Precision Farming Vehicle

*Implementation of an autonomous vehicle for precision farming

Neaz Mahmud
Electronic Engineering(A4)
Hamm-Lippstadt University of Applied Science
Lippstadt, Germany
Neaz.mahmud@stud.hshl.de

Moshiour Rahman Prince
Electronic Engineering(A4)
Hamm-Lippstadt University of Applied Science
Lippstadt, Germany
moshiour-rahman.prince@stud.hshl.de

Abdullah Al Forkan
Electronic Engineering(A4)
Hamm-Lippstadt University of Applied Science
Lippstadt, Germany
Abdullah-al.forkan@stud.hshl.de

Nnaemeka Valentine Eze
Electronic Engineering(A4)
Hamm-Lippstadt University of Applied Science
Lippstadt, Germany
Nnaemeka-valentine.eze@stud.hshl.de

***Abstract*— This document is a submission of an autonomous vehicle in precision farming that is achieving increasing popularity. Thanks to the high impact it may have on crops, sustainability, efficiency, reduction of chemical use, and optimization of human effort and yield. With this vision, this project aimed to develop a prototype of an autonomous vehicle with multiple functionalities. This paper presents an overview of the hardware, design, and advanced system solutions and concept outcomes obtained in the project world. We implement a ground-based transportation system for precision farming.**

Keywords—Autonomous, vehicle, sensor, diagram, Arduino

I. Introduction

Precision agriculture is innovative; the fundamental requirement of agriculture modernization is to improve the efficiency of agriculture production without affecting various factors. In the farming automation technology is so correct identification and positioning for the agricultural objects with tracking the farming area. Therefore, in this prototyping course, an autonomous vehicle is proposed to make a vehicle that follows the line to a designated place automatically and detects the obstacle using the sensors. This project contains three main components that are an infrared sensor as a movement guide, an ultrasonic sensor as an obstacle detector and a color sensor system as a line follower. Infrared sensor Line followers are single-type mobile robots with the ability to follow a line that has a hardwires control circuit on board very precisely. As a result, a vehicle for line follower is designed for autonomous driving using Arduino UNO.

We finally present the hardware, software solutions, and prototype implemented and tested for different types of requirements. Furthermore, we describe an authentic use case in which an autonomous vehicle collaborates with precision farming without human intervention.

II. Precision Farming

Precision farming has revolutionized the way people produce crops and farm as well as our understanding of farming in general. Using precision tech has helped farmers reach a level of profitability that was previously unheard of and created a vast industry that thrives with strictly calculated ROI, yield projections, and cost reduction. Aside from the staggering profitability of employing precision farming techniques, it has helped create a sustainable attitude to farming. For struggling farmers and farms the world over, precision farming represents not only a beacon of hope but also a tangible and very real way to save their farms and provide long-term consistency. All this is done while reducing manual workload.

III. Objective

The objective of this project is to achieve practical knowledge about industrial prototyping. This project also contains some other objectives also. In this project, we must be able to follow a line with our prototyped vehicle. This autonomous vehicle must be capable of taking turns by the following line. If there is an obstacle this vehicle must detect the obstacle. In the given environment, the vehicle must be reached the endpoint. This vehicle will follow the line using a color sensor and must have to be unrepeatable.

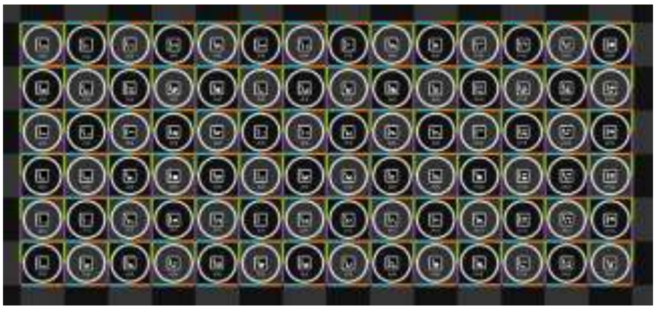


Figure:1: Environment

IV. System Model

System modeling is an aspect of system theory that looks into the construction principles of autonomous and natural systems. It has a practical application in the design of technology and electronics systems. System models can be used for different purposes such as to describe and explain real systems, compare systems and understand a system's behavior. System modeling has now come to mean representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified

Prototyping Lab, Electronic Engineering Dept, Hamm-Lippstadt University of Applied Science.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

Modeling Language. Engineers use these models to discuss design proposals and to document the system for implementation.

We implement this project by starting the system modeling. More than multiple SysML and UML diagrams were created for modeling the appropriate system. Some of them are described here:

- Requirement Diagram
- State Machine Diagram
- Block Definition Diagram
- Internal Block Diagram
- Sequence Diagram

A. *Requirement Diagram*

The requirements diagram describes a contract between the principal and all those who create the system design and implement the system. A requirement specifies flows or conditions that have to be met by the system.

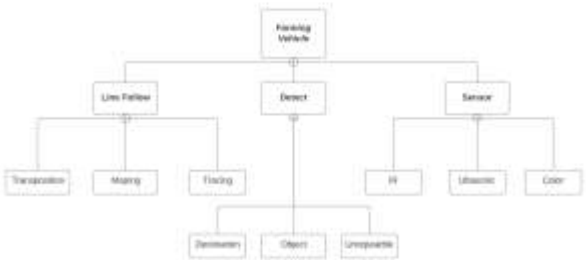


Figure:2: Requirement Diagram

- The vehicle should be able to follow the line.
- The Vehicle must be able to detect the given color line
- The vehicle will be traced randomly given position.
- The vehicle must have to map the track by staying on the color line.
- The Vehicle must be able to detect different colors.
- The vehicle should reach the destination by following the color.
- The vehicle must be able to recognize the object and complete multiple tasks.
- The vehicle should be unrepeatable and move forward to another line. Also must have to be able to take a turn.
- The vehicle should be able to use different sensors.
- The Vehicle will use an IR sensor for following the line and making turns at different stages.
- The vehicle must be able to use an ultrasonic sensor and detect the obstacle.
- The color sensor must be able to get the color value and be efficient enough to change the direction.

B. *State Machine Diagram*

The UML state model is used in SysML completely and in unchanged form. The state machines and the corresponding diagram are independent of the respective discipline of the system we model, and they do not contain any software-specific elements. Every system, regardless of whether it is a software or hardware system, or a social or biological system has states and state transitions that can be described in a state model.

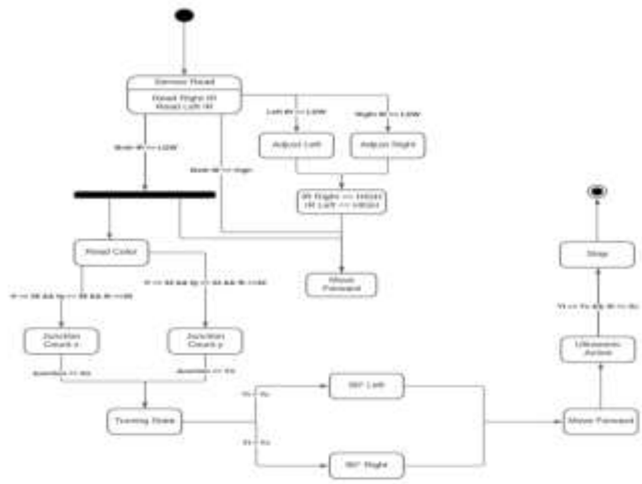


Figure:3: State Machine Diagram

C. *Block Diagram*

The Block diagram describes the structure of a system. They possess information about the block itself (attribute), or they reference other blocks they are bound with (association). In addition to the purely static structure, blocks also describe operations (i.e., behavior they can execute).

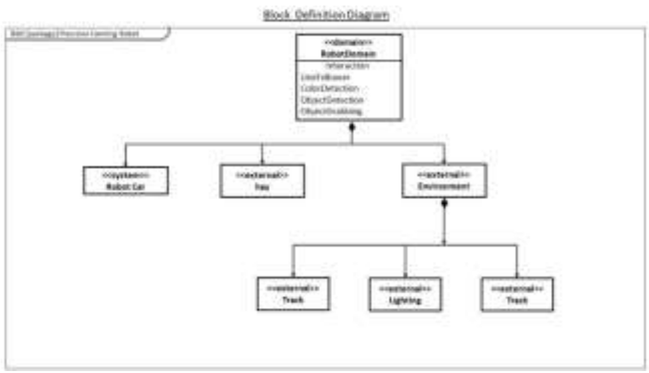


Figure:3: Block Definition Diagram

D. *Internal Block Diagram*

A UML information flow describes only abstract types—information that flows between two elements. SysML extends the information flow by a way to use connectors in the internal block diagram to describe the flow of specific objects. The flowing object is a property defined in the context of the block. While the flow port describes the objects that can flow there, the item flow describes what really flows.

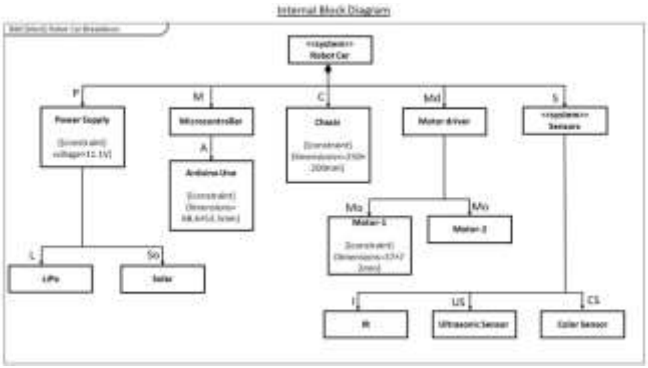


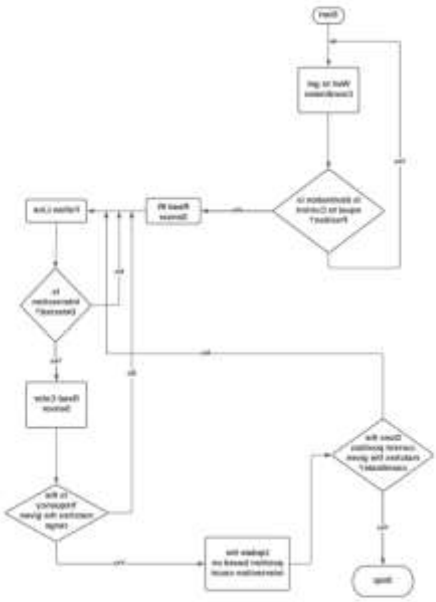
Figure:4: Internal Block Diagram

E. *Sequence Diagram*

Figure:5: Sequence Diagram

A sequence diagram is a Unified Modeling Language (UML) diagram that illustrates the sequence of messages between objects in an interaction. A sequence diagram consists of a group of objects that are represented by lifelines and the messages that they exchange over time during the interaction.

V. Flow Chart



VI. Hardware Components

The hardware components are the required hardware elements for the implementation of a prototype vehicle. This prototype builds with some 3d printed parts, laser-cut base parts, and a few electronic components. All the components are known as hardware components also. For this vehicle, we use three different types of elements.

- 3D Printing PLA
- Plywood
- Electronic

Here is the list of our prototyping vehicles:

- Microcontroller
- Bread Board
- Sensor
- Battery
- Motors
- Motor Driver
- Wheel
- Wooden Chassis

A. *Microcontroller*

Arduino Uno is a microcontroller board The Arduino UNO WiFi Rev.2 is the easiest point of entry to basic IoT with the standard form fact. The Arduino UNO WiFi Rev.2 has 14 digital input/output pins—5 can be used as PWM outputs—6 analog inputs, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller. Simply connect it to a computer with a USB cable or power it with an AC adapter or battery to get started.

Figure:6: Arduino Uno WIFI Rev.2

Microcontroller	ATMega 4809
Voltage	5V
Flash Memory	48
Digital Pins	20
PWM output	5
Weight	8.9 g
Dimension	W-53.4 L-68.6 mm

B. *Bread Board*

A breadboard is used for building temporary circuits. It is useful to designers because it allows components to be removed and replaced easily. It is useful to the person who wants to build a circuit to demonstrate its action, then reuse the components in another circuit.

Figure:5: Bread Board

Material	White abs plastic
Pins	400
Voltage Current	300V/3-5A
Dimension	82*55*8.5mm

C. *Sensor*

A sensor is a device that detects and responds to some type of input from the physical environment. The specific input could be light, heat, motion, moisture, pressure, or any one of a great number of other environmental phenomena. The output is generally a signal that is converted to a human-readable display at the sensor location or transmitted electronically over a network for reading or further processing. The vehicle requirement was to use 3 sensors.

- IR Sensor
- Ultrasonic Sensor
- Color Sensor

D. *Ultrasonic Sensor*

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves and converting the reflected sound into an electrical signal. Ultrasonic sensors have two main components: the transmitter and the receiver . to calculate the distance between the sensor and the object, the sensor measures the time it takes between the emission of the sound by the transmitter to its contact with the receiver.

Figure:6: Ultrasonic Sensor

Model	ST1099/ME140
Power Supply	5V
Measuring Angel	30 degree
Ranging Distance	2 -400 cm

Dimension	W-2.1L-4.5 cm

E. *IR Sensor*

The Line Tracker functions by illuminating a surface with its infrared light LED and then the infrared light sensor measures the reflected infrared radiation. Based on the reflected radiation's intensity, the Line Tracker can determine how light or dark the surface is below the sensor.

Figure:7: IR Sensor

Model	IDUINO ST1140
Voltage	3.3-5 V
Current	20 mA
Execution	Infrared
Dimension	28*10 mm

F. *Color Sensor*

A color sensor detects the color of the material. This sensor usually detects color in RBG scale. This sensor can categorize the color as red, blue or green. These sensors are also equipped with filters to reject the unwanted IR light and UV light. To detect the color of material three main types of equipment are required. A light source to illuminate the material surface, a surface whose color has to be detected and the receivers which can measure the reflected wavelengths.Colour sensors contain a white light emitter to illuminate the surface. Three filters with wavelength sensitivities at 580nm, 540nm, 450nm to measure the wavelengths of red, green and blue colors respectively.

Figure:6: Ultrasonic Sensor

Model	TCS3200
Voltage	3V-5V
Chip	Tcs3200
Dimension	⌀37mm

G. *Battery*

A lithium-polymer (LiPo, LIP or Li-Poly) battery is a type of rechargeable battery that uses a soft polymer casing so that the lithium-ion battery inside it rests in a soft external "pouch." It may also refer to a lithium-ion battery that uses a gelled polymer as an electrolyte. However, the term commonly refers to a type of lithium-ion battery in a pouch format. Lithium-polymer batteries are lighter and more flexible than other kinds of lithium-ion batteries because of their soft shells, allowing them to be used in mobile and other electronic devices, as well as in remote control vehicles.

Figure:7: Battery

Model	Conrad Li-ion Battery
Voltage	11.1 v
Capacity	3200 mAh
Cell Count	3
Dimension	138*46*24 mm

H. *Motors*

A DC Gear motor, is also called DC Geared Motor, Geared Dc Motor and gearhead motor or gearbox motor. It consists of a electric DC motor and a gearbox or gearhead; these gearheads are used to reduce the DC motor speed, while increase the DC motor torque. Therefore user can get lower speed and higher torque from gear motor.

Figure:8: Motor

Voltage	12v
Torque	60 Ncm
Speed	174 rpm
Efficiency	66%
Dimension	37*72mm

I. *Motor Driver*

The Motor Driver is a module for motors that allows you to control the working speed and direction of two motors simultaneously. Motor requires high amount of current whereas the controller circuit works on low current signals. So the function of motor drivers is to take a low-current control signal and then turn it into a higher-current signal that can drive a motor.

Figure:9: Motor Driver

Driver	L298N
Voltage	5V
Current	2A
Dimension	43*43*27mm

J. *Wheel*

A wheel is a circular component that is intended to rotate on an axle bearing. For making this prototype we use three wheels, two basic and one front wheel.

Figure:10: Wheel

Dimension	36 mm
Quantity	3
Front Wheel	5-10 mm

K. *Wooden Chassis*

For this prototype we had two different options for building chassis. Those are 3D printing and laser cutting. We choose laser cutting with plywood for chassis

Figure:11: Chassis

Dimension	290*180 mm
Thickness	6 mm

VII. DESIGN

To design the vehicle, we choose to use laser cutting and printed 3D parts. The advantage of laser cutting parts they are not so much costly and take a short time. The laser cut parts are hard enough and easily useable. Out team task was implementing a prototype vehicle. According to many tastings phrase some parts could be damage. The laser cut parts can be restore in very short time. Also 3D printed parts was strong enough. The complication with 3D printings parts was much time absorbing. The advantage of 3D printing is that the parts can be modeled and printed according to user taste. We used the solid works for design the vehicle. Also we used some other application for printing and leaser cutting parts.

For some of the components—such as the motors and the array sensor—the 3D file can be found on the web and can be imported into the model. But some of the parts that need to be printed have to be manually designed. There are a few tips in designing a vehicle. First, the parts must have a certain thickness. order to not bend or even break. Second, a distance between certain parts is recommended, in the case of some of the electronic parts (Arduino, motor driver) and the battery holder. This also applies for screws, because there are a lot of lengths for them and they could either enter certain printed parts or touch the track. To avoid this, the screws can be cut, or the design can be made in such way that it includes the screws and are therefore no longer a problem. Finally, the dimensions of the vehicle may vary depending on what it is intended for. The line follower competitions have some restrictions, for example the cars maximum dimensions are X=30 cm, Y=20 cm Z=20 cm.

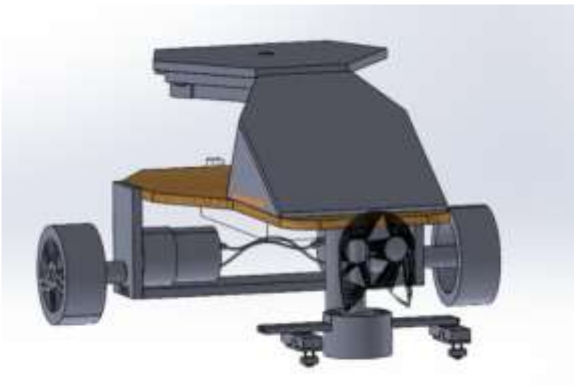


Figure:12: CAD Design

VIII. ALGORITHM

Autonomous vehicles first interact with their surrounding using mounted sensors. This vehicle is expected to navigate successfully in complex environments, locate and track objects, detect obstacles, determine their own location.

Coordinate	X0	Y0	X1	Y1	X2	Y2	Action	Direction
	0	0	>0				Forward Till X1	East
	0	0	>0	>0			Sharp left	North
	0	0	>0	<0			Sharp Right	South
	0	0		>0	>X1		Sharp Right	East
	0	0		<0	<X1		Sharp Right	West
	0	0		<0	>X1		Sharp left	East
	0	0		>0	<X1		Sharp left	West
	0	0			>X1	>Y1	Sharp left	North
	0	0			<X1	<Y1	Sharp left	South
	0	0			>X1	<Y1	Sharp Right	South
	00	0			<X1	>Y1	Sharp Right	North
	0	0				==Y2	STOP	

Figure:12: Turning Algorithm based on given coordinate table.

The robot car will always start from (0,0) and move forward toward East direction. After reaching X1 point it will check if the given Y1 coordinate is greater or smaller than 0. If it is greater than 0, it will take a 90° left turn toward North and move forward till Y1. Else If it is less than 0, it will take a 90° Right turn toward South and move forward till Y1.

After reaching Y1 point it will check if the given X2 coordinate is greater or smaller than X1. If (Y1>0) && (X2>X1), it will take a 90° Right turn toward East and move forward till X2. Else If (Y1>0) && (X2<X1), it will take a 90° Left turn toward West and move forward till X2.

Else If (Y1<0) && (X2>X1), it will take a 90° Left turn toward East and move forward till X2. Else If (Y1<0) && (X2<X1), it will take a 90° Right turn toward West and move forward till X2.

After Reaching X2 point, it will check if the given Y2 coordinate is greater or smaller than Y1. If (Y2>Y1) && (X2>X1), it will take a 90° Left turn toward North and stop at Y2. Else If (Y2>Y1) && (X2<X1), it will take a 90° Right turn toward North and stop at Y2.

Else If (Y2<Y1) && (X2>X1), it will take a 90° Right turn toward South and stop at Y2. Else If (Y2<Y1) && (X2<X1), it will take a 90° Left turn toward South and stop at Y2

RIGHT IR SENSOR	LEFT IR SENSOR	ACTION
HIGH	HIGH	FORWARD
HIGH	LOW	ADJUST LEFT
LOW	HIGH	ADJUST RIGHT
LOW	LOW	FORWARD && JUNCTION COUNT

The Car has two IR sensors attached on the front side. Following outside of the line was implemented to keep the car on the line while running.

Whenever both IR sensors are on the black line it will give (HIGH-HIGH) reading. In that case the car will move forward. If the Right IR sensor comes on the colored line, It will give a (LOW) reading. The Right motor will move backward, and the left motor will move forward to adjust the car on the line.

The same logic goes for the other scenario also. If the Left IR sensor comes on the colored line, It will give a (LOW) reading. The Left motor will move backward and the Right motor will move forward to adjust the car on the line.

Red Frequency	Green Frequency	Blue Freuency	Line Color
>=20 && <=22	>=30 && <=32	>=42 && <=44	Orange
>=23 && <=25	>=33 && <=35	>=45 && <=47	Purple
>=26 && <=29	>=36 && <=38	>=48 && <=50	Green
>=20 && <=22	>=39 && <=41	>=51 && <=53	Blue

While crossing a junction, Both IR sensors will give (LOW-LOW) reading. The car will still move forward and check for color frequency with the help of RGB color sensor (TCS3200). If the frequency matches with the given range and based on the range Junction count X or Junction count Y will be incremented. The Color sensor does not give the same frequency read every time, so IR sensors were also use for the Junction count.

This Junction count will help the car to determine its current position and make a decision about when to turn on which direction

```
void color ()
{
  //color sensor, frequency reading from different colors
  digitalWrite(10, LOW);
  digitalWrite(11, LOW);
  frequency_red = pulseIn(4, LOW);
  Serial.print("Red = ");
  Serial.print(frequency_red);
  Serial.print(", ");
  digitalWrite(10, HIGH);
  digitalWrite(11, HIGH);
  frequency_green = pulseIn(4, LOW);
  Serial.print("Green = ");
  Serial.print(frequency_green);
  Serial.print(", ");
  digitalWrite(10, LOW);
  digitalWrite(11, HIGH);
  frequency_blue = pulseIn(4, LOW);
  Serial.print("Blue = ");
  Serial.println(frequency_blue);
}
```

IX. RESULT

This vehicle is a prototype.When it will be actually implemented in real time, there will be practical errors but those can be solved with better components and more time .Once it will be fixed it will be helpful application for health care centers Due to structure and technical lacuna of the robot ,it is not possible to transport high load goods. However in future if the center of gravity and other technical aspects are taken off, it is also possible to transport high load goods.

X. CONCLUSION

In its current form the vehicle is enough capable that it can follow any line. We must build a vehicle that has light weight and fast, because for precision farming points are awarded based upon the distance covered and the speed of the overall vehicle. Before making this kind of project it is required to relocate the perfect resources and have best information about those parts. Though it has some limitations on physical and environmental but the simple vehicle is more effective and profitable for people. Implementing this project based on path finder can make our life easier. It can be modified and upgraded with new features and options regarding its usability. This product is economically cheaper. It's a step creating revolution implementing autonomous vehicles in needs off precision farming and mankind.

XI. RECOMMENDATION

In the future, Our team believe that this type of vehicle will play an important role in everyday life. Although the simple control system is suitable for this type of robot, there is room for improvement. One such improvement would be an AI (perception type) of system that can process the track before the vehicle is there. That would be an interesting idea to implement and perfect. In addition to sensors, it would use a video camera that would be manipulated with the help of an object-detection AI. The disadvantage is that it would need a processor, taking up more space—an issue for small projects. All that we said, for large platforms—like vehicles—this should not be a problem.As our team said earlier, a line follower vehicle has the advantage that it can be used without human help. It can perform tasks in farming fields where there aren't enough staff , or where work is much too dangerous (industrial) or for simply replacing boring or long-term activities (driving).

XII. ACKNOWLEDGEMENT

Team members from Group A4, we are grateful because we managed to complete Precision Farming Vehicle project by under observation our two teachers Kristian Rother and Gido Wahrmann. We also sincerely thank our lecturer Prof. Stefan Henkler for the guidance and encouragement in finishing this project and teaching us in this course. Last but not least, we would like to express our gratitude to our friends and respondents for the support.

References

[1] https://learning.oreilly.com/library/view/systems-engineering-with/9780123742742/ *(references)*

[2] A. Attar, A. Ansari, A. Desai, S. Khan, and D. Sonawale, "Line Follower And Obstacle Avoidance Bot Using Arduino," no. 4, pp. 18–21, 2017.

[3] X. Wang, S. H. Kim, C. Chen, L. Chen, H. He, and L. Qian, "Development and Applications of Line Following Robot Based Health Care Management System," ACS Appl. Mater. Interfaces, vol. 7, no. 27, pp. 14785–14792, 2015.

[4] D. S. Vidhya Delicia Perlin Rebelo, M. D. Cecilia Jane, S. Linford William Fernandes, and M. Clarissa Joella Costa, "Obstacle Detection using Ultrasonic Sensors," IJIRST- International J. Innov. Res. Sci. Technol., vol. 2, no. 11, pp. 316–320, 2016, [Online]. Available: www.ijirst.org.

[5] M. Engin and D. Engin, "Path planning of line follower robot," EDERC 2012 - Proc. 5th Eur. DSP Educ. Res. Conf., pp. 1–5, 2012, doi: 10.1109/EDERC.2012.6532213.

[6] https://www.farmmanagement.pro

Contributor
Moshiour Rahman Prince
Neaz Mahmud
Abdullah Al Forkan

Appendices :

Arduino Code

```
#include <NewPing.h>

const int TRIG = A1 ;
const int ECHO = A0 ;

int val;
int junctionx = 0 ;
int junctiony = 0 ;
bool East, West, North, South ;
unsigned long lastMillis1;
unsigned long lastMillis2;
int ena = 5 ;
int enb = 6;
int in1 = A2 ;
int in2 = A3 ;
int in3 = A4 ;
int in4 = A5 ;
int IRRIGHT = 7 ;
int IRLEFT = 8 ;
int x1 = 0;
int x2 = 2;
int y1 = 0;
int y2 = 1;

int a = -1; //if y2 positive . -1 if
y2 negative
int b = -1 ;//if x2 positive . -1 if
x2 negative

void setup()
{
  East = true;
  pinMode(7, INPUT);
  pinMode(8, INPUT);
  pinMode(ena, OUTPUT); //Motor
  pinMode(in1, OUTPUT); //Motor
  pinMode(in2, OUTPUT);//Motor
  pinMode(enb, OUTPUT);//Motor
  pinMode(in3, OUTPUT);//Motor
  pinMode(in4, OUTPUT);//Motor

  lastMillis1 = millis();
  lastMillis2 = millis();

  pinMode(TRIG, OUTPUT);
  pinMode(ECHO, INPUT);
  Serial.begin(9600);
}



void loop() {
  if (digitalRead(7) == HIGH &&
digitalRead(8) == HIGH)
  {
    val = 1;
  }
  else if (digitalRead(7) == LOW &&
digitalRead(8) == HIGH)
  {
    val = 3;
  }

  else if (digitalRead(7) == HIGH &&
digitalRead(8) == LOW)
  {
    val = 2;
  }

  else if (digitalRead (7) == LOW &&
digitalRead (8) == LOW)
  {
    if ((millis() - lastMillis1) >=
2000  && East == true)
    {
      val = 4;
      if (junctionx == x1)
      {
        LLeft();
        East = false;
        North = true;
      }
      else if (junctionx == x2)
      {
        if ((a == -1) || (b==1))
        {
          LLeft();
          East = false;
          North = true;
        }
        else if ((a ==1) || (b==-1))
        {
          RRight();
          East = false;
          North = true;
        }
      }
    }

    else if ((millis() - lastMillis2)
>= 2000  && North == true)
    {
      val = 5;
      if (junctiony == y1)
      {
        if (b==1)
        {
          RRight();
        East = true;
        North = false;
        }
```

```
        else if (b==-1)
        {
          LLeft();
        East = true;
        North = false;
        }

      }
      else if (junctiony == y2)
      {
        stopped();
        East = true;
        North = false;
      }
    }
    else {
      val = 1;
    }
  }


  switch (val)
  {
    case 1:
      digitalWrite(in1, LOW);
      digitalWrite(in2, HIGH);
      digitalWrite(in3, LOW);
      digitalWrite(in4, HIGH);
      analogWrite(enb, 110);
      analogWrite(ena, 110);
      Serial.print("Going Fronttt");
      Serial.println('\n');
      break;

    case 2:
      digitalWrite(in1, LOW);
      digitalWrite(in2, HIGH);
      digitalWrite(in3, HIGH);
      digitalWrite(in4, LOW);
      analogWrite(enb, 100);
      analogWrite(ena, 90);
      Serial.print("Going Left");
      Serial.println('\n');
      break;

    case 3:
      digitalWrite(in1, HIGH);
      digitalWrite(in2, LOW);
      digitalWrite(in3, LOW);
      digitalWrite(in4, HIGH);
      analogWrite(enb, 90);
      analogWrite(ena, 100);
      Serial.print("Going Right");
      Serial.println('\n');
      break;

    case 4 : //Number of junction
needed to cross
      digitalWrite(in1, LOW);
      digitalWrite(in2, HIGH);
      digitalWrite(in3, LOW);
      digitalWrite(in4, HIGH);
      analogWrite(enb, 110);
      analogWrite(ena, 110);
      Serial.print("Going Fronttt");
      Serial.println('\n');
      junctionx = junctionx + 1 ;
      Serial.println('\n');
      Serial.print("X");
      Serial.println(junctionx);
      Serial.println('\n');
      lastMillis1 = millis();
      break;


    case 5:
      digitalWrite(in1, LOW);
      digitalWrite(in2, HIGH);
      digitalWrite(in3, LOW);
      digitalWrite(in4, HIGH);
      analogWrite(enb, 110);
      analogWrite(ena, 110);
      Serial.print("Going Fronttt");
      Serial.println('\n');
      junctiony = junctiony + 1 ;
      Serial.println('\n');
      Serial.print("Y");
      Serial.println(junctiony);
      Serial.println('\n');
      lastMillis2 = millis();
      break;
  }
}
void LLeft ()
{
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
  delay(2500);

  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  analogWrite(enb, 110);
  analogWrite(ena, 110);
  delay(700);

  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  analogWrite(enb, 100);
  analogWrite(ena, 90);
  Serial.print("Taking left turn");
  Serial.println('\n');
  delay(1600);

}
```

```
void RRight ()
{
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
  delay(2500);

  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  analogWrite(enb, 110);
  analogWrite(ena, 110);
  delay(700);

  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  analogWrite(enb, 100);
  analogWrite(ena, 90);
  delay(1850);
  Serial.print("Taking right turn");
  Serial.println('\n');
}

void stopped()
{
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
  Serial.print("Stopped");
  Serial.println('\n');
  delay(5000);
}

void ultraac()
{
  int duration, distance;
  digitalWrite(TRIG, HIGH);
  delay(1);
  digitalWrite(TRIG, LOW); // Measure
the pulse input in echo pin
  duration = pulseIn(ECHO, HIGH);//
Distance is half the duration devided
by 29.1 (from datasheet)
  distance = (duration / 2) / 29.1;
  Serial.write(distance);
  Serial.print("Distance");
  Serial.println(distance);
  //allows all serial sent to be
received together

}

void ultradeac()
{
  int duration, distance;
  digitalWrite(TRIG, HIGH);
  delay(1);
  digitalWrite(TRIG, LOW); // Measure
the pulse input in echo pin
  duration = pulseIn(ECHO, HIGH);//
Distance is half the duration devided
by 29.1 (from datasheet)
  distance = (duration / 2) / 2.91;
  Serial.print("Distance");
  Serial.println(distance);
  //allows all serial sent to be
received together

}
/*


      //Color sensor, frequency
reading from different colors
      digitalWrite(S2, LOW);
      digitalWrite(S3, LOW);
      frequency_red = pulseIn(OUT,
LOW);
      Serial.print("Red = ");
      Serial.print(frequency_red);
      Serial.print("; ");
      digitalWrite(S2, HIGH);
      digitalWrite(S3, HIGH);
      frequency_green = pulseIn(OUT,
LOW);
      Serial.print("Green = ");
      Serial.print(frequency_green);
      Serial.print("; ");
      digitalWrite(S2, LOW);
      digitalWrite(S3, HIGH);
      frequency_blue = pulseIn(OUT,
LOW);
      Serial.print("Blue = ");
      Serial.println(frequency_blue);

*/
```

Eidesstattliche Erklärung

Hiermit bestätige ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen sowie Hilfsmittel genutzt habe. Alle Ausführungen, die anderen Quellen im Wortlaut oder dem Sinn nach entnommen wurden, sind deutlich kenntlich gemacht. Außerdem versichere ich, dass die vorliegende Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Prüfungsleistung war.

Affidavit

I hereby confirm that I have written this paper independently and have not used any sources or aids other than those indicated. All statements taken from other sources in wording or sense are clearly marked. Furthermore, I assure that this paper has not been part of a course or examination in the same or a similar version.

Moshiour Rahman Prince	19.06.22 Soest,	
Name, Vorname	Ort, Datum	Unterschrift
Last Name, First Name	Location, Date	Signature

Eidesstattliche Erklärung

Hiermit bestätige ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen sowie Hilfsmittel genutzt habe. Alle Ausführungen, die anderen Quellen im Wortlaut oder dem Sinn nach entnommen wurden, sind deutlich kenntlich gemacht. Außerdem versichere ich, dass die vorliegende Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Prüfungsleistung war.

Affidavit

I hereby confirm that I have written this paper independently and have not used any sources or aids other than those indicated. All statements taken from other sources in wording or sense are clearly marked. Furthermore, I assure that this paper has not been part of a course or examination in the same or a similar version.

Mahmud, Neaz	Lippstadt, 19.06.2022	
Name, Vorname	Ort, Datum	Unterschrift
Last Name, First Name	Location, Date	Signature

Eidesstattliche Erklärung

Hiermit bestätige ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen sowie Hilfsmittel genutzt habe. Alle Ausführungen, die anderen Quellen im Wortlaut oder dem Sinn nach entnommen wurden, sind deutlich kenntlich gemacht. Außerdem versichere ich, dass die vorliegende Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Prüfungsleistung war.

Affidavit

I hereby confirm that I have written this paper independently and have not used any sources or aids other than those indicated. All statements taken from other sources in wording or sense are clearly marked. Furthermore, I assure that this paper has not been part of a course or examination in the same or a similar version.

Forkan, Abdullah Al	Hamm, 19.06.2022	Forkan
Name, Vorname	Ort, Datum	Unterschrift
Last Name, First Name	Location, Date	Signature