MANSOURA UNIVERSITY
FACULTY OF ENGINEERING
MTE DEPARTMENT

A Robot to improve Library management

A Formal Design Report

Submitted to the faculty of Engineering in

Partial Fulfillment of the Requirements for the Degree of

Bachelor of Engineering in Mechatronics.

Done By:

Islam Adel Ahmed Elnagar	800161969
Ahmed Mounier Ahmed Sabaa	803091337
Mohamed Mahmoud Radwan	800138038
Ahmed Ehab Mahfouz Mahfouz	800160085
Adham Hanaa Mourad Ebrahim	800162026
Sara Hassan Bdeir Rabea	800160068
Sahar Ali Mohamed Dawood	800159877
Mohamed Zahran Elbaz Zahran	800168621
Mohamed Elsayed Elsayed Elrazaz	800136647

Ebrahim Noureldin Mohamed Badrawy

Hassan Hossam Hassan Zwain

Under the supervision of:-

Prof., Dr. Mohamed Rezk & Dr. mahmoud Hamouda

Eng. Ahmed Elhelw

June 2024

Acknowledgment

We want to describe our appreciation to the Faculty of Engineering, and the mechatronics department for their infinite efforts and support through our educational journey. We are grateful to Prof., Dr. Mohamed Rezk & Dr. mahmoud Hamouda & Eng. Ahmed Elhelw for their constant support and we couldn't have ended this project without their directions, experienced supervision, and encouragement and, all other department staff guided us through our university educational journey, never accepting less than our best efforts. We are grateful to them all.

Also, we would like to thank everyone who helped us in our project, including the workers and technicians, for their engagement with us, which helped us achieve this project and our family members who were the core of passion and support in each step in our life.

Abstract

This research project's main goal is to invent a robot that can perform a specific task in the library, to understand this task clearly, we need to go through the problem first, analyze it, measure all risks and failure possibilities to find the suitable solution for the problem we are facing in our project. this project is providing really great benefits for all academic persons and readers and serve big sector of persons and universities as it can help save efforts for them and the librarian in addition to help them have good navigation and searching ability to reach the required book as fast as possible.

The robot has the ability to scan the books existed on shelves and store names and location for each book on a database so that it is no longer hard for any person to find a book in the library, you no longer have to go search for any book and go through many sections and shelves lastly, now the books can be reached by not only librarian but also every person on the library, it is easy, fast and convenient.

Now, a robot dimensions of 430 MM x 310 MM x 620 MM and manage its operation by Raspberry Pi and Arduino with the assistance of ROS and computer vision can provide a real solution for saving time and effort in searching for a book, it is not only a solution for finding a book but also the books database will be stored on cloud which the librarian have access to and he can manage the process of borrowing the books from the library as it will be easy if he mentioned the location for each book and as well add each person who borrowed a book from the library.

Keywords: Library – Robot– Arduino –Database – Computer vision

Table of Contents

List of Figures

List of Tables

List of graphs

Chapter 1

Introduction

There is a strong believe we have about engineering and engineers, as we have gone through many education stages and we have worked in many projects and innovations through our education process we have a strong believe that our role is to be problem solver for the problems we are facing in our life and one of us, the engineers that working on this project had faced a great problem while navigating for a book in the library and he begin to think about it and how to provide reliable, affordable and easy to work and develop solution for the problem of spending too much time to search for a book and find a section related to this book so we begin to think on how to create a robot that can do this process.

The solution to any problem can be reached through some common steps, the first one is to understand the problem, to think deeply about the problem core and begin to analyze it and find the related factors that connected to it, work in how to solve each related factor individually step by step you can solve each problem and link them together to get a prober solution for the problem.

The problem we are facing is the time consumed for searching for a specific book on the library so we were searching for a book about finance while we were in the library and started searching for this book by through multiple steps, first we have to find the finance section then we will have to search for the book itself, it is known that there is two methodologies in arrangement for the books in the library which we are going to discuss in the next section clearly but generally, our robot can go through both of them and manage the database for the books and as well store in it the borrowed books.

Libraries around the world in public places or in educational organizations like schools and universities are using two methods for classifications as shown in figure1.1, the first one is Dewey Decimal Classification System which is generally called the Dewey decimal system which depends on specifying relative location for the book using relative index and the topics is classified using specific period of numbers for example philosophy books takes numbers from 100 to 199

and there will be a description for each section so it is easy to add extra books on the same section and adjusting the numbers for each section and topic, this classification is used in more than 200,000 libraries in around 135 countries, the second classification system is library of congress classifications which depends on dividing all the knowledge into classes this classes is divided into sub classes, there is around twenty one basic classes each class is represented by a letter and each subclass is represented by second digit letter as well for example class N refers to Art and has subclasses NA that represents Architecture and so on, this method of classification is mostly used in united states of America and was invented on eighteenth century by James Hanson.

Figure 1. 1 classification of books on the library

Our robot is considered a great solution for the arrangement of the books in the library, it does not only scan the books in the library and specify each book location, but also it can help the librarian to check whether there are any mistakes in the of the books that lead to losing any books on the library, instead of storing the information for students and researches who is going to borrow any of the books on the library. definitely, this robot will make it so easy for librarian to manage all the books on the library and as well make it easy for students and researchers to find the desired book on the fastest way possible.

Chapter 2

Background of the study

It is really oblivious to feel the difference between working on the library now and before and as well when it comes to the students, researchers and university professors, it will be much easier to find a book they are willing to have, whenever you are in a problem and try to fix it, it is really important to put yourself on the position of the one who is facing this problem, show him empathy, not only that but also try and think in how to find a solution for the problem.

Previously, we as a team working together had faced many problems on our life, most of them was in technical issue related to our field of study as our specialization is considered one of the most significant specializations between all engineering department, Mechatronics engineering can handle many branches of engineering, which gives us great advantage on applying all knowledge we have in multiple fields of study on solving many problems, but before thinking of a solution we have to think and consider all scientific bases we are aware of, we have to consider concepts like safety, customer experience, environment enhancement, cost efficiency, reliability, durability and easy to maintain.

2.1 Statement of the problem

- Librarian takes long time to document the person who borrowed a book.

- Students and researchers find sometimes find it hard to find a book on a section.

- Workers in the public libraries will suffer in rearrange the books after any person borrowed a book or even take it to read a part of it.

- Librarian will find it hard to check if there are any missing books on each section.

- Having a database that include all the details about each section on the library can't be done manually by employ a person who have to go through each shelf to check it.

2.2 Objectives of the project

The main goal of this project is to design and manufacture an automated robot that can go through the library, scan boxes and store each book location in a data base, the robot will be able to navigate through the library by creating a map using ROS mapping and then navigate through the library to scan the shelves of boxes which is classified by one of the classification methods we have discussed previously, the robot can work on and store all the data base for the boxes on a cloud that the librarian will have access to from the main computer in the library.

2.3 Significance of the Study

The robot will help save time and effort for librarian to check any lost books or find a book.

- **It will help increasing the productivity and knowledge for researchers and students.**
- **It will help to save time and effort for researchers and university professors that consumed in finding a book.**
- **It will help reduce effort for public library workers to navigate for lost boxes and rearrange it.**
- **It will increase the national income for our country as it will increase the knowledge for readers which most of them provides productive solutions for problems.**
- **It will increase our university rank among other universities as we participate in providing smart solution for public problems like this one.**

2.4   Scope and Delimitation

The design project has the following scope and delimitations:

- **Applicable for the use on public, universities and school libraries.**
- **Applicable to be used in high all environments, whether crowded or quite one.**
- **Applicable to scan unlimited number of boxes as it will upload the database on big storage station.**
- **Not applicable to work in libraries with multiple floors, it supposed to be a robot for each floor.**
- **Not applicable to be used in libraries with long height shelves as there is a limit for vertical linear actuator that move camera up and down through shelves.**

2.5   Definition of terms

Arduino Mega microcontroller – It is a microcontroller board dependent on the Microchip ATmega2560 microcontroller. The board is outfitted with sets of computerized and simple information/yield sticks that might be interfaced to different development sheets (shields) and other circuits. The board has 54 digital input/output pins 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. [1]

The Raspberry Pi: it is a small, affordable single-board computer designed to promote computer science education and DIY projects. It features a range of ports and connectors, enabling it to interface with various peripherals and sensors. Despite its compact size, it offers significant processing power and versatility. Widely used in robotics, home automation, and other innovative applications, the Raspberry Pi supports various operating systems and programming languages.[2]

Linear actuator – It is a vertical linear actuator that robot uses to mover camera used for scanning books through shelves, there are an motor connected to it in order to produce the required linear motion and it is widely used in many applications like computer numerical control machines. [3]

DC gear motor – It is considered an actuator that converts electrical energy to mechanical energy and it is a particular kind of electrical engine that is intended to create high force while keeping up

a low strength, or low speed, engine yield. Stuff engines can be found in a wide range of utilizations and are likely utilized in numerous gadgets in your home. [4]

DC Motor driver- it's an electronics component that link between the low voltage signal of microcontroller and the high-power circuit of motor and it also can alternate the direction of motor whether to go forward or backward and there are several types of it, we choose the suitable one based on many considerations like the driven motor, the required speed, operation time and power consumption of motor. [5]

Chapter 3

Review of related literature and studies

3.1   Conceptual literature

Our project is consisting of many components but one of the main parts are raspberry pi and Arduino microcontroller through which we will control the operation of the robot. The raspberry pi will be responsible for navigation, mapping and the giving signals to Arduino to drive the motors and process the sensors data, The Arduino will be the best choice to get your electronic devices get connected with. It has many numerical input/output pins to be easy for dealing and connecting any sensor or actuator with it, there are six analog which is used as inputs and power jack. It has most needed elements to make the microcontroller16 MHz ceramic resonator work well. You have just to connect it to any PC or even a mobile phone with a USB cable or power it with an AC-to-DC adapter or battery to be able to work with it. You can play with the Arduino without worrying too much about doing something wrong, but for just in situation it happened then you can change the chip for few dollars or so and can be replaced freely. "Mega" is the Italian word use for 1 and was selected to mark the release of Arduino Software IDE. The Mega board and version 1.0 of Arduino IDE were the main version of Arduino, now it had developed to newer releases. The Mega board is the first in a sequences of USB Arduino boards, and the orientation model for the Arduino platform; for a wide-ranging list of current, past, or out-of-date panels see the Arduino index of boards that will be helpful.

It will be connected to the motor driver which is a BTS7960 Single Channel 43A H-bridge Controller, it will drive two motors one for driving the whole robot which will be attached to two rear wheels and there will be two free wheels.

3.2 Research literature

When it comes to robots and automated machines, there are many companies and developers around the world worked on various types of robots as there are a driving potential on each person characteristics to provide reliable solution for complex tasks and also this robots can work more efficiently than human in many cases, as for navigation and scanning specifically, there are various studies about it, some of them is seeking for accuracy, reliability, fun or productivity so we had read a lot in this area and will discuss some of this local and foreign studies in the upcoming points.

3.3 Foreign studies

3.3.1 Isolation of Dots for Arabic OCR using Voronoi Diagrams (Bahrain)

When it comes to reading the name of a book using computer vision, we are going to use Optical Character Recognition which depends on recognition of the objects on the image by going through multiple processes starts by pre-processing, segmentation, feature extraction and classification. All of this steps the system will go through whenever it is going to work on recognizing any character on any language, it had to make the picture clearer and remove un necessary details and focus only in important details and when it comes to the Arabic language, one of the major factors that leads to mis recognizing the Arabic characters, is the absence of dots in some Arabic letters as shown in figure 3.1. and in this study, the developers used Voronoi diagram in order to assist them into recognizing the Arabic letters that commonly leads to conflict at the presence of dots.[5]
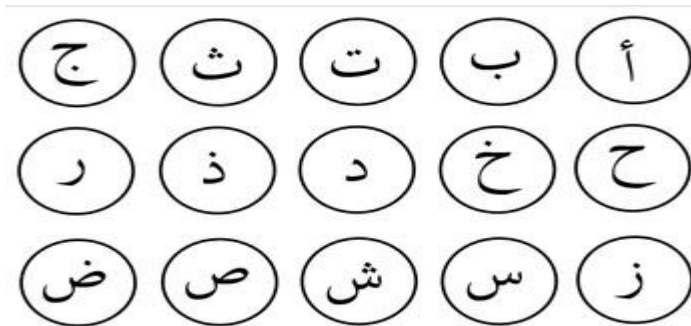
Figure 3. 1 Arabic characters with dots

3.3.2 ORAN SYSTEM: A BASIS FOR AN ARABIC OCR (Kingdom of Saudi Arabia)

Arabic language is considered a big challenge for all the scientists and developers around the world who works on OCR as the words on Arabic language is always written connected together, so if we compared it to different language, we will find out that it is one of the hardest language that OCR can work on and provide high accurate result, there are some other languages that is similar to Arabic like Chinese, Roman characteristics and Japanese. There are available OCR's for them like omni page and recognition, but the developers still working on them in order to enhance the accuracy for it, In this study the developers worked hardly in rising the accuracy of Arabic OCR by depending on representing the image in binary method and then apply special analogy depends on partite sets and graphs, this analogy will be trained multiple times until it reached the top accuracy level and this analogy block diagram is represented in below figure3.2.

intend to do.[6]
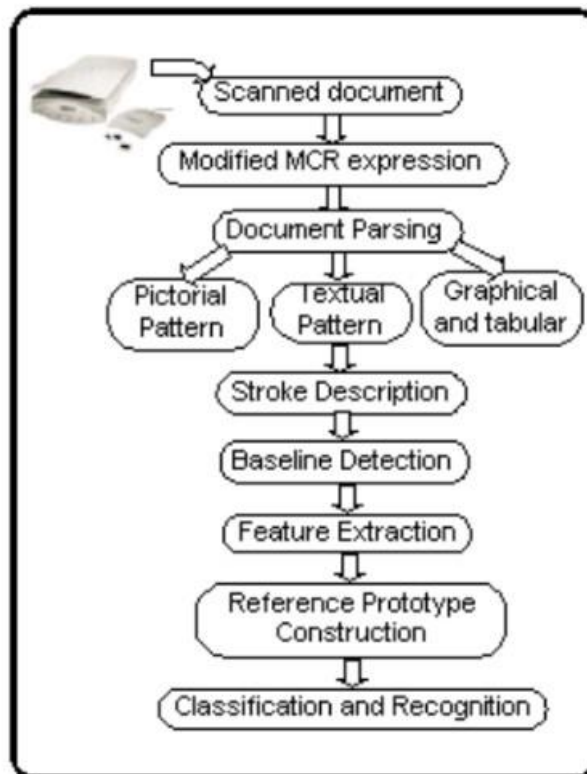
Figure 3. 2 block diagram for Oran system

3.3.3 Sonar-based real-world mapping and navigation (USA)

Mapping and navigation becomes one of the most popular research topics as it has a great range of applications like self-driving cars, robots that works in close environment, and open as well, in this study the developers worked on providing a map for unknown and unstructured environment by depending on sonar sensor which sends signals and then receive it, based on the analysis for this signals, it can create a map for the surrounded environment, the sonar mapping concept is used in many applications to generate a full map used for path planning and navigation, and it is commonly used in autonomous mobile robot navigation system which is called Dolphin and this map is used to develop application that represents this environment by uploading this map into a cloud so it is easy for user to manage, control and monitor the robot that work in this environment as shown in figure3.3.[7]
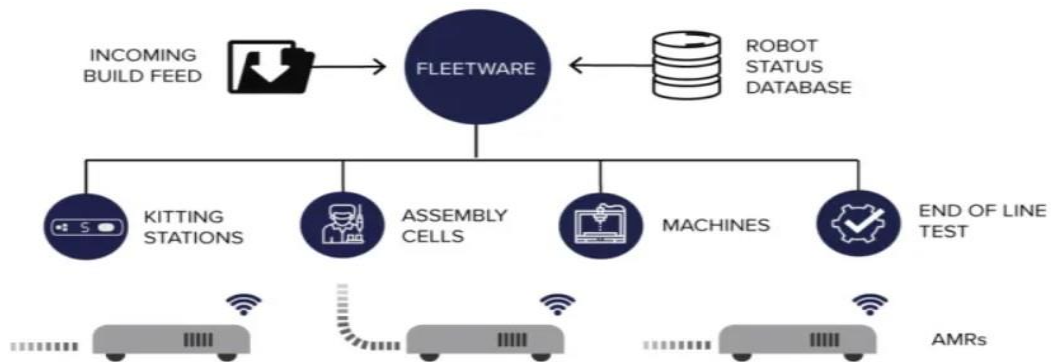
Figure 3. 3 Map storing in a cloud database for future use.

3.3.4 Path planning of line follower robot (Turkey)

Our main study concept of motion is line following as our robot will follow a line parallel to the library shelves, it is simple to use infrared sensor to detect the line and then follow it, what makes it much more accurate and fast is the way of processing the signal comes from infrared sensor, in this study, they had used ARM cortex-3 Based microcontroller to process the signal comes from the sensor and as well, perform some mathematical approaches on the signal in order to give the most accurate result for the motion of the motors, they used partial integral derivative control in

order to enhance the motion of the robot and increase the robot accuracy, not only they will control the direction of the motors, but also they controlled the speed of the motors in order to end the track fast as the application requires, so this robot supposed to be weightless and small in design as shown in figure3.4. [8]
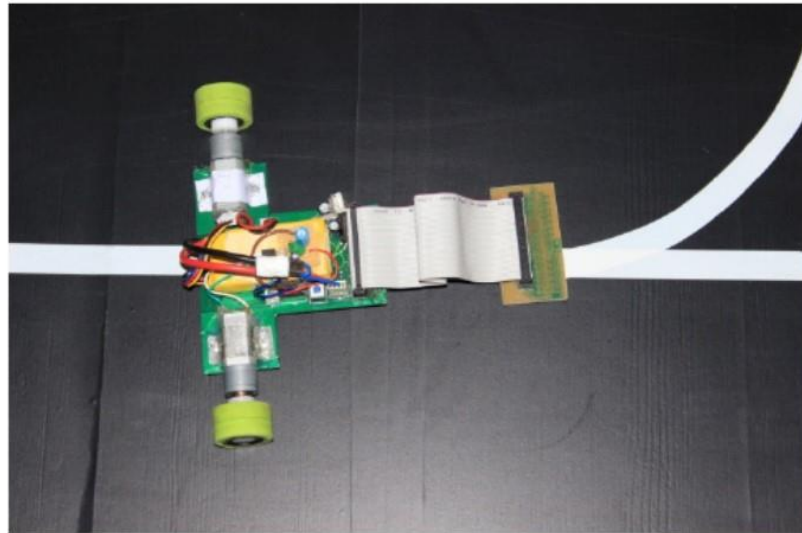
Figure 3. 4 Design of the line follower robot.

3.3.5 Comparing OCR and RFID application in the Container Shipping Industry Alternatives Criteria (United Kingdom)

In order to provide the best solution for a problem we have to consider all possible ways for implementing any project, in our study we are going to use camera as the main source of information for the books, it will detect the absence of books on shelves and read each book name, there are another method of detecting the absence and presence of books, which is using radio frequency identification layers, which depends on sending and receiving signals by which it can detect the existence of any book, but in our project it will be better, if we used OCR instead of RFID as it is more efficient, reliable, cost reductive product and process its operation fast although it is less in accuracy than RFID but the difference in accuracy between them will not affect the operation of our project totally and its operation is shown in figure 3.5, it is much more simple than OCR and have a narrow area of development.[9]
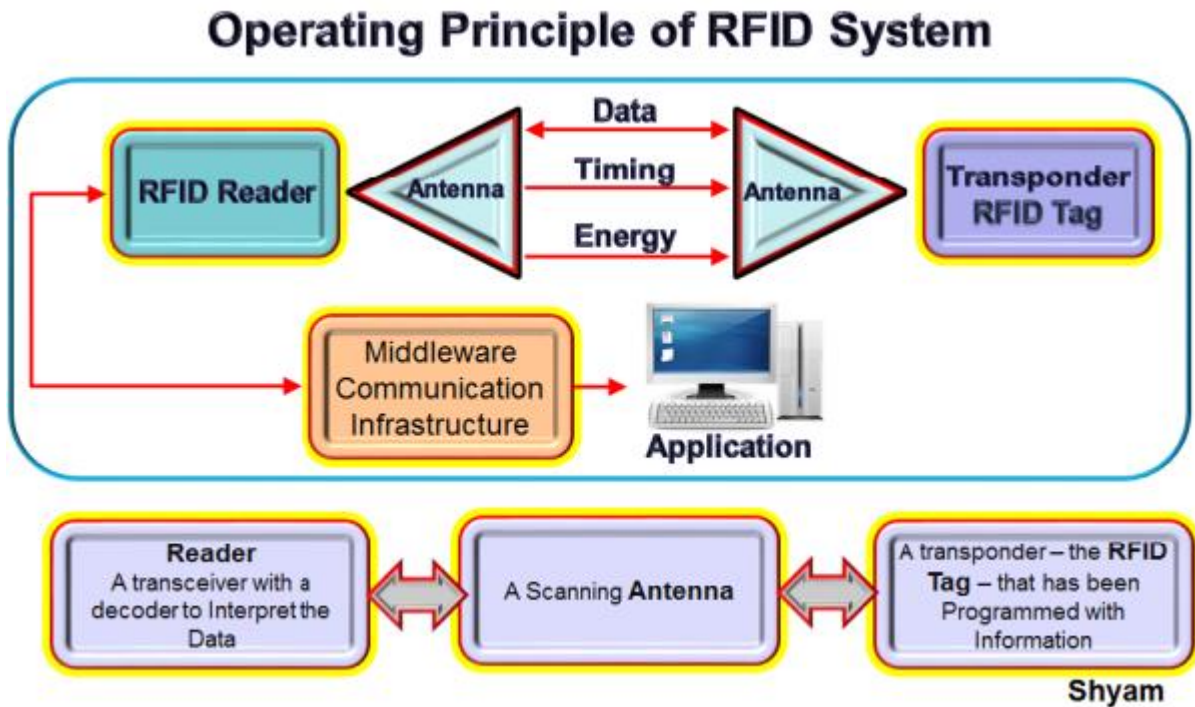
Figure 3. 5 Work flow for RFID

3.4   syntheses

Eventually, there are big progress on many fields related to our projects, starting by deploying OCR in the recognition of Arabic letters and how hard to use modules for detection of Arabic letters used previously on English, but there are multiple researches applied on such field in order to increase the accuracy for detecting the letters, but when it comes to navigation and going through the library, there is many studies that covers how the robot can go through the library, by not only using the line following methodology, but also using cameras, sensors and radar to read the dimensions of the area and develop and detailed map for the environment the robot is going to operate on and this multiple studies make the door open for continuous development on our project.

Chapter 4

Design specifications

In design specifications detailed explanation for this design project will be discussed under the light of its processes, functions, means and different evaluations will be performed to justify technically the design approach for this designed project like weight decision matrix.

4.1   Research paradigm

In this section we will discuss the research paradigm which includes the types of approaches taken during development of the project, data collection and its translation, methodology, strategy, logical justification etc.

- **Input-Process-Output (IPO) model:**

The input-output chart is representing what is the sources of information to the system central processing unit which can be raspberry pi is the Kinect camera, OCR camera, and then the Arduino will transfer data from rotary encoder, this information will be processed and then give orders to the motor drivers to run the motors, the Kinect camera signal will go to the raspberry pi to be used for mapping and navigation and the OCR camera signal will use to scan the books, it will read the names for the books on the shelves and then store it on a database as shown in figure 4.1.

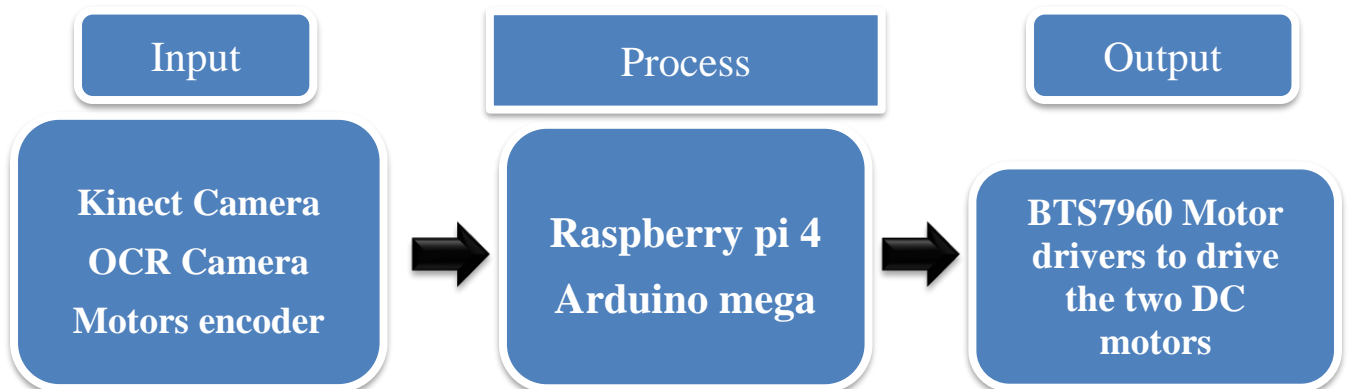| Input	Process	Output
Kinect Camera OCR Camera Motors encoder	**Raspberry pi 4 Arduino mega**	**BTS7960 Motor drivers to drive the two DC motors**

Figure 4. 1 Input-process-output

4.2 Functions

Library assistant robot operation can be categorized into three sections, the first one is main driving mechanism for the robot, the robot mainly consisted of four wheels, two rear wheels driven by two DC geared motors, which mainly chosen to produce low speed, high torque, in order to be able to drive the whole robot frame slowly and give a chance for the OCR camera to capture a picture for the books on shelves and Kinect for mapping and navigation, this motors is driven using two 12volts battery, the two front wheels is free wheels that attached to the robot chassis and they reduce the friction and make the movement smooth, the second section is the camera which will be attached to the top plate of the robot, it will move up and down, this camera will be connected to computer that will have Python program which will be used for detection of the books then read each book name and store the names and location on the database and the last section is the raspberry pi which is responsible for driving the whole robot as it will navigate the library first and create a map for the navigation in the environment to make it possible to move in the library and avoid obstacles.

4.3 Project development

Mechatronics engineering is a combination of mechanical engineering, electrical engineering and programming. So, we have many concepts we will go through in our project beginning with:

4.3.1 Raspberry pi

The Raspberry Pi 4 is a versatile and powerful single-board computer that significantly enhances performance over its predecessors. Equipped with a Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC running at 1.5GHz, it offers up to 8GB of LPDDR4-3200 SDRAM. Its connectivity options include dual-band 802.11ac wireless, Bluetooth 5.0, Gigabit Ethernet, two USB 3.0 ports, and two USB 2.0 ports. The Raspberry Pi 4 supports dual 4K displays via two micro-HDMI ports, making it suitable for a wide range of applications from basic computing tasks to advanced multimedia projects and it is shown in figure 4.2.

In the realm of robotics and the Robot Operating System (ROS), the Raspberry Pi 4 stands out due to its enhanced processing power and connectivity. ROS, an open-source framework for robot software development, benefits from the Pi 4's improved CPU and memory capabilities, facilitating more complex computations and real-time data processing. The Raspberry Pi 4's GPIO pins allow for easy integration with various sensors and actuators, essential for robotic applications. Additionally, its USB 3.0 ports support high-speed peripherals, crucial for advanced robotics tasks like real-time video processing and sensor fusion.

Using Raspberry Pi 4 with ROS in robotics projects allows developers to build sophisticated, low-cost robots capable of performing a variety of tasks. The Pi 4's compatibility with a wide range of ROS packages enables seamless integration of hardware and software components. This makes it an ideal platform for educational robotics, hobbyist projects, and even professional research and development. With the ability to connect to high-speed networks and process large datasets efficiently, the Raspberry Pi 4 helps streamline the development process, reducing both time and cost for robotics projects.

Figure 4. 2 Raspberry pi 4

4.3.2 Arduino mega

The Arduino Mega is a microcontroller board based on the ATmega2560, designed for projects requiring more input/output pins than typical Arduino boards. It boasts 54 digital I/O pins (15 of which support PWM output), 16 analog inputs, and 4 hardware serial ports (UARTs). The board operates at 5V and can be powered via a USB connection or an external power supply (6-20V), with an automatic selection between power sources and it is shown in figure 4.3.

The Mega 2560 is compatible with most shields designed for the Arduino mega, ensuring versatility in various applications. It includes a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. With 256 KB of flash memory, 8 KB of SRAM, and 4 KB of EEPROM, the Arduino Mega 2560 is well-suited for complex projects that require extensive memory and multiple peripherals.
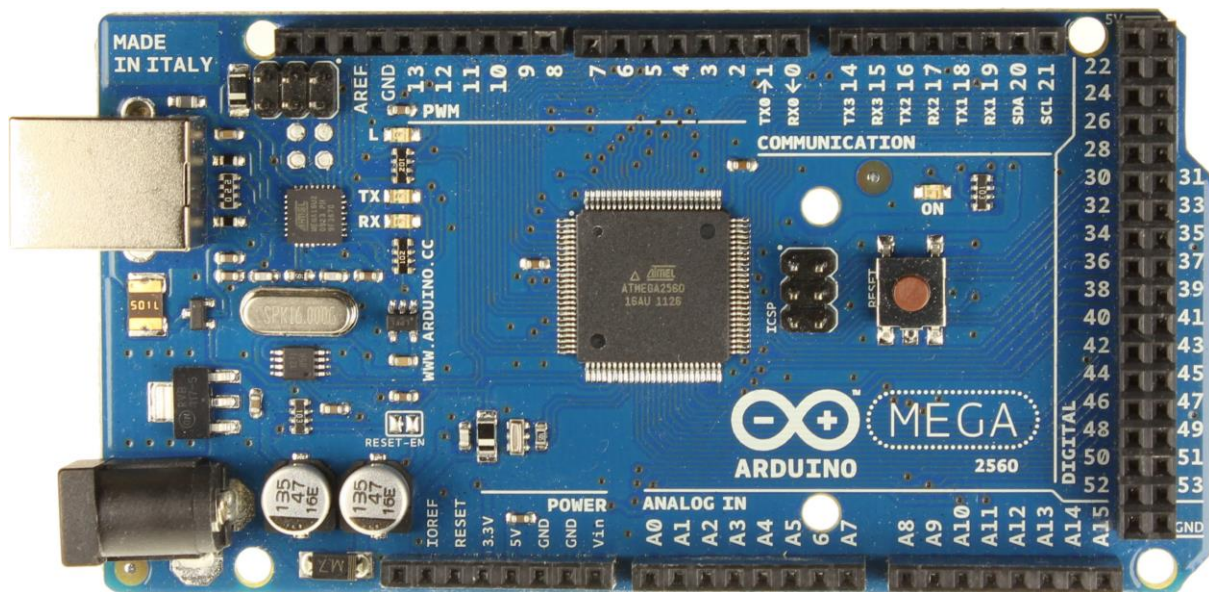
Figure 4. 3 Arduino Mega

4.3.3 Sensors

It is a device that converts physical property to electrical signal and then this electrical signal is delivered to the Arduino, during this transmission process there is some electrical process done to get a suitable signal for Arduino board which make this digitalized signal something understandable to us in the code and mainly it is converted to zero and one form and there are many types of sensors that can work with Arduino shown in figure 4.4.
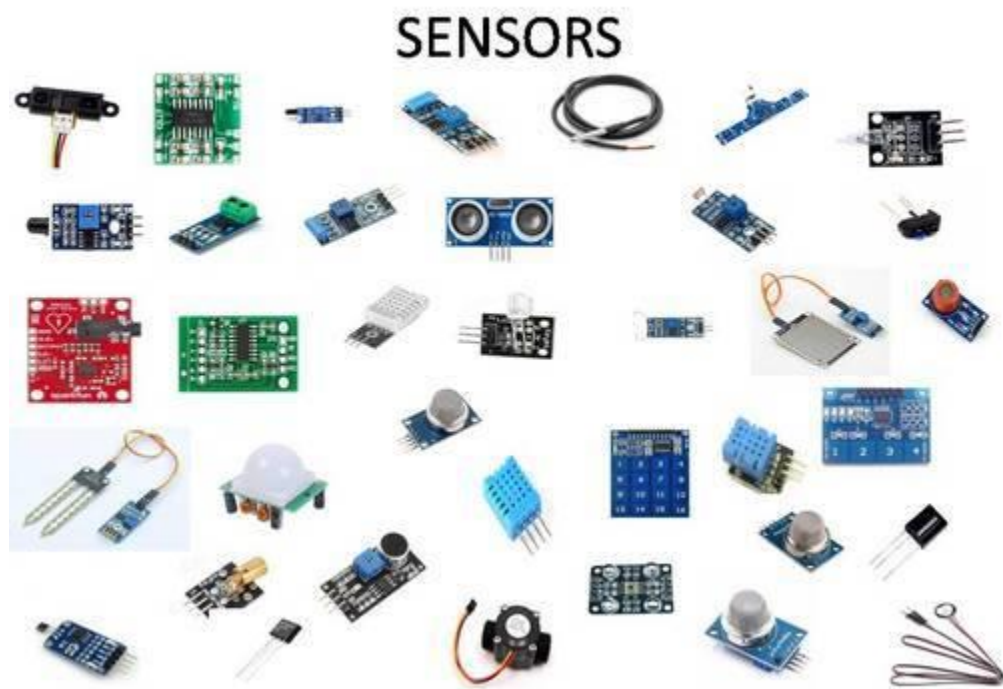
Figure 4. 4 Different types of sensors compatible with Arduino

In our project we used rotary encoder and the camera shown in figure 4.5, which is considered a complex sensor as it sends a wide variety of signals range in order to form an image, this data goes through various process in order to detect the shape of the book, recognize the name and finally store this name on a database that the system has access for and we chose Logitech C270 USP 720p camera to be used in our project in addition to the second camera which is Kinect that's used for mapping and navigation.

Figure 4. 5 Logitech USP camera used in project

4.3.4   Kinect camera

The Kinect camera, originally developed for the Xbox gaming console, has found significant applications in robotics, particularly in mapping and navigation using the Robot Operating System (ROS). It utilizes RGB-D (Red, Green, Blue-Depth) sensing technology to capture detailed 3D information about its surroundings. In ROS, the Kinect camera can be integrated with SLAM (Simultaneous Localization and Mapping) algorithms to create accurate maps of environments. This is achieved by leveraging the camera's depth sensing to gather spatial data, which is then processed by ROS packages like rtabmap or rgbdslam to build real-time 3D maps and assist in autonomous navigation and it is shown in figure 4.6.

Using a Kinect camera for mapping and navigation in ROS offers several advantages. It provides a robust, low-cost solution for capturing high-resolution depth data, essential for precise mapping. The ROS framework supports a variety of tools and packages that facilitate the integration and utilization of Kinect data. For instance, RTAB-Map (Real-Time Appearance-Based Mapping) allows robots to perform real-time SLAM, using the Kinect to both map the environment and

localize within it. Additionally, the Kinect's integration in ROS enables testing and validation of navigation algorithms in simulation before deploying them in real-world scenarios, reducing the risk of hardware damage during development

Figure 4. 6 Kinect camera

4.3.5   DC Motor

A direct current (DC) motor is a sort of electric machine that changes over electrical energy into mechanical energy. DC engines take electrical force through direct flow, and convert this energy into mechanical revolution.

DC engines utilize attractive fields that happen from the electrical flows produced, which control the development of a rotor fixed inside the yield shaft. The yield force and speed rely on both the electrical info and the plan of the engine.

The JGY-370 12V Worm Gear Motor with Encoder and Bracket is designed for applications requiring precise control and high torque in a compact form. This motor features a worm gear mechanism that provides a significant reduction ratio, resulting in high torque output while

maintaining a low speed of 32 RPM. The worm gear design also ensures that the motor is self-locking, preventing back-driving and allowing for stable operation under load without needing additional braking mechanisms. This makes it ideal for use in robotics, automation, and other applications where precision and stability are crucial and it is shown in figure 4.7.

Figure 4. 7 JGY-370 12V Worm Gear Motor

Additionally, this motor is equipped with an encoder, which provides feedback on the motor's position and speed. This feature enables closed-loop control systems to achieve higher accuracy in motion control tasks. The included motor bracket facilitates easy mounting and integration into various projects. With its robust construction and versatile functionality, the JGY-370 motor is suitable for hobbyists and professionals alike who need reliable and controllable motion solutions.

- **Operating Voltage: 12V DC**
- **Speed: 32 RPM**
- **Torque: 40 kg-cm**
- **Gearbox Type: Worm Gear**
- **Encoder: Yes**
- **Mounting Bracket: Included**
- **Dimensions: Compact size suitable for various applications**
- **Self-Locking: Yes, prevents back-driving under load**

4.3.6 DC motor speed-torque calculations

In a DC motor, the relationship between speed and torque is inversely proportional, meaning as the load on the motor increases (resulting in higher torque), the speed decreases, and vice versa. This relationship can be described using the motor's speed-torque characteristic curve, which is typically linear for DC motors. The no-load speed (the speed at which the motor runs with no load) is at the intercept where torque is zero, while the stall torque (the maximum torque the motor can produce when the shaft is not rotating) is at the intercept where speed is zero. The slope of this line is determined by the motor's voltage and internal resistance, with higher voltages resulting in higher no-load speeds and higher stall torques.

We have already developed an Excel sheet for the calculations represented below.

range of speed		wheel diameter	
0.25-.035 m/s			0.1
	0.25		
calculate the RPM of the motors			
RPM = ((60 * Speed /(3.14 * Diameter of Wheel)			
	47.74648293		
Calculation of motor torque			
number of wheels (including castor)			
	4		
number of motors			
	2		
co. friction assumed			
	0.6		
wheel radius in M			
	0.05		
Mass of the robot m around (KG)			
	5		
Number of direct-drive wheels			
	2		
total weight of the robot+ payload			
W=mg in kg*N/Kg			
	49		
the weight acting in 4 wheels			
2*N1 +2*N2			

Assume that the robot is stationary, The maximum torque is required when the robot starts moving. It should also overcome friction.

We can write the frictional force as robot torque = 0 until the robot moves. If we get the robot torque in this condition, we get the maximum torque.

$\mu * N * r - T = 0$, where μ is the coefficient of friction, N is the average weight acting on each wheel, r is the radius of wheels, and T is the torque.

N = W/number of direct drive wheels; for computing the maximum torque).		
24.5		
$\mu * N * r - T = 0$		
Torque N-M		
0.735		
Torque kg-cm		
7.49491416538757		

we will assume that the required motor torque is 25 KG-CM as we have neglected some little other factors like: the friction of the free wheels and the acceleration force of the motors

4.3.7 Computer vision

Computer vision is considered a part of artificial intelligence and mainly it is an application for artificial intelligence, as Artificial intelligence enables computer to think, computer vision enables a computer to see, analyse and take action, the difference between the human sight and computer vision is the ability for being able to take action and analyse a picture with no need for previous training in addition to detect the distances and depth in a picture, so how it works?

Computer vision mainly depends on previous training and data analysis in order to be able to recognize any image, at first we have to give the program various amount of pictures at which there will be a training for the program to extract the features in the image, and this concept is called deep learning with the usage on conventional neural network which on its construction, is similar to the human neural network in the brain that make multiple connections with multiple elements and features in order to be able to recognize the input for the image, not only deep learning is used in computer vision put also machine and there is a difference between them

although they may drive us to the same result, deep learning depends on extraction of the features in the image by going through feature extraction and classification using the conventional neural network but in machine learning the developer assist on feature extraction and the classification is done by conventional neural network and this difference is shown in the figure 4.8.
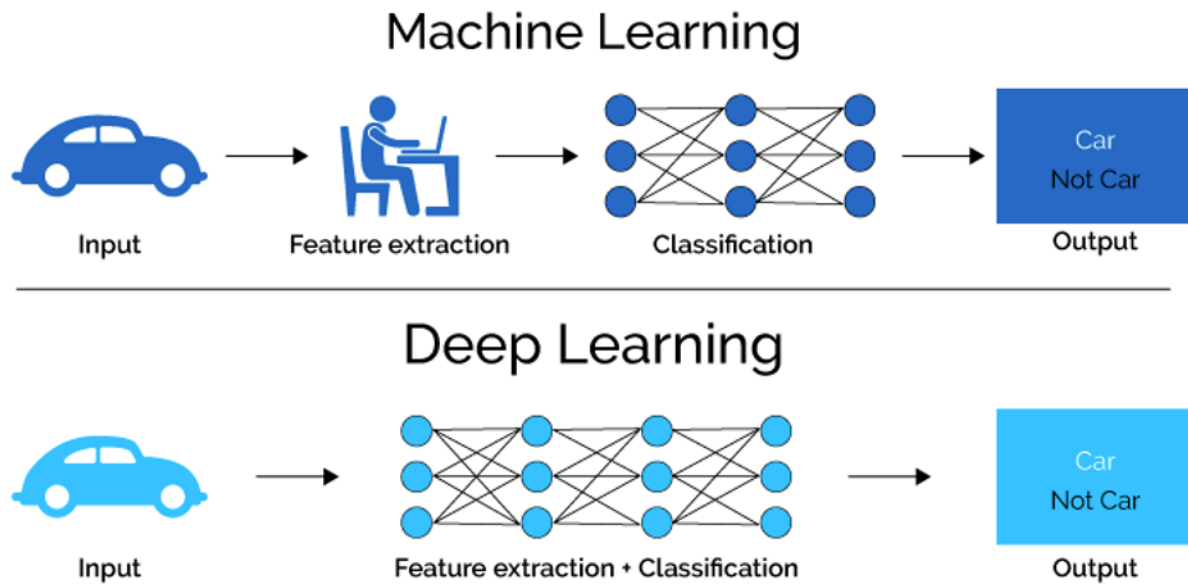
Figure 4. 8 Difference between machine learning and deep learning

4.3.8   QR code generating for books

Creating QR codes for labeling books in a library represents a modern and efficient approach to managing a library management system. By assigning a unique QR code to each book, librarians can streamline the process of cataloging, tracking, and managing the inventory. These QR codes can store essential information such as the book's title, author, genre, and shelf location. When attached to the books, these codes allow for quick and accurate data retrieval. This system reduces manual entry errors, enhances the speed of the check-in and check-out process, and ensures that library records are always up-to-date. Additionally, integrating QR codes can facilitate easier inventory audits and more effective space management by providing real-time insights into book availability and placement.

Scanning these QR codes brings several practical benefits to both librarians and patrons. For librarians, scanning a book's QR code with a handheld scanner or a smartphone can instantly update the library's database, reflecting the book's status—whether it's being borrowed, returned, or relocated. This real-time updating capability significantly improves the accuracy of the library's records and inventory management. For patrons, scanning the QR code can provide immediate access to the book's information, including its availability, reviews, and related recommendations. It can also enable self-service checkouts, reducing wait times and enhancing the overall user experience. By integrating QR code technology into the library management system, libraries can offer a more efficient, accurate, and user-friendly service as represented in figure 4.9.

Figure 4. 9 QR code example

4.3.9   mechanical design

In order to reach the best mechanical design motion mechanisms, you have to go through several steps necessary for the design process as shown in figure 4.10.

The first step as shown is to recognize the need as the understanding of the problem is the half way for the solving the problem, then begin to analyze the problem and the related factors to the problem then work on the concept of the problem and try to get in touch with the sectors who are affected by this problem, then begin to go through detailed design and apply analysis on this design like stress-strain analysis, after that begin to make small less reliable prototype and test your system and then go through making your project from much reliable material and watch out the life cycle of your desired machine and develop it based on changes happened to it through the long term use.

In our project we had gone through all steps until reaching the prototype and testing stage and used Solidworks for making the CAD model of our project then we will go on manufacturing the machine based on the considerations we had gained from working on the design processes steps.

Recognize the Need

Problem Definition

Gathering of Information

Concept Generation

Concept Selection

Communication

Detailed Design and Analysis

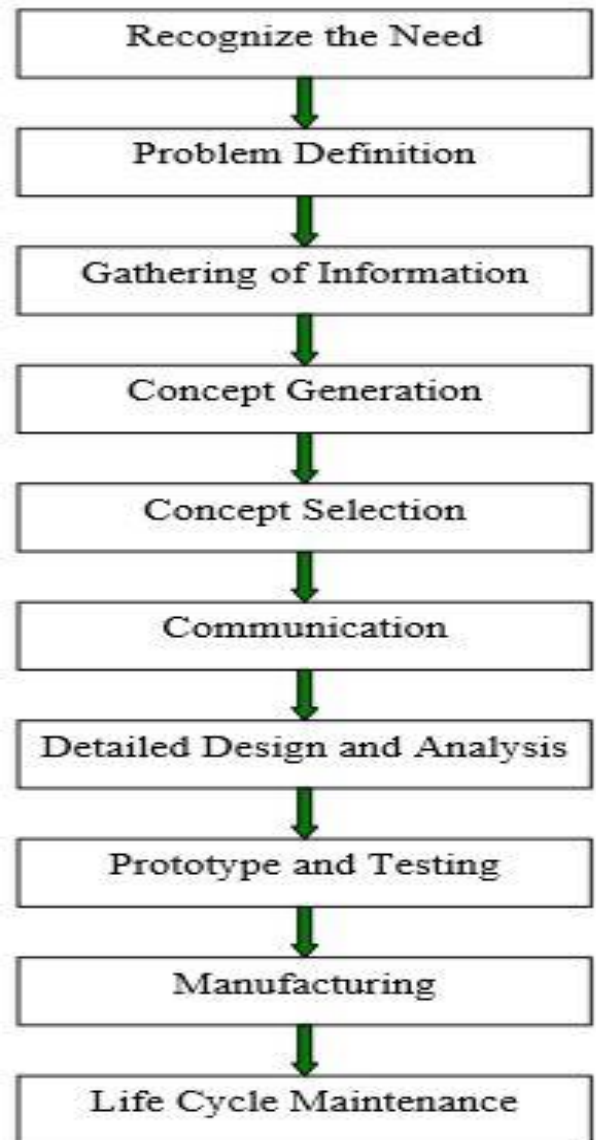Prototype and Testing

Manufacturing

Life Cycle Maintenance

Figure 4. 10Steps of product design processes

4.3.10 Design for Manufacturing / Assembly (DFM/DFA)
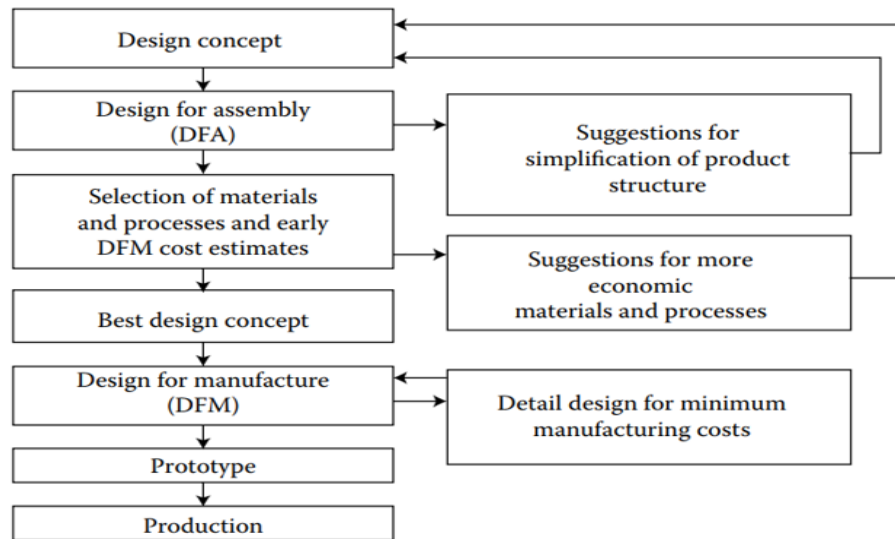
Figure 4. 11 DFMA process diagram

DFMA is a mix of two strategies, Design for Manufacturing (DFM) and Design for Assembly (DFA). This mix permits an item configuration to be expertly produced and effectively gathered with least industry cost. Using DFMA, we can evade, recognize, evaluate and eliminate waste and assembling failure inside a creation plan.

DFM/A will be a break from the organization. With DFM/A, the Design and Manufacturing Engineers cooperate as a group in arising the item's assembling and getting together techniques momentarily with the plan. Traditionally, the planning engineer plans the item at that point give the drawings to the assembling who at that point decides the assembling and get-together cycles.

Numerous specialists naturally particular the two into DFM and DFA since they have been very much characterized independently for quite a while. For powerful utilization of DFM/A the two exercises must exertion as one to acquire the best advantage and its diagram is on figure 4.11.

So, in our project, we decided to manufacture it from 4 mm acrylic sheets and cut it using laser cutting CNC machine then assembly it based on the CAD model and fix the motors and wheels which is manufactured from artelon and then connect all the electronics components together including the motors, batteries, controllers and cameras.

4.4 Robotics

is a sub-area of design and science that incorporates mechanical designing, electrical designing, software engineering, and others? Advanced mechanics manage the plan, development, activity, and utilization of robots and PC frameworks for their control, tangible input, and data handling. A robot is a unit that actualizes this collaboration with the actual world dependent on sensors, actuators, and data handling. The industry is a vital use of robots, or to be exact Industry 4.0, where mechanical robots are utilized.

Communitarian robots or Co-bots are getting progressively significant. In a developing number of territories in industry, ordinary modern robots are being supplanted by or sponsored up with shared robots. Co-bots cooperate with people underway cycles and are not, at this point kept separate from their human collaborators with defensive gadgets, as commonplace modern robots. Contrasted with conventional mechanical robots, shared robots are more modest, can be utilized all the more deftly, and are simpler to program.

As innovation advances, so too does the extent of what is viewed as mechanical technology. In 2005, 90%, everything being equal, could be discovered collecting vehicles in auto plants. These robots comprise principally of mechanical arms entrusted with welding or screwing on specific pieces of a vehicle. Today, we're seeing an advanced and extended meaning of mechanical technology that incorporates the turn of events, creation and utilization of bots that investigate Earth's harshest conditions, robots that help law-requirement and even robots that aid pretty much every aspect of medical care.

There are uncountable meanings of what PC writing computer programs is, yet the one I use is:

"Writing computer programs is the way you cause PCs to tackle issues."
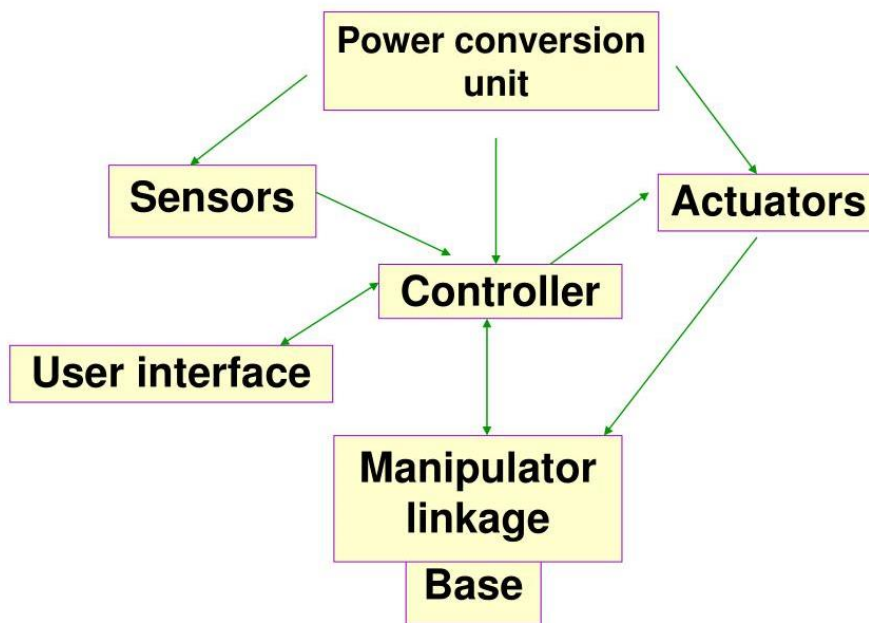
There are two key sayings here which are significant:

you: without the developer (you), the PC is unreasonable. It does what you advise it to do.

tackle issues: PCs are instruments. They are compound apparatuses, in fact, yet they are not mysterious or enchanted: they exist to make undertakings simpler.

PC projects (or programming) are what make PCs effort. Without programming, present-day PCs are simply muddled apparatuses for transforming power into heat. It's product on your PC that work your working framework, program, email, games, film player – just around everything.

Writing computer programs is an innovative undertaking: there is no set-in stone strategy to take care of an issue, similarly that there is no correct technique to paint an image. There are decisions to be made, and one way may appear to be well than another, however that doesn't mean the other is erroneous! With the correct guides and experience, a developer can abilities programming to take care of a limitless number of issues – from disclosing to you when your next train will reach to playing your #1 music. The prospects are hindered exclusively by your creative mind and the key components for robotics is represented in figure 4.12.

Figure 4. 12 Key components for robotics

These three concepts are the elementary logical structures in computer programming:
- Sequence: running commands in order
- Selection: creating choices
- Repetition: making the same thing more than once

Add to these views the capability to deal with inputs and outputs and to store information, and you have the tools to solve the common of all computing problems.

4.5 Path Planning Algorithm

The path planning algorithm makes it possible that the drone flies autonomous in a much better way than without the algorithm. The algorithm does however need some sort of map from which it can get information on the possible paths.

Multiple path planning algorithms are found and it is investigated what is the best to implement. For example, there is the Dijkstra algorithm and A* and D* and many more. The best algorithm is chosen based on a few criteria: the execution time, processing time. The path planning is chosen also based on the travel time and distance travelled. Based on these criteria the A* algorithm is chosen. It does best in all these aspects, although the difference are really small. The best thing about the A* algorithm is that it is really Smart, it really is the shortest path (known for now). There is a difference in optimal paths for example: one can go 5 meters to the right (x-direction) and then 5 meters to the right (y-direction) however this is not optimal, it would be much better to go in a diagonal path. This difference would not be visible by only identifying the execution time and processing time since this would be the same. The difference is only noticeable in the amount of distance (number of blocks). Distance is also a vague term, what is meant? in this case it would be Manhattan distance, which is in case the amount of blocks travelled. The results of the different algorithms cannot be shown since this hugely depends on the environment of the path (many obstacles or not). It can however be noted that the A* algorithm comes is on average 10 percent better than the other algorithms investigated.

4.5.1　How does the algorithm work?

Result: Optimal path between point A and point B initialization;

- **make an open list containing only the starting node**
- **make an empty closed list**

while the destination node has not been reached do consider the node with the lowest f score in the open list

  if this node is our destination node, then we are finished

      else

        put the current node in the closed list and look at all of its neighbors

        for each neighbor of the current node **do**

          if neighbor has lower g value than current and is in the closed list

            then

              replace the neighbor with the new, lower, g value; current node is

              now the neighbor's parent

          Else

          If current g value is lower and this neighbor is in the open list

            Then

              replace the neighbor with the new, lower, g value

              change the neighbor's parent to our current node

           else

           his neighbor is not in both lists add it to the open list and set its g

          end

        end

      end

    end

  end

end

the A* algorithm is the best and It overall wins because the algorithm really has the most optimal path instead of vertical and horizontal transitions and undoubtedly, Further research needs to be done to actually implement it and integrate it with the autonomous fight itself.

4.6   Navigation system

Drone mapping and navigation involve using drones to create detailed maps and navigate complex environments. This process requires a combination of hardware, software, and advanced algorithms.

Drone mapping and navigation using vision position estimation harness advanced computer vision techniques to enhance the drone's capability to autonomously navigate and map its surroundings. At the heart of this technology lies Visual Odometry (VO) and Simultaneous Localization and Mapping (SLAM). VO algorithms estimate the drone's position and orientation by analyzing consecutive frames captured by onboard cameras, detecting features such as edges or corners, and tracking their movement. This motion estimation is refined by combining visual data with inertial measurements from the drone's IMU (Inertial Measurement Unit), which records accelerations and angular velocities. SLAM takes this a step further by building a map of the environment while simultaneously keeping track of the drone's location within that map. This dual process of mapping and localization enables drones to navigate complex environments accurately and efficiently, even in GPS-denied scenarios such as indoors or densely built urban areas.

- **Simultaneous Localization and Mapping (SLAM)**

It is a critical technique in vision-based drone navigation, enabling drones to build a map of an unknown environment while simultaneously keeping track of their location within it. SLAM combines information from onboard cameras and sensors to create a detailed, real-time map of the surroundings. There are several types of SLAM, including monocular SLAM, stereo SLAM, and visual-inertial SLAM. Monocular SLAM uses a single camera to capture images and relies on detecting and matching features to estimate depth and build the map. Stereo SLAM uses two cameras to directly measure depth, providing more accurate spatial information. Visual-inertial SLAM integrates visual data with IMU readings to enhance accuracy and robustness, particularly useful in dynamic or cluttered environments. The process involves continuously updating the map

as new visual data is acquired and refining the drone's estimated trajectory through techniques like bundle adjustment and loop closure, which correct errors and ensure consistency in the map. This comprehensive mapping and localization capability make SLAM indispensable for autonomous navigation in complex, unstructured environments.

4.7 ROS

ROS, which stands for Robot Operating System, is often referred to as an operating system for robots, but that's not entirely accurate. ROS is actually an open-source framework that provides a collection of tools and libraries specifically designed to simplify the development of robotics software. It's like a giant toolbox filled with pre-built components that we can use to construct our robot's control system.

ROS offers several key advantages for robotics developers. First, it provides hardware abstraction. This means no need to worry about the specific details of how each piece of hardware on your robot works. ROS handles the communication between the software and the hardware components, allowing you to focus on writing code that controls the robot's behavior. Second, ROS facilitates communication between different software components. Imagine your robot has separate programs for processing sensor data, making navigation decisions, and controlling the motors. ROS enables these programs to communicate seamlessly with each other, ensuring that all parts of the robot work together effectively.

ROS also boasts a large and active community of developers. This means that there's a wealth of existing code and resources available for common robotics tasks. If we are building a robot that needs to perform tasks like object recognition or mapping its environment, there's a good chance there's already a ROS package that can help you do it. This can save a tremendous amount of time and effort compared to starting from scratch. By leveraging the vast resources of the ROS community.

And the figure 4.13 shows how the ROS Master and ROS nodes communicate with the help of topics and services and this example we have already implemented while we were starting working with ROS.
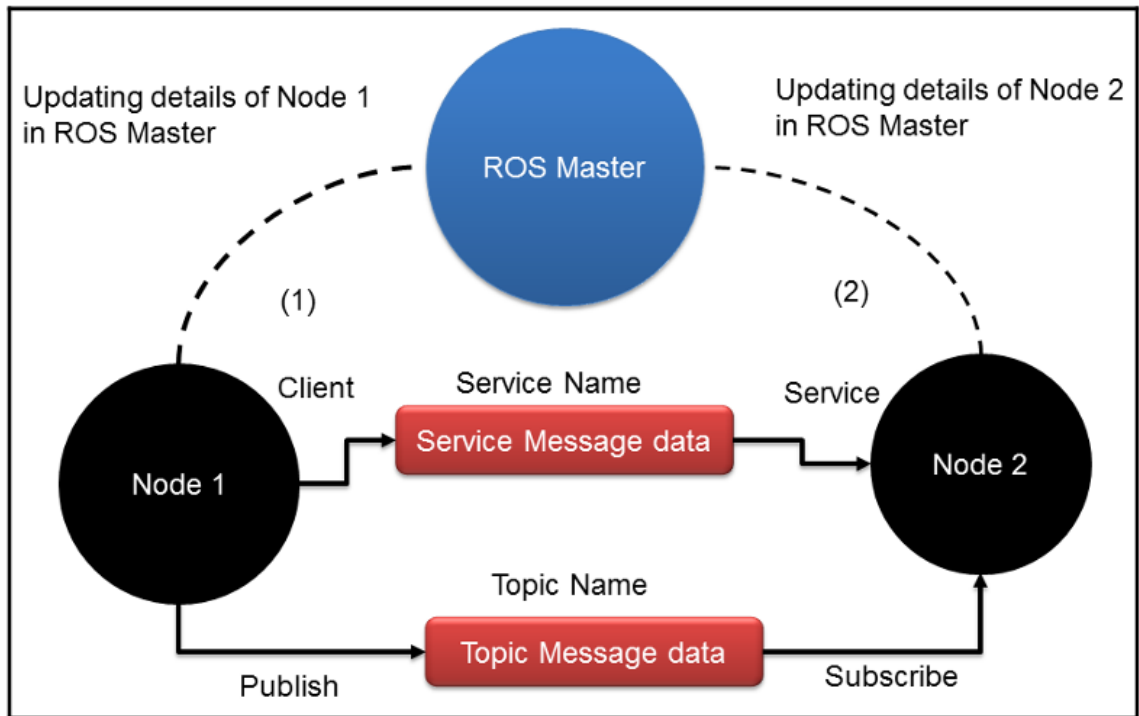
Figure 4. 13 Communication between ROS nodes

Runtime tools are what you'll use to run your ROS code on your robot or in a simulation environment. Key examples include:

ROS Core: This is the heart of ROS and provides essential functionalities like node management, message passing, and parameter server access.

- ROS Nodes: These are independent processes that communicate with each other using ROS topics (data streams) and services (remote procedure calls). You'll write your robot control code as ROS nodes.

- ROS Topics and Services: As mentioned earlier, topics allow nodes to publish and subscribe to data streams, enabling communication between different parts of your robot software. Services provide a way for nodes to request and receive specific pieces of information from each other.

- RViz: This is a powerful 3D visualization tool that lets you view sensor data, robot poses, and other information published by your ROS nodes. It's a valuable tool for debugging and understanding the behavior of your robot software.

There are many other ROS tools available, but these are the core ones you'll use for most robotics projects. The specific tools you'll need may vary depending on the complexity of your robot and your development preferences.

4.8   Design standards

ANSI/IEEE std 991-1986 provides standard guidelines for logic circuit diagrams. It includes definitions, requirements for assignment of logic levels, application of logic symbols, presentation techniques, and labelling requirements. According to standard a logic circuit diagram predominantly uses symbols for logic functions to depict the overall function of a circuit. The standard divided the logic circuit diagram into two types' i.e. Basic and detailed circuit diagram. The proponent's circuit diagram is according to standard Basic Circuit Type Diagram. It provides logic symbols and other necessary functional symbols, together with their signal and major control path connections. [11]

IEEE 1451 Transducers (Sensors and Actuators interface) - is a set of transducers (Sensors and Actuators) interface standards published by the Institute of Electrical and Electronics Engineers (IEEE) Instrumentation and Measurement Society. Since project deals with interfacing sensors and actuators to the microcontroller this specific standard was implemented. [12]

IEEE std 315-1971 (included with supplement ANSI/IEEE std 315A-1986) provides detailed guidelines for using graphic symbols for electrical and electronic diagrams. The proponent's project involves a set of resistors with different ohm ratings, capacitors, diodes and sensors. Project circuit diagrams are designed according to the standard graphic symbol's specifications. [13]

ISO 128 Mechanical Design - ISO 128 is an international standard (ISO), about the general principles of presentation in technical drawings, specifically the graphical representation of objects on technical drawings. [14]

ANSI/IEEE std 100-1996 (Sixth Edition) provides a standard dictionary for electrical and electronic terms. This standard describes more than 20, 254 terms and definition associated with electrical and electronic. Each year an average of 7000 new and revised terms are added into this standard. Each entry is considered as an official standard of IEEE. The proponents used

extensively this standard while defining and using an acronym in the design, development and documentation of associated project components e.g., transducers, amplifiers, timer, resistors, capacitors etc. [15].

4.9   Project issues

- The body of the robot is manufactured from thin layers of Acrylic in order to be able to assembled and modified easily and this may lead it to broken.

- The DC motors high speed and controlling methods may lead to wrong results for the sensor while following the line so we have to use encoders for feedback.

- The OCR used in detection the names of the book may not produce good results if it was applied on Arabic books.

- It is not possible for make the robot move from floor to another if the library consists of multiple floors.

- The operation of mapping and navigation may be little slow as the processing of the data is little complicated and the raspberry is not powerful like normal computer.

- Not using IMU sensor may lead to obstacles crash as the feedback from encoder can be little dependable.

4.10 Bill of materials

Table 4.1 below shows proponent project bill of material which specifies the total quantity used along with unit and total price.

Table 4. 1 Bill of Material

No	Description	Quantity	Unit Price	Total Price
1	Logitech C505 camera	1	1800	1800
2	Manufacturing	1	900	900
3	Raspberry pi 4	1	6800	6800
4	Microcontroller Arduino mega	1	1500	1500
5	Power supply 12V	1	300	300
6	LI-Batteries	8	70	560
7	Sheet acrylic	1	1900	1900
8	On/Off switch	1	20	20
9	CNC Laser cut wood	1	600	600
10	Wiring, painting, transportation, etc.	1	850	850
11	Damage, redesign, testing and other incidents during development	1	950	950
12	Kinect camera	1	2500	2500
13	Raspberry pi 4 case and Caple	1	1600	1600
			Total (EGP)	**20280**

4.11 Block diagram

The following graph of 4.1 shows the block diagram that represent the robot several components of inputs and outputs. Raspberry pi handles the two signals from the two cameras and Arduino Microcontroller integrates various controls, and manages various inputs and outputs of the robot and OCR module plays the role of getting the books name and store it in the database.
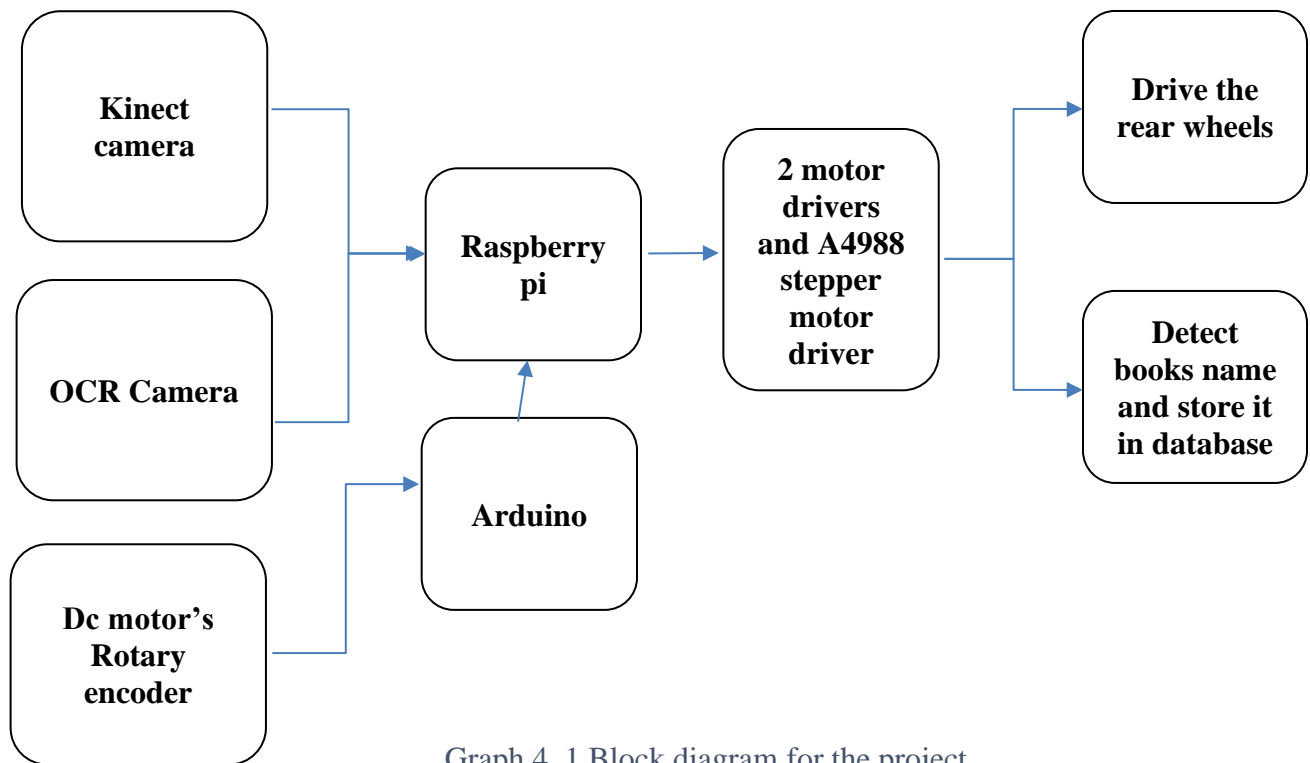
Graph 4. 1 Block diagram for the project

4.12 Circuit Diagram

Graph 4.2 Below is the circuit diagram of the main components of this project and we attached the connections of each component and we will explain each component and how it's working in this project.
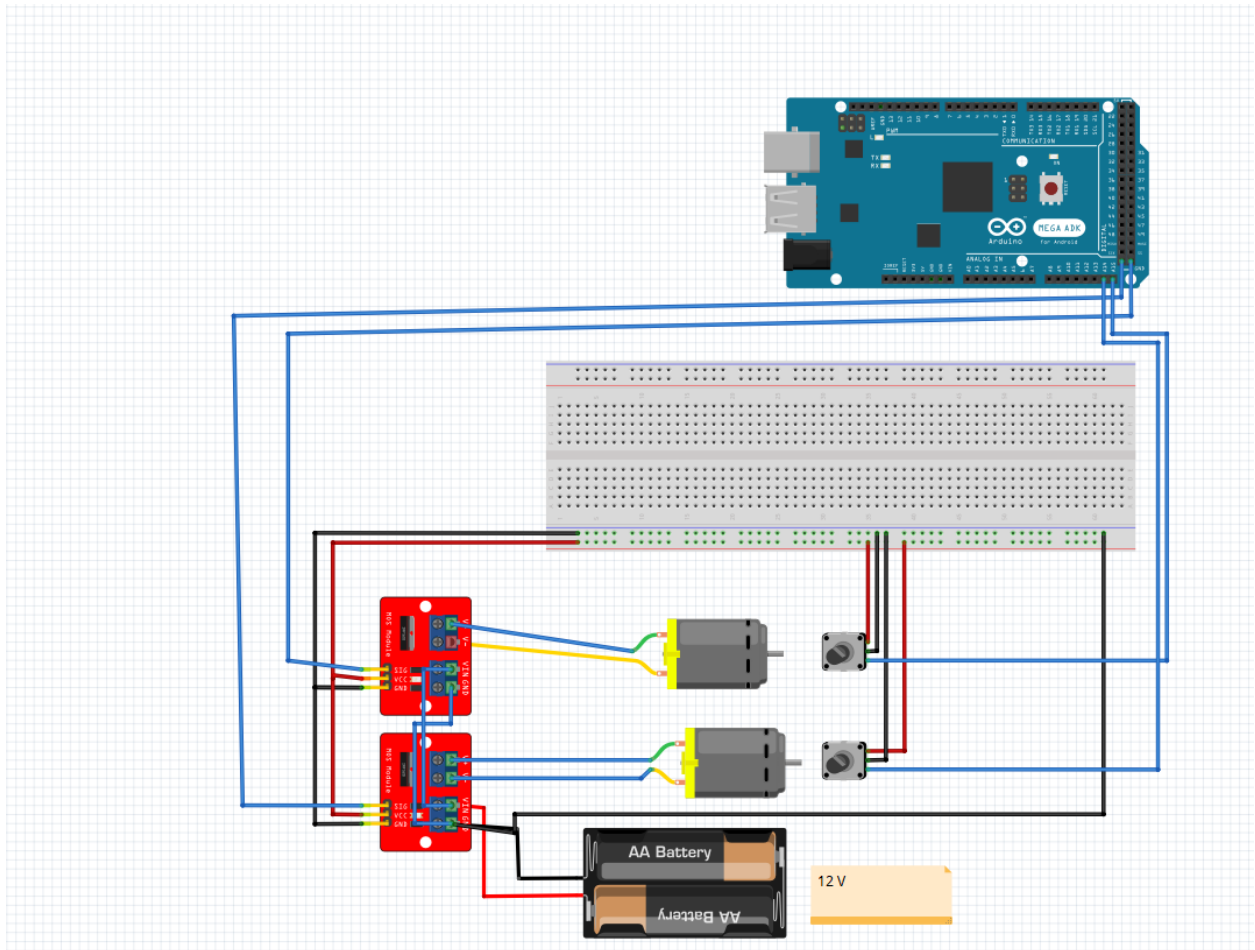
Graph 4. 2 Electronics circuit diagram

We used Fritzing for drawing the circuit diagram of the project which is only one circuit diagram for driving the robot and send feedback from the rotary encoder to the Arduino which will get the data from raspberry pi to driver the motors based on the operation and feedback from the encoder and the two motors is driven by two motor drivers BTS7960 connected to power distribution board that deliver 12V dc power.

4.13 Gantt chart

Table4.2 below shows proponent Gantt chart .Gantt chart provides the detail of project activities that are divided among the participants of the project. The grey gradient block below represents each participant. The horizontal axis at the top represents the schedule of project activities divided in weeks. The left vertical axis represents various project activities.

Table 4. 2 Gantt Chart

Activity	Sep				Oct				Nov				Dec			
Tasks(week)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Documentation																
Designing and installation																
Programming & Prototype Construction																
Testing and Evaluation																
Development																
Final Assembly																

	Mechanical committee
	Hardware committee
	Software committee

Chapter 5

Design procedure, functional analysis & implementation.

In this part of our research project thesis different design procedures, function analysis of design project and implementations will be discussed thoroughly. Mostly all the technical detail and specifications of design, hardware components and software ends will be discussed in detail and different techniques and tools will be used to justify different discussions regarding choosing specific designs and perspectives to optimize the project design.

5.1 Project description

Design research project library assistant robot consists of three parts with dimensions of 550*350*650 mm (L*W*H), the first part is the base part which is which consists of the main vehicle which holds the whole chassis, this part consists of one acrylic sheet that holds the DC motors and the free wheels, there are two upper acrylic sheets connected by metal rods screw and nut, the two motors are attached to two rear wheels, the base frame holds the electronics circuit, power supplies circuit, and the batteries, Kinect camera is in front used for mapping and navigation, the second part is the OCR camera will be attached to the top plate of the robot, the third part is the outside chassis which is manufactured from a thin layer of wood 2MM width and to be flexible to cover the whole robot, we have made lots of cuts on it and this cuts allow the wood to bend on the outside shape of the robot.

5.2 Project Functions Analysis

There are two sources of information in the robot, the first one is the Kinect camera, which is used for driving the robot in two stages, the first one is mapping at which the Raspberry will create a map for the working environment and then it will be used for navigation and at this stage the OCR camera will operate to scan the box on shelves to get the names for the books on the shelf then store this names on a database which will be accessible to the library visitors and the librarian as well, all the functions on the robot will be applied using the two signals from this two cameras

with the help of rotary encoder to get a feedback from the motors to make the motion accurate and smooth.

5.3   Project implementation

In this part, the project implementation steps will be discussed. The implementation procedure is concluded in the following points:

- Putting the manufacturing and testing steps and plan.

- Gathering and buying of necessary mechanical fixed parts and components.

- Testing the behavior of the encoder with Arduino and serial communication.

- Testing the behavior of the Raspberry Pi with Arduino and serial communication.

- Testing the DC motors and drivers and how can we manage the speed, torque, and power consumption for it.

- Design the CAD model of the robot based on the existing fixed components.

- Setup Linux OS and start working with ROS and learn its concepts.

- Export URDF from Solidworks and create a simulation for the robot in ROS Gazebo and RVIZ.

- Manufacturing the different parts and assembling them based on the assembly of the CAD model.

- Assembly of the robot chassis with the motors and the drivers.

- Testing the control and power connections.

- Calibrating and testing the machine operation.

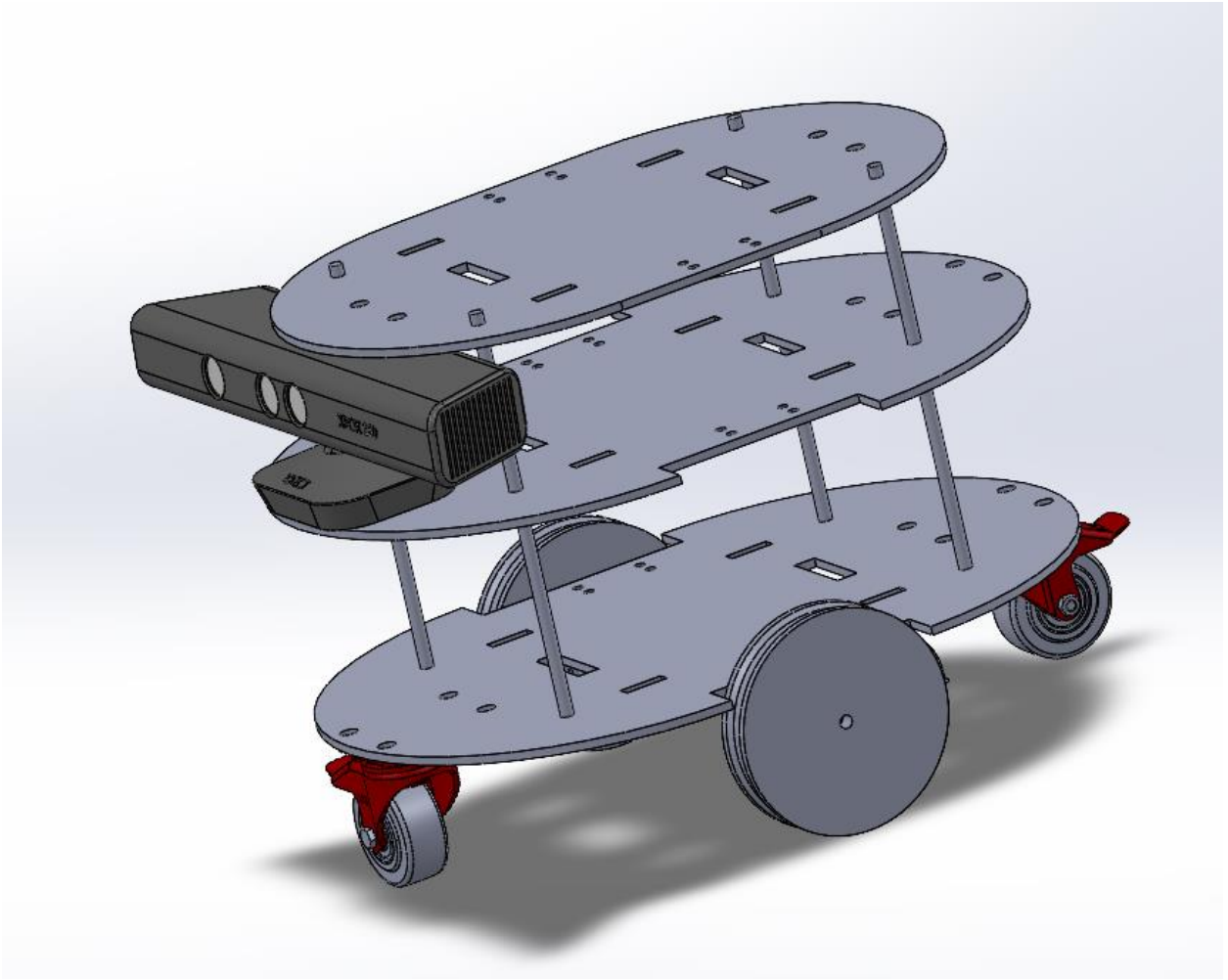- Final testing and see the results of the project.
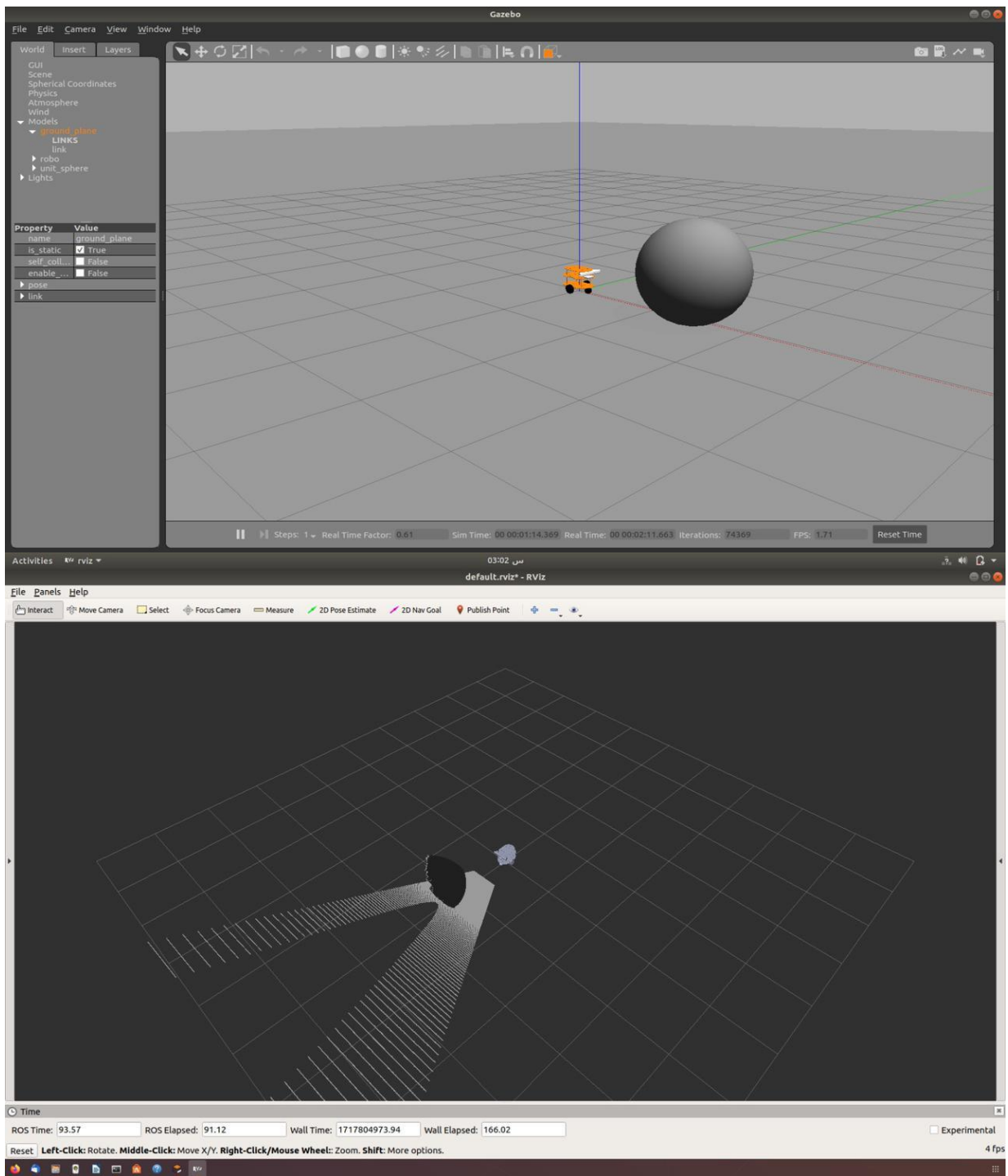
Figure 5. 1 Project Implementation

5.4 FMEA

Failure Modes and Effects Analysis (FMEA) (table 5.1) is a systematic, proactive method for evaluating a process to identify where and how it might fail and to assess the relative impact of different failures, to identify the parts of the process that are most in need of change.

Table 5. 1FMEA

Items	Component Description	Failure Mode	Effects	Safeguard	Actions
1	DC Motors	No power/ Barely moving	The robot will not move	Check the connection/ Check the programming	Check power supply Check the dual channel relays Replace or Fix
2	Arduino Mega	No power	All machine stops.	Check connections	Check the power supply source, check for the circuit breaker position. Shutdown the power, check the battery.
3	Power Supply	No power	All process stops.	Check connections	Check the power supply source, check the input source of the power. Check power breakers.
4	Kinect camera	No signal	Move randomly	Replace it with another one	Apply individual connection and testing on it
5	OCR camera	Stop scanning box	Scanning shelves problem	Check connections with the Raspberry Pi	Reassembly the camera and calibrate it well

Chapter 6
Project summary

6.1   Summary

Experience is priceless, this is the major part we had gained while the journey of making this project, we had experienced dealing with many software programs in order to end up this project, we start it with Microsoft office word which we used in writing all documentations of our project, the Arduino Ide was also great programming program as it provides many examples and codes to understand the operation of the microcontroller, raspberry pi and ROS at which we have experienced how to link between different parts and apply mapping and navigation using SLAM, the main part that we exerted too much effort on is the mechanical design of the robot and the electronics as it the motors we used has many hard issues like the high current consumption so we had experienced too many solutions to drive this motors and control its direction in addition to it was not possible to use low power motors instead of this two high torque motors so we had to do this, the mechanical design is full of hard challenges beginning with choosing the right material for manufacturing this robot, the CAD model which has too many details in order to decrease possibilities for any faults then dealing with technicians which really something hard as they have their own way of creating and understanding the object, then finally the calibration process which was really hard as it needs too much patience in order to reach the best result.

6.2   Conclusion

I and the other group members had exerted too much effort to make this robot reach the final stage of assembly, calibration, and operation, we hope that doing our best and catch the purpose of working on such projects, passion and enjoy what we were doing was the driving principles for us to work on this project, the jobs market outside our university needs too much passion and hard work to achieve much more and make a successful career we are seeking for.

Chapter 7

Recommendations

Generally, this type of robot can be used for only showing the idea of scanning the box in the library and assisting in fixing the problem of missing the location of the box in the library, assisting in finding it in the fastest way, increasing the productivity for readers and researchers. Still, there are some considerations we will have to follow up on like, what if the library is big and it needs high processing time and power, what will happen if one of the components gets disconnected? We thought about providing some enhancements to the robot to make it more reliable and efficient like:

- Replacing the Arduino and motor driver circuits with PLC or Classic control to work for a long duration.

- Replace the breadboard with a printed circuit board to avoid any misconnection or components break down.

- Testing the robot more in a complex environment to reach the best results.

- Try different operation algorithms to enhance the results.

Appendices

- **Appendix A (2D CAD drawings)**

wheels

A3
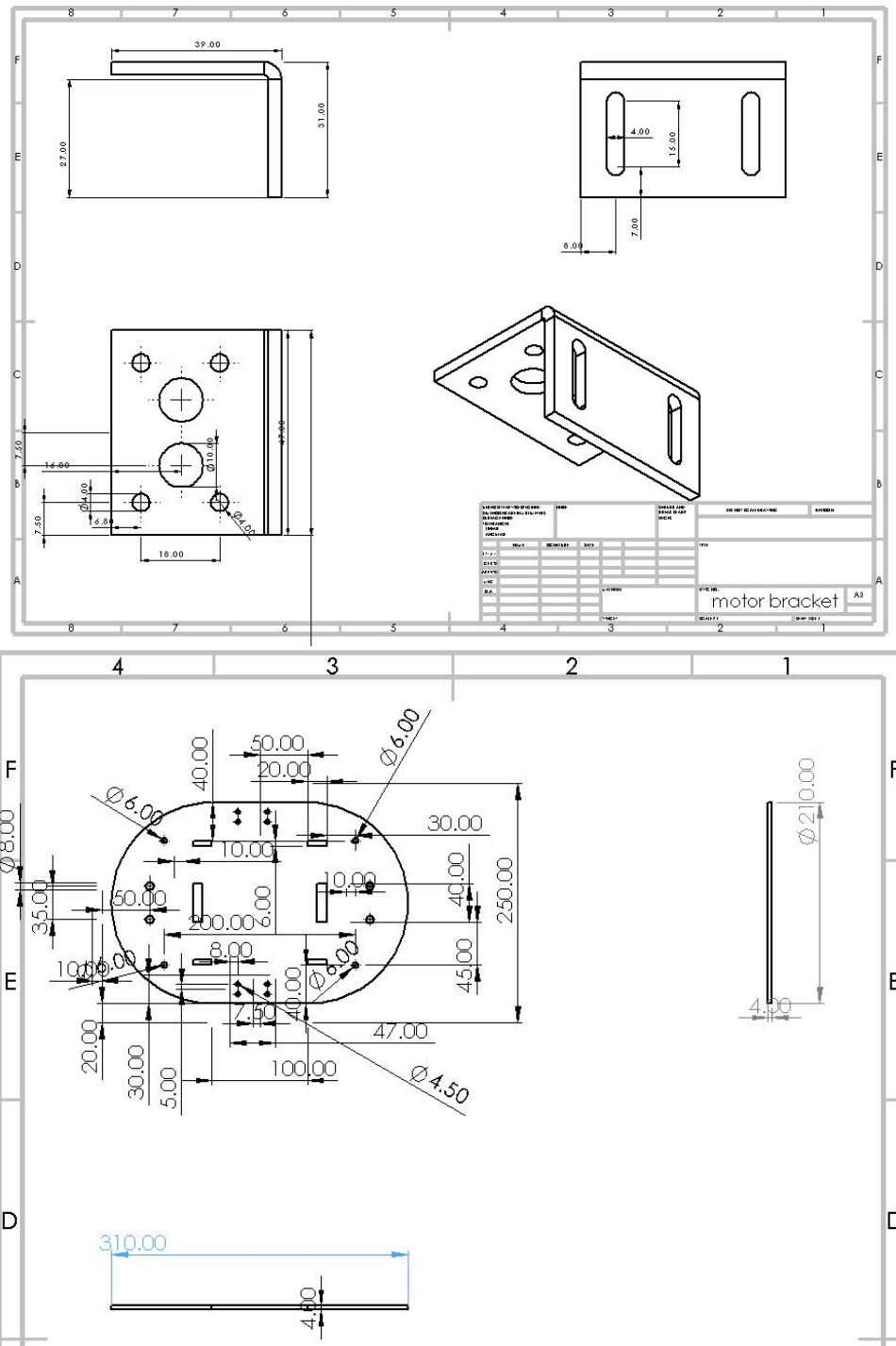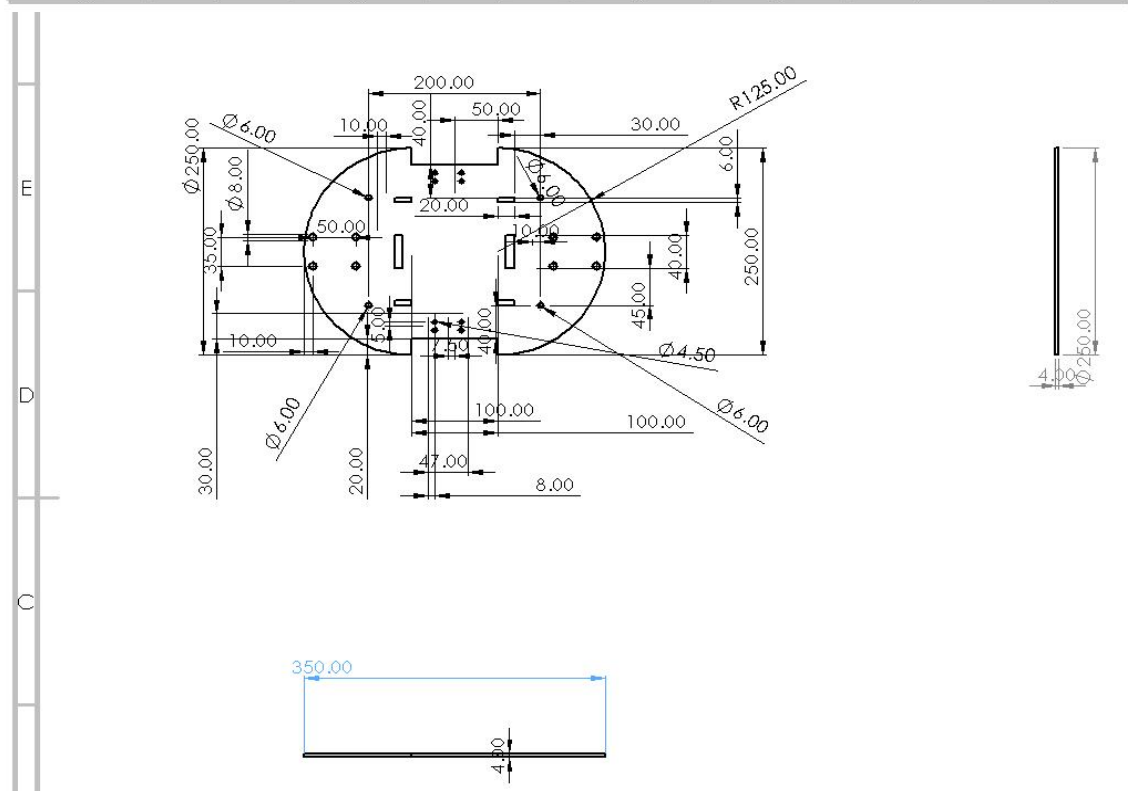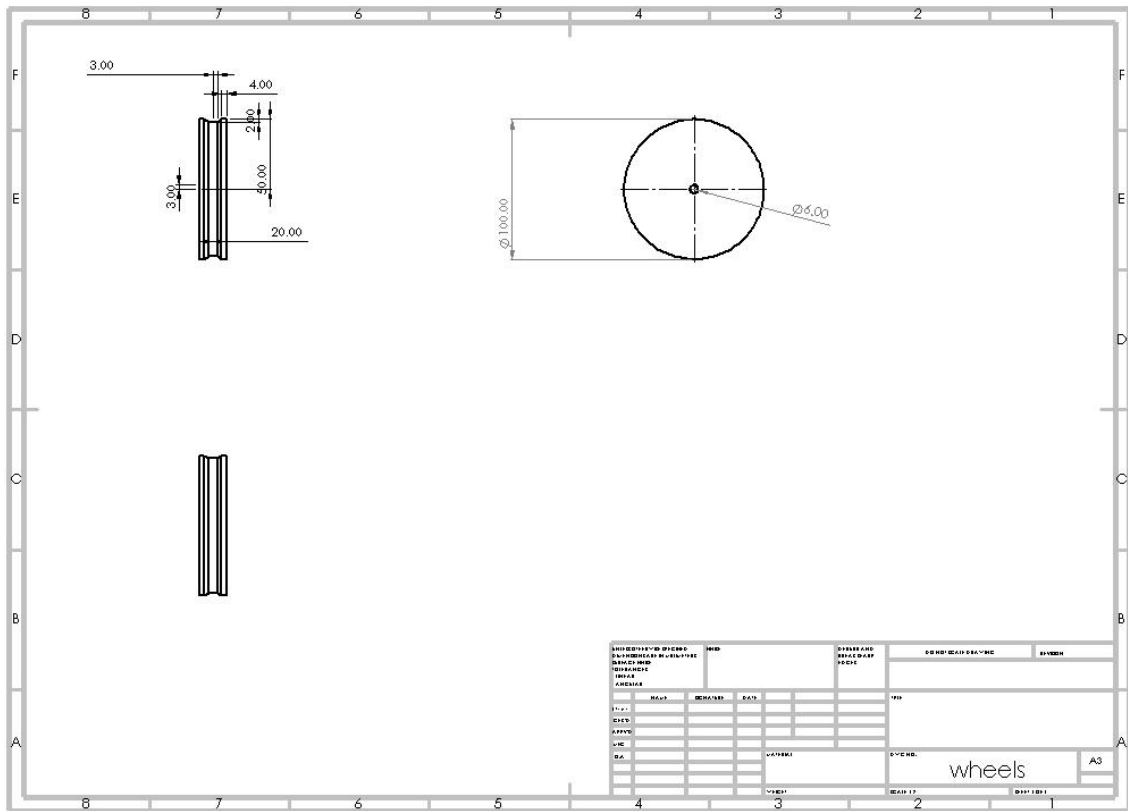
• **Appendix B (QR detection code)**

```python
import cv2
from pyzbar.pyzbar import decode
import sqlite3


def create_table(conn):
    # Create a new table with the desired schema
    conn.execute('''CREATE TABLE IF NOT EXISTS BOOKS
                (BOOK_NAME TEXT NOT NULL);''')


def extract_qrcodes_from_camera():
    # Initialize the camera
    cap = cv2.VideoCapture(0)

    print("Press 'q' to capture the QR code and exit.")

    extracted_data = []

    while True:
        # Capture frame-by-frame
        ret, frame = cap.read()

        if not ret:
            print("Failed to grab frame")
            break

        # Display the resulting frame
        cv2.imshow('frame', frame)
```

```python
        # Convert the frame to grayscale
        gray_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

        # Detect QR codes in the frame
        qr_codes = decode(gray_frame)

        # Extract and decode the data from QR codes
        for qr_code in qr_codes:
            qr_data = qr_code.data.decode("utf-8")
            extracted_data.append((qr_data,))  # Add book name to the list

        # Break the loop on 'q' key press
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break

    # Release the camera and close windows
    cap.release()
    cv2.destroyAllWindows()

    return extracted_data

def insert_books(conn, extracted_data):
    # Insert extracted book names into the table
    conn.executemany("INSERT INTO BOOKS (BOOK_NAME) VALUES (?)", extracted_data)

def main():
    # Connect to SQLite database
    conn = sqlite3.connect('librarynewdb.sqlite')

    # Create the table if it doesn't exist
    create_table(conn)
```

```python
    # Extract QR codes and their data from the camera
    extracted_data = extract_qrcodes_from_camera()

    # Insert extracted book names into the table
    insert_books(conn, extracted_data)

    # Commit changes to the database
    conn.commit()

    # Close the database connection
    conn.close()

if __name__ == "__main__":
    main()
```

- **Appendix C (ROS codes)**

Launch file

```xml
<launch>
 <!-- Map Server -->
 <node          name="map_server"          pkg="map_server"          type="map_server"
args="/home/eslam/my_robot_ws/src/my_robot_navigation/map/arche.yaml" />
 <!-- AMCL (Adaptive Monte Carlo Localization) -->
 <node name="amcl" pkg="amcl" type="amcl" output="screen">
  <param name="odom_frame_id" value="odom" />
  <param name="base_frame_id" value="base_footprint" />
  <param name="global_frame_id" value="map" />
  <!-- Adjust parameters as needed -->
 </node>
```

```xml
<!-- Move Base -->
<node name="move_base" pkg="move_base" type="move_base" output="screen">
  <rosparam file="$(find my_robot_navigation)/config/costmap_common_params.yaml" command="load" />
  <rosparam file="$(find my_robot_navigation)/config/local_costmap_params.yaml" command="load" />
  <rosparam file="$(find my_robot_navigation)/config/global_costmap_params.yaml" command="load" />
  <rosparam file="$(find my_robot_navigation)/config/base_local_planner_params.yaml" command="load" />
  <rosparam file="$(find my_robot_navigation)/config/move_base.yaml" command="load" />

</node>


<!-- RViz -->
<node name="rviz" pkg="rviz" type="rviz" args="-d $(find my_robot_navigation)/rviz/navigation.rviz" />
</launch>
```

- **Slam.launch**

```xml
<?xml version="1.0"?>
<launch>
  <!-- Node to convert depth image to laser scan -->
  <node name="depthimage_to_laserscan" pkg="depthimage_to_laserscan" type="depthimage_to_laserscan" output="screen">
    <param name="output_frame_id" value="link_kinect" />
    <remap from="image" to="/camera/depth/image_raw" />
    <remap from="scan" to="/scan" />
  </node>

  <!-- Include Gazebo launch file -->
  <include file="$(find robot_is)/launch/gazebo.launch" />
```

```
<!-- Include GMapping launch file -->
<include file="$(find robot_isnav)/launch/gmapping.launch" />


<!-- Include teleop launch file -->
<include file="$(find robot_isnav)/launch/mybot_teleop.launch" />


<!-- Include RViz display launch file
<include file="$(find robot_is)/launch/display.launch" />
-->
</launch>
```

- **Gazebo.launch**

```
<?xml version="1.0"?>
<launch>

 <include file="$(find gazebo_ros)/launch/empty_world.launch">
  <arg name="paused" value="false"/>
  <arg name="use_sim_time" value="true"/>
  <arg name="gui" value="true"/>
  <arg name="recording" value="false"/>
  <arg name="debug" value="false"/>
 </include>


<param       name="robot_description"       command="$(find       xacro)/xacro      '$(find
robot_is)/urdf/robot_is.xacro'"/>


<!-- Spawn a robot into Gazebo -->
 <node name="spawn_urdf" pkg="gazebo_ros" type="spawn_model" output="screen" args="-
urdf -param robot_description -model robo" />
```

```xml
    <!-- Publish joint states

 <node  name="joint_state_publisher"  pkg="joint_state_publisher" type="joint_state_publisher"
/>

<node name="robot_state_publisher" pkg="robot_state_publisher" type="robot_state_publisher"
/>
-->
</launch>
```

- **Plugins gazebo**

```xml
<?xml version="1.0"?>
<!-- A gazebo plugin file that connects the URDF-written robot with the following gazebo plugins:
1: "differential_drive_controller" for differential drive simulation
2: "camera_controller" for Kinect simulation
Written by Muhammad Asem m7mad3asem@gmail.com -->
<robot name="robot_is">

  <!-- +++++++++++++++++ Diff drive plugin +++++++++++++++++++ -->
<gazebo>
   <plugin name="differential_drive_controller" filename="libgazebo_ros_diff_drive.so">
    <legacyMode>false</legacyMode>
    <alwaysOn>true</alwaysOn>

    <updateRate>20</updateRate>

    <leftJoint>joint_left</leftJoint>
    <rightJoint>joint_right</rightJoint>

    <wheelSeparation>0.21</wheelSeparation>
```

```xml
      <wheelDiameter>0.02</wheelDiameter>

      <torque>20</torque>
      <commandTopic>cmd_vel</commandTopic>
      <odometryTopic>odom</odometryTopic>
      <odometryFrame>odom</odometryFrame>
      <robotBaseFrame>base_footprint</robotBaseFrame>
      <!-- Added Parameters -->
      <rosDebugLevel>INFO</rosDebugLevel>
      <publishWheelTF>true</publishWheelTF>
      <publishOdomTF>true</publishOdomTF>
      <publishWheelJointState>true</publishWheelJointState>
      <wheelAcceleration>1.0</wheelAcceleration>
      <wheelTorque>10.0</wheelTorque>
      <odometrySource>1</odometrySource>
      <publishTf>true</publishTf>

  </plugin>
</gazebo>

<!-- +++++++++++++ Chassis Orange color +++++++++++++++++ -->

<gazebo reference="base_link">
  <material>Gazebo/Orange</material>
</gazebo>

<!-- +++++++++++++++++ Left wheel Blue Color +++++++++++++++++ -->

<gazebo reference="Link_left">
  <material>Gazebo/Black</material>
</gazebo>
```

```xml
<!-- ++++++++++++++ Right wheel Blue Color +++++++++++++++++ -->

<gazebo reference="Link_right">
 <material>Gazebo/Black</material>
</gazebo>

<gazebo reference="frontlink">
 <material>Gazebo/Black</material>
</gazebo>

<gazebo reference="backlink">
 <material>Gazebo/Black</material>
</gazebo>

<!-- +++++++++++++++++ Kinect Plugin +++++++++++++++++ -->

<gazebo reference="link_kinect">
 <sensor type="depth" name="camera">
  <always_on>1</always_on>
  <visualize>true</visualize>
  <camera>
   <horizontal_fov>1.047</horizontal_fov>
   <image>
    <width>640</width>
    <height>480</height>
    <format>R8G8B8</format>
   </image>
   <clip>
    <near>0.1</near>
    <far>100</far>
```

```xml
      </clip>
    </camera>
    <plugin name="camera_controller" filename="libgazebo_ros_openni_kinect.so">
        <alwaysOn>true</alwaysOn>
        <updateRate>10.0</updateRate>
        <cameraName>camera</cameraName>
        <frameName>kinect_optical</frameName>
        <imageTopicName>rgb/image_raw</imageTopicName>
        <depthImageTopicName>depth/image_raw</depthImageTopicName>
        <pointCloudTopicName>depth/points</pointCloudTopicName>
        <cameraInfoTopicName>rgb/camera_info</cameraInfoTopicName>

<depthImageCameraInfoTopicName>depth/camera_info</depthImageCameraInfoTopicName>
        <pointCloudCutoff>0.4</pointCloudCutoff>
        <hackBaseline>0.07</hackBaseline>
        <distortionK1>0.0</distortionK1>
        <distortionK2>0.0</distortionK2>
        <distortionK3>0.0</distortionK3>
        <distortionT1>0.0</distortionT1>
        <distortionT2>0.0</distortionT2>
        <CxPrime>0.0</CxPrime>
        <Cx>0.0</Cx>
        <Cy>0.0</Cy>
        <focalLength>0.0</focalLength>
     </plugin>
   </sensor>
 </gazebo>

</robot>
```

References

[1] Shaker, M., Hamzah, M., Osama, N. (2019, March). The Specifications of Arduino Mega Microcontroller. Researchgate.com.

[2] Raspberry pi website. https://www.raspberrypi.com/products/.

[3] Miles, B. (2001). Common applications for linear rails. Linearmotiontips.com.

[4] Tom, V., Glenn, M., Raphael, F., Bram, V. (2015. Augast). Modeling and design of geared DC motors for energy efficiency: Comparison between theory and experiments. Researchgate.com.

[5] Nilesh, C. (2014). Brushed Dc motor driver. Rubo.in.

[6] Ahmed, z., Mohamed, S., Liong, C. (2007,January). Isolation of Dots for Arabic OCR using Voronoi Diagrams. Researchgate.com.

[7] Mansour, A., Ahmed, J. (2007, July). Optical Character Recognition System for Arabic Text Using Cursive Multi-Directional Approach. Journal of Computer Science 3 (7): 549-555, 2007.

[8] Alberto, E. (2009, June). Sonar-based real-world mapping and navigation. IEEE Journal on Robotics and Automation.

[9] Mostafa, E., Dilsad, E. (2012, septemper). Path planning of line follower robot. Education and Research Conference (EDERC), 2012 5th European DSP.

[10] Taih, C. (2009, July). Comparing OCR and RFID application in the Container Shipping Industry Alternatives Criteria. Researchgate.com.

[11] IEEE Standard Graphic Symbols for Logic Functions. (1984). IEEE STANDARD.

[12] IEEE Standard for Logic Circuit Diagrams. (1986) IEEE STANDARD.

[13] Adnan Aqeel. Introduction to Arduino Mega. (2018). The Engineering Projects.com. IEEE. (1975).

[14] IEEE GRAPHIC SYMBOL STANDARD (315-1975). IEEE.

[15] IEEE. Song, E. Y., & Lee, K. (2008). Understanding IEEE 1451-Networked smart transducer interface standard-What is a smart transducer. IEEE Instrumentation & Measurement Magazine, 11(2), 11-17.