UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	AUNI ZAHIDAH BINTI MOHD RASOL
Date of Birth	:	
Title	:	GRIPPER CONFIGURATION AND CONTROL DESIGN ON OMNIDIRECTIONAL MOBILE MANIPULATOR ARM
Academic Session	:	SEMESTER I 2021/2022

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

(Supervisor's Signature)

ASSOC. PROF. IR. DR. ADDIE IRAWAN BIN HASHIM

New IC/Passport Number
Date: 18 FEBRUARY 2022

Name of Supervisor
Date: 18/2/2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of Engineering in Electrical and Electronics Technology

____

(Supervisor's Signature)

Full Name : ASSOC. PROF. IR. DR. ADDIE IRAWAN BIN HASHIM

Position : ASSOCIATE PROFESSOR

Date : 18/2/2022

____

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : AUNI ZAHIDAH BINTI MOHD RASOL

ID Number : TF18021

Date : 18 FEBRUARY 2022

GRIPPER CONFIGURATION AND CONTROL DESIGN ON OMNIDIRECTIONAL MOBILE MANIPULATOR ARM

AUNI ZAHIDAH BINTI MOHD RASOL

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Electrical Engineering Technology (Power & Machine) With Honours

Faculty of Electrical & Electronics Engineering Technology

UNIVERSITI MALAYSIA PAHANG

REBRUARY 2022

ACKNOWLEDGEMENTS

First of all, I would like to thank Allah for giving me the strength in succeeding this thesis. Everything is under His control.

This project would not have been successful without the help of my supervisor Dr Addie Irawan, helping me from beginning to the end. It was a great opportunity to study with him

The prayers and support from my father, family, friends and people around really helped me in completing this project. This thesis would not have been possible if it weren't for the help of my friends. Thank you to everyone who has helped me along the way, both personally and indirectly.

ABSTRAK

Lengan robot yang diartikulasikan terdiri daripada pautan yang menyambungkan satu siri sendi berputar kepada hujung efektor. Projek ini akan meningkatkan fungsi lengan robotik 3-DoF. Sistem Manipulator Omnidirectional (OMSys) sekarang telah berjaya dibina dan dilaksanakan oleh bekas pelajar yang bertanggungjawab bagi projek ini. Walau bagaimanapun, ia tidak lengkap. Isu pertama yang perlu ditangani adalah menjadi lengan dengan fungsi angkat-dan-letak, pencengkam telah ditambahkan pada hujung effector. Kedua ialah, untuk mengawal hujung efektor bersama-sama dengan mecanum dan lengan 3DoF. Oleh itu, kod semasa yang diubah suai untuk lengan, mecanum dan pencengkam, bersama-sama dengan alat kawalan jauh, adalah penyelesaian untuk mengawal pergerakan lengan dan kenderaan. Hasil yang dijangkakan adalah hasil yang diramalkan. Terdapat dua hasil jangkaan yang telah diramalkan untuk projek ini. Pertama, hasil yang dijangkakan untuk projek ini ialah mewujudkan konfigurasi pencengkam pada manipulator lengan OMSys untuk operasi angkat-dan-letak. Kedua, untuk mereka bentuk sistem kawalan untuk hujung efektor pencengkam. Keputusan ini boleh dicapai dengan menggabungkan kedua-dua mecanum dan lengan manipulatornya. Seterusnya, semua motor disambungkan ke papan Arduino yang mengandungi kod pengaturcaraan yang boleh mengawal kedua-dua mecanum dan manipulatornya dari jauh.

ABSTRACT

An articulated robotic arm is made up of links that connect a series of rotating joints to an end effector. This project will enhance the function of 3-DoF robotic arm. The present Omnidirectional Manipulator System (OMSys) has been successfully constructed and implemented by the former students who were in charge of this project. However, it is not incomplete. The first issue to address is to be an arm with pick-and-place function, a grasper/gripper was added at the end effector. Second is, to control the end effector together with mecanum and 3DoF arm. Therefore, modified current code for arm, mecanum, and grasper, along with Remote configuration, is the solution for controlling the arm and vehicle movement. The Expected result are forecasted result. There are two expected outcomes that have been predicted for this project. First, the expected result for this project is establish the grasper/gripper configuration on the OMSys arm manipulator for pick-and-place operation. Second, to design a control system for the end effetor of the grasper/gripper. This result can be achieved by combine both mecanum and its manipulator arm. Next, all the motor are connected to Arduino board containing programming codes that can remotely control both mecanum and its manipulator.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

OMSys	Omnidirectional Manipulator System

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Project Background

In this very modern age, the use of robots over human labor in daily life to perform accurate and precise work is becoming more preferable, thanks to its better performance and minimal risks. An articulated robotic arm consists of links that connect a particular number of rotary joints in series to an end effector. There is three parts that will be improved in this project which is the 3-DoF Robotic Arm, the Monitoring System and the mecanum Wheel.

The subject of mobile robotics is much like an interdisciplinary course made up of electrical, mechanical, and electronic engineering, as well as computer science. Founded in the 1980s, omnidirectional robot platforms are a common solution for mobile manipulation. Prior to the 20th century, many industrial worker skills were replaced by mobile manipulators, because of the changes in the manufacturing industry. Workers using automated tools such as autonomous and adaptable mobile manipulators are required in every contemporary workplace because of the extensive automation and production in industrialized nations. Mobile manipulators intended to assist humans in manual activities, such as wire stripping and closing open circuits, were among the earliest types of mobile manipulator. Additionally, mobile manipulators were often used for inspecting, cleaning, and transporting objects. DoF stands for the number of degrees of freedom of the robotic arm. Usually, servo motors are used to drive the joints. Figure 1 below shows a SolidWorks model of the considered 3 DoF robotic arm.

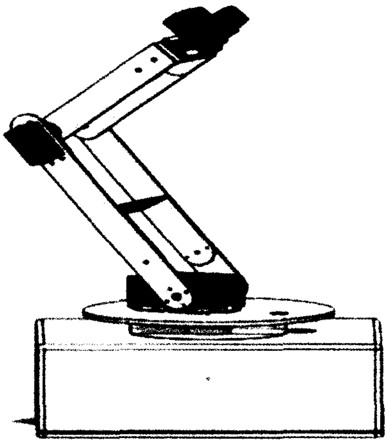

Figure 1: A SolidWorks model of the considered 3 DOF robotic arm [1]

Omnidirectional vehicles have 3 degrees of freedom (DOF) on the ground. They can achieve longitudinal motion, lateral motion, center-point steering motion, and any composite motion of above three, so they are suitable for highly maneuverable, narrow or accurate positioning occasions. In the omnidirectional wheel, the wheel velocity can be divided into the components in the active direction and in the passive direction. The active component is directed along the axis of the roller in contact with the ground, while the passive one is rotating, at least one roller, and maximum of two rollers are in contact with the ground. However, theoretically, just one point of the roller is in contact with the ground. The area of this surface traverses the roller from one side to another, depending on the sense of wheel rotation. This makes the direction of the traction force be done by the traversing sense of the contact of the roller with the surface.

Robots have long borne the potential in everything, especially in human work. Human work does sometimes have limits. For example, entering a place where humans cannot enter which have narrow space such as underbuilding, sewer or tunnel. This monitoring system is built to solve this problem. The circuit is separate from the main circuit which makes it easier to attach and detach without changing the original system of the OMA. The electronic devices are connected to the transmitter and receiver. This can result in a wireless system and can bring to outdoor.

1.2 Problem statement

The key issue from the previous project is that there is no main purpose for the three-degree-of-freedom robotic arm. It only moves in direction right, left, up, and down Additionally, the prior project had a difficulty with controlling the arm and vehicle motions, which was resolved in this project. It is necessary to modify the controller's setup so that it can be used to control the arm, mecanum wheel, and grasper.

1.3 Objective

The goal of this research is to construct a three-degree-of-freedom arm for pick-and-place operations. As a result, there are three objectives in this project:

i. To establish the grasper/gripper configuration on the OMSys arm manipulator

ii. To design a control system for end-effector of the grasper/gripper

1.4 Research Scope

Scope for this project in develop the 3DoF robot arm for pick-and-place is selection of grasper, the weight of the target is equal to the selection of the servo motor, modified current code of the arm and mecanum wheel and last but not least remote configuration.

1.5 Thesis Organisation

Thesis organization is as shown below:

Chapter 1

Chapter 1 is about the introduction of the project and explain the purpose of this SDP. It covers project background of the current 3DoF robot arm, problem statement and the objectives.

Chapter 2

Literature review is included in these chapter. It was all about the development of the 3DoF robot arm for pick-and-place operation.

Chapter 3

This chapter will discuss about the methodology that used to complete this project. It includes the flowchart of the OMSys, selection of Gripper, code for gripper, arm and mecanum and remote configuration.

Chapter 4

Chapter 4 discuss about the result of the placement of the gripper and control system for gripper.

Chapter 5

In this chapter, project result will be discussed here to confirm whether this project objective has been achieved or not together with the recommendation.

CHAPTER 2

Literature Review

2.1 Introduction

Literature reviews is one of the research methods which is defined as an evaluative report of statistics and facts found in the literature related to scope of study. The review was consisting of describe, summarize, and clarify this literature. In description, the author names, sources and all the details will be stated. Moreover, for the summary, it will only be stated what had been understood and what is related to this project. For clarification part, it can be defined as the conclusion and decision made after the respective literature review had been done.

A literature review is more than the search for information as it gathers the methods and solutions and come out a best solution. The reference sources can be come from research papers, websites, books, articles, journals and magazines. The processes of doing literature reviews are included read, evaluate, analyses and cite but it must be related to the field of research.

2.2 Mobile Manipulator Robot

The robotic arm design is influenced by a number of variables such as the geometry of the manipulator, dynamics involved, the structural characteristics of the linkage system (manipulator), and the actuator characteristics. The hand of the robot carries an end-effector, which might be any tool or gripping mechanism. The design of the end-effector is crucial to the satisfactory performance of the robotic arm and hence its design is dependent on the shape, size, and weight of the object to be gripped. [1]

Industrial manipulators have been employed in a wide variety of applications, mostly in the manufacturing sector, although their utility is restricted to situations requiring a very big manipulator. industrial space, including aerospace manufacturing, shipbuilding, and manufacture of wind turbines. Manipulation of mobile devices is a means of circumventing these constraints and therefore qualifies as a critical not just for the manufacturing sector, but also for robots for professional service. Additionally, mobile manipulators find use in virtual or physical labs for education and research.

According to Christof Röhrig, Daniel Heß, Frank Künemund, 2017 [2], stated that on the basis of commercially accessible mobile platforms and robotic arms, research institutions have created their own mobile manipulators. Cody, developed at the Georgia Institute of Technology, is made up of two MEKA Robotics arms and a mecanum wheeled Segway base. Cody was created primarily for the purpose of doing research on service robots in the health care sector. Another example in this field is Cornell University's POLAR (PersOnaL Assistant Robot), which is comprised of a 7-DOF Barrett arm placed on a Segway Omni base. Additionally, Willow Garage's PR2 and Fraunhofer IPA's Care-O-bot 4 are notable examples in this area. In contrast to the professional and domestic service sectors, industrial mobile manipulators work in more organized settings. They do, however, need a greater degree of operational efficiency, such as speed, precision, and durability, in order to be appropriate for industrial applications. Figure 2 show the example of mobile manipulator, OmniMan consists of a Universal Robots UR5 robotic arm mounted on mecanum wheeled mobile platform from MIAG Fahrzeugbau GmbH (Braunschweig, Germany).

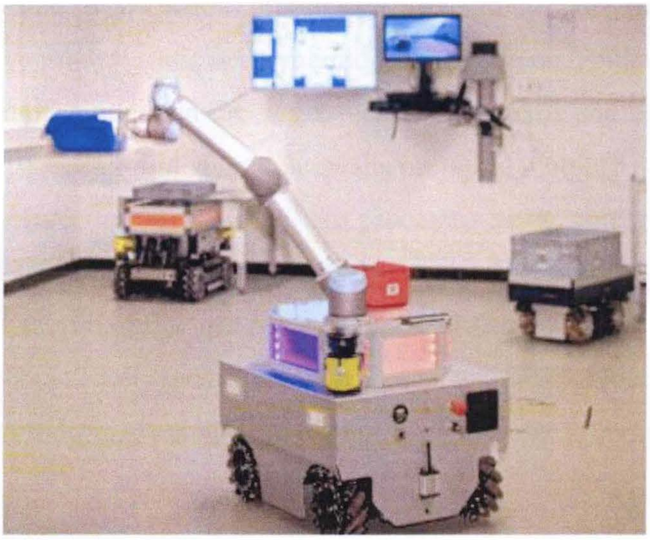

Figure 2: Mobile manipulator OmniMan: UR5 robotic arm on a Mecanum wheeled mobile platform [2]

Ching-Chih Tsai1, Ching-Zu Kuo, Chun-Chieh Chan, Xiao-Ci Wang, 2013[3] said in proposal paper that the proposed navigation system is composed of three modules: odometry, nonsingular terminal sliding-mode (NTSM) dynamic motion controller, and global path planner, which have been implemented using the SoPC technology. The odometry is constructed by using a numerical method and a kinematic model of the robot, in order to keep track of the current position and orientation of the robot over short distances. A nonsingular terminal sliding-mode dynamic controller is well derived to achieve simultaneous point stabilization and trajectory tracking. A hybrid PSO (particle swarm optimization)-RGA (real-coded genetic algorithm) algorithm is proposed to find an optimal path between a starting and ending point in a given grid environment.

As shown in Figure 3, the MWOMR with an anthropomorphous dual-arm manipulator is equipped with the 3D sensing optical module, MESA 4000, mounted on a two-degrees-of-freedom robotic head module, a small size personal computer with CPU of Intel Core2 Duo E6850 3.0GHz, an embedded PIV controller and a navigation controller using two Stratix II edition of Altera Nios II development kits, and two seven degrees-of-freedom arms.

The suggested MWOMR's mobile manipulations with the twin arms are categorized into two modes: static and dynamic. The mobile manipulators must be halted in the static operation mode, and the target locations of items and both hands must be made accessible to a visual-guided module on the proposed mobile robot. As a result, the pictures from the Kinect or MESA 4000 are processed using object identification and localization algorithms that are suitable for the pictures. Additionally, the posture of the target objects and the end effectors are calculated in Cartesian space using SURF to enable rapid and robust object identification and localization. In dynamic mode, the dual arms and movable platform are controlled concurrently in order to complete a cooperative job.

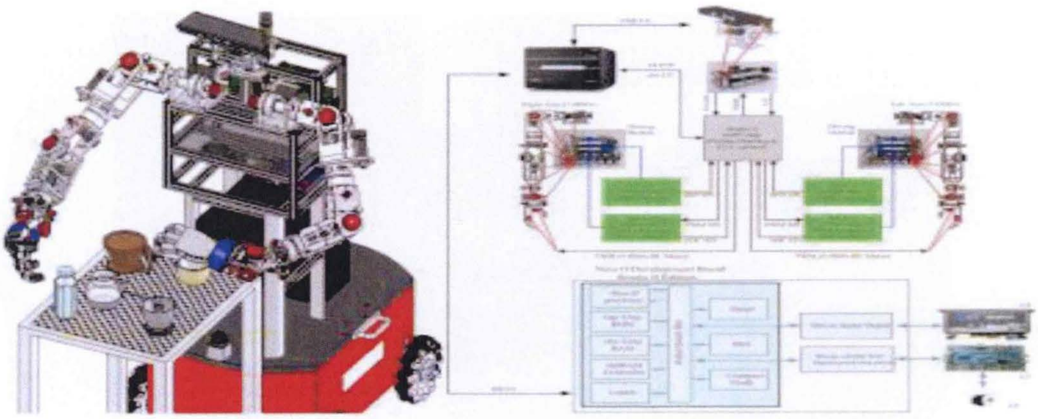

Figure 3: Structure of the MWOMR with 7DOF dual arms [3]

Universal Robots' UR5 is a six-degree-of-freedom flexible and lightweight robotic arm in figure 4[4]. The arm can carry a payload of 5 kilograms and weighs 18.4 kilograms without the end effector. The control box adds 15kg to the weight (light versions weigh between 4.3 and 4.7kg). Interaction with the controller is available through the PolyScope teach pendant programmed, URScript, or the C-API. More significantly, it supports ROS drivers, making it compatible with CoppeliaSim's online interface. UR5 is composed of seven segments that are linked by six revolute joints. CoppeliaSim solves forward and inverse kinematics using a damped least squares pseudoinverse method applicable to any kind of mechanism.

A base, joints, segments, a tip, and a target comprise the kinematic chain; together, they create the hierarchical scene for the robotic arm. The two dummies UR5 ikTarget and UR5 ikTip are connected kinematically and are used to mimic the arm in inverse kinematic mode. The mathematical model of UR5 is extensively described in the literature and is based on the Denavit-Hartenberg parameters or on the exponential product. The arm's CAD model is accessible in CoppeliaSim's collection of non-mobile robots, courtesy of Universal Robots. Simultaneously, simpler geometries are utilized to decrease computing complexity, like with the Mecanum wheels, which are reduced to spheres. The RG2 gripper is used as the end effector. It has a maximum stroke of 110 mm and a payload of 2 kg. Additionally, this gripper is included in CoppeliaSim's library, along with the actuation script.

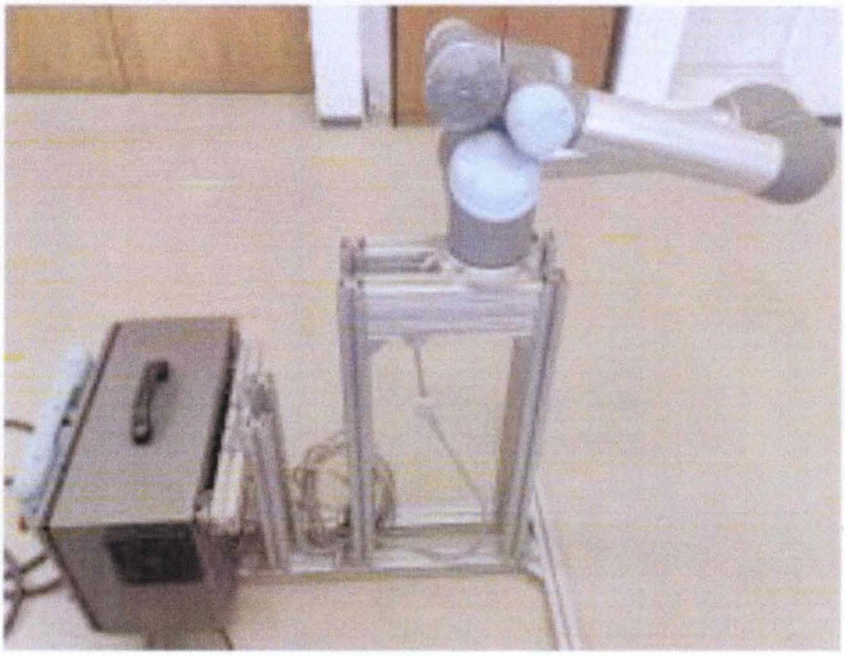

Figure 4: UR5– folded position [4]

A four-wheel mobile vehicle with a robotic arm is developed. The robotic arm has three degrees of freedom, and it possesses two revolute joints, one prismatic joint, and a claw end device. The four-wheel mobile vehicle is composed of two DC motors with two Hall sensors. These was according to S. -P. Kuo, K. -Y. Tsai and Y. -G. Leu, 2020 [5]. In order to control mechanical arm, due of the absence of a rotating joint at the base of the robotic arm, the robot utilizes the four-wheel vehicle to adjust its heading.

After facing the item to be grabbed, the robot arm begins the manipulation phase. Inverse kinematics is used to determine the change value of this robotic arm. The image-processing-derived object coordinates are replaced into the inverse kinematics formulae. The microprocessor determines the rotation degree and distance travelled. Three sets of joints comprise this robotic arm. The joint 1 is perpendicular to the four-wheel vehicle cover and is used to modify the arm crossbar's inclination. The joint 2 at the arm crossbar's end controls the crossbar's extension length, while the joint 3 changes the angle between the claw portion and the crossbar.

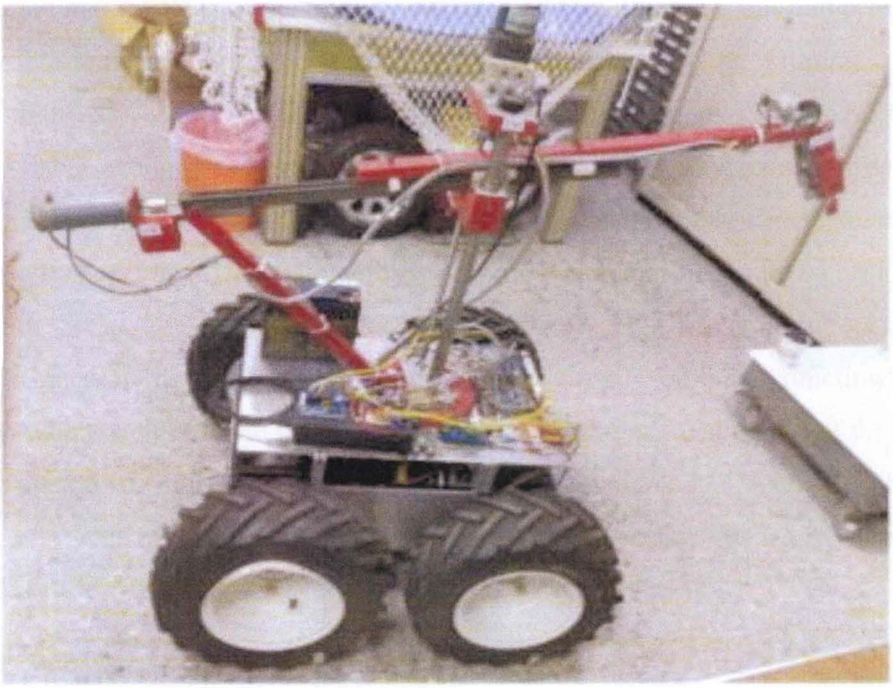

Figure 5: Example of the four-wheeled mobile robot [5]

2.3 Pick-and-place function

For the technologies and coordination of end effector, according to A. Farooq, M. Irfan, A. Saeed, T. Ansar, S. Arshad and N. Chumuang, [6] 2019, stated that in order to improve reliability of modern robotic systems, generally two parallel approaches are often used. First approach is to understand its failure modes and it's modeling while the second approach is to analyze failure of available field data and use models for stimulation and predicting reliability of robotic system. Figure 6 illustrates how the proposed four-degree-of-freedom robotic arm manipulator is operated from an autonomous ground vehicle with the addition of two head-on-flippers. The prototype hardware system as a whole is split into three primary components. The first task is to design and construct a robotic manipulator and arm gripper mechanically using CAED software. We simulated and validated the electronics circuits in the second module before fabricating the final printed circuit boards for controlling primarily the high voltage motor

driving modules. The third and final step included integrating and testing the solar charger with 12 V DC batteries. Robotic arm manipulators are used in industries, medical, and natural catastrophe to carry out operations with high focus and precision. A 4-DOF robotic manipulator has been built. PWM produced by the Arduino controller board from the RF receiver controls the High Torque RC servo motors on each DOF joint link. The linear actuator and gripper are the two primary components of our developed robotic manipulator. For to and from, we utilized a four-bar actuator mechanism, while for pick and place, we utilized a finger type gripper capable of lifting up to 1.5 kg.

To calculate the necessary magnitude of the gripper force as a function of these variables, where weight is the force that tends to cause the pat to slide out of the gripper. Torque equals force multiplied by distance. Torque = Item Weight Arm Length + Arm Weight 1/2 Arm Length are some additional variables to consider while lifting up an object. We added up all forces exerted to half the length of the arm.

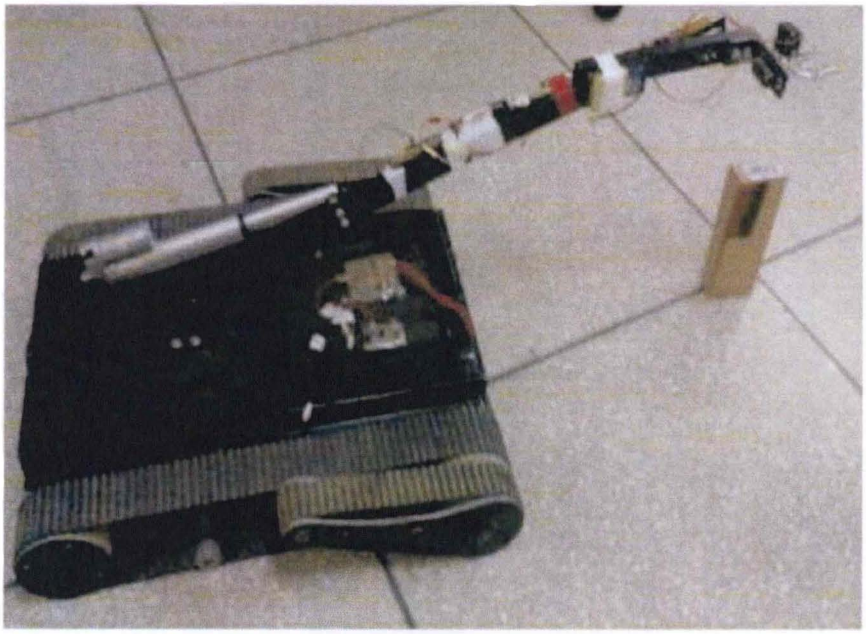

Figure 6: Example of robotic manipulator prototype on a UGV for picking a random object [6]

Z. Li, H. Ma, Y. Ding, C. Wang and Y. Jin, 2020 [7], presents an improved deep deterministic policy gradient algorithm based on a six-DoF (six multi-degree-of-freedom) arm robot. Improved the experience pool of the traditional DDPG (deep deterministic policy gradient) algorithm by adding a success experience pool and a collision experience pool. For a six-DOF arm robot, ROS model is used to created six-DOF arm robot. The UR series arm robot is the world's first collaborative arm robot and a seminal example of a collaborative robot. As a result, we chose the UR5 arm robot model for this article. The UR5 arm robot's work area is 850mm in radius. It is equipped with six motors that regulate the rotation of the arm robot's six axes. The UR5 arm robot is a collaborative robot in the traditional sense as shown in figure 7.

Figure 7: Example of the UR5 arm robot model [7]

Move It is comprised of a number of ROS packages for move operation functions, such as motion planning, operation control, 3D sensing, kinematics, and collision detection. It is a software tool suite. And its motion planning is based on established techniques such as RRT (Rapidly Exploring Random Trees), PRM (Probabilistic Roadmaps), and others. If the distance between the two locations in which the arm robot travels in the configuration space is too great, move It will often have a wide range of motion to ensure consistent movement, and it will collide readily with the surrounding area. Figure 8 illustrates the route planned by the RRT algorithm, whereas Figure 9 illustrates the route planned by the PRM method. Arm robot was trained using deep reinforcement learning, and the arm robot's control amount is the angle of six axes. We can avoid using moveit's in-house planning package by introducing the third-party product arbotix. Arbotix is a servo simulator controller written in ROS that can easily control the whole robotic arm by manipulating the arm robot's six motors.

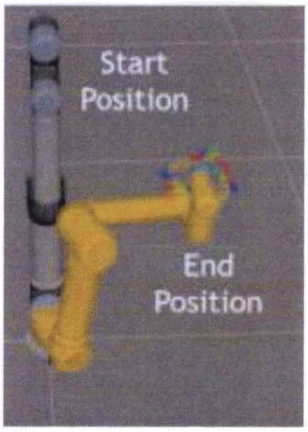

Figure 8: Example of the path planned by the RRT algorithm [7]

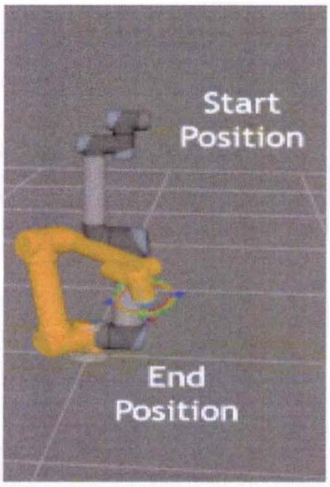

Figure 9: Example of the path planned by the PRM algorithm [7]

Robot manipulators are categorized into three following categories based on their kinematic structure which is serial, parallel and hybrid. According to the Dynamic Modelling of a 3-DOF Articulated Robotic Manipulator Based on Independent Joint Scheme (2017), serial manipulators are a series of links and joints create an open kinematic chain [8]. There are five variations of serial manipulator's arms that industrial robotic manipulators often use: Cartesian, cylindrical, polar, SCARA, and revolution. Parallel manipulators are s connections between links and joints create a complete kinematic chain as shown in Figure 10. Hybrid manipulators is one that combines serial and parallel manipulators as in Figure 11.

Figure 10: Example of Tripod robot consists of three parallel arms [8]

Figure 11: Example of Hybrid autonomous industrial mobile manipulator [8]

The point that will take out from this literature review is coordination between arm manipulator and mecanum wheels. The kinematic and inverse kinematic also the improvement of 6DOF that will used in 3DoF robot arm.

CHAPTER 3

METHODOLOGY

3.1 Introduction

These chapter explain the detail of the work flow for pick-and-place 3DoF robot arm. It involves the flowchart, hardware selection, electronic device, programming language and remote controller configuration.

3.2 Flowchart of the Omnidirectional Manipulator System

This section shows the flow of the Omnidirectional Manipulator System (OMSys) that controlled by play station 2 (PS2). This project using programming language via Arduino IDE. Prior to developing the main programme, a subroutine programmed called basic programme is developed to verify the Arduino's connection to the Omniarm hardware. A flow chart in Figure 12 is created to identify the terminal and to assist in troubleshooting when an error occurs. It is used in the basic programmed to manage the Arduino's pin connections and to control the speed of a motor through Arduino. The primary programmed of this OMSys robot utilizes up to 12 if else statements and up to 12 buttons on the remote controller to control each of its movements.

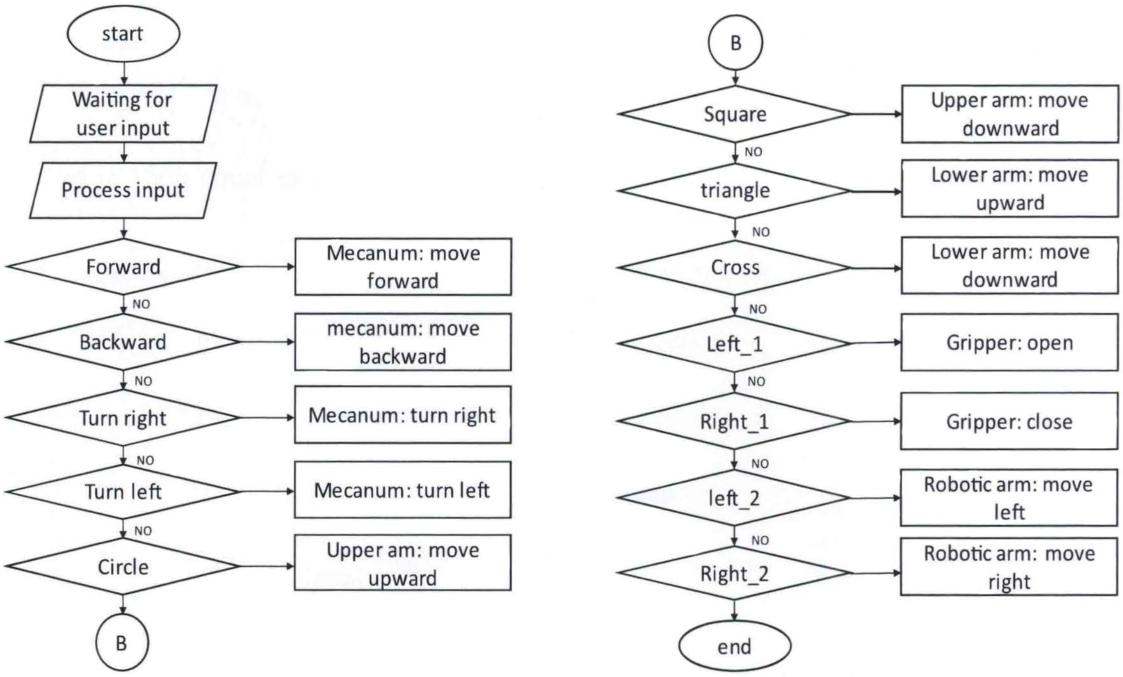

Figure 12: Flowchart of the OMSys

3.3 Selection of grasper and motor

The end effector is the primary function of 3DoF. The load must be picked and placed, and the target load is 150g. Grasper must be sturdy enough to support the load. The selected motor has adequate torque to keep the grasper from loosening.

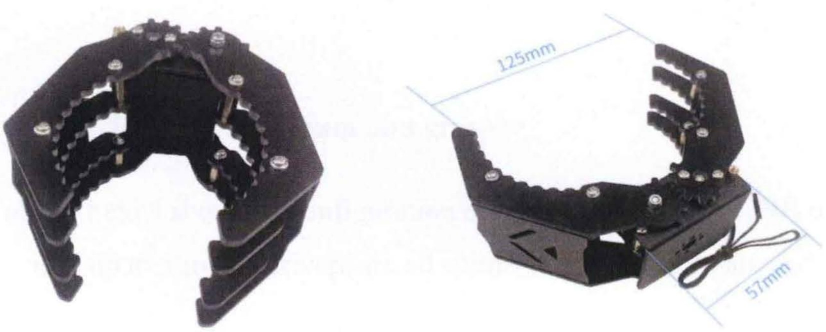

Figure 13 (a) closed gripper and (b) open gripper

Figure 13 above show a LOBOT Metal RC Robot Arm Gripper for the selected grasper for this project. It was made from metal. The grabbing weight is 500g the This grasper can open up 125mm as shown in figure 13 (b).

The load is 150g equal to 1.4709975N. LDX-335MG that shown in figure 14 below is the selected servo motor with blocking torque 17 kg/cm 7.4V which is can up to 1,6671305N. Operating speed is 0.16sec/60 degree 7.4V. The detail of the specification LDX-335MG is shown in appendix A.

Figure 14 LDX-335MG

3.4 Code for arm, mecanum and grasper

The table 1 below show the configuration of Arduino Mega. The INB of all motor driver which used up to 5 motor drivers are all connected with NOT gate and input from signal INA.

Table 1 Pin Configuration of Arduino

Input from Arduino (pin)	PWM (pin)	Type of Control
50	33	Motor 1
53	34	Motor 2
56	35	Motor 3
59	36	Motor 4
62	37	DC Worm Gear
74	21	Linear Actuator 1
79	43	Linear Actuator 2
9	-	Servo motor

3.5 Remote configuration

The figure 14 above shows the configuration of the OMSys. PS2 wireless are used to control the whole project. The controller used 2.4GHZ frequency and capability for have vibration feedback. Transmitter (Tx) and receiver (Rx) are used separately and using double of the AAA battery as then power supply to the controller. The remote configuration can be seen in figure 15 below.

Figure 15 configuration of remote controller

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This section will show the detailed of the testing result for the 3DoF robot arm. It will be divided into two part which is the placement of the gripper and control system for the grasper/gripper.

4.2 Placement of gripper/grasper

Grasper was placed at the end of the effector to be a pick-and-place 3DoF. The arm is still stable when the gripper is placed. When arm is turn off, it still in the perfect condition and stable. Grasper was connected to Arduino and powered by 7.4v battery. It was attach using M2 screw. Figure 16 below the attachment of gripper

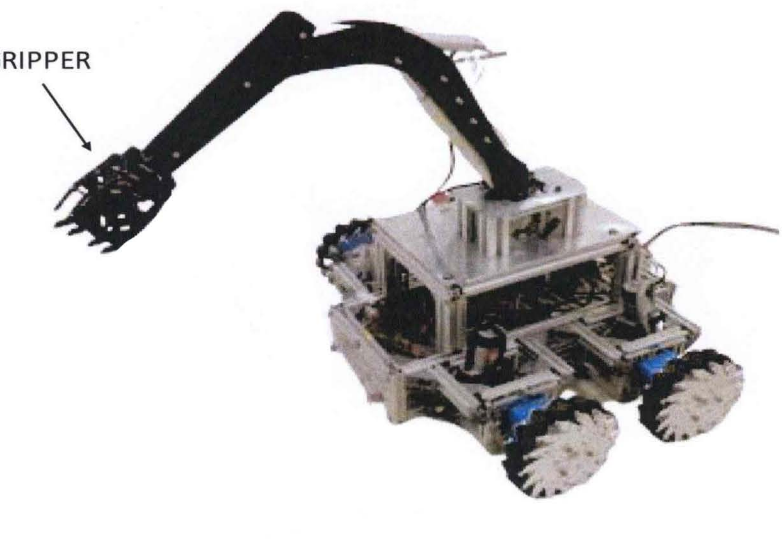

Figure 16 attachment of gripper

4.3 Control system of gripper/grasper

Grasper was controlled by play station 2 (ps2) controller. It using wireless ps2 which means it using transmitter and receiver. It transmits to give signal to Arduino. Using software Arduino IDE, a coding for grasper has been created and shown in appendix B and C.

CHAPTER 5

CONCLUSION

5.1 Introduction

In conclusion, the purpose of the project is to pick-and-place load. With this addition, the robotic arm generously functions as a pick-and-place 3DoF arm. From this project, it can conclude that grasper was attached at the end of the effector and still stable even without power supply on it. Unfortunately, this project was not successful when paired with a controller. These was a main point for this project and as the report, it can be seen that the objective was not achieved.

5.2 Recommendation for future project

Recommendation for the future, coding for the overall project can be done using Simulink MATLAB. By enhancing the torque of the servo motor, it can pick a heavy load which is more than 150g.

REFERENCES

[1] Farman, Madiha & Al-Shaibah, Muneera & Aoraiath, Zoha & Jarrar, Firas. (2018). Design of a Three Degrees of Freedom Robotic Arm. International Journal of Computer Applications. 179. 12-17. 10.5120/ijca2018916848.

[2] C. Röhrig, D. Heß and F. Künemund, "Motion controller design for a mecanum wheeled mobile manipulator," 2017 IEEE Conference on Control Technology and Applications (CCTA), 2017, pp. 444-449, doi: 10.1109/CCTA.2017.8062502

[3] C. Tsai, C. Kuo, C. Chan and X. Wang, "Global path planning and navigation of an omnidirectional Mecanum mobile robot," 2013 CACS International Automatic Control Conference (CACS), 2013, pp. 85-90, doi: 10.1109/CACS.2013.6734112

[4] I. Zamfirescu and C. Pascal, "Modelling and simulation of an omnidirectional mobile platform with robotic arm in CoppeliaSim," 2020 24th International Conference on System Theory, Control and Computing (ICSTCC), 2020, pp. 667-672, doi: 10.1109/ICSTCC50638.2020.9259773.

[5] S. -P. Kuo, K. -Y. Tsai and Y. -G. Leu, "Object Moving System of Four-wheel Mobile Vehicle with Mechanical Arm," 2020 International Automatic Control Conference (CACS), 2020, pp. 1-4, doi: 10.1109/CACS50047.2020.9289797.

[6] A. Farooq, M. Irfan, A. Saeed, T. Ansar, S. Arshad and N. Chumuang, "Design and Development of a Low-Cost Indigenous Solar Powered 4-DOF Robotic Manipulator on an Unmanned Ground Vehicle," 2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), 2019, pp. 1-5, doi: 10.1109/iSAI-NLP48611.2019.9045302

[7] Z. Li, H. Ma, Y. Ding, C. Wang and Y. Jin, "Motion Planning of Six-DOF Arm Robot Based on Improved DDPG Algorithm," 2020 39th Chinese Control Conference (CCC), 2020, pp. 3954-3959, doi: 10.23919/CCC50068.2020.9188521

[8] Sadegh, Hossein & Zarabadipour, Hassan. (2014). Modeling, Simulation and Position Control of 3DOF Articulated Manipulator. Indonesian Journal of Electrical Engineering and Informatics. 2. 132-140. 10.11591/ijeei.v

APPENDICES

Appendix A: LDX 335MG

Weight	60g
Size	40*20*40.5mm
Speed	0.16sec/60°7.4V
Stall Torque	15kg/cm 6V; 17kg/cm 7.4V
No-load Current	100mA

Appendix B: Basic Program

```
//M1

const int INA_1_pin = 50; //grey, Motor 1 direction
```

```
const int INB_1_pin = 51; //white, Motor 1 direction

const int PWM_1_pin = 33; //orange, Motor 1 speed

const int INA_2_pin = 53; //grey, Motor 2 direction

const int INB_2_pin = 52; //white, Motor 2 direction

const int PWM_2_pin = 34; //orange, Motor 2 speed


//M2

const int INA_3_pin = 56; //grey, Motor 3 direction

const int INB_3_pin = 57; //white, Motor 3 direction

const int PWM_3_pin = 35; //orange, Motor 3 speed

const int INA_4_pin = 59; //grey, Motor 4 direction

const int INB_4_pin = 58; //white, Motor 4 direction

const int PWM_4_pin = 36; //orange, Motor 4 speed


//M3

const int INA_5_pin = 62; //grey, DC WORM GEAR

const int INB_5_pin = 63; //white,
```

```
const int PWM_5_pin = 37; //orange,


//M4

const int INA_6_pin = 68; //grey, LINEAR ACTUATOR BAWAH@1 direction

const int INB_6_pin = 69; //white,  direction

const int PWM_6_pin = 39; //orange,  speed


//M5

const int INA_7_pin = 74; //grey, LINAER ACTUATOR ATAS @ 2  TX1

const int INB_7_pin = 75; //white,direction

const int PWM_7_pin = 41; //orange, speed


void setup()

{

 pinMode(INA_1_pin, OUTPUT);
```

```
pinMode(INB_1_pin, OUTPUT);

pinMode(PWM_1_pin, OUTPUT);

pinMode(INA_2_pin, OUTPUT);

pinMode(PWM_2_pin, OUTPUT);

pinMode(INA_3_pin, OUTPUT);

pinMode(INB_3_pin, OUTPUT);

pinMode(PWM_3_pin, OUTPUT);

pinMode(INA_4_pin, OUTPUT);

pinMode(INB_4_pin, OUTPUT);

pinMode(PWM_4_pin, OUTPUT);


pinMode(INA_5_pin, OUTPUT);

pinMode(INB_5_pin, OUTPUT);

pinMode(PWM_5_pin, OUTPUT);

pinMode(INA_6_pin, OUTPUT);

pinMode(INB_6_pin, OUTPUT);

pinMode(PWM_6_pin, OUTPUT);

pinMode(INA_7_pin, OUTPUT);
```

```
pinMode(INB_7_pin, OUTPUT);

pinMode(PWM_7_pin, OUTPUT);


delay(300);

}

void loop()

{

digitalWrite(INA_1_pin,HIGH);

analogWrite(PWM_1_pin, 255);

digitalWrite(INA_2_pin,HIGH);

analogWrite(PWM_2_pin, 255);

digitalWrite(INA_3_pin,HIGH);

analogWrite(PWM_3_pin, 255);

digitalWrite(INA_4_pin,HIGH);

analogWrite(PWM_4_pin, 255);

digitalWrite(INA_5_pin,HIGH);

analogWrite(PWM_5_pin, 255);

digitalWrite(INA_6_pin,HIGH);
```

```
analogWrite(PWM_6_pin, 255);

digitalWrite(INA_7_pin,HIGH);

analogWrite(PWM_7_pin, 255);


delay(5000);

digitalWrite(INA_1_pin,LOW);

analogWrite(PWM_1_pin, 255);

digitalWrite(INA_2_pin,LOW);

analogWrite(PWM_2_pin, 255);

digitalWrite(INA_3_pin,LOW);

analogWrite(PWM_3_pin, 255);

digitalWrite(INA_4_pin,LOW);

analogWrite(PWM_4_pin, 255);



digitalWrite(INA_5_pin,LOW);

analogWrite(PWM_5_pin, 255);

digitalWrite(INA_6_pin,LOW);
```

```
analogWrite(PWM_6_pin, 255);

digitalWrite(INA_7_pin,LOW);

analogWrite(PWM_7_pin, 255);

delay(5000);

}
```

Appendix C: Full program

```
#include <SoftwareSerial.h>

#include <Cytron_PS2Shield.h>
```

```
#include<Servo.h>

Cytron_PS2Shield ps2(25,26); // SoftwareSerial: Rx and Tx pin

//Cytron_PS2Shield ps2; // HardwareSerial

const int INA_1_pin = 50; //grey, Motor 1 direction

const int PWM_1_pin = 33; //orange, Motor 1 speed

const int INA_2_pin = 53; //grey, Motor 2 direction

const int PWM_2_pin = 34; //orange, Motor 2 speed

const int INA_3_pin = 56; //grey, Motor 3 direction

const int PWM_3_pin = 35; //orange, Motor 3 speed

const int INA_4_pin = 59; //grey, Motor 4 direction

const int PWM_4_pin = 36; //orange, Motor 4 speed

const int INA_5_pin = 62; //grey, Motor 5 direction

const int PWM_5_pin = 37; //orange, Motor 5 speed

const int INA_6_pin = 68; //grey, Motor 6 direction

const int PWM_6_pin = 39; //orange, Motor 6 speed

const int INA_7_pin = 74; //grey, Motor 7 direction

const int PWM_7_pin = 41; //orange, Motor 7 speed

Servo servo_X;
```

```
int val;

void setup()
{
 ps2.begin(115200); // This baudrate must same with the jumper setting at PS2 shield
 Serial.begin(115200);

 pinMode(INA_1_pin, OUTPUT);
 pinMode(PWM_1_pin, OUTPUT);
 pinMode(INA_2_pin, OUTPUT);
 pinMode(PWM_2_pin, OUTPUT);
 pinMode(INA_3_pin, OUTPUT);
 pinMode(PWM_3_pin, OUTPUT);
 pinMode(INA_4_pin, OUTPUT);
 pinMode(PWM_4_pin, OUTPUT);
 pinMode(INA_5_pin, OUTPUT);
 pinMode(PWM_5_pin, OUTPUT);
 pinMode(INA_6_pin, OUTPUT);
```

```
pinMode(PWM_6_pin, OUTPUT);

pinMode(INA_7_pin, OUTPUT);

pinMode(PWM_7_pin, OUTPUT);

servo_X.attach(43);

delay(300); //added delay to give wireless ps2 module some time to startup, before
configuring it

}



void loop()

{

if(ps2.readButton(PS2_DOWN) == 0) // 0 = pressed, 1 = released

{

digitalWrite(INA_1_pin,HIGH);

analogWrite(PWM_1_pin, 180);

digitalWrite(INA_2_pin,LOW);

analogWrite(PWM_2_pin, 180);

digitalWrite(INA_3_pin,HIGH);

analogWrite(PWM_3_pin, 180);
```

```
digitalWrite(INA_4_pin,LOW);

analogWrite(PWM_4_pin, 180);

}

else if (ps2.readButton(PS2_UP) == 0) // 0 = pressed, 1 = released

{

digitalWrite(INA_1_pin,LOW);

analogWrite(PWM_1_pin, 180);

digitalWrite(INA_2_pin,HIGH);

analogWrite(PWM_2_pin, 180);

digitalWrite(INA_3_pin,LOW);

analogWrite(PWM_3_pin, 180);

digitalWrite(INA_4_pin,HIGH);

analogWrite(PWM_4_pin, 180);

}

else if (ps2.readButton(PS2_RIGHT) == 0) // 0 = pressed, 1 = released

{
```

```
digitalWrite(INA_1_pin,HIGH);

analogWrite(PWM_1_pin, 255);

digitalWrite(INA_2_pin,HIGH);

analogWrite(PWM_2_pin, 255);

digitalWrite(INA_3_pin,LOW);

analogWrite(PWM_3_pin, 255);

digitalWrite(INA_4_pin,LOW);

analogWrite(PWM_4_pin, 255);

}


else if (ps2.readButton(PS2_LEFT) == 0) // 0 = pressed, 1 = released

{

digitalWrite(INA_1_pin,LOW);

analogWrite(PWM_1_pin, 255);

digitalWrite(INA_2_pin,LOW);

analogWrite(PWM_2_pin, 255);

digitalWrite(INA_3_pin,HIGH);

analogWrite(PWM_3_pin, 255);
```

```
digitalWrite(INA_4_pin,HIGH);

analogWrite(PWM_4_pin, 255);

}


else if (ps2.readButton(PS2_CROSS) == 0) // 0 = pressed, 1 = released(linear2)

{

digitalWrite(INA_5_pin,HIGH);

analogWrite(PWM_5_pin, 255);

}


else if (ps2.readButton(PS2_TRIANGLE) == 0) // 0 = pressed, 1 = released(linear 2)

{

digitalWrite(INA_5_pin,LOW);

analogWrite(PWM_5_pin, 255);

}


else if (ps2.readButton(PS2_CIRCLE) == 0) // 0 = pressed, 1 = released(linear 1)

 {
```

```
digitalWrite(INA_6_pin,HIGH);

analogWrite(PWM_6_pin, 255);

}

else if (ps2.readButton(PS2_SQUARE) == 0) // 0 = pressed, 1 = released(linear 1)

{

digitalWrite(INA_6_pin,LOW);

analogWrite(PWM_6_pin, 255);

}

else if (ps2.readButton(PS2_LEFT_2) == 0) // 0 = pressed, 1 = released(arm motor)

{

digitalWrite(INA_7_pin,HIGH);

analogWrite(PWM_7_pin, 255);

}

else if (ps2.readButton(PS2_RIGHT_2) == 0) // 0 = pressed, 1 = released(LINEAR 1)

{
```

```
digitalWrite(INA_7_pin,LOW);

analogWrite(PWM_7_pin, 255);

}


else if (ps2.readButton(PS2_RIGHT_1) == 0) // 0 = pressed, gripper close

{

 for (val = 0; val <= 180; val += 1)

{

servo_X.write(val);

delay(10);

}

}


else if (ps2.readButton(PS2_LEFT_1) == 0) // 0 = pressed, 1 gripper open

 {

 for (val = 0; val <= 180; val += 1)

{

servo_X.write(val);
```

```
delay(10);
}
}

else
{
analogWrite(PWM_1_pin, 0);
analogWrite(PWM_2_pin, 0);
analogWrite(PWM_3_pin, 0);
analogWrite(PWM_4_pin, 0);
analogWrite(PWM_5_pin, 0);
analogWrite(PWM_6_pin, 0);
analogWrite(PWM_7_pin, 0);
}
}
```