Line Follower Robot

Ahmad Samir
Abderahman Shazly

September 18, 2017

Line Follower Robot

A DETAILED MINI-PROJECT REPORT

AT FAB-LAB INTERNSHIP

This page was lifted Blanck

Abstract

Line follower is an autonomous robot which follows either black line in white area or white line in black area. Robot must be able to detect particular line and keep following it.

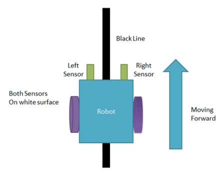
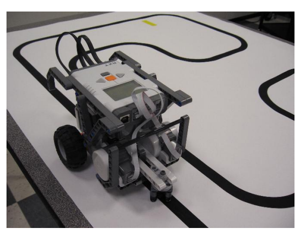


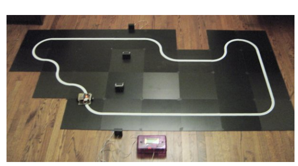
Figure 1: Working of Line Follower

For special situations such as cross overs where robot can have more than one path which can be followed, predefined path must be followed by the robot.

In our project , we asked to design the line follower robot to follow black line in white area and take right or left turn whenever cross overs turn arrives.



(a) Black line



(b) white line

Figure 2: Different area - Line Follow

Contents

Chapter 1

Introduction

Figure 1.1: Multi-Degree-Robot

An autonomous robot is a robot that performs behaviors or tasks with a high degree of autonomy, which is particularly desirable in fields such as spaceflight, household maintenance (such as cleaning), waste water treatment and delivering goods and services.

Some modern factory robots are "autonomous" within the strict confines of their direct environment. It may not be that every degree of freedom exists in their surrounding environment, but the factory robot's workplace is challenging and can often contain chaotic, unpredicted variables. The exact orientation and position of the next object of work and (in the more advanced factories) even the type of object and the required task must be determined. This can vary unpredictably (at least from the robot's point of view).

One important area of robotics research is to enable the robot to cope with its environment whether this be on land, underwater, in the air, underground, or in space.

/// A fully autonomous robot can:
1-Gain information about the environment.
2-Work for an extended period without human intervention.
3-Move either all or part of itself throughout its operating environment without human assistance.
4-Avoid situations that are harmful to people, property, or itself unless those are part of its design specifications.

An autonomous robot may also learn or gain new knowledge like adjusting for new methods of accomplishing its tasks or adapting to changing surroundings.

Like other machines, autonomous robots still require regular maintenance.

// cet here here $https://en.wikipedia.org/wiki/Autonomous_robot$

Chapter 2

Planning

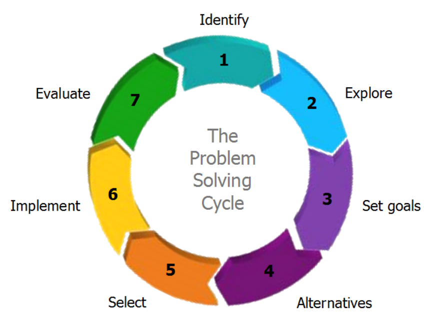


Figure 2.1: Problem-Solving-Steps

2.1 Identify

At the begining we got an identifying mission from the instructor " Ahmed Abdelbasit "

> 1- Our 3rd mini project is to make a line follower robot able to pass through the track shown in the attached photo.
> 2-The track is divided into 5 check points each increases your grade by a single point.
> 3-Our track line width is 4 cm.
> 4-Your robot must stop automatically at the last check point.

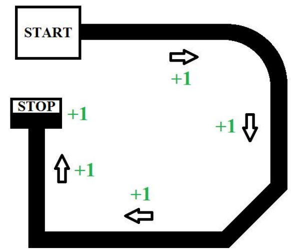


Figure 2.2: Idenfying-Mission

5-You must design and fabricate a creative robot chassis using maximum 2 sheets of wood.

6-Documentation must be written in Latex, follow the technical report layout we discussed and delivered in pdf file format.

7-Due date is next Monday 18/9/2017

2.2 Explore

The mission was clear enouf for us , so we start sarshing on the internet web sites to collect more information about "Line Follower Robot" , watching videos of working ones to notice other mistakes to avoid them and reading different codes. We ended with a good conception about what we have to do , then we moved to what we missid and what we do not know about our resorces in our "Mini-Fab-Lab".

2.3 Set Goals

We found some amazing projects and some other insignificant , but we was restricted by our limited time and resorces , so our goals :

1-Use minimum components
2-Write an efficient orginzed code
3-Build a butiful car model
4-Write a good profitional documentation
5-Make a good presentation
and nice to have :
6-Make a PCB instead of using bread-board

2.4 Components and Alternatives

The component we choosed ware:

-Two continouse DC motors

-Two wheels (10cm diam)

-One Caster-Ball

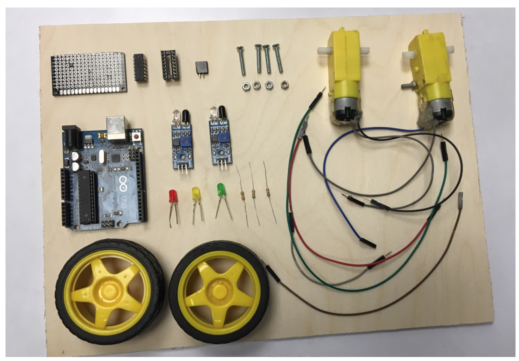
-One L293D H-Bridge IC

-Arduino uno kit.

The Alternatives ware to choose between:

-Two IR-Sensors VS Two LDR-Sensors

we choosed IR-Sensors , because it adjustable.



(a) Diff-Components

(b) Lithuime-Bettary

(c) Caster-Ball

Figure 2.3: Components

Chapter 3

Learning

Thought and planing is just the beginning.First of all we start to analyze and devide the project into pecies , software and hardware.Figure 3.2

At software part we build a small circuit by H-bridge to test the motors and sensros to make sure it works.Then we connect the arduino and wrote a simple code to choose a sutable speed of motors which modify later in trial model.

From the other side at the hardware , there was available an olde model its design was near to our design and we were allowed to try the road three times , so we used in it experimental try to test the software and to avoid hardware mistakes that may meet us.

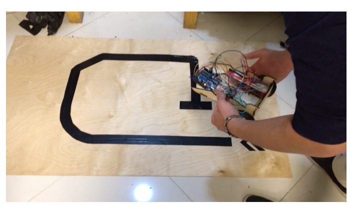


Figure 3.1: Experimental-Try

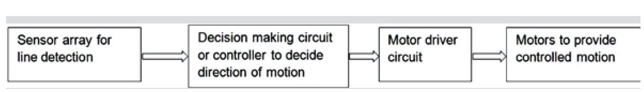


Figure 3.2: Analyzing

Chapter 4

Executing

4.1 Electric and Software

Building a Semi-Final Circuit via a bread-board and other components.Figure 4.1

Then write a sutabale code Figure 4.2, after that we tested that code Fig4.3 (a) , we recogize the field reason which was the sensor placed from the motors and its installation Fig4.3 (b) and fixed it temporarily and keeped in mind in design stage.

Figure 4.1: Semi-Final Circuit

Setup.ino

```
// Right motor
#define Rmotor 5   //pin of the motor
#define Rspeed 140 //to control the speed of Right Motor
#define in3    3   // direction of the motor
#define in4    4   // direction of the motor
unsigned long currentMillis = millis();
unsigned long previousMillis = 0;
const long interval = 250;

//Left motor
#define Lmotor 6   //pin of the motor
#define Lspeed 140 //to control the speed of Left Motor
#define in2    2   // direction of the motor
#define in3    3   // direction of the motor

//sensors
int RSensor = 11;
int LSensor = 12;
int CSensor = 13;

void setup() {
  // put your setup code here, to run once:
pinMode(Rmotor,OUTPUT);
pinMode(in3,OUTPUT);
pinMode(in4,OUTPUT);
pinMode(Lmotor,OUTPUT);
pinMode(in2,OUTPUT);
pinMode(in3,OUTPUT);

pinMode(RSensor,INPUT);
pinMode(LSensor,INPUT);
pinMode(CSensor,INPUT);
}
```

(a) Code-Setup-Part

Function_Difinition.ino

```
// Function Defi
void right() {
  analogWrite(Lmotor,Lspeed);
  analogWrite(Rmotor,0);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in2, LOW);
}

void left() {
  analogWrite(Lmotor,0);
  analogWrite(Rmotor,Rspeed);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in2, LOW);
}

void forward() {
  analogWrite(Lmotor,100);
  analogWrite(Rmotor,100);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in2, LOW);
}

void Stop() {
  analogWrite(Lmotor,0);
  analogWrite(Rmotor,0);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in2, LOW);
}

void StayintheAction (){
  if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;
    if ( digitalRead(RSensor)==HIGH && digitalRead(LSensor)==LOW ){
      right ();
      }

      }
}
```

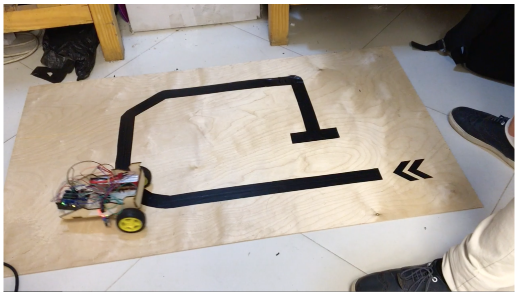
(b) Code-Fun-Diff-Part

Loop.ino

```
void loop() {
  // put your main code here, to run repeatedly:
  if ( digitalRead(RSensor)==LOW && digitalRead(LSensor)==LOW ){
     forward();
  }
  else if ( digitalRead(RSensor)==LOW && digitalRead(LSensor)==HIGH ){
    left ();
  }
  else if ( digitalRead(RSensor)==HIGH && digitalRead(LSensor)==LOW ){
    right ();
    StayintheAction;
    }
  else {
    Stop ();
    }
}
```

(c) Code-Loop-Part

Figure 4.2: Code Components



(a) Experiment-Try-Failed-One

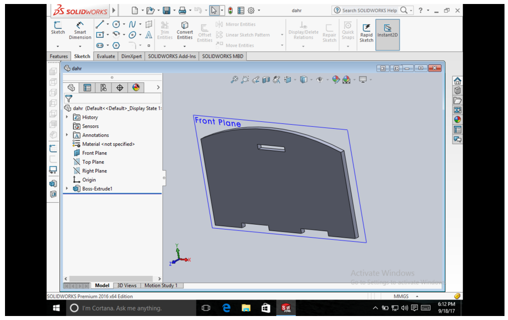
(b) Experiment-Try-Sucsses-Two

Figure 4.3: Experiments

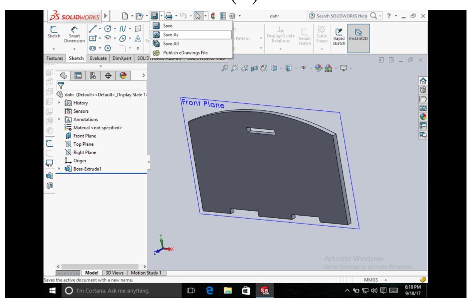
4.2 Mechanics and Hardware

We designed the care body on SolidWorks , then assembeled the to insure it is fine , finally saved it as .dfx Figure .

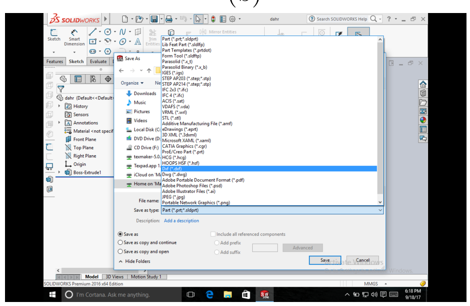
After that we moved to Laser cutter and used ”moshi” prog to control the machine.



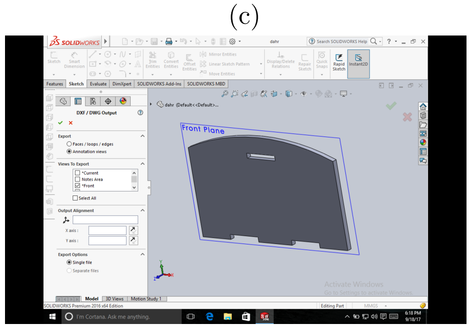
(a)



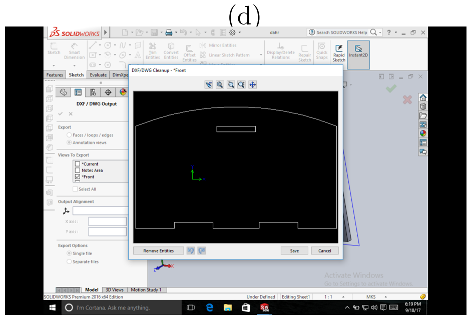
(b)



(c)



(d)



(e)

Figure 4.4: From Designing To Laser Cutter

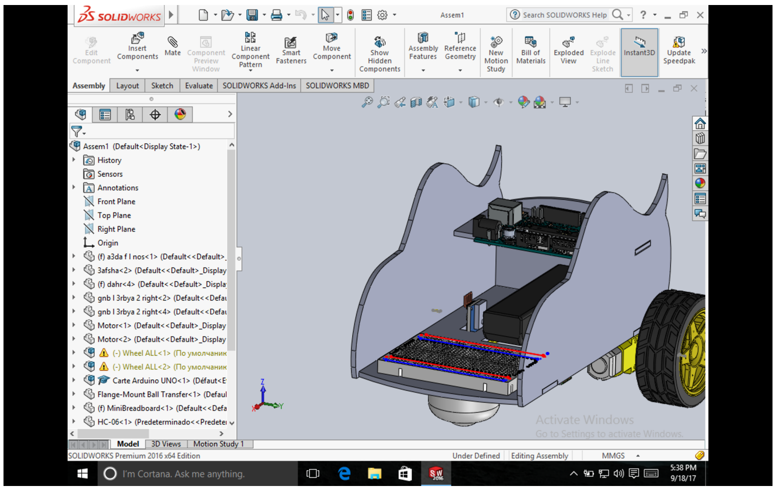
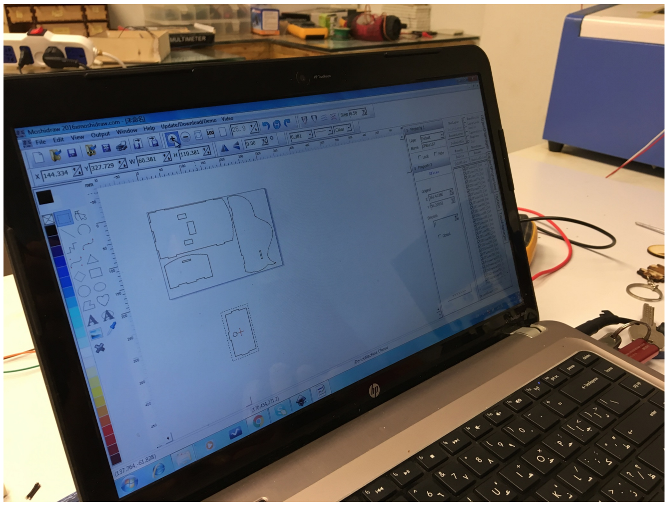
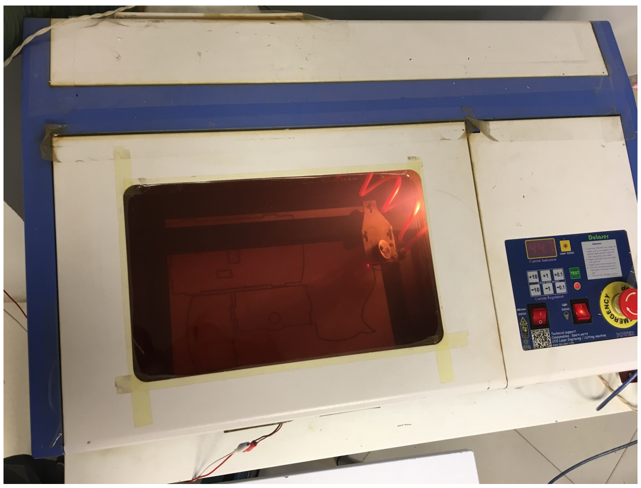


Figure 4.5: Assembly



(a)



(b)

Figure 4.6: Laser Cutter

Chapter 5

Conclusion

We finished with important , critical instructions:

-Time managment is critical.

-Use external power supply with the motors (for more than one) to avoid mess up the arduino kit.

-Profishinal coumentaion is very important.

-Confirm using the laser cutting machine.

-Confirm using H-bridge IC

List of Tables References

List of Figures

References

$https://en.wikipedia.org/wiki/Autonomous_robot$
$https://www.youtube.com$
$https://www.arduino.cc$