The Profibus Timed Token MAC Protocol for Real-Time Communications

Hong-Hee Lee*, Gwan-Su Kim** and Eui-Heon Jung**
* Department of Electrical Engineering, Ulsan University, Ulsanl, Korea
(Tel : +82-52-259-2187; E-mail: hhlee@mail.ulsan.ac.kr)
**Department of Electrical Engineering, Ulsan University, Ulsan, Korea
(Tel : +82-52-259-2867; E-mail: gskim94@mail.ulsan.ac.kr)

Abstract: This paper describes how to use Profibus networks to support real-time industrial communications, that is, how to ensure the transmission of real-time messages within a maximum bound time. Profibus is based on a simplified timed token protocol, which is a well-proved solution for the real-time communication systems. However, Profibus differs from the timed token protocol, thus the usual timed token protocol has to be modified in order to be applied in Profibus. In fact, the real-time solutions for networks based on the timed token protocol rely on the possibility of allocating specific bandwidth for the real-time traffic. This means that a minimum amount of time to transmit the real-time messages is always guaranteed whenever each token is arrived. In other words, with the Profibus protocol, at least, one real-time message should be transmitted per every token visit in the worst case. It is required to control medium access properly to satisfy the message deadlines. In this paper, we have presented how to obtain the optimal network parameter for the Profibus protocol. The selected network parameter is valid regardless of the behavior of asynchronous messages.

Keywords: fieldbus, timed-token protocol, profibus, real-time system

1. INTRODUCTION

Most real-time applications such as industrial process control depend on digital computers. Because Profibus is based on a simplified timed-token protocol which is a well proved solution for the real time communication system, Profibus differs from the traditional timed-token protocol. The timed-token protocol is a token-passing protocol which guarantees each node to share of the network bandwidth[1]. In the Profibus, an access to the communication medium is controlled by a token that is passed subsequently from a node to another node. Messages are usually separated into two classes: synchronous and asynchronous. Because the synchronous message used for the real time communication has a deadline constraints, it should be guaranteed to share of the network bandwidth. Thus, it is important to make sure the deadlines of synchronous messages.

The idea behind the timed-token protocol is how to control the token rotation time. A protocol parameter called the target token rotation time(TTRT) provides the expected token rotation time. An amount of TTRT is assigned to each node, and the synchronous bandwidth(H_i) is the maximum permitted time for the node to transmit synchronous messages whenever it receives the token. The allocated synchronous bandwidth is always longer than the transmission time. When a node receives the token, it is general to transmit its synchronous messages. In other hand, if the elapsed time from the previous token departure at the same node is less than the value of TTRT, that is, in case that the token is arrived earlier than the expected TTRT, it can transmit its asynchronous messages. To guarantee that synchronous message deadlines are satisfied, the system designer must carefully choose network parameters such as the synchronous bandwidth, the TTRT, and the buffer size. The synchronous bandwidth is the most critical one among the above mentioned three network parameters. If the synchronous bandwidth allocated to a node is too small, the node might not have enough network access time to transmit messages before their deadlines. Conversely, a large synchronous bandwidth results in a long network access time, and this may cause message deadlines to be missed. Thus, the proper selection of TTRT is very important.

Let τ be the token walk time around the network. Then, the proportion of time taken due to the token walking is given by τ /TTRT. The maximum network utilization available to user application is then 1-τ /TTRT. A smaller TTRT may make the network utilization unavailable and limits the network capacity. On the other hand, if TTRT is too large, the token for a node to meet message deadlines may not arrive in time. According to Johnson and Sevcik's theorem, the token rotation time is bounded by 2TTRT. In this paper, this theorem is applied to modify Profibus protocol properly.

Each node has buffers for outgoing and incoming synchronous messages. The size of these buffers also affects the performance of the real time network. A buffer that is too small may not keep the messages stable due to the buffer overflow. In opposite case where buffer is too large, it will waste memory excessively.

The Profibus MAC is based on both a token passing procedure and a master-slave procedure. A token passing procedure is used for communication between the master stations each other, and a master-slave procedure is used for communication between the master station and the slave station.

The Profibus token passing procedure uses a simplified timed token protocol. One important parameter to consider in this kind of protocols is the target token rotation time(T_{TR}), which is set and assigned for all network nodes at the network initialization time. Whenever a station receives the token, it transmits its highest priority message within the allocated time period by the synchronous bandwidth(H_i). The lower priority messages can be transmitted only if the previous token rotation time is shorter than T_{TR}. Therefore, the amount of time that a station may hold the token is dynamically adjusted to the speed of token rotation.

The main difference between the Profibus token passing protocol and the timed token protocol is the absence of synchronous bandwidth allocation in Profibus. When the station receives a token in a late time, that is, in case that the token rotation time is greater than the target token rotation time, only the highest priority message can be transmitted. As a consequence, in Profibus, the lower priority traffic capabilities may be affected drastically by the higher priority traffic capabilities. In fact, when a station receives a token early, that is, in case that the token rotation time is shorter than the target token rotation time and the lower priority traffic is not constrained to stay in that station, the subsequent stations may be limited to transmit one higher priority message in case of holding the token.

2. PROFIBUS NETWORKS

As previously mentioned, the Profibus MAC includes a token passing procedure which is used by the master stations to communicate with each other and a master-slave procedure which is used by master stations to communicate with the slave stations. Fig. 1 illustrates this hybrid-operating mode.

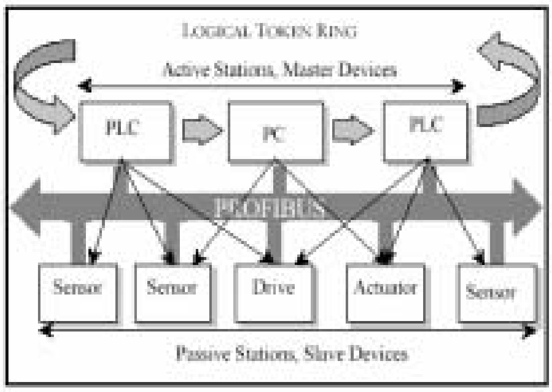

Fig. 1 Hybrid-operating mode

The MAC protocol is implemented with the layer 2 of the OSI reference model which is called Fieldbus Data Link(FDL) in Profibus. In addition to controlling the bus access and the token cycle time, the FDL is also responsible for the provision of data transmission services for the FDL user (e.g. the application layer).

Profibus supports four basic data transmission services such as Send Data with No acknowledge (SDN), Send Data with Acknowledge (SDA), Request Data with Reply (RDR) and Send and Request Data(SRD). The SDN is an unacknowledged service mainly for broadcasting from an active station to all the other stations on the bus. Conversely, all the other services except SDN are based on a real dual relationship between the initiator that is the master station with the token and the responder. The responder which can be passive or active station has to send a response frame or an acknowledge frame immediately when it receives the service request such as SDA, RDR and SRD only except SDN.

The operating principle of Profibus MAC is described as below. After receiving the token, the measurement of the token rotation time begins. This measurement stops at the arrival time of the next token and this measured interval becomes the real token rotation time(T_{RR}). T_{RR} is the significant factor in case of carrying out the low priority message cycles. In order to keep the system reaction time stable, the target rotation time(T_{TR}) of the token must be defined in a Profibus network.

Each master station may always execute at least one high priority message cycle after token receipt regardless of the real token rotation time. In order to perform the lower priority message cycles, T_{RR} must be shorter than T_{TR} at the instant of execution. Otherwise, the station retains the low priority message cycles and transmits them at the next token receipt. Once a message cycle is started, it should be completed even if it retries any required transmission regardless of the token holding time.

3. FRAME WORK FOR TIMED TOKEN PROFIBUS PROTOCOL

3.1 Network and message model

In order to get the network and message model, we assume the following constraints:

(1) there are n nodes in the ring type network.
(2) the network works well without any faults.
(3) the message transmission is controlled by the timed token protocol.
(4) outgoing messages at a node are queued in FIFO order.

The token walk time is denoted by τ . τ includes the node-to-node delay and token transmission time.

We denote the n streams of synchronous messages as $S_1, \ldots, S_n$ with stream S_i incident on node i. Each synchronous message stream S_i may be characterized as $S_i = (C_i, D_i)$ where C_i is the maximum amount of time required to transmit a message in the stream and D_i is the relative deadline of messages in the stream. In Profibus, C_i should also include possible message retries. The relative deadline is the maximum amount of time from a message arrival to the completion of its transmission.

The utilization, U, of a synchronous message set is defined by

$$U = \sum_{i=1}^{n} \frac{C_i}{\max(D_i, P_i)} \tag{1}$$

U can be regarded as the proportion of time required for synchronous traffic to the deadline in the network.

3.2 Timed token protocol constraints

In the timed token protocol, access to the ring is controlled by a token that circulates through the nodes. The idea behind the timed-token protocol is how to control the token rotation time. A protocol parameter called the target token rotation time(TTRT) provides the expected token rotation time. An amount of TTRT is assigned to each node, and the synchronous bandwidth(H_i) is the maximum permitted time for the node to transmit synchronous messages whenever it receives the token. The allocated synchronous bandwidth is always longer than the transmission time. When a node receives the token, it is general to transmit its synchronous messages.

We denote the token walk time by τ . The token walk time includes the node-to-node delay and the token transmission time. Thus, τ is the portion of TTRT but the message transmission time is not included in τ. Let α be the ratio of τ to TTRT, i.e., $\alpha = \tau$ /TTRT. Thus, α represents the proportion of the time excluding the message transmission to TTRT.

The timed token protocol requires that the total bandwidth allocated to synchronous messages should be shorter than the available network bandwidth, i.e.,

$$\sum_{i=1}^{n} H_i \le \text{TTRT - } \tau \qquad (2)$$

This constraint, known as the protocol constraints [2], is necessary for stable operation of the protocol.

The physical meaning of the above inequality is that the summation of the synchronous bandwidth assigned to the nodes in the token ring can not exceed the effective ring bandwidth. Theoretically, the total available time to maintain the real-time traffic, during the token rotation cycle, can be as long as T_{TR}. However, the network factors such as ring latency and other protocol or network overheads reduce the total available time to persist the real-time traffic. We define τ which excludes the synchronous message transmission time from T_{TR}.

The deadline means the allocated synchronous capacities to the nodes and the synchronous messages must be transmitted within their deadlines. In order to guarantee to transmit the synchronous message for a node, it is necessary to have some information about the token visit cycle to that node. Fortunately, extensive studies have already been carried out on the timing properties for the timed token protocol. In this paper, we have studied the properties of Johnson and Sevcik[3,4]. Johnson and Sevcik have shown that the maximum amount of time that may pass between two consecutive token arrivals at a node can approach $2T_{TR}$. This time bound holds regardless of the behavior of asynchronous messages in the network. Generalized Johnson and Sevcik's theorem is as follows.

For $1 \le i \le n$ and $j \ge 1$, the maximum amount of time that may pass between the j-th token arrival at node i and the (j+k)-th token arrival at node i satisfies

$$T_{i,(j+k)} - T_{i,j} \le (k+1)T_{TR} - H_i \qquad (3)$$

In particular, the time between consecutive token visits to a node i satisfies

$$T_{i,(j+1)} - T_{i,j} \le (k+1)T_{TR} - H_i \qquad (4)$$

This is the relationship between two consecutive token arrivals at a node that has been derived by Johnson and Sevcik.

We denote the maximum time between consecutive token visits to a station i by $T_{cycle}^{(i)}$.

Equation (4) is rewritten as follows.

$$T_{cycles}^{(i)} \le (k+1)T_{TR} - H_i \qquad (5)$$

4. SYNCHRONOUS BANDWIDTH ALLOCATION

The selection of appropriate values for the synchronous bandwidths H_i is a crucial step to satisfy the message deadlines. This selection proposes a new synchronous bandwidth allocation scheme in order to guarantee the time constraints of synchronous messages with arbitrary deadlines.

Synchronous bandwidth allocation schemes may be divided into two classes: local allocation schemes and global allocation schemes. These schemes differ in the type of information they may use. A local synchronous bandwidth allocation scheme uses only the information available locally at node i for allocating H_i. Locally available information at node i includes the parameters of stream S_i (i.e., Ci, Pi, and Di). τ and TTRT are also locally available at node i, because these values are known to all nodes. On the other hand, a global synchronous bandwidth allocation scheme can use global information in order to allocate the synchronous bandwidth to a node. Global information includes both locally available information of a node and information for the parameters related with the synchronous message streams at the other nodes.

A locally allocation scheme is preferable to the global allocation scheme in respect to a network management. Even if the parameters of stream S_i on node i change, it is sufficient to calculate only the synchronous bandwidth H_i of node i. The synchronous bandwidths at other nodes are not changed because they are calculated independently on S_i. This makes a local allocation scheme flexible and suitable for use in dynamic environments.

In a global allocation scheme, if the parameters of S_i are changed, then it may be necessary to compute all of the synchronous bandwidth at other nodes. Thus a global allocation scheme is not suitable to a dynamic environment. On the other hand, a global allocation scheme may perform better than a local one because it uses extra information. However, it is known that the performance of the local allocation scheme is very close to that of the optimal synchronous bandwidth allocation scheme when message deadlines are equal to message periods [1,4]. According to reason mentioned above, this paper concentrates on the local synchronous bandwidth allocation schemes.

Several local synchronous bandwidth allocation schemes have been proposes both for the case of $D_i = P_i$ and for the case of $D_i \ne P_i$. All of these schemes have a similar performance in the worst case. In this article, we consider a simplified version of the scheme proposed by Malcolm and Zhao and by Zheng and Shin. In these schemes, the synchronous bandwidth for node i is allocated as

$$\frac{C_i}{\left\lfloor \frac{D_i}{T_{TR}} - 1 \right\rfloor} \le H_i \qquad (6)$$

A message must be sent within D_{H_i} time in order to satisfy its deadline. Eq.(6) suggests that for a message at node i to satisfy its deadline, the synchronous bandwidth H_i must be sufficient to send the message within the token visit time $\left\lfloor \frac{D_i}{T_{TR}} - 1 \right\rfloor$.

Eq. (6) is rewritten as follows.

$$\frac{C_i}{\left\lceil \frac{T_{TR} + n \times C_{\max}}{T_{TR}} - 1 \right\rceil} \le H_i \qquad (7)$$

The synchronous bandwidth are allocated using the scheme in Eq. (6), then the worst case utilization, U^*, of the network is given by

$$U^* = \frac{\left\lceil \dfrac{T_{TR} + n \times C_{\max}}{T_{TR}} - 1 \right\rceil}{\left\lceil \dfrac{T_{TR} + n \times C_{\max}}{T_{TR}} + 1 \right\rceil}(1-\alpha) \qquad (8)$$

5. SELECTION OF TTRT

This section considers how to select T_{TR}. When a station 1 receives the token, it has to send messages during a time length T_{TH}. In fact, the token can be hold in a station even if the token holding time, T_{TH}, has elapsed in case of starting to transmit the message, including the non real-time message, issued by that station before the token holding time. This may happen because once Profibus starts to send a message it will be continued until the end of the message cycle even if T_{TH} time has elapsed.

For simplification, we can derive the expression for T_{cycle} using the maximum length for a message including high and low priority traffic in the network.

$$T_{cycle} = T_{TR} + C_i + Ch_i^{(1)} + Ch_i^{(2)} + Ch_i^{(3)} + Ch_i^{(4)} + \tau \quad (9)$$

$$T_{TR} \le \frac{1}{\sum_{i=1}^{nh} \frac{1}{D_i}} - n \times C_{\max} + \tau \qquad (10)$$

The evaluation of T_{cycle} is carried out as following process. First, the master station i with token starts to transmit the message. This master can actually hold the token during T_{TH} plus the duration of its longest message. In this situation, all the following masters in the logical ring will receive a token late, thus only one high priority message can be transmitted. Fig.2 illustrates this scenario.

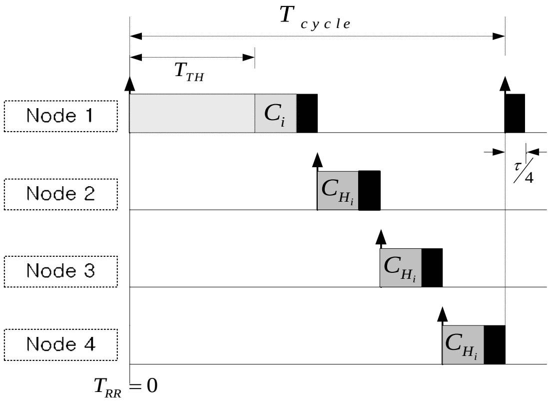

Fig. 2 Token arrival time

6. CONCLUSION

In this paper, we have presented how to obtain the optimal network parameter for the Profibus protocol. Our methodology by using the timed-token protocol for the real time communication is very useful to select TTRT in order to improve the network performance, especially in the Profibus network. Because the synchronous bandwidth allocation method uses only local information at a node, we can create or modify synchronous message streams locally without reinitializing the network. The proposed network parameter selection method can be applied to the Profibus network. And also, the selected network parameter is valid regardless of the behavior of asynchronous messages because it is selected with the assumption of the worst case of asynchronous traffic.

ACKNOWLEDGMENTS

This work was supported by the Korea Science and Engineering Foundation(KOSEF) through the Network-based Automation Research Center(NARC) at University of Ulsan.

REFERENCES

[1] R. M. Grow. “A timed token protocol for local area networks.” *In Proceedings of Electro/82, Token Access Protocols,* pp. 17/3, 1982.

[2] G. Agrawal, B.Chen, W. Zhao, and S. Davari. “Guaranteeing synchronous message deadline in high speed token ring networks with timed token protocol,” *In Proceeding of the 12th IEEE International Conference on Distributed Computing Systems*, pp. 468-475, 1992.

[3] M. J. Johnson, “Proof that timing requirements of the FDDI token ring protocols are satisfied,” *IEEE Trans. on Communications*, COM35(6), pp. 620-625, 1987.

[4] K. C. Sevcik and M. J. Johnson, “Cycle time properties of the FDDI token ring protocol,” *IEEE Trans. On Software Engineering*, SE13(3), pp. 376-385, 1987.