COMMUNICATION PROTOCOL FOR ROBOTIC ARM IN NUCLEAR POWER PLANTS

Ilona Usuman[1] and Hary Gunarto[2]

1. ***Faculty of Math and Natural Sciences, Gadjah Mada University***
2. **Ritsumeikan Asia Pacific University, JAPAN**

ABSTRACT

Robotic Arm Environment In Nuclear Power Plant consists of robots, controlled instrumentations, computers, and sensors. Its use in the production segment of segment of *nuclear power industries promises a variety of benefits ranging from high utilization to high volume of productivity[7.8]. Functions and organizations of industrial and Nuclear Robotic Arm have experienced important improvements. While nodes were initially introduced by grouping machines and then simply interconnecting the inputs and outputs of their controllers, it is now assumed that each device can be attached to a network and be able to exchange information reciprocally. The need for the design of a node communication network becomes urgent. The coordination among devices and instrumentation are mostly under the control of one or more computers. In this paper, we will study the communication protocol needed for Robotic Arm Environments. The proposed Robotic Arm communication protocol should satisfy the requirements and characteristics of data traffic in this environment. Large program files from the main computer take several seconds to be down loaded into each device and instrument at the beginning of Robotic Arm operation. Messages for data checking, status monitoring and reporting usually need to be transmitted in a periodic time with deterministic time delay. Other type of message used for emergency reporting is quite short in size and must be transmitted and received with almost instantaneous response.*

A reliable Robotic Arm protocol that support a real time communication with bounded delay time is needed for Robotic Arm In Nuclear Power Plants. We proposed a modification of standard IEEE 802.4 Token Bus protocol to implement a prioritized access scheme. The performance of the proposed protocol is presented and compared with the standard Token Bus. Both analytical and simulation techniques are used to verify the performance of the proposed protocol.

Keywords : *Robot Arm, Nuclear, Instrumentation, Sensor*

* Corresponding author.
E-mail address: ilona@ugm.ac.id

Published @ 2nd Jogja International Physics Conference Proceeding, Gadjah Mada University, 2007年9月

1. INTRODUCTION

A Robotic Arm Network consists of robots, Controlled instrumentations, computers, sensors, and other stand alone systems such as inspection machines. The use of robotic arm in the production segment of power plant industries promises a variety of benefits ranging from high utilization to high volume of productivity. Each Robotic Arm cell or node, as proposed by Berman and Maimon[1], will be located along a material handling system such as a conveyor or automatic guided vehicle. The production of each part or work-piece will require a different combination of manufacturing nodes. The movement of parts from one node to another is done through the material handling system. At the end of part processing, the finished parts will be routed to an automatic inspection node, and subsequently unloaded from the Robotic Arm.

Functions and organizations of industrial Robotic Arm have experienced important improvements recently. The various nodes are incorporated into a single system, and each device or machines is attached to a network and are able to exchange information reciprocally. The coordination among nodes and the control of the part production throughout the nodes will be accomplished under the supervision of one or more computers.

In this paper, we will study the communication protocol needed for Robotic Arm. In section 2, we discuss the expected traffic characteristics in this environment. The proposed Robotic Arm protocol is discribed in section 3. In sections 4 and 5, the analysis and simulation results of the proposed protocol are discussed. The last section summarizes and concludes the results of the investigation.

2. TRAFFIC CHARACTERISTICS OF ROBOTIC ARM NETWORK

A recent paper presented by Schutz [2] proposed the communication and data characteristics of Robotic Arm Network. The Robotic Arm data traffic consists of large files and short messages, and mostly come from devices and instruments. The message size ranges between a few bytes to several hundreds of bytes. Executive software and other data, for example, are files with a large size, while messages for machining data, instrument to instrument communications, status monitoring, and data reporting are transmitted in small size.

There is also some variation on response time. Large program files from a main computer usually take about 60 seconds to be down loaded into each instrument or node at the beginning of Robotic Arm operation. Messages for instrument data need to be sent in a periodic time with deterministic time delay. Other type of messages used for emergency reporting is quite short in size and must be transmitted and received with almost instantaneous response.

The demands for reliable Robotic Arm protocol that support all the Robotic Arm data characteristics are now urgent. The existing IEEE standard protocols do not fully satisfy the real time communication requirements in this environment [3]. The delay of CSMA/CD is unbounded as the number of nodes increases due to the message collisions. Token Bus has a deterministic message delay, but it does not support prioritized access scheme which is needed in Robotic Arm communications. Token Ring provides prioritized access and has a low message delay, however, its data transmission is unreliable. A single node failure which may occur quite often in Robotic Arm causes transmission errors of passing message in that node. In addition, the topology of Token Ring results in high wiring cost.

A design of Robotic Arm protocol that supports a real time communication with bounded message delay and reacts promptly to any emergency signal is needed. Because of machine failure and malfunction due to heat, dust, and electromagnetic interference is common, a prioritized mechanism and immediate transmission of emergency messages are needed so that a suitable recovery procedure can be applied. We propose a modification of standard Token Bus to implement a prioritized access scheme. This scheme allows transmission of short and periodic messages with a low delay compared to the one for long messages.

3. PROPOSED ROBOTIC ARM PROTOCOL

The topology of our proposed Robotic Arm network is shown in Fig 1. It comprises of stations which are arranged according to their physical locations. Predecessor node always has higher address, and successor node has lower address, except for the lowest and highest node. For the lowest node, its successor is the highest node. The highest node has the lowest node as its predecessor. This physical arrangement has the advantage of minimizing unnecessary token delay between nodes.

The natural node ordering also supports a reliable communication and helps in detecting node failure promptly. The delay between nodes can be measured and memorized so that the duration in which successor node has to respond to previous token is exactly known. If the successor node does not respond within this deterministic time period, then the node will determine that the successor node fails. This mechanism for detecting a node failure is another essential feature of Robotic Arm protocol because the probability of node failure is very high in this environment.

The bus access control in our Robotic Arm network is a variation of Token Bus. Fig 2 shows the access scheme of the proposed Robotic Arm protocol. We developed a priority mechanism with two different message classes: high(H) and low(L) priority messages. The transmission of high priority messages is done more frequently than the low priority messages. In each token rotation, all nodes sequentially send their high priority messages, but only one node is allowed to send its low priority messages. Then, on the next token rotation, the successor node of the previous one can access the bus for its low priority messages. Consequently, high priority messages are transmitted in each token rotation, whereas low priority messages are sent after waiting for N token rotations. N is the total number of nodes in the system. Note that two different token rotation times are identified. The first one, T_{c1}, corresponds to high priority message transmission. The other token rotation time, T_{c2} which equals to NT_{c1} refers to low priority message access.

4. ANALYSIS OF THE PROTOCOL

We consider a Robotic Arm protocol with two different priority classes. As before, let us define the first class of traffic as the high priority and the second class of traffic as the low priority. The channel activity for these two class of messages is shown in Fig 2. Assume that each node has an average token transmission time of $\overline{V}$, and average service time of $\overline{X_1}$ and $\overline{X_2}$ for the first and the second class messages respectively. For N nodes system, the mean token rotation time, T_{c1}, can be expressed as summation of token transmission, and the transmission of the first and second class messages.

$$T_{c1} = N(\overline{V} + N_1\overline{X_1}) + N_2\overline{X_2}$$

Note that the mean number of first class messages in each node, N_1 which arrive within token rotation T_{c1}, equals to $\lambda_1 T_{c1}$. The total number of second class messages, N_2, which arrive within token rotation T_{c2}, equals to $\lambda_2(NT_{c1})$. Then, we have the following:

$$T_{c1} = N\overline{V} + N\lambda_1 T_{c1}\overline{X_1} + \lambda_2 NT_{c1}\overline{X_2}$$

Substitution of $N\lambda_1\overline{X_1} = \rho 1 =$ rho_1 @, and $N\lambda_2\overline{X_2} = \rho 2$ will bring

$$T_{c1} = N\overline{V} + (\rho 1 + \rho 2)T_{c1}$$

Note that the sum of traffic load for both priority classes equals to total traffic ρ . Then,

$$T_{c1} = N\overline{V} + \rho T_{c1}$$

Solving for T_{c1} , we have the following expression:

$$T_{c1} = \frac{N\overline{V}}{(1-\rho)} \tag{1}$$

The calculation for waiting time of both classes of messages involves two queueing delays. The first queueing delay is due to a message already in service (residual time) W_p which can be represented as priority queue system. The other delay W_T is due to messages in other nodes that have to be served before this node. Note that W_T equals to half of token rotation for a specific priority class under consideration [4]. Thus,

$$W = W_p + W_T$$

The analysis of priority queueing system is proposed by Cobham [5]. For M/G/1 system, the average waiting time for the p priority class is given by:

$$W_p = \frac{\Sigma_{i=1}^{p} \lambda_i \overline{X_i^2 / 2}}{\left(1 - \Sigma_{i=1}^{p-1} \rho i\right)\left(1 - \Sigma_{i=1}^{p} \rho i\right)} \tag{2}$$

where ρi, λ_i, and $\overline{X_i^2}$ are the traffic load, mean arrival rate, and the second moment of service time for message in each class respectively. We use the notation W_F for the overall mean waiting time of the first (high) priority class, and W_s for the overall mean waiting time of the second (low) priority class. For the first priority class, we have:

$$W_F = W_1 + W_T$$

where W_1 is obtained from equation (2) by setting P = 1. By substitution of $W_T = T_{c1} / 2$ and T_{c1} from equation (1), we can write the overall mean waiting time as follows:

$$W_F = \frac{\lambda_1 \overline{X_1^2}}{2(1-\rho)} + \frac{N\overline{V}}{2(1-\rho)} \tag{3}$$

Similarly, the overall mean waiting time of the second priority class
W_s can be expressed as

$$W_s = W_2 + W_T$$

where W_2 equals to W_p in equation (2) with P = 2 . As shown in Fig 2, the token rotation time for second priority class T_{c2} equals to NT_{c1}. Therefore, the overall mean waiting time for the second priority class is given by the following:

$$W_s = \frac{\lambda_1 \overline{X_1^2} + \lambda_2 \overline{X_2^2}}{(1-\rho 1)(1-\rho)} + \frac{N^2 \overline{V}}{2(1-\rho)} \tag{4}$$

Let us turn now to the special case (P=1) in which there is a single priority level (no priorities as indicated in standard Token Bus). If we set P=1 in (2), and if T_{c1} is the only system token rotation time, we obtain the formula

$$W = \frac{\lambda \overline{X^2}}{2(1-\rho)} + \frac{N\overline{V}}{2(1-\rho)} \tag{5}$$

This is the formula for expected waiting time of messages in standard Token Bus [6].

5. SIMULATION RESULTS

The simulation results of our proposed Robotic Arm protocol are presented in Fig 3. To demonstrate its performance, the results are compared with the standard Token Bus. The same parameters: message length of 100 bytes and traffic load of 70 % are used for both protocol simulations. Other network parameters such as cable length and token transmission time are kept the same, except the size of delay between nodes. In Robotic Arm protocol, the delay equals to one round trip propagation time divided by the number of nodes in the system. The performance of the protocol in terms of number of nodes has been studied.

Fig 3 shows the average delay as a function of number of nodes. High priority messages have a lower average delay than the messages for Token Bus. It is quite reasonable because high priority messages need not wait as long as they would be in Token Bus. Conversely, the low priority messages of the Robotic Arm protocol show a higher delay than the ones for Token Bus. Note that low priority messages are transmitted after waiting for N token rotations.

In Figs 4 and 5, the simulation results of Robotic Arm protocol are compared to the analytical results discussed in the previous section. Fig 4 shows the message delay for the first priority message, and Fig 5 is for the second priority message. The average message delays do not vary significantly between the analytical and the simulation results. The results yield a good accuracy over the entire range of number of nodes, and consistently confirmed the accuracy of the simulation results.

6. CONCLUSION

A variation of Token Bus protocol designed for Robotic Arm Communication in nuclear power plants is presented. The proposed Robotic Arm protocol results in a small average delay for high priority messages. The low priority messages, as a result, are forced to wait some additional time compared to the one in cyclic service system (Token Bus). The protocol requires an arrangement of the nodes. A simple algorithm, however, can be implemented to do the nodes arrangement automatically.

The Robotic Arm protocol supports a real time communication with bounded delay time and provides a prioritized access mechanism. These features are important because machine failure or malfunction due to heat, dust, and electromagnetic interference need to be reported immediately so that a suitable recovery procedure can be applied. Both analytical and simulation results verify the advantages of our proposed protocol.

REFERENCES

[1] O. Berman, O. Maimon : "Cooperation Among Flexible Manufacturing Systems ", IEEE Journal of Robotics and Automations vol. RA-2 March 1986, pp 24.

[2] Henrik Schutz : "Types of Networks", I &CS the Industrial and Process Control Magazine, March 1985.

[3] Hary Gunarto, Y. P. Chiang: " An Industrial FMS Communication Protocol" IEEE International Phoenix Conference, Feb. 1987, pp 53.

[4] S. W. Fuhrmann and R. B. Cooper: "Application of Decomposition Principle in M/G/1 Vacation Model ...", AT &T Bell Lab Tech. Journal, vol 64 June 1985, pp 1091.

[5] Alan Cobham: "Priority Assignment in Waiting Line Problems", Operations Research, no 2, 1954, pp 70.

[6] M. Colvin and A. Weaver: "Performance of Single Access Classes on the IEEE 802.4 Token Bus", IEEE Trans. on Comm., vol COM-34, no 12, Dec. 1986, pp 1253.

[7] Jae Kwon Kim and Byung Kook Kim: “Implementation of Reliable Robot Control System for Nuclear Power Plant Manipulation”, Proceedings of the 32nd International Symposium on Robotics, Korea, 19-21 April 2001.

[8] János Sebestyén Jánosy: Instrumentation and Control System Renewal using Simulation at PAKS Nuclear Power Plant, The European Multidisciplinary Society for Modelling and Simulation Technology Conference, ISC 2007, June 11-13 2007, Delft University of Technology, Delft.