

2019 기초설계 2조 4주차 발표



일주일간 공부한 것들...

01

풀업 / 풀다운 저항

- 아두이노 내장 풀업 이용
- 회로에서의 의미와 원리 이해

02

PWM

analogWrite()의
구체적인 원리 이해

03

map() 함수

- map() 함수의 원리와 개념
- 조도 센서를 이용한 활용 예시

04

종합 프로젝트

- 프로그래밍 실습
- 아날로그 입력과 출력을
이용한 심화 예제

개념 공부

함께 정해 두었던 진도와 교안으로 각자 학습 후, 공부한 내용을 문서로 정리하고 기초적인 실습을 한 동영상과 코드를 올렸습니다.

5강 led의 밝기 조절하자 (analog in)

포토셀- 빛의 세기에 따라 저항값 달라진다.

포토레지스터로 LED 밝기 조절하는 실습하는데 LED 밝기가 너무 낮아서 실습이 어려웠다.

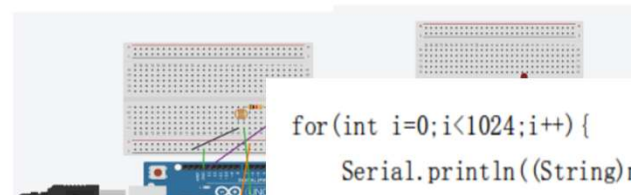
5v 끌어오는 이유? 포토셀에 전원공급하기 위해서.

아날로그 전압측정기로 생각.

실험이 제대로 되지 않음. 왜일까? 캐드먼저 작성하고 회로로 구현 해보려 해도 마찬가지로 //

<http://www.arduino.cc/en/Guide/Troubleshooting#upload> 같은 오류 발생.

5강 내용으로 응용센서랑 led를 여러개 연결하여 보는 예제를 구현중이었는데 갑자기 위와 같은 문제가 있다면서 컴파일링이 안되기 시작했다.



▲ 조도 센서

```
for(int i=0;i<1024;i++){  
  Serial.println((String)map(i, 0, 1023, 0, 512));  
}
```

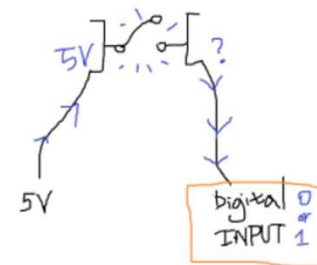
로 간단한 함수를 만들어서 변형 결과

fromHigh의 값이 반이 될 경우 반환되는값이 2배.

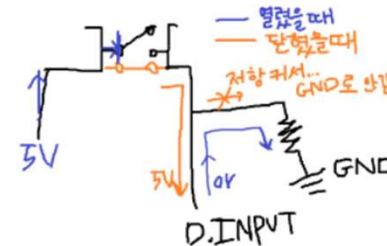
값이 2배가 될 경우 반환되는값이 절반.

fromLow를 1024 fromHigh를 2047로 바꿀시 -512~0까지의 값이 나온다.

위의 두 개의 경우에는 그림으로 대략 이해가 가능하나 마지막 경우가 이해가 부족함.



1 플로팅 노드 상태를 설명하기 위한 그림. 스위치가 닫힐 때 INPUT으로 5V(디지털로는 1)가 들어가는 것은 맞으나, 스위치가 열렸을 때 어디에도 연결되지 않은 것과 마찬가지로 되어 0V도 5V도 아닌 상태가 되어, 값을 0 또는 1로 랜덤하게 받는다. 아두이노 공식 홈페이지는 digitalRead 함수에 대해 이렇게 설명한다. "핀이 아무데도 연결되지 않으면, digitalRead()은 HIGH나 LOW 중 아무 값이나 임의로 반환할 수 있습니다."



< 풀다운 저항 >

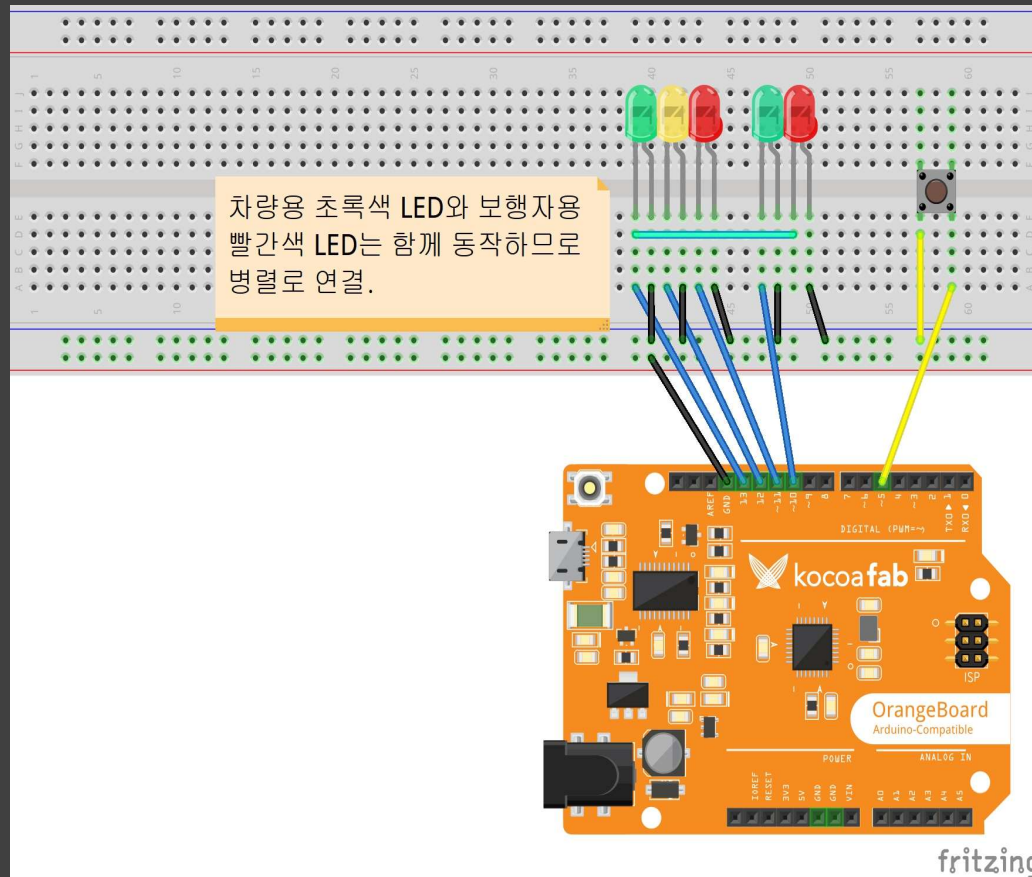
▲ 풀업/풀다운

◀ map() 함수

풀다운 / 풀업 원리 공부

사용 목적, 회로에서의 의미와 작동 원리를 알아보았습니다.

활용 예제: 신호등 게임



```
#define green 13
#define yellow 12
#define red 11

#define walk_green 10

#define ON(x) digitalWrite((x),1)
#define OFF(x) digitalWrite((x),0)

void setup() {
  pinMode(5, INPUT_PULLUP);
  pinMode(green, OUTPUT);
  pinMode(yellow, OUTPUT);
  pinMode(red, OUTPUT);
  pinMode(walk_green, OUTPUT);
}

void loop() {

  if (digitalRead(5) == 0) {
    cross_street();
  }
  else ON(green);
}

void cross_street() {
  OFF(green);
  ON(yellow); delay(1000);
  OFF(yellow); ON(red); ON(walk_green);
  delay(3000);
  for (int i = 0; i < 5; i++) {
    OFF(walk_green); delay(500);
    ON(walk_green); delay(500);
  }
  OFF(walk_green);
  OFF(red);
}
```

map() 함수

맵 함수의 기본 구조

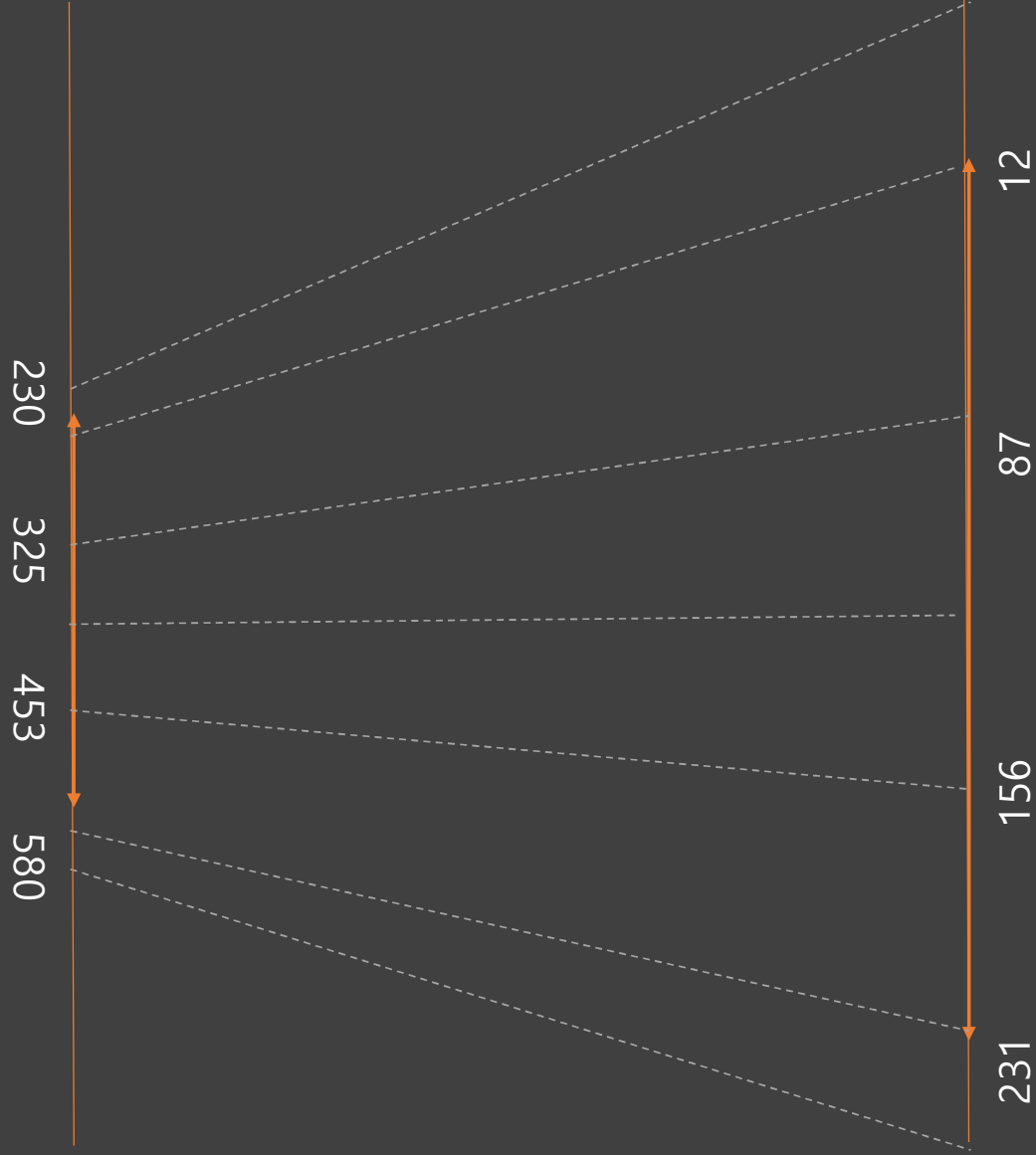
```
long map(long x, long in_min, long in_max, long out_min, long out_max) {  
    return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;  
}
```



$$\text{반환 값} = \text{출력범위 최소값} + (\text{입력 값}(x) - \text{입력 범위 최소값}) \times \frac{\text{출력 값 범위의 길이}}{\text{입력 값 범위의 길이}}$$

map() 함수

맵 함수는 어디에 쓰일까?



map() 함수

맵 함수는 어디에 쓰일까?

아날로그 출력을
단계별로 나누어 사용하는 데에도
유용하지 않을까?

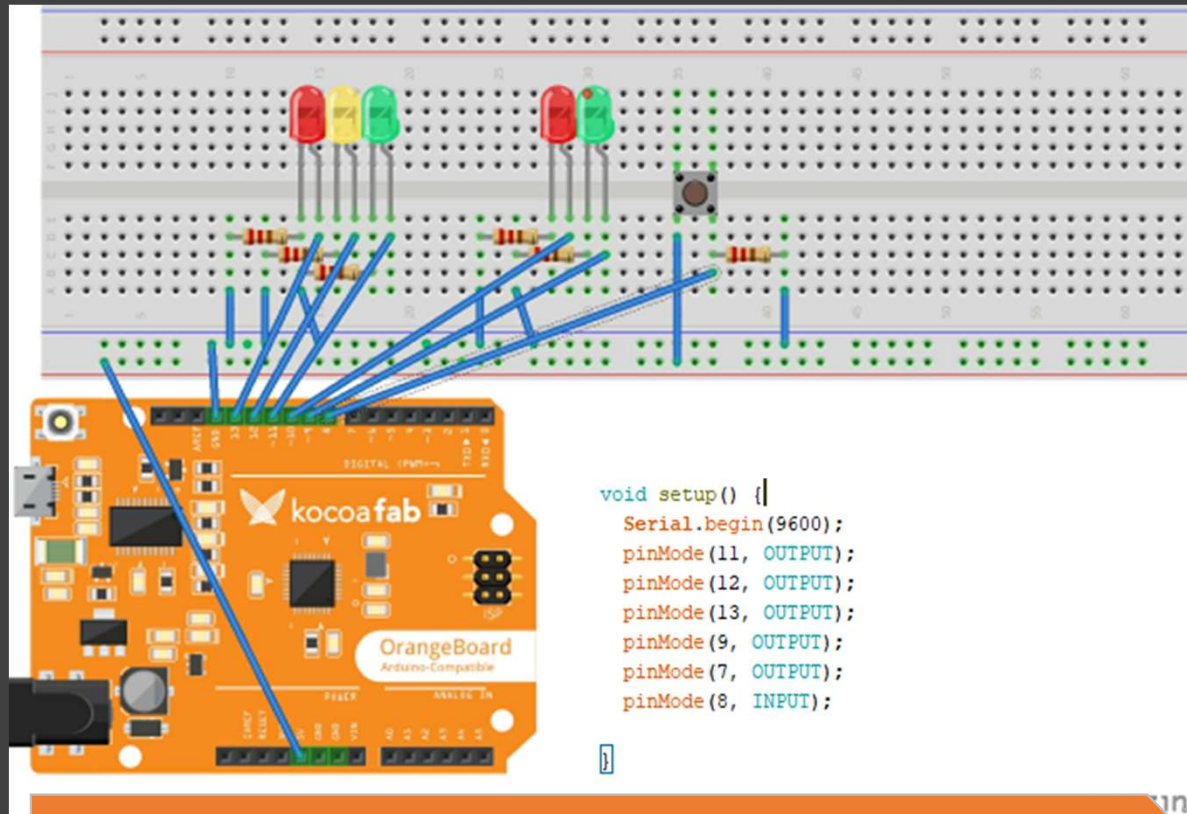
```
void setup() {  
  Serial.begin(9600);  
  pinMode(11, OUTPUT);  
  // Because pin 11 can make PWM OUTPUT  
}  
  
void loop() {  
  int x = Serial.parseInt();  
  Serial.println(x);  
  if(x > 0){  
    for(int i=0; i<=x; i++){  
      analogWrite(11, map(i, 0, x, 0, 255) );  
      delay(700);  
    }  
    digitalWrite(11, LOW);  
  }  
  else;  
}
```

시리얼 모니터에서
숫자를 입력 받아

밝기 단계를 차례로 올린다

응용 프로젝트 1

프로그래밍 연습 - 게임 만들기

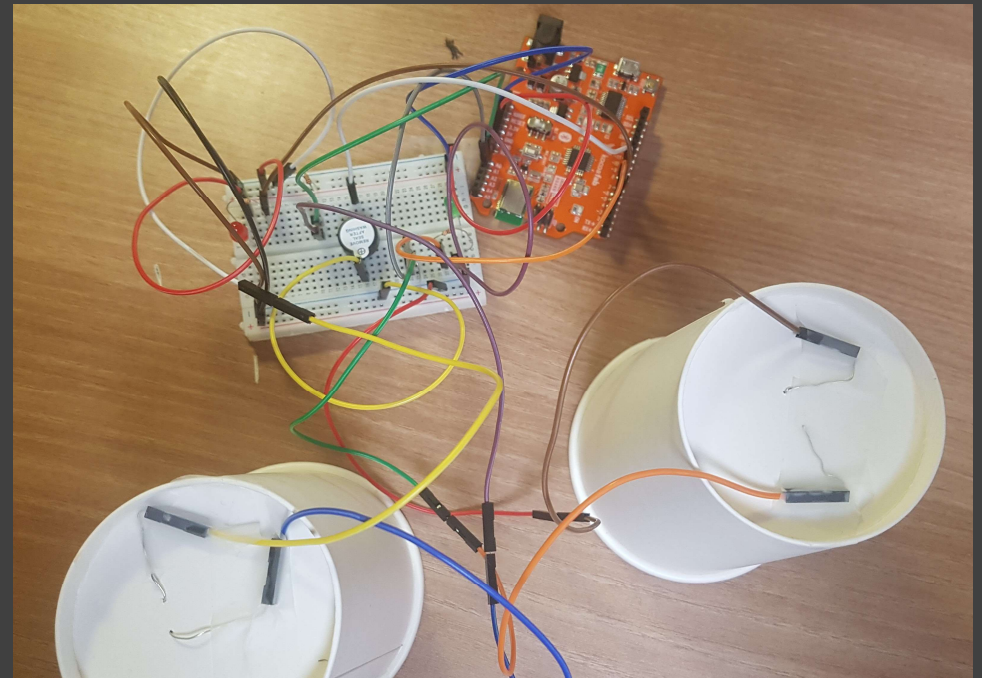
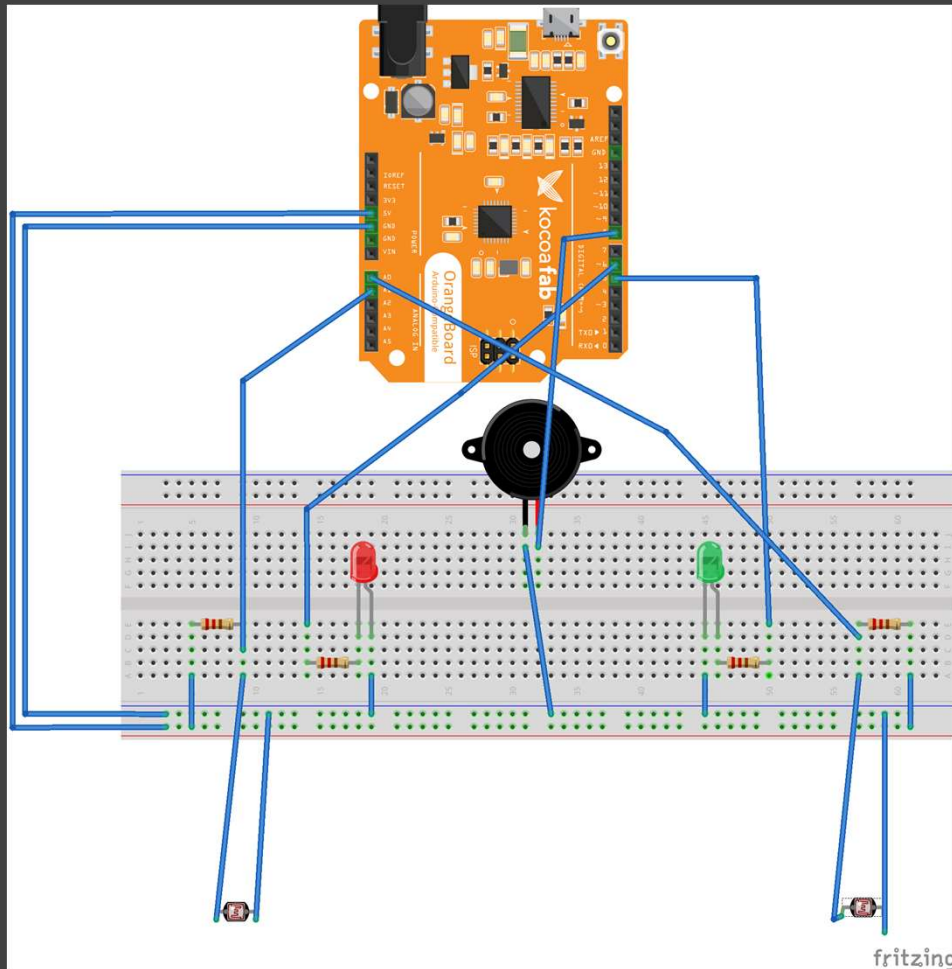


컴퓨터에서 무작위로 LED를 깜빡입니다.
사용자는 깜빡이는 LED의 순서를 기억하고
시리얼 모니터에 그 순서를 정확하게 입력해야 승리합니다.

```
void loop() {  
  // put your main code here, to run repeatedly:  
  if (digitalRead(8)) {  
    int ra = random(5, 8);  
    int que[ra];  
    int an = 0;  
    for (int i = 0; i < ra; i++) {  
      int ra2 = random(11, 14);  
      digitalWrite(ra2, 1);  
      delay(300);  
      digitalWrite(ra2, 0);  
      delay(300);  
      que[i] = ra2 + 38;  
    }  
    for (int i = 0; i < ra; i) {  
      if (Serial.available()) {  
        int ans = (int)Serial.read();  
        if (ans == que[i]) {  
          an = an + 1;  
        }  
        i++;  
      }  
    }  
    delay(1000);  
    if (an == ra) {  
      digitalWrite(9, 1);  
      delay(500);  
      digitalWrite(9, 0);  
    }  
    else {  
      digitalWrite(7, 1);  
      delay(500);  
      digitalWrite(7, 0);  
    }  
    an = 0;  
  }  
}
```


응용 프로젝트 2

프로그래밍 연습 - 게임 만들기 2 : 이번 주에 배운 내용 최대한 활용하는 과제 설정



응용 프로젝트 2

프로그래밍 연습 - 게임 만들기 2 : 이번 주에 배운 내용 최대한 활용하는 과제 설정

```
// 2P 반응속도 게임 [[[ Cup Game For 2P ]]]
// 빨강이 1P, 초록이 2P
//
// 게임 설명:
// 부저가 울릴 때 상대방보다 먼저 종이컵을 뒤집으면 승리. 패배할 수록 당신의 LED 불빛은 꺼져간다.
//
// 주의사항:
// 새로운 환경에서 연결 시, 조도 기준 새로 세팅할 것.
// --> void cupState() <--

#define RND_MX 8
#define RND_MN 5
#define LIFE 5

const int p1 = 5; const int pCup1 = A0; int lifel = LIFE - 1; // player 1 data
const int p2 = 6; const int pCup2 = A1; int life2 = LIFE - 1; // player 2 data
int rd = 0, ar = 0; // rd is for random(), ar is for analogRead()
// parameter
byte state = 0;
byte gamemode = 0;

int level[LIFE] = {0};

void setup() {
    Serial.begin(9600);
    pinMode(8, OUTPUT); // Piezo Buzzer
    pinMode(p1, OUTPUT);
    pinMode(p2, OUTPUT);
    randomSeed( analogRead(A5) ); // A5 OUTPPUT: random integer [0,1023]
    for (int i = 0; i < LIFE; i++) level[i] = map(i, 0, LIFE - 1, 0, 255);
}
```

```
void loop() {
    blinkTwice();
    if (!gamemode) {
        printStatus();
        randSignal();
    }
    else {
        /* Game over. Need to Reset */
        gamemode = 0;
        lifel = LIFE - 1;
        life2 = LIFE - 1;
        blinkTwice();
    }
}

void printStatus() {
    Serial.println("Player 1 HP : " + (String) (lifel + 1) );
    Serial.println("Player 2 HP : " + (String) (life2 + 1) + "\n");
}

void blinkTwice() {
    analogWrite(p1, 0);
    analogWrite(p2, 0);
    delay(400);
    analogWrite(p1, level[lifel]);
    analogWrite(p2, level[life2]);
    delay(400);
    analogWrite(p1, 0);
    analogWrite(p2, 0);
    delay(400);
    analogWrite(p1, level[lifel]);
    analogWrite(p2, level[life2]);
}
```

초기설정된 LIFE 개수에
따라 LED 밝기 단계를
미리 나누어 둬

응용 프로젝트 2

프로그래밍 연습 - 게임 만들기 2 : 이번 주에 배운 내용 최대한 활용하는 과제 설정

```
void randSignal() {
    rd = random( RND_MN , RND_MX + 1 );
    delay(rd * 750);
    /* Random Beep Signal */
    tone(8, 500);
    /* Cheat Test */
    cupState();
    if (state) {
        cheat();
        return; // Restart the Round.
    }
    for (int i = 0; i < 300; i++) {
        delay(5);
        cupState();
    }
    noTone(8);
    Serial.println("Player " + (String) state + " WON !!!");

    /* 게임 점수 처리 */
    switch (state) {
        case 1: damage2(); state = 0; break;
        case 2: damage1(); state = 0; break;
    }
}

void cupState() {
    ar = analogRead(A0);
    if (ar < 350 && !state) state = 1;
    ar = analogRead(A1);
    if (ar < 350 && !state) state = 2;
}

void cheat() {
    Serial.println("Player " + (String)state + " has cheated. Restart The Round. \n");
    noTone(8);
    if (state == 1) damage1();
    else damage2();
    state = 0;
}

void damage1() {
    if (lifel > 0) lifel --;
    else gamemode = 1;
    analogWrite(pl, level[lifel]);
}

void damage2() {
    if (life2 > 0) life2 --;
    else gamemode = 1;
    analogWrite(p2, level[life2]);
}
```

부저 소리가 울리는 주기는 무작위

어려웠던 점

```
int i=0;

void setup() {
  Serial.begin(9600);
  pinMode(8, OUTPUT); // Piezo Buzzer
}

void loop() {
  delay(2000);

  tone(8, 500, 1500);
  Serial.println("case 1");

  delay(2000);

  for(i=0; i<100; i++){
    tone(8, 500, 15);
    Serial.println(i);
  }
  Serial.println("case 2");
}
```

tone() 함수를 for문 안에 넣으면
왜 정상 작동하지 않을까?

여러 값을 넣으며 관찰해본 결과
for문을 끝까지 돌리지 않고
단 한 번만 실행하고 넘어가는 듯하다.



tone 울리는 시간

연산작업 하는 시간

▲ 기대했던 작동결과

향후 학습 계획

응용 프로젝트로 게임을 만들면서
아두이노 간의 분업 및 통신의 필요성을 느꼈음

아두이노의 통신

- 아두이노 기기 간의 직접적인 통신 연결
- 블루투스 연결
- LAN 연결 ? (추가 부품이 필요한 것 같아 확정은 아님)