Development of a New 5 DOF Mobile Robot Arm and its Motion Control System

Hyeung-Sik Choi*
Division of Mechanical and Information Engineering, Underwater Vehicle Research Center, Korea Maritime University
Changman Lee
Graduate Student, Department of Mechanical Engineering, Korea Maritime University, Korea
Changhun Chun
DASA Tech, 420-734 Wonmi-Ku, Bucheon, Kyungki-do, Korea

In this paper, a new revolute mobile robot arm with five degree of freedom (d.o.f) was developed for autonomous moving robots. As a control system for the robot arm, a distributed control system composed of the main controller and five motor controllers for arm joints was developed. The main controller and the motor controllers were developed using the ARM microprocessor and the TMS320c2407 microprocessor, respectively. A new trajectory tracking algorithm for the motor controllers was devised employing pre-generated off-line trajectory data. Also, a 3-D simulator based on the openGL software to simulate the motion of the robot arm was developed. To validate the performance of the robot system, experiments to track a specified trajectory were performed.

Key Words : Motion Control System, Mobile Robot Arm, Microprocessor, Trajectory Profile

1. Introduction

Recently, autonomous moving robots with the operating robot arm working in hazardous environments to substitute for human beings, helping old or disable people, and guiding people in the museum or exhibition have been developed and operated in the human environments. As a theoretical research on the mobile manipulator for coordination and control of the manipulator suitable for farming was studied by Thompson et al. (1995), an integrating control of locomotion and manipulation to develop applications in outdoors was studied by Koyachi et al. (1995) and the mobile manipulator to derive coordinated motion along a given desired end-point trajectory was studied by Huang, Sugano, and Tanie (1998). In experimental approach, a heavy industrial manipulator PUMA560 was employed and its global motion from one point to another taking into account different constraints was studied by Perrier, Dauchez, and Pierrot (1998), and a study on the design and coordination of the mobile manipulator system of humanoid Hadaly-2 was performed by Morita, Shibuya, and Sugano (1998).

Especially, out of mobile robots, the biped walking robot has a limited load capacity. Since it should carry the control system, batteries, and sensors with itself, light-weighted robot arms are required. Light-weighted robot arms have been developed and applied to biped walking robots such as ASIMO (asimo.honda.com), HRPII stu-

* Corresponding Author,
E-mail : hchoi@mail.hhu.ac.kr
TEL : +82-51-410-4297; **FAX** : +82-51-000-000
Professor, Division of Mechanical and Information Engineering, Underwater Vehicle Research Center, Korea Maritime University. (Manuscript **Received** September 20, 2005; **Revised** May 18, 2006)

died by Harada (2005), and KHR studied by Kim et al. (2003).

In the control system, the same kind of microprocessor for the robot arm and autonomous moving base robot is required since they are under the same control structure. Though a high capacity microprocessor such as that of PC is required due to fast processing and large memory capacity in the control system of the autonomous moving robot including the robot arm, it is almost impossible to board that processor due to a large size and heavy weight. Especially, not like industrial robots, since the robots move autonomously with the on-boarded control system, batteries, and motor drivers, their control system are constrained to be small and light but have high performance. In this reason, the most appropriate microprocessor among small or medium-sized ones should be selected and be composed as an efficient control system. Therefore, out of microprocessor devices Atmega (ATmega128 User Manual), Intel80xx (www.intel.com), ARM (www.samsung.com), PIC (www.microchip.com), and dsp24xx (TMS320LF2407A User Guide), the most appropriate device should be selected and composed in a efficient control structure. Out of these, the fastest processor is DSP-based control system, and one of the implementation was performed by Han and Hashimoto (2006).

So far, a number of robot arms for the autonomous moving robots such as ASIMO, HRPII, and KHR2 were developed, but the detail information on the control system was not open. In this reason, for a new robot arm of an autonomous moving robot, a new control system should be developed using its own microprocessor. In addition to this, in developing the control system, a new position and velocity control algorithm with a trajectory planning using the selected microprocessor should be developed. Also, a kinematics model for the robot arm should be set up for actual operation and its control.

In this paper, we designed and constructed a light-weighted five d.o.f mobile robot arm as in Fig. 1 and its control system, and set up its kinematics model. As a control system, we composed a modular distributed control system for the joint motors utilizing the TMS320C2408 microprocessor which is cheap, compact, and has low capacity but high performance. Also, as a main controller to control the whole process of the robot arm, we composed an embeded control system loaded on the robot arm utilizing the ARM microprocessor. Since composing a new control system for the robot arm using one-chip microprocessors has small memory capacity and relatively slower processing time than those of industrial robots, we developed a new position and velocity control system including the control algorithm. In addition to this, we developed a 3D graphic simulator which can give order data to the main controller and accept monitoring data from the distributed motor control system in real time. Also, we developed a communication system connecting the PC, the main controller, and the motor controller.

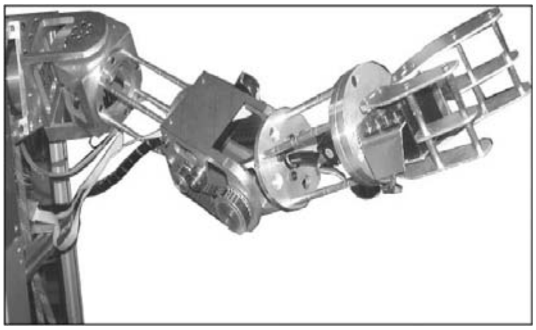


Fig. 1 d.o.f revolute robot arm

2. An Analysis on the Kinematics of the Robot Arm

We performed an analysis on the kinematics of the light-weighted five d.o.f mobile robot arm to identify joint angles in accordance with the position of the end effector in the Cartesian coordinate. After defining a homogeneous transformation matrix about the position of the end effector with respect to the reference coordinate through analysis on the kinematics, we obtained an analytic solution of the joint angles in accordance with the position in the Cartesian coordinate using a geometric approach.

2.1 Analysis on the kinematics

We established a coordinate as Fig. 2 to analyze the kinematic structure of the robot arm according to the Denavit-Hartenberg (D-H) notation referred to Spong, and Vidyasagar (1989). For convenience, the third axis was put on the second axis.

The obtained parameters of the four links are presented in Table 1 according to the established coordinate as in Fig. 2.

Where actual parameters of the robot are $d_1=$ 86 mm, $a_1=275$ mm, $d_4=318$ mm. A homogeneous transformation matrix A_i of the joint i and the joint i-th link are expressed as follows

$$A_{i1}=R_{z,\theta}\cdot T_{z,d}\cdot T_{x,a}\cdot R_{x,\alpha}=\begin{bmatrix} c_i & 0 & -s_i & 0 \\ s_i & 0 & c_i & 0 \\ 0 & -1 & 0 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

If the transformation matrix relating the coordinate of the end effector and the basis of the robot arm are substituted by the obtained parameters in Table 1 to Eq. (1) in sequence for $i=1, \cdots, 4$, then the following equation is obtained as

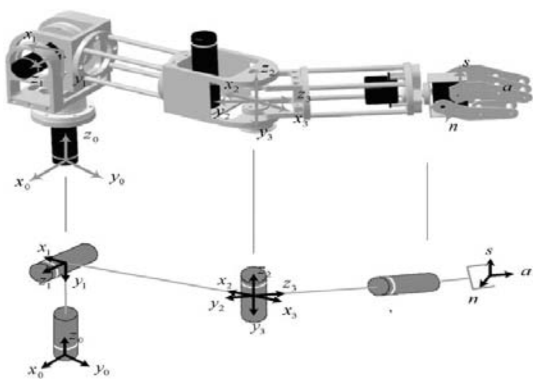


Fig. 2 Establishment of coordinate by D-H notation

Table 1 Parameter of robot arm

	θ_i	d_i	a_i	α_i
1	θ_1^*	d_1	0	90°
2	θ_2^*	0	a_2	90°
3	θ_3^*	0	0	90°
4	θ_4^*	d_4	0	0

$$T_0^4=A_1A_2A_3A_4$$
$$=\begin{bmatrix} (c_1c_2c_3-s_1s_3)c_4-c_1s_2s_4 & -(c_1c_2c_3-s_1s_3)s_4-c_1s_2c_4 & -c_1c_2s_3-s_1c_3 & (-c_1c_3s_3-s_1c_3)d_4+a_2c_1c_2 \\ (s_1c_2c_3-c_1c_3)c_4-s_1s_2s_4 & -(s_1c_2c_3+c_1s_3)s_4-s_1s_2c_4 & -s_1c_2s_3+c_1c_3 & (-s_1c_2s_3+c_1c_3)d_4+a_2s_1c_2 \\ -s_2c_3c_4-c_2s_4 & s_2c_2s_4-c_2c_4-c_2c_4 & s_2s_3 & s_2s_3d_4-a_2s_2+d1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where $c_i=\cos\theta_i$, $s_i=\sin\theta_i$.

2.2 Analysis on the inverse kinematics

For the solution of the joint parameters for continuous control of the robot arm, the analytic solution is better than the numerical solution which requires long calculation time and might cause no convergence. In this paper, we obtained an analytic solution for the joint angle of the robot arm using a geometrical method.

The solution of θ_2 in Fig. 3 is expressed as follows :

$$\begin{aligned}\theta_2&=A\tan(S_z, \sqrt{P_x^2+P_y^2}) \\ &=A\tan(P_z-d_1, \sqrt{P_x^2+P_y^2})\end{aligned} \quad (3)$$

Also, the projection of the robot arm to x_0y_0 suface results in Fig.4 to solve for θ_1 and θ_3.

Using the Cosine low, the following equation is obtained.

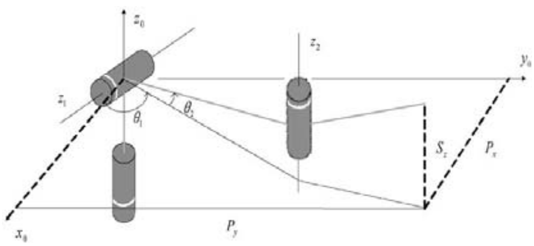


Fig. 3 Projection tosurface

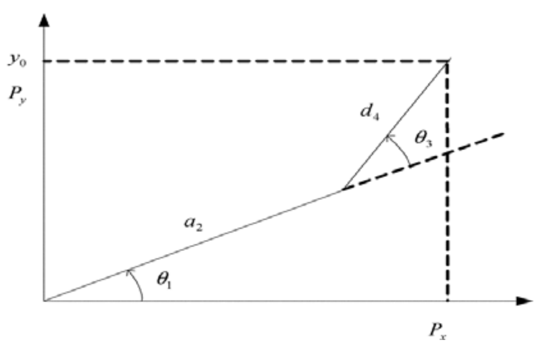


Fig. 4 Projection to surface

$$\cos\theta_3=\frac{P_x^2+P_y^2-a_2^2-d_4^2}{2a_2d_4}=D \quad (4)$$

and, as a result, $\theta_3=A\tan(D, \pm\sqrt{1-D^2})$ is obtained. Using this equation, we can solve for θ_1 as

$$\begin{aligned}\theta_1=&A\tan(P_x, P_y)\\&-A\tan(a_2+d_4\cos\theta_3, d_4\sin\theta_3)\end{aligned} \quad (5)$$

R_0^3 is the matrix of the robot arm, which is expressed as

$$R_0{}^3=\begin{bmatrix} c_1c_2c_3-s_1s_3 & c_1s_2 & -c_1c_2s_3-s_1c_3 \\ s_1c_2c_3+c_1s_3 & -s_1s_2 & -s_1c_2s_3+c_1c_3 \\ -s_2c_3 & -c_2 & s_2s_3 \end{bmatrix} \quad (6)$$

and

$$R_3^4=\begin{bmatrix} c_4 & -s_4 & 0 \\ s_4 & c_4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Solving for θ_4, we can use $R_3^4=(R_0^3)^TR$, where

$$(R_0^3)^T=\begin{bmatrix} c_1c_2c_3-s_1s_3 & s_1c_2c_3+c_1s_3 & -s_2c_3 \\ c_1s_2 & -s_1s_2 & -c_2 \\ -c_1c_2s_3-s_1c & -s_1c_2s_3+c_1c_3 & s_2s_3 \end{bmatrix} \quad (8)$$

where the matrix R is known rotation matrix whose parameters are decided by the orientation of the operation tool in the Cartesian coordinate.

$$R=\begin{bmatrix} (c_1c_2c_3-s_1s_3)c_4-c_1s_2s_4 & -(c_1c_2c_3-s_1s_3)s_4-c_1s_2c_4 & -c_1c_2s_3-s_1c_3 \\ (s_1c_2c_3-c_1s_3)c_4-s_1s_2s_4 & -(s_1c_2c_3+c_1s_3)s_4-s_1s_2c_4 & -s_1c_2s_3+c_1c_3 \\ -s_2c_3c_4-c_2s_4 & s_2c_3s_4-c_2c_4-c_2c_4 & s_2s_3 \end{bmatrix} \quad (9)$$

θ_4 has a number of solutions according to configuration of the robot arm.

3. The Control of the Robot Arm

3.1 The hardware structure of the controller for the joint motor

In this paper, we composed the control system capable of the trajectory control of the robot arm using the one-chip microprocessor with the small memory capacity and relatively slower processing time than PC. The developed total control system for the control of the robot arm consists of a host PC, main controller, joint motor controller, and communication system. The structure of the control system is shown in Fig. 5 where the host PC sends the order of the operation signal to the main controller, which sends the control signal to the joint motor controller through RS232 communication. The main controller composed of ARM board whose specification is shown in Table 2, was composed to send orders of sequential or continuous motion to the joint motor controller according to the trajectory plan of the robot arm received from the host PC and to transmit the encoder signal received from each joint motor to the PC through wireless communication system. The joint motor controller based on the TMS3204LF2407A microprocessor device whose specification is shown in Table 2 for the motor control was developed to control the servomotors,

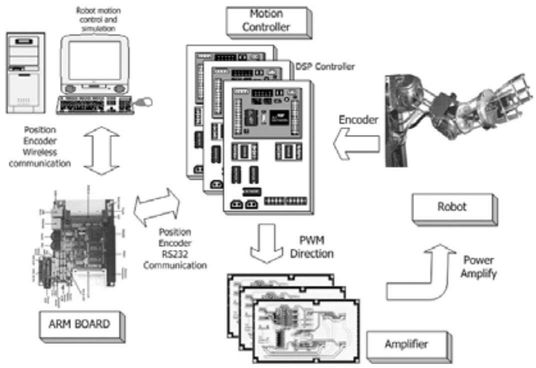


Fig. 5 Composition of motion controller

Table 2 Specification ot microprocessors

	TMS320LF2407	ARM
Bit	16bit	32bit
Speed	40MHz	400MHz
I/O	40channel	120channel
Memory	32K Word*16Bits Flash 2.5K Word*16Bits RAM	64M SDRAM 32M Flash

Table 3 Property of motion controller

Parts	Model name	Function
MCU	TMS320LF2407A	PWM, Encoder counter, UART, Timer, GPIO
Level Shifter	74LVHT4245	3.3V $\Longleftrightarrow$ 5V
DeadTime Generator	CD4538	Dual Precision, Monostable

and its properties are shown in Table 3.

The controller was built in a modular control board capable of controlling two motors as shown in Fig. 6. The controller was designed to generate the pulse width modulation (PWM) signal and the direction signal for the motor and to send them from the ARM board to the motor driver. The motor driver was developed to supply the power which is proportionally amplified with the duty ratio to the joint motors according to the PWM signal and the direction signal of the motor. In this paper, we constructed the motor driver by composing a bridge circuit with power transistors, which is shown in Fig.7.

In the control system, the PC sends the operation order of the robot arm through the wireless Ethernet communication system to the ARM board. It monitors the states of the joint motor of the robot arm, and manages the edition of total control program including control of input and output signals. The systematic flow diagram to explain whole structure of the control system is described in Fig. 8.

3.2 The trajectory of the joint motor and the control algorithm

For velocity control of the robot arm along the desired trajectory, it is important to design an adequate trajectory profile and to control the position and velocity of the joint motors. In this paper, we designed a new trajectory control system to control the position and velocity of the joint motors by using the TMS320LF2407A microprocessor device. The TMS320LF2407A microprocessor device is a fast processor but has limited memory capacity so that its RAM memory only contains about 1500 trajectory points. For tracking control of the motor along a trajectory, firstly a desired on-line target trajectory data composed of a set of points on the trajectory should be generated, which cause time delay and deteriorate the performance of the controller. In this reason, to reduce the calculation time of tracking points on the trajectory in the controller, we developed a new off-line table method including a trajectory data for the motors to track. By this method, a set of data of the points on the desired trajectory are

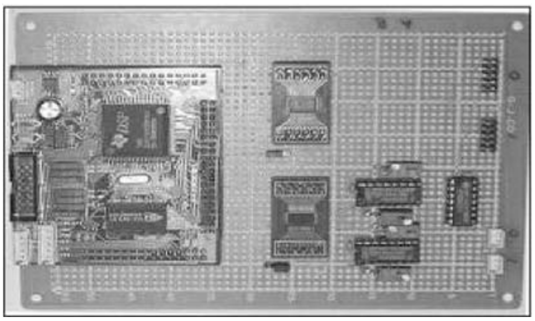


Fig. 6 Outlook of motion controller

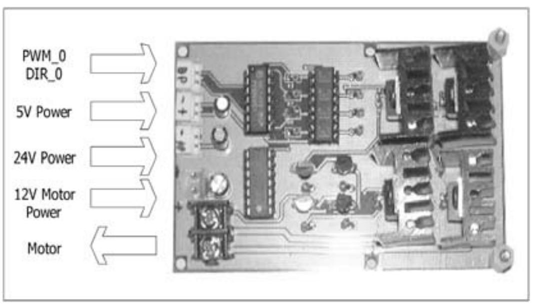


Fig. 7 Outlook of the motor driver

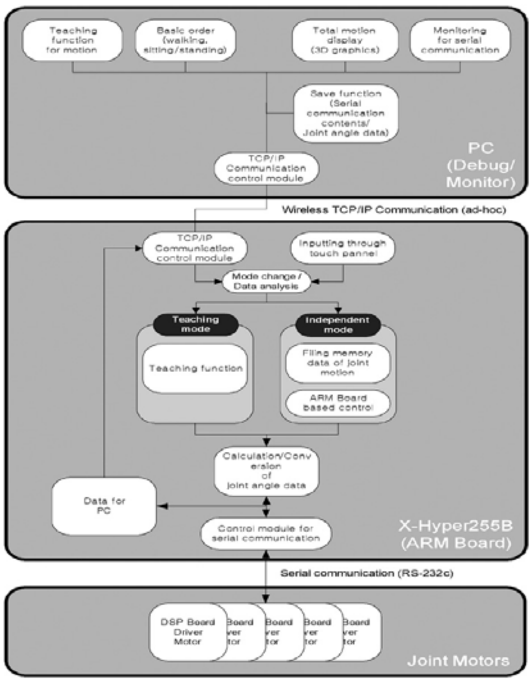


Fig. 8 The flow diagram of whole structure of the control system

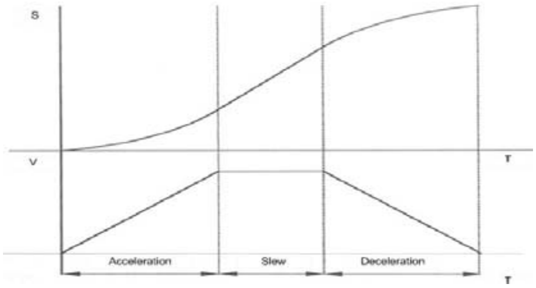


Fig. 9 Reference profile for motors

generated and formed as a table in advance. Employing the method, we designed a control algorithm to remove errors between the desired states and the actual states of the position and velocity of actuating motors along the trajectory. In this paper, we applied a position and corresponding trapezoidal velocity profile shown in Fig. 9.

The trajectory profile that the joint motor should track is composed of the position profile and the velocity profile which is its differential function. We made a table of off-line data composed of around 500 points on the trajectory and controlled the motor to pass the points with specified velocities. The tracking time between two points was set using the interrupt function in the microprocessor. As data of tracking points of the table increase, the velocity of the motor gets smoother. But, for faster interrupt frequency, high speed CPU is required. However, the TMS320LF2407A microprocessor has limited memory capacity and calculation speed. Therefore, we selected a near optimal 500 Hz interrupt frequency for the microprocessor through a large number of control experiments. A rotational velocity and a distance of the motors are designed using the interrupt function of microprocessor. The design method is explained according to the following equations.

The total distance for the motors to track is defined as

$$S_t = S_a + S_c + S_r \tag{10}$$

where S_t is the total distance, S_a and S_r are the distance during acceleration and deceleration, respectively, and S_c is a distance during the constant velocity. If S_t is decided by the inverse kinematics analysis, S_c can be decided by deciding S_a and S_r arbitrarily. Also, relations between the maximum velocity $V_{\max}$ and the deceleration velocity time T_a are expressed as follows:

$$S_a = S_r = V_{\max}{}^* T_a / 2 \tag{11}$$

The total arrival time of all the motors is expressed by the sum of the constant velocity time and the acceleration-deceleration time. Using this, the arrival time of the robot arm to destination is designed to coincide with that of the joint motors.

$$T_t = T_a + T_r + S_c / V_{\max} \tag{12}$$

where T_t is the total time, T_a is the acceleration velocity time, and I_p is the interrupt period. The numbers of interrupt generation at acceleration, deceleration, and constant velocity are T_r/I_p, T_a/I_p, and $S_c/(V_{\max}\ I_p)$, respectively, and the total interrupt numbers are the sum of them. We designed the velocity trajectory of the joint motors using the interrupt period and its numbers since the total interrupt generation numbers decide the number of rotational velocity of the motor.

We programmed a code to generate a table of off-line data of the reference velocity of the motors automatically for the profile of acceleration and deceleration trajectory. The data table for the position trajectory is generated by integration of the velocity data in the table at every interrupt period. The position profile for the acceleration and deceleration section in Fig. 8 is an S-type polynomial, and which are expressed by the following equations.

$$y = a + bt + ct^2 + dt^3 \tag{13a}$$

$$y' = b + 2ct + 3dt^2 \tag{13b}$$

$$y'' = 2c + 6dt \tag{13c}$$

Eqs. (13a), (13b), and (13c) represent a position, a velocity and acceleration profile for the acceleration section of the trajectory, respectively, and a, b, c and d are parameters to decide. We applied a PID control algorithm widely used as the robot control algorithm in the industry since it takes a little calculation time and is easy to code. The

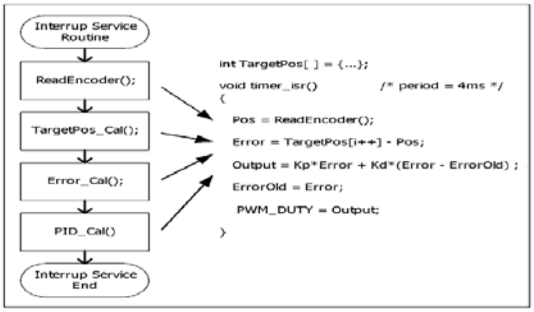


Fig. 10 Block diagram of interrupt routine

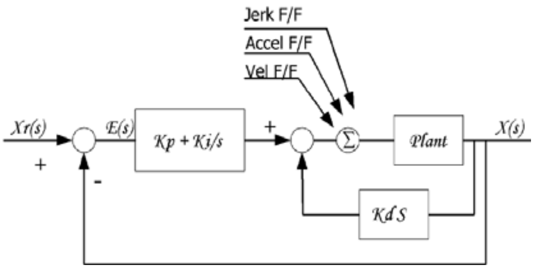


Fig. 11 PID controller with feedforward term

loop time of controller was set 2 ms by using the interrupt function and designed the PID control system capable of completing calculation of the control data within that time. An example for the control algorithm executed within interrupt time is shown as Fig. 10.

In this paper, we applied the PID controller with a feed-forward term to compensate for the gravity as shown in Fig. 11.

3.3 Communication system design using LINUX

We composed the main controller using the embedded ARM board to control the five controllers for the joint motors and the system. The embedded LINUX capable of real-time processing with small memory capacity is called the real-time embedded LINUX. In this paper, for the controller of the robot arm, we composed an operation system of the ARM board using the real-time LINUX referred to (Labrosse, 2002) and developed a communication system for it. For composing the controller using the real-time LINUX for various systems of the robot arm, compatibility between the microprocessor of the host PC

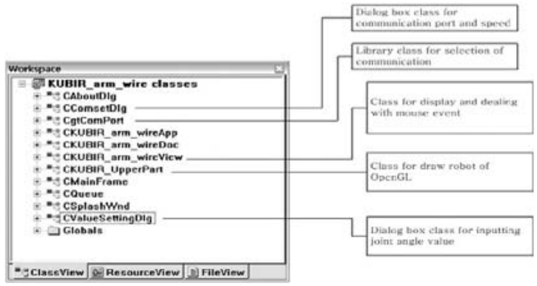


Fig. 12 Class structure of the simulator program

and that of the ARM board using different compilers should exist in the cross environments having different compilers. In this paper, we first built a compatible environment for developing cross systems such as the boot loader, kernel, and file system, and then boarded them on the ARM board. And then, we built a device driver required for controlling environmental devices around the main board such as the Ethernet communication system. Also, we developed a new library which makes it easy to control the joint motors and implements various motions for the control system of the robot arm. Finally, we composed a structure of the system to implement the wireless Ethernet communication system which uses PC as the server and the ARM Board as the client, and to drive the motion controller through RS 232 serial communication as shown in Fig. 5. Also, by using the thread function which makes it possible to treat serial inputs as parallel mode, we composed a parallel processing system to solve the problem of waiting time for receiving data in serial communication system.

3.4 A design of the motion generation program

We developed a 3D graphic simulator which makes it possible to monitor and simulate the motion of the robot arm in real time in advance. The simulator was composed of OpenGL that is a 3D graphic tool employing the pre-designed 3D model of the robot arm based on the AUTOCAD design tool. The simulator was developed using Microsoft Visual Studio 6.0 and MFC software, and its development structure of the system class is described in Fig. 12.

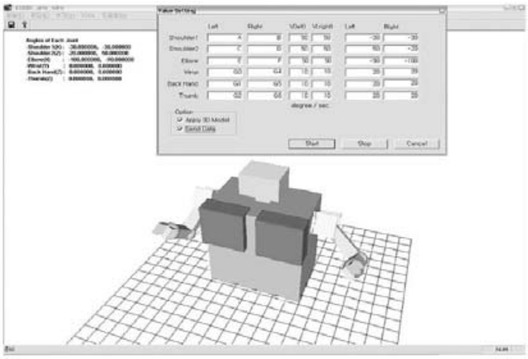


Fig. 13 Window program of motion generator

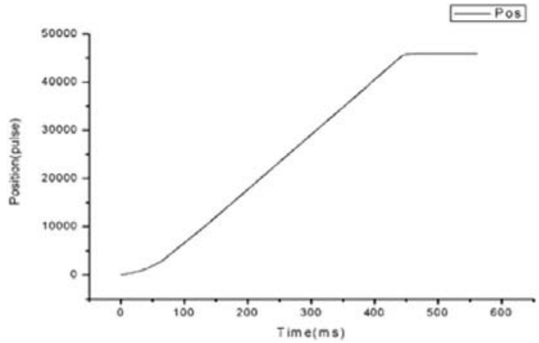


Fig. 14(a) Tracking results of the position control

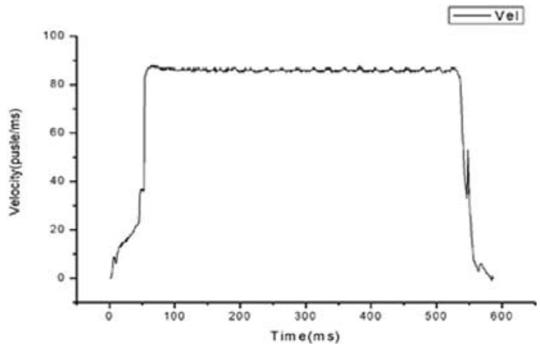


Fig. 14(b) Tracking results of the velocity control

The window screen of the whole structure including the main body of the robot is shown in Fig. 13. The dimensions of the modeling robot were reduced proportionally with that of the real robot. If we write the desired data of the joint angles and hit the transmission button at the input dialog box on the screen of the PC, then, the robot arm is controlled by receiving the order from the motion controller. Also, it is programmed to present the variation of motion of the graphic picture of the robot arm at every 100 ms according to the motion data from the arm joints.

4. Performance Test and Results

The performance test of developed system was made for the basic position control and the tracking ability along the trapezoidal velocity profile described in chap. 3

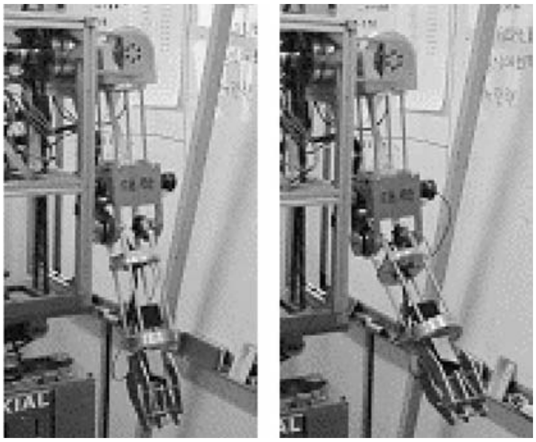
4.1 The trajectory control test of the joint motor

The test of the servo motor controller applied to the joint of the robot arm was performed. As results of the performance test, the tracking state of the joint motors along the trapezoidal velocity and its position trajectory were shown. The conditions of the performance test and parameters of the controller are specified as follows:

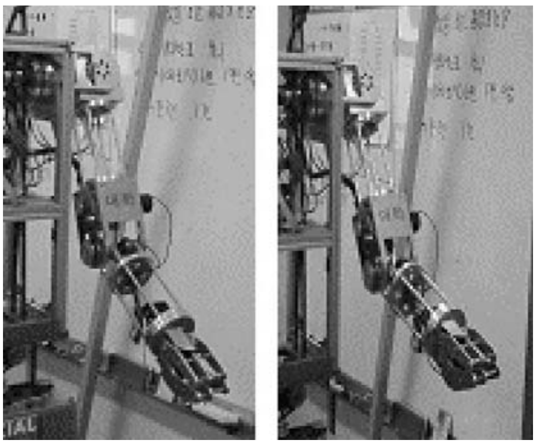
Control period: 1 kHz, Proportional gain Kp=1, Derivative gain Kd=6, and Integral gain Ki=0.05

The encoder pulse numbers per one revolution of the motor was set 6000. The goal destination of the motor was set 45000 pulses (7.5 revolution), the arriving time is 0.58 (second), and the average velocity was set 90 pulse/ms. The result of the control is shown in Fig. 14.

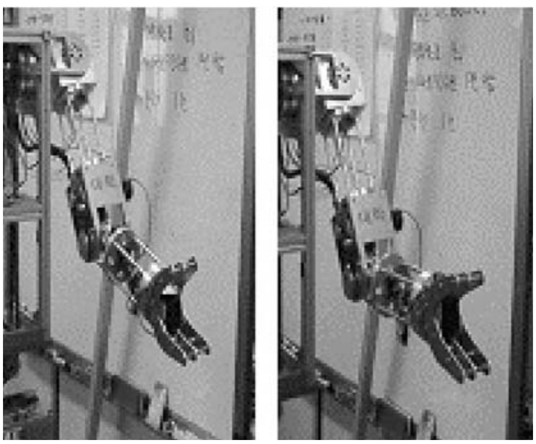
In the performance test, the modular DSP controller employing the PID control algorithm was applied to the DC servo motors, and Fig. 14 represents the results of the states of the motor tracking along the profiles of the trapezoidal velocity and its integrated position. According to the test results, the position tracking result shows almost no position errors, but the velocity tracking result shows some tracking errors and some chattering. Through the performance test, it is known that these small velocity errors and chattering do not have much effect on the position errors. Also, the velocity chattering and errors can be modulated by the gain tunning and altering the number of the passing points along the trajectory profile. In this paper, the operation purpose of the developed robot arm for the moving robot is to reach the arm to an object and to grasp or to touch it in



(a)



(b)



(c)

Fig. 15 Motion control of robot arm

the specified time. Therefore, we paid more attention on the position control than the velocity control.

Fig. 15 represents the sequential states of the robot arm motion according to the developed controller for the joint motors. The robot arm showed smooth motion without chattering. Through the performance tests of the motor and robot arm, we identified that the developed control system makes it possible for the robot arm to reach to the designated object along the trajectory in the specified time.

5. Conclusion

In this paper, we present a new revolute robot arm and its new distributed control system utilizing the one-chip microprocessor. The developed structure of the control system was composed of the main controller, a motion controller for five joint motors, and communication systems. The main controller and the motor controller were composed of the ARM microprocessor and the TMS320c2407 microprocessor appropriate for the embedded control system.

For the control system, a new table method for the motor controller to track the trajectory was devised by employing off-line trajectory data in advance. Also, related communication system was developed among the PC, the main controller, and the motion control system. In addition to this, we developed a 3D simulator based on the openGL software to simulate the motion of the robot arm.

To validate the performance of the robot system, tests to track a trajectory were performed. According to the test results, the position control result showed good performance despite of relatively lower performance in tracking velocity profile. Accordingly, the developed robot including the embedded control system was validated to be appropriate for the mobile robot arm.

Acknowledgments

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD)″(KRF-2005-D00065).

References

Han, S. H. and Hashimoto, H., 2005, "A Robust Real Time Adaptive Controller Design for Robot Manipulator with Eight-Joints Based on DSPs," *Journal of Mechanical Science and Technology*, Vol. 19, No. 10.

Harada, K., Kajita, S., Saito, H., Morisawa, M., Kanehiro, F., Fujiwara, K., Kaneko, K. and Hirukawa, H., 2005, "A Humanoid Robot Carrying a Heavy Object," *Proc. IEEE International Conf. on Robotics and Automation*, pp. 1724～1729.

Huang, Q., Sugano, S. and Tanie, K., 1998, "Motion Planning for a Mobile Manipulator Considering Stability and Task Constraints," *Proc of the IEEE Int. Conf. on Robotics and Automation*, pp. 2192～2198.

Kim, J. H., Park, S. W., Park, I. W. and Oh, J. H., 2003, "Development of a Humanoid Biped Walking Robot Platform KHR-1 — Initial Design and Its Performance Evaluation," *in Proceedings of 3rd IARP Int. Workshop on Humanoid and Human Friendly Robotics*, pp. 14～21.

Koyachi, N., Aral, T., Adachi, H., Asami, K. and Itoh, Y., 1995, "Hexapod with integrated Limb Mechanism of Leg and Arm," *Proc. of the Int. Conf. on Robotics and Automation*, pp. 1952～1957.

Labrosse, J. J., 2002, "MicroC/OS-II Real Time Kernel,," CMP Books, pp. 11～12.

Morita, T., Shibuya, K. and Sugano, S., 1998, "Design and Control of the Mobile Manipulation Symbiotic Humanoid : Hadaly-2," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 1315～1320.

Perrier, C., Dauchez, P. and Pierrot, F., 1998, "A Global Approach for Motion Generation of Non-Holonomic Mobile Manipulators," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 2971～2976.

Spong, M. W. and Vidyasagar, M., 1989, "Robot Dynamics and Control," John Wiley & Sons.

Thompson, P., Rombaut, M., Rzbatel, G., Pierrot, F., Liegeois, A. and Sevile, F., 1995, "Design and Contol of a Mobile Manipulator for Weed Control," *Proc. of the Int. Conf. on Advanced Robotics*, pp. 933～938.

ATmega128 User Manual, Atmel. Inc, 2004.

TMS320LF2407A User Guide, Texas Instrument. Inc.

http ://asimo.honda.com/

http ://www.intel.com/cd/corporate/

http://www.samsung.com/Products/Semiconductor/SystemLSI/DevelopmentTools/32bitscores01.htm

http ://www.microchip.com/stellent