REAL TIME SYSTEM AND INTERNET OF THINGS FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA

LOCKBOX SECURE+

GROUP B4

Bisma Alif Alghifari	**2106721402**
Daffa Anis Fahrizi	**2106731365**
Ibrahim Rijal	**2106633323**
Muhammad Najih Aflah	**2106653880**

PREFACE

In the ever-evolving landscape of security, the LockBox Secure+ emerges as a beacon of innovation, revolutionizing the way we safeguard our most prized possessions. In an era where the intersection of technology and security has become pivotal, this smart safe is not merely a storage solution but a guardian of trust. The LockBox Secure+ is a testament to the relentless pursuit of creating a seamless blend between cutting-edge technology and an aesthetic, user-friendly design.

Imagine a secure enclave where your valuables find sanctuary, guarded not just by impenetrable material but by a symphony of smart features. With its intuitive Numpad Pad interface, the LockBox Secure+ seamlessly combines ease of access with impenetrable security. However, its prowess extends beyond; equipped with an integrated camera system, it captures and sends images of anyone attempting or succeeding in accessing the safe directly to the owner via Telegram. Adding another layer of protection, a sensitive vibration sensor triggers an image capture and activates a buzzer, ensuring that any unauthorized movement is met with swift detection and alert.

Depok, December 08, 2023

Group B4

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

The realm of traditional safes is marked by a range of security challenges that demand urgent attention. Existing safes, often reliant on antiquated key-based or basic lock mechanisms, exhibit vulnerabilities in the face of contemporary security threats. The lack of advanced features, such as real-time monitoring and immediate alerts, creates a significant void in the ability to safeguard valuable possessions effectively. The absence of proactive security measures and the inability to provide visual evidence during unauthorized access attempts leave users without the means to respond promptly to potential threats[1].

Furthermore, the limitations of conventional safes are magnified in scenarios where timely owner notification is paramount following unauthorized access attempts. The absence of smart connectivity and monitoring capabilities hinders the establishment of a robust security framework. As security threats undergo constant evolution, the inadequacies of traditional safes become increasingly pronounced, necessitating a modern and intelligent safe solution that addresses these inherent vulnerabilities comprehensively[1].

In essence, the prevailing model of traditional safes falls short in delivering a holistic security experience. Recognizing these shortcomings, the LockBox Secure+ initiative aims to usher in a new era of safe technology by introducing innovative features. However, a detailed exploration of these features and the proposed solution will be explained in the next section.

1.2 PROPOSED SOLUTION

The LockBox Secure+ aims to address these security challenges by introducing innovative features. By incorporating a smart lock with a Numpad Pad, the safe ensures a higher level of security compared to traditional key-based or combination locks. The built-in camera serves as a crucial component, capturing images of individuals attempting or successfully gaining access to the safe. These images are then promptly transmitted to the owner through a secure communication channel like Telegram[2][3].

In addition to the camera, the inclusion of motion sensors adds another layer of security. These sensors detect any movement or tampering with the safe and trigger instant alerts. Simultaneously, a buzzer provides an audible notification, enhancing the overall security system's effectiveness. This multifaceted approach to security not only deters potential intruders but also empowers safe owners with real-time information and the ability to respond promptly to security threats[2][3].

In summary, the LockBox Secure+ addresses the shortcomings of traditional safes by providing a comprehensive, technologically advanced security solution. By leveraging smart technologies and real-time communication capabilities, this safe redefines the standards for safeguarding valuables in both residential and commercial settings, offering peace of mind and proactive security measures to users[3].

1.3 ACCEPTANCE CRITERIA

The acceptance criteria of this project are as follows:

1. The Numpad Pad interface must effectively control the locking and unlocking mechanisms of the safe.
2. The vibration sensor must detect any unauthorized movements or tampering with the safe.
3. Upon detection, the sensor should trigger the capture of images and activate the audible alert system (buzzer).
4. Images should be promptly and securely transmitted to the safe owner via the Telegram messaging platform.
5. If possible, add a pairing feature so that the ESP connection credentials settings are not hardcoded.

1.4 ROLES AND RESPONSIBILITIES

The roles and responsibilities assigned to the group members are as follows:

Roles	Responsibilities	Person
Role 1	Make the code for the Accelerometer - Buzzer and the main ESP32 code, Hardware Soldering and assembly of the overall components. Writing Report.	Ibrahim Rijal
Role 2	Make the code for the Keypad and Servo, Creating the user manual, Creating the Presentation (PPT), Creating the README. Writing Report.	M. Najih Aflah
Role 3	Make the code for the ESP32 Cam, and the ESP-NOW connection. Code for overall components integration, Integration Testing.	Daffa Anis Fahrizi
Role 4	Hardware Soldering and assembly of the overall components, Code Documentation, Testing final product Functionality. Writing Report.	Bisma Alif Alghifari

Table 1.4.1 Roles and Responsibilities

1.5 TIMELINE AND MILESTONES

LockBox Secure+ GanttChart														
		WEEK 1							WEEK 2					
No	Task Description	27	28	29	30	1	2	3	4	5	6	7	8	9	10
1	Hardware Design completion														
2	Software Development														
3	Integration and Testing of Hardware and Software														
4	Final Product Assembly and Testing														

Table 1.5.1 Gantt Chart

CHAPTER 2

IMPLEMENTATION

2.1 HARDWARE DESIGN AND SCHEMATIC

The implementation phase of the project commences with a meticulous exploration of the hardware design and schematic, pivotal components defining the operational dynamics of the envisioned safety box. Employing an array of sophisticated tools and technologies, this section unveils the intricacies of the hardware framework designed to ensure the security and functionality of the safety box system. Each selected tool serves a strategic purpose, collectively contributing to the creation of a robust and reliable mechanism. The following tools are utilized to bring the aspect above to fruition:

a) Safety Box Simulation Box

A specialized box is employed for simulating the real-world environment of the safety box. This aids in practical testing and validation of the implemented features.

b) Esp32

The Esp32 serves as a crucial component, facilitating the connection to the buzzer and contributing to the overall functionality of the safety box.

c) ESP Cam

Employed for capturing images of individuals attempting or successfully accessing the safety box. The images are then sent to the owner via the Telegram platform, enhancing security measures.

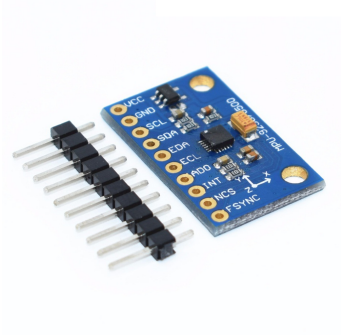
d) Num Pad

Integrated into the design to serve as the mechanism for accessing the safety box. It provides a secure and user-friendly input method for users.

e) Buzzer

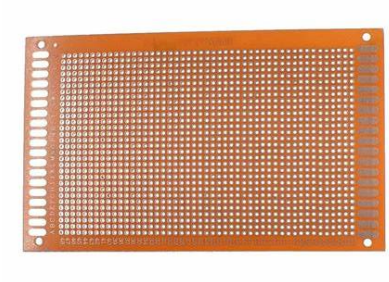
Utilized for alerting the user in response to specific events, such as unauthorized access attempts or detected motion.

f) Accelerometer



Functions as a motion sensor, detecting any movement near the safety box and triggering signals sent to the buzzer for immediate notification.

g) Protoboard/PCB Dot Matrix (2)



These components play a crucial role in connecting various hardware elements, ensuring a seamless interaction between them. The protoboard and PCB Dot Matrix contribute to the overall stability and reliability of the hardware setup.

h) Servo

Functions as a lock mechanism, controlled by the Esp32. This servo-driven mechanism adds an extra layer of security to the safety box.

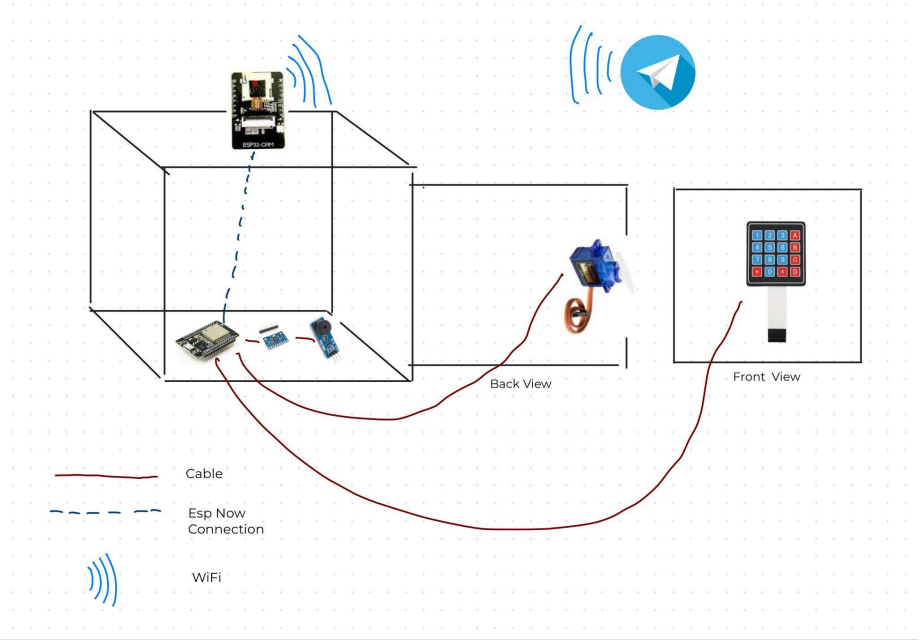


Fig 2.1.1. Hardware Design

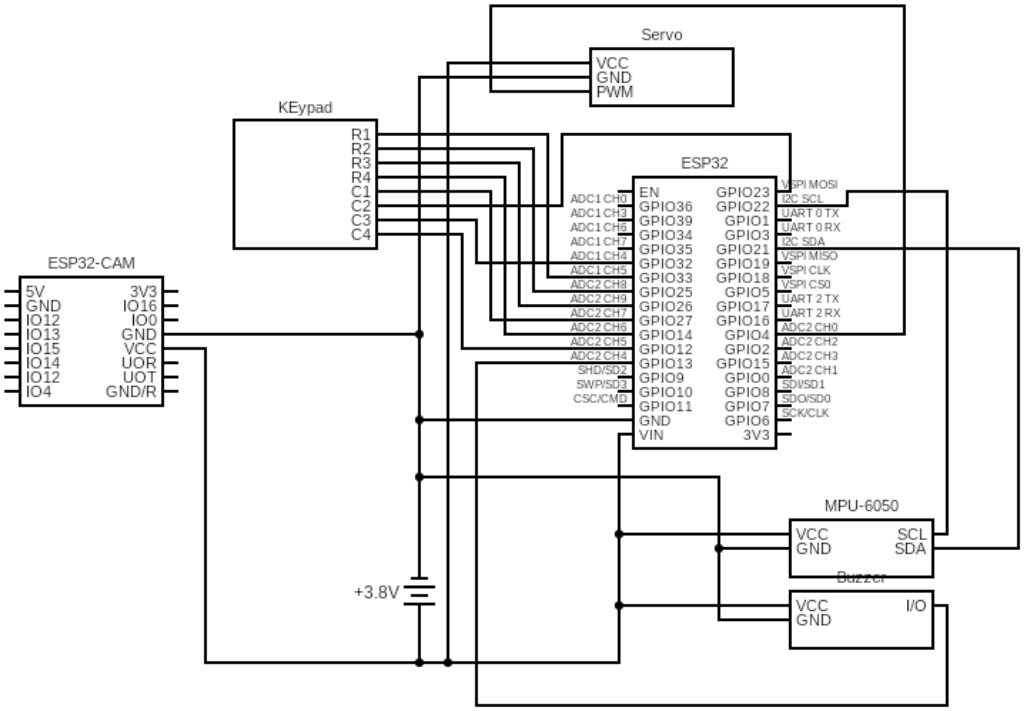


Fig 2.1.2. Hardware Schematic

2.2 SOFTWARE DEVELOPMENT

In the software development phase, our approach is a bit different – we utilize the Telegram platform as the backbone of our system, acting as both the user interface and the monitoring system. Rather than developing a new standalone software, we creatively leverage the capabilities of Telegram to seamlessly integrate it with the hardware components of the safety box.

At the heart of this integration lie two crucial components: the BOT Token and the CHAT ID. The Bot Token is a unique key provided by Telegram when you create a bot using the BotFather. This token serves as a form of authentication, allowing the Telegram API to recognize and authorize your bot to interact with Telegram servers. It's essentially a secret code that proves your bot's legitimacy[4].

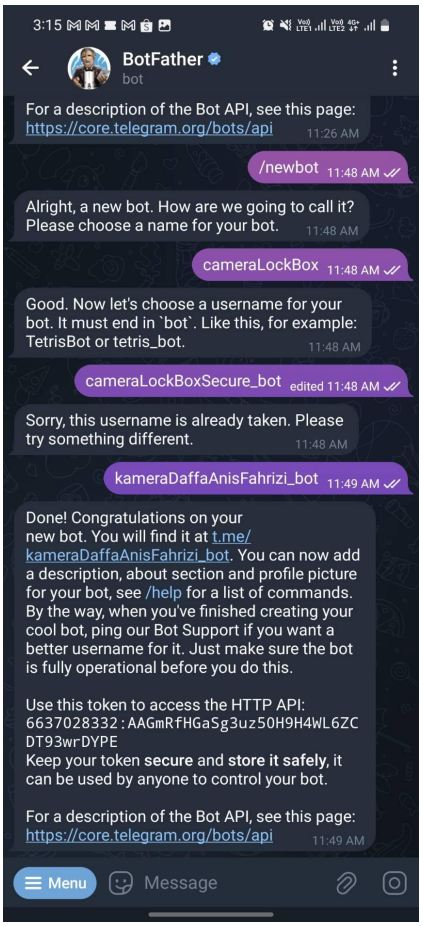


Fig 2.2.1. BotToken Creation

On the other hand, the Chat ID is a numerical identifier assigned to each individual or group chat on Telegram. It acts as an address, specifying the destination for messages or

media you want to send. To obtain the Chat ID, you typically initiate a conversation with your bot and then use the Bot API to retrieve information about the received message, including the Chat ID[4].

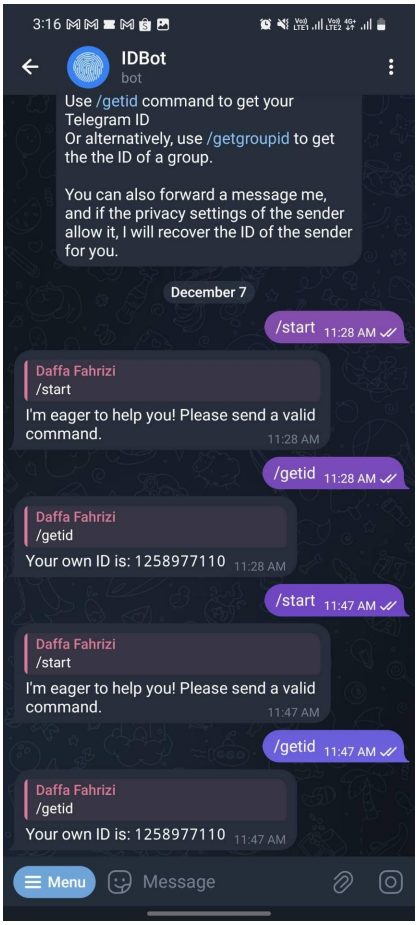


Fig 2.2.2. ChatID Creation

To set up a connection between my ESPcam and Telegram for sending photos, we started by creating a Telegram bot using the BotFather. This process generated a unique Bot Token that functions as a key for authentication. Following that, we initiated a conversation with my bot on Telegram to obtain the Chat ID, a specific identifier for the chat or user where we intend to send messages. we retrieved this Chat ID by using the Bot API to gather information about a message we sent[4][5].

With the Bot Token and Chat ID in hand, we proceeded to program my ESPcam using the Arduino IDE. we embedded these identifiers into my code, allowing the ESPcam to send messages and photos to the designated Telegram chat. To facilitate photo capture, we integrated the necessary code for the camera module on the ESPcam[5].

```
// Initialize Telegram BOT
String BOTtoken = "6637028332:AAGmRfHGaSg3uz50H9H4WL6ZCDT93wrDYPE";  // your Bot Token (Get from Botfather)

// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
String CHAT_ID = "1258977110";
```

Fig 2.2.3. Code Integration Example

When a photo is taken, we utilized the Telegram API to send the image to the specified Chat ID using the Bot Token. Here are some of the temporary results:

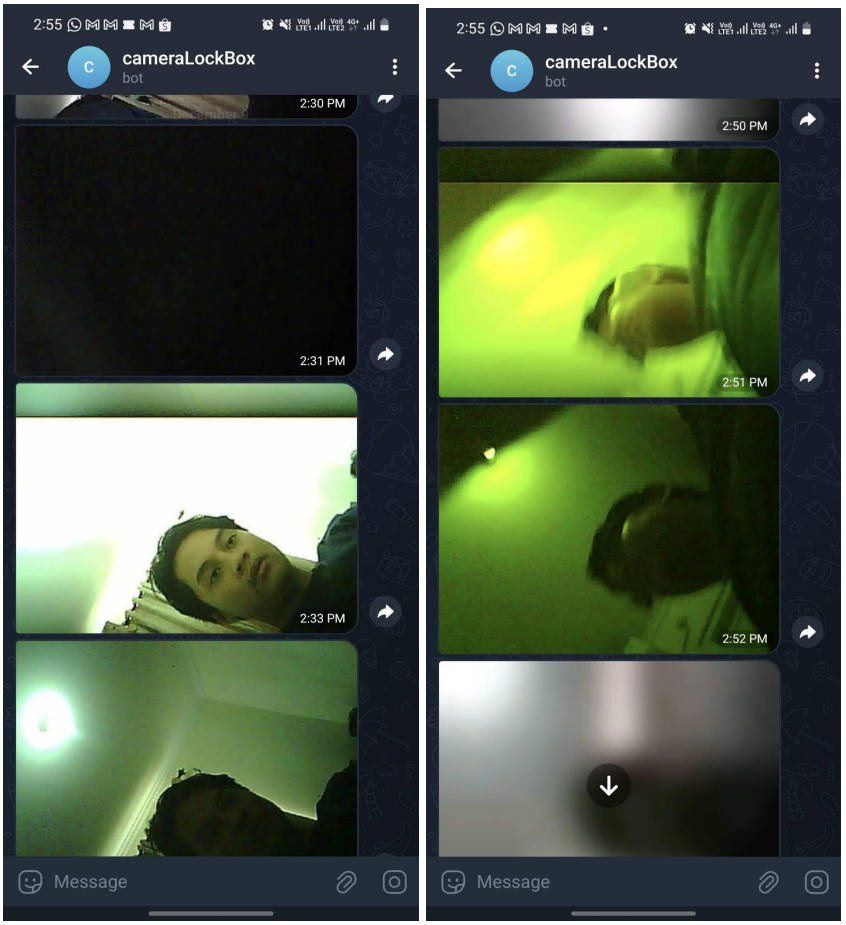


Fig 2.2.4. Temporary Telegram Results

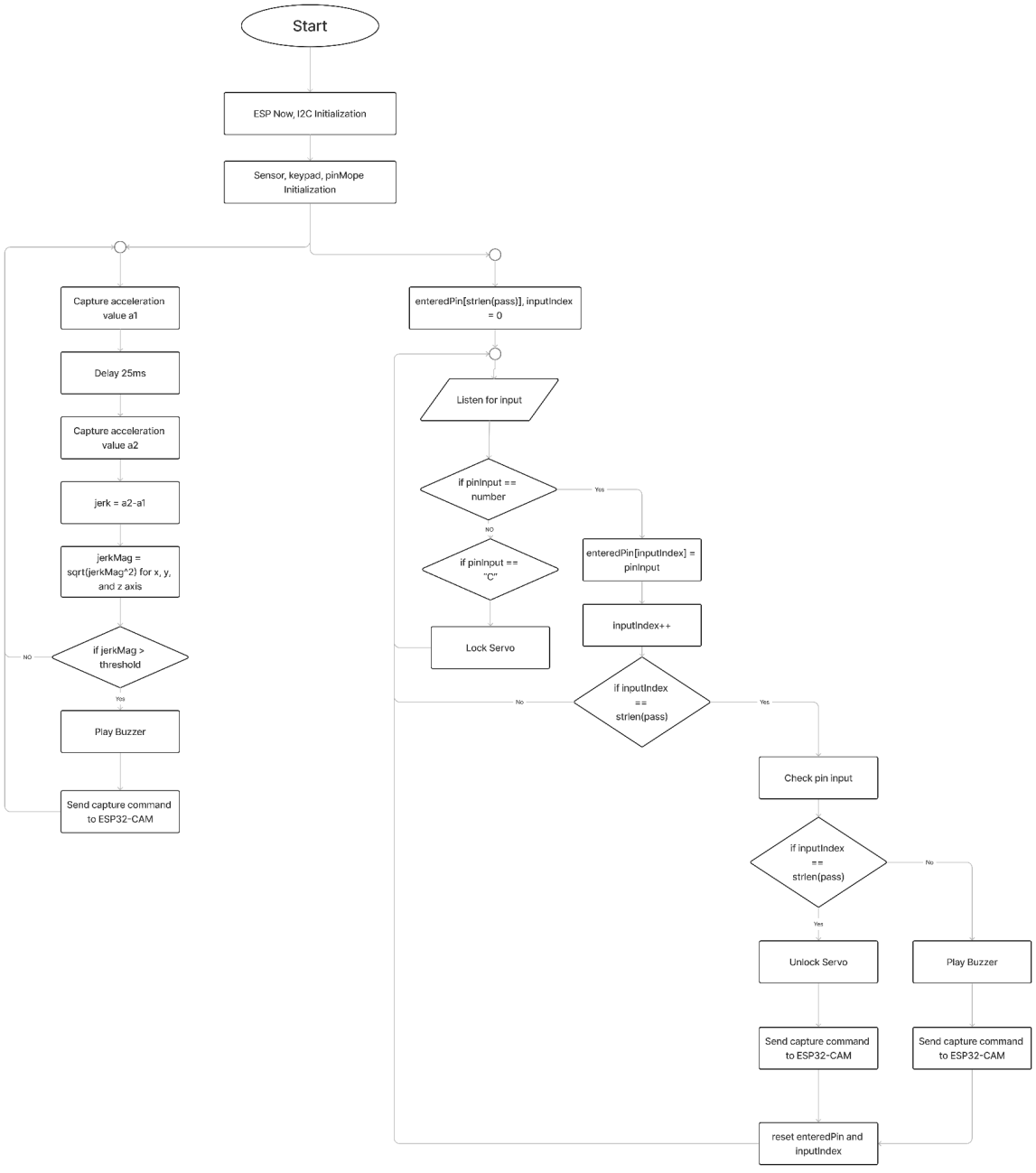


Fig 2.2.5. Flowchart ESP32 (ESP Now)

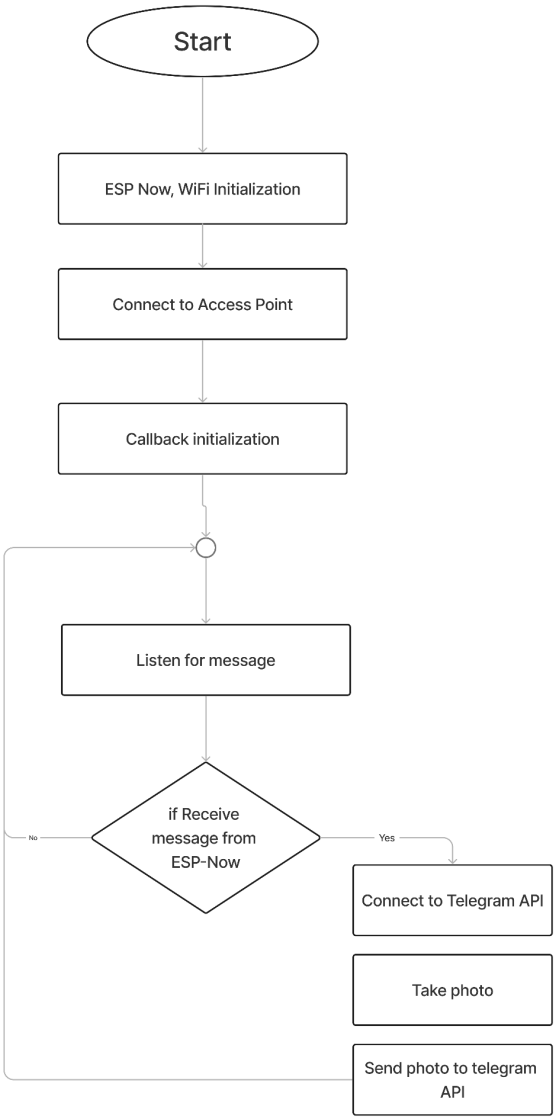


Fig 2.2.5. Flowchart Esp-Cam

2.3 HARDWARE AND SOFTWARE INTEGRATION

In the process of integrating the software and hardware, our team has meticulously designed and implemented code for both the ESP32 and ESPCAM devices. On the ESPCAM side, the Telegram communication functionality has been seamlessly woven into the code using libraries such as WiFi, UniversalTelegramBot, and ArduinoJson. The configuration of the camera is a critical aspect, and we've meticulously set parameters such as frame size and JPEG quality to optimize photo transmission to Telegram. Additionally, the integration

includes error-handling mechanisms to gracefully manage potential issues during the photo capture and transmission process[5].

Moving to the ESP32, which serves as the brains of the operation, the integration incorporates sophisticated functionalities. The MPU6050 sensor facilitates motion detection, a pivotal feature triggering the ESPCAM to capture photos when suspicious activity is detected. The use of FreeRTOS tasks ensures efficient multitasking, allowing simultaneous execution of tasks like motion detection, keypad input monitoring, and PIN verification[5][6].

Speaking of the keypad, it plays a central role in the system, enabling users to input PINs for verification. The keypad integration involves a task-driven approach, providing responsiveness to user inputs. PIN verification is intricately handled, and successful entries trigger actions such as unlocking the safety box and initiating a photo capture event on the ESPCAM[6].

The servo locking mechanism, another hardware component controlled by the ESP32, has been seamlessly integrated into the system. The servo motor, serving as the actuator for the locking mechanism, responds dynamically to correct PIN entries, ensuring a secure and controlled unlocking process. This not only enhances the security of the LockBox Secure+ but also adds a layer of user interaction[7].

Audio feedback is provided through a buzzer, enhancing the user interface by providing confirmation during PIN entry and system events. The choice of ESP-NOW for communication between ESP32 and ESPCAM ensures low-latency and reliable data exchange, contributing to the overall responsiveness and effectiveness of the system[8].

The integration is designed to be scalable, allowing for future enhancements or additions to the system. By encapsulating functionalities into well-defined tasks and modules, we've created a flexible and extensible architecture that accommodates potential expansions seamlessly. This integration represents a synergy of software and hardware, working harmoniously to create a robust, responsive, and user-friendly IoT security system for the LockBox Secure+ project.

CHAPTER 3

TESTING AND EVALUATION

3.1 TESTING

Testing Procedures:

Motion Detection Testing:

- Verify the MPU6050 sensor functionality by simulating motions near the safe to trigger detection.
- Validate the accuracy of motion alerts and the reliability of capturing images.

Keypad Input and Servo Testing:

- Test the Keypad input functionality by entering valid and invalid codes.
- Assess the servo's response to valid inputs and its reliability in locking and unlocking the safe.

ESP Connection Testing:

- Confirm the connectivity between the ESP32 devices to ensure proper communication.
- Validate the reliability of data transmission using ESP Now connection.

WiFi Connection Testing:

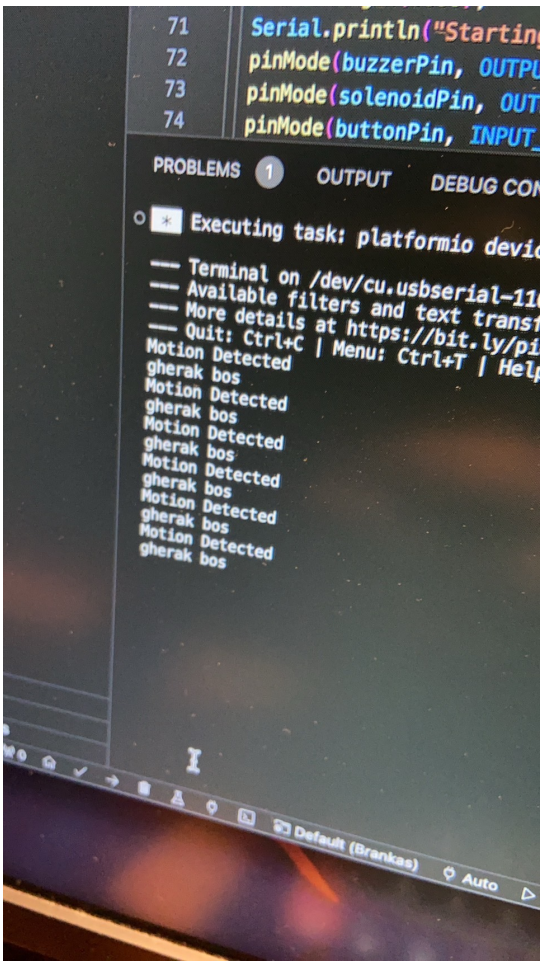
- Test the WiFi connectivity of the ESP32 Cam to ensure consistent and reliable internet access.
- Verify the stability of the connection for transmitting images via Telegram.

Telegram Integration Testing:

- Test the functionality of sending images and messages to the designated Telegram chat.
- Validate the reception and promptness of notifications on the owner's device.

3.2 RESULT

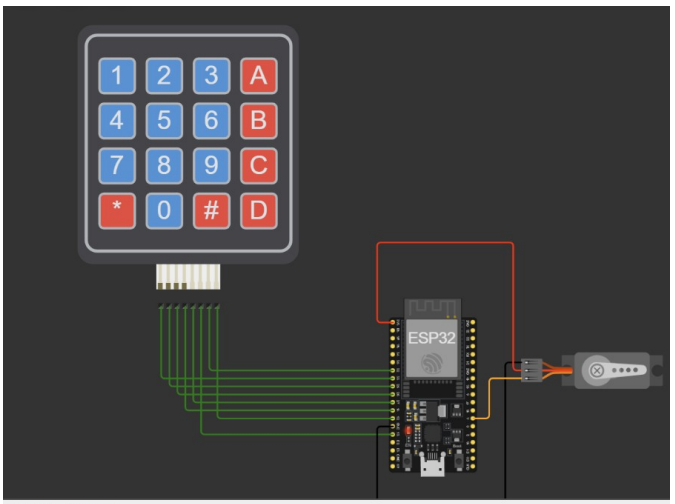
a. Testing for the motion detection



Motion detection is done by calculating what is called by “jerk”. The MPU (Motion Processing Unit) is utilized to obtain initial and final acceleration values along the X, Y, and Z axes. These values are converted to floating-point format and normalized for accurate calculations. The jerk, indicating the abrupt changes in acceleration, is computed for each axis by subtracting the initial acceleration from the final acceleration. The magnitude of the jerk is then determined using the Euclidean norm, providing a comprehensive measure of motion intensity. If the jerk magnitude surpasses a predefined threshold of 1.2, the system identifies significant motion.

b. Testing for the Keypad input and the servo

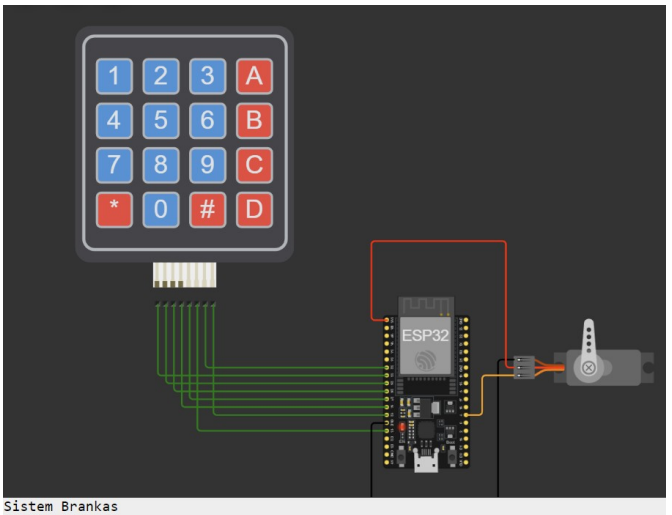
- Testing if the Pin is correct




```
Sistem Brankas
Silahkan Masukkan PIN
1
PIN yang dimasukkan: 1
2
PIN yang dimasukkan: 12
3
PIN yang dimasukkan: 123
4
PIN yang dimasukkan: 1234

PIN Benar! Membuka Brankas
```

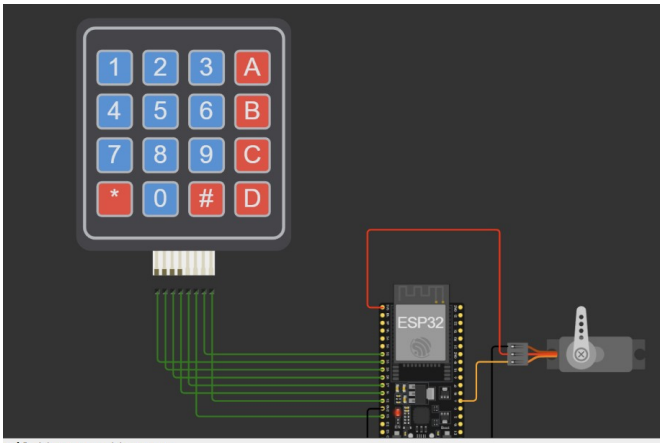
- Testing if the Pin is False




```
Sistem Brankas
Silahkan Masukkan PIN
2
PIN yang dimasukkan: 2
3
PIN yang dimasukkan: 23
6
PIN yang dimasukkan: 236
5
PIN yang dimasukkan: 2365

PIN Salah! Brankas Tetap Tertutup
```

- Testing to close back the safe



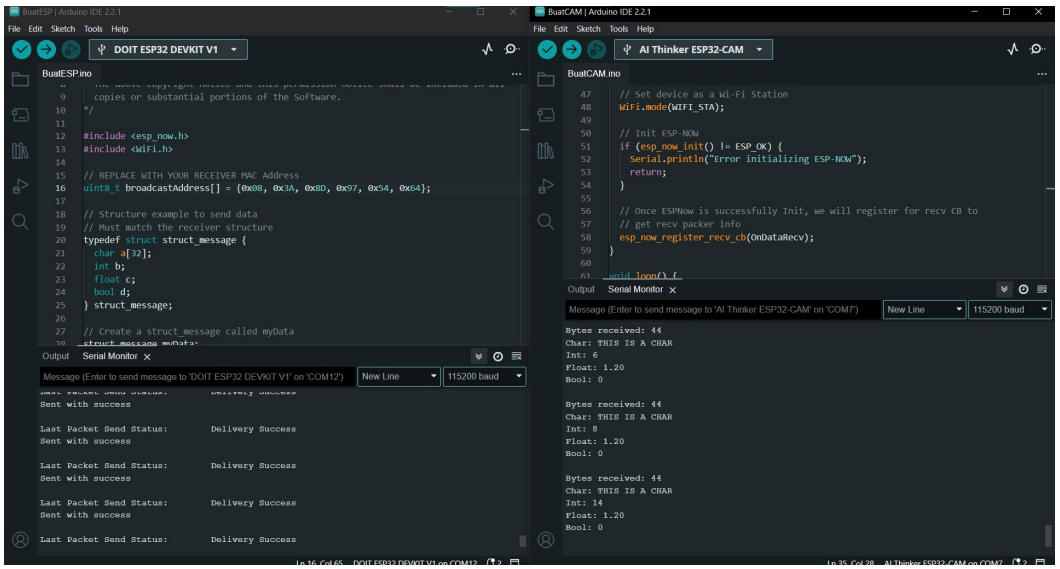

```
Silahkan Masukkan PIN
1
PIN yang dimasukkan: 1
2
PIN yang dimasukkan: 12
3
PIN yang dimasukkan: 123
4
PIN yang dimasukkan: 1234

PIN Benar! Membuka Brankas
C

Kembali ke posisi awal / Menutup Brankas
```

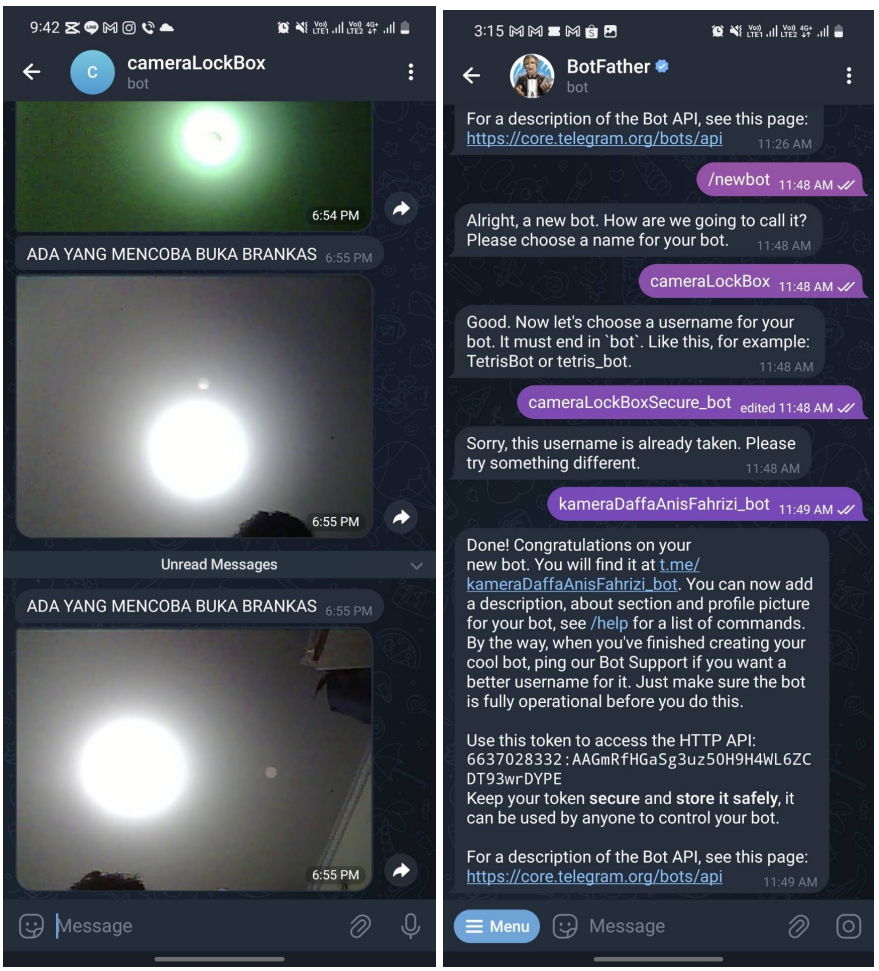
The Keypad and Servo testing process validates the core functionality of the LockBox Secure+, ensuring accurate PIN input and servo activation for secure access to valuables. The system's responsiveness to both correct and incorrect entries showcases its reliability and security measures. Adjustments or fine-tuning may be required based on the observed outcomes to enhance user experience and security.

c. ESP Connection Testing



ESP32 devices demonstrated consistent and reliable communication through the ESP Now connection

d. Telegram Integration Testing



Telegram integration was successful, allowing the system to send images and notifications promptly

Fig 3. Testing Result

Fig 4. Testing Result

Fig 5. Testing Result

Fig 6. Testing Result

Donec at iaculis leo. Integer congue sed lacus suscipit iaculis. Nulla a augue ut sapien rutrum consectetur. Sed ac dignissim lorem. Maecenas hendrerit nisl a metus posuere, vel vehicula metus eleifend. Mauris blandit, dolor nec malesuada tempor, purus nibh aliquet

nibh, at faucibus leo felis a nisi. Donec pharetra leo risus, in vestibulum dui laoreet in. Nulla facilisi. Etiam nec consequat justo. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam erat volutpat. Etiam pharetra eleifend hendrerit.

3.3 EVALUATION

In the evaluation of our project, several key areas have been identified for improvement. Firstly, addressing the issue with power consistency is paramount. The current use of an old iPhone battery has resulted in occasional instability, causing disruptions in the normal functioning of the device, particularly affecting components like the buzzer and accelerometer. To enhance reliability, we should explore more consistent power sources to ensure continuous and stable operation.

Secondly, in terms of the locking mechanism, our team has recognized the limitations of using a servo. To establish a more secure and robust locking system, we propose replacing the servo with a solenoid accompanied by a spring for additional safety measures. This adjustment eliminates the need for a hook extension, simplifying the locking process and enhancing the overall security of the system.

Furthermore, we envision incorporating new features for remote management. One such feature involves the ability to lock and unlock the safety box remotely through Telegram. Additionally, a status monitoring system will be implemented to provide real-time feedback on the lock condition. This enhancement not only adds convenience but also improves the overall accessibility and control of the safety box.

Lastly, to enhance the user experience and surveillance capabilities, a feature enabling remote viewing of the current condition will be integrated. This involves remotely capturing images from the ESPcam, providing users with visual insights into the surroundings of the safety box. These proposed improvements aim to address functionality, security, and user interaction aspects, elevating the overall effectiveness and user satisfaction with our project.

CHAPTER 4

CONCLUSION

In conclusion, the LockBox Secure+ project has successfully materialized as an innovative and secure IoT solution for safeguarding valuables. The integration of a Numpad interface has proven effective in controlling the locking and unlocking mechanisms of the safe, providing users with a straightforward and reliable means of access. The inclusion of a vibration sensor has enhanced the security features, promptly detecting any unauthorized movements or tampering attempts on the safe. One of the notable achievements is the seamless integration with the Telegram messaging platform. The project has met the acceptance criteria by promptly capturing images upon unauthorized access attempts and triggering an audible alert through the buzzer. These captured images are then securely transmitted to the safe owner via Telegram, providing real-time awareness and enabling swift response to potential security threats.

As we look towards further development, the LockBox Secure+ project lays a solid foundation for enhancements and expansions. The use of ESP-NOW for communication between devices, the integration of the MPU6050 sensor for motion detection, and the utilization of FreeRTOS for multitasking highlight the project's technical robustness. To elevate the project based on the evaluation conducted, future developments could explore refining the locking mechanism, implementing additional security features, and optimizing power management for prolonged operation.

The project's success in meeting the acceptance criteria showcases its practicality and effectiveness in securing valuable possessions. With a well-established foundation, there is room for iterative improvements and customization to cater to specific user needs. Overall, LockBox Secure+ not only fulfills its primary objective but also opens avenues for continuous refinement, ensuring it remains at the forefront of IoT security solutions.

REFERENCES

[1] AAIsecurity, "The Dangers of Traditional Lock and Key Security," [Online]. Available: https://www.aaisecurity.co.uk/news/the-dangers-of-traditional-lock-and-key-security/. [Accessed: December 4, 2023].

[2] Keymitt, "Why Are Smart Locks Actually Safer Than Traditional Locks?" [Online]. Available: https://keymitt.com/blogs/keymitt-blog/why-are-smart-locks-actually-safer-than-traditional-locks. [Accessed: December 4, 2023].

[3] Vivint, "Smart Door Lock vs. Traditional," [Online]. Available: https://www.vivint.com/resources/article/smart-door-lock-vs-traditional. [Accessed: December 4, 2023].

[4] RandomNerdTutorials, "Telegram with ESP32-CAM to Take Photos and Send Alerts," [Online]. Available: https://randomnerdtutorials.com/telegram-esp32-cam-photo-arduino/. [Accessed: December 7, 2023].

[5] Hackster.io, "ESP32-CAM Motion Detection and Image Capture for Telegram," [Online]. Available: https://www.hackster.io/ashshaks/esp32-cam-motion-detect-send-captured-image-to-telegram-54f5f2. [Accessed: December 8, 2023].

[6] ESP32IO, "ESP32 Keypad Tutorial," [Online]. Available: https://esp32io.com/tutorials/esp32-keypad. [Accessed: November 8, 2023].

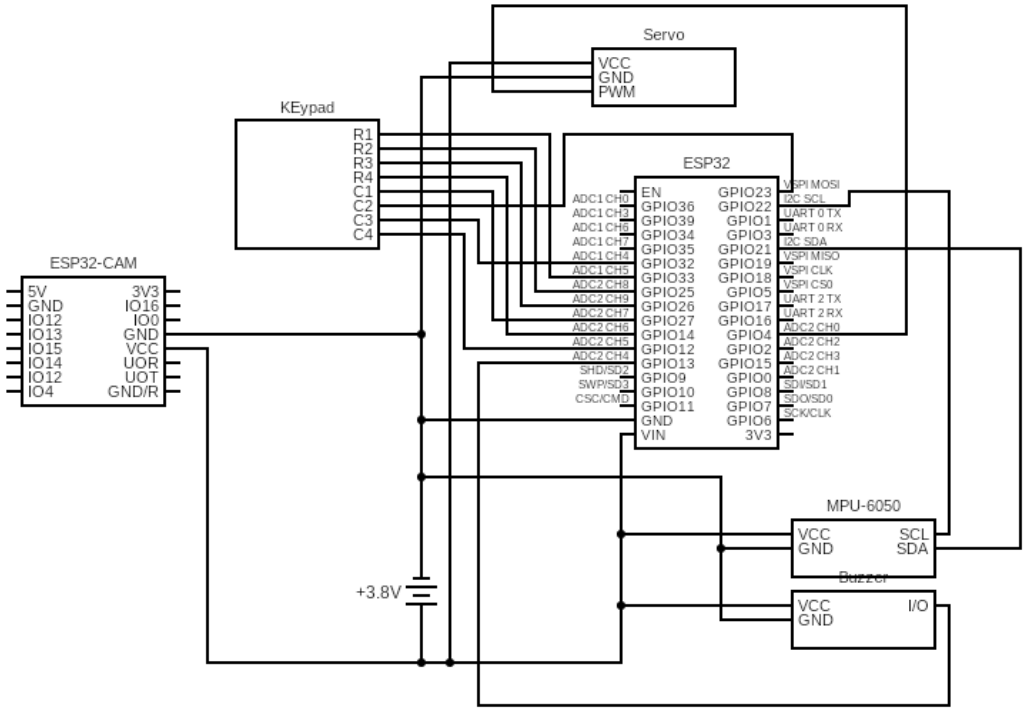
[7] MakerGuides, "How to Drive Servo Motors Using ESP32," [Online]. Available: https://www.makerguides.com/how-to-drive-servo-motors-using-esp32/. [Accessed: November 9, 2023].

[8] Random Nerd Tutorials, "ESP32 MPU-6050 Accelerometer and Gyroscope," [Online]. Available: https://randomnerdtutorials.com/esp32-mpu-6050-accelerometer-gyroscope-arduino/. [Accessed: November 10, 2023].

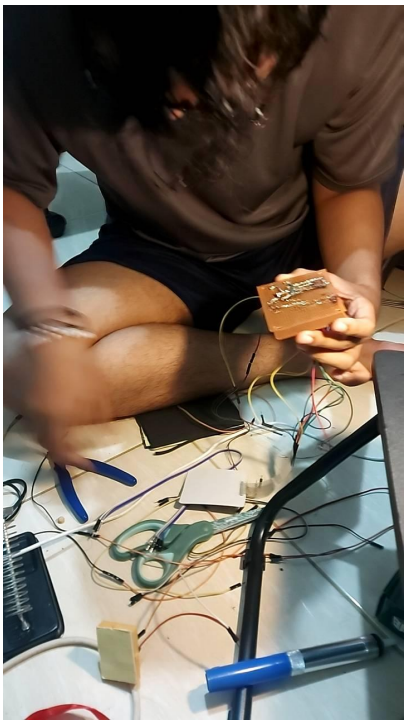
[9]

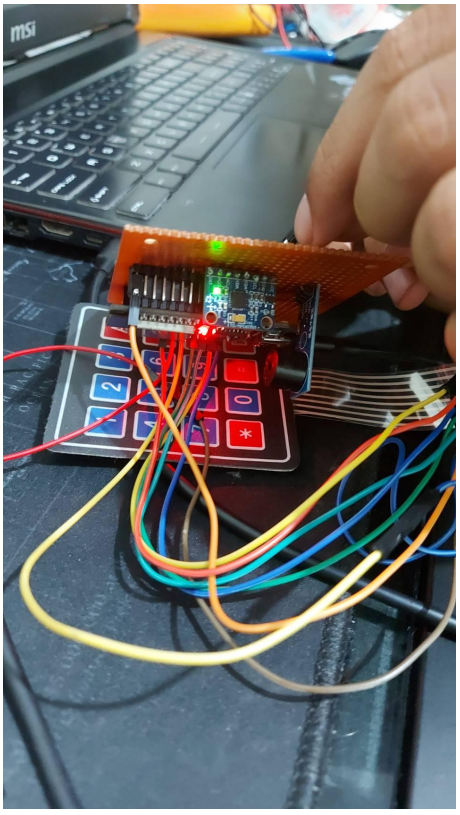
APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation




msi