'91 KACC 1991. 10. 22~24

PERFORMANCE AND PARAMETER REGION FOR REAL TIME USE IN IEEE 802.4 TOKEN BUS NETWORK

Hong Seong Park, Deok-Woo Kim and Wook Hyun Kwon

Engineering Research Center for
Advanced Control and Instrumentation
and
Automation and Systems Research Institute

Seoul National University
Sillim_dong Kwanak_gu, Seoul 151-742, KOREA
Tel. +82-2-887-0040 FAX. +82-2-871-7010

ABSTRACT

This paper derives the upper and the lower bound of the mean cycle time and the mean service time of the class 6 and the class 4, within which the minimum utilization constraint of the class 4 is guaranteed. Also, derived are conditions under which the token bus network is stable or unstable. These bounds and stable conditions are represented in terms of the high priority token hold time, the token rotation time and the arrival rate and the total station number etc. This paper suggest a parameter tuning algorithm in a partially symmetric token bus network with two classes, which maximizes the token rotation time for a suitable high priority token hold time and at the same time meets the stability condition of the network, the real time constraint and the minimum utilization constraint of the class 4.

1. INTRODUCTION

The distributed systems, which are used in the manufacturing environment like a CIM(Computer Integrated Manufacturing) system, require the Local Area Networks(LAN's) for the data exchange between computers, robots, data acquisition systems, and other devices. Through these networks the several types of messages with different requirements are transferred. Real time messages for alarming and controlling are required to be transmitted as quickly as possible within a specified time limit. Non real time messages for file transfer are not required to be transmitted quickly. Therefore, messages used in the networks can be classified into the real time and the non real time class.

In networks, generally, the priority mechanisms are adopted for managing the these classes of messages. In the IEEE 802.4 token bus network[1] considered in this paper, the priority mechanism of is designed to control the time constraints of the messages associated with the devices connected to the network. More specifically, in the IEEE 802.4 standard the optional priority mechanism with 4 classes (class 6, 4, 2, and 0) allocates the different service time to each class by setting timers called THT(High Priority Token Hold Timer of class 6) and TRT_i(Token Rotation Timer of class i, i=4, 2, and 0). In the real time application, the class 6 can deal with real time messages and the class 4, 2 and 0 can deal with non real time messages as explained in the next section. This paper considers a partially symmetric token bus network with two classes; i.e. the class 6 for the real time application and the class 4 for the non real time application.

It is well known that the high priority token hold time and the target token rotation time as well as the arrival rate of the messages have a great effect on the mean cycle time, the mean service time and the mean waiting time of each class. For the token bus system with a single class, the mean cycle time and the mean waiting/delay time were investigated under the assumption of the infinite buffer capacity at each station[2] or under the assumption of a single buffer capacity at each station[3]. For the token bus system with multiple classes, Hong and Ray[4] suggested an analytic model for the average delay time of each class. But the network models in these studies[2-4] are different from the IEEE 802.4 token bus network since there exist two types of the timers, THT and TRT_i, in the IEEE 802.4 token bus network while no timer[2-3] or the only one type timer[4] similar to TRT_i is used. Pang and Tobagi[5] derived the mean cycle time and the throughput of the IEEE 802.4 token bus network under the unrealistic assumption of heavy load. Under fairly realistic assumptions the mean cycle time and the mean service time of each class are very difficult to be represented by the analytic forms. In this paper, under fairly realistic assumptions, we present the upper and the lower bounds, instead of analytic forms, of the mean cycle time and the mean service time of each class.

It is also well known that the high priority token hold time and the target token rotation time as well as the arrival rate of the messages have great effects on the real time constraints, the minimum utilization of the class 4, and the stability of the network. Jayasumuna et al.[6] have derived the conditions on the high priority token hold time and the target token rotation time of class 4 which satisfy the real time constraint and the minimum utilization constraint of the class 4 for the partially symmetric token bus network. The network must be stable in the real time applications since the arriving message can be canceled in the unstable network. In this viewpoint, the result derived by Jayasumuna et al. is insufficient for the real time application since the stability condition is not added. Under assumptions that the synchronous class messages are generated according to a periodic pattern and that the traffic of the other class is heavy, Montuschi et al.[7] have derived the conditions of the high priority token hold time which stabilize the token bus network. In this paper, under fairly realistic assumptions such that messages of the classes are generated according to a random pattern and the traffic of all the classes is not always busy, we derive the conditions under which the given network is stable or unstable.

By combining these stable conditions and the conditions derived by [6], we can also derive the conditions of the high priority token hold time, the target token rotation time and the arrival rate of messages, which meet the real time constraint and the minimum utilization constraint of the class 4 and at same time stabilize the network with two classes generated according to a random pattern. By using these derived conditions, the parameter tuning algorithm is presented for a partially symmetric token bus network with two classes, which maximizes the network throughput and at same time meets the stability condition of the network, the real time constraint and the minimum utilization constraint of the class 4. The parameters obtained from this algorithm, are the optimal values of the high priority token hold time and the target token rotation time.

In Section 2, the IEEE 802.4 token bus network is reviewed and the token bus model considered in this paper is given. Also, several constraints used in this paper are given. In Section 3, we derive the upper and the lower bounds of the mean cycle time and the upper bound of the mean service times of the class 6 and the class 4. Also, the stable conditions and the unstable conditions of the given token bus model are derived. In Section 4, the parameter tuning algorithm is suggested, from which we can obtain the high priority token hold time and the target token rotation time to stabilize the network and to guarantee the real time constraint and the minimum utilization constraint of the class 4. Examples are also given. Finally, we have the conclusion in Section 5.

2. IEEE 802.4 TOKEN BUS MODEL

The IEEE 802.4 token bus network[1] can be considered conceptually as a logical token ring network and provides an optional 4-class priority mechanism : class 0(lowest), 2, 4, and 6(highest). Fig. 2.1 shows the IEEE 802.4 token bus model consisted of several stations, where one station is composed of separate queues for each class.

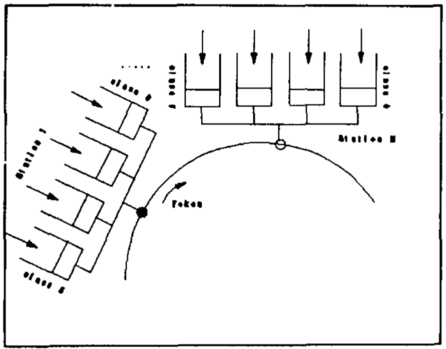

Fig. 2.1 Token Bus Model

When a station receives a token, the token is passed internally from the highest priority class to the lowest priority class before it is passed to the next station. And, when a class receives the token, its messages are transmitted either until its timer expires or until its queue is empty. The high priority token hold timer exists in the class 6 and the token rotation timer i exists in the class i(i=4, 2, and 0). When the class 6 receives the token, the high priority hold timer is reset to the value of the high priority token hold time, which is denoted by T_h. Messages are transmitted until the queue of the class 6 is empty or until the this timer expires. And, when this timer expires or the queue is empty, transmission of the message should be completed, and the token is passed to the next lower priority class. The three lower classes have each parameter called the target token rotation time, denoted by $T_{r,k}$, where k indicates the class k(k=4, 2, and 0). The time instant that the token arrives at the class k in the previous cycle or in the current cycle is denoted by $T_{p,k}$ or by $T_{c,k}$, respectively. If the time interval measured in the class k, $(T_{c,k} - T_{p,k})$, is less than $T_{r,k}$, transmission of messages in the class k is available either during the time interval $[T_{r,k} - (T_{c,k} - T_{p,k})]$ or until the queue of the class k is empty. When the target token rotation timer expires or the corresponding queue is empty, transmission of the message should be completed, and the token is passed to the next lower priority class or the next station.

Though the corresponding timer expires, the transmission of the current message should be completed before the token is transmitted to the next class or to the next station. We denote the maximum and mean value of this overshoot for class k by Tx_k and by Tm_k, respectively. And the mean cycle time and the mean service time of the class k are denoted by Ec and by H_k. The mean service time of each class as well as the delay of the message depend on the T_h or on the $T_{r,k}$. These values can be set by the station management.

For the real time applications, all messages arrived in the previous cycle time must be transmitted. This is possible in the class 6 since we can set the high priority token hold time to transmit all messages arrived in the previous cycle. But the class i(i = 4,2,0) is not used in the real time applications. This reason is as follows; as mentioned above, each class's service time can be zero or not be zero. Even when the service time is not zero, the service time in one cycle time is different from that in the other cycle time since the time interval measured in the class k, $(T_{c,k} - T_{p,k})$, is varied. Therefore, it is not guaranteed that all messages arrived in the previous cycle time is always transmitted.

Let us consider our model as the partially symmetric token bus network with N stations and one station has two classes such as the class 6 and the class 4. The class 6 corresponds to messages required for the real time applications and the class 4 consists of messages required for the non real time applications. Since one station generates the messages of the class 6 or the class 4 or both in the partially symmetric token bus network, we can denote the number of the station that generates the messages of the class 6 or the class 4 by N_6 or by N_4, respectively. $T_{r,4}$ is abbreviated as T_r hereafter. It is assumed that the operation of the network is error-free and the overhead time associated with the maintenance is neglected.

In next section, we suggest the upper bound within which the mean cycle time and the mean service time of each class can be taken in the token bus model. Also, the stable conditions and the unstable conditions of the given token bus model are given.

3. Conditions on Real Time Application and Stability

As mentioned in Section 1, the mean cycle time and the mean service times of each class in IEEE 802.4 token bus network are difficult to be obtained as the mathematical expressions. Thus we derive the upper/lower bounds of these performances. We can derive the mean cycle time from the following expressions[2].

$$EC = \sum_{1}^{N} T_t + \sum_{1}^{N_6} P_6 H_6 + \sum_{1}^{N_4} P_4 H_4 \quad (3.1)$$

where T_t is a token passing overhead time and P_i is a probability that class i will be severed. Let U_i and u_i denote the mean utilization of network by all queue of class i and by a queue of class i, respectively. Then, the following relation,

$$U_4 = N_4 u_4 \quad (3.2)$$

holds for class 4. Ec can be also expressed in terms of u_i as

First, the upper bound and the lower bound of the mean cycle time will be derived.

$$EC = \sum_{1}^{N} T_t + \sum_{i}^{N_6} u_6 EC + \sum_{1}^{N_4} u_4 EC \quad (3.3)$$

Theorem. 3.1 (Mean Cycle Time) The upper bound and the lower bound of the mean cycle time is given by

$$\frac{NT_t}{1-Un_4} \le EC \le \frac{NT_t+N_4(T_r+Tm_4)}{1+N_4-N_6\lambda_6(T_h+Tm_6)} \quad (3.4)$$

where λ_6 is the arrival rate of messages offered to the queue 6 and Un_4 is the minimum utilization of the class 4.

Proof: Since the mean utilization of class 4, U_4 is less than or equal to the minimum utilization of class 4, the following equation,

$$U_4 \ge Un_4 \quad (3.5)$$

can be derived. By Kuhen[2],

$$P_6 = \lambda_6 EC \quad (3.6)$$

From (3.1), (3.2), (3.3), (3.5) and (3.6),

$$EC \ge NT_t + N_6 \lambda_6 H_6 EC + Un_4 EC \quad (3.7a)$$

is obtained. Since $H_6 \ge 0$, it summarized to

$$EC \ge \frac{NT_t}{1-Un_4} \quad (3.7b)$$

From [6],

$$u_4 \le \frac{T_r+Tm_4-EC}{EC} \quad (3.8)$$

is obtained. And From (3.1), (3.2), (3.3), (3.6) and (3.8),

$$EC \le \frac{NT_t+N_4(T_r+Tm_4)}{1+N_4-N_6\lambda_6 H_6} \quad (3.9)$$

can be obtained. Since $H_6 \le T_h + Tm_6$, it follows that

$$EC \le \frac{NT_t+N_4(T_r+Tm_4)}{1+N_4-N_6\lambda_6(T_h+Tm_6)} \quad (3.10)$$

Thus, (3.4) is obtained from (3.7b) and (3.10). The proof is completed.

Generally, since the mean utilization of class 4 depends on parameters such as the mean utilization of class 6, the minimum utilization of class 4 can be limited to a value. Lemma 3.1 shows the upper bound of the minimum utilization of class 4.

Lemma 3.1 (Upper bound of Un_4) The maximum utilization of class 4 is given by

$$Un_4 \le \frac{N_4(T_r+Tm_4)+NT_tN_6\lambda_6(T_h+Tm_6)-NT_tN_4}{NT_t+N_4(T_r+Tm_4)} \qquad (3.11)$$

Proof: From (3.5), the following relationship,

$$\frac{NT_t}{1-Un_4} \le \frac{NT_t+N_4(T_r+Tm_4)}{1+N_4-N_6\lambda_6(T_h+Tm_6)}$$

can be derived. If this equation is summarized in terms of Un_4,

$$Un_4 \le \frac{N_4(T_r+Tm_4)+NT_tN_6\lambda_6(T_h+Tm_6)-NT_tN_4}{NT_t+N_4(T_r+Tm_4)}$$

is given. The proof is completed.

Next, consider the upper bound of the mean service time of class 6, H_6. In the class 6, messages are transmitted until the queue of the class 6 either has nothing more to transmit or the high priority token hold timer expires, from which these bounds are derived.

Theorem 3.2 (mean service time of class 6) The maximum value of the mean service time is given by

$$H_6 \le \frac{(1-Un_4)(T_r+Tm_4)N_4+NT_t\left(N_6\lambda_6(T_h+Tm_6)-Un_4-N_4\right)}{N_6\lambda_6\left(NT_t+N_4(T_r+Tm_4)\right)} \qquad (3.12)$$

Proof: (3.7a) can be changed to

$$H_6 \le \frac{Ec-NT_t-Un_4Ec}{N_6\lambda_6Ec} \qquad (3.13)$$

From $Ec \le Ec_{\max}$ and (3.4), it is summarized to

$$H_6 \le \frac{(1-Un_4)(T_r+Tm_4)N_4+NT_t\left(N_6\lambda_6(T_h+Tm_6)-Un_4-N_4\right)}{N_6\lambda_6\left(NT_t+N_4(T_r+Tm_4)\right)}$$

This completes the proof.

Third, consider the upper bound of the mean service time of class 4, H_4. If the time interval measured in the class 4, $(T_{c,4} - T_{p,4})$ is less than T_r, transmission of messages in the class 4 is available during the time interval $[T_r - (T_{c,4} - T_{p,4})]$ or until the queue of the class 4 is empty. The bound indicated above is obtained from these conditions.

Theorem. 3.3 (mean service time of class 4) The mean service time of class 4, H_4 is given by

$$H_4 \le \frac{T_r+Tm_4-NT_t-N_6(T_h+Tm_6)}{1+N_4} \qquad (3.14)$$

Proof: From (3.1), $H_6 \le T_h + Tm_6$ and $P_6 \le 1$,

$$Ec \le NT_t+N_6(T_h+Tm_6)+N_4P_4H_4 \qquad (3.15a)$$

is obtained. Since $P_4 \le 1$, it follows that

$$Ec \le NT_t+N_6(T_h+Tm_6)+N_4H_4 \qquad (3.15b)$$

From [6], the maximum value of the mean cycle time as

$$EC_{\max} = \frac{N_6(T_h+Tm_6)+N_4(T_r+Tm_4)+NT_t}{1+N_4} \qquad (3.16)$$

is given and is obtained in the worst case. Comparing between (3.15b) and (3.16), the relation as

$$NT_t+N_6(T_h+Tm_6)+N_4H_4 \le \frac{N_6(T_h+Tm_6)+N_4(T_r+Tm_4)+N}{1+N_4}$$

holds. (3.14) is obtained from summarizing this relation. The proof is completed.

It is known that the network becomes unstable when, on the average, all stations generate more messages than the maximum number of messages that can be transmitted. In this case, it is said that the network is working in a global instability. In contrast, it is said that the network is working in a local instability when, on the average, one or more stations generate more messages than the maximum number of messages that can be transmitted. Let λ_4 denote the arrival rate of messages offered to the queue 4. The condition for the global instability can be derived from the condition for the local instability. Thus, the local instability and stability is considered in this paper. If the arrival rate, λ_i is greater than the maximum value of the service rate of messages, it can be said that the class i is locally unstable(i=6,4). The conditions for a class 6 and a class 4 to be unstable are expressed in (3.17) and (3.18).

$$\lambda_6 > (T_h+Tm_6)/Ec \qquad (3.17)$$

$$\lambda_4 > [(T_r+Tm_4)-Ec]/Ec \qquad (3.18)$$

Then the sufficient condition for class 6 to be unstable can be obtained from (3.17).

Theorem. 3.4 (Local instability of class 6) A sufficient condition for one queue of class 6 to become unstable is expressed by relation as

$$\lambda_6 > \frac{(1-Un_4)(T_h+Tm_6)}{NT_t} \qquad (3.19)$$

Proof: For the network the minimum mean cycle time,

$$Ec_{\min} = \frac{NT_t}{1-Un_4} \qquad (3.20)$$

is obtained from (3.4). Thus, (3.19) is obtained from (3.17) and (3.20). The proof is completed.

Also, the sufficient condition for class 4 to be unstable can be obtained from (3.18).

Theorem. 3.5 (Local instability of class 4) A unstable condition for one queue of class 4 is expressed by relation as

$$\lambda_4 > \frac{(T_r+Tm_4)(1-Un_4)-NT_t}{NT_t} \qquad (3.21)$$

The above relation (3.21) can be obtained from (3.18) and (3.20). Theorem 3.4 and Theorem 3.5 state that each class of each station in the network satisfies relation (3.19) and (3.20) to be locally unstable. The next step is to derive conditions for the global stability.

Consider a condition that each class of one station is stable. The network does not become unstable even for the maximum duration of a average token cycle time, unless (3.17) or (3.18) holds, and this is a sufficient condition. These condition follow that

$$\lambda_6 < \frac{T_h + Tm_6}{Ec} \qquad (3.22)$$

$$\lambda_4 < \frac{(T_r + Tm_4) - Ec}{Ec} \qquad (3.23)$$

Then the sufficient condition for class 6 to be stable can be obtained from (3.22).

Theorem 3.6. (Local Stability of class 6) A sufficient condition for class 6 of one station to become stable is given by relation as

$$\lambda_6 < \frac{(1+N_4)(T_h+Tm_6)}{NT_t+N_4(T_r+Tm_4)+N_6(T_h+Tm_6)^2} \qquad (3.24)$$

Proof: The maximum mean cycle time ,

$$Ec_{\max} = \frac{NT_t+N_4(T_r+Tm_4)}{1+N_4-N_6\lambda_6(T_h+Tm_6)} \qquad (3.25)$$

is obtained from (3.4). From (3.22) and (3.25), it follows that

$$\lambda_6 < \frac{(1+N_4)(T_h+Tm_6)}{NT_t+N_4(T_r+Tm_4)+N_6(T_h+Tm_6)^2}$$

This completes the proof.

Also the sufficient condition for class 4 to be stable can be obtained from (3.23).

Theorem. 3.7. (Local Stability of class 4) A stable condition for class 4 of one station is given by relation as

$$\lambda_4 < \frac{(T_r+Tm_4)(1-N_6\lambda_6(T_h+Tm_6))-NT_t}{NT_t+N_4(T_r+Tm_4)} \qquad (3.26)$$

The avobe relation (3.26) can be obtained from (3.23) and (3.25). Theorem 3.6 and Theorem 3.7 state that each class of each station in the network satisfies relation (3.24) and (3.26) to be locally stable. As seen above, the stability depends on the arrival rate and the high priority token hold time and etc. Since the network should be stable in the real time control, we must stabilize the network by using these relations. In the next section, the tuning of parameters is considered.

4. Parameter tuning for the real time use

The real time application must send the message within a certain time limit, which can be satisfied with ensuring that the token returns to this station within the time limit denoted by D_a[7]. Thus, the class 6 must be used to manage the real time application and the T_h must be such that all messages generated during the last token cycle time can be sent to. Let's define the access delay as the time interval between two successive transmission periods of the queue. For the real time applications, the maximum access delay of the class 6 message must be allowed to be less than the T_h. Since the access delay of the class 6 message depends on the T_h and the arrival rate of class 6 messages, it can be considered that it is bounded by some values of the T_h and the arrival rate. In the case that the arrival rate is not considered, the real time constraints can be guaranteed in the case that the maximum access delay in the class 6 is less than D_a, but the stability of the network cannot be guaranteed. The real time constraint and the constraint for the minimum utilization of the class 4 are as follows[7]:

$$(T_r+Tx_4)+(N_6-1)(T_h+Tx_6) \le D_a \qquad (4.1)$$

$$T_r+Tm_4 > \left(1+\frac{Un_4}{N_4}\right)\frac{[NT_t+N_6(T_h+Tm_6)]}{1-Un_4} \qquad (4.2)$$

For the real time application, the class 6 must be stable. Section 3 has shown that the instability of the class 6 can be due to the wrong choice of the T_h , of T_r and of the message arrival rate. The network parameters can be suitably tuned by achieving a satisfactory tradeoff between the average and the maximum token cycle time, the amount of class 6 message that are transmitted and the need to avoid instability. The problem can be also tackled from a different point, when the network is planned, the total number of all stations and stations with class 6 and stations with class 4 is decided and the arrival rate of the message in each class is decided. Then, we carefully choose the T_h and T_r so that the network is stable. By using several theorems mentioned in section 3, we can determine the range of parameters which stabilize the network. Then we check that the given constraints are satisfied in the derived ranges of parameters. If the given constraints are not satisfied, we adjust the parameters or the given constraints. It is known that increasing the target token rotation time of the class 4 while the other parameters keep constant will increase the total utilization of the network[6]. Therefore, we use our performance measure of the total utilization of the network. From the following algorithm, we get the optimal parameters' values to maximize the total utilization of the network.

```
Parameter Tuning Algorithm For Real Time Control

step 1) set the following parameter's values.
        $N$, $N_6$, $N_4$, $Tx_6$, $Tm_6$, $Tx_4$, $Tm_4$, $\lambda_6$, and $\lambda_4$.
step 2) Select the one choice.
    2.1) If one want to stabilize both the class 6 and the class 4,
        2.1.1) obtain the common area of $T_h$ and $T_r$ to stabilize both the class 6 and
               the class 4 from the given parameters, (3.24) and (3.26).
        2.1.2) go to step (3).
    2.2) If one want to stabilize the class 6.
        2.2.1) obtain the common area of $T_h$ and $T_r$ to stabilize both the class 6 and
               from the given parameters and (3.24).
        2.2.2) go to step (4).
```

```
step 3) if the common area in step (2) dose not exist,
            go to step (1) to change the parameters' values and to stabilize the class 6 and
            the class 4.
        else
            go to step (4).
step 4) set the desired values of the real time limit $D_a$ and the minimum utilization of the
        class 4 $Un_4$ in (4.1) and (4.2).
step 5) obtain the common area of the high priority token hold time $T_h$ and the
step 6) obtain the common area of $T_h$ and $T_r$ from step (2) and step (5).
        If the common area does not exist,
            $D_a$ and $Un_4$ are changed for the common area to exist. Repeat step (6).
            Go to step (1) if the common area does not exist although these parameters
            are changed.
        else
            go to step (7).
step 7) select the suitable high priority token hold time $T_h$. Then obtain the optimal
        target token rotation time $T_r$ to maximize the total utilization of the network in the
        common area of step (6).
```

The aforementioned algorithm and the results can be used conveniently to select the network parameters for the real time application and to detect situations of instability. For instance, let us consider networks having parameters' values listed in Table 4.1. Given these parameters' values it is possible to compute the suitable values of T_h and T_r which prevent the network from being unstable, and which is used in the real time application and which guarantees the minimum utilization of class 4. From the suggested equations, we obtain the areas that T_h and T_r can take for the network to be stable and to be used in the real time applications and to guarantee the minimum utilization of class 4. And these areas, which are obtained from the above algorithm and values listed in Table 4.1, are shown in Fig. 4.1 - Fig. 4.3.

Table 4.1 List of Parameters

	Da	N_6	Tx_6	Tx_4	Un_4	N_4	N	T_t	Tm_6	Tm_4	λ_6	λ_4	
1	200	10	0.2	0.8	0.2	10	10	1	0	0	0.04	0.02	Fig. 4.1
2	200	5	0.2	0.8	0.7	10	10	1	0	0	0.04	0.02	Fig. 4.2
3	40	10	0.2	0.8	0.2	10	10	1	0.2	0.8	0.04	0.02	Fig. 4.3

5. Conclusion

In this paper, the partially symmetric IEEE 802.3 token bus network with the class 6 and the class 4 is considered and is analyzed under fairly realistic assumptions. And the parameter tuning algorithm is suggested in order to obtain the suitable high priority token hold time and the optimal target token rotation time for real time use.

If the suggested algorithm is very useful for parameter setting when the network is installed for the first time or when new stations are added, the high priority token hold time and the target token rotation time can be easily set. This paper deals with the partially symmetric token bus network. The more general case of the asymmetric network need to be investigated for real time use.

References

[1] ANSI/IEEE Standard 802.4: Token Passing Buss Access Method and Physical Layer Specification, IEEE, 1985.

[2] P.J. Kuhen, "Multiqueue Systems with Nonexaustive Cyclic Service," Bell System Tech. Jou., vol.58, pp.671-698, Mar.,1979.

[3] T.Takine, Y.Takahashi and T. Hasegawa,"Exact Analysis of Asymmetric Polling Systems with Single Buffers," IEEE Tr. on Comm., vol.36,pp.1119-1126, Oct., 1988.

[4] S.H. Hong,"Analysis of the Priority Scheme in Token Bus Protocol," ACC, pp.511-516, 1989.

[5] J.W.M. Pang and F.W. Tobagi,"Throughput Analysis of IEEE 802.4 Priority Scheme in Token Passing Protocol Under Heavy Load," IEEE Tr. on Computer, vol.37,pp694-702.

[6] A.P. Jayasumana and G.G Jayasumuna,"On the Use of the IEEE 802.4 Token Bus in distributed Real_Time Control Systems,"IEEE Tr. Indus. Elec., vol.36, no.3,pp.391-397.

[7] P.Montushi, A. Valenzano and L. Ciminiera, "Requirements for Token Holding Times in Timed -Token Protocols," Phoenix Conf. Proc. on Computers and Communications, pp.598-604, Mar., 1990.

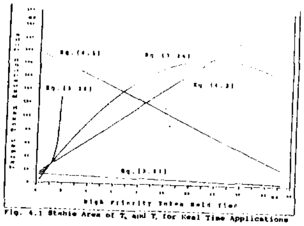

Fig. 4.1 Stable Area of T_h and T_r for Real Time Applications

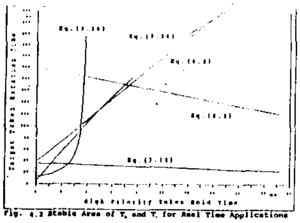

Fig. 4.2 Stable Area of T_h and T_r for Real Time Applications

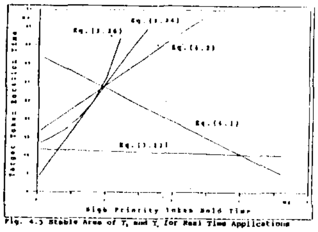

Fig. 4.3 Stable Area of T_h and T_r for Real Time Applications