



Voice Controlled Robot

By

Nouman Salameen
Anas Al-shweiki

Supervisor

Dr. Yousef Sweiti

*Submitted to the College of Engineering
in partial fulfillment of the requirements for the
Bachelor degree in Mechatronics Engineering*

Palestine Polytechnic University

May 2019

Palestine Polytechnic University
Collage of Engineering
Mechanical Engineering Department
Hebron – Palestine

Voice Controlled Robot

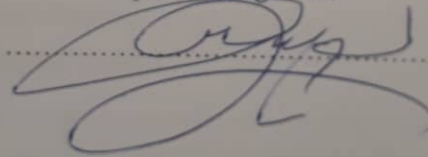
Project Team:

Anas T. Al-Shweiki

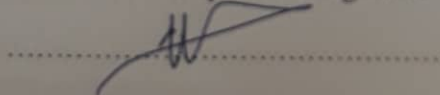
Nouman Al-Salameen

Submitted to the Collage of Engineering
In partial fulfillment of the requirements for the
Bachelor degree in Mechatronics Engineering.

Supervisor Signature



Chair of the Department Signature



2019

Table of Contents

Chapter 1: Introduction.....	3
1.1 Introduction	3
1.2 Problem definition	3
1.3 Recognition of the need	4
1.4 Expected output.....	4
1.5 Methodology.....	4
1.6 Conceptual design.....	5
1.7 System Architecture.....	6
1.7.1 Control Unit.....	6
1.7.2 Robot Unit.....	7
1.8 General Flow Chart.....	8
1.9 Budget.....	9
Chapter 2: Mechanical Design of the Mobile Robot.....	10
2.1 Introduction.....	10
2.2 Design Requirement.....	11
2.3 Design through SolidWorks and Material Selection.....	11
2.4 Mechanical Component.....	12
2.5 Structure and Load Analysis.....	14
Chapter 3: Mathematical model	17
3.1 Kinematic Model.....	17
3.2 Dynamic Model.....	20
Chapter 4: Electrical design Wheel Mobile Robot.....	26
4.1 Motor Selection and Driving Circuit.....	26
4.2 Sensors Selection.....	29
4.3 Communication Circuits Selection.....	31
4.4 Processing unit selection.....	32
4.5 Battery Selected.....	33
Chapter 5: Control System Design for the Robot Motion.....	34
5.1 Introduction.....	34
5.2 BitVoicer Software.....	34
5.3 Path planning.....	35
5.3.1 Matlab/Simulink Software.....	35
5.3.2 Tracker Design.....	36

Chapter 6: Planning	39
6.1 Planning problem.....	39
6.2 Path and timing law.....	39
6.3 Differential flatness.....	40
6.4 Path planning.....	41
6.5 Planning via Cartesian polynomials.....	41
Chapter 7: Practical work and experiment	42
7.1 Introduction	42
7.2 normal environments and noisy environment	43
Chapter 8: Conclusion and future work	44
7.1 Conclusion	44
7.2 Future work	44
References and resources	45

List of Figures

Figure 1.1: Conceptual design for WMR.....	5
Figure 1.2: Control Unit.....	6
Figure 1.3: Hardware part	7
Figure 1.4: Wheel Mobile Robot Flow Chart.....	8
Figure 2.1: The Selected Robot Frame	10
Figure 2.2: The Solid Works Interface	12
Figure 2.3: The Robot Base	13
Figure 2.4: Stress Distribution	15
Figure 2.5: The Stress distribution Location.....	15
Figure 2.6: The Resulted Maximum Deflection Due to Load	16
Figure 3.1: Wheel Mobile Robot.....	18
Figure 4.1: Desired robot motion profile	27
Figure 4.2: MC33926Motor Driver	28
Figure 4.3: Drivers Connection	29
Figure 4.4: Optical Encoder.....	30
Figure 4.5: Hc-SR04 Sensor.....	30
Figure 4.6: Wi-Fi RN-171 Module.....	31
Figure 4.7: Arduino Uno ATmega 328.....	32
Figure 5.1: BitVoicer Interface	35
Figure 5.2: Block diagram of the wheels motion control.....	36
Figure 5.3: Trajectory tracking simulation results	37
Figure 5.4: Input torque levels applied to right and left wheels.....	38

List of Tables

Table 1.1: Budget.....	9
Table 4.1: Specification of MC33926 Motor Driver.....	29
Table 4.2: Encoder Specification.....	30
Table 4.3: HC-SR04 Specification.....	30
Table 4.4: The comparison between communications method.....	31
Table 4.5: RN-171 Wi-Fi specification.....	31
Table 4.6: general comparison between microcontrollers.....	32
Table 4.7: specification of microcontroller.....	33
Table 4.8: Robot Components Consumption.....	33
Table 7.1: experiment result in normal environment.....	43
Table 7.1: experiment result in noise environment.....	43

Abstract

The purpose of robotics in commercial intention has come to be quite essential for executing challenging work into more conveniently simple way. There are a lot of researches working on to enhance the connection between humans and robot.

The projects' purpose is to control a WMR through voice commands which analyzed using a software called BitVoicer which identifies the command and turns it into a form that the microcontroller can understand to perform the given command as it's given from the user.

Also we used MATLAB to make a simulation for the motors we used in the project and to do so we used a certain feature in the MATLAB called Simulink to get the needed torque for each motor and the needed voltage with the designed control system.

We have experimented the WMR in two environments the first in the normal environment without noise which gives a perfect identifying to the user's command and the second environment when there's noise which affects the identification of the user's commands so the BitVoicer gives about 80% of the commands applied as it should be performed.

الملخص :

يعمل الكثير من الباحثين على تطوير طرق تؤدي إلى تسهيل العمليات المعقدة وتبسيطها بحيث تكون سهلة التعامل ، ومن هذه الطرق هي التحكم عن طريق الأوامر الصوتية و يعتبر هذا المشروع امتدادا لما تم التوصل إليه من نتائج أبحاث الباحثين في هذا المجال .

يقوم مبدأ العمل على أن يقوم المستخدم بإعطاء أوامر صوتية يكون الروبوت على دراية مسبقة بها ثم يقوم بتحليلها وتحويلها إلى صيغة يستطيع المتحكم الدقيق فهمها وتحويلها إلى إشارات كهربائية يتم تحويلها إلى الفعل الذي طلبه المستخدم .

في خضم هذا المشروع قد قمنا باختبار الروبوت في بيئتين مختلفتين : أولاهما هي البيئة الطبيعية والتي أعطت نتائج ممتازة من ناحية التعرف الصوتي وكم الثقة في الإشارات الصوتية المتلقاة من المستخدم وعلى الناحية الأخرى عند اختبار الروبوت في البيئة الثانية ألا وهي بيئة الضجيج قد لوحظ أن الاستجابة من قبل المحلل الصوتي تقل بشكل كبير نسبيا مما يستدعي إعادة الأمر مرة أخرى من قبل المستخدم ، حتى عندما يستطيع الروبوت التعرف على الأمر من قبل المستخدم فإن الثقة المؤكدة للأمر الذي استقبله المحلل الصوتي لا تزال ضعيفة نسبيا مقارنة بالبيئة الطبيعية .

Chapter 1

Introduction

This chapter deals with a general introduction of the project and then talks about the problem definition, and the reason why this topic is chosen? And how the project is implemented, also this chapter considers the conceptual design for this project and the expected costs of this project. We will also talk about the outputs of this project.

1.1 Introduction

Since the nineteenth century huge efforts were made by scientists and developers to create new methods of controlling things more efficiently and with better results. Control theory made significant steps in the next century, new mathematical techniques made it possible to control, more accurately, and significantly more complex dynamical systems. These techniques include developments in optimal control in the 1950's and 1960's, followed by progress in robust and optimal control methods in the 1970's and 1980's. One of those control methods was the control through the voice commands which was developed by Bell Laboratories in 1952. This way of control makes it easier to the humans to interact with machines within the lowest effort and the best performance we can get.

1.2 Problem definition

In this project the WMR is controlled by voice command and this method allows the user to interact with the robot in terms which are familiar to most people. The advantages of speech activated robot are hands-free and fast data input operations, the proposed robot is capable of understanding the meaning language command.

1.3 Recognition of the need

The main purpose of this work is to design and construct a differential drive robot which is controlled by voice commands. The robot should be able to work based on the supervisor commands. After the starting command the robot should be able to receive some predefined command. It should also be small so it doesn't require much space which makes it easier to carry and flexible to move at the small space available at working area of robot. It should be simple to change the controller and read the data from its sensors interest for spectators.

Also, the robot should:

- be able to interact with the supervisor.
- be able to define the obstacles that face it.
- be small so it is easy to carry.
- be stable and not make any unwanted movements.
- be safe and don't harm anybody in its environment.

1.4 Expected output

The output of this project is to completely design and implement a small mobile robot (prototype) should be able to receive and execute the commands of the supervisor such as moving from one place to another with a predefined distance and with the fitted angle. And should be able to avoid the obstacles that face it to ensure the safety of the others and protect himself.

1.5 Methodology

In this project, we will control the range of mechanical components (main frame, wheel, etc.), electrical components (DC-motors, encoder, capacitors...etc.), and electronic components (bitvoicer server, spark fun ...etc.) using the modern method of sound waves and completely combination for above components by using robust tracking controller. Where this project will address many of the problems, including the problem of accuracy in achieving what is required.

With this project we will achieve the required skills of Mechatronics Engineering design.

1.6 Conceptual design

The Microphone receives the command of the user and sends it to the bit voicer server through a WIFI module to analyze it, when the command is analyzed the controller translates it to the needed outcome and sends the data to the motor drivers with the needed voltage and turns it to the connected wheels as shown in Figure (1.1)

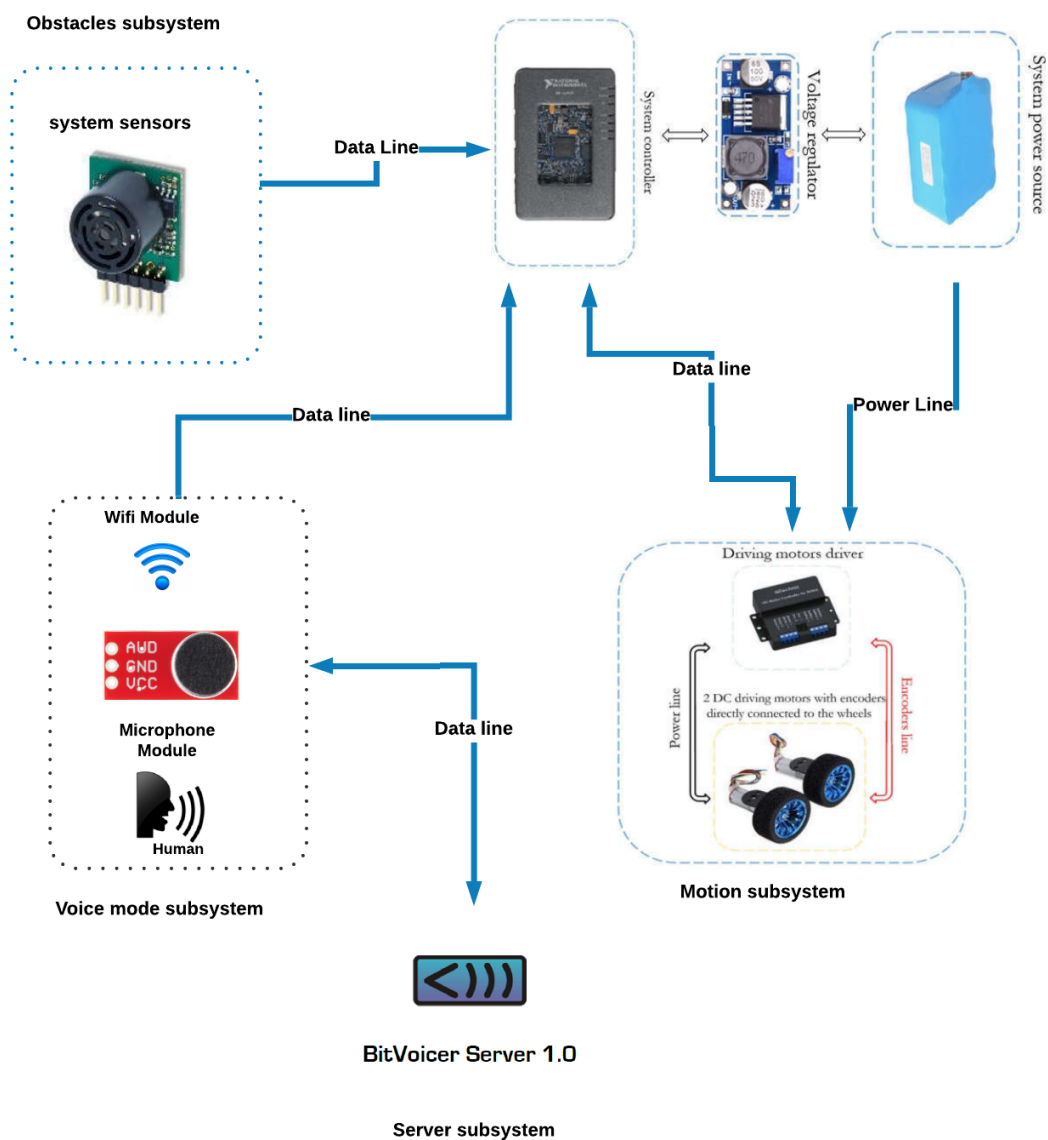


Figure1.1: Conceptual design for WMR

1.7 System Architecture

The system architecture consists of mainly two blocks:

1.7.1 Control unit

When any person pronounces the activation command, the microphone interfaced to control unit will fetch the audio signal and send the signal to Microcontroller to process it, the received speech signal will be analyzed, operated and corresponding command will be given as output as shown in Figure (1.2).

The output from the BitVoicer server will be transmitted to the robot unit via Wifi module interfaced to control unit.



Figure 0.2: Control Unit

1.7.2 Robot Unit

The Wi-Fi module sends the voice command to speech recognizer to analyze it, when the command is analyzed the Arduino translates it to the needed outcome and sends the data to the motor drivers with the needed voltage and turns it to the connected wheels and the encoders perform as a feedback for the robot to watch the accomplishment of the needed output as illustrated in Figure (1.3).

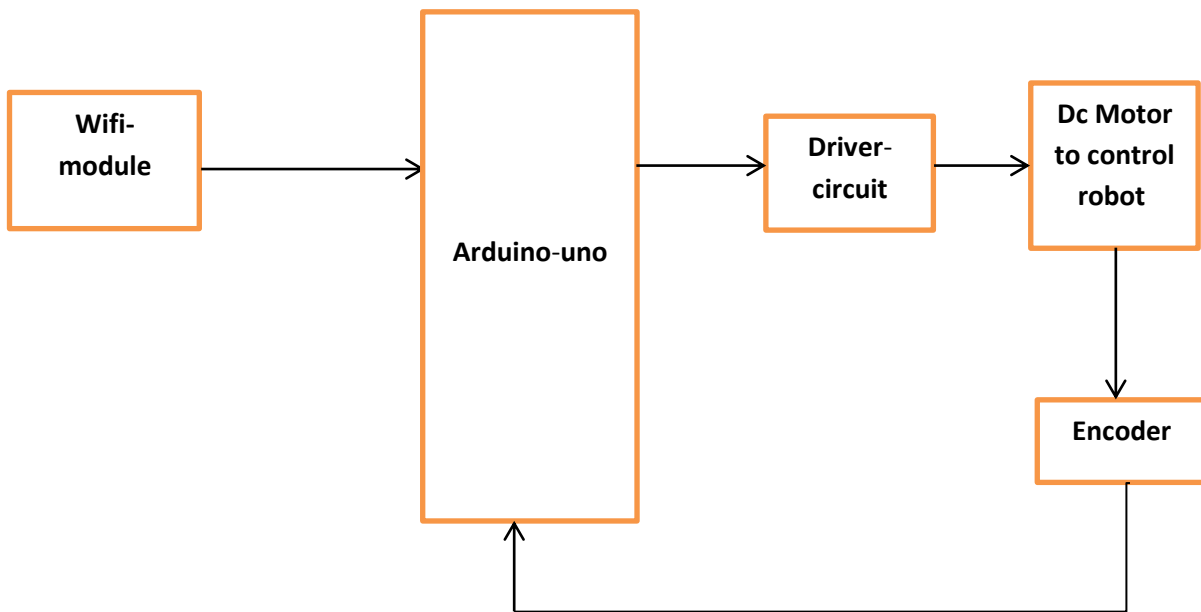


Figure 0.3: Hardware part

1.8 General Flow Chart

Shown in Figure (1.4) below is the general flow chart that shows the operation of the mobile robot when receiving a voice command to execute some task.

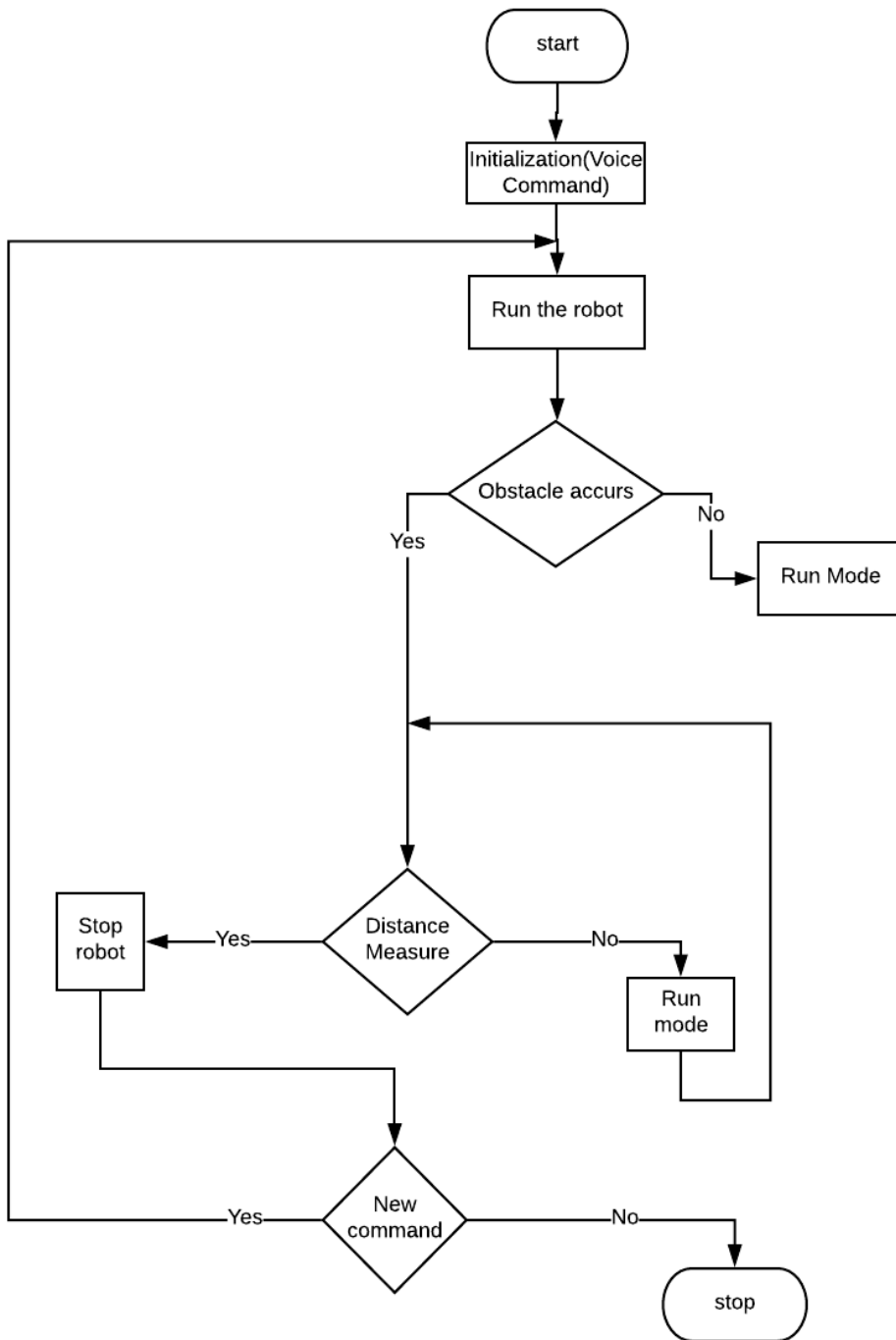


Figure 1.4: Wheel Mobile Robot Flow Chart

1.9 Budget

The expected necessary to implement this project is shown in Table (1.1)

Table 1.1: Budget

Tool &device	No.	Piece Price (NIS)	Price (NIS)
Motor	2	90	180
Frame	1	150	150
Arduino	1	90	90
Bit voicer	1	150	150
Motor driver	2	80	160
Encoder	2	100	200
Spark fun	1	1	350
Microphone	1	60	60
Capacitors, Leds, ..,etc.	Multiple	-	20
Batteries	4	20	20
Others	-	500	500
Total			1900

Chapter 2

Mechanical Design of the Mobile Robot

2.1 Introduction

In this chapter we are going to discuss the mechanical design of the robot frame shown in Figure (2.1), calculation of each component of the robot in term of strength, geometry, durability and material properties will be explained. Thus, robot main frame will operate in the system without any failure or defect occur.

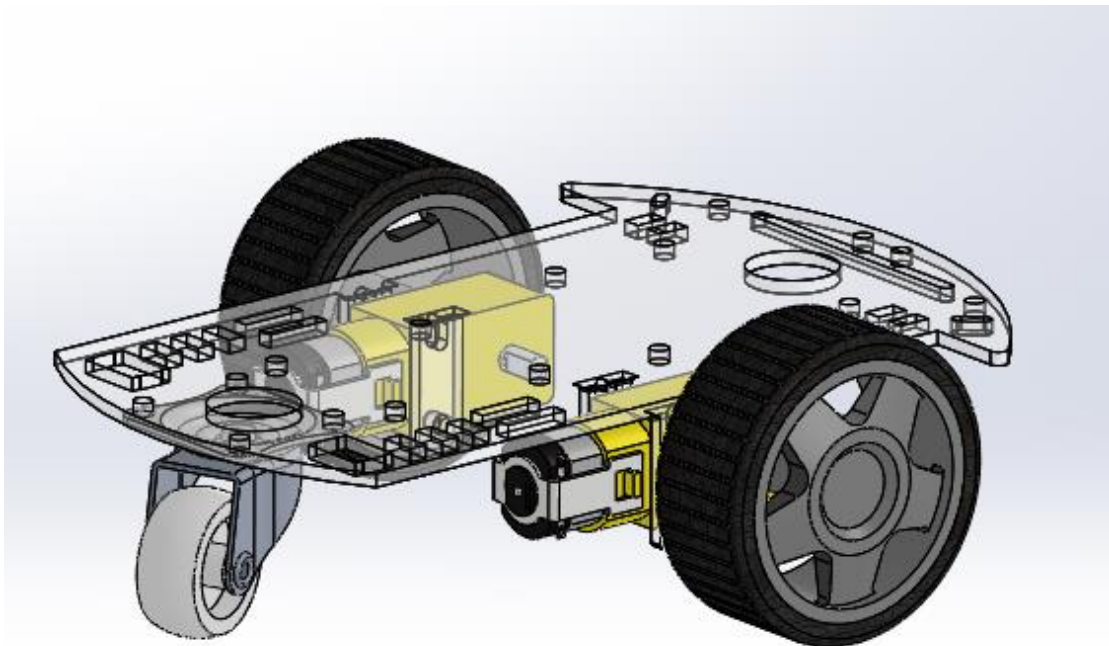


Figure 2.1: The Selected Robot Frame

2.2 Design Requirement

The first step in designing any project is to define the project requirement. In other words, choosing the specification which your project supposed to do. Then you have to make many designs and choose one of them which satisfy our requirements, but in our case a pre-manufactured model was chosen and through this chapter we will consider the analysis of this model. The robot requirements are:

1. Easy to carry.
2. Low weight.
3. Small size.
4. Flexible movement.
5. Stable and not make any unwanted movements.
6. Safe and do not harm anybody in its environment.

2.3 Design through SolidWorks and Material Selection

SolidWorks is a solid modeling computer-aided design (CAD) and computer-aided engineering (CAE) computer program that runs on Microsoft Windows. SolidWorks is published by Dassault Systems. It's a three dimensional interactive application, and it is able to make part design for every part of any machine or project, and combining those parts to produce the assembled design or project.

It also allows the ability to estimate the stress and strain and load analysis for the machine or the project. Figure (2.2) shows the SolidWorks interface.

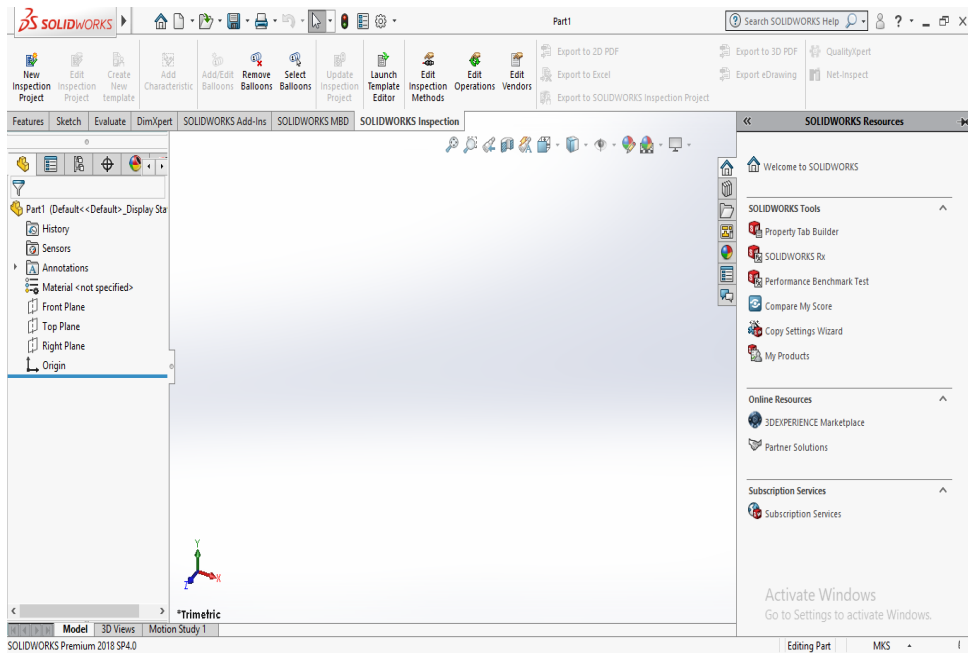


Figure 2.2: The SolidWorks Interface

2.4 Mechanical Components

1. The main frame

is the robot base which is the Achrylic frame which forms the body of the robot which carries all the mechanical and electrical components. The main frame should be light and have enough space to satisfy the requirements and to contain the other components. As shown in Figure (2.3) an Achrylic material is used to manufacture the base.

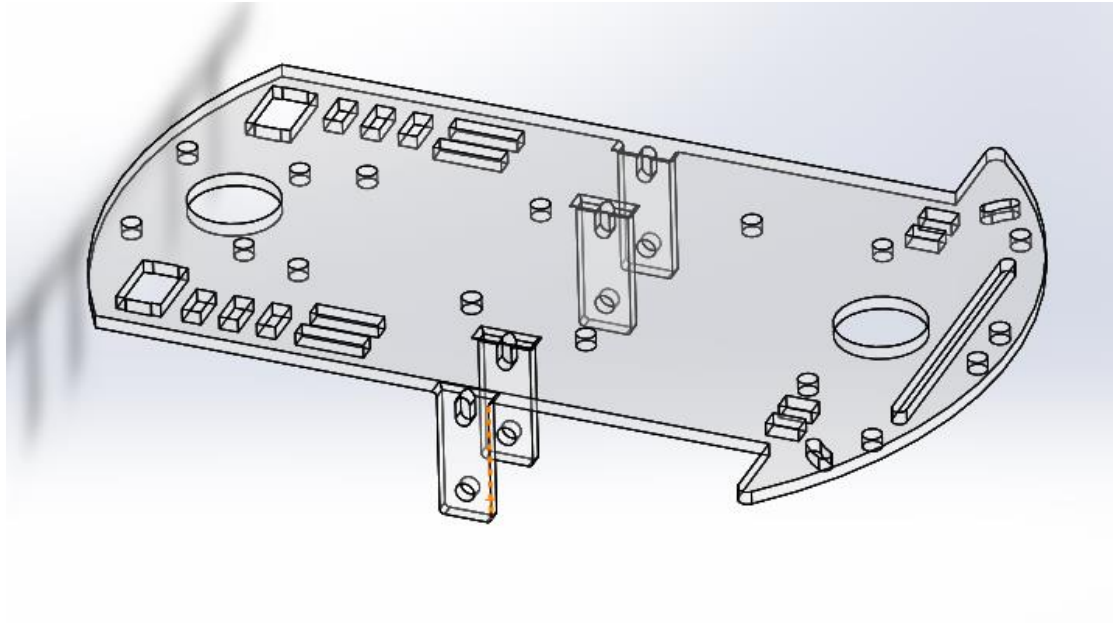


Figure 2.3: The Robot Base

The base dimensions are (200mm×150mm) which offers enough space for the other electrical connections on the base surface.

2. Wheels

The wheel is a mechanical part used to allow heavy objects to be moved easily facilitating movement or transportation while supporting a load or performing labor in machines. Our robot has two types of wheels, the driving wheels and caster wheel. The driving wheels are used to drive the robot and make steering. Driving wheels must be strong and rigid enough to carry the robot load, the following specifications can be selected for the driving wheels:

- 1- Diameter 65 mm
- 2- Tire width 22 mm
- 3- Tire cross-section (height) 23mm
- 4- Width on the wheel bearing 29mm
- 5- Inner diameter 2.44mm bearings
- 6- Weight per wheel 50 g
- 7- Plastic rim black
- 8- Rubber tires

3. The caster wheel

Is used to enable relatively easy rolling movement and stability of the robot and to ensure enough friction with the ground, the following specification can be selected for the caster wheel

- 1- Material: Aluminum alloy (the hub), Rubber (the outer circumference)
- 2- Outside Diameter: 30 mm
- 3- Perimeter: 170 mm
- 4- Thickness: 13 mm

2.5 Structure and Load Analysis

In this section we will make the stress and deflection analysis to find the maximum stress and maximum deflection, which act when the maximum possible load placed on the robot, the maximum load is 1 Kg. The stress and deflection analysis are as follows.

The base of the mobile robot is made of Acrylic which has a yield strength of 45MPa. The design process will be applied when the mechanical structure is in its critical configurations, in other words the configurations those yield the maximum stress on the structure.

The resulted stress distribution is illustrated in Figure (2.4), and Figure (2.5) shows location where the maximum stress occurs, the maximum resulted shear stress on the plate is 26.7MPa.

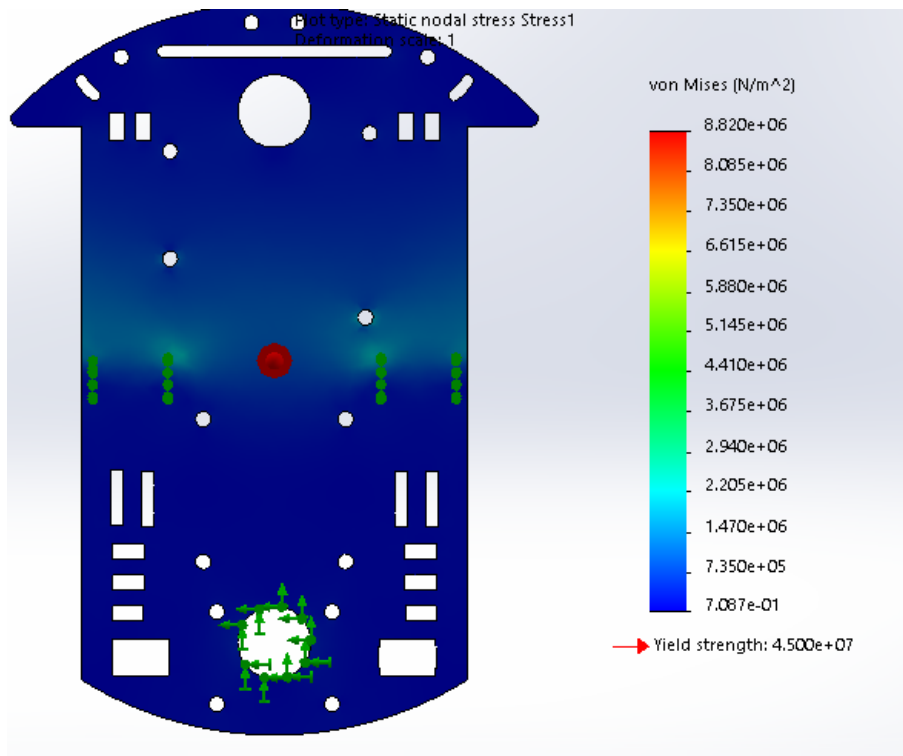


Figure 2.4: Stress Distribution

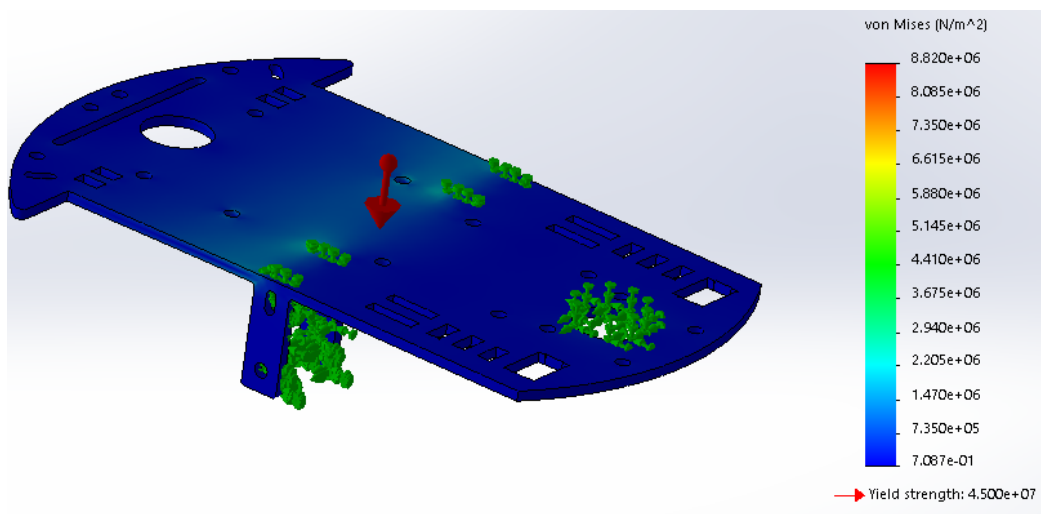


Figure 2.5 The Stress distribution Locations

About the deflection, the resulted maximum deflection occurred in the far area from the wheels due to no support exists, which means that we should avoid putting load to minimize the deflection value. Figure (2.6) describes the resulted deflection due to the load.

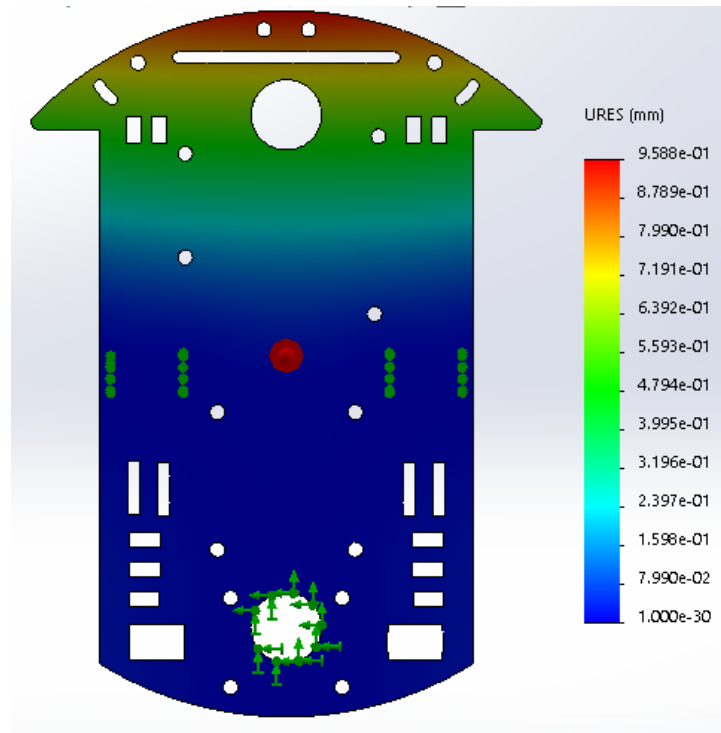


Figure 2.6: The Resulted Maximum Deflection Due to Load

Where the maximum deflection was 9.7 mm at the shown area due to the lack of supports on this area.

Chapter 3

Mathematical model

This chapter focuses on kinematics and dynamics of the wheel mobile robot (WMR) in order to describe the robot motion within its environment.

Lagrange dynamic approach is used to determine the robot dynamics model which will be used later to design the required control system to control robot position dynamics under a given voice command.

3.1 Kinematic Model

1-Kinematic Constraints of the Wheel Mobile Robot (WMR)

The motion of a WMR is characterized by two non-holonomic constraint equations, which are obtained by two main assumptions:

- No lateral slip motion: This constraint simply means that the robot can move only in a curved motion (forward and backward) but not sideward. In the robot frame, this condition means that the velocity of the center-point of A is zero along the lateral axis are shown in Fig (3.1):

$$\dot{y}_r = 0 \quad (3.1)$$

Expanding Equation (3.1) gives

$$-\dot{x}_A \sin \theta + \dot{y}_A \cos \theta = 0 \quad (3.2)$$

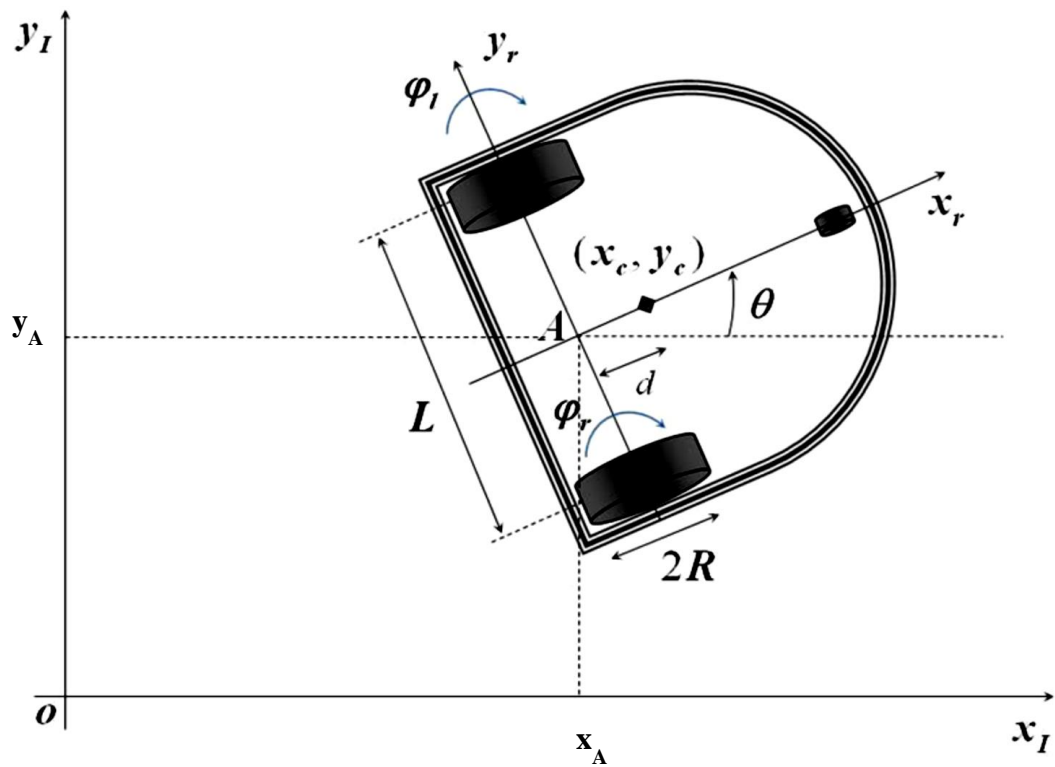


Figure31: Wheel Mobile Robot

- Pure rolling constraint: The pure rolling constraint represents the fact that each wheel maintains a one contact point with the ground, i.e., the translational velocity v of the center point of a wheel in the direction of rolling is related to its angular velocity as

$$v_{wheel} = R \dot{\theta}_{wheel} \quad (3.3)$$

where, R is the radius of the wheel, and $\dot{\theta}_{wheel}$ is the angular velocity of the wheel

Robot Kinematics

Kinematic modeling is the study of the motion of mechanical systems without considering the forces that affect the motion. For the WMR, the main purpose of kinematic modeling is to represent the robot velocities as a function of the driving wheels' velocities along with the geometric parameters of the robot.

Now, in view of the assumed constraints (pure rolling with no lateral slip) the linear and translational velocity of the robot frame (centered at A) can be expressed as

$$\begin{Bmatrix} \dot{x}_A \\ \dot{y}_A \\ \dot{\theta} \end{Bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} v \\ \omega \end{Bmatrix} \quad (3.4)$$

Where v and ω are the linear and angular velocities of the WMR respectively. The linear velocity of the WMR in the Robot Frame is therefore the average of the linear velocities of the two wheels

$$v = \frac{R(\dot{\theta}_R + \dot{\theta}_L)}{2} \quad (3.5)$$

And the angular velocity of the WMR about the normal (z-axis) is

$$\omega = \frac{R(\dot{\theta}_R - \dot{\theta}_L)}{L} \quad (3.6)$$

Equations (3.5 and 3.6) can be inverted to get the inverse velocity equations as

$$\begin{Bmatrix} \dot{\theta}_R \\ \dot{\theta}_L \end{Bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & L/2 \\ 1 & -L/2 \end{bmatrix} \begin{Bmatrix} v \\ \omega \end{Bmatrix} \quad (3.7)$$

3.2 Dynamic Model

Dynamics is the study of the motion of a mechanical system taking into consideration the different forces that affect its motion unlike kinematics where the forces are not taken into consideration. The dynamic model of the WMR is essential for simulation analysis of the (WMR) motion and for the design of various motion control algorithms.

There are two methods to derive the dynamics model for the (WMR)

- 1-Newton-Euler method.
- 2-Lagrange method.

Lagrange method is chosen to derive the system model.

Lagrange dynamic approach:

Lagrange dynamic approach is a very powerful method for formulating the equations of motion of mechanical systems. This method, which was introduced by Lagrange, is used to systematically derive the equations of motion by considering the kinetic and potential energies of the given system.

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial L}{\partial \mathbf{q}} = \mathbf{Q} \quad (3.8)$$

Where $L=T-V$ is the Lagrangian function, T , is the kinetic energy of the system, V is the potential energy of the system, $\mathbf{q} = [\theta_R \quad \theta_L]^T$ is the generalized coordinates vector, and \mathbf{Q} is the generalized force vector.

The first step in deriving the dynamic model using the Lagrange approach is to find the kinetic and potential energies that govern the motion of the WMR. Furthermore,

since the WMR is moving in the $\{x_I, y_I\}$ plane, the potential energy of the WMR is considered to be zero.

Consider the WMR shown in Fig (3.1), the Kinetic energy of the robot can be expressed as follows:

$$T = \frac{1}{2}mv_c^2 + \frac{1}{2}I_c\dot{\theta}^2 + \frac{1}{2}I_o(\dot{\theta}_R^2 + \dot{\theta}_L^2) \quad (3.9)$$

Where,

m is the total mass of the robot (kg), I_c is the moment of inertia of the entire robot about the vertical axis passing through the center of mass of the robot (kg.m^2), I_o is the moment of inertia of combined drive motor (rotor) and wheel about the motor axis (kg.m^2), $\dot{\theta}_R$ and $\dot{\theta}_L$ are angular velocities of the right and left wheels about their rotation axes respectively (rad/s).

Now, the absolute instantaneous velocity (v_c) of the robot center of mass can be expressed in terms of its rectangular components as

$$v_c^2 = \dot{x}_c^2 + \dot{y}_c^2 \quad (3.10)$$

Thus, the total Kinetic energy of the WMR system becomes

$$T = \frac{1}{2}m(\dot{x}_c^2 + \dot{y}_c^2) + \frac{1}{2}I_c\dot{\theta}^2 + \frac{1}{2}I_o(\dot{\theta}_R^2 + \dot{\theta}_L^2) \quad (3.11)$$

Now, the components of velocity of point A, can be expressed in terms of $\dot{\theta}_R$ and $\dot{\theta}_L$ as:

$$\dot{x}_A = \frac{R}{2}(\dot{\theta}_R + \dot{\theta}_L)\cos\theta \quad (3.12)$$

$$\dot{y}_A = \frac{R}{2}(\dot{\theta}_R + \dot{\theta}_L)\sin\theta \quad (3.13)$$

$$\dot{\theta} = \frac{R(\dot{\theta}_R - \dot{\theta}_L)}{L} \quad (3.14)$$

Where

\dot{x}_A : Absolute velocity component of point A in direction of X-axis (m/s).

\dot{y}_A : Absolute velocity component of point A in direction of Y-axis (m/s).

Equations (3.12) to (3.14) can also be written in compact form representing the absolute velocity of robot frame (with origin at A) as follows

$$V = \frac{R}{2} \begin{bmatrix} \cos \theta & \cos \theta \\ \sin \theta & \sin \theta \\ 2/L & -2/L \end{bmatrix} \begin{Bmatrix} \dot{\theta}_R \\ \dot{\theta}_L \end{Bmatrix} \quad (3.15)$$

Where $V = [\dot{x}_A \quad \dot{y}_A \quad \dot{\theta}]^T$, now we need to find the absolute velocity of point C, taking into consideration the distance (d) between A and C, it can be easily found that

$$\dot{x}_c = \dot{x}_A - d\dot{\theta} \sin \theta \quad (3.16)$$

$$\dot{y}_c = \dot{y}_A + d\dot{\theta} \cos \theta \quad (3.17)$$

Where

\dot{x}_c : Absolute velocity component of point C in direction of X-axis (m/s).

\dot{y}_c : Absolute velocity component of point C in direction of Y-axis (m/s).

d : distance between points A and C (m).

Therefore, the total Kinetic energy of the mobile robot in terms of $\dot{\theta}_R$ and $\dot{\theta}_L$ can be found by substituting equations (3.12-14) into Equations (3.16, 3.17) and then substituting the result into equation (3.11) which gives

$$T = \left(\frac{mR^2}{8} + \frac{(I_C + md^2)R^2}{2L^2} + \frac{I_o}{2} \right) \dot{\theta}_R^2 + \left(\frac{mR^2}{8} + \frac{(I_C + md^2)R^2}{2L^2} + \frac{I_o}{2} \right) \dot{\theta}_L^2 + \left(\frac{mR^2}{4} - \frac{(I_C + md^2)R^2}{L^2} \right) \dot{\theta}_L \dot{\theta}_R \quad (3.18)$$

Now, Lagrange equations describing the robot dynamics becomes

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_R} \right) + \frac{\partial L}{\partial \theta_R} = T_R - f \dot{\theta}_R \quad (3.19)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_L} \right) + \frac{\partial L}{\partial \theta_L} = T_L - f \dot{\theta}_L \quad (3.20)$$

Where

T_R and T_L are right and left actuation torque, respectively in (N.m), $f \dot{\theta}_R$ and $f \dot{\theta}_L$ are viscous friction values of right and left wheel-motor system, respectively in (N.m).

Dynamic equations of motion of the mobile robot resulted from Equations (3.19, 3.20) can be expressed as

$$a\ddot{\theta}_R + b\ddot{\theta}_L = T_R - f \dot{\theta}_R \quad (3.21)$$

$$b\ddot{\theta}_R + a\ddot{\theta}_L = T_L - f \dot{\theta}_L \quad (3.22)$$

Where

$$a = \left(\frac{mR^2}{4} + \frac{(I_C + md^2)R^2}{L^2} + I_o \right) \quad (3.23)$$

$$b = \left(\frac{mR^2}{4} - \frac{(I_C + md^2)R^2}{L^2} \right) \quad (3.24)$$

Equations (3.21) to (3.22) we can find from

$$\begin{aligned} T_R &= a\ddot{\theta}_R + b\ddot{\theta}_L + f\dot{\theta}_R \\ T_L &= b\ddot{\theta}_R + a\ddot{\theta}_L + f\dot{\theta}_L \end{aligned}$$

Equations (3.21) to (3.22) can be written in compact form as

$$\mathbf{M}\ddot{\boldsymbol{\theta}} + \mathbf{N}\dot{\boldsymbol{\theta}} = \mathbf{u} \quad (3.25)$$

Where,

$\boldsymbol{\theta} = \begin{Bmatrix} \theta_R \\ \theta_L \end{Bmatrix}$ is the wheels angular displacements vector

$\mathbf{u} = \begin{Bmatrix} T_R \\ T_L \end{Bmatrix}$ is the input vector (torques) driving the robot wheels

$\mathbf{M} = \begin{bmatrix} a & b \\ b & a \end{bmatrix}$ is the inertia matrix

$\mathbf{N} = \begin{bmatrix} f & 0 \\ 0 & f \end{bmatrix}$ is the viscous damping matrix

The state space model corresponding to the mathematical model shown in Equation (3.25) can be obtained by assuming the following state vectors:

$$\begin{aligned} \mathbf{x}_1 &= \boldsymbol{\theta} \\ \mathbf{x}_2 &= \dot{\boldsymbol{\theta}} \end{aligned} \quad (3.26)$$

Which yields to the following state space model describing the dynamics of the robot motion

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx} \end{aligned} \quad (3.27)$$

Where,

$$\mathbf{x} = \begin{Bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{Bmatrix} = \begin{Bmatrix} \theta_R \\ \theta_L \\ \dot{\theta}_R \\ \dot{\theta}_L \end{Bmatrix}, \text{ is the state vector } \in \mathfrak{R}^4$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{2 \times 2} & \mathbf{I}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & -\mathbf{M}^{-1}\mathbf{N} \end{bmatrix} \text{ is the system matrix } \in \mathfrak{R}^{4 \times 4}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0}_{2 \times 2} \\ \mathbf{M}^{-1} \end{bmatrix} \text{ is the input matrix } \in \mathfrak{R}^{4 \times 2}$$

$$\mathbf{y} = \begin{Bmatrix} \dot{\theta}_R \\ \dot{\theta}_L \end{Bmatrix} \text{ is the output vector } \in \mathfrak{R}^2$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 2} \end{bmatrix} \text{ is the output matrix } \in \mathfrak{R}^{2 \times 4}$$

Chapter 4

Electrical design Wheel Mobile Robot

This chapter is concerned motor selection, driving circuit, battery selected and communication circuit's selection, also we will talk about processing the system's information.

4.1 Motor Selection and Driving Circuit

1-Driving motor selection:

The purpose of the WMR is to move from one point to another, and to ensure that purpose it is important to generate enough Torque with a sensible speed to drive the robot to its desired destination.

Our choice for the motors depends on the following requirements:

Mass of the WMR is equal 2 kg

Radius of drive wheel is equal 0.0325m

Separating distance between the driving wheels is equal to 0.15 m

The desired velocity equal 0.6 m/s

And the desired time to reach the maximum speed equals 0.5s

According to the above requirement, and adopting trapezoidal velocity profile for each wheel rotation (Figure 4.1) we obtain the desired linear acceleration of the robot as

$$a_c = v/t = 0.6/0.5 = 1.10\text{m/s}^2 \quad (4.1)$$

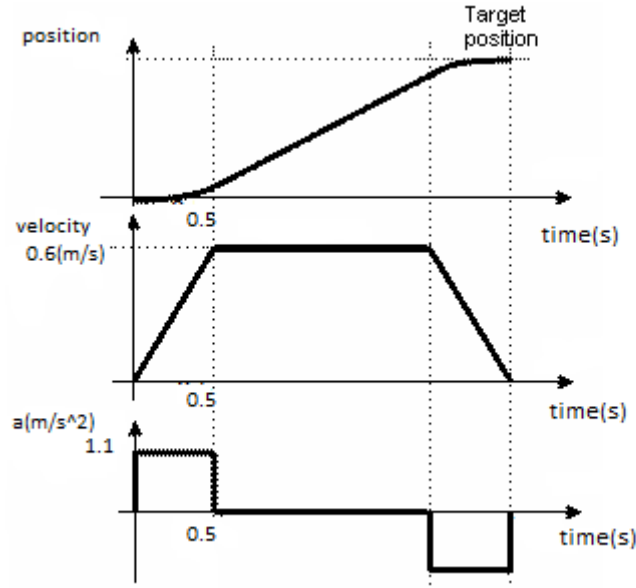


Figure (4.1) : Desired robot motion profile

The minimum value of the required coefficient of friction μ between the robot wheels and the ground to ensure rolling without slip can be computed by applying Newton's second law to the entire robot during linear acceleration as

$$F = \mu mg = ma_c \rightarrow \mu = a_c / g = 1.1 / 9.8 = 0.112 \quad (4.2)$$

The required motor torques needed to move the robot can be roughly estimated using from the robot dynamic model given by Equations (3.21 and 3.22).

$$T_R = a\ddot{\theta}_R + b\ddot{\theta}_L + f\dot{\theta}_R = 0.918 \text{ Nm}$$

$$T_L = b\ddot{\theta}_R + a\ddot{\theta}_L + f\dot{\theta}_L = 0.918 \text{ Nm} \quad (4.3)$$

To find the motor speed n in [rpm], we apply the following formula

$$n = \frac{v}{R} \times \frac{60}{2\pi} \quad (4.4)$$

That gives the robot wheel speed is equal to 176.38 rpm (which is equivalent to 18.46 rad/s), i.e., the selected motor should be integrated with a gear box to provide this steady speed at the computed operating torque.

The motor power required to drive each wheel for this robot can be compute as

$$P = T_R \dot{\theta}_R = T_L \dot{\theta}_L = 0.918 \times \frac{2\pi n}{60} = 18.11 \text{ w} \quad (4.5)$$

2-Motor Driver

Motor drivers is a device that is used to amplify the voltage signal coming from the Microcontroller to produce the needed voltage for running the electrical motor, drivers is used for preventing any damage for the other components of the robot system especially the Microcontroller from reverse current due to the motors rotation.

The driver is shown in figure (4.2) is selected with the specifications shown in table (4.1)

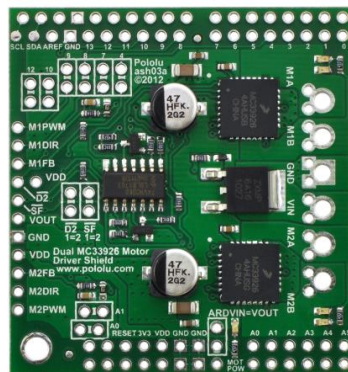


Figure (4. 2): MC33926Motor Driver

Table (4. 1): Specification of MC33926 Motor Driver

Operating voltage range	5-28 V
Output Current	1A Per Motor
PWM	Up to 20KHz

Pololu Dual MC33926 DC Motor Driver shown in figure (4.3) allows speed and direction control of two DC motors at the same time and shows the connection between the microcontroller and the motors through the driver.

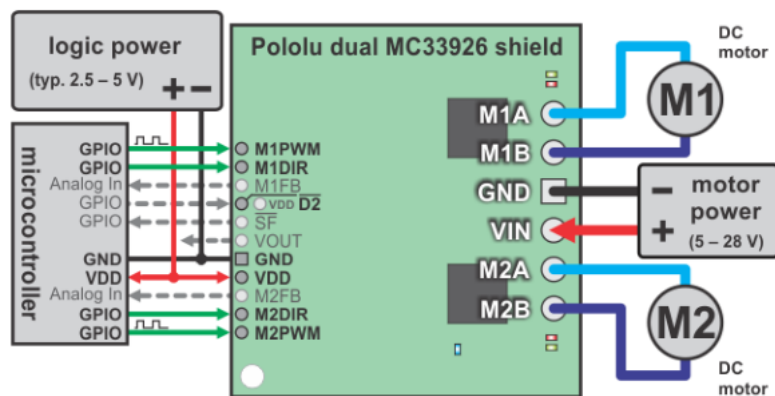


Figure (4. 3): Driver's Connection.

4.2 Sensors Selection

In this project we need two types of sensors, first type is displacement sensor and the second type is the range sensor.

1-Displacement Sensor

Displacement Sensors are used to measure the distance, in this project we are going to use an optical encoder to read the distance moved by the robot, and to offer a feedback of the position and the speed control of the robot, by compromising the wheel rotation with the PWM produced by Microcontroller, the encoder is shown in figure (4.4), and specification in table (4.2)

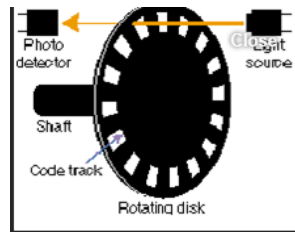


Figure (4.4): Optical Encoder

Table (4. 2): Encoder Specification

Input Voltage	5V
Pulses per revolution	100 pulses
Weight	0.05 kg

2-Range Sensor

Range sensor used to measure the range between the Robot and objects around it, it's important require for avoiding obstacles.

We choose Ultra Sonic Sensor shown in figure (4.5), with Specification in table (4.3)



Vcc- Connects to 5V of positive voltage for power
 Trig- A pulse is sent here for the sensor to go into ranging mode for object detection
 Echo- The echo sends a signal back if an object has been detected or not. If a signal is returned, an object has been detected. If not, no object has been detected.
 GND- Completes electrical pathway of the power.

Figure (4. 5): Hc-SR04 Sensor

Table (4. 3): HC-SR04 Specification

Working voltage	DC 5V
Working current	15 mA
Working frequency	40-50 Hz
Max Range	4 m
Min Range	2 cm
Measuring Angle	15 degree

4.3 Communication Circuits Selection

There are many ways that robot take tasks from human like: Wi-Fi communication, FM communication and Bluetooth communication, these communication ways is discussed in Table (4.4)

Table (4. 4): the comparison between communications method

Communication type	Range/m	Consumption energy	Cost	Options
Wi-Fi	Depends on internet	High	Medium	>25
FM	100	Medium	Medium	15
Bluetooth	1-100	High	Medium	>20

The way communication choice is Wi-Fi, his way is very good to send and receive information and has good technical details, when using this method, we need a strong internet access to give a large range, the Wi-Fi circuit shown in figure (4.6) with specification in table (4.5).



Figure (4.6): Wi-Fi RN-171 Module

Table (4. 5): RN-171 Wi-Fi specification

Type	RN 171 XV
Processor	32 bit
ADC	14-bit resolution
Input voltage	3.3 VDC
Operating temperature range	-40 c to +80 c

4.4 Processing unit selection

The processing unit system is a brain of robot, and allow the robot acts as a complete stand-alone system.

There are many type of microcontroller using to processing information like as: PIC, Arduino Uno and my RIO, these microcontrollers are discussed in Table (4.6)

Table (4. 6): general compression between microcontrollers

Microcontroller	Frequency	Cost	Programming
Arduino Uno	16MHZ	Medium	Easy
PIC18f4550	48MHZ	Medium	Hard
My RIO	667MHZ	High	Easy

The Microcontroller choice is Arduino for many reasons, cost is medium, frequency is good and easy to programming the Microcontroller shown in figure (4.7) with specification in table (4.7).

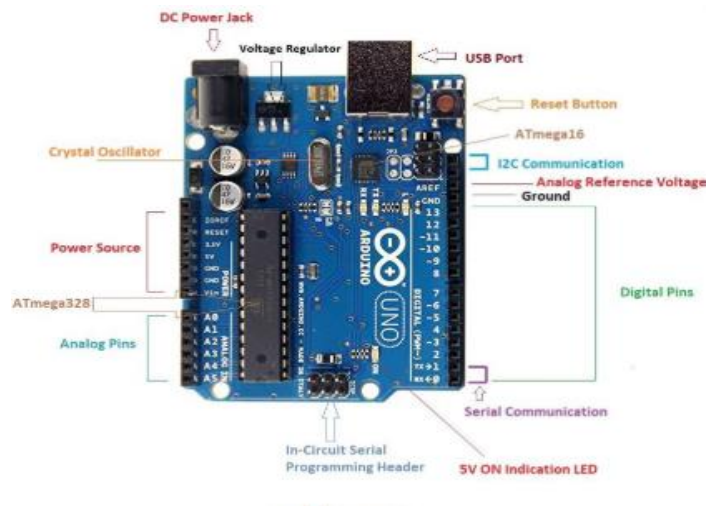


Figure (4.7): Arduino Uno ATmega 328

Table (4. 7): specification of microcontroller

Type	ATMega328
Frequency	16MHz
PWM pins	6 pins
Temperature Range	-40c to +85c
Operating Voltage	5-5.5V

4.5 Battery Selected

The battery selected is a dry used to supply the main energy consumption from robot system, we need 4 * 1.5 V batteries for the microcontroller, also we need 12 V for the

DC Motor, sensors, and Wi-Fi Module, table (4.8) discuss the robot components consumption.

Table (4. 8): Robot Components Consumption

Arduino Uno ATMega 328	50mA
Wi-Fi RN-171	180mA
Ultrasonic sensor	220mA for each motor
Driving motor	15mA
Other components such as(Microphone,...etc)	30mA

Chapter 5

Control System Design for the Robot Motion

5.1 Introduction

The robot is controlled by two processors, one is located within the robot's body and another worked as operation platform, the two are connected by WLAN. PC with sound card is used for speech recognition and signals transfer. The microcontroller is used to control the robot to take the corresponding action. Once connection is established, voice signals are processed and transferred between PC and the microcontroller. The hardware structure of our mobile robot control system is described in Chapter 1. Speech control robot system includes two parts: speech recognition module and control module. Speech recognition system is to provide an analysis of the human's voice in order to determine what action the robot needs to take to satisfy the operator's request.

5.2 BitVoicer Software

BitVoicer is a speech recognition application that enables simple devices, with low processing power, to become voice-operated. To do that, BitVoicer uses the PC processing power to analyze audio streams, identify the sentences present in these streams and send commands to a microcontroller connected to it. The BitVoicer software gives the ability to recognize the voice in multiple languages including English which is transmitted from the connected microphone on the microcontroller through the wireless module, BitVoicer interface is illustrated in Figure (5.1).

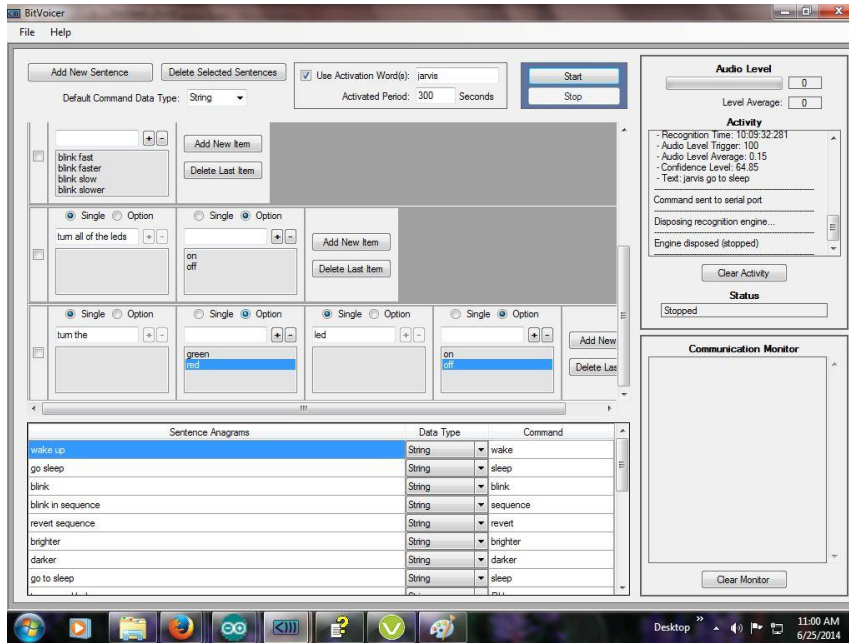


Figure (5.1): BitVoicer Interface.

5.3 Path planning.

The movement of WMR is based on two scenarios, first of them is the movement due to certain commands which is predefined such as Go Forward, Reverse, Left or Right with a desired value to move. The second is giving a command consists of the desired coordinates (x_f, y_f, θ_f) relative to its initial position (x_I, y_I, θ_I) .

5.3.1 Matlab / Simulink Software

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and proprietary programming language developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python. MATLAB software package has been assisting engineers to design and test system models for different fields of engineering. It provides a deep understanding of system parameters and how they affect its performance. Since it is a programming

package, it can interact with other programming languages to provide engineers who use those languages the flexibility to run their codes within a MATLAB environment. Engineering students can build their model and test it before implementing it in the real process. This paper presents and describes some dynamic models and control systems for Mechatronics engineering students using MATLAB and Simulink. Those models will be used as laboratory experiments to expose students to different topics and techniques in control engineering. This will enhance students' knowledge and make them familiar with the MATLAB environment.

5.3.2 Tracker Design

Tracking control of parallel wheeled differential drive mobile robot is considered. The robot has to reach the final goal by following a referenced trajectory. The initial position of the robot may be on or off the path.

In this work, the rotation of the robot wheels is controlled by a discrete-time servo tracking controller with discrete-time integrator in the forward path as shown in Figure (5.2), this control structure is well known for control engineer and is explained in most digital control textbooks. Here it is assumed that all states are measured and available for feedback using the incremental encoders on each of the driving wheels.

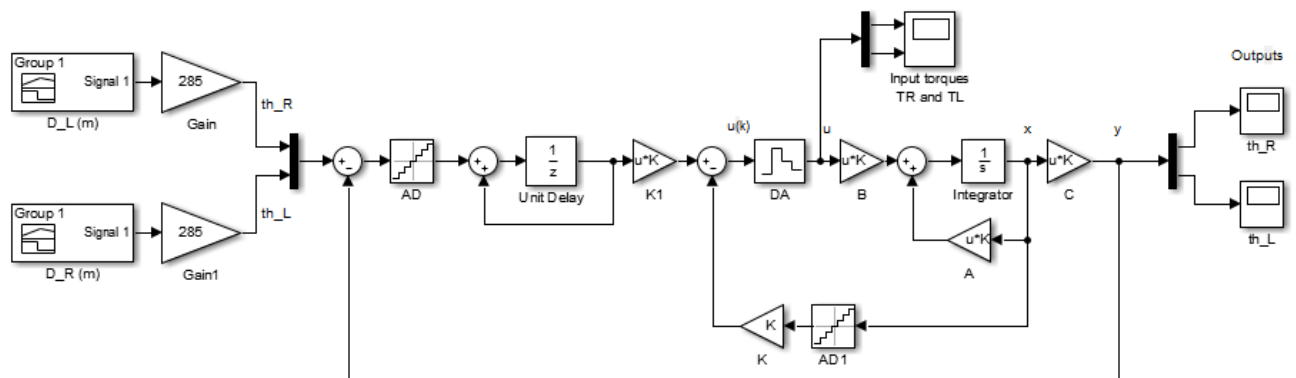


Figure (5.2): Block diagram of the wheels' motion control

Controller gain matrices $\mathbf{K} \in \mathbb{R}^{4 \times 4}$ and $\mathbf{K}_1 \in \mathbb{R}^{2 \times 2}$ are computed using pole placement method, the corresponding Matlab m-file is shown in the attached appendix.

Simulation results for the right and left wheels of the robot in translational motion with trapezoidal velocity profile similar to that shown in Fig (4.1) are shown in Fig

(5.2). Here Equation (3.7) is used to determine the desired rotation trajectory for the right and left wheels. Note that the controller managed to provide excellent tracking of the displacement profile with good tracking of the velocity profile. The corresponding input torques are shown in Figure (5.3), it can be recognized that the torque levels are within the range obtained using the dynamic equations of the robot ($\approx 1\text{Nm}$).

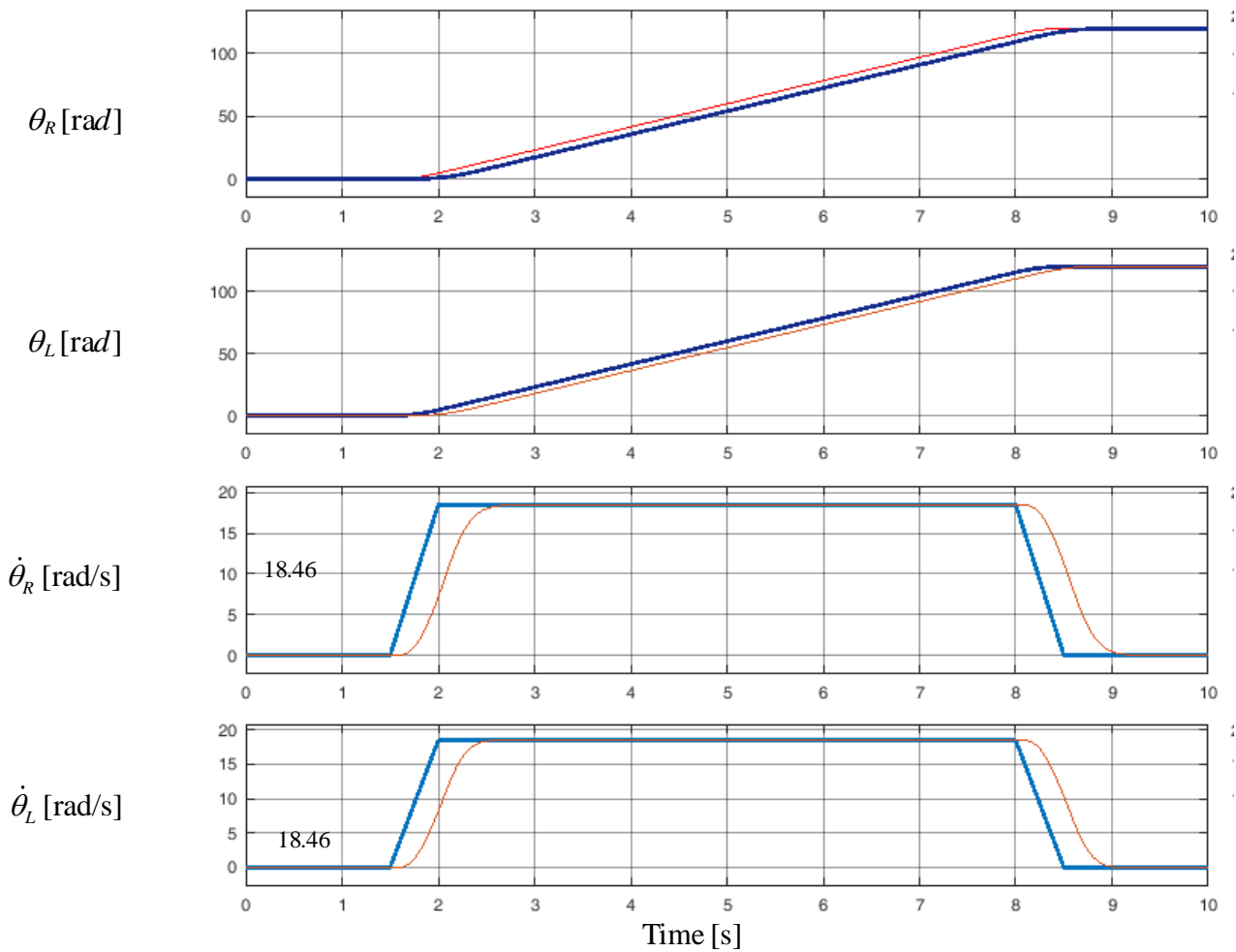


Figure (5.3): Trajectory tracking simulation results (— input, — output)

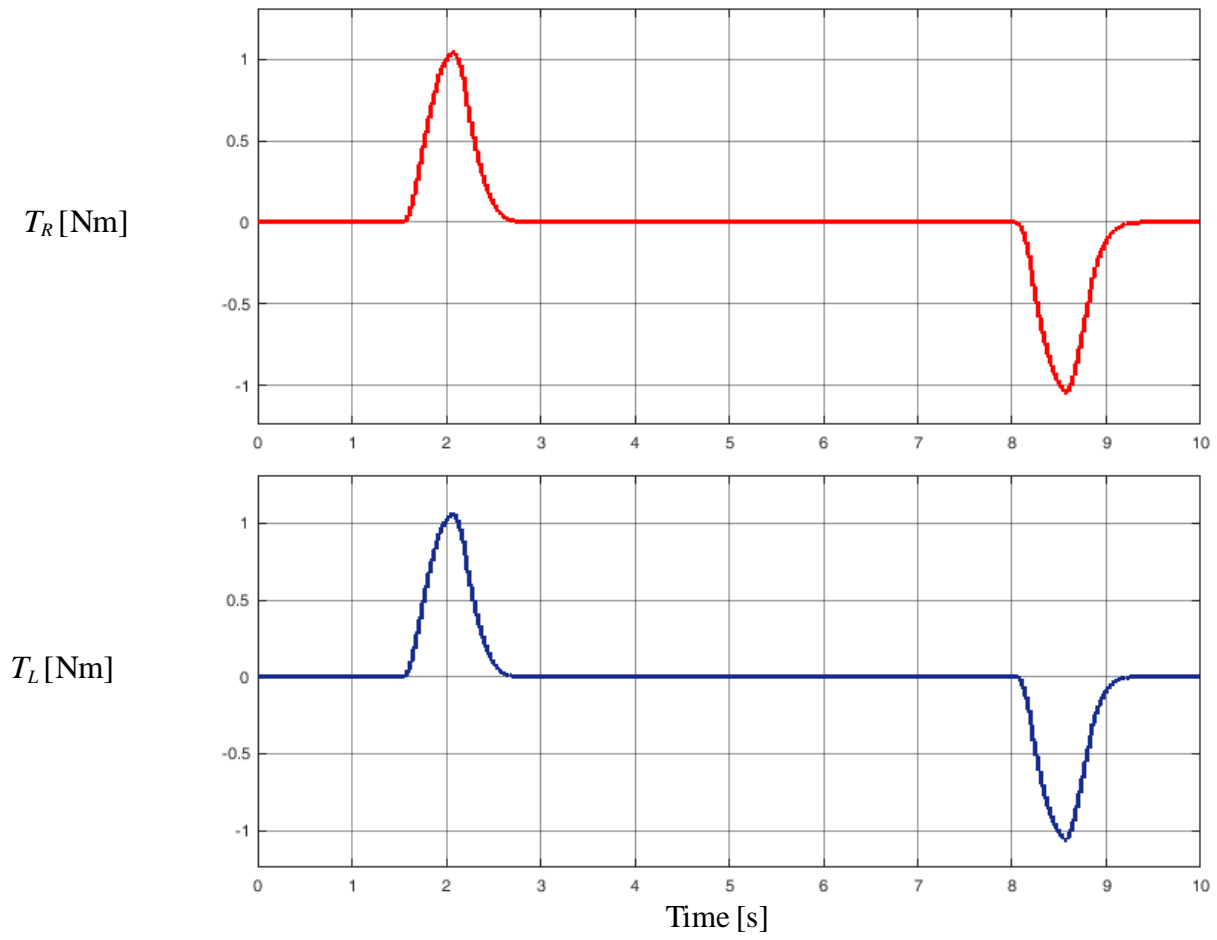


Figure (5.4): Input torque levels applied to right and left wheels

Chapter 6

Planning

6.1 Planning problem

For the execution of a specific robot task, mobile robot needs to translate from initial pos(x_I, y_I, θ_I) to final pos (x_f, y_f, θ_f) through specific time (ΔT), given two motor torques and non-holonomic constraints mentioned in Section (3.1). The adopted solution depends on the analysis and method presented in [16].

6.2 Path and timing law

Assume that the robot plans a trajectory $\mathbf{q}(t)$, for $t \in [t_I, t_f]$, that leads a mobile robot from an initial configuration $q(t_I) = q_I$ to a final configuration $q(t_f) = q_f$ in time $\Delta T = t_f - t_I$.

The trajectory $\mathbf{q}(t)$ can be broken down into a geometric path $\mathbf{q}(s)$, where $\frac{dq(s)}{ds} \neq 0$ for any value of \mathbf{S} , and a timing law $s = s(t)$ where $s(t)$ is monotonically increasing function of time on $t \in [t_I, t_f]$, i.e. $\dot{s}(t) \geq 0$, where the value of s is the arc length along the path. Generalized velocity vector can then be obtained as

$$\dot{q} = \frac{dq}{dt} = \frac{dq}{ds} \dot{s} = \dot{q} \dot{s} \quad (6.1)$$

where the prime symbol denotes differentiation with respect to \mathbf{s} . The generalized velocity vector is then obtained as the product of the vector \dot{q} by the scalar \dot{s} . The Nonholonomic constraints in equation (4.2) can be re expressed as

$$A(q)\dot{q} = A(q)\dot{q}\dot{s} = 0 \quad (6.2)$$

The choice of a timing law $s=s(t)$, for $t \in [t_I, t_f]$ will identify a particular trajectory along the path. Due to the pure rolling constraints in equations (3.2, 3.3) the following condition for geometric admissibility of the path is considered

$$[\sin \theta - \cos \theta - 0]\dot{q} = \dot{x} \sin \theta - \dot{y} \cos \theta = 0 \quad (6.3)$$

Therefore, all the admissible paths for the unicycle can be formulated as

$$\dot{x} = \tilde{v} \cos \theta \quad (6.4)$$

$$\dot{y} = \tilde{v} \sin \theta \quad (6.5)$$

$$\dot{\theta} = \tilde{\omega} \quad (6.6)$$

Where \tilde{v} , $\tilde{\omega}$ are related to v,w by

$$v(t) = v(s)\dot{s}(t) \quad (6.7)$$

$$\omega(t) = \omega(s)\dot{s}(t) \quad (6.8)$$

Equation (6.3) expresses the fact that the tangent to the cartesian path must be aligned with the robot sagittal axis. In another words, no edges or sharp points are allowed on the path.

6.3 Differential flatness

kinematic model of unicycle mobile robot exhibits a property known as differential flatness, that is particularly relevant in planning problems. Consider the following nonlinear system

$$\dot{x} = f(x) + G(x)u \quad (6.9)$$

$$y = h(x) + d(x)u \quad (6.10)$$

Such system is differentially flat if there exists a set of outputs y, where states x and control inputs u can be expressed as unique functions of y and its derivatives.

$$x = x(y, \dot{y}, \ddot{y}, \dots, y^{(r)}) \quad (6.11)$$

$$u = u(y, \dot{y}, \ddot{y}, \dots, y^{(r)}) \quad (6.12)$$

Outputs y are called flat outputs. Cartesian coordinates [x ,y] of unicycle mobile robot are considered flat outputs, consider geometric model in Equation (4.8), by defining an output Cartesian path [x(s),y(s)] one can calculate the orientation from

$$\theta(s) = a \tan^2(\dot{y}(s), \dot{x}(s)) + K \pi \quad ; \quad k=0.1 \quad (6.13)$$

Where , k defines if the robot is moving forward (k = 0) or backward (k = 1) and ($a \tan^2 \theta$) is a variation of arctangent that calculates the angle between the x axis and the line passing through point (x ,y) from origin. The geometric inputs that drive the robot along the Cartesian path are obtained from (6.4), (6.13) as:

$$v(s) = \pm \sqrt{(\dot{x}(s))^2 + (\dot{y}(s))^2} \quad (6.14)$$

$$\omega(s) = \frac{\ddot{y}(s)\dot{x}(s) - \ddot{x}(s)\dot{y}(s)}{(\dot{x}(s))^2 + (\dot{y}(s))^2} \quad (6.15)$$

6.4 Path planning

Our robot admits a set of flat outputs y that exploited to solve planning problems efficiently. In particular, the problem of planning a path that leads the robot from an initial configuration $q(s_I) = q_I = [x_I, y_I, \theta_I]^T$ to a final configuration $q(s_f) = q_f = [x_f, y_f, \theta_f]^T$

6.5 Planning via Cartesian polynomials

The planning problem can be solved by interpolating the initial values (x_I, y_I) and the final values (x_f, y_f) of the flat outputs x, y . By letting $(s_I = 0)$ and $(s_f = 1)$ and using the following cubic polynomials

$$x(s) = s^3 x_f - (s-1)^3 x_I + \alpha_x s^2 (s-1) + \beta_x s (s-1)^2 \quad (6.16)$$

$$y(s) = s^3 y_f - (s-1)^3 y_I + \alpha_y s^2 (s-1) + \beta_y s (s-1)^2 \quad (6.17)$$

That automatically satisfy the boundary conditions on x, y . The orientation at each point being related to \dot{x}, \dot{y} by (6.10).

The values of $\alpha_x, \alpha_y, \beta_x, \beta_y$ in equation (6.16) and (6.17) are computed as shown in equations (6.18) and (6.19).

$$\begin{bmatrix} \alpha_x \\ \alpha_y \end{bmatrix} = \begin{bmatrix} K \cos \theta_f - 3x_f \\ K \sin \theta_f - 3y_f \end{bmatrix} \quad (6.18)$$

$$\begin{bmatrix} \beta_x \\ \beta_y \end{bmatrix} = \begin{bmatrix} K \cos \theta_I + 3x_I \\ K \sin \theta_I + 3y_I \end{bmatrix} \quad (6.19)$$

The choice of k has a precise influence on the obtained path. The evolution of the robot orientation along the path and the associated geometric inputs can then be computed by using Equations (6.13), (6.14) and (6.15) respectively.

Chapter 7

Practical work and experiment

7.1 Introduction

This chapter includes the experimental and practical work of wheel mobile robot (WMR), we know that the user speaks a word to the microphone and analyses it by BitVoicer software then send data to the microcontroller and turns it into action.

The predefined commands of the WMR is robot go forward, robot go reverse, robot go right, robot go left with specific distance also command as do angle 45 degree , and the robot executes the corresponding action .

During the tests, we have experimented the robot in two environments the first is the normal environment which gives a perfect results based on the speech recognition in BitVoicer with high confidence, on the other hand the second environment was noisy and we have noticed that the confidence of the speech recognition was less than the normal environment due to the noise while giving commands, this would make a need to repeat the given command again to obtain more confidence in the speech recognition.

7.2 Normal environments and Noisy environment

The wheel mobile robot has perfect performance and robust in the normal environment and the recognition accuracy reached 100%, the results are shown in table (7.1).

Table (7.1): Experiment result in normal environment

No.	Command	Test times	Correct times
1	Robot go forward 1m	3	3
2	Robot go reverse 1m	3	3
3	Robot go right	3	3
4	Robot go left	3	3
5	Robot do angle 45 degree	3	3

In the noisy environment the recognition accuracy has decreased distinctly for many reasons, Especially the sounds that confuses the speech recognition and affects the speech confidence, the experiments result are shown in table (7.2).

Table (7.2): Experiment result in noise environment

No.	Command	Test times	Correct times
1	Robot go forward 1m	3	2
2	Robot go reverse 1m	3	1
3	Robot go right	3	2
4	Robot go left	3	1
5	Robot do angle 45 degree	3	2

Chapter 8

Conclusion and future work

8.1 Conclusion

In Our project, we have designed and build a voice controlled robot that analyzes the voice using software called BitVoicer which identifies the command and turns it into a form that the microcontroller can understand to perform the task as it's given from the user.

The robot moves through two independently actuated wheels and stabilized by a castor wheel, the robot is able to perform a predefined movement commands with certain distances as the user selects.

8.2 Future work

Some of the improving suggestions we would like to represent is adding a kalman filter to reduce the noise, also adding a Global positioning system (GPS) to navigate the system to be able to perform out door, build an optimal path planning and mobile robot navigation to obtain a better obstacle avoidance .

References and resources :

[1] *Kinematic Model of Wheeled Mobile Robots -Department Of Mechanical Engineering, National Institute of Technology- Rourkela, Rourkela, India – published in Int. J. on Recent Trends in Engineering & Technology, Vol. 05, No. 04, Mar 2011.*

[2] *Robot Control Based on Voice Command ,Proceedings of the IEEE International Conference on Automation and Logistics Qingdao, China September 2008.*

[3] Ahmad Abu Hatab, R. D. (2013). *Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework. Advances in Robotics & Automation, 02(02).*
<https://doi.org/10.4172/2168-9695.1000107>.

[4] Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2010). *Robotics: Modelling, Planning and Control. Springer London. Retrieved from <https://books.google.ps/books?id=jPCAFmE-logC>*

[5] Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). *Introduction to Autonomous Mobile Robots. MIT Press. Retrieved from <https://books.google.ps/books?id=4of6AQAAQBAJ>*

[6] *Linear Systems Control Deterministic And Stochastic Methods – Elbert Hendricks.*

[7] *Review of Voice Control Robot Applications International Journal of Advances in Engineering, 2015, 1(9), 671 - 674*
ISSN: 2394-9260 (printed version); ISSN: 2394-9279 (online version);
[url:http://www.ijae.in](http://www.ijae.in)

[8] Tzafestas, S. G. (2014). *Introduction to Mobile Robot Control. Introduction to Mobile Robot Control. <https://doi.org/10.1016/B978-0-12-417049-0.00001-8>*

[9] *Tracking Controller of Mobile Robot* DivyaAneesh, Assistant Professor, Biomedical Engineering Department, Sahrdaya College of Engineering & Technology, Thrissur Kerala 2012 International Conference on Computing, Electronics and Electrical Technologies [ICCEET]

[10] *Path Planning Of Mobile Robot* 2012 9th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI) Daejeon, Korea / November 26-29, 2012.

[11] *Adaptive Navigation of Mobile Robots with Obstacle Avoidance* IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 13, NO. 4, AUGUST 1997

[12] *Digital Control System Analysis and Design* 3rd ED , Charles Philips Auburn University.

[13] *Linear Systems Control, Deterministic and Stochastic Methods*, Elbert Hendricks.

[14] *MODELLING OF MOBILE ROBOT DYNAMICS* Edouard Ivanjko1 , Toni Petrinic´ 2 , Ivan Petrovic´ 1 1University of Zagreb, Faculty of Electrical Engineering and Computing 10000 Zagreb, Unska 3, Croatia 2HEP d.d., 10000 Zagreb, Ulica grada Vukovara 37, Croatia

[15] *Tennis-Ball Collecting Robot* , Majdi Sultan, Sohaib Abu Omar, Mahmoud Sharabati , 2018

[16] Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2010). *Robotics: Modelling, Planning and Control*. Springer London. Retrieved from <https://books.google.ps/books?id=jPCAFmE-logC>

Appendix A
MATLAB Codes

```

clear all
clc
R=0.0325; L=0.15;
a=0.02; b=0.01;f=0;
M=[a b;b a];
N=[f 0;0 f];
A=[zeros(2,2) eye(2);zeros(2,2) -inv(M)*N];
B=[zeros(2,2);inv(M)];
C=[eye(2) zeros(2,2)];
n=4; m=2;
% from desired wn, we get T
T=.03;
% discretization of the plant
[F,G]=c2d(A,B,T);
% Extended model
Fe=[F zeros(n,m);-C eye(m)];
Ge=[G;zeros(m,m)];
Mc=ctrb(Fe,Ge);
Rc=rank(Mc);
%desired closed loop poles
z1=0.8; z2=0.9; z3=1; wn=10;
p1=-z1*wn+j*wn*sqrt(1-z1^2);
p2=-z2*wn+j*wn*sqrt(1-z2^2);
p3=-z3*wn+j*wn*sqrt(1-z3^2);
Pc=[p1 conj(p1) p2 conj(p2) p3 conj(p3)];
Pcz=exp(Pc*T);
Ke=place(Fe,Ge,Pcz);
K=Ke(:,1:n);
K1=-Ke(:,n+1:n+m);

```

Function

```
[Xdot, Ydot, Thdot, V, W]=fcn(ts, xi, yi, xf, yf, th
etai, thetaf, ki, kf, D
irection)
%Define S(t) coefficients with Tf-Ti=3 sec
p1 =0.024691358024691; p2 =-
0.185185185185185;
p3=0.370370370370370;
p4 = 0; p5 = 0; p6 = 0;
%The above constants obtained by solve for
Sf=1
S= p1*ts.^5 + p2*ts.^4 +p3*ts.^3 + p4*ts.^2
+p5*ts +
p6;%timing law S = s(t)
Sd=5*p1*ts.^4 + 4*p2*ts.^3 +3*p3*ts.^2 +
2*p4*ts +p5;
%Define direction of robot velocity
"1:Forward -1:Backward"
Dir=Direction;
c
%Find the X(s) & Y(s) coefecients as
mentioned in Robotics
Modelling
%Planning and Control by Siciliano, Page
493
alphax=kf*cosd(thetaf)-3*xf;
alphay=kf*sind(thetaf)-3*yf;
betax=ki*cosd(thetai)+3*xi;
betay=ki*sind(thetai)+3*yi;
%Compute the cubic polynomials "X(s) &
Y(s)" as mentioned in
Robotics
%Modelling Planning and Control by
Siciliano, Page 49
%Define the polynomial constants "For X(s)"
Ax=(xf-xi+alphax+betax); Bx=(3*xi-alphax-
2*betax); Cx=(betax-
```



```

3*xi); Dx=xi;
Xs=(Ax*S.^3 + Bx*S.^2 + Cx*S+ Dx); %X(S)
Xsd=(3*Ax*S.^2 + 2*Bx*S + Cx); %X'(S)
Xsdd=(6*Ax*S + 2*Bx); %X''(S)
%Define the polynomials constants "For
Y(s)"
Ay=(yf-yi+alphay+betay); By=(3*yi-alphay-
2*betay); Cy=(betay-
3*yi); Dy=yi;
Ys=(Ay*S.^3 + By*S.^2 + Cy*S + Dy); %Y(S)
Ysd=(3*Ay*S.^2 + 2*By*S + Cy); %Y'(S)
Ysdd=(6*Ay*S + 2*By); %Y''(S)
Vs= sqrt((Xsd).^2 + (Ysd).^2) *Dir;
V=Vs.*Sd;
%For W~(s)
%Define W~(s) coefficients as Handwritten
calculations
M=[ (12*Ay*Bx + 6*Ax*By)*S.^2 + (6*Ay*Cx +
4*By*Bx)*S +
(2*By*Cx) ]; %y''(s)x'(s)
N=[ (12*By*Ax + 6*Ay*Bx)*S.^2 + (6*Ax*Cy +
4*By*Bx)*S +
(2*Bx*Cy) ]; %x''(s)y'(s)
num=M-N; %y''(s)x'(s)-x''(s)y'(s)
den=Vs^2;
Ws=num./den;
W=Ws.*Sd;
thetas=atan2(Ysd,Xsd);
%Estimate motors velocities
Vr=(2*V+L*W)/2; Vl=(2*V-L*W)/2;
Wr=Vr/R; Wl=Vl/R;
%DDMR kinematics
Xdot = ((R/2) * (Wr+Wl) * (cos(thetas)));
Ydot = ((R/2) * (Wr+Wl) * (sin(thetas)));
Thdot=((R/L) * (Wr-Wl));

```