Microcomputing Systems Term Project

By

The DLCKs

Derek Logan Harper, Coy Trent Bryant,

Kallen Vittorio Bullock, Lewis Forrest Bates

Class number: ECE 3130-001

Submission Date: April 19, 2023



# Table of Content

Title Page..................................................................................................................... 1

Table of Contents.........................................................................................................2

Introduction..................................................................................................................3

Project Specifications and Descriptions.......................................................................5

Detailed Implementation..............................................................................................6

Analysis........................................................................................................................10

Description of The Teamwork Experience...................................................................11

# Introduction

**Project Overview**

This project incorporates the STM32L031K6 Microprocessor that is attached to the EDUBASE-L452 PCB. For this project, the microcontroller interfaces with the LCD, a 7-segment display, and an external keypad attached by a ribbon cable. Using the components specified we created a digital lock system.

**Applications**

An application this could be used for is any type of safe where a user interface requires a PIN code to gain access. Another Application could be a security gate that requires the correct PIN to prevent trespassing.

**Priority of the Applications**

The importance is significant in the sense that the program utilized by the application is the frontline defender of the material or contents that the user has deemed vital using their discretion.

**What We Have Done**

We have created a program that allows the user unlimited attempts to gain access. We have allowed an administrative PIN to alter the user’s access by changing or deleting pre-existing PINS.

We did enable a user-friendly code with well-descriptive instructions to guide the user.

We have equipped the board with an external 4x4 mesh keypad with jumper wires.

We have not implemented an exit of the program; thus, it will run until it loses its power source

We did not limit the number of attempts the users can try before they are locked out.

**Problems Encountered During the Process**

One issue we discovered was we were able to overwrite a previous pin without having to delete the pin first.

Another issue we encountered was deleting a PIN would default back to “0000” which is previously an unusable PIN, however, once the admin deleted a pre-existing PIN, “0000” became a useable PIN unintentionally.

We also ran into an issue where the master PIN could be added as a user PIN by the admin. This caused the unintentional consequence of unlocking the program which was not the admin’s intended purpose.

An ongoing problem we were unable to correct properly was illuminating the 7-segment simultaneously with the buzzer for the 10-second duration. Due to the nature of the 7-segment display needing a continuous quick delay, we could not properly display “FAIL” while the buzzer was sounding.

**Problems Fixed During the Process**

As we previously stated we were able to override a previous PIN without having to delete the PIN first. We fixed this problem by setting a conditional IF statement that only allowed the pin to be added if the current PIN is 0000.

For the second issue, we created a conditional IF statement that only allowed entry into the program if the user’s input was not 0000.

Our approach for the third problem was to compare the newly added pin with the master pin. If the added pin matched the master PIN, it would not accept this PIN and display “Invalid PIN” on the LCD.

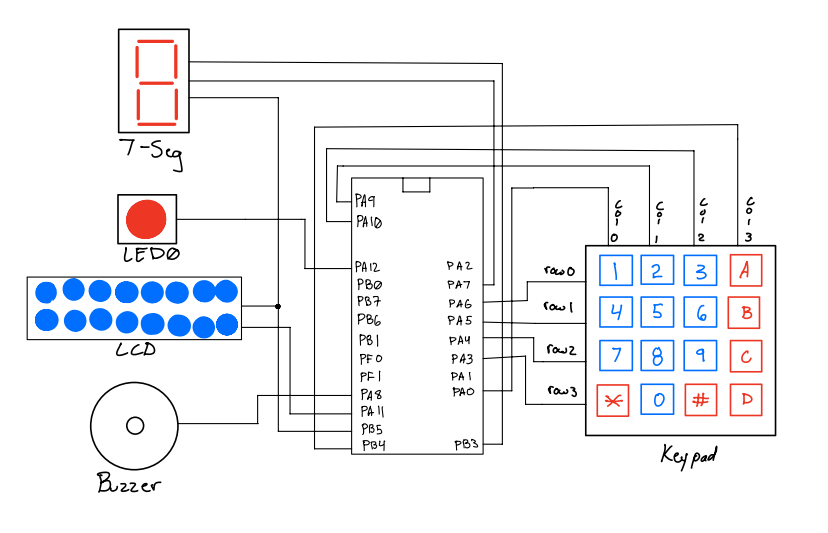
**Brief Functionality of the Project**

We are implementing a 4-pin Digital lock with 6 correct PINS and 1 Master PIN. If the user successfully enters 1 of the 6 PINS, the lock will open allowing access to the contents. The user will know if the PIN is correct when the LCD displays “Correct PIN, Now Opening”, the 7-segment display illuminates “PASS”, and LED\_0 is enabled for a 10-second duration. The User will know if the PIN is incorrect when the LCD displays “Incorrect PIN, Please Try Again” for 10 seconds while the 7-segment Display illuminates “FAIL”, and the Buzzer sounds alternately for 10 seconds. If the user inputs the Master PIN a menu emerges on the LCD enabling the user to either display PINS, delete PIN, delete all PINS, and add a PIN.

**Discussion of the Rest of the Report**

Throughout this report, we will continue to dig deeper into the programming flow and structure of the digital lock, how we started with a general knowledge of what our code should perform like and make it a possibility, and how we analysis and tested our code to make it run flawlessly making it the most user-friendly as possible and captures our final debugging. We will then wrap up with discussing how the team worked on the project.

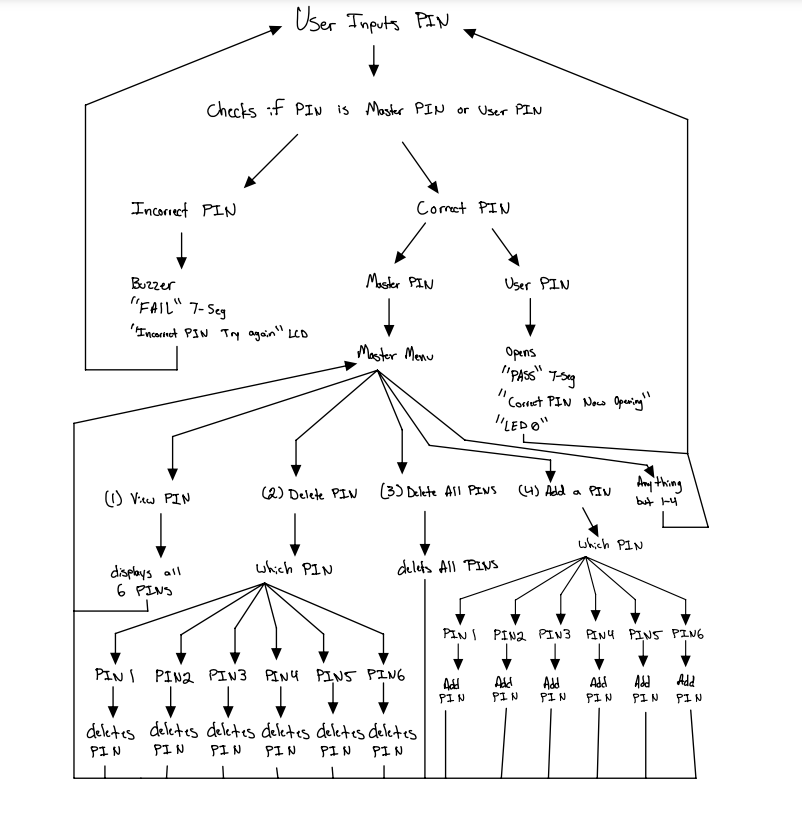
# Project Specifications and Description



The Keypad is what the user will use to send data to the labeled pins that will then input digits to the LCD shift registrar PB5. Once the user inputs the correct pin, the microcontroller will send data to the LED0, the shift registrar that will then send data to the LCD and the seven-segment. If the user does not input the correct pin, then the microcontroller will send data to the LED0, the buzzer and the shift registrar that will send serial data to the LCD and the seven- segment.

# Detailed Implementation

**Flowchart**



This flowchart shows how the user will navigate the program when operating the digital lock. The user can only truly have four different kinds of paths the user can input. These inputs are as follows: a correct pin, an incorrect pin, an invalid pin, and the master pin. The master pin is used as an accessor for the digital lock settings where the master can view existing pins, delete a certain pin, delete all pins, and add a certain pin. When these actions are over the program will default back to the master menu then the master can exit back to the main menu. The one invalid pin is zero since this is our indicator for an empty password and returns to the main menu or the master menu depending on which menu was accessed previously by the user or master. The correct pins are the pins we default set or the ones that the master has changed them to and has achieved access. The last type is the incorrect pin which will indicate to the user that the pin inputted is not the correct pin. When the correct or incorrect pin is inputted, the system will do the desired operation then return back to the main menu.

**The Code**

Within this same ZIP file, we attached the C code for the Program.

**Pseudocode**

1. The main functionality

Main()

//This is the digital lock

Initialize the Peripherals;

while(1)

{

Asks user for a pin;

Retrieves pin;

Checks pin;

Response based on check;

}

This pseudo code represents the main functionality of our project. This is the process in which the program goes through indefinitely. This loop cannot be broken since it would defeat the purpose of a lock.

1. The keypad

Reading KeyPad()

//Used for the user's input

Set all columns high;

While(until button is pressed){}

While(1){

Scan each column breaking when pressed row is found;

}

While(until button is released)

{}

return(int based on column and row scan);

This pseudo code represents how all the data used throughout the program is inputted. This code also represents how the user is able to interact with the keypad.

1. The Correct path & Incorrect path

Correct and Incorrect paths

//This is when a correct or incorrect pin is inputted

If (the pin is in one of the 6 pins) {

Instruct LCD to write;

Write to LCD;

Do "Pass" on 7-segment routine;

Clear all peripherals;

}

Else {

Instruct LCD to write;

Write to LCD;

Do "Fail" on 7-segment routine & Buzzer routine;

Clear all peripherals;

}

1. Master pin path

Master Pin path

//This is like the settings paths

While(input pin == master pin && end == 0){

Display Menu;

Retrieves user input;

If(input == 1)

Shows all pins;

Else if(input == 2)

Delete one pin;

Else if(input == 3)

Delete all pins;

Else if(input == 4)

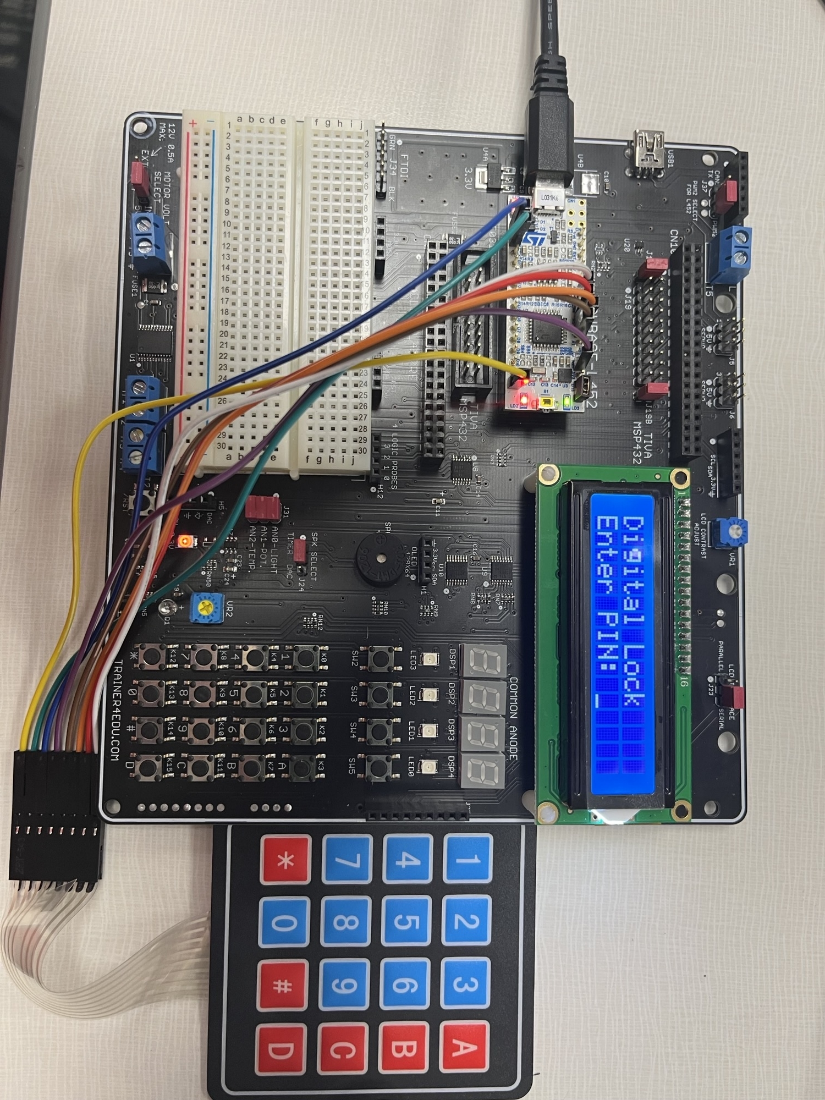
Add pins;

Else

Try again;

}

These two pseudo codes are in the same function since each is dependent on the user’s input pin. These two also display to the user what they can do.

**Image of Project**

# Analysis

In order to functionally test the digital lock, we ran through a series of checks inside each part of the program’s code. Initially, upon starting the program up for the first time, we checked if the display reflected what the code intended and input our first four-digit pin to see if the numbers would display on the LCD. Once the initial menu was working, we then ran through a series of checks to confirm existing pins match with the input pin that would then enable LED0 and display the proper lines on the LCD reassuring the user entered the correct pin.

After confirming correct pin numbers gave the desired output we wanted, incorrect pins were tested to ensure a false acceptance was not triggered. Using a series of incorrect pins, we verified the warning tone from the buzzer and a “FAIL” message displayed on the 7-segment each time without fail.

Finally, the admin password was checked to make sure each section performed as intended. Using the master password, we checked each section thoroughly for individual faults. For selection one of the admin menu, the pin display, the pin display would be checked anytime a pin was deleted or added. For selection two of the admin menu, deleting a pin, we verified pin numbers were correctly deleted and returned to a “0000” value for the selected pin and displayed the proper message accordingly. For selection three of the admin menu, deleting all the pins, all six pin numbers were checked to make sure each of the pins were deleted properly and the appropriate message displayed. For selection four of the admin menu, adding a pin number, this portion took a little more involvement than the previous three for a few reasons. We tested adding new pins to a previously existing slot to make sure it wouldn’t override previous pin numbers which eliminates the purpose of deleting a pin. Following that, there were two different numbers tested to make certain they could not be added as regular pin numbers. These two numbers were the null “0000” default pins upon deletion and the master pin to prevent conflict. Once those checks were established, the final test was to validate only the six pin options were selected with a display message stating anything more is an invalid pin.

When looking at public health, safety, and welfare, the project fits in well with this line of concern. There are many things in society that this digital lock can protect. In the modern day, people’s information is a gold mine for thieves and unjust individuals, and the digitization of society makes people especially vulnerable to this issue. A digital lock is a safety net that any individual, irrespective of technical knowledge, can understand and use effectively without much effort. When looking at health, the health records of someone are a personal set of data that can be secured through a digital lock enabling the privacy that everyone deserves. In reference to safety, a digital lock could either keep trouble out or simply keep trouble in (such as a prison compound). And welfare is no less important. This digital lock is a means to protect valuable assets that a person worked hard to obtain. No one wants to put forth effort to obtain something just to have it taken away.

The digital lock works well no matter the circumstances or environment it is utilized in. Whether it’s a global economy, a diverse cultural space, or varying economic environments, the digital lock has many uses no matter these conditions. Although our project is not specifically designed for every set of circumstances, most places in the world today use or can use digital locks to protect their property, family, money, or privacy.

To better the project in reference to the previously described situations, we believe the specific circumstances would need to be known to curtail the program to their needs. For a more robust need, the code would probably require more checks and ensure guessing would lock out the program upon a certain number of attempts. That said, the digital lock in and of itself would need little altering to fit the mentioned criteria since it's universally useable.

# Description of the Teamwork Experience

We went into the project open to all our ideas collectively. During the planning phase of the project, everyone was encouraged to volunteer for any part of the project they wanted to accomplish. We split up the work amongst ourselves to what each of us was comfortable working on, and this turned out to be beneficial with a successful product as a result.

During each phase of the project, everyone had an opportunity to lead their respective section. Whether it was starting the initial code, building upon that code to enhance the quality, debugging issues, or helping comment on each section, everyone had their opportunity to chip in and make the project work with as few flaws as we believe possible. Throughout the process, we would constantly meet to discuss milestones, check progress, or even just see if there were any quality-of-life functions we could incorporate.