Development of Parallel Delta Robot System Controller Based on Raspberry Pi and FPGA

Article *in* Applied Mechanics and Materials · May 2016

DOI: 10.4028/www.scientific.net/AMM.835.698

Development of Parallel Delta Robot System Controller based on Raspberry Pi and FPGA

Wanayuth Sanngoen[1,a], Watcharin Po-ngaen[2b],
Chirot Charitkhuan[3,b], Kitsada Doungjitjaren[4,*]

[2,4]Faculty of Technical Education, King Mongkut's University of Technology North Bangkok

[1,3]Department of Computer Engineering, Department of Mechatronics, Sripatum University

[a]wanayuth.sa@spu.ac.th, [b]vpn@kmutnb.ac.th , [c]chirot@spu.ac.th , [*]kitsada.dou@gmail.com

Keywords: Raspberry Pi, FPGA, Look Up Table

Abstract. This paper presents a system control design on 2 axes "Delta Industrial Robot". The control system utilizes both the embedded technology and the "Raspberry Pi" which is a low power consumption PC with open source operating system. The user can communicate with the PC and FPGA (Field Programmable Gate Array) via the system GUI interface. The FPGA acts as dedicated hardware to provide both the efficiency and flexibility for the closed-loop servo control system.

To calculate all related kinematics equations for the robot, look up table approach has been implemented to reduce the calculation complexity. Python programs are also implemented to create and search the look up tables. The inverse kinematics is the key to find the position of the robot. A test load of 0.2 kg at the velocity of 2.618 m/s, the S.D. of the end effector's position of the test result is within 0.06 mm.

Introduction

There are many introductions of utilizing various embedded systems to control various robots. Many of such embedded systems are combination of ARM microcontrollers and FPGA. The main reasons behind such approaches are: the decimal calculations can easily be done on the microcontrollers and the FPGA has unique capability of parallel processing. The combination of the above features results in an efficient control system, this reduces complexities and allows reconfigurable control architecture which is very important in controlling each individual industrial robots [1-3]. For the "Biped Robot" which requires simple and very small size controller, DSP (Digital Signal Processor) is being used as main controller and FPGA is being used as motor controllers. Such approach provides an efficient way of controlling the "Biped Robot" [7]. An alternative new approach is to use Raspberry Pi with the open source Linux operating system to control the above robots. The smaller size of Raspberry Pi should reduce the development cost and provide reliable control system similar to those of dedicated controllers [6].

In order to fully control all the positional motions of the industrial robot, related dynamic equations of all moving mechanical parts have to be calculated to find the required positional motion. The result(s) of the calculation is then compared to the actual positional movement(s) of the industrial robot. This approach will in-turn enhanced the modification of the dynamic equations to increase the accuracies and faster calculation.

There are proposals which introduce forward kinematics calculation by way of comparing the calculated data with the actual positional movement of the Delta Industrial Robot [4]. To speed up the kinematics calculation for the robot manipulators, look up tables and relative angle of the sine and cosine in trigonometric functions are utilized. The fourth order series with its coefficients are stored in the look up tables, these data are calculated in the inverse trigonometric and square root functions. The result of the calculation is then fed into the hardware within the FPGA platform for its extremely fast computation [5].

This goal of this research is to develop a control system for typical Delta Industrial Robot utilizing both the embedded technology and FPGA technology to obtain flexibility, increase the efficiency for faster response and better control resolution for the Delta Industrial Robot. The FPGA will be used as dedicated hardware, the Raspberry Pi will act as the front-end processing unit for the FPGA, this approach will increase the efficiency, reduce the current complexity and reduce the amount of time-consuming in controlling the Delta Industrial Robot.

Design of Control System

From the "Conceptual Design and Dimensional Synthesis of Novel 2-DOF Translational Parallel Robot for Pick and Place Operations" in the Journal of Mechanic, the following are related dynamic equations for Figure 1.

$$H = l_2 + \frac{h}{2} + l_1 \sin\theta_u \tag{1}$$

$$A\sin(\theta - \theta_u) + B\sin(\theta - \theta_u) + C = 0 \tag{2}$$

$$A = -2\left(H + \frac{h}{2}\right)l_1 \tag{3}$$

$$B = 2\left(\frac{b}{2} + e\right)l_1 \tag{4}$$

$$C = \left(\frac{b}{2} + e\right)^2 + \left(H + \frac{h}{2}\right)^2 + l_1^2 - l_2^2 \tag{5}$$

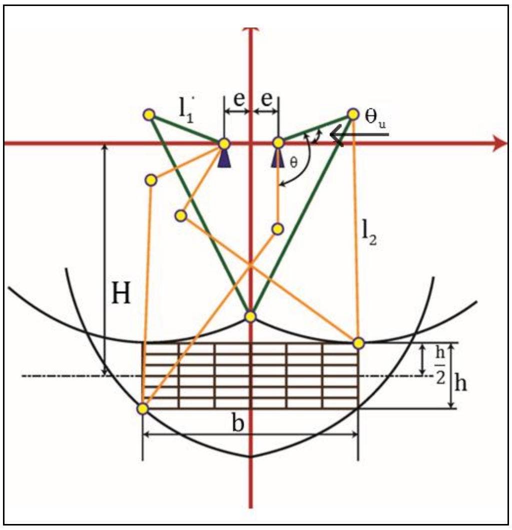

Figure 1. Variable assignments on the 2-axes Delta robot.

where:

H : the vertical distance from the rotational point of the upper arm of the delta robot to the center point of the working envelop of the robot.

e : the distance from center point (0,0) between both the rotational points of the upper arms of the robot.

l_1 : the length of the upper arm portion of the robot.

l_2 : the length of the lower arm portion of the robot.

θ_u : the angle of rotation of the upper arm in reference to the line drawn between 2 starting rotational points of the upper arms.

θ : the maximum angle of rotation of the upper arm of the robot.

h : the width of the working envelop of the Delta robot.

b : the length of the working envelop of the Delta robot.

To control typical 2 axes-Delta Robot, there are 2 basic principles of controlling them. With forward kinematics, knowing the angle of each axis of the Delta robot, the final position of the end effector can be obtained. With inverse kinematics, knowing the required final position of the end effector, the angle of each axis of the Delta robot can be obtained as well.

Forward kinematics

First step, draw a typical structural connection of the Delta robot as those of Figure 2 to simplify the motion analysis of various parts of the robot. Second step, assigned all related mathematical variables on the structural connection of Figure 2, details of each individual variables are as follow:

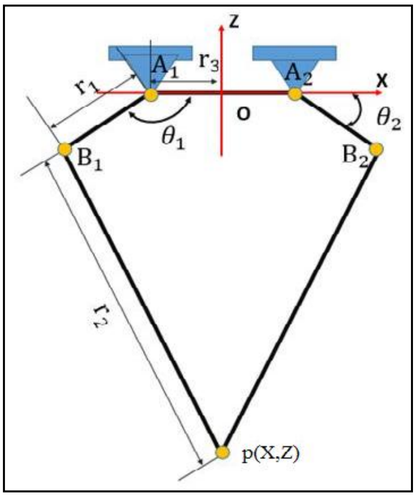

Figure 2. A typical structural connection of the Delta robot.

A_1 & A_2 : the rotational point of each upper arm respectively.

B_1 & B_2 : the rotational point at the end of each upper arm of the Delta robot.

θ_1 & θ_2 : the respective rotational angle of each upper arm on the XZ plane.

O : the origin (0,0) of the XZ plane.

r_1 : the length of the upper arm portion of the robot. (150) mm

r_2 : the length of the lower arm portion of the robot. (403.11) mm

r_3 : the distance between point A_1 and point O. (50) mm

p : the end effector position of the robot.

Third step: develop forward movement equations starting from end effector position of the robot with the position vector of p in (6)

$$p = (X , Z) \tag{6}$$

and position vectors at points B_1 and B_2 are b_1 and b_2 respectively as follow:

$$b_1 = \left(r_1\cos\theta_1 - r_3 r_1 \sin\theta_1\right) \ , \ b_2 = \left(r_1\cos\theta_1 + r_3 r_1 \sin\theta_1\right) \tag{7}$$

With values of p, b_1, b_2 the magnitude of $|pb|$ can be obtained from where

$$|pb_i| = r_2 \qquad\qquad i = 1, 2 \tag{8}$$

By substituting the result of (1) and (2) into (3), we can obtain the following:

$$\left(x - r_1\cos\theta_1 + r_3\right)^2 + \left(z - r_1\sin\theta_1\right)^2 = r_2^{\ 2} \ , \ \left(x - r_1\cos\theta_1 - r_3\right)^2 + \left(z - r_1\sin\theta_1\right)^2 = r_2^{\ 2} \tag{9}$$

From equation (9), obtain the value of x as follow:

$$x = \frac{r_1\left(\sin\theta_1 - \sin\theta_2\right)z}{(2r_3 - r_1\cos\theta_1 + r_1\cos\theta_2)} + \frac{r_1 r_3\left(\cos\theta_1 + \cos\theta_2\right)}{(2r_3 - r_1\cos\theta_1 + r_1\cos\theta_2)} \tag{10}$$

Assign the variable e and f as follow:

$$e = \frac{r_1\left(\sin\theta_1 - \sin\theta_2\right)}{(2r_3 - r_1\cos\theta_1 + r_1\cos\theta_2)} \ , \ f = \frac{r_1 r_3\left(\cos\theta_1 + \cos\theta_2\right)}{(2r_3 - r_1\cos\theta_1 + r_1\cos\theta_2)} \tag{11}$$

Therefore:

$$x = ez + f \tag{12}$$

Substitute the equation (9) into equation (7) to obtain (13) as follow:

$$x = \left(1-e^2\right)z^2 + 2\left(ef - er_1\cos\theta_1 + er_3 - er_1\sin\theta_1\right)z + f^2 - 2\left(r_1\cos\theta_1 - r_3\right) - 2r_1r_3\cos\theta_1 + r_3^2 + r_1^2 - r_2^2 \quad (13)$$

Assign the following variables d, g, h as follow:

$$d = 1-e^2,\ g = 2\left(ef - er_1\cos\theta_1 + er_3 - er_1\sin\theta_1\right),\ h = f^2 - 2\left(r_1\cos\theta_1 - r_3\right) - 2r_1r_3\cos\theta_1 + r_3^2 + r_1^2 - r_2^2 \quad (14)$$

Therefore:

$$dz^2 + gz + h = 0 \quad (15)$$

From equations (10), (11) and (12) with input values of θ_1 and θ_2, the calculation results in 2 values of z, one for positive value, the other one is negative value. Since the end point p (in Figure 2) resides on the negative side of z, choose the negative value of z. This implies that the values of both the angle and z must be negative angles as well.

Inverse Kinematics

Rearrange both the equations (7) and (8), to obtain equations (14) and (15) respectively as follow:

$$\theta_i = 2\tan^{(-1)}(Z_i) \qquad \text{where } i = 1,2 \quad (16)$$

and

$$Z_i = \frac{-b_i \pm \sqrt{b_i^2 - 4a_ic_i}}{2a_i} \qquad \text{where } i = 1,2 \quad (17)$$

where

$$a_1 = r_1^2 + y^2 + \left(x + r_3\right)^2 - r_2^2 + 2\left(x + r_3\right)r_1,\ b_1 = -4yr_1, c_1 = r_1^2 + y^2 + (x + r_3)^2 - r_2^2 - 2(x + r_3)r_1 \quad (18)$$

$$a_2 = r_1^2 + y^2 + \left(x - r_3\right)^2 - r_2^2 + 2\left(x + r_3\right)r_1, b_2 = -4yr_1, c_2 = r_1^2 + y^2 + \left(x - r_3\right)^2 - r_2^2 - 2\left(x - r_3\right)r_1 \quad (19)$$

The implementation of look up table in FPGA

To obtain all of the data set in the respective look up table, the respective values of end point p(X,Z) (within the working envelop of the robot) must be calculated using the Inverse Kinematics approach in the Python program. The lengths of the upper arm and lower arm of the Delta robot are: 150 mm. and 403 mm. respectively. The maximum of traveling length on z axis is 200.03 mm. The maximum length of the traveling length on x axis is 536.65 mm. Table 1 is a typical creation of look up table from related Inverse Kinematics equations.

Table 1 Typical look up table from related Inverse Kinematics equations.

Position X,Z [mm]	Look up table angles of arm movement θ_1 , θ_2 [degree]
0.00,-350	0.00,0.00
………..	………….
-140.00,-350	340.61,37.45
………..	………….
-150.00,-350	340.12,40.47

Control System Design

The difference between using the Raspberry Pi and standard off the shelf ARM microcontroller is the Raspberry Pi is a low power commercialized board with an option to use open source Linux operating system. It is easier to develop the GUI interface to control the FPGA as the proposed system in Figure 3. To control the robot, series of commands are sent to the FPGA via the Address/Data lines. The generated pulse trains which are specified by the above Data portion are sent to the Driver AC Servo to move the robot arm to the required position.

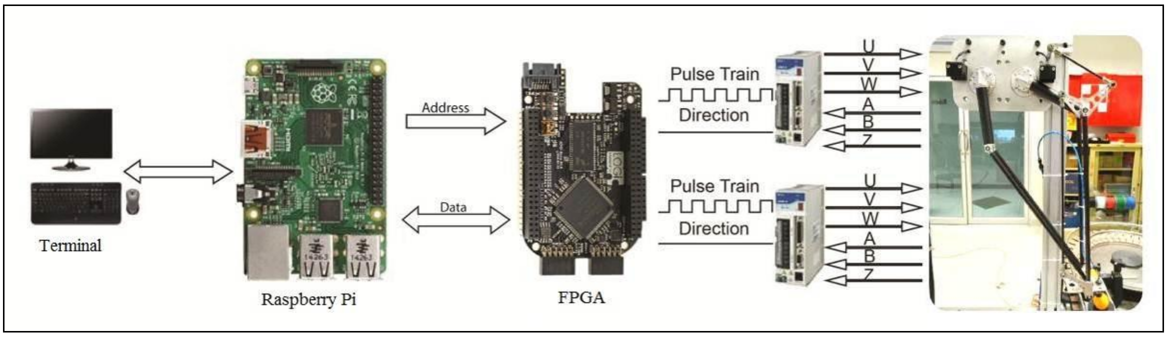

Figure 3. Proposed Raspberry Pi & FPGA connection

To control the robot, there are 2 main portions as follow:

The Raspberry Pi communicates with the user through the GUI interface. The look up table is being used to reduce the calculation on the mechanical part motion. The Raspberry Pi acts as front-end computer to send/receive the data to/from the FPGA.

The FPGA analyzes the received data for the Mapping Kinematics process and converts them into appropriate pulse train format for higher precision robot motion as shown in Figure 4.

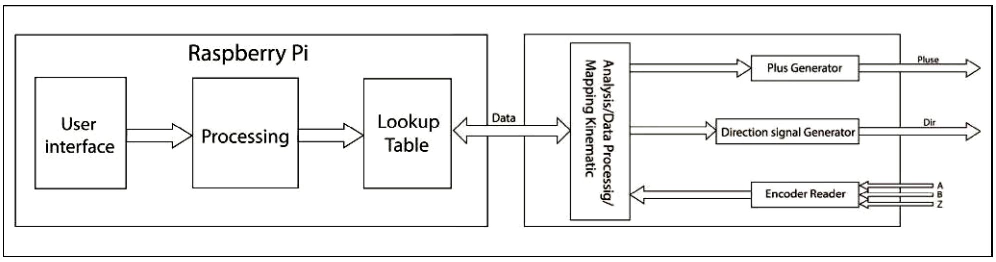

Figure 4. Proposed control system design on FPGA

The flow chart of control program on Raspberry Pi and the flow chart of motor control on FPGA are shown on Figure 5.

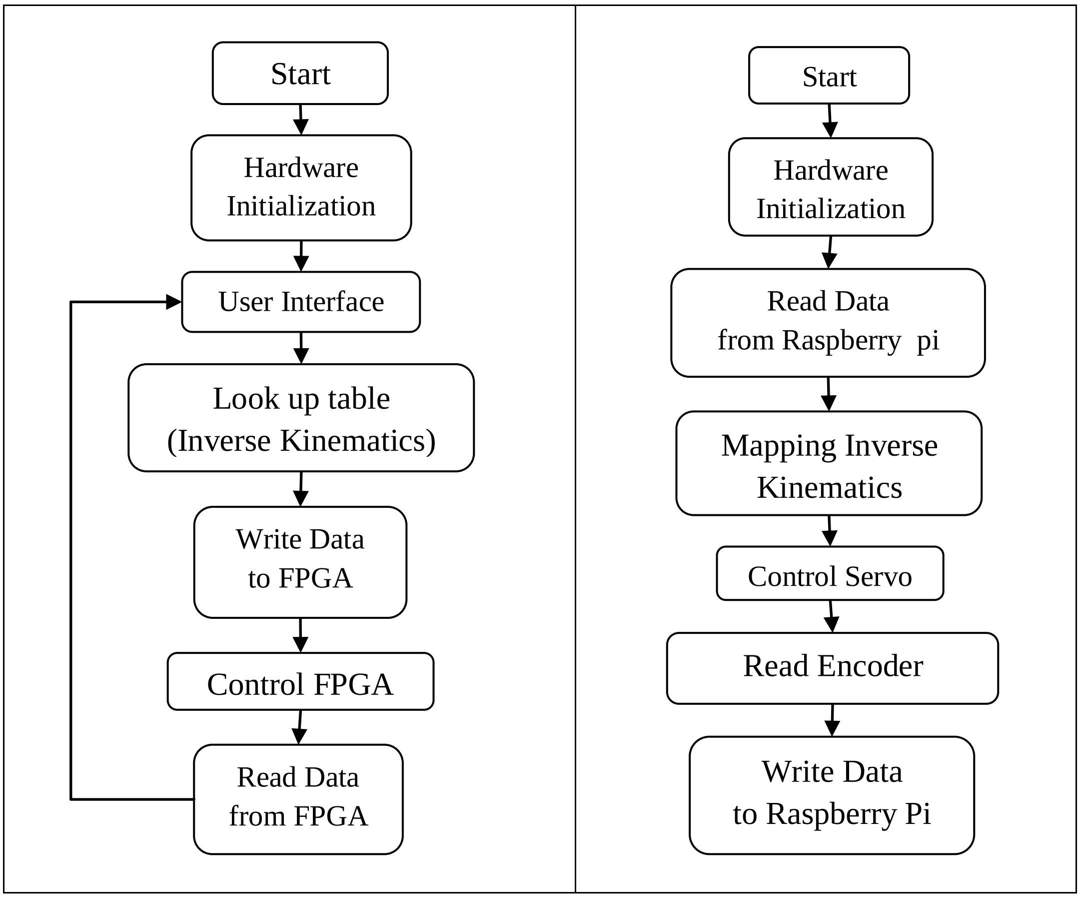

Figure 5. Raspberry Pi program flow chart and motor control flow chart on FPGA

Experimental results

The goal of controlling the Delta robot is to control the movement of robot arm to the required position by ways of controlling the movement on both the X and Z axes. The experiments are conducted as follow:

Ten tests of the 2-axes Delta robot which carried 0.2 kg from the starting point (0,0) to the end point (300,0) at the speed of 2.618 m/s on the XZ plane were conducted, the test results are shown in Figure 6.

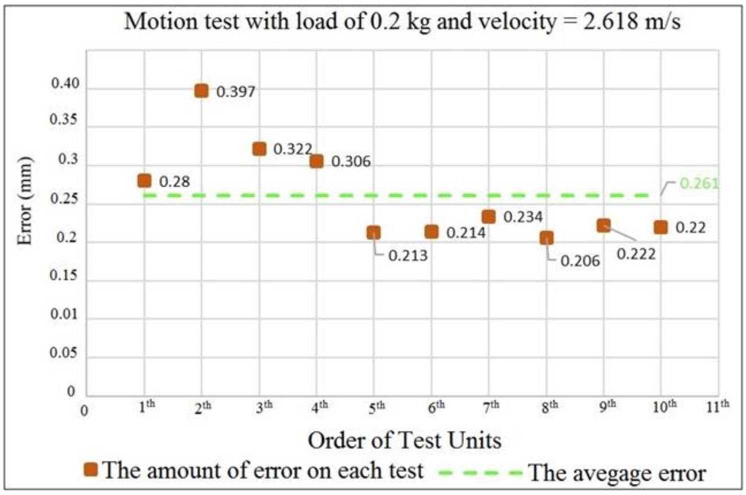

Figure 6. The positional errors for a test load of 0.2 kg at the speed of 2.618 meters per second

From Figure 6, the average of errors (from the tests) is equal to 0.261 mm. and the Standard Deviation (S.D.) is at 0.06 mm.

Summary

In this research, we propose embedded control system for the 2 axes Delta robot. The embedded control system is a combination of Raspberry Pi and FPGA. The advantages of this approach are: Raspberry Pi is a small size PC at low power consumption with Linux open source operating system, the reprogrammable FPGA offers the flexibility of control system design within the FPGA itself. The look up table approach speeds up the execution of dynamic equations for the Delta robot movements. A test load of 0.2 kg at the velocity of 2.618 m/s, the S.D. of the end effector's position is within 0.06 mm.

REFERENCES

[1] Xingqiang He, Zhengdong Wang, Haitao Fang, Kai HeAn, RuxuDu "Embedded Robot Controller Based on ARM and FPGA" 2014 IEEE

[2] Min Xu , Wenzhang Zhu, Ying Zou "Design of a Reconfigurable Robot Controller Based on FPGA" 2008 IEEE DOI 10.1109/SEC.2008.33

[3] Lin Zhang, Peter Slaets and Herman Bruyninckx "An Open Embedded Hardware and Software Architecture Applied to Industrial Robot Control" Proceedings of 2012 IEEE International Conference on Mechatronics and Automation August 5 - 8, Chengdu, China

[4] Mahfuzah Mustafa, Rosita Misuari, Hamdan Daniyal "Forward Kinematics of 3 Degree of Freedom Delta Robot" The 5th Student Conference on Research and Development –SCOReD 2007 11-12 December 2007, Malaysia

[5] Min-Kuang Wu, Ying-Shieh Kung, Yi-Hsien Huang, Tz-Han Jung "Fixed-Point Computation of Robot Kinematics in FPGA" 2014 International Confernce on Advanced Robotics and Intelligent System (ARIS 2014), June 6-8 2014,Nat Taiwan Univ. of Sci. and Tech, Taipei, Taiwan, R.O.C.

[6] Viren Pereira, Vandyk Amsdem Fernandes and Junieta Sequeira "Low Cost Object Sorting Robotic Arm using Raspberry Pi" 2014 IEEE Global Humanitarian Technology Conference - South Asia Satellite (GHTC-SAS) | September 26-27, 2014 | Trivandrum

[7]Sung-nam Oh, Sung-Ui Lee, and Kab-ll Kim "Design of a Biped Robot Using DSP and FPGA" International Journal of Control, Automation, and Systems Vol. 1, No. 2, June 2003stems Vol. 1, No. 2, June 2003