جامعة الأمير محمد بن فهد
PRINCE MOHAMMAD BIN FAHD UNIVERSITY

College of Engineering

Department of Electrical Engineering

Fall 2019-2020

Senior Design Project Report

Autonomous Mobile Robotic Library System

In partial fulfillment of the requirements for the Degree of Bachelor of Science in Electrical Engineering

Team Members

	Student Name	Student ID
1	Fahad Alsubaie	201500450
2	Ibrahim Alharbi	201101220
3	Aws Aleryani	201503139
4	Nassim Mahamoud	201503130
5	Ahmed Alhafidh	200901602

Project Advisor:

Dr. Sadiq Alhuwaidi

Abstract

The main objective of this project is to design a robot that handles library services effectively, develop a smart system to maintain a library using controller based system, reduce the load and the time consumption of human services, and ease and simplify the job of monitoring the library services and saving expenses by reducing human dependency.
The robot performs multipurpose services and assistance for library users. It brings and returns books for students and records database. The robot interacts between students and library system.

Table of *Contents*

1. Introduction

1.1 Project Definition

The goal of our project (An Autonomous Mobile Robotic Library System) is to design a smart human-robot interface, which will perform multipurpose services and assistance for library users; mainly bringing and returning books for students and other related services. The robot interacts between students and library systems.

1.2 Project Objectives

- Design a robot that handles library services effectively.
- Develop a smart system to maintain a library using controller-based systems.
- Reduce the load and the time consumption of human services.
- Ease and simplify the job of monitoring the library services.
- Saving expenses by reducing human dependency.

1.3 Project Specifications

- Controlling and monitoring book lending requests.
- Lifting capability of reaching 1.5 meters high shelves.
- Handling library books autonomously.
- Smart robotic detection of obstacles and shelves.

1.4 Product Architecture and Components

This system works automatically. This robot receives the request from the tablet, then sends it to the controllers. Controllers work as the brain of this robot and send the information to the motors to move. Also, the controllers send signal to the pixy cam to start working and following the line. The robot starts moving to that shelve that has the book. When the robot reaches the correct shelve, it stops and sends a signal to the controllers and the controllers will send a signal to the arm to move to the right shelve and the RFID to check the books. In the end, when the robot takes the right book, it will send the signal to the controllers to start going back to the main place, and it will do the same order to return the book to the right shelve.

1.5 Applications

- This autonomous robot can be used in warehouse like what Amazon did.
- It can be used as a waiter for restaurant.
- It can be used for electric vehicle (EV) charger.

2. Literature Review

2.1 Project background

The libraries are major parts of our modern lives, for increasing amounts of books which are being stored in libraries worldwide with contents from different parts of the world. As a result, finding books in libraries task has become difficult even for library employees. Therefore, we are aiming to build a robotic system that is able to autonomously detect books using RFID technology, and locate them in their shelves or bringing them to library users.

Some common problems are:

- Time consumption to search for books.
- Load on library staff to process individual search requests.
- No regular updating of digital database.
- No interactive technology involved.

Previous solutions involved detecting books using barcode, which is not efficient as barcode scanners are missing many books, but our method of implementing RFID's is precise in detecting books even at a peak of a second.

We are aiming to provide the following solutions:

- Develop a smart robotic system to maintain a library using controller-based systems.
- Reduce the load on human services.
- Ease and simplify the job of monitoring the library services.
- Saving expenses by reducing human dependency.

2.2 Previous Work

1- Microcontroller Based Robotic Arm Development for Library Management System

Other people in different universities have done some previous similar projects. This definitely highlights the importance of having an autonomous robot within a workspace. For instance, the Department of Electrical Engineering Meghnad Saha Institute of Technology, Kolkata, India published an article concerning an autonomous robot, which could pick books and return books in different shelves within a library. In their project, when returning a book, a bar code scanner used to identify the book in order to determine where to be placed. In addition, they used IR sensors to locate the shelves where the book should be picked or returned. It is actually a smart way to localize a book. Indeed, this project is similar to ours. However, the components used to build it will differ to the ones which will be used to build ours. For example, they used IR while we are going to use RFID. They used wifi to connect the system to the Arduino while we may use Zigbee and so on. [1]

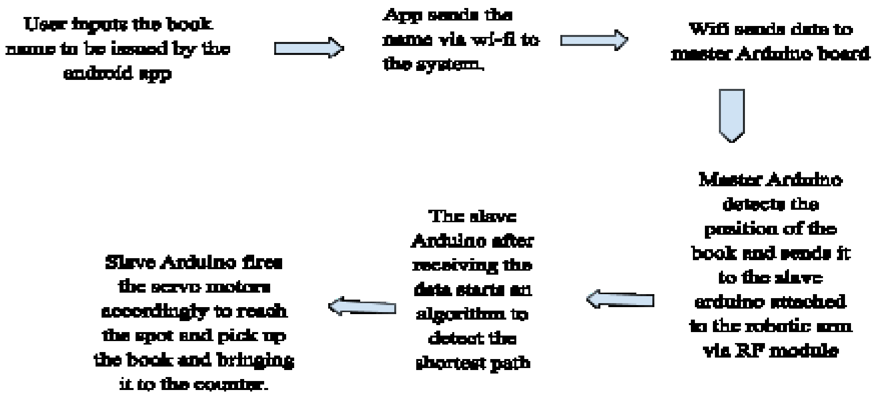


Figure 2.2.1: Project Block Diagram

2 – A Mobile Robot for Autonomous Book Retrieval

In this research paper, a robot had been created using a laptop to monitor the subsystems. The robot consists of two hands' system to catch the books. However, the robot is not precise as it uses barcodes to collect books. The robot has customized cross-platform compatible software, written in C++ language, forms the intelligence of the robot while an onboard laptop computer, running on Linux platform of Red Hat distribution version 9.0, provides the processing and execution of intelligence of the robot. A Graphical User Interface (GUI) developed using

Borland's Kylix R acts as the human–machine interface allowing a user to submit a desired book call number. DC motors are used for locomotion. Infrared line tracer sensors are used for the navigation of the robot with lines embedded in the floors marking the routes and giving the robot a priori knowledge of the paths it has to take. Barcoding technology is used to provide digital signatures to library books, encoding its call number into the barcode. The employment of an innovative and practical system, a typical and inexpensive barcode reader, provides the technology with which books are implemented. Current ongoing work includes improvising the performance of the robot and incorporation of features such as circumnavigation around obstacles, localization and automatic self-recharging that will solve the problems of deployment in a realistic environment. [2]

The project that we are going to design and build has two advantages than the one above, which are: 1- fixing the system's power problem by implementing a continuous charging method. 2- Developing more subsystems to manage the library database using RFID, which is a more precise method of locating books.

3 - Mobile Robot Capable of Crossing Floors for Library Management

As the world revolves on autonomous in order to reduce the labor and materials. Beijing Institute of Technology published an article under a title of (Mobile Robot Capable of Crossing Floors for Library Management). This article discusses about a team trying to develop an intelligent robot that can deliver books to readers, shelve books in its correct position and can interact with readers without labor participation. This paper has the same goals as our project in addition to the design of the robot. They use looks similar in many aspects with slightly different features. [3]

4 - Warehouse Robots for Fulfillment Centers

In industrial nowadays, industrial robots have become essential in industries. More than that, the delivery services have been shifted, and medical assistants and other applications include autonomous robots. For example, Behemoth retailer Amazon put retail automation technology in the spotlight with the 2012 purchase of Kiva Systems, a company which developed warehouse robots and related technologies, and which was acquired for $775 million. Now known as Amazon Robotics, this system and design reflects the same idea and purpose of our project providing the delivery and returning objects and ensuring the possibility of interacting with a human is relatively low. The system is using the mapping for robot navigation. The Amazon Robotics system has proven its efficiency and so we try to build a similar system but this time in library. We are searching and developing a robot to install it in library instead of a warehouse like Amazon. [4]

2.3 Comparative Study

As can be seen from the comparison study Table 2.3.1, our project offers the advantage of all the previous projects combined in terms of simplicity, efficiency, low price, returning to home charge, multitasking and automatic monitoring.

Table 2.3.1: Projects Comparison Study.

Projects	Simplicity	Efficiency	Low Price	Return to Home Charge	Multitask	Automatic Monitoring
Microcontroller Based Robotic Arm Development for Library Management System	√	X	√	X	X	X
Amazon Robotics system	X	√	X	X	√	X
A Mobile Robot for Autonomous Book Retrieval	X	√	X	X	√	X
Our Project	√	√	√	√	√	√

3. System Design

3.1 Design Constraints

3.1.1 Design Constraints: Engineering Standards

The engineering constraint we faced in the designing period of the project is the size of the robot, which will ensure that the robot will not be unbalanced at 2 m height while carrying 2 kg book, and will meet the correct design size to move freely in the library without affecting the motor specifications.

We solved this issue using CAD software for planning and anticipating the worst-case scenarios.

3.1.2 Design Constraints: Economic

The Economic constraint we faced in our project is minimizing the costs, especially for the pulley that will raise the arm to the shelves. We found out that we could use actuators, which are much cheaper than pulleys to do the lifting process and with better accuracy.

3.2 Design Methodology

The controller (Arduino) is supplied power by the batteries, which in turn powers the DC motors of the chassis. The controller receives orders from the tablet, and orders the robot to go to the specific shelve to either pick or return the book using Pixycam for movement and RFID tags for detecting books. The tablet should coordinate the library systems between users and the robot. Figure 3.2.1 shows the block diagram of the project architecture.

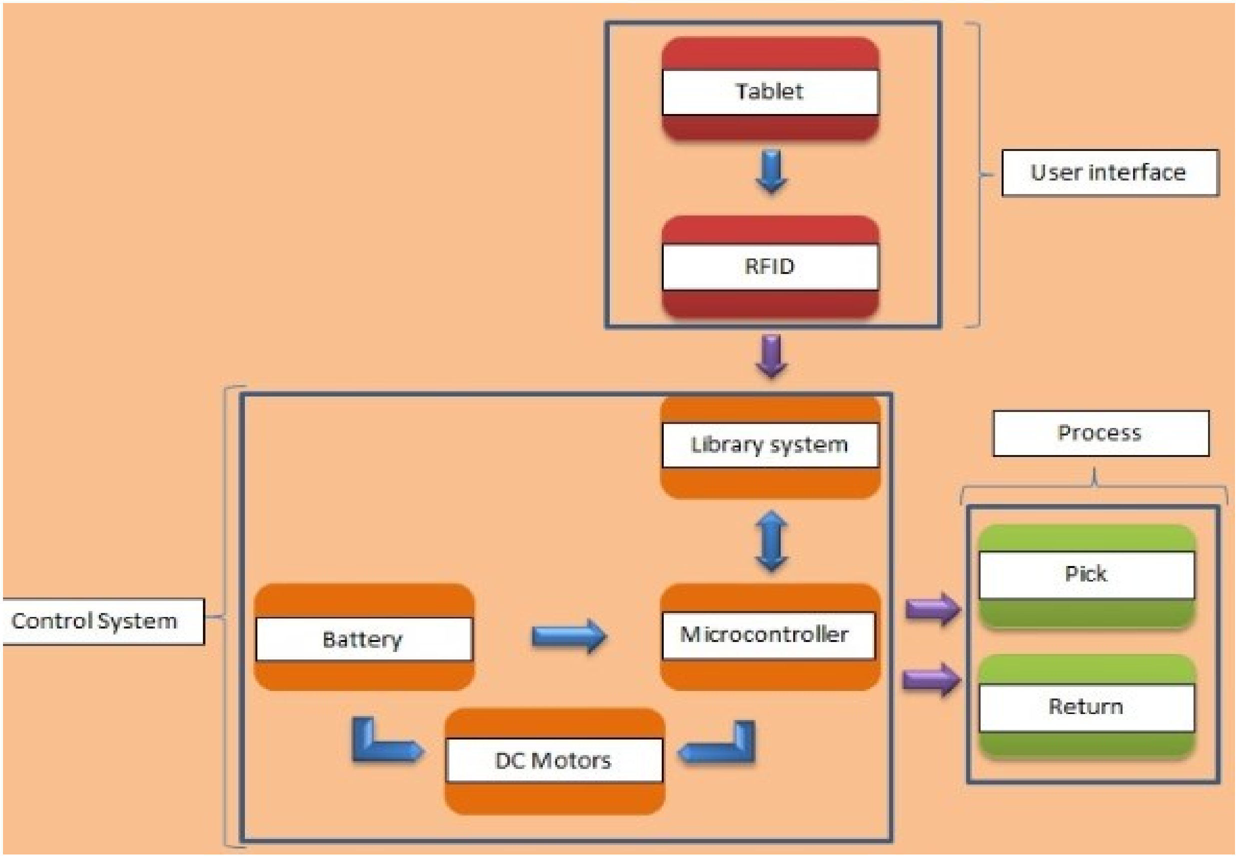


Figure 3.2.1: The Project Architecture Block Diagram

3.3 Product Subsystems and Components

3.3.1 Product Subsystem 1: Movement Motors

Our robot will have to move from point A to point B in order to perform either a picking or a deposing task at a reasonable speed of 0.3 m/s. For that, there is only one option that can perform this task, which is the using of motors.

Several types of motors can be used to meet our requirements such as:

- DC motor.
- Step motor.
- Servomotor.
- AC motor.

However, one type of motor has to be used. Thus, through our research, we decided that the DC motors are much suitable to our project compared to the others. The reason is that: DC motor has continuous displacement while stepper motor's motion has an angle incremental and servomotors are not continuous. Indeed, as our power supply is a battery then AC motor is not needed.

3.3.2 Product Subsystem 2: Moving Mechanism Using Pixy2 Cam

To reach a specific destination, we chose the using of Pixy2 Cam to dictate the path that the robot must follow. Some other options can be used to perform the same task such as:

- IR sensors
- GPS module

However, Pixy2 Cam is much more reliable, multitasking, accurate and easy to use compared to any of the listed above. For those reasons, we decided to go with Pixy2 Cam.

3.3.3 Product Subsystem 3: RFID Book Recognition

This subsystem is the most important one for any robot made for library. Therefore, we took sometimes to research the most reliable and effective in the local market within our budget. To meet those requirements, we decided to purchase a passive RFID which is the RC522. This passive RFID has a ranch from 0 to 5 cm which satisfies our requirement. In addition, it is the cheapest passive RFID available in the local market, and its reader can read PMU students' IDs. This means that we do not need to buy extra RFID tags in order to stick them on the books. Different alternatives of this one could have been:

- Active RFID: This kind of RFID are not at all suitable for our project on the way that they are expensive, have long detection range, and consume more power.
- Other types of passive RFID: Those RFID are more expensive and have a long detection range which is not recommended for our subsystem.

3.3.4 Product Subsystem 4: Lifting Mechanism

To reach different shelves level, a need of an elevator system needed to be designed. Thus, we decided to choose to incorporate a 3D printer design model for our system of elevator. The reason behind that is that 3D printers are precise. Such feature is one of the two important requirements for our lifting subsystem. For those reasons, we bought stepper motors and lead screws made specifically for 3D printers in order to install them in our lifting mechanism. Other options were on the table for example a pulley elevator model. This model demands too many mechanical equipment to be involved. In addition, we noticed that this model will cause a stability problem.

3.3.5 Product Subsystem 5: Handling Book Mechanism

Another important subsystem is to choose our handling book mechanism. When you do literature search, you will notice uncountable ways of grabbing a book. So, with a high variety of grabbing mechanisms, a decision to choose the right one was really difficult. At the end, we decided go with an actuator which will be equipped at its tip with an electromagnet. So far, we did not yet implement this system. However, we already have the equipment. Choosing those two equipment for a handling mechanism appears to be our best solution in order to avoid dealing with mechanical equipment like grabbing mechanism in which it might not be able to hold the required object. In addition, we intended to avoid more mechanical issues in which a lot of calibrations are demanded. For those reasons, an actuator is the best choice. Noticing that with an electromagnet, we are sure that the grabbing part will not be a problem.

3.3.6 Product Subsystem 6: 3D printed Core Material Reinforced with Carbon Fiber

This subsystem is the structure of our robot. At first, we decided to build a workshop structure made of fibber as it is light. However, when added just the DC motors and the battery, we surprisingly discovered that it became much heavier than expected. To solve this problem, we decided to go with a 3D printed core as a predesigned and virtual testing were required before printing. This will make sure to avoid any surprises and to avoid wasting more money for an unreliable structure with poor design.

3.4 Implementation

From last semester, we started building our base to make it move from point A to point B. Thus, we decided to focus in two subsystems, which are described previously. To meet our specifications, we chose the DC motor with regards to some calculations as shown in Figure 3.4.1. Those calculations find the DC motors that can meet our specifications (weight and speed) provided which we can find them in the market. In order to ease our calculations, we chose primarily four different diameters of tires (d=0.065 m, d= 0.13 m, d=0.085 m, and d=0.1m) available in the market. With any of those diameters, we can reach a speed of 0.3 m/s if we choose any angular speed greater than 44.08 rpm. To solve the weight problem, we just calculated the torque of each one then chose the DC motor that had a speed greater or equal to 44.08 rpm, and had a torque bigger enough to bear the weight. The fourth case is the measurement we chose to go with as we found a DC motor that meets the entire requirement.

```
%%First Case%%
d1 = 0.065; %diameter in m
r1 = d1/2;
g = 9.81;
m = 22.2; %wight in kg
v = 0.3; %speed in m/s
F1 = m*g;
T1 = r1*F1
rpm1 = (60*v)/(2*pi*r1)
EW1 = T1/4
%%Second Case%%
d2 = 0.13; %diameter in m
r2 = d2/2;
g = 9.81;
m = 22.2; %wight in kg
v = 0.3; %speed in m/s
F2 = m*g;
T2 = r2*F2
rpm2 = (60*v)/(2*pi*r2)
EW2 = T2/4
%%Third Case%%

d3 = 0.085; %diameter in m
r3 = d3/2;
g = 9.81;
m = 22.2; %wight in kg
v = 0.3; %speed in m/s
F3 = m*g;
T3 = r3*F3
rpm3 = (60*v)/(2*pi*r3)
EW3 = T3/4

%%Fourth Case%%

d4 = 0.1; %diameter in m
r4 = d4/2;
g = 9.81;
m = 22.2; %wight in kg
v = 0.3; %speed in m/s
F4 = m*g;
T4 = r4*F4
rpm4 = (60*v)/(2*pi*r4)
EW4 = T4/4
```

```
T1 = 7.0779
rpm1 = 88.1474
EW1 = 1.7695

T2 = 14.1558
rpm2 = 44.0737
EW2 = 3.5390

T3 = 9.2557
rpm3 = 67.4068
EW3 = 2.3139

T4 = 10.8891
rpm4 = 57.2958
EW4 = 2.7223
```

Figure 3.4.1: Matlab Motor Calculations.

As described in the subsystem section, to follow the line, we chose Pixy2 Cam. The reasons we chose this are described in details in 3.3.2 subsystem section. Before implementing the two subsystems together, we tested the four DC motors by powering them up with low power supply to check if they are working. Then, we did a test for the pixy2 Cam by using Arduino mega 2560.

Finally, we built a temporary chassis through a workshop, and integrated the four DC motors and the pixy for a demo as shown in the Figure 3.4.2.

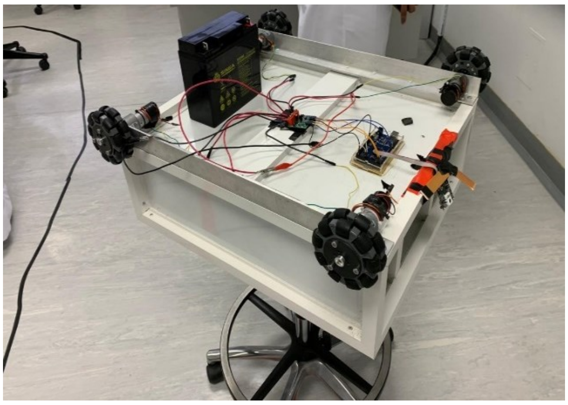


Figure 3.4.2 (A): Project Shape from Bottom.

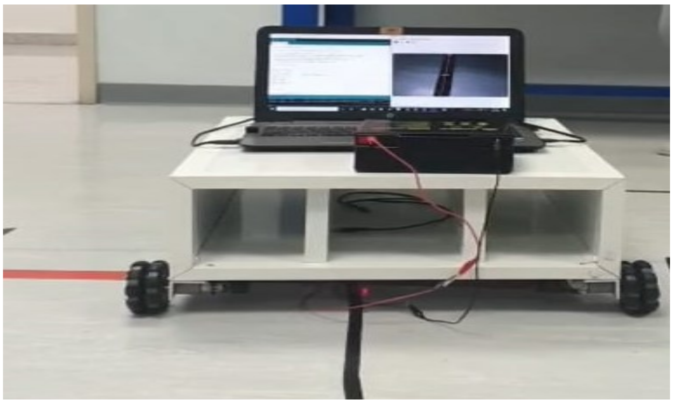


Figure 3.4.2 (B): Project Shape while Moving.

In Assessment III course, we started with the user interface system which is our third system. In order for our robot to pick or return a book, it needs first to detect the book. Thus, we chose to use the RC522 passive RFID to detect the preregistered book of the library. The choice of choosing this specific RFID is explained in 3.3.3 subsystem section. Figure 3.4.3 shows the RC522 passive RFID reader and tags.

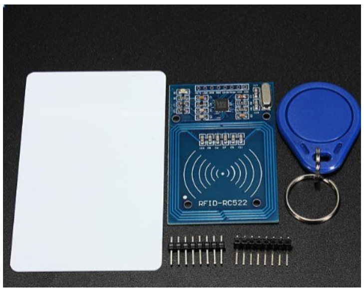
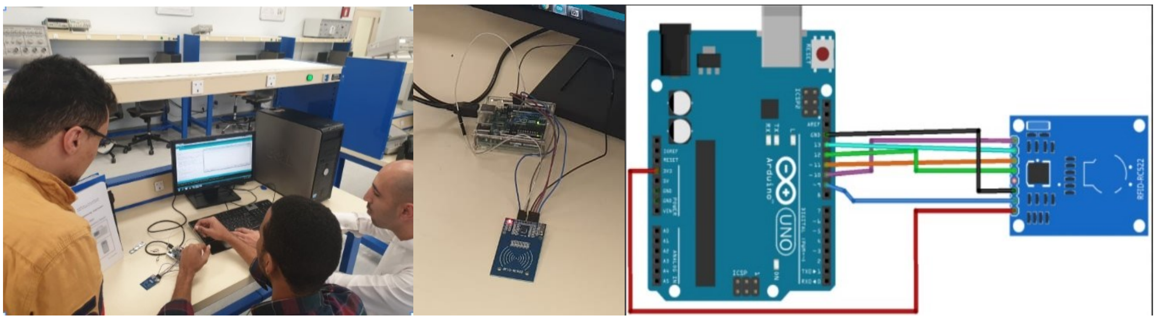


Figure 3.4.3: RFID Kit.

The RFID subsystem was integrated and implemented by the group members as shown in Figure 3.4.4.



Figures 3.4.4 A, B, C: RFID Subsystem Implementation.

After implementing the circuit shown above, we tested it in the lab and the system works as expected.

At this time, we are working in parallel to finish three subsytems at the same time. Those subsytems are lifting mechanism, handling mechanism, and the body structure. The components chosen to built and implement those subsytems are already in our hands. The preferred chosen components are explained in 3.3.4 , 3.3.5 , and 3.3.6 in subsytem section. Figure 3.4.5 shows the progress of the lifting mechanism.

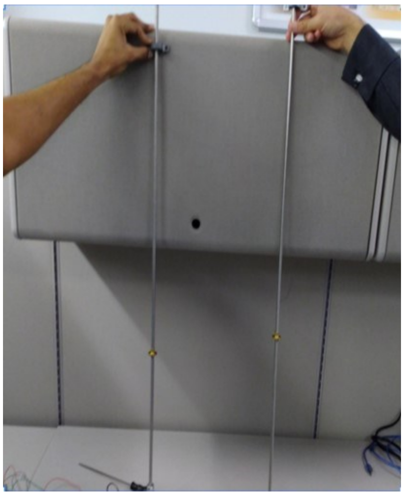


Figure 3.4.5: Lifting Mechnism Testing.

4. System Testing and Analysis

4.1 Subsystem 1: Movement Motors

Objective: The objective of the testing is to verify that the calculations of motor specification meet the actual requirement. In another words, verifying that the motors can handle the weight of the robot, and move as desired without mistakes.

Instruments: The main instruments used were the digital multimeter (DMM) and the DC Power Supply, which were used inside PMU circuit lab to test if the motor current consumption is within the range limit. Such tests indicates that the motors can handle the weight of the robot.

Results: The results of the motors current consumption under 15 kg of weight was 5 A, indicating that it still can hold more weight estimated to be as 7.5 A.

4.2 Subsystem 2: Moving Mechanism using Pixy2 Cam

Objective: The aim of the testing is to verify that the PixyCam accuracy meets our requirements. The testing mainly verifies that the PixyCam vectors following can be used to move smoothly into the required shelve, and recognize which path to follow in case of intersections of different shelves line vectors.

Instruments: The main instruments used were tapes and the chassis with Pixy2 Cam. A test was done to move the robot from point to point using Pixy2 Cam moving mechanism where intersection between two lines was available.

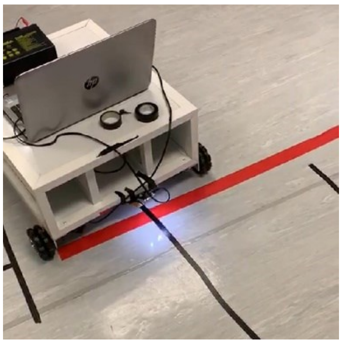


Figure 4.2.1: Intersection Recognition Movement.

Results: The results of the test showed positive outcomes as the robot moved smoothly without facing difficulties in recognizing which vector line to follow. Moreover, the movement is smooth and did not affect the chassis balance.

4.3 Subsystem 3: RFID Book Recognition

Objective: The aim of the test was to verify that the RFID reader can recognize the books using tags placed at the end of the book from a specific distance. This distance between the book and the reader in the test was simulated to be similar to the distance between the robot's elevator plate and the reader.

Instruments: The main instruments used were the books occupied with RFID tags, and the RFID readers, which are shown in figures 3.4.4 of previous chapter

Results: The results of the test showed that the RFID tags and readers used where well suited for our project as they immediately and correctly recognized the books which were passed through them.

Figures 4.3.1 & 4.3.2 show the process of book scanning and recognition during testing.

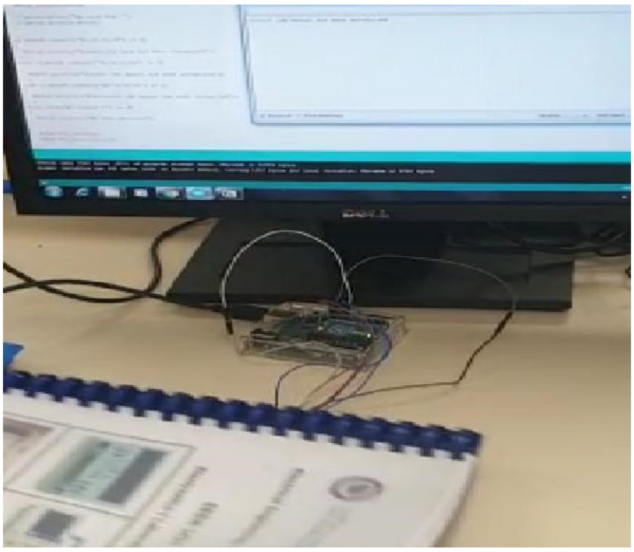


Figure 4.3.1: RFID Book Scanning.

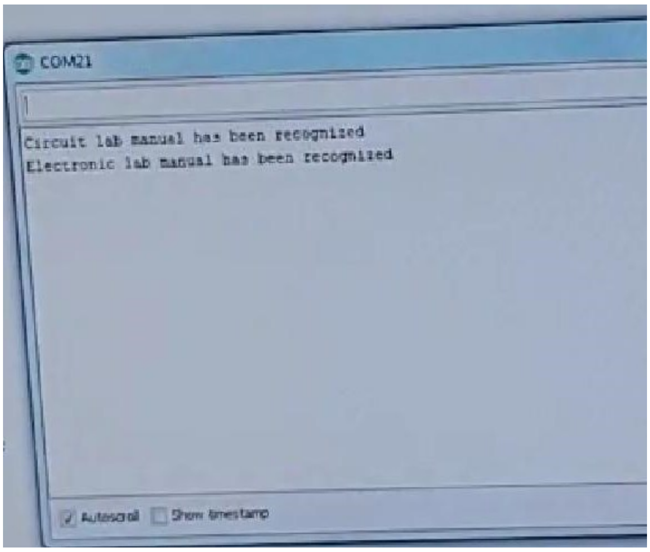


Figure 4.3.2: RFID Book Recognition.

4.4 Subsystem 4: Lifting Mechanism

Objective: The aim of the test is to test how effective and fast can the elevator plate rise up or go down, and test the code accuracy in going to the desired level.

Instruments: The main instruments used were the stepper motors and the lead screws attached to them.

Results: The results of the test showed that the stepper motor required to be replaced as their performance was weak, and the sound was noisy. As we replaced the stepper motors, we noted that there was minimal noise, and the elevator was fast in rising and going down due to the special lead screws with maximum slope.

4.5 Subsystem 5: Handling Book Mechanism

Test was not concluded as the subsystem coding is still in progress.

4.6 Subsystem 6: 3D Printed Core Material Reinforced with Carbon Fiber

The robot is undergoing simulations in design before starting the printing. The robot will be ready to be tested after printing and assembling takes place.

4.7 Overall Results, Analysis and Discussion

Overall, we can say that the results of the testing were successful after several trials. However, it is worth to note that many obstacles may arise in case if we combine all the subsystems together within the robot. Therefore, we estimate the progress percentage for each subsystem with respect to the combined overall robot as follows.

Table 4.1: Subsystems progress percentage.

Subsystem	Progress Percentage
Movement Motors	100%
Moving Mechanism using Pixy2 Cam	100%
RFID Book Recognition	90%
Lifting Mechanism	60%
Handling Book Mechanism	20%
3D printed core material reinforced with fibber carbon	50%

5. Project Management

5.1 Project Plan

At this point we are working on:

- The frame of the lifting subsystem mechanism.
- Building the lifting plate and assembling it within the frame.
- Testing accuracy of reaching the desired shelve.
- Starting Subsystem 5 (Handling Subsystem).
- Printing core structure.

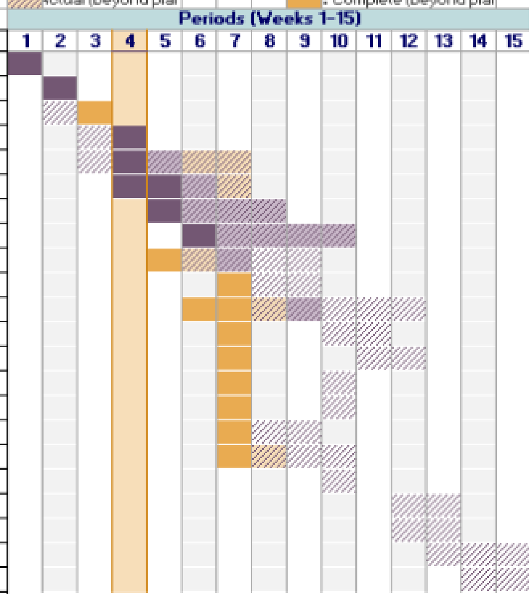
Title: Autonomous Mobile Robotic Library System | Advisor: Dr. Sadiq Alhuwaidi | Design II (ASSE 3) | Spring 2020

Project PLAN & Progress
ProgRpt No. 3
Plan updated (Date): March 6, 2020
Instructor: Dr. Sadiq Alhuwaidi

Fahad Alsubaie 201500450 (FAH)
Ibrahim Alharbi 201101220 (IBR)
Aws Aleryani 201503139 (AWS)
Nassim Mahamoud 201503130 (NAS)
Ahmed Alhafidh 200901602 (AHM)

Period Highlight: 4 | Plan | Actual | Actual (beyond plan) | % Complete (beyond plan)

ACTIVITY	PLAN START	PLAN DURATION	Assigned To	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Creating Second Semester Plan	1	1	IBR & FAH	1	1	100%
Design subsystem 3 (lifting capabilty)	2	1	NAS & AWS	2	1	100%
Ordering Lifting Components	2	1	AHM	3	1	100%
Testing the Stepper Motors	3	2	IBR & NAS	4	1	55%
Testin the lifting mechanism Separately	3	3	FAH & AWS	4	4	10%
Coupling the lead screws with the stepper motors	4	3	AHM & IBR	4	4	70%
Design the plate	5	4	NAS & AWS	5	4	25%
Implementing the whole lifting part on the robot	6	5	ALL	6	5	15%
Coding and testing the Lifting Mechanism	7	3	AS, IBR , AV	5	3	10%
Prepare midterm presentation	8	2	AHM & FAH	7	1	95%
Design subsystem 4 (Handling Mechanism)	9	4	ALL	6	4	50%
Ordering Hand	10	2	IBR & FAH	7	1	100%
Design Subsystem 5 (Book recognition using RFID	11	2	ALL	7	1	100%
RFIDs Components	10	1	IBR &FAH	7	1	100%
Testing RFID	10	1	AS, AWS, F	7	1	100%
Design Subsystem 6 (Obstacle Avoidance using U	8	2	FaH & AHM	7	1	100%
Subsytem 6 components and testing	9	2	AWS & NAS	7	2	55%
Assembling and testing Handling concept	10	1	AWS & NAS	0	0	0%
Coding, combining and testing all subsystems toge	12	2	ALL	0	0	0%
Prepapre final presentation and final report	12	2	ALL	0	0	0%
Prepare project demo	13	3	ALL	0	0	0%
Submit Rpt/PPT/Brochure ...	14	2	ALL	0	0	0%



Progress Details:
We finished subsytem 5.
We are waiting for the stepper motors to come in order to finish our subsytem 3.
We ordered the components for our subsytem 4
Subsytems 6 & 3 in progress

Issues (delay ...):
Due to the current crisis in the world especially in China, our orders are delayed.

Figure 5.1: Progress Report

5.2 Contribution of Team Members

Table 5.1: Team Contribution towards Project.

Task	Nassim	Aws	Ibrahim	Ahmed	Fahd	Task Total
Search & acquire components	25%	20%	20%	20%	15%	100%
Design Subsystems	20%	25%	20%	20%	15%	100%
Test Subsystems	20%	20%	20%	20%	20%	100%
Write Reports & Presentations	25%	25%	15%	15%	20%	100%

5.3 Project Execution Monitoring

Activities:

- Biweekly meetings with advisor.
- Weekly team meetings on Thursdays.
- Testing each subsystem within week of components arrival.

5.4 Challenges and Decision Making

So far, the following below are the challenges we faced:

Fall 2019-2020:

- Designing and balancing the robot.
  Solution: Consulted 3D engineers to take care of the designing and simulation of the robot before performing the 3D printing.

- Motor Driver damaged which needed to be replaced as fast as possible.
  Solution: Replaced the driver as fast as possible with different kinds of driver. Therefore, adapting and changing the code logic to meet the criteria.

Spring 2020:

- Designing the lifting plate and frame is risky before insuring lead screws correct positioning through testing. Therefore, lifting mechanism had to be delayed.
- Stepper motor manufacturing malfunctioning, which caused a delay to order the new stepper motors from USA.
- Components arrival delay from China because of Corona Virus.
- City Lockdown because of Corona Virus caused a huge delay in our progress.

5.5 Project Bill of Materials and Budget

Table 5.2: Project Budget

No.	Description	Quantity	Unit Cost (SR)	Total Cost (SR)
1	Microcontroller IRs, Ultrasonic	1	150	150
2	Motor driver	4	50	200
3	RFID Reader	1	50	50
4	3D Printer	1	3000	3000
5	Battery	1	170	170
6	DC motor & Servo Motors	4/4	262.5/100	1450
7	Pixy2 cam	1	200	200
8	Actuator & Lead Screws	2	336/80	992
			Total	6212

6. Future Work and Expected Final Prototype/Results

6.1 Future Work

For the future work plans, we came across situations in which there was better techniques to implement our project applications.

These plans include the following:

- PixyCam technology to detect shelves.
- Complex path technique to move and detect paths across the library.
- Silent hydraulic or pressure-based lifting mechanism using expert mechanic advices.
- Implementing self-recharge efficient, precise and safe techniques using complex coding approaches.

6.2 Expected Prototype

In our project, most of the subsystems that compose our expected prototype were explained and shown in Sections 3 and 4. Thus, we would expect our product to have an appearance as shown in Figure 6.2.2.

Our initial design was implemented using CAD software as shown in figure 6.2.1, but after consulting experts, we ended up with the design in Figure 6.2.2.

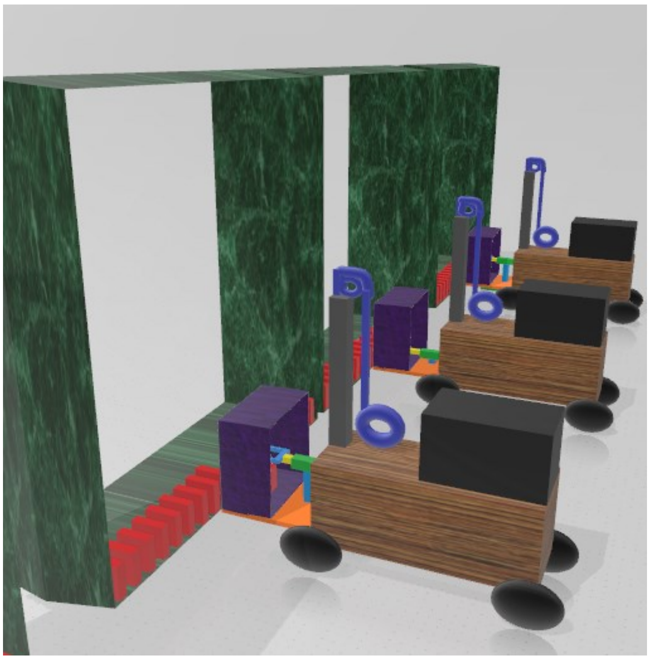


Figure 6.2.1: Initial CAD Prototype.

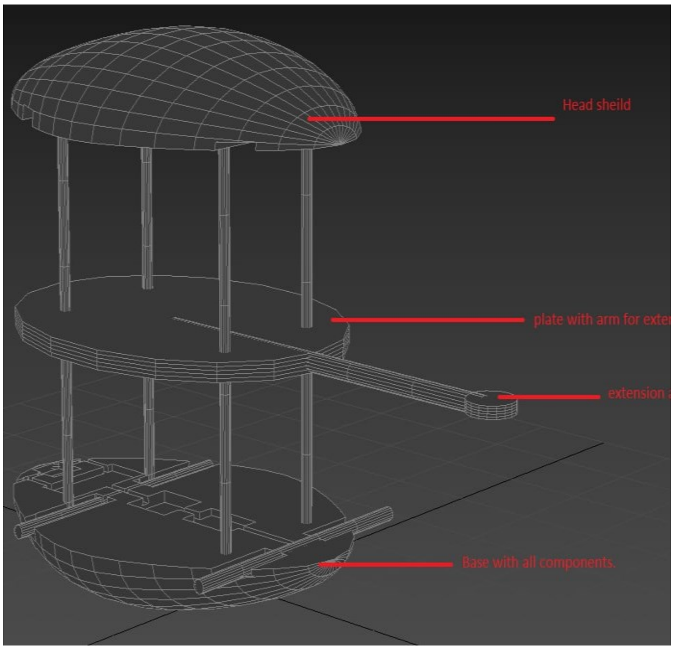


Figure 6.2.2: Expected Prototype.

7. Limitations and Challenges

7.1 Limitations

- Covering a small library due to battery limitations.
- Bottom level shelves cannot be accessed.
- The robot is only able to follow a specific path for each shelve.
- The robot has limitation in performing two tasks at a time. i.e. cannot bring and return books at the same time.

7.2 Challenges

Fall 2019-2020:

In this term, we faced those following challenges in which we were able to overcome perfectly.
- Designing and balancing the robot.
- Motor driver damaged which needed to be replaced as fast as possible.

Spring 2020:

In this term, some of the challenges mentioned were overcome. However, the last one was not in our control.
- Designing the lifting plate and frame is risky before insuring lead screws correct positioning through testing.
- Stepper motor manufacturing malfunctioning.
- Components arrival delay from China.
- Due to COVID 19, no progress done after March 7, 2020.

8. Conclusion

In conclusion, our project to design a robot that handles library services has finished in term of design and testing in every aspect.However, due to the unfortunate COVID-19 spreading around the globe and the Saudi government new procedures to block this virus from spreading, our work was restricted on the final stage of producing our final product. We could proudly say that our team has achieved high progress in the Design Methdology semeste (Fall 2019 - 2020), we made the Implementation, system testing and analysis of 2 subsystems, and we managed to implement the remaining 3 systems in the Assesment III semeste (Spring 2020). However we aim to combine all the subsystems as soon as the pandemic is over, and we are able to present the demo of our work.

9. References

- [1] https://arxiv.org/ftp/arxiv/papers/1812/1812.11316.pdf
- [2] https://link.springer.com/content/pdf/10.1007%2F978-3-540-73424-6.pdf
- [3] ieeexplore.ieee.org/document/8816274
- [4] https://www.nanalyze.com/2018/08/warehouse-robots-retail-automation/

Appendix A: Progress Reports

The progress reports of the Design Methodology Course and the Assessment 3 Course are shown in the figures:

1- Design Methodology Course

Autonomous mobile robotic library system							Design Methodology & PM	Fall 2019/2020
Fahad Alsubaie 201500450								
Ibrahim Alharbi 201101220							Project PLAN	
Aws Aleryani 201503139								
Nassim Mahamoud 201503130							Course Instructor: Dr. Samir El-Nakla & Project Advisor: Sadiq Alhuwaidi	
Ahmed Alhafidh 200901602							Period Highlight: 1	Plan / Actual
ACTIVITY	PLAN START	PLAN DURATION	Assigned To	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE	Actual (beyond plan)	% Complete (beyond plan)
							Periods (Weeks 1-15)	
Form teams	1	1	ALL	1	1	100%		
Select topic & advisor	2	1	ALL	2	2	100%		
Write proposal and plan	2	1	ALL	2	2	100%		
Background research & PPT1	3	2	ALL	3	2	100%		
Search and acquire components	3	3	ALL	3	4	60%		
Design subsystem 1 (Base)	4	3	ALL	4	4	50%		
Test subsystem 1	5	4	ALL	14	2	10%		
Learning Center of Gravity	5	2	ALL	5	3	80%		
Motor Calculations	5	2	AWS,NAS	5	3	100%		
Design Building Learning	6	5	IBR, FAH	6	5	60%		
Chassy Building	6	1	ALL	13	2	30%		
Prepare midterm Presentation	7	3	ALL	7	3	100%		
Wheel Calculations	8	2	AHM	9	2	80%		
3D Design Learning	9	3	AWS	9	3	100%		
Design subsystem 2 (LANE Follwing)	9	4	ALL	15	1	0%		
Test subsystem 2	10	2	ALL	16	1	0%		
Understanding LANE Following	10	1	ALL	11	4	10%		
Sensors Assembling	11	2	FAH, IBR	12	2	0%		
Learning Arduino Programming	12	2	AWS,NAS	12	2	35%		
Chassy Powering	12	2	AHM	13	2	15%		
Prepapre final report	12	3	ALL	12	3	0%		
Prepapre final presentation	13	2	ALL	13	2	0%		

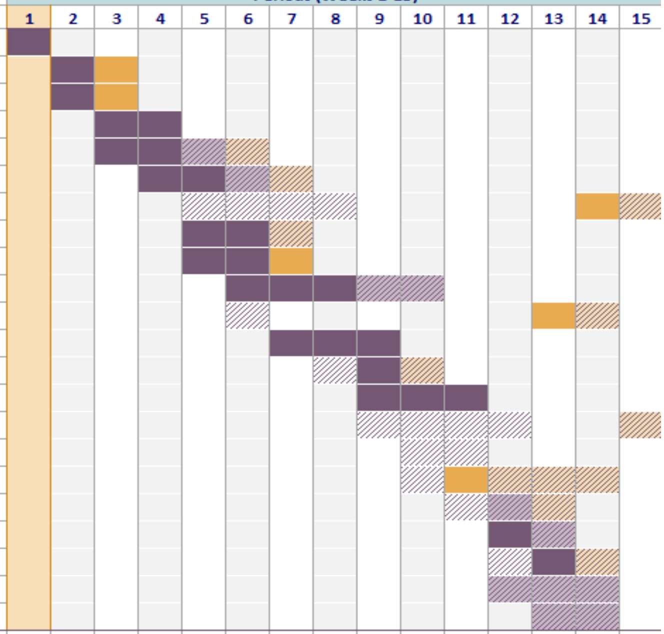
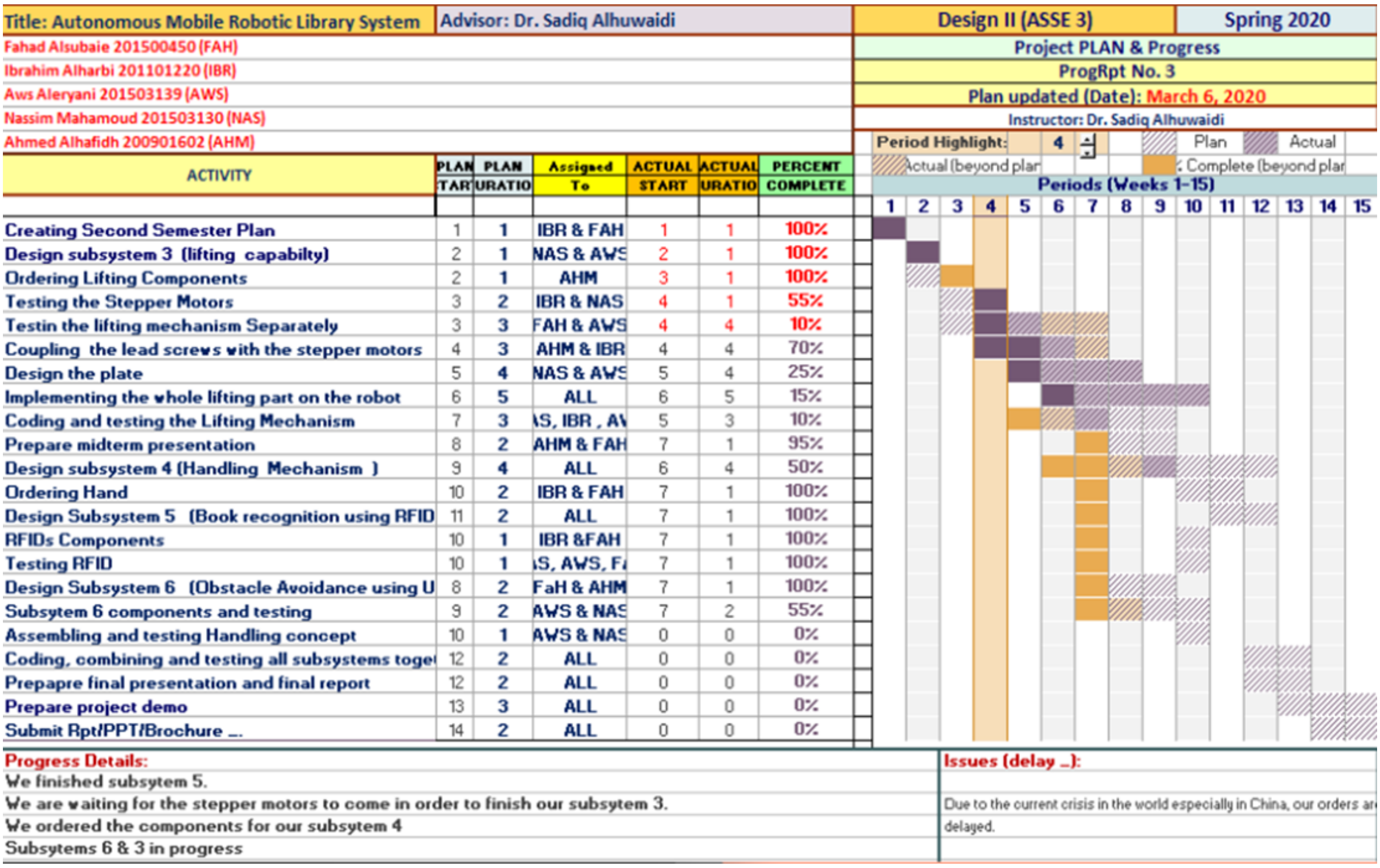


Figure 1: Design Methodology & Project Management Progress Report

2- Assessment III Course



Title: Autonomous Mobile Robotic Library System | Advisor: Dr. Sadiq Alhuwaidi | Design II (ASSE 3) | Spring 2020

Fahad Alsubaie 201500450 (FAH)
Ibrahim Alharbi 201101220 (IBR)
Aws Aleryani 201503139 (AWS)
Nassim Mahamoud 201503130 (NAS)
Ahmed Alhafidh 200901602 (AHM)

Project PLAN & Progress
ProgRpt No. 3
Plan updated (Date): March 6, 2020
Instructor: Dr. Sadiq Alhuwaidi
Period Highlight: 4 | Plan | Actual | Actual (beyond plan) | % Complete (beyond plan)

Periods (Weeks 1-15)

ACTIVITY	PLAN START	PLAN DURATION	Assigned To	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Creating Second Semester Plan	1	1	IBR & FAH	1	1	100%
Design subsystem 3 (lifting capabilty)	2	1	NAS & AWS	2	1	100%
Ordering Lifting Components	2	1	AHM	3	1	100%
Testing the Stepper Motors	3	2	IBR & NAS	4	1	55%
Testin the lifting mechanism Separately	3	3	FAH & AWS	4	4	10%
Coupling the lead screws with the stepper motors	4	3	AHM & IBR	4	4	70%
Design the plate	5	4	NAS & AWS	5	4	25%
Implementing the whole lifting part on the robot	6	5	ALL	6	5	15%
Coding and testing the Lifting Mechanism	7	3	AS, IBR , AW	5	3	10%
Prepare midterm presentation	8	2	AHM & FAH	7	1	95%
Design subsystem 4 (Handling Mechanism)	9	4	ALL	6	4	50%
Ordering Hand	10	2	IBR & FAH	7	1	100%
Design Subsystem 5 (Book recognition using RFID	11	2	ALL	7	1	100%
RFIDs Components	10	1	IBR &FAH	7	1	100%
Testing RFID	10	1	AS, AWS, FA	7	1	100%
Design Subsystem 6 (Obstacle Avoidance using U	8	2	FaH & AHM	7	1	100%
Subsytem 6 components and testing	9	2	AWS & NAS	7	2	55%
Assembling and testing Handling concept	10	1	AWS & NAS	0	0	0%
Coding, combining and testing all subsystems toge	12	2	ALL	0	0	0%
Prepapre final presentation and final report	12	2	ALL	0	0	0%
Prepare project demo	13	3	ALL	0	0	0%
Submit Rpt/PPT/Brochure ...	14	2	ALL	0	0	0%

Progress Details:
We finished subsytem 5.
We are waiting for the stepper motors to come in order to finish our subsytem 3.
We ordered the components for our subsytem 4
Subsytems 6 & 3 in progress

Issues (delay ..):
Due to the current crisis in the world especially in China, our orders are delayed.

Figure 2: Assessment III Progress Report

Appendix B: Bill of Materials

The following table shows the cost of equipment and services that we have bought and used for our robot.

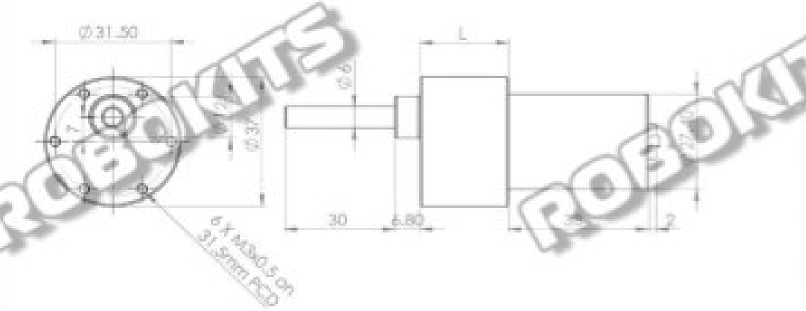
Table 1: Project Budget:

No.	Description	Quantity	Unit Cost (SR)	Total Cost (SR)
1	Microcontroller IRs, Ultrasonic	1	150	150
2	Motor driver	4	50	200
3	RFID Reader	1	50	50
4	3D Printer	1	3000	3000
5	Battery	1	170	170
6	DC motor & Servo Motors	4/4	262.5/100	1450
7	Pixy2 cam	1	200	200
8	Actuator & Lead Screws	2	336/80	992
			Total	6212

Appendix C: Datasheets

Model Number RKI-1147	Johnson Motor High Torque DC Geared 12V 60RPM - Grade A

Mechanical Drawing (Dimensions are in mm)



10RPM	L=28.5mm
30RPM	L=28.5mm
60RPM	L=26.5mm
100RPM	L=26.5mm
200RPM	L=25mm
300RPM	L=25mm
600RPM	L=22mm
900RPM	L=22mm

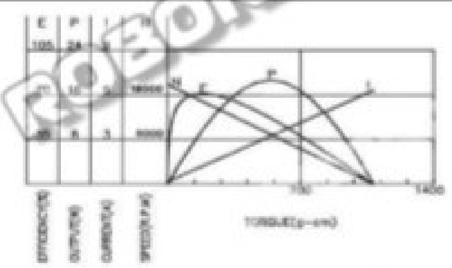
Technical specifications of Motor

Motor Type	High Torque Geared Motor
Operating voltage	12V DC
Motor Speed at Output Shaft (RPM)	60RPM
Stall Torque(Kgcm)	38Kgcm (30Kgcm is Gearbox Breaking Torque)
Rated Torque(Kgcm)	15.2Kgcm
Gear Ratio	1 : 270
Weight	200gm

Base Motor Specifications

Nominal voltage(V)	12V
No-Load Speed (RPM)	18000
Rated Torque (gcm)	151
No-load current (A)	0.8A
Load current(A)	up to 7.5 A(Max)

RHINO



Base Motor Characteristic Graph

Figure 4: DC Motor Datasheet

WWW.ROBOKITS.CO.IN

EASY TO USE, VERSATILE ROBOTICS KITS

Add raw power and simple connectivity to your robotics applications with this 18V compatible 20A capable Dual DC motor driver. It is ideal for application where two motors require up to 20 Amperes of current during startup and during normal operations.

It comes with a simple TTL/CMOS based interface that can connect directly to the IOs of an MCU. It has a breaking feature that can guarantee immediate halt on the shaft of motors in most high power applications and also includes protection circuitry to avoid any electrical fluctuations affecting the normal operation of an MCU.

This Package Includes

- Upto 18V compatible 20A Dual DC motor driver circuit

Features

- Simple connectivity to IO pins of any MCU.
- Compatible with motors rated up to 18V
- Can easily deliver 20A of current during normal operation
- Braking feature included without affecting the performance of an MCU

Applications

- Simple DC motor applications that require forward and backward driving of motors
- DC motor applications requiring speed control via PWM input
- Halting or braking a DC motor during operation

Electrical Charateristics

Input Voltage: 7V minimum to 18V maximum

Continuous Current (< 10seconds) ~ 20A

Continuous Current (< 60seconds) ~ 10A

Continuous Current (> 60seconds) ~ 5A (without heat sink on MOSFETS)

Absolute Maximum Peak Current ~ 50A

No short circuit protection on output of the driver

Connector Information

http://www.robokits.co.in
http://www.robokitsworld.com

Page 2

Dual DC Motor Driver 20A [RKI-1341]

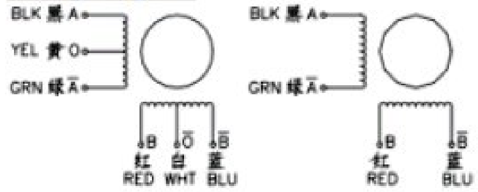
Figure 5: Motor Driver Datasheet

MotionKing (China) Motor Industry Co., Ltd.

17HS4401

2 Phase Hybrid Stepper Motor
17HS series-Size 42mm(1.8 degree)

Wiring Diagram:

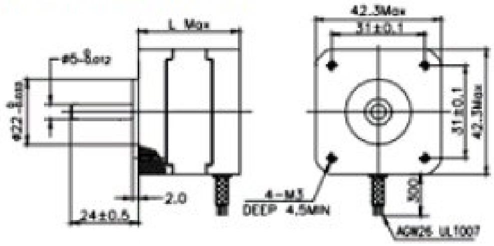


Electrical Specifications:

Series Model	Step Angle (deg)	Motor Length (mm)	Rated Current (A)	Phase Resistance (ohm)	Phase Inductance (mH)	Holding Torque (N.cm Min)	Detent Torque (N.cm Max)	Rotor Inertia ($g.cm^2$)	Lead Wire (No.)	Motor Weight (g)
17HS2408	1.8	28	0.6	8	10	12	1.6	34	4	150
17HS3401	1.8	34	1.3	2.4	2.8	28	1.6	34	4	220
17HS3410	1.8	34	1.7	1.2	1.8	28	1.6	34	4	220
17HS3430	1.8	34	0.4	30	35	28	1.6	34	4	220
17HS3630	1.8	34	0.4	30	18	21	1.6	34	6	220
17HS3616	1.8	34	0.16	75	40	14	1.6	34	6	220
17HS4401	1.8	40	1.7	1.5	2.8	40	2.2	54	4	280
17HS4402	1.8	40	1.3	2.5	5.0	40	2.2	54	4	280
17HS4602	1.8	40	1.2	3.2	2.8	28	2.2	54	6	280
17HS4630	1.8	40	0.4	30	28	28	2.2	54	6	280
17HS8401	1.8	48	1.7	1.8	3.2	52	2.6	68	4	350
17HS8402	1.8	48	1.3	3.2	5.5	52	2.6	68	4	350
17HS8403	1.8	48	2.3	1.2	1.6	46	2.6	68	4	350
17HS8630	1.8	48	0.4	30	38	34	2.6	68	6	350

*Note: We can manufacture products according to customer's requirements.

Dimensions: unit=mm



Motor Length:

Model	Length
17HS2XXX	28 mm
17HS3XXX	34 mm
16HS4XXX	40 mm
16HS8XXX	48 mm

Figure 6: Stepper Motor Datasheet

RobotShop www.robotshop.com

La robotique à votre service! - Robotics at your service!

Schematic: arduino-mega2560-schematic.pdf

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

Figure 4: Arduino Mega 2560 Datasheet

Operating current :13-26mA/DC 3.3V

Idle current :10-13mA/DC 3.3V

Sleep current: <80uA

Peak current: <30mA

Operating Frequency: 13.56MHz

:Supported card types

Product physical characteristics: size: 40mm×60mm

Environmental Operating temperature: -20-80 degrees Celsius

Environmental Storage Temperature: -40-85 degrees Celsius

Relative humidity: relative humidity 5% -95%

Module interfaces SPI Parameter

Data transfer rate: maximum 10Mbit/s

Figure 5: RFID Datasheet

Specifications

Load (LBS)		No Load Current (A)				Full Load Current (A)				Speed (inch/sec)	
Dynamic	Static	12VDC	24VDC	36VDC	48VDC	12VDC	24VDC	36VDC	48VDC	No Load	Full Load
35	75	1.0	0.5	0.3	0.3	5.0	2.5	1.7	1.3	2.00	1.38
50	100	1.0	0.5	0.3	0.3	5.0	2.5	1.7	1.3	1.14	0.83
75	150	1.0	0.5	0.3	0.3	5.0	2.5	1.7	1.3	0.95	0.70
110	220	1.0	0.5	0.3	0.3	5.0	2.5	1.7	1.3	0.79	0.59
150	300	1.0	0.5	0.3	0.3	5.0	2.5	1.7	1.3	0.37	0.28

Stroke	1" to 40"
Limit Switch	Internal - Non-Adjustable
Limit Switch Feedback	Customizable
Screw Type	ACME Screw
Motor Type	Brushed or Brushless DC Motor
Connector Type	See Page 5
Wire Length	40" (customizable)
Housing Material	6062 Aluminum Alloy
Rod Material	Aluminum Alloy/Stainless Steel (customizable)
Gear Material	Polyformaldehyde (35 lbs only)/Powder Metallurgy Steel Alloy
Color (Shaft)	Silver
Color (Motor End)	Silver
Noise	<45dB
Duty Cycle	25% (5 minutes on, 15 minutes off)
Operational Temperature	-25°C to 65°C (-13°F to 149°F)
Protection Class	IP54 (IP65 customizable)
Feedback Options	See PA-14P Feedback Linear Actuator
Certifications	CE/RoHS
Mounting Brackets	See Page 6
Mounting Ends	Customizable

2

Figure 6: Actuator Datasheet

Description:

T8 1000mm Stainless Steel Lead Screw with Shaft Coupling and Mounting Support

Specification:

Lead screw	
Material	stainless steel
Diameter	8mm
Pitch	2mm
Lead	8
Total Length	1000mm
Screw nut	
Material	brass
Diameter	8mm
Pitch	2mm
Lead	8
Mounted ball bearing	
Material	zinc alloy
Bore diameter	5mm
Hole diameter	8mm
Total length	55mm
Height	29mm
Shaft coupling	
Material	aluminium
Shaft	6.35 x 8mm
Length	25mm
Diameter	18mm
Quantity	1 set (1 screw + 1 Screw nut + 2 Mounted ball bearing + 1 shaft coupling)

Figure7: Lead Screw Datasheet

The links for the above datasheets:

- https://cdn.shopify.com/s/files/1/0061/7735/7891/files/Mini_Linear_Actuator_PA-14.pdf?v=8946735387817795481
- http://www.datasheetcafe.com/17hs4401-datasheet-stepper-motor/
- https://robokits.download/downloads/Dual_DC_Motor_Driver_20A.pdf
- https://robokits.download/downloads/RKI-1147.pdf
- https://electrinostore.com/aeYRbn
- https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf

Appendix D: Program Codes

The Arduino codes used are divided into three categories:

- Chassis Movement Code
- Lifting Plate code
- RFID Code

Each category codes are shown in the following pages, and it is worth mentioning that the final integrated code can only be created and combined after implementing and testing the whole systems together to add the missing parts.

1- Chassis Movement Code

```
#include <Pixy2.h>
#include <PIDLoop.h>

#define X_CENTER         (pixy.frameWidth/2)
#define dir_l 42
#define dir_r 44
#define pwm_l 6
#define pwm_r 5
#define brk_l 22
#define brk_r 24

Pixy2 pixy;
PIDLoop headingLoop(5000, 0, 0, false);

void setup()
{
	//motors.setLeftSpeed(0);
	//motors.setRightSpeed(0);
	pinMode (dir_l,OUTPUT);
	pinMode (dir_r,OUTPUT);
	pinMode (pwm_l,OUTPUT);
	pinMode (pwm_r,OUTPUT);
	pinMode (brk_l,OUTPUT);
	pinMode (brk_r,OUTPUT);
	//digitalWrite(brk_l, HIGH);
	// digitalWrite(brk_r, HIGH);
	pixy.init();
	// Turn on both lamps, upper and lower for maximum exposure
	pixy.setLamp(1, 1);
	// change to the line_tracking program.  Note, changeProg can use partial strings, so 
for example,
	// you can change to the line_tracking program by calling changeProg("line") instead of 
the whole
	// string changeProg("line_tracking")
	pixy.changeProg("line");

	// look straight and down
	pixy.setServos(500, 1000);
	Serial.begin(9600);
}


void loop()
{
	//Serial.print("Starting...\n");


	int8_t res;
	int32_t error;
	int left, right;
	char buf[96];

	// Get latest data from Pixy, including main vector, new intersections and new 
barcodes.
	res = pixy.line.getMainFeatures();

	// If error or nothing detected, stop motors
	if (res<=0)
	{
		analogWrite(pwm_l, 0);
		analogWrite(pwm_r, 0);
```

```
            //motors.setLeftSpeed(0);
            //motors.setRightSpeed(0);

            Serial.print("stop ");
            Serial.println(res);
            return;
    }

    // We found the vector...
    if (res&LINE_VECTOR)
    {
            // Calculate heading error with respect to m_x1, which is the far-end of the
vector,
            // the part of the vector we're heading toward.
            error = (int32_t)pixy.line.vectors->m_x1 - (int32_t)X_CENTER;

            pixy.line.vectors->print();

            // Perform PID calcs on heading error.
            headingLoop.update(error);

            // separate heading into left and right wheel velocities.
            left = headingLoop.m_command;
            right = -headingLoop.m_command;

            // If vector is heading away from us (arrow pointing up), things are normal.
            if (pixy.line.vectors->m_y0 > pixy.line.vectors->m_y1)
            {
                    // ... but slow down a little if intersection is present, so we don't
miss it.
                    if (pixy.line.vectors->m_flags&LINE_FLAG_INTERSECTION_PRESENT)
                    {
                            left += 100;
                            right += 100;

                    }
                    else // otherwise, pedal to the metal!
                    {
                            left += 150;
                            right += 150;

                    }
            }
            else  // If the vector is pointing down, or down-ish, we need to go
backwards to follow.
            {
                    left -= 100;
                    right -= 100;

            }
            //motors.setLeftSpeed(pwm_l);
            //motors.setRightSpeed(pwm_r);
            analogWrite(pwm_l, left);
            analogWrite(pwm_r, right);

    }
}
```

2- Lifting Plate Code Test

```
byte dir1= 4, dir2= 5;
byte brk1= 6, brk2=7;
byte pwm1= 2, pwm2= 3;

#define x        2500                 // smaller values may make the motor produce more speed
and less torque
#define y        2000
#define stepsPerRevolution 200         // you can the number of steps required to make a
complete revolution in the data sheet of your motor

void stepperClockwise(byte spd);
void stepperAnticlockwise(byte spd);


void setup() {
pinMode(dir1, OUTPUT);pinMode(dir2, OUTPUT);
pinMode(brk1, OUTPUT);pinMode(brk2, OUTPUT);
pinMode(pwm1, OUTPUT);pinMode(pwm2, OUTPUT);

Serial.begin(9600);
}


void loop() {
char direction;
direction= Serial.read();

stepperAnticlockwise(200);
delay(1000);

stepperClockwise(200);
delay(1000);
}

void stepperAnticlockwise(byte spd){
    for (int i = 0; i < (stepsPerRevolution/4) ; i++) {
    analogWrite(pwm1,spd);
    analogWrite(pwm2,spd);


    digitalWrite(dir2,LOW);
    digitalWrite(dir1,LOW);
    delayMicroseconds (x);

    digitalWrite(dir2,HIGH);
    digitalWrite(dir1,LOW);
    delayMicroseconds (x);

    digitalWrite(dir2,HIGH);
    digitalWrite(dir1,HIGH);
    delayMicroseconds (x);

    digitalWrite(dir2,LOW);
    digitalWrite(dir1,HIGH);
    delayMicroseconds (x);

     analogWrite(pwm1,0);
    analogWrite(pwm2,0);

  }
 }
```

```
void stepperClockwise(byte spd){

    for (int i = 0; i < (stepsPerRevolution/4) ; i++) {
    analogWrite(pwm1,spd);
    analogWrite(pwm2,spd);


    digitalWrite(dir1,HIGH);
    digitalWrite(dir2,HIGH);
    delayMicroseconds (x);

    digitalWrite(dir1,LOW);
    digitalWrite(dir2,HIGH);
    delayMicroseconds (x);

    digitalWrite(dir1,LOW);
    digitalWrite(dir2,LOW);
    delayMicroseconds (x);

    digitalWrite(dir1,HIGH);
    digitalWrite(dir2,LOW);
    delayMicroseconds (x);

analogWrite(pwm1,0);
    analogWrite(pwm2,0);
      }

}
```

3- RFID Code

```
#include "SPI.h"
#include "MFRC522.h"
#define SS_PIN 10
#define RST_PIN 9
#define SP_PIN 8
MFRC522 rfid(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;

void setup() {
  Serial.begin(9600);
  SPI.begin();
  rfid.PCD_Init();
}

void loop() {
  if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial())
    return;
  MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);

  if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&
    piccType != MFRC522::PICC_TYPE_MIFARE_1K &&
    piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
    Serial.println(F("Your tag is not of type MIFARE Classic."));
    return;
}

  String strID = "";
  for (byte i = 0; i < 4; i++) {
    strID +=
    (rfid.uid.uidByte[i] < 0x10 ? "0" : "") +
    String(rfid.uid.uidByte[i], HEX) +
    (i!=3 ? ":" : "");
  }
  strID.toUpperCase();

if (strID.indexOf("B3:C8:1C:06") >= 0)
{
  Serial.println("Probability book has been recognized");
  }
else if(strID.indexOf("7B:16:21:39") >= 0)
{
  Serial.println("Circuit lab manual has been recognized");
  }
else if(strID.indexOf("AB:C6:20:39") >= 0)
{
  Serial.println("Electronic lab manual has been recognized");
  }
else if(strID.indexOf("") >= 0)
{
  Serial.println("No book detected");
  }

  rfid.PICC_HaltA();
  rfid.PCD_StopCrypto1();
}
```

Appendix E: Operation Manual

This robot was made to help students and instructors in receiving and returning books.

The project works in 2 modes: The first one is bringing the books to the home station, and the second one is returning books to the shelf before returning to home station.

1. **Bringing the books:**

   - Determining the type of the book using four digits code typed in keypad.
   - The keypad sends the signal to the Arduino.
   - The Arduino sends the signal to the motor and the pixy cam to follow the designated line to the right shelve.
   - The Arduino sends the signal to the stepper motor to move up/down to make the plate in the right position.
   - The Arduino sends the signal to the actuator (Hand) to move inside the shelf, and the electromagnet attached to the actuator end will pull the book.
   - The robot will move to the base station by following back the same line.

2. **Returning the books:**

   - Students / instructors will but the book in the robot plate and the RFID will recognize the code and send it to the Arduino.
   - The Arduino sends the signal to the motor and the pixy cam to follow the line to the right shelve.
   - The Arduino sends the signal to the stepper motor to move up/down to make the plate in the right shelf position.
   - The Arduino will send the signal to the extension to push the book inside the shelf in the right location.
   - The robot will move back to the base station by following the same line.