

〈 창의 SW 기초 설계 〉

최종 보고서

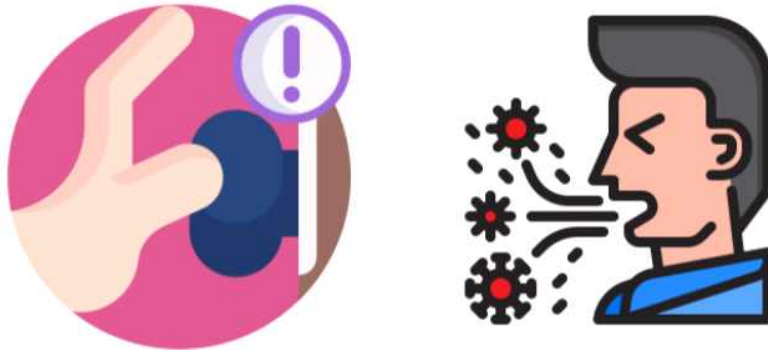
: Touch Free Door Lock

분반	스마트 001
지도 교수	송진우 교수님
조	4조
학과	지능기전공학부 스마트기기공학전공
학번	18012576
이름	장윤정

- 목차 -

1. 설계 배경 및 기획 의도	p.3
2. 제작물 소개	p.4
1) Touch Free Door Lock 이란?	p.4
2) 필요 물품	p.4
3. 구현 상세 내용	p.5
1) 기능 순서도	p.5
2) 하드웨어 설계	p.5~8
3) 소프트웨어 설계	p.8~11
4. 시연	p.12~13
5. 다양한 시도 및 차별성	p.13~14
6. 한계점 및 극복 방안	p.14
7. 기대효과 및 발전 가능성	p.14~15
8. 요약	p.15

1. 설계 배경 및 기획 의도



문 손잡이에 '바이러스 씨앗'... "최대 5일간 생존"

서울시 "집단감염 사무실 문손잡이·에어컨서 코로나19 바이러스 검출"

사무실 문 손잡이에도 코로나... '재택' 이후가 문제다

공공건물 문 손잡이 '코로나19' 불안하다

① COVID-19

: 코로나 바이러스가 전파되기 시작한 이후로 사람들은 위생 관리와 감염 경로 차단에 많은 신경을 쓰고 있습니다. 공공장소에서 누가 만졌을지도 모르는 손잡이를 잡기 싫다는 생각, 누구나 한 번쯤은 해 보았을 것입니다. 심지어 손잡이에 접촉한 후 손을 씻을 화장실을 찾지 못하거나 손 소독제마저 없다면 매우 찝찝한 기분이 들게 됩니다. 이와 같은 이유로 불과 1년이라는 시간 동안 ‘비대면 비접촉’, ‘언택트 시스템’의 수요는 폭발적으로 증가하였습니다. 저는 이러한 시대 상황과 수요를 고려한 작품을 만들고 싶다는 생각을 하였고, 저희 4조는 “Touch Free Door Lock”이라는 작품을 설계하게 되었습니다.

② 스마트 광진구

: 이번 기초설계의 대주제는 ‘스마트 광진구’입니다. “Touch Free Door Lock”은 강의실, 사무실, 화장실 등 문과 잠금장치가 필요한 공간이라면 어디든지 적용 가능합니다. 광진구에 존재하는 건물들에 저희의 시스템을 적용한다면 ‘스마트한 방식으로 방역 강화에 기여하는 광진구’라는 점에서 ‘스마트 광진구’라는 대주제와도 적합하다고 생각합니다.

2. 제작물 소개

1) Touch Free Door Lock 이란?



“Touch Free Door Lock” 은 문을 자동으로 여닫는 기능과 잠금장치 작동 기능을 비접촉 방식으로 구현한 시스템입니다. 기능 목록은 아래와 같습니다.

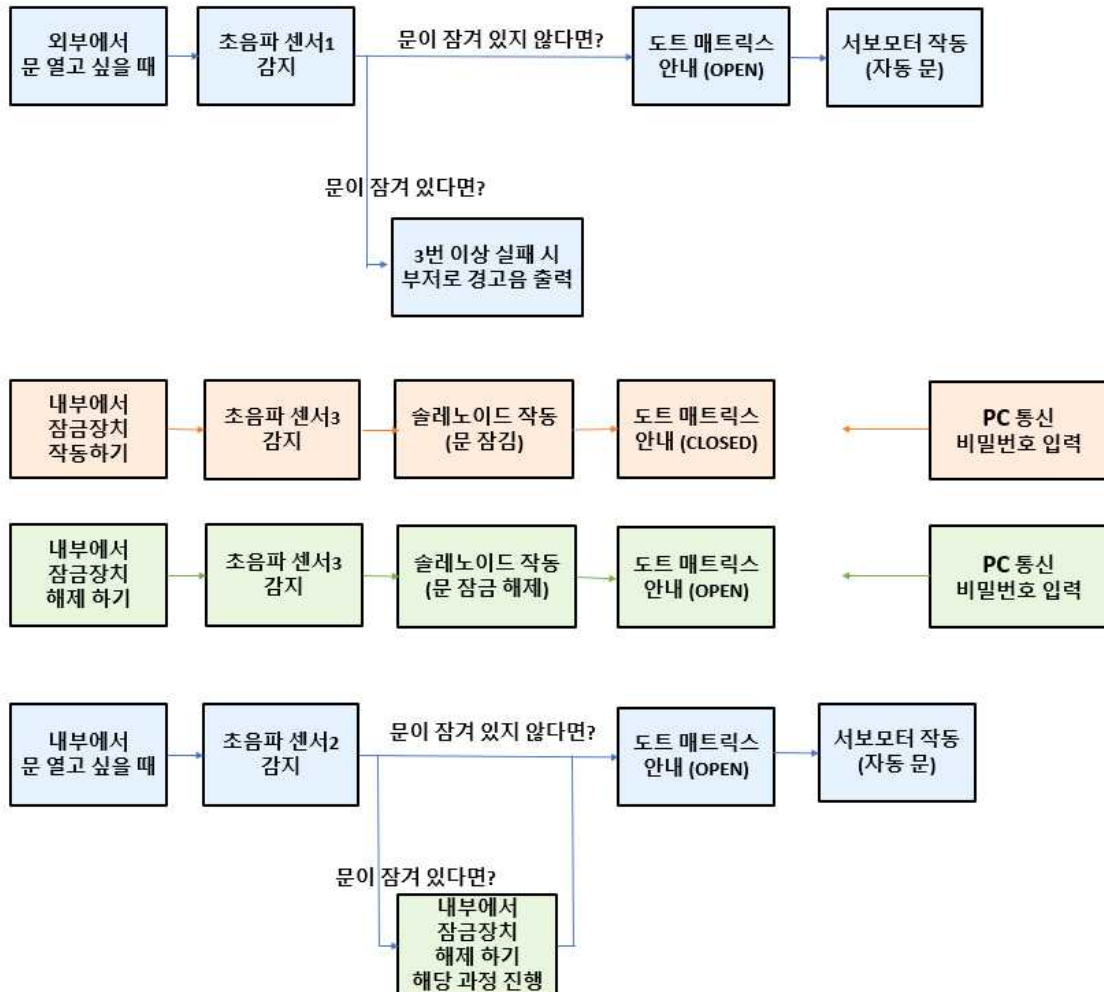
- ① 비접촉 센서 감지 방식 문 자동 개폐
- ② 비접촉 센서 감지 방식 잠금장치 작동
- ③ 잠금장치는 PC 통신을 통해서도 제어 가능
- ④ 문이 잠겨있을 때, 3회 이상 문을 열고자 시도한다면 경고음 출력
- ⑤ 경고음이 출력된 시간은 PC 통신을 통해 기록
- ⑥ 도트 매트릭스를 이용하여 문 잠금 상태 실시간 파악 (OPEN, CLOSED)

2) 필요 물품

No.	물품명	수량	용도
1	아두이노 우노, USB 케이블	1	마이크로컨트롤러, 개발 환경 제공
2	MDF 얇은 합판	-	외관 제작
3	초음파 센서	3	사람 접근 감지
4	서보 모터	1	자동문 개폐
5	12V 솔레노이드	1	잠금장치
6	1채널 5V 릴레이 모듈	1	솔레노이드와 아두이노 연결
7	아두이노 호환 전원 어댑터	1	솔레노이드 전원 공급
8	8x32 도트 매트릭스	1	OPEN, CLOSED 상태 출력
9	수동 부저 모듈	1	경고음 출력

3. 구현 상세 내용

1) 기능 순서도



2) 하드웨어 설계

① 초음파 센서 1

- 설명 : 초음파를 이용하여 거리를 측정하는 센서

- 동작 원리 :

사람의 귀에 들리지 않을 정도의 높은 주파수 (약 20kHz 이상)의 소리인 초음파가 가지는 특성을 이용한 센서이다. 트리거(Trig)핀을 통해 10μs 길이의 펄스를 입력하면 센서에서 전방을 향해 초음파를 발사하고, 물체에 부딪혀 반사되는 초음파가 돌아오는 시간의 길이를 계산하여 에코(ECHO)핀에서 펄스를 출력한다. 에코핀에서 출력되는 하이 신호 폭을 측정하여 초음파 센서 모듈과 물체와의 거리를 알아낼 수 있다. 감지 거리는 최소 2cm에서 최대 400cm까지의 측정범위를 가지고 있다.

- 용도 : 문 외부에서 사용자 접근을 감지

- 핀 맵 :

아두이노 우노	초음파 센서
디지털 4 핀	Trig 핀
디지털 5핀	Echo 핀
5V 핀	VCC 핀
GND 핀	GND 핀

② 초음파 센서 2

- 설명 : 초음파를 이용하여 거리를 측정하는 센서
- 동작 원리 : ‘초음파 센서 1’의 동작 원리와 동일
- 용도 : 문 내부에서 사용자 접근을 감지
- 핀 맵 :

아두이노 우노	초음파 센서
디지털 4 핀	Trig 핀
디지털 5핀	Echo 핀
5V 핀	VCC 핀
GND 핀	GND 핀

③ 초음파 센서 3

- 설명 : 초음파를 이용하여 거리를 측정하는 센서
- 동작 원리 : ‘초음파 센서 1’의 동작 원리와 동일
- 용도 : 문 내부에서 사용자 접근을 감지
- 핀 맵 :

아두이노 우노	초음파 센서
디지털 4 핀	Trig 핀
디지털 5핀	Echo 핀
5V 핀	VCC 핀
GND 핀	GND 핀

④ 서보 모터

- 설명 : 각도 및 각속도의 제어가 가능한 모터
- 동작 원리 : PWM(펄스 폭 신호 제어)에 따라 회전의 각도($0^{\circ} \sim 180^{\circ}$)가 정해진다.
- 용도 : 문 자동 개방
- 핀 맵 :

아두이노 우노	서보 모터
디지털 3 핀	오렌지색 선
5V 핀	적색 선
GND 핀	흑색 선

⑤ 12V 솔레노이드

- 설명 : 도어락 잠금장치에 사용되는 솔레노이드

- **동작 원리** : 솔레노이드는 구리 와이어의 커다란 코일로 만들어져 있습니다. 전류 공급 여부에 따라 솔레노이드가 작동하게 됩니다. 솔레노이드는 12V의 전원을 필요로 하지만 아두이노만으로는 이를 제어하기 힘들기 때문에 릴레이 모듈과 외부전원을 사용하게 됩니다. 릴레이 모듈은 자력으로 접점을 끌어당기거나 놓는 원리를 이용한 전력용 스위칭 모듈입니다. 릴레이는 코일에 전류를 흘리면 자석이 되는 성질을 이용하여 전자석이 되었을 때 철판을 끌어당겨 그 철판에 붙어 있는 스위치의 접점을 열거나 닫습니다. 릴레이는 전기적으로 독립된 회로를 연동시킬 수 있다는 장점이 있으며 5V와 같은 저 전압계로 구성된 회로의 동작 때문에 DC 대 전류의 회로를 ON/OFF 시킬 수 있습니다. 이는 릴레이의 코일 부분과 접점 부분이 완전하게 절연되어 가능합니다. 릴레이 모듈에서의 NC는 Normal Close의 줄임말로 전원을 입력하지 않을 때 NC와 COM이 연결되어 있다가 전원을 입력할 때 NC와 COM이 떨어집니다. 이와 반대로 NO는 Normal Open의 줄임말로 전원이 입력되지 않을 때 접점이 떨어져 있다가 전원이 입력될 때 NO와 COM이 연결됩니다.

- **용도** : 잠금장치

- **핀 맵** :

아두이노 우노	릴레이 모듈	릴레이 모듈	12V 솔레노이드	전원 어댑터
디지털 13핀	SIG 핀	NC 핀		적색 선
5V 핀	VCC 핀	COM 핀	적색 선	
GND 핀	GND 핀	NO 핀		
			흑색 선	흑색 선

⑥ 도트 매트릭스

- **설명** : LED를 매트릭스 형태로 배열하여 다양한 숫자나 문자 등을 표현할 수 있도록 한 디스플레이 장치, 8X32 도트 매트릭스는 8X8 도트 매트릭스를 4개 이어 붙인 모듈

- **동작 원리** : 모양을 만들 때 한 번에 출력하는 것이 아닌, 각 라인이 빠르게 돌아가며 데이터를 입력합니다. 이 때 사람의 눈에서는 잔상 효과로 인해 도트 매트릭스의 모든 줄이 빛나는 것처럼 보이게 됩니다. 이런 제어법을 다이내믹 구동 방법이라고 합니다.

- **용도** : 잠금 상태 (OPEN, CLOSED) 실시간 출력

- **핀 맵** :

아두이노 우노	8X32 도트 매트릭스
디지털 10 핀	DIN 핀
디지털 11 핀	CS 핀
디지털 12 핀	CLK 핀
5V 핀	VCC 핀
GND 핀	GND 핀

⑦ 수동 부저

- **설명** : 신호를 입력하면 소리 출력이 가능한 센서

- **동작 원리** : 수동 부저는 부저에 구동 회로가 없어 아두이노 스케치에서 제공하는 라이브러리인 톤(Tone) 함수를 이용하여 음을 출력할 수 있다. 전원만 인가되면 작동하는 능

동부저와 달리 프로그래밍에 의한 신호 제어가 필요하다. 동작 전압은 5V이며, 입력 신호는 PWM(펄스 폭 신호 제어) 이다.

- 용도 : 잠겨있는 상태에서 외부에서 3번 이상 접근 시도 시 경고음 출력

- 핀 맵 :

아두이노 우노	수동 부저
디지털 2 핀	S핀
5V 핀	+핀
GND 핀	-핀

3) 소프트웨어 설계


- 프로그램 : 아두이노 IDE (Sketch)

- 사용하는 센서와 부품의 작동 원리, 동작 전압 등을 파악하고 프로그래밍 해야 합니다.

- 이해를 돕기 위해 중요한 흐름을 의사 코드 형식으로 작성해 보았습니다.

```
void loop() {
    if (초음파 센서 1 감지 || 초음파 센서 2 감지) {
        if (솔레노이드 잠금장치 ON) {
            도트 매트릭스에 "CLOSED" 출력;
            이 상태로 몇 번 접근하는지 COUNT 해서 3번 이상일 때 부저음 출력;
        }
        else if (솔레노이드 잠금장치 OFF) {
            도트 매트릭스에 "OPEN" 출력;
            서보 모터 작동;
        }
    }
    else 도트 매트릭스에 "CLOSED" 출력;

    if (초음파 센서 3 감지) {
        짝수 횟수 감지 시 솔레노이드 잠금장치 ON;
        홀수 횟수 감지 시 솔레노이드 잠금장치 OFF;
    }
}
```



- 상세한 코드는 아래에 첨부합니다.

```
#include <Servo.h>
#include <LedControl.h>
#include <swRTC.h>

#define BUZZER 2 //부저
#define PIN_SERVO 3 //서보모터
#define SONAR1_TRIG 4 //초음파센서1(내부에서 문 여는 용도)
#define SONAR1_ECHO 5
#define SONAR2_TRIG 6 //초음파센서2(외부에서 문 여는 용도)
#define SONAR2_ECHO 7
#define SONAR3_TRIG 8 //초음파센서3(내부에서 잠금장치)
```



```

#define SONAR3_ECHO      9
#define DOT_DIN          10 //도트 매트릭스 DIN핀
#define DOT_CS           11 //도트 매트릭스 CS핀
#define DOT_CLK          12 //도트 매트릭스 CLK핀
#define SOLENOID         13 //솔레노이드
Servo servo; // 문을 조작하는 서보모터를 동작시키기위해서 작성합니다. ( 객체 생성 )
int angle = 0;
int sol = 0;
int count = 0;
LedControl dot_matrix=LedControl(DOT_DIN,DOT_CLK,DOT_CS,4); // (DIN, CLK, CS, 연결할 도트 매트릭스의 개수)
swRTC rtc;

//도트 매트릭스 출력할 배열 정의
int n_matrix = 4;
byte OPEN_1[8] = {B00000000,B00111100,B01000010,B01000010,B01000010,B00111100,B00000000};
byte OPEN_2[8]= {B00000000,B00000010,B00000010,B00111110,B01000010,B01000010,B00111110,B00000000};
byte OPEN_3[8]= {B00000000,B01111110,B00000010,B01111110,B01111110,B00000010,B01111110,B00000000};
byte OPEN_4[8]= {B00000000,B01000010,B01100010,B01010010,B01001010,B01000110,B01000010,B00000000};
byte CLOSED_1[8] = {B00000000,B11011100,B01000010,B01000001,B01000001,B01000010,B01011100,B00000000};
byte CLOSED_2[8] = {B00000000,B00110111,B01001000,B10000100,B10000100,B01001000,B00110000,B00000000};
byte CLOSED_3[8] = {B00000000,B11011111,B01010001,B11011100,B11000011,B01010001,B11011111,B00000000};
byte CLOSED_4[8] = {B00000000,B01111011,B10001000,B10001011,B10001011,B10001000,B01111011,B00000000};

void setup() {
    Serial.begin(9600);
    Serial.print("Password : "); //시작 비밀번호 입력
    digitalWrite(SOLENOID, HIGH);
    pinMode(BUZZER, OUTPUT);
    servo.attach(PIN_SERVO);
    pinMode(SONAR1_TRIG, OUTPUT);
    pinMode(SONAR1_ECHO, INPUT);
    pinMode(SONAR2_TRIG, OUTPUT);
    pinMode(SONAR2_ECHO, INPUT);
    pinMode(SONAR3_TRIG, OUTPUT);
    pinMode(SONAR3_ECHO, INPUT);
    pinMode(SOLENOID, OUTPUT);
    rtc.stopRTC(); //정지
    rtc.setTime(9, 00, 00); //시, 분, 초
    rtc.setDate(21, 6, 2021); //일, 월, 년
    rtc.startRTC(); //시작
    for(int ii=0; ii<n_matrix; ii++) { // 매트릭스 0번부터 3번까지 세팅
        dot_matrix.shutdown(ii,false); // 0~3번까지 매트릭스 절전모드 해제
        dot_matrix.setIntensity(ii,8); // 매트릭스의 밝기 0~15 사이의 정수
        dot_matrix.clearDisplay(ii); // 매트릭스를 초기화
    }
}

void loop() {
    int a = Serial.parseInt();
    if(a == 1234){
        Serial.println("OPEN!");
        digitalWrite(SOLENOID, LOW); //비밀번호 1234 입력하면 잠금장치 OFF
        turn_on_led_open(OPEN_1, OPEN_2, OPEN_3, OPEN_4, dot_matrix); // OPEN 출력
    }
    if(a == 4321){
        Serial.println("CLOSED!");
        digitalWrite(SOLENOID, HIGH); //비밀번호 4321 입력하면 잠금장치 ON
        turn_on_led_closed(CLOSED_1, CLOSED_2, CLOSED_3, CLOSED_4, dot_matrix); //CLOSED 출력
    }
}

```

```

}
long duration1, duration2, duration3;
//초음파 센서 1
digitalWrite(SONAR1_TRIG, LOW);
digitalWrite(SONAR1_ECHO, LOW);
delayMicroseconds(2);
digitalWrite(SONAR1_TRIG, HIGH);
delayMicroseconds(10);
digitalWrite(SONAR1_TRIG, LOW);
duration1 = pulseIn(SONAR1_ECHO, HIGH);
//초음파 센서 2
digitalWrite(SONAR2_TRIG, LOW);
digitalWrite(SONAR2_ECHO, LOW);
delayMicroseconds(2);
digitalWrite(SONAR2_TRIG, HIGH);
delayMicroseconds(10);
digitalWrite(SONAR2_TRIG, LOW);
duration2 = pulseIn(SONAR2_ECHO, HIGH);
//초음파 센서 3
digitalWrite(SONAR3_TRIG, LOW);
digitalWrite(SONAR3_ECHO, LOW);
delayMicroseconds(2);
digitalWrite(SONAR3_TRIG, HIGH);
delayMicroseconds(10);
digitalWrite(SONAR3_TRIG, LOW);
duration3 = pulseIn(SONAR3_ECHO, HIGH);

if(duration1 < 200 || duration2 < 200){ // 초음파센서 1, 2 감지

    if(sol % 2 != 0){ // 솔레노이드 잠겨있으면
        turn_on_led_closed(CLOSED_1, CLOSED_2, CLOSED_3, CLOSED_4, dot_matrix); //CLOSED 출력
        count++; //잠겨있는 상태로 접근 시도하는 횟수 세고
        Serial.println(count);

        if (count % 3 == 0){ // 세번 시도하면 부저 출력하는 부분

            Serial.print("!!!WARNING!!!"); //시리얼 모니터의 경고 출력
            Serial.print(rtc.getYear(), DEC);
            Serial.print("/");
            Serial.print(rtc.getMonth(), DEC);
            Serial.print("/");
            Serial.print(rtc.getDay(), DEC);
            Serial.print(" ");
            Serial.print(rtc.getHours(), DEC);
            Serial.print("h ");
            Serial.print(rtc.getMinutes(), DEC);
            Serial.print("m ");
            Serial.print(rtc.getSeconds(), DEC);
            Serial.print("s ");
            Serial.println("!!!WARNING!!!");

            delay(100);
            tone(BUZZER, 523); //경고음 출력
            delay(100);
            noTone(BUZZER);
        }
        delay(500);
    }
}

```

```

else{ // 솔레노이드 열려있으면
    turn_on_led_open(OPEN_1, OPEN_2, OPEN_3, OPEN_4, dot_matrix); // OPEN 출력
    delay(100);
    for(angle = 50; angle >=0; angle--){ //서보모터 작동해서 문 열고
        servo.write(angle);
        delay(100);
    }
    for(angle = 0; angle <= 50; angle++){ //문 닫음
        servo.write(angle);
        delay(100);
    }
}

else{ // 초음파 센서 1, 2에 감지가 안되면
    turn_on_led_closed(CLOSED_1, CLOSED_2, CLOSED_3, CLOSED_4, dot_matrix); // CLOSED 출력 되어 있음
    delay(500);
}

if(duration3 < 200){ //초음파 센서 3 감지하면
    sol++;
    if(sol % 2 != 0){
        digitalWrite(SOLENOID, HIGH); // 솔레노이드 ON
        Serial.println("잠금장치 ON");
        delay(500);
    }
    else {
        digitalWrite(SOLENOID, LOW); // 솔레노이드 OFF
        Serial.println("잠금장치 OFF");
        delay(500);
    }
}

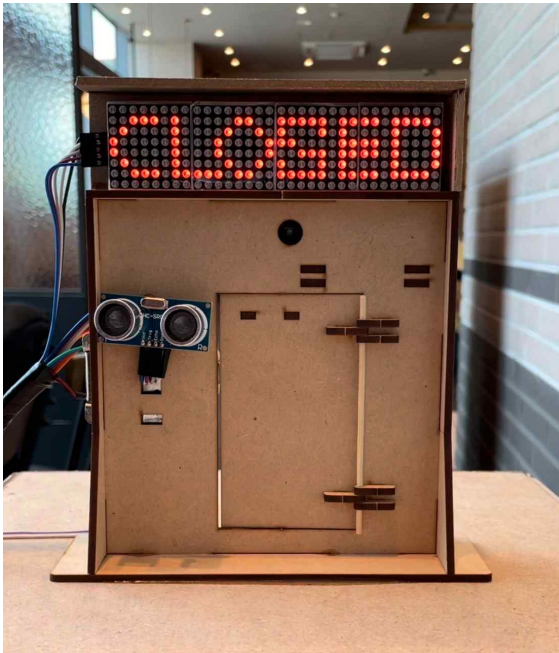
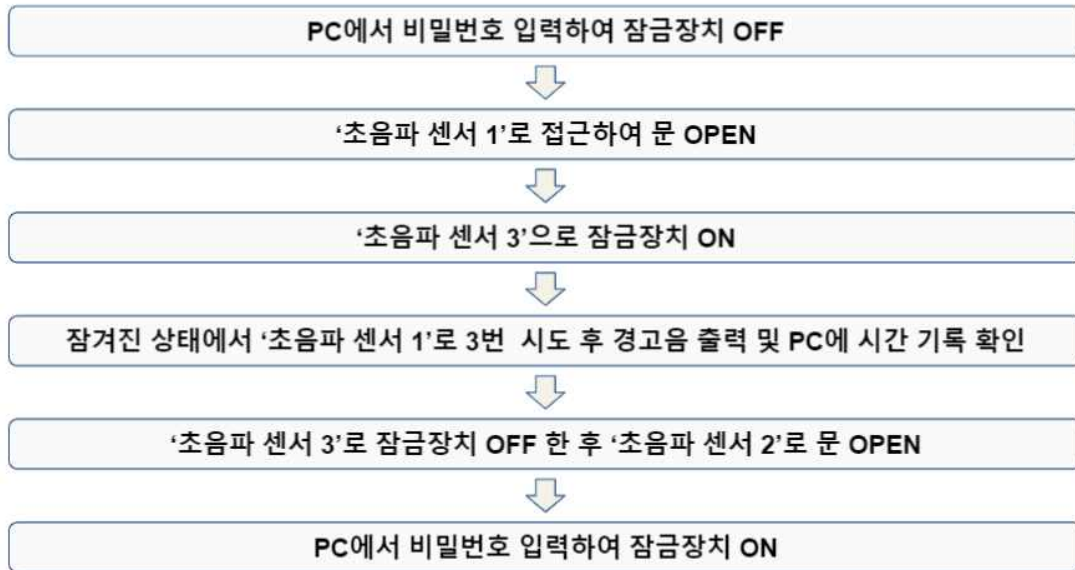
}

void turn_on_led_open(byte* CLOSED_1, byte* CLOSED_2, byte* CLOSED_3, byte* CLOSED_4, LedControl lc){
    // lc.setRow 명령어는 행(Row) 기준으로 매트릭스를 제어
    // lc.setRow(matrix_number,Row,value)
    for(int ii=0; ii<8; ii++) {
        lc.setRow(0,ii,OPEN_1[ii]); // 0번째 매트릭스에 mat1 출력
        lc.setRow(1,ii,OPEN_2[ii]); // 1번째 매트릭스에 mat2 출력
        lc.setRow(2,ii,OPEN_3[ii]); // 2번째 매트릭스에 mat3 출력
        lc.setRow(3,ii,OPEN_4[ii]); // 3번째 매트릭스에 mat4 출력
    }
}

void turn_on_led_closed(byte* CLOSED_1, byte* CLOSED_2, byte* CLOSED_3, byte* CLOSED_4, LedControl lc){
    // lc.setRow 명령어는 행(Row) 기준으로 매트릭스를 제어
    // lc.setRow(matrix_number,Row,value)
    for(int ii=0; ii<8; ii++) {
        lc.setRow(0,ii,CLOSED_1[ii]); // 0번째 매트릭스에 mat1 출력
        lc.setRow(1,ii,CLOSED_2[ii]); // 1번째 매트릭스에 mat2 출력
        lc.setRow(2,ii,CLOSED_3[ii]); // 2번째 매트릭스에 mat3 출력
        lc.setRow(3,ii,CLOSED_4[ii]); // 3번째 매트릭스에 mat4 출력
    }
}

```

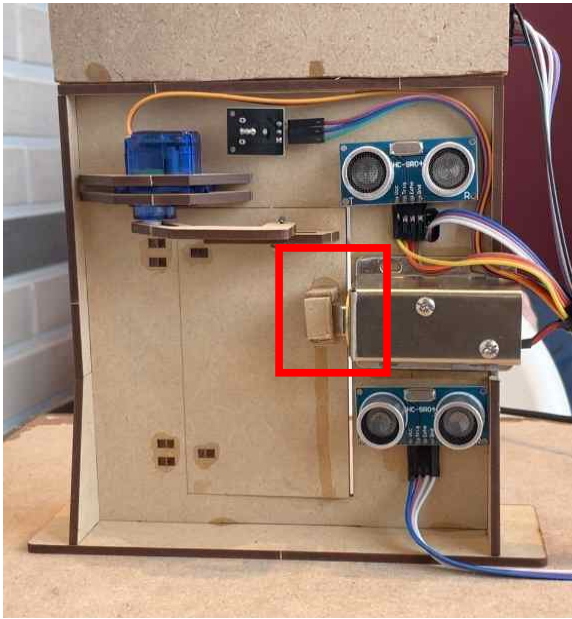
4. 시연



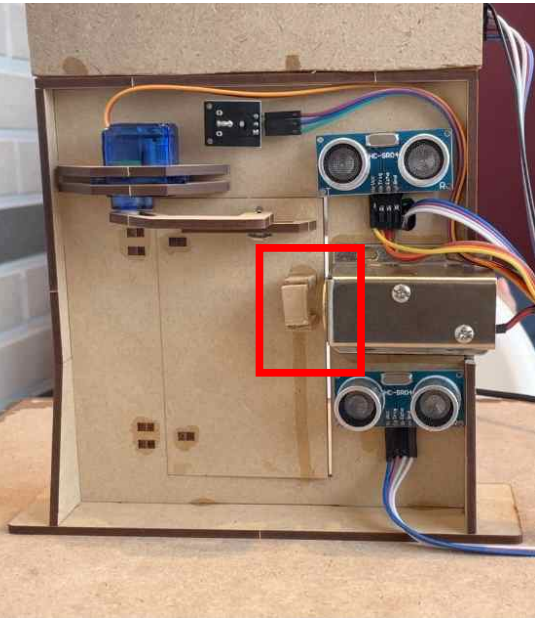
[외부 1] 문 잠겨있는 상태



[외부 2] 문 열고 있는 상태



[내부 1] 잠금장치 잠겨있는 상태



[내부 2] 잠금장치 해제된 상태

5. 다양한 시도 및 차별성

① 초음파 센서 사용

: 기존 아두이노 자동문들이 적외선 센서를 주로 사용하는 것과 다르게, 저희 조는 초음파 센서를 사용하였습니다. 초음파 센서는 실내의 전기 불빛과 낮과 밤의 명암 변화에 대해서 일정한 특성을 나타내주기 때문에, 빛 변화가 큰 실내에서는 초음파 센서가 더 좋은 효과를 줄 것으로 기대하였습니다. 초음파 센서는 기본 단위가 시간(마이크로초)이므로 이를 거리에 대한 조건으로 변경하고 싶다면 cm로 변환하는 식을 추가 해주어야 합니다.

② ALL 비접촉 방식 & 솔레노이드 잠금장치

: 기존의 아두이노 도어락들은 키패드를 이용한 접촉식 시스템이 많지만, 저희 조는 잠금 장치의 작동 또한 비접촉으로 가능하게 구현하였습니다. 모든 기능에서 한 번도 직접적인 접촉이 이뤄지지 않습니다. 해당 잠금장치는 솔레노이드를 이용한 잠금장치를 사용하였습니다. 잠금장치 또한 서보모터로 구현해 볼지 고민하였으나 아두이노 프로젝트에서 도어락으로 많이 사용된다는 솔레노이드를 사용하게 되었습니다. 덕분에 솔레노이드의 원리와 릴레이 모듈 연결, 외부 전원 공급 등 새로운 공부를 많이 하게 되었습니다.

③ 8X32 도트 매트릭스 사용

: 처음에는 키트에 들어있는 단일 도트 매트릭스를 사용해볼까 고민하였으나 핀이 너무 많이 필요해지는 관계로 8X32 도트 매트릭스 모듈을 사용했습니다. 또 LCD 디스플레이

도 고민해보았으나 한 눈에 상태를 표현하기 위해서는 도트 매트릭스가 더 나을 것 같다는 결론을 내렸습니다.

④ PC 통신 (잠금장치 제어 및 보안 기능)

: PC 통신을 어떤 식으로 이용하면 좋을지 고민을 많이 하였습니다. 결과적으로 잠금장치 제어와 보안 기능에 사용하게 되었습니다. 이는 스마트 빌딩의 관리인이 PC를 제어한다는 가정하에 이루어집니다. 관리인이 출근하면 PC에 비밀번호 입력을 통해 문의 잠금장치를 해제할 수 있습니다. 역시나 퇴근을 할 때도 PC에 비밀번호를 입력하여 문을 잠그고 나갈 수 있습니다. 또 보안 기능을 구현하기 위해, 잠겨있는 상태에서 외부인이 출입을 3번 이상 시도할 경우 경고음을 울림과 동시에 그 기록과 시간을 PC에 저장합니다.

6. 한계점 및 극복 방안

① 자동문 기능

: 저희는 자동문 기능을 서보모터를 이용해 구현하였습니다. 한번 열리고 난 후 시간 간격을 두고 다시 닫히는 식으로 동작합니다. 하지만 실제로 사람이 문을 통과하는 경우는 다양한 케이스가 존재합니다. 여러 명이 동시에 들어가는 경우, 크고 무거운 짐을 옮기는 경우, 어린아이가 문틈에서 장난치는 경우 등 다양한 케이스를 고려하지 못한 것이 아쉽습니다. 미니 제작물을 넘어서 실제 문에 적용하려면 사람이 완전히 통과하기 전까지는 문이 닫히지 않는 방향으로 설계를 다시 해야 할 것 같습니다. 또한, 서보모터보다 더 강한 힘을 줄 수 있는 모터를 사용해야 합니다.

7. 기대효과 및 발전 가능성



① COVID-19 방역 강화

: 제작물의 가장 큰 기획 의도이자 기대효과는 COVID-19 방역 강화입니다. 문손잡이에도 장기간 생존하는 코로나 바이러스를 막기 위해 “Touch Free Door Lock”을 이용한다면, 바이러스 감염 경로를 차단할 수 있다는 점에서 큰 의미를 지니게 됩니다.

② 인공지능을 이용한 모션 감지, 얼굴 인식 사용

: 현재 사용하고 있는 초음파 센서보다 한 단계 더 발전하기 위해서는 인공지능을 이용할 수 있습니다. 모션 감지 기술을 이용해 암호 모션을 취했을 때만 접근을 허가하거나, 얼굴 인식 기술을 이용해 지정된 사람의 얼굴만 인식하도록 할 수도 있습니다. 기획 초반에는 구글의 티처블 머신을 이용해 이와 같은 기능을 구현해 보고자 하였으나, 기초설계 과목에서는 아두이노와 센서에 집중하는 것이 더 좋을 것 같아 초음파 센서를 사용하게 되었습니다.

③ IoT 스마트 빌딩 전용 스마트 도어락

: 궁극적으로는 기술을 더 발전시켜 스마트 빌딩 전용 스마트 도어락을 만들 수 있습니다. 이는 광진구의 건물들에 쓰이게 되며 ‘스마트 광진구’라는 대주제를 만족시킬 수 있습니다. 이 정도로 발전하기 위해서는 확실한 보안 문제 해결, 스마트폰 애플리케이션과의 연동 등으로 자유로운 제어가 가능해야 할 것입니다.

8. 요약

(Touch Free Door Lock 이란?)

- 비접촉 자동문, 잠금장치 시스템이며 PC와의 통신도 가능합니다.
- 기대효과는 감염 경로 차단으로 인해 COVID-19 방역에 도움이 되는 것과 스마트 빌딩에 적용 가능한 IoT 서비스 제공입니다.
- 사용한 물품과 센서는 아두이노 우노, MDF 합판, 8X32 도트매트릭스, 12V 솔레노이드, 1채널 릴레이 모듈, 외부전원 어댑터, 서보모터, 수동 부저가 있습니다.