Keyless Smart Lock (Secured Illini)
ECE 445 Final Report - Spring 2025

Project #61
Andrew Ruiz, Sebastian Sovailescu, Bowen Cui

Professor: Arne Fliflet
TA: Sanjana Pingali

Abstract

This report presents the design, implementation and evaluation of a novel smart bike-lock system. A servo motor secures the bicycle, while Bluetooth Low Energy enables seamless, keyless user authentication. Integrated inertial sensors detect tampering or unauthorized movement, triggering real-time alerts to a cloud-based dashboard for continuous security monitoring. Emphasis was placed on user convenience, reliability, and resistance to tampering. The design process included initial concept development, prototyping, testing, and iterative refinement. Overall, the system achieves its goals of user convenience, dependable operation and strong resistance to tampering, offering an effective, user-friendly approach to bicycle security.

Contents:

1. Introduction

1.1. Problem

Bike theft remains a major issue in urban and suburban areas, with millions of bicycles stolen annually due to the shortcomings of conventional locks. Despite the use of U-locks and chain locks, thieves easily bypass them using bolt cutters, angle grinders, and lock-picking tools. According to 529 Garage, over two million bikes are stolen each year in North America, discouraging cycling and undermining sustainable transportation efforts. Research by Sidebottom et al. (2009) highlights that even high-security locks can be compromised within minutes, exposing the need for more effective theft prevention measures. Additionally, improper locking techniques further contribute to the problem, leaving bicycles vulnerable. Addressing these security gaps is essential to protecting cyclists and promoting bicycle use as a reliable mode of transportation.

1.2. Solution

We propose a smart bike lock equipped with tracking, a keyless locking mechanism via Bluetooth, and an integrated siren that offers a comprehensive solution to the problem of bike theft. WiFi tracking ensures that stolen bikes can be quickly located and recovered, significantly increasing the chances of retrieval compared to traditional locks. The keyless locking mechanism eliminates vulnerabilities associated with physical keys or combinations, reducing the risk of lock picking or brute-force attacks. By using Bluetooth connectivity, cyclists can securely lock and unlock their bikes through a smartphone app, adding convenience while maintaining security. Additionally, a built-in siren serves as an active deterrent by emitting a loud alarm when unauthorized tampering is detected, drawing attention and discouraging thieves. This multi-layered security approach not only makes theft more difficult but also increases the likelihood of intervention before a bike is stolen. By integrating these advanced features we will be helping to reduce bike theft rates and promote cycling as a secure mode of transportation.

1.3. High Level Requirements

- Electronic locking system that can be controlled with Bluetooth.
- A buzzer will sound for 10 seconds when our anti-theft algorithm detects suspicious activity within a locked state. Theft attempts will be determined when excessive movement is detected which sensitivity will be experimented with.
- Can send and receive real-time alerts, or temperature readings over WiFi using a ThingSpeak dashboard. All communications will be safe.
- 3.3V Indicator LEDs to indicate the lock’s current state.

2. Design

2.1 Introduction

The smart bike lock will have a U-lock shape, combining durability and security while accommodating the necessary electronics. The main housing at the base encloses the PCB, battery, IMU, and siren. A servo motor will control the locking bolt, allowing for electronic and backup manual operation. The U-shaped shackle, aluminum wire, will resist cutting and prying. To withstand outdoor conditions, the lock will feature a weather-resistant steel enclosure. Designed for easy mounting and portability, the lock will balance security, usability, and smart connectivity.

2.2 Block Diagram

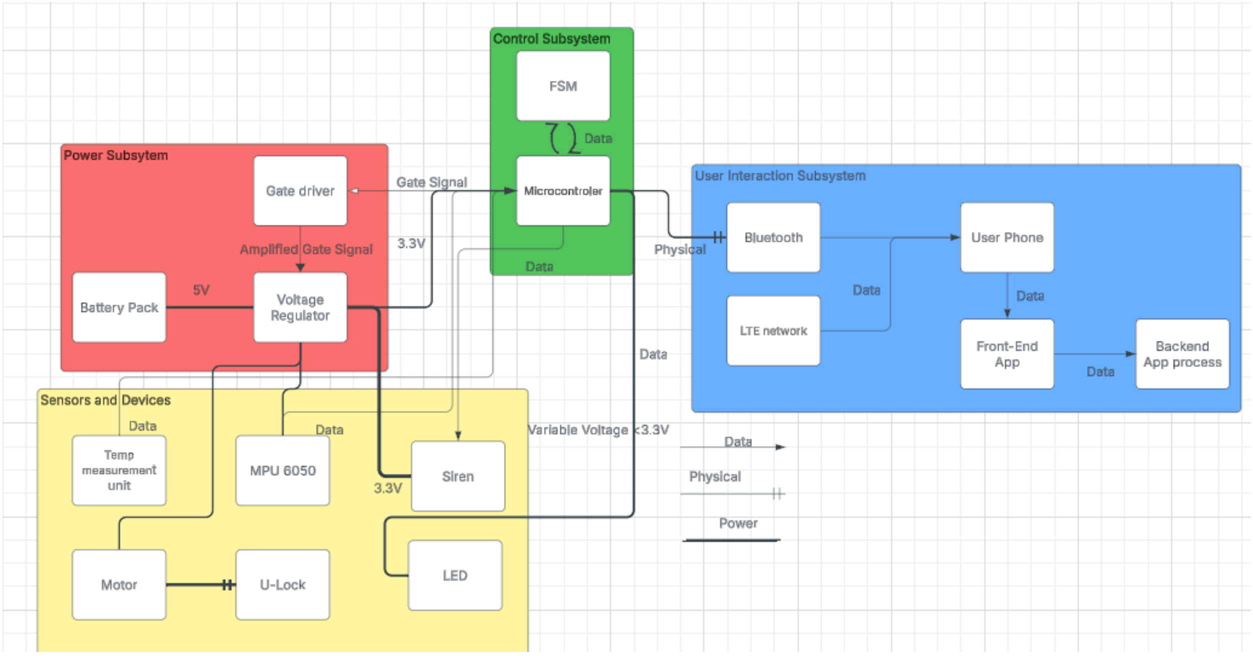


Figure 1: Initial Block Diagram

2.3. Design Details

2.3.1 Physical Design

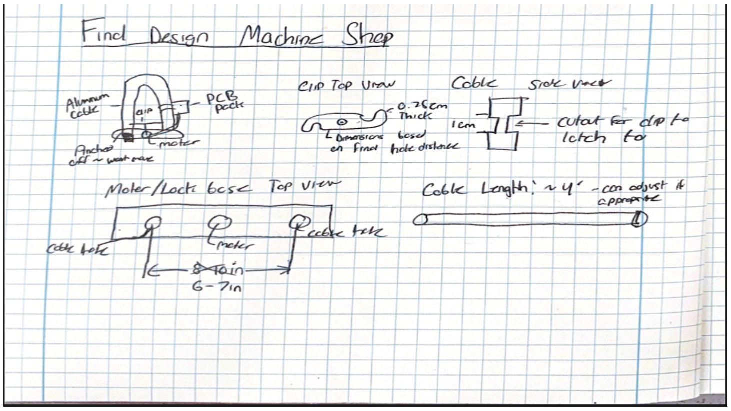


Figure 2 : Physical Design Schematic for Machine Shop

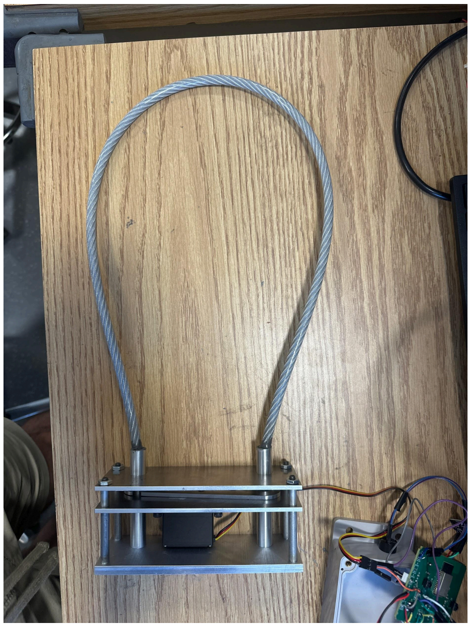


Figure 3 :Fully constructed lock with servo attached from machine shop

2.3.2 PCB Design

The final PCB design integrates the essential programming circuitry directly onto the board. It also includes dedicated ports for the USB-UART bridge, IMU, servo motor, battery, and siren, allowing for streamlined assembly and reliable connections to all peripheral devices. This integrated layout simplifies both development and deployment while ensuring compactness and ease of use.

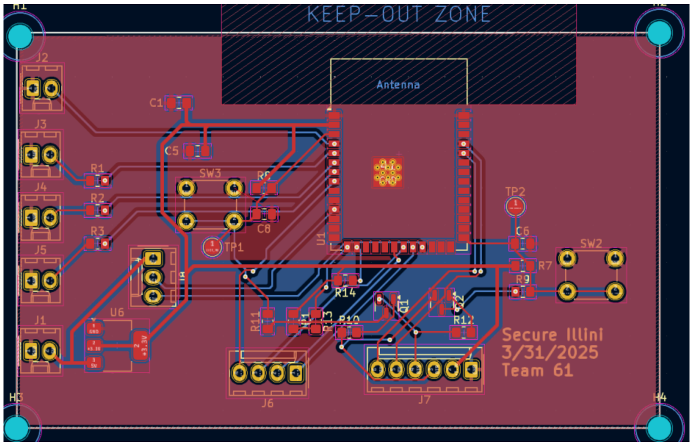


Figure 4 :Final PCB Layout

2.3.3. Power subsystem

Although this subsystem is physically small, consisting of only a few components, it plays a critical role in the overall functionality of the smart lock. At its core is a 4.8V 2000 mAh Tenergy rechargeable battery pack, which serves as the primary power source for the device. Another key component in this subsystem is the LM1117 3.3V voltage regulator, which steps down the 4.8V from the battery to a stable 3.3V rail used to power sensitive components such as the ESP32 microcontroller and the programming interface. Meanwhile, the 4.8V rail from the battery is used to directly power peripherals like the IMU and the servo motor. To ensure stability and smooth voltage regulation, several bypass capacitors ranging from 0.1 µF to 10 µF are included in the design. Additionally, a recharge port is built into the battery pack to allow for convenient recharging without disassembling the system. This compact yet essential power subsystem ensures the safe and efficient operation of all electronic components within the smart lock.

TENERGY
Item No. 11001
NiMH RECHARGEABLE BATTERY PACK
4.8V 2000mAh
MUST READ INSTRUCTIONS BEFORE USE
www.Tenergy.com

Figure 5 :4.8V 2000mAh battery used

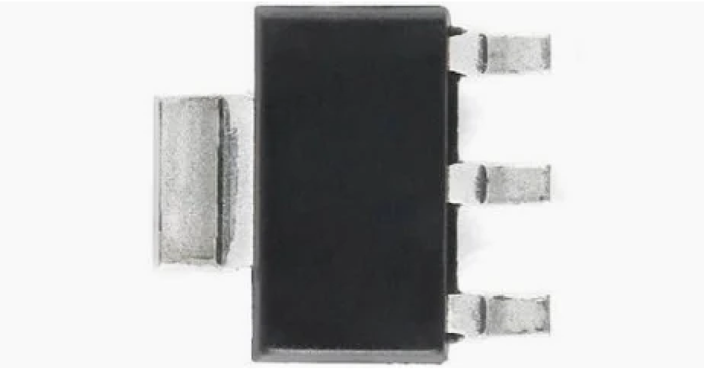


Figure 6 :LM1117 Linear voltage regulator

2.3.4. Control subsystem

The control subsystem consists of an ESP32-S3-WROOM-1U implementing a three-state finite-state machine (LOCKED, UNLOCKED, ALARM) that manages the entire system operation. The processor:

- Maintains state transitions based on sensor inputs and commands
- Controls the servo motor with precise timing for locking/unlocking
- Processes BLE commands through a custom service with callback architecture
- Analyzes MPU6050 accelerometer data to detect tampering attempts
- Manages alarm responses with timed visual and audio alerts
- Communicates system status to ThingSpeak via formatted HTTP requests
- Generates distinct audio patterns for different system events\

A list of all the signals can be seen in Figure 8

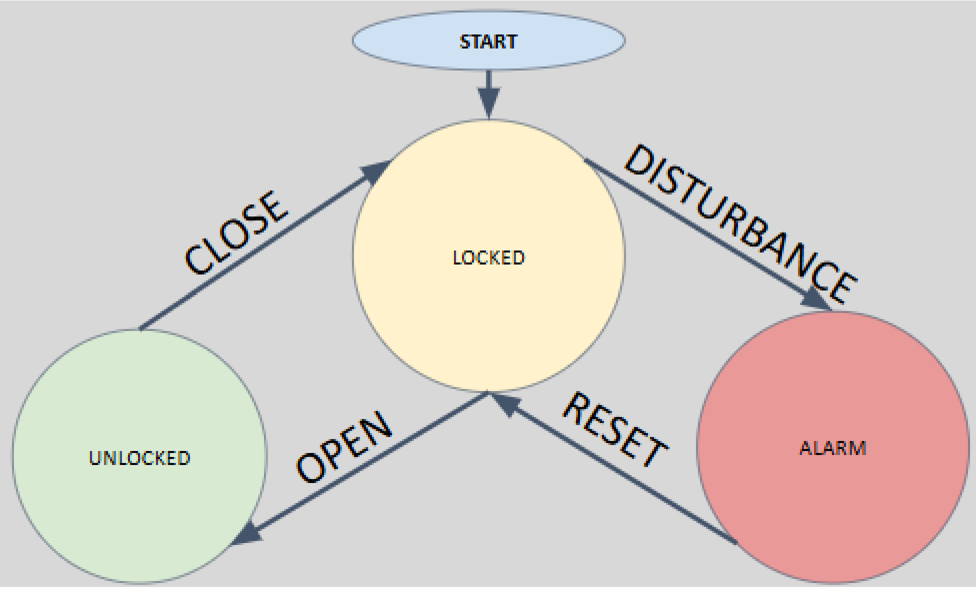
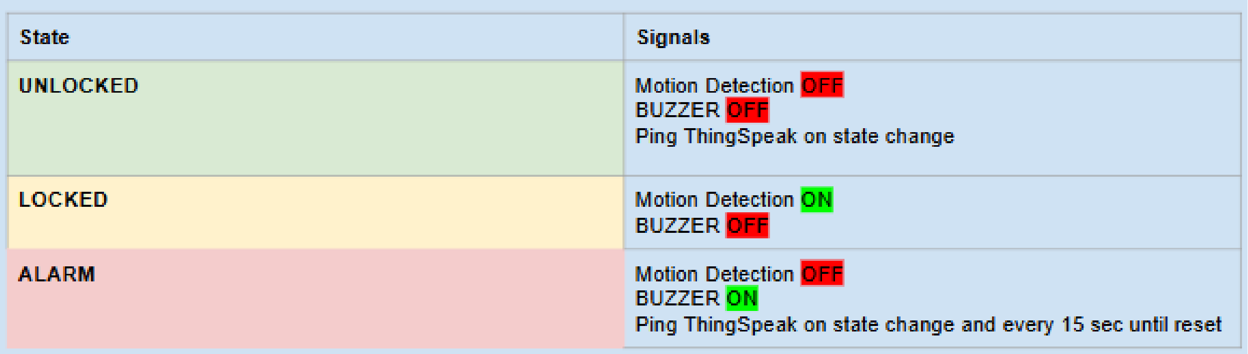


Figure 7: Finite state machine of the control unit



State	Signals
UNLOCKED	Motion Detection OFF BUZZER OFF Ping ThingSpeak on state change
LOCKED	Motion Detection ON BUZZER OFF
ALARM	Motion Detection OFF BUZZER ON Ping ThingSpeak on state change and every 15 sec until reset

Figure 8 : FSM State signal Table

2.3.5. Locking subsystem

The locking mechanism integrates Bluetooth communication with servo motor control to provide secure remote operation. The ESP32 establishes a BLE server and processes authenticated commands through a custom service. When receiving the "open" command, the system transitions to UNLOCKED state, rotating the servo motor from 0° to 90°. Similarly, the "close" command returns the system to LOCKED state, rotating the servo from 90° to 0°. The servo utilizes allocated PWM timers with 1000-2000µs pulse width range for precise positioning. LED indicators provide visual feedback during operation, and each state transition triggers a ThingSpeak update to maintain cloud synchronization of the lock status.

2.3.6. Anti-Theft subsystem

The anti-theft subsystem utilizes the MPU6050 accelerometer to detect unauthorized tampering attempts. When the system is in LOCKED state, it continuously monitors acceleration across all three axes to detect suspicious movement.

The entire system relies on the Euclidean norm equation which calculates the total magnitude of acceleration using the IMU readings. It first computes the square root of the sum of squares of acceleration components from all three axes

$$\boxed{|v| = \sqrt{x^2 + y^2 + z^2}}$$

Figure 9: Euclidean norm equation, where x, y, and z are the magnitudes of acceleration

The purpose of this calculation is to detect tampering by measuring the overall motion intensity of the device. When this value exceeds the IMU_THRESHOLD (13.0 m/s²), the system identifies it as a potential theft attempt and transitions to the ALARM state.

2.3.7. Dashboard subsystem

The ThingSpeak dashboard serves as the cloud monitoring interface for our smart lock system, providing real-time alerts and status updates. The ESP32 transmits critical data including lock state, temperature, and three-axis acceleration measurements via HTTP POST requests whenever state changes occur or alarm conditions are detected.

Users receive immediate notifications through the ThingSpeak platform when tampering is detected, as the system automatically transitions to ALARM state and begins sending periodic updates every 15 seconds. Our testing demonstrated that alert notifications reached users within 22 seconds of a tampering event, providing sufficient response time for intervention. The dashboard's visual interface allows users to monitor lock status remotely, track historical events, and verify system operational parameters from any internet-connected device, enhancing security awareness without requiring physical presence at the lock location.

2.4. Design Verification

2.4.1 RV Table

Requirement	Verification Method	Result
The lock shall be secure within 2 seconds of command.	Measure time from Bluetooth command to movement with stopwatch	Lock secured within 1.5s
Lock shall unlock only after valid authentication	Test it doesn’t open with invalid commands	Only opened and closed on proper commands
The lock shall detect and alert on tampering	Simulate tampering (e.g., vibrational or mechanical force) and check alert response.	IMU properly detected movement and signaled the siren.
There shall be a mechanical override.	Utilize mechanical unlock master key	While mechanical override is possible, it is not how initially intended therefore unsuccessful test.
Siren will trigger 90dB siren within 1 sec of tampering	Simulate tampering and time response time	Siren only capable of 70dB but successful timing

The electronic locking system shall be able to withstand at least 1000 lbs of pulling force without mechanical failure.	Apply increasing force with with force gauge until failure	Unsuccessful, motor was displaced from force well below 1000 lbs
The device shall send real-time alerts and receive commands over WiFi and/or Bluetooth using the custom web app	Confirm and demonstrate wifi communication with web app	Successful Test
3.3V Indicator LEDs shall show lock state (Locked/Unlocked) and battery status (e.g., Good, Low, Critical).	Demonstrate LED with different lock states and battery level	Successful Alarm status LED, due to design limitations with housing, Unsuccessful battery status LED
The system shall measure and report temperature data to the web application in real time.	Vary the temperature environment and verify reported values	Successful Test
The siren shall sound for 10 seconds once activated.	Trigger alarm and use a timer to verify siren duration.	Design choice led to a shorter duration than 10 sec, Unsuccessful test
Battery life can last up to 7 days	Leave plugged in and running for 7 days	Ran out of time to test but mathematically should be capable

Table 1: Requirements and Verification Table

2.4.2 Battery Life Test

Although we did not conduct a formal 7-day battery life test, we estimated the lock's performance under both ideal and peak usage scenarios. Under passive operating conditions, the current consumption of the main components was as follows: the ESP microcontroller drew 20 mA, the IMU consumed 3.9 mA, the status LED used 4 mA, the servo motor idled at 7.7 mA, and the siren remained inactive at 0 mA. This results in a total passive current draw of approximately 35.6 mA. Using a 2000 mAh battery, this corresponds to an estimated battery life of roughly 2.34 days under continuous passive operation. In contrast, peak current values during active operation were significantly higher, with the ESP drawing up to 300 mA and the servo reaching 200 mA, while other components remained at similar levels. Although this peak usage is not sustained continuously, it represents the upper bound of power demand during events such as locking/unlocking or alarm activation. To better understand the system's real-world performance, we generated plots showing the average power consumption over a 24-hour period and the projected battery life under varying usage conditions. These visualizations provide insight into how different activity levels impact battery longevity and help guide future optimization efforts.

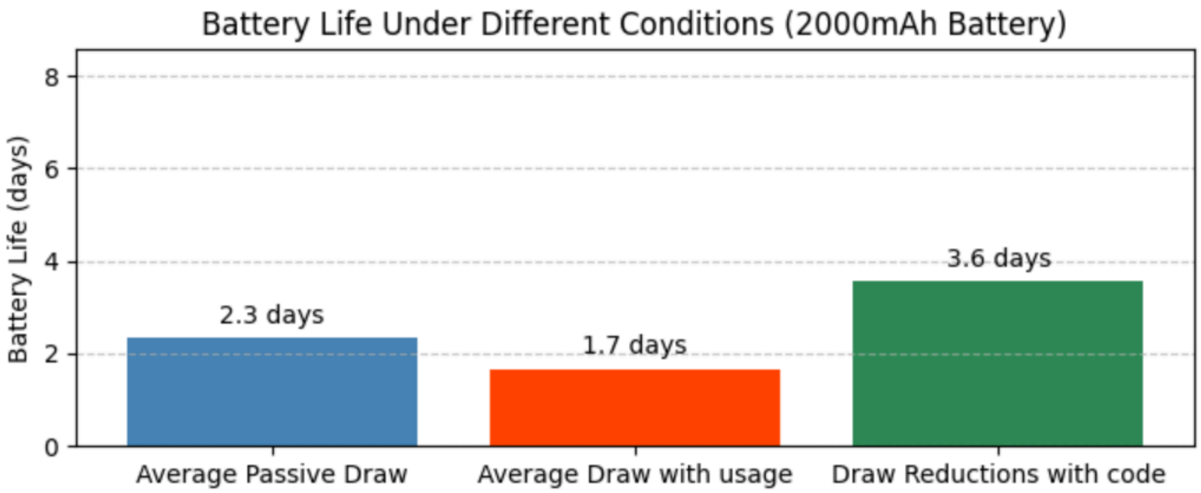


Figure 10 : Calculated battery life under different operating conditions

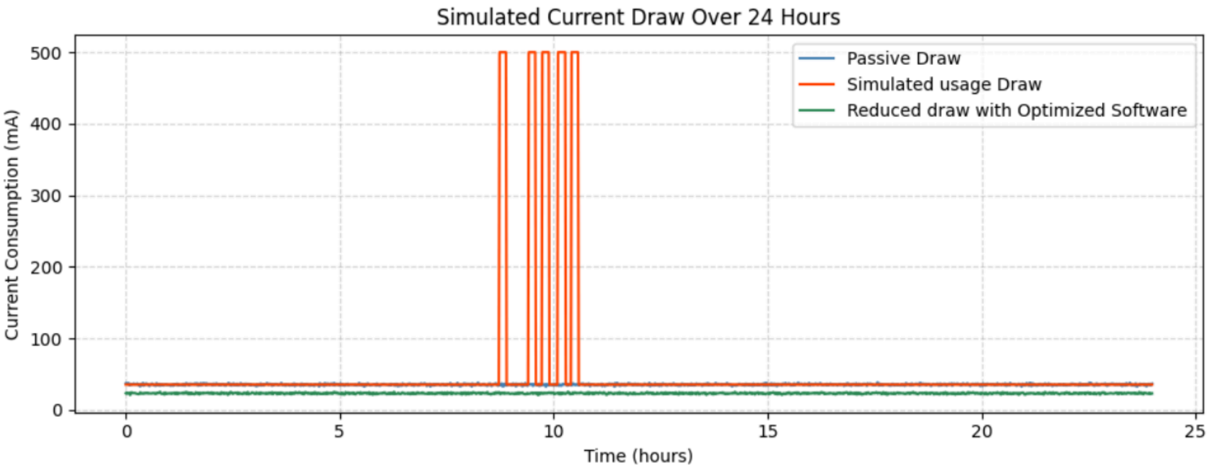


Figure 11 : Simulated power consumption over the course of a 24 hour period for different operating conditions.

2.4.3 Alarm Test

In order to ensure that our Alarm subsystem was capable of sending real time alerts, we performed a test to ensure the ThingSpeak dashboard updated in real time as tampering was detected. According to Figure 12, The first graph shows the timestamp at which we began tampering with the lock. Since it is in state 1 this means that it is in the locked state. The second graph shows the timestamp 22 seconds later that shows the device has been tampered with and thus is now in state 2 or the alarmed state. 22 seconds is reasonable to be considered real time as it is fast enough for the user to respond and intervene.

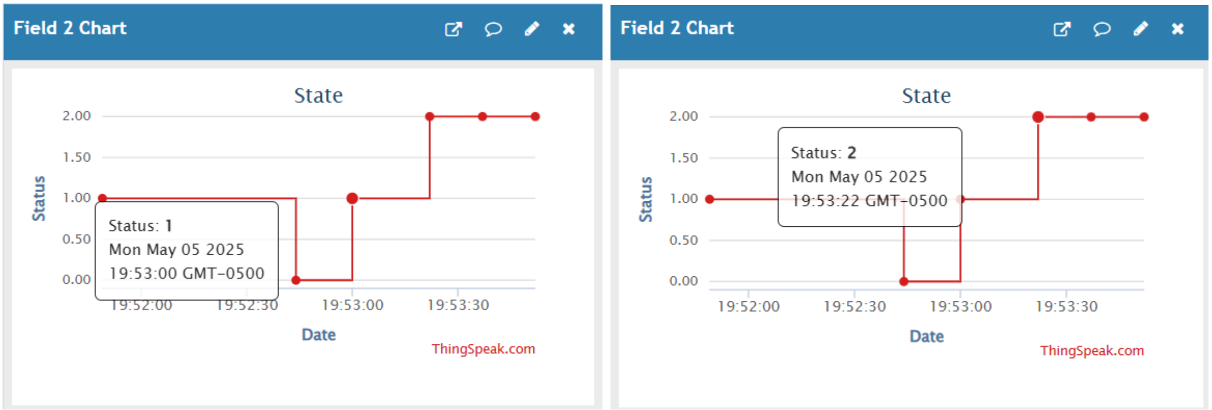


Figure 12: ThingSpeak State data from Alarm Test showing how within 22 sec of motion the state was able to update thus proving capability of real-time alerts

2.5 Design Process

2.5.1 Challenges

Throughout development, we encountered several challenges that impacted our timeline and testing process. Initially, the programming circuit failed to upload code reliably, which hindered early firmware development. Additionally, the stepper motor we intended to use only operated unidirectionally due to hardware and configuration limitations. We also needed multiple iterations of the PCB, as early designs lacked proper USB-UART bridge connections, further delaying testing. Troubleshooting was also complicated by bridged solder contacts beneath the ESP32, which caused signals to be unreadable and required careful rework to resolve.

2.5.2 Solutions

To address the challenges we faced, several key solutions were implemented. We resolved the programming issue by swapping the CTS and RTS pins of the USB-UART bridge, enabling successful code uploads. Due to the limitations with the stepper motor, we switched to a servo motor and updated the lock design, with help from the machine shop, accordingly. Our third PCB revision included the necessary USB bridge connections and proved successful. To recover lost time from earlier setbacks, we accelerated the testing and validation phases. Additionally, we ensured more reliable performance by carefully aligning the ESP32 during soldering and increasing the baking time to eliminate soldering defects.

3. Cost Analysis

3.1 Labor

Given as this project is for a class the true cost of labor will be $0, but we wanted to calculate how much a project like this would cost at market rate. Assuming a team of three individuals, a hardware engineer, a software engineer, and a product designer working on the project, we can estimate labor costs based on typical hourly wages.
Hardware Engineer: Responsible for circuit design, PCB layout, and integration of electronic components such as Bluetooth modules and locking mechanisms. Estimated hourly rate: $50–$70.
Software Engineer: Develops firmware for microcontrollers, mobile app connectivity, and security features like encryption. Estimated hourly rate: $60–$80.
Product Designer: Designs the physical enclosure, ensuring durability, weather resistance, and usability. Estimated hourly rate: $45–$65.
If each team member works 40 hours per week for 12 weeks, the total labor cost can be estimated as follows:

Hardware Engineer: $50 × 40 × 12 = $24,000 (minimum estimate)
Software Engineer: $60 × 40 × 12 = $28,800
Product Designer: $45 × 40 × 12 = $21,600
Total Estimated Labor Cost:
At minimum rates, the total labor cost for 12 weeks would be $74,400, while at higher rates, it could exceed $100,000. Additional costs may arise from extended development time, testing, and unforeseen challenges. Clearly it is beneficial to be designing this project as a passion project for this class as labor can be the biggest cost to the design.

3.2 Parts

Description	Manufacturer	Part #	Quantity	Cost
Servo Motor	HiTEC	HS-311	1	$13.49
Microcontroller	Espressif	ESP32-S3 module	3	$13.80
Piezo buzzer sensor	Adafruit	SBZ-204	1	$1.62
6-axis	HiLetgo	MPU-6050	3	$10.99

Accelerometer Gyroscope Sensor				
4.8V 2Ah Battery Pack	Tenergy	Battery	1	$16.99
Plastic Electronic Project Box	WeiMeet	IP65	1	$7.99
LEDs	Digikey	QBL7IB60D	2	$0.38 x 2 = $0.76
Push-button	Adafruit	1683	1	$3.33
USB-UART bridge	HiLetgo	FT232RL	1	$6.49
Voltage Regulator	Digikey	LM1117MP-3.3	1	$1.38
Total				$76.84

Table 2: Component Part Number and Price Table

3.3 Total costs

Labor ($74,400) + Parts ($76.84) + Machine shop hours ($100 x 15) = 75,977 ~ **$76,000**

4. Ethics and Safety Considerations

4.1 Ethics

Developing a smart bike lock with tracking, Bluetooth locking, and an alarm system involves several ethical and safety considerations, particularly in alignment with the IEEE Code of Ethics and the ACM Code of Ethics. Privacy is a major concern, especially with location tracking, which must be handled responsibly to prevent misuse. To ensure user data security, we will require explicit consent for location tracking. Additionally, system reliability is critical, as malfunctions could leave users stranded. In accordance with ethical guidelines to “avoid harm,” we plan to integrate redundant unlocking methods, such as a backup PIN entry or an emergency override.

4.2 Safety

Beyond ethical concerns, our design must comply with applicable safety and regulatory standards. Because our smart lock uses Bluetooth and GPS, it must adhere to FCC Part 15 regulations for radio frequency emissions and comply with UL 437 standards to ensure resistance to physical attacks like cutting or drilling. Furthermore, we must account for state laws regarding electronic tracking devices, ensuring that location data remains private and is accessible only to the rightful owner. To address potential safety issues, such as false alarm activations, we will implement adaptive sensitivity settings for the alarm.

By adhering to these ethical and safety standards, we aim to develop a secure, reliable, and compliant smart lock that effectively reduces bike theft while minimizing risks to users.

5. Conclusions

5.1 Summary

In conclusion, although some design decisions were made with the intent of this being a demonstration project lock, we successfully constructed a device that is structurally sound and incorporates all the features we aimed for, such as keyless locking and an accelerometer-based alarm system. We demonstrated the ability to remotely lock and unlock the device via Bluetooth and to receive real-time data through Wi-Fi. The broader impact of our project in a global societal context is that our device serves as a deterrent to would-be thieves, not only by physically resisting break-in attempts, but also by sounding an alarm when tampered with. This dual-layered protection helps discourage theft and prevent opportunistic crimes.

5.2 Future Work

Rather than designing a full electronic lock, which can be costly and difficult to secure, we would like to explore the idea of a smart attachment as a more practical alternative. Most existing physical locks already offer strong physical protection but lack electronic safety features. Our new solution involves using our same smart attachment box that houses the majority of the electronic components and can be easily mounted onto a bike or the lock itself. This approach offers a cost-effective way to enhance the security of existing lock systems on the market without compromising on functionality or safety.

6. References

ACM. *ACM Code of Ethics and Professional Conduct*. Association for Computing Machinery, 2018, https://www.acm.org/code-of-ethics.

FCC. *Title 47 CFR Part 15 – Radio Frequency Devices*. Federal Communications Commission, https://www.ecfr.gov/current/title-47/chapter-I/subchapter-A/part-15.

IEEE. *IEEE Code of Ethics*. Institute of Electrical and Electronics Engineers, https://www.ieee.org/about/corporate/governance/p7-8.html.

Project 529. (n.d.). New research claims 2 million bikes are stolen in North America every year. Retrieved from https://www.pinkbike.com/news/new-research-claims-2-million-bikes-are-stolen-in-north-america-every-year.html

UL. *UL 437 – Standard for Key Locks*. Underwriters Laboratories, https://www.ul.com.

U.S. State Laws on Electronic Tracking Devices. *Legal Considerations for GPS Tracking*, National Conference of State Legislatures, https://www.ncsl.org.

Sebitian. *ECE445*. GitHub, last updated **6 Mar 2025**, https://github.com/Sebitian/ECE445.

Sidebottom, A., Thorpe, A., & Johnson, S. D. (2009). Using targeted publicity to reduce opportunities for bicycle theft: A demonstration and replication. European Journal of Criminology, 6(3), 267–286. https://doi.org/10.1177/1477370809102168