

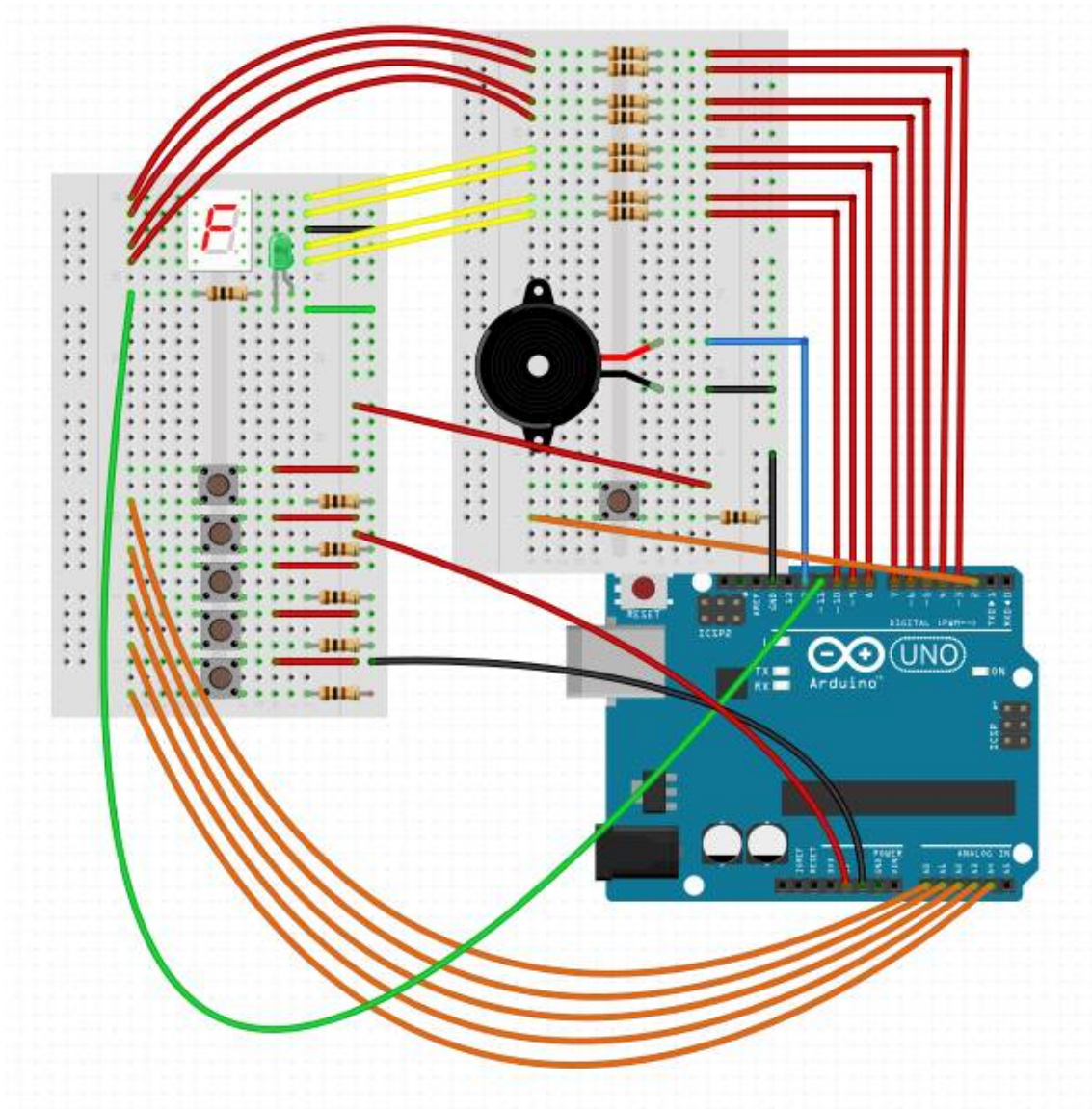
# 센서활용 프로그래밍 보고서

주제 : 엘리베이터
팀원 : 20192613 김규열, 20191776 노지원
결과 보고서
<b>1. 작품 개요</b> 실생활에서 볼 수 있는 전자기기 중 상당히 흥미로운 고기능 기기인 엘리베이터를 생각해 내어 센서활용 프로그래밍 팀 프로젝트의 주제로 채택하였습니다.
<b>2. 작품 설명</b> <ul style="list-style-type: none"> <li>• 엘리베이터 기능 <ul style="list-style-type: none"> <li>- 승강 기능 (세그먼트 숫자로 올라가는 엘리베이터 카의 현재 위치를 실시간으로 표시해 줍니다.)</li> <li>- 선택 층 도착 시 자동 개폐 (개폐 표현은 LED켜짐 꺼짐으로 표현하였고 원하는 층에 도착 후 analogWrite로 서서히 변하는 밝기로 자동 개폐과정을 표현했습니다.)</li> <li>- 엘리베이터 카 정지 상태에서의 수동 개폐 (층 입력이 없는 멈춘 상태에서의 수동 열림, 닫힘을 LED 밝기 변화를 통해 구현했습니다.)</li> <li>- 정지상태에서 수동 열림 사용시, 수동 닫힘, 일정 시간 이후 강제 닫힘과 구현 (delay가 아닌 millis 함수를 사용하여 일정 시간(2초)동안 닫힘 버튼을 누르거나, 일정 시간이(2초) 지나면 LED가 꺼집니다.)</li> <li>- 현재 위치와 같은 층 선택 시 세그먼트 점멸 (같은 층이라는 의미로 세그먼트를 전체 점멸하고 LED 밝기 변화로 자동 개폐를 구현했습니다.)</li> <li>- 문이 일정 시간 초과하여 열려 있을 경우 경고음 발생과 함께 강제 닫힘 (도착 후 자동 열림 기능 이후 수동 열림 버튼을 일정 시간 누를 시 수동 BUZZER를 통해 경고음이 발생하고 강제로 LED가 꺼집니다(문 닫힘을 의미).)</li> <li>- 비상 버튼을 누르면 강제 1층 (attachInterrupt를 사용해서 외부 인터럽트를 구현해 모든 행동을 중단하고 층을 강제로 1씩 줄어들게 해서 1층에 도착하도록 만들어 주고, 이후 모든 입력을 받지 않습니다.)</li> </ul> </li> </ul>



<탭으로 기능을 분리별로 분리>

4



5

6

7

<배선>

**//arduino\_Elevator**

```

    unsigned int curmil, premil;
    unsigned int cmil, pmil;
    int onebtn=0;
    int twobtn=0;
    int thrbtn=0;
    int opbtn=0;
    int clbtn=0;
    int now=1;//현재승강기 위치
    int pressd=0;//층버튼눌림

    boolean Emrused=0;
    boolean *emr=&Emrused;
    int i=0;
    int *lux=&i;

int digits[3][7]=
{

{0,0,0,0,0,1,1},//1
{1,0,1,1,1,1,0},//2
{1,0,0,1,1,1,1},//3

};
void setup() {
digitalWrite(12,0);
for(int pin = 3;pin<13;pin++){
    pinMode(pin,OUTPUT);
}
Serial.begin(9600);
attachInterrupt(0,Emr,RISING);
}

void loop() {
    Serial.println("loop실행");
    thrbtn=digitalRead(A0);
    twobtn=digitalRead(A1);
    onebtn=digitalRead(A2);
    opbtn=digitalRead(A3);
    clbtn=digitalRead(A4);

```

```
analogWrite(11,*lux);
for(int pin =3; pin<10; pin++){
    digitalWrite(pin,digits[now-1][pin-3]);
}

if(!Emrused){
if((onebtn||twobtn)||thrbtn){
floorFc();
openFc();
}

if(opbtn==1){
    cmpOpen();
    cmil=millis();
    pmil=cmil;
    while(!((cmil-pmil)>2000)){
        cmil=millis();
    }
    if(digitalRead(A4)){
        Serial.println("수동열림완전열림후수동닫힘실행");
        cmpClose();
        Serial.println(*lux);

        break;
    }
}
}

closeFc();

}
}
}
```

```
//openF
```

```
void openFc(){
```

```
Serial.println("open실행");
```

```
pressd=0;
```

```
for(;*lux<255;*lux=*lux+5){ //점점 열림
```

```
Serial.println(*lux); //열림(빛)출력
```

```
analogWrite(11,i);
```

```
delay(50);
```

```
}
```

```
delay(2000);
```

```
if(!Emrused)
```

```
closeFc();
```

```
Serial.println("open종료");
```

```
}
```

```
void closeFc(){
```

```
Serial.println("close실행");
```

```
unsigned long curmil=millis();
```

```
unsigned long premil=curmil;
```

```
for(;*lux>0;*lux=*lux-5){ //점점 닫힘
```

```
analogWrite(11,*lux);
```

```
delay(30);
```

```
curmil=millis();
```

```
if((curmil-premil)<=5000){ //닫히지않고 열려있는시간이 5초이상
```

```
if(digitalRead(A3)){ //열림버튼을 누를 경우
```

```
cmpOpen();
```

```
if((curmil-premil)>=2500){
```

```
Serial.println("강제닫힘실행");
```

```
tone(12,440,500);
```

```
delay(500);
```

```
}
```

```
}
```

```
}
```

```
}
```

```
Serial.println("close종료");
```

```
}
```

```
//floorF
```

```
void floorFc(){
  Serial.println("floor실행");
  if(onebtn==1){
    pressd=1;
  }
  if(twobtn==1){
    pressd=2;
  }
  if(thrbtn==1){
    pressd=3;
  }

  if(now==pressd){//깜빡깜빡
    for(int now=0; now<3; now++){
      for(int pin =3; pin<10; pin++){
        digitalWrite(pin,0);
      }
      delay(100);
      for(int pin =3; pin<10; pin++){
        digitalWrite(pin,digits[pressd-1][pin-3]);
      }
      delay(100);
    }

    }else if(now>pressd){
    for(now;now>=pressd;now--){
      for(int pin =3; pin<10; pin++){
        digitalWrite(pin,digits[now-1][pin-3]);
      }
      delay(1000);
      if(now==pressd)
        break;
    }//내려감
    tone(12,880,300);
    delay(300);
tone(12,880,300);
delay(300);
```

```
tone(12,784,300);
delay(300);
tone(12,740,300);
delay(300);

}else if(now<pressd){

    for(now:now<=pressd;now++){
        for(int pin =3; pin<10; pin++){
            digitalWrite( pin,digits[now-1][ pin-3]);
            //세그먼트 총표시
        }
        delay(1000);
        if(now==pressd)
            break;
        }//올라감
        tone(12,880,300);
delay(300);
tone(12,880,300);
delay(300);

tone(12,784,300);
delay(300);
tone(12,740,300);
delay(300);

    }

}
```



```
//cmpOpenF
void cmpOpen(){
  Serial.println("cmpOpen실행");
  for(;*lux<255;*lux=*lux+5){ // *lux값대로 점점 열림
    analogWrite(11,*lux);
    delay(30);
    if(digitalRead(A4)){
      cmpClose();
      Serial.println(*lux);

      break;
    }
  }

  Serial.println("cmpOpen종료");
}

void cmpClose(){
  Serial.println("cmpClose실행");
  delay(1000);
  for(*lux>0;*lux=*lux-5){ // i값에서 점점 닫힘
    analogWrite(11,*lux);
    delay(30);
  }
  Serial.println("cmpClose종료");
}
```

**//Emergency**

```
void Emr(){
  Serial.println("Emr실행");
  tone(12,1020,20);

  delayMicroseconds(20000);

  tone(12,440,20);

  delayMicroseconds(20000);

  for(now;now>=1;now--){

    delayMicroseconds(10000);

    for(int pin =3; pin<10; pin++){
      digitalWrite(pin,digits[now-1][pin-3]);
    }

  }

  now=1;
  *lux=255;
  *emr=true;
  Serial.println("Emr종료");
}
```

### 3. 핵심기술 (특징 및 장점)

- 기능별로 코드를 탭으로 나누어 수정을 쉽게 할 수 있도록 했습니다.
- 반복문 속에 millis함수를 사용하여 시간 지연에 관련된 기능을 구현했습니다.
- attachInterrupt를 통해 비상과 관련된 기능을 구현했습니다.
- 추가적인 공부를 통해 엘리베이터 밝기 변수를 포인터 변수로 설정하여 함수들과 간의 상호작용이 가능하도록 구현하였습니다.
- 정갈한 배치를 통해 핀을 다 이용했음에도 불구하고 합선의 위험을 최소화하였습니다.

### 4. 느낀 점, 아쉬웠던 점

- 평소 수업을 통해 이론적인 학습과 실습을 했지만, 시작부터 주체적으로 이끌어나가는 프로젝트를 통해 시험보다 더 많은 지식을 얻게 되었다고 생각합니다.
- 아두이노 보드의 포트 수와 같은 물리적 한계에 부딪히게 돼서 더 다양한 기능을 사용하지 못해 아쉬웠습니다.
- 초반 기획과 실제 설계에서 예상하지 못한 상황들을 해결해나가면서 팀원과 의견 조율을 하는 방법을 배울 수 있었습니다.
- 배웠던 모든 센서와 모듈을 한 번씩 사용해 보고 싶었지만, 핀과 보드 공간이 부족하여 사용하지 못하였는데, 이후 학기에 기회가 된다면 더 많은 기능들을 사용해 보고 싶습니다.  
(ex 조도센서를 이용한 손끼임 방지, 가변저항기를 이용한 수동 개폐 장치 등)