

디지털 시스템 설계 및 실험 기말 프로젝트 보고서

팀 : 12

이름 : 임영준

학번 : 2018170925

이름 : 박상우

학번 : 2019171097

프로젝트 주제

Non-Numeral Digital Lock

프로젝트 개요

1. 기획 동기



현 도어락의 문제점

1. Alkali aluminosilicate glass 재질

-Vickers Hardness 480 – 670 HV 로 강철의 3 배 정도의 경도를 가지나, 반복된 충격에는 왼쪽 사진과 같이 흠집과 '스크래치'가 발생할 수 있다.

2. 반복된 충격으로 인해서 표면에 난 흠집은 왼쪽 사진과 같이 어느 숫자를 입력했는지를 유추 가능하게 한다.

사진: 스크래치가 나서
비밀번호를 예측할 수 있는 도어락

해결 방안

1. 비밀번호의 길이를 늘린다

-비밀번호 입력 시간이 증가하고, 비밀번호를 잊어버리기 쉽다

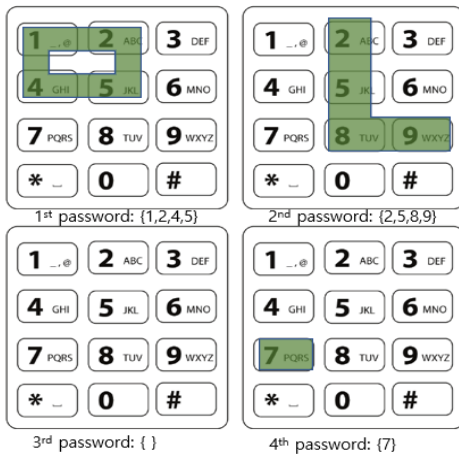
2. 특수문자를 추가한다.

-사용자들이 보통 특수문자를 기존 비밀번호의 맨 앞이나, 맨 뒤에만 추가하므로 보안성의 증가가 크지 않다.

외우기 어렵지 않으면서 보안성이 강한 방안이 필요함

2. 기획 컨셉

Non-Numeral Digital Lock



- 비밀번호는 4 개의 패턴 (Integer Set)으로 구성
- 각각의 Integer Set 내에서는 입력의 순서를 구분하지 않음
*(1, 2, 3, 4) 순으로 입력 = (1, 3, 4, 2)순으로 입력
- 각 Integer Set 은 {}부터 {1,2,3,4,5,6,7,8,9}까지 가능

Difficulty-Adjustable	아무것도 입력하지 않음 ({}), {}, {}, {} 부터 아주 복잡한 패턴까지 설정 가능
Easily Memorizable	각 Integer Set은 패턴 또는 도형으로 시각적으로 기억할 수 있으므로 사용자가 암기하기 쉬움
Easy to adapt	기존의 사용하던 키패드는 그대로 두고 소프트웨어적으로만 변경하는 것이므로 새로 기기를 구매할 필요 없이 소프트웨어 변경만으로 적용 가능
Hard to Breach	각 Integer Set마다 2^9 개의 경우의 수가 가능하므로 비밀번호는 총 $2^{36} = 68719476736$ 의 경우가 가능 (일반 4자리 비밀번호의 경우 6561의 경우가 가능)

이전 사진과 같이 4개의 숫자가 사용됨을 알고 있을 때:

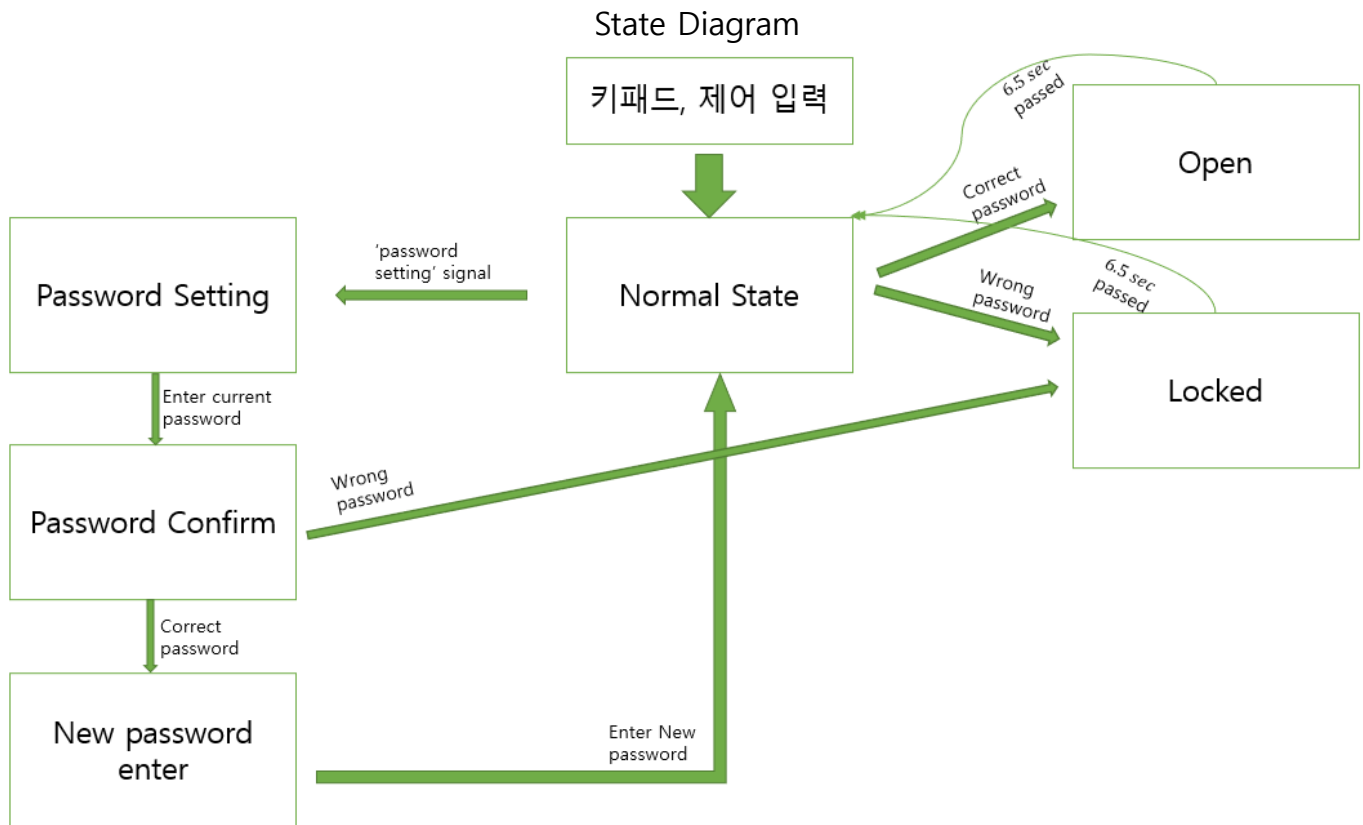
일반 4자리 비밀번호: $4 \times 4 \times 4 \times 4 = 256$ 개의 경우

Non-Numeral : $2^4 \times 2^4 \times 2^4 \times 2^4 = 65536$ 개의 경우

프로젝트 상세 설명 및 구현

다음과 같은 동작을 지원한다: password enter기능, password setting 기능, undo 기능
Servo Motor를 구동하기 위해서 정격 클럭은 10kHz로 설정했다

1. 기본 작동구조



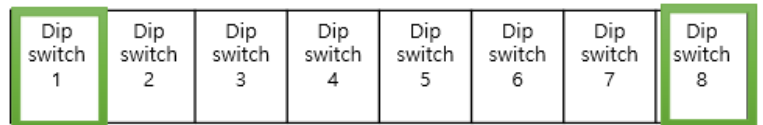
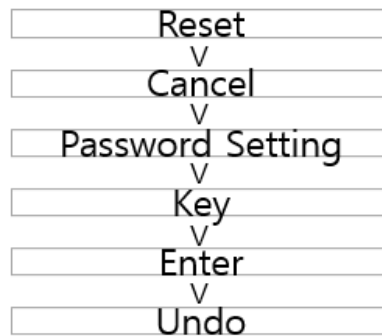
- Normal State, Password Setting, Open, Close로 구분된다.
- Normal State에서 Password 입력이 맞으면 Open, 틀리면 Locked state가 된다
- Normal State에서 Password Setting 버튼이 눌리면 Password Setting state가 된다
- Password Setting state에서는 먼저 현재 비밀번호를 한번 더 입력하게 한다. 만약 맞으면 새로운 비밀번호를 입력하고, 틀리면 Locked state가 된다.
- Open, Locked state에서는 6.5초가 지나면 Normal State가 된다

입력 배치



Key(1~9): 1에서 9까지의 입력을 제공한다.
 Undo(*): 현재 입력중인 내용을 취소하거나, 이전 입력을 지우고 다시 입력한다
 Enter(0): 현재 입력중인 integer set을 확정짓고 다음 번째로 넘어간다.
 Cancel(#): 현재 작업중인 모든 내용을 중지하고 normal state로 돌아간다.

hierarchy of buttons:

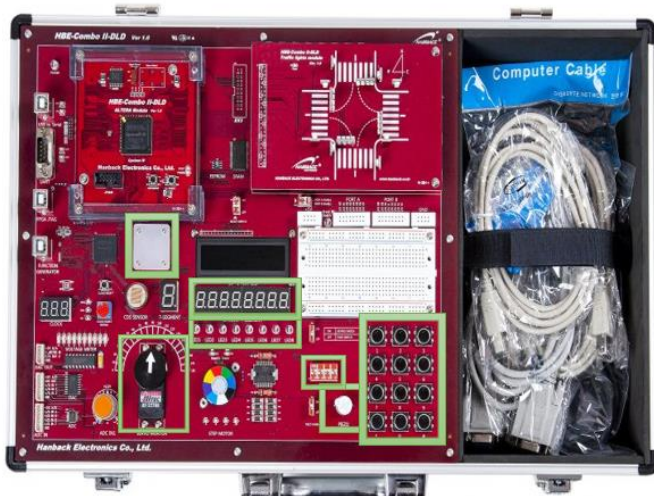


Dip Switch 1: Password Setting 버튼
 Dip Switch 8: Reset버튼 (positive reset)

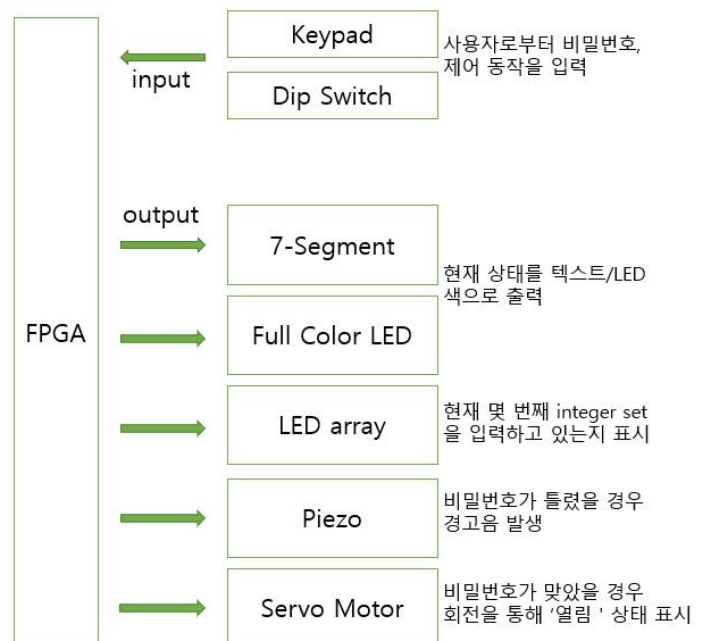
비밀번호는 Key-Enter-Key-Enter-Key-Enter-Key-Enter 순으로 입력된다.

마지막 Enter가 눌렸을 때 현재 비밀번호와 대조한 뒤 Open 또는 Closed State로 이동한다

출력 배치



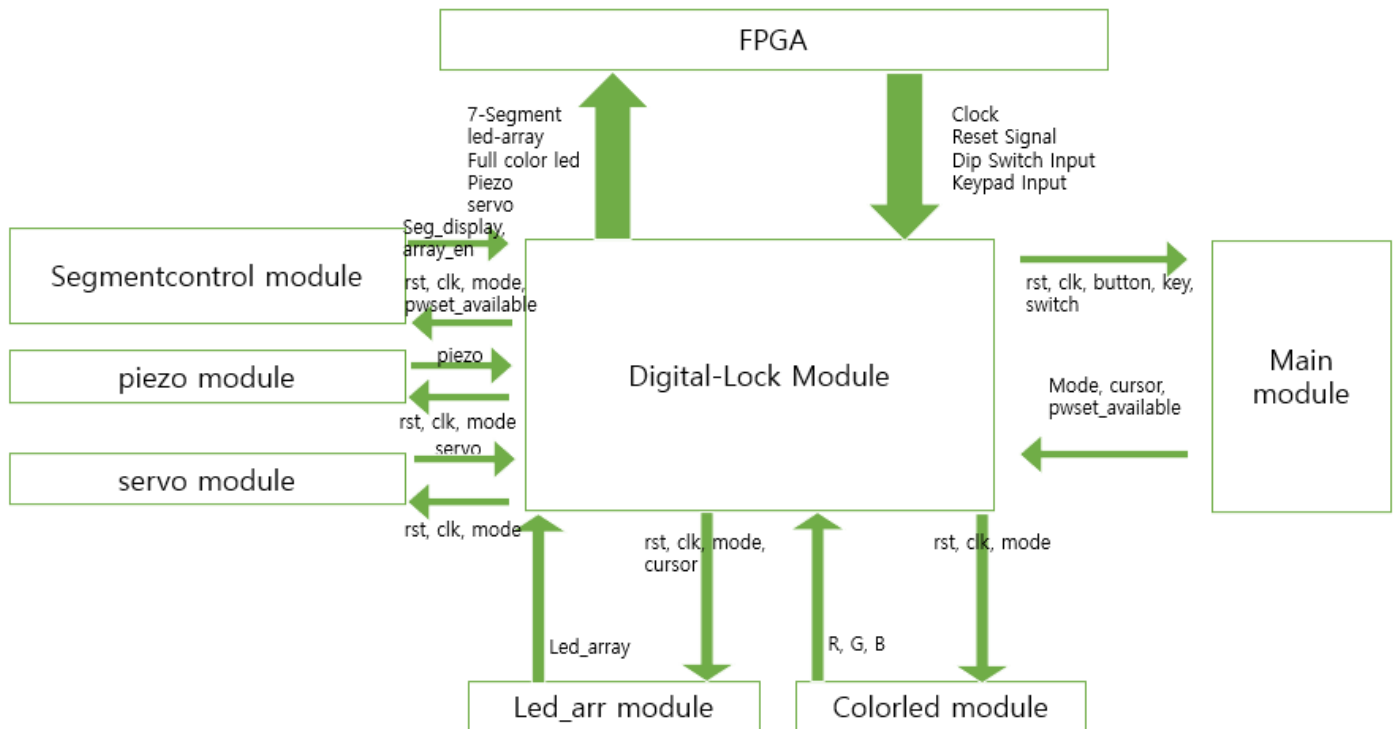
Input: Keypad, Dip Switch
 Output: 7-segment, LED array, full color LED, piezo, servo



2. 소프트웨어 구현구조

Module의 구성

*Digital_lock module은 단순히 하위 모듈들을 연결하는 역할만 한다

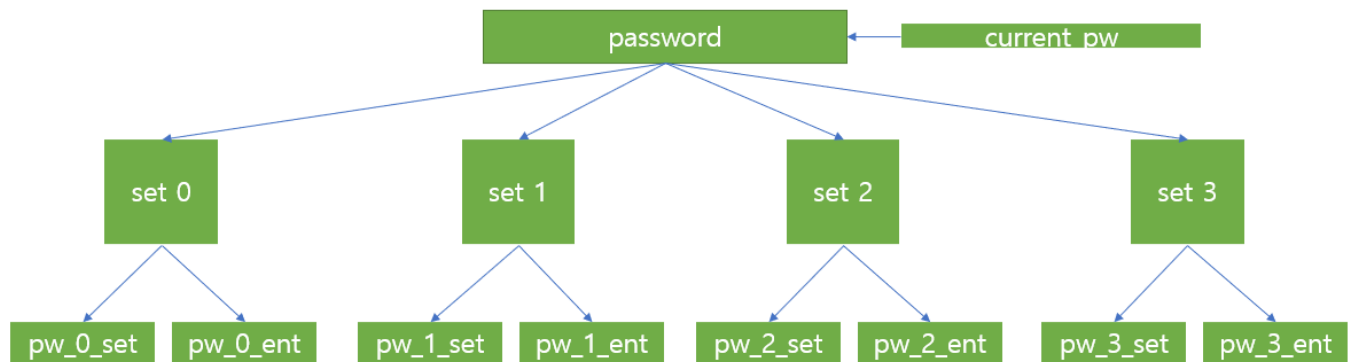


주요 wire

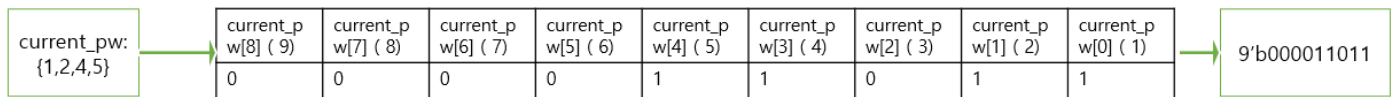
mode	<p>현재 State를 나타낸다</p> <p>2'b00: Normal State</p> <p>2'b01: Password Setting</p> <p>2'b10: Open</p> <p>2'b11: Locked</p>
cursor	<p>현재 입력되고 있는 비밀번호 위치를 나타낸다</p> <p>2'b00: 1st password being entered</p> <p>2'b01: 2nd password being entered</p> <p>2'b10: 3rd password being entered</p> <p>2'b11: 4th password being entered, open, close state</p>
pwset_available	password setting state 여부를 나타낸다

R, G, B	<p>현재 state에 따라 Full Color LED의 색을 달리한다.</p> <p>Normal State: White</p> <p>Password Setting: Blue</p> <p>Open: Green</p> <p>Lock: Red</p>
led_array	<p>현재 입력되고 있는 비밀번호 위치에 따라 LED array의 켜짐을 달리한다</p> <p>. LED 표시</p> <p>1st password: ○○○○</p> <p>2nd password: ●○○○</p> <p>3rd password: ●●○○</p> <p>4th password: ●●●○</p> <p>open, close state: ●●●●</p>
array_en	7-Segment의 enable을 담당한다
seg_display	<p>7-Segment에 전시되는 Data를 담당한다</p> <p>현재 State에 따라서 표시를 달리한다</p> <p>Normal State "ENTER"</p> <p>Password Setting(현재 비밀번호 확인) "ENTER PW"</p> <p>Password Setting(새 비밀번호 입력) "NEW PW"</p> <p>Open "OPEN"</p> <p>Locked "LOCK"</p>
piezo	mode가 Lock일 때 경고음을 낸다
servo	mode가 Open일 때 모터를 180도에 위치시킨다

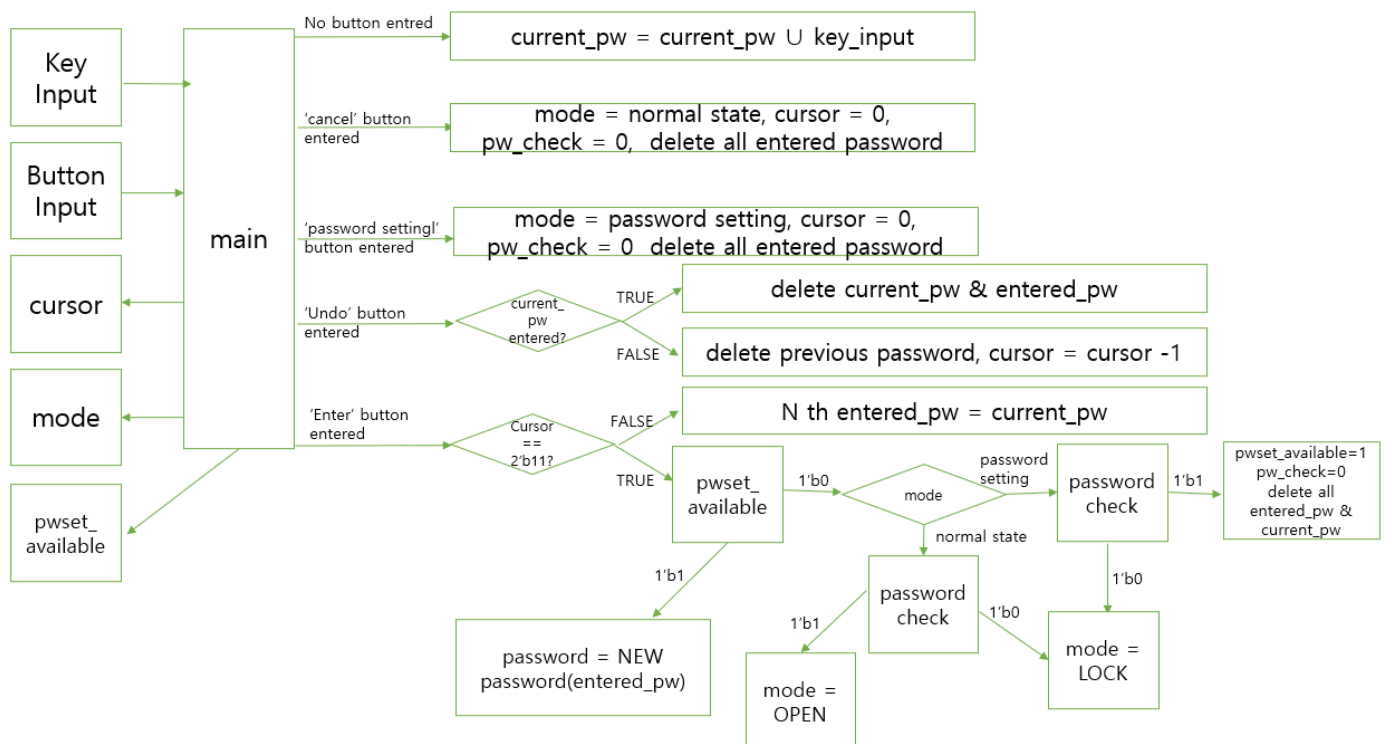
password의 구성



pw_0_set	설정된 password
pw_0_ent	사용자가 입력한 password
current_pw	사용자가 현재 실시간으로 입력중인 내용
*pw_n_set, pw_n_ent, current_pw는 모두 9bit의 크기이다	

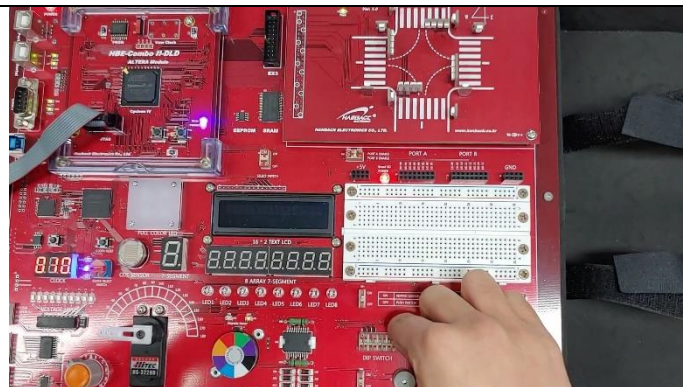


main module의 flowchart

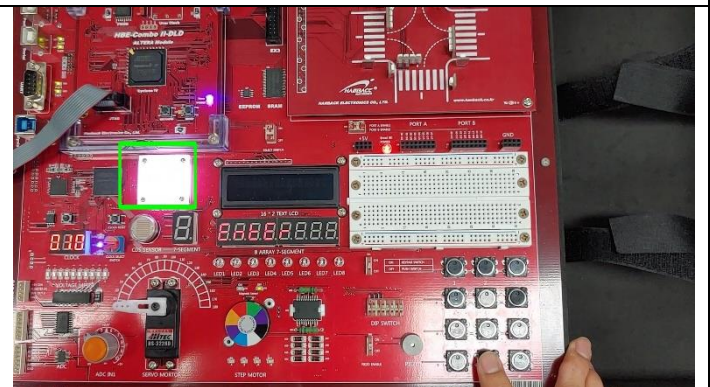


구현 결과

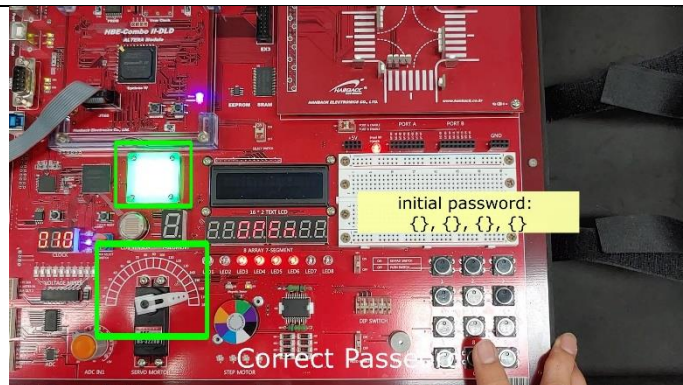
Reset On 상태



Normal State



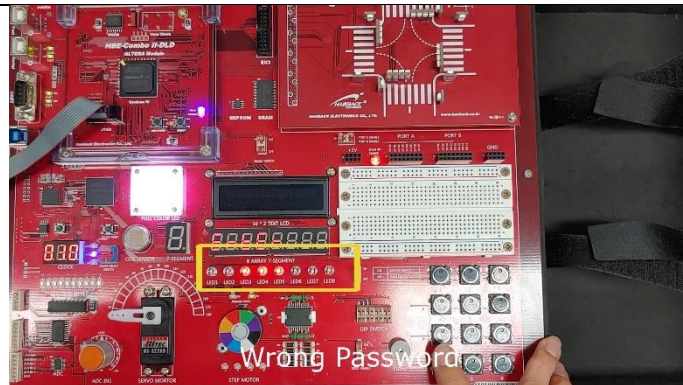
Open State



Lock State



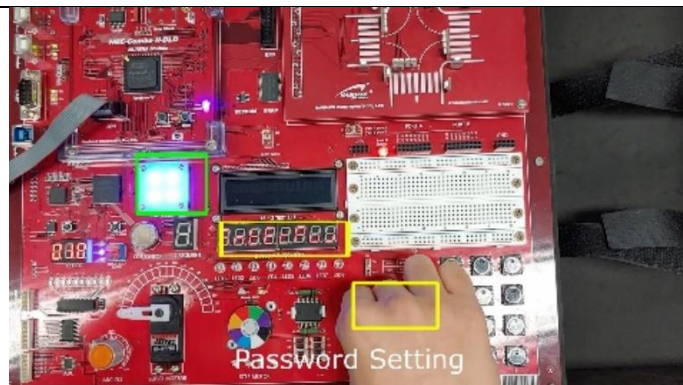
Wrong Password Entered



Undo & Correct Password



Password Setting



Enter New Password 1



Enter New Password 2



Enter New Password 3



Enter New Password 4



Enter Changed Password



Enter Previous Password



Cancel During Password Setting



Reset During Password Entering



Initial Password Entered



seq_led_array[2]	Output	PIN_L4	2	B2_N1	PIN_L4	2.5 V (default)	8mA
seq_led_array[1]	Output	PIN_L2	2	B2_N1	PIN_L2	2.5 V (default)	8mA
seq_led_array[0]	Output	PIN_L1	2	B2_N0	PIN_L1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_P5	2	B2_N1	PIN_P5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_L5	2	B2_N1	PIN_L5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_R1	2	B2_N0	PIN_R1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_P2	2	B2_N0	PIN_P2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_N1	2	B2_N0	PIN_N1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_M1	2	B2_N0	PIN_M1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_P2	2	B2_N0	PIN_P2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_P3	2	B2_N1	PIN_P3	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J7	1	B1_N2	PIN_J7	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J6	1	B1_N1	PIN_J6	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J5	1	B1_N2	PIN_J5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_K8	1	B1_N2	PIN_K8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_E3	1	B1_N0	PIN_E3	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_F2	1	B1_N1	PIN_F2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_G5	1	B1_N0	PIN_G5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_H8	1	B1_N1	PIN_H8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_AB11	3	B3_N0	PIN_AB11	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J4	1	B1_N3	PIN_J4	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_L6	2	B2_N0	PIN_L6	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J1	1	B1_N3	PIN_J1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J2	1	B1_N3	PIN_J2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_H1	1	B1_N3	PIN_H1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_L7	2	B2_N0	PIN_L7	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_M5	2	B2_N0	PIN_M5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_M2	2	B2_N0	PIN_M2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_M3	2	B2_N0	PIN_M3	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_M4	2	B2_N0	PIN_M4	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_M6	2	B2_N0	PIN_M6	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J3	1	B1_N3	PIN_J3	2.5 V (default)	8mA

seq_led_array[2]	Output	PIN_L8	1	B1_N2	PIN_L8	2.5 V (default)	8mA
seq_led_array[1]	Output	PIN_N6	2	B2_N1	PIN_N6	2.5 V (default)	8mA
seq_led_array[0]	Output	PIN_N8	2	B2_N2	PIN_N8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_AB18	4	B4_N1	PIN_AB18	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_V3	2	B2_N3	PIN_V3	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_D2	1	B1_N1	PIN_D2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_C2	1	B1_N1	PIN_C2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B2	1	B1_N0	PIN_B2	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_H10	8	B8_N1	PIN_H10	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_J8	1	B1_N1	PIN_J8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_H5	1	B1_N2	PIN_H5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_H6	1	B1_N1	PIN_H6	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_V15	4	B4_N2	PIN_V15	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_U1	2	B2_N1	PIN_U1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_Y1	2	B2_N2	PIN_Y1	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_A5	8	B8_N2	PIN_A5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_A8	8	B8_N0	PIN_A8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B4	8	B8_N2	PIN_B4	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B10	8	B8_N0	PIN_B10	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_C8	8	B8_N1	PIN_C8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B5	8	B8_N2	PIN_B5	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_C10	8	B8_N0	PIN_C10	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B6	8	B8_N1	PIN_B6	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B8	8	B8_N1	PIN_B8	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B7	8	B8_N1	PIN_B7	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_B9	8	B8_N0	PIN_B9	2.5 V (default)	8mA
seq_led_mat_switch	Input	PIN_A3	8	B8_N3	PIN_A3	2.5 V (default)	8mA

결과 및 토의

Conclusion

Non-Numeral Digital Lock

- 비밀번호 입력, 설정, 입력취소, reset 모두 기획했던 대로 정상적으로 작동함
- Behavioral Modeling을 주로 사용하였고, 복잡한 작업은 타 모듈 또는 함수로 분리하여 적용함
- 대부분의 기능은 FPGA 보드에서 구현하였지만, MCU 와 기타 입출력장치를 이용하여 구현하여 현실에 확장할 수 있을 것으로 보임
- 프로젝트 과정을 통해 디지털 회로 설계에 대한 기초적인 경험을 쌓을 수 있었음

Required Improvements

readability

실시간 입력 내용을 확인하기 어려움

->사용자가 자신의 입력 내용을 시각적으로 볼 수 있는 방법이 필요

	column 1	column 2	column 3
row1			
row2			
row3			

← {1, 2, 3, 4}를 입력했을 때의 예시

Data Security

password 데이터가 별다른 암호화 없이 저장됨

->hash function 등을 이용한 데이터의 암호화가 필요

Quick
entering time

기존의 4자리 비밀번호 방식보다 입력하는데 걸리는 시간이 훨씬 오래 걸림

-> 이는 버튼 식 키패드로 구현했기 때문이며 터치패드를 사용하고 swipe-gesture를 통해서 입력하면 기존 방식보다 입력시간을 줄일 수 있을 것으로 보임

Power saving

기기가 항상 Normal State에 존재

->'저전력 대기 모드'의 추가가 필요

User Settings

Open, Lock의 유지 기간, LED 색, 경고음 등을 사용자가 설정 할 수 있는 방법이 필요함