Tennis-Ball Collecting Robot

By

Majdi Sultan

Sohaib Abu Omar

Mahmoud Sharabati

Supervisor

Prof. Dr. Karim Tahboub

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

Palestine Polytechnic University

May 2018

Palestine Polytechnic University
Collage of Engineering and Technology
Mechanical Engineering Department
Hebron - Palestine

Tennis-Ball Collecting Robot

By

Majdi Sultan

Sohaib Abu Omar

Mahmoud Sharabati

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

Supervisor signature

.....

Prof. Dr. Karim A Tahboub

Testing Committee signature

....

Dr. Yousef Sweiti *Dr. Jasem Tamimi*

Department Head signature

.....

Dr. Iyad Hashlamon

Abstract

Tennis is a racket sport played on a rectangular and flat surface court. It can be played individually against a single opponent (singles) or between two teams of two players each (doubles). Modern tennis balls are made of hollow vulcanized rubber with a felt coating with a diameter of 65.41 to 68.58 mm and a weight of 56 to 59.4 gram.

Tennis players need to collect the played balls from different locations in the court. In training sessions many balls are used and played. This makes the collection a tedious and tiring process. Some tools and mechanisms are used to facilitate the collection process such as pick up hoppers and tubes.

In this graduation project, we designed and built a nouvelle autonomous / semi-autonomous mobile robot to specifically collect up to fifty tennis balls. The robot navigates through two independently-actuated wheels and is stabilized by a castor. The nouvelle collection mechanism is based on a spinning brush that pushes tennis balls into a lightweight basket on top of the robot. A motor makes the drum spins at a desired speed that is experimentally verified. Three successive working prototypes have been designed, built, and tested to make sure the adopted ideas and conceptual designs are applicable and robust.

The mobile robot has its own on-board intelligence where it is equipped with a camera, sonars, and a myRIO embedded device. The robot can identify, localize, and collect tennis balls autonomously or, alternatively, a tennis player can command the robot to move it and collect the balls through a mobile-phone application. [In the future, global cameras and a central image-processing system will be used for identifying and localizing all tennis balls on the court. This system is thought as well to take care of optimal path planning, collision avoidance, and mobile robot navigation].

ملخص

تعتبر رياضة التنس واحدة من اشهر الالعاب في العالم، يتابز فيها لاعبين على ملعب ذو سطح مستو، بحيث يحمل كل لاعب مضرباً ليستخدمه في ضرب الكرة، يتبارى فيها لاعبان في مباريات الفردي، أو فريقان مكونان من لاعبين في مباريات الزوجي. تصنع كرات التنس الحديثة من المطاط و تكون مكسوة باللباد. تكون الكرة ذات شكل خارجي منتظم و تكون أقسامها متصلة بدون خيوط، و يتراوح قطرها بين 63.5 ملم و 66.7 ملم، و وزنها يتراوح بين 56.7 غرام و 58.5 غرام.

يتعين على لاعبي التنس جمع الكرات التي يتم لعبها من مواقع مختلفة في الملعب حيث ان العديد من الكرات يتم استخدامها في مرحلة التدريب. مما يجعل هذه المهمة شاقة ومتعبة. يتم استخدام بعض الأدوات والآليات لتسهيل عملية الجمع مثل الانابيب المفرغة.

تم في هذا المشروع تصميم و بناء روبوت متنقل يقوم بالتقاط و جمع كرات التنس بمساحة تخزينية تصل الى 50 كرة، يتنقل الروبوت بواسطة عجلتان مستقتان بالاضافة الى عجلات مساعدة. و تم تصميم نظام جمع الكرات و ربطها بمحرك كهربائي لتدويرها بالسرعة المطلوبة. و لتحقيق اهداف المشروع و الوصول الى التصميم النهائي تم بناء ثلاث نماذج تحاكي التصميم النهائي لمعالجة المشاكل المتعلقة بالتصميم و الوصول الى الحل الامثل.

يحتوي الروبوت على انظمة فرعية لجعل الروبوت ذكي الحركة، وتتمثل هذه الانظمة في المتحكم الذكي، اضافة الى نظام معالجة الصور الذكي، و نظام الاستشعار المكون من مستشعرات الحركة الذكية. يقوم الروبوت بالتعرف على الكرات و تمييزها وجمعها بطريقة الية او بدلا من ذلك يمكن التحكم في كامل النظام عن طريق تطبيق ذكي يثبت على الاجهزة اللوحية الذكية. [مستقبلاً سيتم اضافة نظام رؤية شامل لتغطية كامل الملعب و التعرف على جميع الكرات في الملعب، وسيتم بناء نظام لاختيار المسار الامثل لحركة الروبوت لجعله اكثر ذكاء] .

Dedication

We dedicate our research project to our beloved parents for their continuous invaluable support and encouragement all through the years and to our dear siblings for providing us with a comfortable environment for study and research.

Acknowledgement

We would like to express our gratitude to our supervisor, Prof. Karim Tahboub, for his full support and guidance and remarkable suggestions. We would also like to thank our teachers for all the efforts they have exerted to make us qualified engineers who can assume-with confidence-our role in building our community. Thanks are also due to our classmates and friends for their cooperation and encouragement.

We would also like to thank the Deanship of Graduate Studies and Scientific Research at Palestine Polytechnic University for their financial support.

Table of Contents

List of Figures

List of Tables

Chapter One: Introduction

1.1 Introduction

1.2 Recognition of the need

1.3 Literature review and existing solutions

1.4 Approach to find an alternative design

1.5 Alternative design

1.6 Candidate design

1.1 Introduction

Tennis is a racket sport played on a rectangular and flat surface court. It can be played individually against a single opponent (singles) or between two teams of two players each (doubles). Modern tennis balls are made of hollow vulcanized rubber with a felt coating with a diameter of 65.41 to 68.58 mm and a weight of 56 to 59.4 gram.

Tennis players need to collect the played balls from different locations in the court. In training sessions many balls are used and played. This makes the collection a tedious and tiring process.

Mobile robots are widely used in field of applications to reduce human efforts and save time. Mobile robots can navigate an uncontrolled environment autonomously without the need for physical or electro-mechanical guidance devices [15].

In this project we propose a solution to solve tennis ball collecting problem by designing and building nouvelle autonomous / semi-autonomous mobile robot to specifically collect up to fifty tennis balls, faster than human.

1.2 Recognition of the need

In this section, we define the need of the tennis ball collecting problem and the requirements to be met by the designer. The main task is collecting tennis balls from ground upon the players request, so the following requirements are listed.

- The type of play field is rough-ground.
- The dimension of the play field is 10m wide by 24.6m long [25].
- The balls are randomly distributed in the field.
- The collection speed is suggested to be thirty balls in five min.
- The collecting process must collect not less 80% of the total balls.
- The storing area must be detachable and allows for emptying process.
- The on board electrical power source must be a rechargeable battery.
- The operating time of batteries should not be less than one hour without recharge and the charger power source is AC 220V.
- The robot must be safe for end users.
- The collection process should not destroy the ball.
- The robot must be user friendly with convenient interface.
- The total cost must not exceed $1000.

1.3 Literature review and existing solutions

In this section, we present the most important papers published in IEEE/ASME Transactions on Mechatronics, about this subject. And we will present them in terms of designing, applying, function specification and equipment used in each in it.

1.3.1 Development of an autonomous tennis ball retriever robot as an educational tool

I. Elamvazuthi et.al. present a tennis ball retriever robot prototype, the robot sweeps the court as shown in Figure (1.1) and the robot navigates a left side navigation Figure (1.2). The performance of the robot was quite (60% of total balls were collected (12/20 balls)) [1].

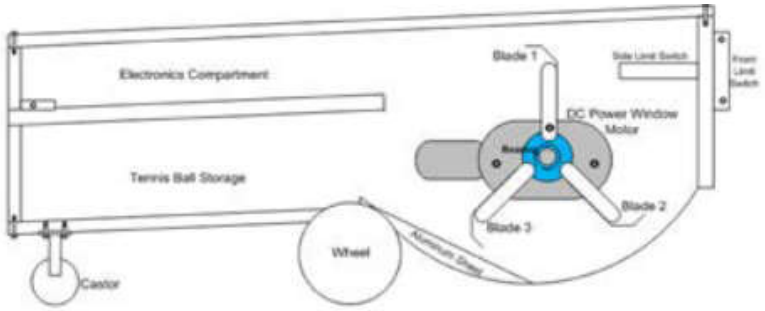


Figure 1.1: Tennis ball retriever robot by sweeping the court

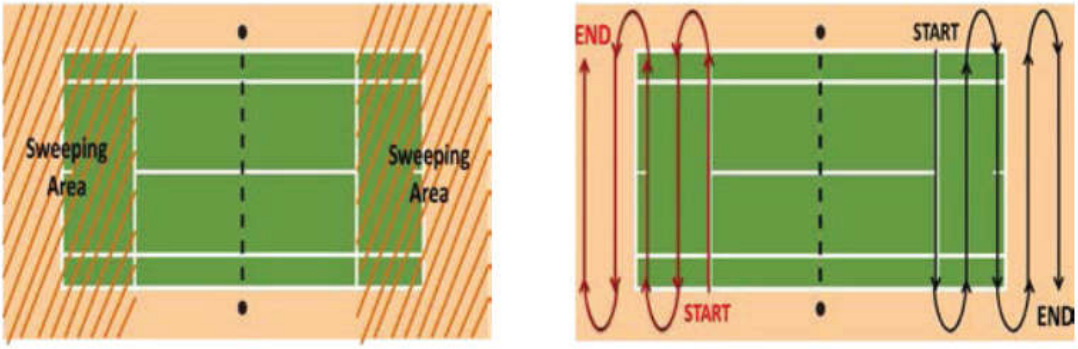


Figure 1.2: Method of navigation

1.3.2 An intelligent tennis ball collecting vehicle using smart phone touch-based interface

Chen presented an intelligent tennis ball collecting vehicle which is an autonomous vehicle equipped with a novel tennis ball collector. The robot has a novel with multiple channel tennis ball collector comprises a pair of parallel acrylic discs joined by four springs, each was drilled a hole at their center from which they can be attached to an axle. To accommodate the tennis balls in between, the two discs are made of resilient material and four springs are attached in Figure (1.3). As the discs rolling through the tennis balls, the ball is then squeezed in between the two discs as shown in Figure 1.3 [2].

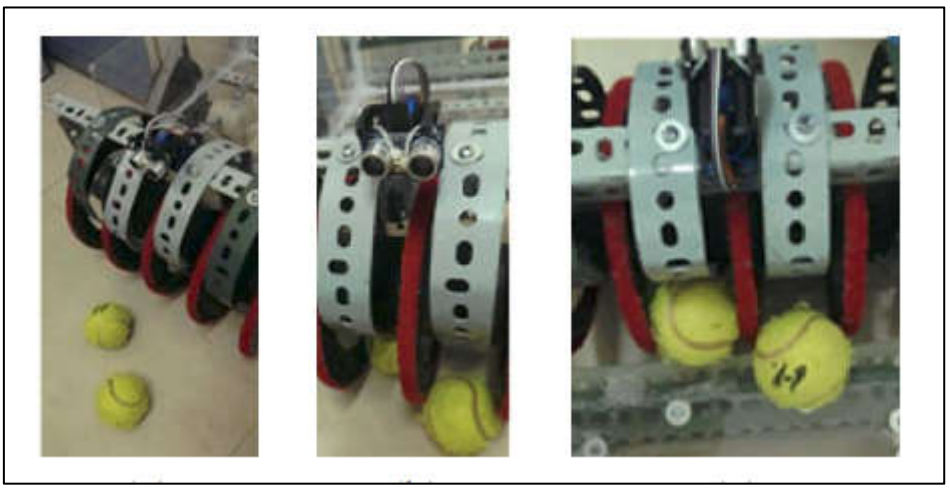


Figure 1.3: Robot a novel with multiple channel tennis ball

1.3.3 Robotic tennis ball collector “Tennibot”

H. Eletrabi presented an intelligent tennis ball collecting robot has a multiple sensors and local camera. When the robot detects a ball the robot moves towards it and picks it up then the robot moves on to the next ball until it collects all balls. Tennibot solution is shown in Figure (1.4) [3].

Figure 1.4: Tennibot robot

1.3.4 Inwatec tennis ball collector

Inwatec group used a Bosch AHM38G lawn mower, they unmount the cylinder and cut all the blades replace the metal blades (that cuts the grass) with two blades of some soft material (1-2 mm rubber) and mount the new blades with some bolts/nuts as shown in Figure (1.5) [4].

Figure 1.5: Inwatec tennis ball collector

1.4 Approach to find an alternative design

1.4.1 Goal of the project

The project goal is to solve the tennis ball collecting problem by designing and implementing an autonomous mobile robot that meets the user requirements mentioned in section 1.2.

1.4.2 Achieving system requirements

Once we have a firm grasp of the system requirements we consider exactly how we go to meet them. At this stage in the process, we made a brain stormy to generate copious numbers of ideas to solve the problem and select viable approaches out of a great many candidate solutions.

Number of solutions conceived that look likely to meet project requirements. To choose which candidate designs to move forward we construct prototype to serve to validate our initial ideas and demonstrate the potential of a given approach, serve to inform our estimates of how difficult the project is likely to be.

1.5 Alternative design

From literature review and existing solutions [3,4,5,6], the existing method of collecting the ball do not match our needs. Our needs to design an autonomous / semi-autonomous mobile robot, the robot alternatively collect up to fifty tennis balls, faster than human. So we have to create new solution to meet our needs.

In this section we listed the alternative ideas which suggested to help solving the collecting problem. The ideas are a result of open-ended questions, brain stormy and YouTubes. We follow approach to get the candidate design which studying the existing systems and solutions which are like balls collecting in its principle, so we study the trash collecting method (garbage), crops collecting method, grass cutting method (lawn mower) and existing methods of collecting the balls. So, the following ideas are listed:

1.5.1 Collecting using vacuum and suction

The concept of this idea is implemented in Figure (1.6) [5], the collecting mechanism uses an air to pick up the balls from ground like sweeper in work principal. This idea is type of trash collecting method. This method will collect all existing balls on ground. But this method requires very accurate positioning to work well, high power source and in addition the ground may contain a rubbish and objects other than balls. All these disadvantages do not match our needs.

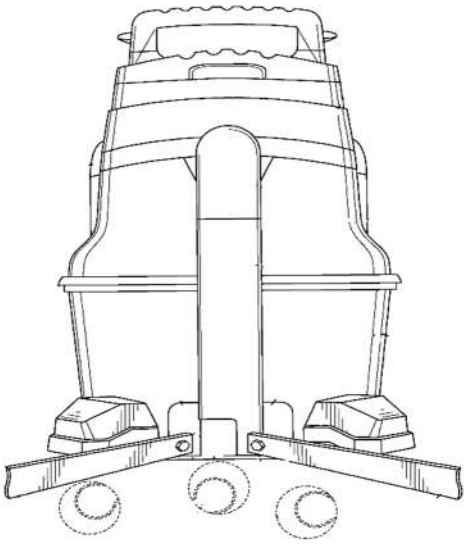


Figure 1.6: Vacuum collecting robot

1.5.2 Collecting using two cylinders with vertical actuators

The concept of this idea as shown in Figure (1.7) [6], its principal of work it uses two rotating cylinders attached with two vertical motors to collect the balls to the storage area like garbage. This method collects with high speed, can be used with scattered balls, but it needs a high accurate positioning and has a low collecting speed which does not matches our needs.

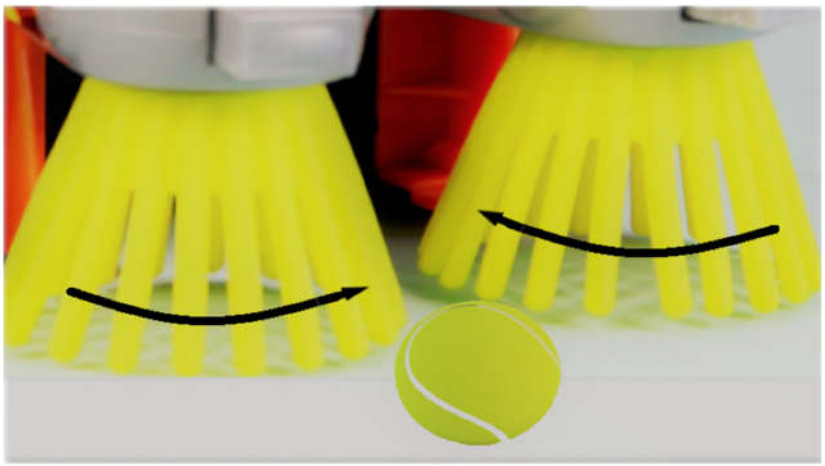


Figure 1.7: Collecting using two cylinders with vertical actuators

1.5.3 Throwing instead of collecting

The principal of this idea is to have a mechanism to throw the balls to specific area using air pump stead of collect it from ground. This method does not match our needs because it collects the balls from ground taking long time.

1.5.4 Collecting using gripper arm tool

This idea shown in Figure (1.8) [7] uses a gripper arm to pick up the balls by catch each ball, then through it to storage area. This method is effective for small amount of balls because it needs a high accuracy of positioning to pick up the ball that will take too much time to make the collection. This alternative does not meet our need. [7]

Figure 1.8: Gripper-tool collecting arm

1.5.5 Collecting using multiple channel tennis ball collector

This idea has a novel with multiple channel tennis ball collector shown in Figure (1.9) [8], comprises a pair of parallel acrylic discs joined by four springs, each was drilled a hole at their center from which they can be attached to an axle. To accommodate the tennis balls in between, the two discs are made of resilient material and four springs are attached. As the discs rolling through the tennis balls, the ball is then squeezed in between the discs, it is similar to crops collecting method. This method high accuracy of positioning to pick up the ball, so it does not meet our needs.

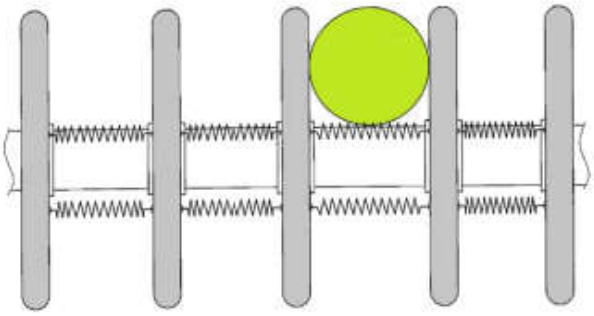


Figure 1.9: Robot a novel with multiple channel tennis ball

1.5.6 Collecting using lawn mower mechanism

This mechanism collects the balls like grass cutting method (lawn mower) Figure (1.10) [9], this method collects the balls with high efficiency and in short time that meet our needs, but the problem of this solution it is not automatic, it requires a human to push it toward balls.

Figure 1.10: Lawn mower mechanism

1.5.7 Collecting using a rough conveyor to pick up the balls

The concept of this idea is to use an oblique conveyor attached to robot, the balls is picked up to storage area by moving a rough conveyor as shown in Figure (1.11) [10]. [It looks like crops collecting method. this method will collect the balls with high efficiency, but the collecting speed is low which do not meet our need.

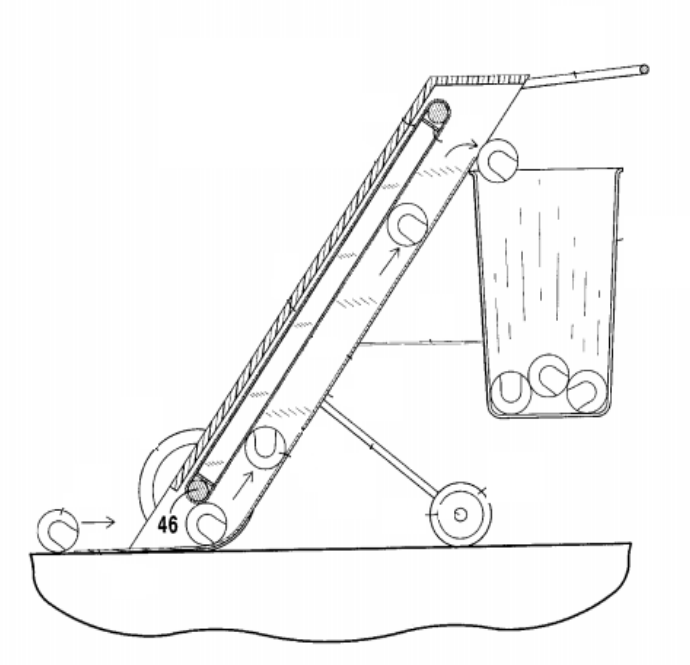


Figure 1.11: Rough conveyor collector

1.5.8 Collecting using Velcro (scotch)

This alternative idea uses a scotch to pick up the balls from ground, because the balls are made from cloth strips stitched together with thread and stuffed with feathers like fibers, it has an adhesion property with scotch as shown in Figure (1.12) [11]. This method will collect the balls with high speed. But in the long term this method does not effective, because it destroys the outside shape of balls, and it is required to change the scotch of mechanism from time to time.

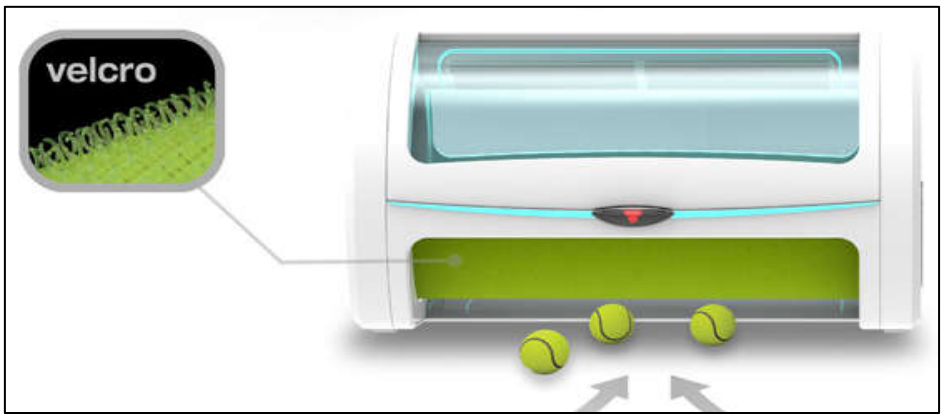


Figure1.12: Rough conveyor collector

1.5.9 Collecting by jump over a ball

The concept of this idea is to use a mechanism with tube shape to pick up the balls see Figure (1.13) [12] this method will pick up all balls from ground and store them inside the shape itself. This design needs a high accurate positioning and has a low collecting speed which is not meet our needs.

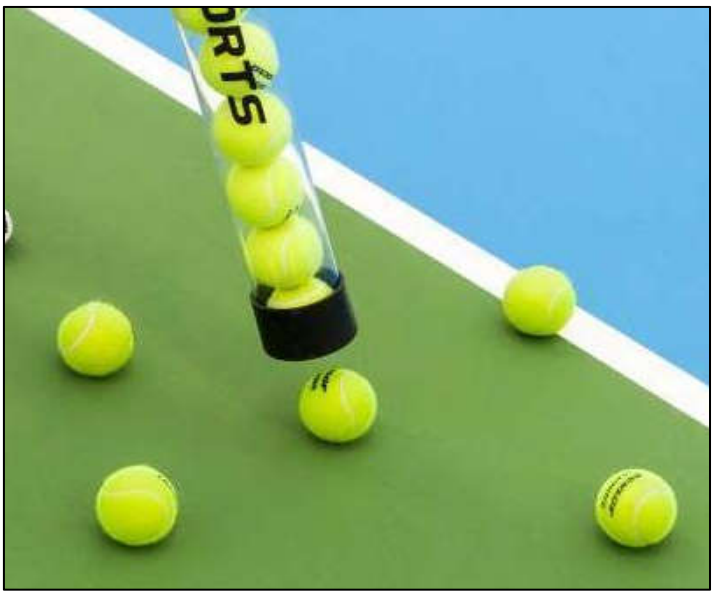


Figure 1.13: Collecting by jump over a ball

1.6 Candidate design

Based on brain storming and alternative designs, we proposed that the existing solutions do not match our needs so we have to create a new design. We take the design idea from two applications, the first one called "tennibot" Figure (1.14), we take the frame design from it, the movement of robot and the wheels design as shown in Figure (1.15), other wheels are casters used for robot stability.

Figure 1.14: Tennibot ball collector robot

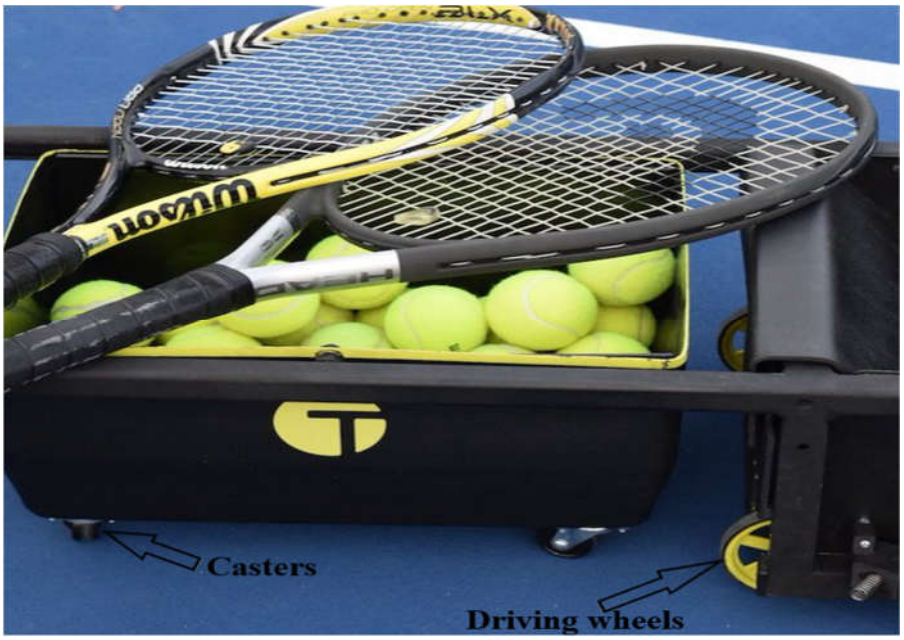


Figure 1.15: Tennibot driving wheels

The Tennibot solution [3] requires a high positioning to operate accurately.

The second solution takes from hand pushed lawnmower Figure (1.16), for that idea we will replace the metal blades (that cuts the grass) with two blades of some soft material (1-2 mm plastic or rubber) to collect the balls and push them to storing area as shown in Figure (1.17) [4]. We will take the collecting mechanism of the second solution and automate it to make it spins by a motor.

Figure 1.16: Hand pushed Lawnmower

Figure1.17: Collecting mechanism with metal blades

The Inwatic collects the balls with high speed, the two flexible blades are mention with the rotating rod to push the balls inside, an inclined surface allows the balls to move smoothly toward the storing area. Storing area can handle up to fifty tennis balls.

Chapter Two: Conceptual design and functional specifications

2.1 Introduction

2.2 Conceptual design schematic

2.3 Robot motion

2.4 Robot Navigation

2.5 Robot collecting system

2.6 Robot storing system

2.7 Operating modes

2.1 Introduction

This section describes the tennis ball collecting robot workflow, including the system components (subsystems), parts functions and relations between elements.

The tennis ball collecting robot is a differential drive mobile robot (DDMR) that has two independently-actuated wheels and is stabilized by a castor. The robot has its own on-board intelligence where it is equipped with a camera, sonars, and a computer (controller) embedded device. Figure (2.1) shows the robot in the tennis court.

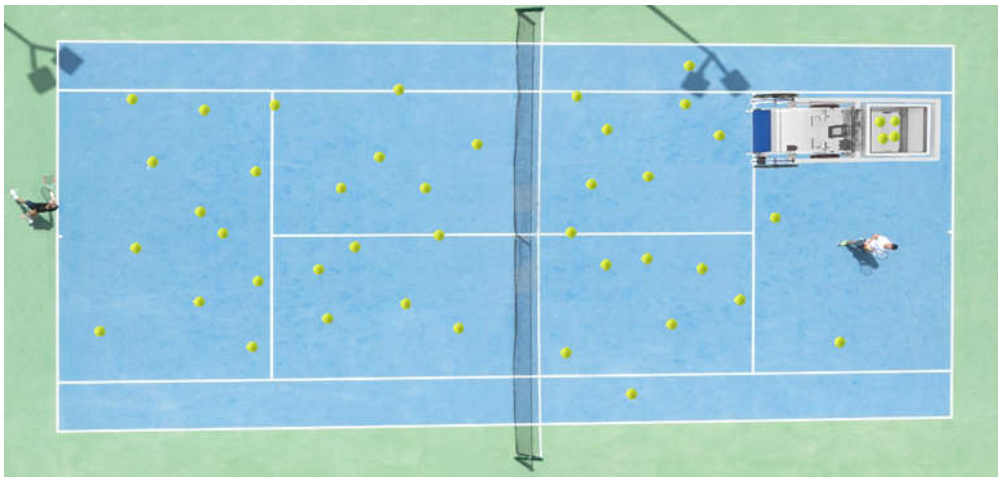


Figure 2.1:Tennis ball collecting robot in tennis court

2.2 Conceptual design schematic

Figure (2.2) explains the system conceptual design.

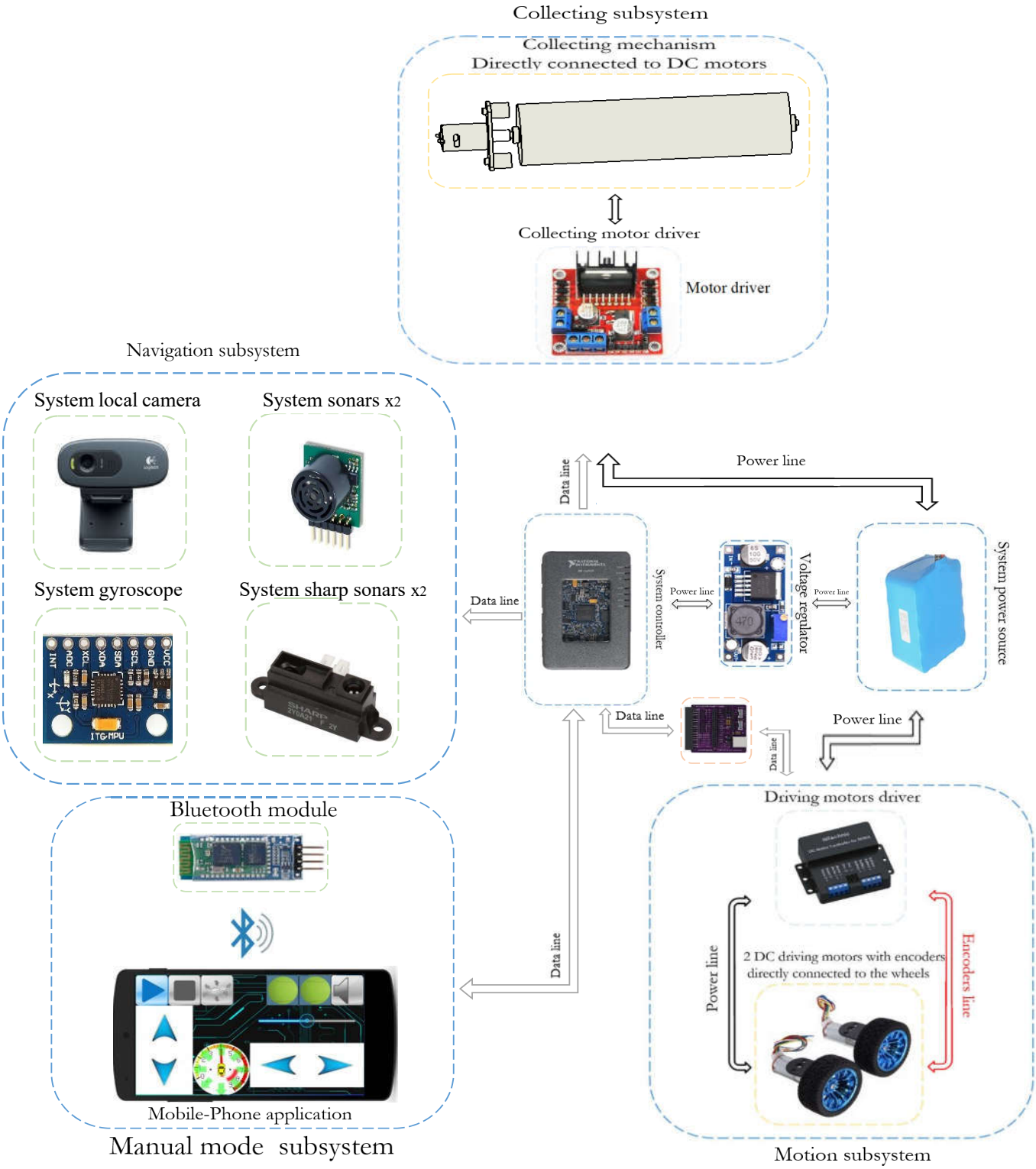


Figure 2.2: Conceptual design schematic

2.3 Robot motion

Figure (2.2) represents our tennis ball collecting robot driven by two independent wheels actuated by two DC motors with encoder for each one used to move the robot in different directions with controlled speed. The robot is stabilized by four castors used to support it and give relatively easy movement.

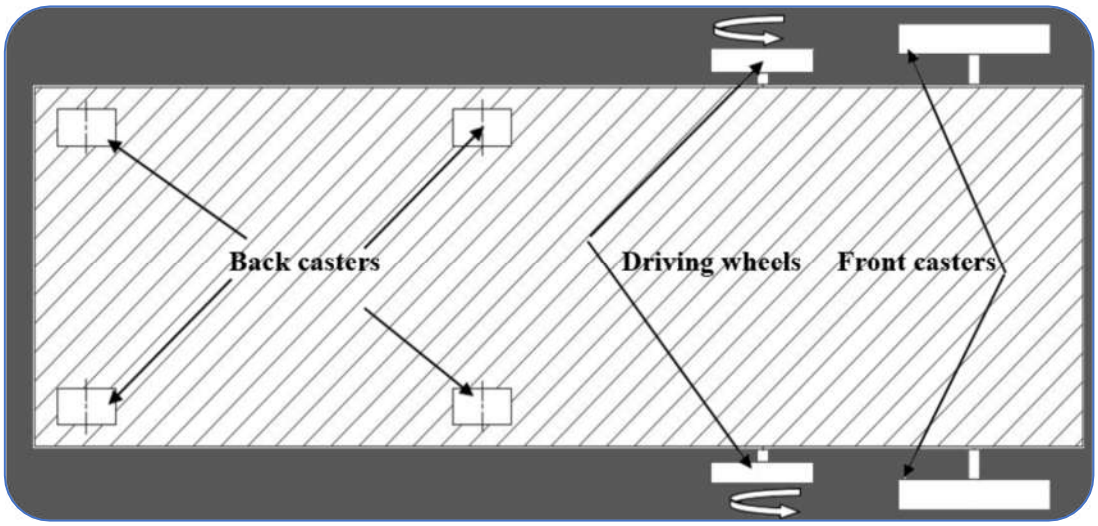


Figure 2.3: Robot configuration

Two front casters are used to drag the robot like travel-bag as shown in Figure (2.3) [14].

Figure 2.4: Drag the robot like travel-bag

2.4 Robot Navigation

The robot has two sonars and single camera used for perception and localization, they are connected with robot controller. The controller uses the sensors data to decide the robot motion to command the driving wheels motors.

2.5 Robot collecting system

The collecting process is the process of picking up the balls from field ground. The robot collection mechanism is based on a spinning drum that pushes tennis balls into a storing basket. A DC motor makes the drum spins at a desired speed.

2.6 Robot storing system

The robot has a lightweight basket on top of the robot used to store the picked balls. The basket has a capacity of fifty balls and it can be detach.

2.7 Operating modes

The robot operates autonomously / semi autonomously through two different modes. It can identify, localize and collect tennis balls autonomously or, alternatively, the human can command the robot through a mobile-phone application to drive it toward the balls and collect them.

Chapter Three: Design of structure and mechanical components

1.1 Introduction

2.2 Conceptual design schematic

2.3 Robot motion

2.4 Robot Navigation

2.5 Robot collecting system

2.6 Robot storing system

2.7 Operating modes

3.1 Introduction

In this chapter we will discuss the mechanical design of selected robot shown in Figure (3.1), calculation of each component in term of strength, geometry, durability and material properties will explained. Thus, these components will operate in the system without failure or defect.

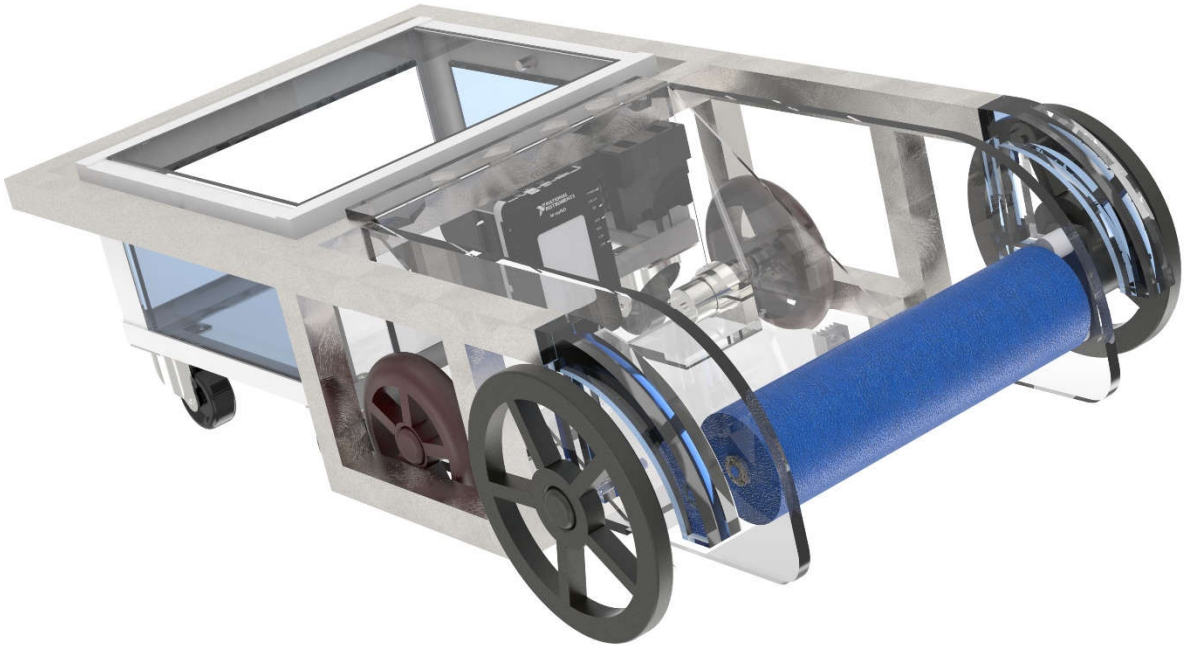


Figure 3.1: Selected tennis-ball collecting robot

3.2 Design requirements

The first step in designing any project is to define the project requirements. In other words, to define the specifications which your project supposed to do. Then you have to make many designs and choose one of them which satisfy your requirements, so the following requirements are listed for mechanical design:

Enable to collect fifty balls in five min.

1. Detachable storing area and allows for emptying process.
2. Power sources rechargeable battery.
3. Be safe for end-users.
4. User friendly.
5. Convenient interface.

3.3 Design through CATIA and material selection

The word CATIA is an acronym of Computer Aided Three-dimensional Interactive Application, and it is one of the "best" software program that are used for CAD, CAE and CAM, Figure (3.2) shows the CATIA software interface, and it is able to make part design for every part of any machine or project, and it is able to estimate stress and strain and load analysis for the machine project.

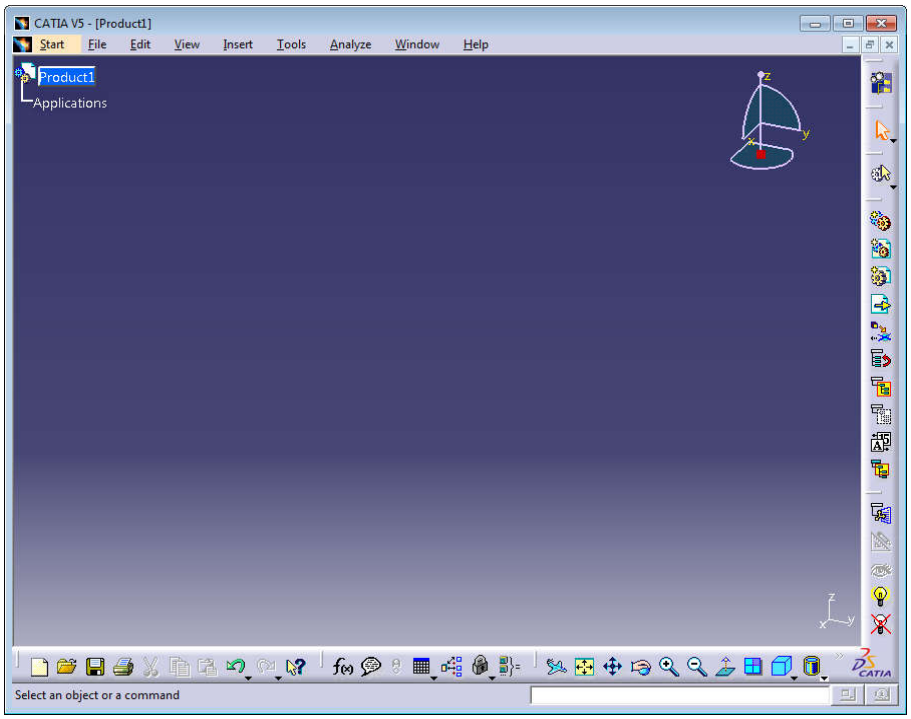


Figure 3.2: CATIA software

The mechanical components of candidate design are divided into four components as mentioned in Chapter 2 mechanical frame, wheels, collecting mechanism and storing area. In this section we will select the material for each component.

3.3.1 Mechanical frame

The robot mechanical frame consists from two main parts, which are the aluminum frame which forms the body of the robot and the supporting frame which carries all mechanisms, mechanical and electrical parts and the collecting mechanism. The mechanical frame must be heavy enough and made of high damping capacity material, as shown in Figure (3.3), an aluminum material is selected to manufacture for the mechanical frame.

Figure 3.3: Mechanical frame

The supporting frame is directly mentioned to the frame, it must have an ability to carry the electrical components and collecting mechanism, the plastic material is selected to manufacture since it has an ability to turning by the CNC machine. see Figure (3.4).

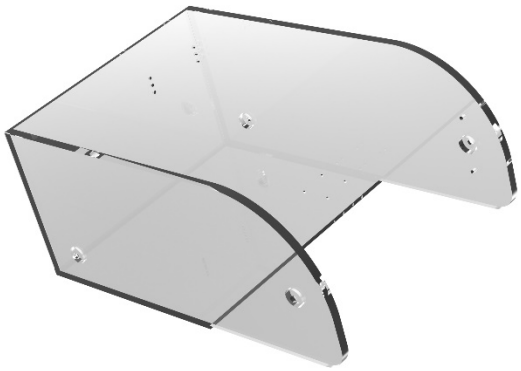


Figure 3.4: Mechanical supporting frame

In addition, the mechanical frame has a ramp and metallic cover. The ramp is designed with angle of 35 degree, it is made from light plastic material as shown in Figure (3.5). The ramp helps the balls to reach the storing area.

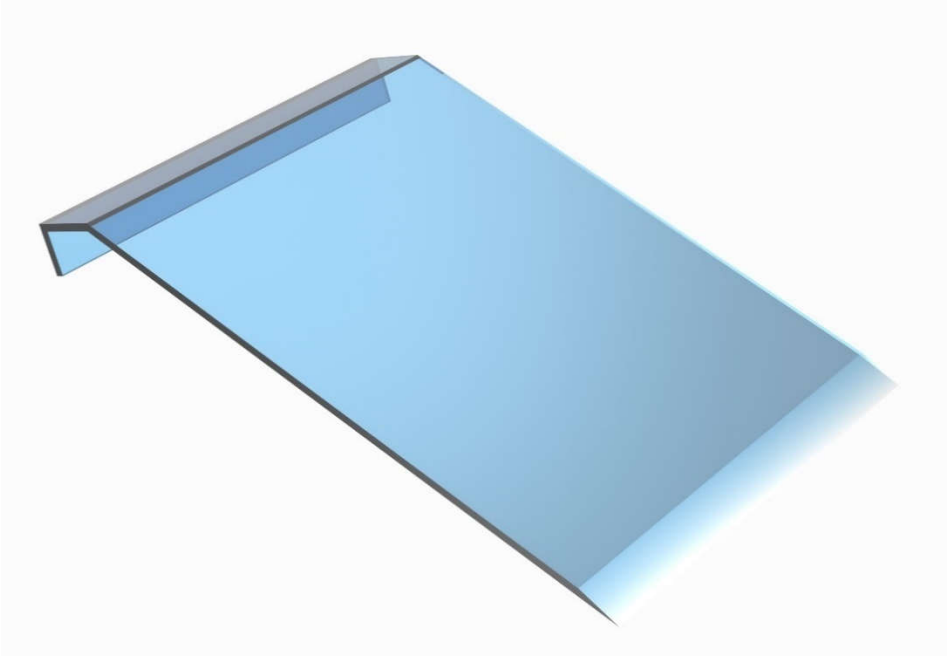


Figure 3.5: The ramp of robot

The metallic cover designed to cover the head of robot, it is made from light sheet material as shown in Figure (3.6).

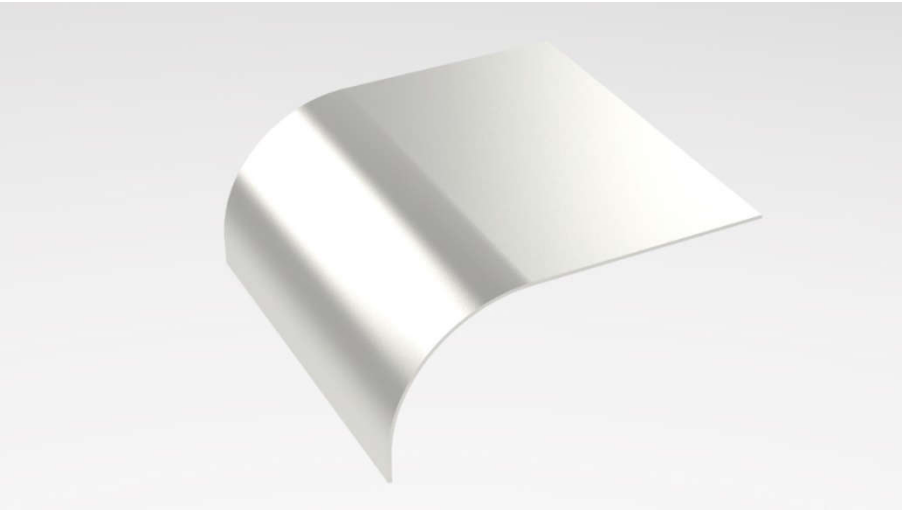


Figure 3.6: The metallic cover of robot

3.3.2 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 casters.

The driving wheels are used to drive the robot and make steering, it is directly mentioned with the driving motors gears. See Figure (3.7).

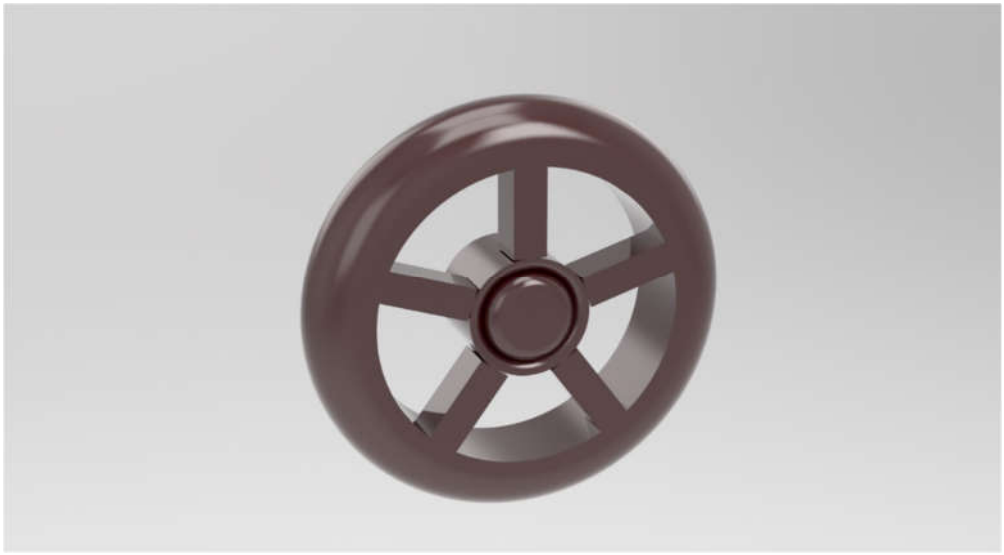


Figure 3.7: Driving wheel

Driving wheels must be strong and rigid enough to carry the robot load, the following specifications can be selected for the driving wheels [26]

- Diameter 140 mm
- Tire width 30 mm
- Tire cross-section (height) 23mm
- Width on the wheel bearing 29mm
- Inner diameter 8mm bearings
- Axes 8mm x 60mm
- Weight per wheel 256 g
- Plastic rim black
- PU (polyurethane) transparent medium hardness 82A
- Bearings: ABEC 7 for fast and easy rolling

The casters are used to enable relatively easy rolling movement and stability of the robot, our robot has front casters and back casters. The front casters that is shown in Figure (3.8) are used to manually drag the robot.

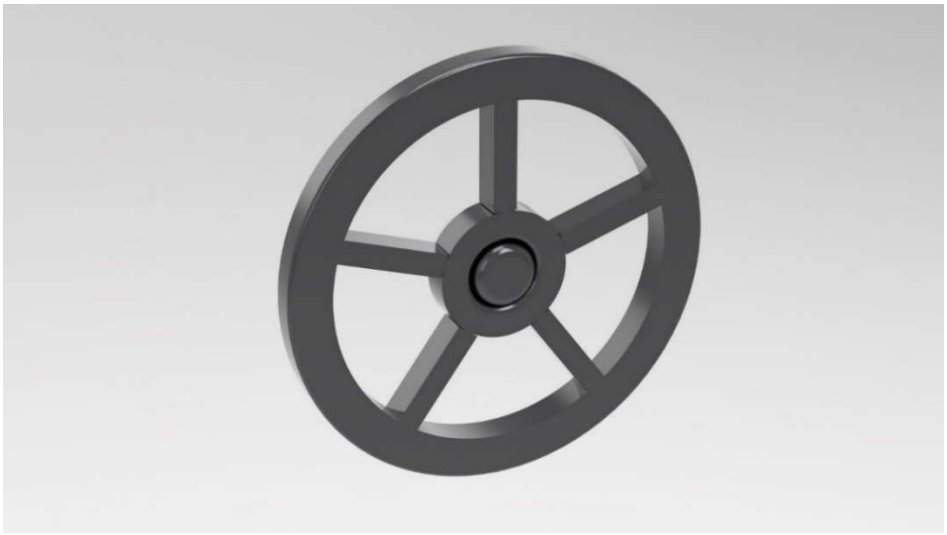


Figure 3.8: Front caster wheel

Front casters construct to manually drag (drive) the robot like a travel bag it must be strong enough to carry the load. The following specification can be selected for the front caster wheels [27].

- Material: Aluminum alloy (the hub), Rubber (the outer circumference)
- Outside Diameter: 200 mm
- Perimeter: 628 mm
- Thickness: 32 mm
- Inner Bore Diameter**:** 8 mm

The back casters that are shown in Figure (3.9) are used to give the robot equilibrium and stability, help the it to move freely and carry the storage area load.

Figure 3.9: Back caster wheel

3.3.3 Collecting mechanism

This part consists of metal cylindric rod made from plastic and multi plates of sponge as shown in Figure (3.10). The rod designed with following specifications:

- It has a capability to connect with motor to rotate it.
- It has elasticity compresses.

Figure3.10: Collecting mechanism rod

There are two bearings mentioned with the rod sides to reduce the friction to rotate the rod freely.

3.3.4 Storing area

This part designed to store the collected ball inside as shown in Figure (3.10), it has a capacity of fifty balls. The storing area designed to be detachable to enable for emptying.

Figure 3.11: storing area

3.4 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 10 Kg which equal approximately 100. The stress and deflection analysis are as follow.

3.4.1 Stress Analysis

Mobile Robot detailed design:

Aluminum 6061 alloy is selected to be the metal for the frame, because of it has a light weight and high stiffness, the yield strength is (56 MPa) and plastic (acrylic) for the storing area and the support frame, its yield strength is (45MPa). The mobile robot mechanical structure must not exceed a deflection of 0.23 mm. 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 stress analysis will be perform using SolidThinking by applying 30 N vertically downwards at the storing area and 20 N vertically downwards at the support frame. The stress equal to $3.797MPa$ shown in the following Figure (3.12), location where the maximum stress occurs shown in Figure (3.13).

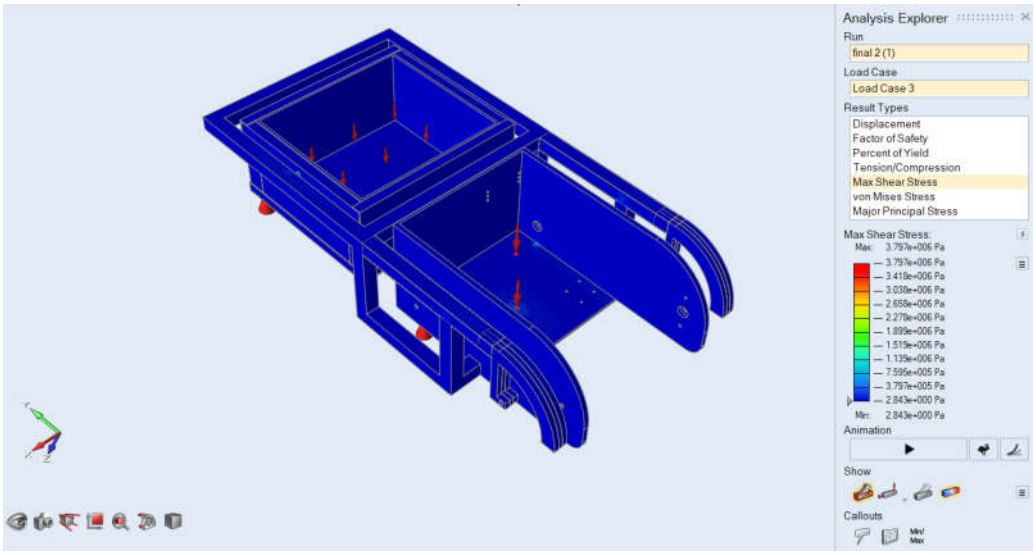


Figure 3.12: Stress result due to the mass of the structure

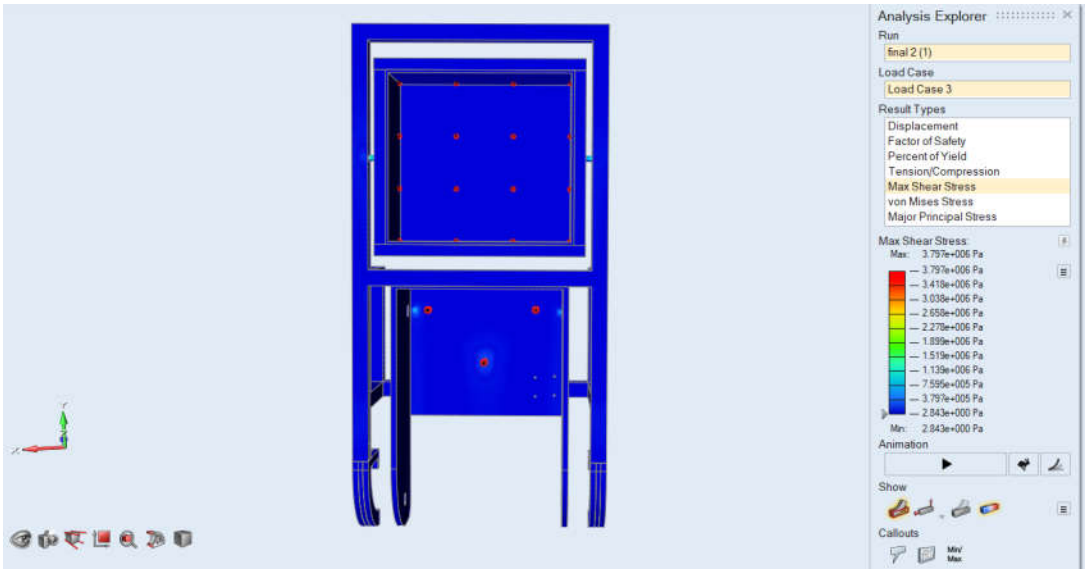


Figure3.13: Location where the maximum stress occurs

The results show that the maximum von Mises stress is 6.6661 MPa whereas the maximum bending stress in the Aluminum part is $56MPa$ and that in the plastic (acrylic) pars is $45MPa$. This leads to the following minimum factors of safety:

$$n_{Al} = \frac{\sigma_u}{\sigma_e} = \frac{56}{6.661} = 8.407,$$

$$n_{plastic} = \frac{\sigma_u}{\sigma_e} = \frac{45}{6.661} = 6.756,$$

The Figure (3.14) below shows the minimum factor of safety.

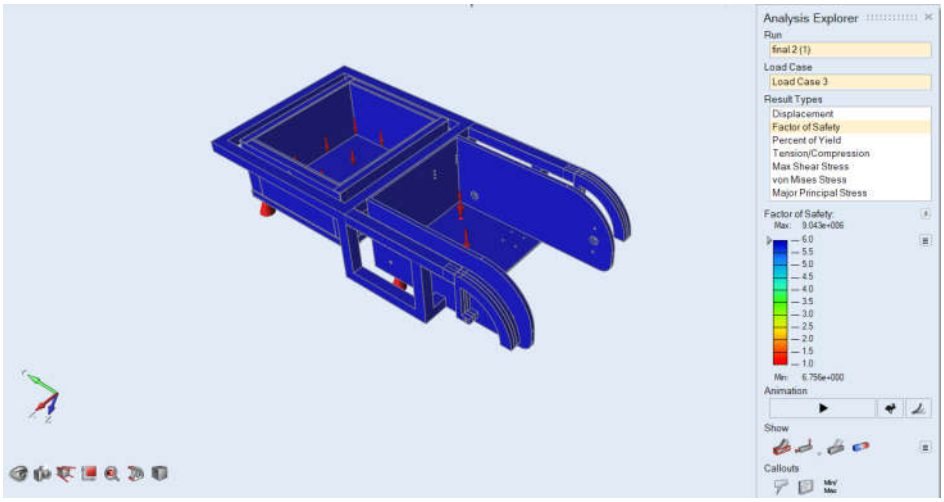


Figure 3.14: the minimum factor of safety

The deflection due to bending shown in the Figure (3.15), and the maximum deflection is $0.24\ mm$.This deflection is acceptable and does not contradict with assumption criteria of the design.

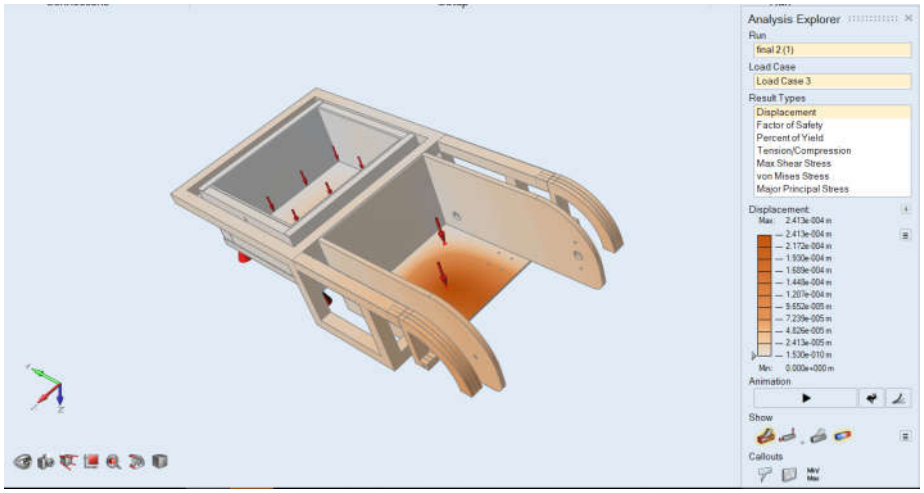


Figure 3.15: The maximum deflection

Chapter Four: Kinematics and dynamics of the mobile robot

4.1 Introduction

4.2 Non-Holonomic motion

4.3 Kinematics model

4.4 Dynamic model

4.5 Actuator dynamics

4.1 Introduction

In this chapter kinematics and dynamics of a differential drive mobile robot are derived and the differences between the two models and limitations of the kinematic model are explained [16].

4.2 Non-Holonomic motion

4.2.1 Introduction

Wheels are by far the most common mechanism to achieve locomotion in mobile robots. Any wheeled vehicle is subject to kinematic constraints that reduce its local mobility [15], for instance, a car can reach any final configuration in its plane, but it cannot move sideways. depending on the goal configuration, it requires to perform a series of maneuvers (such as parallel parking) to reach the desired state.

So, the holonomic and non-holonomic systems have to be defined. Consider a mechanical system whose configuration $q \epsilon C$ is described by a vector of generalized coordinates, where C is the configuration space of the proposed system and coincides with R^n. For considered system, a constraint is called Kinematic when it only involves generalized coordinates (q) and velocities ($\dot{q}$).

Kinematic constraints are generally expressed in *Pfaffian form* [16] (i.e. they are linear in the generalized velocities) as follow:

$$a_i^T(q)\dot{q} = 0 \qquad i = 1, \dots, k < n \tag{4.1}$$

Or in matrix

$$A^T(q)\dot{q} = 0 \tag{4.2}$$

A mechanical system for which all the kinematic constraints are integrable to the form (4.3) where, m_i is the integration constant, is called *holonomic* (or *integrable*) system.

$$h_i(q) = m_i \qquad i = 1, \dots, k < n \tag{4.3}$$

On the other hand, a mechanical system that is subject to at least one nonintegrable constraint is called nonholonomic (or nonintegrable) system. Although such constraint limits the local mobility of the system, due to its non-integrable nature, the accessibility to C is not affected. Hence, the generalized coordinates are not reduced.

4.2.2 Non-holonomic constraint

Wheels are typical sources of non-holonomic constraints. Consider the disk in Figure (4.1) with generalized coordinates $q = [x\ y\ \theta]^T$, where (x, y) are the Cartesian coordinates of the contact point of the wheel with the ground and θ is the orientation of the wheel with respect to the x axis. Due the non-holonomic motion, the two following assumptions are considered:

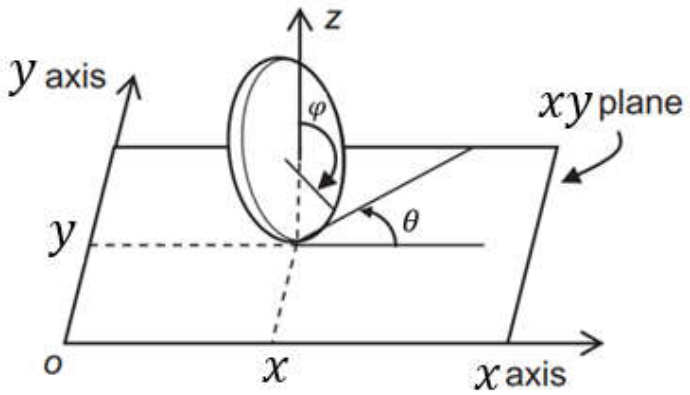


Figure 4.1: Generalized coordinates for a disk rolling on a plane

- **No lateral slip motions**

This constraint is described in Equation (4.4) which simply means that the robot can move only in a curved motion (forward and backward) but not sideward where $\dot{y}_r$ is the robot velocity along y axis, that means there is no velocity component for the contact point perpendicular to the plane containing the disk.

$$\dot{y}_r = 0 \tag{4.4}$$

- **Pure rolling constraint**

The pure rolling constraint which expressed in Equations (4.5) and (4.6) [23] represents the fact that each wheel maintains a one contact point (P) with the ground as shown in Figure (4.2).

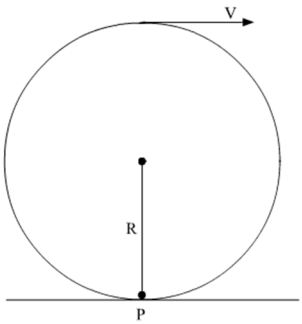


Figure 4.2: Pure rolling motion constraint

$$\dot{x}\sin\theta - \dot{y}\cos\theta = 0 \tag{4.5}$$

By using *Pfaffian* form, equation 4.4 can written as following:

$$[\sin\theta \quad -\cos\theta \quad 0]\dot{q} = 0 \tag{4.6}$$

Equation 4.6 is not integrable causing the nature of the wheel to be nonholonomic.

4.3 Kinematics model

4.3.1 Introduction

Differential drive mobile robot (DDMR) is considered as unicycle vehicle that has a single orientable wheel [16], its configuration is completely described by q.

Kinematic modeling is the study of the motion of mechanical systems without considering the forces that affect the motion, so the goal is to represent the robot velocities as a function of the driving wheels velocities along with the geometric parameters of the robot.

4.3.2 Robot kinematics

Consider the mobile robot in Figure (4.3). Using generalized coordinate vector $q = [x\ y\ \theta]^T$, the robots posture can be defined on its whole configuration space.

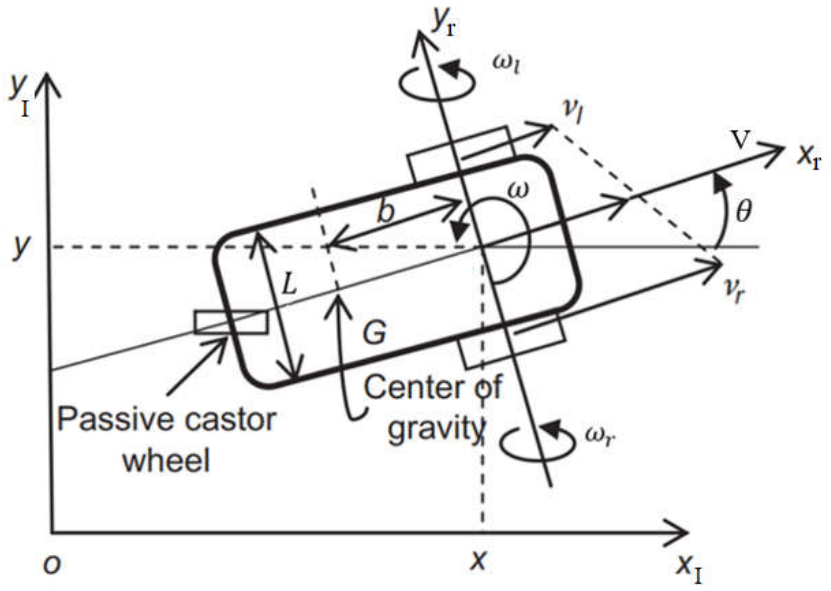


Figure 4.3: Differential drive mobile robot

Consider the pure rolling constraint in equation (4.6) entailing that the velocity of point P in Figure (4.2) is zero. Consider the following matrix:

$$G(q) = [g_1(q) \quad g_2(q)] = \begin{bmatrix} cos\,\theta & 0 \\ sin\,\theta & 0 \\ 0 & 1 \end{bmatrix} \tag{4.7}$$

whose columns $g_1(q)$ and $g_2(q)$ are a basis of the null space of the matrix associated with the Pfaffian constraint [16]. All the admissible generalized velocities at q are therefore obtained as a linear combination of $g_1(q)$ and $g_2(q)$. The kinematic model of the unicycle is then

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos\,\theta \\ sin\,\theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \tag{4.8}$$

where, the inputs have clear physical interpretation, v and ω are the linear velocity and angular velocity of the robot, respectively, as shown in Figure (4.3).

It is important to consider that there is a one-to-one correspondence between v and ω and the actual velocity inputs $w_R$and w_L of the right and left wheel [20], respectively, as follow

$$v = \frac{R(\omega_R+\omega_L)}{2} \quad \omega = \frac{R(\omega_R-\omega_L)}{2L} \tag{4.9}$$

where R is the radius of the wheels and L is the distance between their centers.

4.4 Dynamic model

The robot kinematics do not represent the actual inputs of the robot motors (i.e. forces and/or torques). In another words, we consider that the motors give enough torques to drive the robot when dealing just with a kinematic model. The dynamic model of the DDMR is essential for simulation, analysis of robot motion, and for the design of motion control algorithm.

A non-holonomic DDMR with n generalized coordinates $(q_1, q_2, \dots, q_n)$ and subject to m constraints can be described by the following equations of motion:

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) = B(q)\tau - A^T(q)\lambda \tag{4.10}$$

Where,

$M(q)$ an $n \times n$ symmetric positive definite inertia matrix, $V(q, \dot{q})$, is the centripetal and Coriolis matrix, $G(q)$ is the gravitational vector, $B(q)$ is the input matrix, τ is the input vector, $A^T(q)$ is the matrix associated with the kinematic constraints, and λ is the Lagrange multipliers vector [17].

There are two methods to derive the dynamic model for the DDMR. *Newton-Euler* method that describes the system in terms of all the forces and momentum acting on the system based of direct interpretations of Newtons Second Law of Motion. And the other one *Lagrange* method that derive the equations of motion by considering the kinetic and potential energies of the given system. *Lagrange* method is chosen to derive the system model.

4.4.1 Lagrange dynamic approach

The Lagrangian L of the mechanical system as the difference between its kinetic and potential energy as follow:

$$L = T - V \tag{4.11}$$

Where, L is the Lagrange function, T, and V are kinetic and potential energies respectively.

The Lagrange equation can be written in the following form:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) + \frac{\partial L}{\partial q_i} = F - A^T(q)\lambda \tag{4.12}$$

Where,

q_i is the generalized coordinates, F is the generalized force vector, A is the constraints matrix, and λ is the vector of Lagrange multipliers associated with the constraints [17].

The first step in deriving the dynamic model using the Lagrange approach is to find the kinetic and potential energies, since the differential drive mobile robot is moving in the X_I-Y_I plane, the potential energy is considered to be zero.

Consider the DDMR shown on Figure (4.3), the kinetic energy of the robot is given as follow:

$$T = \frac{1}{2}mv^2 + \frac{1}{2}I\dot{\theta}^2 \tag{4.13}$$

Where,

m is robot total mass, I is the moment of inertia of the robot about the vertical axis.

The velocity v is computed as a function of the generalized coordinates using the general velocity equation in the inertial frame.

$$v^2 = \dot{x}^2 + \dot{y}^2 \tag{4.14}$$

Where x and y are components of the center of robot.

The kinetic energy of the robot system become

$$T = \frac{1}{2}m\dot{x}^2 + \frac{1}{2}m\dot{y}^2 + \frac{1}{2}I\dot{\theta}^2 \tag{4.15}$$

Applying equation (4.12) in Lagrangian function, $L = T$, the equations of motion is given by

$$m\ddot{x} = C_1 \tag{4.16}$$

$$m\ddot{y} = C_2 \tag{4.17}$$

$$I\ddot{\theta} = C_3 \tag{4.18}$$

Where,

(C_1, C_2 and C_3), are coefficients related to the kinematic constraints, which can be written in terms of the Lagrange multipliers vector λ and the constraints matrix A.

$$A^T(q) = \begin{bmatrix} -sin\,\theta \\ cos\,\theta \\ 0 \end{bmatrix} \tag{4.19}$$

Now, the obtained equations of motion (4.16) to (4.18) can be represented in the general form given by Equation (4.10)

$$M(q)\ddot{q} = B(q)\tau - A^T(q)^{\lambda} \tag{4.20}$$

where,

$$M(q) = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix}$$

$$B(q) = \frac{1}{R}\begin{bmatrix} cos\,\theta & cos\,\theta \\ sin\,\theta & sin\,\theta \\ L & -L \end{bmatrix}$$

for the purpose of control and simulation, the system described by Equation (4.20) will transformed into an alternative form which is more convenient [23].

We should eliminate the constraint term $A^T(q)\lambda$ in equation (4.20) since the Lagrange multipliers λ_i are unknown. This is done first by defining a matrix G(q) which is described in equation (4.7) whose columns are a basis for the null space of $A^T(q)$ [16], so that

$$G^T(q).A^T(q) = 0 \tag{4.21}$$

From equation (4.21) we note the following

$$\dot{q} = G(q)\eta \tag{4.22}$$

where η is the vector whose components of this vector are the driving velocity *v* and the steering velocity *ω.*

Next, taking the time derivative of Equation (4.22) gives

$$\ddot{q} = \dot{G}(q)\eta + G(q)\dot{\eta} \tag{4.23}$$

Substituting Equation (4.23) in the main Equation (4.20), we obtain

$$M(q)\left[\dot{G}(q)\eta + G(q)\dot{\eta}\right] = B(q)\tau \tag{4.24}$$

Rearranging Equation (4.24) and multiplying both sides by $G^T(q)$ leads to

$$G^T(q)M(q)G(q)\dot{\eta} \; + \; G^T(q)M(q)\dot{G}(q)\eta \; = G^T(q)B(q)\tau \tag{4.25}$$

where the last term is identically zero. The new matrices

$$\bar{M}(q) = G^T(q)M(q)G(q) \tag{4.26}$$

$$\bar{V} = G^T(q)M(q)\dot{G}(q) = 0 \tag{4.27}$$

$$\bar{B} = G^T(q)B(q) \tag{4.28}$$

The dynamic equations are reduced to the form

$$\bar{M}(q)\dot{\eta} = \bar{B}(q)\tau \tag{4.29}$$

where,

$$\bar{M}(q) = \begin{bmatrix} m & 0 \\ 0 & I \end{bmatrix} \tag{4.30}$$

$$\bar{B}(q) = \frac{1}{R}\begin{bmatrix} 1 & 1 \\ L & -L \end{bmatrix} \tag{4.31}$$

Equation (4.29) shows that the robot dynamics are expressed only as a function of the robot linear and angular accelerations $(\dot{v}, \dot{\omega})$ and the driving motor torques (τ_R, τ_L). For the simulation purpose the equation of motion (4.29) can be also transformed into an alternative form as follows

$$\dot{v} = \frac{1}{mR}(\tau_R + \tau_L) \tag{4.32}$$

$$\dot{\omega} = \frac{L}{IR}(\tau_R - \tau_L) \tag{4.33}$$

4.5 Actuator dynamics

The DC motors are generally used to drive the wheels of the robot. Since the wheels motor are symmetrical, it is considered to derive the model for one of them. There are two classes of DC motors control which are a filed-current controlled and armature-voltage controlled.

An armature-controlled dc motor shown in Figure (4.4) [22] which is the case for our robot system, the armature voltage V_a is used as the control input while keeping the conditions in the field circuit constant [20]. For a permanent-magnet dc motor, we have the following equations for the armature circuit:

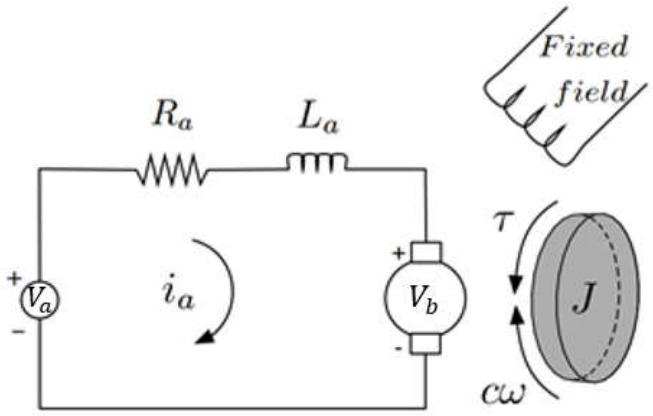


Figure 4.4: Circuit equivalent of a DC motor

In an armature-voltage controlled structure, the motor torque is linearly dependent on the armature current by

$$\frac{\tau_m(s)}{I_a(s)} = K_m \tag{4.34}$$

where,

τ_m(s) is the motor torque in S-domain and K_m is called the motor torque constant.

Based on circuit model provided in Figure (4.4), and considering the back-EMF voltage (V_b), induced by the rotation of armature winding, the voltage relation on the armature will be

$$V_a = V_R + V_L + V_b \tag{4.35}$$

Where, V_R, V_L are voltages across R_a,L_a respectively.

Back-EMF has a linear relation to the motor angular speed (ω_m) through back EMF constant K_b as follow

$$V_b = K_b \omega_m \tag{4.36}$$

By substitute equation (4.36) in equation (4.35) and taking the Laplace transform, the following equation is achieved

$$V_a(s) - V_b(s) = V_a(s) - K_b \omega_m(s) = (R_a + L_a s) I_a(s) \tag{4.37}$$

Now consider the wheel of the robot has inertia of J shown in Figure (4.5) [22] connected to the motor has a damping friction c

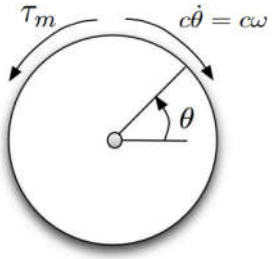


Figure 4.5 :Torque applied to a free body

The resulting torque across the block is given by

$$\tau_m = J\dot{\omega}_m + c\omega_m \tag{4.38}$$

Taking Laplace transform of Equation (4.38) the following transfer function is achieved

$$\frac{\omega_m(s)}{T_m(s)} = \frac{1}{Js + c} \tag{4.39}$$

Using Equations (4.34 to 4.39) the transfer function from armature voltage to angular velocity is

$$\frac{\omega_m(s)}{V_a(s)} = \frac{K_m}{L_a s + R_a(J_{eq}s + c)K_b K_m} \tag{4.40}$$

Closed loop block diagram of DC motor model expressed in Equation (4.40) is shown in Figure (4.6)

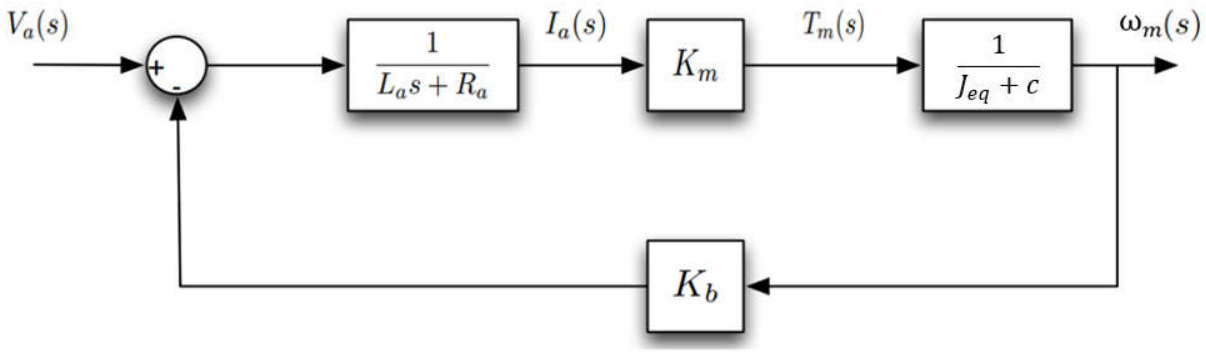


Figure4.6: DC motor closed loop model

Table 4.1 shows the dynamic model parameters and their nominal values of driving motors which are obtained based on calculations and from motor data sheets [24].

Table 4. 1: DC driving motor parameters

Parameter	Description	Nominal Value
K_m	Torque Constant	0.44 N.m/Amp
R_a	Armature Inductance	3 ohm
R	Wheel Radius	0.065 m
M	Mass	7 K_g
J_{eq}	Inertia	3.7119 $K_g.m^2$
L	Distance between wheels	0.27 m
β	Friction Constant	0.056 N.m.s
K_b	Back emf Constant	0.76 V/(rad/sec)
L_a	Armature Inductance	0 H
N	Gear Ratio	40

Chapter Five: Selection of electrical actuators and processing unit

5.1 Introduction

5.2 Processing unit

5.3 Component selection

5.4 Processing unit selection

5.1 Introduction

This chapter introduces the design of electrical and information processing subsystems, explaining the motors design, electric drives, encoders, sensors and controller.

5.2 Electrical parts

The robot electrical parts include sensors, motors and interfacing circuits that connect these sensors and actuators with the system controller. The robot electrical parts are listed as follow:

5.2.1 Robot motors

Based on the conceptual design that is presented in chapter 2, the robot has two types of motors which are the driving motors and the collecting mechanism motor.

1) Driving motors

Since the operation of the system requires change the robot direction which necessary to make the robot steering, it is important for the system driving motor used in such a system to have:

- The ability to generate enough torque independently of the speed of the motor.
- The ability to control the speed independently of the torque value.
- Simple and precise control.

To satisfy the above tasks, the driving motors must generate enough torque and speed to drive the robot with the desired motion at any state (no load to full load state), so the following calculation done for the driving motors:

To choose the motor we should first identify the following specifications

- Total mass of robot (m)= 10 [Kg].
- The desired velocity (V)= 2 [m/s].
- Radius of drive wheel (r)= 0.07 [m].

Now the maximum acceleration that is needed to drive the robot is calculated based on straight line acceleration motion equations

$$x = V_0 t + \frac{1}{2} a t^2 \tag{5.1}$$

where, V_0 is the ball initial velocity, t is the time of movement and a is the ball desired acceleration.

That gives the desired acceleration $= 1.6\ [m/s^2]$

Based on Newton formula, the force [F] needed to move any inertia [m] is given by the formula

$$F = ma \tag{5.2}$$

That gives the maximum force $F = 16\ [N]$.

To estimate the motors torque, we apply the following formula

$$\tau = Fr \tag{5.3}$$

But we have two wheels with its motors, so the needed torque τ= 0.56 [Nm].

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

$$\omega = \frac{V}{r} * \frac{60}{2\pi} \tag{5.4}$$

That gives the robot speed in [rpm] w= 272.83 [rpm].

2) Collecting mechanism motor

The collecting motor makes the brush spins at a desired speed that calculated based on equation under constant acceleration motion Equations (5.5 to 5.9) and experimentally verified.

$$v^2_{f_{ball}} = v^2_{i_{ball}} - 2gy \tag{5.5}$$

$$v^2_{f_{ball}} = (v_{i_{ball}} \sin\theta)^2 - 2gy \tag{5.6}$$

$$0 = (v_{i_{ball}} \sin\theta)^2 - 2gy \tag{5.7}$$

$$v^2_{i_{ball}} = \frac{2gy}{\sin^2\theta} \tag{5.8}$$

$$v_{i_{ball}} = v_{i_{motor}} = \sqrt{\frac{2gy}{\sin^2\theta}} \tag{5.9}$$

Equations (5.5 to 5.9) are valid for the consideration of the ball center of mass velocity so the double of obtained velocity $v_{i_{motor}}$ should be taken.

5.2.2 Motors drivers

The motor driver is a device that acts as intermediary between the robot microcontroller, batteries and motors. The robot microcontroller has a low output voltage signal, and the

actuators a high voltage signal to operate, so the driver used to amplify the signal, control the direction of actuators and it protect microcontroller from reverse current comes from motors. The driver should be able to provide the required power output as well as compatible to interface with the system controller.

5.2.3 Sensors

Different types of sensors are required to complete the robot tasks as follow:

1) Displacement sensors

Designing a real time closed loop feedback control system is essential to control the speed and position of the motors and satisfy the path planning of motion which requires a sensor to be relatively noiseless with a fast response, so the information received from sensors accurately reflects the state of the system.

2) Range sensor

Range sensor used to measure the range between the robot and the rigid objects around it. It is required for robot navigation and operating avoiding obstacles.

3) Orientation sensor

In order to sense the robot direction and correct its orientation, motion sensor should be used to ensure desired orientation (without drift).

4) Camera sensor

Vision is the most powerful sense. It provides an enormous amount of information about the environment and enables rich, intelligent interaction in dynamic environments. So it is required a sensors to capture a series of images and process it.

5.2.4 Power source

As an essential requirement of electronic device, power must be provided to support all the internal components function well. For this specific autonomous robot system, power source should be portable and powerful enough to supply the main energy consumption from robot's motion, as well as supporting all the other components in various situations. Table 5.1 shows the robot components power consumption.

Table 5.1: Robot components power consumption

myRIO-1900	100 mA as a maximum current on each connector
Driving motor	1.3 A for each motor
Collecting motor	0.78 A
Ultrasonic	15 mA
GP2Y0A21 Sharp distance sensor	40 mA

Summation of currents that needed 3.590A, so the battery requires to operate mobile robot is 4Ah.

5.3 Processing unit

As a mechatronic application, computer and information system are essential parts of the mobile robot system. In this project, these parts are

- Personal Computer is hardware used for microcontroller programming and implementation, a set of software packages that are used to design, simulate, and control the entire system.

- Real hardware controller that allows the robot acts as a complete stand-alone system.

5.4 Component selection

5.4.1 Motors selection

Tennis ball collecting robot required two types of motors, which are the driving motors and a collecting mechanism motor

1) Driving motors

Two motors are required to drive the robot. Driving motors should be identical. The driving motor is shown in Figure (5.1) are selected with specifications shown in Table 5.2 [28].

Figure5.1: Driving motors

Table 5.2: Driving DC motor specifications

Input Voltage	12V
No Load speed	150 rpm
Loading speed	137.5 rpm
Rated torque	3.5 Kg.cm
Rated current	**0.9 A**

2) Collecting mechanism motor

Collecting mechanism motor is required to spins the brush with a desired speed. The motor shown in Figure (5.2) is selected with specifications shown in Table 5.3 [29].

Figure 5.2: Collecting mechanism motor

Table 5.3: Collecting mechanism DC motor specifications

Input Voltage	16 V
No Load speed	6300 rpm
Loading speed	5350 rpm
Rated torque	15.8 N.m
Rated current	0.78A

5.4.2 Motors drivers

The motor driver used to amplify the controller signal and protect it from motor generated current, it is also used to control the motor speed and direction two types of driver are used in this project as follow

1) Driving motors driver

This type of drivers used with the robot driving DC motor. A HiTechnic DC Motor Controller shown in Figure (5.3) is used. The motor controller has connections for two DC motors and two encoders, it has two H-bridge outputs to control the speed and direction of two DC motors.

Figure5.3: HiTechnic DC Motor Controller

2) Collecting mechanism motor driver

A L298N DC motor driver shown in Figure (5.4) is used. The L298N is a dual H-Bridge motor driver which allows speed and direction control of two DC motors at the same time. The module can drive DC motors that have voltages between 5 and 35V, with a peak current up to 2A.

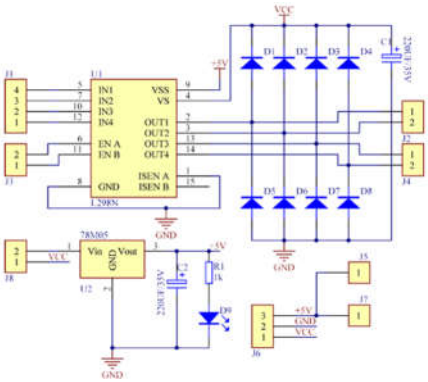


Figure 5.4: L298N DC motor driver schematic

5.4.3 Sensors selection

1) Displacement sensors

Encoders are common type of displacement sensors. There are different types of encoders, one of them is the optical encoder which is a type of rotary encoder that uses a sensor to identify position change as light passes through a patterned encoder disk. The selected encoders are shown in Figure (5.5) with specifications shown in Table 5.4 [28].

Figure 5.5: Optical encoder

Table 5.4: Sonar sensor specifications

Input Voltage	5 V
Pulses per revolution.	1,440 pulses
Weight	0.3 kg

2) Range sensors

The range sensors are used to measure the distance between the robot and any physical object. A common type of range sensors are the sonars which is shown in Figure (5.6). The sonars do not require physical contact with the object is being detected. It allows the robot to sense an obstacle without having to come into contact with it that prevent possible entanglement, allows for better obstacle avoidance and possibly allows the robot to distinguish between obstacles of different shapes and sizes.

Figure 5.6: Sonar sensor

The sonar shown in Figure 5.6 is selected with the following specifications that are shown in Table 5.5 [30].

Table 5.5: Optical encoder specifications

Operating voltage	DC 3 - 5 V
Measuring range	20 cm – 720 cm
Resolution	1 cm
Response frequency	15 HZ (full range)
Operating current	9 mA
Weight	5 g

Another type of range sensors called Sharp GP2Y0A21YK0F is used with specifications shown in Table 5.6 [31], it detects the obstacles with small distance. Sharp sonar which shown in Figure (5.7) is selected to be mentioned at the robots back edges.

Figure 5.7: Sharp sensor

Table 5.6: Sharp GP2Y0A21YK0F sensor specifications

Operating voltage	DC 4.5-5.5 V
Measuring range	10 cm - 80 cm
Resolution	1 cm
Update period	38 ± 10 ms
Operating current	30 mA
Weight	3.5 g

3) Orientation sensor

Orientation sensor is used to detect the motion direction. There are wide types of orientation sensors. One type is called MPU6050 which show in Figure (5.8) is selected with specifications shown in Table 5.7 [32]. The MPU6050 is used as a gyroscope sensor to ensure desired orientation.

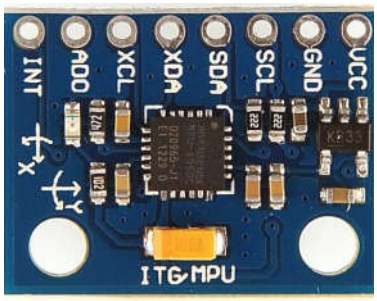


Figure 5.8: MPU6050 sensor

Table 5.7: MPU6050 sensor specifications

Operating voltage	DC 3 V-5 V
Communication modes	standard IIC communications protocol
Gyroscope range	±250, ±500, ±1000, and ±2000 °/sec (dps)
Acceleration range	±2g, ±4g, ±8g, and ±16g.

5.4.4 Power source selection

A rechargeable lithium ion battery shown in Figure (5.10) is used to supply the main energy consumption from robot system which has the specifications shown in Table 5.9.

Figure 5.9: Lithium ion battery

Table 5.8: Lithium ion battery specifications

Manufacturer	J&Y
Chemistry	Li-ion
Cell Type	18650
Voltage	12 V
Capacity	4 Ah
Charge Mode	CC-CV
Discharge Current	2 A
Charge Temp	0~45 ºC
Discharge Temp	-20~60 ºC
Weight	0.9 kg
Dimension	60*60*65 mm
Cycle Life	500 times

5.5 Processing unit selection

The NI myRIO device shown in Figure (5.11) is used to allow the robot acts as a complete stand-alone system. MyRIO is a real-time embedded evaluation board and made by National Instruments. It is programmed using the LabVIEW programming language and development environment. NI myRIO which shown in Figure (5.11) has the following specifications shown in Table 5.10 [33].

Figure 5.10: NI myRIO computer

Table 5.9: myRIO specifications

Processor type	Xilinx Z-7010
Processor speed	667 MHz
Processor cores	2
Nonvolatile memory	512 MB
USB host port	USB 2.0 Hi-Speed
Wireless Radio mode	IEEE 802.11 b,g,n
Rated voltage	6 – 16 V
Operating current	1.5 A max

Chapter Six: Planning

6.1 Planning problem

6.2 Path and timing law

6.3 Differential flatness

6.4 Path planning

6.5 Planning via Cartesian polynomials

6.6 Planning results

6.1 Planning problem

For the execution of a specific robot task, mobile robot needs to translate from initial pos $(x_i\ y_i\ \theta_i)$ to final pos $(x_f\ y_f\ \theta_f)$ through specific time (ΔT), given two motor torques and nonholonomic constraints mentioned in Section (4.1). 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 $q(t)$, 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 $\Delta T = t_i - t_f$. The trajectory $q(t)$ can be broken down into a geometric path $q(s)$, where $\frac{dq(s)}{ds} \neq 0$ for any value of 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

$$\dot{q} = \frac{dq}{dt} = \frac{dq}{ds}\dot{s} = q'\dot{s} \tag{6.1}$$

where the prime symbol denotes differentiation with respect to s. The generalized velocity vector is then obtained as the product of the vector q' by the scalar $\dot{s}$.

The Nonholonomic constraints in equation (4.2) can be re expressed as

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

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 (4.4,4.5) the following condition for geometric admissibility of the path is considered

$$[\sin\theta \quad -\cos\theta \quad 0]q' = x'\sin\theta - y'\cos\theta = 0 \tag{6.3}$$

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

$$x' = \tilde{v}\cos\theta \tag{6.3}$$

$$y' = \tilde{v}\sin\theta \tag{6.4}$$

$$\theta' = \tilde{\omega} \tag{6.5}$$

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

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

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

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

$$x' = \mathrm{f(x)} + \mathrm{G(x)u} \tag{6.9}$$

$$y = h(x) + d(x)u \tag{6.10}$$

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.

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

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

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

$$\theta(s) = a\tan^2(y'(s), x'(s)) \; + \; k\pi \qquad k = 0{,}1 \tag{6.13}$$

where, k defines if the robot is moving forward ($k \; = \; 0$) or backward ($k \; = \; 1$) and $a\tan^2$ 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:

$$\tilde{v}(s) = \pm\sqrt{(x'(s))^2 + (y'(s))^2} \tag{6.14}$$

$$\tilde{\omega}(s) = \frac{y''(s)x'(s) - x''(s)y'(s)}{(x'(s))^2 + (y'(s))^2} \tag{6.15}$$

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 $\boldsymbol{q}(s_i) = \boldsymbol{q_i} = [\, x_i \; y_i \; \theta_i]^T$ to a final configuration $\boldsymbol{q}(s_f) = \boldsymbol{q_f} = [\, x_f \; y_f \; \theta_f]^T$

5.5 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

$$x(s) = s^3 x_f - (s-1)^3 x_i + \alpha_x s^2 (s-1) + \beta_x s (s-1)^2 \tag{6.16}$$

$$y(s) = s^3 y_f - (s-1)^3 y_i + \alpha_y s^2 (s-1) + \beta_y s (s-1)^2 \tag{6.17}$$

that automatically satisfy the boundary conditions on x, y. The orientation at each point being related to x', 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).

$$\begin{bmatrix} \alpha_x \\ \alpha_y \end{bmatrix} = \begin{bmatrix} k \cos\theta_f - 3x_f \\ k \sin\theta_f - 3y_f \end{bmatrix} \tag{6.18}$$

$$\begin{bmatrix} \beta_x \\ \beta_y \end{bmatrix} = \begin{bmatrix} k \cos\theta_i + 3x_i \\ k \sin\theta_i + 3y_i \end{bmatrix} \tag{6.19}$$

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.

6.6 Planning results

The timing law $s(t)$ is supposed as a 5th order polynomial with the following conditions, $s_i = 0,\ s_f = 1, \dot{s}_i = 0,\ \dot{s}_f = 0, \ddot{s}_i = 0,\ \ddot{s}_f = 0$ and $\Delta T = 3$ sec. Figure (6.1) shows $s(t)$ and Figure (6.2) shows $\dot{s}(t)$.

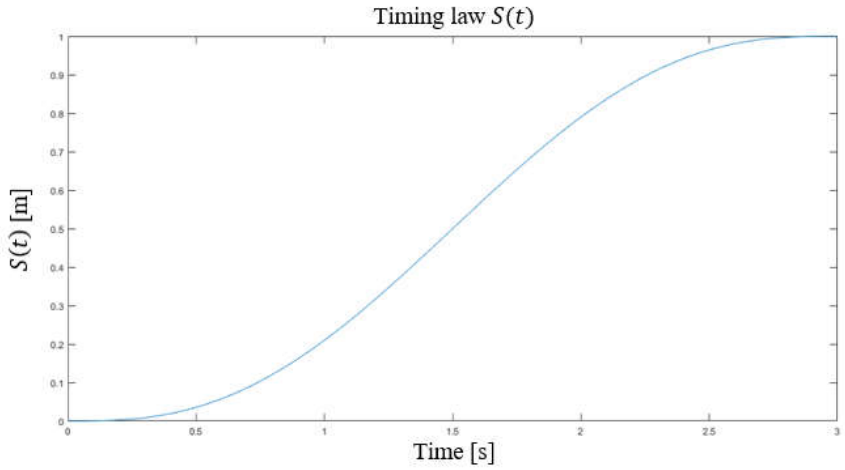


Figure 6.1: Timing law s(t)

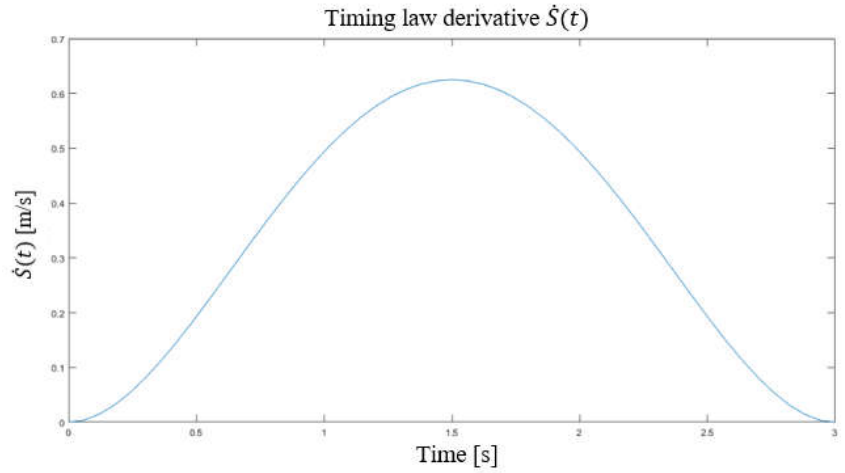
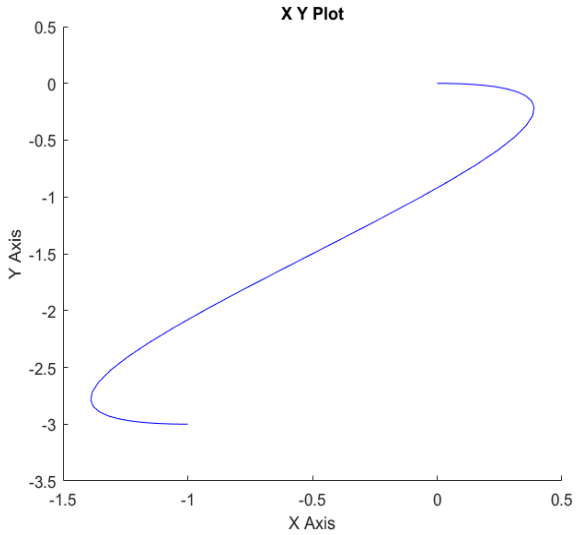
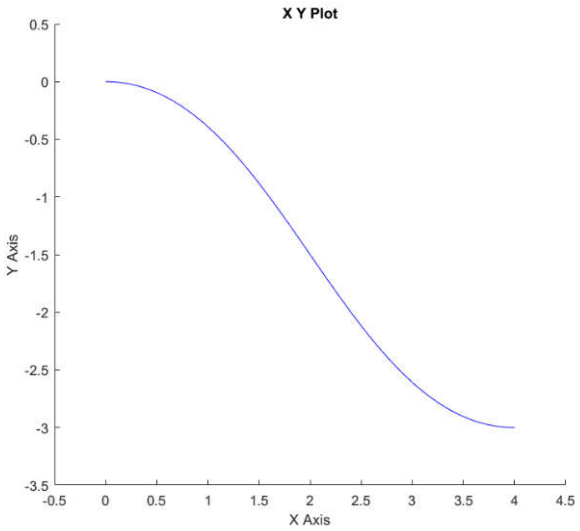


Figure 6.2: Timing law derivative $\dot{S}(t)$

The Equations (6.13) to (6.19) are used to solve the planning problem. Planning MATLAB algorithm is shown in Appendix A. The typical paths produced by the planner are shown in Figures (6.3 to 6.5). As already noticed, the unicycle never inverts its motion, that is forward because k > 0. For k < 0, the manoeuvres would have been performed in backward motion with different paths.



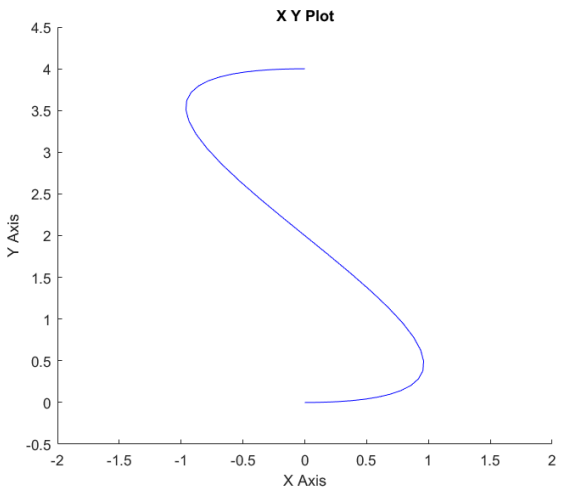
(a)



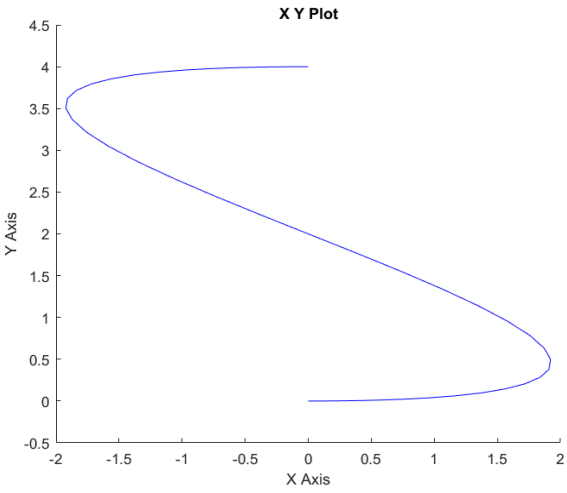
(b)

Figure 6.3: parking manoeuvre (K=5)

Planning a parallel parking manoeuvres via cubic Cartesian polynomials are shown in Figure (6.4a) with k = 10 and Figure (6.4b) with k = 20.



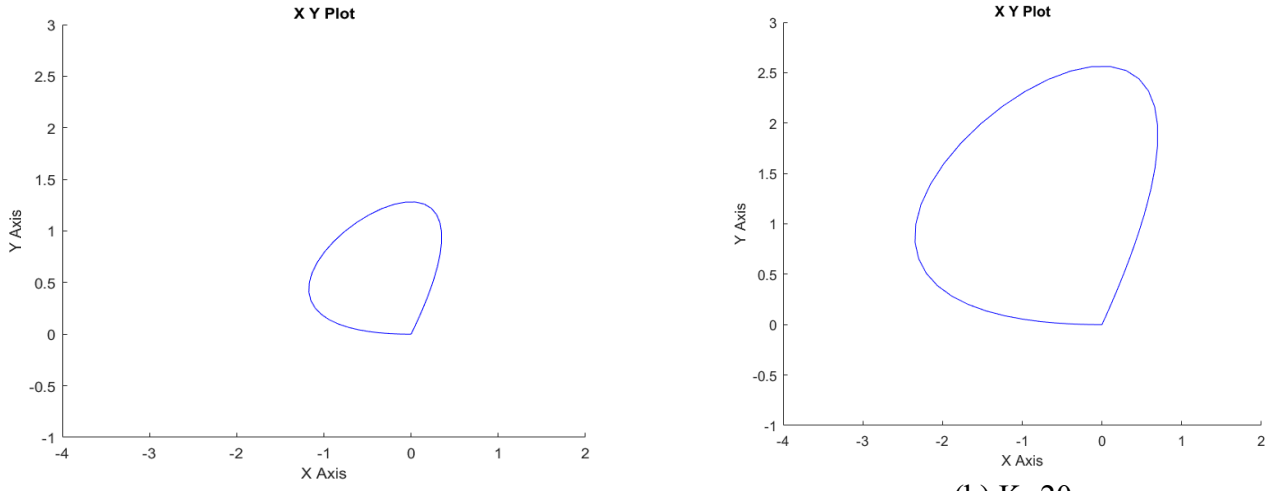
(a) K=10



(b) K=20

Figure 6.4: Parallel parking manoeuvre

Planning a pure reorientation manoeuvres via cubic Cartesian polynomials are shows in Figure (6.5a) with k=10 and Figure (6.5b) with k=20.



(a) K=10 (b) K=20

Figure 6.5: pure reorientation manoeuvre

Chapter Siven: Mobile robot control

7.1 Introduction

7.2 Inner loop control design and simulation results

7.3 Outer loop control design and simulation results

7.1 Introduction

The robot must move with the desired speed that decided by the planning and to navigate without drifting even when one wheel radius differ from the other so three levels of control (inner loop, outer loop and human in the loop) are considered. The inner and the outer loops are built and implemented using MATLAB/Simulink software and the human in the loop is implemented by a mobile application. Decentralized control architecture based on PID controller is used for the inner loop to achieve the desired wheel speed (servo level) and the outer loop is designed based on PI controller for outer loop to ensure the desired orientation (without drift). Further a human commanding the robot can be considered as the core of a third loop.

7.2 Inner loop control design and simulation results

In this section the schemes of decentralized controllers are implemented in order to control the speed of dynamic plant of the mobile robot (servo level). The block diagram of such implementation is shown in Figure (7.1).

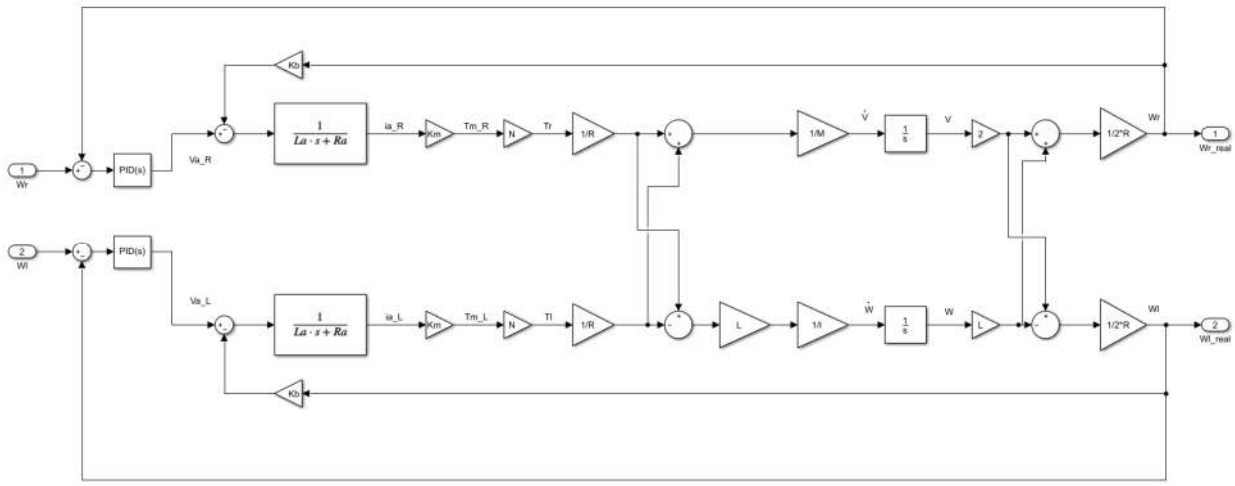


Figure 7.1: Decentralized PID controller architecture for servo level

PID controller is designed based on tuning and the following results are obtained.

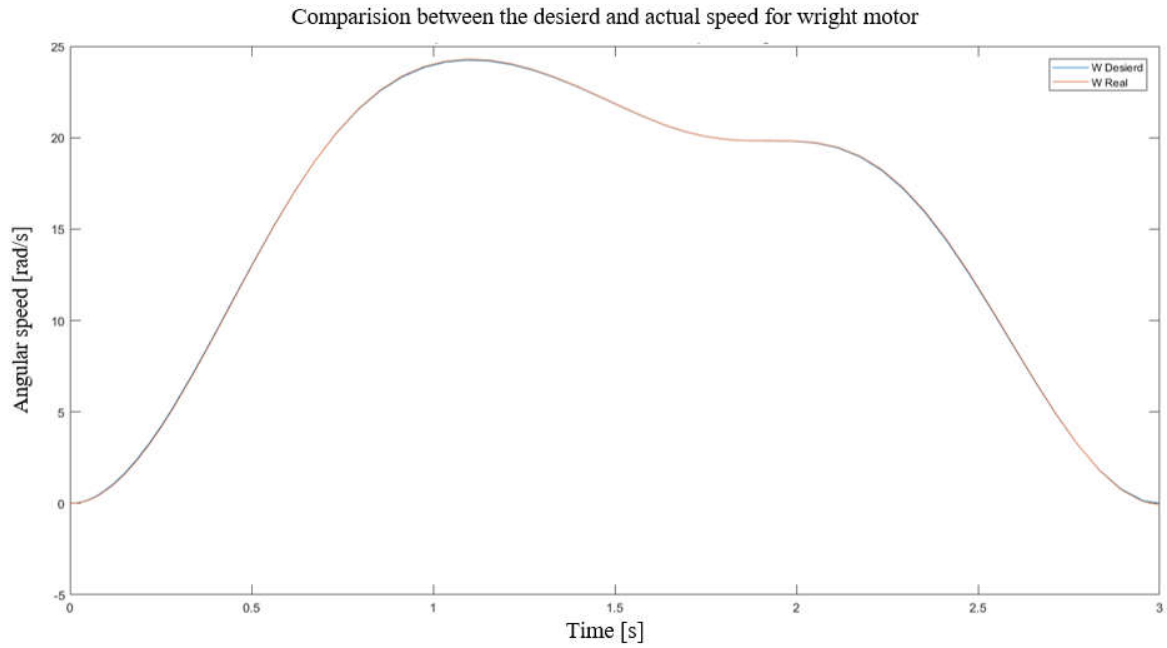


Figure 7.2: Desired and actual speed of right motor

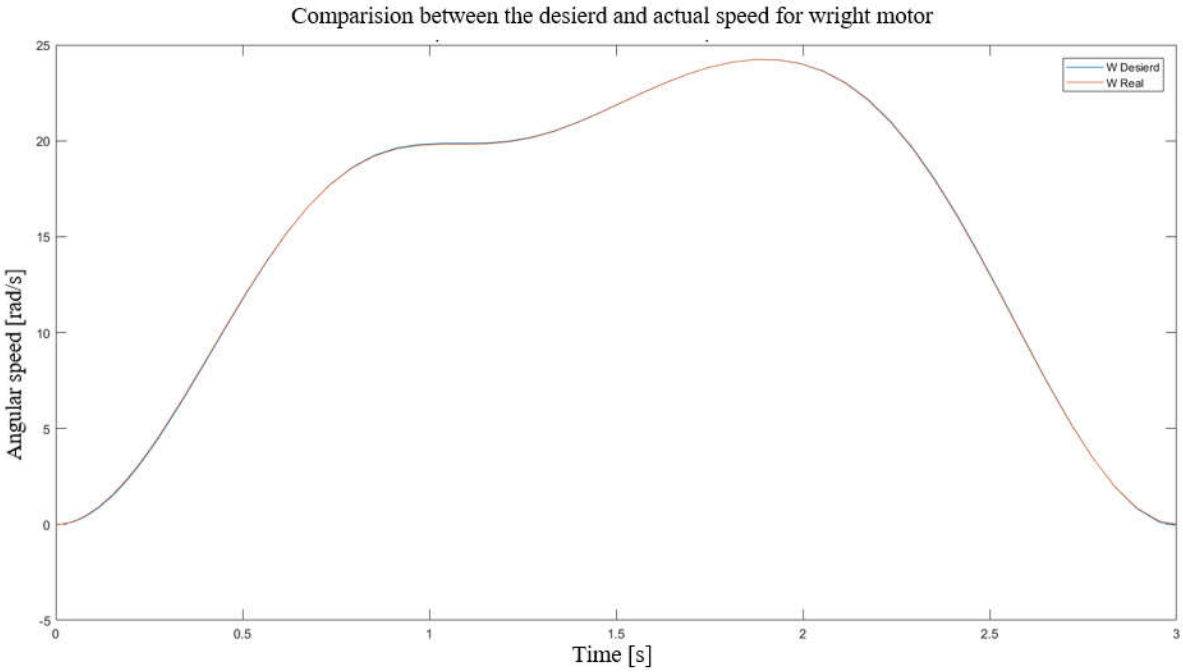


Figure 7.3: Comparison between the desired and left actual motors speeds

7.3 Outer loop control design and simulation results

In this section the scheme of outer loop controller is implemented in order to ensure desired orientation (without drift). The block diagram of such implementation is shown in Figure (7.4).

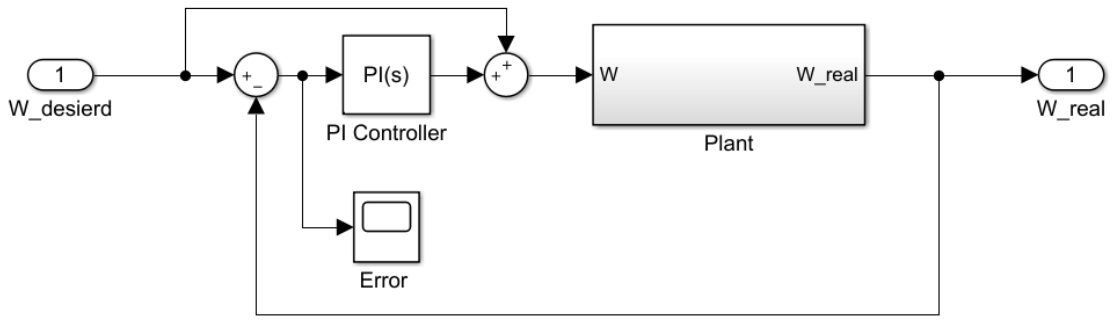


Figure 7.4: PI controller architecture for outer loop

Gyroscope sensor signal is used to measure the actual robot angular velocity ω_{real}. A PI controller with feedforward is designed to make the error (desired and actual angular velocities) become zero. The PI controller computes a compensated angular velocity. The PI controller transfer function formula is shown in Equation (7.1).

$$G_{PI} = K_p + \frac{K_i}{S} \tag{7.1}$$

where, K_p and K_i are the proportional and integral gain respectively. Equation (7.2) shows the tuned PI transfer function.

$$G_{PI} = 2.7 + \frac{1.3 * e^5}{S} \tag{7.1}$$

For simulation and to cause undesired robot angular velocity, one of the wheels is made (in the simulation) to have a different diameter. The results are shown in Figures (7.5-7.7) are obtained by reduce the size of right wheel diameter to %95 of its real size.

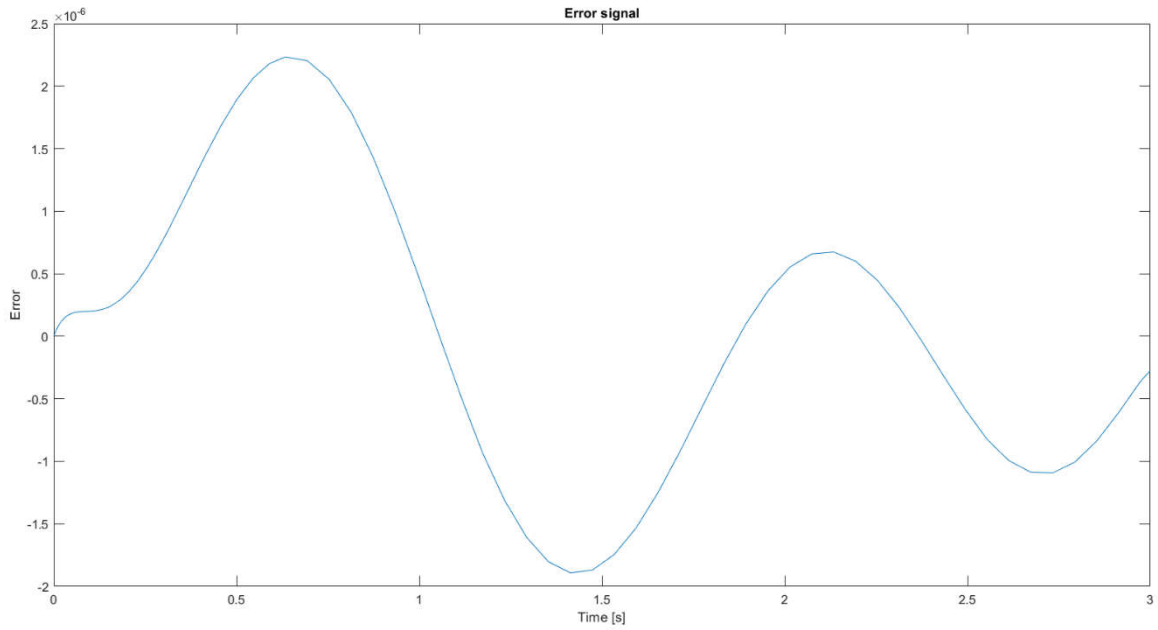


Figure 7.5: Error signal for outer loop controller

The robot path results with consideration of drifting are shown in Figures (7.6), (7.7).

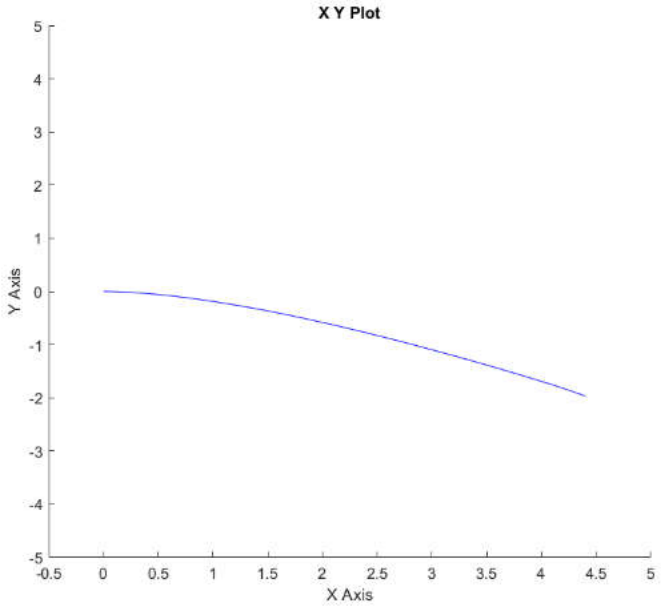


Figure 7.6: Robot path without controller

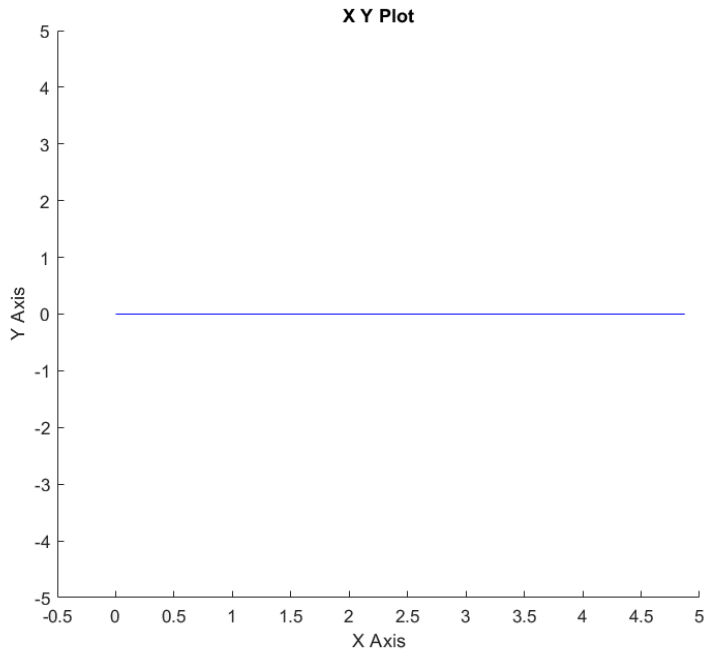


Figure 7.7: Robot path without controller

Chapter Eight: Practical work and experiments

8.1 Introduction

8.2 Manual mode implementation

8.3 Autonomous / semi-autonomous mode

8.1 Introduction

This chapter includes the experimental and practical work of the Tennis-Ball Collecting Robot which presents myRIO and LabVIEW implementation.

The implementation is done for two operating modes which are the manual mode and autonomous / semi-autonomous mode.

8.2 Manual mode implementation

This mode promotes a tennis player to command the robot to move to and collect balls through a mobile-phone application. The manual mode implementation in LabVIEW environment is shown in Figure (8.1).

Due this mode the robots act as follow.

When the manual mode is selected the Bluetooth module activate and the system waits the user connection. After the connection is established the system start receiving the user commands. A case structure is used to control the system based on the received command. An I2C technique is used for interfacing the system controller with the driving motors controller. A digital pulse width modulation technique is user for interfacing the system controller with the collecting mechanism motor driver. The system returns its default values when it shuts down.

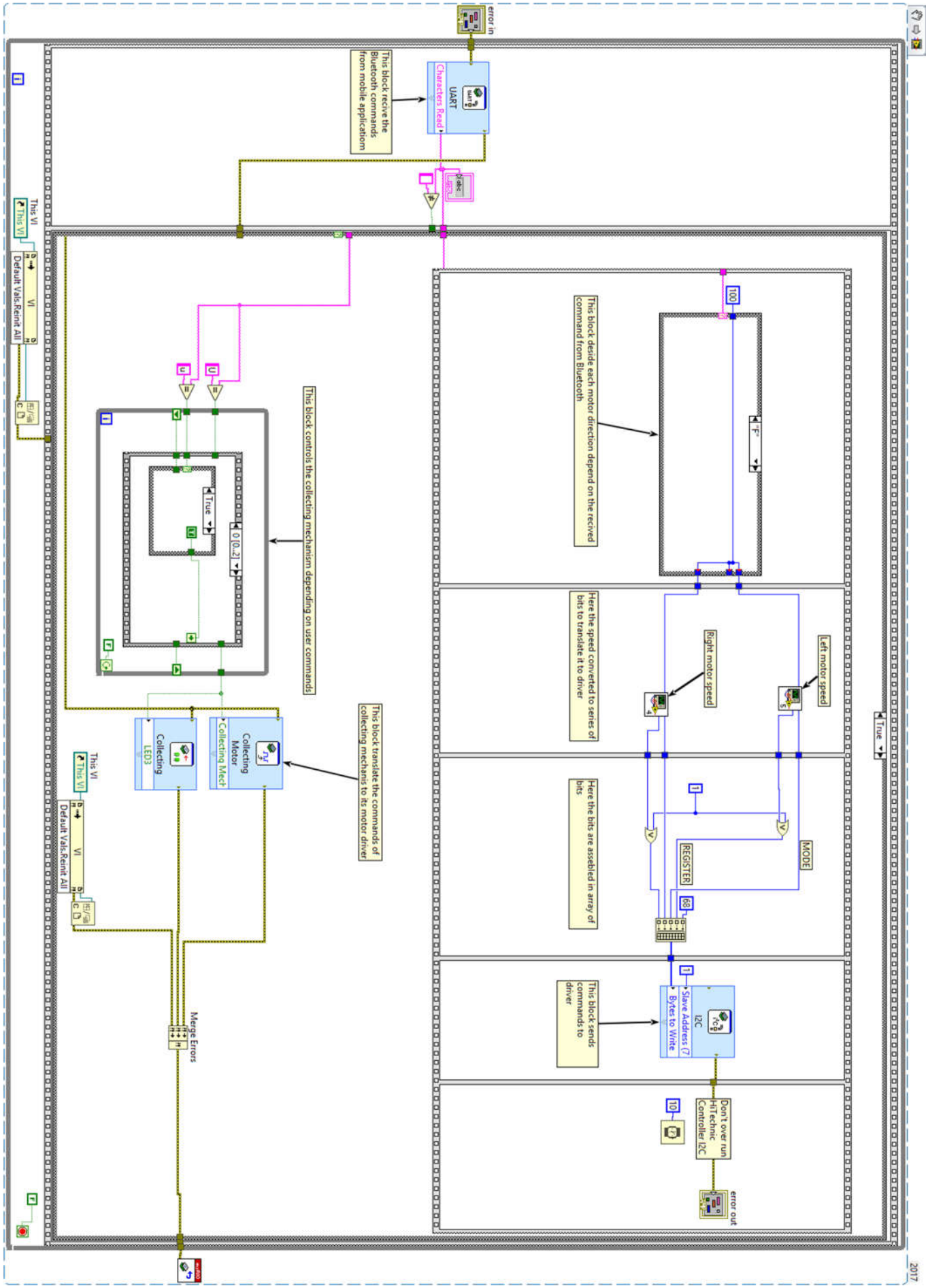


Figure 8.1: Manual mode LabVIEW code

Figure (8.2) shows the mobile-phone application.

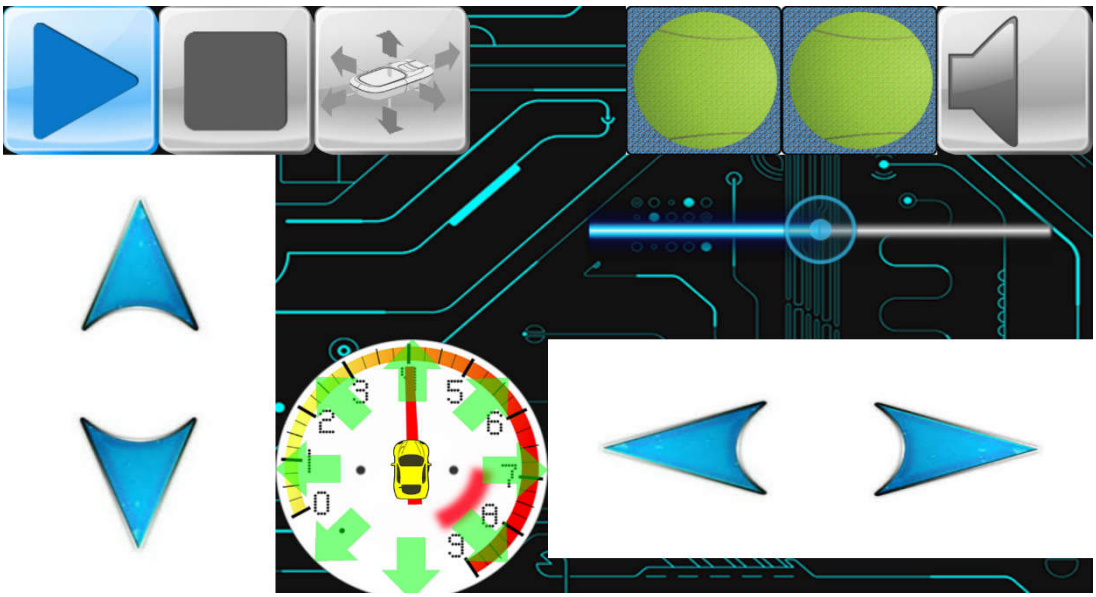


Figure 8.2: Mobile-phone application.

8.3 Autonomous / semi-autonomous mode

In this mode, the robot is moved from initial pos to final pos depending on planning algorithm mentioned on chapter 6. Figure (8.3) shows the front panel of the LabVIEW implements a graphical user interface of planning algorithm and Figure (8.4) shows the programmed algorithm.

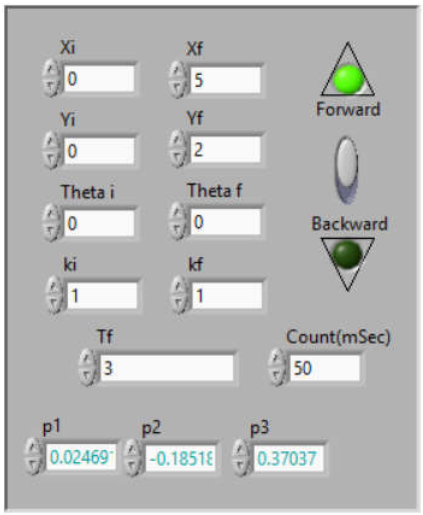


Figure 8.3: Front panel of the LabVIEW implementation of planning algorithm

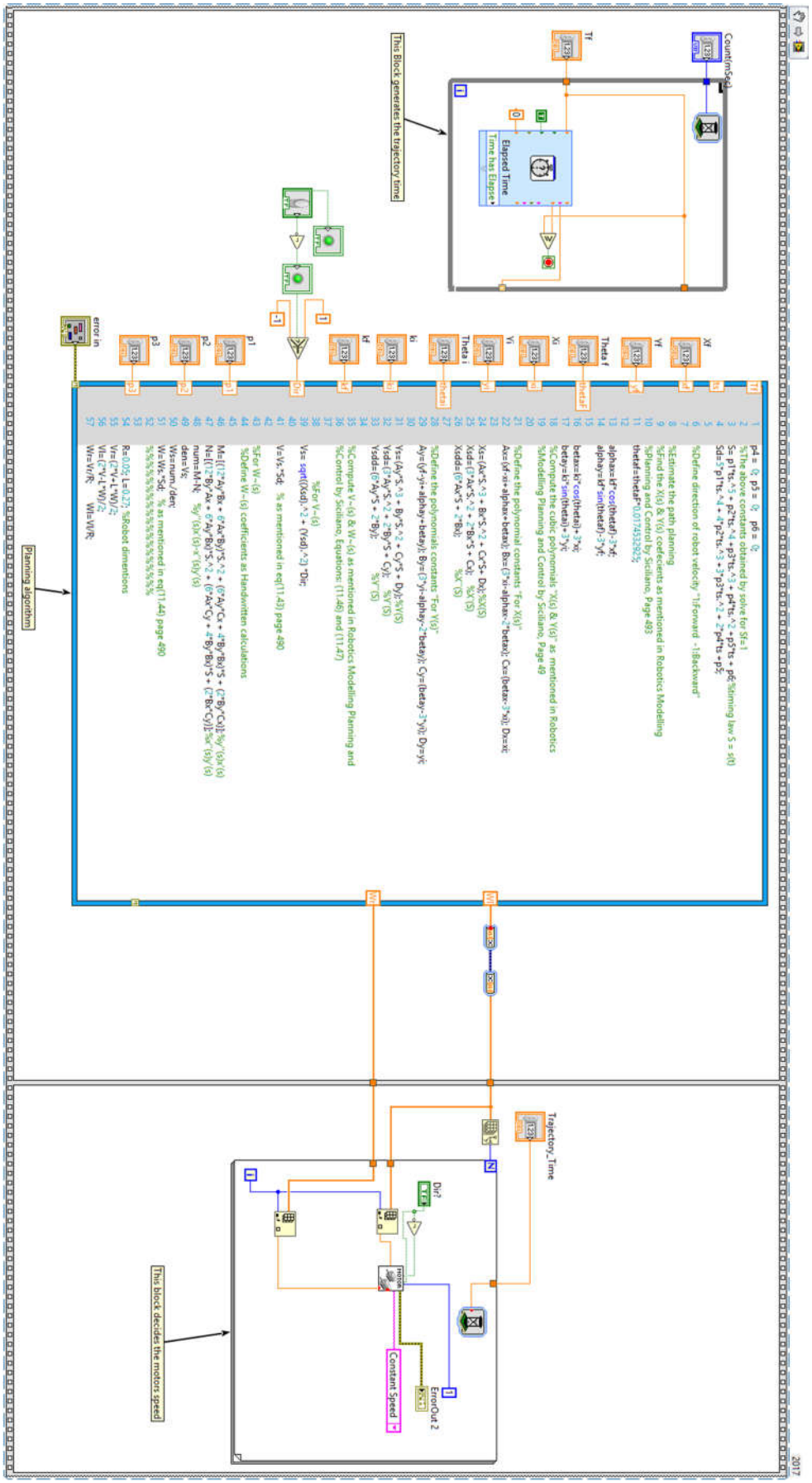


Figure 8.4: Block diagram of the LabVIEW implementation of planning algorithm

As shown in Figures (8.3, 8.4) a graphical user interface is used to enter planning parameters. A time generator algorithm is used to generate the trajectory time, the planning algorithm decides each motor speed. A servo level control (inner loop) is used to achieve the desired speed.

Above codes are executed on the robot impeded device and the robot controlled by the mobile application and moved based on user commands and collect the balls when the manual mode is activated, then the robot controlled through the autonomous / semi-autonomous mode, the robot commanded based on the planning algorithm and moved from initial pos to final pos following the generated trajectory.

Chapter Nine: Conclusion and future work

9.1 Conclusion

9.2 Future work

9.1 Conclusion

In this graduation project, we designed and built a nouvelle autonomous / semi-autonomous mobile robot to specifically collect up to fifty tennis balls. The robot navigates through two independently-actuated wheels and is stabilized by a castor. The nouvelle collection mechanism is based on a spinning brush that pushes tennis balls into a lightweight basket on top of the robot. A motor makes the brush spins at a desired speed that is experimentally verified. Three successive working prototypes have been designed, built, and tested to make sure the adopted ideas and conceptual designs are applicable and robust.

The mobile robot has its own on-board intelligence where it is equipped with a myRIO embedded device. The robot can navigate from initial pos to final pos based on a programed path planning algorithm or, alternatively, a tennis player can command the robot to move to and collect balls through a mobile-phone application.

9.2 Future work

In this section, some of future work suggestions are presented. Adding a global cameras to cover all court and a central image-processing system for identifying and localizing all tennis balls on the court. Build an optimal path planning, collision avoidance, and mobile robot navigation.

Appendix A

Prototypes

A.1 First Prototype

The purpose of this prototype is to test our alternative design of the collecting mechanism which shown in Figure (A.1). This prototype is built from available tools and toys.

Figure A.1: Collecting mechanism

The prototype moves only forward and reverse without steering, so we connect front wheel by a motor with a small gearbox to drive the system and we used bearings without limitation to make the rotation of rear wheel free.

Finished prototype shown in Figure (A.2) its structure made from toys, the collecting mechanism made from some soft material (2 blades of soft Plastic), a12V DC driving motor is used to drive the system, a 5V DC motor is used to actuate the collecting mechanism, a 12V lithium rechargeable batteries is used as a power source and the used microcontroller is Arduino UNO that shown in Figure (A.3), HC-05 Bluetooth module is used which shown in Figure (A.4), L298N H_Bridge Module is used as a motors driver which shown in Figure (A.5) and a MOSFET Lm317 module is used to control the speed of collecting mechanism. A mobile-phone application shown in Figure (A.6) is used to control the system via Bluetooth connection.

Figure A.2: Finished prototype

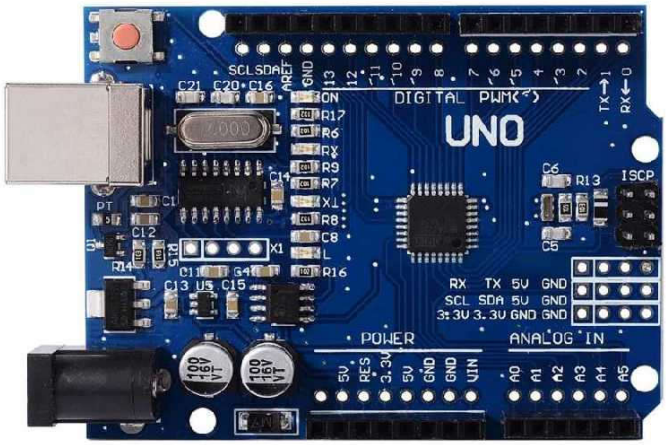


Figure A.3: Arduino Uno

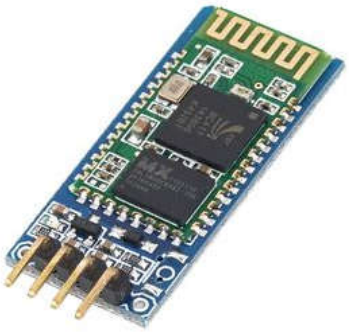


Figure A.4: Bluetooth HC-05 module

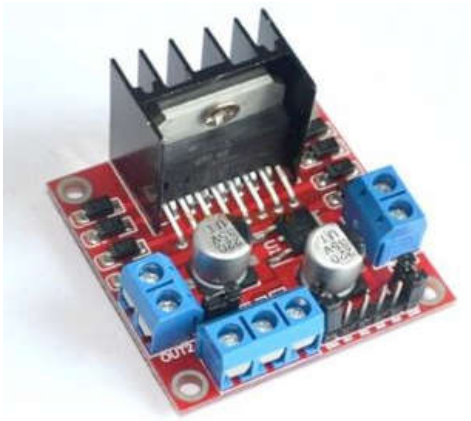


Figure A.5: H_Bridge Module (L298N)

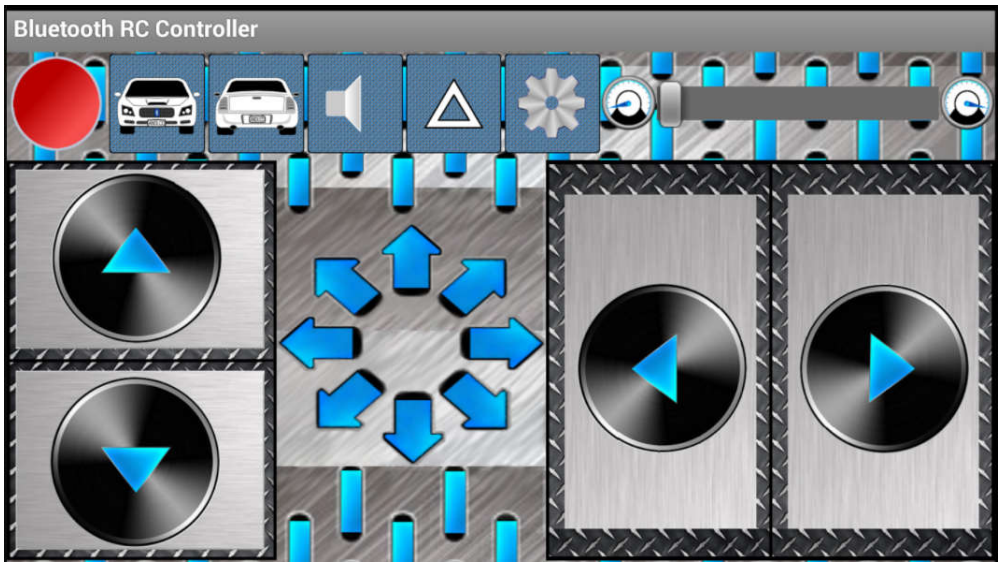


Figure A.6: Arduino Bluetooth RC Car

The result of the prototype its collect the balls in with acceptable efficiency and we conclude that our alternative design archives the purpose.

A.2 Second Prototype

The purpose of this prototype is to apply the collecting mechanism on real robot to collect the tennis balls. Figure (A.7) shows the second prototype.

Figure A.7: Second prototype

The prototype moves forward and reverse with steering, two independent wheels are actuated by two independent DC motors used to drive the robot.

The robot structure made from iron material, the collecting mechanism made from some soft material (2 blades of soft Plastic) and its rod made from strong plastic material, two12V DC

driving motors are used to drive the system, a 12V DC motor is used to actuate the collecting mechanism, a 12V lithium rechargeable batteries is used as a power source and the used microcontroller is Arduino MEGA that shown in Figure (A.8), HC-05 Bluetooth module is used which shown in Figure (A.4), L298N H_Bridge Module is used as a motors driver which shown in Figure (A.5) and a MOSFET Lm317 module is used to control the speed of collecting mechanism. A mobile-phone application shown in Figure (A.6) is used to control the system via Bluetooth connection.

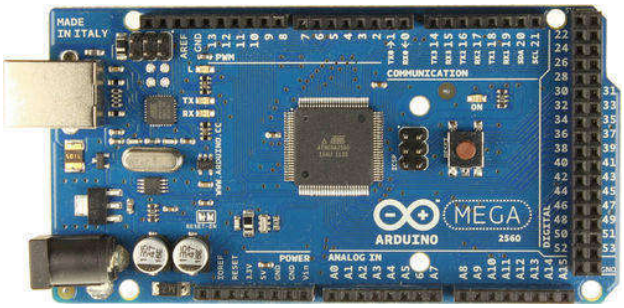


Figure A.8: Arduino MEGA

The second prototype its collect the balls in with acceptable efficiency and later on an alternative design for collecting mechanism is used which shown in Figure (A.9) collects the balls best than the old one.

Figure A.9: New collecting mechanism

Appendix B

MATLAB Codes

B.1 Planning algorithm

```
function
[Xdot,Ydot,Thdot,V,W]=fcn(ts,xi,yi,xf,yf,thetai,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)

%Compute V~(s) & W~(s) as mentioned in Robotics Modelling
Planning and
%Control by Siciliano, Equations: (11.46) and (11.47)

               %For V~(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;    % as mentioned in eq(11.44) page 490


thetas=atan2(Ysd,Xsd);  % as mentioned in eq (11.45) page 491

%Robot dimentions:
R=0.0715;      %wheel radius in meters
L=0.35;    %the distance between wheels centers

%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));

%%%%%%%%%%%%%
%%%%%END%%%%%
%%%%%%%%%%%%%
```

B.2 Robot parameters

```
%Matlab script written to identify the robot parameters
%including its physical dimentions and the driving
%Dc motors parameters
%Written by:
%Mahmoud Sharabati
%Majdi Sultan
%Sohaib Abu Omar
%v1.0 31.03.2018
Km= 0.44;     %Torque Constant [N.m/Amp]
La= 0;        %Armature Inductance[H]
Ra= 3;        %Armature Resistance[ohm]
Rr=0.0715     %Right Wheel Raduis [m]
Rl=0.0715;    %Left  Wheel Raduis [m]
M=  11.5;     %Robot Total Mass   [Kg]
I=  0.0104;   %Robot Interia      [Kg.m2]
L=  0.35;     %Distance between the wheels [m]
N=40;         %Ratio
Kb= 0.76;     %Back EMF Constant 0.0487 V/(rad/sec)
```

References

[1] I. Elamvazuthi et al., “Development of an Autonomous Tennis Ball Retriever Robot As an Educational Tool,” Procedia Comput. Sci., vol. 76, no. Iris, pp. 21–26, 2015.

[2] H. K. Chen and J. M. Dai, “An intelligent tennis ball collecting vehicle using smart phone touch-based interface,” Proc. - 2016 IEEE Int. Symp. Comput. Consum. Control. IS3C 2016, pp. 362–365, 2016.

[3] Haitham Eletrabi, Robotic Tennis Ball Collector “Tennibot”, pages. Available at http://www.tennibot.com. (last visit date November 2017)

[4] Inwatec, Inwatec Tennis Ball Collector - www.inwatec.dk, Youtube, Published on 4/09/2012, 0:24, https://www.youtube.com/watch?v=d1Fco37SC2E

[5] Mailman, C. J. (2012). U.S. Patent No. 8,313,396. Washington, DC: U.S. Patent and Trademark Office.

[6] The Trash Pack Street Sweeper TV Spot. (n.d.). Retrieved December 12, 2017, from https://www.ispot.tv/ad/777E/the-trash-pack-street-sweeper

[7] Chawla, N., Alwuqayan, W., Faizan, A., & Tosunoglu, S. (2012). Robotic Tennis Ball Collector, (February 2018), 1–6.

[8] Chen, H. K., & Dai, J. M. (2016). An intelligent tennis ball collecting vehicle using smart phone touch-based interface. Proceedings - 2016 IEEE International Symposium on Computer, Consumer and Control, IS3C 2016, 362–365. http://doi.org/10.1109/IS3C.2016.100.

[9] ExMark Lawn Mowers. (n.d.). Retrieved December 10, 2017, from http://more.crunchybeachmama.com/2013/03/exmark-lawn-mowers.

[10] Valdes-Rodriguez, R. (2000). U.S. Patent No. 6,079,930. Washington, DC: U.S. Patent and Trademark Office.

[11] Yu, Y., & Kim, S. (2014, July 31). TENNIS BALL BOY. Retrieved December 11, 2017, from https://ifworlddesignguide.com/entry/133917-tennis-ball-boy

[12] 10-S Tenn-Tube - 20 Ball. (n.d.). Retrieved December 15, 2017, from http://www.10-s.com/Products/Ball-Tubes/IF2003

[13] AHM 38 G | Hand mowers | Lawnmowers | Garden. (n.d.). Retrieved December 13, 2017, from https://www.bosch-garden.com/gb/en/garden-tools/garden-tools/ahm-38-g 3165140578929-199958.jsp#tab_technical.

[14] M.dhgate.com. (n.d.). 18" 20" Men's Large Vintage Genuine Leather laptop Travel Wheeled Duffle hand luggage bag Carry On Rolling Duffel bags men. Retrieved December 30, 2017, from https://m.dhgate.com/product/18-quot-20-quot-men-039-s-large-vintage-genuine/387536439.html.

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

[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

[17] 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.

[18] Neĭmark JI, Fufaev NA (1972) Dynamics of Nonholonomic Systems: Translations of Mathematical Monographs. American mathematical Society.

[19] Bloch A, Crouch P, Baillieul J, Marsden J (2003) Nonholonomic Mechanics and Control.

[20] Joshi, B., Shrestha, R., & Chaudhary, R. (2014). Modeling , Simulation and Implementation of Brushed DC Motor Speed Control Using Optical Incremental Encoder Feedback DC Motor Model Simulink Modeling , Simulation and Parameter estimation. Proceeding of IOE Graduate Conference, (November), 497–505.

[21] Richard C Dorf and Robert H Bishop. Modern control systems. Pearson, 2011.

[22] Anvari, I. (2013). Non-holonomic Differential Drive Mobile Robot Control & Design: Critical Dynamics and Coupling Constraints (Doctoral dissertation, Arizona State University).

[23] 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

[24] F. (n.d.). CHEER 4 FTC. Retrieved March 1, 2018, from http://cheer4ftc.blogspot.com/p/motors.html

[25] Tennis. (n.d.). In Wikipedia. Retrieved October 8, 2017, from http://en.wikipedia.org/wiki/Psychology.

[26] A. (n.d.). Retrieved November 17, 2017, from https://www.amazon.co.uk/Razor-electric-scooter-E90-Spacer/dp/B00DKDBKEE/ref=sr_1_1?s=sports&ie=UTF8&qid=1527614352&sr=1-1&keywords=140mm scooter wheel.

[27] A. (n.d.). Retrieved November 18, 2017, from https://www.amazon.co.uk/JD-Bug-200mm-Wheel

Bearings/dp/B01EI520CS/ref=sr_1_3?s=sports&ie=UTF8&qid=1527614020&sr=1-3&keywords=200mm scooter wheel

[28] Pitsco Education: TETRIX®. http://www.tetrixrobotics.com/ (2014). Accessed 10 Apr 2017.

[29] "RS-550PC-8019", standard specifications for RS-550 motor, [online] Available: product.mabuchi-motor.com/detail.html? id=115.

[30] Gross, B. (2013). MaxSonar Operation on a Multi-Copter. Retrieved from MaxBotix: http://www. maxbotix. com/articles/067. htm.

[31] Sharp GP2Y0A21YK0F Distance Measuring Sensor Unit Measuring distance: 10 to 80 cm Analog output type, http://www.pololu.com/file/0J85/gp2y0a21yk0f.pdf.

[32] http://www.botched.co.uk/pic-tutorials/mpu6050-setup-data-aquisition

[33] Instrument, N. (2013). NI myRIO− 1900 User guide and specifications. National instruments Corporation.