POLITEKNIK SULTAN SALAHUDDIN ABDUL AZIZ SHAH

SMART ENCLOSED SHOES RACK

NAME	**REGISTRATION NO**
BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN	08DEU20F2025

JABATAN KEJURUTERAAN ELEKTRIK

SESI 2 2021/2022

POLITEKNIK

SULTAN SALAHUDDIN ABDUL AZIZ SHAH

SMART ENCLOSED SHOES RACK

NAME	**REGISTRATION NO**
BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN	08DEU20F2025

This report submitted to the Electrical Engineering Department in fulfillment of the requirement for a Diploma in Electrical Engineering

JABATAN KEJURUTERAAN ELEKTRIK

SESI 2 2021/2022

CONFIRMATION OF THE PROJECT

The project report titled " SMART ENCLOSED SHOES RACK " has been submitted, reviewed and verified as a fulfills the conditions and requirements of the Project Writing as stipulated

Checked by:

Supervisor’s name : PUAN IRMA BAIZURI BINTI MOHD AKHIR

Supervisor’s signature:

Date :

Verified by:

Project Coordinator name :

Signature of Coordinator :

Date :

“I acknowledge this work is my own work except the excerpts I have already explained to our source”

1. Signature :

Name : **BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN**

Registration Number : **08DEU20F2025**

Date : 01/06/2023

DECLARATION OF ORIGINALITY AND OWNERSHIP

TITLE : SMART ENCLOSED SHOES RACK

SESSION: SESI 2 2022/2023

1. I, **BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN(08DEU20F2025)**

   is a final year student of **Diploma in Electrical Engineering, Department of Electrical, Politeknik Sultan Salahuddin Abdul Aziz Shah**, which is located at **Persiaran Usahawan,40140 Shah Alam Selangor Darul Ehsan**. (Hereinafter referred to as 'the Polytechnic').

2. I acknowledge that 'The Project above' and the intellectual property therein is the result of our original creation /creations without taking or impersonating any intellectual property from the other parties.

3. I agree to release the 'Project' intellectual property to 'The Polytechnics' to meet the requirements for awarding the **Diploma in Electrical Engineering** to me.

Made and in truth that is recognized by;
(**BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN**
(Identification card No: - 020821040332)

) .
....…………………………
) **BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN**

In front of me, **PUAN IRMA BAIZURI BINTI MOHD AKHIR** (810302-07-5396)
As a project supervisor, on the date:

) .....………………
) **PUAN IRMA BAIZURI BINTI MOHD AKHIR**

ACKNOWLEDGEMENTS

I have taken efforts in this Project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them. I am highly indebted to (PUAN IRMA BAIZURI BINTI MOHD AKHIR) for their guidance and constant supervision as well as for providing necessary information regarding the Project & also for their support in completing the Project.

I would like to express my gratitude towards my parents & member of (POLITEKNIK SULTAN SALAHUDDIN ABDUL AZIZ SHAH) for their kind co-operation and encouragement which help me in completion of this Project. I would like to express my special gratitude and thanks to industry persons for giving me such attention and time.

My thanks and appreciations also go to my colleague in developing the Project and people who have willingly helped me out with their abilities.

ABSTRACT

At this time, playing sports is a step taken by most people to continue to maintain health, it is because playing sports has also become a great trend as soon as it can help someone stay healthy because when we play sports there is a lot of sweat production even by walking for 30 minutes. Among the important criteria to be followed before, during and after. Among them is dressing and wearing appropriate shoes, most people will emphasize clothing, especially the right type of fabric, but many also do not know how to take care of a person's body, which is the skin. Especially the feet, shoes actually need appropriate care methods and are very important to ensure that the user is in good condition. The way to ensure shoes are in good condition is to use shoes deodorant and shoes sanitizer. As we know, shoes deodorant has the role of a pleasant fresh smell, while shoes sanitizer provides protection and maintenance of hygiene before someone uses the shoes. The idea of producing this product is to help users stay fragrant and have a layer of protection. Smart Enclosed Shoes Rack consists of ESP32 as a microcontroller. The ESP32 is a low-power chip (SoC) that has built-in WiFi and Bluetooth modules. It is a continuation of the popular ESP8266 chip. This product does not use the ESP8266 module because the ESP32 has better features and is also more stable.

Battery 18650 is also used with volt 2200mAh because they are lighter weight, very small battery memory effect, and can be shaped as desired. LiPo batteries have a higher energy density than other types of batteries. In other words, this battery can accommodate more energy with a lighter weight. And this battery can be recharged by using b3 compact charger.

Push button (yellow) in this product works to start in the program cycle. The buzzer plays a role with a beep sound when the liquid either shoes sanitizer or shoes deodorant wants to spray and a beep sound 2 times when the liquid stops. A switch acts as a device that can disconnect an electrical circuit, stop the flow of electrical current or change the direction of flow from one conductor to another. The function of the 2 channel relay module is as a connecting switch for two networks at once. Ultrasonic Sensor to measure the percentage of liquid in the Smart Enclosed Shoes Rack. Inside this Smart Enclosed Shoes Rack there is a place to hang shoes and under the shoe hanger a special space to place the product.

Key Word: ultrasonic, ESP32, buzzer

ABSTRAK

Pada masa ini, bersukan adalah satu langkah yang diambil oleh kebanyakan orang untuk terus menjaga kesihatan, kerana bersukan juga telah menjadi trend yang hebat sebaik sahaja ia dapat membantu seseorang untuk kekal sihat kerana apabila kita bersukan banyak peluh. pengeluaran walaupun dengan berjalan kaki selama 30 minit. Antara kriteria penting yang perlu dipatuhi sebelum, semasa dan selepas. Antaranya berpakaian dan memakai kasut yang sesuai, kebanyakan orang akan menitikberatkan pakaian terutama jenis fabrik yang betul, tetapi ramai juga yang tidak tahu menjaga tubuh badan seseorang iaitu kulit. Terutamanya kaki, kasut sebenarnya memerlukan kaedah penjagaan yang sesuai dan amat penting bagi memastikan pengguna berada dalam keadaan baik. Cara untuk memastikan kasut dalam keadaan baik adalah dengan menggunakan deodoran kasut dan pembersih kasut. Seperti yang kita tahu, deodoran kasut berperanan sebagai bau segar yang menyenangkan, manakala pembersih kasut memberikan perlindungan dan penyelenggaraan kebersihan sebelum seseorang menggunakan kasut. Idea menghasilkan produk ini adalah untuk membantu pengguna kekal wangi dan mempunyai lapisan perlindungan. Rak Kasut Tertutup Pintar terdiri daripada ESP32 sebagai mikropengawal. ESP32 ialah cip berkuasa rendah (SoC) yang mempunyai modul WiFi dan Bluetooth terbina dalam. Ia adalah kesinambungan cip ESP8266 yang popular. Produk ini tidak menggunakan modul ESP8266 kerana ESP32 mempunyai ciri yang lebih baik dan juga lebih stabil.

Bateri 18650 juga digunakan dengan volt 2200mAh kerana ia lebih ringan, kesan memori bateri yang sangat kecil, dan boleh dibentuk seperti yang dikehendaki. Bateri LiPo mempunyai ketumpatan tenaga yang lebih tinggi daripada jenis bateri lain. Dengan kata lain, bateri ini boleh menampung lebih banyak tenaga dengan berat yang lebih ringan. Dan bateri ini boleh dicas semula dengan menggunakan pengecas kompak b3.

Butang tekan (kuning) dalam produk ini berfungsi untuk bermula dalam kitaran program. Buzzer memainkan peranan dengan bunyi bip apabila cecair sama ada sanitizer kasut atau deodoran kasut mahu disembur dan bunyi bip 2 kali apabila cecair berhenti. Suis bertindak sebagai peranti yang boleh memutuskan litar elektrik, menghentikan aliran arus elektrik atau menukar arah aliran dari satu konduktor ke konduktor yang lain. Fungsi modul geganti 2 saluran adalah sebagai suis penyambung untuk dua rangkaian sekaligus. Penderia Ultrasonik untuk mengukur peratusan cecair dalam Rak Kasut Tertutup Pintar. Di dalam Rak Kasut Terlampir Pintar ini terdapat tempat untuk menggantung kasut dan di bawah penyangkut kasut ruang khas untuk meletakkan produk.

Kata Kunci: ultrasonik, ESP32, buzzer

TABLE OF CONTENTS

LIST OF TABLES

TABLE **TITLE** **PAGE**

LIST OF FIGURES

FIGURE **TITLE** **PAGE**

CHAPTER 1

1 INTRODUCTION

1.1 Introduction

Description This Smart Enclosed Shoes Rack is an innovation that involves two types of liquid, which is shoes sanitizer and shoes deodorant. The origin of this idea is because we have investigate many people don't have time to take care of shoes. Therefore, the shoes are not kept well and cause the smell and quality of the shoes to not be kept. So with this new product, there are no more problems. The advantage of this product is that when we are far away from it, we can still control it from a distance and know how much water there is. General knowledge, we are known for famous brands such as NIKE, ADIDAS, PUMA and so on for famous brands in Malaysia and abroad. We also know how meticulous the stitching is and the fabric used is also very premium from other brands. The price which almost reaches RM3k for someone who loves shoes or for an athlete is not a problem because of the interest. In addition, nowadays, society almost forgets the most important appearance, which is shoes. Especially in people who don't use shoes when leaving the house. So with the appearance of this product, it can highlight the importance of shoe care . For them, shoes are something that makes them happy every day and they will love shoes more than anything else. Shoes are also one of the main focuses for someone who takes care of appearance and for athletes they also need comfort when using them. Careful care will be taken of the shoes. For those who are addicted to these shoes, they don't care how much they have to pay for the equipment needed to wipe shoes, shoe sanitizer and shoe spray every time they use it. Therefore, they need a special bag that can be taken anywhere. But for this project, I applied it into a box. In this box it combines automatic dispenser, shoe deodorant & shoe sanitizer. This automatic dispenser will spray at the time we set, which is periodic. For example, 15 minutes or 30 minutes for one spray involving shoe deodorant & shoe sanitizer. This product, I have combined it with the use of IoT which will be able to identify when the liquid in the shelf has reached the minimum level, it will connect through an application that has been connected through a mobile phone. With this very smart shoe rack, it will no longer be a problem for them to care for these shoes .

1.2 Background Research

The liquid used is an alcoholic liquid. But it does not cause air pollution. This is because this liquid will be sprayed periodically and not in an open place. So it is an environmentally friendly product. This product uses a microcontroller , ESP32. ESP32 is an open source microcontroller / single board microcontroller and is one of the most popular Open Source Hardware projects. Designed to facilitate the use of electronics in various fields. The hardware has an Atmel AVR processor .

1.3 Problem Statement

- Don't have much time for shoe care .
- The shoes worn by athletes will become rotten and have a lot of bacteria if they are not taken care of properly .
- Not having a place of care ; if send it to the store, it will involve a high cost

1.4 Research Objectives

1. To develop a new automatic product with two liquid which is shoes deodorant and shoes sanitizer .
2. To reduce the movement of users in ensuring the quantity of shoes deodorant and shoes sanitizer when the data limit can be issued through the application on the phone as a sign of shortage .
3. To innovate the shoes sanitizer and shoes deodorant in shoes rack to keep shoes in good condition .

1.5 Scope of Research

1. Ultrasonic Sensor
    - Investigating liquid level , percentage of liquid
2. Place:
    - Home
    - Office
3. Target user:
    - All level age

The appearance of this product are link between the application Blynk on product and the phone . In this project i am using ESP32 as microcontroller for my project which will process the data from input and control the output . For example when the liquid (shoes deodorant and shoes sanitizer) already reach the minimum level, the data will connect to the Blynk application which is on the phone . Then it will show the percentage of the liquid in the Smart Enclosed Shoes Rack .

1.6 Project Significance

The origin of this idea is after I did a study on some PSA athletes about the problems they face after doing sports activities. And after doing research, I also have the same problem which is not enough long time to do shoe care, and shoe care is not organized. So as a result of this project, I have studied from several sides from the point of view of the suitability of the place, the speed of using it and the effectiveness of this product on shoes. I have used LIPO batteries because they are easy to recharge. The ultrasonic sensor used is to see what percentage of the liquid is in the special chamber of this product through a smart phone only(Blynk) . The use of a buzzer in my project is also very necessary because it works for notifications when liquids are sprayed periodically (shoes sanitizer & shoes deodorant). The shoe rack that I produced is also complete and comes with a little ventilation space because it doesn't want to steam. I have done this project from the beginning of the first week of lectures until the tenth week of lectures. This product has been helped by my father and mother and my friend Puteri Khairul Bariyah Binti Mohd Sulaiman. I am very grateful to them for their support.

1.7 Chapter Summary

An introducing a project involving 2 types of liquid which is shoe sanitizer and shoe deodorant in a special shoes rack for shoe care. If the liquid is reduced, then there will be a massage delivery via smartphone using apps Blynk .

CHAPTER 2

2 LITERATURE REVIEW

2.1 Introduction

This chapter reviews existing project created to get an idea about the project design, conception and any information that related to improve the project. This chapter also explains and discuss about source or article that related to the project. It is consist of the products that have been appeared in the market nowadays. This chapter is also contained the theory of the components, equipment and programming that is used in the project.

2.2 Automatic Sanitizer Dispensing Machine (Literature Review Topic 1)

The main objective of this paper is an ultrasonic sensor senses the hand placed near it, the ESP32 is used as a microcontroller, which senses the distance and the result is the pump running to pump out the hand sanitizer.

2.2.1 Previous Research (Subtopic Literature Review Topic 1)

ESP32 microcontroller is used since it is easy to program, has inbuilt ADC, DAC. The input to the Arduino is given using an ultrasonic sensor ,which is used to sense the distance, it emits ultrasonic frequency from one side and the notes the time taken by sound wave to get reflected back. When the sensor senses the hand, at a distance less than 7cm from the sensor, the ESP32 gives a 100ms pulse from its digital output pin. The pump cannot be used directly, hence a relay is used as a switch. The relay accepts the pulse from ESP32 and makes the pump run. The pump is 3 to 12V submersible type ,which pumps out a few drops of hand sanitizer onto the hands, after pumping, the distance is sensed for every 1000ms (1s) for scanning purposes.

2.3 Control System (Literature Review Topic 2)

A control system is a set of mechanical or electronic devices that regulates other devices or systems by way of control loops. Typically, control systems are computerized. Control systems are a central part of production and distribution in many industries. Automation technology plays a big role in these systems. A **control system** manages, commands, directs, or regulates the behavior of other devices or systems using control loops. It can range from a single home heating controller using a thermostat controlling a domestic boiler to large industrial control systems which are used for controlling processes or machines. The control systems are designed via control engineering process. For continuously modulated control, a feedback controller is used to automatically control a process or operation. The control system compares the value or status of the process variable (PV) being controlled with the desired value or setpoint (SP), and applies the difference as a control signal to bring the process variable output of the plant to the same value as the setpoint.
For sequential and combinational logic, software logic, such as in a programmable logic controller, is used.

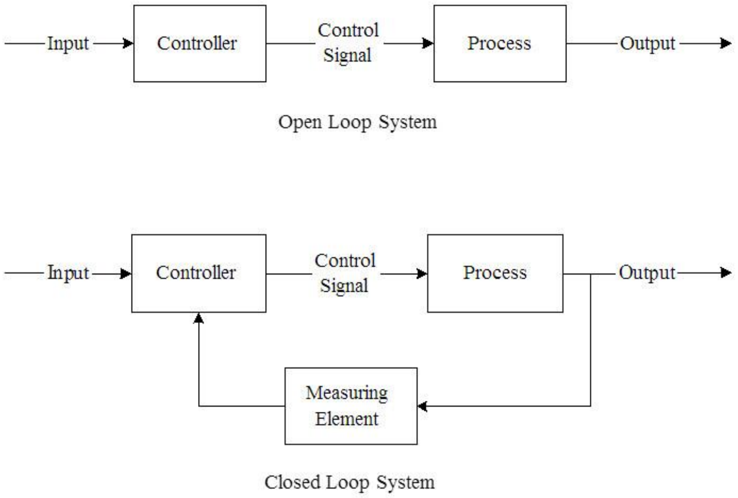


Figure 2.1: Block diagram of open loop and closed loop system

2.3.1 Microcontroller

A microcontroller is an integrated circuit (IC) device used for controlling other portions of an electronic system, usually via a microprocessor unit (MPU), memory, and some peripherals. These devices are optimized for embedded applications that require both processing functionality and agile, responsive interaction with digital, analog, or electromechanical components. The most common way to refer to this category of integrated circuits is “microcontroller" but the abbreviation “MCU” is used interchangeably as it stands for “microcontroller unit”. You may also occasionally see “µC” (where the Greek letter mu replaces “micro”). “Microcontroller” is a well-chosen name because it emphasizes defining characteristics of this product category. The prefix “micro” implies smallness and the term "controller" here implies an enhanced ability to perform control functions. As stated above, this functionality is the result of combining a digital processor and digital memory with additional hardware that is specifically designed to help the microcontroller interact with other components

2.3.2 Programmable Logic Control (PLC)

A programmable logic controller is a type of tiny computer that can receive data through its inputs and send operating instructions through its outputs. Fundamentally, a PLC’s job is to control a system’s functions using the internal logic programmed into it. Businesses around the world use PLCs to automate their most important processes. A PLC takes in inputs, whether from automated data capture points or from human input points such as switches or buttons. Based on its programming, the PLC then decides whether or not to change the output. A PLC’s outputs can control a huge variety of equipment, including motors, solenoid valves, lights, switchgear, safety shut-offs and many others. The physical location of PLCs can vary widely from one system to another. Usually, however, PLCs are located in the general vicinity of the systems they operate, and they’re typically protected by a surface mount electrical box. Skip to the end if you’re interested in seeing the electrical junction boxes that help protect PLCs. PLCs largely replaced the manual relay-based control systems that were common in older industrial facilities. Relay systems are complex and prone to failure and, in the 1960s, the inventor Richard Morley introduced the first PLCs as an alternative. Manufacturers quickly realized the potential of PLCs and began integrating them into their work processes. Today, PLCs are still a fundamental element of many industrial control systems. In fact, they’re still the most used industrial control technology worldwide. The ability to work with PLCs is a required skill for many different professions, from the engineers designing the system to the electrical technicians maintaining it .

2.3.3 Arduino

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board. The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board -- you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package.

2.4 Chapter Summary

This chapter discusses the literature reviews of five journals that I discovered to be relevant to this study. One Atmel Atmega328p, one microcontroller, and the other Arduino were used from this five journals

CHAPTER 3

3 RESEARCH METHODOLOGY

3.1 Introduction

A very careful plan is being implemented to realize this project as a ready-to-use product with security features. To ensure timely completion of the Project, a step-by-step process is followed. This product is also an environmentally friendly product because the liquid used is only dispersed in a closed room.

3.2 Project Design and Overview.

Smart Enclosed Shoes Rack consists of ESP32 as a microcontroller. The ESP32 is a low-power chip (SoC) that has built-in WiFi and Bluetooth modules. It is a continuation of the popular ESP8266 chip. This product does not use the ESP8266 module because the ESP32 has better features and is also more stable. Battery 18650 is also used with volt 2200mAh because they are lighter weight, very small battery memory effect, and can be shaped as desired. LiPo batteries have a higher energy density than other types of batteries. In other words, this battery can accommodate more energy with a lighter weight. And this battery can be recharged by using b3 compact charger. Push button (yellow) in this product works to start in the program cycle. The buzzer plays a role with a beep sound when the liquid either shoes sanitizer or shoes deodorant wants to spray and a beep sound 2 times when the liquid stops. A switch acts as a device that can disconnect an electrical circuit, stop the flow of electrical current or change the direction of flow from one conductor to another. The function of the 2 channel relay module is as a connecting switch for two networks at once. Ultrasonic Sensor to measure the percentage of liquid in the Smart Enclosed Shoes Rack. A step-down transformer converts high voltage with low current into low voltage with high current. The main function of the step down transformer is to lower the voltage and adjust it to the electrical capacity. Inside this

Smart Enclosed Shoes Rack there is a place to hang shoes and under the shoe hanger a special space to place the product.

3.2.1 Block Diagram of the Project

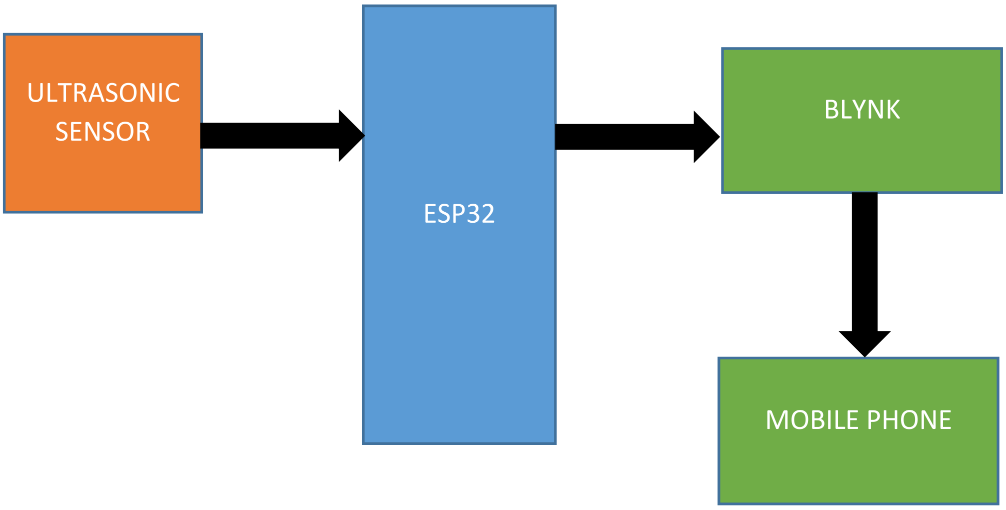


Figure 3.2.1: Flow chart of operation of the system

3.2.2 Flowchart of the Project 2

Figure 3.2 shows the circuit diagram of the whole system. It is show that Flowchart of the Smart Enclosed Shoes Rack

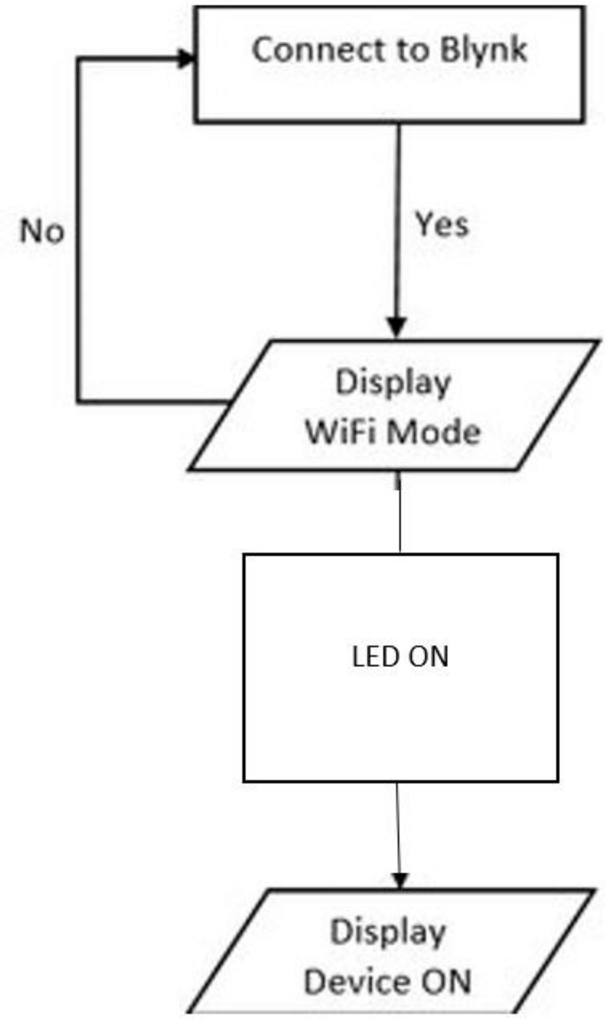


Figure 3.2: Flow chart of operation of the system

*Images may be subject to copyright

3.2.3 Project Description

Inside a shelf that combines an automatic dispenser, shoe deodorant & shoe sanitizer automatic to make a spray. every 15 or 30 minutes will release a periodic spray between shoe deodorant and shoe sanitizer. If the deodorant and shoe sanitizer has reached the minimum level, it will send information to the application that has been connected to the mobile phone. This rack aims to make it easier for users to take care of their shoes.

3.3 Project Hardware

As mentioned in the above chapter, the controller was designed using ESP32. This microcontroller recognizes live stills, reports results and manages push buttons, level controls and the entire project workflow. The entire process is managed by an ESP32 Microcontroller.

3.3.1 Schematic Circuit

Figure 3.3 shows the overall circuit diagram of this Project of Smart Enclosed Shoes Rack

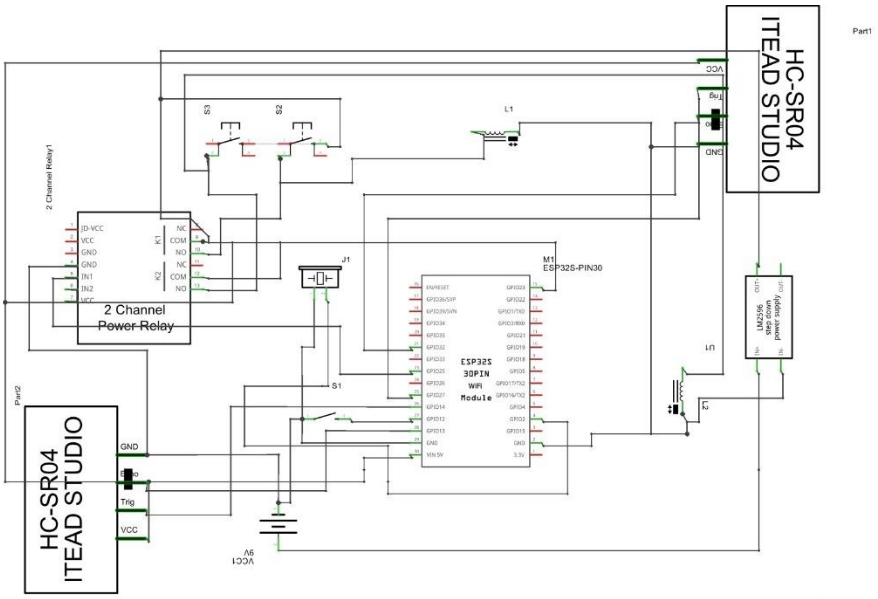


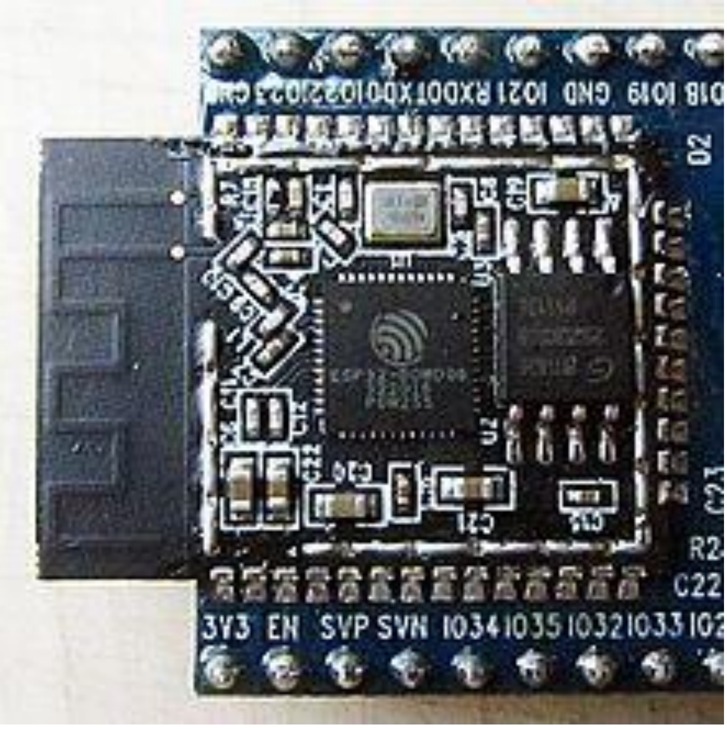
Figure 3.3: Circuit Diagram

*Images may be subject to copyright

3.3.2 Description of Main Component

3.3.2.1 Component 1 (ESP32)

ESP32 is a series of low-cost, low-power system on a chip microcontrollers with integrated Wi-Fi and dual-mode Bluetooth. The ESP32 series employs either a Tensilica Xtensa LX6 microprocessor in both dual-core and single-core variations, Xtensa LX7 dual-core microprocessor or a single-core RISC-V microprocessor and includes built-in antenna switches, RF balun, power amplifier, low-noise receive amplifier, filters, and power-management modules. ESP32 is created and developed by Espressif Systems, a Shanghai-based Chinese company, and is manufactured by TSMC using their 40 nm process. It is a successor to the ESP8266 microcontroller.



3.3.2.2 Component 2 (2 CHANNEL RELAY)

The 2 Channels Relay Module is a convenient board which can be used to control high. voltage, high current load such as motor, solenoid valves, lamps and AC load. It is designed to interface with microcontroller such as Arduino, PIC and etc. The relays terminal (COM, NO and NC) is being brought out with screw terminal. It also comes with a LED to indicate the status of relay.

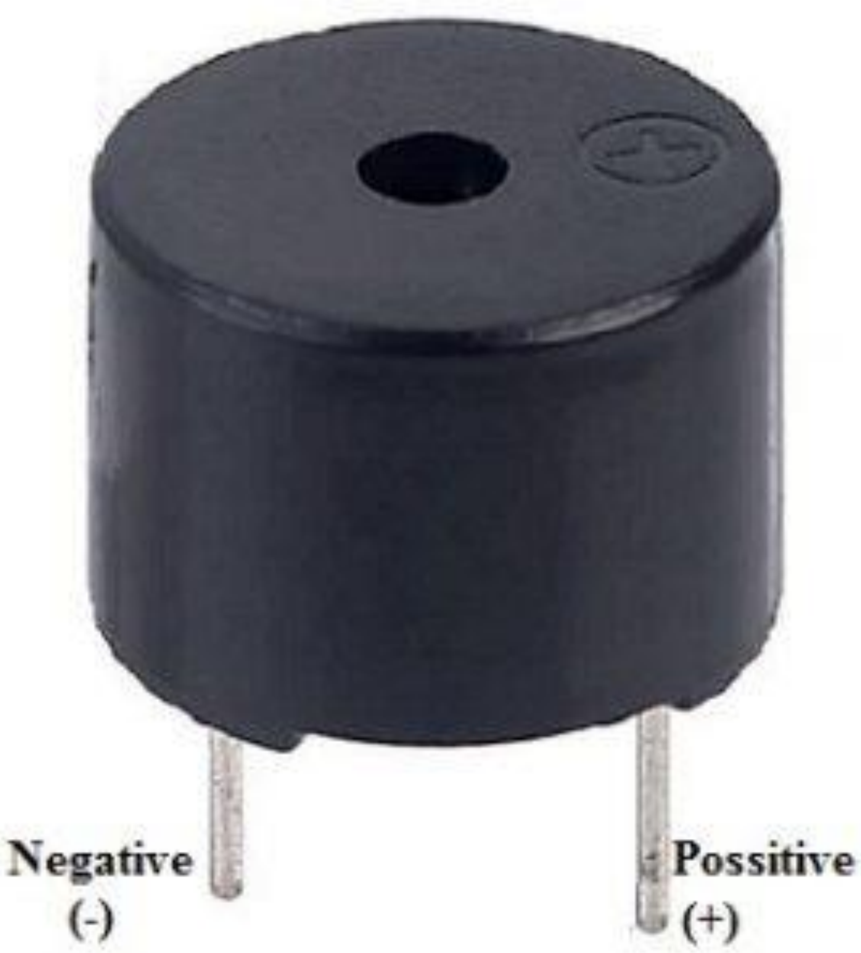
3.3.2.3 Component 3 (BATTERY 18650 LiPo)

An 18650 battery is a type of lithium-ion rechargeable battery. The numbers "18650" refer to the battery's dimensions: it is 18mm in diameter and 65mm in length. 18650 batteries are commonly used in electronic devices such as laptops and flashlights, as well as in electric vehicles and other high-power applications. They are known for their high energy density, long lifespan, and relatively low self-discharge rate.

Some types of 18650 have been modified adding either a button top and/or internal protection circuit. This can increase the physical length of an “18650” battery from 65mm to 70mm or in certain cases even longer. If you're using an 18650 battery for a consumer product, you should always verify with that product's manufacturer for the exact specifications required to power the device.

3.3.2.4 Component 4 (BUZZER)

An audio signaling device like a beeper or buzzer may be electromechanical or piezoelectric or mechanical type. The main function of this is to convert the signal from audio to sound. Generally, it is powered through DC voltage and used in timers, alarm devices, printers, alarms, computers, etc. Based on the various designs, it can generate different sounds like alarm, music, bell & siren. The pin configuration of the buzzer is shown below. It includes two pins namely positive and negative. The positive terminal of this is represented with the '+' symbol or a longer terminal. This terminal is powered through 6Volts whereas the negative terminal is represented with the '-'symbol or short terminal and it is connected to the GND terminal.



3.3.2.5 Component 5 (SWITCH ON/OFF)

A control is defined as an **on-off switch** when its function is to open or close an electrical circuit in a stable manner. If the closing or opening occurs in a non-stable or momentary manner, we are talking about a momentary on-off switch or push-button on-off switch, more briefly called on-off push-button.
The **push-button** is used to **open or close a specific electrical circuit in a momentary** or non-stable manner. This means that opening or closing occurs as long as pressure is maintained on the button in question. In the case of the on-off switch, the button with the I/O marking is used to switch on or off any equipment permanently, until a new command is given.

Specifically, the bipolar switch has the positions open and closed, while the push-button can be NO, normally open, or NC, normally closed. Push-buttons are defined as normally open when the application requires the button to be pressed to close the circuit. For example in work tools, such as a drill, which switches off as soon as the button is released. Normally closed is when, conversely, the application requires the button to be pressed to open the circuit. Just think of those applications with continuous operation that have a button that must be held down for the time interval in which you want to interrupt its operation.

3.3.2.6 Component 6 (ULTRASONIC SENSOR)

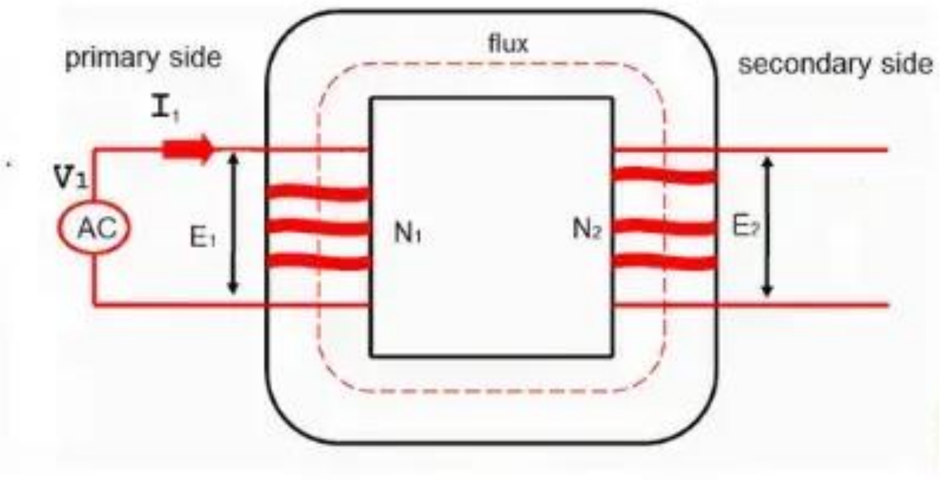
Ultrasonic sensors are used primarily as proximity sensors. They can be found in automobile self-parking technology and anti-collision safety systems. Ultrasonic sensors are also used in robotic obstacle detection systems, as well as manufacturing technology. In comparison to infrared (IR) sensors in proximity sensing applications, ultrasonic sensors are not as susceptible to interference of smoke, gas, and other airborne particles (though the physical components are still affected by variables such as heat).

Ultrasonic sensors are also used as level sensors to detect, monitor, and regulate liquid levels in closed containers (such as vats in chemical factories). Most notably, ultrasonic technology has enabled the medical industry to produce images of internal organs, identify tumors, and ensure the health of babies in the womb.

3.3.2.7 Component 7 (STEP DOWN TRANSFORMER)

A **step-down transformer** is a type of transformer that converts the high voltage (HV) and low current from the primary side of the transformer to the low voltage (LV) and high current value on the secondary side of the transformer. The reverse of this is known as a step up transformer.

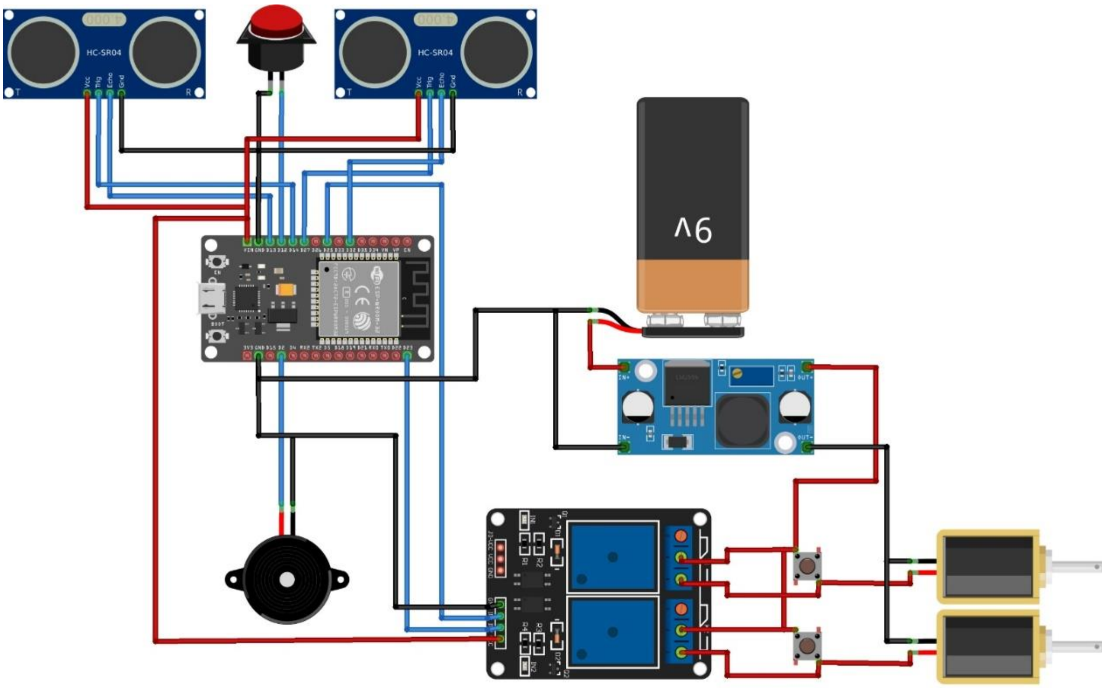
A transformer is a type of static electrical equipment that transforms electrical energy (from primary side windings) to magnetic energy (in transformer magnetic core) and again to the electrical energy (on the secondary transformer side). A step-down transformer has a wide variety of applications in electrical systems and transmission lines.



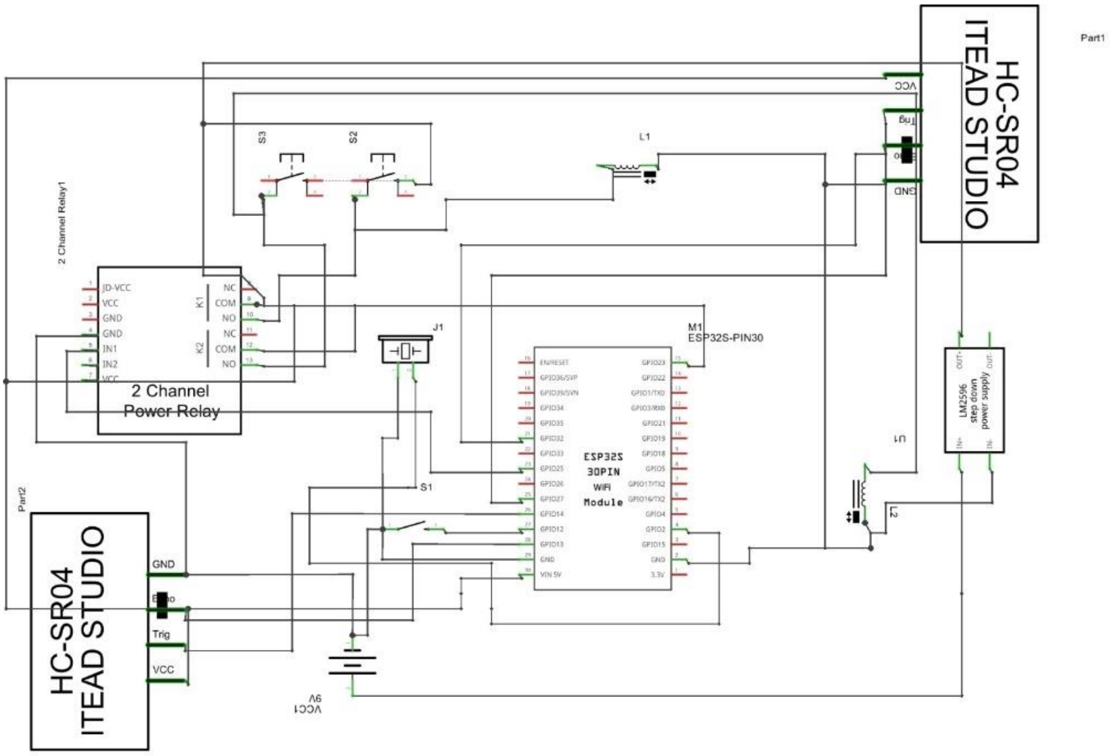
3.3.2.8 (PUSH BUTTON)

A push-button (also spelled pushbutton) or simply button is a simple switch mechanism to control some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal.[1] The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed. Buttons are most often biased switches, although many un-biased buttons (due to their physical nature) still require a spring to return to their un-pushed state.

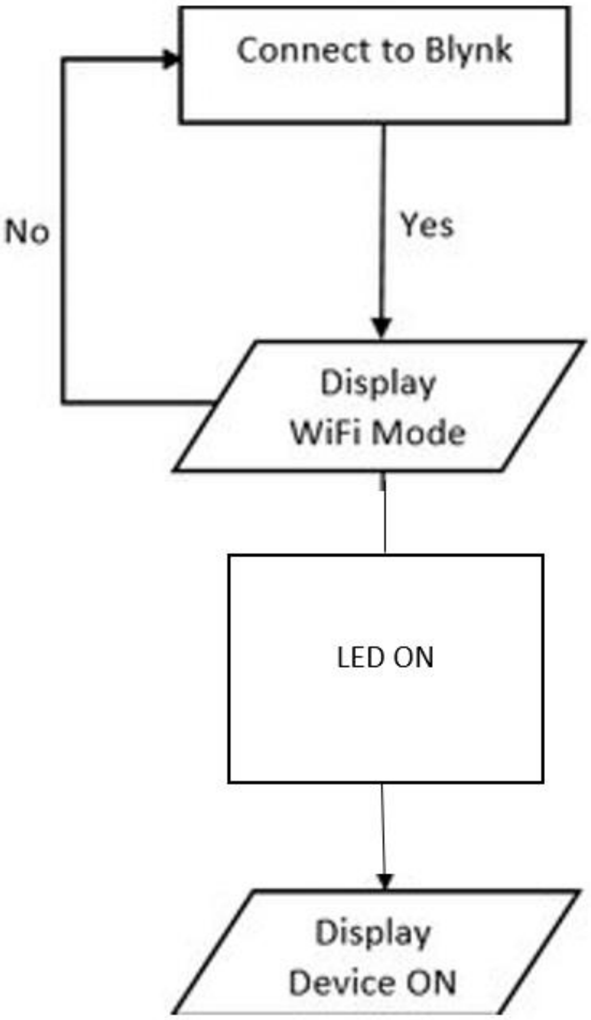
3.3.3 Circuit Operation



3.4 Project Software



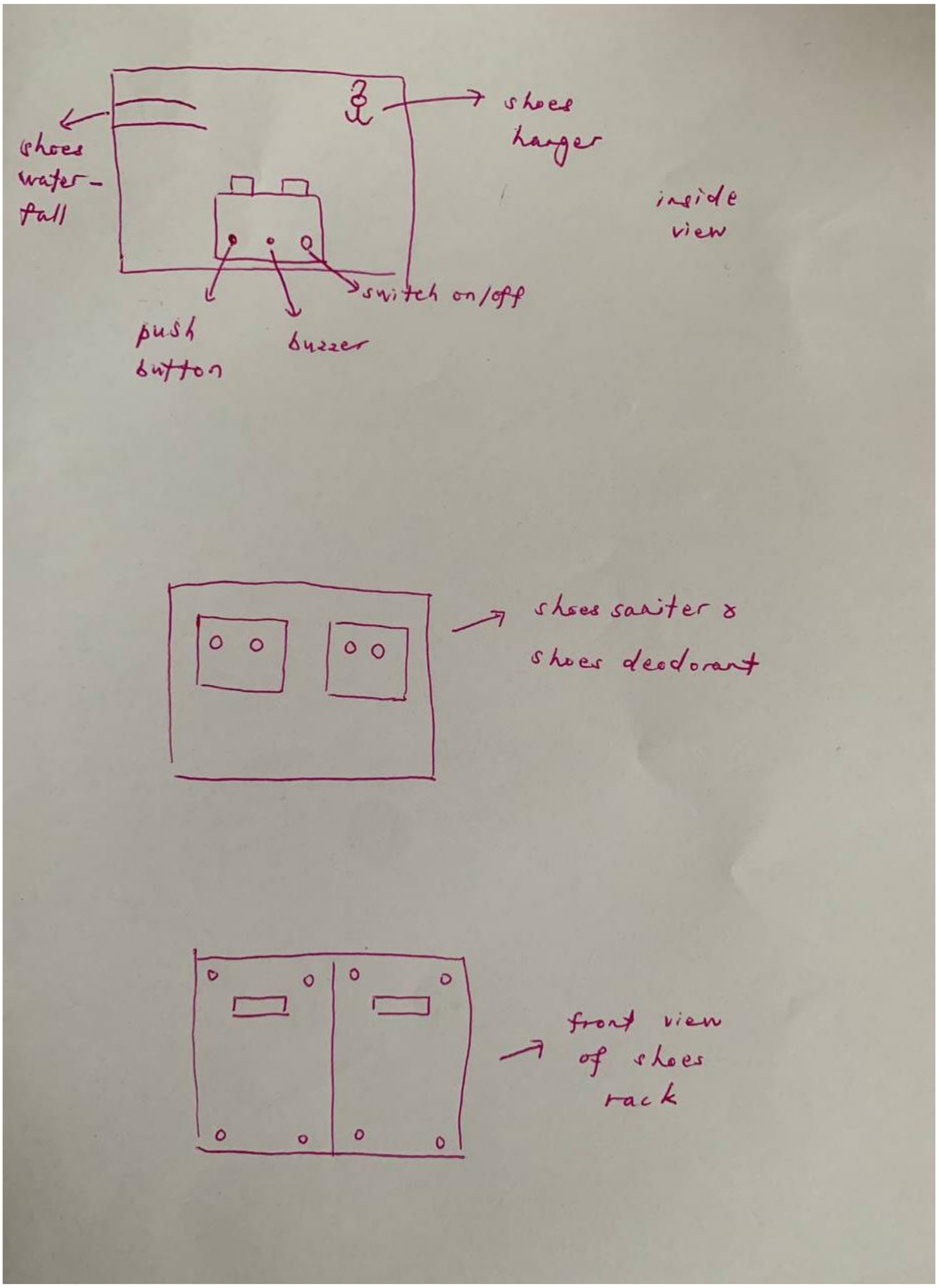
3.4.1 Flowchart of the System



3.4.2 Description of Flowchart

We must connect the wifi from the smartphone to the project, when the buzzer sounds, the sign has been successfully connected. And the process will run according to the coding we set. If it cannot be connected to wifi, then the program will not run successfully.

3.5 Prototype Development



3.5.1 Mechanical Design/Product Layout

Figure 3.4 shows the design of the product Smart Enclosed Shoes Rack

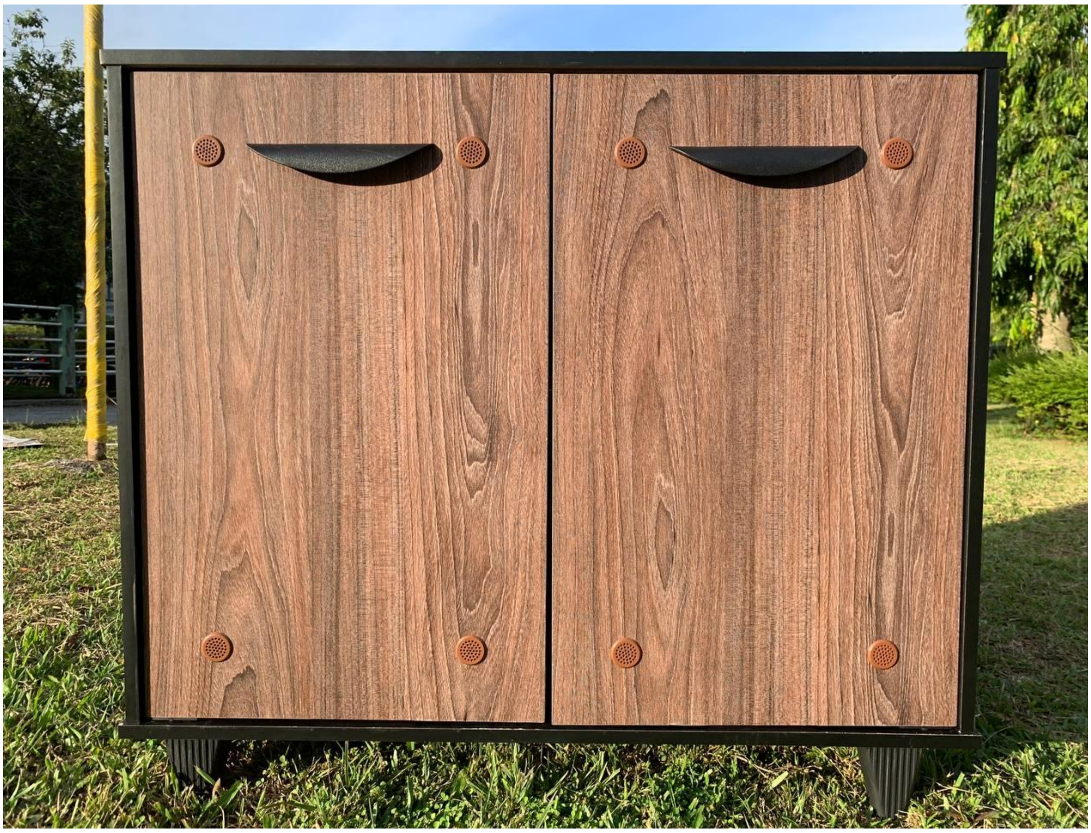


Figure 3.4: Front view of the Project

3.6 Sustainability Element in The Design Concept

In this project, I did not use a spray that releases a lot of water because it will cause damage to the shoes due to excessive use of shoes deodorant or shoes sanitizer. And it will not be a pollution to the air because the shoe cleaning process will take place in a closed space which is in this Smart Enclosed Shoes Rack.

3.7 Chapter Summary

The project's research methodology is covered in this chapter. I've included the project's block diagram, project description, hardware, circuit diagram, component description, circuit operation, project flowchart, and flowchart description in this study technique.

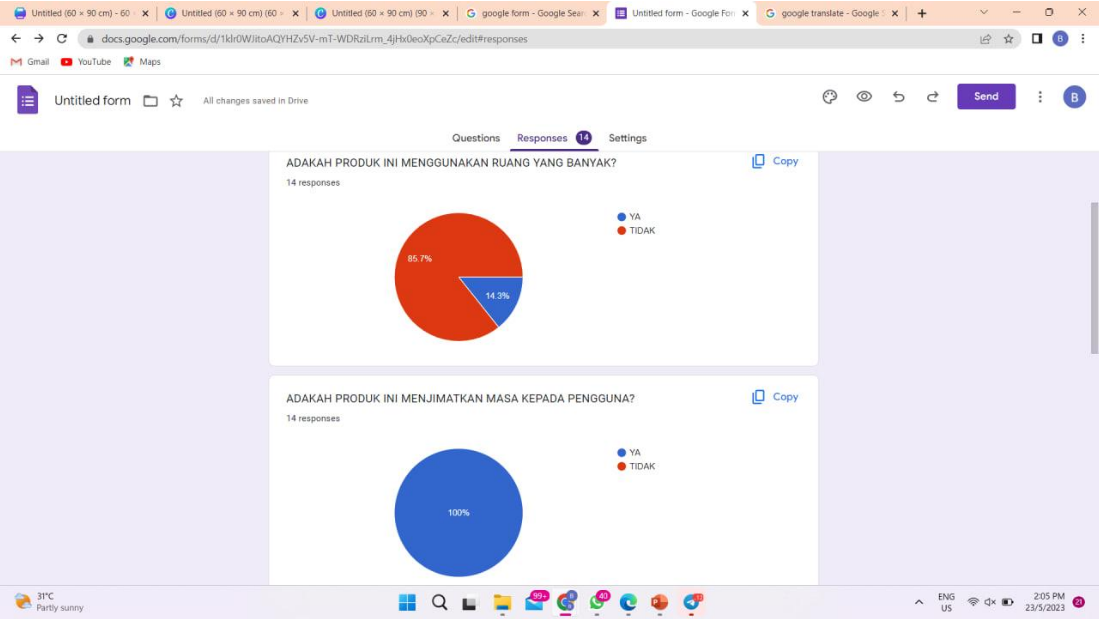
CHAPTER 4

4 RESULTS AND DISCUSSION

4.1 Introduction

The steps taken in the observation to examine every point of view from the user have already been implemented. The next step is to place the product in one place for a period of time and ensure that users give feedback on their improvement or acceptance of this Smart Automatic Handwash. It is digitally recorded by creating a google form that is entered by scanning a QR code.

4.2 Results and Analysis



Untitled form

All changes saved in Drive

Questions Responses 14 Settings

ADAKAH PRODUK INI MENGGUNAKAN RUANG YANG BANYAK?

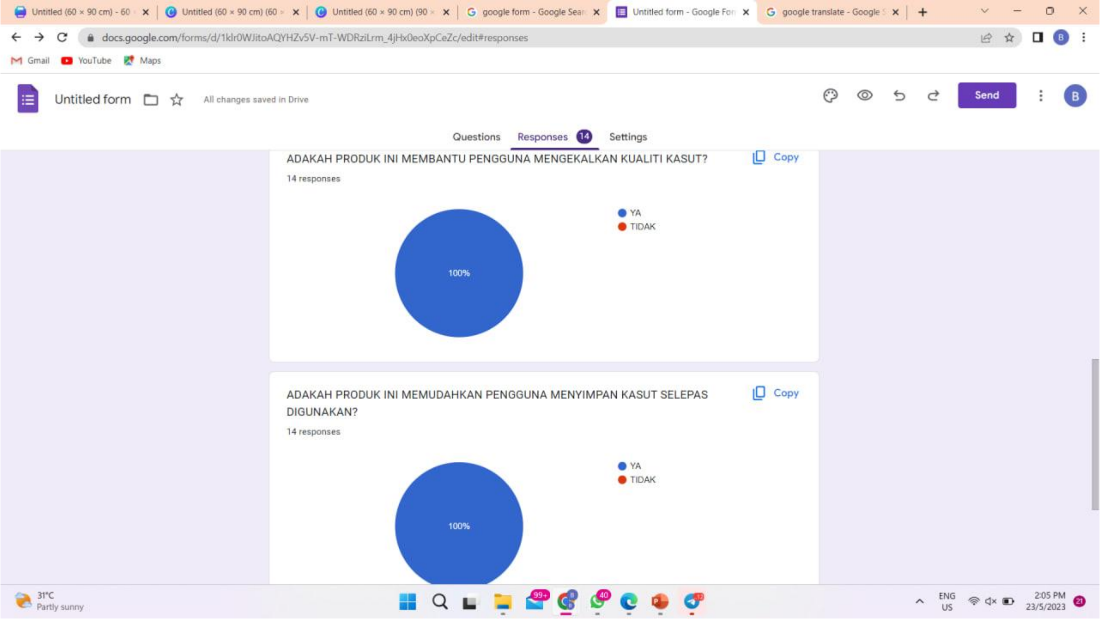
14 responses

Copy
YA
TIDAK
85.7%
14.3%

ADAKAH PRODUK INI MENJIMATKAN MASA KEPADA PENGGUNA?

14 responses

Copy
YA
TIDAK
100%
docs.google.com/forms/d/1klr0WJitoAQYHZv5V-mT-WDRziLrm_4jHx0eoXpCeZc/edit#responses
Send
31°C
Partly sunny
ENG
US
2:05 PM
23/5/2023

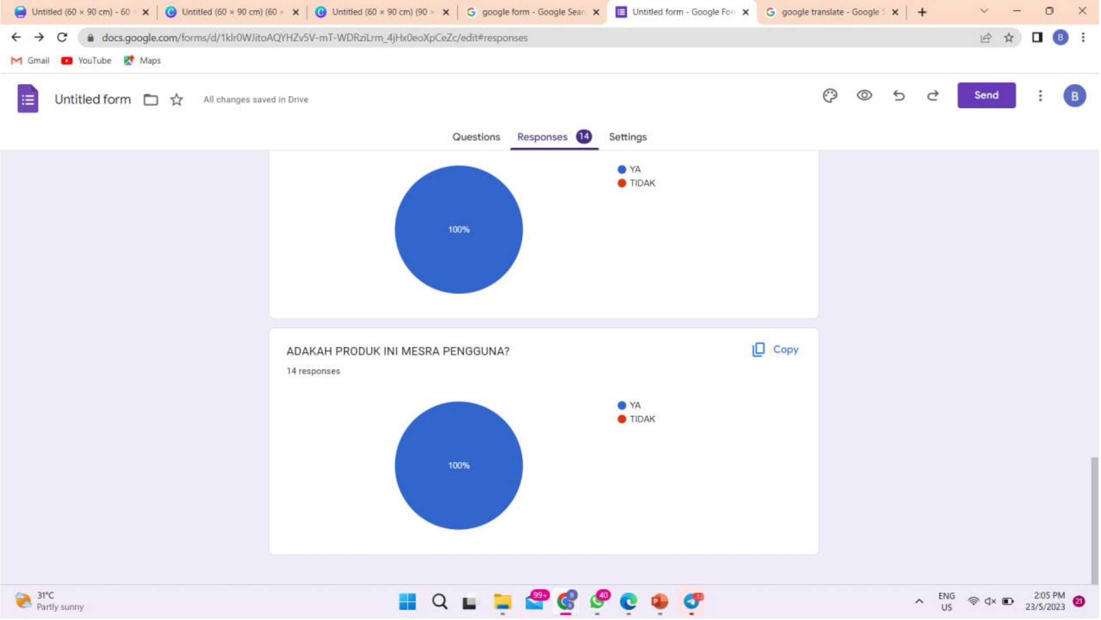


ADAKAH PRODUK INI MEMBANTU PENGGUNA MENGEKALKAN KUALITI KASUT?

14 responses

ADAKAH PRODUK INI MEMUDAHKAN PENGGUNA MENYIMPAN KASUT SELEPAS DIGUNAKAN?

14 responses



ADAKAH PRODUK INI MESRA PENGGUNA?

14 responses

4.3 Discussion

All respondents gave different feedback. From the results, some steps can be taken as a way to improve in the future. Each of them gave feedback after trying to use the Smart Enclosed Shoes Rack product. Smart Enclosed Shoes Rack is a new tool that wants to be given full support to ensure that the product can get a high marketability scope for the common good. Considering that more and more infectious diseases involve contact, in this way the responsible party may help reduce the spread and help the special group by making it easier for them to stay hygienic without needing a lot of movement and will reduce the risk of harm.

4.4 Chapter Summary

The project's results and discussion is covered in this chapter. I've included the project's introduction, results and analysis, and discussion.

CHAPTER 5

5 CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

All studies play an important role to produce a conclusion, before the conclusion and discussion. it is the result of observation from each part to examine the available conclusions. For example, all products have been designed but all of them are isolated with their own functions. There is no special combination as completely designed as the Smart Automatic Handwash that has been created.

5.2 Conclusion

This Smart Enclosed Shoes Rack has been innovated from two types of liquid which are shoes sanitizer and shoes deodorant. This innovation is the result of not having enough time for shoe care. So this product is the most appropriate to make a choice. And no more problems can be caused because two types of liquid have been combined in one space. Just put/hang the shoes in the rack and the program will run.

5.3 Suggestion for Future Work

I assure that this Smart Enclosed Shoes Rack will be a product that will be loved by users. This is because, this product does not require a long time and it will save the user's time. And this product can enter more than 3 shoes and will sanitize and deodorize all shoes at the same time. This product is also a user-friendly and environmentally friendly product. This product can also be used by all ages.

5.4 Chapter Summary

The project's conclusion and recommendation is covered in this chapter. I've included the project's introduction, conclusion and suggestion for future work.

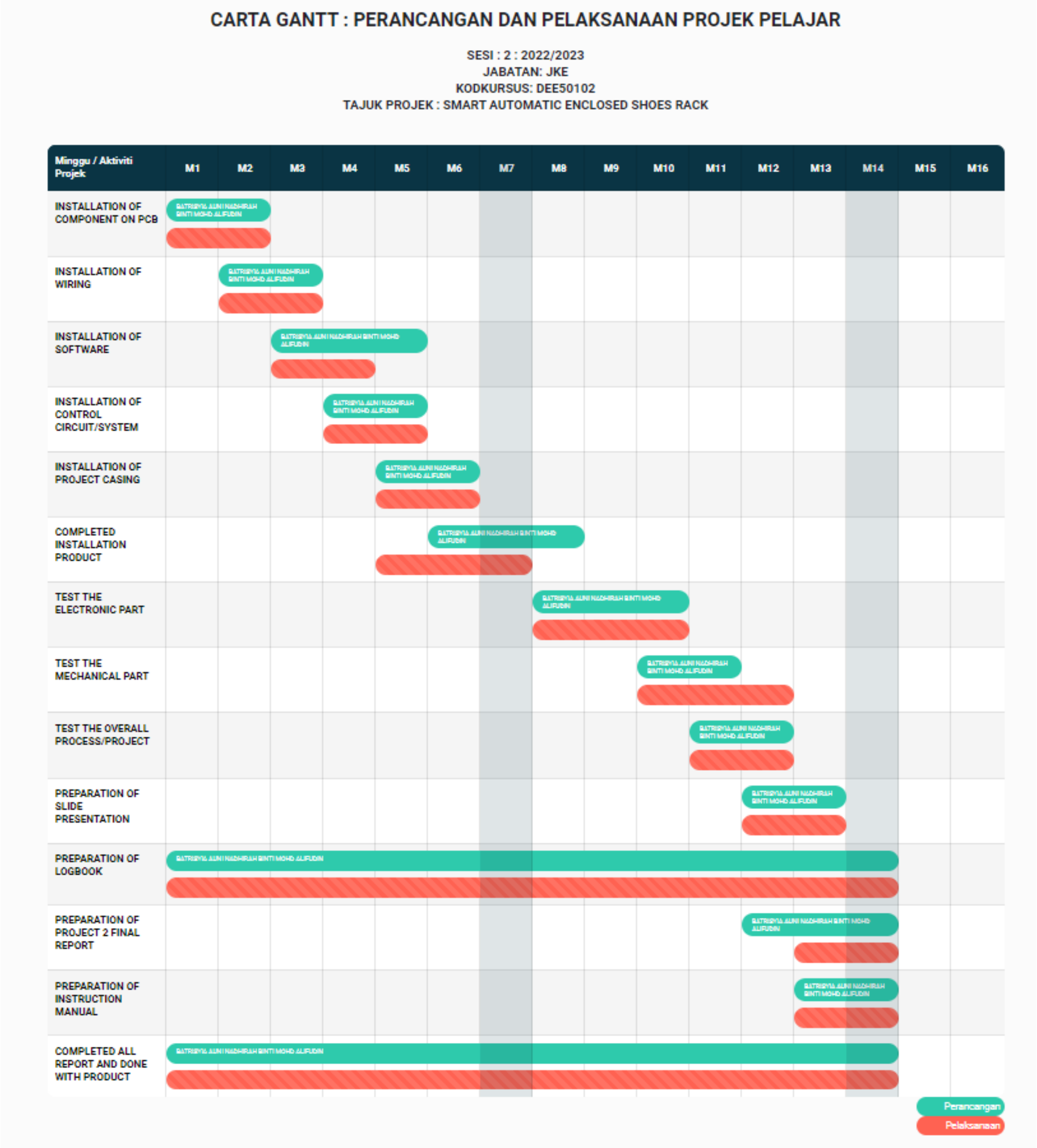
CHAPTER 6

6 PROJECT MANAGEMENT AND COSTING

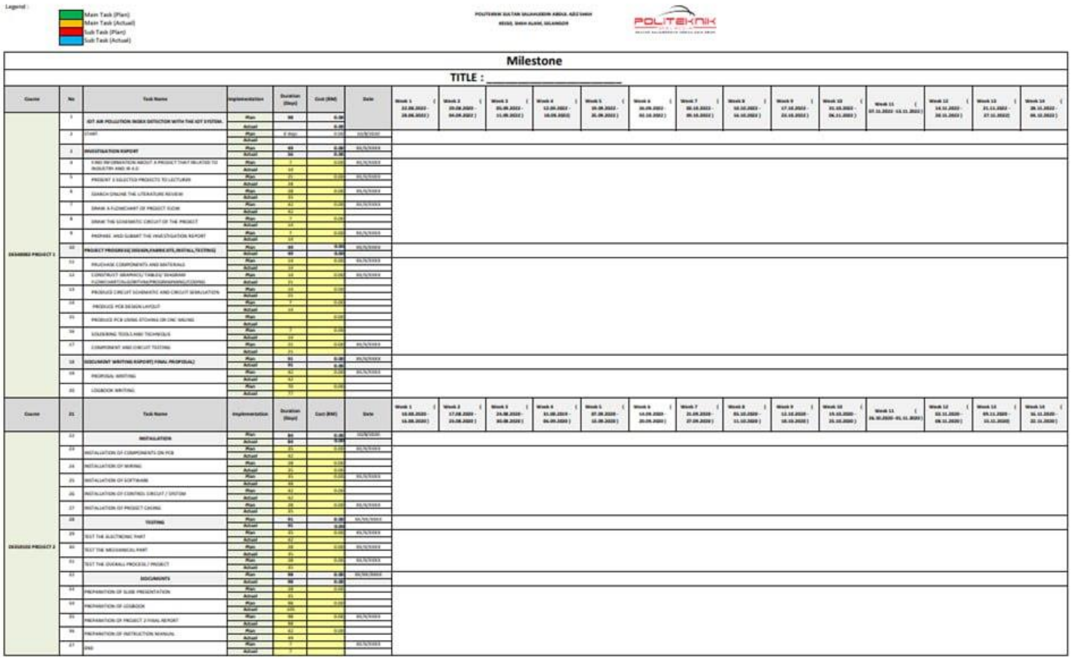
6.1 Introduction

Because we don't have a sponsor, we had to use our own money to buy the majority of the essential components and supplies for the project. The estimated price is RM1000. The development costs are still manageable for the upcoming months. The research indicates that it is doable and attainable.

6.2 Gant Chart and Activities of the Project



6.3 Milestone



Legend :

🟩 Main Task (Plan)
🟨 Main Task (Actual)
🟥 Sub Task (Plan)
🟦 Sub Task (Actual)

POLITEKNIK SULTAN SALAHUDDIN ABDUL AZIZ SHAH
40150, SHAH ALAM, SELANGOR

Milestone

TITLE : ____________________

Course	No	Task Name	Implementation	Duration (Days)	Cost (RM)	Date	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14
[illegible] PROJECT 1	1	IOT AIR POLLUTION INDEX DETECTOR WITH THE IOT SYSTEM.	Plan Actual	[illegible]	[illegible]															
	2	START	Plan Actual	[illegible]	[illegible]	[illegible]														
	3	INVESTIGATION REPORT	Plan Actual	[illegible]	[illegible]	[illegible]														
	4	FIND INFORMATION ABOUT A PROJECT THAT RELATED TO INDUSTRY AND IR 4.0	Plan Actual	[illegible]	[illegible]	[illegible]														
	5	PRESENT 3 SELECTED PROJECTS TO LECTURER	Plan Actual	[illegible]	[illegible]	[illegible]														
	6	SEARCH ONLINE THE LITERATURE REVIEW	Plan Actual	[illegible]	[illegible]	[illegible]														
	7	DRAW A FLOWCHART OF PROJECT FLOW	Plan Actual	[illegible]	[illegible]	[illegible]														
	8	DRAW THE SCHEMATIC CIRCUIT OF THE PROJECT	Plan Actual	[illegible]	[illegible]															
	9	PREPARE AND SUBMIT THE INVESTIGATION REPORT	Plan Actual	[illegible]	[illegible]	[illegible]														
	10	PROJECT PROGRESS(DESIGN,FABRICATE,INSTALL,TESTING)	Plan Actual	[illegible]	[illegible]	[illegible]														
	11	PURCHASE COMPONENTS AND MATERIALS	Plan Actual	[illegible]	[illegible]	[illegible]														
	12	CONSTRUCT GRAPHICS/ TABLES/ DIAGRAM FLOWCHART/ALGORITHM/PROGRAMMING/CODING	Plan Actual	[illegible]	[illegible]	[illegible]														
	13	PRODUCE CIRCUIT SCHEMATIC AND CIRCUIT SIMULATION	Plan Actual	[illegible]	[illegible]															
	14	PRODUCE PCB DESIGN LAYOUT	Plan Actual	[illegible]	[illegible]															
	15	PRODUCE PCB USING ETCHING OR CNC MILING	Plan Actual		[illegible]															
	16	SOLDERING TOOLS AND TECHNIQUE	Plan Actual	[illegible]	[illegible]															
	17	COMPONENT AND CIRCUIT TESTING	Plan Actual	[illegible]	[illegible]	[illegible]														
	18	DOCUMENT WRITING REPORT(FINAL PROPOSAL)	Plan Actual	[illegible]	[illegible]	[illegible]														
	19	PROPOSAL WRITING	Plan Actual	[illegible]	[illegible]	[illegible]														
	20	LOGBOOK WRITING	Plan Actual	[illegible]	[illegible]															
Course	21	Task Name	Implementation	Duration (Days)	Cost (RM)	Date	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14
[illegible] PROJECT 2	22	INSTALLATION	Plan Actual	[illegible]	[illegible]	[illegible]														
	23	INSTALLATION OF COMPONENTS ON PCB	Plan Actual	[illegible]	[illegible]	[illegible]														
	24	INSTALLATION OF WIRING	Plan Actual	[illegible]	[illegible]															
	25	INSTALLATION OF SOFTWARE	Plan Actual	[illegible]	[illegible]	[illegible]														
	26	INSTALLATION OF CONTROL CIRCUIT / SYSTEM	Plan Actual	[illegible]	[illegible]															
	27	INSTALLATION OF PROJECT CASING	Plan Actual	[illegible]	[illegible]	[illegible]														
	28	TESTING	Plan Actual	[illegible]	[illegible]	[illegible]														
	29	TEST THE ELECTRONIC PART	Plan Actual	[illegible]	[illegible]	[illegible]														
	30	TEST THE MECHANICAL PART	Plan Actual	[illegible]	[illegible]	[illegible]														
	31	TEST THE OVERALL PROCESS / PROJECT	Plan Actual	[illegible]	[illegible]	[illegible]														
	32	DOCUMENTS	Plan Actual	[illegible]	[illegible]	[illegible]														
	33	PREPARATION OF SLIDE PRESENTATION	Plan Actual	[illegible]	[illegible]															
	34	PREPARATION OF LOGBOOK	Plan Actual	[illegible]	[illegible]															
	35	PREPARATION OF PROJECT 2 FINAL REPORT	Plan Actual	[illegible]	[illegible]	[illegible]														
	36	PREPARATION OF INSTRUCTION MANUAL	Plan Actual	[illegible]	[illegible]															
	37	END	Plan Actual	[illegible]		[illegible]														

6.4 Cost and Budgeting

This project involves the cost of purchasing components and materials throughout its implementation. components involving cost are hardware ESP32, Battery 18650 LiPo, Buzzer, 2 Channel Relay ,Switch , Ultrasonic Sensor , Shoes Sanitizer , Shoes Deodorant , Shoes Rack . All of these components are purchased through online purchase methods to make it easier as well as save on costs. The overall gross budget estimate in the implementation of this project is RM and other expenses is at RM 300 as shown in Table 1 . According to this budget cost, this project is can be considered as a less costly project compared to other projects that can cost over a thousand ringgit. The cost of the project is also in line with one of the key features of a good project developer that is low cost but have a high quality project.

No.	Component and materials	The unit price	Quantity	Total
1	ESP32	RM29	1	RM29
2	BATTERY 18650 LIPO	RM20	1	RM20
3	BUZZER	RM20	1	RM20
4	2 CHANNEL RELAY	RM6	1	RM6
5	SWITCH	RM3	1	RM3
6	ULTRASONIC SENSOR	RM5	3	RM10
7	SHOES DEODORANT	RM20	1	RM20
8	SHOES SANITIZER	RM20	1	RM20
9	SHOES RACK	RM120	1	RM120
	TOTAL COSTING			**RM243**
	List of other costing			RM1000
			TOTAL:	**RM1243**
			OVERALL TOTAL:	**RM1243**

6.5 Chapter Summary

Costing and project management have both been covered in this chapter. A gantt chart and the project's operations are given. The project's cost and budget list, complete with component quantities and the amount, is also included**.**

REFERENCES

1. John M. Boyce, M.D, Didier Pittet, M.D, Centers for Disease Control and Prevention. Guideline for Hand Hygiene in Health-Care Settings: Recommendations of the Healthcare Infection Control Practices Advisory Committee and the HICPAC/SHEA/APIC/IDSA Hand Hygiene Task Force. MMWR 2002; 51(No. RR-16)

2. Klevens RM, Edwards JR, Richards CL, Estimating healthcare-associated infections in U.S. hospitals, 2002. Public Health Rep 2007; 122:160166.

3. Last Minutes Engineers , (2022) , How Water Level Sensor Works and Interface it with Arduino , [Online] , Available , https://lastminuteengineers.com/water-level-sensor-arduinotutorial/#:~:text=The%20water%20level%20sensor%20is%20extremely%20simple%20to%20use%20and,fro m%203.3V%20to%205V

4. Arduino Get Startes , (2018) , Arduino - Water Sensor , [Online] , Available , https://arduinogetstarted.com/tutorials/arduino-water-sensor

5. Circuits Autodesk Instructables , (2022) , Hpw To Use the Water Level Sensor – Arduino Tutorial , [Online] , Available , https://www.instructables.com/How-to-use-a-Water-Level-Sensor-Arduino-Tutorial/

6. Awesome Inventive , (2020) , How To Make Automatic Sanitizer Mist Sprayer || Sanitizer Dispenser , [YouTube] , Available , https://youtu.be/zjgXUaaNRSY

7. Technoreview85 , (2021) , How To Make Automatic Sanitizer Mist Sprayer || Sanitizer Dispenser , [YouTube] , Available , https://youtu.be/JQNAcU_voNc

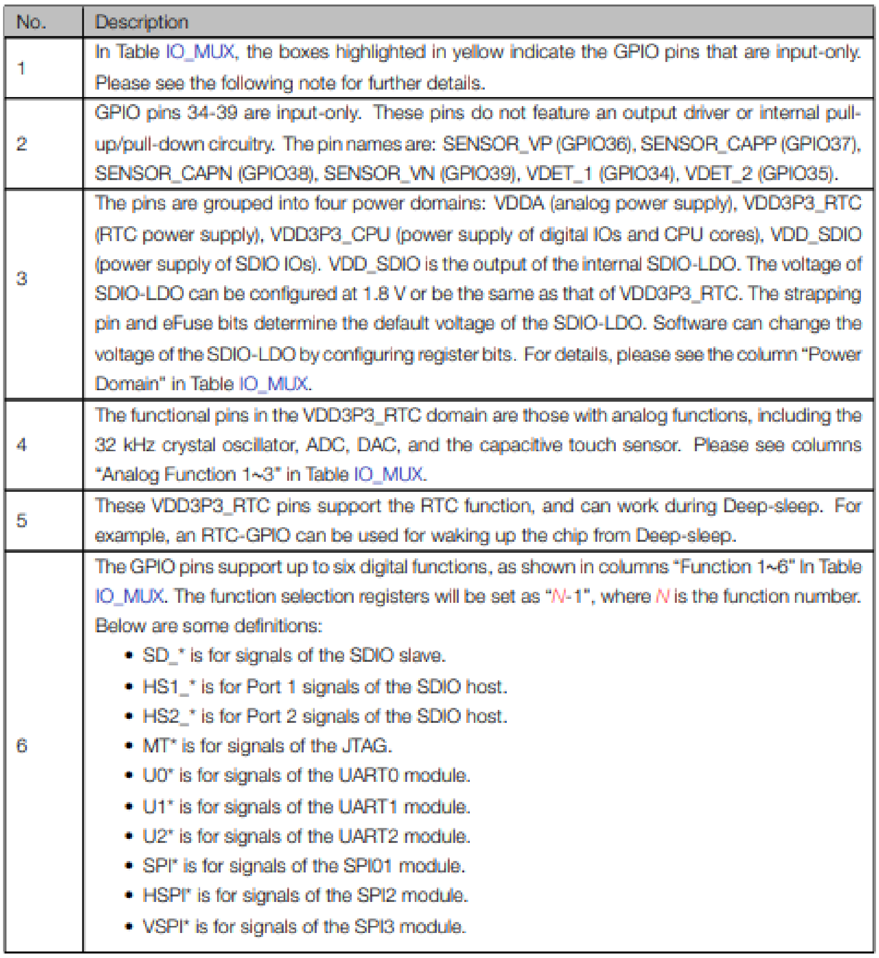
8. Nekobee , (2019) , Homemade Automatic Sprayer / Penyembur Air Using Arduino – Malaysia, [YouTube] , Available , https://youtu.be/AcuOxoNg0vI

9. Tech & Lifestyle , (2020) , How to Use Air Wick Freshmatic (Summer Delights) - Automatic Air Freshener , [YouTube] , Available , https://youtu.be/FC3k3bMwWlI

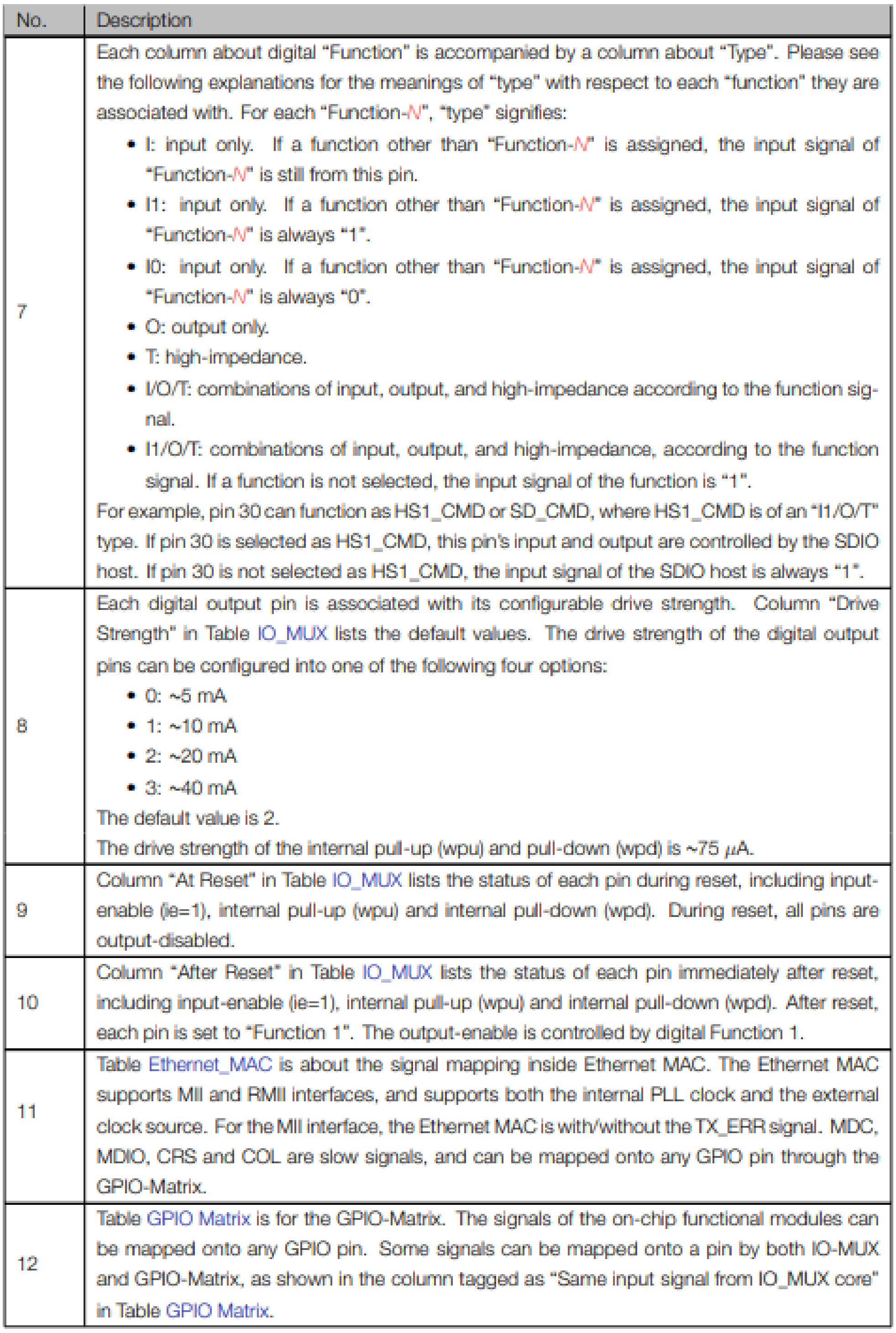
10. The Geek Pub , (2019) , Arduino Water Level Sensor Tutorial , [YouTube] , Available https://youtu.be/n7WRi5U5lQk

APPENDICES

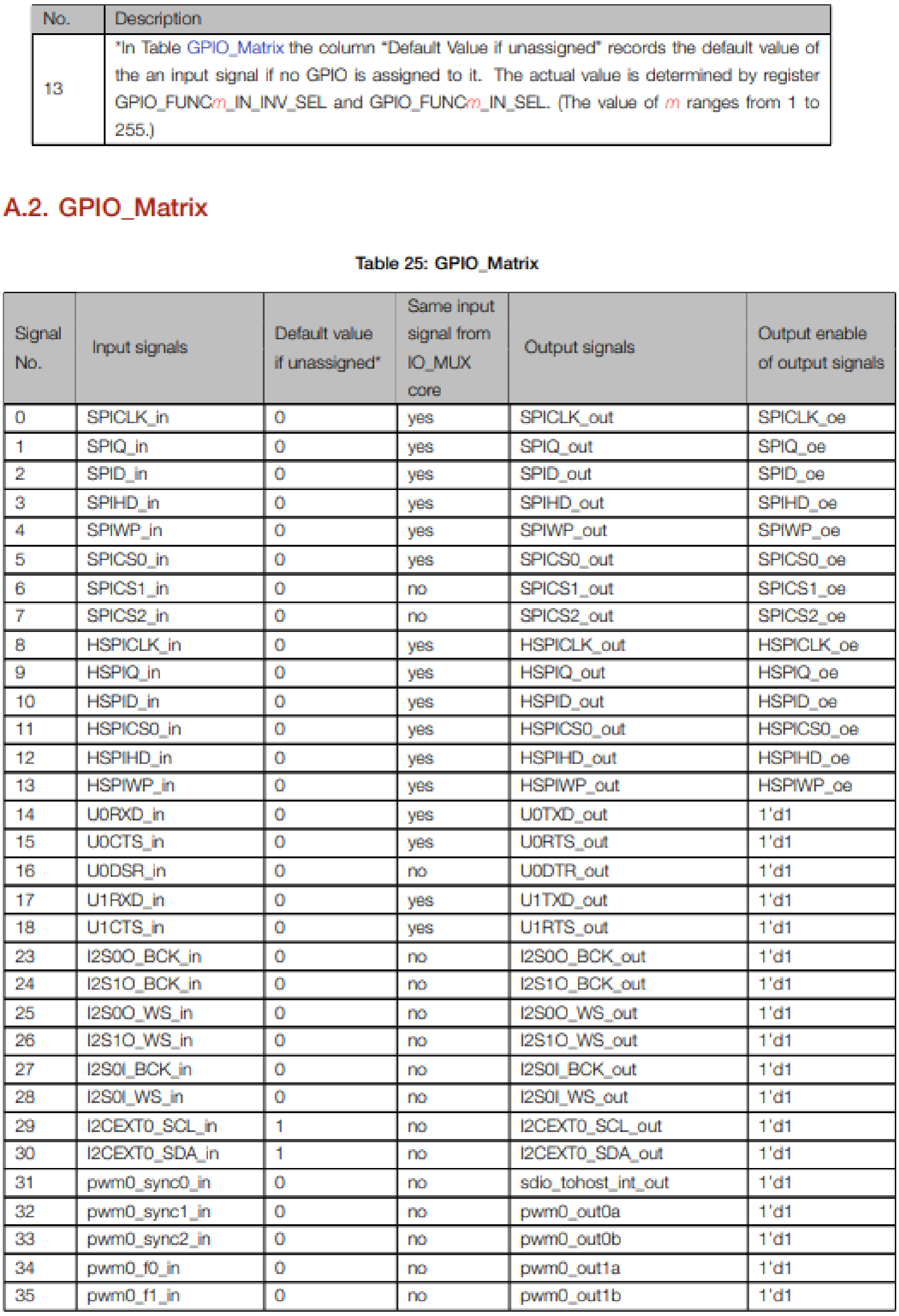
APPENDIX A- DATA SHEET



No.	Description
1	In Table IO_MUX, the boxes highlighted in yellow indicate the GPIO pins that are input-only. Please see the following note for further details.
2	GPIO pins 34-39 are input-only. These pins do not feature an output driver or internal pull-up/pull-down circuitry. The pin names are: SENSOR_VP (GPIO36), SENSOR_CAPP (GPIO37), SENSOR_CAPN (GPIO38), SENSOR_VN (GPIO39), VDET_1 (GPIO34), VDET_2 (GPIO35).
3	The pins are grouped into four power domains: VDDA (analog power supply), VDD3P3_RTC (RTC power supply), VDD3P3_CPU (power supply of digital IOs and CPU cores), VDD_SDIO (power supply of SDIO IOs). VDD_SDIO is the output of the internal SDIO-LDO. The voltage of SDIO-LDO can be configured at 1.8 V or be the same as that of VDD3P3_RTC. The strapping pin and eFuse bits determine the default voltage of the SDIO-LDO. Software can change the voltage of the SDIO-LDO by configuring register bits. For details, please see the column "Power Domain" in Table IO_MUX.
4	The functional pins in the VDD3P3_RTC domain are those with analog functions, including the 32 kHz crystal oscillator, ADC, DAC, and the capacitive touch sensor. Please see columns "Analog Function 1~3" in Table IO_MUX.
5	These VDD3P3_RTC pins support the RTC function, and can work during Deep-sleep. For example, an RTC-GPIO can be used for waking up the chip from Deep-sleep.
6	The GPIO pins support up to six digital functions, as shown in columns "Function 1~6" In Table IO_MUX. The function selection registers will be set as "N-1", where N is the function number. Below are some definitions: • SD_* is for signals of the SDIO slave. • HS1_* is for Port 1 signals of the SDIO host. • HS2_* is for Port 2 signals of the SDIO host. • MT* is for signals of the JTAG. • U0* is for signals of the UART0 module. • U1* is for signals of the UART1 module. • U2* is for signals of the UART2 module. • SPI* is for signals of the SPI01 module. • HSPI* is for signals of the SPI2 module. • VSPI* is for signals of the SPI3 module.



No.	Description
7	Each column about digital "Function" is accompanied by a column about "Type". Please see the following explanations for the meanings of "type" with respect to each "function" they are associated with. For each "Function-*N*", "type" signifies: • I: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is still from this pin. • I1: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is always "1". • I0: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is always "0". • O: output only. • T: high-impedance. • I/O/T: combinations of input, output, and high-impedance according to the function signal. • I1/O/T: combinations of input, output, and high-impedance, according to the function signal. If a function is not selected, the input signal of the function is "1". For example, pin 30 can function as HS1_CMD or SD_CMD, where HS1_CMD is of an "I1/O/T" type. If pin 30 is selected as HS1_CMD, this pin's input and output are controlled by the SDIO host. If pin 30 is not selected as HS1_CMD, the input signal of the SDIO host is always "1".
8	Each digital output pin is associated with its configurable drive strength. Column "Drive Strength" in Table IO_MUX lists the default values. The drive strength of the digital output pins can be configured into one of the following four options: • 0: ~5 mA • 1: ~10 mA • 2: ~20 mA • 3: ~40 mA The default value is 2. The drive strength of the internal pull-up (wpu) and pull-down (wpd) is ~75 μA.
9	Column "At Reset" in Table IO_MUX lists the status of each pin during reset, including input-enable (ie=1), internal pull-up (wpu) and internal pull-down (wpd). During reset, all pins are output-disabled.
10	Column "After Reset" in Table IO_MUX lists the status of each pin immediately after reset, including input-enable (ie=1), internal pull-up (wpu) and internal pull-down (wpd). After reset, each pin is set to "Function 1". The output-enable is controlled by digital Function 1.
11	Table Ethernet_MAC is about the signal mapping inside Ethernet MAC. The Ethernet MAC supports MII and RMII interfaces, and supports both the internal PLL clock and the external clock source. For the MII interface, the Ethernet MAC is with/without the TX_ERR signal. MDC, MDIO, CRS and COL are slow signals, and can be mapped onto any GPIO pin through the GPIO-Matrix.
12	Table GPIO Matrix is for the GPIO-Matrix. The signals of the on-chip functional modules can be mapped onto any GPIO pin. Some signals can be mapped onto a pin by both IO-MUX and GPIO-Matrix, as shown in the column tagged as "Same input signal from IO_MUX core" in Table GPIO Matrix.

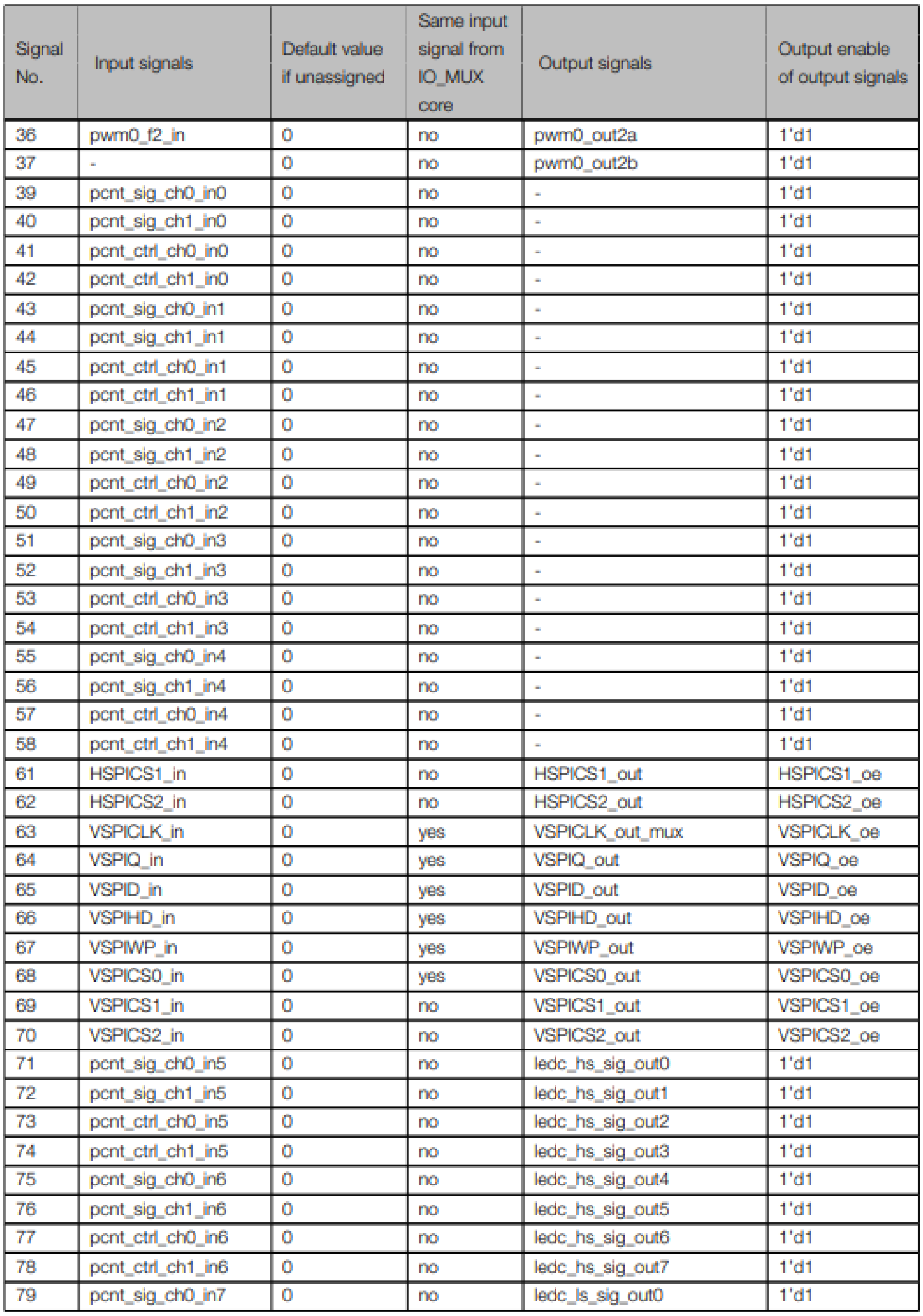


No.	Description
13	*In Table GPIO_Matrix the column "Default Value if unassigned" records the default value of the an input signal if no GPIO is assigned to it. The actual value is determined by register GPIO_FUNC*m*_IN_INV_SEL and GPIO_FUNC*m*_IN_SEL. (The value of *m* ranges from 1 to 255.)

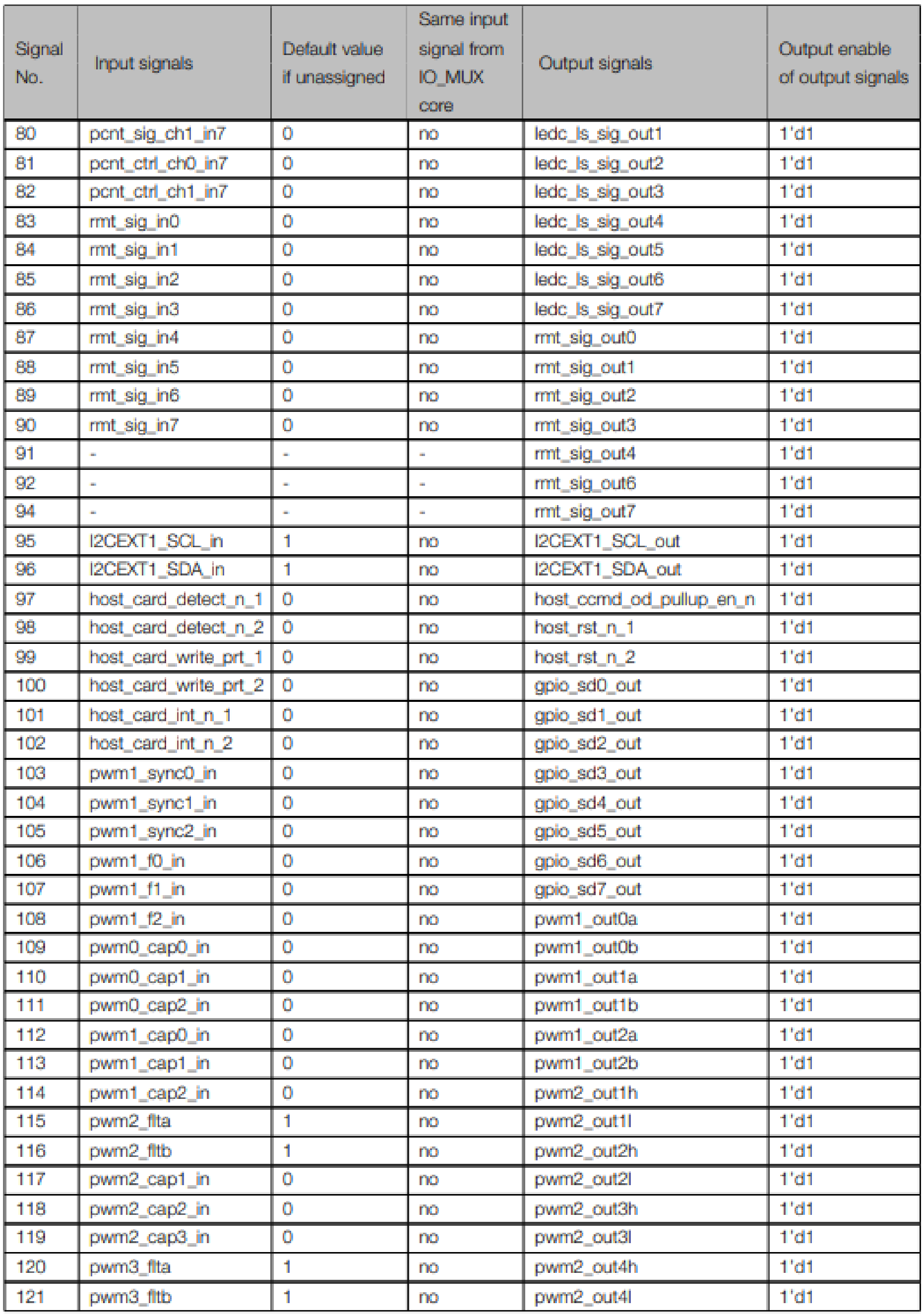
A.2. GPIO_Matrix

Table 25: GPIO_Matrix

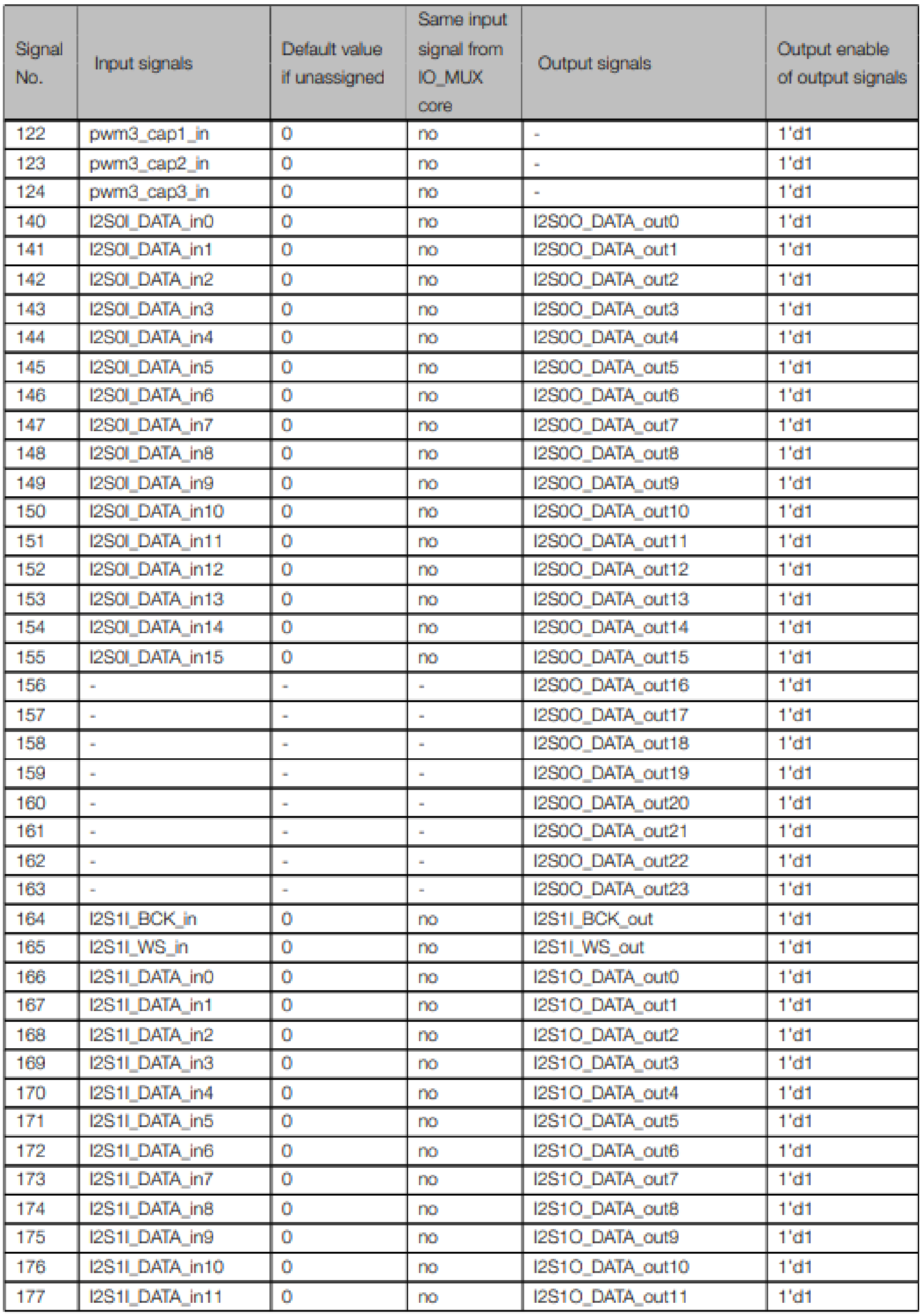
Signal No.	Input signals	Default value if unassigned*	Same input signal from IO_MUX core	Output signals	Output enable of output signals
0	SPICLK_in	0	yes	SPICLK_out	SPICLK_oe
1	SPIQ_in	0	yes	SPIQ_out	SPIQ_oe
2	SPID_in	0	yes	SPID_out	SPID_oe
3	SPIHD_in	0	yes	SPIHD_out	SPIHD_oe
4	SPIWP_in	0	yes	SPIWP_out	SPIWP_oe
5	SPICS0_in	0	yes	SPICS0_out	SPICS0_oe
6	SPICS1_in	0	no	SPICS1_out	SPICS1_oe
7	SPICS2_in	0	no	SPICS2_out	SPICS2_oe
8	HSPICLK_in	0	yes	HSPICLK_out	HSPICLK_oe
9	HSPIQ_in	0	yes	HSPIQ_out	HSPIQ_oe
10	HSPID_in	0	yes	HSPID_out	HSPID_oe
11	HSPICS0_in	0	yes	HSPICS0_out	HSPICS0_oe
12	HSPIHD_in	0	yes	HSPIHD_out	HSPIHD_oe
13	HSPIWP_in	0	yes	HSPIWP_out	HSPIWP_oe
14	U0RXD_in	0	yes	U0TXD_out	1'd1
15	U0CTS_in	0	yes	U0RTS_out	1'd1
16	U0DSR_in	0	no	U0DTR_out	1'd1
17	U1RXD_in	0	yes	U1TXD_out	1'd1
18	U1CTS_in	0	yes	U1RTS_out	1'd1
23	I2S0O_BCK_in	0	no	I2S0O_BCK_out	1'd1
24	I2S1O_BCK_in	0	no	I2S1O_BCK_out	1'd1
25	I2S0O_WS_in	0	no	I2S0O_WS_out	1'd1
26	I2S1O_WS_in	0	no	I2S1O_WS_out	1'd1
27	I2S0I_BCK_in	0	no	I2S0I_BCK_out	1'd1
28	I2S0I_WS_in	0	no	I2S0I_WS_out	1'd1
29	I2CEXT0_SCL_in	1	no	I2CEXT0_SCL_out	1'd1
30	I2CEXT0_SDA_in	1	no	I2CEXT0_SDA_out	1'd1
31	pwm0_sync0_in	0	no	sdio_tohost_int_out	1'd1
32	pwm0_sync1_in	0	no	pwm0_out0a	1'd1
33	pwm0_sync2_in	0	no	pwm0_out0b	1'd1
34	pwm0_f0_in	0	no	pwm0_out1a	1'd1
35	pwm0_f1_in	0	no	pwm0_out1b	1'd1



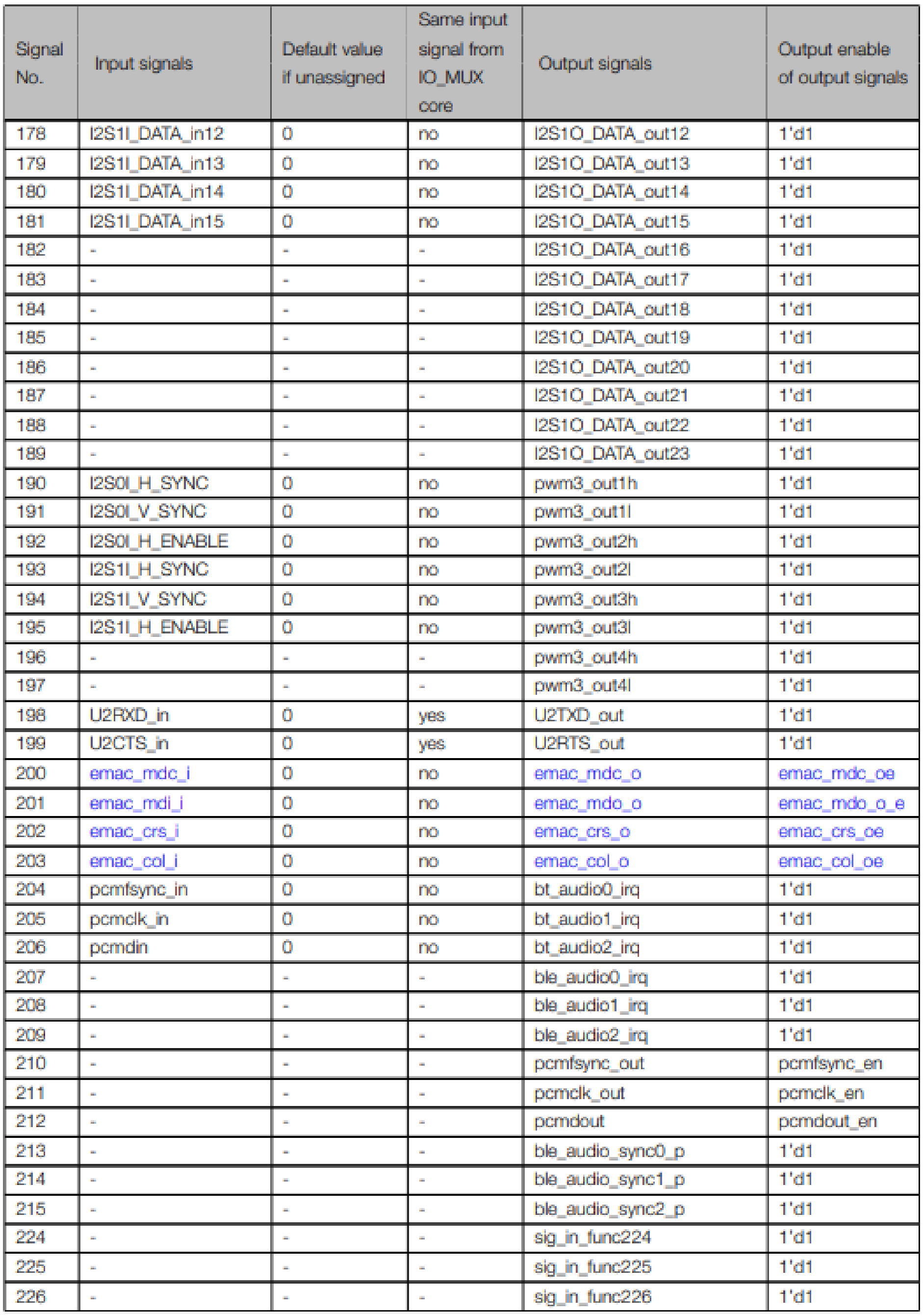
Signal No.	Input signals	Default value if unassigned	Same input signal from IO_MUX core	Output signals	Output enable of output signals
36	pwm0_f2_in	0	no	pwm0_out2a	1'd1
37	-	0	no	pwm0_out2b	1'd1
39	pcnt_sig_ch0_in0	0	no	-	1'd1
40	pcnt_sig_ch1_in0	0	no	-	1'd1
41	pcnt_ctrl_ch0_in0	0	no	-	1'd1
42	pcnt_ctrl_ch1_in0	0	no	-	1'd1
43	pcnt_sig_ch0_in1	0	no	-	1'd1
44	pcnt_sig_ch1_in1	0	no	-	1'd1
45	pcnt_ctrl_ch0_in1	0	no	-	1'd1
46	pcnt_ctrl_ch1_in1	0	no	-	1'd1
47	pcnt_sig_ch0_in2	0	no	-	1'd1
48	pcnt_sig_ch1_in2	0	no	-	1'd1
49	pcnt_ctrl_ch0_in2	0	no	-	1'd1
50	pcnt_ctrl_ch1_in2	0	no	-	1'd1
51	pcnt_sig_ch0_in3	0	no	-	1'd1
52	pcnt_sig_ch1_in3	0	no	-	1'd1
53	pcnt_ctrl_ch0_in3	0	no	-	1'd1
54	pcnt_ctrl_ch1_in3	0	no	-	1'd1
55	pcnt_sig_ch0_in4	0	no	-	1'd1
56	pcnt_sig_ch1_in4	0	no	-	1'd1
57	pcnt_ctrl_ch0_in4	0	no	-	1'd1
58	pcnt_ctrl_ch1_in4	0	no	-	1'd1
61	HSPICS1_in	0	no	HSPICS1_out	HSPICS1_oe
62	HSPICS2_in	0	no	HSPICS2_out	HSPICS2_oe
63	VSPICLK_in	0	yes	VSPICLK_out_mux	VSPICLK_oe
64	VSPIQ_in	0	yes	VSPIQ_out	VSPIQ_oe
65	VSPID_in	0	yes	VSPID_out	VSPID_oe
66	VSPIHD_in	0	yes	VSPIHD_out	VSPIHD_oe
67	VSPIWP_in	0	yes	VSPIWP_out	VSPIWP_oe
68	VSPICS0_in	0	yes	VSPICS0_out	VSPICS0_oe
69	VSPICS1_in	0	no	VSPICS1_out	VSPICS1_oe
70	VSPICS2_in	0	no	VSPICS2_out	VSPICS2_oe
71	pcnt_sig_ch0_in5	0	no	ledc_hs_sig_out0	1'd1
72	pcnt_sig_ch1_in5	0	no	ledc_hs_sig_out1	1'd1
73	pcnt_ctrl_ch0_in5	0	no	ledc_hs_sig_out2	1'd1
74	pcnt_ctrl_ch1_in5	0	no	ledc_hs_sig_out3	1'd1
75	pcnt_sig_ch0_in6	0	no	ledc_hs_sig_out4	1'd1
76	pcnt_sig_ch1_in6	0	no	ledc_hs_sig_out5	1'd1
77	pcnt_ctrl_ch0_in6	0	no	ledc_hs_sig_out6	1'd1
78	pcnt_ctrl_ch1_in6	0	no	ledc_hs_sig_out7	1'd1
79	pcnt_sig_ch0_in7	0	no	ledc_ls_sig_out0	1'd1



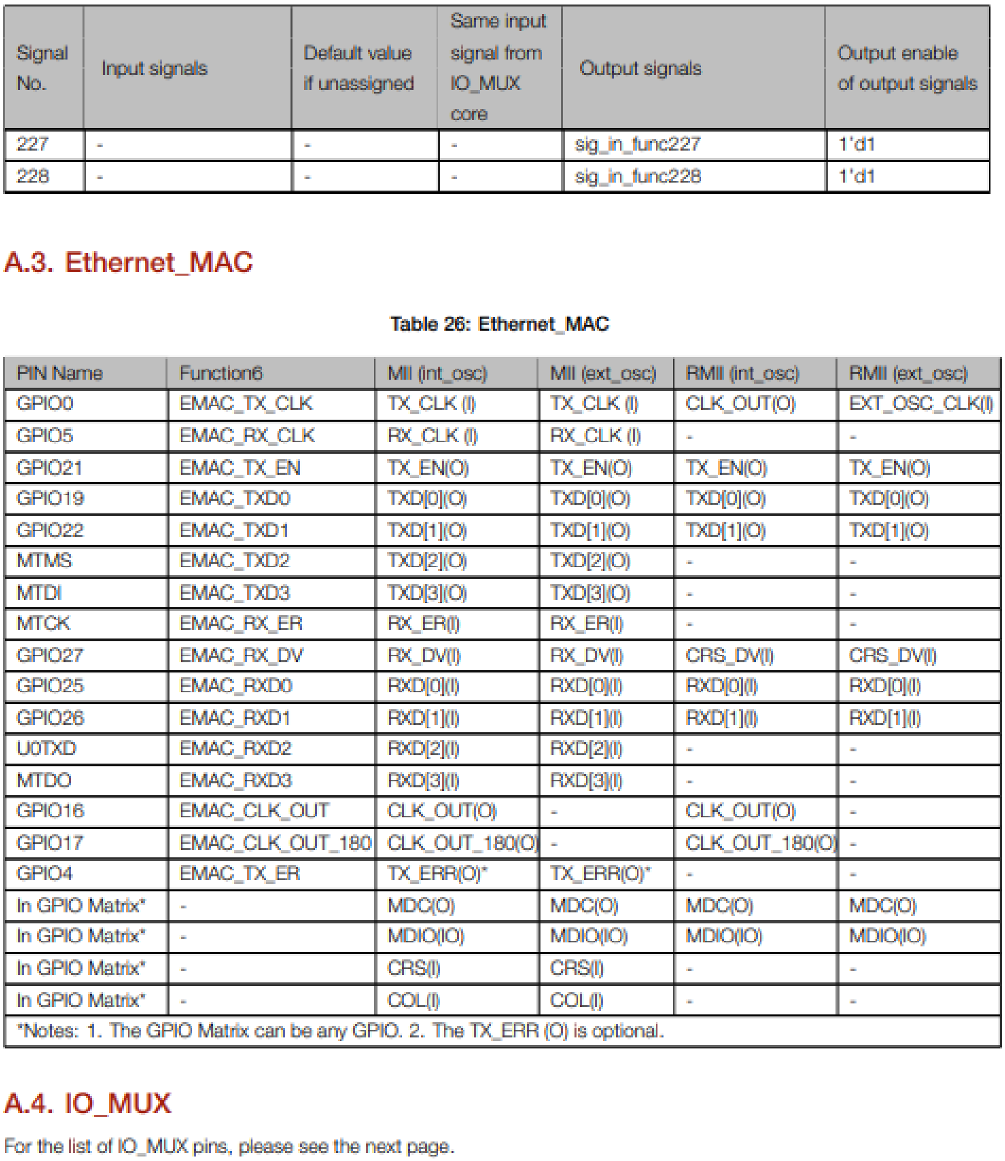
Signal No.	Input signals	Default value if unassigned	Same input signal from IO_MUX core	Output signals	Output enable of output signals
80	pcnt_sig_ch1_in7	0	no	ledc_ls_sig_out1	1'd1
81	pcnt_ctrl_ch0_in7	0	no	ledc_ls_sig_out2	1'd1
82	pcnt_ctrl_ch1_in7	0	no	ledc_ls_sig_out3	1'd1
83	rmt_sig_in0	0	no	ledc_ls_sig_out4	1'd1
84	rmt_sig_in1	0	no	ledc_ls_sig_out5	1'd1
85	rmt_sig_in2	0	no	ledc_ls_sig_out6	1'd1
86	rmt_sig_in3	0	no	ledc_ls_sig_out7	1'd1
87	rmt_sig_in4	0	no	rmt_sig_out0	1'd1
88	rmt_sig_in5	0	no	rmt_sig_out1	1'd1
89	rmt_sig_in6	0	no	rmt_sig_out2	1'd1
90	rmt_sig_in7	0	no	rmt_sig_out3	1'd1
91	-	-	-	rmt_sig_out4	1'd1
92	-	-	-	rmt_sig_out6	1'd1
94	-	-	-	rmt_sig_out7	1'd1
95	I2CEXT1_SCL_in	1	no	I2CEXT1_SCL_out	1'd1
96	I2CEXT1_SDA_in	1	no	I2CEXT1_SDA_out	1'd1
97	host_card_detect_n_1	0	no	host_ccmd_od_pullup_en_n	1'd1
98	host_card_detect_n_2	0	no	host_rst_n_1	1'd1
99	host_card_write_prt_1	0	no	host_rst_n_2	1'd1
100	host_card_write_prt_2	0	no	gpio_sd0_out	1'd1
101	host_card_int_n_1	0	no	gpio_sd1_out	1'd1
102	host_card_int_n_2	0	no	gpio_sd2_out	1'd1
103	pwm1_sync0_in	0	no	gpio_sd3_out	1'd1
104	pwm1_sync1_in	0	no	gpio_sd4_out	1'd1
105	pwm1_sync2_in	0	no	gpio_sd5_out	1'd1
106	pwm1_f0_in	0	no	gpio_sd6_out	1'd1
107	pwm1_f1_in	0	no	gpio_sd7_out	1'd1
108	pwm1_f2_in	0	no	pwm1_out0a	1'd1
109	pwm0_cap0_in	0	no	pwm1_out0b	1'd1
110	pwm0_cap1_in	0	no	pwm1_out1a	1'd1
111	pwm0_cap2_in	0	no	pwm1_out1b	1'd1
112	pwm1_cap0_in	0	no	pwm1_out2a	1'd1
113	pwm1_cap1_in	0	no	pwm1_out2b	1'd1
114	pwm1_cap2_in	0	no	pwm2_out1h	1'd1
115	pwm2_flta	1	no	pwm2_out1l	1'd1
116	pwm2_fltb	1	no	pwm2_out2h	1'd1
117	pwm2_cap1_in	0	no	pwm2_out2l	1'd1
118	pwm2_cap2_in	0	no	pwm2_out3h	1'd1
119	pwm2_cap3_in	0	no	pwm2_out3l	1'd1
120	pwm3_flta	1	no	pwm2_out4h	1'd1
121	pwm3_fltb	1	no	pwm2_out4l	1'd1



Signal No.	Input signals	Default value if unassigned	Same input signal from IO_MUX core	Output signals	Output enable of output signals
122	pwm3_cap1_in	0	no	-	1'd1
123	pwm3_cap2_in	0	no	-	1'd1
124	pwm3_cap3_in	0	no	-	1'd1
140	I2S0I_DATA_in0	0	no	I2S0O_DATA_out0	1'd1
141	I2S0I_DATA_in1	0	no	I2S0O_DATA_out1	1'd1
142	I2S0I_DATA_in2	0	no	I2S0O_DATA_out2	1'd1
143	I2S0I_DATA_in3	0	no	I2S0O_DATA_out3	1'd1
144	I2S0I_DATA_in4	0	no	I2S0O_DATA_out4	1'd1
145	I2S0I_DATA_in5	0	no	I2S0O_DATA_out5	1'd1
146	I2S0I_DATA_in6	0	no	I2S0O_DATA_out6	1'd1
147	I2S0I_DATA_in7	0	no	I2S0O_DATA_out7	1'd1
148	I2S0I_DATA_in8	0	no	I2S0O_DATA_out8	1'd1
149	I2S0I_DATA_in9	0	no	I2S0O_DATA_out9	1'd1
150	I2S0I_DATA_in10	0	no	I2S0O_DATA_out10	1'd1
151	I2S0I_DATA_in11	0	no	I2S0O_DATA_out11	1'd1
152	I2S0I_DATA_in12	0	no	I2S0O_DATA_out12	1'd1
153	I2S0I_DATA_in13	0	no	I2S0O_DATA_out13	1'd1
154	I2S0I_DATA_in14	0	no	I2S0O_DATA_out14	1'd1
155	I2S0I_DATA_in15	0	no	I2S0O_DATA_out15	1'd1
156	-	-	-	I2S0O_DATA_out16	1'd1
157	-	-	-	I2S0O_DATA_out17	1'd1
158	-	-	-	I2S0O_DATA_out18	1'd1
159	-	-	-	I2S0O_DATA_out19	1'd1
160	-	-	-	I2S0O_DATA_out20	1'd1
161	-	-	-	I2S0O_DATA_out21	1'd1
162	-	-	-	I2S0O_DATA_out22	1'd1
163	-	-	-	I2S0O_DATA_out23	1'd1
164	I2S1I_BCK_in	0	no	I2S1I_BCK_out	1'd1
165	I2S1I_WS_in	0	no	I2S1I_WS_out	1'd1
166	I2S1I_DATA_in0	0	no	I2S1O_DATA_out0	1'd1
167	I2S1I_DATA_in1	0	no	I2S1O_DATA_out1	1'd1
168	I2S1I_DATA_in2	0	no	I2S1O_DATA_out2	1'd1
169	I2S1I_DATA_in3	0	no	I2S1O_DATA_out3	1'd1
170	I2S1I_DATA_in4	0	no	I2S1O_DATA_out4	1'd1
171	I2S1I_DATA_in5	0	no	I2S1O_DATA_out5	1'd1
172	I2S1I_DATA_in6	0	no	I2S1O_DATA_out6	1'd1
173	I2S1I_DATA_in7	0	no	I2S1O_DATA_out7	1'd1
174	I2S1I_DATA_in8	0	no	I2S1O_DATA_out8	1'd1
175	I2S1I_DATA_in9	0	no	I2S1O_DATA_out9	1'd1
176	I2S1I_DATA_in10	0	no	I2S1O_DATA_out10	1'd1
177	I2S1I_DATA_in11	0	no	I2S1O_DATA_out11	1'd1



Signal No.	Input signals	Default value if unassigned	Same input signal from IO_MUX core	Output signals	Output enable of output signals
178	I2S1I_DATA_in12	0	no	I2S1O_DATA_out12	1'd1
179	I2S1I_DATA_in13	0	no	I2S1O_DATA_out13	1'd1
180	I2S1I_DATA_in14	0	no	I2S1O_DATA_out14	1'd1
181	I2S1I_DATA_in15	0	no	I2S1O_DATA_out15	1'd1
182	-	-	-	I2S1O_DATA_out16	1'd1
183	-	-	-	I2S1O_DATA_out17	1'd1
184	-	-	-	I2S1O_DATA_out18	1'd1
185	-	-	-	I2S1O_DATA_out19	1'd1
186	-	-	-	I2S1O_DATA_out20	1'd1
187	-	-	-	I2S1O_DATA_out21	1'd1
188	-	-	-	I2S1O_DATA_out22	1'd1
189	-	-	-	I2S1O_DATA_out23	1'd1
190	I2S0I_H_SYNC	0	no	pwm3_out1h	1'd1
191	I2S0I_V_SYNC	0	no	pwm3_out1l	1'd1
192	I2S0I_H_ENABLE	0	no	pwm3_out2h	1'd1
193	I2S1I_H_SYNC	0	no	pwm3_out2l	1'd1
194	I2S1I_V_SYNC	0	no	pwm3_out3h	1'd1
195	I2S1I_H_ENABLE	0	no	pwm3_out3l	1'd1
196	-	-	-	pwm3_out4h	1'd1
197	-	-	-	pwm3_out4l	1'd1
198	U2RXD_in	0	yes	U2TXD_out	1'd1
199	U2CTS_in	0	yes	U2RTS_out	1'd1
200	emac_mdc_i	0	no	emac_mdc_o	emac_mdc_oe
201	emac_mdi_i	0	no	emac_mdo_o	emac_mdo_o_e
202	emac_crs_i	0	no	emac_crs_o	emac_crs_oe
203	emac_col_i	0	no	emac_col_o	emac_col_oe
204	pcmfsync_in	0	no	bt_audio0_irq	1'd1
205	pcmclk_in	0	no	bt_audio1_irq	1'd1
206	pcmdin	0	no	bt_audio2_irq	1'd1
207	-	-	-	ble_audio0_irq	1'd1
208	-	-	-	ble_audio1_irq	1'd1
209	-	-	-	ble_audio2_irq	1'd1
210	-	-	-	pcmfsync_out	pcmfsync_en
211	-	-	-	pcmclk_out	pcmclk_en
212	-	-	-	pcmdout	pcmdout_en
213	-	-	-	ble_audio_sync0_p	1'd1
214	-	-	-	ble_audio_sync1_p	1'd1
215	-	-	-	ble_audio_sync2_p	1'd1
224	-	-	-	sig_in_func224	1'd1
225	-	-	-	sig_in_func225	1'd1
226	-	-	-	sig_in_func226	1'd1



Signal No.	Input signals	Default value if unassigned	Same input signal from IO_MUX core	Output signals	Output enable of output signals
227	-	-	-	sig_in_func227	1'd1
228	-	-	-	sig_in_func228	1'd1

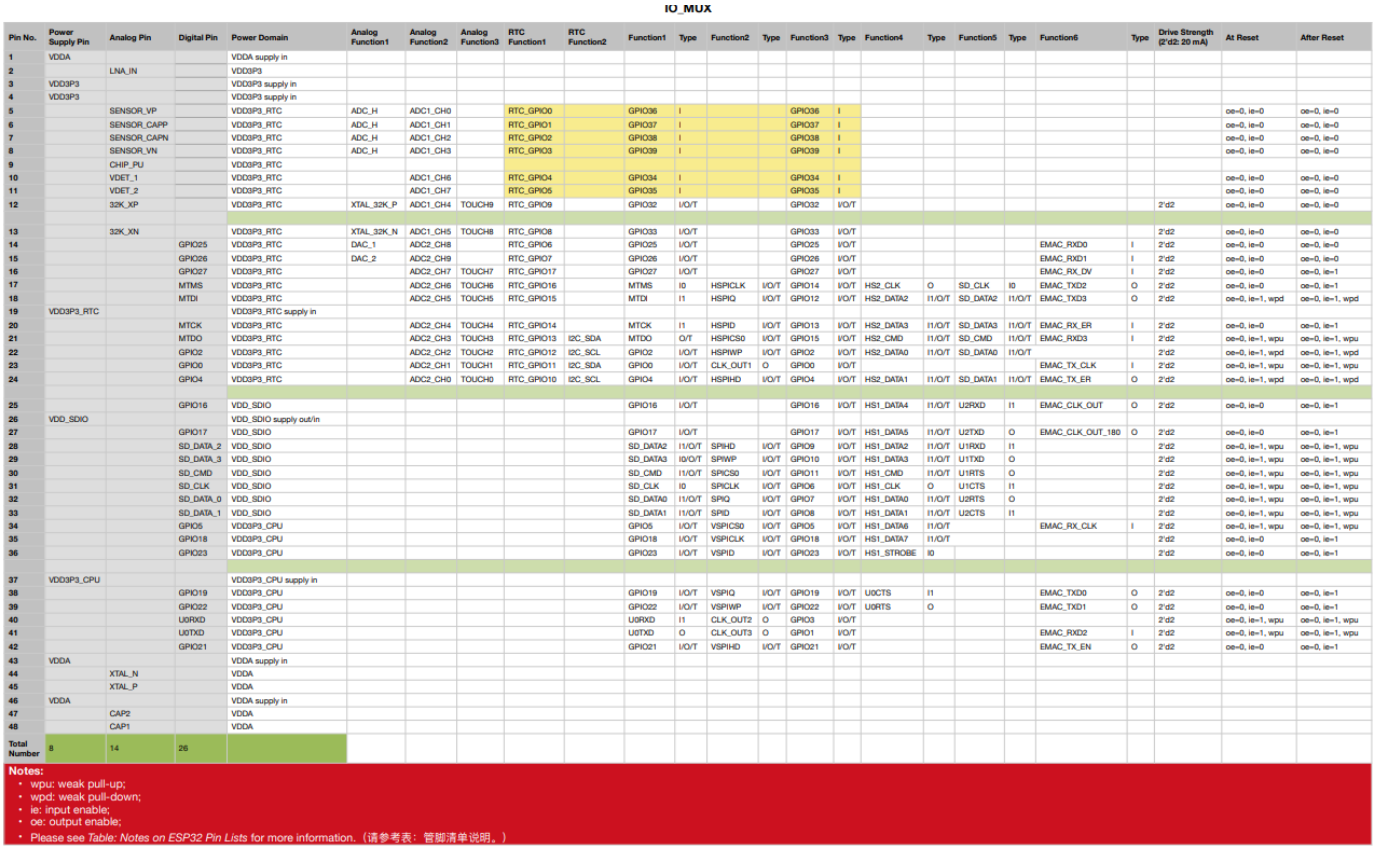
A.3. Ethernet_MAC

Table 26: Ethernet_MAC

PIN Name	Function6	MII (int_osc)	MII (ext_osc)	RMII (int_osc)	RMII (ext_osc)
GPIO0	EMAC_TX_CLK	TX_CLK (I)	TX_CLK (I)	CLK_OUT(O)	EXT_OSC_CLK(I)
GPIO5	EMAC_RX_CLK	RX_CLK (I)	RX_CLK (I)	-	-
GPIO21	EMAC_TX_EN	TX_EN(O)	TX_EN(O)	TX_EN(O)	TX_EN(O)
GPIO19	EMAC_TXD0	TXD[0](O)	TXD[0](O)	TXD[0](O)	TXD[0](O)
GPIO22	EMAC_TXD1	TXD[1](O)	TXD[1](O)	TXD[1](O)	TXD[1](O)
MTMS	EMAC_TXD2	TXD[2](O)	TXD[2](O)	-	-
MTDI	EMAC_TXD3	TXD[3](O)	TXD[3](O)	-	-
MTCK	EMAC_RX_ER	RX_ER(I)	RX_ER(I)	-	-
GPIO27	EMAC_RX_DV	RX_DV(I)	RX_DV(I)	CRS_DV(I)	CRS_DV(I)
GPIO25	EMAC_RXD0	RXD[0](I)	RXD[0](I)	RXD[0](I)	RXD[0](I)
GPIO26	EMAC_RXD1	RXD[1](I)	RXD[1](I)	RXD[1](I)	RXD[1](I)
U0TXD	EMAC_RXD2	RXD[2](I)	RXD[2](I)	-	-
MTDO	EMAC_RXD3	RXD[3](I)	RXD[3](I)	-	-
GPIO16	EMAC_CLK_OUT	CLK_OUT(O)	-	CLK_OUT(O)	-
GPIO17	EMAC_CLK_OUT_180	CLK_OUT_180(O)	-	CLK_OUT_180(O)	-
GPIO4	EMAC_TX_ER	TX_ERR(O)*	TX_ERR(O)*	-	-
In GPIO Matrix*	-	MDC(O)	MDC(O)	MDC(O)	MDC(O)
In GPIO Matrix*	-	MDIO(IO)	MDIO(IO)	MDIO(IO)	MDIO(IO)
In GPIO Matrix*	-	CRS(I)	CRS(I)	-	-
In GPIO Matrix*	-	COL(I)	COL(I)	-	-
*Notes: 1. The GPIO Matrix can be any GPIO. 2. The TX_ERR (O) is optional.					

A.4. IO_MUX

For the list of IO_MUX pins, please see the next page.



IO_MUX

Pin No.	Power Supply Pin	Analog Pin	Digital Pin	Power Domain	Analog Function1	Analog Function2	Analog Function3	RTC Function1	RTC Function2	Function1	Type	Function2	Type	Function3	Type	Function4	Type	Function5	Type	Function6	Type	Drive Strength (2'd2: 20 mA)	At Reset	After Reset
1	VDDA			VDDA supply in																				
2		LNA_IN		VDD3P3																				
3	VDD3P3			VDD3P3 supply in																				
4	VDD3P3			VDD3P3 supply in																				
5		SENSOR_VP		VDD3P3_RTC	ADC_H	ADC1_CH0		RTC_GPIO0		GPIO36	I			GPIO36	I								oe=0, ie=0	oe=0, ie=0
6		SENSOR_CAPP		VDD3P3_RTC	ADC_H	ADC1_CH1		RTC_GPIO1		GPIO37	I			GPIO37	I								oe=0, ie=0	oe=0, ie=0
7		SENSOR_CAPN		VDD3P3_RTC	ADC_H	ADC1_CH2		RTC_GPIO2		GPIO38	I			GPIO38	I								oe=0, ie=0	oe=0, ie=0
8		SENSOR_VN		VDD3P3_RTC	ADC_H	ADC1_CH3		RTC_GPIO3		GPIO39	I			GPIO39	I								oe=0, ie=0	oe=0, ie=0
9		CHIP_PU		VDD3P3_RTC																				
10		VDET_1		VDD3P3_RTC		ADC1_CH6		RTC_GPIO4		GPIO34	I			GPIO34	I								oe=0, ie=0	oe=0, ie=0
11		VDET_2		VDD3P3_RTC		ADC1_CH7		RTC_GPIO5		GPIO35	I			GPIO35	I								oe=0, ie=0	oe=0, ie=0
12		32K_XP		VDD3P3_RTC	XTAL_32K_P	ADC1_CH4	TOUCH9	RTC_GPIO9		GPIO32	I/O/T			GPIO32	I/O/T							2'd2	oe=0, ie=0	oe=0, ie=0
13		32K_XN		VDD3P3_RTC	XTAL_32K_N	ADC1_CH5	TOUCH8	RTC_GPIO8		GPIO33	I/O/T			GPIO33	I/O/T							2'd2	oe=0, ie=0	oe=0, ie=0
14			GPIO25	VDD3P3_RTC	DAC_1	ADC2_CH8		RTC_GPIO6		GPIO25	I/O/T			GPIO25	I/O/T					EMAC_RXD0	I	2'd2	oe=0, ie=0	oe=0, ie=0
15			GPIO26	VDD3P3_RTC	DAC_2	ADC2_CH9		RTC_GPIO7		GPIO26	I/O/T			GPIO26	I/O/T					EMAC_RXD1	I	2'd2	oe=0, ie=0	oe=0, ie=0
16			GPIO27	VDD3P3_RTC		ADC2_CH7	TOUCH7	RTC_GPIO17		GPIO27	I/O/T			GPIO27	I/O/T					EMAC_RX_DV	I	2'd2	oe=0, ie=0	oe=0, ie=1
17			MTMS	VDD3P3_RTC		ADC2_CH6	TOUCH6	RTC_GPIO16		MTMS	I0	HSPICLK	I/O/T	GPIO14	I/O/T	HS2_CLK	O	SD_CLK	I0	EMAC_TXD2	O	2'd2	oe=0, ie=0	oe=0, ie=1
18			MTDI	VDD3P3_RTC		ADC2_CH5	TOUCH5	RTC_GPIO15		MTDI	I1	HSPIQ	I/O/T	GPIO12	I/O/T	HS2_DATA2	I1/O/T	SD_DATA2	I1/O/T	EMAC_TXD3	O	2'd2	oe=0, ie=1, wpd	oe=0, ie=1, wpd
19	VDD3P3_RTC			VDD3P3_RTC supply in																				
20			MTCK	VDD3P3_RTC		ADC2_CH4	TOUCH4	RTC_GPIO14		MTCK	I1	HSPID	I/O/T	GPIO13	I/O/T	HS2_DATA3	I1/O/T	SD_DATA3	I1/O/T	EMAC_RX_ER	I	2'd2	oe=0, ie=0	oe=0, ie=1
21			MTDO	VDD3P3_RTC		ADC2_CH3	TOUCH3	RTC_GPIO13	I2C_SDA	MTDO	O/T	HSPICS0	I/O/T	GPIO15	I/O/T	HS2_CMD	I1/O/T	SD_CMD	I1/O/T	EMAC_RXD3	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
22			GPIO2	VDD3P3_RTC		ADC2_CH2	TOUCH2	RTC_GPIO12	I2C_SCL	GPIO2	I/O/T	HSPIWP	I/O/T	GPIO2	I/O/T	HS2_DATA0	I1/O/T	SD_DATA0	I1/O/T			2'd2	oe=0, ie=1, wpd	oe=0, ie=1, wpd
23			GPIO0	VDD3P3_RTC		ADC2_CH1	TOUCH1	RTC_GPIO11	I2C_SDA	GPIO0	I/O/T	CLK_OUT1	O	GPIO0	I/O/T					EMAC_TX_CLK	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
24			GPIO4	VDD3P3_RTC		ADC2_CH0	TOUCH0	RTC_GPIO10	I2C_SCL	GPIO4	I/O/T	HSPIHD	I/O/T	GPIO4	I/O/T	HS2_DATA1	I1/O/T	SD_DATA1	I1/O/T	EMAC_TX_ER	O	2'd2	oe=0, ie=1, wpd	oe=0, ie=1, wpd
25			GPIO16	VDD_SDIO						GPIO16	I/O/T			GPIO16	I/O/T	HS1_DATA4	I1/O/T	U2RXD	I1	EMAC_CLK_OUT	O	2'd2	oe=0, ie=0	oe=0, ie=1
26	VDD_SDIO			VDD_SDIO supply out/in																				
27			GPIO17	VDD_SDIO						GPIO17	I/O/T			GPIO17	I/O/T	HS1_DATA5	I1/O/T	U2TXD	O	EMAC_CLK_OUT_180	O	2'd2	oe=0, ie=0	oe=0, ie=1
28			SD_DATA_2	VDD_SDIO						SD_DATA2	I1/O/T	SPIHD	I/O/T	GPIO9	I/O/T	HS1_DATA2	I1/O/T	U1RXD	I1			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
29			SD_DATA_3	VDD_SDIO						SD_DATA3	I0/O/T	SPIWP	I/O/T	GPIO10	I/O/T	HS1_DATA3	I1/O/T	U1TXD	O			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
30			SD_CMD	VDD_SDIO						SD_CMD	I1/O/T	SPICS0	I/O/T	GPIO11	I/O/T	HS1_CMD	I1/O/T	U1RTS	O			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
31			SD_CLK	VDD_SDIO						SD_CLK	I0	SPICLK	I/O/T	GPIO6	I/O/T	HS1_CLK	O	U1CTS	I1			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
32			SD_DATA_0	VDD_SDIO						SD_DATA0	I1/O/T	SPIQ	I/O/T	GPIO7	I/O/T	HS1_DATA0	I1/O/T	U2RTS	O			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
33			SD_DATA_1	VDD_SDIO						SD_DATA1	I1/O/T	SPID	I/O/T	GPIO8	I/O/T	HS1_DATA1	I1/O/T	U2CTS	I1			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
34			GPIO5	VDD3P3_CPU						GPIO5	I/O/T	VSPICS0	I/O/T	GPIO5	I/O/T	HS1_DATA6	I1/O/T			EMAC_RX_CLK	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
35			GPIO18	VDD3P3_CPU						GPIO18	I/O/T	VSPICLK	I/O/T	GPIO18	I/O/T	HS1_DATA7	I1/O/T					2'd2	oe=0, ie=0	oe=0, ie=1
36			GPIO23	VDD3P3_CPU						GPIO23	I/O/T	VSPID	I/O/T	GPIO23	I/O/T	HS1_STROBE	I0					2'd2	oe=0, ie=0	oe=0, ie=1
37	VDD3P3_CPU			VDD3P3_CPU supply in																				
38			GPIO19	VDD3P3_CPU						GPIO19	I/O/T	VSPIQ	I/O/T	GPIO19	I/O/T	U0CTS	I1			EMAC_TXD0	O	2'd2	oe=0, ie=0	oe=0, ie=1
39			GPIO22	VDD3P3_CPU						GPIO22	I/O/T	VSPIWP	I/O/T	GPIO22	I/O/T	U0RTS	O			EMAC_TXD1	O	2'd2	oe=0, ie=0	oe=0, ie=1
40			U0RXD	VDD3P3_CPU						U0RXD	I1	CLK_OUT2	O	GPIO3	I/O/T							2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
41			U0TXD	VDD3P3_CPU						U0TXD	O	CLK_OUT3	O	GPIO1	I/O/T					EMAC_RXD2	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
42			GPIO21	VDD3P3_CPU						GPIO21	I/O/T	VSPIHD	I/O/T	GPIO21	I/O/T					EMAC_TX_EN	O	2'd2	oe=0, ie=0	oe=0, ie=1
43	VDDA			VDDA supply in																				
44		XTAL_N		VDDA																				
45		XTAL_P		VDDA																				
46	VDDA			VDDA supply in																				
47		CAP2		VDDA																				
48		CAP1		VDDA																				
Total Number	8	14	26																					

Notes:

- wpu: weak pull-up;
- wpd: weak pull-down;
- ie: input enable;
- oe: output enable;
- Please see *Table: Notes on ESP32 Pin Lists* for more information.（请参考表：管脚清单说明。）

APPENDIX B- PROGRAMMING

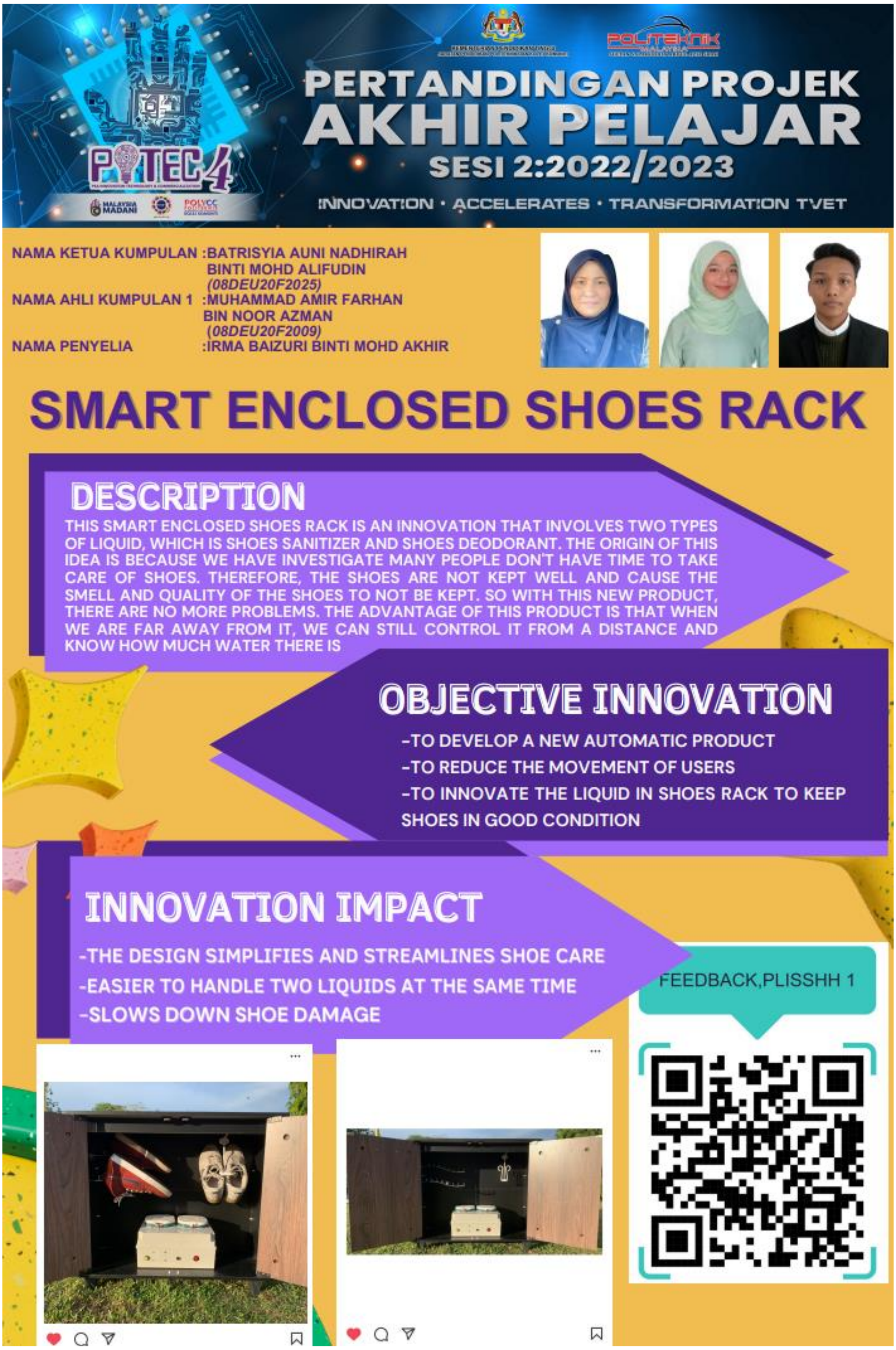
```
#define BLYNK_TEMPLATE_ID           "TMPL6EW1PUsWp"
#define BLYNK_TEMPLATE_NAME         "IoT"
#define BLYNK_AUTH_TOKEN            "JP8Fvuq2eLy6REUvoyfY_2gyVmStO5p0"

#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

char ssid[] = "abc";
char pass[] = "123456789";

const int trigPin1 = 27; //27
const int echoPin1 = 32; //32
long duration1;
int distance1;

const int trigPin2 = 14; //27
const int echoPin2 = 13; //32
long duration2;
int distance2;

int percentage_1 = 0;
int percentage_2 = 0;

const int buttonPin = 12;
int buttonState = 0;
const int buzzer =  2;
const int relay1 =  23;
const int relay2 =  25;

int state_1 = LOW;
unsigned long previous_time_1 = 0;
const long interval_1 = 5000;          // sanitizer - 12 saat = 12000

int state_free_1 = LOW;
unsigned long previous_time_free_1 = 0;
const long interval_free_1 = 10000;    // 2 minit = 120000

int state_2 = LOW;
unsigned long previous_time_2 = 0;
const long interval_2 = 5000;          // sanitizer - 12 saat = 12000
```

```
int state_free_2 = LOW;
unsigned long previous_time_free_2 = 0;
const long interval_free_2 = 10000;    // 10 minit = 600000

int state_3 = LOW;
unsigned long previous_time_3 = 0;
const long interval_3 = 3000;          // deodorant - 20 saat = 20000

int state_free_3 = LOW;
unsigned long previous_time_free_3 = 0;
const long interval_free_3 = 10000;    // 15 minit = 900000

int state_4 = LOW;
unsigned long previous_time_4 = 0;
const long interval_4 = 5000;          // sanitizer - 20 saat = 20000

int state_free_4 = LOW;
unsigned long previous_time_free_4 = 0;
const long interval_free_4 = 10000;    // 10 minit = 600000

int state_5 = LOW;
unsigned long previous_time_5 = 0;
const long interval_5 = 5000;          // shoe deodorant - 12 saat = 
12000

int state_free_5 = LOW;
unsigned long previous_time_free_5 = 0;
const long interval_free_5 = 5000;    // 2 minit = 120000

int state_6 = LOW;
unsigned long previous_time_6 = 0;
const long interval_6 = 4000;         // deodorant - 12 saat = 12000

int state_free_6 = LOW;
unsigned long previous_time_free_6 = 0;
const long interval_free_6 = 5000;    // 5 saat = 5000

int flag_1 = 0;

int starts = 0;

int flag_noty1 = 0;
int flag_noty2 = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  pinMode(buttonPin, INPUT_PULLUP);

  digitalWrite(relay1, HIGH);
  digitalWrite(relay2, HIGH);

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);

  digitalWrite(buzzer, HIGH);
  delay(100);
  digitalWrite(buzzer, LOW);
  delay(100);
  digitalWrite(buzzer, HIGH);
  delay(100);
  digitalWrite(buzzer, LOW);
}

void loop()
{
  Blynk.run();
  start_button();
  measure_1();
  measure_2();
  delay(150);
}

BLYNK_WRITE(V1)
{
  int data1 = param.asInt();
  digitalWrite(buzzer, HIGH);
  delay(300);
  digitalWrite(buzzer, LOW);
  if(data1 == 1)
  {
    starts = 1;
  }
}

void measure_1()
{
  digitalWrite(trigPin1, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin1, LOW);
  duration1 = pulseIn(echoPin1, HIGH);
  distance1= duration1*0.034/2;
  Serial.print("distance1: ");
  Serial.print(distance1);

  percentage_1 = map(distance1, 0, 12, 100, 0);

  Serial.print(" percentage_1: ");
  Serial.print(percentage_1);

  Blynk.virtualWrite(V5,percentage_1);

  if(percentage_1 < 10 && flag_noty1 == 0)
  {
    Blynk.logEvent("blynk_perfume", "alert");
    flag_noty1 = 1;
  }

  else if(percentage_1 > 10 && flag_noty1 == 1)
  {
    flag_noty1 = 0;
  }
}

void measure_2()
{
  digitalWrite(trigPin2, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin2, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin2, LOW);
  duration2 = pulseIn(echoPin2, HIGH);
  distance2= duration2*0.034/2;
  Serial.print("  distance2: ");
  Serial.print(distance2);

  percentage_2 = map(distance2, 0, 12, 100, 0);

  Serial.print(" percentage_2: ");
  Serial.println(percentage_2);

  Blynk.virtualWrite(V6,percentage_2);

  if(percentage_2 < 10 && flag_noty2 == 0)
  {
    Blynk.logEvent("blynk_senitize", "alert");
    flag_noty2 = 1;
  }
  else if(percentage_2 > 10 && flag_noty2 == 1)
  {
    flag_noty2 = 0;
  }

}

void start_button()
{
  buttonState = digitalRead(buttonPin);
  if (buttonState == LOW)
  {
    digitalWrite(buzzer, HIGH);
    delay(300);
    digitalWrite(buzzer, LOW);
    starts = 1;
  }

  if(starts == 1)
  {
    if(flag_1 == 0)
    {
      masa_1();
    }
    else if(flag_1 == 1)
    {
      masa_free_1();
    }
    else if(flag_1 == 2)
    {
      masa_2();
    }
    else if(flag_1 == 3)
    {
      masa_free_2();
    }
    else if(flag_1 == 4)
    {
      masa_3();
    }
    else if(flag_1 == 5)
    {
      masa_free_3();
    }
    else if(flag_1 == 6)
    {
      masa_4();
    }
    else if(flag_1 == 7)
    {
      masa_free_4();
    }
    else if(flag_1 == 8)
    {
      masa_5();
    }
    else if(flag_1 == 9)
    {
      masa_free_5();
    }
    else if(flag_1 == 10)
    {
      masa_6();
    }
    else if(flag_1 == 11)
    {
      masa_free_6();
    }

  }
}

void masa_1()
{
  unsigned long current_time_1 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_1);

  if (current_time_1 - previous_time_1 >= interval_1)
  {
    previous_time_1 = current_time_1;
    if (state_1 == LOW)
    {
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      state_1 = HIGH;
    }
    else
    {
      state_1 = LOW;
      current_time_1 = 0;
      flag_1 = 1;
    }
  }
```

```
}

void masa_free_1() //delay 2 minit
{
  unsigned long current_time_free_1 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_free_1);

  if (current_time_free_1 - previous_time_free_1 >= interval_free_1)
  {
    previous_time_free_1 = current_time_free_1;
    if (state_free_1 == LOW)
    {
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      delay(100);
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      state_free_1 = HIGH;
    }
    else
    {
      state_free_1 = LOW;
      previous_time_free_1 = 0;
      flag_1 = 2;
    }
  }
}

void masa_2()
{
  unsigned long current_time_2 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_2);

  if (current_time_2 - previous_time_2 >= interval_2)
  {
    previous_time_2 = current_time_2;
    if (state_2 == LOW)
    {
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      state_2 = HIGH;
    }
    else
    {
      state_2 = LOW;
      current_time_2 = 0;
      flag_1 = 3;
    }
  }
}

void masa_free_2() //delay 10 minit
{
  unsigned long current_time_free_2 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_free_2);

  if (current_time_free_2 - previous_time_free_2 >= interval_free_2)
  {
    previous_time_free_2 = current_time_free_2;
    if (state_free_2 == LOW)
    {
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      delay(100);
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      state_free_2 = HIGH;
    }
    else
    {
      state_free_2 = LOW;
      previous_time_free_2 = 0;
      flag_1 = 4;
    }
  }
}
```

```
void masa_3()
{
  unsigned long current_time_3 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_3);

  if (current_time_3 - previous_time_3 >= interval_3)
  {
    previous_time_3 = current_time_3;
    if (state_3 == LOW)
    {
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      state_3 = HIGH;
    }
    else
    {
      state_3 = LOW;
      current_time_3 = 0;
      flag_1 = 5;
    }
  }
}

void masa_free_3() //delay 10 minit
{
  unsigned long current_time_free_3 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_free_3);

  if (current_time_free_3 - previous_time_free_3 >= interval_free_3)
  {
    previous_time_free_3 = current_time_free_3;
    if (state_free_3 == LOW)
    {
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      delay(100);
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      state_free_3 = HIGH;
    }
    else
    {
      state_free_3 = LOW;
      previous_time_free_3 = 0;
      flag_1 = 6;
    }
  }
}

void masa_4()
{
  unsigned long current_time_4 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_4);

  if (current_time_4 - previous_time_4 >= interval_4)
  {
    previous_time_4 = current_time_4;
    if (state_4 == LOW)
    {
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      state_4 = HIGH;
    }
    else
    {
      state_4 = LOW;
      current_time_4 = 0;
      flag_1 = 7;
    }
  }
}

void masa_free_4() //delay 10 minit
{
  unsigned long current_time_free_4 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_free_4);

  if (current_time_free_4 - previous_time_free_4 >= interval_free_4)
  {
    previous_time_free_4 = current_time_free_4;
    if (state_free_4 == LOW)
    {
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      delay(100);
      digitalWrite(relay1, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay1, HIGH);
      digitalWrite(buzzer, LOW);
      state_free_4 = HIGH;
    }
    else
    {
      state_free_4 = LOW;
      previous_time_free_4 = 0;
      flag_1 = 8;
    }
  }
}

void masa_5()
{
  unsigned long current_time_5 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_5);

  if (current_time_5 - previous_time_5 >= interval_5)
  {
    previous_time_5 = current_time_5;
    if (state_5 == LOW)
    {
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      state_5 = HIGH;
    }
    else
    {
      state_5 = LOW;
      current_time_5 = 0;
      flag_1 = 9;
    }
  }
}

void masa_free_5() //delay 10 minit
{
  unsigned long current_time_free_5 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_free_5);

  if (current_time_free_5 - previous_time_free_5 >= interval_free_5)
  {
    previous_time_free_5 = current_time_free_5;
    if (state_free_5 == LOW)
    {
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      delay(100);
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      state_free_5 = HIGH;
    }
    else
    {
      state_free_5 = LOW;
      previous_time_free_5 = 0;
      flag_1 = 10;
    }
  }
}

void masa_6()
{
  unsigned long current_time_6 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_6);

  if (current_time_6 - previous_time_6 >= interval_6)
  {
    previous_time_6 = current_time_6;
    if (state_6 == LOW)
    {
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      state_6 = HIGH;
    }
    else
    {
      state_6 = LOW;
      current_time_6 = 0;
      flag_1 = 11;
    }
  }
}

void masa_free_6() //delay 10 minit
{
  unsigned long current_time_free_6 = millis();

  Serial.print("ct: ");
  Serial.println(current_time_free_6);

  if (current_time_free_6 - previous_time_free_6 >= interval_free_6)
  {
    previous_time_free_6 = current_time_free_6;
    if (state_free_6 == LOW)
    {
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      delay(100);
      digitalWrite(relay2, LOW);
      digitalWrite(buzzer, HIGH);
      delay(100);
      digitalWrite(relay2, HIGH);
      digitalWrite(buzzer, LOW);
      state_free_6 = HIGH;
    }
    else
    {
      state_free_6 = LOW;
      previous_time_free_6 = 0;
      flag_1 = 0;
      starts = 0;
      digitalWrite(buzzer, HIGH);
      delay(3000);
      digitalWrite(buzzer, LOW);
    }
  }
}
```

APPENDIX C- PROJECT MANUAL/PRODUCT CATALOGUE



PERTANDINGAN PROJEK AKHIR PELAJAR

SESI 2:2022/2023

INNOVATION · ACCELERATES · TRANSFORMATION TVET

NAMA KETUA KUMPULAN :BATRISYIA AUNI NADHIRAH BINTI MOHD ALIFUDIN *(08DEU20F2025)*

NAMA AHLI KUMPULAN 1 :MUHAMMAD AMIR FARHAN BIN NOOR AZMAN *(08DEU20F2009)*

NAMA PENYELIA :IRMA BAIZURI BINTI MOHD AKHIR

SMART ENCLOSED SHOES RACK

DESCRIPTION

THIS SMART ENCLOSED SHOES RACK IS AN INNOVATION THAT INVOLVES TWO TYPES OF LIQUID, WHICH IS SHOES SANITIZER AND SHOES DEODORANT. THE ORIGIN OF THIS IDEA IS BECAUSE WE HAVE INVESTIGATE MANY PEOPLE DON'T HAVE TIME TO TAKE CARE OF SHOES. THEREFORE, THE SHOES ARE NOT KEPT WELL AND CAUSE THE SMELL AND QUALITY OF THE SHOES TO NOT BE KEPT. SO WITH THIS NEW PRODUCT, THERE ARE NO MORE PROBLEMS. THE ADVANTAGE OF THIS PRODUCT IS THAT WHEN WE ARE FAR AWAY FROM IT, WE CAN STILL CONTROL IT FROM A DISTANCE AND KNOW HOW MUCH WATER THERE IS

OBJECTIVE INNOVATION

-TO DEVELOP A NEW AUTOMATIC PRODUCT

-TO REDUCE THE MOVEMENT OF USERS

-TO INNOVATE THE LIQUID IN SHOES RACK TO KEEP SHOES IN GOOD CONDITION

INNOVATION IMPACT

-THE DESIGN SIMPLIFIES AND STREAMLINES SHOE CARE

-EASIER TO HANDLE TWO LIQUIDS AT THE SAME TIME

-SLOWS DOWN SHOE DAMAGE

MANUAL PRODUCT

1) Enter and hang the shoes in the shoe rack in the space provided.
2) Press the switch on to open it.
3) Connect the smart phone to wifi.
4) Once connected, the buzzer will blink 2 times to indicate that the program has been successfully connected.
5) Press the yellow push button to start the sequence that has been set in the program.
6) Spray shoes sanitizer and shoes deodorant will be sprayed according to the sequence that has been set in the coding.
7) After finishing the set sequence, it will stop automatically.