OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

An Improved Accurate Trajectory Control System for Industrial Hydraulic Robotic Arms

Osama A. Montasser
Mechanical Power Engineering Department, Faculty of Engineering, Ain Shams University, Egypt,
On leave to join the British University in Egypt, BUE,

Abstract: *A PC based control system was developed in this work to control a hydraulic pick and place robotic arm with a high tracking accuracy. The hydraulic arm was designed, constructed and controlled through an electronic driver circuit designed by the author. The tracking control system is driven by computer software, the commands of which are connected to the arm by means of a data acquisition card to read the rotational angles of its parts and to actuate the driver circuit of its hydraulic system. The present hydraulic robot arm is controlled to carry out desired pick and place tasks. A smart control software program was designed and constructed by the author using C# programming language. The present software program is designed as a Graphical User Interface application, GUI, and therefore it can be easily operated by non-expert users. Inverse kinematics formulas, derived by the author, are processed by the software to convert the coordinates of the initial and object tracking points from Cartesian to Polar systems as needed. Experimental runs were carried out to verify the effectiveness and the accuracy of the present tracking arm. Experiments showed that nine of ten trials were successful to attain a predesigned accurate pick and place task, which is a good percentage, confirmed the high accuracy of the present tracking system.*

Keywords: *Hydraulic controlled systems, industrial robotic arms, inverse kinematics, PC based control systems, pick and place handling robots*

I. INTRODUCTION

Robotics term is practically defined as the study, design and use of robot systems for manufacturing [5]. Performing unpleasant tasks such that, unsafe, hazardous, and highly repetitive are generally done by robots. They have many different functions such as material handling, assembly, arc welding, resistance welding, machine tool load and unload functions, painting, spraying, etc.

Service robot and an industrial robotic are mainly the two different kinds of robots. Excluding manufacturing operations, service robot operates fully or semi autonomously services useful to human being, [6]. On the other hand, industrial robot is an automatically controlled multipurpose multi axis manipulator [5]. Programmed motions are designed for industrial robots to pick and place or move pieces of different shapes to perform several kinds of tasks. An industrial robot system includes not only the design of industrial robots but also the optimum selecting of any devices and/or sensors required for the robot to perform its tasks.

Dancing hand, weight lifting, and color classification, industrial robots were designed by [1] as an example. Eight degrees of freedom robot arm was developed, as well, to be able to pick and place many objects with a lot of shapes [8].

Massive attention of the robot localization problem is clearly observed in the recent robotic literatures. Localization deals with the estimation of the robot position and orientation, its pose, relative to a given proposed trajectory. This is achieved using position sensors. Compensation for sensors noise and errors is an essential matter for accurate tracing operation [9]. Global localization problem is encountered when the initial robot pose is unknown, otherwise it is called pose tracking problem [7]. Approaches providing solutions to global localization problem are proposed by [2, 3 and 10]. The particle filter, Monte Carlo method, has been applied with great success in mobile robot localization [3, and 4], fault detection [11], and map building [12].

In the present work, a PC based control system is designed and implemented, for accurate tracking control of a hydraulic robotic arm. The robotic arm is driven by five hydraulic cylinders to rotate its parts and to open and close its grippers. Motion of the hydraulic cylinders are controlled using 4/3 flow control valves actuated by electric solenoids from both sides. Electronic circuit was designed and implemented by the author to control the valves solenoids using digital voltage signals decided by the control software.

The control system hardware consists of the solenoids driving circuit and feedback sensors, to measure the rotational angle of the arm parts. A control software program was designed and constructed by the author to read sensors signals, to calculate the actual rotational angles of all the arm parts, and to decide and send appropriate control action to the solenoids driver circuit. The control soft and hard ware are connected to each other using a data acquisition card, DAC, attached to a host computer. Limit switches are attached to the arm parts and automatically operated by the control program to ensure safe arm operation.

This paper is organized as follows. The mechanical construction description of the present robotic arm is presented in the following section, number II. In section III, the inverse kinematics of the present arm mechanical geometry configuration is detailed. The present control hard and soft wares are overviewed in section IV. Results of the present experimental work are discussed in section V. In the last section the conclusions of the present work are summarized. References are listed at the end of this paper.

II. MECHANICAL CONSTRUCTION OF THE PRESENT ARM

2.1 Layout of the arm mechanical design

A pick and place arm manipulator is designed and constructed, in this work, as a four bar mechanism with a rotatable base. It is driven by a five hydraulic linear double acting cylinders. Four of the cylinder pistons linear motions are converted to limited rotational swing motions to rotate the arm links, base, shoulder, elbow, and wrist around their hinges. The fifth cylinder is used with the arm gripper as it is mentioned later in this section. Layout of the construction assembly of the present arm is shown in Fig. 1 below;

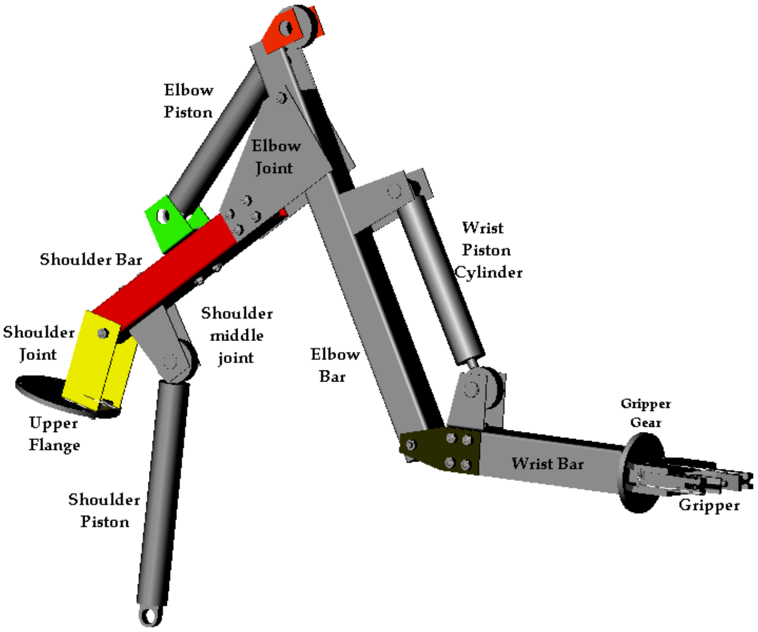


Figure 1 Construction assembly of the present hydraulic arm

The arm is designed to lift approximately a maximum load of 20 kg keeping it balanced over its base. Lengths of arm shoulder, elbow, and wrist bars are designed to be 57.7 cm, 61.7, and 45 cm respectively. Motion of the arm hydraulic cylinders are controlled using 4/3 flow control valves, actuated by electric solenoids from both sides, so it can be moved forward, or backward, or stopped. Electronic circuit was designed and implemented by the author to control the valves solenoids using digital voltage signals decided by the control software.

Four optical encoders of high resolution of 0.044 degree are installed, each on each of the arm links. Encoders' signals for the relative angular motion of each of the link are used as the feedback signal of the closed loop tracking control mode. Limit switches are attached to the hinges of arm parts and sends their signals to the control program to ensure safe arm operation. The control software is connected to the arm to read the signals of the encoders and limit switches and to send appropriate commands to the solenoids driver circuit through a data acquisition card attached to a host computer. Fig. 2, below, shows a schematic diagram for the present arm control system.

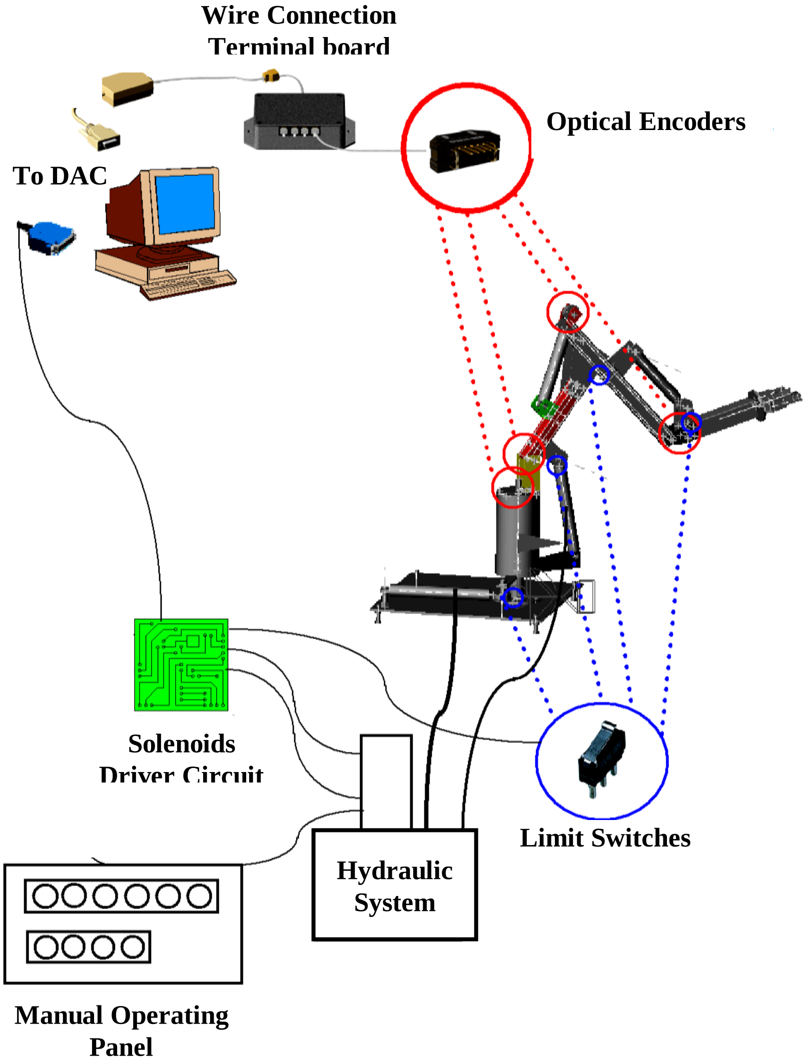


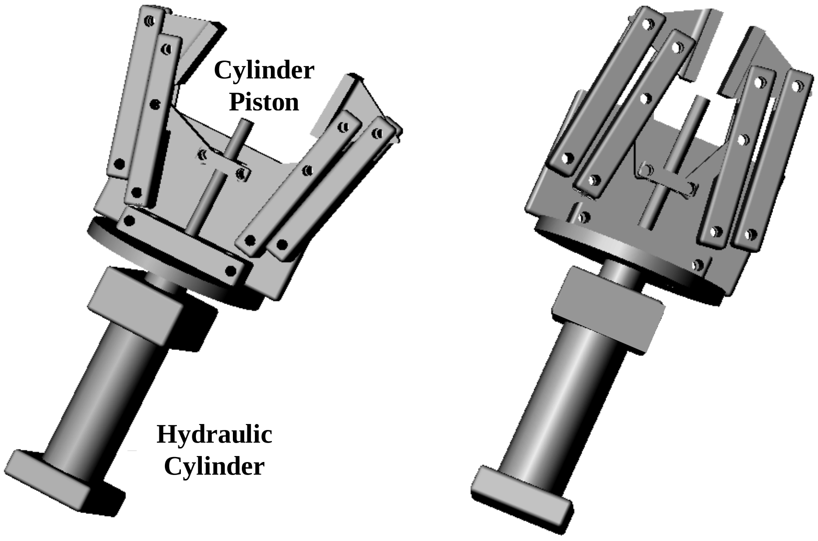
Figure 2 Schematic diagram for the present arm control system, DAC: for Data Acquisition Card

2.2 Gripper of the arm

The gripper end-effectors usually costs about 4-8% of the total cost of any robot. Specially designed end-effectors can cost up to 20% of the total robot cost. End-of-arm tooling in a robot work cell should have the following characteristics:

1. The tooling must be capable of gripping, lifting and releasing an industrial object or family of objects required by the manufacturing process.
2. The tooling may sense the presence of a part in the gripper, using sensors located either on the tooling or at a fixed position in the work cell.
3. Tooling weight must be kept to a minimum because it is added to the picked object weight, the summation of which should not exceed the maximum allowed payload.
4. Containment of the part held by the gripper must be ensured as it affects the maximum acceleration of the gripper and results in loss of the gripper power.
5. The simplest gripper that meets the first four criteria should be the one that should be implemented.
   Most commonly used mechanical finger grippers, can be angular or parallel, are listed as; two fingers grippers, external or internal gripping. Three fingers grippers, simulates the action of thumb, index finger and third finger. Four finger grippers, grasp square and rectangular parts easily.

The two fingers gripper is chosen in this work due to its simplicity and adequacy for our applications. The gripper is connected to the arm wrist link the free end of the arm. The linear motion of the fifth cylinder piston is used to open or close the gripper as shown in Fig. 3 below. The gripper is rotated around its axis using a geared stepper motor.



Opened Gripper **Closed Gripper**

Figure 3 Mechanism of Opening-closing the gripper

III. KINEMATICS ANALYSIS FOR THE PRESENT ARM GEOMETRICAL CONFIGURATION

Kinematics is the analytical study of the geometry of motion of a robot, with respect to a fixed reference coordinates system, without taking into consideration the forces or moments that cause the motion.
In order to be able to trajectory control the present hydraulic arm, kinematics study for the present robotic arm is carried out as follows; Cartesian and polar coordinates, x, y, z, and ϕ and, θ_0, θ_1, θ_2, and θ_3 of the present arm are selected as shown in Fig. 4 below.
Where: θ_0 ; is the rotational angle of the base, θ_1 ; is the rotational angle of the shoulder bar, θ_2 ; is the rotational angle of the elbow bar, and θ_3 ; is the rotational angle of the wrist bar, ϕ ; is the angle of attack of the end-effectors.

l_1; is the shoulder bar length, 57.7 cm, l_2; is the elbow bar length, 61.7 cm, and l_3; is the wrist bar length, 45.0 cm.

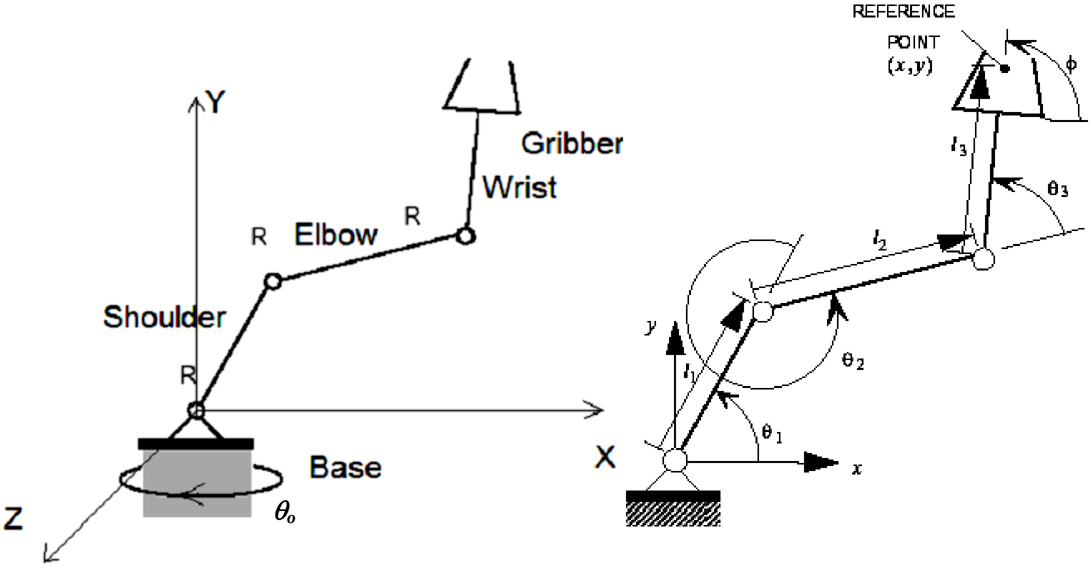


Figure 4 Hydraulic arm Cartesian and polar coordinates

3.1 Arm home position

The home position of the present robotic arm is defined as the position at which all the hydraulic cylinders are fully retracted. Defining the coordinates of vertical plan as X and Y and those of the horizontal plant as *X* and *Z*, the home position is shown in Fig 5 below. When all the hydraulic cylinders are fully extended the maximum rotational angular motions, relative to the corresponding home position are 80° for each of the base, and the shoulder and the elbow bars, while it is equal to 60° for the wrist bar. Angular motion is given a positive sign for counter clock wise rotation, CCW, and a negative sign for clock wise rotation, CW. Values for the home and maximum angular positions are listed in Table 1 below.

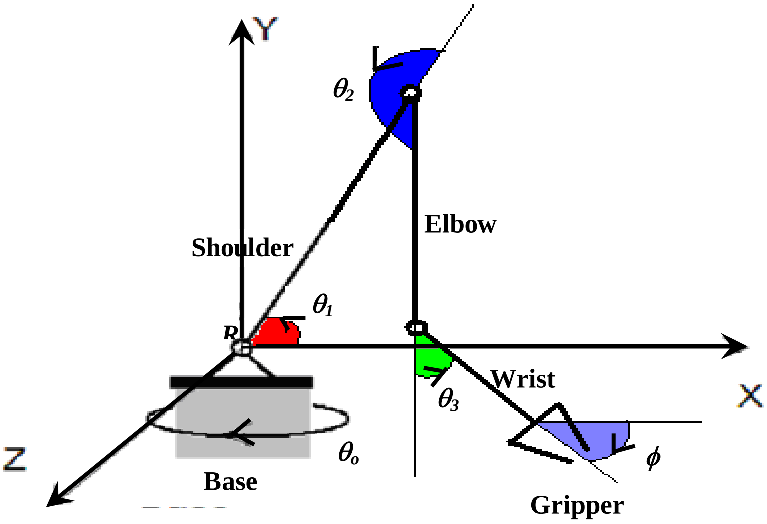


Figure 5 Home position of the present robotic arm

Table 1: Home and maximum angular positions of the present arm links

Link angle	***Home angle, (o)***	***Maximum available angle, (o)***
Base, θ_0	-55	25
Shoulder, θ_1	47	127
Elbow, θ_2	223	303
Wrist, θ_3	30	90

Kinematics is classified into two main categorizes, namely, forward and inverse kinematics. In forward kinematics, conversion from polar to Cartesian coordinates are carried out, while Cartesian to polar coordinates conversion is achieved by applying the inverse kinematic analysis. Derivation of both kinds' formulas for the present arm geometrical configuration is explained as follows:

3.2 Forward kinematics formulas

Forward kinematics formulas calculate unknown values for x, y, z, and ϕ from known values for the length of each of the arm parts, l_1, l_2, and l_3, and θ_0, θ_1, θ_2, and θ_3. Very simple formulas are derived for the position coordinates, x, y, z, and the angle of attack, ϕ as follows:

$$x = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3)$$

(1)

$$y = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3)$$

(2)

$$z = x \sin\theta_0$$

(3)

$$\phi = \theta_1 + \theta_2 + \theta_3$$

(4)

3.3 Inverse Kinematics formulas

Formulas of inverse kinematics are used to calculate unknown polar coordinates, θ_0, θ_1, θ_2, and θ_3 values, given known values for l_1, l_2, l_3, x, y, z, and ϕ. Very simple formula for θ_0 is presented below.

$$\theta_0 = \sin^{-1}\frac{z}{x} \tag{5}$$

Formulas for θ_1, θ_2, and θ_3 are not so easy to be derived in a single step and need to be performed through a sequence of derivation steps as shown below:

Step 1: Rearranging the Forward Kinematics formulas, (1) and (2)

$$x - l_3\cos(\phi) = l_1\cos\theta_1 + l_2\cos(\theta_1+\theta_2) \tag{6}$$

$$y - l_3\sin(\phi) = l_1\sin\theta_1 + l_2\sin(\theta_1+\theta_2) \tag{7}$$

Step 2: Recasting (6) and (7) in one formula

Defining, $x^{'}$ and $y^{'}$ as follows:

$$x' = x - l_3\cos(\phi) \tag{8}$$

$$y' = y - l_3\sin(\phi) \tag{9}$$

(6) and (7) can be written as:

$$x' - l_1\cos\theta_1 = l_2\cos(\theta_1+\theta_2) \tag{10}$$

$$y' - l_1\sin\theta_1 = l_2\sin(\theta_1+\theta_2) \tag{11}$$

Squaring and adding (10) and (11) gives:

$$(-2l_1x')\cos\theta_1 + (-2l_1y')\sin\theta_1 + \left(x'^2 + y'^2 + l_1^2 + l_2^2\right) = 0 \tag{12}$$

Step 3: Solving (12) to get θ$_1$

Defining variables *P*, *Q*, and *R* as:

$$P = -2l_1x', \qquad Q = -2l_1y', and \qquad R = x'^2 + y'^2 + l_1^2 + l_2^2 \tag{13}$$

Form of (12) is simplified to:

$$P\cos\theta_1 + Q\sin\theta_1 + R = 0 \tag{14}$$

To solve (14) in θ_1, γ is defined as:

$$\gamma = atan2\left[\frac{Q}{\sqrt{P^2+Q^2}}, \frac{P}{\sqrt{P^2+Q^2}}\right] \tag{15}$$

Using (15), (14) can be rewritten as

$$\cos\gamma\ \cos\theta_1 + \sin\gamma\ \sin\theta_1 + \frac{R}{\sqrt{P^2+Q^2}} = 0 \tag{16}$$

Using triangle relation gives:

$$\cos(\theta_1 - \gamma) = \frac{-R}{\sqrt{P^2+Q^2}} \tag{17}$$

And thus formula for θ_1 is obtained as:

$$\theta_1 = \gamma + \sigma \cos^{-1}\left(\frac{-R}{\sqrt{P^2+Q^2}}\right), where \quad \sigma = \pm 1 \tag{18}$$

θ_1 therefore has two solutions and thus θ_2 should also has a corresponding couple of solutions so that the summation of θ_1 and θ_2 gives give the same (x, y) coordinates for the reference point. This result is shown in Fig. 6 below.

Step 4: Deriving formulas for θ_2 *and* θ_3

Using (10) and (11), it is easy to get θ_2 formula as:

$$\theta_2 = atan2\left[\frac{y' - l_1 \sin\theta_1}{l_2}, \frac{x' - l_1 \cos\theta_1}{l_2}\right] - \theta_1 \tag{19}$$

θ_3 is simply calculated using θ_1, θ_2, and ϕ values using the relation:

$$\theta_3 = \phi - (\theta_1 + \theta_2) \tag{20}$$

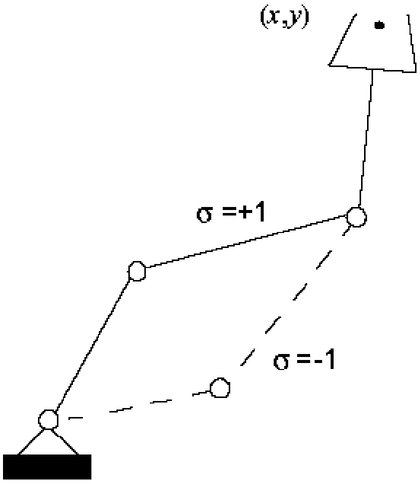


Figure 6 Two solutions for θ_1 and θ_2 give the same (x,y) coordinates for the reference point

IV. PRESENT TRAJECTORY CONTROL SYSTEM

The present control system hardware consists of feedback sensors, to measure the rotational angle of each of the arm parts, and the hydraulic system control circuit. A control software program was designed and constructed by the author to read the actual rotational angles and to decide and send appropriate control action to the hydraulic cylinder actuators to stop/rotate, in the proper direction, each of the arm parts. The control soft and hard wares are connected to each other using a data acquisition card, DAC, attached to a host computer. Limit switches are attached to the arm parts and automatically operated by the control program to ensure safe arm operation.

The robotic arm is driven by five hydraulic cylinders to rotate its parts, base, shoulder, elbow and wrist bars, around their hinges and to open and close the fingers of its grippers. The linear motion direction of the arm hydraulic cylinders are controlled using 4/3 flow control valves actuated by electric solenoids from both sides. The cylinders can be therefore moved forward, extended, or backward, retracted, or stopped. Simple mechanical mechanisms are used to convert the cylinder linear motions to rotational movements of limited swing angles. A schematic diagram for the present hydraulic circuit contains two cylinders as a sample is shown in Fig. 7 below.

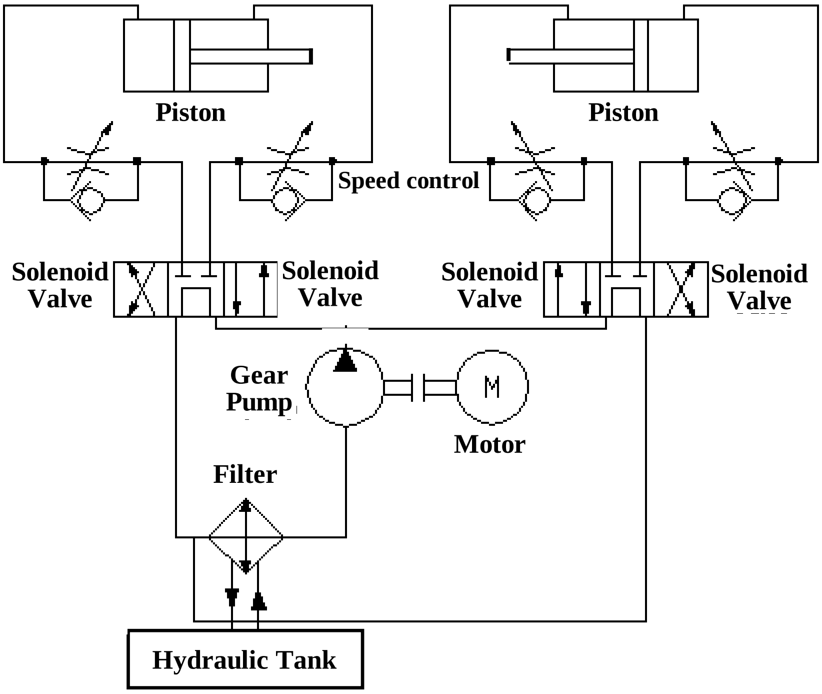


Figure 7 Sample of the present hydraulic circuit contains two cylinders

Electronic circuit was designed and implemented by the author to control the valves solenoids using digital voltage signals decided by the control software. Signals for rotating the arm parts in clock wise, CW, or counter clock wise, CCW, directions are sent by the control program to trace the required object position. Fig. 8 below shows a schematic diagram for a circuit driver to control one solenoid as an example. The cylinders driver circuit contains ten units of that shown in Fig. 8. A digital signal decided by the control program and is sent to the circuit, through the data acquisition card, DAC, to turn on or off the solenoid. To enable an individual cylinder piston to advance, the right solenoid of the 4/3 directional valve is enabled while the left one is disabled. To enable retract motion the action that mentioned in the previous sentence is reversed. Both solenoids are disabled to stop moving the cylinder piston. It is worth noting that turning any of the arm elements in CW or CCW direction is corresponding to move the relevant cylinder piston in advance of retract motion respectively.

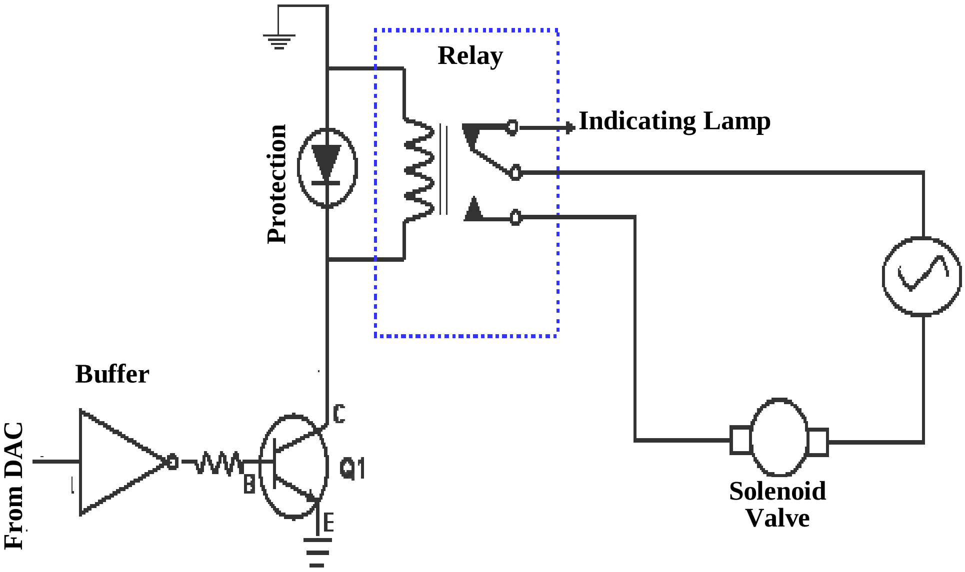


Figure 8 Electronic driver circuit of one solenoid valve, DAC: data acquisition card

Reaching the target position is achieved through implementing pre-decided individual rotational tasks for each of the arm parts. Control soft ware sends command to immediately stop moving the arm part as soon as it finishes its individual task.

The present control software program was designed using C# programming language. It is constructed to be used as a Graphical User Interface application, GUI, so it can be operated by un-trained persons. Users just simply choose the control mode and inter the corresponding tracking data, even in Polar or in Cartesian coordinates to achieve the arm pick and place tracking task.

Inverse kinematics formulas, derived by the author, are processed by the program to convert the coordinates of the start and end tracking points from Cartesian to Polar systems as needed. The program is prepared to do the calibration process, which results in attaining the off line prepared lookup tables, and to construct the specific learning procedures as well.

The control system is designed to be operated in the following control modes:

1- Feedback control mode:

Four optical encoders with a very high resolution, of 0.044 degree, are mounted to the base and to the other four hinged arm links, to measure their angle of rotations and feed them back to the control soft ware program. The required rotational angle of each link is calculated as the difference between the relevant required target coordinate and the corresponding initial position coordinates, $\Delta\theta_{target}$. The software sends commands to rotate all of the arm links in CCW or CW directions according to the sign of the corresponding required $\Delta\theta_{target}$, positive or negative respectively. Data for the actual increase, in each of the arm links angular position, obtained from reading the relevant encoder signal, is compared to the corresponding required $\Delta\theta_{target}$. When the actual and the target $\Delta\theta$ becomes of equal values for any of the arm parts, the software stops the motion of this part immediately. The flow chart of the feedback control program is shown in Fig. 9, presented in the next page.

2- Open loop control mode:

The open loop control does not use the encoders' signals for the current angular position. It uses, instead, lockup tables that are off line prepared tables for the angle of rotations of each arm part via the corresponding time consumed. The lookup tables are prepared by performing the calibration process.

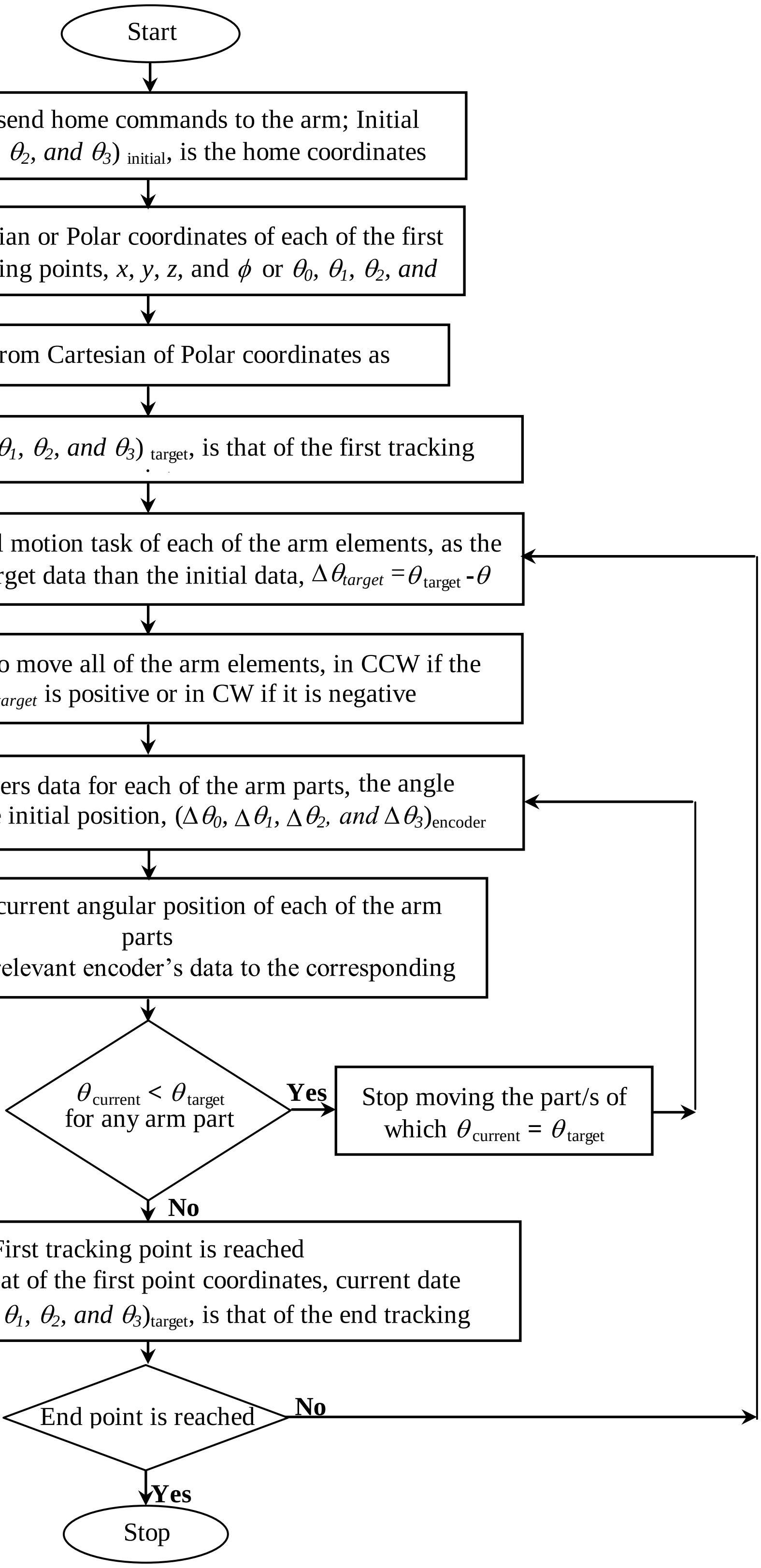


Figure 9 Flow chart of the present feedback tracking control system

Details of calibration process:

Since for any angular movement, the moving angle is related to the moving time interval corresponding to the angular speed of the link. Calibration is performed for arm links, link after link.

The software sends a command to move an individual link starting from its home position and ending at its maximum allowable position, presented in table I above. The software stores readings for the link rotational angle increments of about 0.25 degree, obtained by reading its relevant encoder, against the corresponding elapsed time interval. The calibration process of this link is repeated for five times. The mean values of these five trials are taken as the final lookup table of the calibrated link. The same procedure is repeated for the other three links to get lookup tables of all the arm links. A graph for the look up data is shown in Fig. 10 below.

The software decides the rotational time interval and motion direction required for each of the arm parts corresponding to the required $\Delta\theta$, sign and value. This is achieved using the lookup tables, offline pre prepared and stored in the computer memory, as mentioned above, interpolation are done as needed. The software sends commands to move each of the arm links, in the proper moving direction during the time interval picked up from the lookup tables corresponding to required $\Delta\theta$.

This executed for link after another since the calibration was done for only one link moving at a time. The software stops the motion of the part immediately after the required time interval is elapsed. If more than one part is moved together their speed will be slower than that were available at the calibration time, and therefore the calibration data will be misleading of no use.

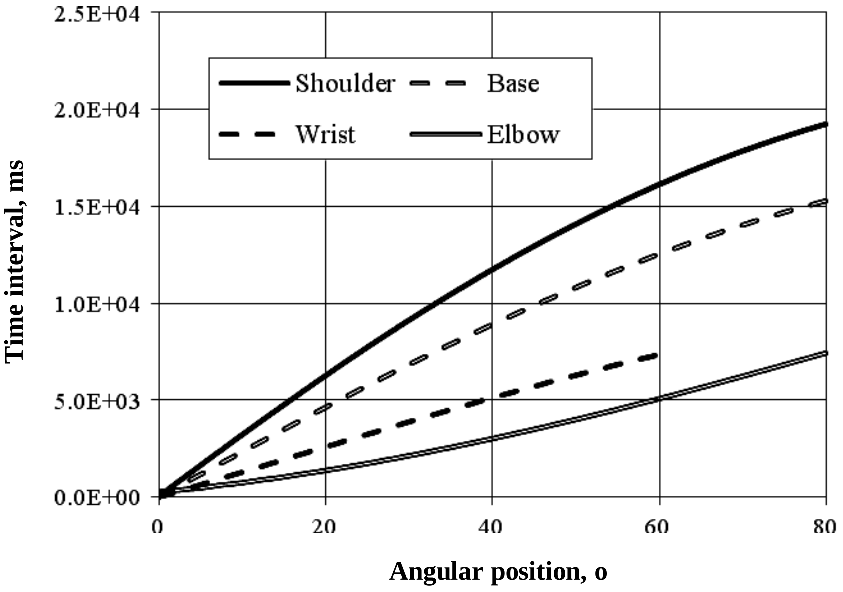


Figure 10 Lookup data for all the arm links

3- Pre-learned control mode:

Learning mode is used to achieve highly repeatable pick and place tasks. The control program uses a stored procedure as a sequence of points in the working space, according to which the arm goes one after another. The procedure is stored as a series of polar coordinates, θ_0, θ_1, θ_2, *and* θ_3, while the arm was feedback controlled to move starting from pick up to place tracking points. The soft ware sends commands to move the relevant arm parts to execute the stored path while running the feedback or the open loop control modes.

V. EXPERIMENTAL RESULTS

Meany experimental runs were carried out to check the present arm operation under different control modes. Experiments were performed using the graphical user interface soft ware, prepared to control the present robotic arm, as follows;

Note: positive and negative signs are assigned to counter clock wise and clock wise rotational angle respectively.

1- Click the HOMING button of the GUI application to move the robotic arm to its home position, the end of retract strokes of all the arm cylinders, $\theta_0 = -55$, $\theta_1 = 47$, $\theta_2 = 223$, and $\theta_3 = 30$ in polar coordinates.

2- Choose the control mode from a list contains: feedback control, or pre-learned control, or open loop control modes.

3- Inter the Cartesian or the polar coordinates of the first and end tracking points, pick up and place points respectively. The control software, thus, performs a sequence of operations as follows:

a. Converts from Cartesian to polar coordinates, if needed, using the inverse kinematic formulas derived for the present arm mechanical configuration. Values for θ_0, θ_1, θ_2, and θ_3 for both the first and end points are now known.
b. Sends a command to open the gripper.
c. Calculate and send commands to execute the angular motion needed for each of the arm parts, $\Delta\theta_0$, $\Delta\theta_1$, $\Delta\theta_2$, and $\Delta\theta_3$ to reach the pickup point starting from its home position.
d. Sends a command to close the gripper to catch the experimental object.
e. Calculate and execute the angular motion needed for each of the arm parts, $\Delta\theta_0$, $\Delta\theta_1$, $\Delta\theta_2$, and $\Delta\theta_3$ to reach the object release point starting from the pickup position.
f. Sends a command to open the gripper to release the experimental object at last.

Figure 11 Experiment 2, pick up the screw driver

Figure 12 Experiment 2, place the screw driver inside a bottle of small diameter hole

Samples of the carried out experiments are presented as follows:

Experiment 1:

It is carried out to pick up a large object, big size book, from a start point of $x_1 = 70$ cm, $y_1 = 5$ cm, $z_1 = -10$ cm and $\phi_1 = 30^o$ to release it in the center of a wide plastic pot at an end point of $x_2 = 70$ cm, $y_2 = 15$ cm, $z_2 = -50$ cm and $\phi_2 = 30^o$.

a. The experiment has been tried for many times without any noticeable error when applying the feedback control mode.
b. Open loop control mode results in an error of ± 0.5o in each of the arm parts movements. This sum up to a small error relative to this easy pick and place task. The release point is not so far from the pot center.
c. Results of applying the pre-learned control mode were found to be better than the open loop mode with respect of releasing the book very near the pot center.

Experiment 2:

This experiment was carried out to achieve a hard pick and place task. A thin screw driver with a diameter 5 mm is held to be placed inside a water bottle of a small opening hole of 15 mm diameter. The coordinates of pick up and place positions are $x_1 = 80$ cm, $y_1 = 3$ cm, $z_1 = -20$ cm and $\phi_1 = 30^o$ and $x_2 = 80$ cm, $y_2 = 30$ cm, $z_2 = -60$ cm and $\phi_2 = 30^o$ respectively.

a. Applying the feedback control mode, the experiment was repeated ten times, nine of which were successful. The unsuccessful trial is most probably to human error in putting the driver or the bottle in their proper position as it is given to the software. Thus this is a good percentage pointing out the high accuracy of the present tracking system to achieve such a hard pick and place task.
b. The error associated with applying the open loop control mode is considered large here and therefore only five of ten trials were successful.
c. Results of applying the pre-learned control mode, based on feedback sensor signals, were found to be almost the same as that of the feedback control mode. This is because that feedback signals are still be used.

A successful trial of experiment 2, is shown in photos 10 and 11 present in the previous page.

VI. CONCLUSION

A PC based accurate tracking control for hydraulic robotic arms is designed and implemented in the present work. A pick and place arm manipulator is designed and constructed as a four bar mechanism mounted on a rotatable base and driven by a hydraulic system.

The present control system hardware consists of feedback sensors, to measure the rotational angle of the arm links, and an electronic driver circuit to control its hydraulic system. A control software program was designed and constructed by the author to read the actual rotational angles of the arm parts and to decide and send appropriate control actions to the its hydraulic driver circuit. The control soft ware is connected to the arm using a data acquisition card attached to a host computer.

Inverse kinematics formulas, derived by the author, are processed by the software program to convert the coordinates of the initial and object tracking points from Cartesian to Polar systems when needed. The present control system is designed to be operated as a feedback control, or an open loop control, using offline prepared lookup tables, or executing a pre-decided stored sequence of points in the working space in the learning mode.

Experimental runs were carried out to verify the effectiveness and the accuracy of the present tracking arm. Conclusions of these experiments results are summarized as follows:

1- Experiments, to pick up a big book object from a predefined position to place it in a wide pot in another end position, have been tried for many times without any noticeable error.
2- Experiments for accurate pick and place tasks were carried out to catch a thin screw driver with a diameter 5 mm to place it inside a water bottle of a small opening hole of 15 mm diameter. The experiment was repeated ten times, nine of which were successful, which is a good percentage referring to the high accuracy of the present tracking system.
3- The open loop control mode and open loop based learning modes were experimentally examined and found to be successful but, as expected, they are of less accuracy than the feedback control mode.
4- Lookup tables need to be updated from time to time due to the unsteady operating conditions such as the oil viscosity, pump discharge, oil pressure, and mechanical friction. These conditions have a great effect on the arm parts speed and therefore on the time consumed to reach certain rotational angles.

REFERENCES

Journal Papers:

[1] R.J. Wang, J.W. Zhang, et al., Multiple-Function Intelligent Robotic Arms, FUZZ-IEEE Journal, Korea, 20-24, 2009, 1995-2000.

[2] Jihua Zhu, Nanning Zheng and Zejian Yuan, An improved technique for robot global localization in indoor environments, International Journal of Advanced Robotic Systems, Vol. 8, No. 1,ISSN 1729-8806, 2011, 21-28.

[3] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, Robust Monte Carlo Localization for Mobile Robots, Artificial Intelligence, Vol. 128, No. 1-2, , ISSN : 11076-9757, 2001, pp. 99-141.

[4] D. Fox, Adapting the sample size in particle filters through KLD-sampling, International Journal of Robotics Research IJRR, Vol. 22, No. 12, ISSN: 0278-3649, 2003, 985-1003.

Books:

[5] Manipulating industrial robots vocabulary (International Organization for Standardization Standard 8373, 1994).

[6] Industrial and service robots (IFR International Federation of Robotics, http://www.ifr.org/home, 2010).

[7] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics, (MIT Press, London, ISBN 0-262-20162-3, 2005).

Proceedings Papers:

[8] L.B. Duc, M. Syaifuddin, et al., Designing 8 Degrees of Freedom Humanoid Robotic Arm, International Conference on Intelligent and Advanced Systems, Kuala Lumpur, 2007, 1069-1074.

[9] A. Milstein, J.N. Sánchez, and E.T. Williamson, Robust Global Localization Using Clustered Particle Filtering, Proc. of the 18th National Conference on Artificial Intelligence, Alberta, Canada, 2002, 581-586.

[10] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, Monte Carlo localization for mobile robots, Proc. of the IEEE International Conference on Robotics and Automation, Michigan, 1999, 1322–1328.

[11] N. Freitas, Rao-Blackwellised particle filtering for fault diagnosis, Proc. IEEE Aerospace Conference, Vol, 4, ISBN: 0-7803-7231-X, 2002, 1767-1772.

[12] A. Doucet, A. Freitas, K. Murphy, and S. Russel, Rao-Blackwellized particle filtering for dynamic Bayesian networks, Proc. of the Conference Uncertainty Artificial Intelligence, Stanford, CA, 2000, 176–183.