IJCTA, 9(12), 2016, pp. 5687-5695

Investigation and Evaluation of Embedded Controller Nodes of the Robotic Arm for Industrial Automation

***R. Puviarasi **Mritha Ramalingam**

Abstract : Robotic arm is widely used for industrial automation. Most programming methods used today involve some degree of programming knowledge or involve creating an entire three dimensional workspace to virtually program the robotic arm. The two major key drawbacks preventing from a more widespread use of robotic arm is the cost involved and the difficulty in programming the arm as per the unique requirements of the job at hand. To address these two limitations, this research work aims at reducing the cost of the industrial arm by using multiple low cost embedded processors working in a coordinated and distributed fashion. Also, this work implements a master-slave based control system for real time control and programming of the robotic arm. This proposed master-slave control system is analogous to lead by nose programming method and overcomes the mentioned limitations.

Keywords : Robot, Node, industrial automation, sensor.

1. INTRODUCTION

Robotic arm is electro-mechanical device with a kind of motorized parts which are generally programmable [1]. A robotic arm is said to have similar functions as human arm. The parts of an arm are connected by joints. These joints allow the arm for articulated motion or linear displacement. Robotic arm is used to move a complex object from one place to another. A major example of this robotic arm is in industries requiring moving heavy objects.

An industrial robot is a mechanically controlled and it is usually reprogrammable in more than three axes. There are different kinds of robotic arms involved in industries today. The robots could be applied in industries that involve welding, painting, assembly, packaging and other applications. These applications require elevated endurance, pace, and accuracy. The robotic arm can be classified into many categories depending on their kinematic configuration. The kinematics causes the motion of the robot arms [2].

A Cartesian robot is a robot arm with three prismatic joints. The joint axes are concurrent with a Cartesian coordinator. It is used in industries requiring high difficult jobs, handling machine tools and arc typed welding. In a Cylindrical robot, the axes form a cylindrical coordinate system. It is used for assembly operations, handling spot welding, and die casting machines. Spherical robot is used in fettling machines and gas welding. Articulated robot is used for spray painting. Parallel robot is used in mobile platform handling cockpit flight simulators. This paper describes the implementation of robotic arm for the use of industrial automation.

* Department of Electronics & Communication Engineering, Saveetha School of Engineering, Saveetha University, Chennai, India.
**Faculty of Computer Systems & Software Engineering, University Malaysia Pahang, Malaysia.

The current programming methods employ some form of programming languages to program the robot. This requires skilled programmers and has some set up time associated with the programming time. Another factor discouraging the widespread use of robotic arm is the cost of the robotic arm. Thus this paper aims to address these two concerns by employing some new approaches including master slave control based programming and using multiple low cost embedded controllers instead of a single powerful computer controlling the arm.

The paper is organized as follows. Section 2 defines the parameters of robotic arm. Section 3 describes the design and evaluation of angle sensor of robotic arm. Section 4 presents the evaluation of control algorithm and performance. Section 5 concludes the paper with future research directions.

2. DEFINING PARAMETERS OF A ROBOTIC ARM

Robots are competent with feats of strength, speed, and apparently smart decisions. The major advantage of automating the robotic arm is to complete the complex industrial jobs with high speed and negligible errors. Figure 1 shows the robotic arm used for the evaluation. There are many dimensions on which the parameters of robotic arm can be defined.

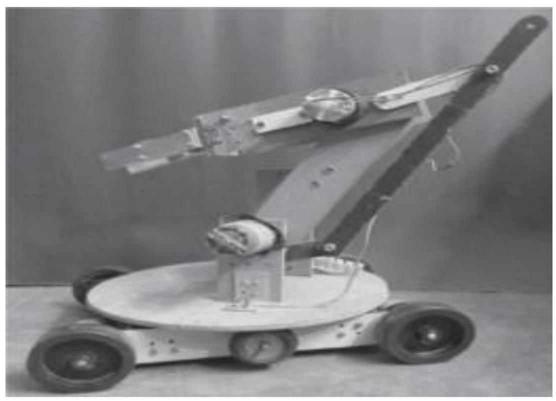

Fig. 1. Robotic arm used for the evaluation

Kinematics of robotic arm is the definite arrangement of axis joints in the robot. Kinematics determines the possible motions of robot arm. The payload was measured by increasing the weight till the robot can lift. The movement of axes of the arm is defined in terms of angular speed of each axis.

Accuracy is tested by measuring the current position of the arm with its commanded position. The accuracy of the arm varies according to speed and motion within the workplace with payload.

Repeatability of robot arm is tested by moving the arm from its home position to other different locations alternately. Accuracy and repeatability are two different measures.

The base of robot arm is powered by DC motors. The details of the gear motors used are given in Table 1.

Table 1. Characteristics of the DC geared motors used to actuate the slave robotic arm

Axis / actuator position	Number of motors used	Speed of the geared motor in revolutions per minute (RPM)	Effective revolution speed of the axis in revolutions per minute (RPM)	Effectiv e torque of the axis in Kg/cm	No load RPM of the non geared motor
Axis 1	4	200	15	20	1200
Axis 2	2	3.5	3.5	25	1200
Axis 3	1	10	10	5	1200
Gripper	1	10	NA	5	1200

Generally, Repeatability is the most vital measure for a robot arm. In an industrial process, Repeatability is subject to the accuracy of the end effectors. For example, if robot arm is commanded to pick a screw by its head, the screw could be at a random angle. A successive attempt to put in the screw into arm's hole could easily fail.

For more complicated applications, such as welding and spray painting, motion must be constantly controlled to follow a path in gap, with controlled direction and velocity. Some of the robots use electric stepper motors as power source and others use hydraulic actuators. The electric motors are faster but the hydraulic are stronger and highly advantageous in complicated applications.

Degree of freedom of robotic arm is quite important to understand. The degree of freedom of a robotic arm is the joint on the arm. It is the place where the arm could bend, translate or rotate from its position.

3. DESIGN AND EVALUATION OF ANGLE SENSOR

A sensor measures the physical quantity and converts it into equivalent electrical signal. The user of the robotic arm, in order to move the robot should make a hand movement. This different movement is sensed by potentiometer. The potentiometer gives the feedback in the form of voltage.

A low cost angle sensor was fabricated from a potentiometer. The potentiometer used was a single turn 10 KΩ linear potentiometer. The potentiometer has a turning range of 300 degrees. This sets the limit on the maximum measurable axis rotation. The design of the potentiometric angle sensor required the use of filtering capacitors to smoothen out the noise in the circuit. This filtering out of the noise is required as this noise could be read by the sensitive 10 bit Analog to Digital Converter (ADC) and injects measurement errors in the system.

The overall block diagram showing the interfacing of the various modules is shown in Figure 2. The communication bus is an I2C bus transferring using a modified I2C data format for increased bus data throughput. Figure 3 illustrates the data transfer when the master slave system is working in programming mode.

data values

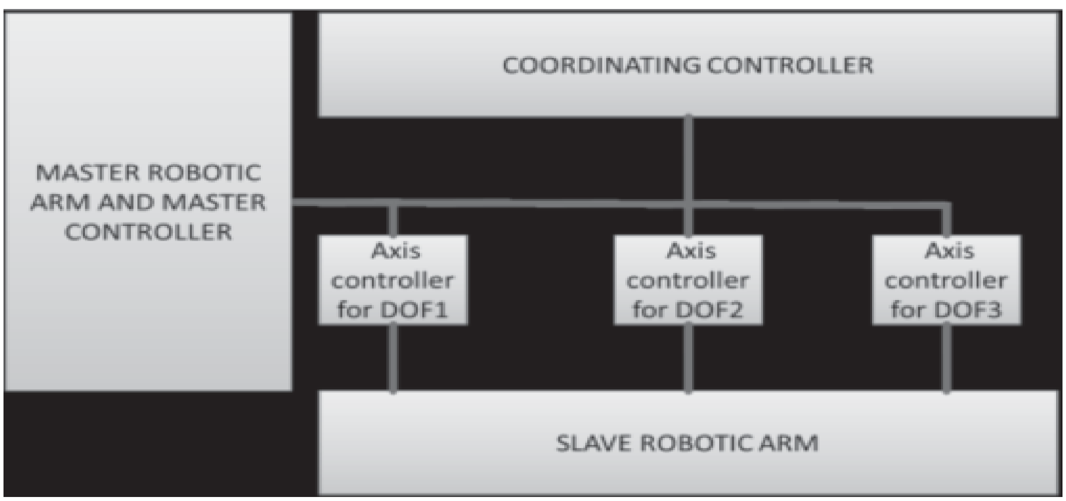

Fig.2 Overall block diagram showing the interconnection between various embedded nodes

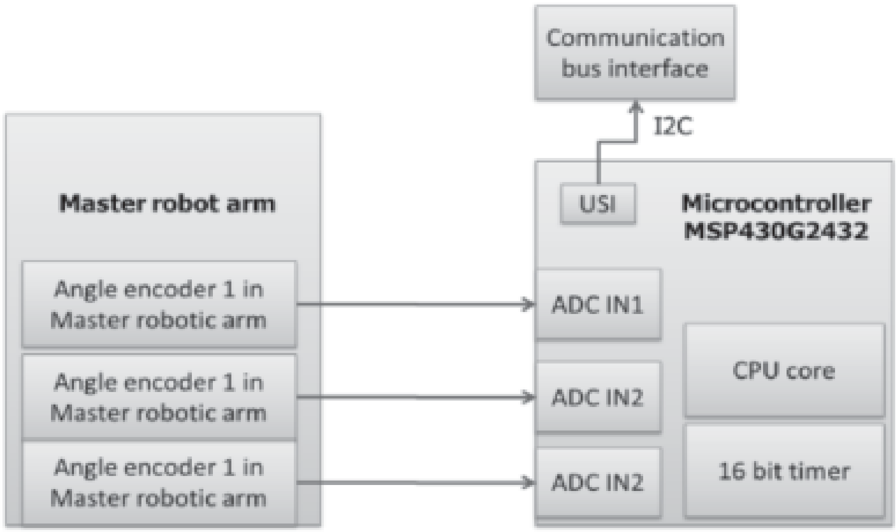

Fig. 3. Data transfer when the system is in programming mode

The master controller node is connected to the master robotic arm and reads the values of the master robotic arm angle sensors. The microcontroller in this node consists of an MSP430G2432 microcontroller. The angle encoder inputs are sampled repeatedly by the ADC port and the data read is verified to be within the maximum and minimum limits. If the data are valid and within the limits, they are transmitted via I2C along with the precise time stamp generated by the timer unit. The timestamp is used to order the data send by the other controller nodes and prevents swapping of stray data by new data if any delay occurs due to the communication interface. The configuration of angle sensor evaluations are shown in Fig.4.The I2C data format used is modified to suit the needs of the paper and is discussed in detail in future sections.

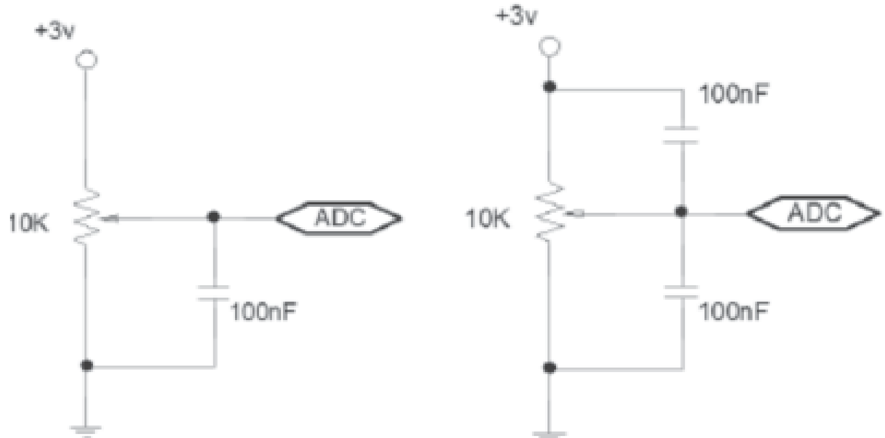

Fig. 4. Configurations of angle sensor evaluated.

Clockwise 1. The angle sensor consisting of one filtering capacitor to reduce noise.

Clockwise 2. The sensor designed with two capacitors which has marginally better performance than the one capacitor design.

Clockwise 3. The design with no capacitors giving noisy output.

Table 2 shows the maximum and minimum values of the angle sensors read via the MSP430 ADC. The table presents the results taken from a sample of 1000. ADC conversions done with no changes is performed on the angle sensor. The noise seen here is a sum of thermal noise present in the resistor and the ADC conversion noise. The effects of the angle sensor without any capacitor and with one and two capacitors connected to filter noise could be easily deduced from the table.

From Table 2, it is evident that with a single capacitor filtering the sensor output, there is a good improvement in the output noise. By adding another capacitor, the performance increases marginally. The design uses two capacitors filtering the potentiometer output before they could be read via the ADC.

Table 2. Noise measured with three configurations of angle sensor design

	Maximum ADC voltage in a sample of 1000 conversions Volts	Minimum ADC voltage in a sample of 1000 conversions. Volts	Maximum error (Maximum – minimum ADC voltage) Volts
Without capacitor	0.821	0.756	0.065
With one capacitor	0.784	0.798	0.014
With two capacitors	0.781	0.792	0.011

I2C communication is used to transfer the data serially to other controller nodes in the paper. I2C communication is device based and addresses a data to each deice. This would require multiple I2C frames to transfer data to each and every controller node. To overcome this and to increase the data throughput, a modified frame format was developed. The modified I2C transmission does not employ a single node intended transmission as traditional I2C. The I2C transmission is of broadcast type where all the data is send as a single frame to a common broadcast address. The I2C frame transmitted is shown in Fig.5. This method is employed to reduce the address overheads which arise when each node is addressed individually.

	Start	Broadcast address + write (0x91)	A	Time stamp	A	Data 0 node 1	A 1	Data 1 Node 1	A 1	Data 0 node 2	A 2	Data 1 Node 2	A 2	Data 0 node 3	A 3	Data 1 Node 3	A 3	Stop
Number of clocks per field	1	8	1	8	1	8	1	8	1	8	1	8	1	8	1	8	1	1

Fig. 5. Modified I2C Data format.

This modified I2C messaging format reduces the communication overhead by removing the transmission of separate addresses for data transfers, thereby reducing the communication delay which would otherwise hinder the real-time behavior of the system.

4. EVALUATION OF THE CONTROL ALGORITHM AND PERFORMANCE

The design was evaluated using a custom made three axis electrically actuated revolute robotic arm. This robotic arm was interfaced with the embedded controller nodes and the system was evaluated to infer the real time behavior, lag and the error of the system.

To evaluate the system, UART communication was established with a PC to study the various parameters within the system in a real time manner with minimal design intrusion. Figure 6 shows the Tera Term trace window showing the trace in progress. The results discussed were obtained using this real time data output.

The UART trace method was used to get the angle sensor values of the master and the slave robotic arm for a single degree of freedom. These values are tabulated in Table 3. . The table also shows the calculated ADC count value and the calculated angle sensor output value. The calculations used to arrive upon these values are as follows.

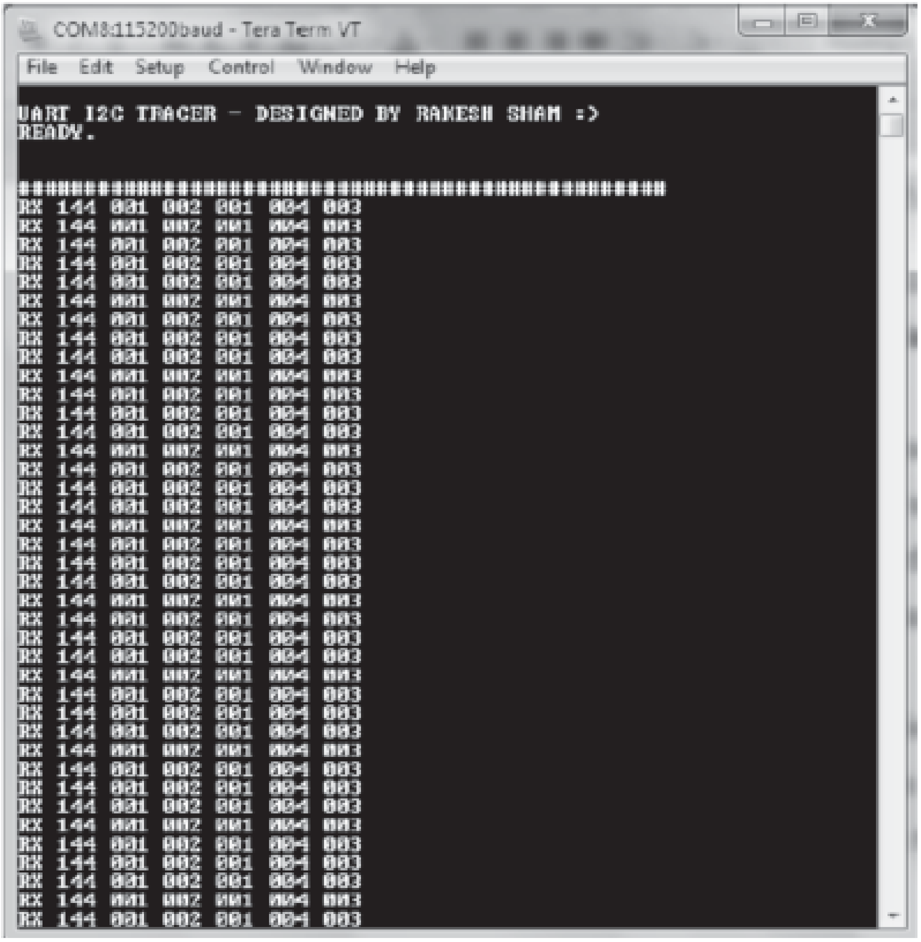

Fig. 6. Tera Term UART trace showing the axis 1 angle sensor data from the master controller

$$\text{ADC count} = (\text{Data } 0 * 256) + \text{Data } 1$$

Where Data 0 and Data 1 are the high and low data bytes send via the I2C. the formula gives a maximum of 1023 for the 10 bit values of the ADC10.

$$\text{Anglevalue} = \text{ADC count} * (300/1023)$$

This formula gives the precise angle output of the axis. The table also shows the error in the system. This error is calculated from the values from the master controller and the value measured at the slave axis.

Table 3. Comparison of the axis data from the master and the slave robotic arm for a single axis (axis 1)

	Master Axis 1 reading		Slave Axis 1 value		Master Axis 1 ADC count (counts)	Slave Axis 1 ADC count (counts)	Master Axis 1 angle value (Degree)	Slave Axis 1 angle value (Degree)	Error (Degree)
Time Stamp	Data 0	Data 1	Data 0	Data 1					
1	3	250	1	163	1018	419	298.5337243	122.8739003	175.6598
2	3	250	1	169	1018	425	298.5337243	124.6334311	173.9003
3	3	250	1	174	1018	430	298.5337243	126.0997067	172.434
4	3	250	1	165	1018	421	298.5337243	123.4604106	175.0733
5	3	250	1	165	1018	421	298.5337243	123.4604106	175.0733
6	3	251	1	166	1019	422	298.8269795	123.7536657	175.0733
7	3	223	1	173	991	429	290.6158358	125.8064516	164.8094
8	3	136	1	172	904	428	265.1026393	125.5131965	139.5894
9	3	95	1	175	863	431	253.0791789	126.3929619	126.6862
10	3	44	1	167	812	423	238.1231672	124.0469208	114.0762
11	2	218	1	169	730	425	214.0762463	124.6334311	89.44282
12	2	174	1	171	686	427	201.1730205	125.2199413	75.95308
13	2	133	1	171	645	427	189.1495601	125.2199413	63.92962
14	2	63	1	173	575	429	168.6217009	125.8064516	42.81525
15	2	33	1	160	545	416	159.8240469	121.9941349	37.82991
16	1	254	1	160	510	416	149.5601173	121.9941349	27.56598
17	1	172	1	141	428	397	125.5131965	116.4222874	9.090909
18	1	124	1	125	380	381	111.4369501	111.7302053	-0.29326
19	1	76	1	109	332	365	97.36070381	107.0381232	-9.67742
20	1	22	1	76	278	332	81.52492669	97.36070381	-15.8358
21	1	7	1	36	263	292	77.12609971	85.63049853	-8.5044
22	1	6	1	5	262	261	76.83284457	76.53958944	0.293255
23	1	7	0	206	263	206	77.12609971	60.41055718	16.71554
24	1	7	0	170	263	170	77.12609971	49.85337243	27.27273
25	1	6	0	128	262	128	76.83284457	37.53665689	39.29619
26	0	220	0	66	220	66	64.51612903	19.35483871	45.16129
27	0	179	0	37	179	37	52.49266862	10.85043988	41.64223
28	0	125	0	9	125	9	36.6568915	2.639296188	34.0176
29	0	38	0	0	38	0	11.14369501	0	11.1437
30	0	0	0	4	0	4	0	1.173020528	-1.17302
31	0	0	0	5	0	5	0	1.46627566	-1.46628
32	0	0	0	0	0	0	0	0	0
33	0	0	0	1	0	1	0	0.293255132	-0.29326
34	0	0	0	36	0	36	0	10.55718475	-10.5572
35	0	0	0	76	0	76	0	22.28739003	-22.2874
36	0	0	0	148	0	148	0	43.40175953	-43.4018
37	0	0	0	231	0	231	0	67.74193548	-67.7419
38	0	4	1	46	4	302	1.173020528	88.56304985	-87.39
39	0	38	1	86	38	342	11.14369501	100.2932551	-89.1496
40	0	85	1	118	85	374	24.92668622	109.6774194	-84.7507
41	0	184	1	152	184	408	53.95894428	119.6480938	-65.6891
42	0	231	1	177	231	433	67.74193548	126.9794721	-59.2375
43	1	20	1	183	276	439	80.93841642	128.7390029	-47.8006
44	1	111	1	219	367	475	107.6246334	139.2961877	-31.6716
45	1	161	1	236	417	492	122.28739	144.2815249	-21.9941
46	1	201	1	249	457	505	134.0175953	148.0938416	-14.0762
47	2	29	2	23	541	535	158.6510264	156.8914956	1.759531
48	2	57	2	25	569	537	166.8621701	157.4780059	9.384164
49	2	78	2	50	590	562	173.0205279	164.8093842	8.211144
50	2	98	2	69	610	581	178.8856305	170.3812317	8.504399
51	2	101	2	62	613	574	179.7653959	168.3284457	11.43695
52	2	101	2	69	613	581	179.7653959	170.3812317	9.384164
53	2	84	2	61	596	573	174.7800587	168.0351906	6.744868
54	2	51	2	67	563	579	165.1026393	169.7947214	-4.69208
55	2	28	2	63	540	575	158.3577713	168.6217009	-10.2639
56	2	13	2	28	525	540	153.9589443	158.3577713	-4.39883
57	2	12	2	14	524	526	153.6656891	154.2521994	-0.58651
58	2	17	1	243	529	499	155.1319648	146.3343109	8.797654
59	2	17	1	179	529	435	155.1319648	127.5659824	27.56598
60	2	17	1	143	529	399	155.1319648	117.0087977	38.12317

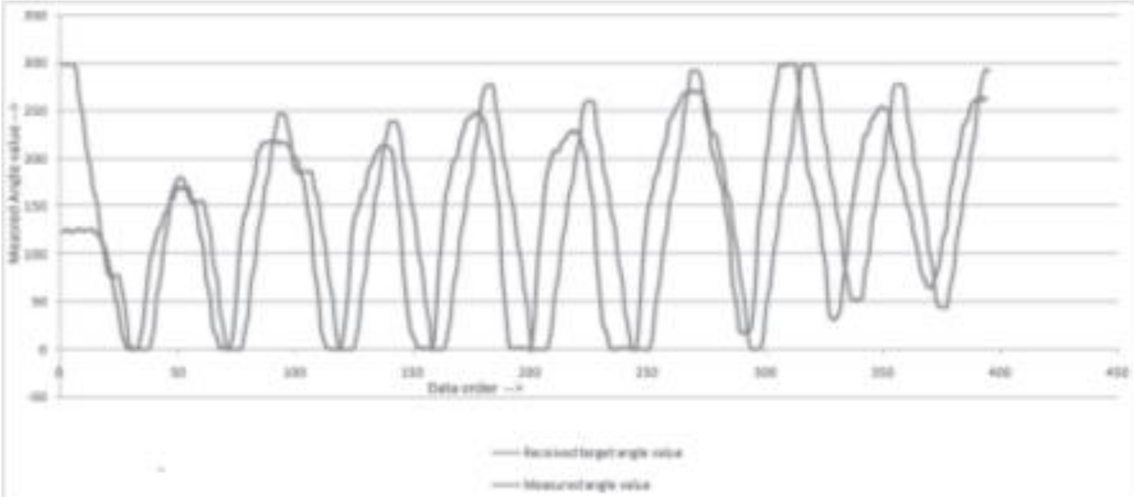

Fig. 7 Graph shows the angle sensor values of the axis 1 of master and slave robotic arms. The lag between master and slave arms is evident from the graph

The graph in Fig.7 shows the output data values. A sample of 395 data values are utilized from the UART trace. The graph plots the received target angle value from the master controller node and the measured slave angle value from the axis controller node in the same graph to allow comparison of the lag of the system.

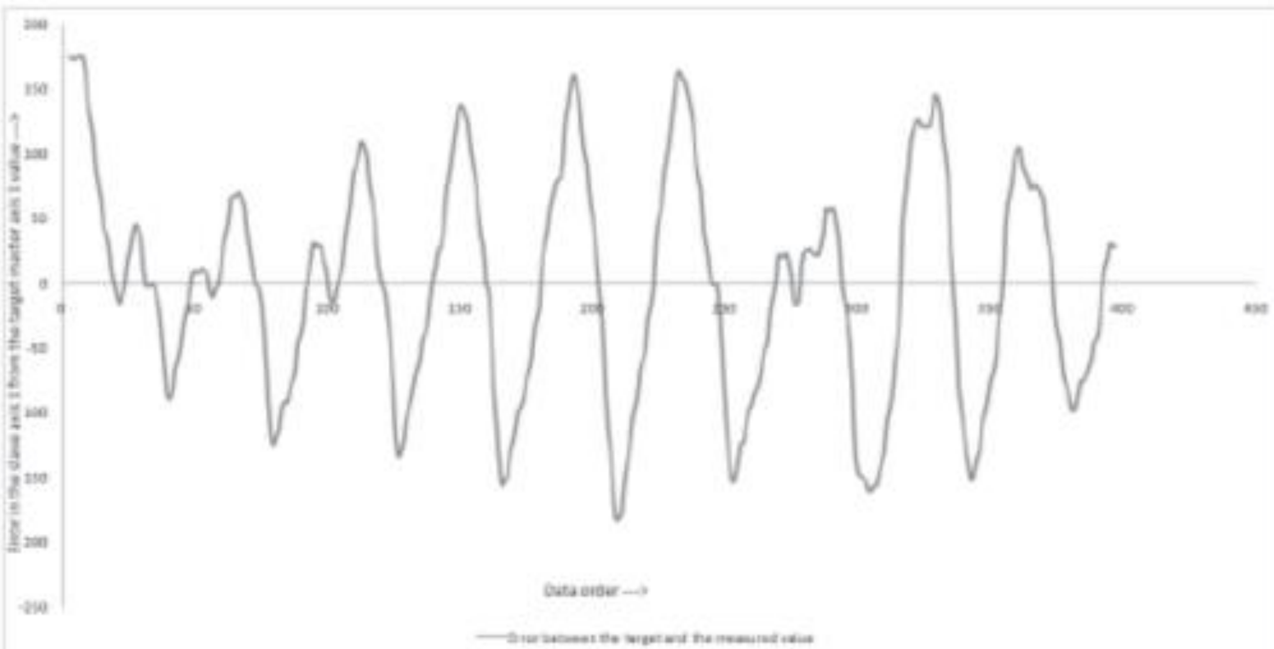

Fig. 8 Graph showing the error between the angle sensor values of the axis 1 of the master and the slave robotic arms

The figure 8 shows the variation of the error in the system as a function of time. Also, the graph specifies that the error is more for high rapid movements in the master arm compared to the slave arm.

Because of the higher error values, this can only be used for jobs that require less precision. The precision could be increased by using more precise angle encoders and by using a higher speed communication bus to transfer the master data.

5. CONCLUSION

In this paper, a master-slave based method of programming and control has been designed using multiple embedded controllers and the performance is evaluated. It is designed multiple nodes based on MSP430G2432 microcontroller and interfaced them to control the slave robotic arm. This research work had designed and evaluated the control algorithm for coherently controlling all the nodes in real time. Further, this work was designed and built a three axis revolute robotic arm using DC geared motors and custom designed angle sensor circuitry. The method has been evaluated electrically on a freedom of three degree actuated robotic arm and the results are discussed. The concept could be expanded further to include more complex arm geometries involving increased degrees of freedom.

Another interesting work would be to expand the scope to a mobile robotic arm platform. This would involve newer challenges as to navigation of the mobile platform and collision avoidance, but the possibilities of such a programmable mobile robotic arm are very high.

6. REFERENCES

1. A Rajan, A. Thomas, R Mathew, "A comparative performance analysis of ARM based web servers with integrated and external Ethernet interfaces for industrial applications", International Journal of Computer Applications, Vol 44, No.21, April 2012, 0975 – 8887.
2. J Iqbal, R Islam, and H Khan, "Modeling and analysis of a 6 DOF robotic arm manipulator", Canadian Journal on Electrical and Electronics Engineering, Vol 3, No 6, July 2012, 300–306.
3. J Kober and J Peters "Learning motor primitives for robotics", 2009 IEEE International Conference on Robotics and Automation
4. G. Hirzinger, A. Albu-Schaffer, M. Hiihnle, I. Schaefer, N. Sporer "On a new generation of torque controlled light-weight robots". Proceedings of the 2001 IEEE International Conference on Robotics & Automation Seoul, Korea. May 21-26, 2001
5. I Yamano, K Takemura, K Endo, T Maeno, "Method for controlling Master-Slave robots using switching and elastic elements", IEEE Intemational Conference on Robotics and Automation, Washington, DC May 2002
6. G Tonietti, R Schiavi and A Bicchi "Design and control of a variable stiffness actuator for safe and fast physical human/ robot interaction" IEEE International Conference on Robotics and Automation, 2005.
7. M Kovac, M Fuchs, A Guignard, J Zufferey, D Floreano , "A miniature 7g jumping robot", 2008 IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, 2008
8. S Wolf and G Hirzinger, "A new variable stiffness design: matching requirements of the next robot generation", 2008 IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, 2008
9. R. Schiavi, G. Grioli, S. Sen, A. Bicchi, "VSA- II: a Novel Prototype of Variable Stiffness Actuator for safe and performing robots interacting with humans", IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, 2008.
10. P Pastor, H Hoffmann, T Asfour, and S Schaal, "Learning and generalization of motor skills by learning from demonstration", 2009 IEEE International Conference on Robotics and Automation Kobe International Conference Center Kobe, Japan.
11. C. C. Kemp, A. Edsinger, and E. Torres-Jara, "Challenges for robot manipulation in human environments [Grand challenges of robotics]," IEEE Robotics and Automation Magazine, vol. 14, no. 1, pp. 20–29, 2007.