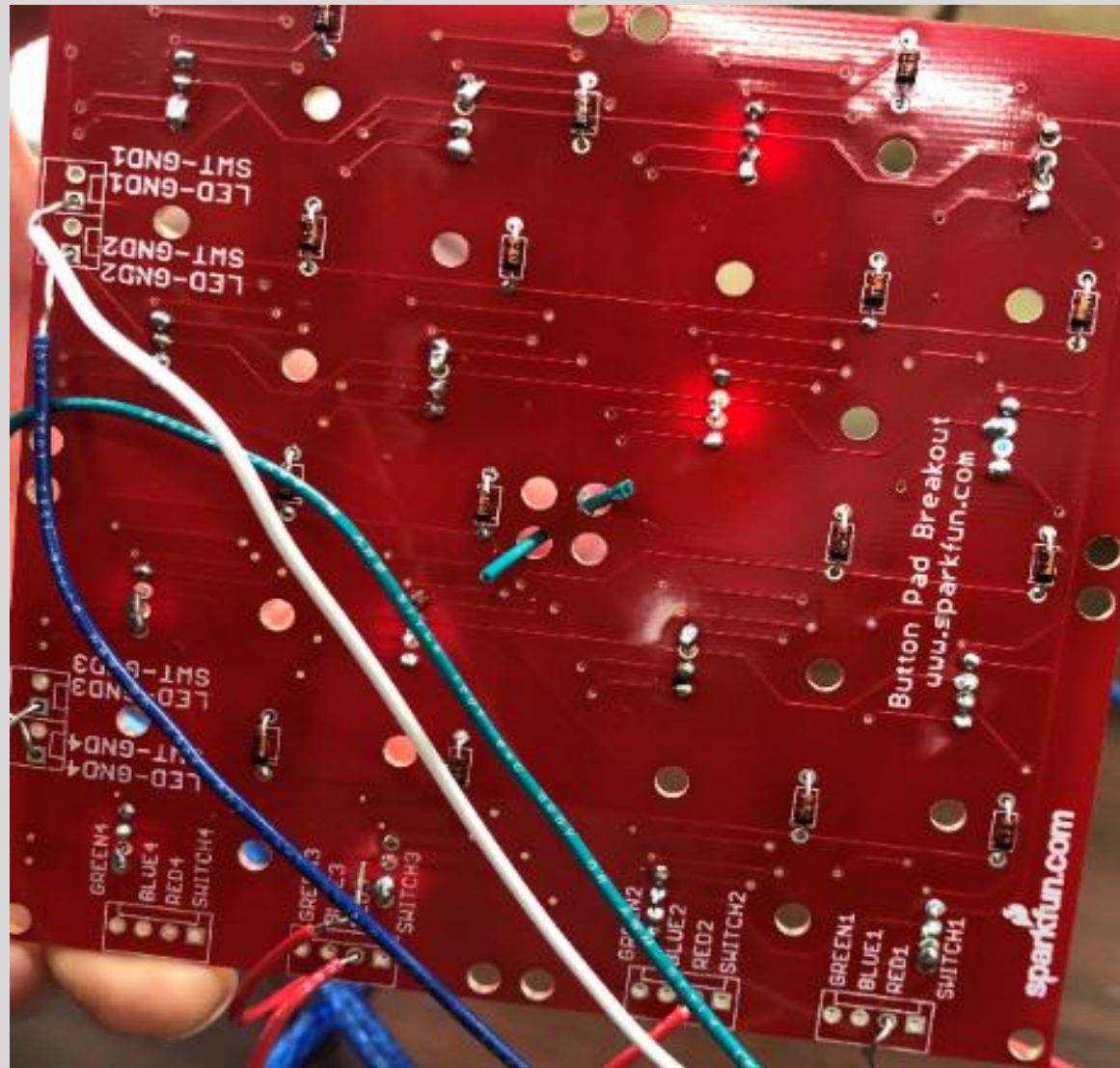
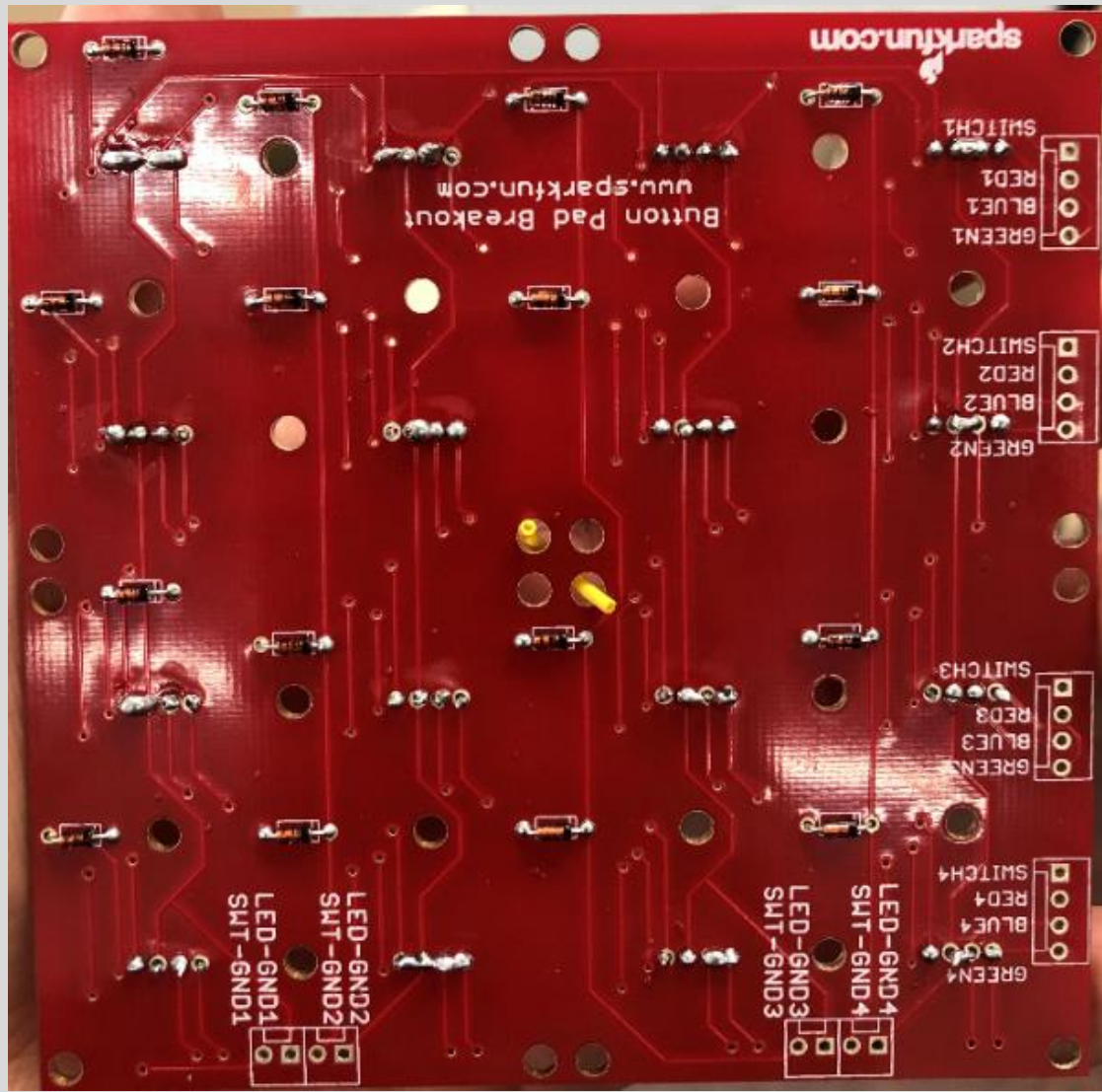
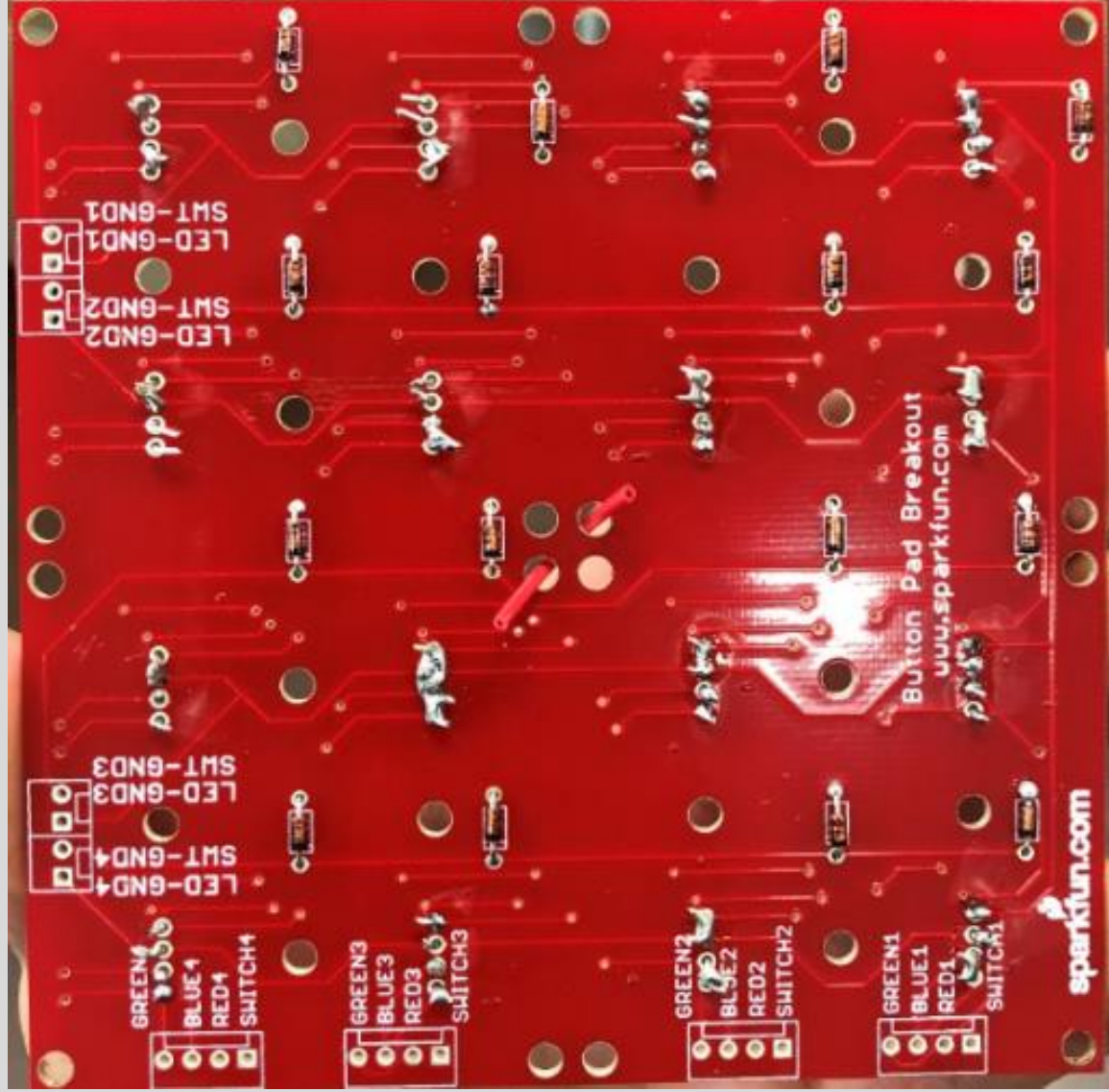
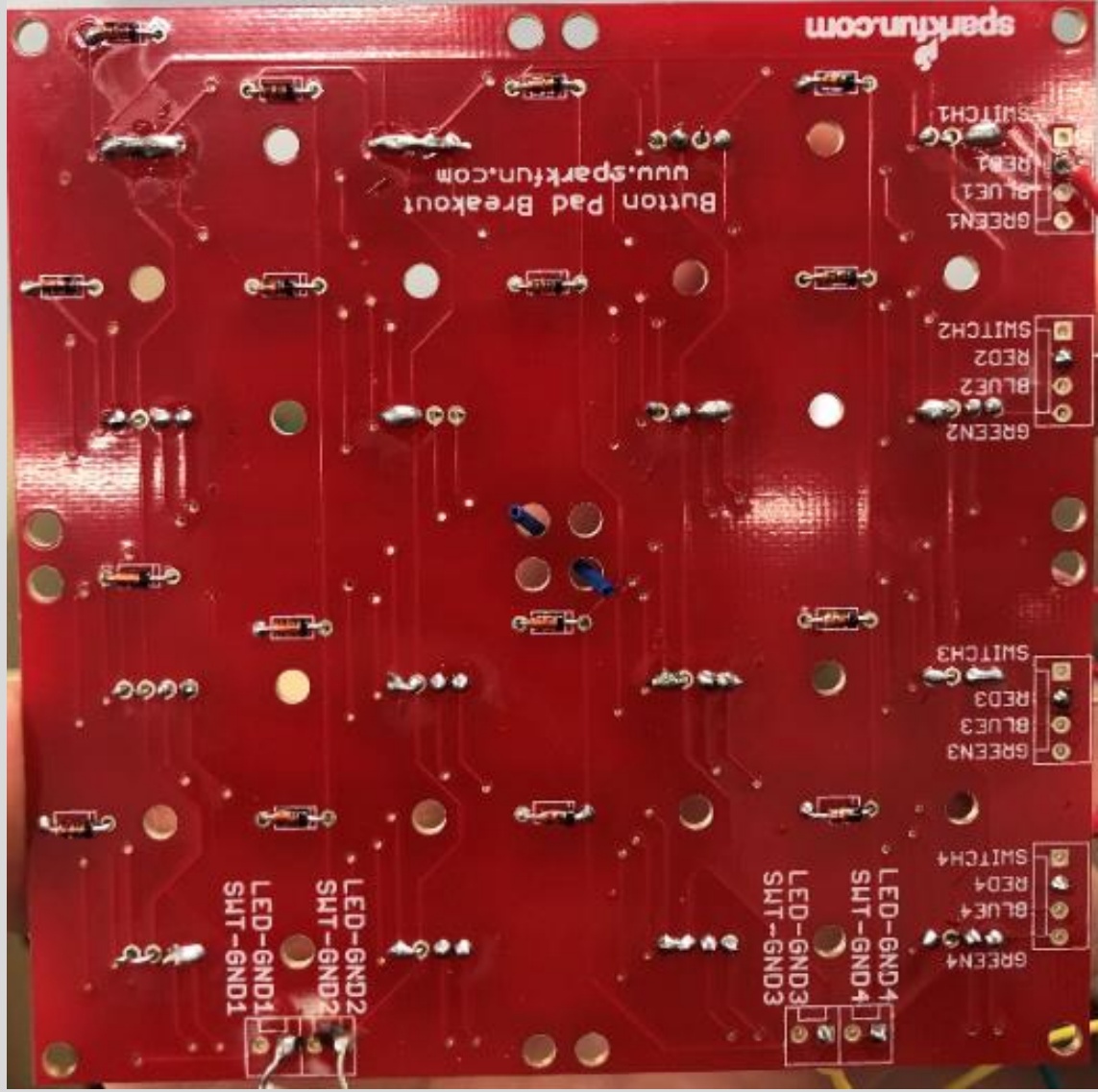


'지뢰 찾기' 진행 상황

복전과생

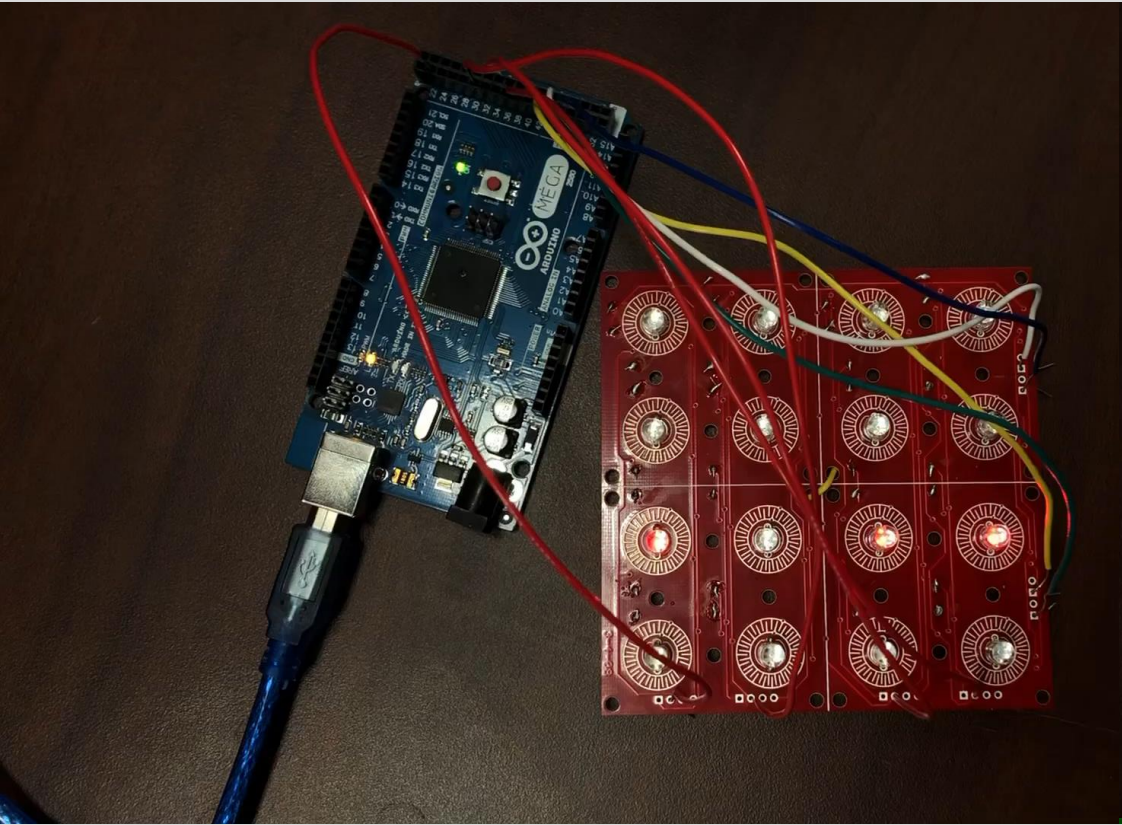
—
김병민
김수연
손병수
이예은



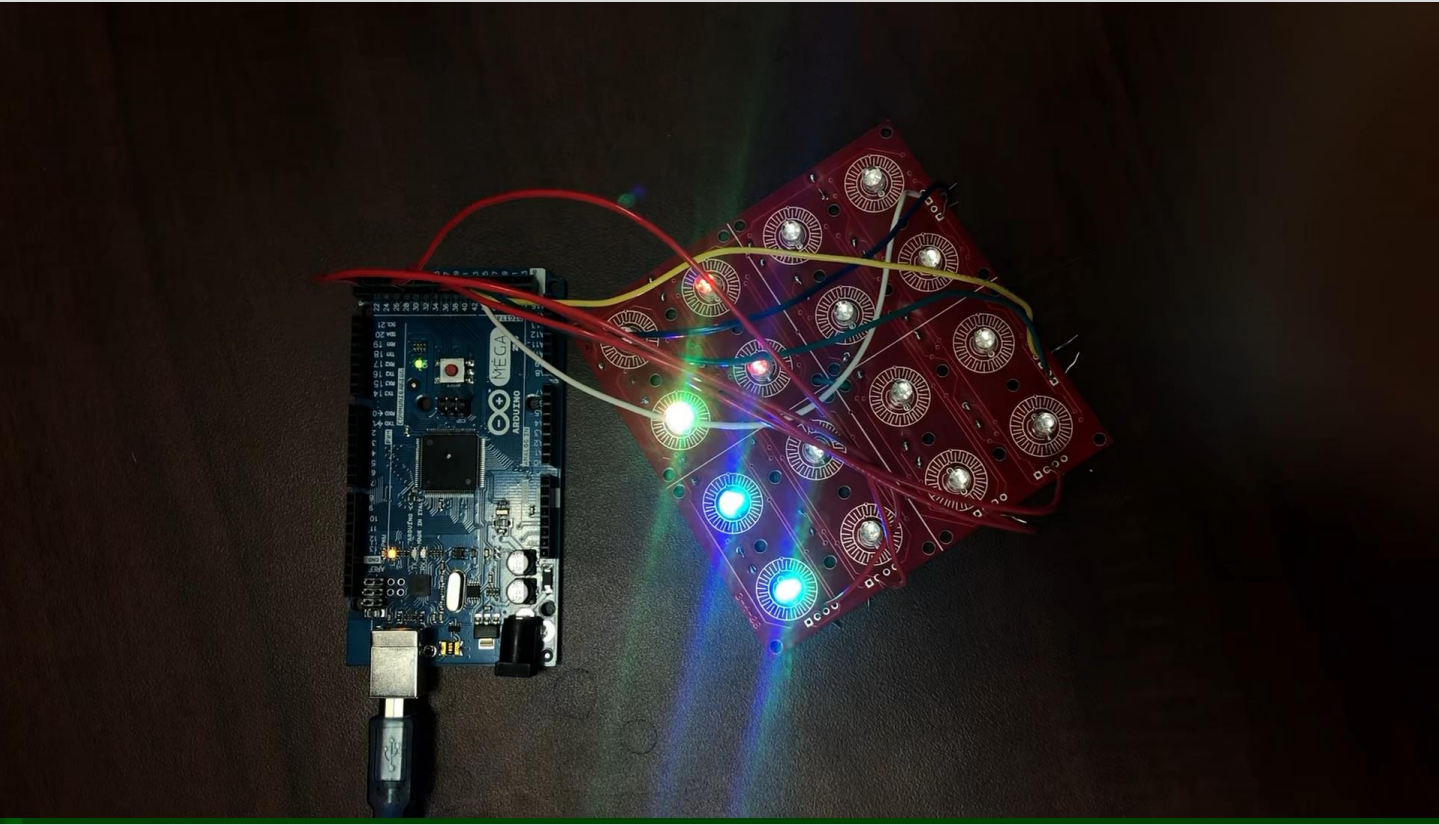


💣 납땜 확인

기판 #1



기판 #2

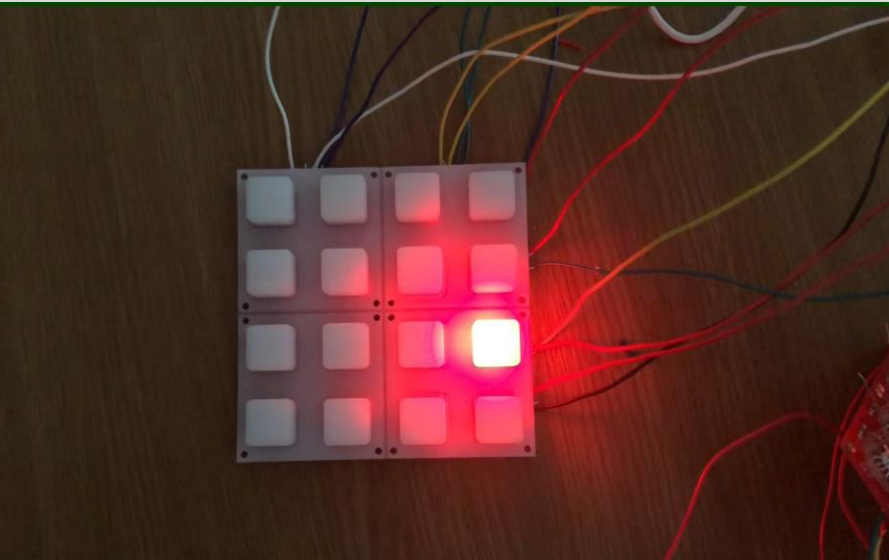


* 정상 작동시 각 LED에 순서대로 빨간색이 켜졌다가 꺼짐

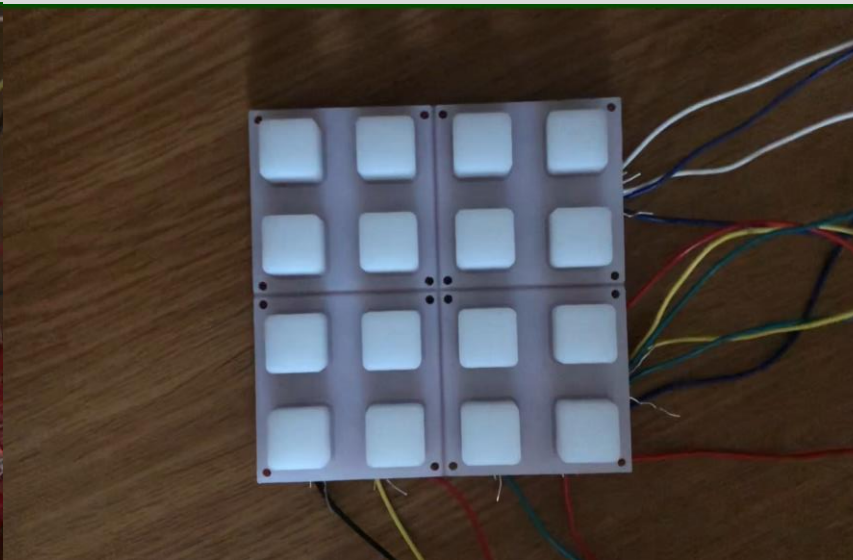
기판 #3, 4 - 아예 불이 안 들어옴

💣 재납땜 (1) - 정상 기판

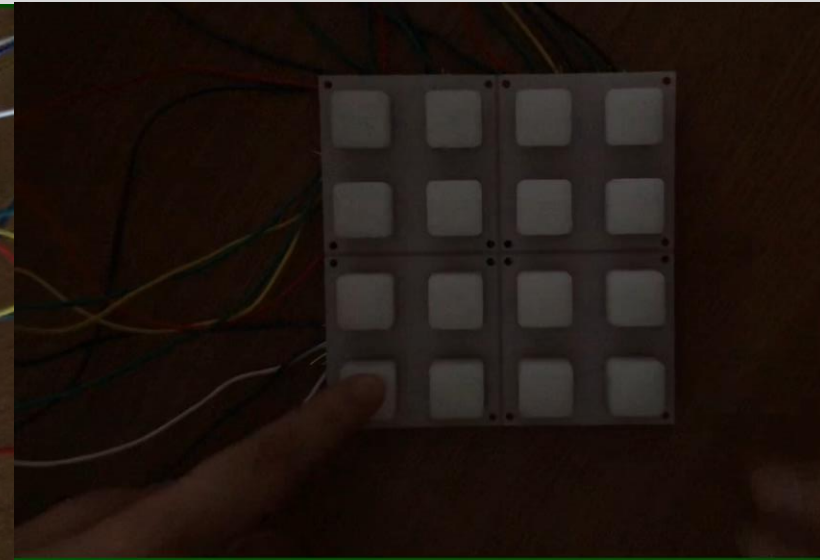
기판 #4



[실행 1]
순서대로 빨간색 점멸



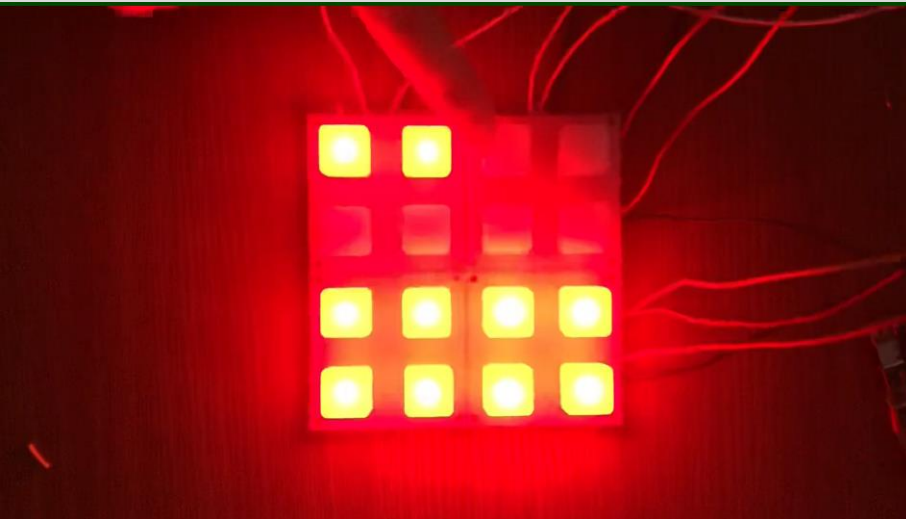
[실행 2]
버튼을 누르면 출력



[실행 3]
버튼을 누를 때 순서대로
'빨, 초, 파' 출력

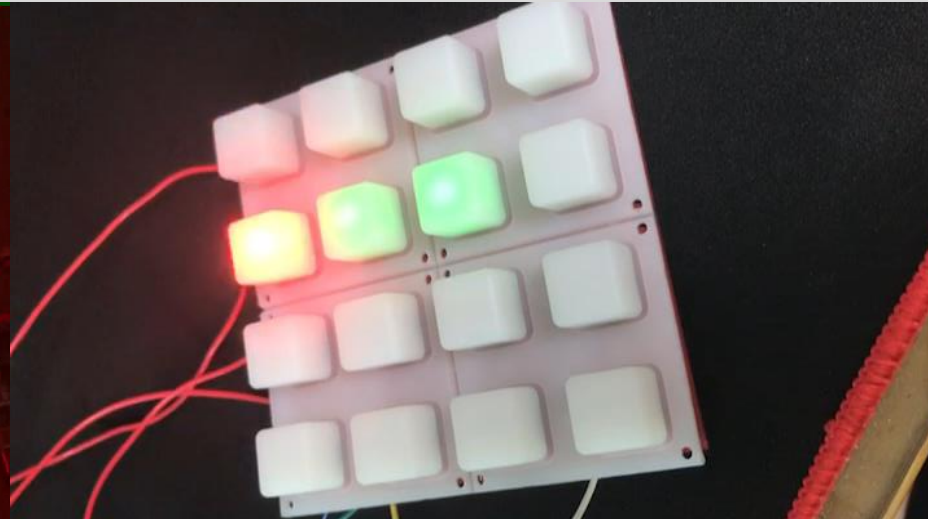
💣 재납땜 (1) - 문제 기판

기판 #1



- 눌러도 반응x
- 다른 LED에 영향을 미침

기판 #2



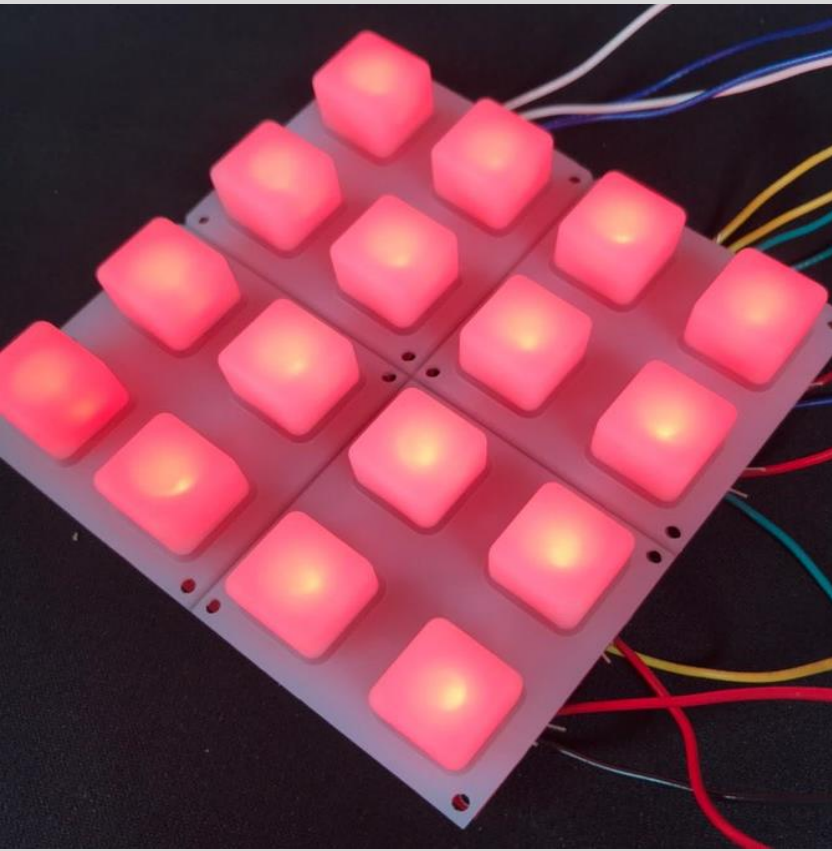
- 초록불이 3개 들어와 있음

기판 #3

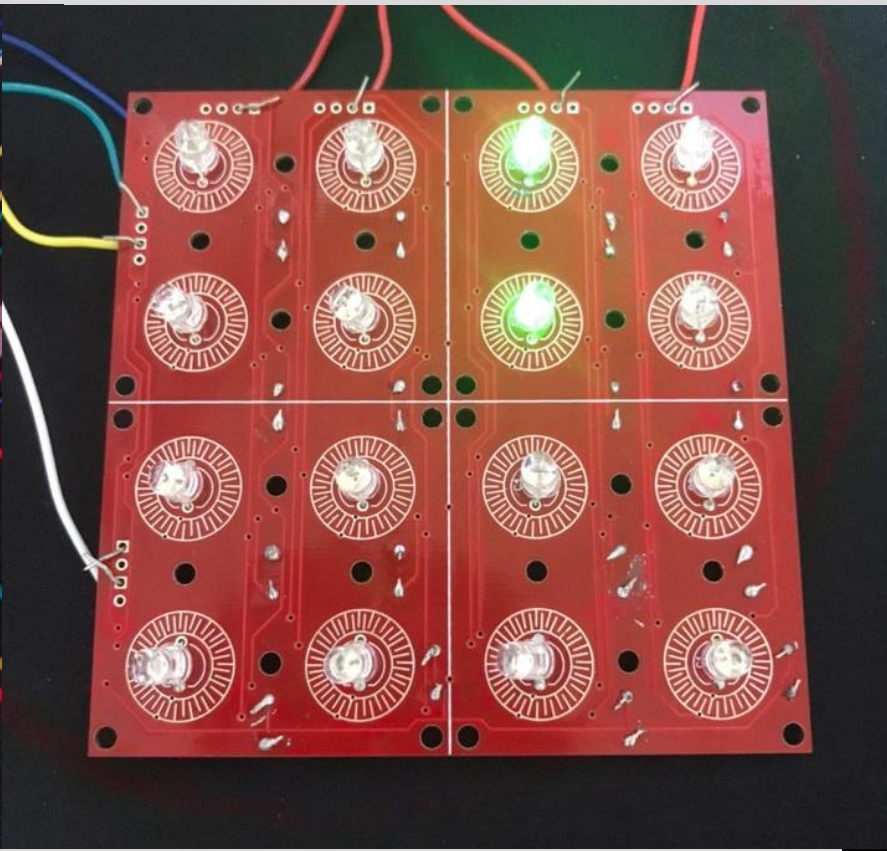
- 여전히 불이 안 들어옴

💣 재납땜 (2)

기판 #1 - 정상



기판 #2

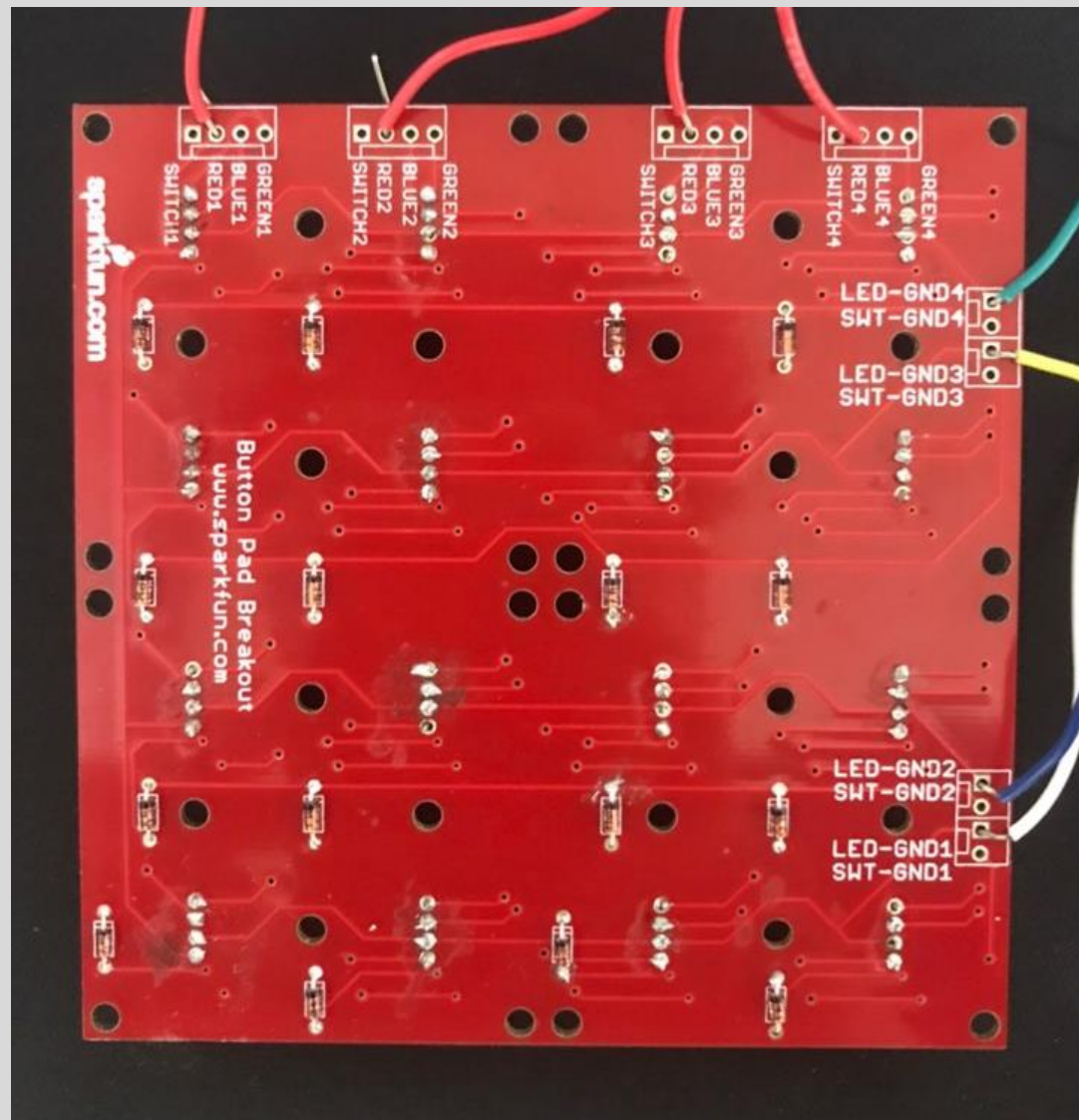
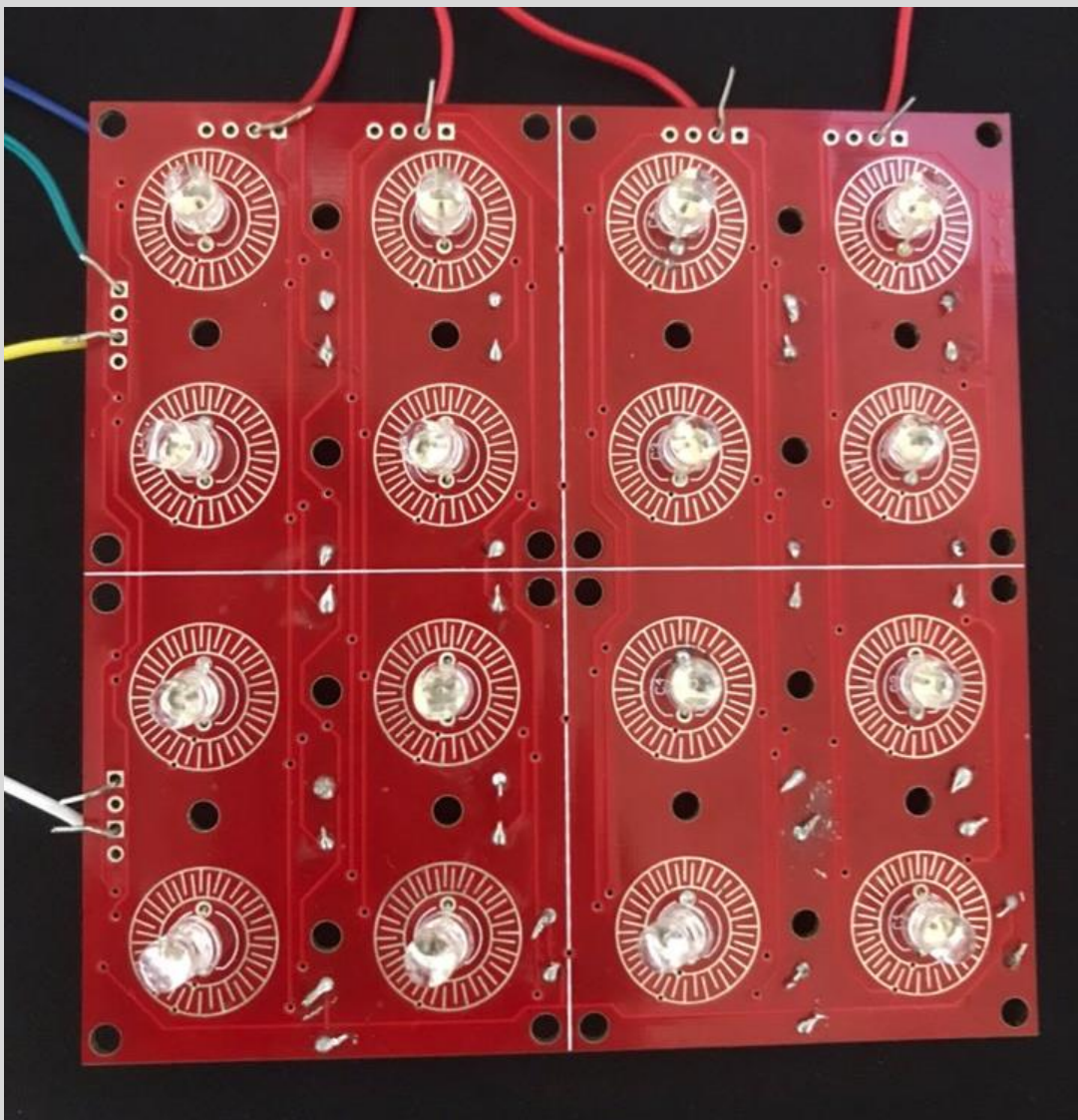


기판 #3 - 정상



- 초록불 3개 -> 2개

💣 재납땜 (3) - 문제 기판





LED 색상 변경

```
#define MAX_DEBOUNCE (3)
```

```
// Global variables
```

```
static uint8_t LED_outputs[NUM_LED_COLUMNS][NUM_LED_ROWS];
```

```
static int32_t next_scan;
```

```
static const uint8_t btnselpins[4] = {50,51,52,53};
```

```
static const uint8_t btnreadpins[4] = {46,47,48,49};
```

```
static const uint8_t ledselpins[4] = {42,43,44,45};
```

```
// RGB pins for each of 4 rows
```

```
//static const uint8_t colorpins[4][3] = {{22,24,26}, {30,31,32},{33,34,35},{36,37,38}};
```

```
static const uint8_t colorpins[4][3] = {{22,26,24}, {30,32,31},{33,35,34},{36,38,37}};
```

```
|
```

```
static int8_t debounce_count[NUM_BTN_COLUMNS][NUM_BTN_ROWS];
```

```
static void setuppins()
```

```
{
```

```
    uint8_t i;
```

```
static void scan()
```

```
{
```

```
    static uint8_t current = 0;
```

```
    uint8_t val;
```

```
    uint8_t i, j;
```

```
|
```

```
    //run
```

```
    digitalWrite(btnselpins[current], LOW);
```

```
    digitalWrite(ledselpins[current], LOW);
```

```
    // choose color
```

```
    for(i = 0; i < NUM_LED_ROWS; i++)
```

```
    {
```

```
        uint8_t val = (LED_outputs[current][i] & 0x03);
```

```
        if(val)
```

```
        {
```

```
            digitalWrite(colorpins[i][val-1], HIGH);
```

```
            digitalWrite(colorpins[i][val-1], HIGH);
```

```
            if(val == 2)
```

```
                digitalWrite(colorpins[i][0],HIGH);
```

```
            if(val == 3)
```

```
                digitalWrite(colorpins[i][0],HIGH);
```

```
        }
```

```
    }
```

```
    delay(1);
```

```
    for( j = 0; j < NUM_BTN_ROWS; j++)
```

```
    {
```

```
        val = digitalRead(btnreadpins[j]);
```



colorpins[4][3]의 핀 번호 순서 변경

기존 : 빨 -> 파 -> 초

변경 : 빨 -> 초 -> 파

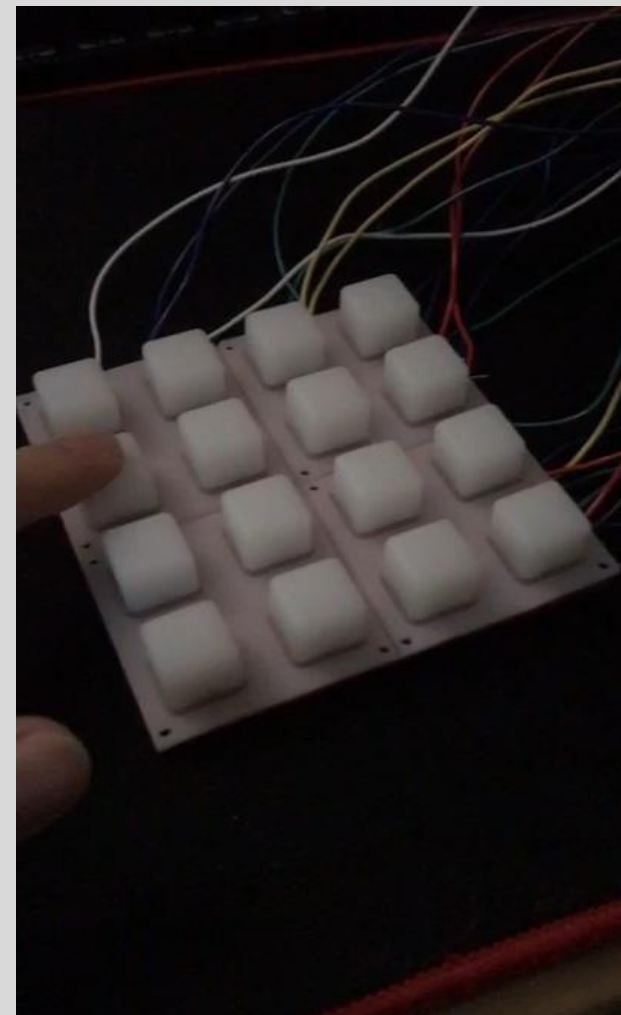


val(색상이 켜지는 순서) 값을 통해 색상 조합

val == 1 빨[i][val-1]

val == 2 파[i][val-1] + 빨[i][0] = 분홍

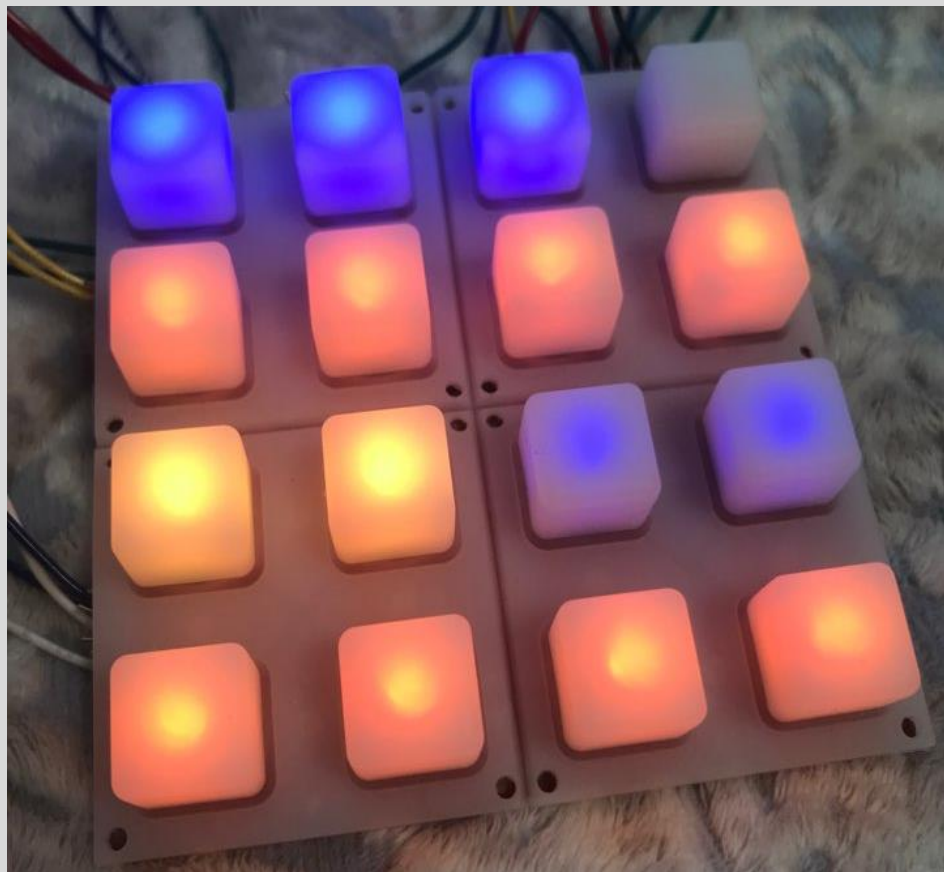
val == 3 초[i][val-1] + 빨[i][0] = 노랑





LED 출력 문제 (진행중)

1. 밝기 문제



2. 더 다양한 색

빨강 플레이어가 버튼을 클릭했을 때

3-1. 플레이어 지뢰와 거리가 1~2 인 버튼

-> 버튼 LED 에 `rgb(255, 0, 0)` 들어온다.

3-2. 플레이어 지뢰와 거리가 3~4 인 버튼

-> 버튼 LED 에 `rgb(255, 99, 99)` 들어온다.

3-3. 플레이어 지뢰와 거리가 5~6 인 버튼

-> 버튼 LED 에 `rgb(255, 198, 198)` 들어온다.

파랑 플레이어가 버튼을 클릭했을 때

4-1. 플레이어 지뢰와 거리가 1~2 인 버튼

-> 버튼 LED 에 `rgb(0, 0, 255)` 들어온다.

4-2. 플레이어 지뢰와 거리가 3~4 인 버튼

-> 버튼 LED 에 `rgb(99, 99, 255)` 들어온다.

4-3. 플레이어 지뢰와 거리가 5~6 인 버튼

-> 버튼 LED 에 `rgb(198, 198, 255)` 들어온다.