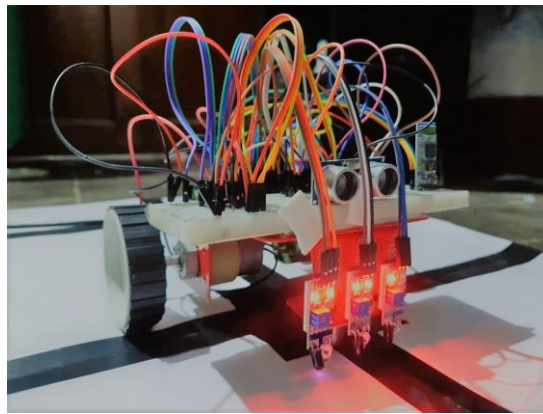


PROJECT COMPLETION REPORT

NAVIGATOR BOT



July 1, 2021

DIY PROJECT

Group – 6

- Pragma Chaudhary – 20ME30044
 - Mohit Agarwal – 20GG20027
 - Arpit Sahu – 20NA30006
 - Anubhav Shrivastava – 20GG20008
-

Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely fortunate to have got this all along the completion of my project work. Whatever I have done is only due to such guidance and assistance and I would not forget to thank them.

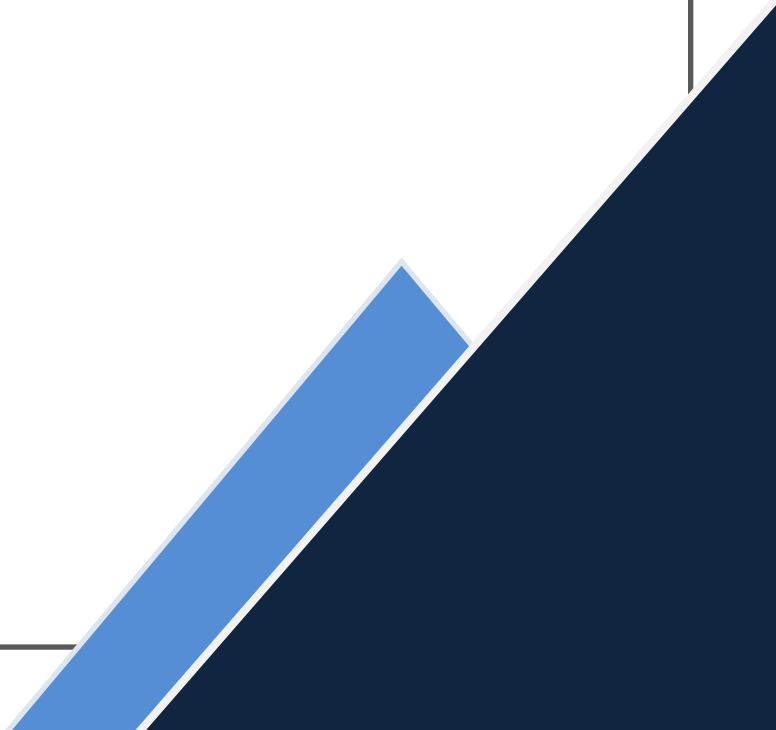
I respect and thank the DIY Faculty for giving us an opportunity to do the project work on Autonomous robots and providing us all support and guidance which made me complete the project on time.

I owe my profound gratitude to our project guide Prof. Debapratim Pandit & TA's who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We again express our gratitude to everyone who have supported us.

Table of contents

Project outline -----	4
Overall design -----	5
Working -----	6
Codes and Explanation -----	7
What we learnt -----	9
References -----	10



Project Outline

In this project we have made a **Navigator Bot**, which basically works on the principle of **line follower robot**. We can say that this bot is a multipurpose bot and can solve many things, so pointing towards a particular inspiration would be misleading. If you want someone to work in your home, restaurant, and many places that you own then this bot can help you in almost everything, just tell him where to go and it will reach there, and will complete the job, further enhancement in the designs can help in doing multiple things.

So basically, this is the basic prototype for a multipurpose robot, which works on a grid and reaches to the coordinate that you provide. Lets see what is the basic idea behind this “Navigator Bot”.

Lets say you have a grid like a cartesian plane, with nodes at the different coordinates, so here we are using three **IR sensors** for navigating the bot through this grid, further using **Ultrasonic sensors** for finding out any obstacles in its path, also **DC motors** and a 360 degree wheel for rotating, we are using an **Arduino UNO R3** board for the main computing and **Arduino IDE** for writing the code for this project.

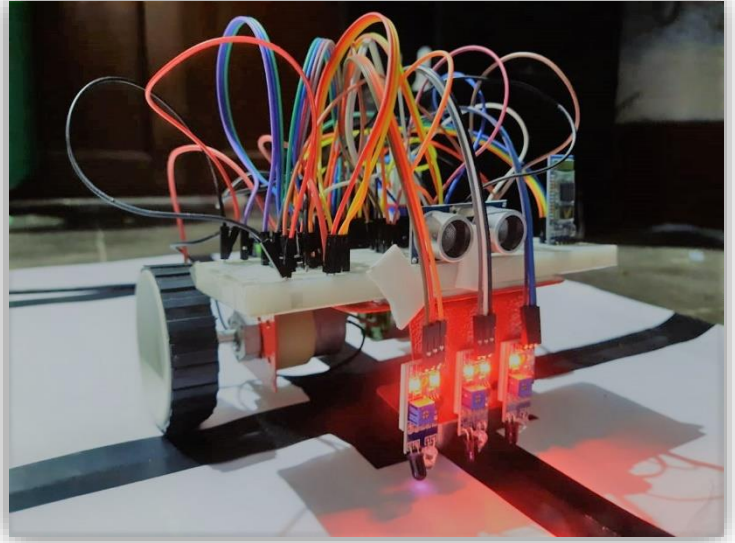
The various applications of this bot can be:

- In restaurants, these can be used like a waiter, also further enhancements in the design can be used to take the orders, delivering the order and stuff.
- In homes, these can be used as the room cleaner if we define the proper grid for the house.
- In offices also it can be used as a workmen for taking the files(also you don't have to give him salary, just a 5V DC source)

Overall Design

Stuff we have used :-

- Metallic chassis for the bot
- Arduino UNO R3 and Breadboard
- IR and Ultrasonic sensors
- HC05 Bluetooth module
- Tyres and Motors, 5V DC source

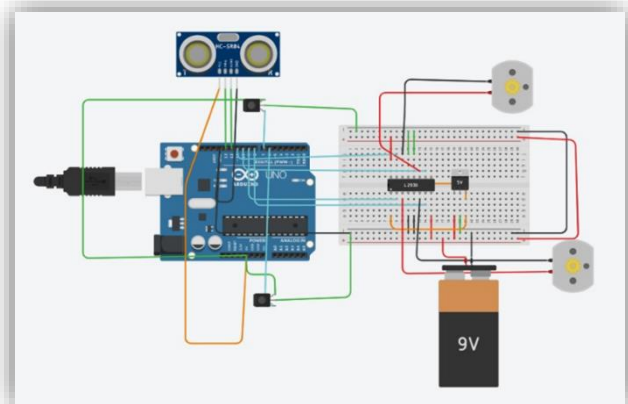


Our bot basically works with the help of **three IR sensors** which read if they are above the black line or not and moves accordingly, also we are using **Ultrasonic sensor** for finding if there is some obstacle in its way or not.

We are using an **HC05 Bluetooth module** for giving the coordinates it has to reach.

Then finally the basic stuff like motors and **breadboard** and **Arduino board**...

Integrating all these we reach to our final design.



Working

After a brief description of the components lets discuss the working of this robot.

Motion on line: The bot maintains it self on the black line by sensing the line from IR sensors and slightly turning left and right by rotating the opposite wheel as required direction.

Turns: The bot reads all the junctions as a place where all the sensors give high and to turn left it rotates its right wheel while keeping the left wheel stationary. Similarly it also turns right by moving the left wheel.

Direction sense: While turning left and right the bot uses two variables 'x' & 'y' to sense the direction and updates them after every turn it takes. Where x is +1 for positive x direction , y is +1 for positive y direction, x is -1 for negative x direction, y is -1 for negative y direction.

So we will initialize this robot in the **+ve y direction** , after that we will give initial {(0,0) in default} and final coordinates to the bot with the help of HC05 bluetooth module and the app interface we have made.

The three IR sensors will read the value, if all of them are above the black line then it will move forward, if all above the white then it will stop, and further conditions accordingly.

Once the bot have got the coordinates, we will then store them in the **temporary variables** and then will **update there values as the bot progresses on the axis.**

Then for more real life stuff we have included ultrasonic sensors that will sense the object Infront of it, if there is some obstacle then it will **reroute itself and will reach to the final destination.**

In this way our bot will reach to the given coordinates

Codes

Navigation code

```
void navigate () {
    if (y>0) {

        while ( f>0 ) {
            lineFollower();
            if ((digitalRead(L_sensor) == HIGH) && (digitalRead(M_sensor) == HIGH) && (digitalRead(R_sensor) == HIGH))
                f--;
        }

        //f=0 & y complete//

        if(e>0)
            turnright();
        else if (e<0)
            turnleft();
    }
    if ( x>0 ) {

        while ( e>0 ) {
            lineFollower();
            if ((digitalRead(L_sensor) == HIGH) && (digitalRead(M_sensor) == HIGH) && (digitalRead(R_sensor) == HIGH))
                e--;
        }
        if(f>0)
            turnleft();
        else if (f<0)
            turnright();
    }
    if (y<0) {

        while ( f<0 ) {
            lineFollower();
            if ((digitalRead(L_sensor) == HIGH) && (digitalRead(M_sensor) == HIGH) && (digitalRead(R_sensor) == HIGH))
                f++;
        }
        if(e>0)
            turnright();
        else if (e<0)
            turnleft();
    }
    if (x<0) {
```

Turn and Direction

```
void turnright() {
    digitalWrite(opL, HIGH);

    digitalWrite(opL2, LOW);

    digitalWrite(opR, LOW);

    digitalWrite(opR2, LOW);
    if (x == 0 && y == 1) {x=1;y=0;}
    else if (x == 1 && y == 0) {x=0;y=-1;}
    else if (x == 0 && y == -1) {x=-1;y=0;}
    else if (x == -1 && y == 0) {x=0;y=1;}
    if (x_temp == 0 && y_temp == 1) {x_temp =1;y_temp=0;}
    else if (x_temp == 1 && y_temp == 0) {x_temp=0;y_temp=-1;}
    else if (x_temp == 0 && y_temp == -1) {x_temp=-1;y_temp=0;}
    else if (x_temp == -1 && y_temp == 0) {x_temp=0;y_temp=1;}
```


Rerouting

```
delay(1000);
if(obs==1){Serial.print("\n Rerouting now"); // rerouting code
{
  while(1){

digitalWrite(opR, LOW);

digitalWrite(opR2, HIGH);

digitalWrite(opL, LOW);

digitalWrite(opL2, HIGH);
if ((digitalRead(L_sensor) == HIGH) && (digitalRead(M_sensor) == HIGH) && (digitalRead(R_sensor) == HIGH)){
break;
}
}
if (x == 0){
  if ( f > 0 ){
    f = f - 1;
  }
  else f = f + 1;
}
else if ( y == 0){
  if ( e > 0 ){
    e = e - 1;
  }
  else e = e + 1;
}

h = e - a;
k = f - b;
x_temp = -1 * x;
y_temp = -1 * y;
turnright();
turnright();

if (x_temp>0 ){

while ( k>0 ){
lineFollower();
if((digitalRead(L sensor) == HIGH) &&(digitalRead(M sensor) == HIGH) && (digitalRead(R sensor) == HIGH))
```

Bluetooth

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

unsigned int integerValue=0; // Max value is 65535
char incomingByte;

void loop() {
  if (Serial.available() > 0) { // something came across serial
    integerValue = 0; // throw away previous integerValue
    while(1) { // force into a loop until 'n' is received
      incomingByte = Serial.read();
      if (incomingByte == '\n') break; // exit the while(1), we're done receiving
      if (incomingByte == -1) continue; // if no characters are in the buffer read() returns -1
      integerValue *= 10; // shift left 1 decimal place
      // convert ASCII to integer, add, and shift left 1 decimal place
      integerValue = ((incomingByte - 48) + integerValue);
    }
    Serial.println(integerValue); // Do something with the value
  }
}
```

12:02

Screen1

scan

connect disconnect

Not Connected

final x

final y

initial x

initial y

enter

What we have learnt

This four-week project has taught us many things, we are thankful to the teachers who assisted us in making the final project.

- We made the whole logic of navigation, rerouting on our own, which made us to brain storm to look at all possibilities even for a small task.
- Also got the first experience in making the app interface, which we have used to make this bot Bluetooth compatible.
- Learnt how to integrate and how to debug code, and yes came to know it's not an easy task!
- Also learnt how to optimize the power consumption.

At the end we would like to say that it wasn't an easy task, we didn't knew that we will have to spend hours in debugging and integration of code when we selected this project, but yes this four weeks has taught us many things other than academics also. It was an amazing experience with an amazing team.

References

- Tinker CAD for making the circuit. -- <https://www.tinkercad.com/>
- Arduino -- <https://www.arduino.cc/>

