International Journal of Control and Automation
Vol. 7, No. 11 (2014), pp. 273-284
http://dx.doi.org/10.14257/ijca.2014.7.11.27

Implementation of an Integrated Controller for a Robot Hand Base on a Vehicle Communication System

JeongHoon Park and Chan-Woo Moon

EECS, Kookmin University
mcwnt@kookmin.ac.kr

Abstract

Intelligent robots commonly adopt Network-Based Control Systems (NBCS). For control of a robot hand or a robot head, many sensor data and control commands need to be transmitted in real time, so they have necessitated communication systems with high transmission capability. Traditionally, CAN is widely used for intelligent robots, but it presents bandwidths that are too low for the control of a high performance system. In this paper, the relation between the transmission period and the performance of a multi-motor control system is analyzed, and a multi-motor control system for a human robot hand is implemented based on the FlexRay communication system, which is an emerging in-vehicle communication network system. FlexRay parameter optimization for a robot system is studied, and a real position control test is conducted, to validate the implemented system.

Keywords: *FlexRay, Intelligent robot, Multi-motor control, Robot Hand, Network Based Control Systems*

1. Introduction

For decades, intelligent robots that can provide service at home and in the office have been the focus of attention. An intelligent robot generally has limited internal space, because it is usually equipped with multiple actuators and sensors, so a network based control system (NBCS) is preferred over a one-to-one interfaced system, to reduce the complexity of electrical wiring [1]. In particular, for a humanoid robot, the development of a small sized actuator in which a controller, sensors and a network controller are integrated has become an industrial trend. Industrial network system, like EtherCat and RS485 can be a good solution for a large scale system such as robot arms and legs. But, in the case of more compact systems like robot hand and head, these network solutions are hardly applicable for an integrated control system because of their hardware complexity. DLR II hand adopted the IEEE 1355 standard with a communication controller FPGA [2]. And it became an exemplary robot hand with the integrated controllers which were installed on each finger. On the other hand, automotive network system which is one of solutions for robotic systems can be implemented with low cost and minimal hardware. And also it has high diagnostic coverage, so it provides very high safety integrity level, and is appropriate for a safety required system.

The CAN (Controller Area Network), which was developed as an automotive network system by Bosch in the 1980s, has been widely used as a control network system for intelligent robots [3, 4]. A CAN based system is easy and cheap to implement, so CAN has extended its applications in the intelligent robot industry. CAN is an event trigger protocol, and sends up to 8 bytes per message at 1Mbps. CAN has a fault detection and processing function inside, and so, is considered to be a robust and reliable protocol, suitable for control applications. But its transmission speed is unsatisfactory for transmission of a large amount of data, it requires multiple channels or a hierarchical structure of a different kind of network protocol

ISSN: 2005-4297 IJCA

to deliver the data on time [5]. Recently, the need for a more complicated network structure has been recognized, as well as the application of X-by-wire technology in the vehicle industry. For this reason, the FlexRay consortium announced the FlexRay protocol for in-vehicle networking. FlexRay is based on time trigger and event trigger protocols, and can realize transmission speeds of up to 10Mbps, with simple hardware [6]. In the vehicle industry, FlexRay is being adopted for a wide range of applications, and main topics of researches are message scheduling and time analysis in automotive system [7]. FlexRay system is applicable to other industrial systems which require high safety integrity level. Heller *et al.*, proposed a FlexRay system for avionic system [8]. In spite of the similarity to automotive system, its role in robotics applications has been limited. Xu *et al.* implemented the FlexRay Controller chip on FPGA for robotic applications [9].

In this paper, an experimental, multi-motor control system for a robot hand is implemented based on the FlexRay system; the relation between communication period and control bandwidth, which is directly related to the performance of the control system, is analyzed; and the optimal FlexRay parameter for the implemented system is investigated. Finally, a comparison between the traditional CAN protocol and the FexRay system is made with a real position control experiment.

2. Analysis of a Multi-motor NBCS for a Robot Hand

2.1. Structure of a Robot Hand Control System

The driving mechanisms of a robot hand can be classified as follows: pneumatic, electric motor driven, wire- driven, and so forth. The developed finger of a robot hand, which is directly driven by electric motor, is shown in Figure 1. The whole robot hand has four fingers, equipped with 13 motors and 13 joint torque sensors. Each finger, excepting the thumb, is driven by 3 BLDC motors, 2 motors for a MCP joint which is a differential type and 1 motor for a PIP joint respectively. Each joint controller is attached on the finger link. The structure of the controller is shown in Figure 2. Each slave motor controller sends position data and joint torque/force sensor data to the master controller, and the master controller sends commands to the motor controller through a communication bus. A multi-motor NBCS for intelligent robots can be implemented mainly in three ways: open-loop position command based, torque command based and speed command based, as shown in Figure 3. Traditionally, the speed command-based system is widely used, although it is easier to implement a high performance control algorithm in the master controller with the torque command-based system. In this paper, the open-loop system is not discussed. An NBCS for a robot hand is designed with four steps:

① Determination of the performance index (bandwidth) of a robot hand controller
② Determination of the transmission period
③ Network parameter optimization with the given transmission period
④ Determination of the number of nodes that one communication channel can drive

Figure 1. Structure of the Developed Finger for a Robot Hand

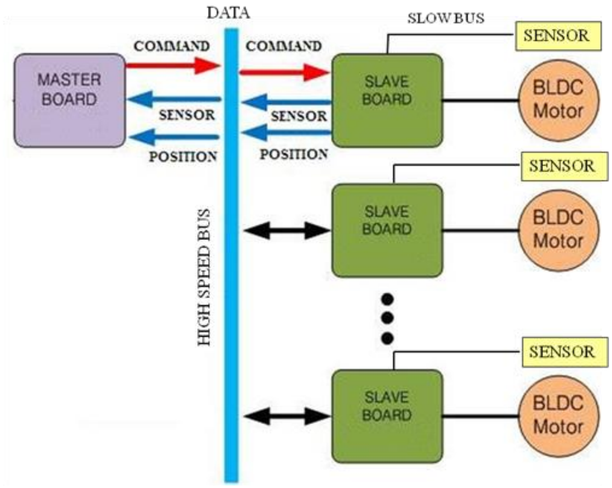
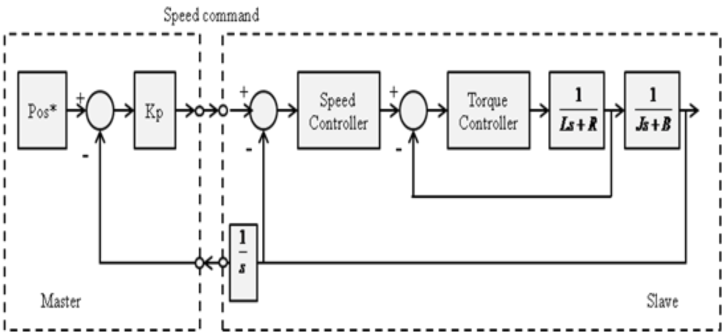
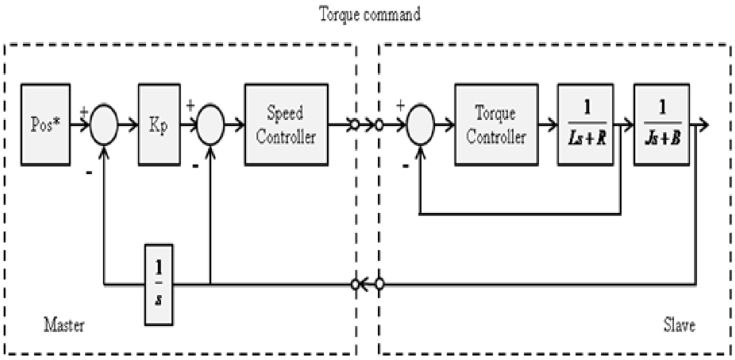


Figure 2. Structure of a Network based Controller



(a) Speed Command based System



(b) Torque Command based System

Figure 3. Control Schemes of NBCS of Multi Motor

2.2. Determination of the Transmission Period of an NBCS according to the Control Method

The transmission period is closely related to the stability and performance of the NBCS. Walsh *et al.* proposed an absolute stability analysis method for the NBCS, based on Lyapunov's second method, but the proposed method is too conservative [10]. On the other hand, Zhang et al. built an augmented discrete system, and obtained a stability condition with respect to the time delay by inspecting the stability of the discrete system [11]. Here, based on Zhang's method, a relation of the control bandwidth and time delay is investigated. The bandwidth of a controller is an important performance index. Let h be the sampling time, which is also the communication period generally, and τ_{delay} be the total communication delay, which includes the network transmission delay and the software execution delay. In most digital controllers, τ_{delay} is longer than h, and it depends on the control software structure. Then, the augmented system for the NBCS would be expressed as in Eq. (1). Γ_0, Γ_1 is function of the system matrix and input matrix, and u is system input [11].

$$z((k+1)h) = \tilde{\Phi}(k) z(kh)$$
$$\tilde{\Phi}(k) = \begin{pmatrix} \Phi & \Gamma_1 & \Gamma_0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -K & 0 & 0 & 0 & 0 \end{pmatrix} \quad (1)$$

Where, $z(kh) = [x^T(kh), u^T((k-l)h), \ldots, u^T((k-1)h)]^T$

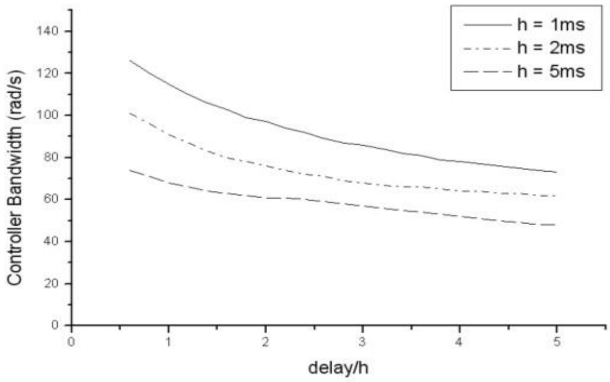
In the case of the torque command based system, the system matrix and input matrix are obtained as in Eq. (2).

$$\dot{x} = \begin{pmatrix} \omega_c & 0 & 0 \\ 0 & 0 & 1 \\ \frac{1}{J} & 0 & -\frac{b}{J} \end{pmatrix} x + \begin{pmatrix} \omega_c \\ 0 \\ 0 \end{pmatrix} u$$
$$y = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} x$$
$$K = \begin{pmatrix} 0 & K_{SI} & K_{SP} \end{pmatrix} \quad (2)$$

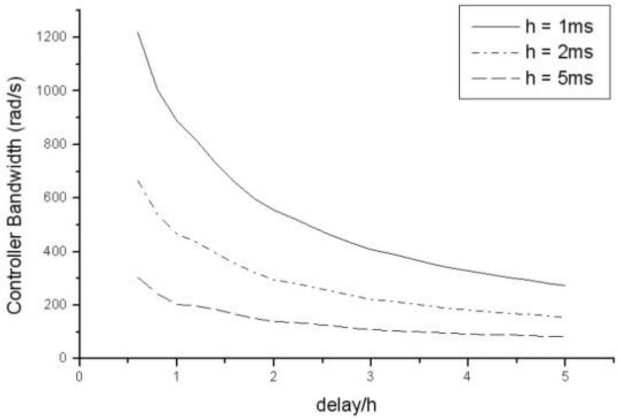
In the case of the speed command based system, the plant is obtained by Eq. (3).

$$\dot{x} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -\frac{K_{SI}}{J} & -\frac{K_{SP}}{J} \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u$$
$$y = \begin{pmatrix} \frac{K_{SI}}{J} & \frac{K_{SP}}{J} & 0 \end{pmatrix} x$$
$$K = K_p \begin{pmatrix} \frac{K_{SI}}{J} & \frac{K_{SP}}{J} & 0 \end{pmatrix} \quad (3)$$

Where, J is the system inertia, K_{SI} and K_{SP} are the integral and proportional gains of the speed controller, respectively, Kp is the gain of the position controller, ω_c is the current loop controller cutoff, and b is the damping coefficient. If τ_{delay} is constant, the whole augmented closed loop system for stability analysis can be obtained as reference [11]. A graph of the bandwidth of the master controller versus the ratio of allowable τ_{delay} to sampling time h for each control scheme is can be obtained in Figure 4, which was obtained by the augmented system. The target motor was the Maxon BLDC 15W motor.



(a) Speed command based system



(b) Torque command based system

Figure 4. Allowable Time Delay τ_{max} of Each Control Scheme Versus Controller Performance

Recently, a commercial non-NBCS showed a high control bandwidth. In particular, in the case of the torque command based scheme, a commercial non-NBCS has at least 1000 rad/s bandwidth, while in the case of the speed command based scheme, it has a bandwidth of over 50 rad/s. So, an NBCS controller needs to have a short communication period, if it is to meet the performance of a non-NBCS controller. But if the communication period decreases, the number of actuators that can be driven by one communication channel would decrease. In both control schemes, there is a trade-off between the control performance and number of actuators, necessitating a high transmission rate. If the target control bandwidth and system structure is given, the transmission period can be determined.

2.3. Network Parameter Optimization and Comparison of CAN and FlexRay NBCS

2.3.1. FlexRay Network Parameter Optimization for a Robotic System: One cycle of FlexRay consists of static segment, dynamic segment, symbol window, and network idle time [6]. The static segment and network idle time are indispensible and the dynamic segment and symbol window are optional. The static segment uses the TDMA(Time Division Multiple Access) scheme, and includes a number of static slots, which have their unique frame IDs. So, the static segment is suitable for periodic control and sensor messages. The dynamic segment uses the FTDMA(Flexible Time Division Multiple Access) scheme. FlexRay system has huge number of parameters to setup, in comparison with CAN system. The parameter optimization problem of the in-vehicle network efficiency of the FlexRay system was described in reference [12]. Here, parameter optimization problem for a robot system with multi motors is formularized. The FlexRay parameters are optimized in view of two objects. The first object is to maximize the number of actuators controlled with one communication channel for a given transmission period. The second object is to minimize the communication period in order to improve the control bandwidth. First, the frame data size of each slot must be the same and an even number of bytes, so the frame size needs to be optimized to drive the maximum number of actuators. Let W denote the number of data words per frame and B_M be the number of data bytes that each actuator sends to the controller, then each actuator sends F_M frames.

$$F_M(W) = \lceil B_M / 2W \rceil \tag{4}$$

Let B_C and N be the number of data bytes that the controller sends to control one actuator and the number of actuators connected to the communication channel, respectively. Then the number of frames F_C occupied by the main controller is obtained as:

$$F_C(N, W) = \lceil B_C \times N / 2W \rceil \tag{5}$$

The number of macroticks per static frame, M_F, is obtained by Eq. (6).

$$M_F = 2 * A + \left\lceil \frac{(Framelength + 11) * B + C + D}{E * (1 - F)} \right\rceil \tag{6}$$

$$Framelength = ((2W + 8) \times 10) + TSS + FSS + FES$$

Where, gdActionPointOffset(A), gdBitMax(B), gdMinPropagationDelay(C), gdMaxPropagationDelay (D), gdMacrotick(E) and cClockDeviationMax(F) are parameters and TSS, FSS and FES are constants, which are defined in the FlexRay specification. The number of static frames F, that can be transmitted in one communication cycle is given by:

$$F(W) = \lfloor Number\ of\ macroticks\ for\ static\ frame / M_F \rfloor \quad (7)$$

Where the number of macroticks assigned for a static frame is the total number of macroticks for one cycle minus the numbers of macroticks of a dynamic segment, symbol window and NIT portion. Then the maximum number of actuators N_{max} that can be controlled with one FlexRay channel is given by the following recursive equation.

$$N_{max} = \max_W \left\lfloor \frac{F(W) - F_C(N_{max}, W)}{F_M(W)} \right\rfloor \quad (8)$$

Secondly, to maximize control bandwidth, cycle period must be minimized. The minimum cycle period h_{min} is obtained as Eq. (9).

$$h_{min} = \min_W \left(\left(\left(F_C(W) + F_M(W) \cdot N \right) \cdot M_F(W) + gdNIT_{min} \right) \right) \cdot gdMacrotick \quad (9)$$

And also, from the specification, h_{min} must satisfy the following constraints.

$$h_{min} \geq 4 \cdot \left(F_C(W) + F_M(W) \cdot N \right) \cdot gdMacrotick$$
$$\geq 10 \cdot gdMacrotick$$

2.3.2. Comparison between the CAN and FlexRay Network for a Robotic System: For the developed robot hand control system, the controller sends a 2 bytes control command for each actuator, and each actuator sends 8 bytes to the controller, including sensor data. Each data is assumed to be transmitted with the same period, 1ms.

● Traditional CAN network based robot hand controller

CAN is an event triggered communication protocol, so collision between messages can occur. Unfortunately, network transmission delay is stochastically varying within the range of the worst case response times (WCRT) [13]. We calculate the number of actuators that one CAN channel can control with two rules, the 60% rule and the worst case response time rule. In the 60% rule case, the number of nodes that can maintain the communication efficiency within 60% is determined. In the latter case, the WCRT of the lowest priority node is set such that the message can be transmitted within one communication period in the worst case. The WCRT can be obtained by the single process time analysis method [13]. The numerical results with 1Mbps Baudrate are shown in Table 1.

Table 1. Number of the Maximum Actuator with One CAN Channel

Rule	messages	Maximum number of actuator
60% rule	5 messages Controller, motor 1~4	4
WCRT	7 messages Controller 1~2, motor 1~5	5

● FlexRay Network based robot hand controller

The numerical result of the parameter optimization with 10Mbps Baudrate, 980 macroticks for static frame, no dynamic frame, no symbol window, 1 μs macrotic and 1 ms sampling time is shown in Table 2. In this case W = 4 or 5 is the optimal data size. The minimum transmission period with no dynamic frame, no simbol window, 2 NIT, and gdActionPointOffset 4 is shown in Table 3.

Table 2. Maximum Number of Actuators with Respect to Data Size with FlexRay System

Length of data	Total static frame number	Maximum number of actuators
6 bytes	35	15
8 bytes	32	25
10 bytes	30	25
12 bytes	28	24

Table 3. Minimum Period with Respect to Number of Actuators

Number of actuators	Minimum transmission period (μs)
6	242
13	512
24	902

3. Implementation and Experiments

3.1. Traditional CAN System

A CAN based system for a test is implemented. Nine motors are connected on 1 CAN channel. The sampling time is 1 ms. The CAN sends data at 1Mbps. Three high-priority messages are assigned to the master controller, and each message includes 6-byte data, and controls three motors. Each motor controller uses one message and delivers 8 bytes of position and sensor data to the master controller. A position control experiment with the speed control based scheme was conducted. The target position was 100 radians or 200 radians. The results are shown in Figure 5. Motors with higher priority are stable but motor 7 (priority order 10) is uncontrollable because sensor data could not be transmitted periodically.

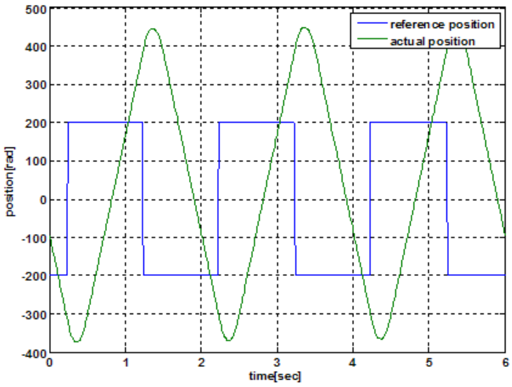


Figure 5. Position Control Experiment Result with CAN, Motor 7 (Uncontrolable)

3.2. The Developed Robot Hand Controller based on FlexRay System

Figure 6 shows the FlexRay based controller board which is installed on the robot finger. The main controller is assigned 1 to 4 slots and motor 1 to motor 13 are assigned from 5 to 17 slots. In this paper, among the 13 motor drivers, only two controllers controlled real motors which drive a robot finger, and the rest were simulators, only transmitting simulated data via a FlexRay bus. Figure 7 shows the position control experiment with 30rad/sec bandwidth, and the order of the frame ID did not affect the result because static segments were transmitted based on the time division scheme.

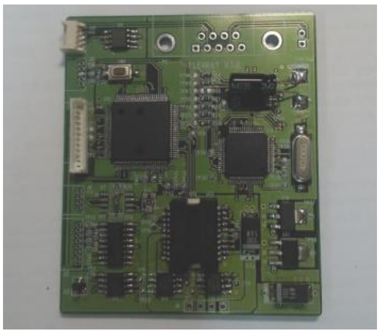


Figure 6. FlexRay based Motor Control Board

Table 4. Frame ID Allocation

Frame ID	Allocation
1~4	controller
5,6	Motor 1,2
7~17	Simulator

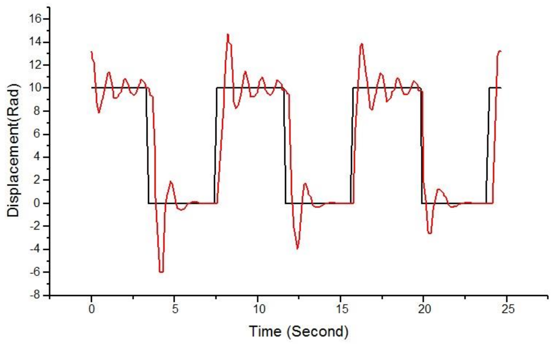


Figure 7. A Real Finger Position Control based on FexRay System

4. Conclusion

A multi-motor control system for a robot hand was implemented based on the FlexRay communication protocol. Recently, FlexRay has attracted much attention for its high bandwidth, high safety integrity level, and simple hardware, but there have been only a few examples of FlexRay design in the robot industry. To determine an adequate transmission period, the stability of two control schemes, the torque control based system scheme and the speed control based system scheme, was analyzed, and the allowable time delay with respect to the controller bandwidth was calculated. Generally, a very high speed data transfer rate of the communication channel is necessary for the torque control based scheme to achieve the performance level of a commercial non network based multi motor control system. But CAN is rather slow for this purpose. The Flexray parameters were optimized in view of two objects. The first object was to maximize the number of actuators controlled with one communication channel for a given transmission period. The second object was to minimize the communication period in order to improve the control bandwidth. For a comparison test, CAN-based and FlexRay- based systems were implemented to test their position control capabilities. For the experimental CAN and FlexRay system, the Infineon 16bit micro controller with the embedded communication controller was used. Because FlexRay is 10 times faster than CAN, it might seem trivial that FlexRay presents a high control bandwidth, but the differences between the communication protocols: the event driven and time driven, were worth analyzing. Recently, FlexRay control chips are increasingly integrated in microprocessor chips, so it can be expected that FlexRay may replace CAN in the field of intelligent robotics which require minimal hardware.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A2040141).

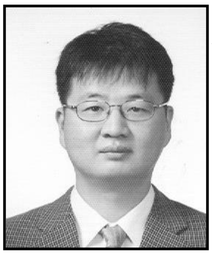
References

[1] F. L. Lian, J. Moyne and D. Tilbury, "Performance Evaluation of Control Networks Ethernet", Control Net and DeviceNet, IEEE Control Systems, vol. 21, no. 1, **(2001),** pp.84-99.
[2] J. Butterfaß, M. Grebenstein, H. Liu and G. Hirzinger, "DLR-Hand II: Next Generation of a Dextrous Robot Hand", IEEE International Conference on Robotics & Automation, **(2001),** pp.109 – 114.
[3] R. Bosch and H. Gmb, "CAN specification version", vol. 2,**(1991)**.
[4] K. Y. Yi, "An Implementation of the Position Controller for Multiple Motors Using CAN", Transaction of KIEE, vol. 51, no. 2, **(2002)**, pp.55 – 60.

[5] Y. S. Cha and B. J.You, "Dual Control Network Method for Efficient Real-time Humanoid Robot Control System", IEEE International Symposium on Industrial Electronics, **(2009),** pp. 1797 – 1802.
[6] FlexRay consortium, FlexRay protocol specification V2.1
[7] H. Jaouani, R. Bouhouch, W. Najjar and S.Hasnaoui, "Hybrid task and message scheduling in hard real time distributed systems over FlexRay bus", Communications and Information Technology, **(2012),** pp. 21 – 26.
[8] C. Heller andR. Reichel, "Enabling FlexRay for avionic data buses", Digital Avionics Systems Conference,**(2009)**.
[9] Y. N. Xu, Y. E. Kim, K. J. Cho, J. G. Chung and M. S. Lim, "Implementation of FlexRay Communication Controller Protocol with Application to a Robot System", ICECS, **(2008)**, pp. 994 – 997.
[10] G. C. Walsh, H. Ye and L. G. Bushnell, "Stability Analysis of Networked Control Systems", IEEE Trans. on Control Systems Technology, vol. 10, no. 3,**(2002)**,pp. 438 – 446.
[11] W. Zhang, M. S. Branicky and S. M. Phillips, "Stability of Networked control Systems", IEEE Control Systems, vol. 21, **(2001)**,pp. 84 – 99.
[12] I. S. Park and M. Sunwoo, "FlexRay Network Parameter Optimization Method for Automotive Applications", IEEE Transactions on Industrial Electronics, vol. 58,**(2011)**, pp. 1449 – 1459.
[13] K. W. Tindell, A. Burns and A. J. Wellings, "Calculating controller area network (CAN) message response times", Control Engineering Practice, vol. 3, no. 8, **(1995)**, pp. 1163 – 1169.

Authors

Jeonghoon Park, Graduate student, EECS, Kookmin Univ.



Chanwoo Moon, Ph.D., Associate professor, Kookmin Univ.