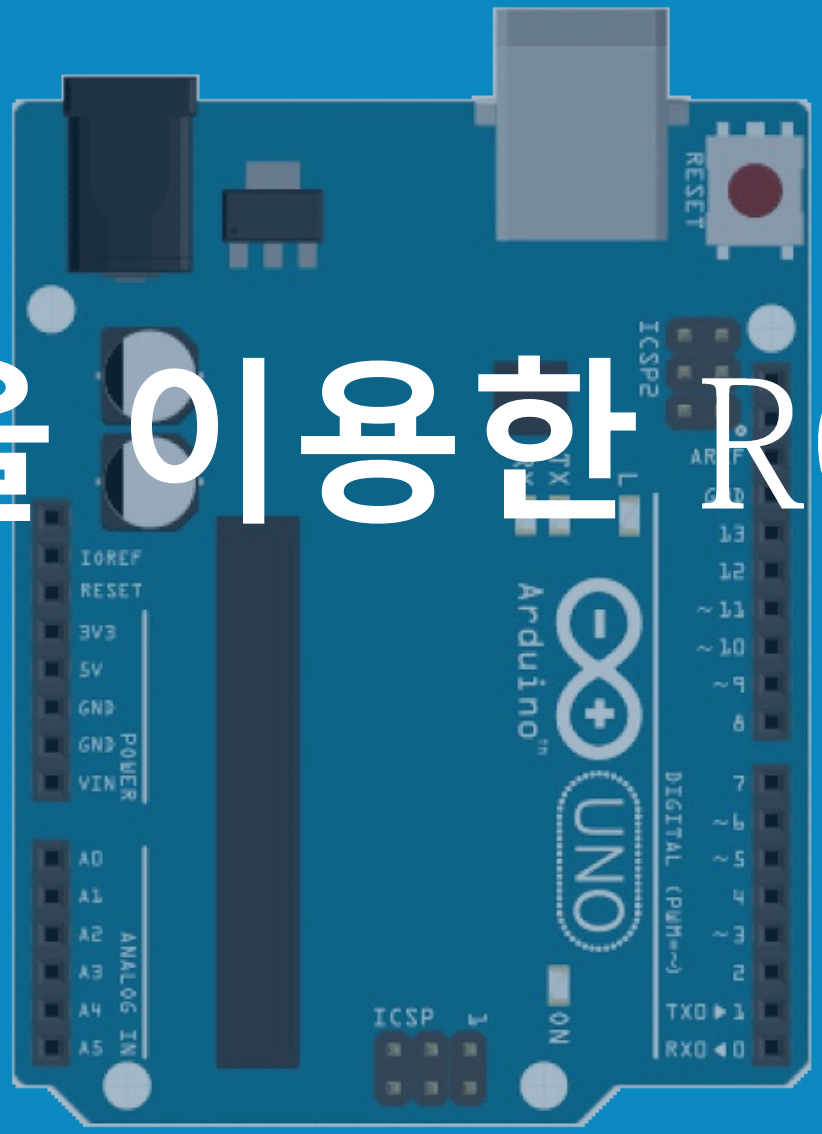
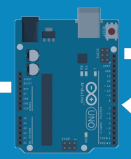
조이스틱을 이용한 RC카 만들기



AR19 이정호
AR23 진우태

목  차

개발 동기

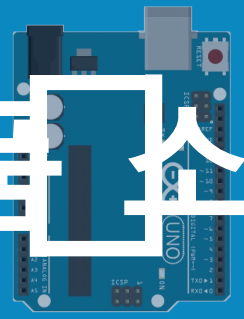
조이스틱?

개발 동기

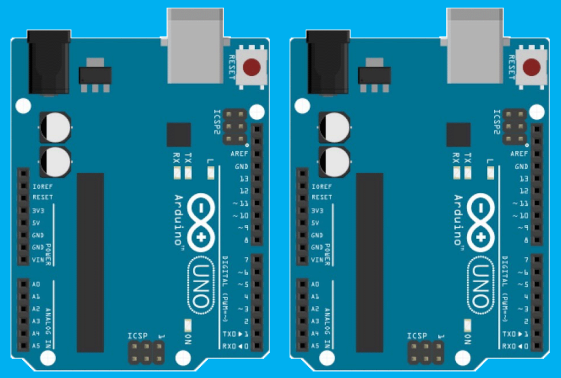
역할분담

AR19 이정호	AR23 진우태
프로젝트아이디어 구상	
PPT 제작 및 자료수집	
아두이노 코드 오류수정 등	
Fritzing 회로 그리기	
조종기 및 본체 제작	
처음부터 끝까지 같이 만나서 프로젝트를 진행했습니다.	

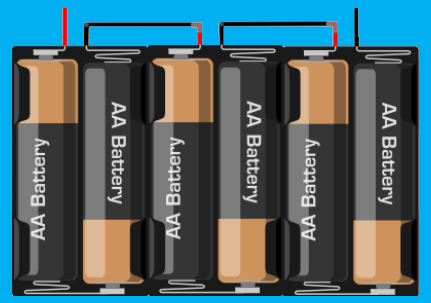
부품 소개



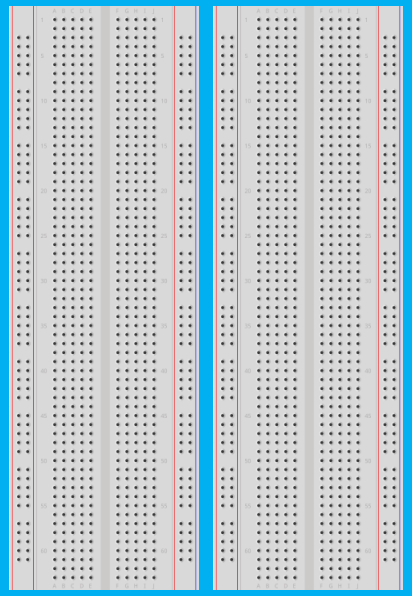
부품 소개



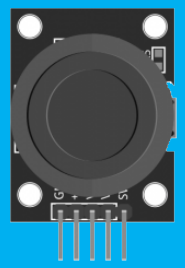
아두이노 UNO X 2



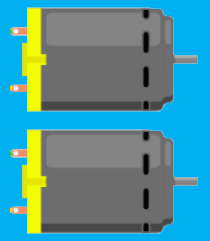
건전지홀더



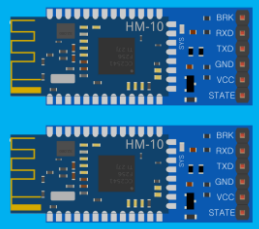
브레드보드 X 2



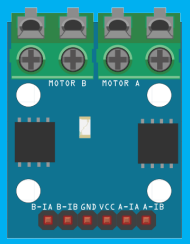
조이스틱



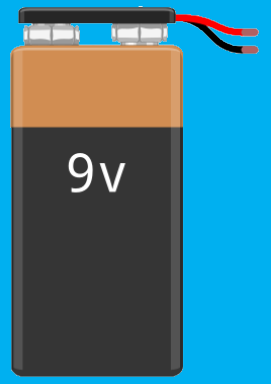
DC모터 X 2



HM-10 X 2

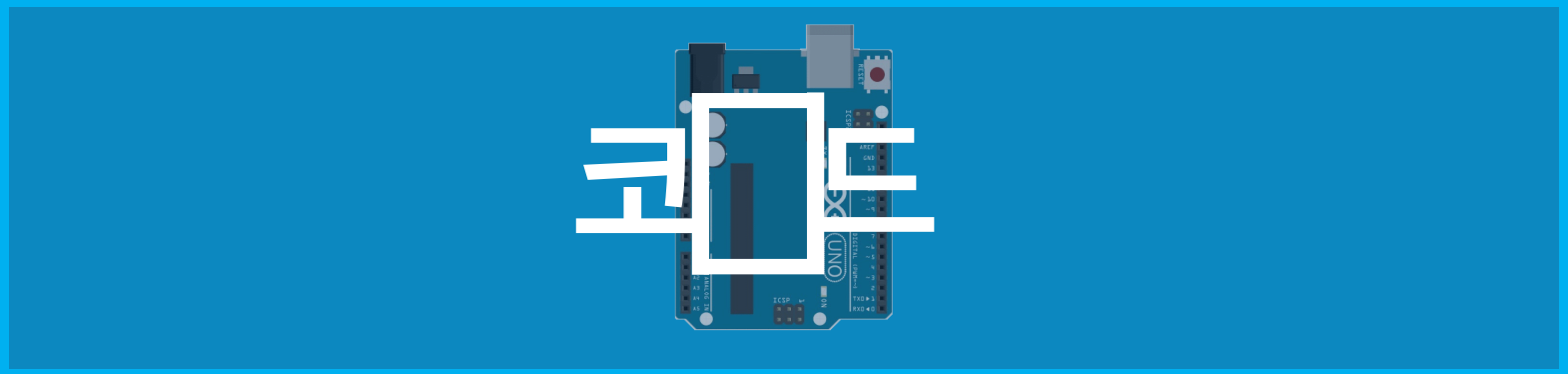


L9110S

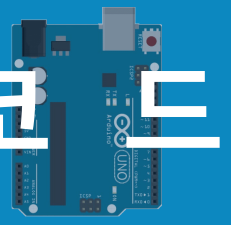


9V 건전지

바퀴 X 3


코드

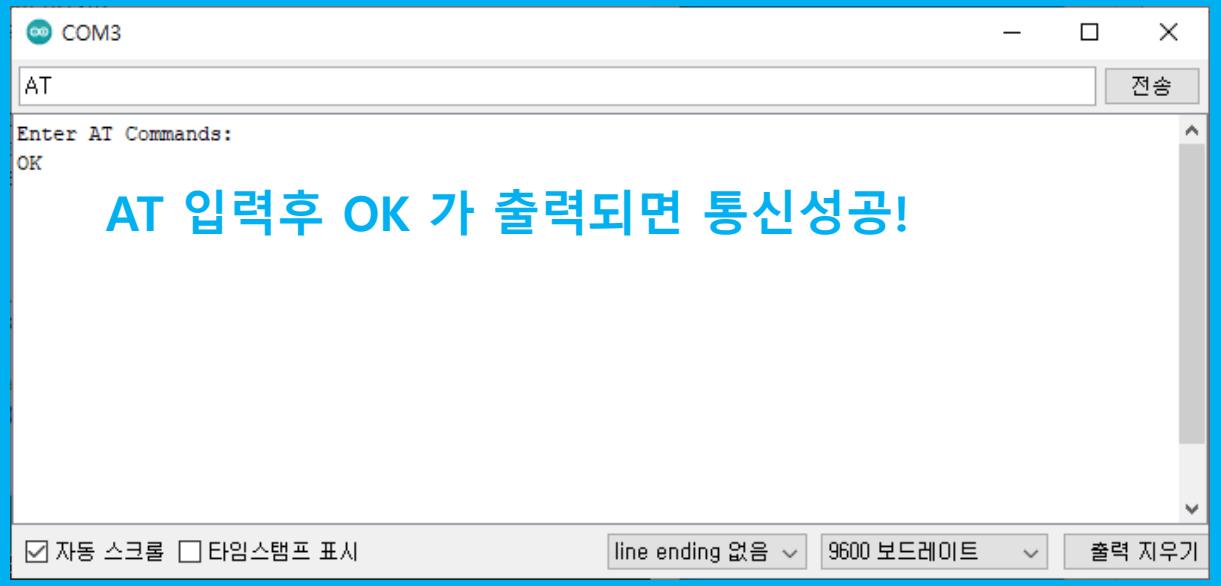
코드

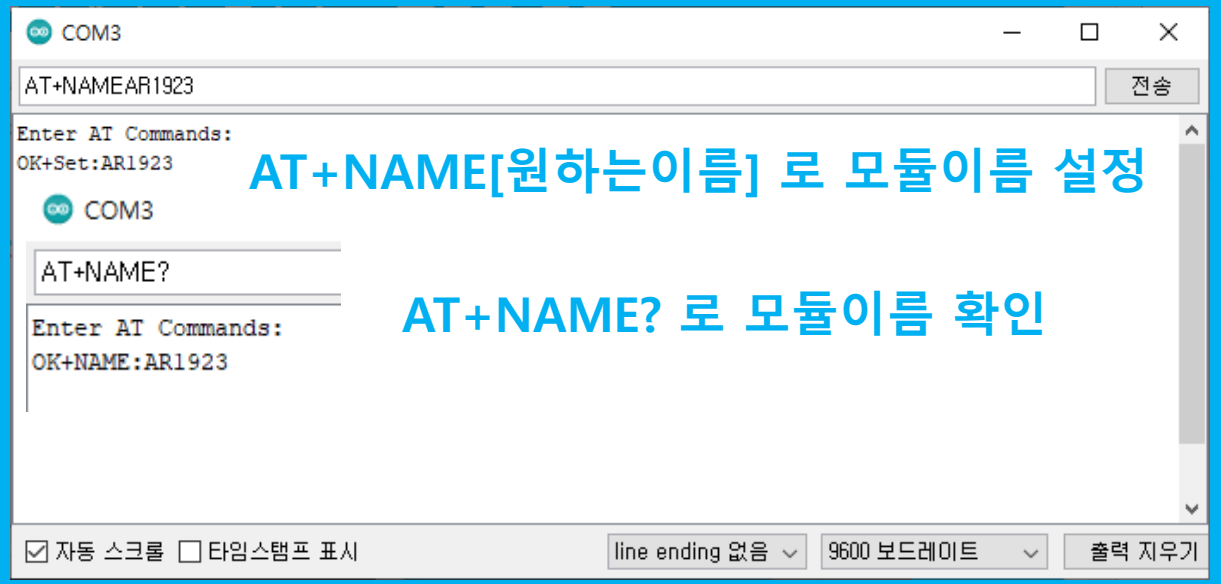


블루투스 연결코드

```
#include <SoftwareSerial.h>
SoftwareSerial AR1923BT(2, 3); //BlueTooth(Rx, Tx)

void setup() {
  Serial.begin(9600);
  Serial.println("Enter AT Commands: ");
  AR1923BT.begin(9600);
}

void loop() {
  if (AR1923BT.available())
    Serial.write(AR1923BT.read());

  if (Serial.available())
    AR1923BT.write(Serial.read());
}
```

커맨드 입력해서 두 블루투스 모듈을 연결 해야한다.





블루투스 설정 커맨드 참고 사이트 : https://hyesunzzang.tistory.com/20

코드

COM3

AT+ROLE0

```
Enter AT Commands:
OK+Set:0
```

하나는 AT+ROLE0 을
입력해 Slave로 설정

COM3

AT+ROLE1

```
Enter AT Commands:
OK+Set:1
```

하나는 AT+ROLE0 을
입력해 Master로 설정

COM3

AT+IMME1

```
Enter AT Commands:
OK+Set:1
```

AT+IMME1 을 입력해
모듈의 동작타입 설정

1: When module is powered on, only respond the AT Command, don't do anything. until AT + START is received, or can use AT+CON,AT+CONNL
0: When power on, work immediately

COM3

AT+ADDR?

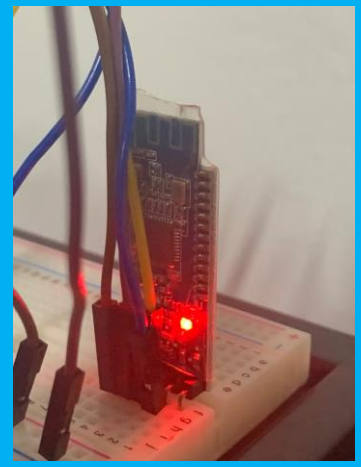
```
Enter AT Commands:
OK+ADDR:D436399B3EF5
```

COM4

AT+ADDR?

```
Enter AT Commands:
OK+ADDR:D436399CC8A0
```

AT+ADDR? 을 입력해
각각의 모듈의 주소를 확인



COM3

AT+CONDD436399CC8A0

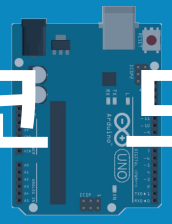
```
Enter AT Commands:
OK+CONNA
```

AT+CON[모듈의주소] 을 입력해
다른 모듈과 블루투스로 연결

HM-10 에서 불빛이 깜빡이다가
블루투스가 연결되면 깜빡이는게 멈추고 불빛이 들어옴

블루투스 설정 커맨드 참고 사이트 : https://hyesunzzang.tistory.com/20

코드

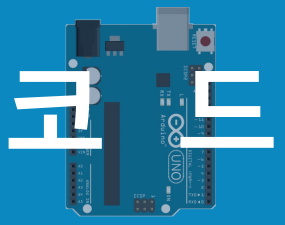


조종기 코드

```
#include <SoftwareSerial.h>
SoftwareSerial AR1923BT(2, 3); // AR1923BT(Rx, Tx)
int BTstates=0;        // 블루투스 신호 상태 저장용 변수

const int X_AXIS =0; //마우스 X 축 (A0)
const int Y_AXIS =1; //마우스 Y 축 (A1)
int xVal=0;
int yVal=0;

void setup() {
  AR1923BT.begin(9600);
  Serial.begin(9600);
}

```

```
void loop() {
  xVal=map(analogRead(X_AXIS),0,1023,100,0); //x축값 읽어 저장
  yVal=map(analogRead(Y_AXIS),0,1023,300,200); //Y축값 읽어 저장

  // 모터 전진 //
  if (xVal >= 60 && yVal >= 225 && yVal <=275)  {
    AR1923BT.write('f');
    Serial.println('f');
  }
  // 모터 후진 //
  else if (xVal <= 40 && yVal >= 225 && yVal <=275)  {
    AR1923BT.write('b');
    Serial.println('b');
  }
  // 모터 좌회전 //
  else if (yVal <= 240 && xVal >= 25 && xVal <= 75)  {
    AR1923BT.write('l');
    Serial.println('l');
  }
  // 모터 우회전 //
  else if (yVal >= 260 && xVal >= 25 && xVal <= 75)  {
    AR1923BT.write('r');
    Serial.println('r');
  }
 else {
    AR1923BT.write('s');
    Serial.print('s');
  }
  delay(100);
}
```

코드

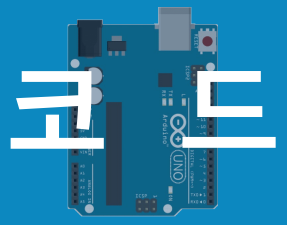


본체 코드

```ino
#include <SoftwareSerial.h>
SoftwareSerial AR1923BT(2, 3); // AR1923BT(Rx, Tx)
int A_1A = 9;
int A_1B = 10;
int B_1A = 5;
int B_1B = 6;
int speed = 250;    // speed: 0~ 255
char AR1923joyBT;
/* L9110s 모터드라이버
   오른쪽모터는
   L9110s A_1A 9
   L9110s A_1B 10
   왼쪽모터는
   L9110s B_1A 5
   L9110s B_1B 6
*/
void setup() {
  //핀을 초기화 하고, 출력설정
  pinMode(A_1A, OUTPUT);
  pinMode(A_1B, OUTPUT);
  pinMode(B_1A, OUTPUT);
  pinMode(B_1B, OUTPUT);
  digitalWrite(A_1A, LOW);
  digitalWrite(A_1B, LOW);
  digitalWrite(B_1A, LOW);
  digitalWrite(B_1B, LOW);
  Serial.begin(9600);
  AR1923BT.begin(9600);
}

```

```ino
void loop() {
  if (AR1923BT.available()) {
    AR1923joyBT = AR1923BT.read();
    Serial.println(AR1923joyBT);
    switch (AR1923joyBT) {
     case 'f' :         // 모터 전진
        //모터A
        analogWrite(A_1A, speed);
        analogWrite(A_1B, 0);
        //모터B
        analogWrite(B_1A, speed);
        analogWrite(B_1B, 0);
        break;

    case 'b' :         // 모터 후진
        analogWrite(A_1A, 0);
        analogWrite(A_1B, speed);
        analogWrite(B_1A, 0);
        analogWrite(B_1B, speed);
        break;

    case 'l' :         // 모터 좌회전
        analogWrite(A_1A, speed);
        analogWrite(A_1B, 0);
        analogWrite(B_1A, 0);
        analogWrite(B_1B, speed);
        break;

    case 'r' :         //모터 우회전
        analogWrite(A_1A, 0);
        analogWrite(A_1B, speed);
        analogWrite(B_1A, speed);
        analogWrite(B_1B, 0);
        break;

    case 's' :         // 모터 정지
      analogWrite(A_1A, 0);
      analogWrite(A_1B, 0);
      analogWrite(B_1A, 0);
      analogWrite(B_1B, 0);
      break;
    }
  }
}
```

코드



본체 시리얼 모니터

COM4

f
b
b
s
s
l
s
r
s
s
s
s
s

자동 스크롤 타임스탬프 표시 line ending 없음

조종기 시리얼 모니터

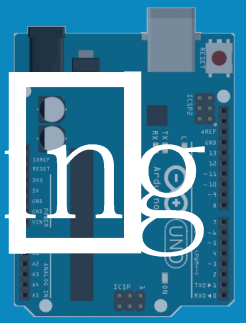
COM3

전송

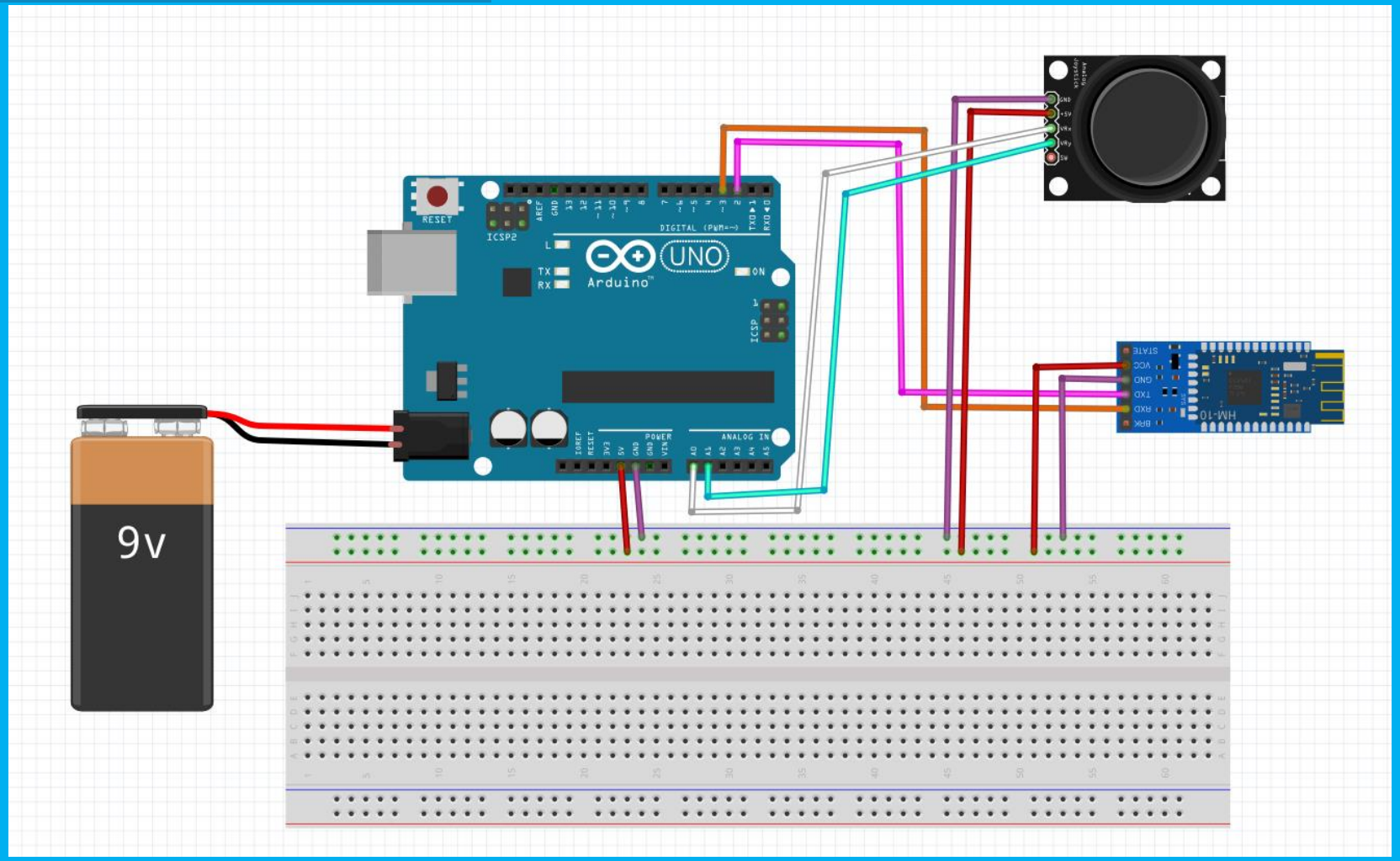
l
l
l
sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
f
sssssssb
sssssssr
r
r
sssl
l
l
sssssssssssssssssss

자동 스크롤 타임스탬프 표시 line ending 없음 9600 보드레이트 출력 지우기

Fritzing 회로

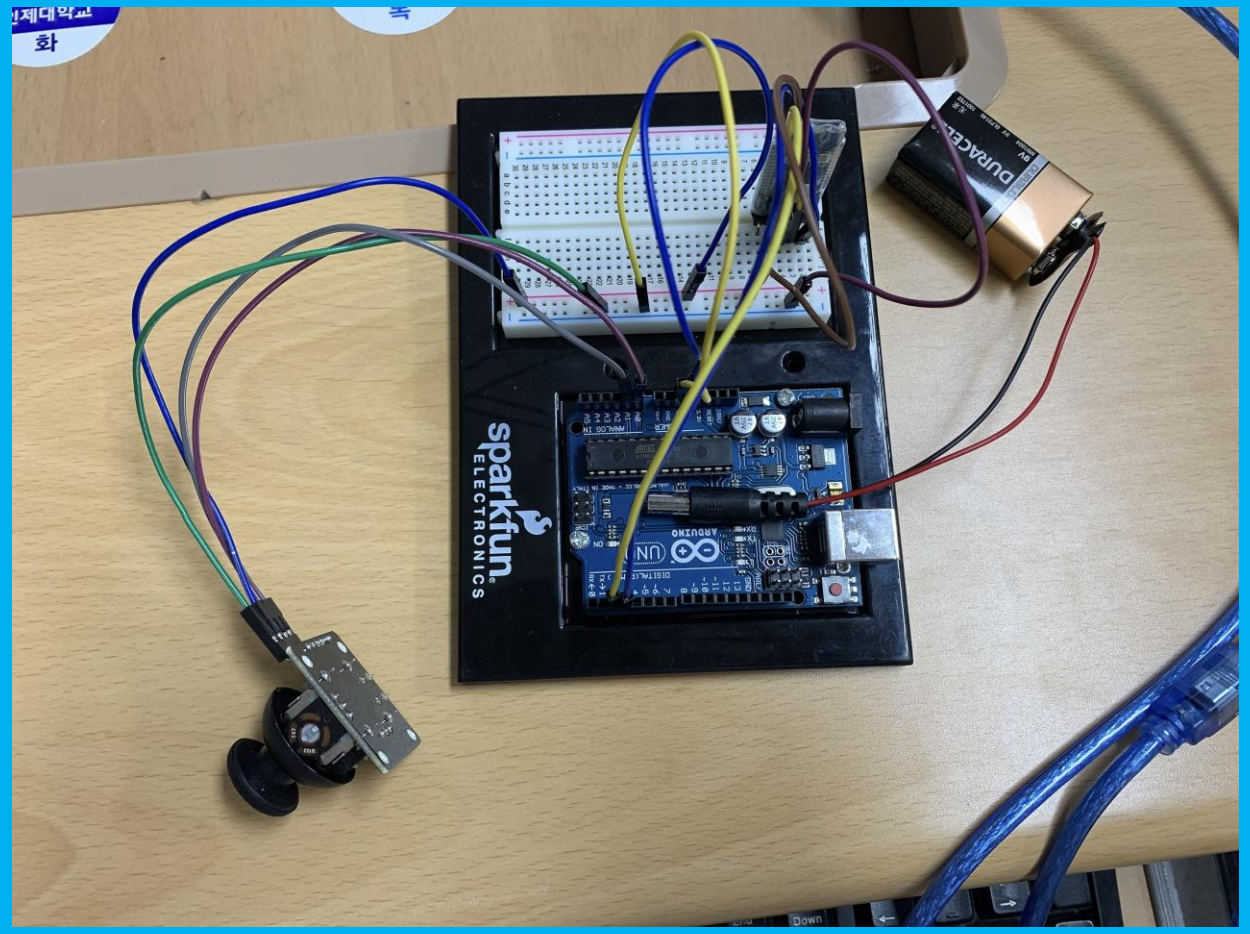


Fritzing 회로



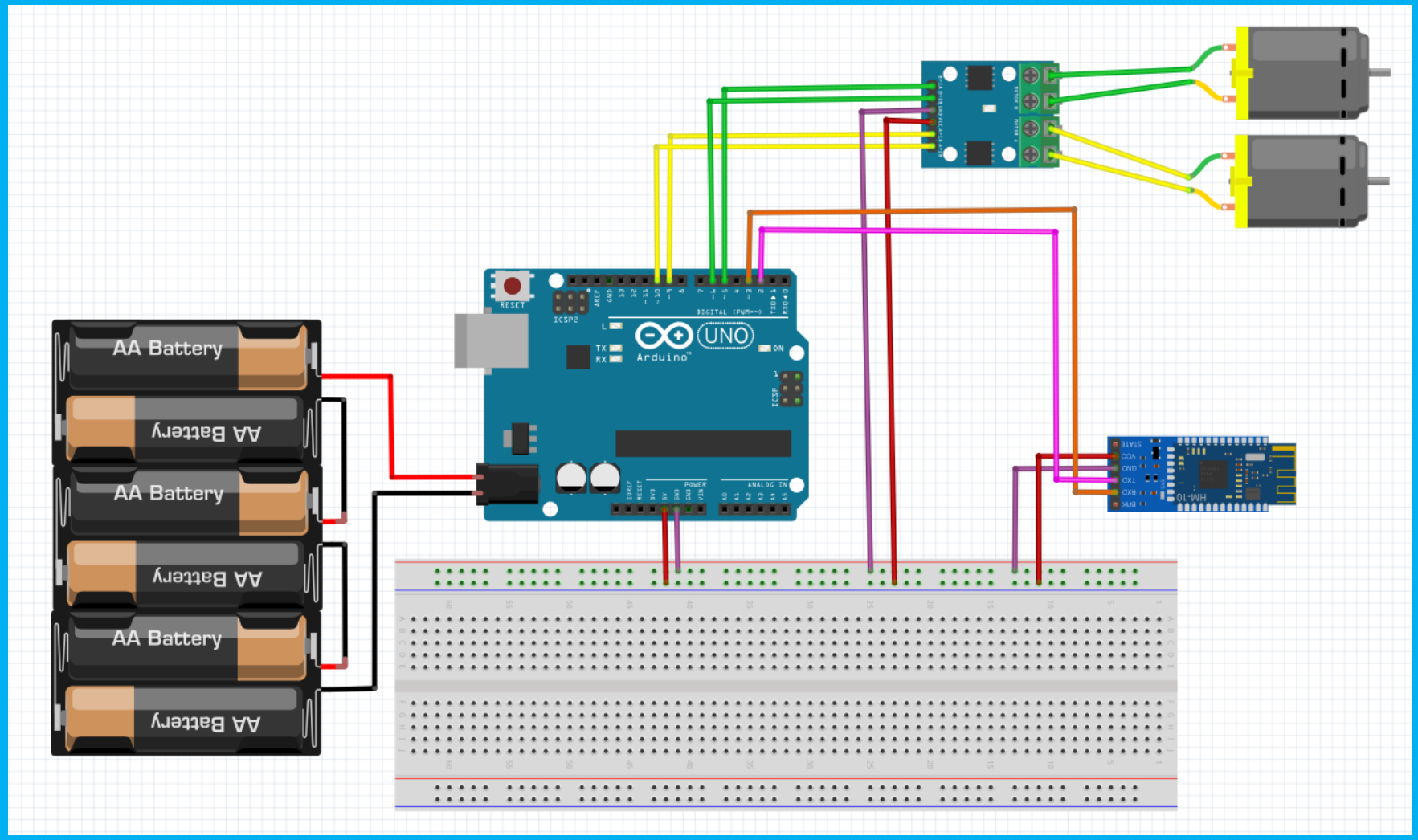
조종기 회로

「 Fritzing 회로 」



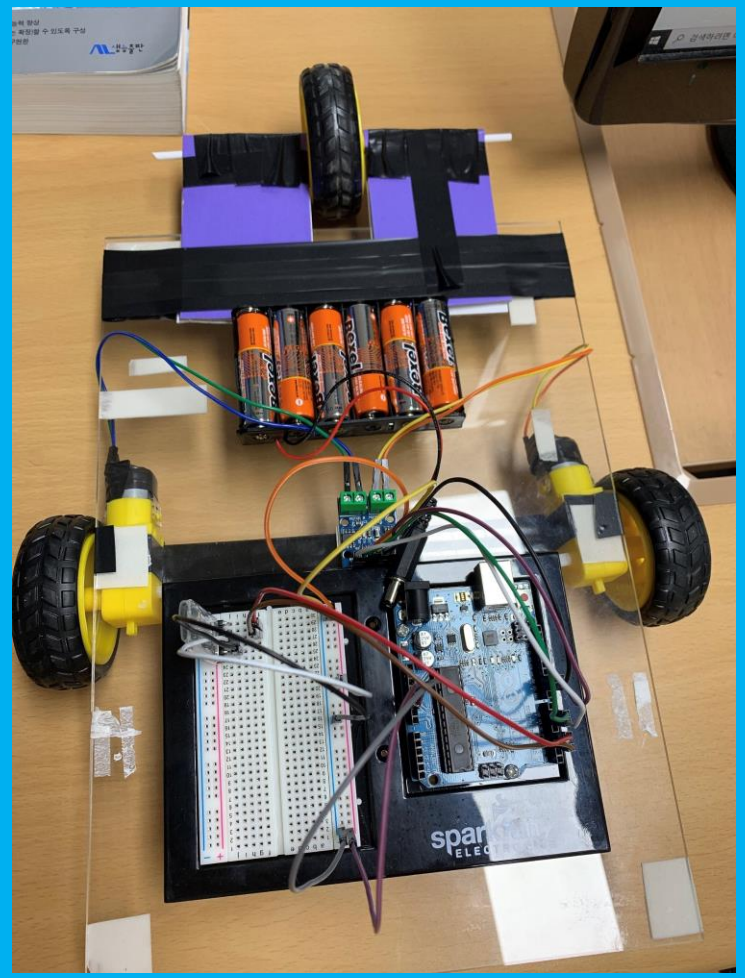
조종기 회로 실물

Fritzing 회로



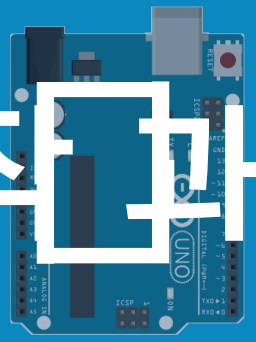
본체 회로

「 Fritzing 회로 」

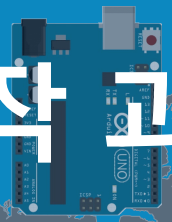


본체 회로 실물

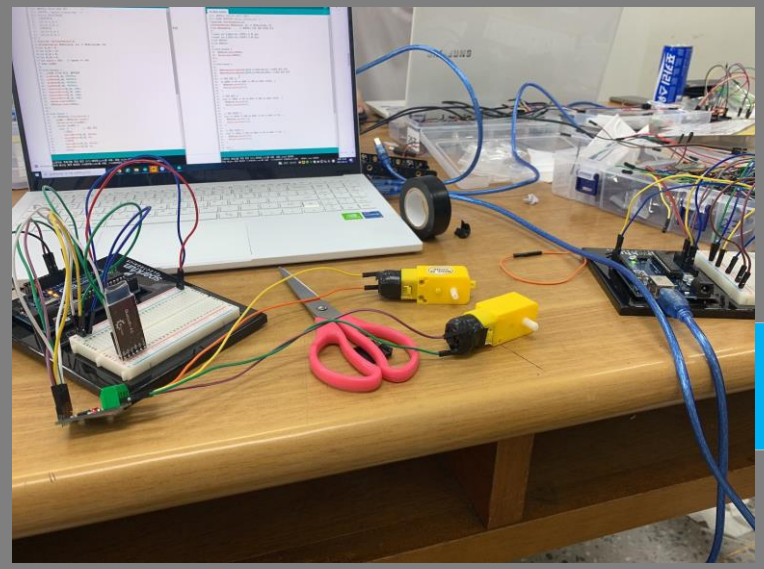
제작 과정

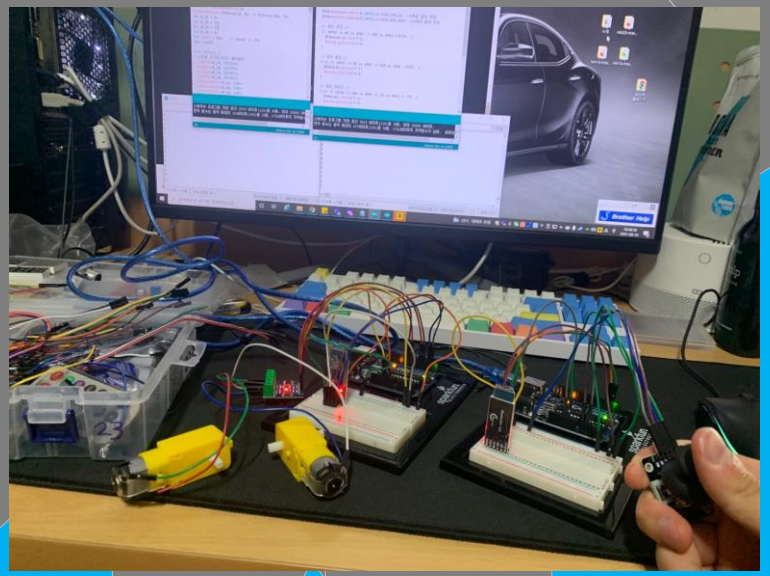


제작 과정

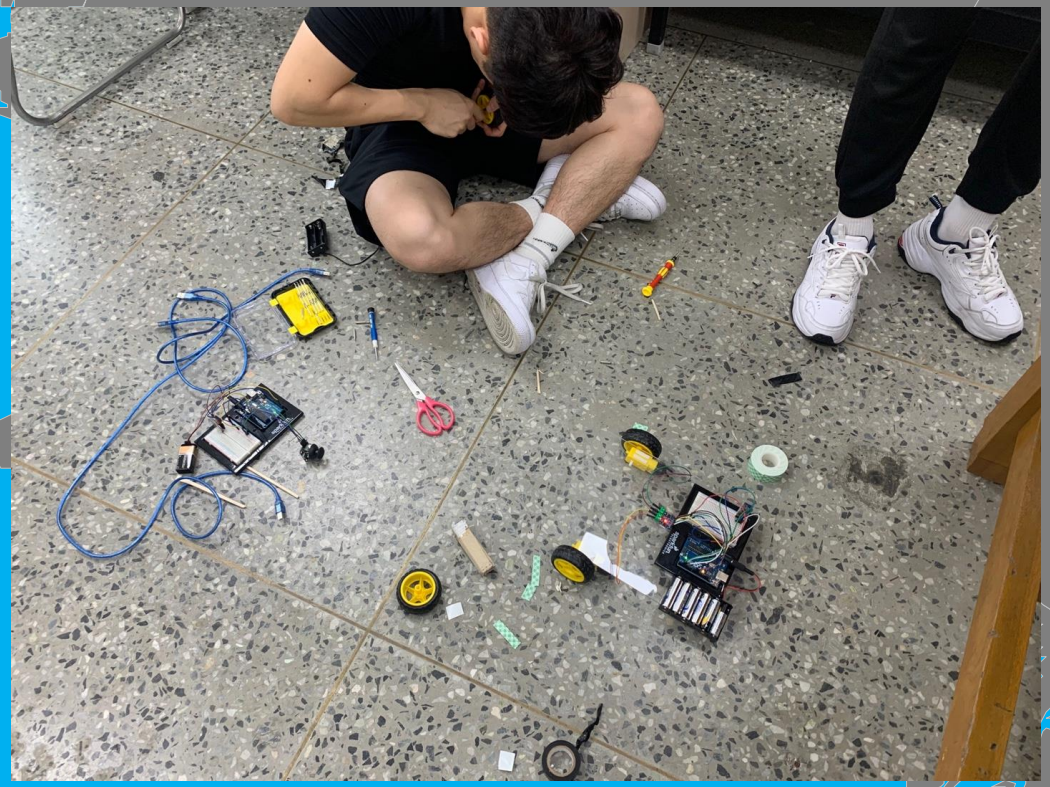


제작 과정





제작 과정



Thank you for listening //