DEVELOPMENT OF AN IOT-BASED SOLAR POWERED ROOF WATER SPRINKLER BY USING ARDUINO

ROOBAKKANTHAN A/L JAYABALAN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEVELOPMENT OF AN IOT-BASED SOLAR POWERED ROOF WATER SPRINKLER BY USING ARDUINO

ROOBAKKANTHAN A/L JAYABALAN

This report is submitted in partial fulfilment of the requirements for the degree of Bachelor of Computer Engineering Technology (Computer Systems) with Honours

Faculty of Electronics and Computer Technology and Engineering
Universiti Teknikal Malaysia Melaka

2025

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJURUTERAAN ELEKTRONIK DAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA I

Tajuk Projek : Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino.

Sesi Pengajian : 2024/2025

Saya ROOBAKKANTHAN A/L JAYABALAN mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (✓):

☐	**SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan.
☐	**TIDAK TERHAD**	

Disahkan oleh:

________________________ ____________________________________

Alamat Tetap:

Tarikh : 27 JANUARI 2025 Tarikh : 27 JANUARI 2025

*CATATAN: Jika laporan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this project report entitled "Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino" is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	
Student Name	:	ROOBAKKANTHAN A/L JAYABALAN
Date	:	27 JANUARI 2025

APPROVAL

I hereby declare that I have checked this project report, and, in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Engineering Technology (Computer Systems) with Honours.

Signature : ....

Supervisor Name : TS. DR. HASRUL NISHAM BIN ROSLY

....

Date : 27 JANUARI 2025

....

Signature : ....

Co-Supervisor Name (if any) : ....

Date : ....

DEDICATION

To my beloved mother, DEVIGA D/O R. KRISHNAN, and father, JAYABALAN S/O M. GOVINDASAMY, who have been the source of inspiration and strength when I'm feeling down and giving up, who continually provide their moral, spiritual, emotional, and financial support.

And

To great lecturer and supervisor, TS, DR, HASRUL NISHAM BIN ROSLY who being a true mentor and an inspiration. The patience and dedication given to me was truly comprehensive and unwavering.

And

To dearest families, BURINIEAMMAN, LEENAAMBIGHAI, DHARSHINEE, HARIITHARAN, GAYETHIRI and YAMUNA who shared their words of advice and encouragement to finish this project.

And

To my friends, NAZRUL ISKANDAR, ZAIM HAKIMI, KAGILNES, SAIFUDDIN, SHAZWAN HAZMI, whose has contributed and guide me in completing my work, giving their thought, idea and solution that greatly help me in solving all the problems that occur in my journey to finish this project.

And lastly,

To the Almighty God, thank you for the guidance, strength, power of mind, protection, skills and for giving me a healthy life. It is because of your mighty power; I was able to finish this project successful.

ABSTRACT

Heating solutions are less commonly needed, however, cooling solutions have need to become most popular as the temperature's increases and the climate changes. That is why this project was launched. The research in this proposal will work towards the development of solar-powered IoT, cutler system, a roof water sprinkler that will enhance the comfort levels indoors as well as contribute positively to the environment. The objectives are elaborated in a theoretical framework that seems to accommodate the levels of efficiency of system design and implementation needed for the successful attainment of the goals. Among the objectives, the project target to use the least number of resources in order to gain the highest returns by employing renewable energy resources and IoT. The research is conducted in a orderly fashion with special focus given to the existing technologies as well as the needs of the users. It consists of such elements as point sensors, controllers, and actuators to allow for the alternation of components as well as prototypes with the help of iterative designs. The tests carried out in the conditions of different surrounding temperatures prove that the system contributes to the stabilization of temperature differences and enhances the qualities of living inside the dwelling. Some development establishes the way that the system can reduce reliance on conventional cooling practices and promote sustainable conduct. It is useful information about how to design the cooling solutions for the houses that are environment friendly. The proposed system in this research revolutionises the indoor climate and plans for the protection of the environment using renewable energy and IoT connectivity. The significance of the project lies in the need to reprise and present new tasks in relation to the climate change in residential spaces within the future research and implementation proposals.

ABSTRAK

Penyelesaian untuk kepanasan persekitaran adalah tidak sepenting, tapi penyelesaian pendinginan perlu menjadi paling popular sebagai peningkatan suhu dan perubahan iklim. Itulah sebabnya projek ini dilancarkan. Penyelidikan dalam cadangan ini akan bekerja ke arah pembangunan IoT bertenaga solar, sistem cutler, pemercik air bumbung yang akan meningkatkan tahap keselesaan dalaman serta menyumbang positif kepada alam sekitar. Matlamat-matlamat itu dibina dalam rangka kerja teori yang seolah-olah memenuhi tahap kecekapan reka bentuk dan pelaksanaan sistem yang diperlukan untuk berjaya mencapai matlamat. Antara matlamat, projek bertujuan untuk menggunakan jumlah terkecil sumber untuk mendapatkan hasil tertinggi dengan menggunakan sumber tenaga boleh diperbaharui dan IoT. Penyelidikan dijalankan dengan cara yang teratur dengan tumpuan khas diberikan kepada teknologi yang sedia ada serta keperluan pengguna. Ia terdiri daripada unsur-unsur seperti sensor titik, kawalan, dan penggerak untuk membolehkan pertukaran komponen serta prototaip dengan bantuan reka bentuk. Ujian yang dijalankan dalam keadaan suhu persekitaran yang berbeza membuktikan bahawa sistem ini menyumbang kepada kestabilan perbezaan suhu dan meningkatkan kualiti hidup di dalam kediaman. Sesetengah pembangunan menetapkan cara bahawa sistem boleh mengurangkan ketergantungan kepada amalan penyejukan konvensional dan mempromosikan tingkah laku yang berkelanjutan. Ia adalah maklumat yang berguna tentang bagaimana untuk merancang penyelesaian pendinginan untuk rumah-rumah yang mesra alam sekitar. Sistem yang dicadangkan dalam penyelidikan ini merevolusi iklim dalaman dan rancangan untuk perlindungan alam sekitar melalui penggunaan tenaga boleh diperbaharui dan sambungan IoT.

ACKNOWLEDGEMENTS

Firstly, I would like to thank my supervisor Ts. Dr. Hasrul Nisham Bin Rosly for his helpful suggestions which have been instrumental throughout the development of this project. The former has not only broadened my knowledge in the academic area but also fostered the desire to stay on and work alongside in me.

I could also extend my appreciation to Universiti Teknikal Malaysia Melaka (UTeM) and my parents for the financial assistance through BDP 1 for the successful completion of the project. Nevertheless, the author wishes to express his appreciation to my fellow colleague Zaim Hakimi, Nazrul Iskandar, Kagilnes, Karthik, Saifuddin, for the willingness of his consent disclosure on the project.

On behalf of everyone that never left me alone when I was studying, I would wish to thank my parents and other family members. I also feel inclined to refer to Deviga, my mother for all the encouragement she has given to me.

Last of all, I would like to extend my heartfelt gratitude to all the staffs, fellow colleagues and classmates, the faculty members, and other who is not mentioned here for their co-cooperativeness and assistances.

TABLE OF CONTENTS

LIST OF TABLES

TABLE **TITLE** **PAGE**

LIST OF FIGURES

LIST OF SYMBOLS

℃ - Degree Celsius

LIST OF ABBREVIATIONS

V	-	Voltage
A	-	Ampere
Ah	-	Ampere per Hour
mA	-	milliampere
mAh	-	milliampere per hour
W	-	Watt
Wh	-	Watt per hour
L	-	Liters
R	-	Resistance
I	-	Current
mA	-	milliampere
Pmax	-	Maximum Power Point
Vmpp	-	Maximum Power Point Voltage
VIN	-	Voltage Input
VOUT	-	Voltage Output
MPPT	-	Maximum Power Point Tracking
PWM	-	Pulse Width Modulation

CHAPTER 1

INTRODUCTION

1.1 Background

There being an influence in today's globalized society, there is increase in temperature because of urbanization, and climate change has also made extreme heat to be more uncomfortable and even dangerous to the people in the area. Air conditioning for example which is a traditional method of cooling is usually effective though it relies on non-renewable resources and is often expensive moreover it is hazardous to the natural environment. Particularly for buildings with private residences, there is an increasing requirement for effective, energy-efficient cooling technologies. One of the biggest problems in implementing this solution is the use of clean, efficient energy for cooling, and utilizing renewable energy resources such as solar power is possible in this case.

Regarding the IoT, it is worth noting that this notion influences the usage of devices and systems significantly, focusing on their capability to build efficient cooperation networks. Through IoT technology, innovation changes which have occurred in many fields such as automation, data sharing or real time control. One of the significant places IoT technologies have transformed is in the realm of environmentalism. IoT devices have been utilized in data gathering and analysis, utilizing the opportunity in conservation and resource sustainability. The application of this project to the real world advances the constantly developing IoT solutions to include the adoption of renewable energy and intelligent environmental control.

Another interesting part is the essential need for introduction of the solar-powered solutions due to increasing concerns with sustainable, green technologies. Solar power implies a green energy solution that is readily available and affordable for deployment in regions interested in the development of renewable power. Due to the tapping of solar energy, IoT devices can run without connecting to conventional non-renewable resources, hence developing a negative impression. In the case of using solar power in the project, this tackles a problem of environmental conservation of energy that needs to power the sprinkler system located on the roof. The project will therefore endeavour to use solar energy in the hope of minimizing the use of the conventional energy sources and therefore extend its credit in playing a role in environmental preservation.

1.2 Addressing Global Warming Through Roof-mounted Water Sprinkler Project.

The El Niño climate currently causes us to feel its effects. Warming Pacific Ocean surface waters define the weather phenomenon known as El Niño. El Niño can affect several countries, including Malaysia, and this warming has the potential to have a major impact on global weather patterns. Malaysia may face drought, increased temperatures, and altered rainfall patterns amid an El Niño event. While we cannot control the temperature outside of our homes, we can control the temperature inside. Here is where a roof-mounted water sprinkler functions. It is possible to lower the house's interior temperature by misting water on the roof.

1.3 Problem Statement

In many residential areas, the interior spaces are likely to find their dwellings hot throughout or at least during certain period hence the comfort is compromised in terms of productivity and health. Other cooling methods commonly used especially the air conditioning are climatically unfriendly. Some of the fluorinated gases involved in these systems if not accurately controlled, find their way to the atmosphere causing depletion of the ozone layer and global warming. Furthermore, the need for cooling creates additional stresses on energy supply and distribution and is linked to greenhouse effect.

The homeowners using normal air conditioning systems face is the high cost involved. The fixing costs of air conditioning units are relatively high at the beginning though they may prove flexible in the long run. These costs are usually burdensome enough to compel people to bear with crowding, stuffy housing or use other informal means of cooling that offer no solution to the issue of excessive heat. Also, maintenance and repair charges for air conditioning systems are high and amount to substantial expenses over the years draining more of the financial resources of homeowners.

Air Cooled cooling method is the most energy consuming because it also depends on non-renewable energy sources hence results to high electricity consumption. This leads to growth in greenhouse products which leads to environmental degradation and climate change considering that the world relies on fossil fuels. However, the usage of electricity in higher amounts may overload the configured power supply system and increased load demand may result in lack of power supply or low voltage known as blackouts or brownouts. This significantly high energy consumption means the need to build more power stations; a process that has been seen to cause depletion of natural resources and emission of pollutants.

1.4 Project Objective

The primary objective of the project is to come up with Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino. that can effectively lower temperature indoors in residential homes. Specifically, the objectives are as follows: Specifically, the objectives are as follows:

1. To develop an automated solution that can respond to the environment optimally through the use sensorial system and network accessibility.
2. To implement IoT water sprinkling technique, the system would need to cool the roof slightly thus bring a recognizable change in the internal temperature of the house spaces.
3. To integrate solar panels as a primary source of energy as sustainability goals and need to move away from conventional/natural energy sources. It is an electricity powered system, which runs on solar energy in a bid to lower the impacts on the natural environment.

1.5 Scope of Project

The identified project “Development of an IoT-based Solar Powered Roof Water Sprinkler by Using Arduino” was done in multi-perspective and this project mainly concern on some of the vital areas to support in the establishment of this innovative idea. I will write a more detailed description of each of the above-mentioned critical pieces of information to give better understanding of the given conceptual framework.

a) Building and employing an Internet of Things based model of roof water sprinkler in an operational framework.
b) When the sensors, controllers, and actuators are incorporated in what is commonly called system, then it will become a logical system.
c) Testing and validation are a tool aimed at establishing and confirming the capacity, as well as efficiency of the system in operation in certain conditions.
d) Record and reporting formats readily signify both the design cycle, mounting process, and the result serving both the academics and practicality.

1.6 Societal and Global Impact

Hence, the idea of the "development of an IoT-based solar-powered roof water sprinkler" is indeed a progressive act of addressing the present societally emerging and future global set challenges. Besides the technological competency exhibited by this attempt, this is an indication of ensuring there is recognition and value for SIGs, social, health, safety, legal and cultural effects which are the paramount and crucial responsibilities in such a process. Seen from the conservation perspective, the system is useful for the proposed system could offer the assurance of energy from the natural source, rainwater and low rates of energy consumption, combat the urban heat island effects and foster environmental conservation.

In terms of health and safety issues, several options regarding such possibilities can be considered regarding possible dangers and prevention measures. The issues such as water contaminated at homes, electrical hazards, precaution, and ergonomics are the difficulties handled by the system and save the individuals from further hazards; thus, it demonstrates the efficiency and effectiveness of the system. At the same time, outing the system will involve compliance with the numerous legal regulations, rules and permits at local, national and international level so that all the parties involved in the system will be Morrison (2007) unethical to engage in illicit behaviors.

1.7 Sustainable Development Goal

The flexibility of integrating the factors of sustainable environment into various projects that are commenced by engineers can be regarded as shifting from a new paradigm towards how innovation must be accomplished. Smart infrastructure on the other hand is the application of the United Nations underlined goals that include the sustainable development goals into an engineering project in which the end goal allows consonance with certain goals. For instance, an engineering project that involves tackling problems of better accessibility to this essential product in rural regions achieves the seventh sustainable development goal. This usually means that when incorporating the principles of SDGs in the design of a particular project, the engineers demonstrate concern of high orders towards societal and ecological impacts such as; climate change, eradication of poverty as well as efficient management of resources in the natural environment.

Life cycle assessment can be described as a detailed analysis of the implementation of a given product, process or service, all the environmental consequences of implementing a particular type of materials and right from the time of purchasing the raw materials through to waste disposal at the end of the cycle. In engineering projects, LCAs help in measuring the estimated cost of environmental effects which stem from the distinct phases of a given project application, not only the manufacturing process or intended use but also the closure. It helps the engineer to identify areas where the organization has been using energy, resources, and time and in the process, emitting greenhouse gases and disposing of wastes, besides showing the areas where such effects on the environment could have been minimized and how resources could have been used optimally.

However, as stated earlier, environmental sustainment in engineering does not only consist of averting or reducing harm to significant extents on the environment but also facilitates the positive environmental accountability and community susceptibility strategies. This is in the areas of energy efficiency and use of clean energy systems and technology, water conservation and the recycled face economy systems in the development and execution of the projects.

1.8 Summary

In Chapter 1, the project introduces an IoT-based solar-powered roof water sprinkler system to address rising temperatures in residential areas. It emphasizes the need for sustainable cooling solutions, highlighting the integration of IoT technology and solar power. The project objectives include developing an automated cooling system and reducing reliance on non-renewable resources. The scope involves prototype development, component integration, testing, and documentation for academic and practical purposes.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Several articles and journals have been referred to, for the Development of an IOT-Based Solar Powered Roof Water Sprinkler By Using Arduino. This section provides a description of the work done and presented by authors that we have referred to for this project. We also researched the components used for my project. Understanding [Global/Current Issue] in Literature.

In this section, analyzing theories and works presented in studies on climate change and weather sensing technologies, this section will discuss how weather sensing projects can be used for global warming enhancement understanding and counteraction of the consequences. A reference on the studies of the use of weather sensing technologies have revealed that detailed data on temperature fluctuations, humidity and precipitation are also signs that give information of the effects of global warming. For instance, to support the conclusion that temperatures are on the rise and that there are rises incidences of heat waves, droughts as well as severe rainfall, researchers tap into the data on weather sensing. Further, relying on weather sensing together with other environmental and socio-economic datasets helps researchers and policymakers to explore the dynamics of climate change, human impact, and ecosystem conditions, and monitor changes. Therefore, weather sensing projects can be seen as more significant endeavors that have a huge potential for contributing to the wrestling with the climate change problem on the global.

2.2 Concept/Theory

2.2.1 Internet of Things (IoT)

With the continuous advancements in technology, a potential innovation of new things becomes unavoidable. A part of the technology advancement is illustrated in Figure 2.1, which summarized how things are sensed and connected start RFID, WSN and IoT [1]. In recent years, significant advancement has been achieved in the industrial environment

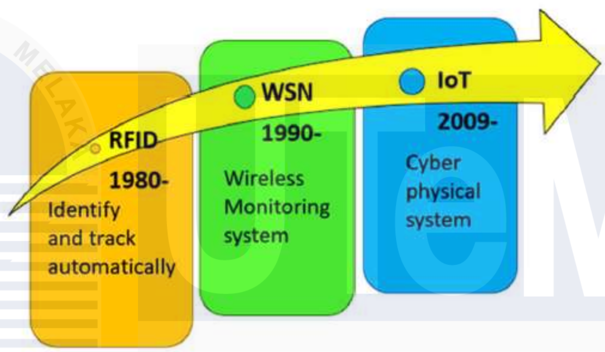

Figure 2.1 Technology Advancement.

domain by the introduction of 4.0. This new revolution was first initiated in 2011 at Hanover Fair, German, focusing on merging production with Information Technology and the Internet as a high-technology strategy for 2020 [2]. The Internet of Things (IOT) provides connectivity for anyone at any time and place to anything at any time and place. With the advancement in technology, we are moving towards a society, where everything and everyone will be connected. The IOT is considered as the future evaluation of the Internet that realizes machine-to-machine (M2M) learning. The basic idea of IOT is to allow autonomous and secure connection and exchange of data between real world devices and applications. The IOT links real life and physical activities with the virtual world [3].

The internet of things is expected to impact various areas of life including smart homes, smart campus, and more. Its usage for scenarios in the smart city paradigm is apparent where many applications are becoming under fast development, notably in the e-health market [4]. Internet of Things (IOT) Over the last few years in the world of wireless telecommunications there is a new technology called Internet of Things (IOT), which has received much attention in academia and industry.

The method uses by the Internet of Things is a wireless or automatic control without knowing about distance. IoT is opening tremendous opportunities for a large number of novel applications that promise to improve the quality of our lives [5]. The implementation of the Internet of Things are usually follow the purpose of the developer in accordance with the application that is created, If a tool created to help monitoring a room then the implementation of the Internet of Things must follow the flow of programming diagram about the sensor in a house, how far is the room can be controlled, and the speed of the internet network used. The development of networking technology and the Internet such as IPv6, 4G, and WiMAX can optimize the implementation of the Internet of Things, and allows the distance that can be passed furthermore, so it is easier for users to control things.

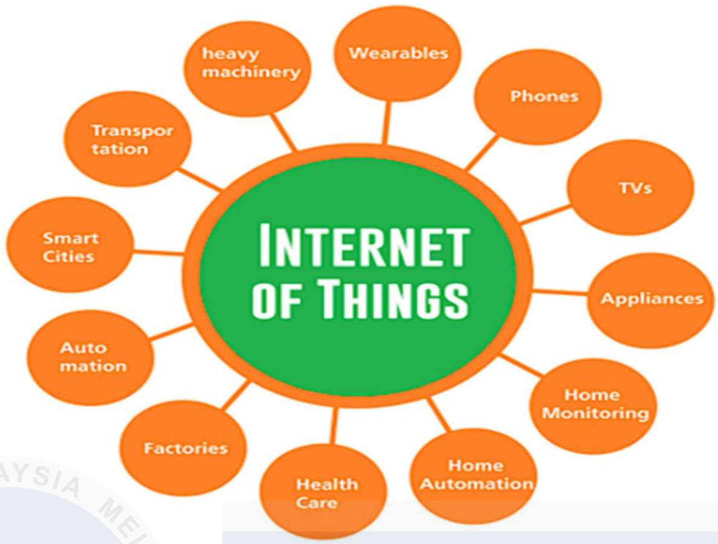

Figure 2.2: The Sectors of IOT Uses.

Based on the Figure 2.2, the 12 sectors of IoT uses is shown. It shall not be forgotten, anyway, that the words ''Internet" and ''Things", when put together, assume a meaning which introduces a disruptive level of innovation into today IOT world. In fact, ''Internet of Things" semantically 146 means' 'a worldwide network of interconnected objects uniquely addressable, based on standard communication protocols" [6]. Internet of Things technologies assessed will be very helpful for people in the real world using the sophisticated technology of Internet. IOT aims to connect the real objects to the Internet, for example, production machinery, automobiles, electronic devices, wearable equipment, and includes any real objects that connected to local and global networks using sensors which is embedded within the object. It can be controlled and provided information to the user. In general, the working system of the Internet of Things is almost the same and easy enough, first every object must have an IP Address. IP Address is an identity in the network that makes these objects can be ordered from other objects in the same network.

2.3 Items of the system

2.3.1 Arduino

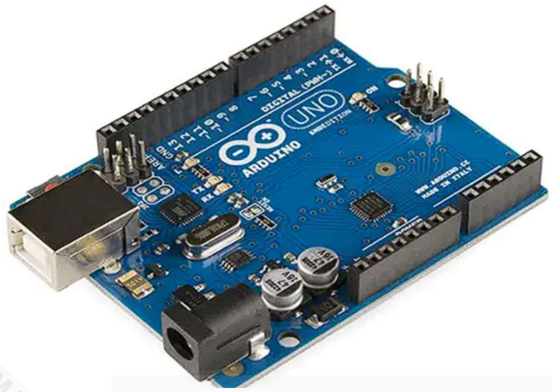

Figure 2.3: Arduino Board.

Arduino is an open-source microcontroller as shown in Figure 2.3 which can be easily programmed, erased, and reprogrammed at any instant of time. Introduced in 2005 the Arduino platform was designed to provide an inexpensive and easy way for hobbyists, students, and professionals to create devices that interact with their environment using sensors and actuators. Based on simple microcontroller boards, it is an opensource computing platform that is used for constructing and programming electronic devices. It is also capable of acting as a minicomputer just like other microcontrollers by taking inputs and controlling the outputs for a variety of electronics devices [7].

A microcontroller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. The important part for us is that a microcontroller contains the processor (which all computers have) and memory, and some input/output pins that you can control. (Often called GPIO - General Purpose Input Output Pins).

It is also capable of receiving and sending information over the internet with the help of various Arduino shields, which are discussed in this paper. Arduino uses hardware known as the Arduino development board and software for developing the code known as the Arduino IDE (Integrated Development Environment).

Built up with the 8-bit Atmel AVR microcontrollers that are manufactured by Atmel or a 32-bit Atmel ARM, these microcontrollers can be programmed easily using the C or C++ language in the Arduino IDE. Unlike the other microcontroller boards in India, the Arduino boards entered the electronic market only a couple of years ago and were restricted to small scale projects only. People associated with electronics are now gradually coming up and accepting the role of Arduino for their own projects. This development board can also be used to burn (upload) a new code to the board by simply using a USB cable to upload. The Arduino IDE provides a simplified integrated platform which can run on regular personal computers and allows users to write programs for Arduino using C or C++.

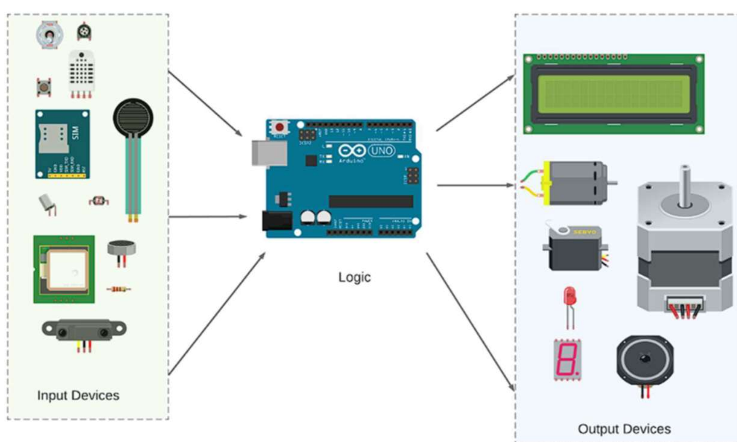

Figure 2.4: Arduino Board Specification.

In Figure 2.4 shown board specification of Arduino. The Arduino is a development board for the ATMEGA328 microcontroller. The Arduino has all the electronic components needed to support the ATMEGA328 on one PCB. There are other useful features like input/output pins, a USB port for communication between the Arduino and a computer, and a 9V DC power connector as shown in the Figure 2.5 Arduino system and I/O Pin. There are many different types of Arduino boards, but the most popular is the Arduino Uno [8].

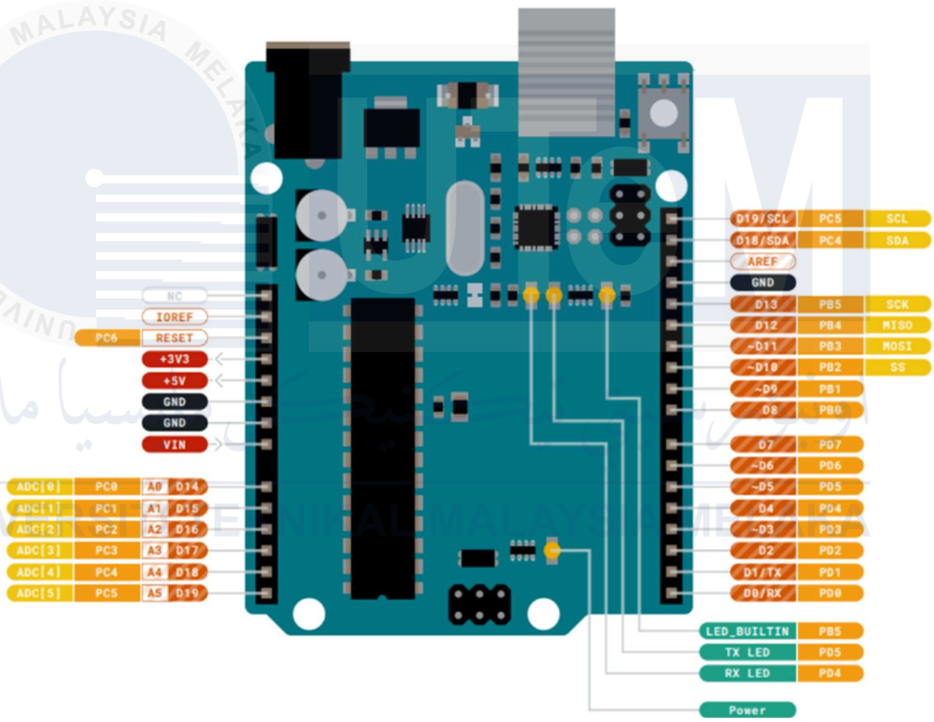

Figure 2.5: Arduino System and I/O Pin.

Table 2.1: Types and Specification of Arduino.

Arduino model	**Microcontroller**	**Features**	**Operating voltage**	**Common applications**
Uno	ATmega328P	Clock speed: 16 MHz Flash memory: 32KB RAM: 2KB GPIO pins: 20 Analog inputs: 6 DAC: No PWM: 6 pins UART, SPI, and I^2C communication Connector: USB	5V	Entry-level hobby projects: RFID access control, Motor control: Temperature monitoring, GSM based projects,
Pro Mini	ATmega328P	Clock speed: 16 MHz Flash memory: 32KB RAM: 2 KB GPIO pins: 22 Analog inputs: 8 DAC: No PWM: 6 pins UART, SPI, and I^2C communication Connector: 6-pin UART No USB-to-Serial Not Shield Compatible	5V	Entry-level hobby projects
Mega 2560 R3	ATmega2560	Clock speed: 16 MHz Flash memory: 256 KB RAM: 8 KB GPIO pins: 54 Analog inputs: 16 DAC: No PWM: 15 pins UART, SPI, and I^2C communication Connector: USB Shield Compatible	5V	Motor control, sensor interfacing, home automation, security systems, embedded systems, IoT applications

Arduino model	Microcontroller	Features	Operating voltage	Common applications
Pro Micro	ATmega32U4	Clock speed: 16 MHz Flash memory: 32 KB RAM: 2.5 KB GPIO pins: 18 Analog inputs: 9 DAC: No PWM: 5 pins UART, SPI, and I^2C communication Connector: USB Not Shield Compatible	5V	Entry-level hobby projects
Leonardo	ATmega32U4	Clock speed: 16 MHz Flash memory: 32 KB RAM: 2.5 KB GPIO pins: 20 Analog inputs: 12 DAC: No PWM: 7 pins UART, SPI, and I^2C communication Connector: USB Shield Compatible	5V	Entry-level hobby projects
MKR WIFI 1010	SAMD21 Cortex M0+	Clock speed: 48 MHz Flash memory: 256KB RAM: 32KB SRAM GPIO pins: 22 Analog inputs: 7 DAC: Yes PWM: 12 pins UART, SPI, and I^2C communication Connector: USB WIFI	3.3V	Projects requiring WIFI connectivity, IoT applications
MKR1000	ATSAMW25H18	Clock speed: 48 MHz Flash memory: 256KB RAM: 32KB GPIO pins: 22 Analog inputs: 7 DAC: Yes PWM: 12 pins UART, SPI, and I^2C communication Connector: USB WIFI	3.3 V	Projects requiring WIFI connectivity, IoT applications.

Arduino model	Microcontroller	Features	Operating voltage	Common applications
Due	AT91SAM3X8E	Clock speed: 84 MHz Flash memory: 512 KB RAM: 96 KB GPIO pins: 54 Analog inputs: 12 DAC: Yes PWM: 12 pins UART, SPI, and I^2C communication Connector: USB Shield Compatible	3.3 V	Industrial automation, security systems, virtual reality, GSM, embedded systems, industrial IoT, wireless sensor networks, drones and robots
MKR Vidor 4000	FPGA: Intel Cyclone 10CL016 SAMD21: Microchip ATSAMD21 (Arm Cortex-M0+ processor)	Clock speed: 200MHz Flash memory: 2MB RAM: 8MB SDRAM GPIO pins: 22 Analog inputs: 7 DAC: Yes PWM: 12 pins UART, SPI, and I^2C communication Connector: USB WIFI Bluetooth	3.3 V	Projects requiring more processing power, image processing, audio and video processing, image fusion, digital signal processing

Table 2. 1 depicted regarding the types and specifications of some of the models of Arduinos such as the Uno, Pro Mini, Mega 2560 R3, Pro Micro, Leonardo, MKR WIFI 1010, MKR1000, Due, and MKR Vidor 4000. Other Arduino models may feature certain benefits, such as higher processing capabilities (as found in the MKR Vidor 4000), more I/O pins (as in the Mega 2560), or some unique features (such as integrated Wi-Fi on the MKR lineup), but for beginners and intermediate users alike, the Arduino Uno is still popular. Furthermore, it could also serve as an added and more solid ground for learning, experimenting, and even creating numerous kinds of electronic projects.

2.3.2 Espressif32 Module

The ESP32 microcontroller as shown in Figure 2.6, developed by Espressif Systems, has garnered significant attention in the field of embedded systems and IoT (Internet of Things) applications due to its versatile features and capabilities. This section reviews the existing literature surrounding the ESP32, highlighting its key attributes, applications, and advancements. The ESP32 is a powerful microcontroller that integrates Wi-Fi and Bluetooth connectivity, making it well-suited for a wide range of IoT applications. It features a dual-core processor, low-power modes, and a rich set of peripheral interfaces, enabling seamless integration with various sensors, actuators, and communication protocols

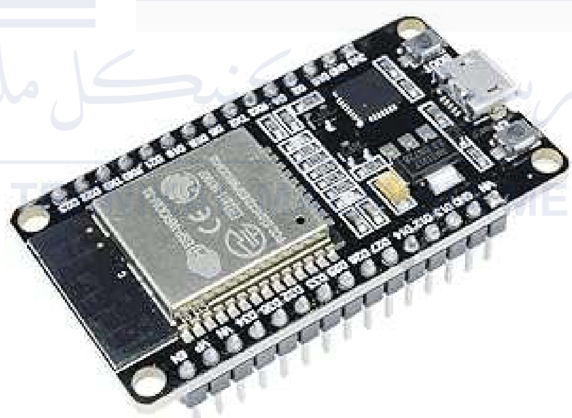

Figure 2.6: ESP32 module.

2.3.2.1 Application of ESP32

Numerous studies have explored the diverse applications of the ESP32 in different domains. From home automation and smart agriculture to industrial monitoring and wearable devices, the ESP32 has been utilized in various projects to enable remote monitoring, data logging, and control functionalities. As shown in Figure 2.7, the process flow of how ESP32 is working [9].

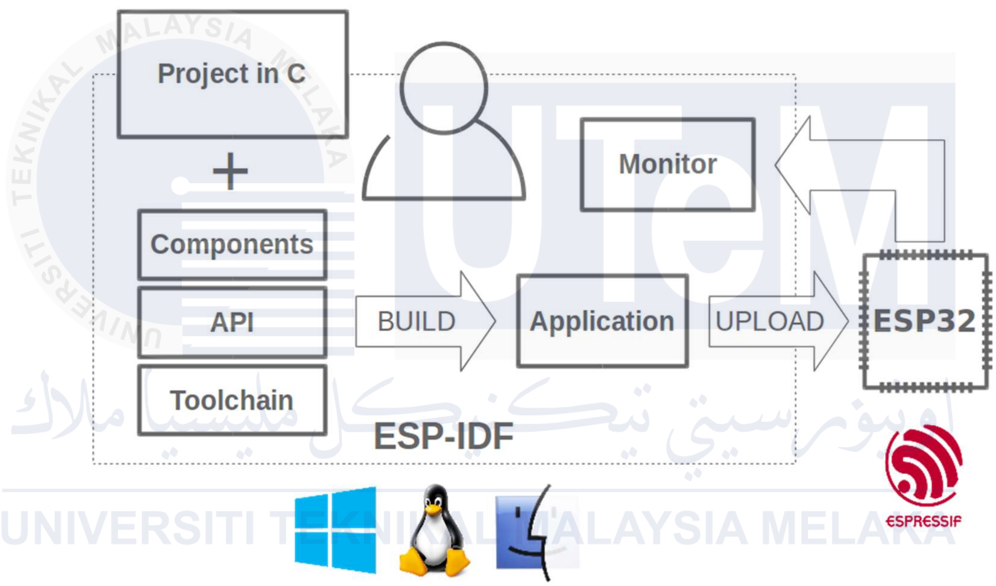

Figure 2.7: Espressif IoT Integrated Development Framework.

Recent developments in ESP32 development have focused on enhancing performance, reliability, and scalability. Through firmware upgrades and software libraries, research efforts have been geared at optimizing power consumption, improving security features, and extending the ESP32's capabilities. The ESP32 has been used by universities and research institutes throughout the world for developing IoT systems, performing experiments, and teaching embedded systems and wireless communications principles [10].

Despite its numerous advantages, the ESP32 also poses certain challenges and limitations. Issues related to power management, software stability, and real-time performance have been reported in some studies, highlighting areas for further improvement and optimization.

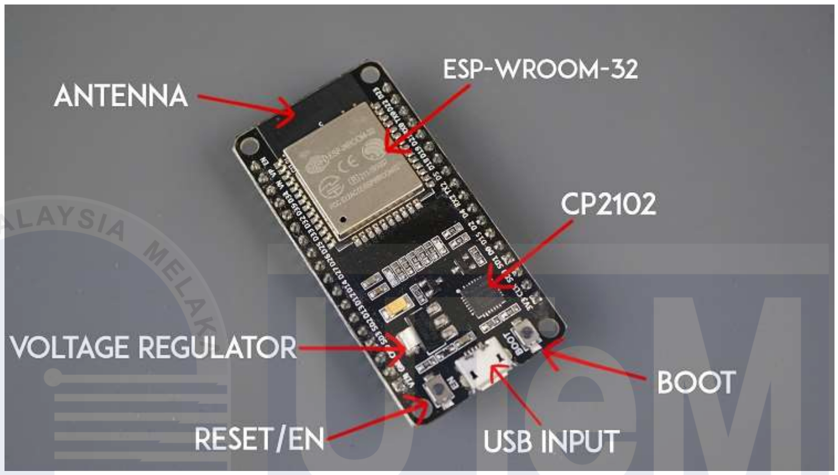

Figure 2.8: ESP 32 Board.

In Figure 2.8, features of ESP 32 board are shown. It has a well-endowed list of features that is befitting the need of a new age IoT or any basic embedment project. Right at the heart of the Devkit is the ESP-WROOM-32 module which is famous for its performance attributes such as two core processors, Wi-Fi and BT presence. The board also comes with an integrated antenna providing strong wireless interfacing of the board. A voltage regulator, essential for the stable management of power supply and important to the system in diverse working conditions. The CY7C68013A USB component allows clean connection with host computer for easy programming and debugging are possible through the CP2102 USB-to-serial converter. Also, the speciation of the program includes a reset button and boot configuration settings that promote usability and development.

2.3.3 DHT11 Sensor

The DHT11, shown in Figure 2.9, is a digital temperature and humidity sensor extensively utilized in a variety of electronic devices and systems for environmental monitoring purposes. This section reviews the literature on the DHT11 sensor, including its operating principles, uses, and advances.

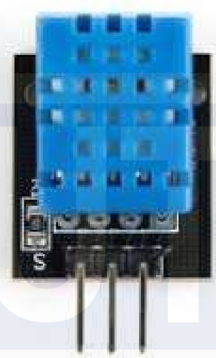

Figure 2.9: DHT11 Sensor.

The DHT11 sensor provides calibrated digital output, making it simple to integrate with microcontrollers and other control systems without requiring external calibration or trimming. It is designed for applications requiring basic temperature and humidity measurements, with an accuracy of ±2°C for temperature and ±5% for humidity. The DHT11 operates within a temperature range of 0°C to 50°C and a humidity range of 20% to 90% RH (Relative Humidity). The DHT11 operates using a single-wire serial interface for communication, simplifying the connection process. It consumes low power (max 2.5mA during measurement) and has a sampling rate of once every second. The sensor has a typical resolution of 1°C for temperature and 1% for humidity, making it suitable for basic monitoring tasks. Its low cost and straightforward interface make it a popular choice for hobbyist and professional projects. [11].

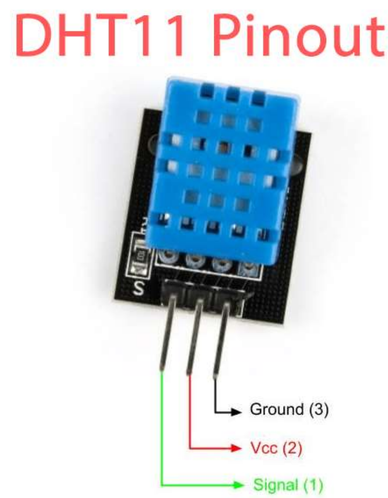

Figure 2.10 DHT11 Pinout.

Figure 2.10 shows the DHT11 pinout. Utilization is common in various electronic circuits and systems within the environmental sensing domain. The DHT11 sensor has four pins, each serving a specific purpose [12]:

I. **VCC (Voltage Supply):** This pin is used to power the DHT11 sensor. It operates on a supply voltage of 3.3V to 5.5V DC, making it compatible with most microcontroller systems, including Arduino.

II. **DATA (Digital Output):** The DATA pin provides a serial digital output representing temperature and humidity readings. Communication occurs through a single-wire protocol, simplifying interfacing and reducing pin requirements.

III. **GND (Ground):** It is the ground or the zero-voltage reference plane of the circuit or a signal reference plane. Control and measure the electrical current to DHT11 sensor in order to maintain the temperature.

2.3.4 Solar Panel with PWM controller

Integration of photovoltaic solar power panels with PWM controllers is one of the widely adopted methods in renewable energy technologies for maintaining cost-effectiveness and simplicity. Solar panels work by converting sunlight into electrical energy, which may be used in various ways. A PWM (Pulse Width Modulation) controller, though less efficient than an MPPT controller, serves as a cost-effective and reliable option for smaller systems. The PWM controller regulates the flow of energy from the solar panel to the battery by adjusting the width of the electrical pulses. This ensures that the battery is charged safely and efficiently without overcharging or overheating. This section briefly reviews the current available literature on solar panels and PWM controllers, highlighting their importance, relevance, and impact on energy reliability and affordability [13].

2.3.4.1 Solar Panel

Solar panels, also known simply as photovoltaic (PV) modules and shown in Figure 2.11 about the solar panel, are primary components in the typical solar system that are designed to convert sunlight into electrical power. This operates on the principle whereby, with the help of semiconductor materials, like silicon, photovoltaic cells are designed to capture the rays of the sun to generate an electric current as soon as photons penetrate the Solar cells. This review goes deeper in discussing the structure and functioning of solar panels, especially for the residential, commercial, and industrial applications of the product.

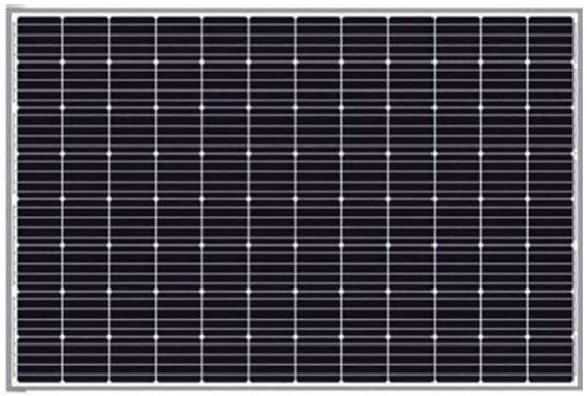

Figure 2.11: Solar Panel.

When the sun rays fall on them, it forms an electric field through which it reliably initiates the flow of electrons to produce a direct current or DC electricity. This can be further converted into the alternating current (AC) through usage of inverters, for example to be fed to the grid or can be stored in batteries for use in standalone systems. Solar products are usually applied in residential, commercial as well as in the industrial area. In off-grid applications, solar panels employments include telecommunication, water pumping, and lighting systems [14].

2.3.4.2 PWM Solar Controllers

In figure 2.12, Pulse width modulation (PWM) main purpose is to switch the solar system controller power devices by applying a constant voltage battery charging. Modern charge controllers used PWM to allow lower amount of power applied to the batteries when the batteries are almost fully charged. PWM allows the battery to be fully charged with less stress on the battery prolonging the battery life. PWM controller works on the concept that when solar cell produces voltage, this voltage is then indicated by voltage indicator. [15].

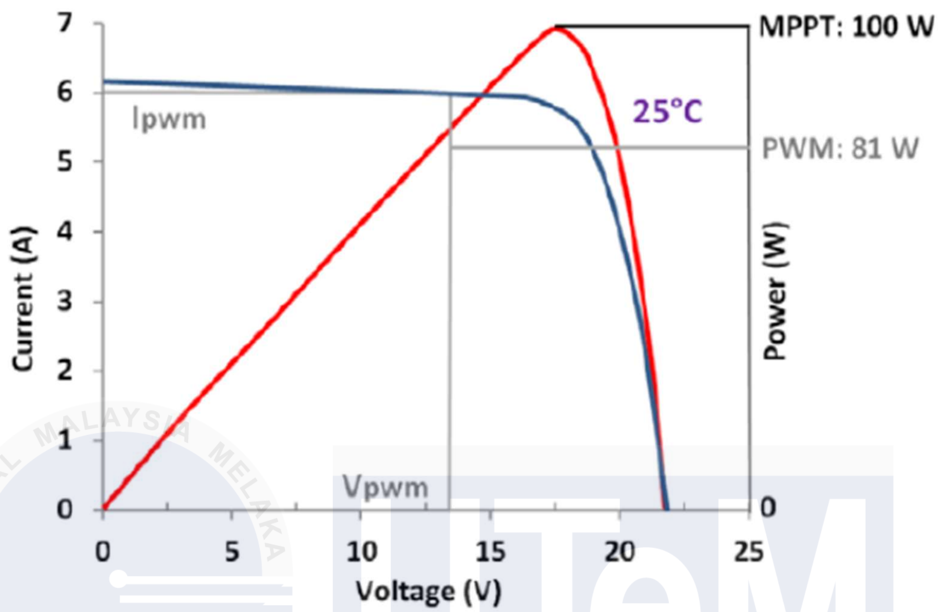

Figure 2.12: Current Vs Voltage curve for PWM.

This figure 2.12 illustrates the relationship between current (I), voltage (V), and power for a solar panel under the control of a Pulse Width Modulation (PWM) controller. Here's a detailed explanation:

1. Axes of the Graph:
    - **X-axis**: Voltage (V) in volts.
    - **Left Y-axis**: Current (I) in amperes.
    - **Right Y-axis**: Power (P) in watts.
2. Graph Components:
    - **Current-Voltage (I-V) Curve** (blue and red lines): Shows how the current varies with voltage at a specific temperature (25°C).
    - **Power-Voltage (P-V) Curve**: Derived from the I-V curve, represents how the power output changes with voltage.

3. Key Points and Regions:

    - **Maximum Power Point (MPP)**: The point where the product of current and voltage is maximized, indicated as "MPPT: 100 W" in the graph.
    - **PWM Operation**: Under PWM control, the solar panel voltage aligns closely with the battery voltage, leading to suboptimal power extraction. The power under PWM is labeled as "PWM: 81 W."
    - **Vpwm and Ipwm**: These indicate the operating voltage and current under PWM control.

4. PWM Controller Characteristics:

    - A PWM controller acts as a simple switch connecting the solar panel to the battery.
    - The panel operates at the battery's voltage, which limits the ability to extract maximum power. For instance, if the battery voltage is below the MPP voltage, the power output is lower than the maximum achievable.

5. Performance Comparison:

    - The graph shows that while the MPP power output is 100 W, the PWM controller only extracts 81 W, leading to a loss of efficiency.
    - This highlights the limitation of PWM controllers compared to Maximum Power Point Tracking (MPPT) controllers, which adjust the operating voltage to maximize power extraction.

PWM charge controllers use technologies similar to other modern high quality battery chargers. Some of the unique benefits of PWM pulsing are:

- Ability to recognize lost battery capacity and to desulfate a battery.
- Dramatically increase the charge acceptance of the battery.
- Maximum high average battery capacities.
- Equalize drifting battery cells.
- Reduce battery heating and gassing.
- Automatically adjust for battery ageing.
- Self-regulate for voltage drops and temperature effects in solar systems.

2.3.5 Water Pump Motor

Figure 2.13: DC 12V Pneumatic Diaphragm Water Pump Motor.

Understanding the crucial role these electromechanical devices play in various sectors is made easier by reading the introduction to the literature review on water pump motors. It starts by putting the importance of water pump motors in the context of enabling the movement of water for home, commercial, and agricultural uses. In Figure 2.13 shows how, it looks like 12V water pump motor.

The basic ideas behind water pump motor operation are described in this section, with special attention to the function of these motors in turning electrical energy into mechanical energy for water pump drive. The introduction outlines the wide range of uses for water pump motors, from industrial processes and water supply systems to agricultural irrigation and livestock watering. The introduction highlights the various applications of water pump motors and their widespread influence on improving efficiency and productivity. DC 12V pneumatic diaphragm water pump motors are essential components in various applications, offering efficient and reliable water pumping solutions [16].

DC 12V pneumatic diaphragm water pump motors utilize compressed air to drive the movement of a flexible diaphragm, resulting in the displacement of water. This displacement creates a pumping action, allowing water to be moved from one location to another. These motors are designed for low-voltage DC operation, making them suitable for portable and off-grid water pumping applications [17].

2.3.5.1 Applications of DC 12V Pneumatic Diaphragm Water Pump Motors

These motors are used in various capacity within agriculture, automobiles, marine, and recreational purposes. They are employed in applications like water distribution and pumping, liquid movement, and other applications like feeding chemicals into solutions. DC 12V pneumatic diaphragm water pump motors are of much demand due to their features such as light size, energy utilization and capability of working in areas that are out of reach of electricity power. Advancements in technology in the recent past seek to refine several parameters like performance, durability and reliability of these motors [18].

2.3.6 Blynk Application

In the literature review section where the few opportunities for the use of Blynk applications are discussed, the reader is first exposed to how this platform revolutionized the Internet of Things (IoT). It begins with setting the scene for Blynk apps, and the significance of remote control and monitoring of IoT devices in the extant literature, while also establishing how the Blynk apps are user-friendly with the focus on the specifications of the domains of application. Figure 2.14 show the logo of the Blynk application.

Figure 2.14: Blynk Application.

In the first section, some basic principles that govern Blynk applications are described; it specifies that Blynk apps operate as clients that can communicate with Blynk servers and connected industrial gadgets. This section focuses on the drag-and-drop capabilities of the platform and what it means for customers to tailor the often-generic IoT device management and data presentation to their specific needs through custom-designed dashboards and widgets [19].

As for the purpose of Blynk apps, the main idea is to provide users with an easy-to-navigate interface of Internet of Things devices' remote control and data monitoring. The integrated drag-and-drop UI enables users to design personalized thematic interfaces and widgets for computers, tablets, and smartphones to control the connected devices. Based upon the client server model adopted by Blynk, the Blynk app acts as the client whenever it has to establish a connection with the Blynk server or the connected IoT devices.

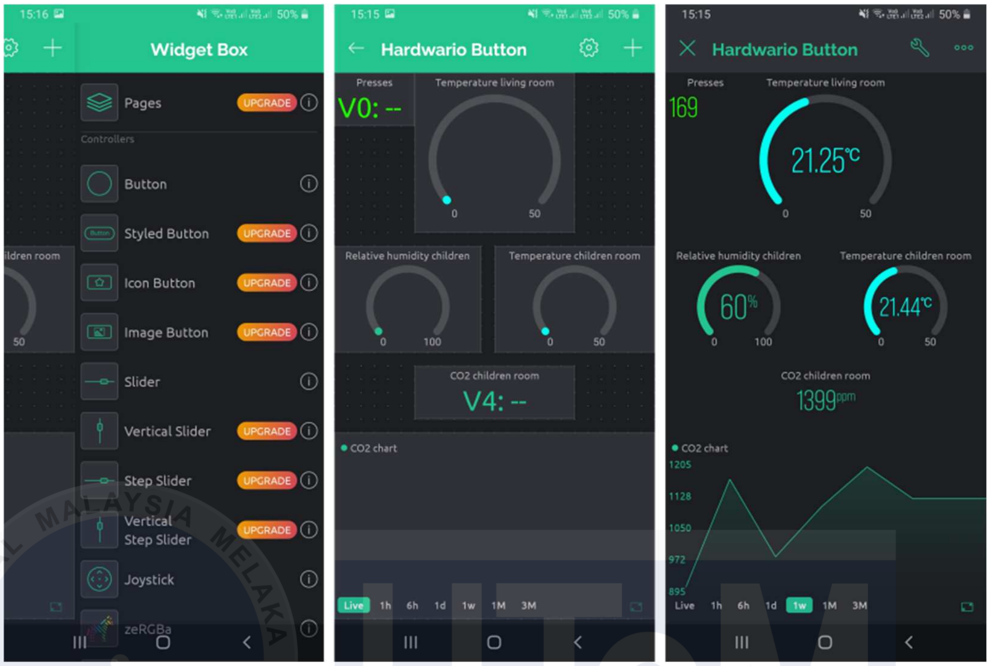

Figure 2.15: Example of Blynk Interface.

In Figure 2.15 shown an example of Blynk application how the user interface is. Managing and observing the different IOT devices from a distance is made simple using Blynk apps. Besides the person's health data sharing and control, users can organize their car statistics, home appliances, and control the environment with the help of personalized users' panels [20].

New development & innovations are specifically aimed at improving the ability of the Blynk apps which refers to timely expansion of its functions, improving the security of its work, and increasing the convenience of its interface. Furthermore, within the course of the diverse platforms of the IoT's evolution, other IoT hardware platforms and communication protocols can also be included to ensure compatibility and interconnectivity of different device hierarchies [21].

2.4 Comparison of My Project and Previous Projects

Table 2.2: Comparison of My Project and Previous Projects.

Features / **Specification**	**IoT Garden Sprinkler System (***Mokh Sholihul Hadi, Pradipta Adi Nugraha***)**	**Solar-Powered Irrigation System (***EFY Bureau***)**	**Automated Lawn Sprinkler System (***Nazirul Mubin Zahari***)**	**My Project: Development of an IoT-based Solar Powered Roof Water Sprinkler**
Primary Objective	Garden irrigation	Agricultural irrigation	Lawn watering	Cooling residential roofs.
Power Source	Solar panels	Solar panels	Solar panels	Solar panels with grid power backup
Control System	Raspberry Pi	Arduino Uno	Raspberry Pi	Arduino Uno, ESP32
Sensors Used	Soil moisture sensor, temperature sensor	Soil moisture sensor, water flow sensor	Soil moisture sensor, temperature sensor	Temperature sensors, water level sensor.
Communication Protocol	Wi-Fi	GSM module	Wi-Fi	Wi-Fi (ESP32),
Water Distribution Method	Drip irrigation	Drip irrigation	Sprinklers	Sprinklers
Automation Level	Semi-automated	Fully automated	Fully automated	Fully automated
Energy Efficiency	High (solar-powered, IoT-controlled)	High (solar-powered, IoT-controlled)	High (solar-powered, IoT-controlled)	High (solar-powered, IoT-controlled, PWM)
Cost Efficiency	High (commercially available)	High (commercially available)	High (commercially available)	Optimized for cost reduction

Table 2.2 shows the comparison of the previous project which is slightly like my current project. With this boost added to it, “Development of an IoT-based solar-powered roof water sprinkler” can offer a better solution with more flexibility, dependability and ease of usage. It is very probable to be rejected in a competitive review so relying on arguments like the second level of automation and energy efficiency, simplicity enough for users and professionals, may be helpful.

2.5 Summary

In the literature review section, chapter 2 critiques the current social and global issues of conventional residential cooling systems and particularly air conditioning discussing the environmental and the financial issues of them. Reviewing the literature of scholarly work, this chapter emphasizes the urgency of hedonistic and economical solutions to address the impacts of increased temperatures on home environments. Further, the literature review discusses various sustainable cooling technologies or measures such as passive cooling, integration of renewable energy sources, and water management to present the possible low-energy solutions or climate change responsive approaches for cooling technologies for energy-efficient and environment friendly building designs, social equity, etc. In summary, Chapter 2 presents a clear state of the art and identification of the major research questions that the current literature has left unanswered” before proceeding with the formulation of new cooling solutions for residential neighbourhoods with subsequent assessment.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The specific objective of this venture is to develop an Internet of Things – IoT based solar operated roof water sprinkler using Arduino Uno. This new technology aims at mitigating heat in living zones by designing sprinkles to release water on the roof particularly in event of excess heat. analysis on the basis project plan shows that this system has facilities like, temperature monitoring, water level control and the use of solar system.

The project titled "Development of an IoT based Solar powered Roof water sprinkler by using Arduino" is much relevant to the present requirements of addressing the heat issue and finding the sustainable and cheap means for cooling. Warm many houses make, reducing ease, and output with exposing health risks to the inhabitants. Some of the earlier cooling technologies like the air conditioning methods, tend to cost a lot of money and are harmful to the ecology in so many ways.

The article "Development of an IoT-based Solar Powered Roof Water Sprinkler using Arduino" will help to enhance the process of homes cooling in this paper. It employs an LM35 temperature sensor to measure high temperatures more than 32 C and in this process use the water pump motor to shower the roof with water, to cool the environment. It also has built-in water level to for monitoring and regulating the volume of water in appropriate sources. This technology is useful since it employs solar panels, thus it operates within the realm of green energy.

3.2 Equipment Used in This Project

In this project, a basic Arduino board is used as the microcontroller while a breadboard circuitry is used for designing circuit connection and jumper wires are used to interconnect the devices. Reliability of the power supply. To ensure the solar power source works as required and keep functioning continuously. We use various sensors and actuators to control the environment and in programming the Arduino and as a source of power during development the USB connection is used. To write and upload code, we need a computer that should have Arduino IDE installed on it. For this we add an ESP32 module so that the project can be configured to connect to the Wi-Fi and Bluetooth.

3.2.1 Arduino Uno R3 ATmega328P

Arduino UNO is a microcontroller board built on the ATmega328P. In Figure 3.1 show Arduino R3 Uno how it is in real life. It contains 14 digital input/output pins (6 of which may be used as PWM outputs), 6 analogue inputs, a 16 MHz ceramic resonator, a USB port, a power connector, an ICSP header, and a reset button. It includes everything necessary to support the microcontroller. You can tamper with your UNO without worrying about making a mistake; worst case scenario, you can replace the chip for a few bucks and start again [22].

Figure 3.1: Arduino R3 Uno.

Table 3.1: Technical specifications for the Arduino UNO R3.

Name	Arduino UNO R3
SKU	A000066
Microcontroller	ATmega328P
USB connector	USB-B
Built-in LED Pin	13
Digital I/O Pins	14
Analog input pins	6
PWM pins	6
UART	Yes
I2C	Yes
SPI	Yes
I/O Voltage	5V
Input voltage (nominal)	7-12V
DC Current per I/O Pin	20 mA
Power Supply Connector	Barrel Plug
Main Processor	ATmega328P 16 MHz
USB-Serial Processor	ATmega16U2 16 MHz
Memory ATmega328P	2KB SRAM, 32KB FLASH, 1KB EEPROM
Weight	25 g
Width	53.4 mm
Length	68.6 mm

Table 3.1 explains the technical specification of Arduino R3 Uno, which is a versatile microcontroller board based on the ATmega328P processor, offering a robust set of features suitable for a wide range of electronic projects. It includes a USB-B connector for easy interfacing with computers, facilitating programming and communication tasks. The board is equipped with a built-in LED on pin 13 for basic status indication.

3.2.2 ESP32 Module

An ESP32 Wi-Fi module is a SOC microchip that is mostly used for constructing endpoint IoT (Internet of Things) applications. It is a cost-effective solo wireless transceiver. It offers internet connectivity for a variety of embedded system applications. In Figure 3.2 shows about the pinouts of ESP32.

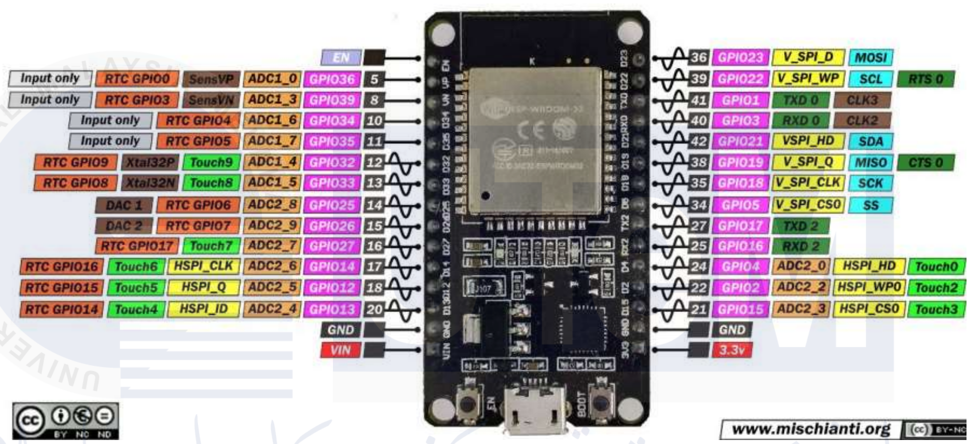

Figure 3.2: ESP32 Pinouts.

3.2.2.1 Features of ESP32

- Processor: Xtensa dual-core or single-core 32-bit LX6 microprocessor
- Clock frequency: 160 or 240 MHz
- Memory: 520 KiB RAM, 448 KiB ROM
- Wireless connectivity: Wi-Fi: 802.11 b/g/n, Bluetooth: v4.2 BR/EDR and BLE
- Security: IEEE 802.11 standard security features, secure boot, flash encryption, 1024-bit OTP
- Operating voltage: 3.3V, but can be operated via 5V-microUSB
- Operating temperature -40°C - 125°C

3.2.3 Water Level Sensor

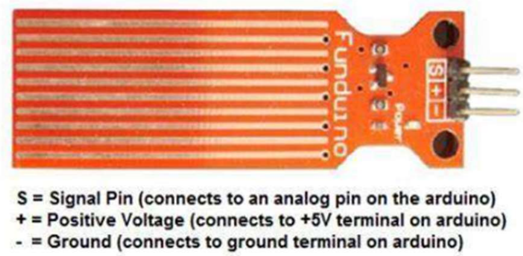

Figure 3.3: Water Level Sensor with Pin Specification.

Figure 3.3 shows the pinout of water level sensor. The sensor consists of three pins, each serving a specific function as described below: The sensor consists of three pins, each serving a specific function as described below:

1) S (Signal):
   - Function: It gives a variable voltage output equivalent to the detected water level by the water level sensor.
   - Connection: This pin should relate to the any of the analogue input pins of the Arduino board so that the microcontroller is able to read what the sensor is outputting.
2) + (VCC):
   - Function: The voltage coming from the source is regulated to provide the right voltage that the sensor needs to do its job.
   - Connection: Power up through the 3V or 5V power pin on an Arduino board.
3) - (GND):
   - Function: It establishes a common reference for the sensor that identifies the power supply and signal for sensors.
   - Connection: This pin must be connected to the ground (GND) pin on the Arduino or the common ground of the power supply.

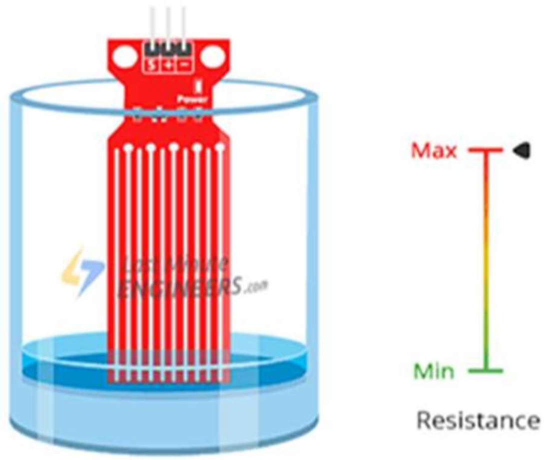

Figure 3.4 Water level sensor detect water level in water tank.

In Figure 3.4, show the application of water level sensor in a container. This water level sensor operates under the provisory measurement of the amount of water by conductivity whereby the amount of conductivity rises as the amount of water increases. The output voltage which is obtained over the LDR is then inputted into the Arduino and with the use of a formula the amount of water in the tank is determined. Well, they work like this, let me give accurate account on how this sensor would:

- Output Voltage: As this sensor is designed to have this capability, it also gives an output voltage in proportion to the level of the water.
- High Water Level: In this case, sensitivity rises with increase in conductivity and declines with increase in resistance: The output voltage also goes up with increase in conductivity although the sensor is dipped more in water.
- Low Water Level: This is because as the sensor is less wet, conductivity is created and hence increases the electrical resistance so that the output voltage becomes altered.

3.2.4 DHT11 Temperature & Humidity Sensor Module

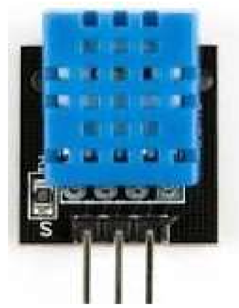

Figure 3.5: DHT11 Temperature & Humidity Sensor Module.

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. 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. [11].

Features:

- Operating Voltage: 3.5V to 5.5V
- Operating current: 0.3mA (measuring) 60uA (standby)
- Size: 28mm x 13mm x 7mm
- Output: Serial data
- Temperature Range: 0°C to 50°C
- Humidity Range: 20% to 90%
- Resolution: Temperature and Humidity both are 16-bit
- Accuracy: ±1°C and ±1%

3.2.5 Solar Panel with PWM Controller and lead

In Figure 3.6 PWM Controller Setup for Solar Panel in an off-grid solar system also prevents reverse current from the batteries to the solar panels during the night or on overcast days. Depending on the kind, it may boost system efficiency and maximise electricity gathering from solar panels. Furthermore, a charge controller usually contains monitoring functions that enable system metrics like current, voltage, and energy to be tracked [23].

Figure 3.6: PWM Controller Setup for Solar Panel and Battery.

3.2.5.1 Specification of Power Source

- Solar Panel: 16V 18W Solar Panel
- Solar Panel Size: 280mmX280mm
- PWM Rated voltage and current: 12V/10V
- PWM USB*2 Output: 5V / 3A MAX
- Battery: Rechargeable Sealed Lead Acid VRLA
- Battery Voltage: 12 V 7.2AH

3.2.6 R385 DC 12V Pneumatic Diaphragm Water Pump Motor

An electromechanical device called a water pump raises water pressure so that it can be moved from one location to another. Modern water pumps are used all over the world to supply water for purposes in municipal, industrial, agricultural, and domestic settings. Figure 3.7 shows R385 DC 12V Pneumatic Diaphragm Water Pump Motor.

Figure 3.7: R385 DC 12V Pneumatic Diaphragm Water Pump Motor.

3.2.6.1 Specification Motor Water Pump

- Pump size: 40 x 90 x 35 (mm).
- Outlet diameter: 6 mm
- External diameter: 8.5 mm.
- Operating voltage of 6~12VDC (Recommended use 9V 1A, Or 12V 1A).
- Operating current of 0.5~0.7A
- Flow rate of 1.5~2L /Min.
- Capable of pumping heated liquids up to 80 degrees Celsius.
- Suck water through the tube from range up to 2m
- Pump water vertically for up to 3m.
- Lifecycle up to 2500 hours.

3.2.7 Jumper

Figure 3.8: Jumper Wire.

A jumper wire is a kind of electrical cable used to link printed circuit boards that are located far apart. Figure 3.8 shows the example look of jumper wire. Connecting a jumper wire to an electrical circuit may short-circuit it and cause a jump. Jumpers (also known as jumper wires) for solderless breadboarding may be purchased in sets or made from scratch. Larger circuits may need more labour. Ready-to-use jumper wires are available in various quality, including those with little plugs connected to the ends.

3.2.8 Water Sprinkler

Figure 3.9: Ways 4mm Nozzle Sprinkler

The 4 ways 4mm nozzle misting system is a versatile and effective solution for various applications. Here are some key points to consider:

- A 4mm nozzle is suitable for misting systems where a medium to fine spray is required, ideal for cooling large areas.
- The 4-way nozzle configuration allows for even distribution of mist in multiple directions, making it suitable for wide-range applications.
- The water pressure generally increases the distance the water can be sprayed. For instance, a garden hose with a nozzle set to high pressure can have a spray radius ranging 5 feet (1.5meter) maximum.

3.3 Block Diagram and Flow Chart

3.3.1 Block Diagram

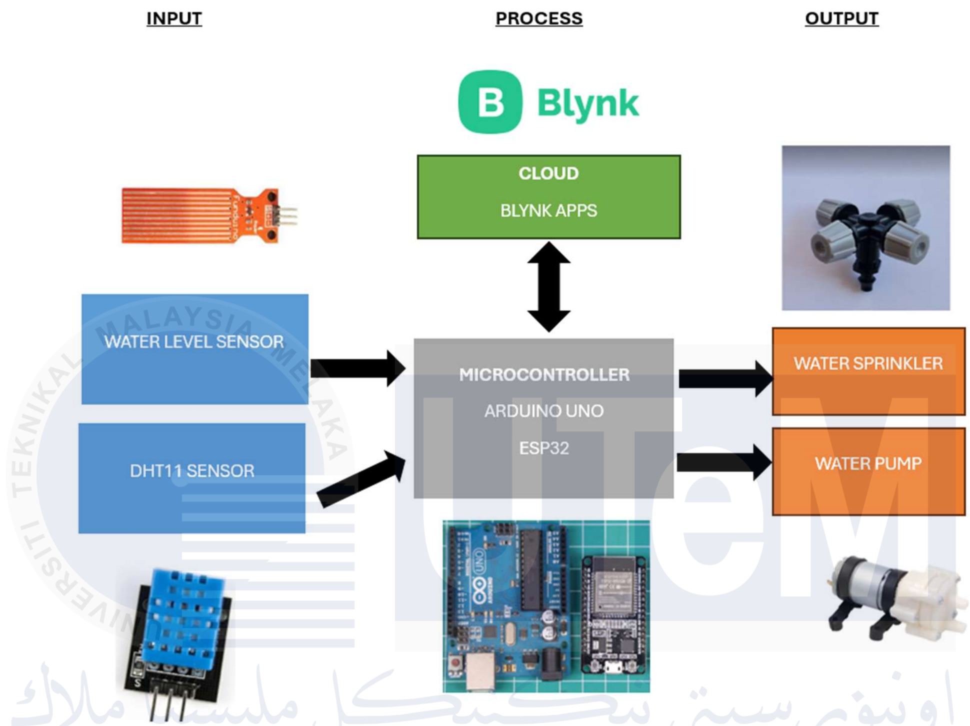

Figure 3.10: Block Diagram

Figure 3.9 Block diagram explains about input, process and output of this project. As for the input water level sensor and temperature sensor. For the process, the sensors will send information to microcontroller, which is Arduino Uno, from there to ESP32 will send the status of the sensor to Blynk Application. Finally for output, water pump and water sprinkler will work as what the microcontroller ask to do.

3.3.2 Flow chart

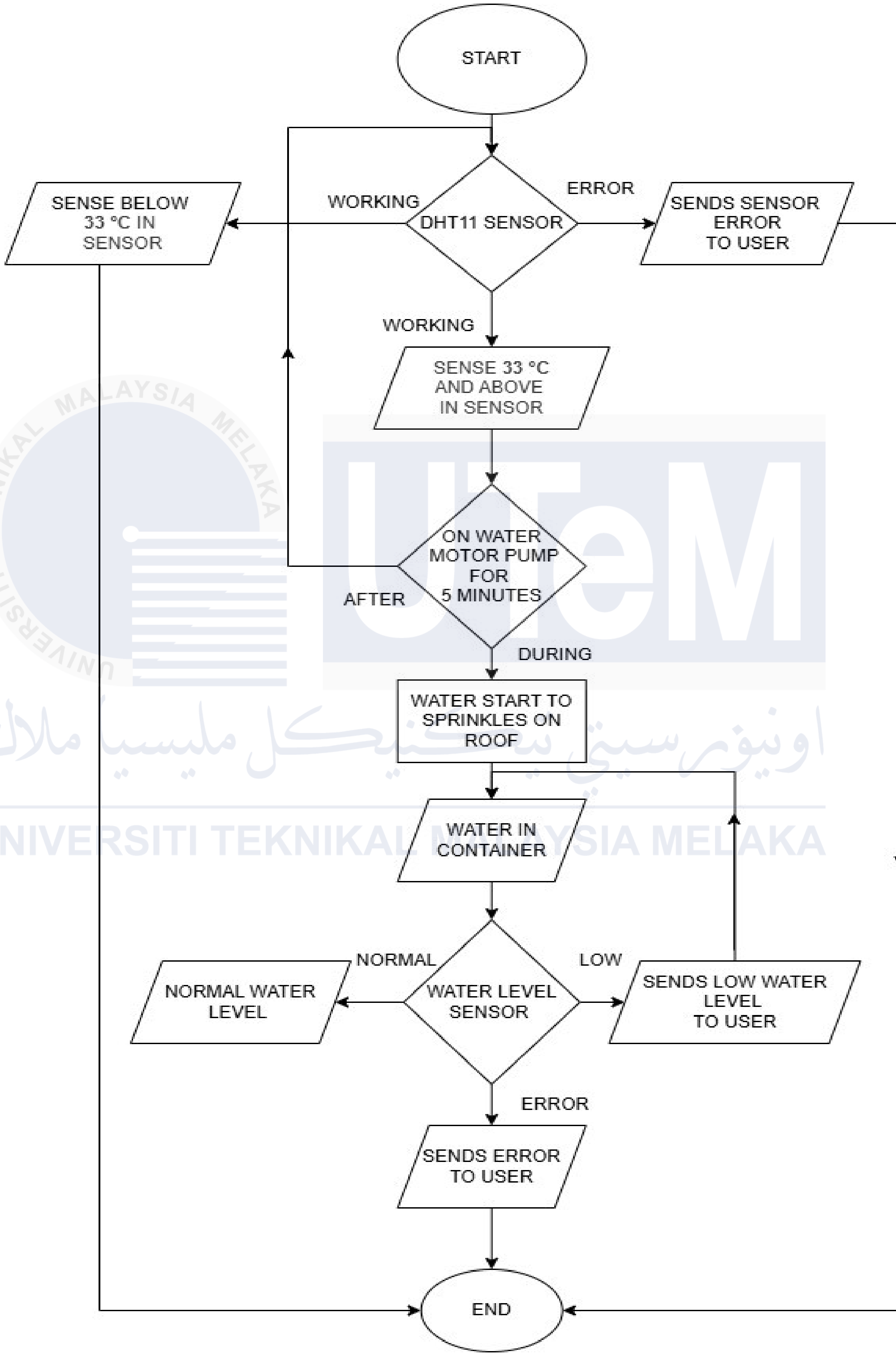

Figure 3.11: Flow Chart.

In Figure 3.10 Flow Chart depicts how the water motor pump system works when different conditions concerning the water levels and temperature are present or when the user is receiving the notification concerning the status of the water motor pump system. Here is a step-by-step explanation of the flowchart:

1) Start: The process begins when the system is powered on.
2) Temperature Sensing (DHT11 Sensor):
    a. The DHT11 sensor continuously monitors the temperature.
    b. If the sensor is working, it proceeds to check the temperature.
    c. If the sensor encounters an error, it sends an error notification to the user and terminates the process.
3) Temperature Threshold Check:
    a. If the sensed temperature is below 33°C, the system loops back to continue monitoring the temperature.
    b. If the temperature is 33°C or above, the system activates the water motor pump.
4) Activate Water Motor Pump:
    a. The water motor pump is turned on for 5 minutes to sprinkle water onto the roof.
5) Water Flow Monitoring:
    a. During this time, water flows through the sprinklers, and the water level in the container is monitored.
6) Water Level Check (Water Level Sensor):
    a. The water level sensor checks the water level in the container:

b. Normal Water Level: If the water level is sufficient, the system continues functioning normally.

c. Low Water Level: If the water level is low, the system sends a "Low Water Level" warning to the user.

d. Sensor Error: If the water level sensor fails, an error message is sent to the user.

7) Loop or End:

    a. If everything is working correctly, the system loops back to monitor the temperature again.

    b. If there is any error (temperature sensor or water level sensor), the process ends with a user notification.

8) End: The process terminates once an error occurs or if the system is manually stopped.

3.4 BDP 2 Milestones

Table 3.2: Gannt Chart of BDP 2.

WEEK / ACTIVITY	1	2	3	4	5	6	7	8	9	10	11	12	13	14
BDP 2 BRIEFING.	■	■												
BUY COMPONENTS.			■	■		■	■							
SOFTWARE & HARDWARE SIMULATION						■	■	■	■	■				
TESTING COMPONENTS & WRITE CODE				■			■	■	■					
DO PROTOTYPE					■	■	■	■	■	■				
MEETING WITH SUPERVISOR		■	■		■	■		■	■			■		
LOGBOOK SUBMISSION							■			■				
REPORT SUBMISSION													■	
PRESENTATION														■

Table 3.2 shows how a Gantt chart is utilized to visually represent project activities. The Gantt chart will outline all the tasks in this project, with the sequence shown against timeframes. Therefore, the project schedule and planning must be carefully planned to complete a project on time. The deadline for critical schedules must be met to ensure that no procedure is missed. We will be able to monitor and assess everything that must be done ahead of time, as well as determine when the activity should be completed.

3.5 Summary

Chapter 3 is methodology, which the requirement component and equipment are listed out for Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino. Not only that, but the block diagram and process flow of this project are shown. Data transmission is typically achieved through WIFI networks. Additionally, solar panels and rechargeable batteries are commonly used as the power source for the system. The BDP 2 milestone is derived by Gannt chart table.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter presents the results and analysis on the Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino. This chapter will present a result, and discussion is a section where it is decisive part of any research paper where it represents the objectives and analysis of the project. This section allows to interpret the data collected and able to proof the project outcome. Within this section, it is also able to present the results of the project outcome and discuss the deficiency of the outcome.

4.2 Hardware Simulation Connections

4.2.1 Schematic Diagram and Simulation

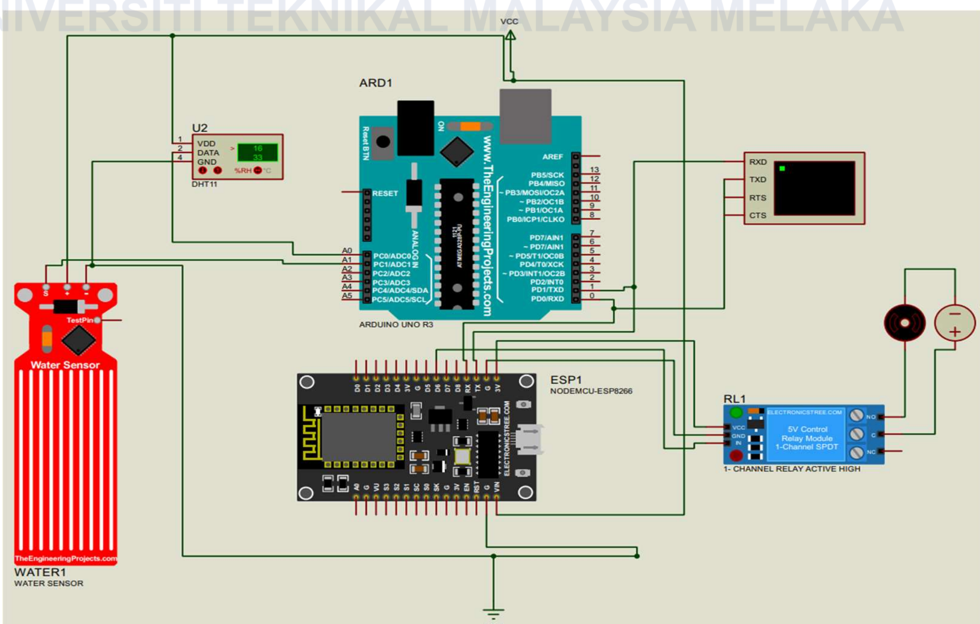

Figure 4.1: Schematic Diagram.

Based on the Figure 4.1, shows about simulation of Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino in Proteus Software. This simulation is successful. The motor is represented water sprinkler, virtual terminal is represented Blynk app and the important is the power supply is using the VCC to replace Solar power and Battery with PWM controller. According to my project, using components are Arduino Uno, ESP32, DHT11, and Water Level sensor. But as for the coding is still in under development.

For the motor, relay is used to control the 12V pump because it allows a low-power control signal from the microcontroller (operating at 3.3V or 5V) to safely switch the higher voltage and current required by the pump. Relays provide electrical isolation between the control circuit and the pump, protecting the microcontroller from power surges or back-EMF generated by the motor. They also handle the high current demands of the pump, which the microcontroller's GPIO pins cannot directly support. When the microcontroller sends a signal, the relay activates, connecting the 12V power supply to the pump, and turns it off when the signal stops. In Figure 4.2 a simulation of Development of An IoT-based Solar Powered Roof Water Sprinkler project.

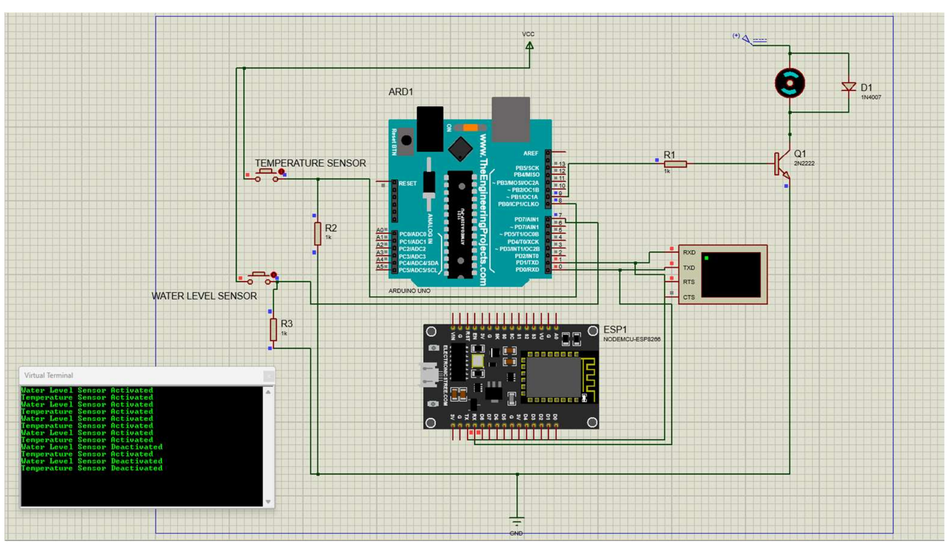

Figure 4.2 Simulation.

4.2.2 DHT11 sensor to Arduino

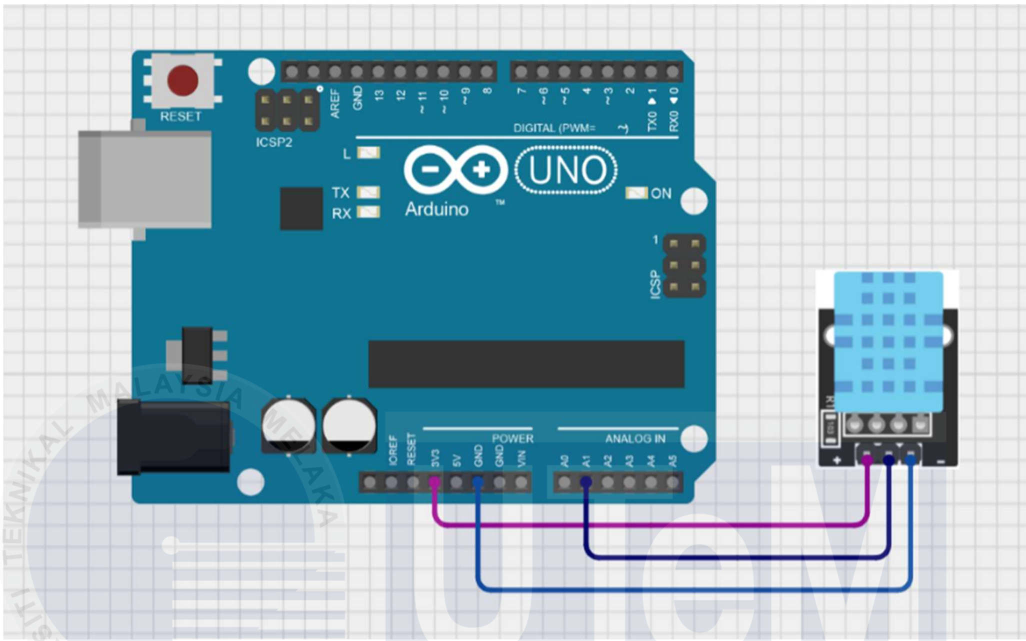

Figure 4.3: DHT11 with Arduino Uno

Figure 4.3 illustrates the connection of a DHT11 temperature and humidity sensor to an Arduino Uno. The DHT11 sensor has three pins connected to the Arduino. The VCC pin of the DHT11 is connected to the 5V pin of the Arduino, providing power to the sensor. The GND pin of the DHT11 is connected to the GND pin of the Arduino, ensuring a common ground. The data pin of the DHT11 is connected to one of the digital pins on the Arduino, typically pin A1 in this case, allowing the Arduino to read the temperature data from the sensor.

4.2.3 Water Level sensor to Arduino Uno

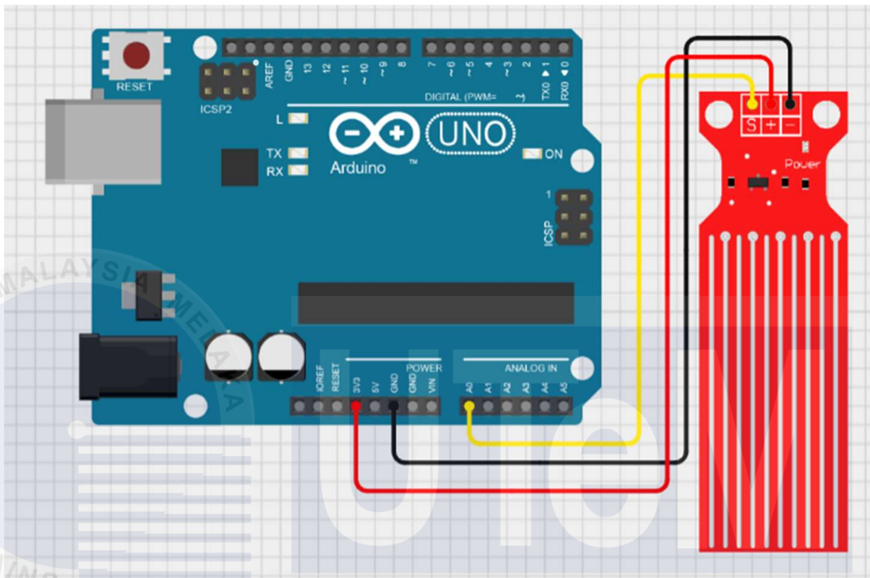

Figure 4.4: Water Level Sensor with Arduino Uno

For Figure 4.4, the connection of a water level sensor to an Arduino Uno R3 was demonstrated. First the water level sensor VCC pin was connected to 3.3V pin of the Arduino Uno to provide power supply. The sensor GND pin was connected to any GND pin on the Arduino to establish a common ground. The water level sensor DATA pin was connected to the Arduino Uno analogue pin 0 (A0).

4.2.4 4.2.4 R385 DC 12V Pneumatic Diaphragm Water Pump Motor

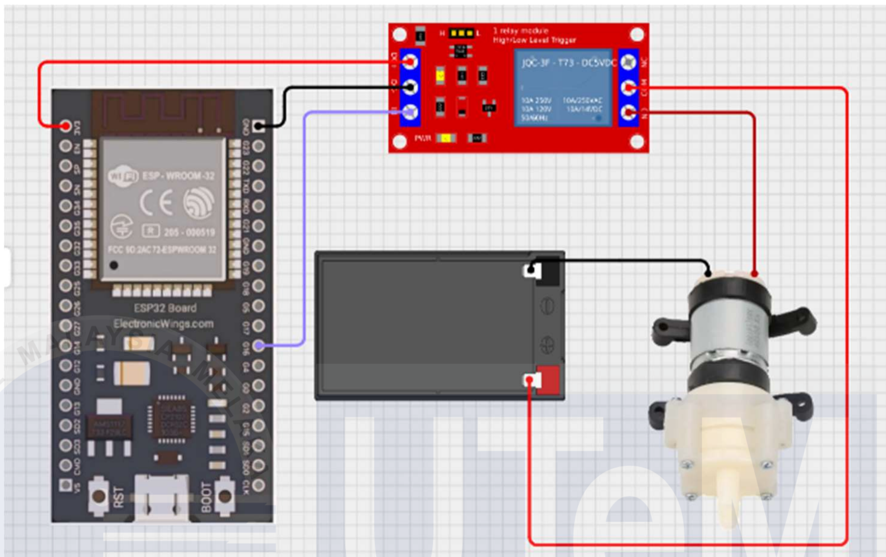

Figure 4.5: Water Pump Motor with Battery, Relay and Arduino Uno.

In Figure 4.5, involves an ESP32 microcontroller, a relay module, a DC pump, and a battery or power source. The ESP32 controls the relay module through GPIO 16, allowing it to act as an electronic switch for the pump. The relay module has its input pins connected to the ESP32, while its power pins (VCC and GND) are connected to the ESP32's 3.3V or 5V output and ground. The relay's common (COM) terminal is connected to the positive terminal of the battery, and its normally open (NO) terminal is connected to the positive terminal of the pump. The pump's negative terminal is connected to the negative terminal of the battery. This configuration enables the ESP32 to toggle the relay's state, thereby controlling the pump's operation. The battery serves as the power source for the pump.

4.2.5 ESP32 module to Arduino Uno

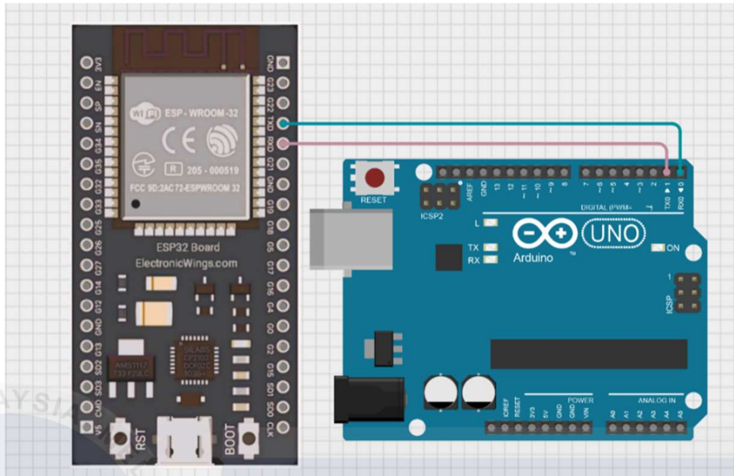

Figure 4.6: ESP32 with Arduino Uno.

This setup establishes a serial communication link between an ESP32 and an Arduino Uno. The TX (Transmit) pin of the ESP32 is connected to the RX (Receive) pin of the Arduino Uno, while the RX (Receive) pin of the ESP32 is connected to the TX (Transmit) pin of the Arduino Uno. This enables data transmission between the two devices, allowing one to send and the other to receive data over the serial connection. This configuration is typically used to share data or control signals between the two boards in projects requiring both microcontrollers.

4.2.6 Solar Cell Charge Battery with Components

This circuit diagram depicts a solar-powered IoT water management system. The solar panel charges a 12V battery through a charge controller, which powers the Arduino Uno and the ESP32. The Arduino communicates with two sensors: a DHT11 (VCC to 5V, GND to GND, and data to A1) and a water level sensor (VCC to 5V, GND to GND, and signal to A0). The ESP32, powered by USB or 3.3V, operates a relay module (GPIO D15 and D16) to power the water pump, which is connected to the battery.

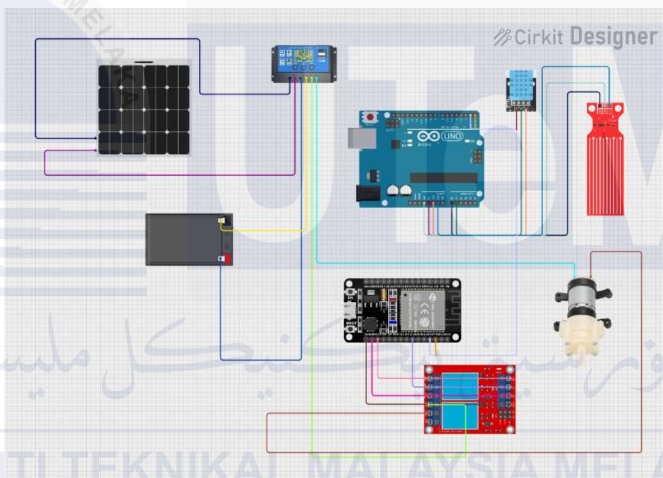

Figure 4.7: Complete prototype circuit with Solar Cell.

This Figure 4.7 depicts a solar-powered, IoT-based automated water management system intended for efficient and sustainable irrigation. The system uses a solar panel to charge a 12V battery via a charge controller, ensuring that the components receive constant power even when there is little sunlight. The main controller is the Arduino Uno, which is linked to a DHT11 sensor for temperature and a water level sensor to monitor water levels. Together, the Arduino and the ESP32 manage sensor inputs and control outputs such as the water pump. The relay module is controlled by the ESP32 and acts as a switch for the water pump, turning it on or off based on predefined conditions.

4.3 Hardware Testing

Before going to project execution, every hardware was tested using a basic source code given by the provider.

4.3.1 DHT11 sensor testing

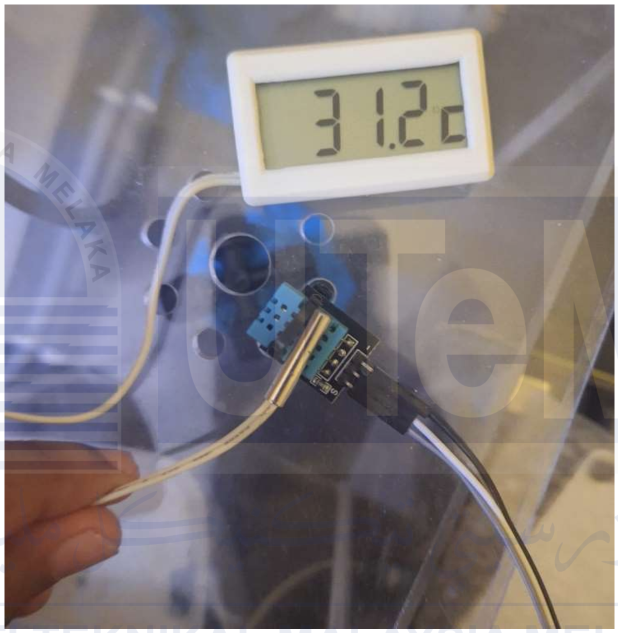

Figure 4.8: Calibrating DHT11

Figure 4.8 demonstrates how to test a DHT11 temperature and humidity sensor with a temperature calibrator. The DHT11 sensor is connected to a circuit, most likely via jumper wires, and placed near a digital thermometer that shows a reference temperature of 31.2°C. The goal of this setup is to ensure the accuracy of the DHT11 sensor by comparing its temperature readings, obtained via a microcontroller like an Arduino or ESP32, to the reference temperature displayed on the digital display. Differences in readings can be identified by ensuring that both sensors are exposed to the same environmental conditions. This procedure is frequently used to calibrate or validate the DHT11 sensor's performance, and any deviations can be corrected in software to improve accuracy.

4.3.2 Water Level sensor testing

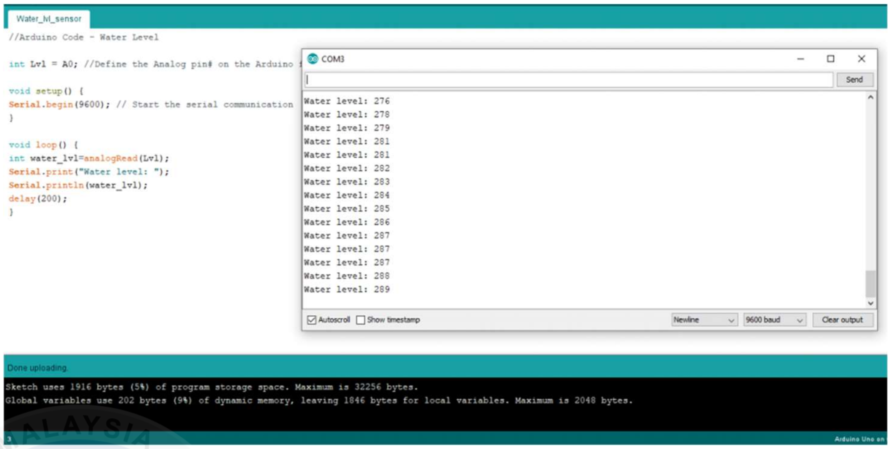

Figure 4.9: Arduino IDE with Water Sensor Program

The Figure 4.9 shows an Arduino IDE interface with a water level monitoring program. The code defines an analogue pin (A0) for reading the water level data. In the setup() function, the serial communication is initialized with a baud rate of 9600. In the loop() function, the water level is read using the analogRead() function and assigned to the variable water_lvl. The value is then printed to the serial monitor along with the label "Water level: ". In Figure 4.10 shows water level sensor is sensing the water according to the coding.

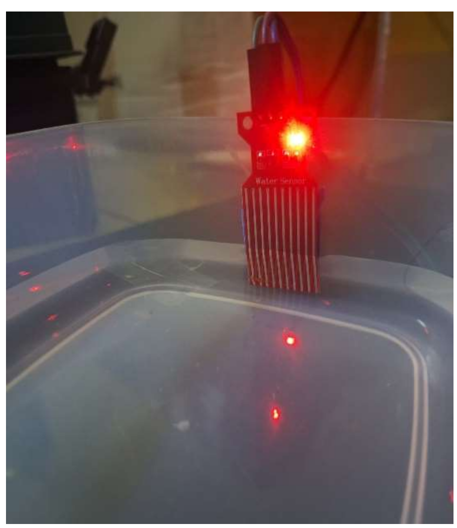

Figure 4.10: Water Level Sensor is Sensing the Water.

4.4 Prototype Preparation

Figure 4.11 shows the prototype that was created for a full circuit and placing all the components on board. The Arduino Uno and ESP32 were neatly arranged within the box and covered with the lid. A hole must be drilled beneath the box for wires. The solar cell and battery were also placed on the board to link to the board in the box and to turn on the prototype for a brief period. Soldering, a cutter, plywood board, clay and acrylic box were used for preparing the prototype. Figure 4.12 shows the connection between Arduino Uno and ESP32 in a box.

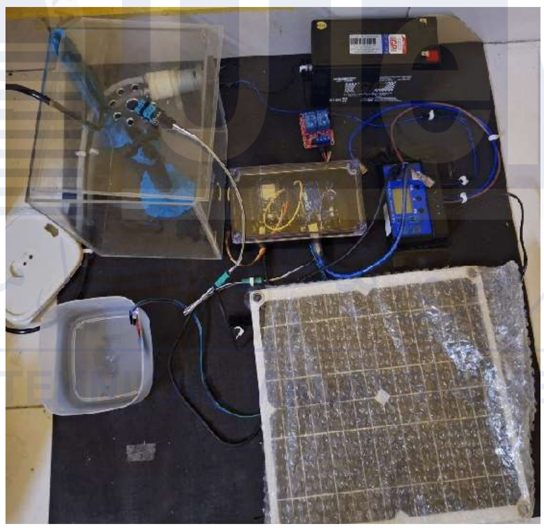

Figure 4.11: Full Prototype Circuit.

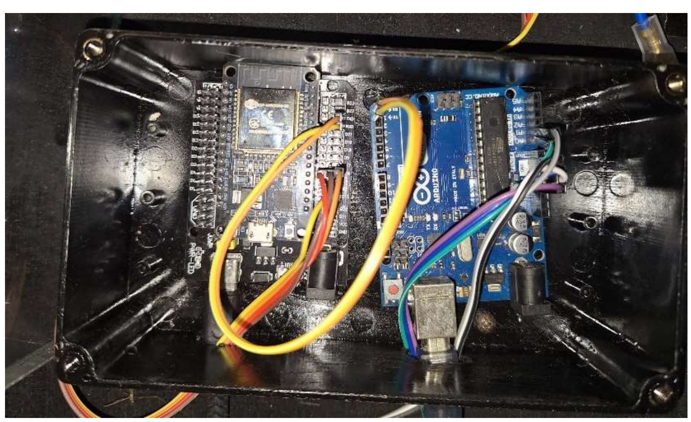

Figure 4.12: Connection between Arduino Uno and ESP32.

4.5 Result

This part consists of the output from the overall system after the Arduino board and ESP32 board supplied with % volt of input power. Figure 4.13 shows the solar will be charging the battery when it detects the light of the sun and the PWM solar controller will show the icon as the results of active charging to the battery. In Figure 4.14 the full prototype of Development of An IoT-Based Solar Powered Roof Water Sprinkler by Using Arduino.

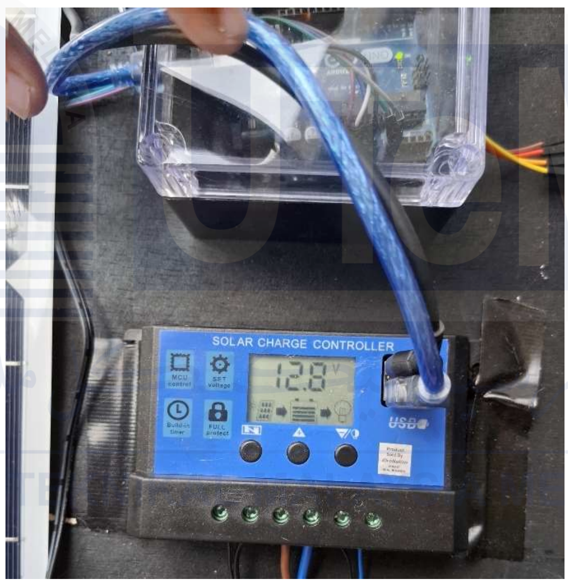

Figure 4.13: PWM Controller showing the Solar Cell is charging the Battery.


Figure 4.14: Full Prototype at Outside.

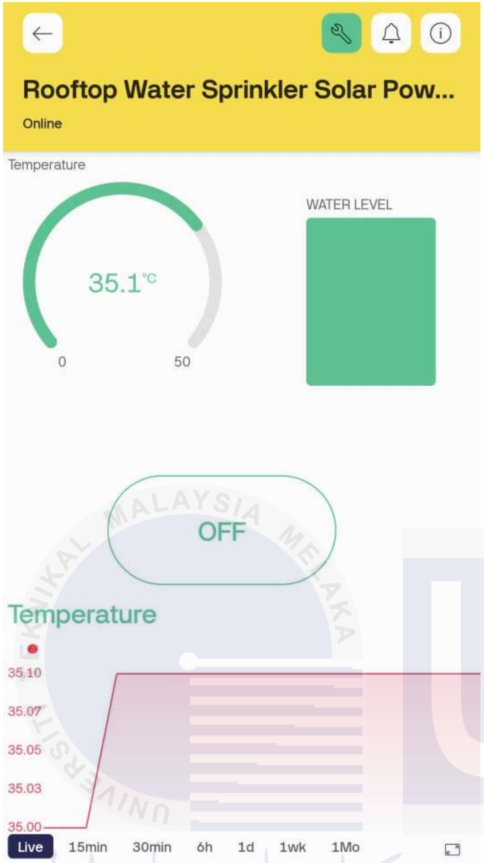

Figure 4.15 Blynk Interface.

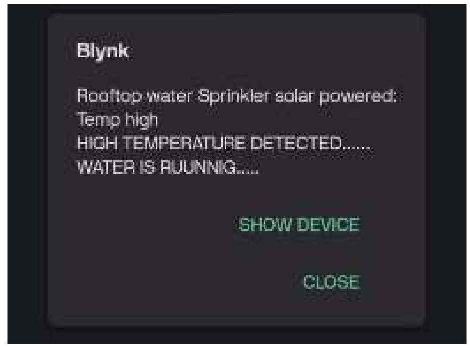

Figure 4.16: Blynk Interface for High Temperature.

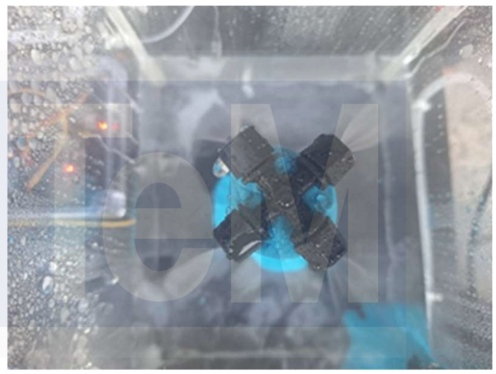

Figure 4.17: Water Sprinkle when Temperature is High.

Figure 4.15 shows the Blynk interface of running the output result of the surrounding temperature, water level measurement, and motor pump state. The connection with ESP32, is to access with cloud (Blynk) to monitor the temperature, water level, and manually motor control. During this process, the sensor of the temperature and water level will continuously show the reading because the surrounding temperature will keep changing. In Figure 4.16 shows the Blynk interface for high temperature. In Figure 4.17, the water starts sprinkle when the temperature is above 33 degrees Celsius.

Figure 4.18: Notification of High Temperature.

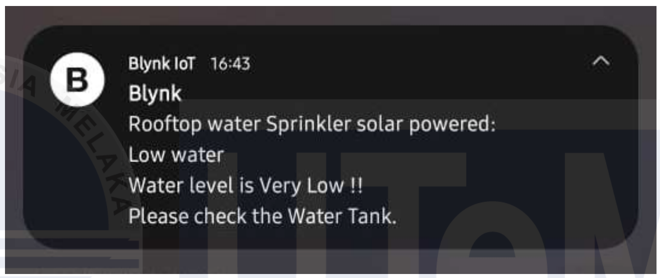

Figure 4.19: Notification of Low Water Level.

In Figure 4.18, Figure 4.19 show that the Blynk application will send the notification when the water level is too low, temperature high to alert the user. Whenever the water pump is running, the temperature sensor will pause the temperature sensor functionality.

Table 4.1 and Table 4.2 shows the setup of the system for Arduino Uno and ESP32 from the coding in Arduino IDE software.

Table 4.1: Arduino Uno Code.

```
#include <DHT.h>    // Define the DHT11 pin and type
#define DHTPIN A1       // Pin connected to the DHT11
#define DHTTYPE DHT11   // DHT11 sensor

// Define the water level sensor pin
const int waterLevelSensorPin = A0; // Analog pin for water level sensor

// Initialize the DHT sensor
DHT dht(DHTPIN, DHTTYPE);

// Variables for non-blocking timing
unsigned long previousMillis = 0;
const unsigned long dhtInterval = 2000; // DHT11 sampling period (2 seconds)

void setup() {
 // Start the DHT sensor
 dht.begin();

 // Start Serial communication with ESP32
 Serial.begin(115200);
}

void loop() {
 unsigned long currentMillis = millis();

   float temperature = dht.readTemperature();
   float humidity = dht.readHumidity();

   // Check if readings are valid
   if (!isnan(temperature) && !isnan(humidity)) {
    // Send temperature to ESP32
    Serial.print("T:");
    Serial.println(temperature);
   } else {
    Serial.println("Error reading DHT11 sensor!");
   }
```

```
  }

  // Simulate water level sensor
  int waterLevel = analogRead(waterLevelSensorPin);
  int waterLevelPercentage = map(waterLevel, 0, 1023, 0, 100);

  // Send water level data to ESP32
  Serial.print("W:");
  Serial.println(waterLevelPercentage);

  delay(500); // Small delay for stability
}
```

Table 4.2: ESP32 Code.

```
#define BLYNK_TEMPLATE_ID "TMPL6Peoo1Mnk"
#define BLYNK_TEMPLATE_NAME "Rooftop water Sprinkler solar powered"
#define BLYNK_AUTH_TOKEN "ygmW7VZVNyt8EX56Ejja7J"

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>

// WiFi credentials
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "rooba";
char pass[] = "88888888";

// Define motor and relay pins
const int motorPin = 16;
const int relayPin = 15;

// Variables to store data from Arduino Uno
float temperature = 0;
int waterLevel = 0;
bool motorState = false;

// Define thresholds
const int lowWaterLevelThreshold = 10; // Water level threshold for low level
// Timing variables
unsigned long lastTemperatureUpdate = 0;
unsigned long lastWaterLevelUpdate = 0;

// Create a timer instance
BlynkTimer timer;

// Function to turn off the motor after 5 minutes
void turnOffMotor() {
  digitalWrite(motorPin, LOW); // Turn off motor
  motorState = false; // Update motor state
  Serial.println("Motor turned off after 5 minutes.");
  Blynk.virtualWrite(V0, 0); // Update motor control in the Blynk app
}

// Function to handle low water level
void handleLowWaterLevel() {
  if (waterLevel < lowWaterLevelThreshold) {
    Blynk.logEvent("low_water"); // Log event in Blynk app
```

```
    Serial.println("Low water level detected! Notification sent.");
  }
}

// Function to check sensor status
void checkSensorStatus() {
  unsigned long currentMillis = millis();

  // Check if temperature sensor data is not received
  if (currentMillis - lastTemperatureUpdate > sensorTimeout) {
    Blynk.logEvent("sensor_error");
    Serial.println("Temperature sensor not detected! Notification sent.");
    lastTemperatureUpdate = currentMillis; // Reset to avoid repeated notifications
  }

  // Check if water level sensor data is not received
  if (currentMillis - lastWaterLevelUpdate > sensorTimeout) {
    Blynk.logEvent("sensor_error");
    Serial.println("Water level sensor not detected! Notification sent.");
    lastWaterLevelUpdate = currentMillis; // Reset to avoid repeated notifications
  }
}

void setup() {
  // Initialize motor and relay pins as output
  pinMode(motorPin, OUTPUT);
  pinMode(relayPin, OUTPUT);

  // Ensure motor and relay are off initially
  digitalWrite(motorPin, LOW);
  digitalWrite(relayPin, LOW);

  // Start Serial communication with Arduino Uno
  Serial.begin(115200);

  // Initialize Blynk
  Blynk.begin(auth, ssid, pass);
  Serial.println("Connected to WiFi and Blynk.");

  // Perform relay toggling at startup
  for (int i = 0; i < 4; i++) {
    digitalWrite(relayPin, HIGH); // Turn relay on
    delay(200);
```

```
    digitalWrite(relayPin, LOW); // Turn relay off
    delay(200);
  }
}

// Virtual Pin V0: Motor control from Blynk app
BLYNK_WRITE(V0) {
    int motorControl = param.asInt();
    if (motorControl == 1) {
        motorState = true;
        digitalWrite(motorPin, HIGH); // Turn on motor
        delay(50); // Short delay to allow transients to settle
    } else {
        motorState = false;
        digitalWrite(motorPin, LOW); // Turn off motor
        delay(50); // Short delay to allow transients to settle
    }
}

void loop() {
  Blynk.run();
  timer.run();

    // Parse temperature data
    if (data.startsWith("T:")) {
      temperature = data.substring(2).toFloat();
      Blynk.virtualWrite(V1, temperature); // Send to Virtual Pin V1
      lastTemperatureUpdate = millis(); // Update last temperature received time

      // Automatic motor control logic based on temperature
      if (temperature > 33 && !motorState) {
    motorState = true;
    digitalWrite(motorPin, HIGH); // Turn on motor
    delay(50); // Short delay to allow transients to settle
    Blynk.logEvent("temp_high");
    Serial.println("Water Pump Turn ON");

    // Set timer to turn off motor after 5 minutes (300000 ms)
    timer.setTimeout(300000, turnOffMotor);
} else if (temperature <= 33 && motorState) {
    digitalWrite(motorPin, LOW); // Turn off motor
    motorState = false;
    delay(50); // Short delay to allow transients to settle
```

```
      Serial.println("Water Pump Turn OFF");
      Serial.println("System Normal");
    }

      }
      // Parse water level data
      if (data.startsWith("W:")) {
        waterLevel = data.substring(2).toInt();
        Blynk.virtualWrite(V2, waterLevel); // Send to Virtual Pin V2
        lastWaterLevelUpdate = millis(); // Update last water level received time

        // Check for low water level and send notification
        handleLowWaterLevel();
      }
    }

    // Check sensor status periodically
    checkSensorStatus();
  }
```

4.6 Data Analysis

The project's study was based on temperature sensing and IOT. Several tests on sensor response time of DHT11 temperature sensor and the response time of BLYNK application with DHT11 sensor. Also, the estimations of battery charging time and battery life when the project is deployed. Showing the graph of current and voltage usage during the device turn on and included the calculation of how long battery life of the prototype. Also shows the graph for solar current and voltage of charge time to the device and the calculation of solar panel how long charge time to the battery.

4.6.1 DHT11 Temperature Sensor Result

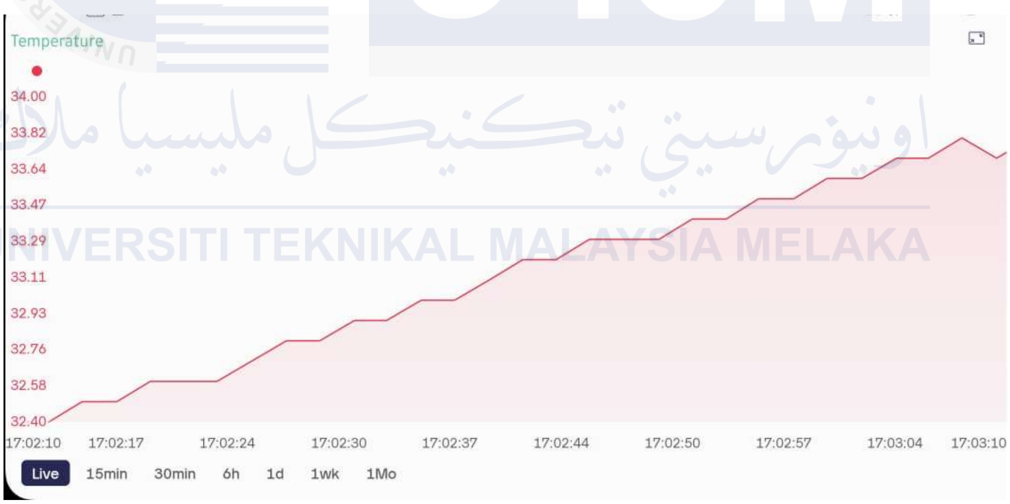

Figure 4.20: Temperature rising due to hot weather.

The Figure 4.20 shows a graph that gradual rise in temperature over time, beginning around 32.40°C and peaking at 34.00°C. This steady increase indicates that the temperature is uncontrolled until it exceeds the 33°C threshold, at which point a system is activated to regulate it.

Figure 4.21: Temperature is decrease due to cooling the environment.

The Figure 4.21 shows a graph that cooling phase initiated by the system once the temperature exceeds 33°C. The graph shows a consistent decline, with the temperature dropping from 34.60°C to around 33.44°C in a relatively short period of time. This demonstrates the system's ability to bring the temperature back down to safe levels after being triggered.

4.6.2 Water Flow Rate

The calculation of water flow rate helps determine the volume of water passing through a system over time. Using the given flow data—1.5 liters in 132 seconds—we can calculate the flow rate in liters per second. From this, we can estimate the total volume of water flow over longer durations, such as 5 minutes and 1 hour. This is useful in various applications, including plumbing, irrigation, and industrial systems.

4.6.2.1 5-Minute Flow of Water

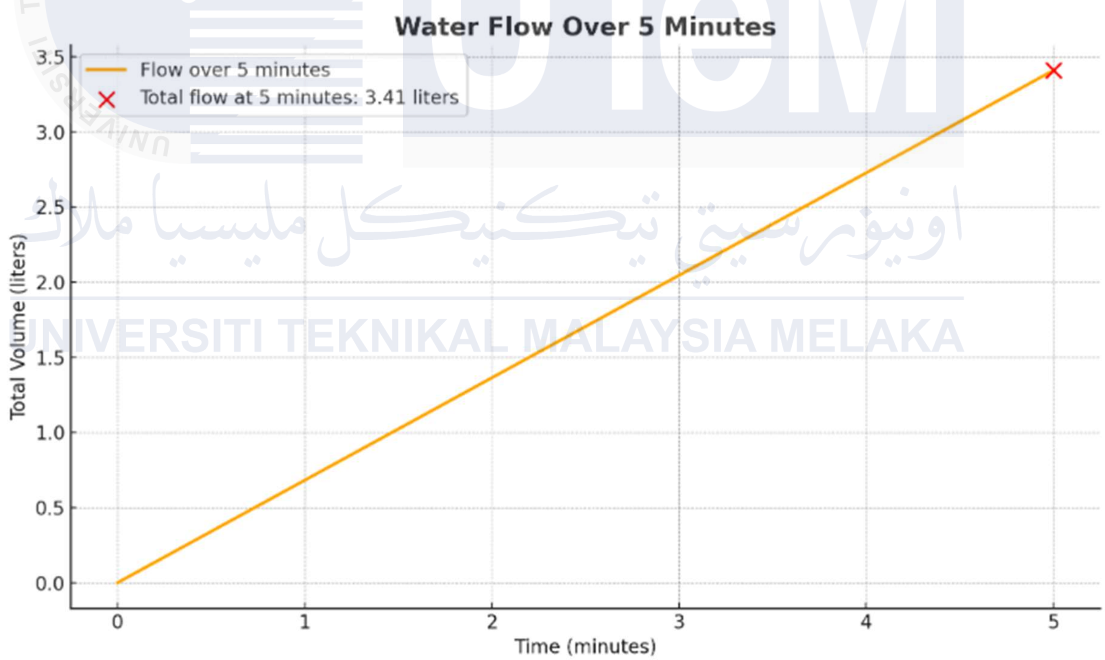

Figure 4.22: Water flow in 5 Minutes.

The Figure 4.22 shows the graph represents the cumulative water flow over a short interval of time (300 seconds). The orange line indicates the total volume of water increasing linearly as time progresses, based on the calculated flow rate of 0.01136 liters per second. At the 5-minute mark, the red marker highlights the total flow reaching 3.41 liters, providing a clear visualization of the short-term water usage.

The calculation for water flow in 5 minutes (300 seconds) is shown below:

$$\boldsymbol{Flow\ Rate = 0.01136 liters/second}$$

$$\boldsymbol{5\ minutes\ =\ 5 \times 60 = 300\ seconds}$$

$$\boldsymbol{Total\ Flow\ (5\ minutes) = Flow\ Rate \times Time\ (300\ seconds)}$$

$$\boldsymbol{Total\ Flow\ (5\ minutes) = 0.01136 \times 300 = 3.41 liters}$$

4.6.2.2 1-Hour Flow of Water

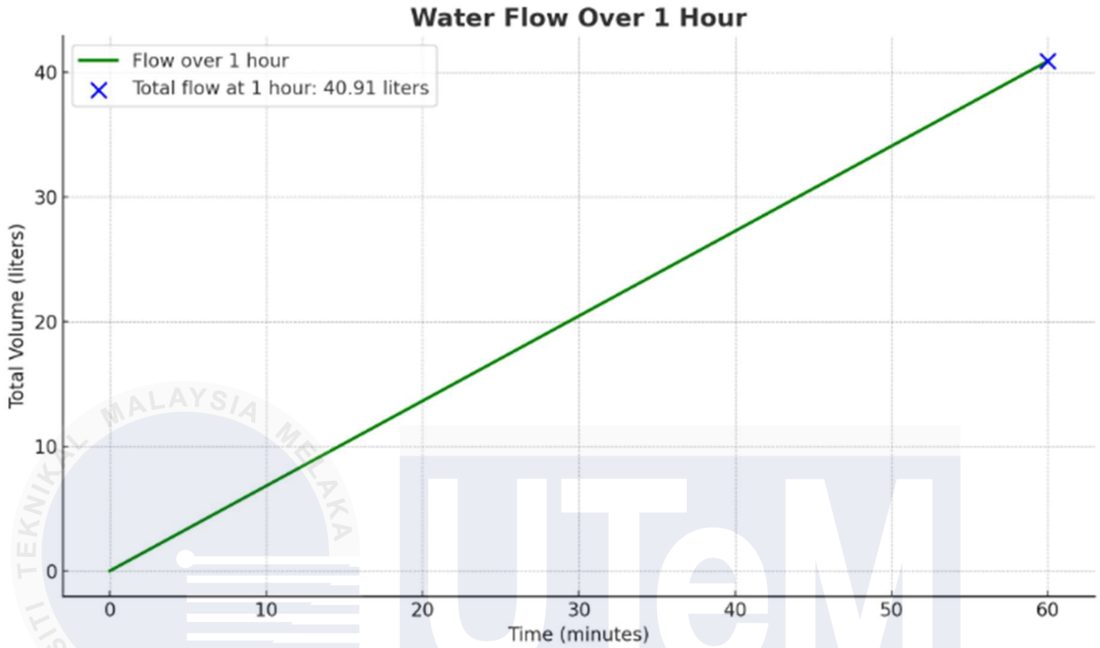

Figure 4.23: Water Flow in 1 hour.

The 1-hour graph illustrates the cumulative water flow over a longer duration (3600 seconds). The green line shows a steady and proportional increase in water volume with time. The blue marker at the 1-hour mark indicates the total flow reaching **40.91 Liters**. This graph effectively demonstrates the long-term behaviour of the water flow system, offering insights into the system's capacity for sustained flow.

The calculation for water flow in 1 hour (3600 seconds) is shown below:

$$\boldsymbol{Flow\ Rate = 0.01136 liters/second}$$

$$\boldsymbol{60\ minutes\ =\ 60 \times 60 = 3600\ seconds}$$

$$\boldsymbol{Total\ Flow\ (1\ hour) = Flow\ Rate \times Time\ (300\ seconds)}$$

$$\boldsymbol{Total\ Flow\ (1\ hour) = 0.01136 \times 3600 = 40.91 liters}$$

4.6.3 The Usage of Battery on Device

The Figure 4.24 shows a graph of current vs. time chart depicts the system's varying current consumption over the operating period. Initially, the current remains at a low level, indicating the operation of low-power components such as the Arduino Uno, ESP32, and sensors (DHT11 and water level sensor). When the temperature rises above 33°C, the R385 DC 12V Pneumatic Diaphragm Water Pump Motor turns on for 5 minutes. During this time, the current surges to meet the motor's increased power demand, which is the system's largest consumer. After the motor operation is completed, the current returns to its baseline level. This fluctuation demonstrates how the system adjusts its power consumption based on operational needs.

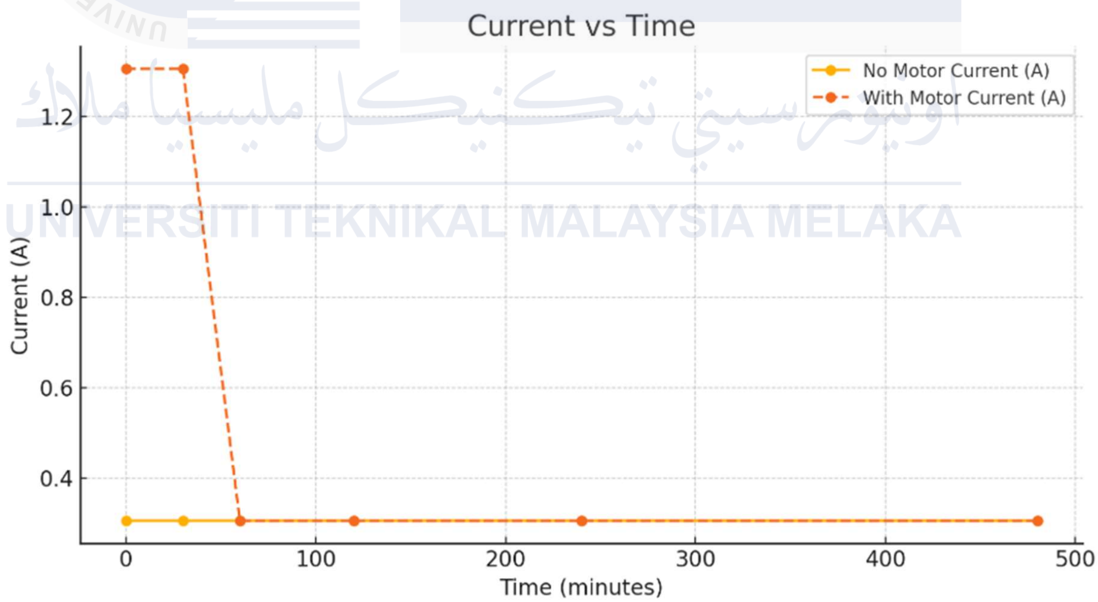

Figure 4.24: Current vs Time Graph for Battery Usage.

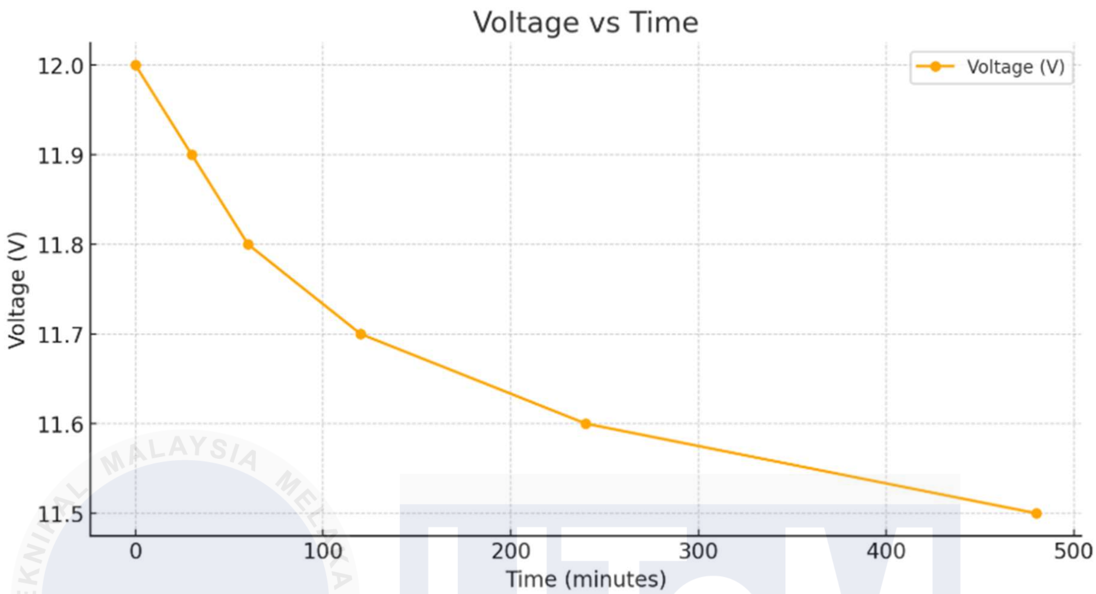

Figure 4.25: Voltage vs Time Graph for Battery Usage.

The Figure 4.25 a graph of voltage vs time chart depicts the gradual depletion of the battery's charge over time. Initially, the voltage remains stable, indicating that the battery can handle the load without significant strain. When the motor is activated, there is a slight drop in voltage due to the increased load, but the system quickly stabilises, demonstrating the battery's ability to support peak demand. After the motor is turned off, the voltage returns to its initial level, demonstrating the system's efficient design and the battery's ability to recover from transient loads. A gradual decline in voltage over time indicates battery discharge, which is to be expected during continuous operation.

4.6.3.1 Calculation for Battery Life

Table 4.2: Current and Voltage Consumption of Each Component.

Components	Current(mA)	Voltage(V)
Arduino Uno	50 mA	5 V
ESP32	240 mA	5 V
DHT11 Sensor	2.5 mA	3.3 V
Water Level Sensor	15 mA	5 V
R385 DC Motor (Active)	2000 mA	12 V

Table 3 shows about the system's components, including current consumption and operating voltage. The Arduino Uno operates at 5V and consumes 50mA, whereas the ESP32 requires 5V and consumes 240mA during operation. The DHT11 sensor, which measures temperature and humidity, operates at 3.3V and draws only 2.5mA. The water level sensor operates at 5V and draws 15mA. Finally, the most power-intensive component is the R385 DC motor, which runs at 12V and draws 2000mA (2A) when active. These specifications are critical for designing the power supply and ensuring the system's efficiency.

So, need few calculation steps to calculate the battery life:

1. Current consumption during daytime:

$$\boldsymbol{Current\ Without\ Motor = 50 + 240 + 2.5 + 15 = 307.5mA}$$

$$\boldsymbol{Current\ With\ Motor = 307.5 + 2000 = 2307.5mA}$$

2. Average current consumption during daytime over 1 hour:

$$\boldsymbol{Daytime\ Average\ Current} = \left(\boldsymbol{Motor\ Current} \times \frac{\mathbf{5}}{\mathbf{60}}\right) + \left(\boldsymbol{Idle\ Current} \times \frac{\mathbf{55}}{\mathbf{60}}\right)$$

$$\boldsymbol{Daytime\ Average\ Current} = (\mathbf{2307.5} \times \mathbf{0.0833}) + (\mathbf{307.5} \times \mathbf{0.9167})$$

$$\boldsymbol{Average\ Current = 192.3 + 281.9 = 474.2mA}$$

3. For current consumption during nighttime:

$$\boldsymbol{Nighttime\ Current = 307.5mA}$$

4. By combining the average current over 24 hours:

$$\boldsymbol{Average\ Current} = (\boldsymbol{Daytime\ Average\ Current} \times \mathbf{2412}) + (\boldsymbol{Nighttime\ Current} \times \mathbf{2412})$$

$$\boldsymbol{Average\ Current} = (\mathbf{474.2} \times \mathbf{0.5}) + (\mathbf{307.5} \times \mathbf{0.5})$$

$$\boldsymbol{Average\ Current = 237.1 + 153.75 = 390.85mA}$$

5. So as for the battery life calculation, the battery capacity is 7.2 Ah (7200 mAh) at 12 V:

$$\boldsymbol{Battery\ Life\ (hours)} = \frac{\boldsymbol{Average\ Current\ Consumption\ (mA)}}{\boldsymbol{Battery\ Capacity\ (mAh)}}$$

$$\boldsymbol{Battery\ Life} = \frac{\mathbf{7200}}{\mathbf{390.85}} = \boldsymbol{18.42hours}$$

4.6.4 The Usage of Solar Cell Charge to Device

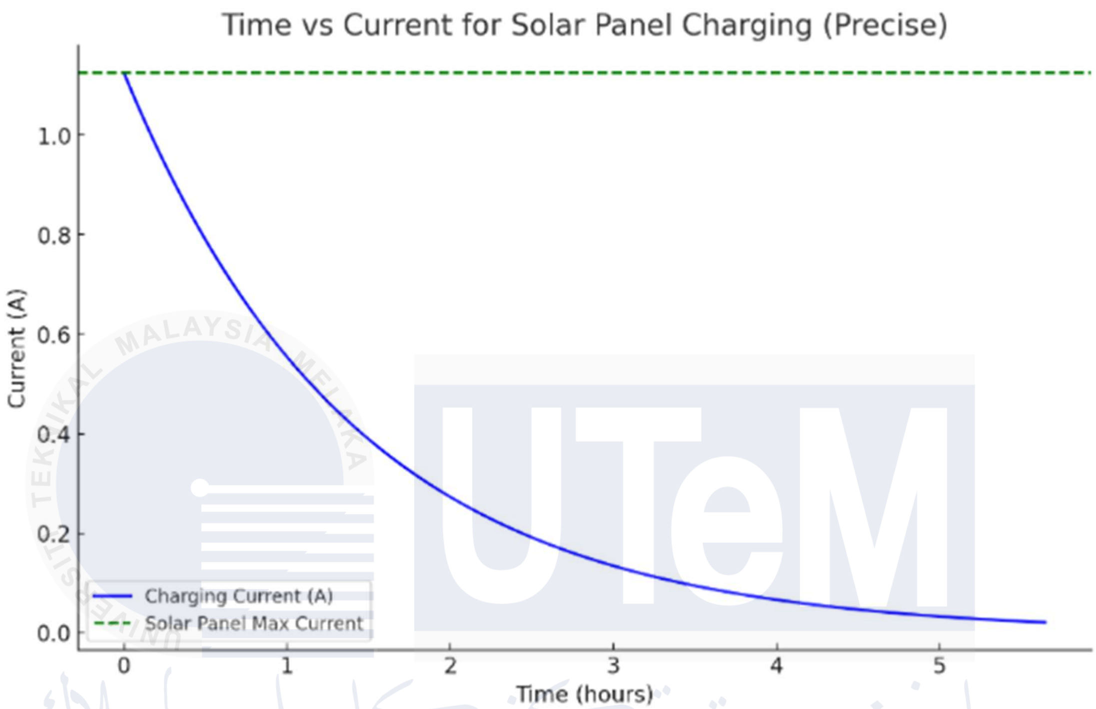

Figure 4.25: Time vs Current for Solar Panel Charging.

The Figure 4.25 shows the current graph shows a decrease in charging current over time, starting at the maximum (1.125 A) and reducing as the battery approaches full charge, reflecting realistic behaviour due to resistance.

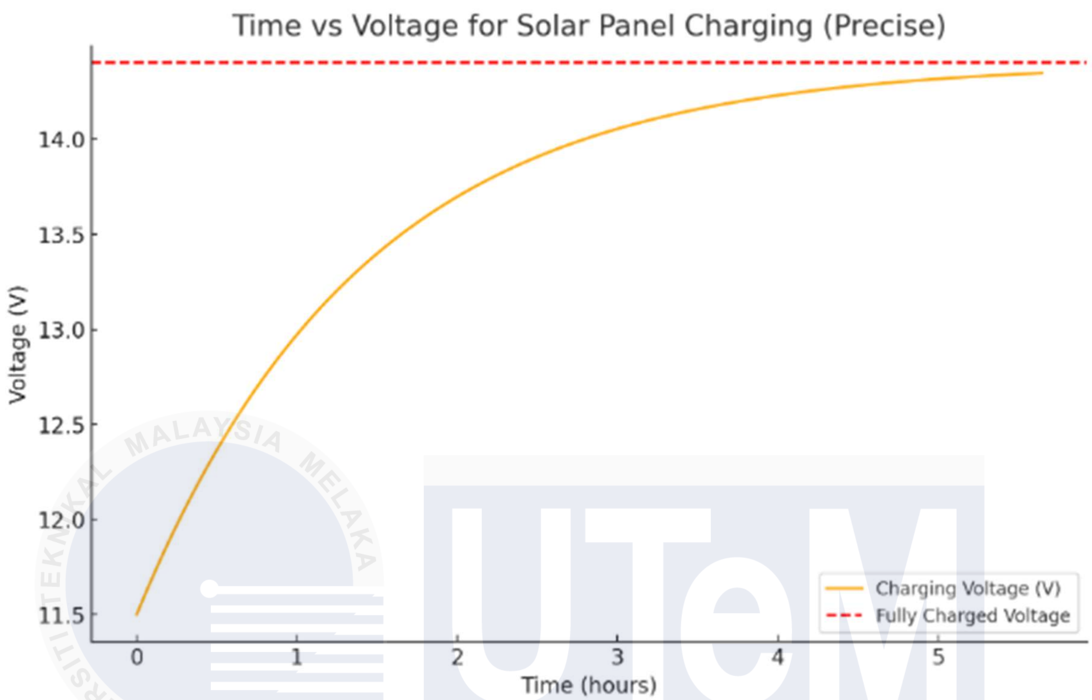

Figure 4.26: Time vs Current for Solar Panel Charging.

The Figure 4.26 shows the voltage graph depicts a gradual increase, starting from the initial battery voltage (11.5 V) and reaching the fully charged state (14.4 V). Together, these graphs illustrate the interplay of current and voltage during the battery charging process, with current decreasing and voltage stabilizing at the end of the charge.

4.6.4.1 Calculation For Solar Panel Charge Time

1. Battery Capacity is 12v, 7.2Ah
2. Find Watt hour Battery capacity:

$$\boldsymbol{Battery\ Capacity\ (Wh) = Voltage \times Capacity\ (Ah)}$$

$$\boldsymbol{= 12V \times 7.2Ah = 86.4Wh}$$

3. Decided Battery is Discharge 80%

$$\boldsymbol{Power\ discharge\ hours = Wh\ \times 80\%}$$

$$\boldsymbol{= 86.4Wh \times 0.8 = 69.12Wh}$$

4. Effective Solar Power with PWM Efficiency (75%):

$$\boldsymbol{P_{effective} = 18W \times 0.75 = 13.5W}$$

5. Estimated Charging Time:

$$\boldsymbol{t_{charge=} \frac{Battery\ Capacity\ (Wh)}{P_{effective}(W)} = \frac{69.12}{13.5} \approx\ 5.12\ hours}$$

4.7 Discussion

The development of an IoT-based solar-powered roof water sprinkler aims to provide an automated solution for lowering indoor temperatures in residential homes while adhering to sustainability goals. This system combines hardware and software components to monitor environmental conditions, activate sprinklers when necessary, and run efficiently on renewable energy sources. The primary goals include creating an automated system capable of providing optimal environmental response, implementing IoT-driven water sprinkling to cool roof surfaces, and utilising solar energy to reduce reliance on traditional energy sources.

The system architecture is made up of Arduino Uno and ESP32 microcontrollers. The Arduino Uno controls the DHT11 temperature and water level sensors, while the ESP32 controls the 12V Pneumatic Diaphragm Water Pump Motor via a 5V relay and provides Wi-Fi connectivity. The sensors are connected using libraries from the Arduino IDE, and programming is done in C++. The Blynk application monitors the data collected by the sensors in real time.

A solar panel, connected to a 10A PWM controller and a 7.2Ah acid battery, provides power to the system. This configuration ensures the system's sustainability by utilising renewable energy, which reduces environmental impact. The solar-powered design demonstrates the project's commitment to reducing reliance on non-renewable energy sources. The operational workflow is straightforward and efficient. The DHT11 sensor continuously monitors the ambient temperature. When the temperature exceeds 33°C, the ESP32 activates the water pump, which sprays water onto the roof for five minutes. This process helps to lower indoor temperatures.

The results showed that the system can successfully lower indoor temperatures while maintaining consistent water flow rates. However, there were some minor delays in data transmission due to Wi-Fi connectivity, which could be improved in future iterations. Despite these minor issues, the system proved to be dependable and effective in meeting its primary goals. Several challenges arose throughout the project, including time management and integrating multiple hardware and software components.

In conclusion, the IoT-based solar-powered roof water sprinkler is a practical and environmentally friendly solution for lowering indoor temperatures in residential homes. By utilising IoT technology and renewable energy, the system meets modern sustainability goals while providing efficient performance. With future upgrades, the system has the potential for broader applications and improved functionality, making it an important contribution to environmentally friendly home cooling technologies.

4.8 Summary

This chapter presents one of the completed deliverables of the project, focusing on the results from the simulation a d development of an IoT-based solar-powered roof water sprinkler. These simulation results can be used to determine the system's operational workability, dependability, and efficiency for proper temperature measurement and sprinkler actuation. These findings make the use of IoT in this context extremely valuable for improving measures of temperature and sprinkling. Testing confirmed the system's ability to reduce indoor temperatures while maintaining consistent operation.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The Development of An IoT-based Solar Powered Roof Water Sprinkler project emphasizes its way to successful development and future effect. The project's goal was to develop an automated system for cooling residential roofs utilizing renewable energy sources of solar electricity. To handle the sprinkling process effectively, the system uses an Arduino Uno microcontroller, an ESP32 Wi-Fi module, DHT11 temperature sensors, and a water level sensor.

Throughout the process, multiple components were carefully integrated to guarantee consistent functioning. When the roof temperature rises over 33-degree Celsius, the temperature sensors activate the sprinkler system. The water level sensor ensures that there is enough water for cooling, while the solar panels supply the energy needed to operate the system, making it self-sustaining and ecologically benign. This initiative targets major challenges such as energy use and greenhouse gas emissions. Traditional air conditioning systems use a lot of energy and harm the environment. The initiative addresses these issues by providing a low-cost, low-power alternative, particularly in locations with insecure or costly electrical supply.

In a nutshell, the Development of an IoT-based Solar Powered Roof Water Sprinkler project, its primary aim of decreasing interior temperatures, but also the feasibility of mixing renewable energy with smart technologies. This program paves the way for future advancements in sustainable home technology, promoting environmentally responsible practices and contributing to a more sustainable future.

5.2 Potential for Commercialization

There is a presupposition that using Development of An IoT-based Solar Powered Roof Water Sprinkle has good market demand as an invention provided towards overcoming the problem of the use of houses and wastage of energy in heating them is likely to be an important invention. In extreme cases, it can become an economy in terms of power use compared to constantly using the internal Air conditioning especially if the country's climate is hot power can be expensive or erratic. The demand for renewable energy, such as the use of solar power, provides for market satisfaction on the buyer's side as well as on the seller's side of the market. Evidently, the inclusion of a feature within the IoT provides added value as it does so through the smartphone app as a way of monitoring and controlling the connected systems and it would be appealing. There is always the possibility of constructing the system at a low cost using the cost of the materials used in developing the various parts of the system and the expansion of the parts of the system guarantees it can suit the small homes and the large business entities as well. Therefore, this system is in the right direction aligned with the current and future market requirements and standards to support a sound long-term affordable and easy-to-implement system best suited for commercialization of the business.

5.3 Future Works

For future improvements, accuracy of the Development of An IoT-based Solar Powered Roof Water Sprinkle results could be enhanced as follows:

i) Wider range of Enhanced Sensor Integration: A humidity and air quality sensor can be installed that will improve the function of the sprinkler in the environment.

ii) Improved Water Management: Total awareness of water recycling/ filtration demonstrates the need to continue researching water-saving technologies, the system can effectively control water consumption, as well as being as environmentally friendly as possible in the cooling process.

iii) User Feedback and Testing: Some of the important methods include the field trials and users' surveys which help to identify possible improvement and solve existing issues. It will facilitate a check on whether the system possesses practical functionality or not and if the application is capable of corresponding to real-life feedback and requirements.

iv) Scalability and Modular Design: During the testing, constructability, design, and selection making the system to be highly flexible and if possible portable shall enhance the applicability of the system on different forms and sizes of building and on different commercial and industrial buildings.

REFERENCES

[1] V. M. a. H. Z. D. T. Matt, Industry 4.0 for SMEs Challenges ,, Springer Nature Switzerland, 2020.

[2] S. U. K. R. Z. a. S. K. Rafiullah Khan, Future Internet: The Internet of Things Architecture, Possible Applications., Proceedings of Frontiers of Information Technology, 2012.

[3] S. U. K. R. Z. a. S. K. Rafiullah Khan, Future Internet: The Internet of Things Architecture, Possible Applications, Proceedings of Frontiers of Information Technology, 2012.

[4] L. T. Y. L. W. a. A. V. Feng Xia, Internet of Things, Wiley Online Library, 2012.

[5] INFSO D.4 Networked Enterprise & RFID INFSO G.2 Micro & Nanosystems,, Roadmap for the Future, 27 May 2008 .

[6] Y. Badamasi, "The working principle of an Arduino " in Electronics, Computer and Computation (ICECCO)," in *2014 11th International Conference*, 2014-Oct.

[7] P. D. M. AuthorHouse, "C Programming for the Pc the Mac and the Arduino Microcontroller System," 12 Nov 2013. [Online]. Available: https://books.google.com.my/books?hl=en&lr=&id=oigWAgAAQBAJ&oi=fnd&.

[8] R. P. ,. P. B. ,. M. S. Jaroslav Sobota, "Raspberry Pi and Arduino boards in control education'," 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667015340684.

[9] E. S. (. Co, "Espressif," 2020. [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/stable/esp32/about.html.

[10] E. Systems, "ESP32 Datasheet," 2022. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

[11] V. K. T. W. V. S. A. A. Shubham Chaudhari, "DHT 11 Sensor," *Real Time Temperature, Humidity Monitoring &Alert System,* vol. 8, no. Dr. DY Patil College of Engineering and Innovation, p. 5, 2023.

[12] V. A. V. V. N. A. R. K. B. Jana Divyashree, "A Comprehensive Study on Temperature and Humidity Sensor," *DHT11 Sensor,* no. D.R.R GOVERNMENT POLYTECHNIC, DAVANGERE, 2024.

[13] S. Patel, "Solar Panels for Off-grid Applications: A Review," *EEE Transactions on Sustainable Energy,* pp. 123-130, 2020.

[14] M. A. Green, Solar Cells: Operating Principles, Technology, and System Applications., University of New South Wales Press, 2017.

[15] A. Beaudet, "How Do I Read Solar Panel Specifications?," altE Store, 13 April 2016. [Online]. Available: https://www.altestore.com/blog/2016/04/how-do-i-read-specifications-of-my-solar-panel/.

[16] J. &. S. R. Boland, Electric Motors and Drives: Fundamentals, Types and Applications, Elsevier, 2018.

[17] S. Lee, "Challenges and Opportunities in Water Pump Motor Technology: A Review," *Journal of Mechanical Engineering Science,* pp. 112-120, 2021.

[18] X. Li, "Future Directions in Water Pump Motor Technology," *Renewable Energy,* pp. 214-226, 2019.

[19] I. Petrov, "Blynk: A Platform for Building IoT Applications," *IEEE Internet of Things Journal,* pp. 412-420, 2018.

[20] L. Wang, "Cloud-based Data Analytics for Blynk Apps," *IEEE Transactions on Industrial Informatics,* 2020.

[21] N. Adam, "Future Directions in Blynk Technology," *EEE Journal on Emerging and Selected Topics in Circuits and Systems,* pp. 214-226, 2019.

[22] Arduino.CC, "Docs Arduino," Arduino, 2024. [Online]. Available: https://docs.arduino.cc/hardware/uno-rev3/#features.

[23] "Solar Charge Controllers," BEIJING EPSOLAR TECHNOLOGY CO. LTD, [Online]. Available: https://www.epever.com/solar-charge-controllers/.