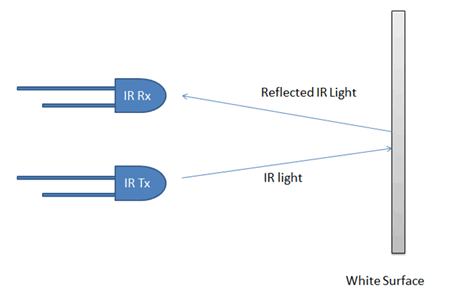
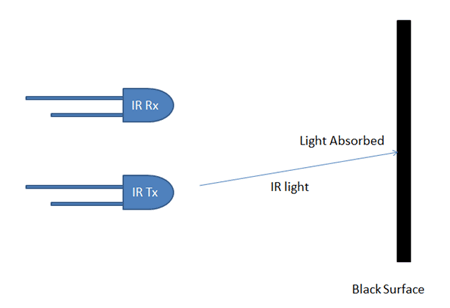
**Functional Blocks of Line Follower Robot using Arduino**

Line follower Robot is a very simple robot that follows a line, either a black line or a white line. Basically, there are two types of line follower robots: one is a black line follower which follows the black line and the second is a white line follower which follows the white line. Line follower actually senses the line and follows it. These robots are used in many applications like factory floor management and warehouses etc.

**Concept of Line Follower:**

The concept of working of line follower is related to light. We use here the behavior of light at the black and white surfaces. When light falls on a white surface it is almost fully reflected and in the case of a 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,** IR (infrared) Transmitters and IR (infrared) receivers also called as photodiodes are used. They are used for sending and receiving light. IR transmits infrared lights. When infrared rays falls on the white surface, it’s reflected back and caught by photodiodes which generate some voltage changes. When IR light falls on a black surface, light is absorbed by the black surface and no rays are reflected back, thus photo diode does not receive any light or rays.

In this Arduino line follower robot when the sensor senses white surface then Arduino gets 1 as input and when senses black line , Arduino gets 0 as input.

**Functional Blocks and its Explanation:**

**Arduino line follower robot** can be divided into 3 sections:

1. **Sensor section**:

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

1. **Control section:**

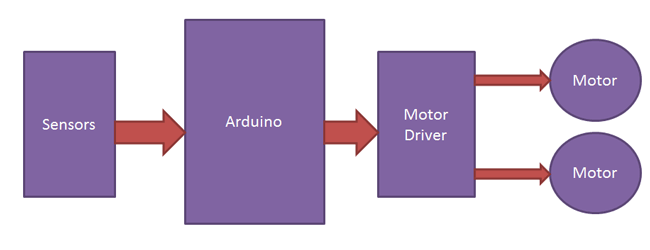
Arduino UNO is used for controlling the whole the process of the line follower robot. The outputs of comparators are connected to digital pin numbers 2 and 3 of Arduino. Arduino read these signals and send commands to driver circuit to driveline follower.

1. **Driver section:**

The driver section consists of motor driver and two DC motors. The motor driver is used for driving motors because Arduino does not supply enough voltage and current to the motor. So we add a motor driver circuit to get enough voltage and current for the motor. Arduino sends commands to this motor driver and then it drives motors.

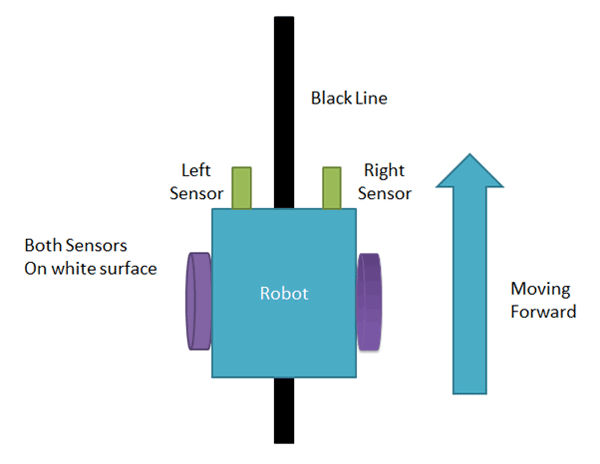
**Working of Line Follower Robot using Arduino:**

The line follower robot senses a black line by using a sensor and then sends the signal to Arduino. Then Arduino drives the motor according to sensors' output.

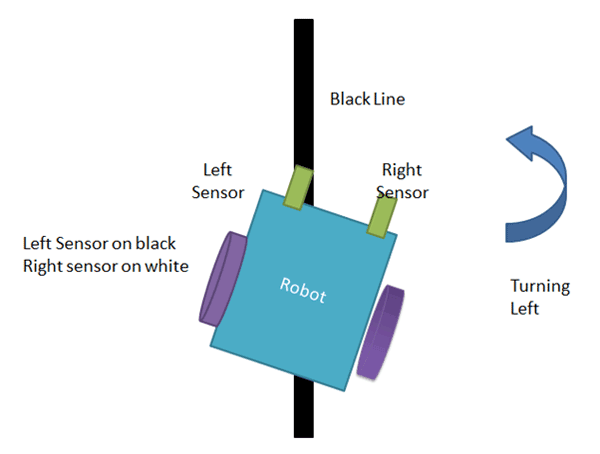


**Fig: Block Diagram of line follower Robot**

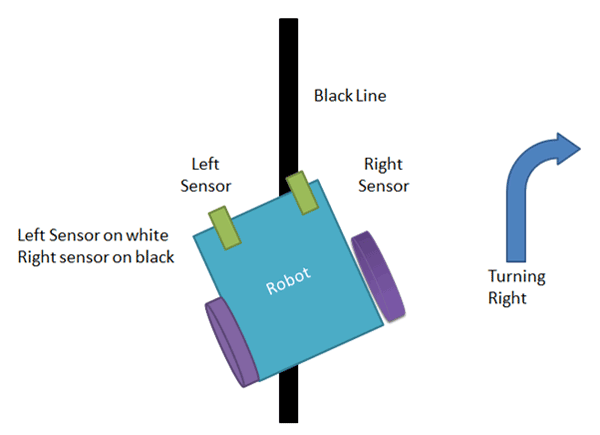
We can use two IR sensor modules namely the left sensor and the right sensor. When both left and right sensor senses white then the robot moves forward.



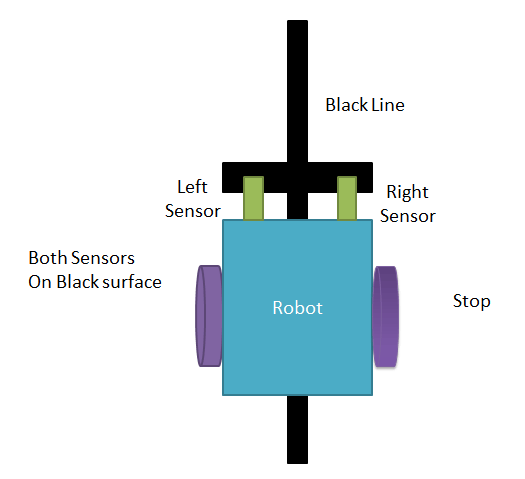
If the left sensor comes on a black line then the robot turn the left side.



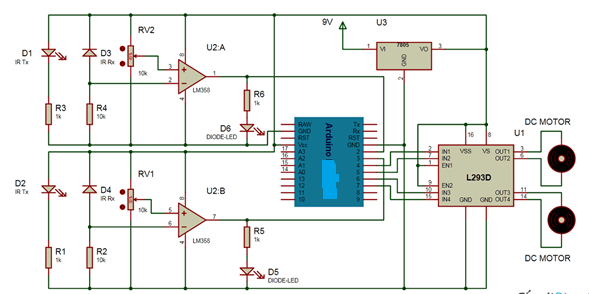
If the right sensor sense black line then robot turn right side until both sensors comes at the white surface. When the white surface comes robot starts moving on forward again.



If both sensors come on the black line, the robot stops.



**Circuit Diagram:**



**The complete circuit diagram for arduino line follower robot**is shown in the above image. The output of comparators is directly connected to Arduino digital pin number 2 and 3. And motor driver’s input pin 2, 7, 10 and 15 is connected to Arduino's digital pin number 4, 5, 6 and 7 respectively. And one motor is connected at the output pin of motor drivers 3 and 6 and another motor is connected at pin 11 and 14.

**Program:**

First of all we define input and output pin, and then in loop, we check inputs and sends output according to inputs to the output pin for the driving motor. For checking the input pin we used “if” statements.

There are four conditions in this line following robot that we read by using Arduino. We have used two sensors namely the left sensor and the right sensor.

**Required Components:**

1. **Arduino UNO:**
2. **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.

1. **IR Module:**

IR Module is sensor circuit that consists IR LED/photodiode pair, potentiometer, LM358, resistors and LED. IR sensor transmits Infrared light and photodiode receives the infrared light.

1. **Power Supply**

A voltage regulator to get 5 volts for Arduino, comparator and motor driver will be used. And a 9-volt battery is used to power the circuit.

**Code:**

/\*------ Arduino Line Follower Code----- \*/  
/\*-------defining Inputs------\*/  
#define LS 2      // left sensor  
#define RS 3      // right sensor

/\*-------defining Outputs------\*/  
#define LM1 4       // left motor  
#define LM2 5       // left motor  
#define RM1 6       // right motor  
#define RM2 7       // right motor

void setup()  
{  
  pinMode(LS, INPUT);  
  pinMode(RS, INPUT);  
  pinMode(LM1, OUTPUT);  
  pinMode(LM2, OUTPUT);  
  pinMode(RM1, OUTPUT);  
  pinMode(RM2, OUTPUT);  
}

void loop()  
{  
  if(digitalRead(LS) && digitalRead(RS))     // Move Forward  
  {  
    digitalWrite(LM1, HIGH);  
    digitalWrite(LM2, LOW);  
    digitalWrite(RM1, HIGH);  
    digitalWrite(RM2, LOW);  
  }  
    
  if(!(digitalRead(LS)) && digitalRead(RS))     // Turn right  
  {  
    digitalWrite(LM1, LOW);  
    digitalWrite(LM2, LOW);  
    digitalWrite(RM1, HIGH);  
    digitalWrite(RM2, LOW);  
  }  
    
  if(digitalRead(LS) && !(digitalRead(RS)))     // turn left  
  {  
    digitalWrite(LM1, HIGH);  
    digitalWrite(LM2, LOW);  
    digitalWrite(RM1, LOW);  
    digitalWrite(RM2, LOW);  
  }  
    
  if(!(digitalRead(LS)) && !(digitalRead(RS)))     // stop  
  {  
    digitalWrite(LM1, LOW);  
    digitalWrite(LM2, LOW);  
    digitalWrite(RM1, LOW);  
    digitalWrite(RM2, LOW);  
  }  
}