# University of Moratuwa

## Faculty of Engineering

Department of Electronic and Telecommunication Engineering

**EN1190 - Engineering Design Project**

**The Effectors**

**Hybrid Smart Timer Plug**

Project Report


Group Members

| Index No. | Name |
|---|---|
| 230013A | D. H. Abeywarna |
| 230100M | H. A. D. C. Chandrakumara |
| 230229P | M. M. A. D. Hansindu |
| 230697V | J. A. H. R. Weerasinghe |

# Contents

# 1 Problem

## 1.1 Identification

Electrical and electronic appliances have become an integral part of modern day-to-day life. In such a world, the usage of the said appliances and their power consumption have to be sustainable. With the overly busy lives of people, overlooking a simple thing like turning a plug off has become commonplace. This could be harmless at times, but also quite dangerous at others.

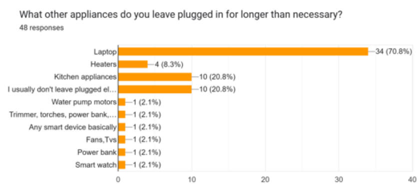
For example, keeping an iron or a heater on for too long could be harmful to both the appliance and the user, as these devices tend to consume a large amount of power. Another example, though not as harmful, is the overcharging of devices such as a phone or a laptop. This is a problem many people face, as proven in a survey we conducted. Charging the battery to 100% can cause strain on the battery over time. And while modern devices tend to possess solutions for this problem, whether by new charging methods or new batteries, most devices currently in use still have this issue.

When using equipment such as a water pump, agricultural watering systems, or even a fan, the ability to use it constrained to a specific time period can be valuable. Water pumps, for example, if left running for too long, can cause much damage. Therefore, we took convenience as well as safety as key factors to consider when designing our solution.

## 1.2 Justification

In order to justify this problem as one that people have experienced in real life, and to gauge their opinions, we conducted a survey. The following are some of the results we gained.


Figure 1: Survey Results 1



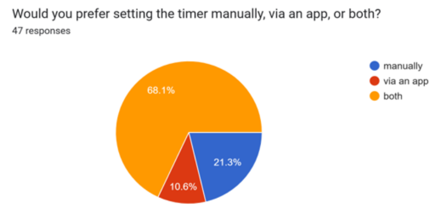




Figure 2: Survey Results 2

## 1.3 Solution

The solution we are suggesting is a plug that will run only for a time period specified by the user. This device could be used with various types of appliances, and thus will be a suitable solution for a plethora of similar problems.

The device can be controlled in two manners, manually through a dial on the plug, and through an app we developed. This will allow users to control the device in whatever method is convenient for them. We have also developed this product in such a way that multiple plugs that a user owns can be controlled through the same app.

# 2 Technical Feasibility

## 2.1 Hardware

The main logical component we used for our product was an ESP32 chip, through which we will be processing the inputs to control a relay. Below are the complete list of components we used.

- ESP32-WROOM-32 Chip
- AMS1117-3.3 V Regulator
- OLED Display
- Rotary Encoder
- 230-5 V Converter
- Relay Module
- AC Power Socket

We designed a PCB that will ensure the proper connection and functioning of the above components, and outsourced the PCB printing process to JLCPCB. The design of the PCB will be illustrated in upcoming pages.

## 2.2 Software

For the app development, we used Flutter as a platform, and VSCode as the IDE. The code for the ESP module was written in C++ with both the Arduino IDE and the PlatformIO extension for VSCode.

To make the connection between the module and the app, we decided to use Google Firebase, which offered the necessary capabilities we required for our project. The existence of such an established backend cloud platform was useful for us since we could use its services directly.

---

The enclosure was designed with Solidworks and printed through Rysera 3D Printing.

---

Therefore, the feasibility of our product was ensured with existing technologies, services, and components aiding in our innovative project.

# 3 Product Architecture

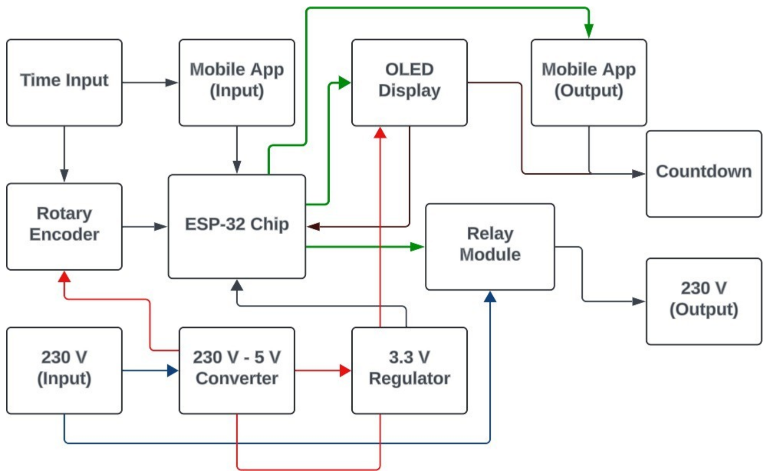



Figure 3: Product Architecture

## 3.1 ESP-32 Chip


Figure 4: ESP-32-WROOM-32

The ESP Module is the main logic component of our product. It handles the inputs and the outputs, as well as the functioning of the timer.

## 3.2 Rotary Encoder

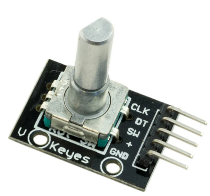
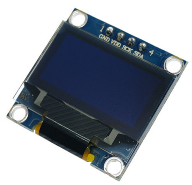



Figure 5: Rotary Encoder

The rotary encoder is the method of input when using the plug manually. The abilities to both rotate and press the encoder's button led us to choose this as a suitable input device.

## 3.3 OLED Display



The OLED display is how we output the countdown, the various menus, and the connection status. We are using a 128x64 display, which proved adequate for our needs.

## 3.4 3.3 V Regulator


The 3.3 V regulator is for regulating the power supply to the ESP chip. The chip works on 3.3 V.

## 3.5 230V - 5V Converter

This converter was used to step down the mains current of 230V to 5V DC, to power up our device.

## 3.6 Relay Module


The power connection to the appliance is supplied through the relay module, and depending on the status of the timer, the module will either allow current to flow, or stop the current. It essentially acts as a switch we can control with a signal.

## 3.7 Mobile App

The app was built using Flutter as a platform and Dart as the language. We currently have the following features.

- Set timer
- Pause/Resume Time
- Reset timer
- User login and registration
- Multiple plugs per user

The feature of logging in adds a layer of security so that only the user can access the plugs they own. These details will be exchanged with the plug during initial setup.

Throughout the process, the countdown status and other commands will be updated in real time between the ESP module and the app, giving users a seamless experience.

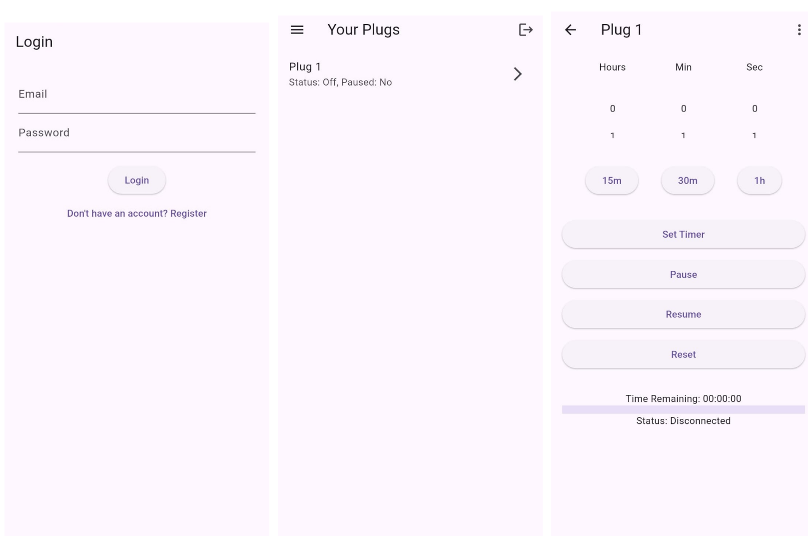



Figure 6: App Screens

# 4 Challenges

Throughout the entire process of building our product, we ran into various problems, and either overcame them, or took a different approach.

## 4.1 Hardware Problems

One such problem was that while the ESP module works on 3.3 V voltage, the relay module needs 5 V to operate. But since the ESP only outputs 3.3 V, using a signal directly from the module was impossible. Therefore, we had to design an amplifier which would convert this 3.3 V into 5 V, before the signal was given to the relay module.

Another was transferring the code to the ESP chip, or 'flashing' as it's called. For this, we had to send the code through a USB cable, but since we were using only the ESP chip, there was no interface to directly upload the code. Therefore, we had to use a USB to TTL adapter, to convert the USB signal to Serial.

## 4.2 Software Problems

While making both the code for the app, and the ESP, since both of them needed to update their status in real time, a unified flow had to be established for both modes of input. The logical structure of this was complex, and we resorted to using multiple functions within functions to build the flow.

Also, at first, we experienced a noticeable delay in the communication through the Firebase database. This was later overcome by further improvements to the code.

# 5 Enclosure

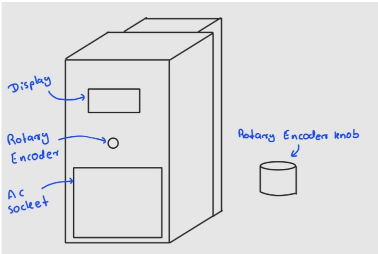



Figure 7: Initial Design

The enclosure was designed with several things in mind. The ease of use, lack of obstructions, and a positive user experience were among our concerns. In the end, we arrived at a design that, as we felt, fit all these.

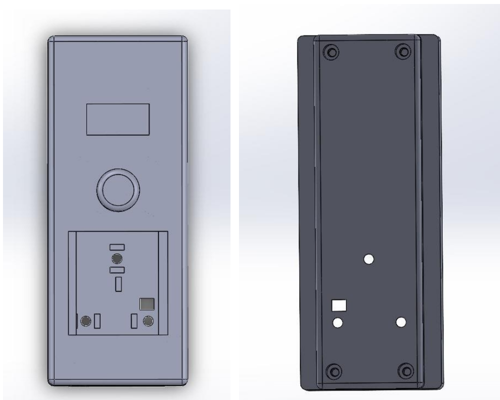


Figure 8: Left: Front View, Right: Back View

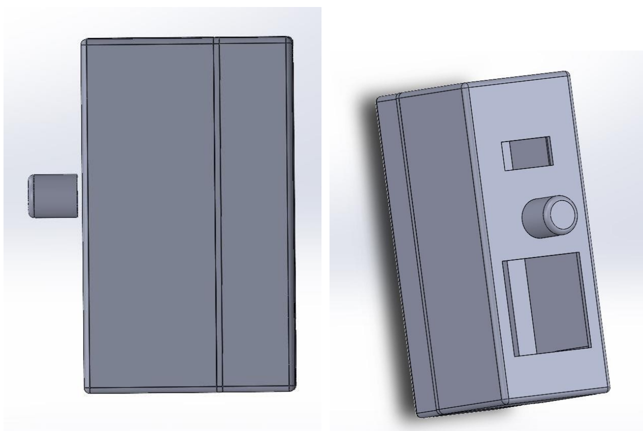


Figure 9: Left: Side View, Right: 3D View

# 6 PCB Design

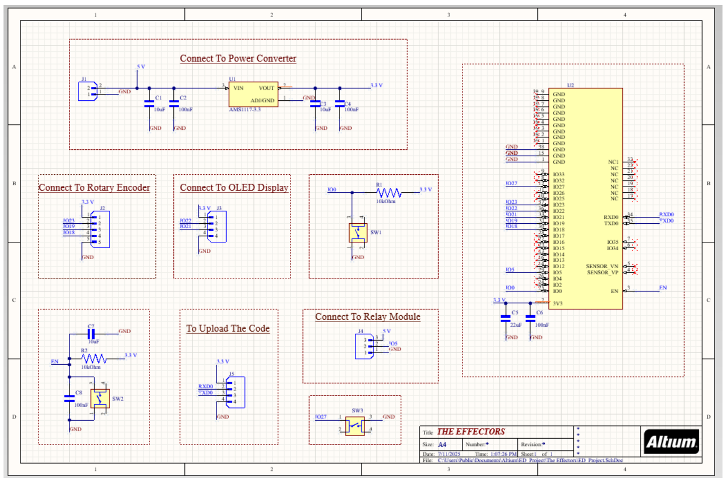



Figure 10: Schematic Design

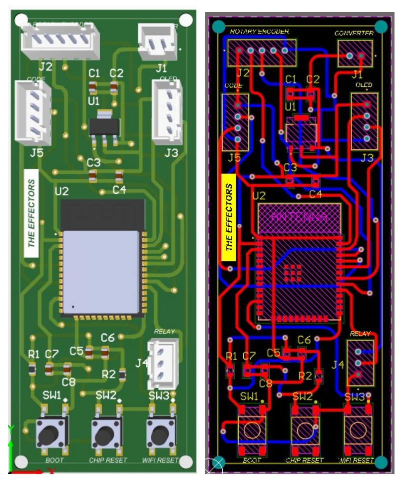



Figure 11: PCB Design

# 7 Final Product


# 8 Task Allocation

| Index No. | Name | Task |
|---|---|---|
| 230013A | D. H. Abeywarna | Software development |
| 230100M | H. A. D. C. Chandrakumara | Enclosure design |
| 230229P | M. M. A. D. Hansindu | PCB design |
| 230697V | J. A. H. R. Weerasinghe | Circuit design and Assembly |

## 9 Project Budget

| | |
|---|---|
| ESP-32 Module | 980.00 |
| 230V-5V Converter | 800.00 |
| AMS1117-3.3 V | 20.00 |
| USB to UART Converter | 550.00 |
| JTS Connectors | 100.00 |
| Capacitors and Resistors | 200.00 |
| PCB | 300.00 |
| Enclosure | 3500.00 |
| AC Power Socket | 70.00 |
| OLED Display | 500.00 |
| Rotary Encoder | 120.00 |
| Relay Module | 120.00 |
| Fuse | 5.00 |
| **Total** | **7265.00** |

Our initial plan was to set the price at around Rs. 5000-6000. But most of these expenses could be further reduced. For example, one USB to UART converter would be enough to upload the code to multiple ESP modules. And the price for the enclosure could be significantly reduced by moving on from 3D printing, and using our own mold. Also, the 230V-5V converter could be replaced with something less expensive. With all these considered, we expect the final budget for a market ready product to be around Rs. 3000.00. So, by setting the price to Rs.6000.00, we might get close to a 100% profit.

## 10 Marketing and Sales

Our primary target markets include households, student accommodations, and small-scale industries where timed control of electrical appliances can significantly enhance both safety and energy efficiency. We identified a clear demand among users of water pumps, irons, chargers, and other devices that either lack built-in timer functions or have malfunctioning timers, despite requiring precise control.

Additionally, small-scale agricultural and industrial settings - such as farms using lighting or watering systems - often lack access to expensive smart home or industrial automation solutions. For these users, our Hybrid Timer Plug provides an affordable and practical alternative.

To effectively reach our target audience, we plan to market the product through social media advertising and form partnerships with local electronic retailers, enabling them to educate customers about the product's benefits. We also intend to participate in technological innovation competitions to increase visibility and gain wider public acknowledgment. Our sales strategy will begin with direct and reseller-based distribution at the local level, with plans to expand into broader smart home and industrial markets as we continue to enhance the product's features.

## 11 Future Improvements

As a next step, we plan to integrate a suitable battery backup system to ensure offline operation during power outages. This will allow the device to save the remaining timer duration and automatically resume the task once power is restored, improving reliability. Another enhancement is the addition of trickle charging detection. This feature will monitor the charging behavior of devices such as smartphones and laptops, cutting off power when full charge is reached - even if the timer is still running - and optionally providing alerts to the user either through a sound from the plug itself or via the mobile app, depending on the situation and user preferences.

We also aim to introduce AI-based scheduling, where users can simply select a regular task, and the system will intelligently suggest appropriate timer settings based on usage patterns. This will make the device even more intuitive and user-friendly. Additionally, the mobile app can be upgraded to include energy usage analytics, enabling users to track their power consumption and make informed decisions. Finally, integrating voice assistant support such as Google Assistant or Alexa will provide hands-free control, making the plug more accessible and convenient to use.