Real time controlled two dof five bar robot manipulator

Fatih Cemal CAN, Hayrettin ŞEN
Izmir Katip Çelebi University
Faculty of Engineering and Architecture
Mechatronics Engineering Department
E-mail: fatihcemal.can@ikc.edu.tr

Abstract

In this research, computer controlled two DoF five bar robot manipulator is investigated. In order to control manipulator, a human machine interface program is developed in Visual C# after completing inverse kinematic analysis of robot manipulator. By the help of inverse kinematics, this program calculates two joint variables for given positions of end point. Then the program sends a data package containing these joint variables to Arduino microcontroller. Arduino microcontroller set the positions of two servos according to calculated joint angles. Also using standard geometries, robot can follow trajectory a line, a circle and a rectangle. Furthermore, a lot of patterns can be generated using function with variable radius and angle of rotation.

Keywords: Two Dof Robot, Five bar, Arduino, Visual C#

1. Introduction

The five-bar linkage mechanism applications have been used in different engineering fields. Researcher working in mechatronics, biomedical, mechanical and electrical and electronics engineering fields designed and implemented the five bar mechanism in their published investigations. Inverse kinematics, link design, medical application, dynamic simulation, calibration and performance were topics of research interest on the five-bar linkage mechanism. Some studies about these topics are presented as follows.

As a biomedical engineering application, a laparoscopic robotic camera system based on five-bar linkage was designed and tested by Kobayashi and et al. [1]. Thisrobotic system reduce the process time for different surgical operations if human assisted camera system is compared to robot assisted the camera system.

Hybrid five-bar mechanism was investigated by researchers [2, 3]. A dynamic simulation and control of these kinds of mechanism was carried out in Simmechanics of Matlab by Zi and et al. [3]. The five-bar was driven by a constant velocity motor and servo motor in this study. The aim of the study was to control tracking trajectory of the end point of the mechanism via Traditional PD control and closed loop PD-type iterative learning control.

Inverse kinematics analysis of six different five-bar planar parallel manipulators which were RRRRR, RRRRP, PRRRP, RPRPR, RRRPR and RPRRP was presented by Alıcı[4]. Sylvester elimination method was used to solve the set of nonlinear equations in his study. Villarreal-Cervantesand et al. [5] optimized design parameters of a five-bar parallel robot by using a novel mechatronic design approach. They designed kinematic and dynamic parameters of the five-bar's links with respect to desired trajectories [5, 6].

A position accuracy calibration of a five-bar planar parallel robot (DexTAR) was established by Joubair and et al. [7]. Experimental validation setup was constructed and the position error reduced to 0.08 mm with in the entire robot workspace of 600x600 mm.

A real-time control of a five-bar parallel robot (DexTAR) was studied using dynamic model of the robot to implement minimum time trajectory planning by Bourbonnais and et al. [8]. They used working mode region to reach points of pick and place operations. Several working modes are considered to reach same pick and place points in their study.

In this study, a five-bar linkage was designed and two links of the mechanism were produced by using 3D printer. In section 2, the inverse kinematics analysis of five-bar manipulator will be presented. In the next section human machine interface design and control algorithm will be explained. In section four electronic hardware and circuit are illustrated along with figures. Then, experimental test results will be presented in section five. Finally, a conclusion will be drawn at the end of the study.

2. Inverse Kinematic Analysis of the Robot Manipulator

Five bar mechanism has two degrees of freedom according to Grubler formulation. In inverse kinematics, end point location (P) is known in Fig. 1. Using this location, two input angles θ_1 and θ_2 must be calculated. This problem can be solved by dividing manipulator into two serial RR manipulators.

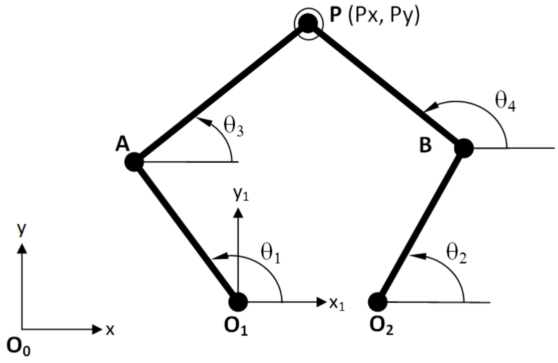


Fig. 1. Two DoF Five Bar Robot

From the first serial kinematic chain, vector loop closure equation (1) is as follows,

$$\mathrm{O_0O_1} + \mathrm{O_1A} + \mathrm{AP} = \mathrm{O_0P} \qquad (1)$$

Second, the second serial kinematic chain, vector loop closure equation (2) is as follows,

$$\mathrm{O_0O_2} + \mathrm{O_2B} + \mathrm{BP} = \mathrm{O_0P} \qquad (2)$$

In this study, user of the robot manipulator changes O_0P vector in human machine interface. However these equations are dependent on each joint angle related to kinematic chain. In order to compute input angles, other joint variables must be eliminated.

$$d_{x1} + i\,d_{y1} + L_1e^{i\,\theta_1} + L_3e^{i\,\theta_3} = P_x + i\,P_y \qquad (3)$$

$$d_{x2} + i\,d_{y2} + L_2e^{i\,\theta_2} + L_4e^{i\,\theta_4} = P_x + i\,P_y \qquad (4)$$

Rearranging equation (3), angle which is need to be eliminated is kept alone,

$$L_3e^{i\,\theta_3} = P_x + i\,P_y - d_{x1} - i\,d_{y1} - L_1e^{i\,\theta_1} \qquad (5)$$

The offset distances in equation (5) can be combined with the end coordinates as follows,

$$L_3 e^{i\,\theta_3} = (P_x - d_{x1}) + i\big(P_y - d_{y1}\big) - L_1 e^{i\,\theta_1} \tag{6}$$

Conjugate of the equation (6) is written as follows,

$$L_3 e^{-i\,\theta_3} = (P_x - d_{x1}) - i\big(P_y - d_{y1}\big) - L_1 e^{-i\,\theta_1} \tag{7}$$

Multiplying equations (6) and (7), angle θ_3 is able to be eliminated.

$$\begin{aligned} L_3^2 = (P_x - d_{x1})^2 + \big(P_y - d_{y1}\big)^2 + L_1^2 \\ - (P_x - d_{x1})L_1\big(e^{i\,\theta_1} + e^{-i\,\theta_1}\big) \\ + \big(P_y - d_{y1}\big)L_1\, i\,\big(e^{i\,\theta_1} - e^{-i\,\theta_1}\big) \end{aligned} \tag{8}$$

Remember that $\big(e^{i\,\theta_1} + e^{-i\,\theta_1}\big)$equals to $2\cos(\theta_1)$ and $i\big(e^{i\,\theta_1} - e^{-i\,\theta_1}\big)$ equals to $2\sin(\theta_1)$ in equation (8). The equation can be simplified using these equalities as follows,

$$A\cos(\theta_1) + B\sin(\theta_1) + C = 0 \tag{9}$$

where $A = -2\,(P_x - d_{x1})L_1$, $B = -2\,\big(P_y - d_{y1}\big)L_1$ and $C = (P_x - d_{x1})^2 + \big(P_y - d_{y1}\big)^2 + L_1^2 - L_3^2$. Half-tangent rule can be used for this equation in order to solve one unknown $(\cos(\theta_1) = \frac{1-t_1^2}{1+t_1^2}, \sin(\theta_1) = \frac{2\,t_1}{1+t_1^2}andt_1 =$ atan$(\frac{\theta_1}{2})$).

$$(C - A)t_1^2 + 2\,B\,t_1 + (C + A) = 0 \tag{10}$$

Two different roots are able to be obtained from the equation. Discriminant must be real for physically realizable result.

$$dis1 = \sqrt{4B^2 - 4(C^2 - A^2)} \tag{11}$$

One unknown t_1 is calculated using discriminant.

$$t_1 = \frac{(-4\,B \mp dis1)}{2(C - A)} \tag{12}$$

One can compute the angle using half tangent value.

$$\theta_1 = 2\tan(t_1) \tag{13}$$

Once θ_1 is known, other angle of kinematic chain is calculated as follows,

$$\begin{aligned} \theta_3 = atan2(P_x - d_{x1} - L_1\cos(\theta_1)\,, P_y - d_{y1} \\ - L_1\sin(\theta_1)) \end{aligned} \tag{14}$$

Similarly, applying same process to the second kinematic chain, we can obtain the discriminant as follows,

$$dis2 = \sqrt{4E^2 - 4(F^2 - D^2)} \tag{15}$$

where $D = -2\,(P_x - d_{x2})L_2$, $E = -2\,\big(P_y - d_{y2}\big)L_2$ and $F = (P_x - d_{x2})^2 + \big(P_y - d_{y2}\big)^2 + L_2^2 - L_4^2$.

Similar to pervious solution, unknown of the second kinematic chain is now known using equation as follows,

$$t_2 = \frac{(-4\,E \mp dis2)}{2(F - D)} \tag{15}$$

$$\theta_2 = 2\tan(t_2) \tag{16}$$

$$\begin{aligned} \theta_4 = atan2(P_x - d_{x2} - L_2\cos(\theta_2)\,, P_y - d_{y2} \\ - L_2\sin(\theta_2)\,) \end{aligned} \tag{17}$$

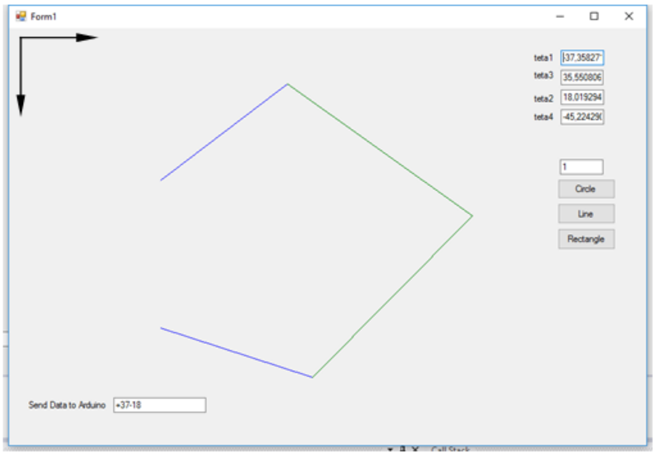
Two solutions are obtained for each angle. Totally four modes can be found for the robot. But, the robot is working on just one mode which is shown in Fig. 1.

3. Visual C# Interface Design and Program for Human - Machine Interaction

The program of the robot is constructed on kinematic analysis of the robot. Two coordinates of the end point location are inputs for this program. User is able to use mouse cursor in order to define end point location of the robot. User must click mouse left button on the Form then he must move cursor wherever he wants. Program solves angles according to end points and redraws lines of the robot.

Human machine interface consists of two parts. The first part is the interface design of the program. This part is related to graphical drawing of the robot, angles in text boxes and control buttons and output of the program. The design of the program is illustrated in Fig. 2.

The second part is algorithm or coding. Follow chart of the visual program is depicted in Fig. 3. This code is written using events of graphical objects such as buttons, lines, form and buttons. This kind of programming is named event based programming. We use load, mouse down, mouse move and mouse up events for our code.



teta1: 37,35827
teta3: 35,550806
teta2: 18,019294
teta4: -45,22429(
1
Send Data to Arduino: +37-18

Fig. 2. Human Machine Interface in Visual C#

Start: Form1_Load event is working. In this event, link lengths and other constant parameters are defined.

Input: Form1_Mousedown, Form1_MouseMove and Form1_MouseUp events are working.

Calculate: dis1 and dis2 are calculated according to inverse kinematic analysis of the robot.

Are roots of dis1 and dis2 real? Program must decide that the results are physically realizable or not. Non real results cannot be used in real environment. Therefore, they are unnecessary.

Calculate:If roots are real, angles are calculated using Equations (13, 14, 16 and 17).

Prepare and send data packages to Arduino: When two angles are calculated, they are packaged in order for sending. The data package includes plus and minus symbols which show the direction of the rotation, and two digit numbers which indicate magnitude of the rotation.

One Data Package

+	9	0	-	4	5

Is Arduino serial connection OK?: In order to send packages to Arduino, COM port number of Arduino must be valid and connected. Baud rate of the connection must be selected correctly.

Is close of Form clicked?: The program runs until close of form is clicked. The program ends if it is clicked.

The flow chart of Arduino program is illustrated in Figure 4. This code waits for the input from interface program, separate according to data package format and sends these to servo motors.

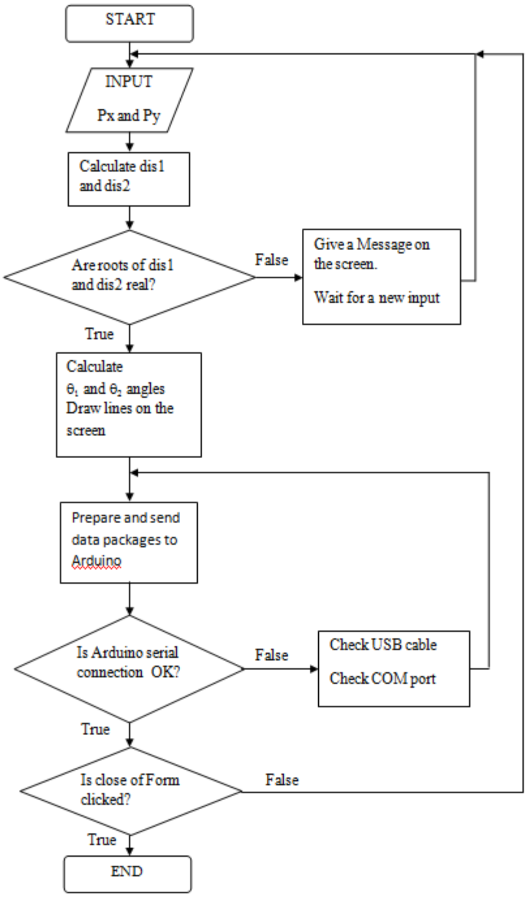

START
INPUT
Px and Py
Calculate dis1 and dis2
Are roots of dis1 and dis2 real?
False
Give a Message on the screen.
Wait for a new input
True
Calculate θ1 and θ2 angles Draw lines on the screen
Prepare and send data packages to Arduino
Is Arduino serial connection OK?
False
Check USB cable
Check COM port
True
Is close of Form clicked?
False
True
END

Fig. 3. Flow Chart of Visual C# Human Machine Interface (HMI) Program

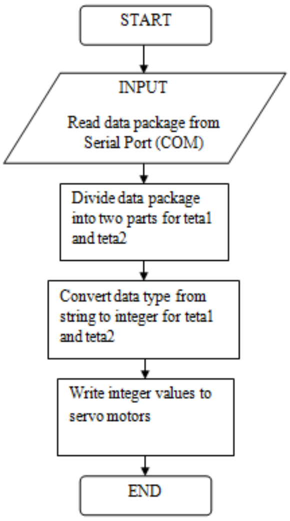

START
INPUT
Read data package from Serial Port (COM)
Divide data package into two parts for teta1 and teta2
Convert data type from string to integer for teta1 and teta2
Write integer values to servo motors
END

Fig. 4. Flow Chart of Arduino Program

4. Electronic Hardware and Circuit

Arduino Mega 1280 Micro Controller is used to control the robot. The robot has two servos connected to digital pins 9 and 10 which are used as PWM (pulse width modulation) ports of Arduino. The electronic circuit of the robot is seen in Figure 5. This circuit is created in Autodesk Circuit Online Software. Arduino Uno is used in the circuit because Arduino mega is not available in this software. One 5V DC power source must be used to supply required power to servo motors. The Arduino code can be simulated in the program. Simulation can be seen in Figure 6.

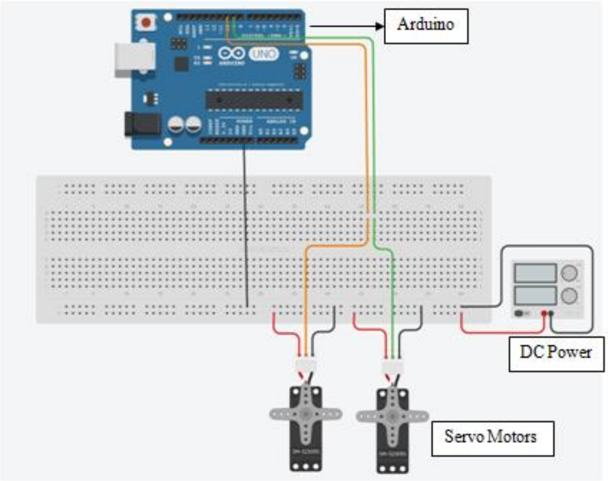
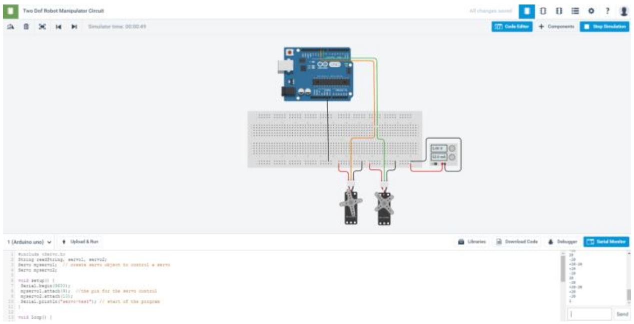


Fig. 5. Electronic Circuit in Autodesk Circuits



```
#include <Servo.h>
String readString, servo1, servo2;
Servo myservo1;  // create servo object to control a servo
Servo myservo2;

void setup() {
 Serial.begin(9600);
 myservo1.attach(9);  //the pin for the servo control
 myservo2.attach(10);
 Serial.println("servo-test"); // start of the program
}

void loop() {
```

Fig. 6. Simulation in Autodesk Circuits

5. The robot construction and Test Results

The robot is constructed using two Emax ES 3005 water proof servo motors and 3-D printed parts. Two link lengths $L_1$and L_2 are chosen same (20 mm) because they are servo links packaged in servo motor box. Other two link lengths L_3 and L_4 are selected to be same (30 mm). Distance between shafts of two servo motors is 19 mm.

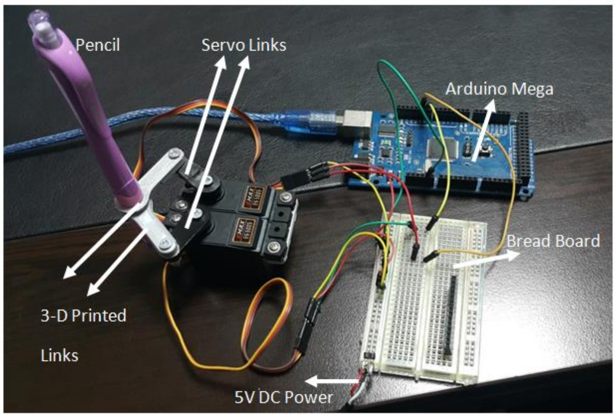


Fig. 7. Electrical connection and whole setup

5.1. Drawing a line and a rectangle

These two shapes are the most basic task to test our robot. Simply, pencil translates forward and backward between two points to draw a line and four lines are used to draw a rectangle (Fig. 8.). We defined the length of the line as 20 mm in the program. The straightness of the line was good but the length of the line was nearly 19.20 mm. This dimension error depends on joint clearances and friction at the joints. Next, the rectangle was drawn. Fillets are created by our robot at the corner. This rectangle was not perfect dimensioned similar to the line. But straightness and appearance of the rectangle is acceptable.

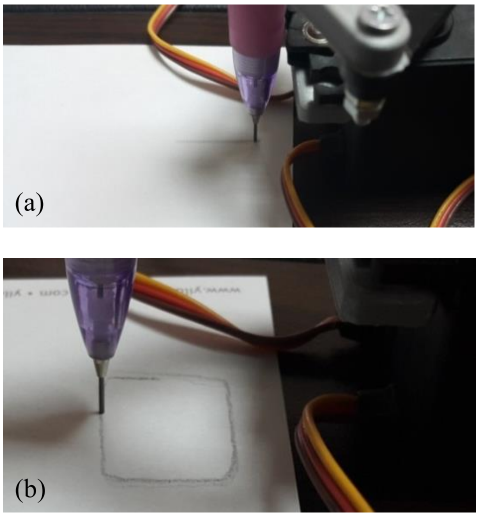
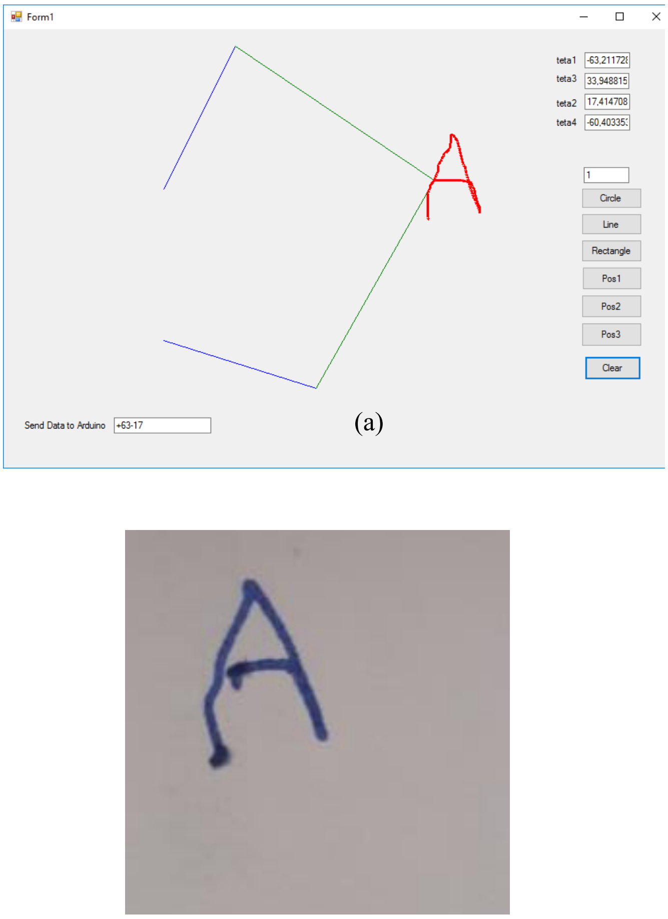


Fig. 8. Drawing Test (a) Line Drawing (b) Rectangle drawing

5.2. Computer Aided Free Hand Drawing

We tried to test our robot manipulator using human machine interface (HMI). User entered required points from HMI, points were sent to microcontroller and microcontroller set angles of two servo motors. Acceptable outputs were obtained as seen from Figures 9 and 10. We firstly tried to draw a letter and then a spiral. The letter and the shape are very close to original shape which is drawn on HMI.

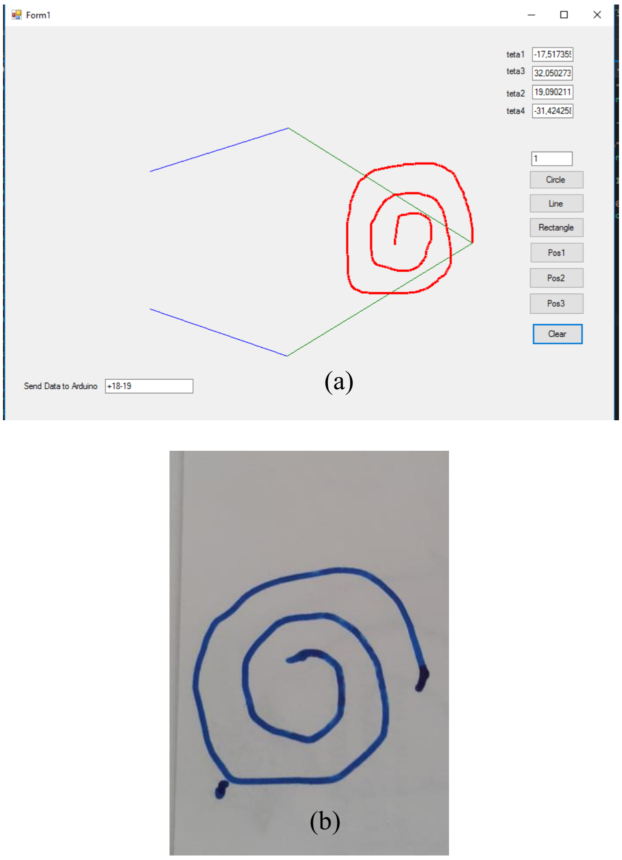


teta1 -63,21172
teta3 33,948815
teta2 17,414708
teta4 -60,40335
1
Send Data to Arduino +63-17

(a)

(b)

Fig. 9. Free hand drawing of letter A (a) Input from Human-Machine Interface (b) Output on the paper



teta1 -17,51735
teta3 32,050273
teta2 19,090211
teta4 -31,42425
1
Send Data to Arduino +18-19

(a)

(b)

Fig. 10. Free hand drawing of a spiral (a) Input from Human-Machine Interface (b) Output on the paper

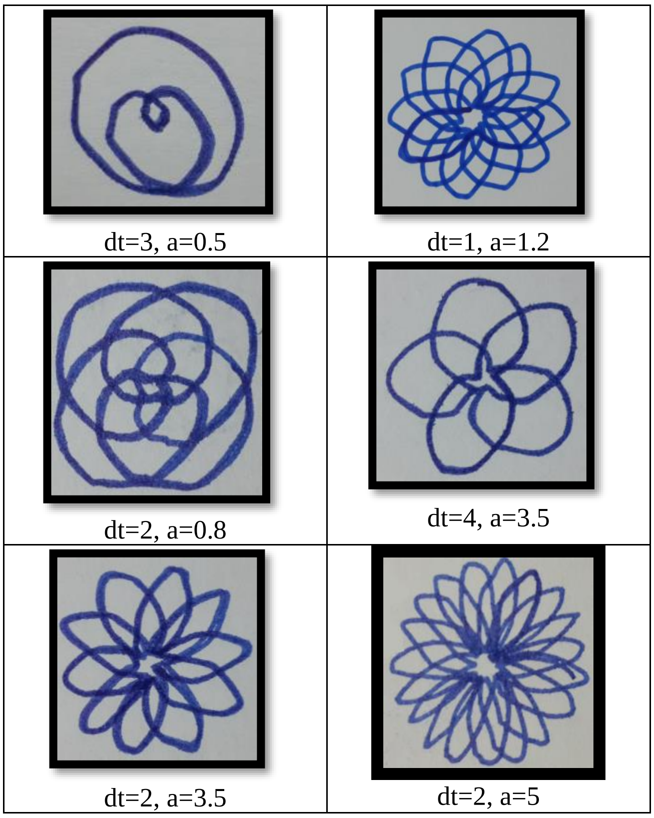
5.3. Drawing a pattern

In this section, a circular pattern drawing will be explained along with examples. We created the pattern by using Equations as follows,

$$P_x = 60 + rcos\left(\beta \frac{\pi}{180}\right) \tag{15}$$

$$P_y = 60 + rsin\left(\beta \frac{\pi}{180}\right) \tag{16}$$

Where Px and Py are coordinates of the end point of the robot, r is radius of circle, β is angle of the rotation of radius. If radius is constant and β angle is changing uniformly, we will get a circle. However, if radius is changed according to a linear function (r=r+a*k, if r>90 then k=-1, if r<1 then k=+1) and also changing rotation of angle (β = β +dt), the patterns can be drawn. According to this pattern configuration and formulation, the end-effector of five-bar robot drew the patterns shown in Fig. 11.



dt=3, a=0.5

dt=1, a=1.2

dt=2, a=0.8

dt=4, a=3.5

dt=2, a=3.5

dt=2, a=5

Fig. 11. Some pattern drawings for different dt and a values

6. Conclusion

The five-bar robot manipulator was constructed and tested in this study. Two servo motors of the robot were controlled using HMI program through Arduino microcontroller. Test results were shown for a line, a rectangle, free hand drawing and patterns. Cost of our robot manipulator is very low (it is nearly 50$) compared to other robot manipulators. Readability and appearance of the test shapes are good. However, accuracy and repeatability are not very well. Therefore, they must be improved. Due to its cost, this robot manipulator will be useful to explain the working principle of the five-bar robot manipulator and application of the robot to engineering students.

References

[1]. Kobayashi, E., Masamune, K., Sakuma, I., Dohi, T., & Hashimoto, D., *A new safe laparoscopic manipulator system with a five-bar linkage mechanism and an optical zoom,* Computer Aided Surgery, 4(4), pages 182-192, 1999.

[2]. Cheng, L., Lin, Y., Hou, Z. G., Tan, M., Huang, J., & Zhang, W. J., *Adaptive tracking control of hybrid machines: a closed-chain five-bar mechanism case,* IEEE/ASME Transactions on Mechatronics, 16(6), pages 1155-1163, 2011.

[3]. Zi, B., J. Cao, and Z. Zhu, *Dynamic simulation of hybrid-driven planar five-bar parallel mechanism based on simmechanics and tracking control,* International Journal of Advanced Robotic Systems, 8(4), pages 28-33, 2011.

[4]. Alici, G., *An inverse position analysis of five-bar planar parallel manipulators,* Robotica, 20(02), pages 195-201, 2002.

[5]. Villarreal-Cervantes, M.G., Cruz-Villar, C. A., Alvarez-Gallegos, J., & Portilla-Flores, E. A., *Differential evolution techniques for the structure-control design of a five-bar parallel robot,* Engineering Optimization, 42(6), pages 535-565, 2010.

[6]. Kim, J., *Task based kinematic design of a two DOF manipulator with a parallelogram five-bar link mechanism,* Mechatronics, 16(6), pages 323-329, 2006.

[7]. Joubair, A., M. Slamani, and I.A. Bonev, *Kinematic calibration of a five-bar planar parallel robot using all working modes,* Robotics and Computer-Integrated Manufacturing, 29(4), pages 15-25, 2013.

[8]. Bourbonnais, F., P. Bigras, and I.A. Bonev, *Minimum-time trajectory planning and control of a pick-and-place five-bar parallel robot,* IEEE/ASME Transactions on Mechatronics, 20(2), pages 740-749, 2015.