

## A Methodology for the Design of 6D Robotic Arm

Younis, Mona

Higher Technological Institute, Mechanical Engineering Department, Next to Small Industries Complex

Nasser, Eman

Higher Technological Institute, Mechanical Engineering Department, Next to Small Industries Complex

Nasser, Amal

Higher Technological Institute, Mechanical Engineering Department, Next to Small Industries Complex

<https://doi.org/10.5109/4738555>

---

出版情報 : Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES). 7, pp.329-335, 2021-10-21. Interdisciplinary Graduate School of Engineering Sciences, Kyushu University

バージョン :

権利関係 :



## A Methodology for the Design of 6D Robotic Arm

Mona Younis<sup>1</sup>, Eman Nasser<sup>1\*</sup>, Amal Nasser<sup>1</sup>

<sup>1</sup>Higher Technological Institute, Mechanical Engineering Department, Next to Small Industries Complex, Industrial Area 2, 10<sup>th</sup> of Ramadan City, Egypt

\*Corresponding author email: Eman.nasser@hti.edu.eg

**Abstract:** This paper presents a technique that has been applied for a design process of 6-degrees of freedom (DOF) robotic arm Manipulator. The kinematic equations of motion are derived using the Denavit - Hartenberg (DH) representation. In this technique, an analytical solution for the inverse kinematic of 6 DOF robotic arm manipulator is presented, to analyze the movement of the arm from one point in space to another point. experiments were carried out to find the accuracy of the developed robotic arm, the results show that the DH kinematic can improve the robot accuracy

**Keywords:** arm Manipulator; inverse kinematic; Denavit – Hartenberg; mechanical design.

### 1. INTRODUCTION

Nowadays, robotic arm manipulator is used in different industrial fields, it can do exactly the same what human being do with same precision [1]. The robot arms structure and design modify depending on various constraints, such as the tasks they have to implement, the operation space in which they have to work, the characteristics and the dimensions of the objects they have to work with, etc. [2]. Usually, a manipulation tasks are given in terms of the desired trajectory of the end-effector. Since servo motors on joints controls the manipulator movement, trajectory design is necessary. In trajectory design process the time sequence of the arm's joints movement to reach from the start point to destination point is described [3]. In order to design the arm trajectory, the kinematics analysis is required.

The robotic arm's kinematics deals with the analysis of the motion's spatial and time-based properties for each link in robot and also the relationship between each link. This offers an empirical explanation of a robot's spatial movements i.e. a relationship between orientation and position of the end effector and their joint variables [4]. Inverse kinematics is necessary for design of the robot, where the kinematics equations are used to find out the parameters of each joint to reach the desired position [5]. Kinematic robots modeling are used in industrial automation for converting robots to semi-autonomous or fully autonomous. The industrial robots are typically composed of a series of rigid links placed on a base according to for complexity and working environment. They operate in the same way as the human arm. Nowadays, robotic arm manipulator with 6 degrees of freedom (DOF) is widely used in the industries [6]. The 6-degrees of freedom (DOF) robotic arm Manipulator design has Been the aim of a number of research in the robotics field .John et al. [7] designed and constructed manipulator robot of 3 degrees of freedom type SCARA , to test the use of inverse kinematics in motors control. Cortés et al. [8] studied the suitable mechanical and suitable control system to optimize the performance 6 degrees of freedom manipulator robot. It is observed that PC-Controller's software has an important role in optimum functionality of the complete system. Ciliz et al. [9] compared the performance of different adaptive control algorithms on a two axis SCARA types. They claimed the multiple model's method can improve the tracking performance. dynamics and Kinematics of SCARA robot was also simulated and derived using various software [10,11]. Additionally different robots controlled with PID were examined by Sahari et al. and

Kern et al. [12,13]. However, to the best of authors knowledge, there are no studies available suggest analytical solution for the inverse kinematics of 6-degrees of freedom (DOF) Manipulator robot. In this paper, an analytical model for 6 DOF Robotic Arm Manipulator design is proposed. The main contents of this article are as follows: Section 2 addresses A Method for the Design of 6 DOF Robotic Arm Manipulator. kinematics analysis and Validation of Inverse Kinematics is demonstrated in Section 3. System Implementation is explained in Section 4. The task results of the robot are presented in Section 5

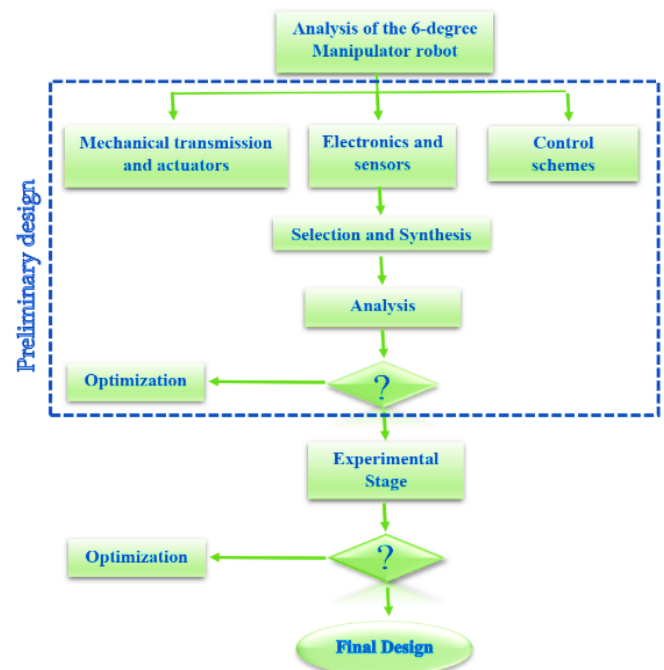


Figure 1: A scheme for a design process stage

### 2. DESIGN METHODOLOGY

In order to build a 6-degrees of freedom (DOF) Manipulator robot, we follow three steps systematic and task-oriented process as illustrated in figure 1. At figure 2 method of manipulator robot design according to systematic and task-oriented process. The methodology consists of use DH representation and analytical solution of the kinematic equation .



Figure 2: A proposed design process for 6 DOF Robotic Arm Manipulator

### 2.1 Design objectives

Design objectives for 6-degrees of freedom (DOF) Manipulator robot can be summarized as follows:

- 1- building and controlling 6-degrees of freedom (DOF) Manipulator robot that can capture objects and move it to the desired location.
- 2- Using the Proportional-Integral-Derivative Control Unit (PID) to design the console that can in control the position of the triggers, allowing the processor to place objects in the exact location.
- 3- Creating a graphical user interface (GUI) that acts as a human interface device to receipt user commands and passing them to a manipulator.

### 2.2 Preliminary design

#### 2.2.1 Mechanical Design

Robotic manipulator consists mainly of links that joint together to perform construct the required mechanism. Figure 3 shows the suggested design of the robotic manipulator. Each part of the robotic manipulator is designed by CAD software program (Solid works 2019) by individually until it is produced by the laser cutter (figure 4) or by 3D printer and then assembled and tied together until the final arm shape.



Figure 3: Suggested design of the robotic manipulator

Because aluminum can offer superior properties such as lightweight, high Formability and very weather-proof [14], it is selected to build all robotic manipulator parts. Figure 4 shows Laser cut layout for all robot parts for the aluminum 6061 sheet. The design of all robot parts is collected in figure 5.

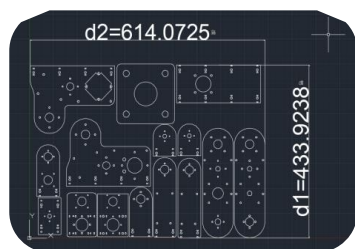


Figure 4 Laser cut layout

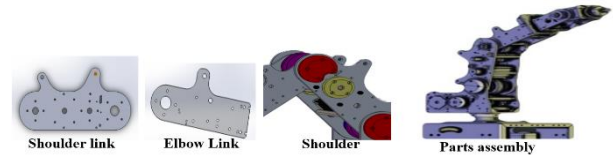
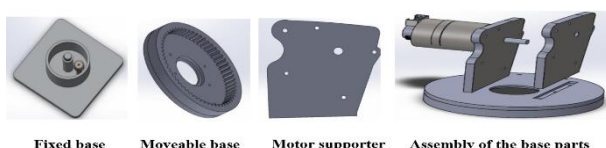


Figure 5 Robot arm parts

Maximum payload for the robotic manipulator is one kg and expected mass of complete robotic manipulator is 25 kg. Maximum height for arm and end effector is 700 mm and the robotic manipulator have six axes for motion. The rotation transmission made by using belt and pulley. Transmission system is placed to each joint to increase the torque of motor. according to the arm design there are six locations for the actuators, table 1 explains the details of transmission system. Figure 6 shows actual picture for manipulator robotic.

Table 1: details of transmission system.

Actuator location	Requirement	Specifications of pully		
base	Reduce the speed to one-third. Increase the torque to three times.	Small pully	Large pully	Stage
		Teeth No. = 30 Inner diameter= 6.35 mm	Teeth No. = 90 Inner diameter = 30 mm	Single Stage
Shoulder	Reduce the speed to nine increase the torque to nine times the torque	Teeth No. = 20 Inner diameter= 8 mm	Teeth No. = 60 Inner diameter = 10 mm	Single Stage
		Teeth No. = 20 Inner diameter=10 mm	Teeth No. = 60 Inner diameter = 10 mm	Second Stage
elbow	Reduce the speed to nine Increase the torque to nine times the torque	Teeth No. = 20 Inner diameter= 6.35 mm	Teeth No. = 60 Inner diameter = 10 mm	Single Stage
		Teeth No. = 20 Inner diameter=10 mm	Teeth No. = 60 Inner diameter = 10 mm	Second Stage
wrist	reduce the speed to one-third as well as increase the torque to three times	Teeth No. = 18 Inner diameter= 5 mm	Teeth No. = 54 Inner diameter = 8 mm	Single Stage



Figure 6 real image for robotic manipulator

### 3. KINEMATIC MODEL

The robotic manipulator has six joints to give it the rotation ability to the desired position. Figure 7 shows

details angle for all joints and relationships between all links. In this study, the standard Denavit - Hartenberg (DH) convention and methodology are used to derive its kinematics. In these methods the kinematic model can be described by making matrix has a relationship of translation or rotation between the all links. table 2 shows DH parameters. kinematic model is based on the Denavit - Hartenberg Coordination system as follow:

- 1- Joint angle ( $\Theta$ ) rotate around  $Z_{n-1}$  axis to match  $X_{n-1}$  to  $X_n$  axis.
- 2- Twist angle ( $\alpha$ ) rotate around  $X_n$  axis to match  $Z_{n-1}$  to  $Z_n$  axis.
- 4- link offset ( $r$ ) is the distance between frame (n) and frame (n-1) along  $X_n$  axis.
- 3- link length ( $d$ ) is the distance between frame (n) and frame (n-1) along  $Z_{n-1}$  axis.

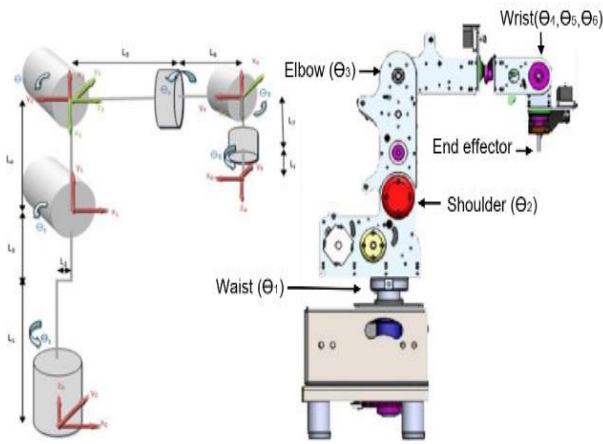


Figure 7 Geometric configuration  
Table 2 DH parameters

The transformations between each two successive joints can be written by simply substituting the parameters from the parameters table into the following matrix:

$$H(n|n) = \begin{bmatrix} \cos(\theta(n)) & -\sin(\theta(n)) \cos(\alpha(n)) & \sin(\theta(n)) \sin(\alpha(n)) & r \cos(\theta(n)) \\ \sin(\theta(n)) & \cos(\theta(n)) \cos(\alpha(n)) & -\cos(\theta(n)) \sin(\alpha(n)) & r \sin(\theta(n)) \\ \theta & \sin(\theta(n)) & \cos(\alpha(n)) & d \\ \theta & \theta & \theta & 1 \end{bmatrix} \quad (1)$$

The 3 X 3 matrix comprising of first three rows and first

	$\theta$	$\alpha$	$r$	$D$
1	$\theta_1$	$\frac{\pi}{2}$	L2	L1+L3
2	$\theta_2 + \frac{\pi}{2}$	0	L4	0
3	$\theta_3 + \pi$	$-\frac{\pi}{2}$	0	0
4	$\theta_4 + \pi$	$-\frac{\pi}{2}$	0	L5+L6
5	$\theta_5 + \frac{\pi}{2}$	$-\frac{\pi}{2}$	0	0
6	$\theta_6$	0	0	L7+L8

three columns is the rotation while the last column represents the position (x, y, z).

$$H_{01} = \begin{bmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & 12 \cos(\theta_1) \\ \sin(\theta_1) & 0 & -\cos(\theta_1) & 12 \sin(\theta_1) \\ 0 & 1 & 0 & 11 \cdot 13 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$H_{12} = \begin{bmatrix} -\sin(\theta_2) & -\cos(\theta_2) & 0 & 14 \sin(\theta_2) \\ \cos(\theta_2) & -\sin(\theta_2) & 0 & 14 \cos(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$H_{23} = \begin{bmatrix} -\cos(\theta_3) & 0 & \sin(\theta_3) & 0 \\ -\sin(\theta_3) & 0 & -\cos(\theta_3) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$H_{34} = \begin{bmatrix} -\cos(\theta_4) & 0 & \sin(\theta_4) & 0 \\ -\sin(\theta_4) & 0 & -\cos(\theta_4) & 0 \\ 0 & -1 & 0 & 15+16 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$H_{45} = \begin{bmatrix} -\sin(\theta_5) & 0 & -\cos(\theta_5) & 0 \\ \cos(\theta_5) & 0 & -\sin(\theta_5) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$H_{56} = \begin{bmatrix} \cos(\theta_6) & -\sin(\theta_6) & 0 & 0 \\ \sin(\theta_6) & \cos(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & 17+18 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

At the base of the robot, it can be started with the first joint and then transform to the second joint, then to the third until to the arm-end of the robot, and eventually to the end effectors. The total transformation between the base of the robot and the hand is a homogeneous transformation matrix consists of the 3 X 3 rotation matrix and the position (x, y, z) for the end-effector with reference to the base.

$$H_{06} = H_{01} * H_{12} * H_{23} * H_{34} * H_{45} * H_{56}; \quad (8)$$

The corresponding variables of each joint could be found with the given location requirement of the end of the manipulator in the given references coordinates system. Inverse kinematic analysis is done by multiplying each matrix of H matrices on the right side of above equation and then equalizing the corresponding elements of the equal matrices of both ends.

$$H_{01} * H_{12} * H_{23} * H_{34} * H_{45} * H_{56} = \begin{bmatrix} R_{xyz}(1,1) & R_{xyz}(1,2) & R_{xyz}(1,3) & x \\ R_{xyz}(2,1) & R_{xyz}(2,2) & R_{xyz}(2,3) & y \\ R_{xyz}(3,1) & R_{xyz}(3,2) & R_{xyz}(3,3) & z \\ \theta & \theta & \theta & 1 \end{bmatrix} \quad (9)$$

Hence  $R_{xyz}$  is the given position and orientation for end-effector about base to achieve (user requirement) known.  $H_{06}$  is the total transformation between the base in the given references coordinates system (function of thetas) thetas are required. From our manipulator specification, Links lengths in mm are:

$$L1 = 215.5; L2 = 19.58; L3 = 173; L4 = 195; L5 = 182; L6 = 120.27; L7 = 62; L8 = 62;$$

and ranges of actuated joints angles are:



$-180 < \theta_1 < 180$ ;  $108 < \theta_2 < 45$ ;  $-15 < \theta_3 < 194$   
 $-180 < \theta_4 < 180$ ;  $-19 < \theta_5 < 148$ ;  $-180 < \theta_6 < 180$

Now solving this equation by equating individual terms of both matrices, we get the inverse solution. There is may be more than one solution or no solution if the desired point is out of the workspace of the manipulator:  
 $H0\_6(1,1) = R_{xyz}(1,1)$ ;  $H0\_6(1,2) = R_{xyz}(1,2)$ ;  
 $H0\_6(1,3) = R_{xyz}(1,3)$ ;  $H0\_6(1,4) = X$ ;  
 $H0\_6(2,1) = R_{xyz}(2,1)$ ;  $H0\_6(2,2) = R_{xyz}(2,2)$ ;  
 $H0\_6(2,3) = R_{xyz}(2,3)$ ;  $H0\_6(2,4) = Y$ ;  
 $H0\_6(3,1) = R_{xyz}(3,1)$ ;  $H0\_6(3,2) = R_{xyz}(3,2)$ ;  
 $H0\_6(3,3) = R_{xyz}(3,3)$ ;  $H0\_6(3,4) = Z$ ;  
Whereas twelve above kinematics equations is a function of  $(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6)$ , with inverse kinematic solutions, the value of each joint can be determined in order to place the arm at a desired position and orientation

### 3.1 Validation of Inverse Kinematics

Robotics Toolbox for MATLAB was used to assure from the accuracy of inverse kinematics model, an object has been placed at a known position and orientation. With this known information from a user. Information includes coordinates of the origin, known values of geometric parameters of the robot, Links lengths and ranges of actuated joints angles. Values of joints angles to get a particular position are:

$\theta_1 = 258.8485$ ,  $\theta_2 = 150$ ,  $\theta_3 = 650$ ,  
 $\theta_4 = 53.9891093531661$ ,  
 $\theta_5 = 106.99310099067357$ ,  $\theta_6 = 18.51038314678336$

which is equal to  $R_{xyz} =$

$\begin{bmatrix} -0.2771 & 0.0928 & 0.9563 & 257.9 \\ -0.5469 & 0.8031 & -0.2364 & -149.8 \\ -0.7900 & -0.5886 & -0.1718 & 650.2 \end{bmatrix}$

By substituting for Links lengths then multiplying each matrix of H matrices on the right side of above equation, H06 is the result, then equalizing each element of H06 matrix (a function of joint angles) to each element of the required matrix  $R_{xyz}$  (known):

$H06(1,1) = -0.2771$ ;  $H06(1,2) = 0.0928$ ;  
 $H06(1,3) = 0.9563$ ;  $H06(1,4) = 257.9$ ;  
 $H06(2,1) = -0.5469$ ;  $H06(2,2) = 0.8031$ ;  
 $H06(2,3) = -0.2364$ ;  $H06(2,4) = -149.8$ ;  
 $H06(3,1) = -0.7900$ ;  $H06(3,2) = -0.5886$ ;  
 $H06(3,3) = -0.1718$ ;  $H06(3,4) = 650.2$ ;

By using MATLAB, joint angles as a result of analytical inverse method to solve above twelve kinematics equations are:

$\theta_1 = -40.7096$ ;  $\theta_2 = 34.6655$ ;  
 $\theta_3 = -10.7738$ ;  $\theta_4 = 40.9022$ ;  
 $\theta_5 = 47.2396$ ;  $\theta_6 = -74.1101$ ;

The inverse kinematic solution has been validated by substituting in the forward model or using Robotics Toolbox for MATLAB. Numerical results together with visual plot of position and orientation of a robot in MATLAB environment (figure 8) gives clear insight of the kinematic behavior of a robot. Inputs of the test MATLAB program are previous outputs of inverse kinematic MATLAB program (joints angles values):  
 $\theta_1 = -40.709$ ;  $\theta_2 = 34.6655$ ;  $\theta_3 = -10.7738$ ;  
 $\theta_4 = 40.9022$ ;  $\theta_5 = 47.2396$ ;  $\theta_6 = -74.1101$ ;  
Out of the test MATLAB program is the desired matrix:

$R_{xyz} =$

$\begin{bmatrix} -0.2771 & 0.0928 & 0.9563 & 257.9 \\ -0.5469 & 0.8031 & -0.2364 & -149.8 \\ -0.7900 & -0.5886 & -0.1718 & 650.2 \end{bmatrix}$

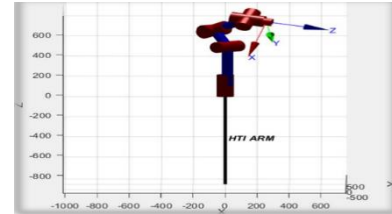


Figure 8 MATLAB Plot for joint angle configuration

## 4. SYSTEM IMPLEMENTATION

The system implementation will consist of hardware and mechanism development and software development.

### 4.1 Mechanism and hardware development

This section deals with the information of important software modules and hardware components such as Controller, Actuators and Sensors used in designing and controlling.

#### Mechanism

The robotic arm mechanism (figure 9) uses parallel structures, that is used consists of 9 links connected appropriately so that 3 parallelogram mechanisms are formed. The structure of the arm ensures that specific links always remain parallel to each other there are 3 sets of parallel links: {1,4,6}, {2,5,8} and {3,7,9}.

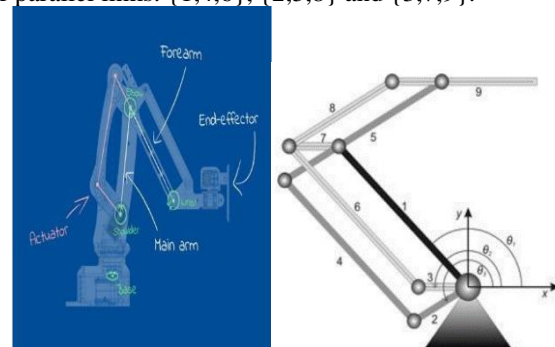


Figure 9: parallel structures of the arm

The Arduino Uno R3 is a microcontroller board based on a removable, dual-in line package (DIP) ATmega328 AVR microcontroller. motor drive in Full Step Mode was used and the 3 MS pins were leaved disconnected and just connect the Direction and the Step pins of the drive to the pins number 3 and 4 on the Arduino Board and as well the Ground and the 5 V pins for powering the board. Also, a 100  $\mu$ F capacitor for decoupling and 12V, 1.5A adapter for powering the motor. suitable motor selection process for each joint was made by calculating the greatest torque at critical condition in the sense that the body weight causes the greatest torque possible, and the safety factor equal to 1.5 in order to make the motion smooth without any vibration. Knowing that the torque is  $T = m.L$ , it is clear from figure 10 that L3 is the largest vertical distance and in the horizontal position, which should be the body in this case. The torque calculation of each motor on its own considering that it carries all the

weight before it is calculated the momentum that causes this weight and figure 11 shows the calculation of the torque:

$$T1 = L1.A1 + 0.5L1.W1$$

$$T2 = L5.A1 + L4.W1 + L3.A2 + L6.W2$$

In this method, the calculation consider that each Lumped mass motor and the momentum caused by the weight A to be the mass multiplied by the vertical distance (L) to the motor axis to measure the torque ( $T = L.A$ ). The momentum caused by each Link (L) and its mass (W) is its mass multiplied by the vertical distance from the center of the Link weight, which is approximately the middle of the Link because it is a regular object (or determined by any CAD program) to the motor axis ( $T = 0.5 LW$ ) The final torque of the motor is the sum of the two plus all the weights ( $T = L.A + 0.5 L. W$ ). Table 2 describes the full details of the selected motors at all joints.

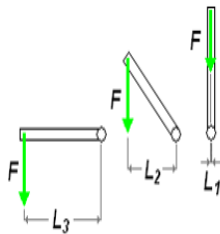


Figure 10 force direction at each joint.

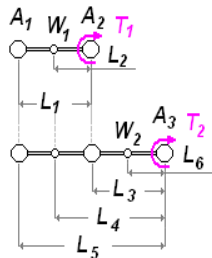
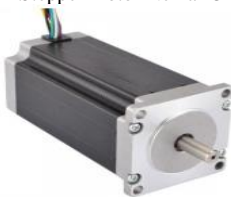


Figure 11 the momentum calculation

NEMA 17 and NEMA 23 bipolar Stepper Motors wires A and C will be connected to the pins 1A and 1B and the B and D wires to the 2A and 2B pins to moving the robot links. Servo actuator was used to make the movement of the gripper, Figure 12 shows Circuit Schematics for wiring connection for each motor.

Table 2 Details of the selected motors at all stages

motor location	Type	Specifications	Stage
base	Stepper motor Nema 23	Torque : 2.2 NM Length : 82 mm Power : 3A Shaft : 6.35 mm 4 wires Step angle : 1.8	Single stage



Shoulder	Stepper motor Nema 23	Torque : 3 NM Length : 112 mm Power : 4.2 A Shaft : 8 mm 4 wires Step angle : 1.8	Second Stage
elbow	Stepper motor Nema 23	Torque: 2.2 NM Length : 82 mm Power : 3A Shaft : 6.35 mm 4 wires Step angle : 1.8	Third Stage
wrist	Three Stepper motors Nema 17	Torque: 50 N CM Length : 48 mm Power : 1.68 A Shaft : 5 mm 4 wires Step angle : 1.8	Fourth stage Fifth stage Sixth stage

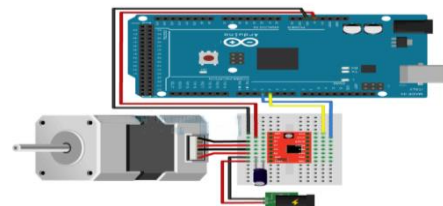


Figure 12 Circuit Schematics

## 4.2 Software Development

Robot GUI on MATLAB apps designer was used to create the user interface. In order to facilitate the process of robot control, the GUI is design to be sample and consists from three sections: Manual section; Hibernate section and Task panel section as shown in figure 13. In manual mode, user can control in each joint of the robot independently from minimum to maximum angle of joint. software limit is adjusted according the maximum angle for each joint to protect robot from breakage at movement if user try to pass the limit of any joint. also, in the Manual mode user can control in the gripper by changing the angle of servo and user can follow communication between pc and robot from Serial port state. In Hibernate section user can control robot movement in Cartesian space configuration (X, Y, Z) and Euler angles. The software has a develop algorithm that translate Cartesian into joint angle configuration. These algorithm takes Cartesian displacement and use inverse kinematics equations to turn them into joint angles, then it takes in consideration micro step and gear ratio of each joint then sent motor steps through serial port to robot controller. User can achieve end effector movement in single axis or multi axis in Cartesian space X, Y, Z or XY, XZ, ZY. It has also reference points for robot as home point, zero point. **Home point button** is point that nearly all motor has consumed minimum power. **Zero-point button** is the reference point in robot kinematic at it all joints are zero. **Hibernate section in robot GUI** It has also panel that display robot end effector position in real time and display also connection mode. **Offline button**

mean serial communication state is off. **ONLINE** mean serial communication state is on It also has **Emergency stop button** that we use as emergency software emergency switch.

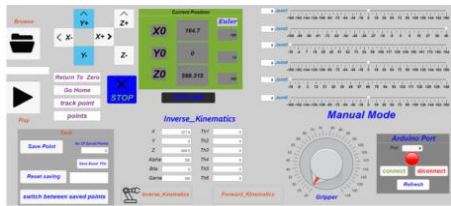


Figure 13 user interface

In task mode user can create task for robot by simple way, by just move robot to specific point and save each point. Then robot could repeat this task immediately and we can. Also save this task as Excel file and make robot do this task late Each point saves as vector of 7 element that 6 for joints angles and one for gripper angle at saved point. Figure 14 shows the flowchart of movement arm robot to finish the task.

## 5. Results and discussion

To evaluate the performance and the accuracy for arm robot numbers of tasks were made with different scenarios, such like picking up an object from one location and placing it onto another location. Figure (15) shows the sequence of robot movement to pick-up object from known source position to another position. Shot 1 shows the home position for the robot; shots 2 and 3 show the arm movement to the destination point; Shot 3 shows the pick-up process for the object. In shots 5 and 6 the robot moves toward destination and in shot 7 the grabber left the object at the destination point. Shots 8 and 9 show the return movement to the home position. Both the source and the destination point should lie inside the operational area.

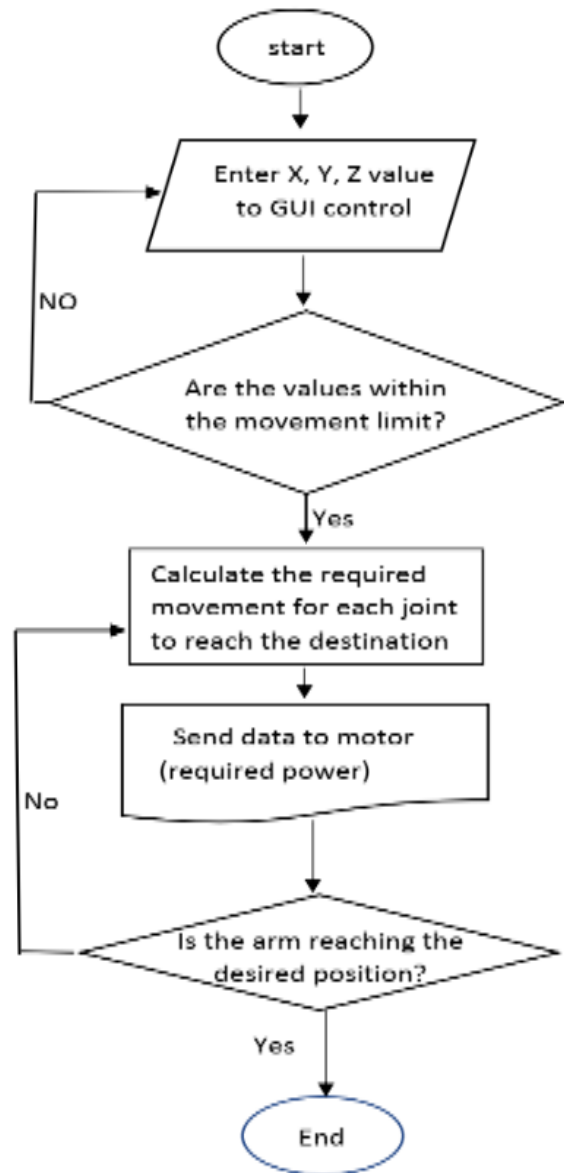


Figure 14 flowchart of movement arm robot



Figure 15 Sequence of pick-up of the object from known position to destination position

## 6. CONCLUSION

Due to robotic arm manipulator has the enough ability to work in different industrial applications, this paper introduces a new design of 6 DOF robotic arm manipulator with an end effector for pick-up operations. The design of proposed arm robot includes 6 degree of freedom that can picks an object from destination on the working space area and places it to the desired position. GUI user interface has been made in Robot GUI on MATLAB to facilitation the robot control. Denavit - Hartenberg (DH) methodology has been used to drive robot kinematics, and to obtain robot's trajectory. To evaluate the efficiency of the arm robot, number of tests were made with different circumstances. The robot passed tests and perform all tasks with high accuracy by ensuring that the object was picked and located in the right position as shown in figure 15.

## 7-REFERENCES

- [1] C. Urrea, and J. K. 'Modeling, simulation and control of a redundant SCARA-type manipulator robot', international Journal of Advanced Robotic Systems, vol. 9, 2012
- [2] Claudio Urrea, Juan Cortés, J. P. 'Design, construction and control of a SCARA manipulator with 6 degrees of freedom', Journal of Applied Research and Technology, vol. 14, pp. 396–404, 2016
- [3] Francis Nturanabo, L. M. and J. B. K. 'Novel Applications of Aluminum Metal Matrix Composites', in Aluminum Alloy and Composites, 2019
- [4] Gao, J., El Sour, M., Keates, S. 'Advances in Manufacturing Technology XXXI', in Multi robot path planning approach for dynamic environment, pp. 275–278, 2017
- [5] Gregor Klančar, I. Š. 'Motion Modeling for Mobile Robots', in Wheeled Mobile Robotics from Fundamentals Towards Autonomous Systems, pp. 13–59, 2017
- [6] John Faber Archila Diaz, Max Suell Dutra, C. J. D. 'Design and Construction Of A Manipulator Type Scara, Implementing A Control System', The International Congress of Mechanical Engineering, pp. 44–51, 2007.
- [7] Jorge Luis Aroca Trujillo, Alexander Pérez-Ruiz, and R. R. S. 'Generation and Control of Basic Geometric Trajectories for a Robot Manipulator Using Compact RIO', Journal of Robotics, pp. 1–17, 2017
- [8] K. S. Mohamed Sahari, K. H. Weng, Y. W. H. 'Design and development of a 4–dof SCARA robot for educational purposes', Jurnal Teknologi, vol. 54, no. 1, pp. 193–215, 2012.
- [9] M. Kemal Ciliz and M. Omer Tuncay 'Comparative experiments with a multiple model based adaptive controller for a SCARA type direct drive manipulator', Robotica, Cambridge University Press, vol. 33, pp. 721–729, 2005.
- [10] M. S. Alshamasin, F. Ionescu, and R. T. A.-K. 'Kinematic modeling and simulation of a SCARA robot by using solid dynamics and verification by matlab/simulink', European Journal of Scientific Research, pp. 388–405, 2009.
- [11] M. T. Das, and L. C. D. (2005) 'Mathematical modelling, simulation and experimental verification of a scara robot', Simulation Modelling Practice and Theory, vol. 13, no. 3, pp. 257–271.
- [12] Mike Wilson, 'Industrial Robots', in Implementation of Robot Systems, pp. 19–33, 2015.
- [13] Paul, R., Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators, 1981.
- [14] Sivčeva, S., Colemana, J., Omerdića, E., Doolya, G. and, Daniel Toala 'Underwater manipulators: A review', Ocean Engineering, vol. 163, pp. 431–450, 2018.