

LockSmith Digital Lock

Omar Abouhussein, Fiyi Ogunkoya, Francis Tengey, Obinna Udo耶
Purdue University

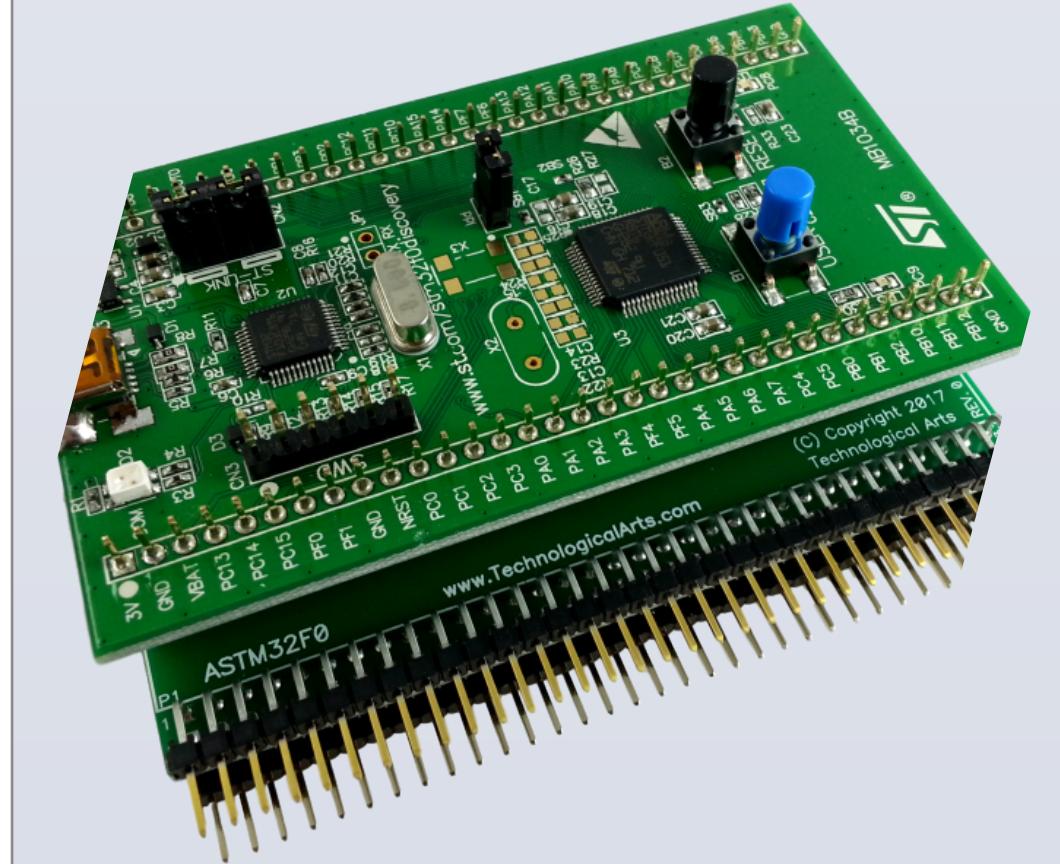
Introduction

Protecting belongings and keeping personal stuff private can be hard at times. Vaults, safes, and piggy banks are all ways to secure your belongings; However, they lack the intellectual aspect in unlocking the box. The goal of this project is to develop a Digital Lock that is only triggered by correctly answering a sequence of mathematical operations, with each answer being a digit to the passcode responsible for opening the lock. This project heavily relies on the integration of a microcontroller to get the correct functionalities.

Objectives

- Set a lock that is only unlocked by answering a sequence of mathematical questions correctly
- Disarm an alarm and lock the box if the user inputs an incorrect passcode/answer
- Disarm an alarm and lock the box if the user tries to input multiple answers consecutively
- Trigger a ‘Time Out’ and lock the box if the user fails to answer the question within 6 seconds

Materials



- STM32F0 Discovery



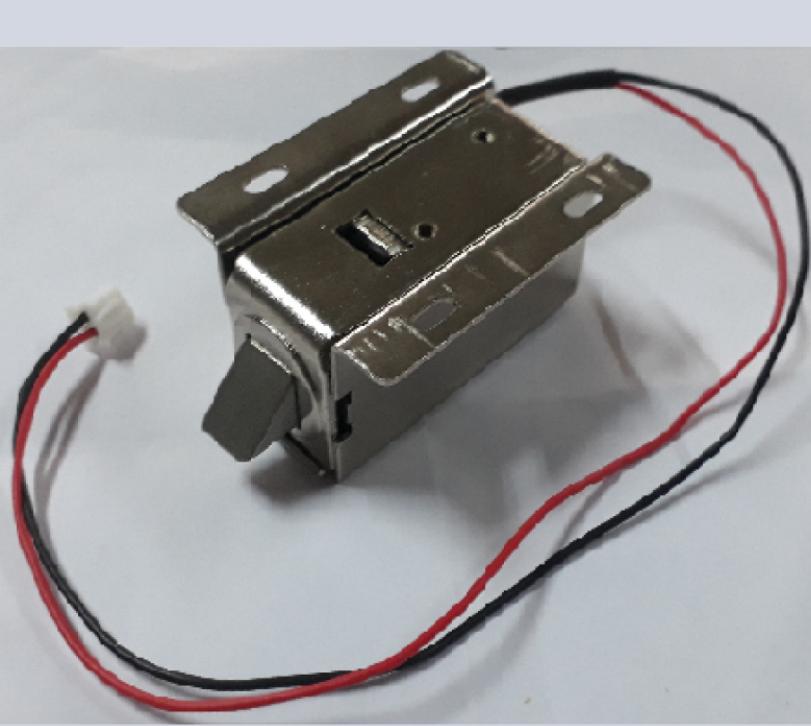
- LCD Display - CFAL1602C-B



- 12-button keypad



- Solenoid Lock



- Circuit Speaker

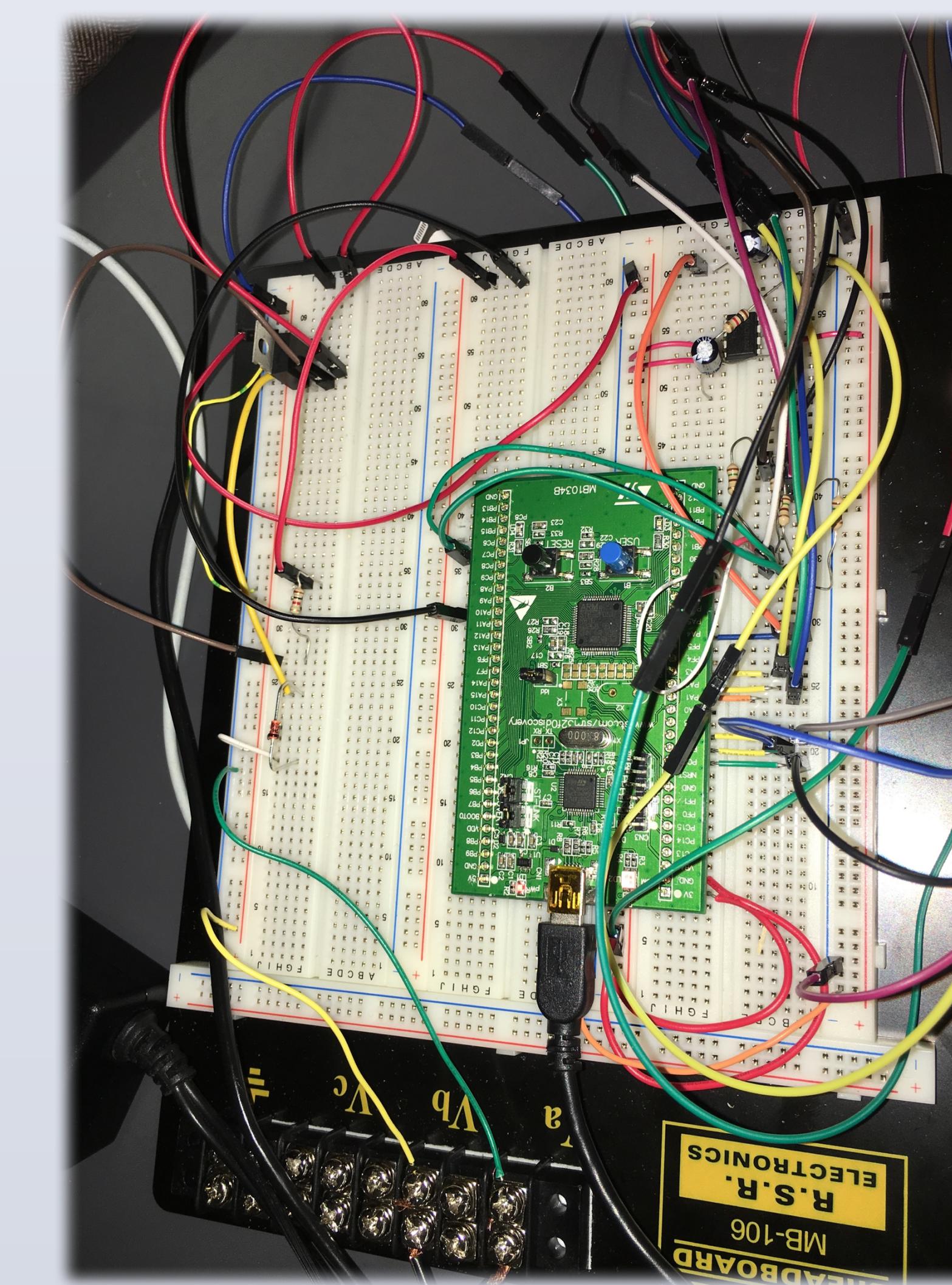
Methods

Programming the software

Displaying the questions in a scrolling display, scanning the input from the keypad, disarming an alarm sound, and triggering the lock are all programmed by the microcontroller STM32F0 Discovery. This was specifically done by using embedded systems and C as a low-level language to enable specific pins of the STM320 to be used as both inputs and outputs. By incorporating peripherals like the Digital-to-Analog Converter (DAC), General Purpose Input Output (GPIO), and Standard Peripheral Interface (SPI), the connection between the software and hardware is facilitated and much simpler.

Hardware and Casing Design

The hardware used to drive and program the whole system is the STM32F0 Discovery microcontroller. This device streamlines project development and allows new programmers to add on to the project once they have learned the functionalities and the essentials.



Though our initial prototypes used breadboards, we plan on developing a PCB-based implementation because we need reliable hardware. Integrating a PCB in this project is essential because it avoids the hassle of overlapping and tangled wires.

Results

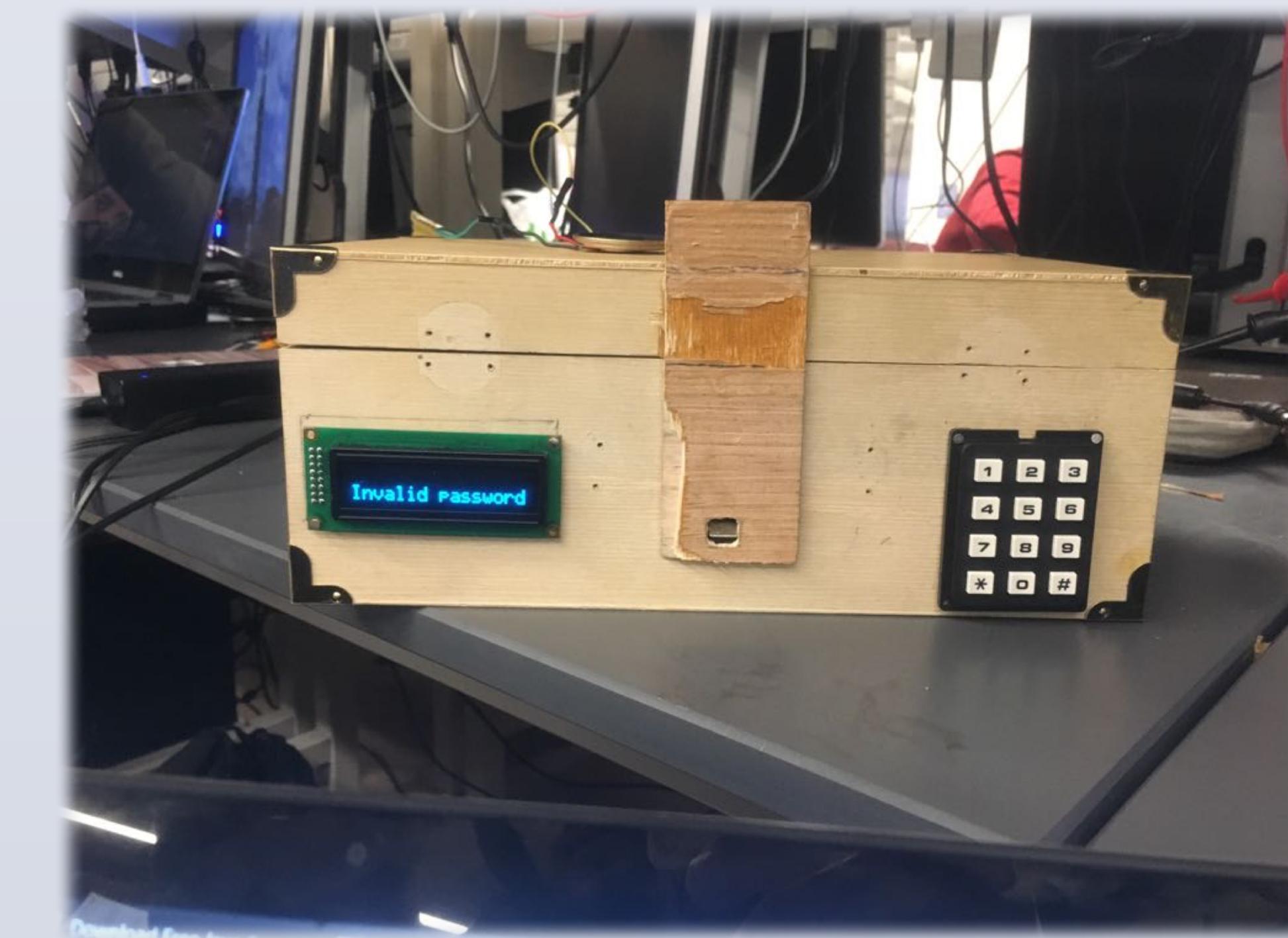
Multiple Attempts of answering a question

If the user guesses the correct answer for one of the question, yet tries to enter the answer more than once, the program will identify this user as an intruder and will not allow the lock to be opened.



Entering an incorrect password

If the user answers the question incorrectly, an alarm will be triggered, the box will remain locked, and the LCD display will notify the user that the password is invalid



Failure to input an answer within allocated time frame

In the case of the failure of the user to input an answer within the allocated timeframe of 6 seconds, the box will remain unlocked, a “TIME OUT” message will appear on the screen, and the alarm sound will be triggered.

Entering the correct password

Once the user correctly answers all the questions, putting in the correct password, a message will appear on the screen notifying the user that the box was unlocked successfully



Conclusion

There were many pieces to this project and the team members tried to address these components with an intellectual approach. The hardest challenge the team faced was linking all these pieces together and have the project come out as a coherent, functioning unit.

Contact

oabouhus@purdue.edu

ftengey@purdue.edu

ooqunkoy@purdue.edu

oudoyeh@purdue.edu