

임베디드 시스템 및 실습

20160067 구소현
20161049 장주영

01 소개

- 주제
- 동기

02 작품 설명

- 전체 작품 도안
- 재료
- 연결도

03 소스코드

- line tracer flow chart
- sudo code
- source code

04 동작 설명

- 동작 동영상
- 동작 설명

05 결론

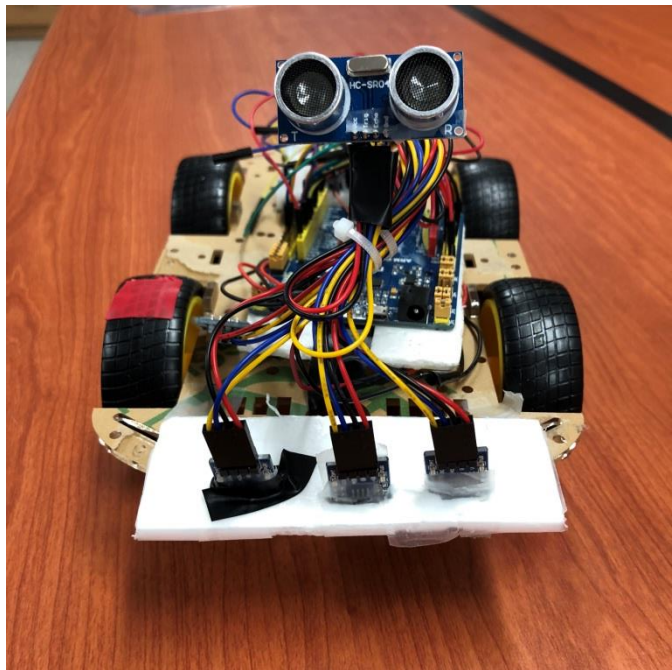
- 장점
- 부족한점
- 역할 분담

INDEX

01 소개

01 소개

무인 서빙 시스템



- route



01 소개



01 소개



안전하고 편하게 배달할 수 있는
방법은 없을까?

01 소개

무인 서빙 시스템



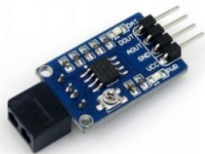
02 작품도안

02 작품도안

- 전체 작품 도안



02 작품도안 - 재료



IR sensor



Bluetooth sensor



DC motor



Dc motor driver



Ultra sonic wave

02 작품도안

- 연결도

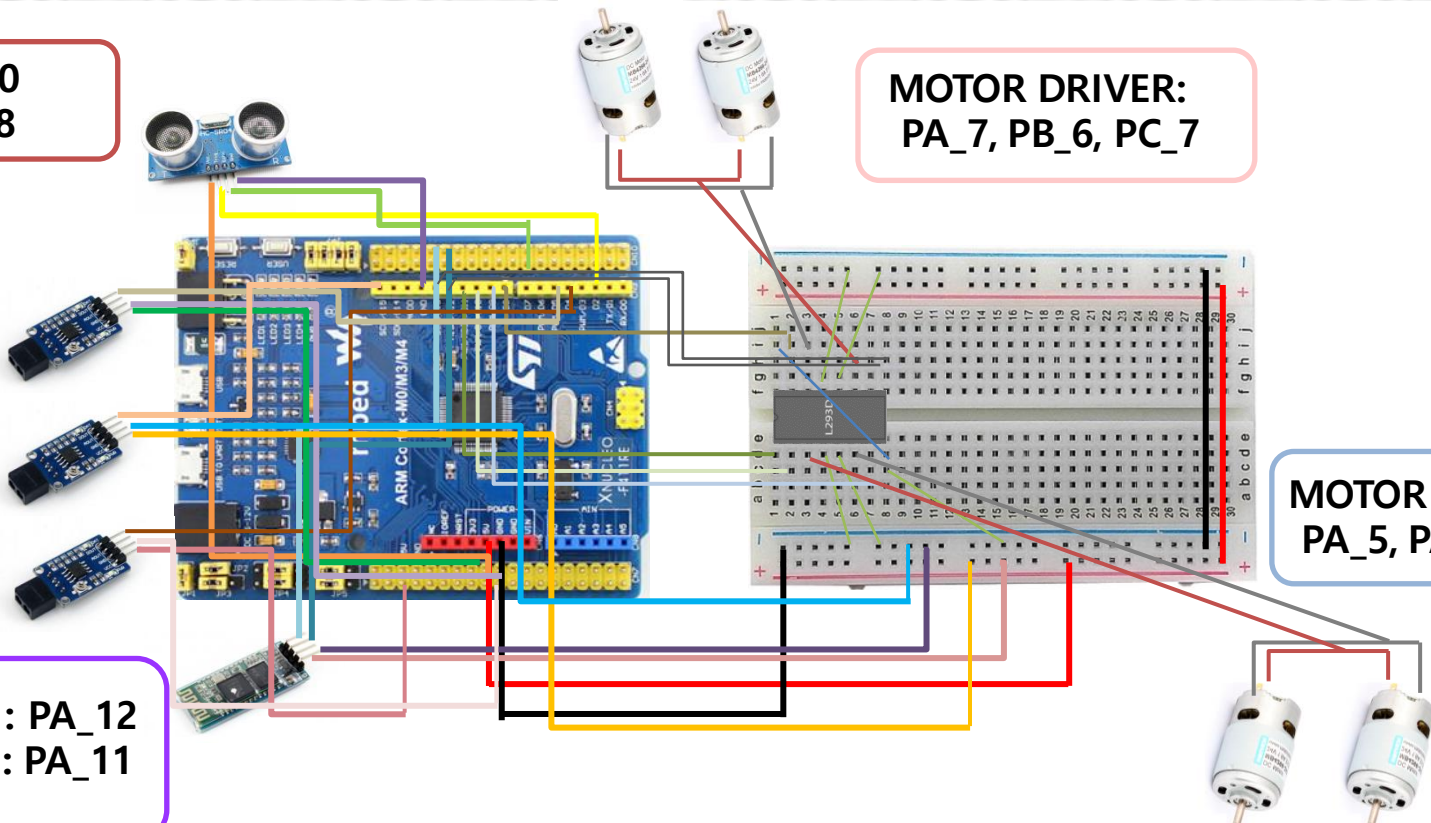
Trig : PA_10
Echo : PA_8

IR (L) : D4
IR (M) : D15
IR (R) : D5

RX : PA_12
TX : PA_11

MOTOR DRIVER:
PA_7, PB_6, PC_7

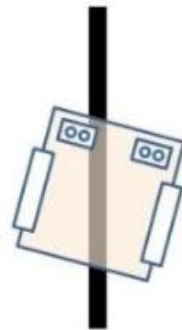
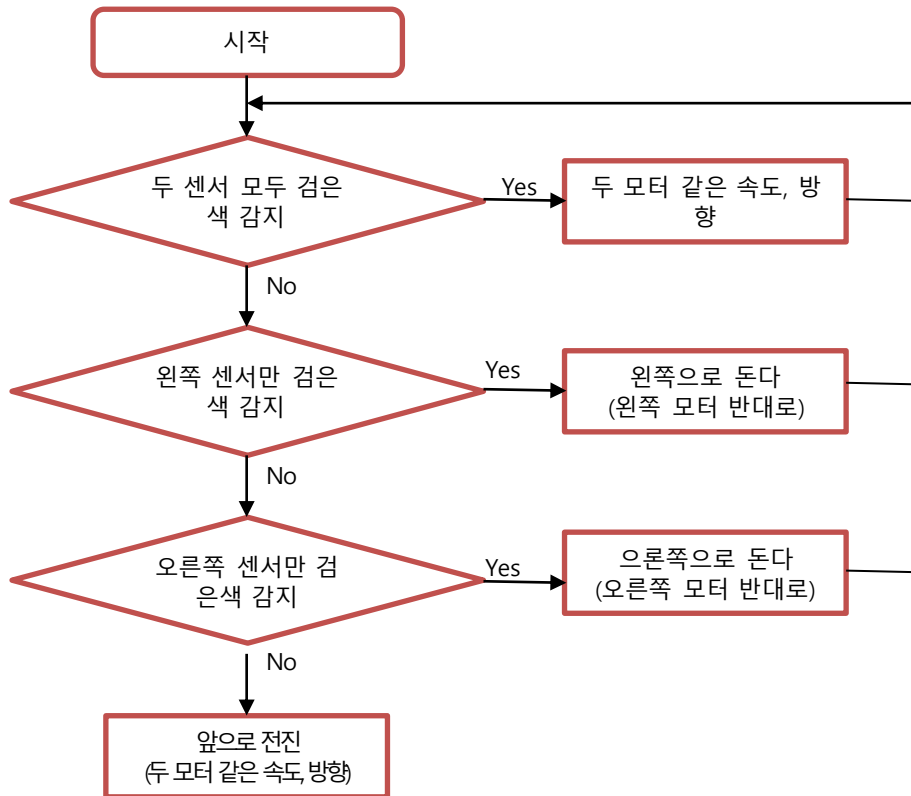
MOTOR DRIVER:
PA_5, PA_6, PA_9



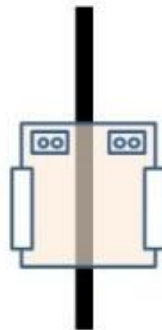
03 소스 코드

03 소스 코드

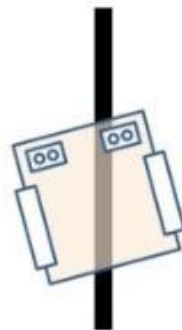
- 라인트레이서 동작 원리



왼쪽 센서가
닿을 경우



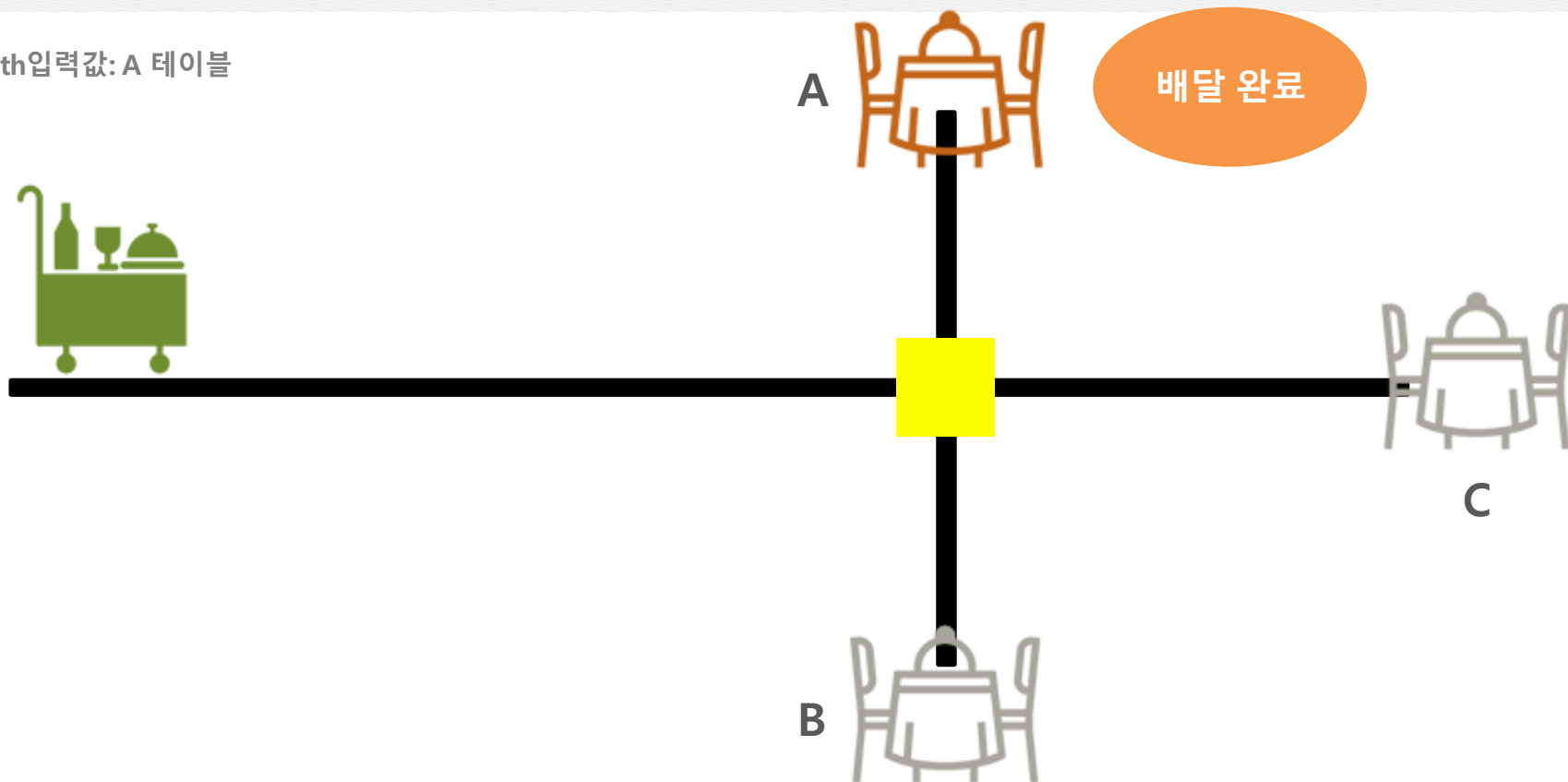
검은선이
중앙일 경우



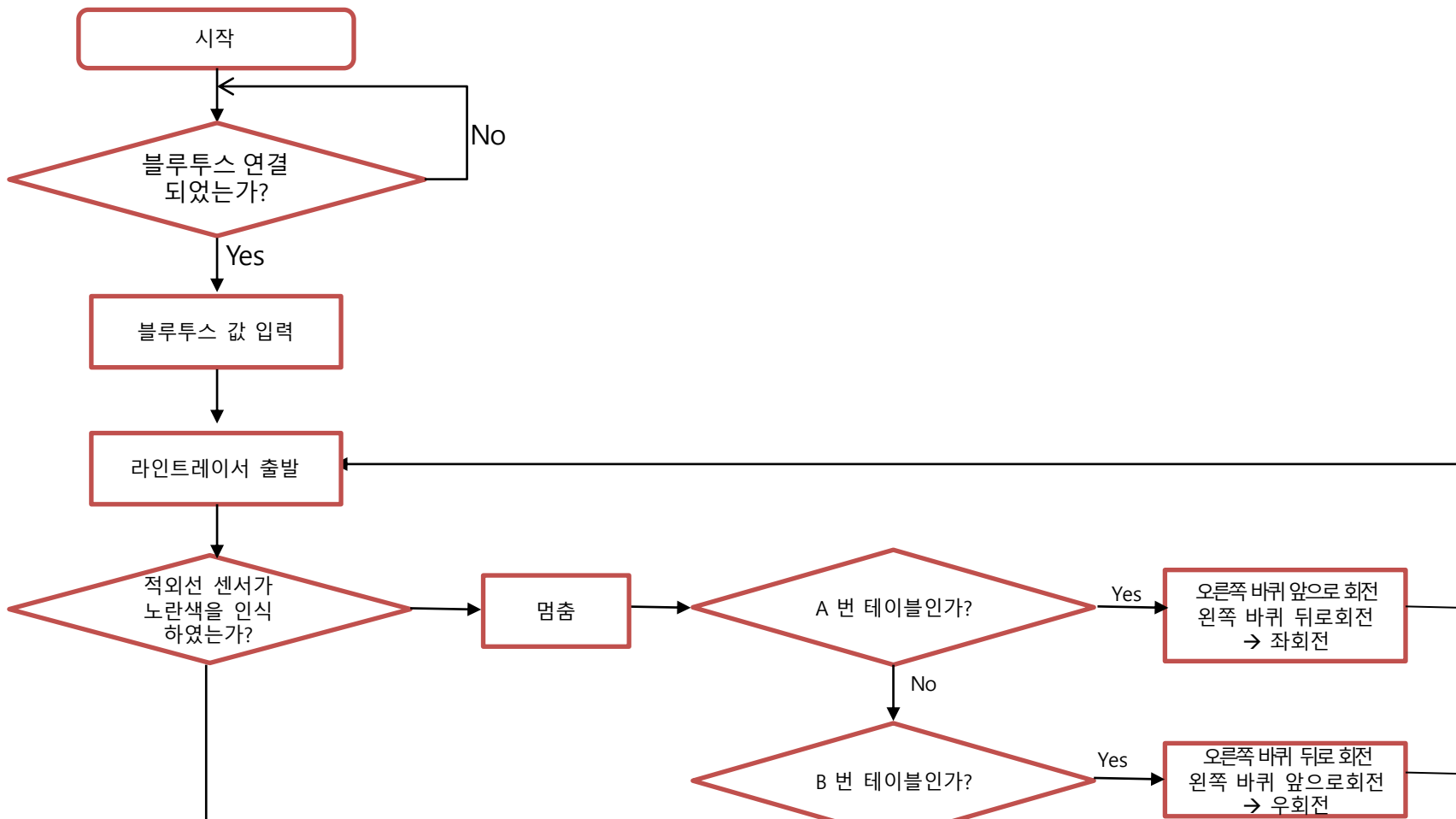
오른쪽 센서가
닿을 경우

03 소스코드

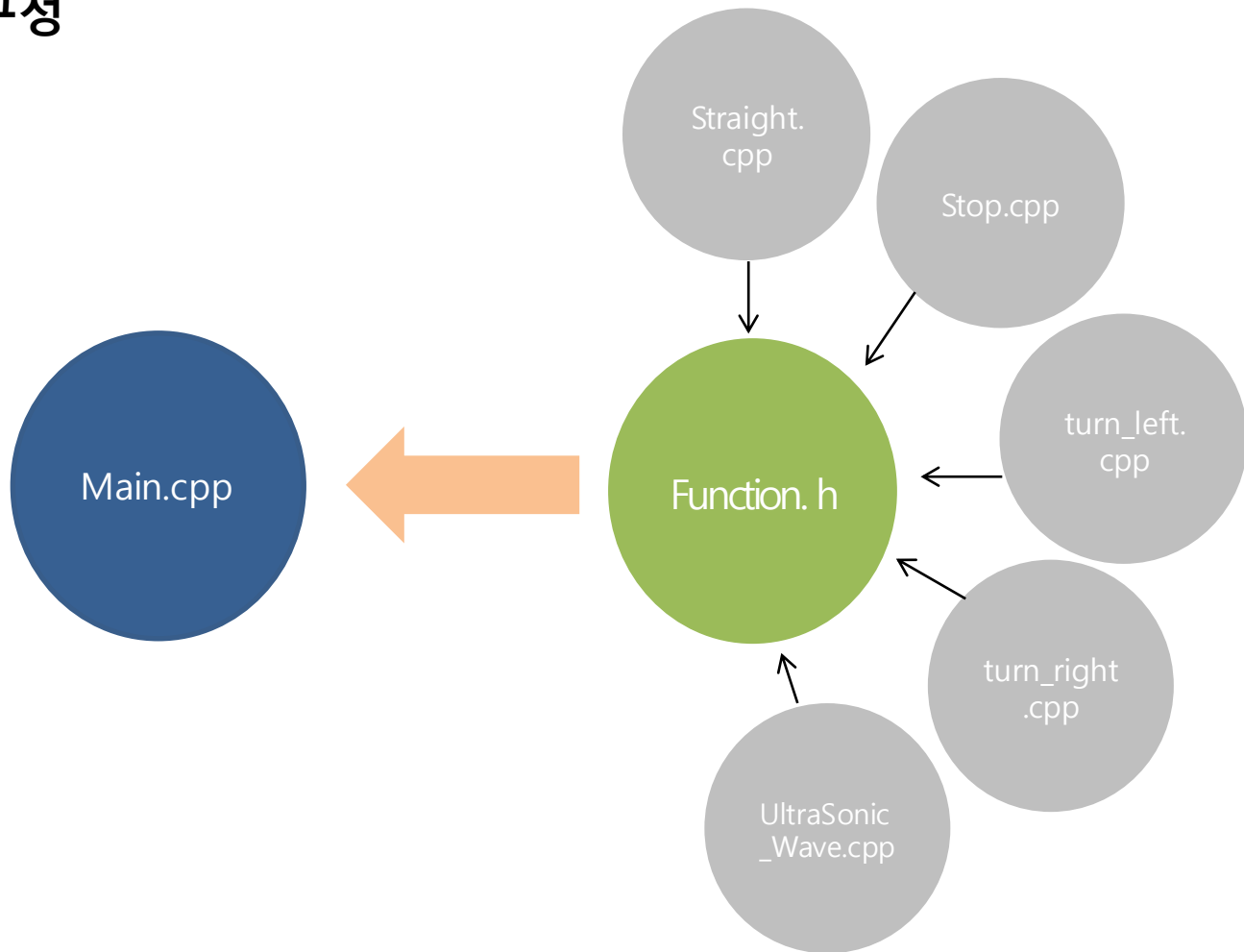
Bluetooth입력값: A 테이블



- Flow chart



◆ 시스템 구성



Function.h

```
#include "mbed.h"
```

```
int UltraSonic_Wave();
```

```
void straight();
```

```
void turn_right();
```

```
void turn_left();
```

```
void stop();
```

//파일을 불러온다.

straight.cpp

```
#include "function.h"

////////// right_motor
extern DigitalOut PIN_01; // dc motor
extern DigitalOut PIN_02;
extern DigitalOut PIN_03;
////////// left_motor
extern DigitalOut PIN_04;
extern DigitalOut PIN_05;
extern DigitalOut PIN_06;
void straight()
```

```
{
```

```
PIN_04 = 1;
PIN_05 = 0;
PIN_06 = 1;
```



역방향

```
PIN_01 = 1;
PIN_02 = 1;
PIN_03 = 0;
```



정방향

```
}
```



Stop.cpp

```
#include "function.h"

////////// right_motor
extern DigitalOut PIN_01; // dc motor
extern DigitalOut PIN_02;
extern DigitalOut PIN_03;

////////// left_motor
extern DigitalOut PIN_04;
extern DigitalOut PIN_05;
extern DigitalOut PIN_06;

void stop()
```

```
{
```

```
    PIN_04 = 1;
    PIN_05 = 0;
    PIN_06 = 0;
```

→ 정지

```
    PIN_01 = 1;
    PIN_02 = 0;
    PIN_03 = 0;
```

→ 정지

```
}
```

turn_left.cpp

```
#include "function.h"
////////// right_motor
extern DigitalOut PIN_01; // dc motor
extern DigitalOut PIN_02;
extern DigitalOut PIN_03;
////////// left_motor
extern DigitalOut PIN_04;
extern DigitalOut PIN_05;
extern DigitalOut PIN_06;
void turn_left()
{
```

```
PIN_04 = 1;
PIN_05 = 1;
PIN_06 = 0;
```

정방향

```
PIN_01 = 1;
PIN_02 = 1;
PIN_03 = 0;
```

정방향



turn_right.cpp

```
#include "function.h"
////////// right_motor
extern DigitalOut PIN_01; // dc motor
extern DigitalOut PIN_02;
extern DigitalOut PIN_03;
////////// left_motor
extern DigitalOut PIN_04;
extern DigitalOut PIN_05;
extern DigitalOut PIN_06;
void turn_right ()
```

```
{
```

```
PIN_04 = 1;
PIN_05 = 0;
PIN_06 = 1;
```



역방향

```
PIN_01 = 1;
PIN_02 = 0;
PIN_03 = 1;
```



역방향

```
}
```



UltraSonic_Wave.cpp

```
#include "function.h"
extern DigitalOut Trig; // choempwa sensor
extern DigitalIn Echo;
Timer t;
float i;
```

```
int UltraSonic_Wave(){
    t.start();
    // 펄스를 내 보낸다
    Trig = 1;
    wait(0.00004);
    Trig = 0;
    // 에코가 high 될 때까지 대기
    while(!Echo);

    // 에코가 high에서 low가 될 때까지 시간 측정
    t.reset();
    while(Echo);
    i = t.read_us();
    i = i/58;
    return i;
```

→ 거리 반환

Main.cpp

```
#include "mbed.h"
```

```
#include "function.h"
```

```
/// ultrasonic
```

```
DigitalOut Trig(PA_10); // choempwa sensor
```

```
DigitalIn Echo(PA_8);
```

```
/// bluetooth
```

```
Serial device(PA_11, PA_12);
```

```
char bt_get;
```

```
/// led for checking
```

```
DigitalOut led2bun(PC_9);
```

```
DigitalOut led3bun(PC_8);
```

```
DigitalOut led4bun(PC_5);
```

```
DigitalOut led(LED1); // definition of the green led on the board as digital output
```

```
/// ir
```

```
DigitalIn ir_left(D4); // definition of the pin D4 as digital input ir_sensor
```

```
DigitalIn ir_right(D5); // definition of the pin D5 as digital input ir-sensor
```

```
DigitalIn ir_Middle(D15); // definition of the pin D15 as digital input
```

```
/// motor
```

```
DigitalOut PIN_01(PA_7); // dc motor
```

```
DigitalOut PIN_02(PB_6);
```

```
DigitalOut PIN_03(PC_7);
```

Main.cpp

```
int main() {
```

```
    int choempwa_value;
```

```
    device.attach(&Rx_Interrupt);
```

```
    int bt_input;
```

```
    while(1) {
```

```
        choempwa_value = UltraSonic_Wave();
```

```
        if(bt_get == 'a')
```

```
            bt_input = 1;
```

```
        if(bt_get == 'b')
```

```
            bt_input = 2;
```

```
        if(bt_get == 'c')
```

```
            bt_input = 3;
```

bt_input에 블루투스
로 받은 값을 넣음

Main.cpp

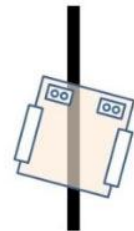
A 테이블인 경우

```
if (bt_input == 1){
```

```
    if (choempwa_value < 7){  
        stop();  
    }
```

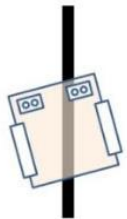
거리가 7cm이내 일 때 정지

```
    else if (ir_left == 1 && ir_right == 0){  
        turn_left();  
        wait(0.05);  
    }
```



왼쪽 센서가
닿을 경우

```
    else if (ir_left == 0 && ir_right == 1){  
        turn_right();  
        wait(0.05);  
    }
```



오른쪽 센서가
닿을 경우

```
    else if (ir_left == 0 && ir_right == 0 && ir_Middle == 0){  
        led2bun = 1;  
        stop();  
        wait(0.1);  
        turn_left();  
        wait(3);  
    }
```

컬러를 인식했을때 A테이블(왼쪽)로 회전

```
    else{  
        straight();
```



Main.cpp

```
else if(bt_input == 2)
{
```

```
...
```

```
    else if (ir_left==0 && ir_right ==0 && ir_Middle==0)
```

```
        led3bun = 1;
```

```
        stop();
```

```
        wait(0.1);
```

```
        turn_right();
```

```
        wait(3);
```

```
    }
```

```
...
```

```
}
```

```
else if(bt_input == 3)
```

```
{
```

```
...
```

```
    else if (ir_left==0 && ir_right ==0 && ir_Middle==0)
```

```
        led2bun = 1;
```

```
        stop();
```

```
        wait(0.1);
```

```
        straight();
```

```
        wait(3);
```

```
    }
```

```
...
```

```
}
```

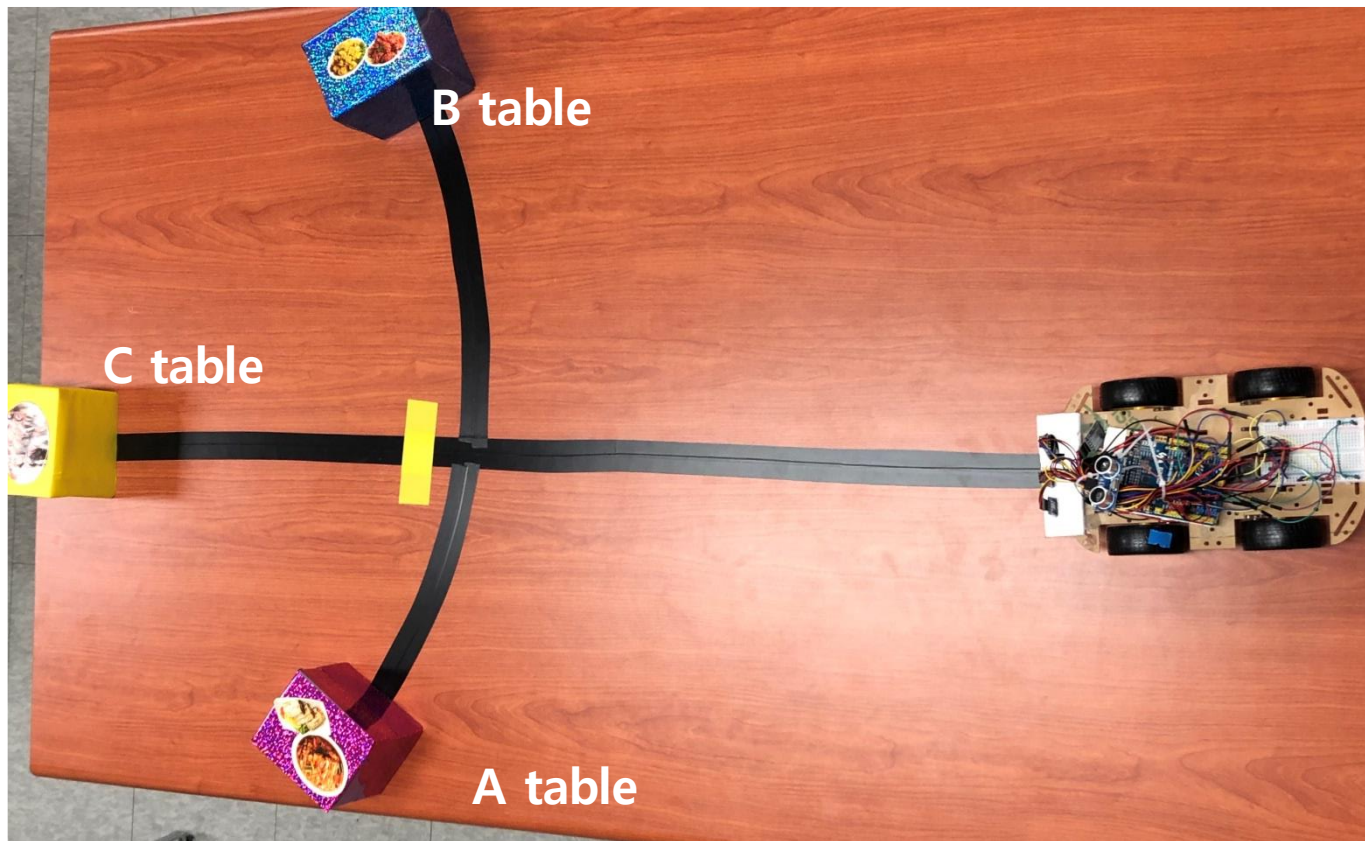
컬러를 인식했을때 B테이블
(오른쪽)로 회전

컬러를 인식했을때 C 테이블
로 직진

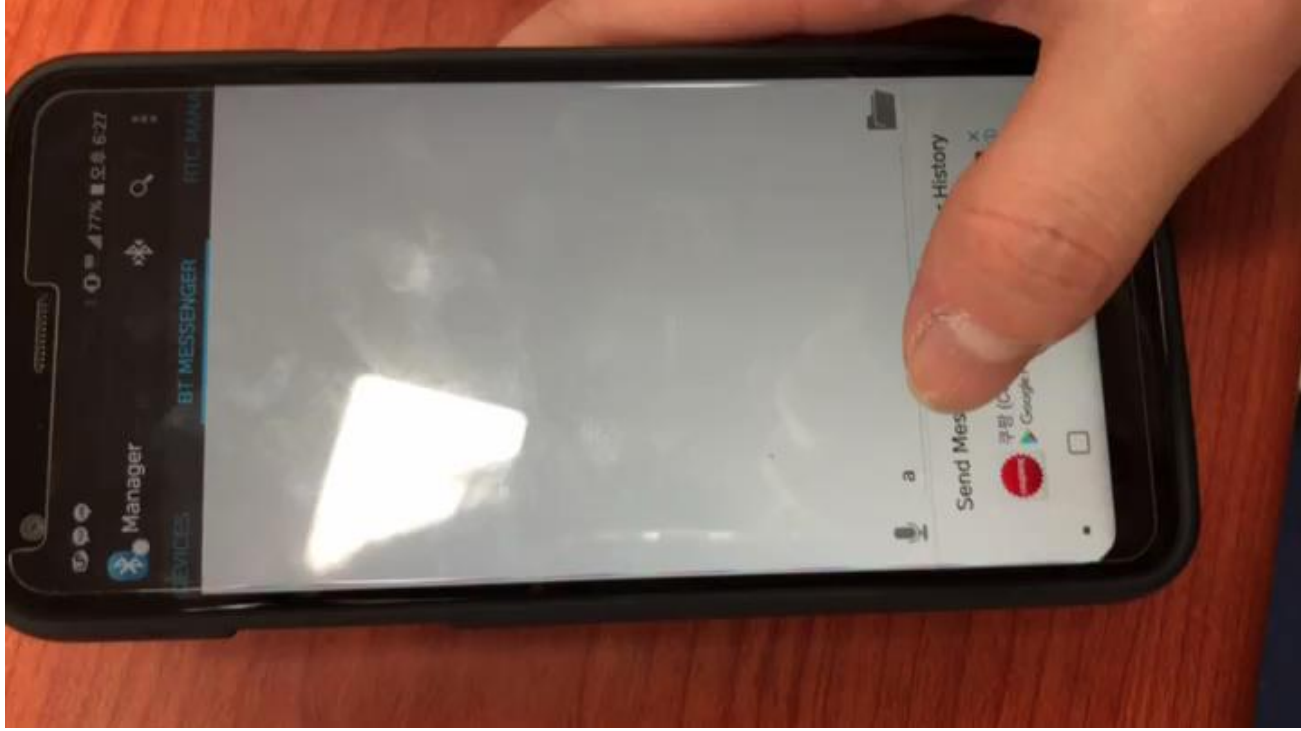
04 동작 동영상

04 동작 동영상

- 전체 작품 도안

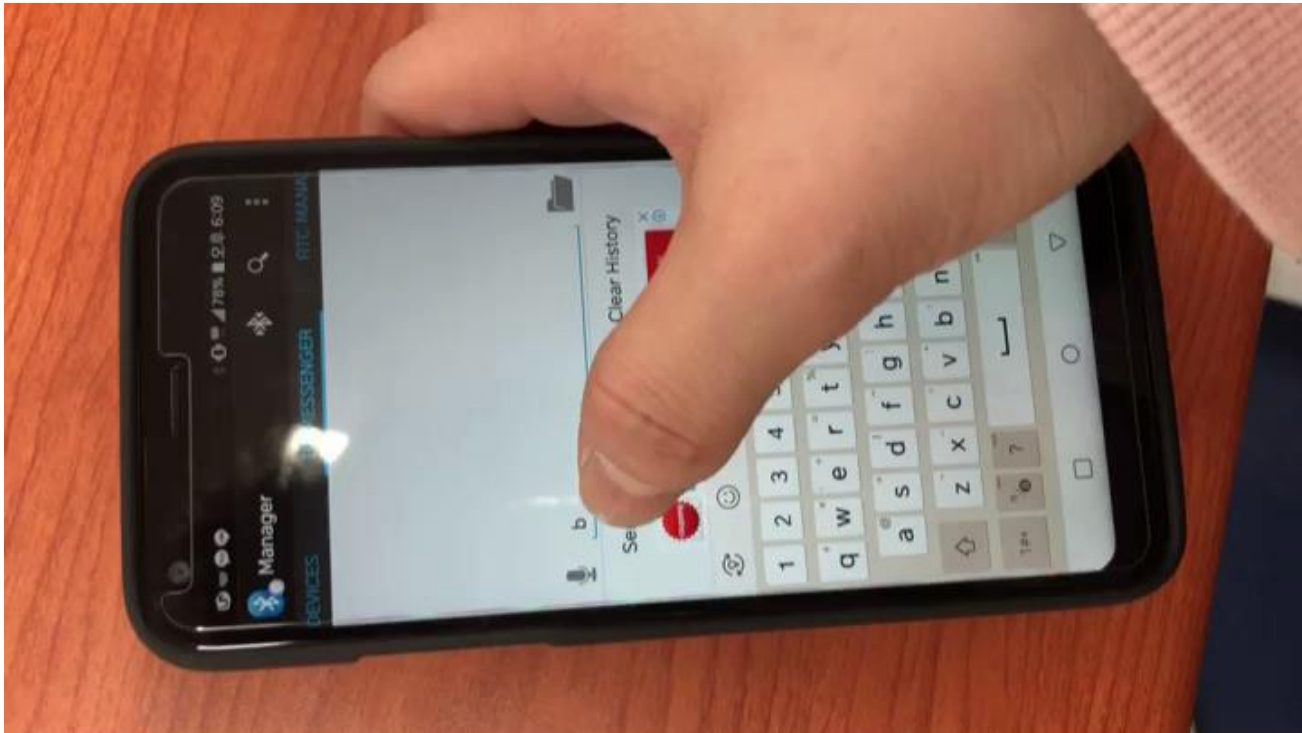


04 동작 동영상 - A 테이블



04 동작 동영상

- B 테이블



04 동작 동영상 - C 테이블



05 결론

05 결론

◆ 장점

- 인건비 절감
- 높은 공간 효율성

05 결론

◆ 부족한점

- 모터 힘이 약함



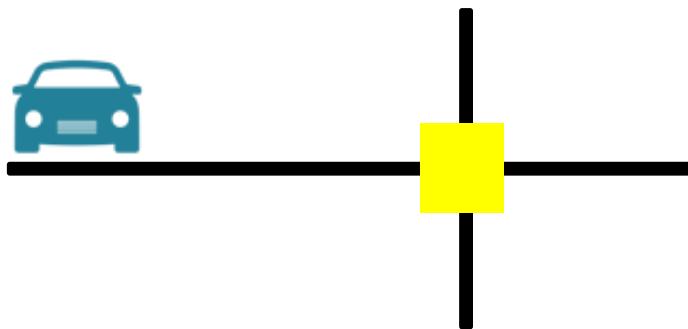
- 하드웨어적 문제



05 결론

◆ 부족한점

- 컬러 인식 IR 센서 사용





Thank You!