Line Follower Robot Arduino

Line Follower Robot follows a black line.

## Things used in this project:-

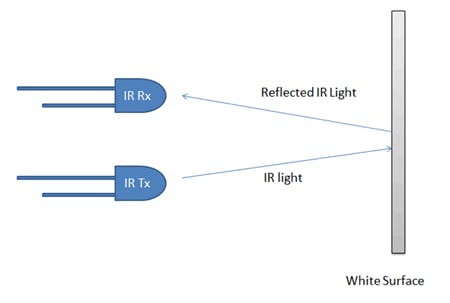
|  |
| --- |
| Hardware components-  1. Arduino UNO & Genuino UNO 2. Dual H-Bridge Motor Driver L298 3. Proximity Sensor(IR Sensor)   Software app:   1. Arduino IDE |

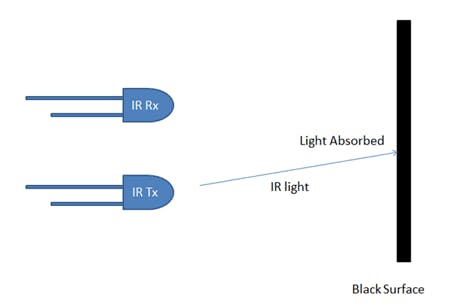
### Line Follower Robot:-

A proximity sensor is a sensor able to detect the presence of nearby objects without any physical contact. A proximity sensor often emits an electromagnetic field or a beam of electromagnetic radiation (infrared, for instance), and looks for changes in the field or return signal.

Concepts of Line Follower

Concept of working of line follower is related to light. We use here the behavior of light at black and white surface. When light fall on a white surface it is almost full reflected and in case of black surface light is completely absorbed. This behavior of light is used in **building a line follower robot**.





In this **arduino based line follower robot** we have used IR Transmitters and IR receivers also called photo diodes. They are used for sending and receiving light. IR transmits infrared lights. When infrared rays falls on white surface, it’s reflected back and catched by photodiodes which generates some voltage changes. When IR light falls on a black surface, light is absorb by the black surface and no rays are reflected back, thus photo diode does not receive any light or rays.

Here in this arduino line follower robot when sensor senses white surface then arduino gets 1 as input and when senses black line arduino gets 0 as input.

### Circuit Explanation

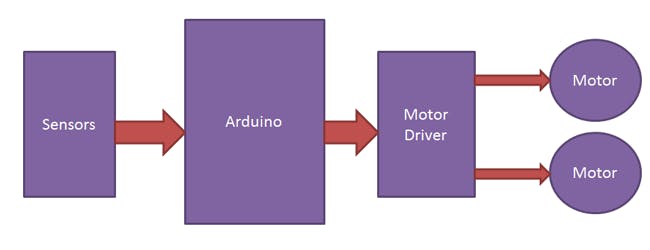
The whole **arduino line follower robot** can be divided into 3 sections: sensor section, control section and driver section.

**Sensor section:**

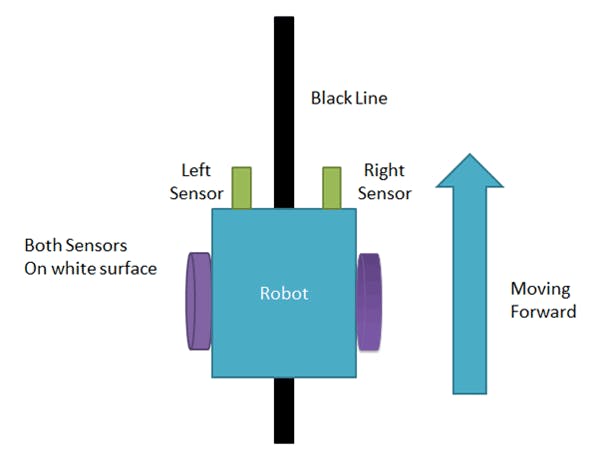
This section contains IR diodes, potentiometer, Comparator (Op-Amp) and LED’s. Potentiometer is used for setting reference voltage at comparator’s one terminal and IR sensors are used to sense the line and provide a change in voltage at comparator’s second terminal. Then comparator compares both voltages and generates a digital signal at output. Here in this **line follower circuit** we have used two comparator for two sensors. LM 358 is used as comparator. LM358 has inbuilt two low noise Op-amps.

### Working of Line Follower Robot using Arduino

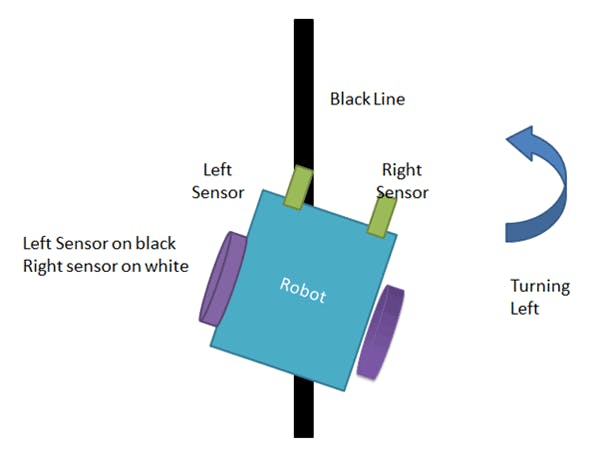
Working of line follower is very interesting. Line follower robot senses black line by using sensor and then sends the signal to arduino. Then arduino drives the motor according to sensors' output.



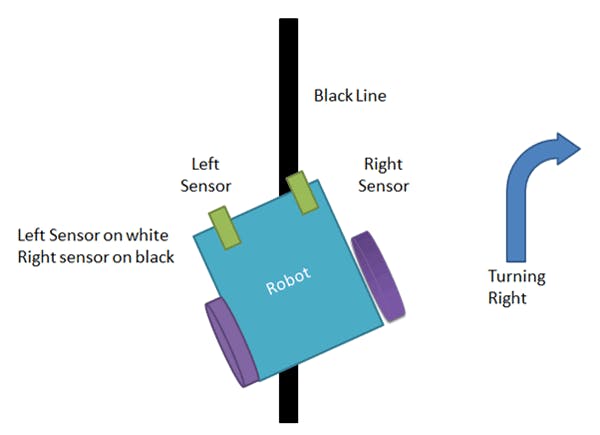
Here in this project we are using two IR sensor modules namely left sensor and right sensor. When both left and right sensor senses white then robot move forward.



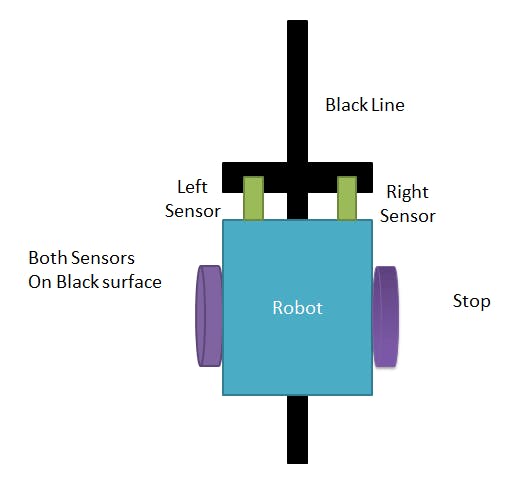
If left sensor comes on black line then robot turn left side.



If right sensor sense black line then robot turn right side until both sensor comes at white surface. When white surface comes robot starts moving on forward again.



If both sensors comes on black line, robot stops.



Code:-

int vSpeed = 110; // MAX 255

int turn\_speed = 230; // MAX 255

int turn\_delay = 10;

//L293 Connection

const int motorA1 = 8;

const int motorA2 = 10;

const int motorAspeed = 9;

const int motorB1 = 12;

const int motorB2 = 13;

const int motorBspeed = 11;

//Sensor Connection

const int left\_sensor\_pin =A0;

const int right\_sensor\_pin =A1;

int left\_sensor\_state;

int right\_sensor\_state;

void setup() {

pinMode(motorA1, OUTPUT);

pinMode(motorA2, OUTPUT);

pinMode(motorB1, OUTPUT);

pinMode(motorB2, OUTPUT);

Serial.begin(9600);

delay(3000);

}

void loop() {

left\_sensor\_state = analogRead(left\_sensor\_pin);

right\_sensor\_state = analogRead(right\_sensor\_pin);

if(right\_sensor\_state > 500 && left\_sensor\_state < 500)

{

Serial.println("turning right");

digitalWrite (motorA1,LOW);

digitalWrite(motorA2,HIGH);

digitalWrite (motorB1,LOW);

digitalWrite(motorB2,HIGH);

analogWrite (motorAspeed, vSpeed);

analogWrite (motorBspeed, turn\_speed);

}

if(right\_sensor\_state < 500 && left\_sensor\_state > 500)

{

Serial.println("turning left");

digitalWrite (motorA1,HIGH);

digitalWrite(motorA2,LOW);

digitalWrite (motorB1,HIGH);

digitalWrite(motorB2,LOW);

analogWrite (motorAspeed, turn\_speed);

analogWrite (motorBspeed, vSpeed);

delay(turn\_delay);

}

if(right\_sensor\_state > 500 && left\_sensor\_state > 500)

{

Serial.println("going forward");

digitalWrite (motorA2,LOW);

digitalWrite(motorA1,HIGH);

digitalWrite (motorB2,HIGH);

digitalWrite(motorB1,LOW);

analogWrite (motorAspeed, vSpeed);

analogWrite (motorBspeed, vSpeed);

delay(turn\_delay);

}

if(right\_sensor\_state < 500 && left\_sensor\_state < 500)

{

Serial.println("stop");

analogWrite (motorAspeed, 0);

analogWrite (motorBspeed, 0);

}

}