

# LINE FOLLOWER ROBOT

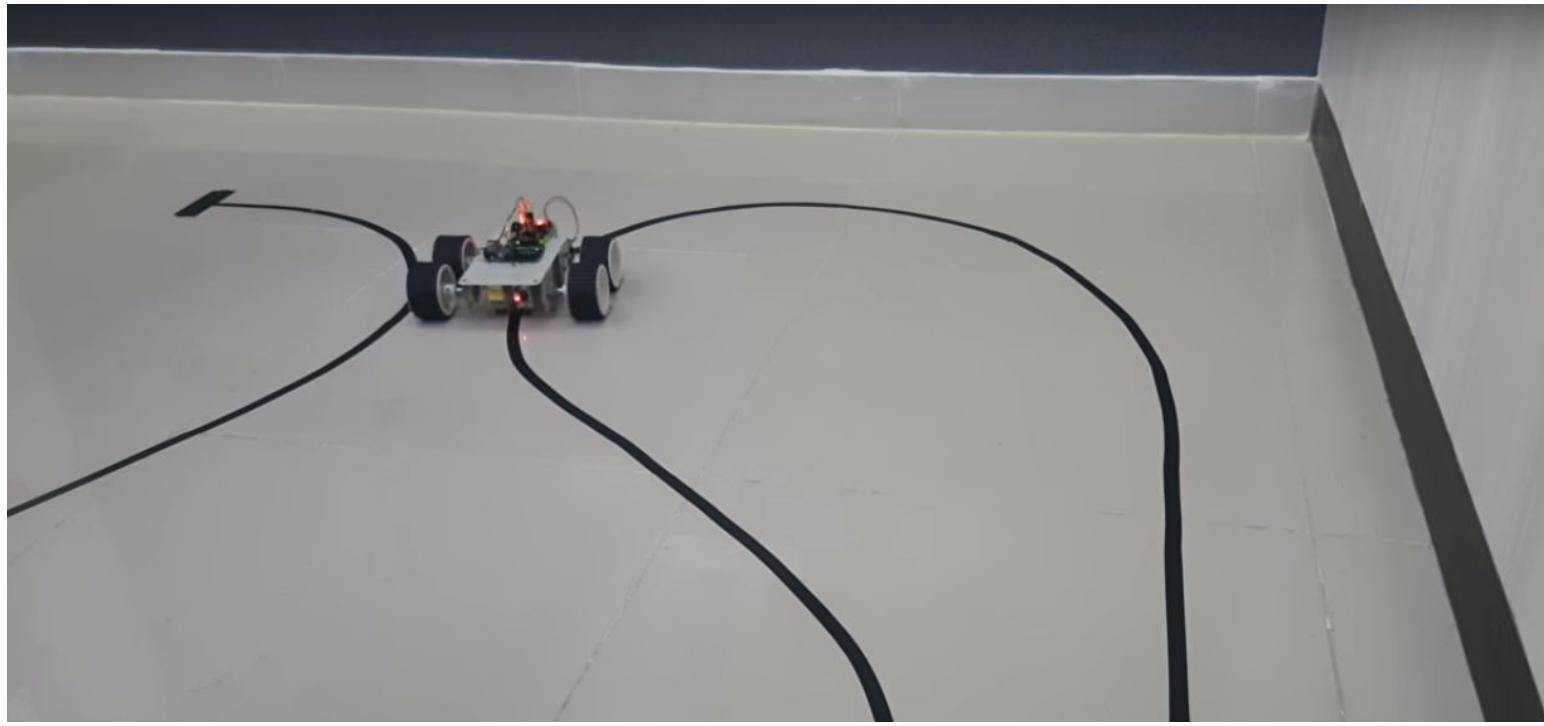
ROBOTICS CLUB  
SECOND SESSION

## FIRST THING FIRST:

1. Slack
2. Google group

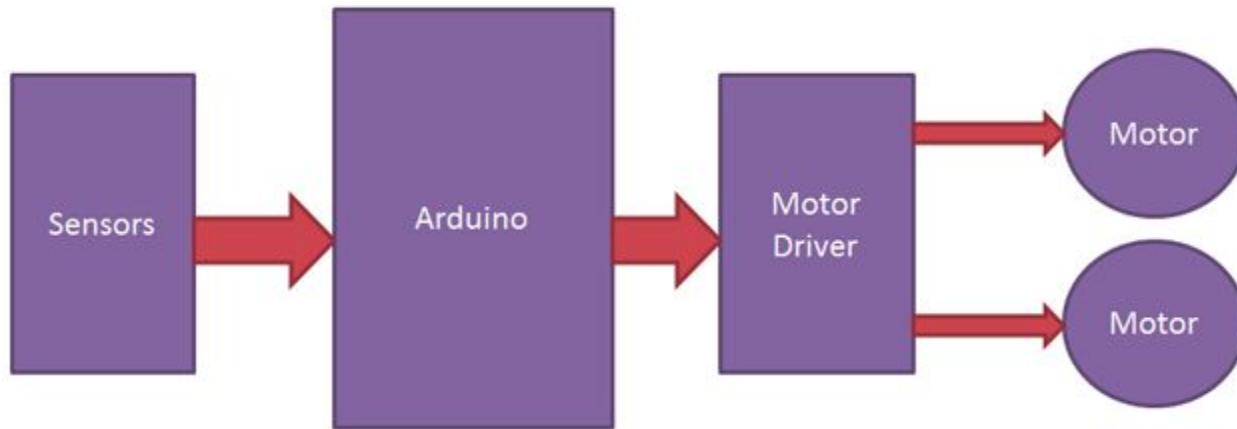
1. CONCEPT
2. CODE
3. HARDWARE
4. CIRCUIT

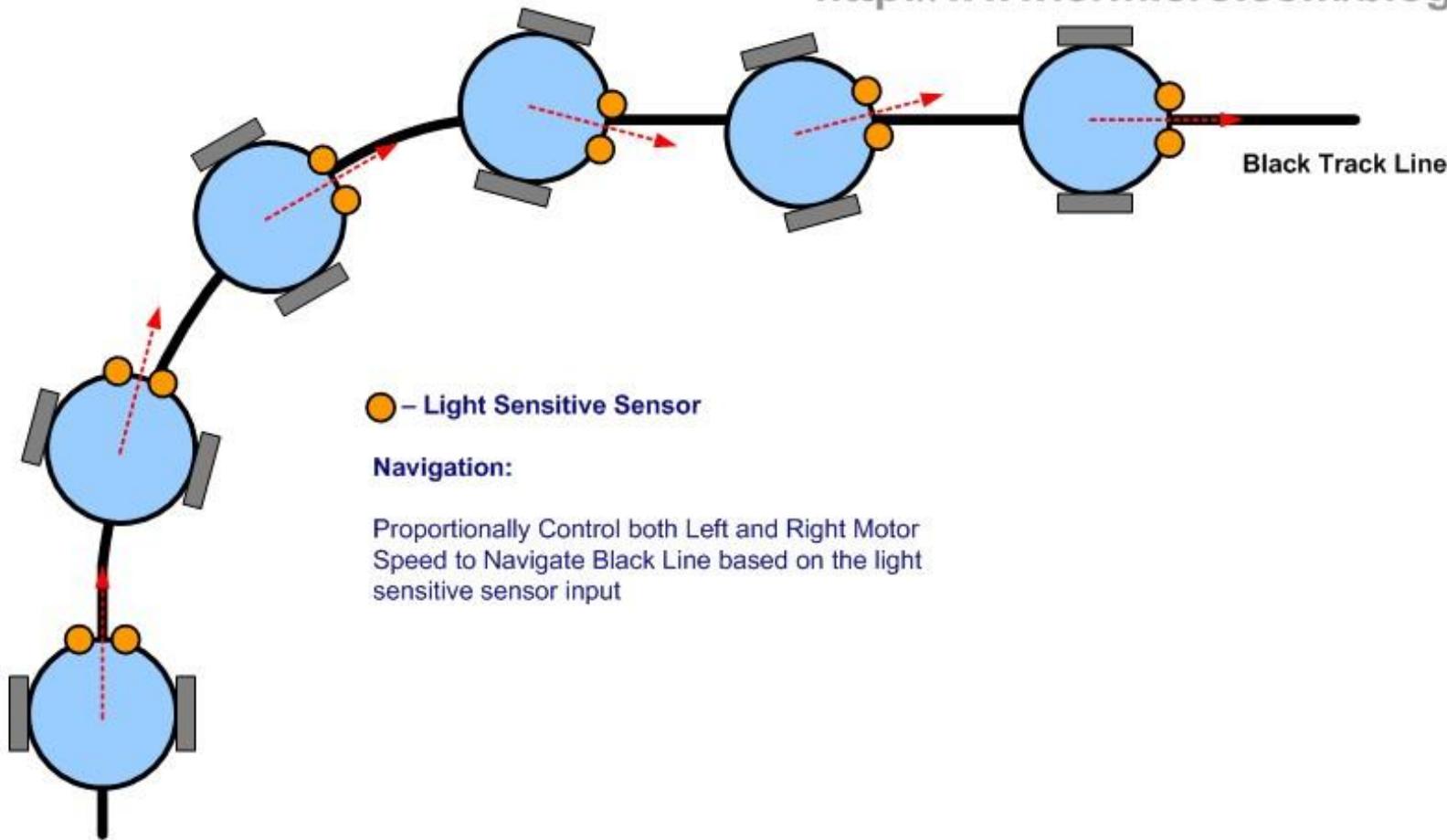
# 1. CONCEPT



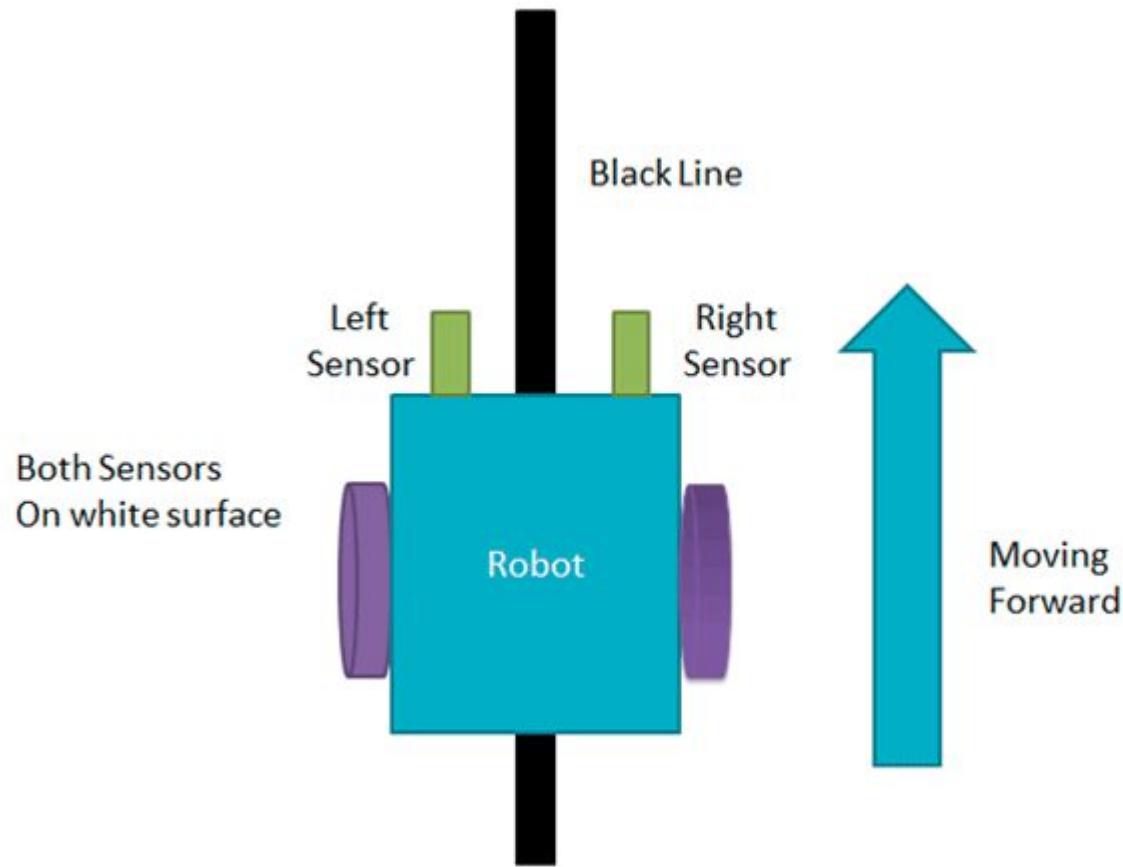
## Working of Line Follower Robot using Arduino

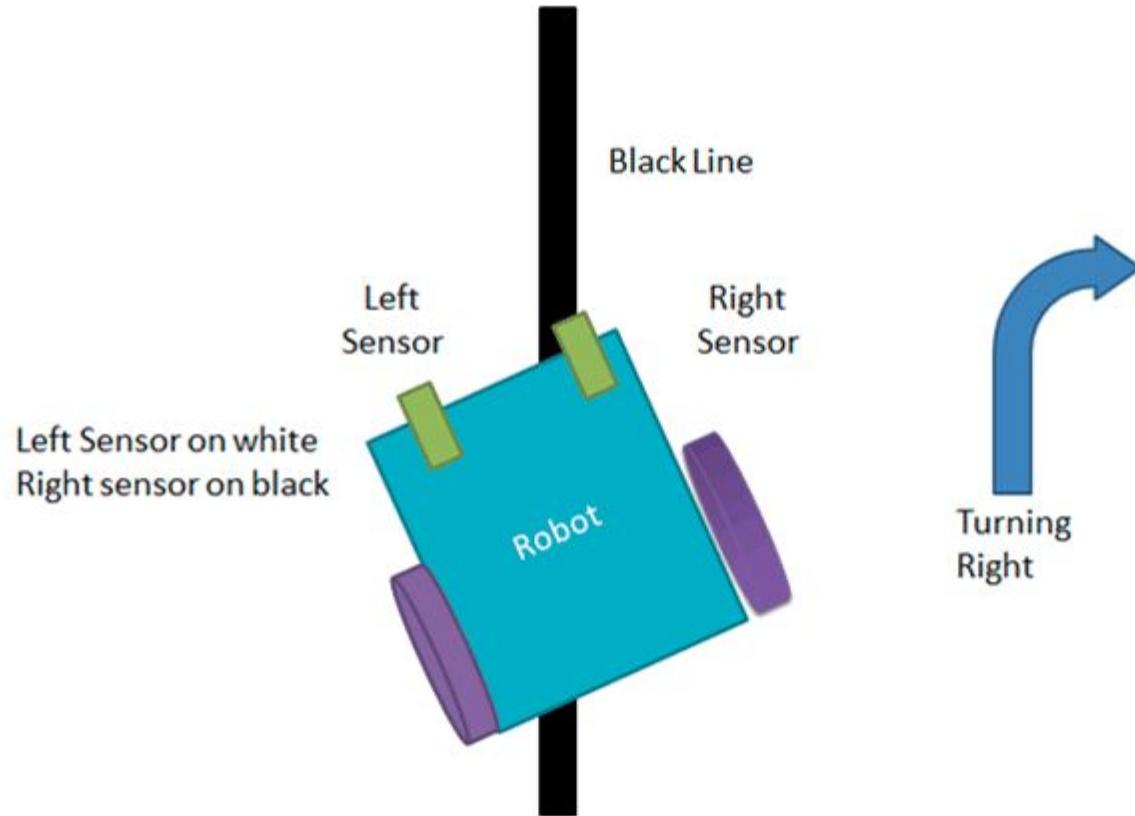
Working of line follower is very interesting. Line follower robot senses black line by using sensor and then sends the signal to arduino. Then arduino drives the motor according to sensors' output.

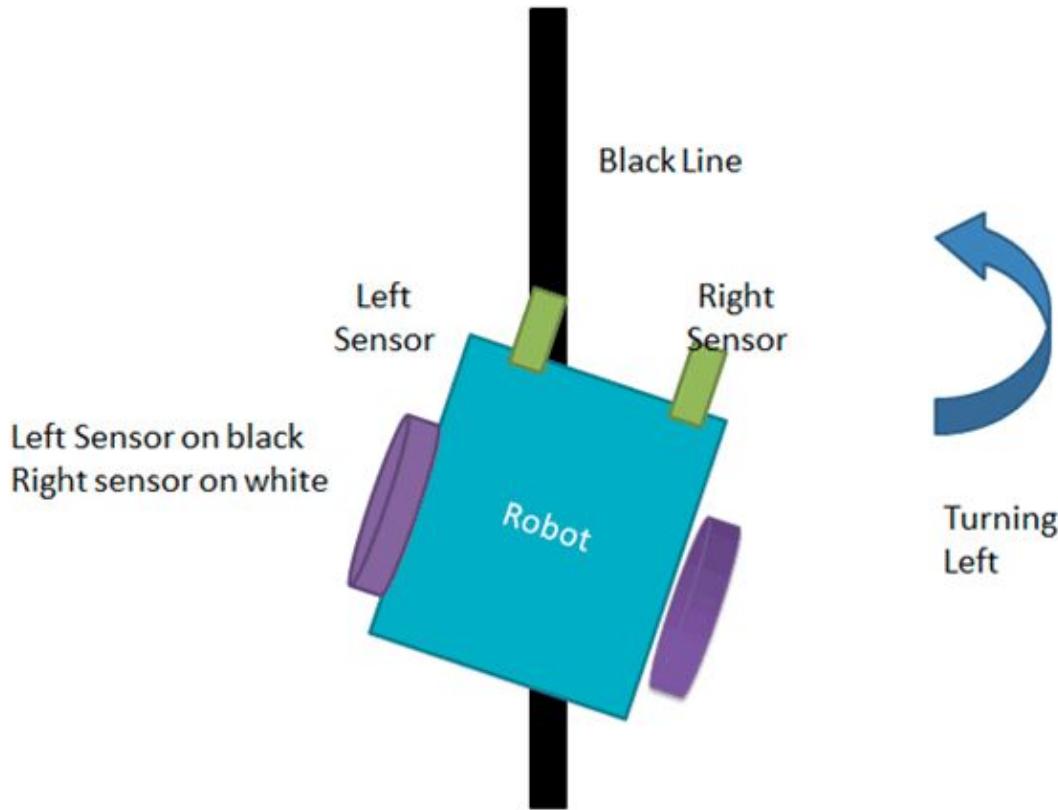


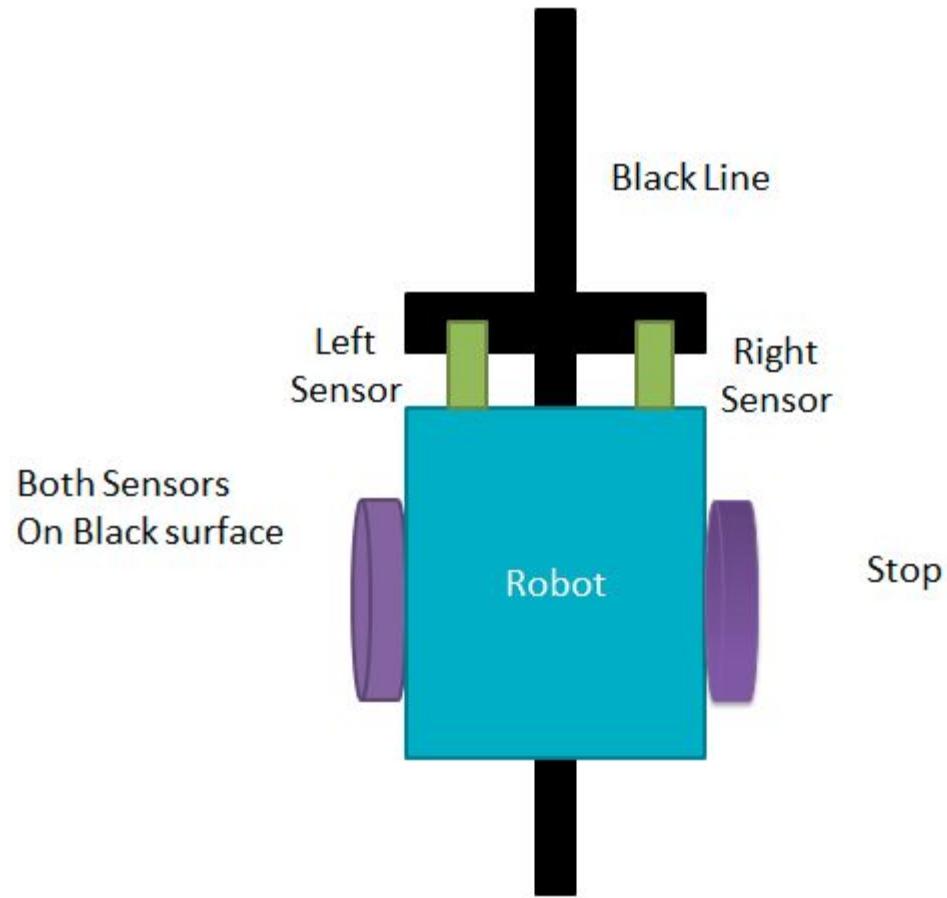


Line Tracking Navigation Principle on The Line Follower Robot (LFR)









# 2. CODE

## **2.1 QUICK LOOK ON PREVIOUS WEEK CODE**

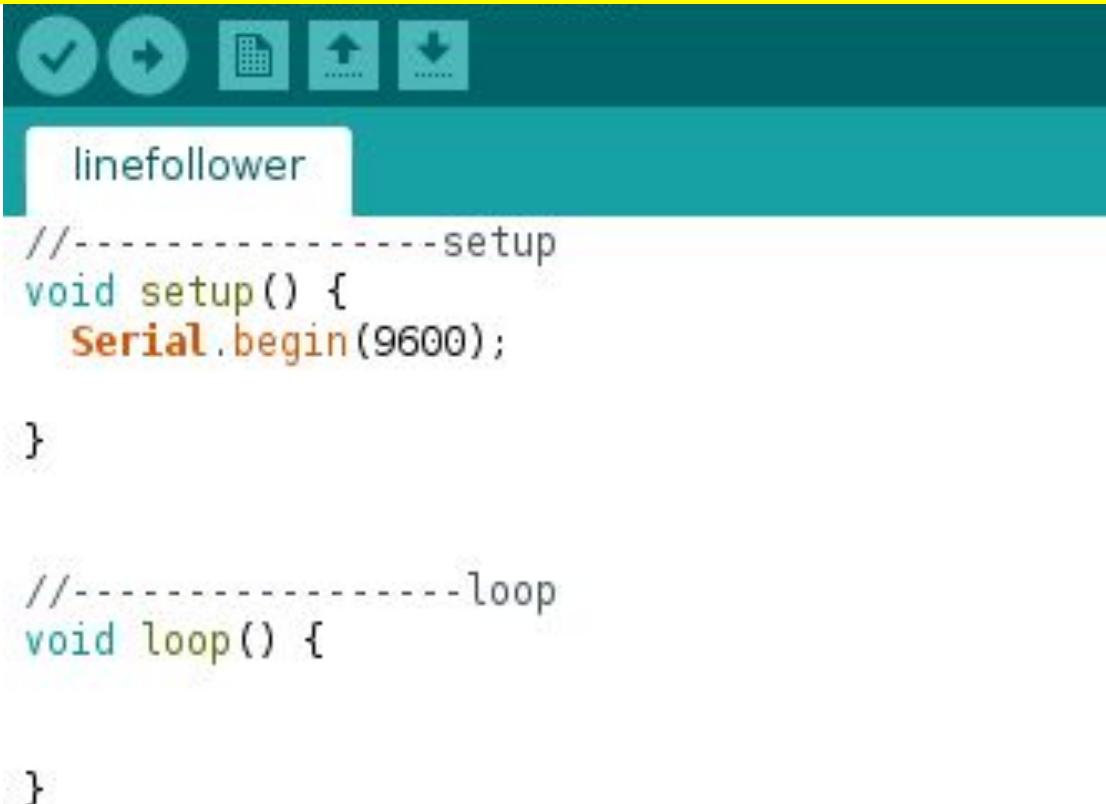
# BASIC SETUP AND LOOP



The screenshot shows the Arduino IDE interface. The title bar includes "Activities", "Places", and the "Arduino IDE" tab. The menu bar contains "File", "Edit", "Sketch", "Tools", and "Help". Below the menu is a toolbar with icons for checkmark, arrow, file, upload, and more. A project window displays the sketch name "linefollower" and its code:

```
void setup() {  
    // put your setup code here, to run once:  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
}
```

# Serial.begin(9600);



The image shows a screenshot of the Arduino IDE. At the top, there is a toolbar with five icons: a checkmark, a right arrow, a file folder, an upload symbol, and a download symbol. Below the toolbar, the sketch name "linefollower" is displayed in a teal header bar. The main code area contains the following code:

```
//-----setup
void setup() {
  Serial.begin(9600);
}

//-----loop
void loop() {
}
```

The line `Serial.begin(9600);` is highlighted in orange, indicating it is selected or being edited.

# GLOBAL DECLARATION

```
linefollower §  
//-----Global declaration  
int a=1;  
char b;
```

```
|  
//-----setup  
void setup() {  
    Serial.begin(9600);  
}
```

```
//-----loop  
void loop() {
```

```
}
```

# FUNCTION

```
//-----Global declaration
int a=1;
char b;

|
//-----function defination
void fun1(){
    Serial.println("fun1 function called");
    digitalWrite(a,1);
}

//-----setup
void setup() {
    Serial.begin(9600);
    pinMode(a, OUTPUT);

}

//-----loop
void loop() {
    fun1(); //calling function fun1
}
```

## **2.2 LINE FOLLOWER CODE**

# GLOBAL DECLARATION

linefollower §

```
-----Global declaration
//pins
int rm_pin=1;//right motor
int lm_pin=2;//left motor
int rs_pin=3;//right sensor
int ls_pin=4;//left sensor
//values
int rm_value;
int lm_value;
int rs_value;
int ls_value;
```

```
-----function defination
```

## pinMode(pin, OUTPUT);

```
//-----setup
void setup() {
    Serial.begin(9600);
    pinMode(rm_pin, OUTPUT); //output
    pinMode(lm_pin, OUTPUT);
    pinMode(rs_pin, INPUT); //input
    pinMode(ls_pin, INPUT);
}

//-----loop
void loop() {

}
```

## read\_sen\_value(); ←FUNCTION

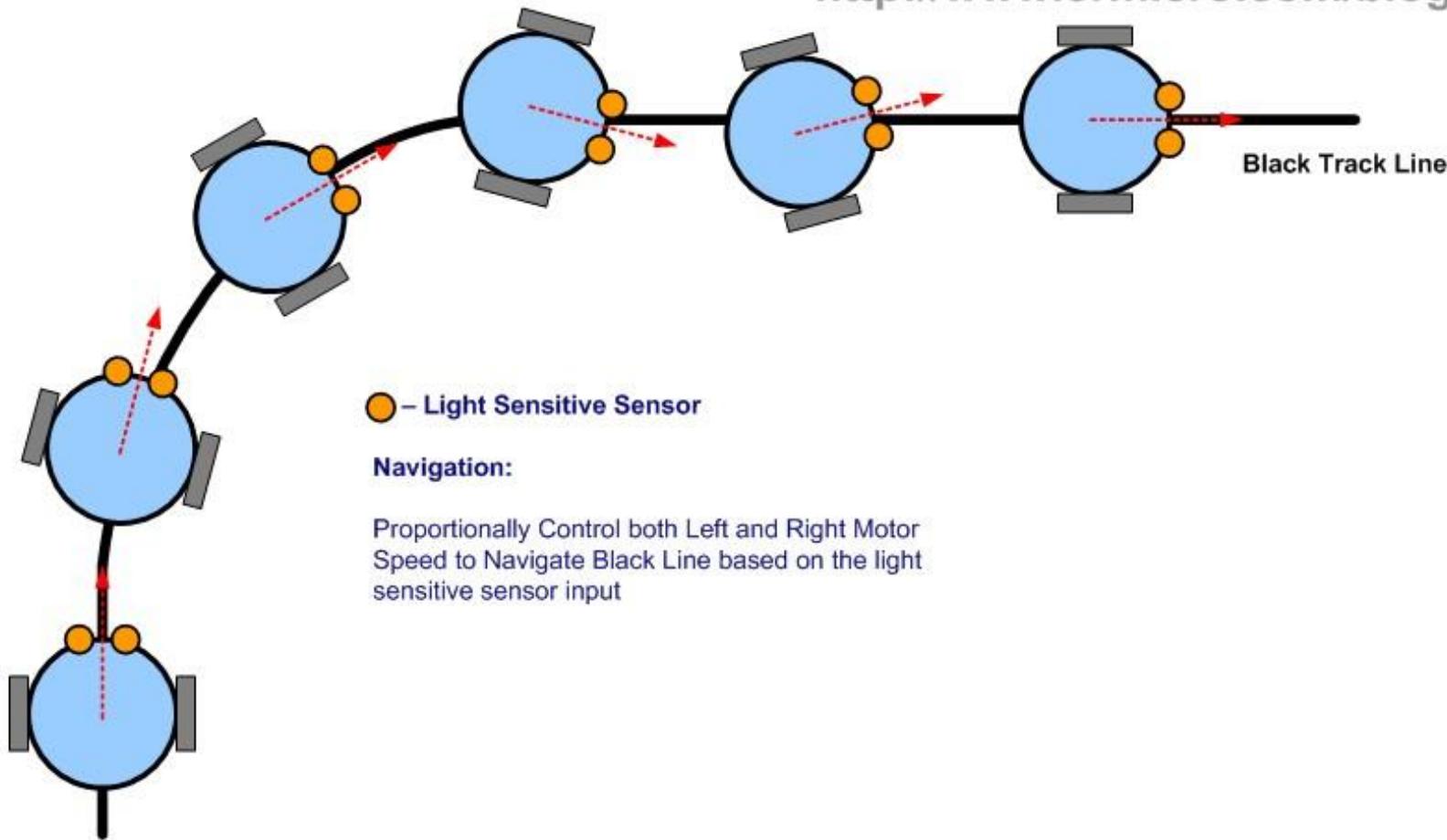
```
//-----function defination
void read_sen_value()
{
    rs_value = digitalRead(rs_pin);
    ls_value = digitalRead(ls_pin);
}

//-----setup
void setup() {
    Serial.begin(9600);
    pinMode(rm_pin, OUTPUT); //output
    pinMode(lm_pin, OUTPUT);
    pinMode(rs_pin, INPUT); //input
    pinMode(ls_pin, INPUT);
}

//-----loop
void loop() {
    read_sen_value();
}
```

**check\_direction\_move(); ← CALL**

```
//-----loop-----//  
void loop() {  
    read_sen_value();  
  
    check_direction_move();  
}
```



Line Tracking Navigation Principle on The Line Follower Robot (LFR)

## check\_direction\_move(); ← DEFINE

```
void check_direction_move()
{
    if(ls_value == 0 && rs_value == 0) //WW
    {
        forward(); //forward
    }
    else if(ls_value == 0 && rs_value == 1)//WB
    {
        right(); //right
    }
    else if(ls_value == 1 && rs_value == 0)//BW
    {
        left(); //left
    }
    else if(ls_value == 1 && rs_value == 1)//BB
    {
        STOP(); //STOP
    }
}
```

**check\_direction\_move(); ← DEFINE**

```
void forward()
{
    digitalWrite(lm_pin, 1); //both high
    digitalWrite(rm_pin, 1);
}
```

## SIMILARLY OTHER FUNCTION

```
void forward()
{
    digitalWrite(lm_pin, 1); //both high
    digitalWrite(rm_pin, 1);
}
void left()
{
    digitalWrite(lm_pin, 1); //left high
    digitalWrite(rm_pin, 0);
}
void right()
{
    digitalWrite(lm_pin, 0); //right high
    digitalWrite(rm_pin, 1);
}
void STOP()
{
    digitalWrite(lm_pin, 0); //both low
    digitalWrite(rm_pin, 0);
}
```

# FULL CODE

```
-----Global declaration-----
//pins
int rm_pin=1;//right motor
int lm_pin=2;//left motor
int rs_pin=3;//right sensor
int ls_pin=4;//left sensor
//values
int rm_value;
int lm_value;
int rs_value;
int ls_value;

-----function defination-----
void read_sen_value()
{
    rs_value = digitalRead(rs_pin);
    ls_value = digitalRead(ls_pin);
}

void check_direction_move()
{
    if(ls_value == 0 && rs_value == 0) //WW
    {
        forward();//forward
    }
    else if(ls_value == 0 && rs_value == 1)//WB
    {
        right();//right
    }
    else if(ls_value == 1 && rs_value == 0)//BW
    {
        left();//left
    }
    else if(ls_value == 1 && rs_value == 1)//BB
    {
        STOP();//STOP
    }
}

void forward()
{
    digitalWrite(lm_pin, 1); //both high
    digitalWrite(rm_pin, 1);
}

void left()
{
    digitalWrite(lm_pin, 1); //left high
    digitalWrite(rm_pin, 0);
}

void right()
{
    digitalWrite(lm_pin, 0); //right high
    digitalWrite(rm_pin, 1);
}

void STOP()
{
    digitalWrite(lm_pin, 0);//both low
    digitalWrite(rm_pin, 0);
}

-----setup-----
void setup() {
    Serial.begin(9600);
    pinMode(rm_pin, OUTPUT);//output
    pinMode(lm_pin, OUTPUT);
    pinMode(rs_pin, INPUT);//input
    pinMode(ls_pin, INPUT);
}

-----loop-----
void loop() {
    read_sen_value();
    check_direction_move();
}
```

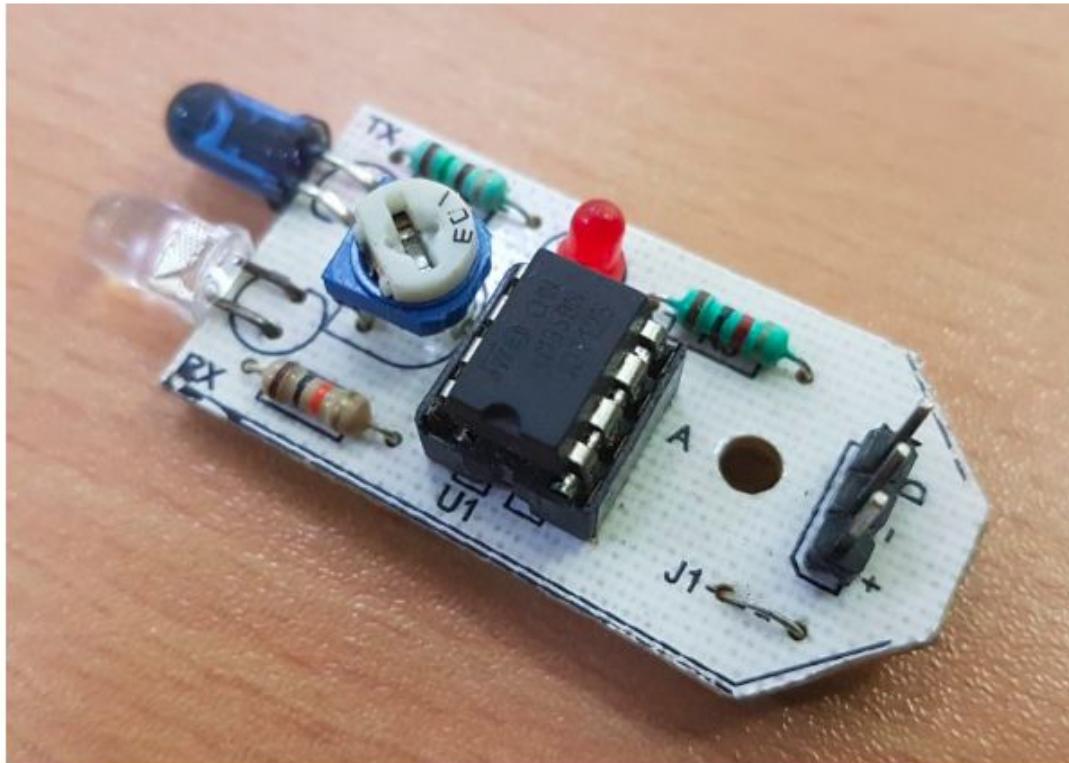
# **3. HARDWARE**

## **Components Required**

- Arduino UNO (or Arduino Nano)
- L293D Motor Driver IC
- Geared Motors x 2
- Robot Chassis
- IR Sensor Module x 2
- Black Tape (Electrical Insulation Tape)
- Connecting Wires
- Power supply

# IR SENSOR

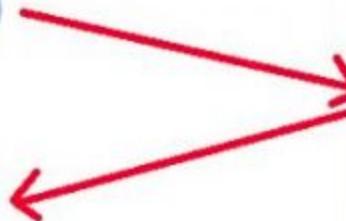
**Sensors (IR Sensor):** We have used IR Sensor Module as the line detecting sensor for the project. It consists of an IR LED and a Photo diode and some other components like comparator, LED etc.



IR LED



Light Coloured Surface



Photodiode

IR LED



Black Surface

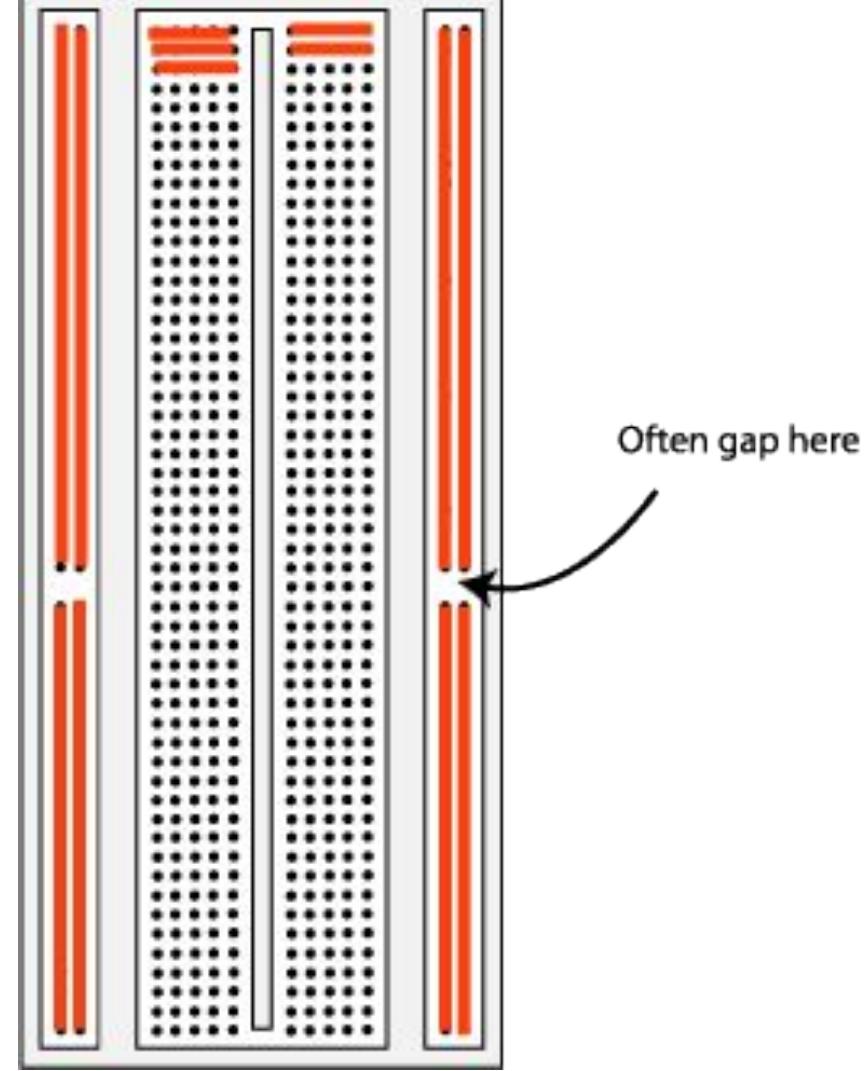


# DC MOTOR

**Motors (Geared Motors):** We have used two geared motors at the rear of the line follower robot. These motors provide more torque than normal motors and can be used for carrying some load as well.



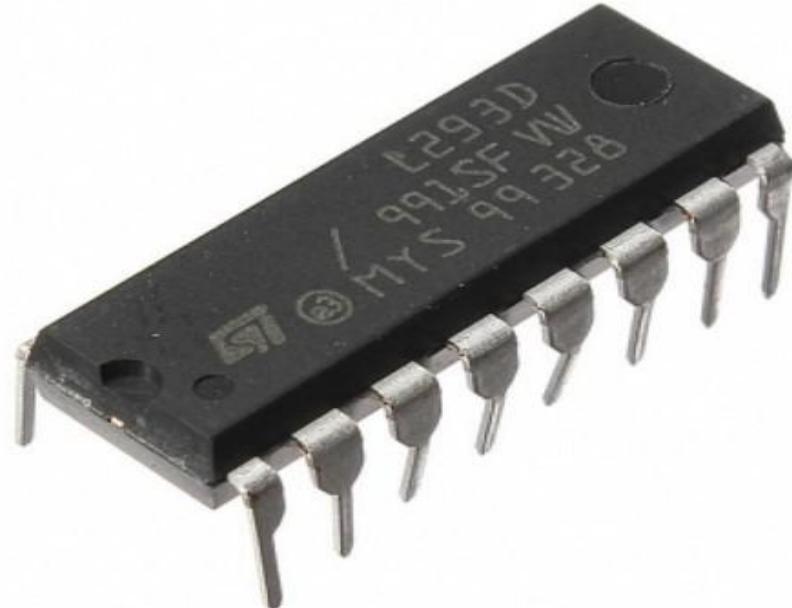
# BREADBOARD

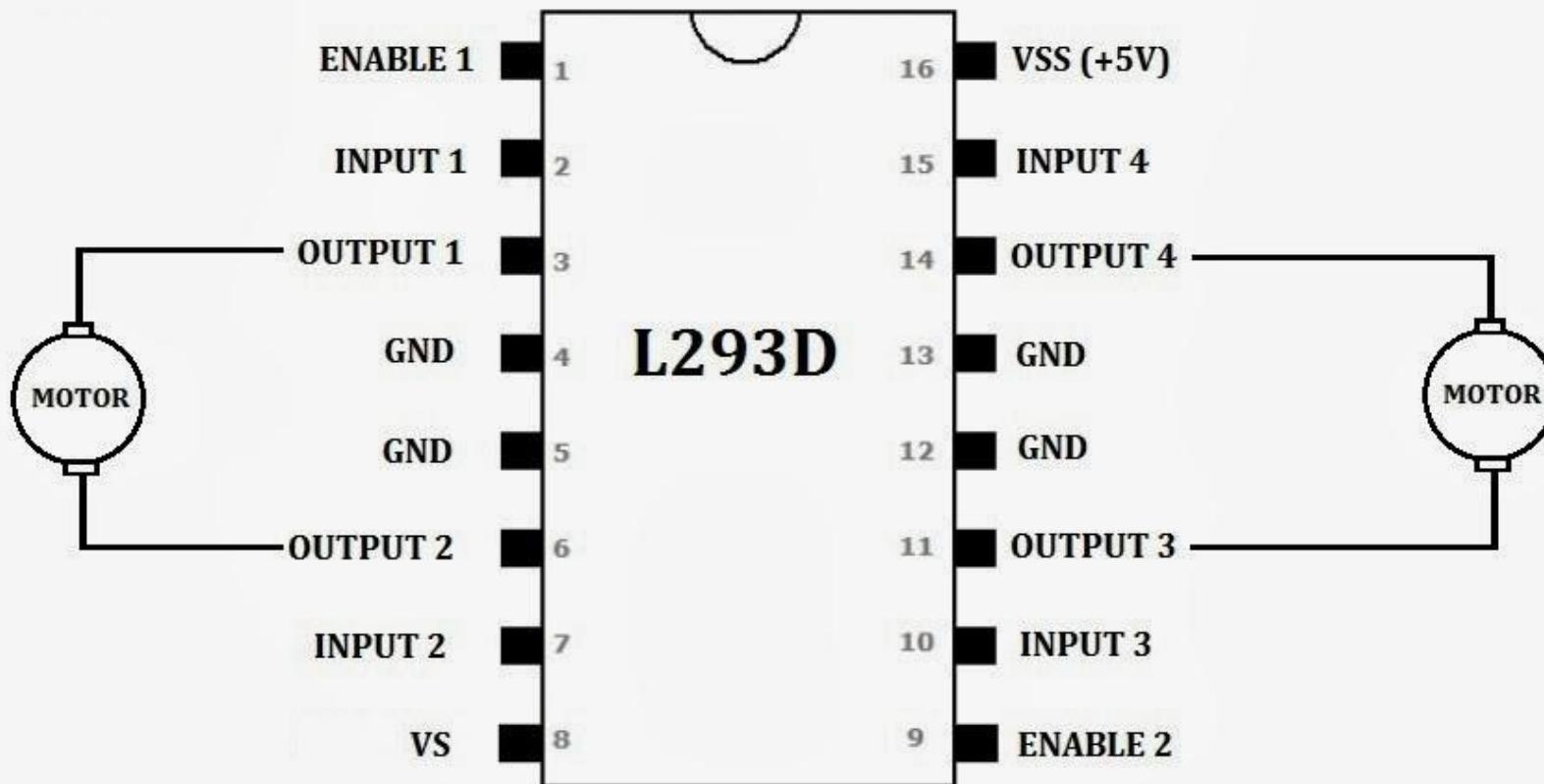


# MOTOR DRIVER (L293D)

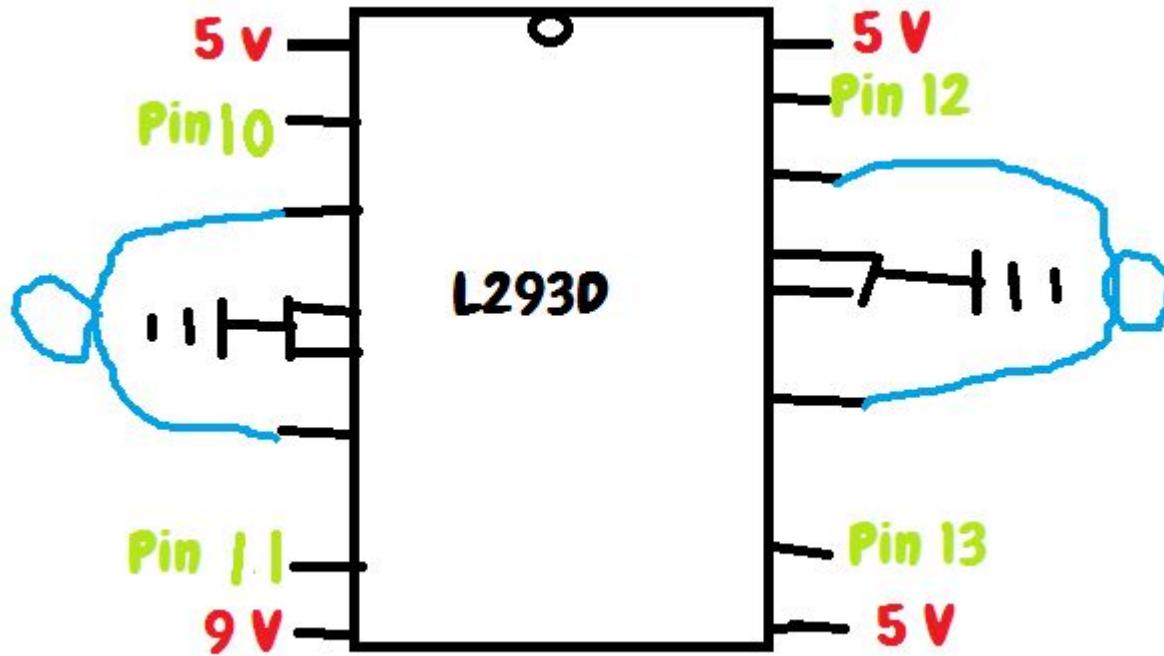
## L293D Motor Driver

L293D is a motor driver IC which has two channels for driving two motors. L293D has two inbuilt Transistor Darlington pair for current amplification and a separate power supply pin for giving external supply to motors.

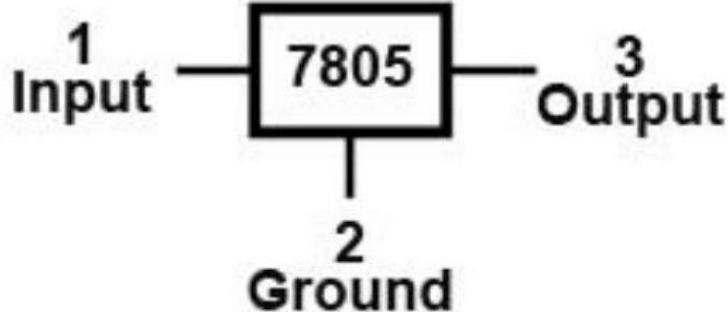
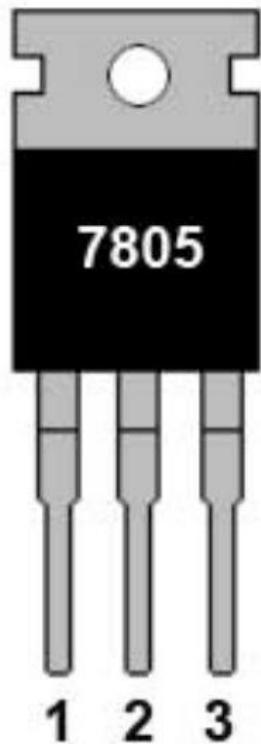




Dual DC Motor Controller

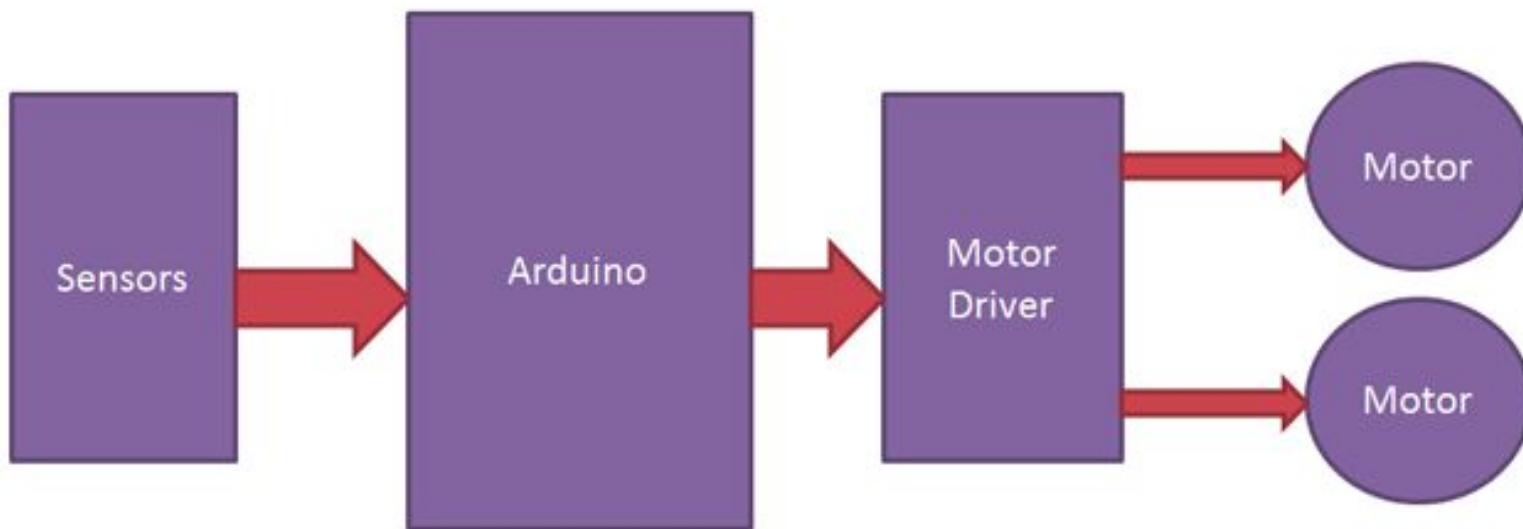


# VOLTAGE REGULATOR (7805)

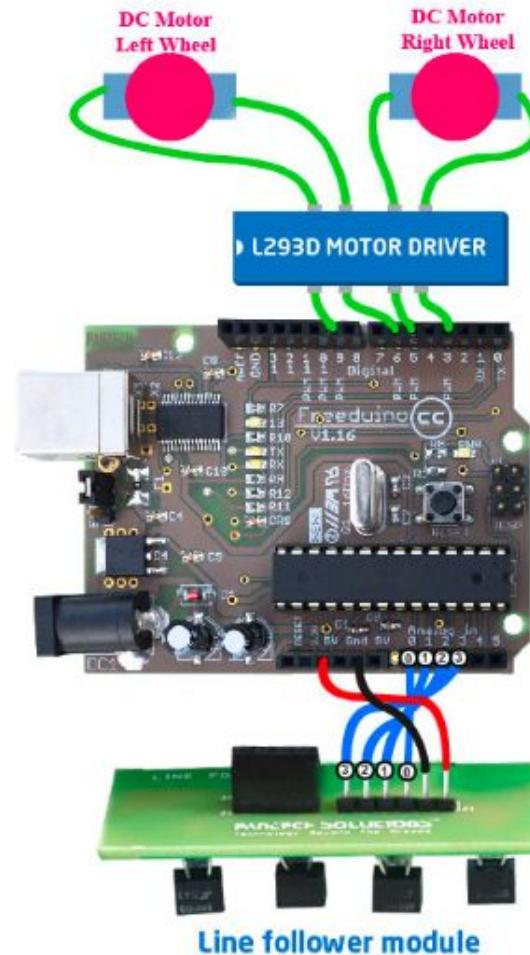


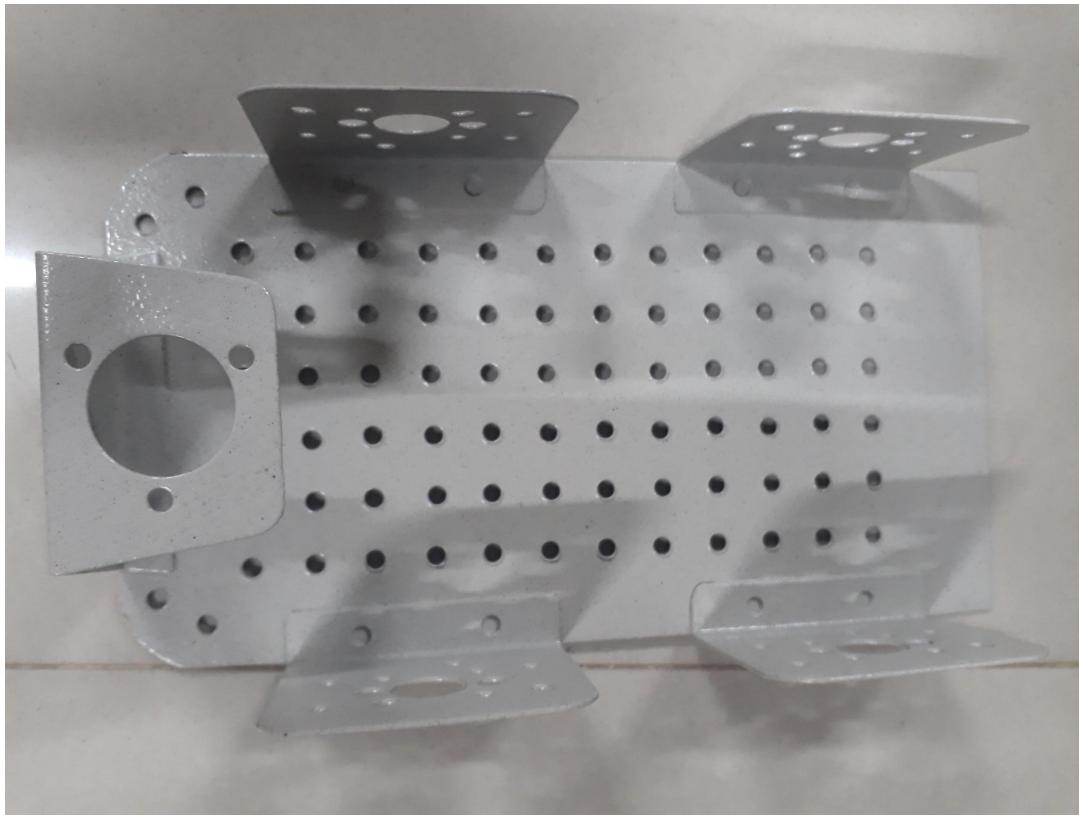
# **4. CIRCUIT**

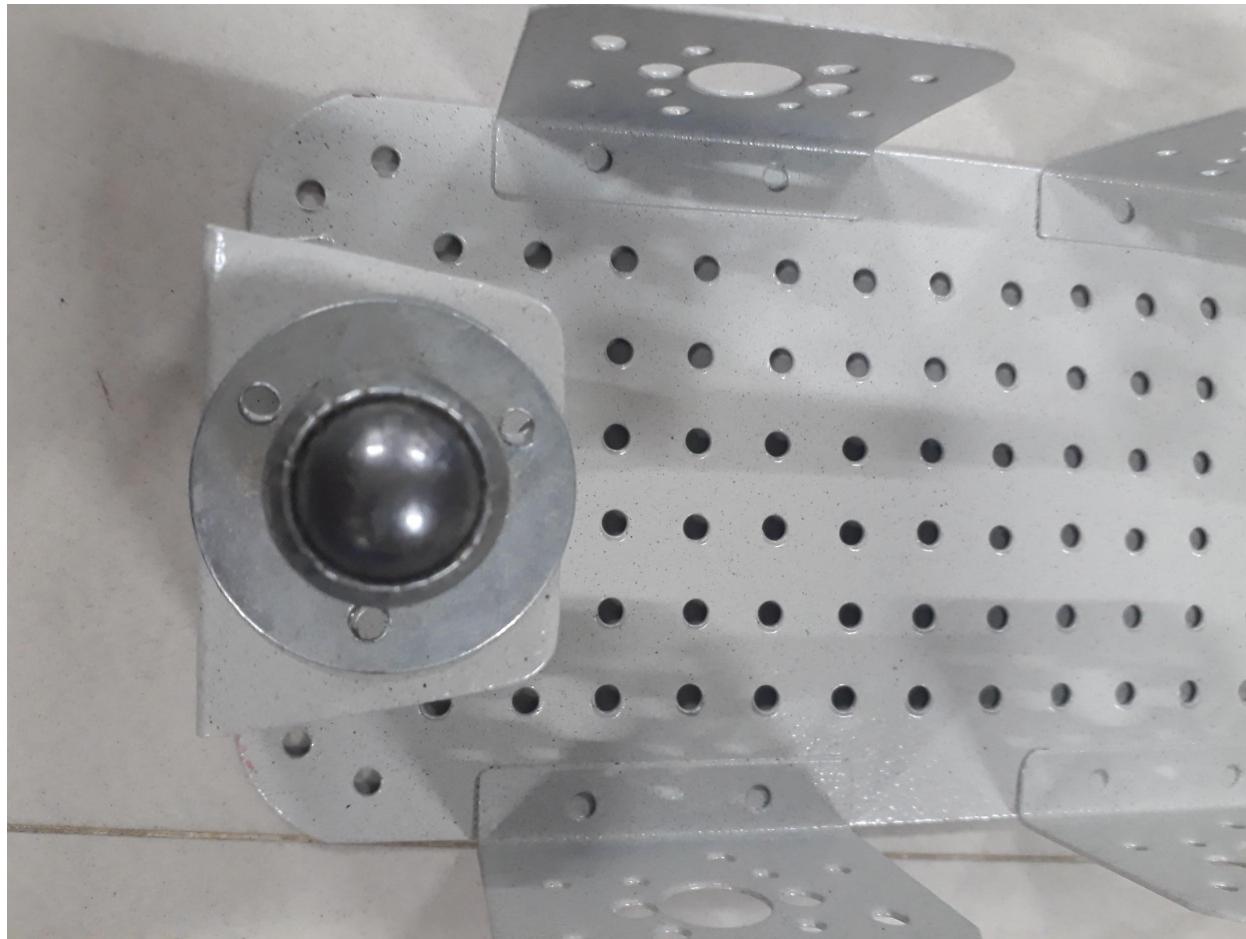
**NEVER CONNECT ARDUINO OR  
SENSORS TO > 5V**

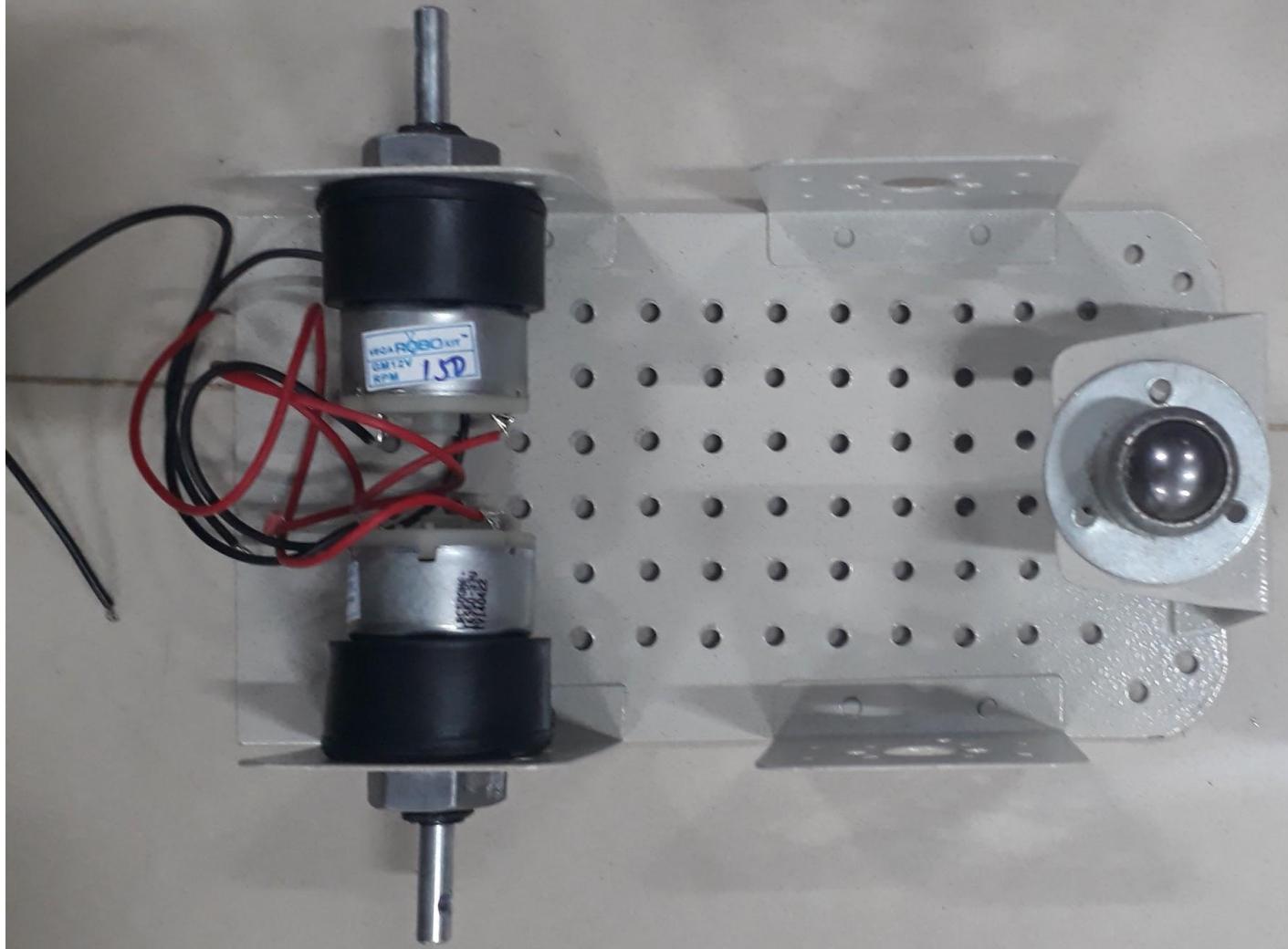


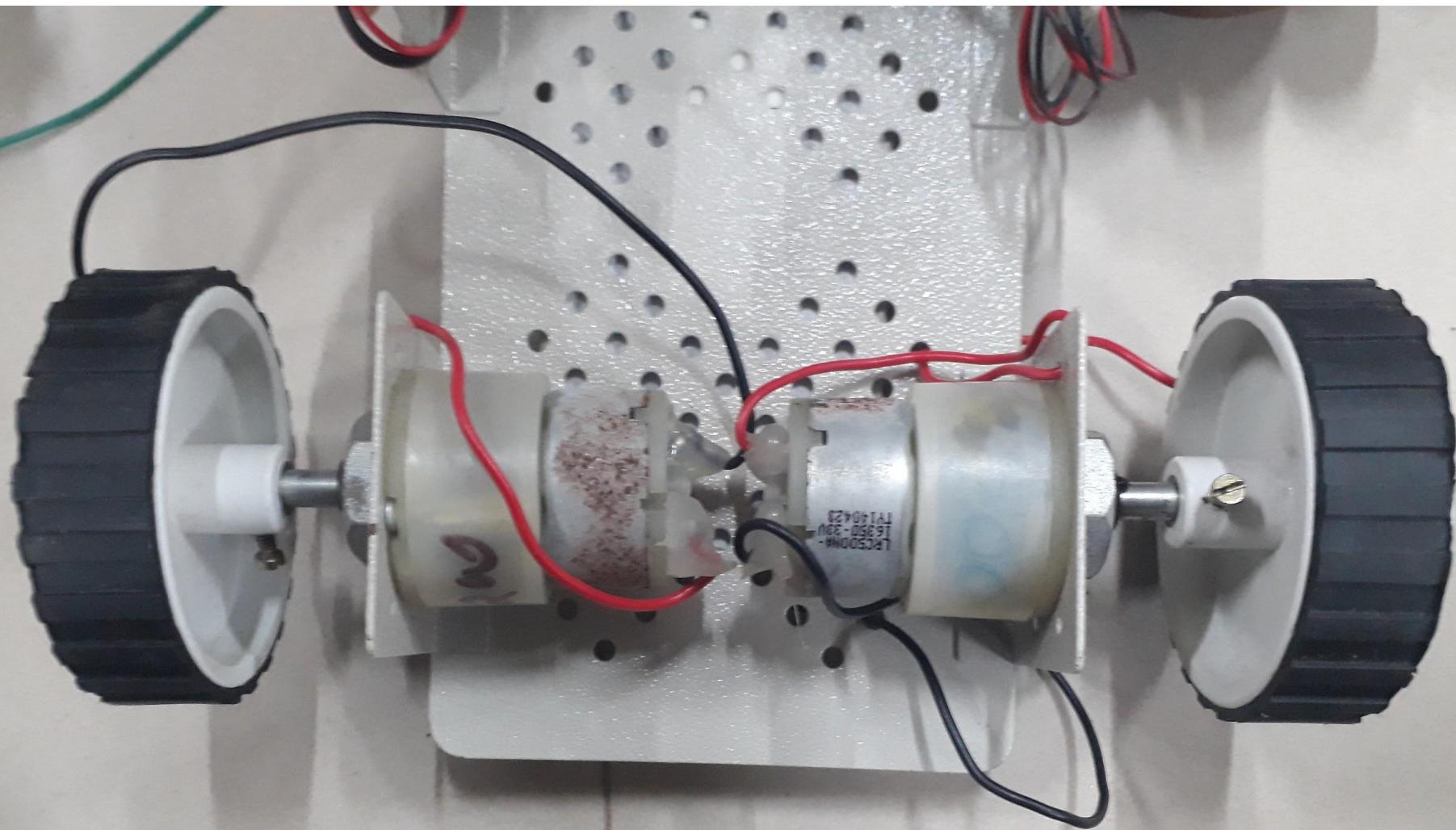
## Connection Circuit diagram with Arduino and Line follower sensor



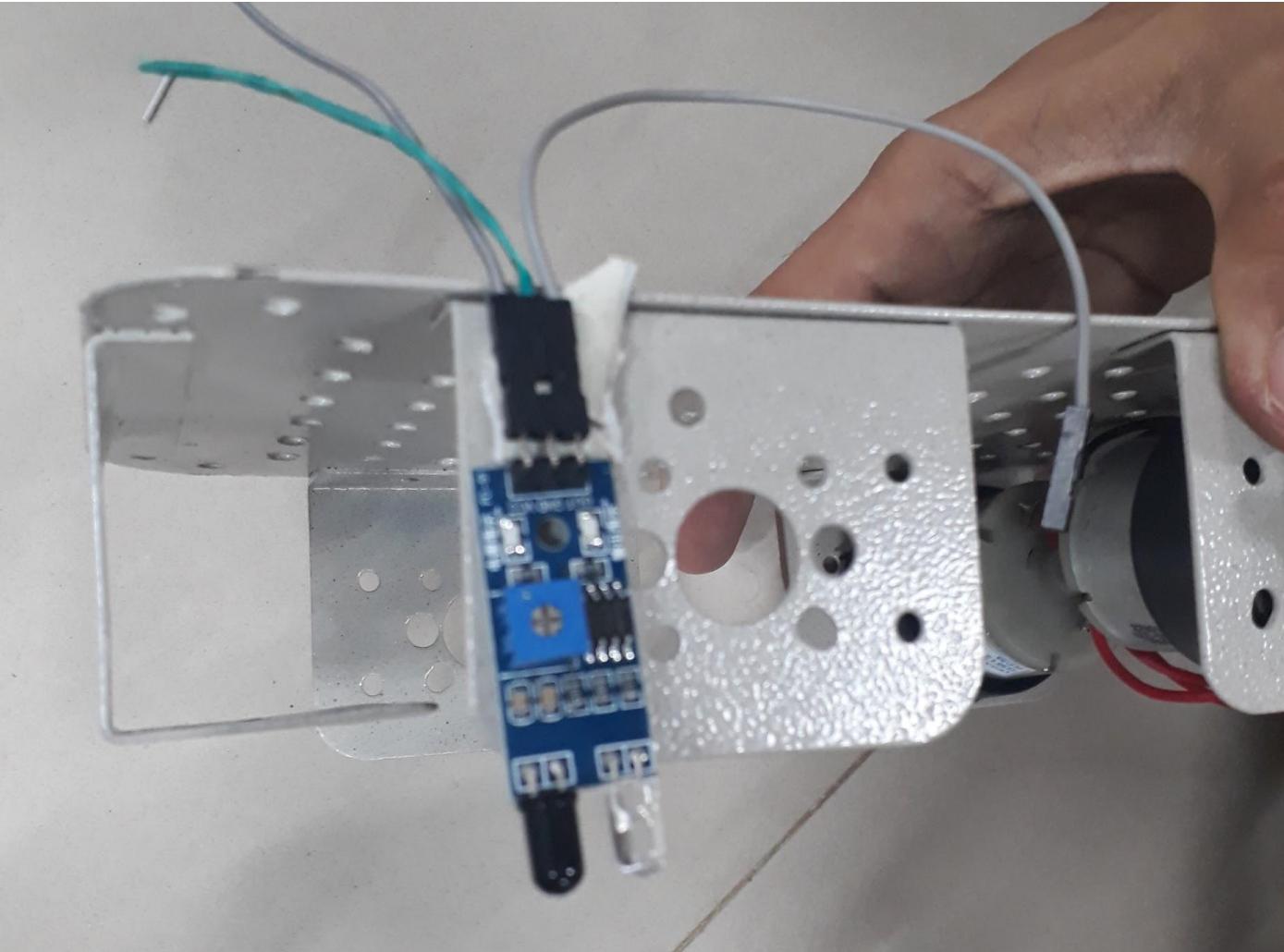




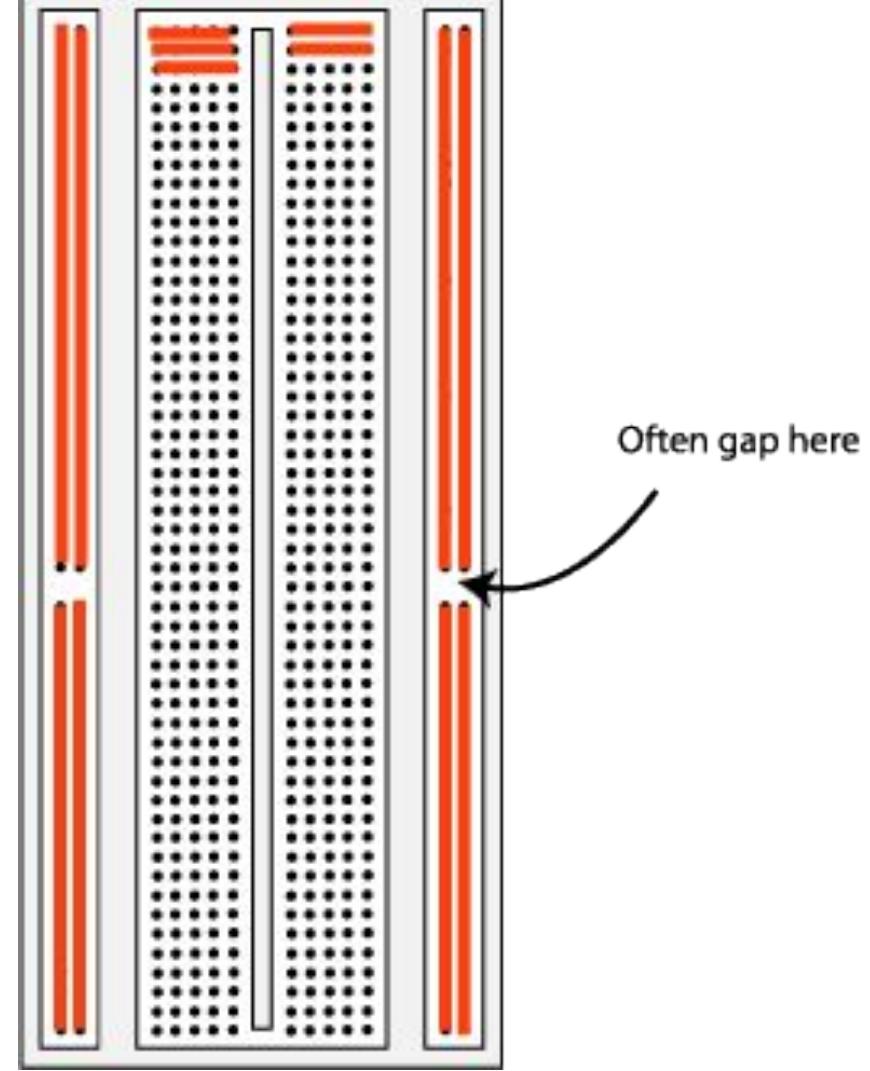


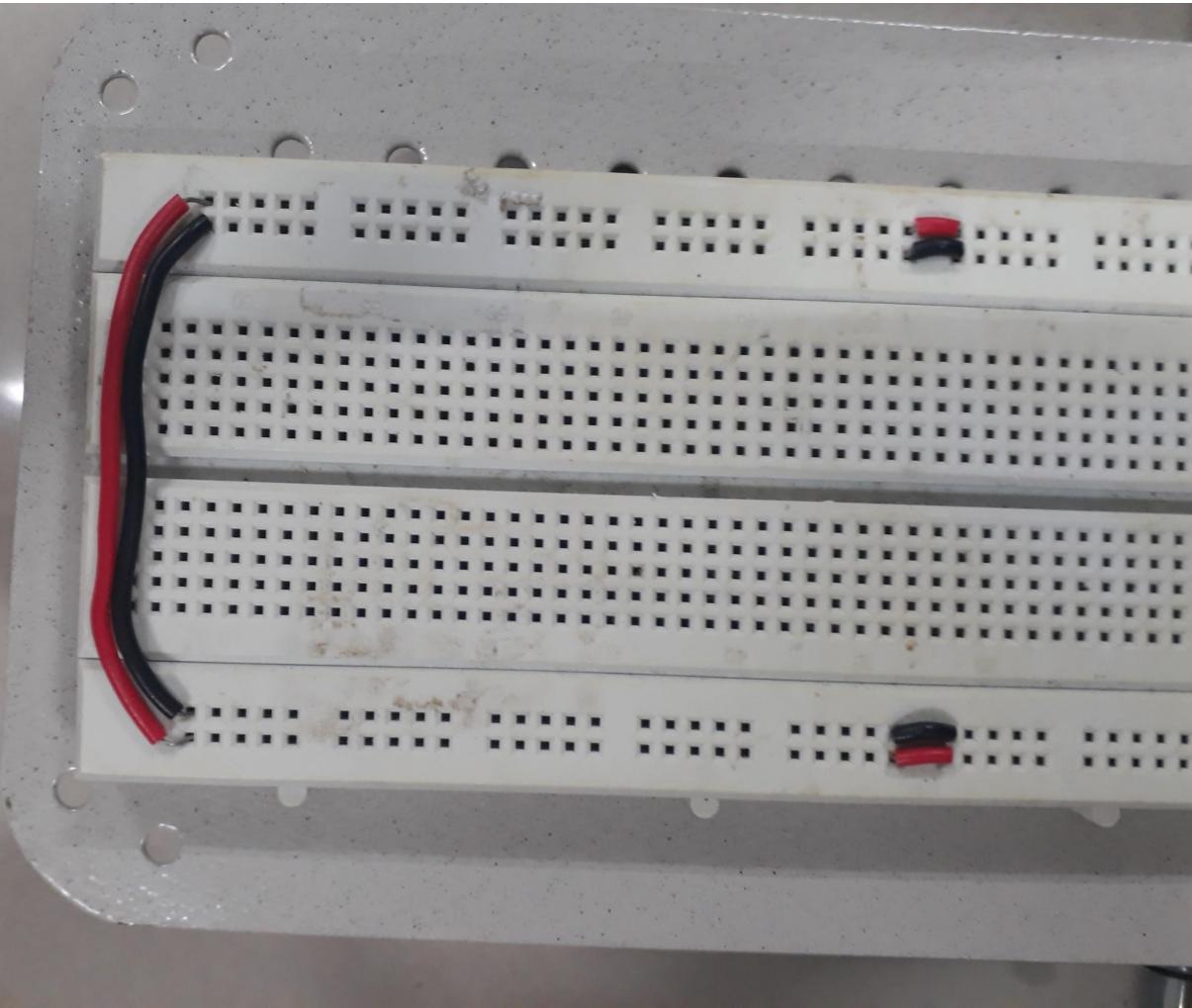


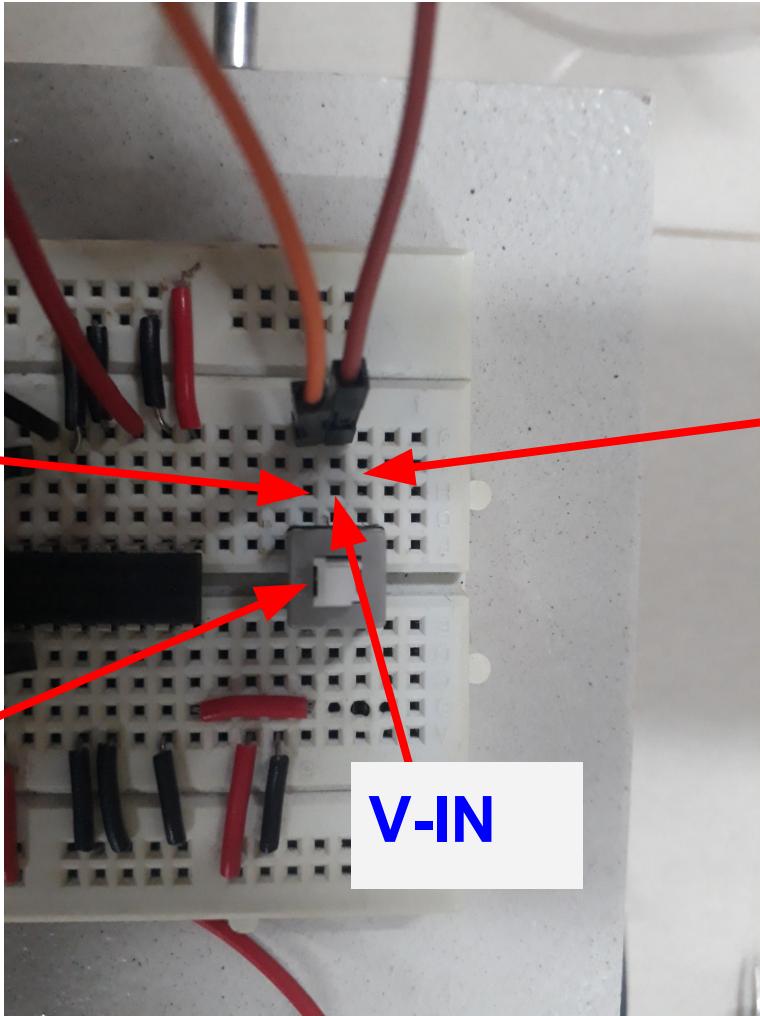




# BREADBOARD







V-OUT

RECTANGLE

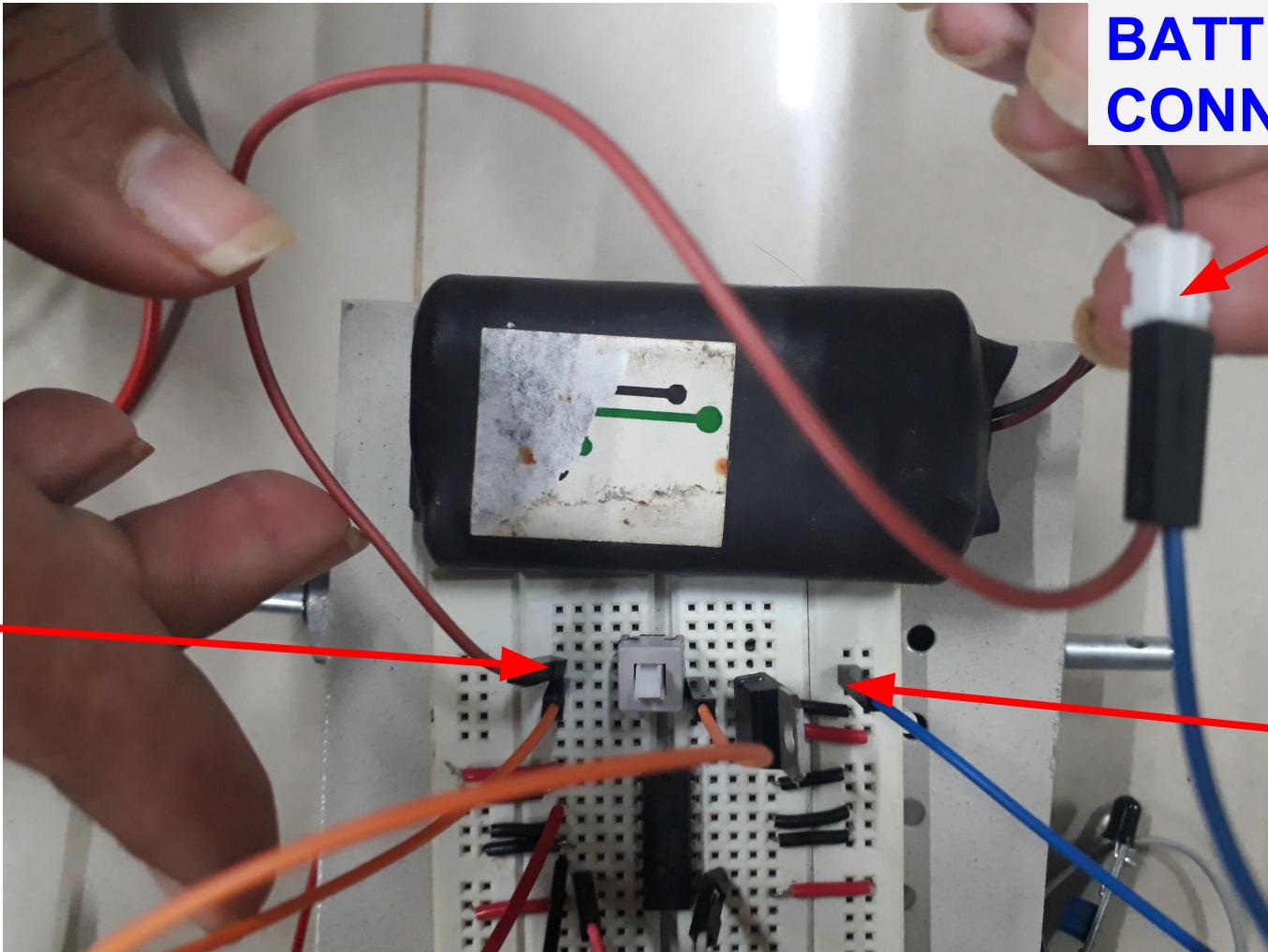
V-IN

ALWAYS ON

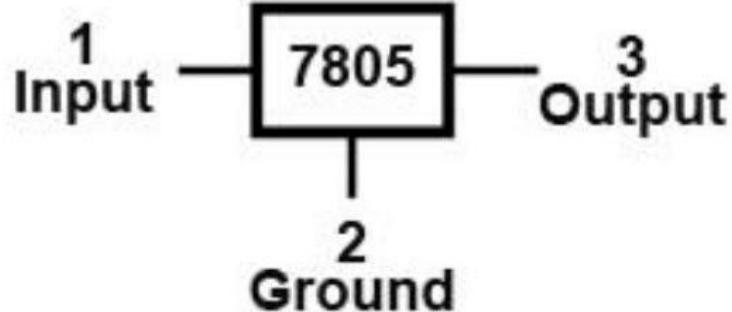
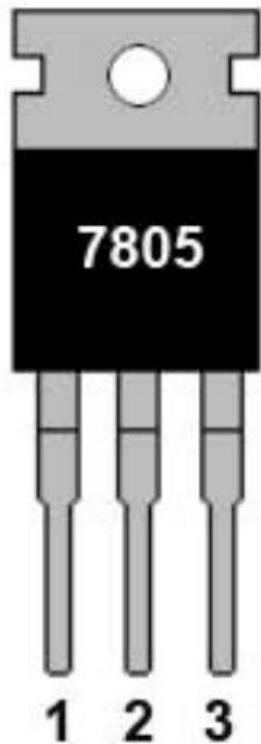
**BATTERY  
CONNECTION**

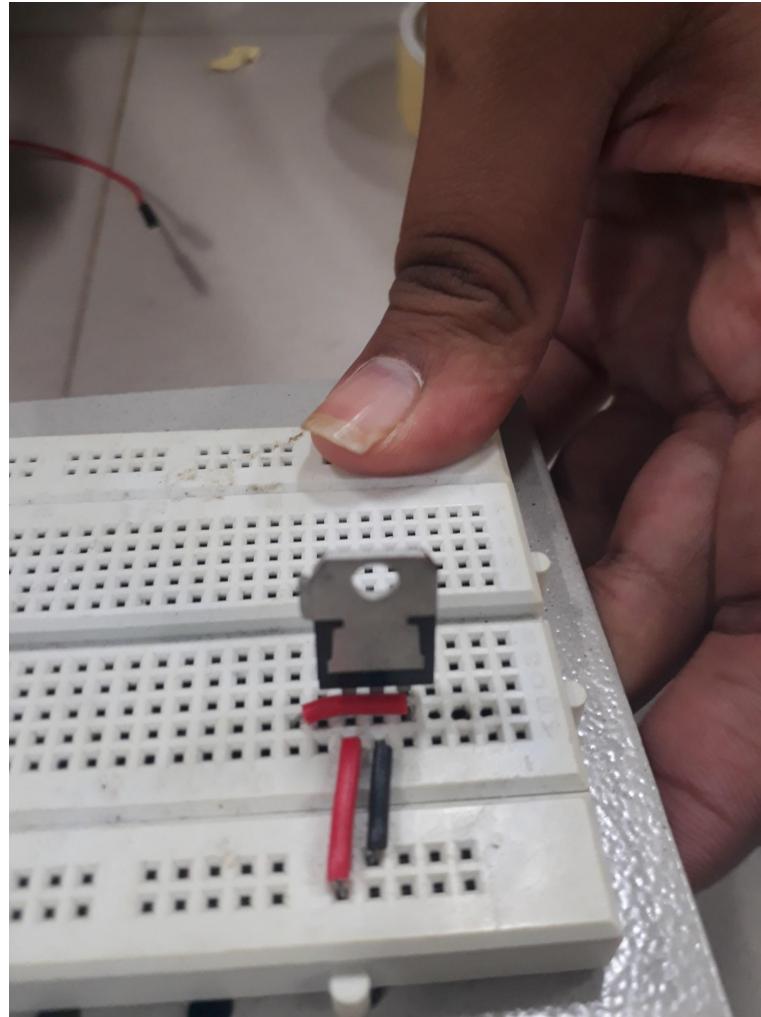
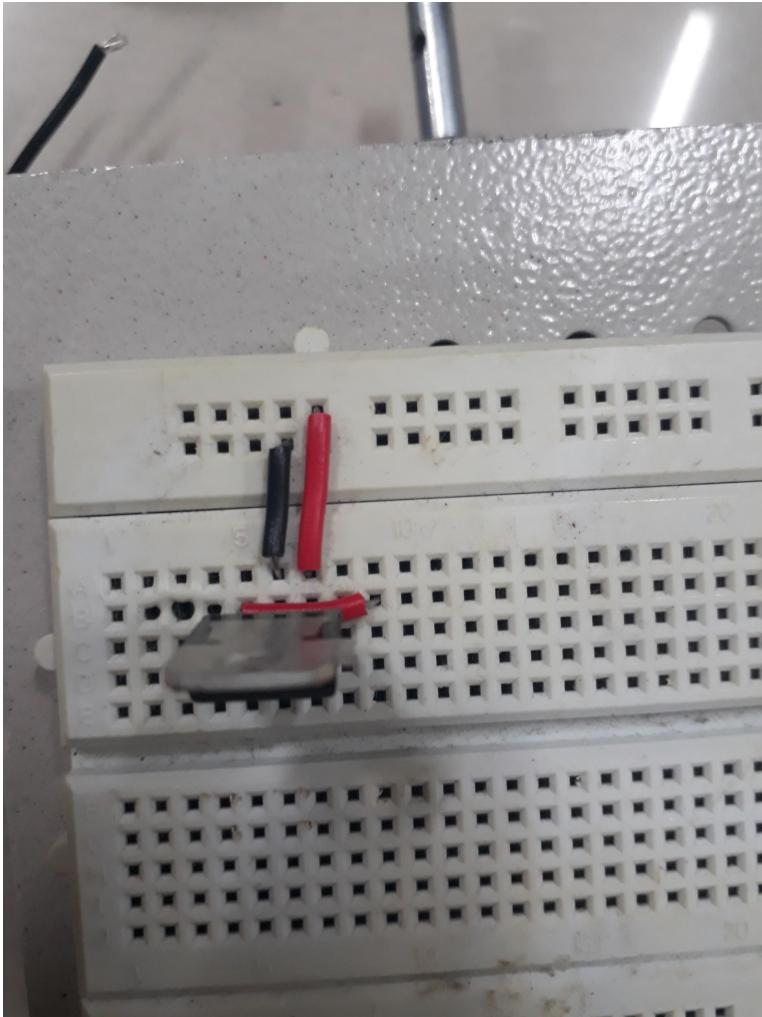
**V-IN**

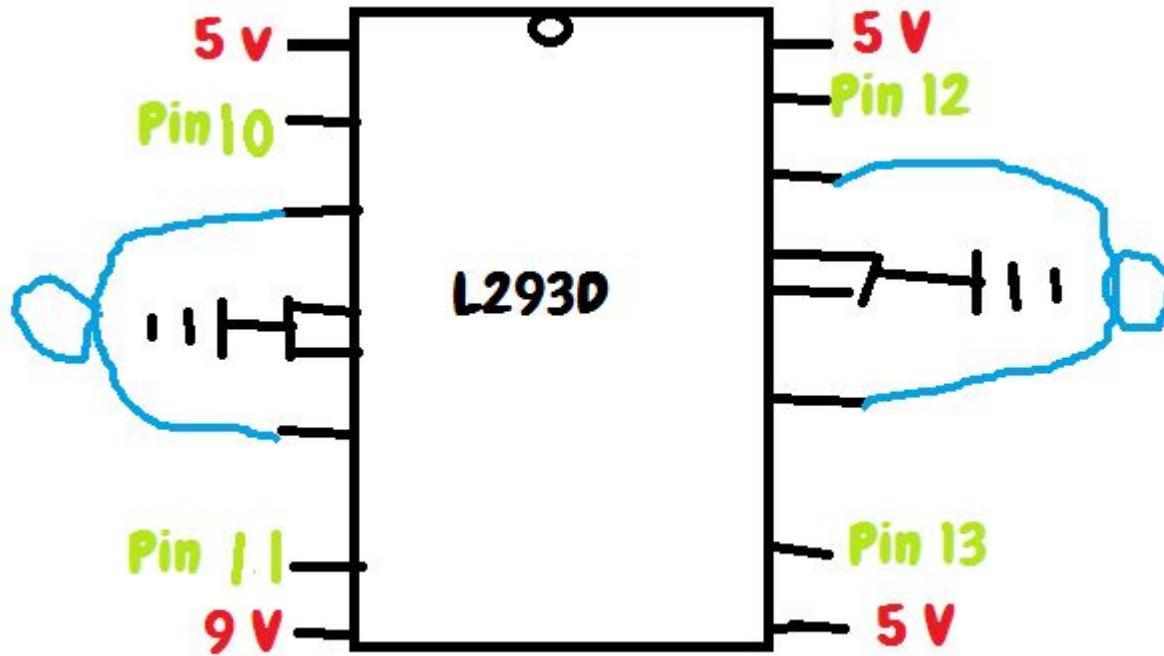
**GND**

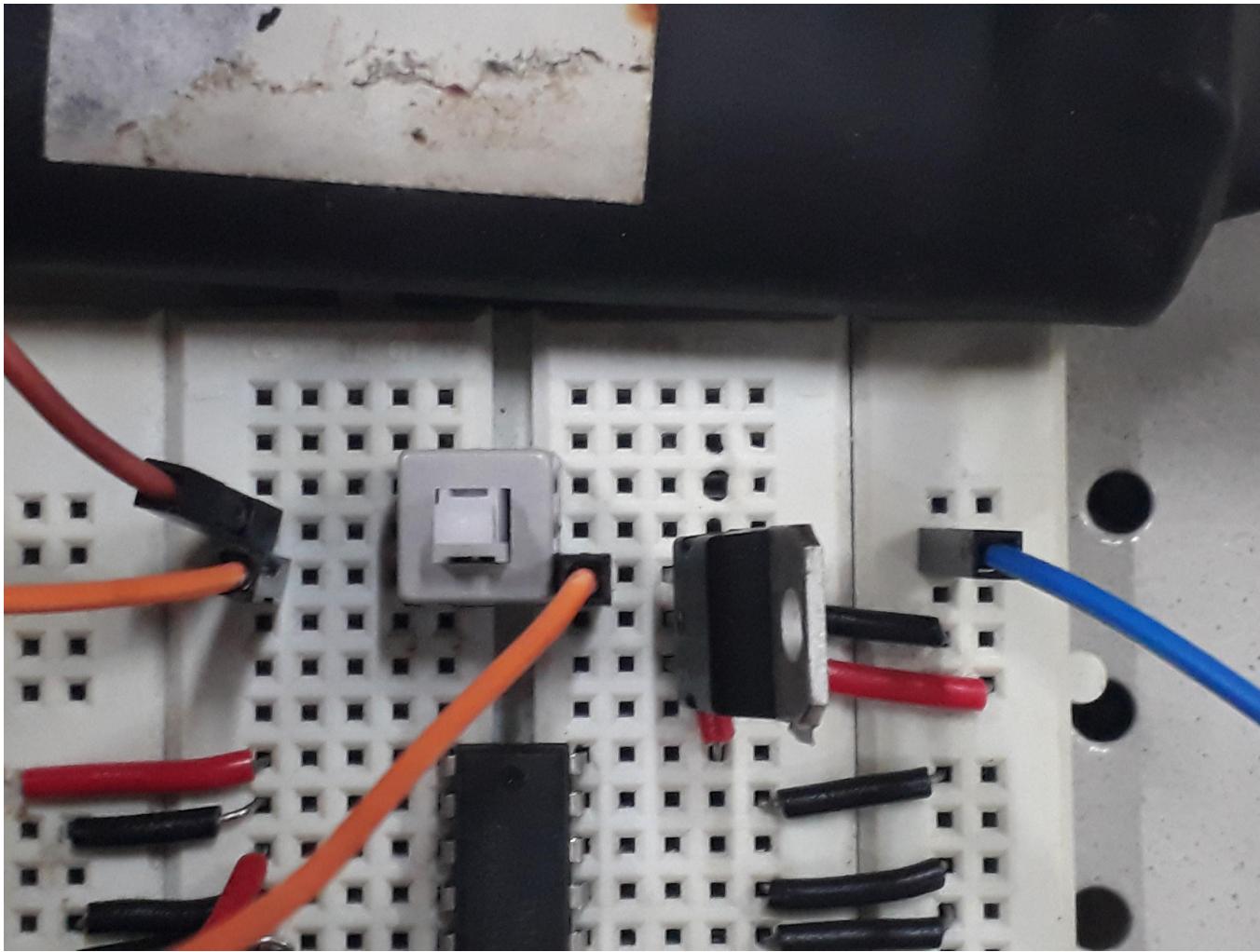


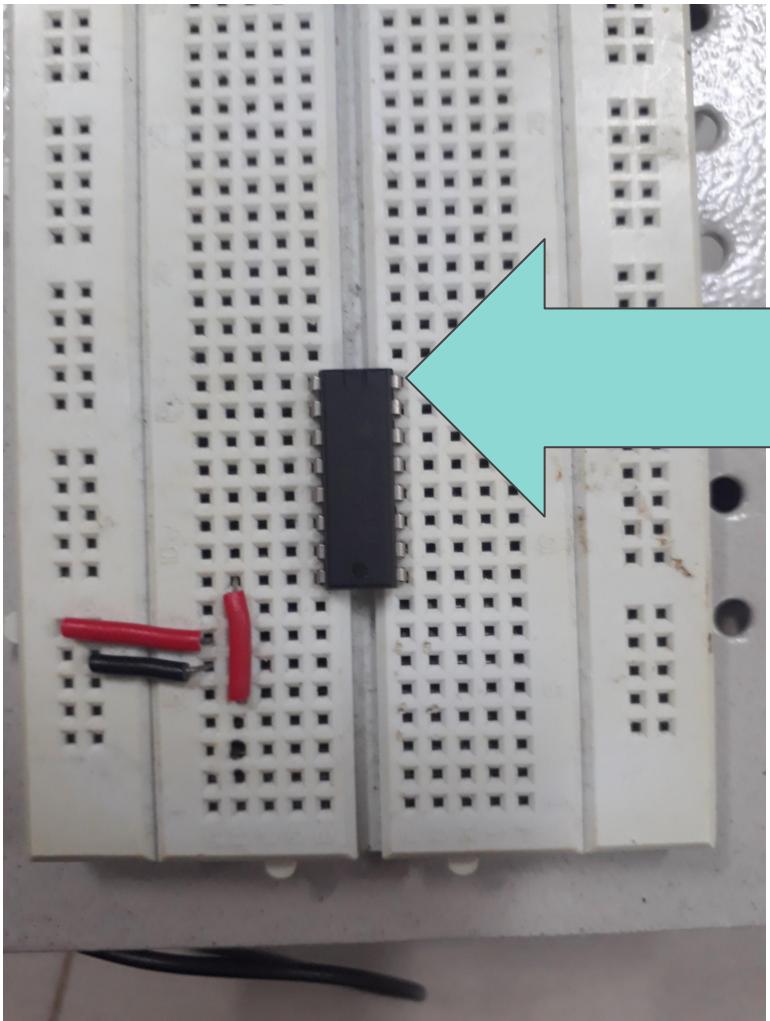
# VOLTAGE REGULATOR (7805)



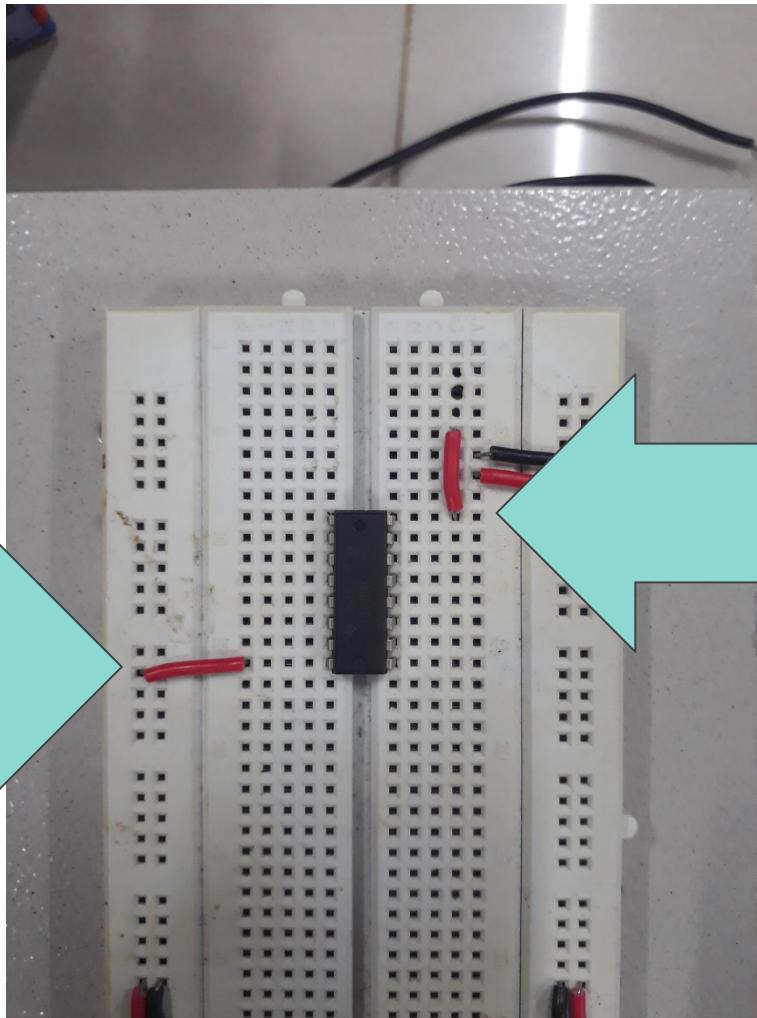






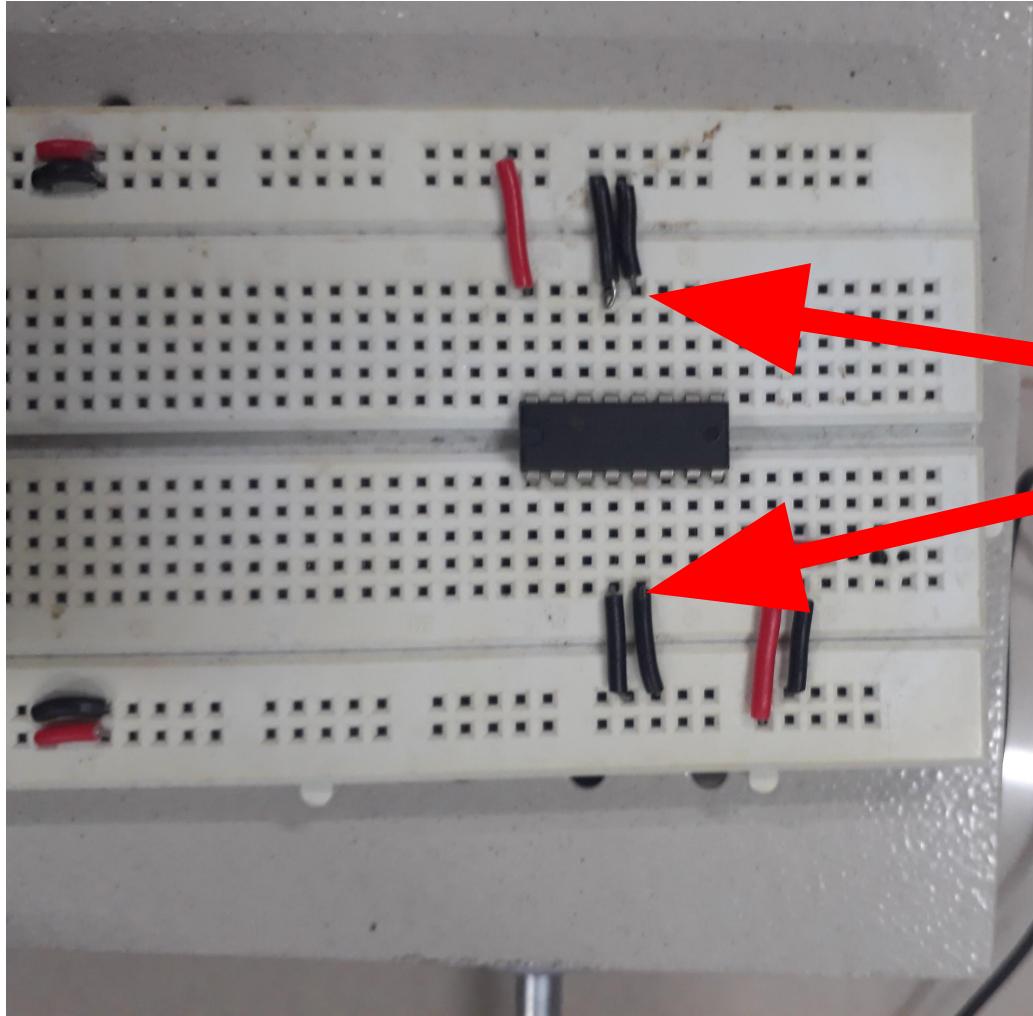


NOTCH



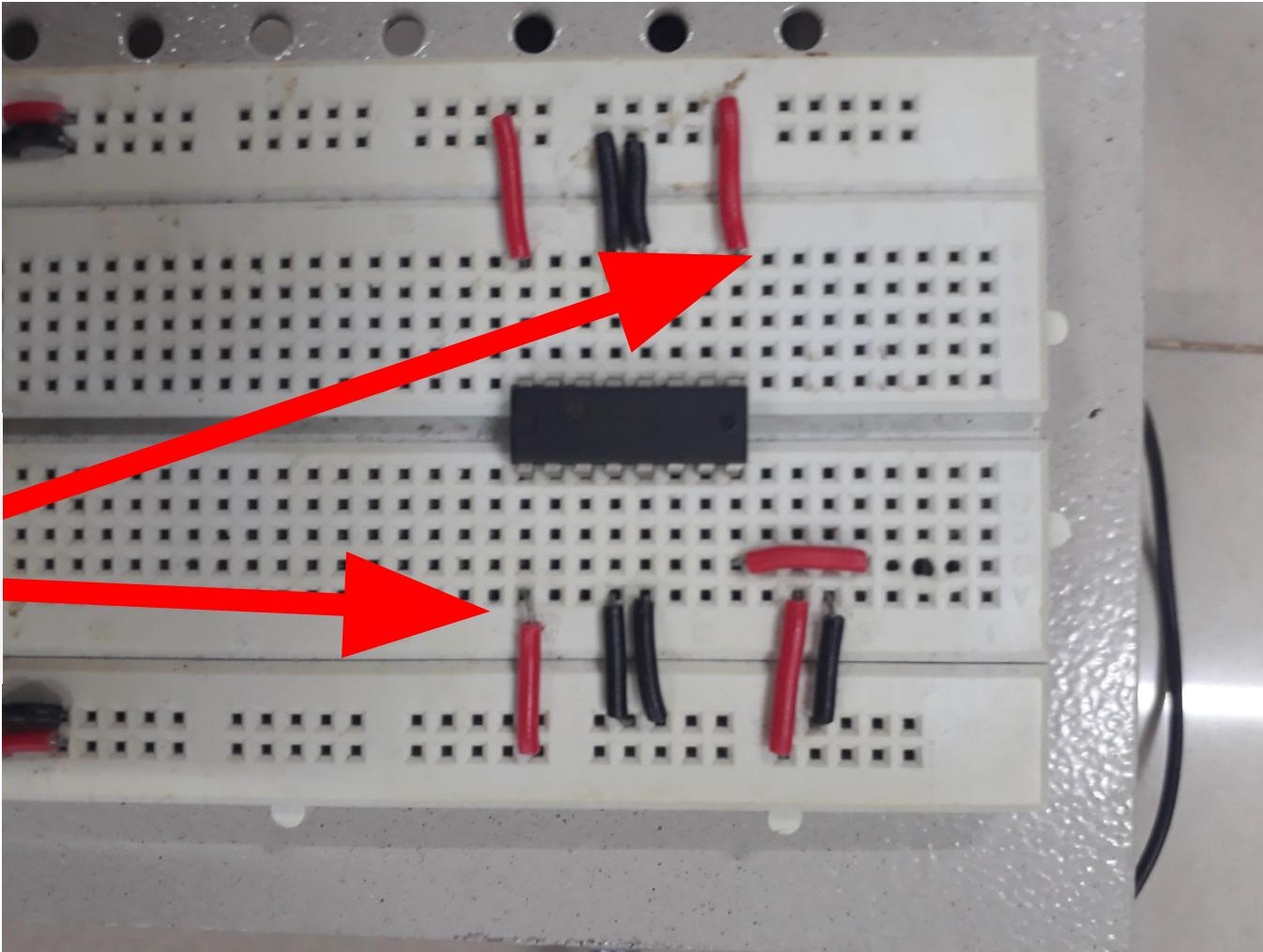
5V(VCC)

8V(VSS)

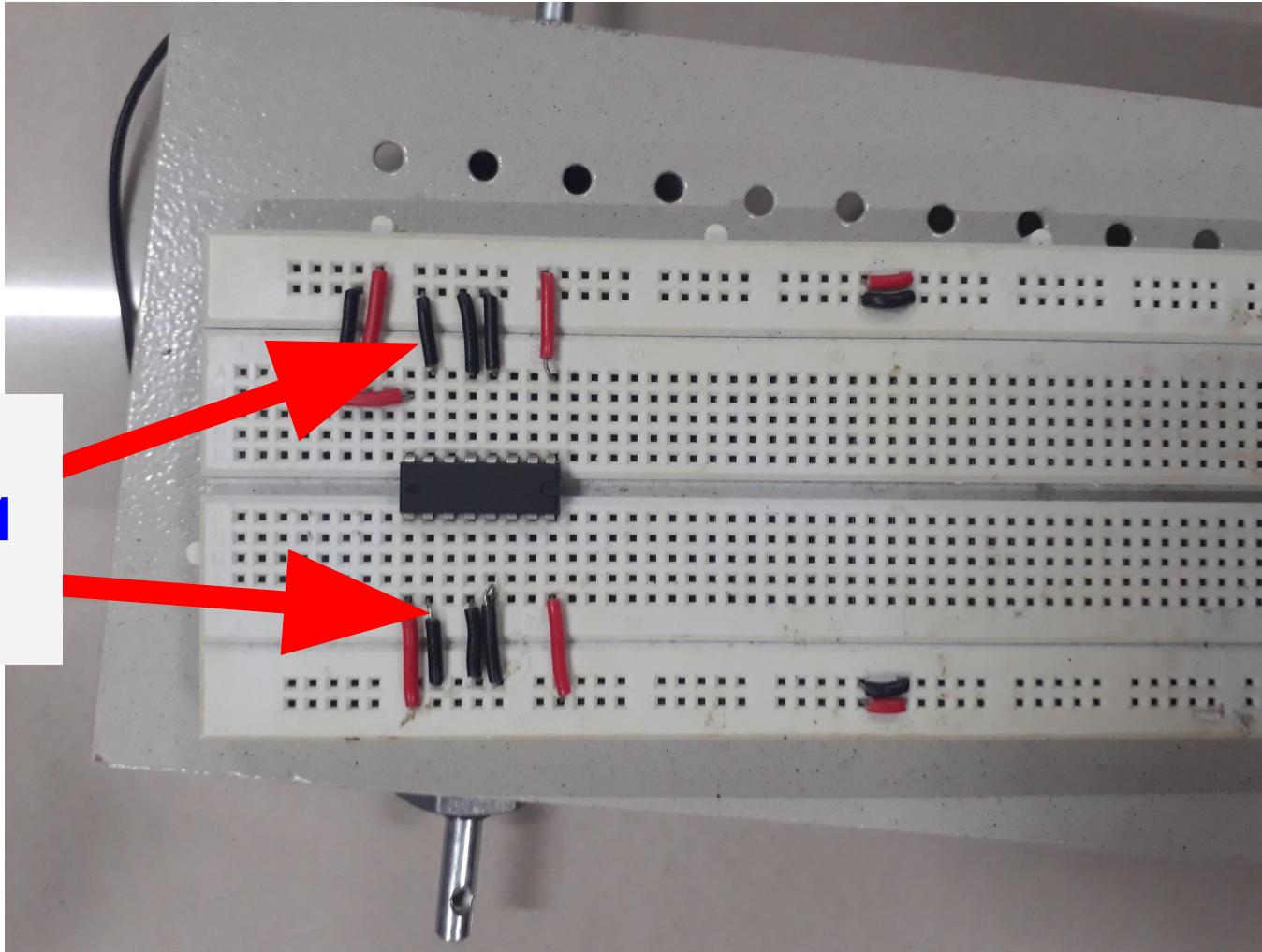


GND

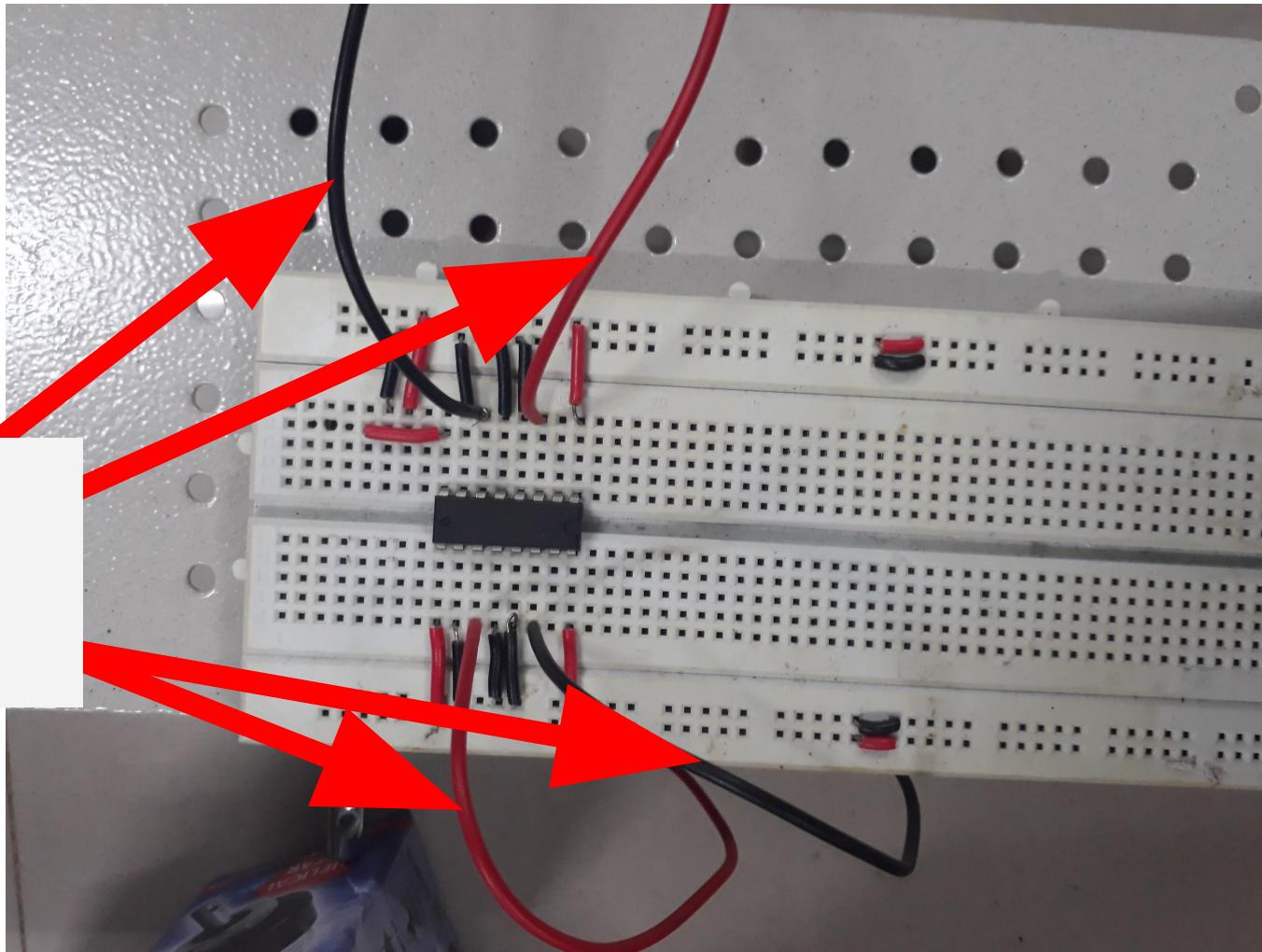
**ENABLE**



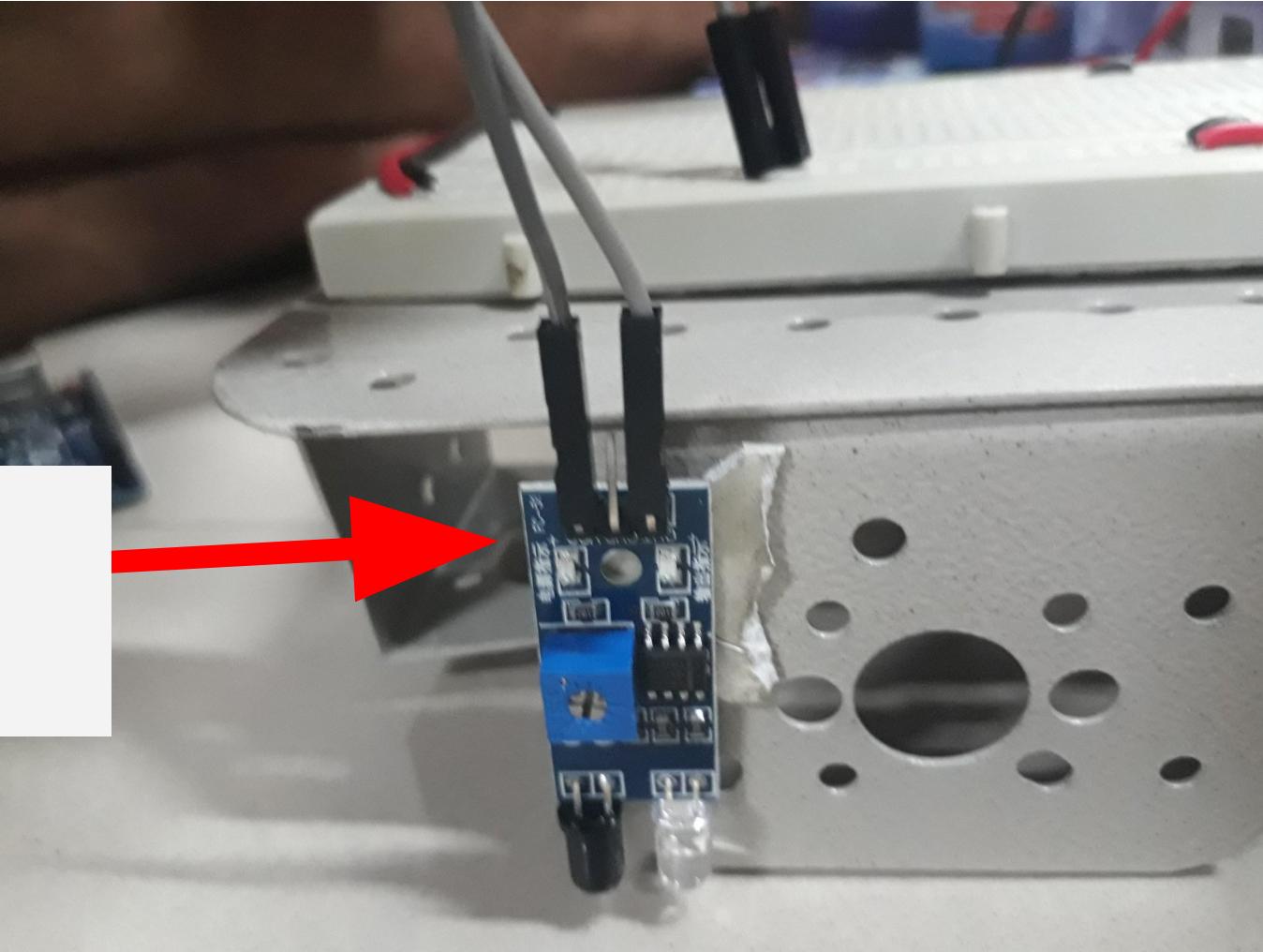
**ONE  
INPUT = 1**

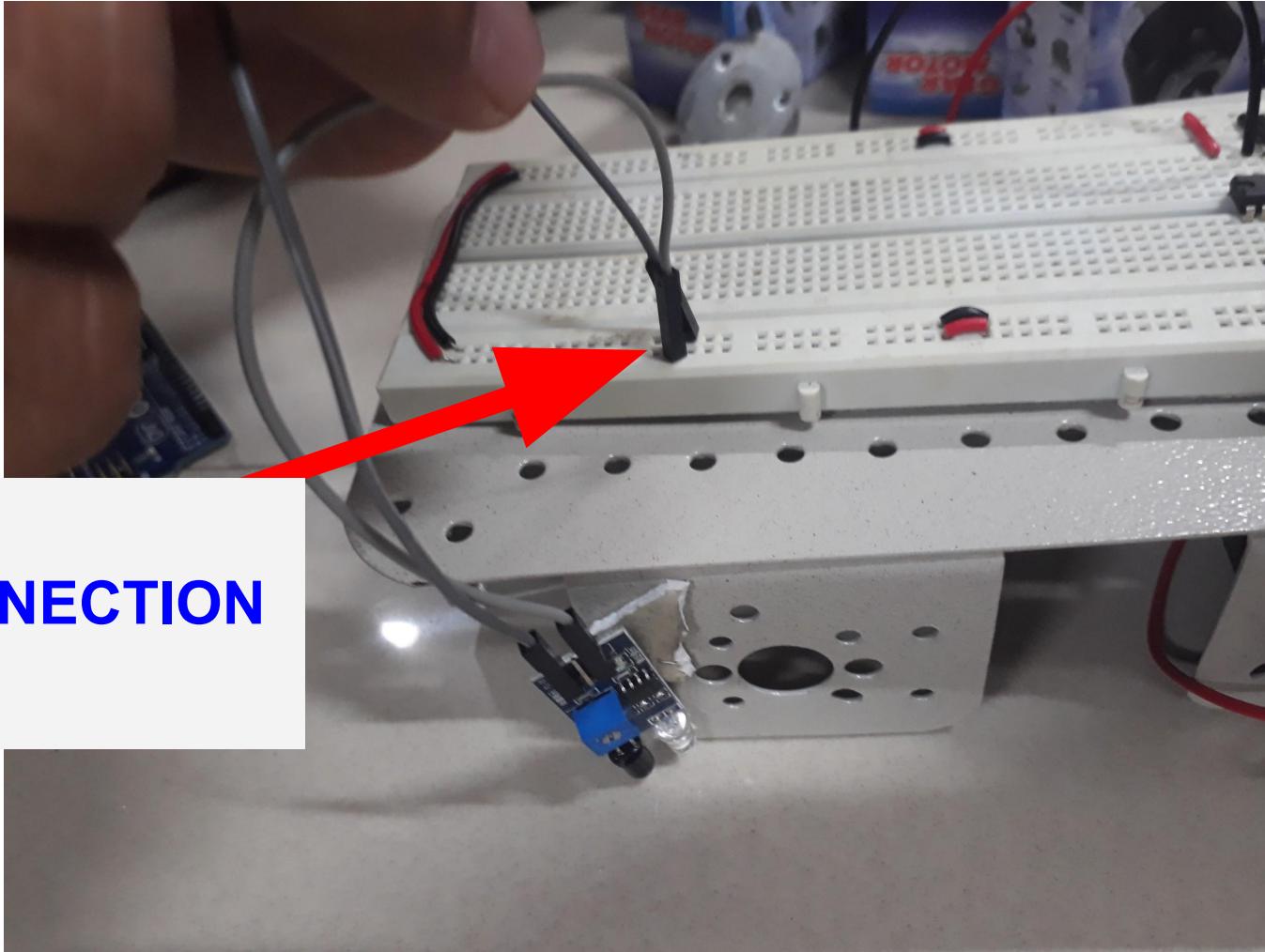


**MOTOR**

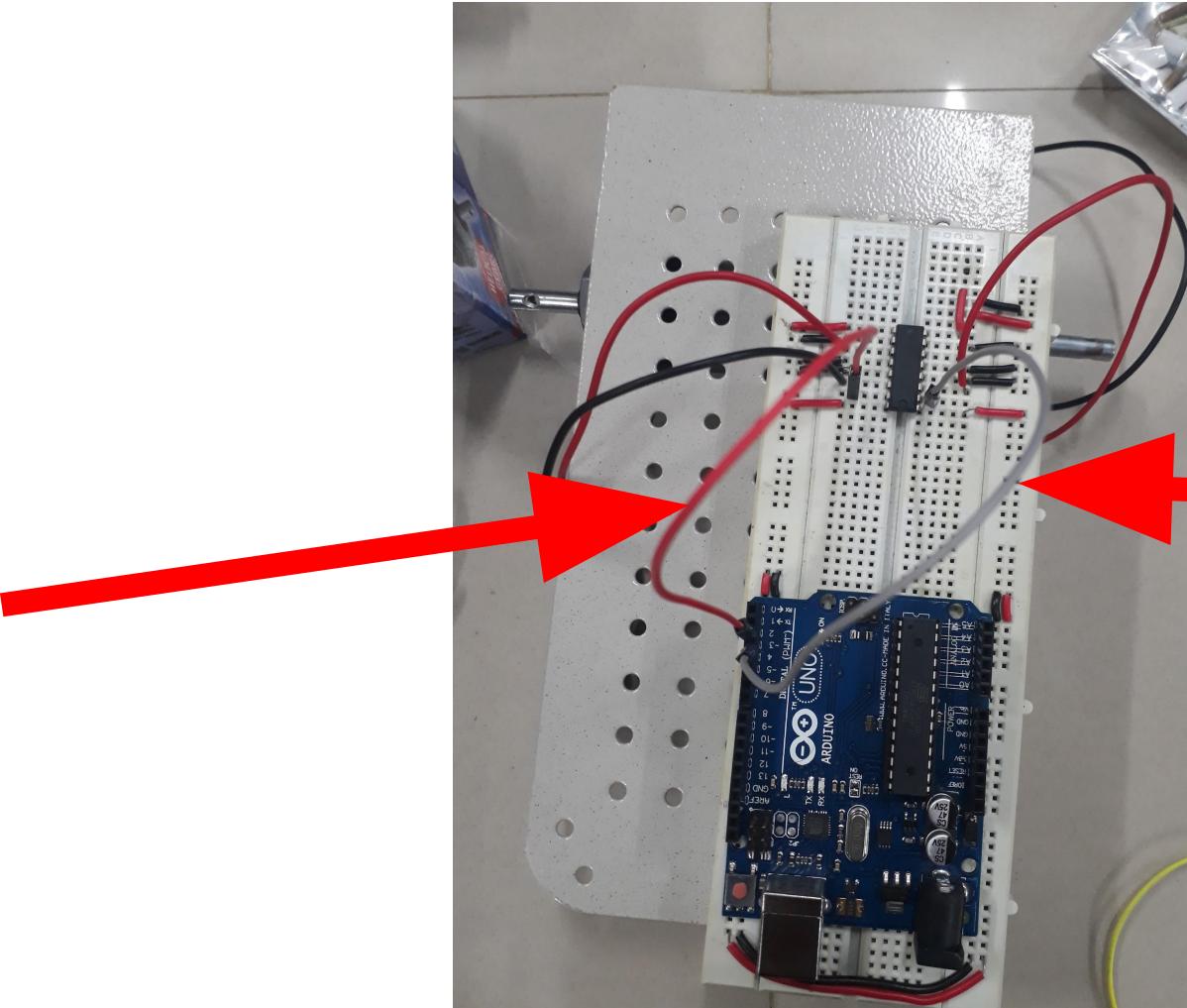


IR



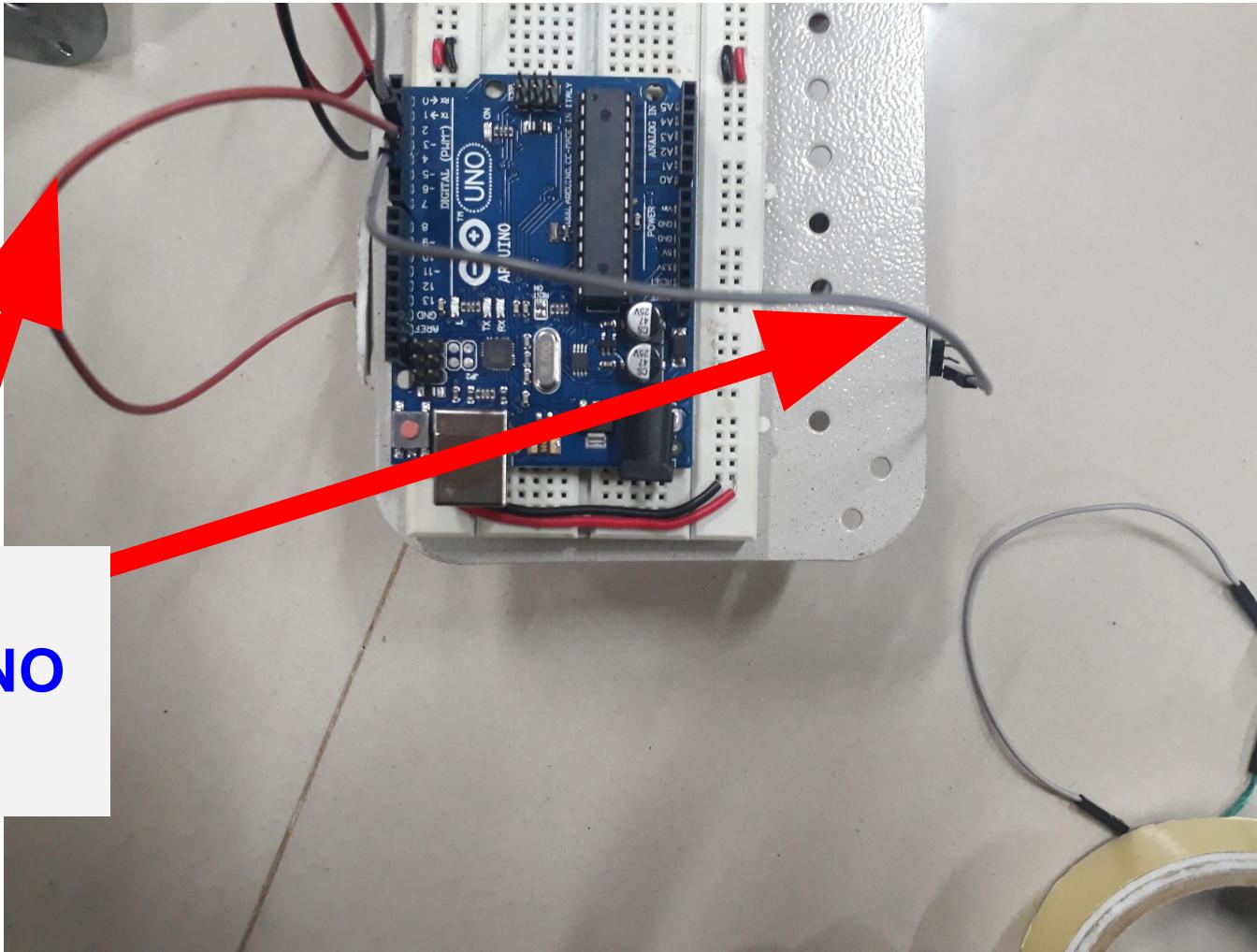


**IR  
CONNECTION**



**ARDUINO  
TO  
MOTOR**

**IR TO  
ARDUINO**



**NEVER CONNECT ARDUINO OR  
SENSORS TO > 5V**

**GIVING ARDUINO POWER**

**(V-IN PIN TO 5V;  
GND TO 0 V)**