Introduction

Background -

This lab focused on developing a basic digital combination lock using pushbuttons, a tri-colored LED for visual status indication, and a piezo buzzer for auditory feedback. The project builds directly on the foundational skills developed in Lab 0, reinforcing digital I/O handling, state machine design, and embedded programming with the ATtiny2313A microcontroller. This lab introduced the concept of sequential input verification, common in real-world systems like electronic safes, digital access panels, and keypad-based entry systems. Unlike the previous lab which used a simple 4-state cycle, this lab requires verifying a variable-length user-entered sequence, providing a practical implementation of user interaction and system state management. The ability to customize the unlock code to a desired input range also introduces considerations for security with flexibility. [1]

Product Functionality -

The product simulates a digital combination lock controlled by five pushbuttons, with a minimum unlock sequence of five digits and a maximum of eight, with a 5-digit range. At startup, the system is in a locked state, indicated by a red LED. As the user presses buttons to input the code:

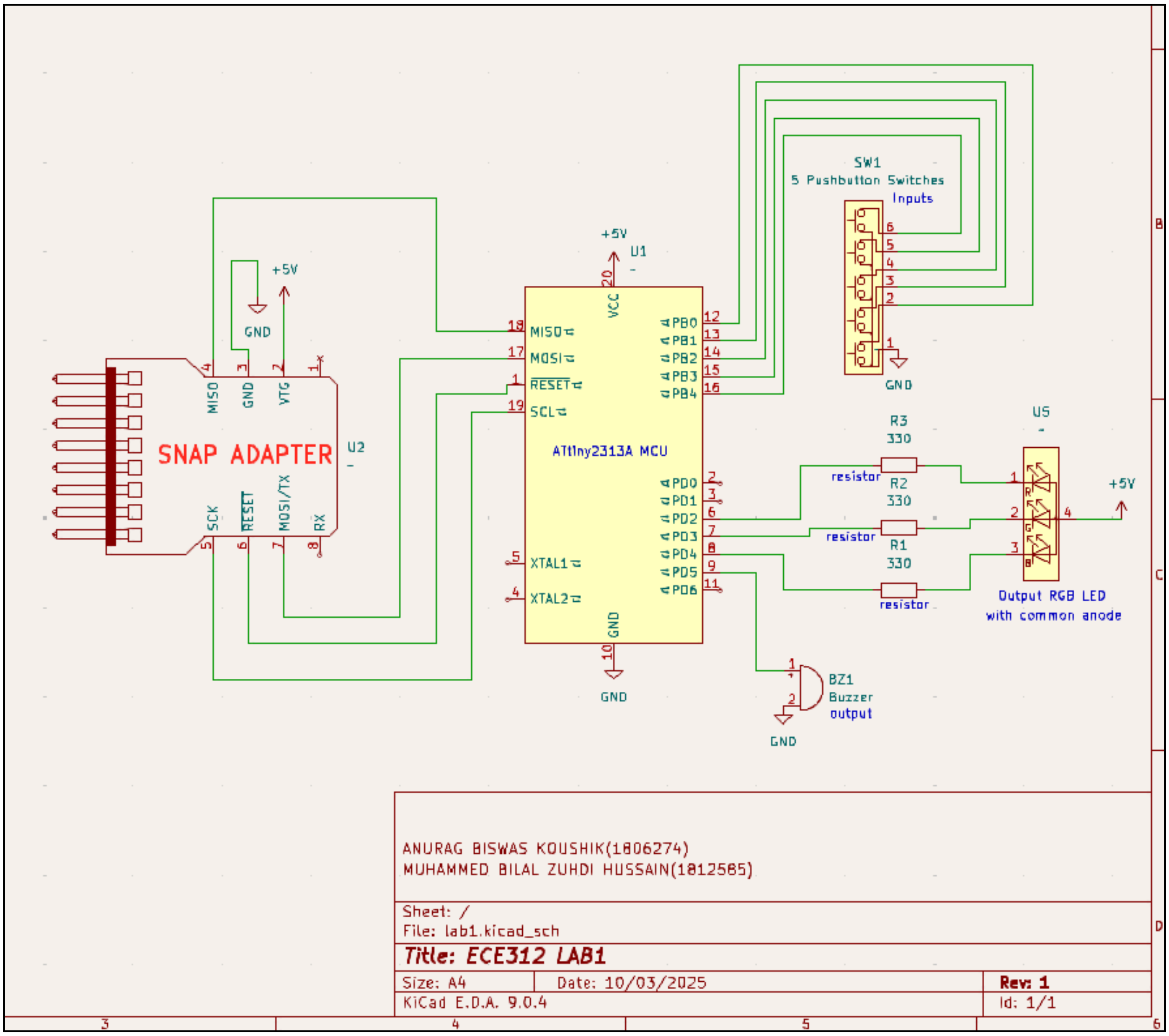
- Each press emits a short beep through a piezo buzzer.
- The LED turns white during the press as visual feedback.
- Upon release, if the full combination has not yet been entered, the LED returns to red.
- Once, the full combination has been entered, the LED flashes red rapidly if the code was incorrect along with the buzzer and the system is reset and locked.
- If the correct code was entered, the LED turns green for 5 seconds, unless a button is pressed to reset and relock the system. If no button is pressed, the system locks automatically after 5 seconds. [1]

The product behavior is managed by a finite state machine (FSM) that processes button inputs, compares sequences, and handles LED/buzzer feedback in real-time. Debouncing is applied to button inputs to prevent false readings, and the system must remain responsive and accurate even if inputs are entered rapidly.

Set-up Environment -

We were instructed to use an ATtiny2313A microcontroller which is programmed via a Microchip SNAP in ISP mode. We used the MPLAB X IDE to write the C code and upload it to the MCU. The schematic for this lab was designed in KiCad using the ECE312 library that was provided to us on Canvas. [1] The circuit consisted of the ATtiny2313A microcontroller, an RGB LED with common anode, a piezo buzzer, 330 ohm resistors and 5 push button switches used as input. We were provided with a DC power supply which we set at 5V and was supplied to the MCU, the LED,the piezo buzzer and the SNAP.

Circuit Schematic



Circuit schematic developed on KiCAD software

Prototype Circuit

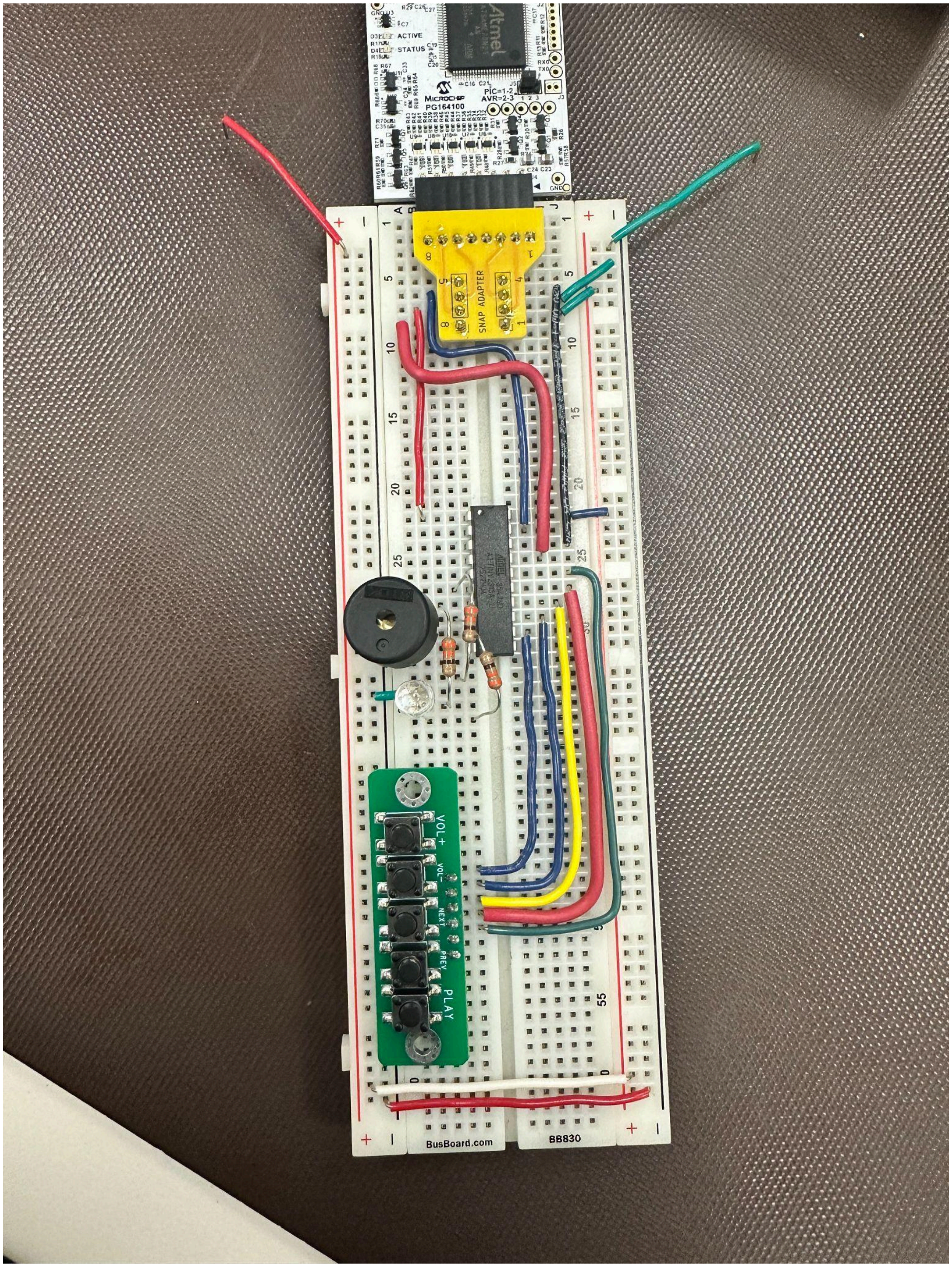


Figure 1.1: The prototype circuit consisting of the switches connected to the Attiny2313A microcontroller as well as the indicator LED and the buzzer, all of them connected to the SNAP module.

Conclusion

The electronic combination lock was successfully designed and implemented using the ATtiny2313A microcontroller, the pushbutton module, RGB LED and a piezo buzzer. The system can accurately detect each press on the push button module and with each press, turns the led white and sounds a tone on the piezo buzzer. While in operation, the lock remains in the locked position which is indicated by the LED being lit up red. The lock can be configured successfully in the range of 5 to 8 presses as required by the lab manual. If the correct sequence of key presses are detected, the LED turns green indicating that the lock has unlocked. After being unlocked, any key presses will immediately lock the system turning the LED red again, or, as an extra feature that me and my lab partner added, the system automatically locks after a set amount of time.

During the development of this project, there were a number of challenges that we faced, the most major one being debouncing the buttons and ensuring no false triggering. Figuring out the frequency logic for the piezo buzzer and calibrating the tones also took an effort. The timings of buzzer tone and the LED light changes were fine tuned a number of times to ensure minimal delay in the program flow. The button press counters were also fine tuned to reset reliably and not have false triggers. All of these issues were resolved through hours of systematic debugging and testing.

The system could be improved in a number of ways. One of them being having the ability to change the password from the hardware side, that is, not having to change the code, but the system would have a mode where it asks for the old password, and then change it by inputting it directly on the system. Having an LCD to display messages would also be a big improvement on the consumer side.

References

[1] “Lab 1: Combination lock,” Canvas, University of Alberta. [Online]. Available: Canvas Portal, University of Alberta. [ACCESSED: 10/09/2025].