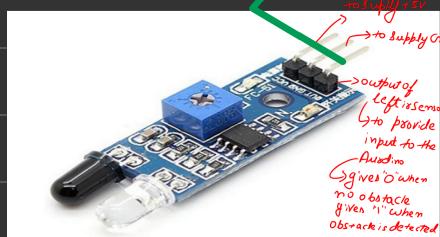
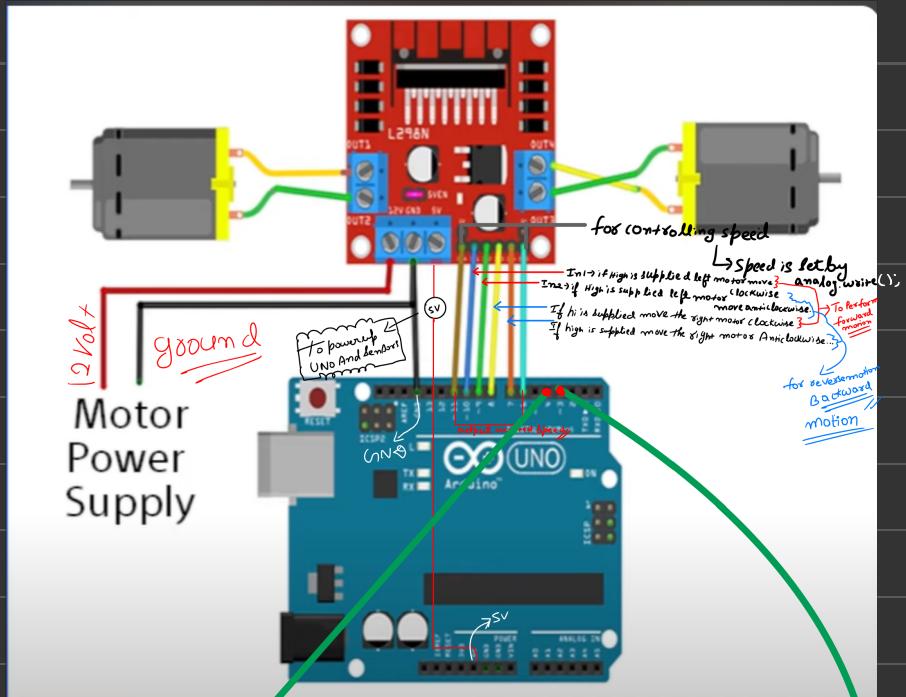


LFR (Explainatory notes)

By
PRIYANSH



understanding the Connections:-



THEORY:-

- IIR left and Right Are Input sensor that provide Input to the Arduino (0 when no obstacle) and (1 when obstacle is present)
- Now when Both ir sensor send 0 means the Bot will move forward.
- Also when Both ir sensor send 1 it signifies dead-end and now Arduino will call stop functions.
- When Left ir is detecting An obstacle and sending input 1 And right ir is not detecting any obstacle and is sending input 0 to Arduino...then UNO would call function go-right(); meaning clear bot there's An obstacle left side to go right...

→ When right is detecting An obstacle and sending input
 | And left is not detecting any obstacle and is
 sending input to Arduino... Then UNO would call
 function go_left(); meaning dear bot there's An obstacle
 right side so go left ...

Now understand functions:-

```
void go_forward() {
    digitalWrite(rmf, HIGH); // Right motor forward → Supply "High" as output to IN1 of microcontroller pin "8" of the UNO. → to move
    digitalWrite(lmf, HIGH); // Left motor forward → same explanation but for Right Side motor...
    digitalWrite(rmb, LOW); // Right motor not backward → IN2 → slow supplied via rmb → to avoid back movement? → A pin named as "not present" at digital pins of UNO...
    digitalWrite(lmb, LOW); // Left motor not backward → IN4 → slow supplied via lmb → to avoid back movement? → A pin named as "not present" at digital pins of UNO...
}
```

```
void go_right() {
    digitalWrite(rmf, LOW); // Right motor stop / reverse
    digitalWrite(rmb, HIGH); // Right motor backward → IN2 will get High output By UNO Via pin rmb; → to rotate left wheel
    digitalWrite(lmf, HIGH); // Left motor forward → IN3 will get High output By UNO Via pin lmf → to rotate right wheel
    digitalWrite(lmb, LOW); // Left motor not backward
}
```

Value of
8 microcontroller pins

```
void go_left() {  
    digitalWrite(rmf, HIGH); // Right motor forward  
    digitalWrite(rmb, LOW); // Right motor not backward  
    digitalWrite(lmf, LOW); // Left motor stop / reverse  
    digitalWrite(lmb, HIGH); // Left motor backward  
}
```

Same logic
As right
Just different
wheel

And stop function will just stop
All wheels By supplying low to In1, In2, In3, In4

In1 → High → Clockwise motion to right
motor
In3 → High → Clockwise motion to left
motor

Anti
In → High → Clockwise motion to right
motor
In → High → Clockwise motion to left
motor