Control of robot arm by using reference arm and LabVIEW

Ozcan CETINKAYA [1], Hilmi KUSCU [2]
[1]*Keşan Vocational College, TrakyaUniversity, Edirne,Turkey*
ozcancetinkaya@trakya.edu.tr
[2]*Engineering Faculty, Mechanical Engineering Department, Trakya University, Edirne, Turkey*
hilmi@trakya.edu.tr

Abstract: In this study, position controlling of a robot arm with Four degree of freedom and an ending function element connected to this robot arm by using reference arm and control software written in computer environment were considered. For this purpose, responsive potentiometers placed on the joints of one of the arms to perceive the joint angles and named as Master Arm. To the joints of other arm, RC servo motors were placed which will move according to the joint angles of master Arm as well as the virtual joint angles produced by the control software and this arm named as Slave Arm. The Control program of Master and Slave Arms that is the exchange of information with computer was provided via an ADC Daq card connected to the computer. Additionally, the positional vectors of the ending function element which were kinematic calculations made according to the joint angles taken from the Master Arm or virtual joint angles produced on the control software were shown on the software for information purpose. In addition to that, Inverse kinematic calculations of x, y, z coordinates for the required position of ending function element on the three dimensional plane's have been made to provide positioning of the slave arm, and the calculated joint angles have also been displayed on the monitor for information.

Key Words: manipulator, robot arm, position control, advanced-kinematics, reverse-kinematics

1. INTRODUCTION

Nowadays, computer assisted design and manufacturing systems have showed an improvement in parallel with the improvement of the computer systems. One or more manipulators which are mostly controlled by a computer and apex processors bounded in these manipulators are used in computer assisted manufacturing systems.

Position control is taken on with the assistance of a manipulator which has four degrees of freedom and a bounded apex processor, also again a manipulator which has four degrees of freedom and a control programme which is setup with a computer. Slave Arm; as to choice made from control programme: It follows the transactions of master arm. It follows the angles of virtual potentiometers in control programme. It moves to the expected point on three-dimensional coordinate platform where apex processor is wanted to reach in control programme and it makes fast acts to the stated coordinates with designated periods.

2 USED COMPONENTS

This study composed of, a master Arm which senses the movements, a Slave arm which follows up the movements of the master arm, a Data collection card which provides the controlling of slave arm by transferring the analog signals received from the master arm to digital data into the computer environment, a Servo motor driver card which provides controlling the servo motors by the analog signals received from the data collection card, and a control software that provides performing all these processes in the computer environment.

2.1. The structure of reference (Master) arm

The Master Arm (in Figure 1) which perceives the movements is manhandled and its joints take an angle value. It is necessary to transfer these values to a form that can be received by computer and read by the assistance of software. Therefore, by placing responsive potentiometers on the joints, the joint angles are transferred to analog signals, and hence, it is provided to transfer the analog signals to an accessible format which will be processed at the software via an I/O card.

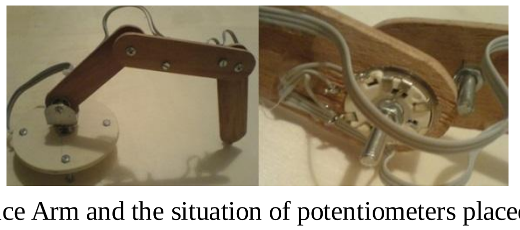

Fig. 1. Reference Arm and the situation of potentiometers placed on the joints

2.2. The structure of the controlled Slave arm

Slave arm (in Figure 2) moves according either to the master arm's joint movements or to the joint angles given by software. To provide this, joint angles processed on the computer are transferred to analog signals from digital, sent to servo motor driver card to control the positions of servo motors. Each joint angle transferred to analog signal, is converted to a signal which controls the axle position (angle) of servo motors by Pulse Width Modulation logic.

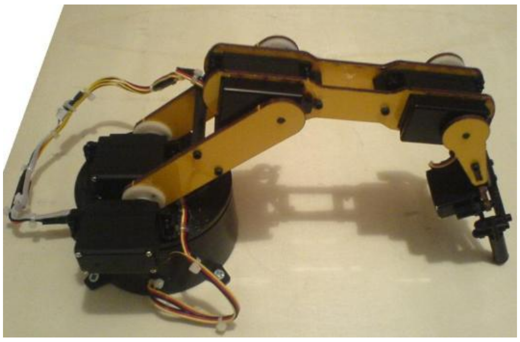

Fig. 2. Controlled Arm

2.3. Mechanical structure of Slave arm

Slave arm has got four rotating joints with four degree of freedom similar to the master arm and an ending function element with a holding part. Its structure is same as the master arm but its design differs because RC Servo motors placed on the joints. Rotation angle of each joint is 180º as of master arm. The main frame made of plastics and the parts of arm made of 4 mm Lexan Polycarbonade material. At the Figure 3 given, the dimensions of the slave arm can be seen.

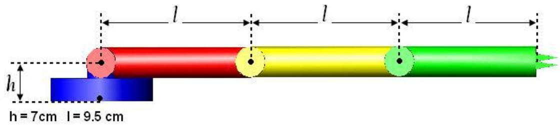

Fig. 3. Dimensions of Slave Arm

2.4. Servo motor driver control card

Servo motor driver card (in Figure 4) converts the analog angle data received from joints of the arm to a signal which its frequency is 50 Hz and modulation range varies in-between 0,5 ms to 2,5 ms to determine the axle positions of servo motors. Servo motor driver card consists of Analog-Digital converter block, Digital output block and PIC feeding block. The joint angle data of data collection card, which is converted to analog data, then, converted to 8 bits digital data by assistance of Microcontroller. A digital output is produced by converting this 8 bits digital data to a signal which its frequency is 50 Hz and modulation range varies in-between 0.5 ms to 2.5 ms by the microcontroller. In this study, PIC16F877/A is used as of microcontroller.

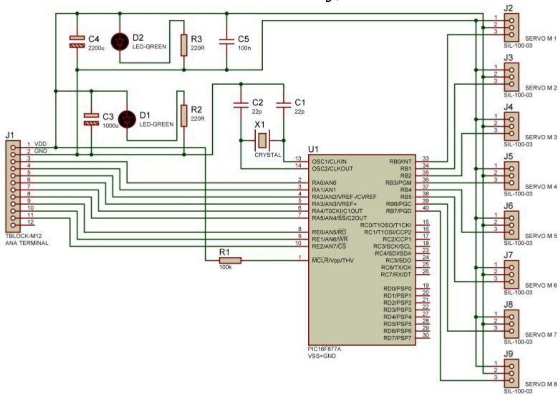

Fig. 4. Principle Schema of Servo Deriver Card

2.5. Microcontroller software

It is necessary to install software to Microcontroller to perform the simulation of and to operate the electronic circuit prepared by ISIS in real life conditions (in Figure 5). The software used in this study was developed with an intermediate level programming language PICBASIC.

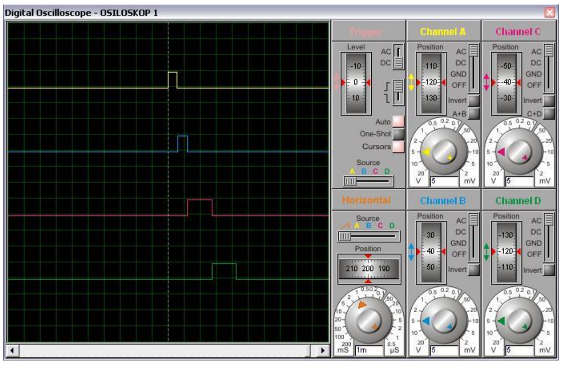

Fig. 5. Ossiloscope view of the control signals of Servo motors at ISIS simulation program

The software to be installed to 16F877A microcontroller was developed with PicBasicPro programming language.

2.6. Analog-Digital converter data input and output card

JAC AIO3320 Analog-Digital Converter (Data Collection) Data Input-Output card (in Figure 6) is used to transfer the analog data received from the potentiometers on the joints of the Master Arm to computer in digital data format, and to send the digital data processed in the computer to the servo motor driver card in analog data format.

Fig. 6.JAC AIO3320 Card and Equipment

2.7. Developed control software

In this study, a user interface software was developed to control the servo motors on the joints of the slave arm with the analog information, which was converted to digital data and processed in several ways in the computer environment, received from the joints of master arm by using AIO3320 data collection card, This software was developed with Graphic Programming Language LabVIEW.

The position vectors of all joints and the position vectors of ending function element, that is Forward kinematic calculation is found by MATLAB script, which is processing under the software. The value of joint angles of the ending function element according to the given coordinates that is Inverse kinematic calculation is also is found by MATLAB script, which is processing under the software.

The software can be used in four different ways by the user. First, the analog data received from the master arm is processed in the computer and the slave arm is controlled. Second, slave arm is controlled based on the joint angles given from the software. Third, the position coordinates of ending function element is given as x, y, z and the slave arm is controlled in this way. The last one, slave arm is controlled by choosing the position coordinates of ending function element from a list of coordinates.

The graphic of the slave arm according to the joint angles are displayed as immediate data. The user interface of the control software is given in Figure 7. (Çetinkaya, 2009)

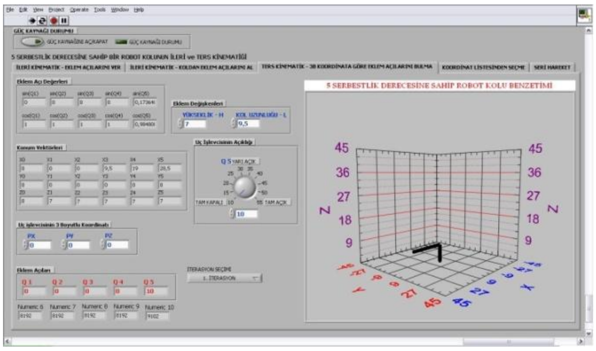

Fig. 7. The user interface of the control software

2.7.1. Forward and Inverse kinematic calculations with MATLAB script

It's necessary to calculate the positioning vectors according to the joint angles received from master arm or joint angles of the slave arm which is provided by the software. For these, it is needed Forward and Inverse kinematic calculations in the software. Since the Forward and Inverse kinematic calculations take long time and complicated with LabVIEW 's own functions, they are calculated by MATLAB script running under LabVIEW. MATLAB script is added on from the function palette to the block diagram of the program.

To perform the Forward and Inverse kinematic calculations, MATLAB script which is running under LabVIEW runs MATLAB at the background and operates the command sequence given to the script, and therefore produces Forward kinematic results based on the joint data and Inverse kinematic results according to the positioning vector data of ending function element.

To perform the Forward and Inverse kinematic calculations, first of all, it is necessary to form the mathematic model of the slave arm and set the Forward and Inverse kinematic equations. Therefore, before all else, Forward and Inverse kinematic models and positioning vectors are determined.

2.7.2. Forward kinematic models and positioning vectors of Master and Slave arms

To form the kinematic model (in Figure 8), first, the joint variables and constants are determined and the coordinate systems placed on the joints.

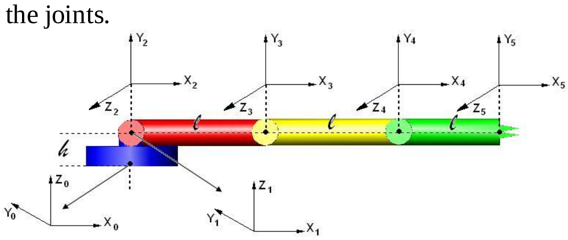

Fig. 8. Mathematic Model of Four Freedom Degree Robot Arm

After the placement of coordinate systems on the joints, D-H variables (Denavit&Hartenberg, 1955) are calculated and written to the Table-1DenavitHartenberg Variables. Invariable parameters which do not vary with the movement of the robot are a_{i-1} longitudes of parts and α_{i-1} axle angles. The varying parameters are, on the other hand, if the joint is swivel, θ_{i-1} joint angle and if the joint is prismatic, d_{i-1} joint crookedness.

Table 1. DenavitHartenberg Variables

Axis No	D-H Variables				Axis Variables
i	αi-1	a_{i-1}	d_i	Q_i	d_i or Q_i
1	$\alpha_0=0$	$a_0=0$	$d_1=h$	Q_1	Q_1
2	$\alpha_1=90$	$a_1=0$	$d_2=h$	Q_2	Q_2
3	$\alpha_2=0$	$a_2=l$	$d_3=h$	Q_3	Q_3
4	$\alpha_3=0$	$a_3=l$	$d_4=h$	Q_4	Q_4
5	$\alpha_4=0$	$a_4=l$	$d_5=h$	Q_5	Q_5

Homogeneous Transformation Matrix of Five Freedom Degree Slave Arm. (Bingül&Küçük, 2005)

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}d_i \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

At the first step, Homogeneous Transformation Matrixes of each joint were calculated from the Homogeneous Transformation Matrix. Then, Homogeneous Transformation Matrixes of each joints were multiplied with each other Homogeneous Transformation Matrix was found. The positioning vector of Homogeneous Transformation Matrix gives the Px, Py and Px position of the ending functioning element on three dimensional coordinates plane. (Çetinkaya, 2009)

$$ {}^{0}_{5}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2) $$

The calculated Transformation Matrixes of five joints multiplied with each other, Transformation Matrix of Slave Arm was obtained and each 5 joint's positioning vectors (,and) were found. (Çetinkaya, 2009)

$$ \begin{aligned} p_{X5} = {} & l.(\cos\theta_4(\cos\theta_1\cos\theta_2\cos\theta_3 - \cos\theta_1\sin\theta_2\sin\theta_3) - \\ & \sin\theta_4(\cos\theta_1\cos\theta_2\sin\theta_3 + \cos\theta_1\sin\theta_2\cos\theta_3)) \\ & + l.(\cos\theta_1\cos\theta_2\cos\theta_3 - \cos\theta_1\sin\theta_2\sin\theta_3) + l.\cos\theta_1\cos\theta_2 \\[1ex] p_{Y5} = {} & l.(\cos\theta_4(\sin\theta_1\cos\theta_2\cos\theta_3 - \sin\theta_1\sin\theta_2\sin\theta_3) - \\ & \sin\theta_4(\sin\theta_1\cos\theta_2\sin\theta_3 + \sin\theta_1\sin\theta_2\cos\theta_3) \\ & + l.(\sin\theta_1\cos\theta_2\cos\theta_3 - \sin\theta_1\sin\theta_2\sin\theta_3) + l.\sin\theta_1\cos\theta_2 \\[1ex] p_{Z5} = {} & l.(\cos\theta_4(\sin\theta_2\cos\theta_3 + \cos\theta_2\sin\theta_3) + \\ & \sin\theta_4(-\sin\theta_2\sin\theta_3 + \cos\theta_2\cos\theta_3)) \\ & + l(\sin\theta_2\cos\theta_3 + \cos\theta_2\sin\theta_3) + l.\sin\theta_2 + h \end{aligned} \quad (3) $$

2.7.3. Inverse kinematic model of Master and Slave arms and joint angles

${}^{0}_{5}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T$ Let's multiply the both sides of matrix of forward kinematics with $\left[{}^{0}_{1}T\right]^{-1}$,

$$ \left[{}^{0}_{1}T\right]^{-1}\,{}^{0}_{5}T = \left[{}^{0}_{1}T\right]^{-1}\,{}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T \quad (4) $$

As it's known, since $\left[{}^{0}_{1}T\right]^{-1}\,{}^{0}_{1}T = I$, the equation will be as given down below.

$$ {}^{1}_{5}T = {}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T \quad (5) $$

At first, inverse of the each transformation matrix are calculated.

${}^{0}_{1}T$ The inverse of the transformation matrix is equal to the matrix of ${}^{0}_{1}T^{-1}$

$$ {}^{0}_{1}T^{-1} = \begin{bmatrix} {}^{0}_{1}R^{T} & -{}^{0}_{1}R^{T} * {}^{0}P_1 \\ 0\;\;0\;\;0 & 1 \end{bmatrix} \quad (6) $$

$$ {}^{0}_{1}R = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \; {}^{0}_{1}R^{T} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7) $$

$$ -{}^{0}_{1}R^{T}\,{}^{0}P_1 = \begin{bmatrix} -\cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & -\cos\theta_1 & 0 \\ 0 & 0 & -1 \end{bmatrix} . \begin{bmatrix} 0 \\ 0 \\ h \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -h \end{bmatrix} \quad (8) $$

$$ {}^{0}_{1}T^{-1} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & -h \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9) $$

Performing the similar operations for the other matrixes, the following inverse kinematic equations were obtained. (Bingül & Küçük, 2005)

1. $p_x \cos\theta_1 + p_y \sin\theta_1 = l.(\cos\theta_4(\cos\theta_2\cos\theta_3 - \sin\theta_2\sin\theta_3) - \sin\theta_4(\cos\theta_2\sin\theta_3 + \sin\theta_2\cos\theta_3)) + l.(\cos\theta_2\cos\theta_3 - \sin\theta_2\sin\theta_3) + l.\cos\theta_2$

2. $-p_x \sin\theta_1 + p_y \cos\theta_1 = 0$

3. $p_z - h = l.(\cos\theta_4(\sin\theta_2\cos\theta_3 + \cos\theta_2\sin\theta_3) + \sin\theta_4(-\sin\theta_2\sin\theta_3 + \cos\theta_2\cos\theta_3)) + l(\sin\theta_2\cos\theta_3 + \cos\theta_2\sin\theta_3) + l.\sin\theta_2 + h$ (10)

4. $p_x(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2) + p_z(\cos\theta_1\sin\theta_2 + \sin\theta_1\cos\theta_2) = l.(\cos\theta_4\cos\theta_3 - \sin\theta_4\sin\theta_3) + l.\cos\theta_3 + l$

5. $p_x(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2) + p_z(-\sin\theta_1\sin\theta_2 + \cos\theta_1\cos\theta_2) = 0$

6. $-p_y - h = l.(\cos\theta_4\sin\theta_3 + \sin\theta_4\cos\theta_3) + l\sin\theta_3 + h$ (11)

7. $p_x(\cos\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2)) + p_y(\sin\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2)) + p_z(\cos\theta_1\sin\theta_2 + \sin\theta_1\cos\theta_2) - l.\cos\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2) = l.\cos\theta_4 + 2l$

8. $p_x(c\cos\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2)) + p_y(\sin\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2))) + p_z(-\sin\theta_1\sin\theta_2 + \cos\theta_1\cos\theta_2) - l.\cos\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2) = 0$

9. $p_x(\sin\theta_3) + p_y(-\cos\theta_3) + 0 - l.\sin\theta_3 - h = l.\sin\theta_4 + h$ (12)

(Çetinkaya, 2009)

10. $p_x(\cos\theta_4\cos\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2) - \sin\theta_4(\sin\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2))) + p_y(\sin\theta_4(\cos\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2)) + \cos\theta_4(\sin\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2))) + p_z(\cos\theta_1\sin\theta_2 + \sin\theta_1\cos\theta_2) - l.\cos\theta_4(\cos\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2)) + l.\sin\theta_4(\sin\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2)) - l.\cos\theta_3(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2) = 3l$

11. $p_x(\cos\theta_4(\cos\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2)) - \sin\theta_4(\sin\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2))) + p_y(\sin\theta_4(\cos\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2)) + \cos\theta_4(\sin\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2))) + p_z(-\sin\theta_1\sin\theta_2 + \cos\theta_1\cos\theta_2) - l.\cos\theta_4(\cos\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2)) + l.\sin\theta_4(\sin\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2)) - l.\cos\theta_3(-\sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2) = 0$

12. $p_x(\sin\theta_3\cos\theta_4 + \cos\theta_3\sin\theta_4) + p_y(\sin\theta_3\sin\theta_4 - \cos\theta_3\cos\theta_4) - l.\cos\theta_4\sin\theta_3 - l.\sin\theta_4\cos\theta_3 - l.\sin\theta_3 - h = h$ (13)

2.7.4. Control program and tabs

Control software was developed with LabVIEW Graphic programming language which runs under Windows XP operating system. Slave arm in our system gains ability of movement according to the joint angles produced in computer environment by this control software, joint angles received from master arm and ending function element coordinates given via the computer environment. Though the Slave arm has got three functions with control software, there are five tabs that derivatives of these functions.

These are;

- Moving Slave arm according to the joint angles produced in computer environment,
- Moving Slave arm according to the joint angles received from Master arm,
- Moving Slave arm according to the three dimensional ending function element coordinates which were produced in computer environment,
- Moving Slave arm according to the three dimensional ending function element coordinate list which were produced in computer environment,
- Periodical moving the Slave arm according to a pre-programmed serial movement.

In Figure 9, general view of the user interface of the control program is given.

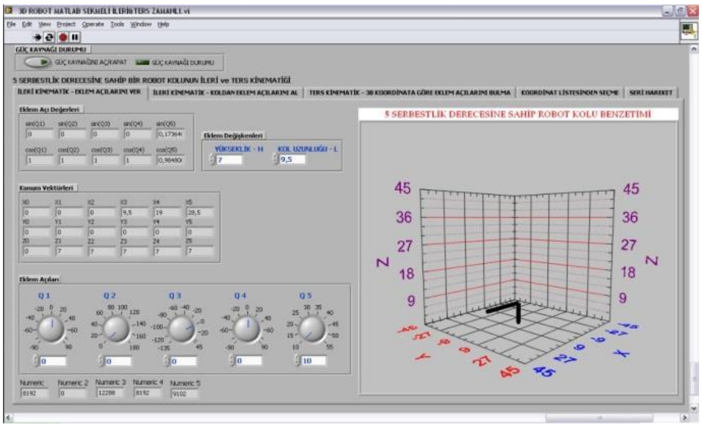

Fig. 9. User Interface of Control Program

2.7.4.1. Forward kinematic – Give joint angles tab

In this tab, the required signal for driving the servo motors according to the joint angles and joint variables produced in the computer environment with the Q1, Q2, Q3, Q4 and Q5 virtual potentiometers given at Figure 10, produced with JAC3320 ADC card.

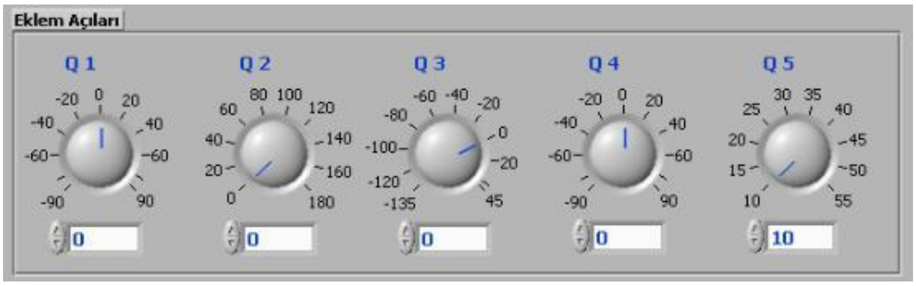

Fig. 10. Virtual Potentiometers

The analog signal for driving the servo motors produced with JAC3320 ADC, transformed to PWM signal format via the servo motor driver card and therefore the servo motors are controlled. In addition to that, forward kinematic and positioning vectors of joints are calculated according to the joint angles produced in computer environment with Q1, Q2, Q3, Q4 and Q5 virtual potentiometers, and the tab is shown at Figure 11.

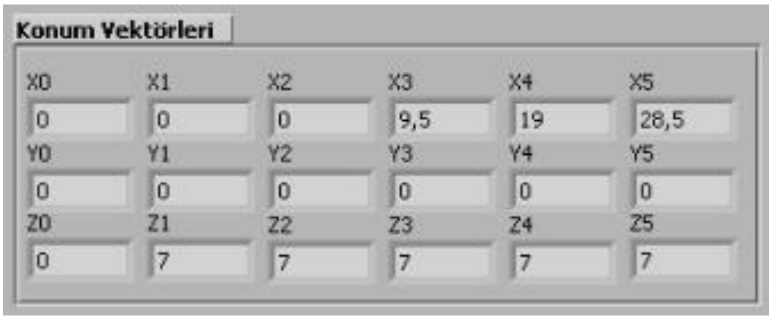

Fig. 11. Positioning Vectors of Joints

The real time graphic with 3D Curve object of the position of Slave Arm on the three dimensional coordinated plane according to this positioning vectors at this tab is shown at Figure 12.

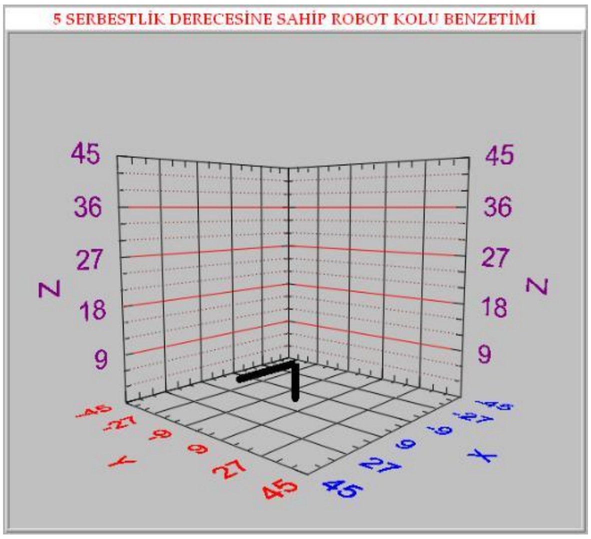

Fig. 12. Three Dimensional Momentary Graphic of Slave Arm

Joint Variables of Slave and Master Arm; The height to main frame H and the length of arm are fixed. Therefore, are displayed as joint variables in the program as shown at Figure 13.

Fig. 13.Joint Variables

Moreover, in this tab, sinus and cosines equivalents of joint angles produced in computer environment with Q1, Q2, Q3, Q4 and Q5 virtual potentiometers are displayed in the software for information purposes as shown at Figure 14.

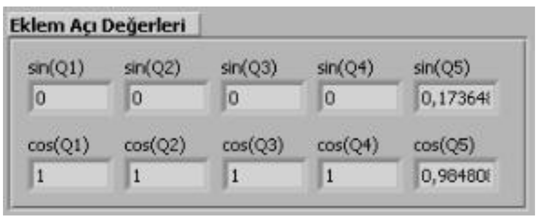

Fig. 14. Sin and Cos equivalents of Joint Angles

In this tab, to be able to perform the forward kinematic calculations, MATLAB script (Figure 15) that is added on to LabVIEW block diagram is used.

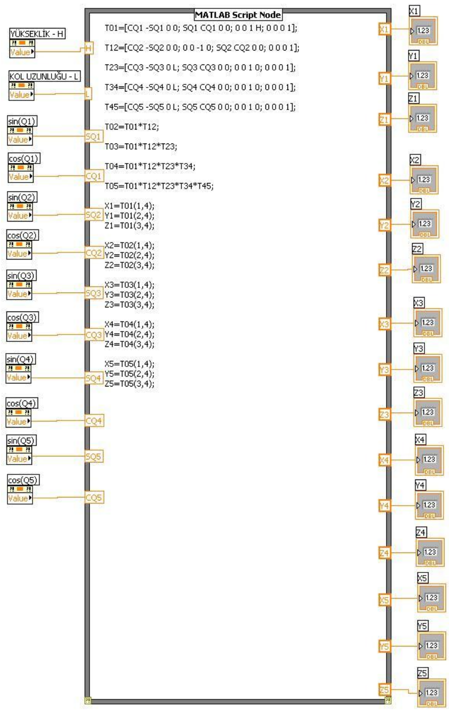

Fig. 15. ForwardKinematic MATLAB script

The below listed MATLAB commands entered to MATLAB script, the forward kinematic calculations of slave arm are found at the background with MATLAB software and used in the program.

2.7.4.2. Forward kinematic – Get the joint angles from the arm tab

The only difference between the "Forward Kinematic – Get the joint angles" from the arm tab which is the second tab of the program and the previous tab "Forward Kinematic – Give Joint Angles Tab" is, moving the slave arm according to the joint angles received data from the real potentiometers placed on the master arms joints instead of producing with virtual potentiometers in the control program. For this purpose, a driver which is given down below at the smart_AtoD.vi (http://ftp.automation.com.tw/labview_driver/labview_driver.exe, 08.05.2008) block diagram (Figure 16) for the potentiometers placed on master arm's joints is used.

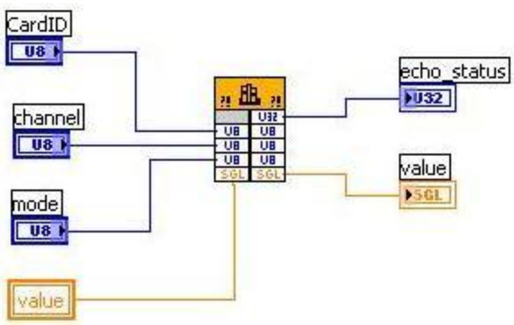

Fig. 16. Smart_AtoD.vi Block diagram

The ordered situation of the driver at the smart_AtoD.vi block diagram to read the no 1 joint angle of the master arm as is given down below at Figure 17.

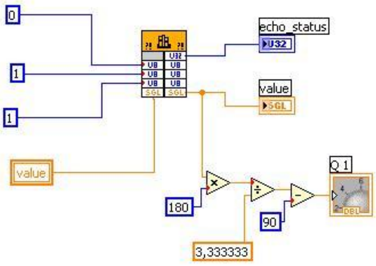

Fig. 17. Ordered situation of the smart_AtoD.vi block diagram to read the no 1 joint angle of the master arm

Here, analog data read from the 1, 2, 3, 4 and 5 numbered channels of JAC3320 ADC card is assigned to Q1, Q2, Q3, Q4 and Q5 display objects. Though Q1, Q2, Q3, Q4 and Q5 objects were control objects in previous tab, in this tab, they are display objects. Since the Q1, Q2, Q3, Q4 and Q5 objects will show the Master arm's joint angles, the received voltage data should be transformed to an angle data in between -90 to +90. For this purpose, first, the analog data which the card read from the channel is assigned to Value variable, then, this assigned voltage data which is varying between 0 to 3,3333 Volt, is transformed to a linear angle data varying between -90 to +90 with the block diagram which is rectangle framed at Figure 18 given down below.

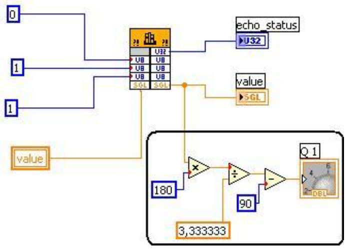

Fig. 18. Transforming the resistance to angle data

There is a 180 degree limitation on the joints of master arm and the potentiometers on the joints. But, the brush (rotating) angle of potentiometers is 270 degrees. Thus, when 5 V applied to potentiometers, because of 180 degrees angle limitation, the received voltage range will be in between 0 to 3.3333 V. Consequently, Q1, Q2, Q3 and Q4 display objects' scales were calibrated 3.3333V for 180 degrees. On the other hand, due to the potentiometer which is setting the opening of the ending functioning element has not got any angle limitation, Q5 object was calibrated with 5V for 45 degrees (in Figure 19) .

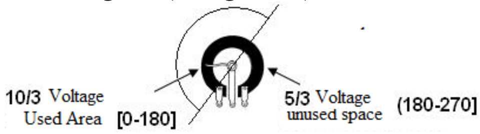

Fig. 19. Angle Limitation on the Potentiometer

2.7.4.3. Inverse kinematic– Finding joint angles according to 3D coordinates

In this tab of the control program, the possible joint angles of the slave arm is found by providing the coordinates according to where the slave arm's ending functioning element should stay on the three dimensional coordinates plane. With assistance of the equations calculated from the inverse kinematic model of slave arm, the joint angles are found according to where the ending functioning element should be.

For calculating the joint angles with assistance of the equations of inverse kinematics, as of in calculations of forward kinematics, the MATLAB script is prepared and used as shown at Figure 20.

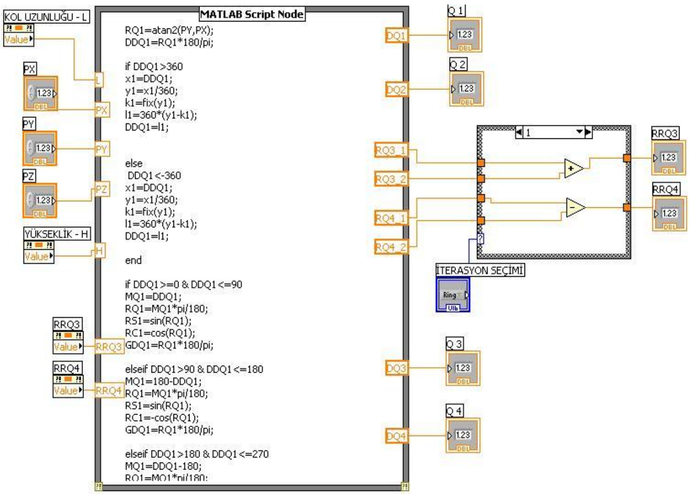

Fig. 20. Inverse kinematic MATLAB script and Block diagram

DQ1, DQ2, DQ3 and DQ4 angles produced with MATLAB script are the angle values of slave arm's joints. The positioning vectors according to these joint angles, momentarily graphic of the slave arm and the analog signals indicating the positions of servo motors on these joints are calculated as in the first tab of the program. Only, to open and close the Slave Arm ending functioning element, Q5 virtual potentiometer is added to the tab. The opening of the ending functioning element is controlled with virtual potentiometer which is shown Figure 21.

Fig. 21. Virtual potentiometer controlling the opening of ending functioning element

The data entry section for the coordinates of ending functioning element on three dimensional planes on the program is shown at Figure 22.

Fig. 22.X, Y, Z coordinate entry of ending functioning element

Section of informative displaying the joint angles calculated from inverse kinematic equations on the program is shown at Figure 23. The Q5 angle value shows the opening of ending functioning element.

Fig. 23. Calculated Joint Angles

Four iterations are formed based on the minus / plusses found at the 3rd and 4th of the inverse kinematic equations. As it is seen on the Block Diagram at Figure 23, in the first iteration, by adding RQ3_1 and RQ3_2, RQ3, then, by adding RQ4_1 and RQ4_2, RQ4 are obtained. In the second iteration, by adding RQ3_1 and RQ3_2, RQ3 and then, by subtracting RQ4_2 from RQ4_1, RQ4 are obtained. In the third iteration, by subtracting RQ3_2 from RQ3_1, RQ3, and then, by adding RQ4_1 and RQ4_2, RQ4 are obtained. In the fourth iteration, by subtracting RQ3_2 from RQ3_1, RQ3, and then by subtracting RQ4_2 from RQ4_1, RQ4 are obtained. Iterations are effective for calculating the joint angles (in Figure 24).

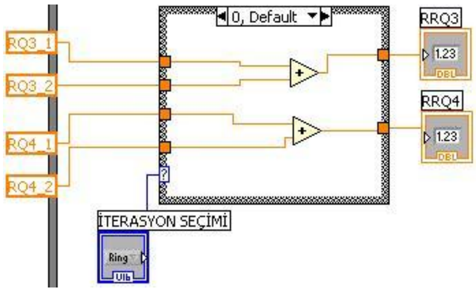

Fig. 24. Iteration Block diagram

These iterations can be selected at the program by the selection list at Figure 25.

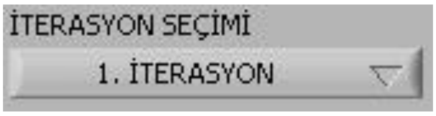

Fig. 25. Iteration Selection List

At the Figure 26 given, the user interface of the tab of “Inverse Kinematic – Calculating the joint Angles according to 3D Coordinates” can be seen.

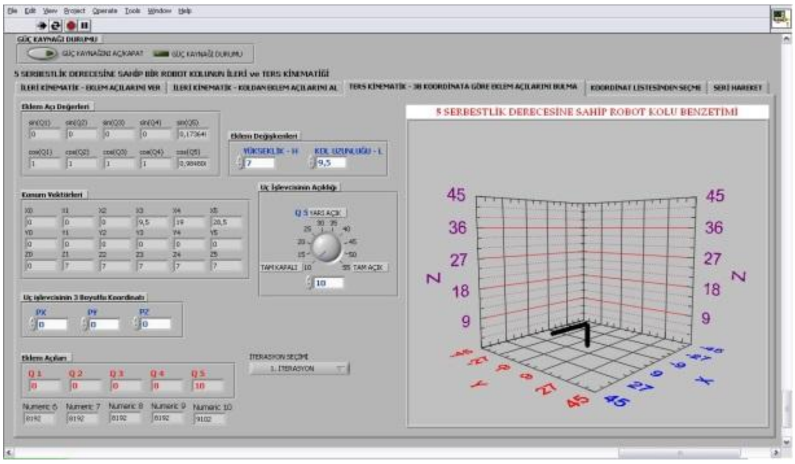

Fig. 26. User interface of the tab of calculating the joint Angles according to 3B coordinates.

3 RESULTS AND SUGGESTIONS

At this study, controlling the slave arm through master arm or control software was projected and this result was obtained.

I. If the positioning vectors of each joint and the ending functioning element, according to the joint angles produced by Q1, Q2, Q3 and Q4 virtual potentiometers placed in the control program, had been found momentarily and errorless controlled via the positioning vectors matrix in the program. In addition to that, it was also verified by the graphic which provides required positions of the ending functioning element and other joints of slave arm three dimensionally as well as real movements. It is seen that, with the Q5 virtual potentiometer placed in the program, the opening of the ending functioning element can be controlled problem-free. (in Figure 27)

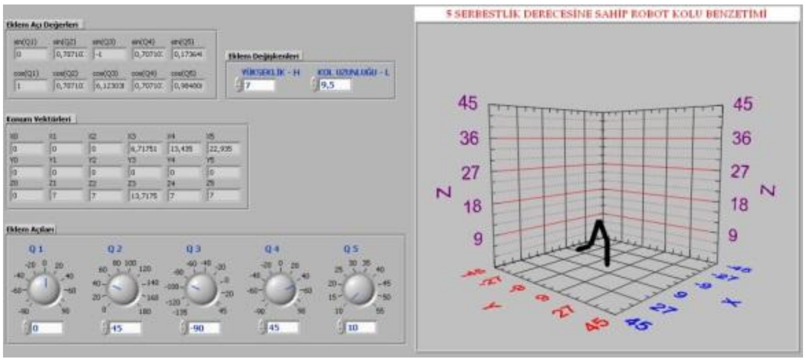

Fig. 27. Positioning Vectors according to the Joint Angles and their Three Dimensional Graphic

II. It was seen and controlled via the positioning vectors matrix placed in the program and via the three dimensional graphic that, when the required x, y, z three dimensional coordinates of the ending functioning element is given at the control program, the joint angles take equal or very close values to required angle values. (in Figure 28)

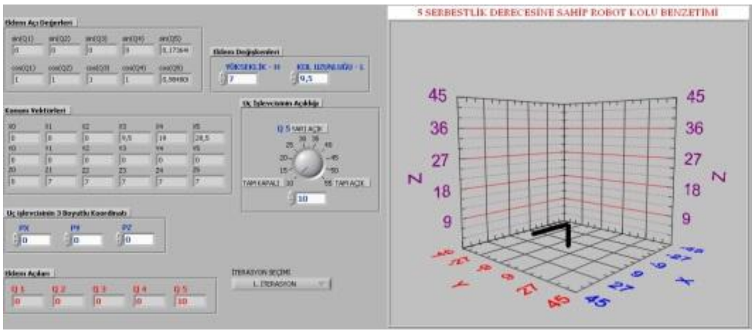

Fig. 28. Joint Angles according to Three Dimensional Coordinates and their Three Dimensional Graphic

III. The joint angles received from Master arm and transformed to analog form were displayed error-free in digital form on the control program, and each joint positioning vector were calculated and displayed at the positioning vectors matrix. By transforming the joint angles of Master Arm into analog form again to control the Slave Arm and the Servo motor card as well as the Slave Arm controlled error-free.

ACKNOWLEDGEMENT
Research described in the paper was supported by the University of Trakya under scientific Foundation projects under No 891. Research described in the paper is a part of the MSc thesis that was written by Özcan ÇETİNKAYA

References

Bingül Z., S. Küçük "Robot Tekniği 1", Birsen Press, 2005. pp. 104-110, 169-176

Cetinkaya O. "Design and Experimental Research of Four Rotating Joints Robot Arm Which Follows the Movements of an Arm." MSc, Trakya University, 2009, pp. 132, 134, 134-139,140-149

Denavit J., R.S.Hartenberg "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices." Journal of Applied Mechanics, June 1955, 215-221.