Protocol Specification, Testing, and Verification, XI
B. Jonsson, J. Parrow and B. Pehrson (Editors)
Elsevier Science Publishers B.V. (North-Holland)

STOCHASTIC PETRI NET MODELING OF THE FDDI NETWORK PROTOCOL

I. F. Akyildiz, G. Chiola†, D. Kofman‡ and H. Korezlioglu‡

College of Computing
Georgia Institute of Technology
Atlanta, GA 30332 U.S.A.

†Dipartimento di Informatica
Universita di Torino
10149 Torino, ITALY

‡Ecole Nationale Superieure Telecommunications
Departement Reseaux
75634 Paris Cedex 13, FRANCE

Abstract

The Fiber Distributed Data Interface (FDDI) is a token ring network protocol over fiber optic rings with a transmission rate of 100 Mbps. The protocol supports a synchronous traffic class which offers guaranteed response time and guaranteed bandwidth as well as an asynchronous traffic class. The access to the medium for these traffic classes is controlled by the timed token protocol. In this paper the performance of the FDDI protocol is analyzed using Stochastic Petri Net (SPN) model. To date all performance studies about FDDI were based on queuing theoretic approaches. This paper is the first attempt to derive an SPN model of an FDDI network with both synchronous and asynchronous traffic consideration. Performance measures are obtained by using a programming tool Great-SPN package.

Key Words: High Speed Local Area Networks, FDDI protocol, Stochastic Petri Nets, Timed Token Protocol, Target Token Rotation Time.

---

* I. F. Akyildiz was supported by the Universite Pierre et Marie Curie (Lab MASI) and the Ecole Nationale Superieure Telecommunications (ENST), Paris, France. G. Chiola was a visiting researcher at the Universite of Pierre et Marie Curie (Lab MASI), Paris, France, with the support of a NATO-Italian CNR research grant during the academic year 1989/90. D. Kofman was supported in part by the Universidad de la Republica del Uruguay, Montevideo, Uruguay.

1 INTRODUCTION

FDDI is a protocol designed for a 100 Mbps token passing ring using a fiber optic medium. The standard supports two traffic classes, synchronous and asynchronous, both handling packet switched traffic. The synchronous traffic is high priority or time critical data. Each station is guaranteed a maximum waiting time before it will receive the token, and may transmit any synchronous traffic for a specified time upon receiving the token. Asynchronous frames are used to carry any other data, and can only be transmitted when all of the capacity of the ring is not needed for synchronous data. Only one station may transmit at a time and the right to transmit is controlled by passing a token from station to station. A station wishing to transmit must wait until receiving the token. After receiving the token, a station may transmit for a specified maximum time (or until all data are sent), and then must pass the token to the next station. Data frames are removed from the ring by the station which transmitted them, after passing completely around the ring.

Each station receives the data which is transmitted by its "upstream" neighbor and passes the data on to the next station, or "downstream" neighbor. If a station detects its own address in the destination address portion of a passing data frame, then it copies the frame into its own buffer space as well as passing the frame on to the next station. In order to keep the token moving around the ring within specified time constraints, and thus guaranteeing that each station may transmit within these limits, each station has two timers: the token rotation timer TRT and the token holding timer THT. During ring initialization the stations negotiate a target token rotation time TTRT. Each station is assigned a percentage of the TTRT for its synchronous packet transmissions. The TRT is initialized at the TTRT value when the token arrives at the station and each time it expires. At the token arrival time two possibilities exist:

i) The TRT timer is expired, i.e., the token is late. In this case the station has no right to transmit asynchronous traffic.

ii) The TRT timer is not expired, i.e., the token is early. In this case the value of the TRT is copied into the THT timer. The station transmits its synchronous traffic and then the THT timer is activated. The asynchronous transmission starts and continues until the THT timer expires as long as there is data to transmit. If there is a packet on the way when the time expires then the station releases the token only after the whole packet has been transmitted.

The FDDI protocol has been developed from the ideas of Grow [5] and Ulm [17]. An OSI standard is available since 1989 [1]. Johnson [6,7] investigated the robustness and the reliability issues of the FDDI protocol. Lundy and Akyildiz [11] specified the protocol using systems of *communicating machines* and analyzed it for safety, liveness and timing properties. Some basic properties of the timed token protocol have been proved by Sevcik and Johnson [7,15]. Analytical and simulation results for delay and throughput are derived in [3,4,8,9,12,14,16,18]. All of these studies are based on queueing theoretic approaches. Moreover, all solutions so far have very restrictive assumptions like one asynchronous traffic type or no synchronous traffic at all. In other words, there is no complete analytical model which captures all the characteristics of the protocol. To our knowledge this paper is the first attempt to develop a complete stochastic Petri net model by which the FDDI performance will be analyzed where both the synchronous and asynchronous traffic are considered.

The practical use of Petri Nets lead to the development of complex software tools to allow the users to define a place-transiton model to validate it, to build the infinitesimal generator of the associated Markov process and to compute the needed steady state or transient parameters at least by a linear system of equations solver. There are several PN software tools available on the market. For our model we used the Great-SPN [2] package which implements the qualitative and the quantitive analysis of GSPNs with instantaneous, exponentially distributed and deterministic transition times.

In this paper we are able to present an exact model of the FDDI protocol. It is easy to verify that the extensions to the FDDI protocol with several priority levels for the asynchronous traffic can easily be included into the Petri net model. The paper is organized as follows. In section 2 we develop the stochastic Petri net model. In section 3 we discuss several experiments and show the performance of the protocol under different workload. In section 4 we conclude our results.

2 PETRI NET MODEL OF THE FDDI

The stochastic Petri net model contains all features defined in the protocol [1]. In the model we consider the synchronous and asynchronous traffic, two independent message sources, two independent buffers, TRT and THT timers, the TTRT parameter, and the propagation delay on the ring (which represents the distances between the stations). The packet length is assumed to be an input parameter of the model. We show the complete model of a particular station in Figure A1 given in the Appendix. Each station has exactly the same structure as in Figure A1. They differ only by the values of some of their characteristic parameters. All stations are connected through the places MAR and TAR of Figure A1. As performance measures we determine the delay time (delay between the arrival of a packet at the MAC layer buffer and its transmission) and the throughput (average number of packets transmitted per second) for synchronous and asynchronous traffic.

We discuss the entire model module by module in the following sections. The model uses instantaneous transitions (thin bars; denoted by t's), and deterministic transitions (thick bars; denoted by T's). Instead of "tokens" from the Petri Net terminology we will use "marks" in our model in order to solve the ambiguity between the actual token of the protocol and the "tokens" of the Petri Nets.

2.1 Sources

We have two separate sources for synchronous and asynchronous traffic. The source for synchronous traffic is shown in Figure 1.

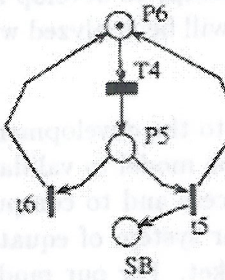

Figure 1: Source for the Synchronous Traffic.

The packets arrive from these sources due to the geometrically distributed times with rate α/los. α is the probability that the transition $t5$ puts a mark in place SB, i.e., a synchronous message is put in the buffer SB. $(1-\alpha)$ is the probability that the transition $t6$ will fire. In both cases a mark is put back in place $P6$ in order to make new packet arrivals possible. The duration time for transition $T4$ is deterministically distributed with the rate los. The asynchronous source is the same as in Figure 2. The only difference to Figure 1 is that the arrival rate for the transition $T3$ is given by the variable $alos$.

2.2 Input Buffers

To model the finiteness of the synchronous buffer, the place SBS is added to Figure 1. The buffer size is given by the variable bus, i.e., the initial number of marks in place SBS. Each time a message is transmitted a mark is put into place SBS to make a place available for a new message in the buffer. The place SB contains the messages which are still in the station. The input buffer $ASBS$ is used for the asynchronous traffic where the buffer size is denoted by the variable $buas$, i.e., the initial number of marks in place $ASBS$.

When a station receives a message, two possibilities occur: In the first case the message was sent by this station, then it will be removed from the ring. In the second case the message was sent by another station, then it must be forwarded. In the latter case, if the station under consideration is the destination station, the message must be read during its retransmission. In any case, if a particular station sends a message, the first message received will be then that message. In fact, all other messages which are on the ring will be removed by other stations without arriving at that particular station.

The places MAR and $M?$ and the transitions MFW and $t46$ (Figure A1) define the procedure which is invoked at each message arrival and departure. Each time a message is sent out by a station a mark is put into place $M?$. A message arrival at a station is modeled by putting a mark into place MAR. In this case there are two possibilities:

a) $M?$ is empty, i.e., the message was put on the ring by another station. The time duration of the transition MFW is denoted by the variable $(pfw + d)$ where pfw is the delay of the messages in passing a station and d is the propagation delay on the optical fiber ring . Firing MFW will put a mark in the place MAR of the next station.

b) $M?$ is not empty, i.e., the message was put on the ring by this station. The transition $t46$ fires and a mark is removed from the place MAR (and hence the message from the ring) as well as from the place $M?$. The place $M?$ becomes empty when all messages which were put on the ring during a cycle have been removed by that station. Note that the message passing time of a station is included in the model.

In case a) we do not distinguish between whether the message is addressed to that station or not. Because the message will be retransmitted in any case whether it is addressed to that station or not. Therefore, the delay time for a station is the same in both cases.

2.3 Station Receiving a Token

When the station receives the token, either there is nothing to transmit, i.e., both buffers for the synchronous and asynchronous traffic are empty, then the token is passed to the next station, or there is something to transmit. The latter case will be discussed in section 2.6. In this section we show how to decide whether or not there is a message to transmit.

In Figure 2 the token arrival is modeled by having a mark in place TAR. By firing the transition $t7$ we create two marks, one in place $T1$ and the other in place $T2$. These two places determine whether the two buffers ASB and SB, given in Figures 3 and 4, are empty or not. This is done in the following way: The transitions $t8$, $t9$ and $t10$ have higher priorities than the transitions $NPDU$ and $YPDU$. If the buffers SB and ASB are empty, the transition $t8$ fires. The mark in place $T1$ is eliminated, $PDU?$ remains empty and $NPDU$ fires. As a result the place $P42$ gets a mark. The transition TFW

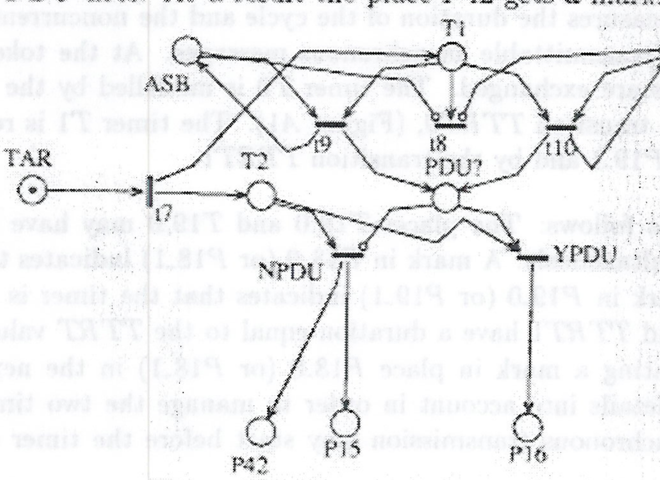

Figure 2: Token Arrival at a Station.

fires with a time duration denoted by tfw (Figure $A1$). The transition TD between the places $P17$ and TAR of the next station represents the propagation delay d of the optical fiber ring (Figure $A1$).

If a station has synchronous or asynchronous messages to transmit, i.e., if one or both of the buffers (SB and ASB) are not empty, then either the transitions $t9$ or $t10$ or both may fire. If only one of them fires, a mark will appear in $PDU?$ which will trigger the transition $YPDU$ to fire and the mark will join the place $P16$ which will indicate that the transmission of the messages may start.

2.4 TRT, THT and TTRT Timers

For the message transmission we need to know if the token is early or late. For this reason each time a token arrives at a station, the TRT timer is restarted. If the token for the next cycle arrives when the TRT timer is expired, then the token is late. Otherwise, it is early. If the token is early, the value of the THT is set equal to the value of the TRT timer. The timer THT starts in the beginning of the transmission of asynchronous messages.

One of the difficult problems to solve in our model is how to copy the TRT value into the THT value. Instead of copying the value of the timer we continue to use the same timer. The problem here is that we cannot stop the timer until the asynchronous transmission begins. We solve this problem by counting the messages which we can transmit during the remaining time before the timer expires. However, those messages are not transmitted immediately instead they are kept in a place for a later transmission which takes place after the end of the synchronous traffic transmission.

Another problem is the reinitialization of the timer for the measurement of the next cycle. Unfortunately we cannot reinitialize the timer, if we continue to use the same timer to count the number of asynchronous messages which will be allowed for transmission. Therefore, we use two separate timers denoted by $T0$ and $T1$. The model contains the current timer which measures the duration of the cycle and the noncurrent timer which counts the number of transmittable asynchronous messages. At the token arrival the roles of the two timers are exchanged. The timer $T0$ is modelled by the places $P18\_0$ and $P19\_0$ and by the transiton $TTRT0$, (Figure $A1$). The timer $T1$ is represented by the places $P18\_1$ and $P19\_1$ and by the transition $TTRT1$.

The timers work as follows: The places $T18\_0$ and $T19\_0$ may have have at most one mark but not simultaneously. A mark in $P18\_0$ (or $P18\_1$) indicates that the timer is expired where a mark in $P19\_0$ (or $P19\_1$) indicates that the timer is expired. The transitions $TTRT0$ and $TTRT1$ have a duration equal to the $TTRT$ value. The timer is reinitialized by creating a mark in place $P18\_0$ (or $P18\_1$) in the next cycle. We need to take several details into account in order to manage the two timers correctly. For example, the asynchronous transmission may start before the timer expires. This has no effect because the counter keeps counting during the transmission of the already counted messages. On the other hand, it may happen that the asynchronous buffer becomes empty during the transmission of the asynchronous messages. In this case, the transmission stops. The solutions for these problems are given in the next sections where we show how to switch between the timers, how to count the asynchronous messages and how to transmit the messages.

2.5 The Switch

The message transmission procedure is activated when a mark arrives at place $P16$ as shown in Figure 2. At this moment we need to exchange the functions of the two timers in order to start the measurement of the next cycle duration and need to decide which of the two timers to use for counting the number of asynchronous messages allowed to be transmitted. These are realized through the switch shown in Figure 3.

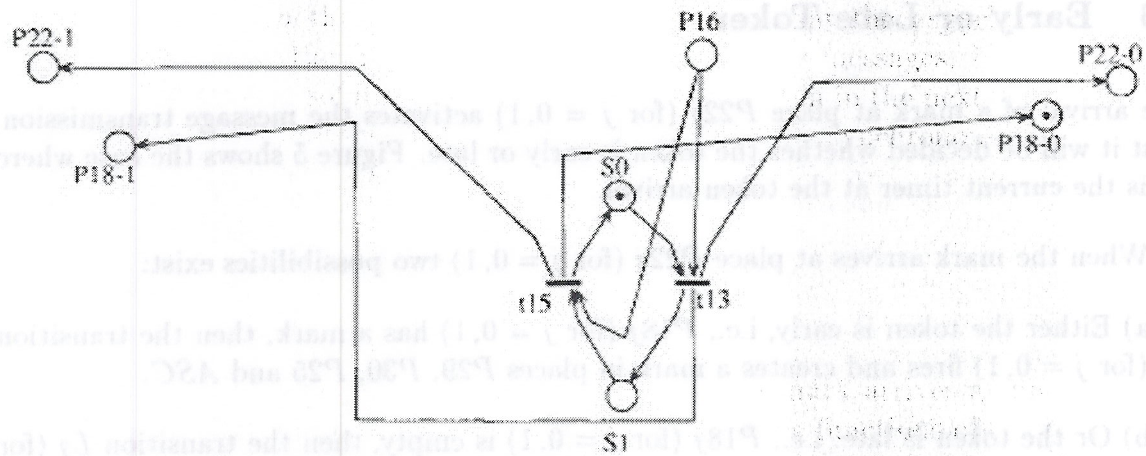

Figure 3: The Switch.

There is only one mark circulating between $S0$ and $S1$. If the mark is at place Si (for $i = 0,1$), this means that Ti is the current timer. Suppose Sj (for $j = 0,1$) has a mark and a switch takes place, then two events occur:

a) A mark arrives at place $P18j$ (for $j = 0,1$) and activates the timer Tj (for $j = 0,1$) which starts to measure the duration of the next cycle.

b) A mark arrives at place $P22j$ (for $j = 0,1$) and activates the transmission of the messages.

The switch of the timers is activated at the token arrival even if there is nothing to transmit. By firing the transition $NPDU$ a mark arrives at place $P15$ which activates the switch only when there is nothing to transmit as shown in Figure 4.

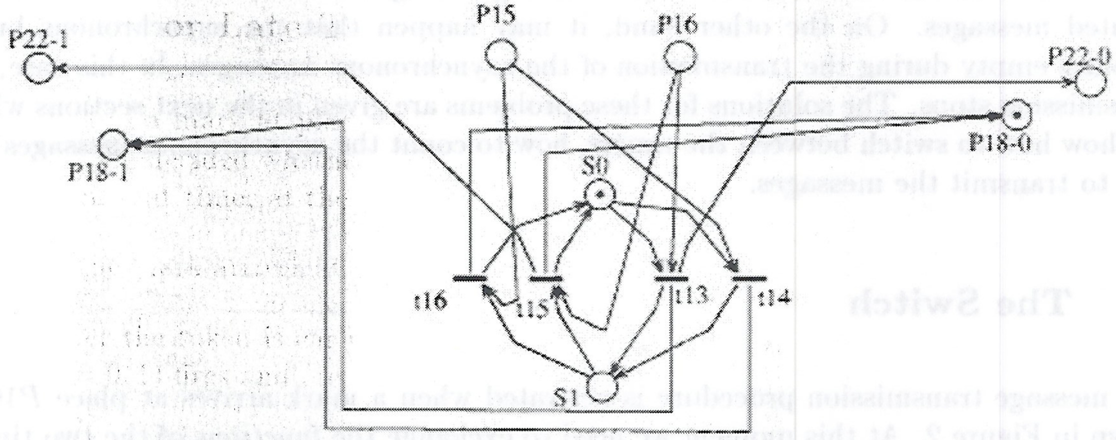

Figure 4: The Complete Switch.

2.6 Early or Late Token

The arrival of a mark at place $P22j$ (for $j = 0, 1$) activates the message transmission. First it will be decided whether the token is early or late. Figure 5 shows the case where $T0$ is the current timer at the token arrival.

When the mark arrives at place $P22j$ (for $j = 0, 1$) two possibilities exist:

a) Either the token is early, i.e., $P18j$ (for $j = 0, 1$) has a mark, then the transition Ej (for $j = 0, 1$) fires and creates a mark in places $P29$, $P30$, $P25$ and ASC.

b) Or the token is late, i.e., $P18j$ (for $j = 0, 1$) is empty, then the transition Lj (for $j = 0, 1$) fires and creates a mark in places $P29$, $P30$ and L.

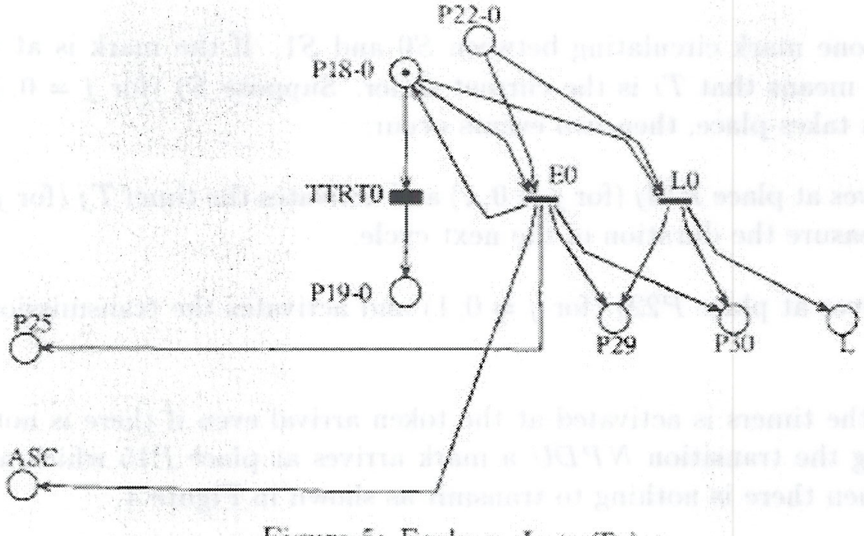

Figure 5: Early or Late Token.

2.7 The Asynchronous Message Counter

As the protocol [1] specifies the asynchronous traffic can be transmitted only if the token is early. The transmission of an asyncronous frame will continue even if time expires. Therefore, if the token is early, a mark is put in place ASC which contains a mark for each asynchronous message that can be transmitted. The place $P25$ and the transitions Ctj (for $j = 0, 1$) in Figure 6 count the number of the asynchronous messages which are allowed to be transmitted.

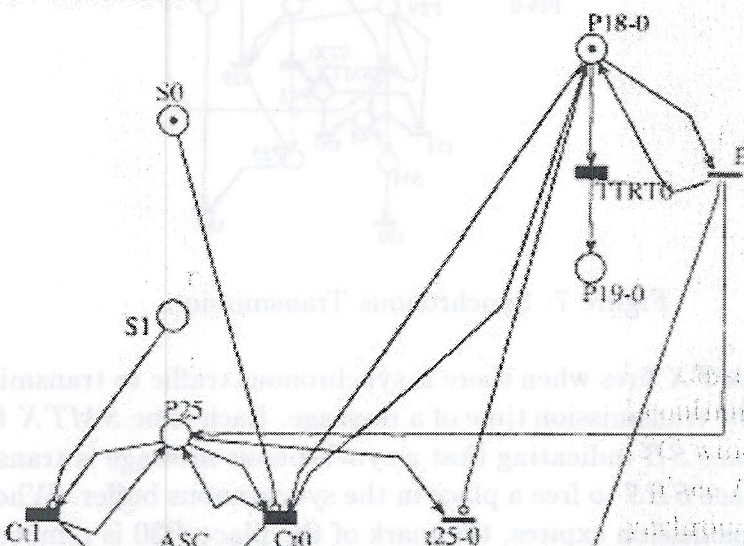

Figure 6: Asynchronous Message Counter.

The transition Ctj (for $j = 0, 1$) is used if Tj (for $j = 0, 1$) is the current timer at the token arrival. Figure 6 shows the case where $T0$ is the current timer. The arrival of a mark in place $P25$ activates the counting. The time for transitions Ctj (for $j = 0, 1$) is equal to the transmission time of a message and is denoted by mtx. Therefore, as long as the timer does not expire, a mark is added to ASC every mtx milliseconds. This means that if THT is between $[x \quad mtx]$ and $[(x+1) \quad mtx]$ at most $[x+1]$ (for $x = 0, 1, ...$) asynchronous messages can be transmitted. When the timer Tj (for $j = 0, 1$) expires, the transitions $t25j$ fire and the mark is removed from $P25$ so that the counting will stop.

2.8 Synchronous Transmission

When the transmission procedure is activated in both cases (early or late token arrival), a mark is put into places $P29$ and $P30$ The mark in place $P30$ activates the timer which measures the synchronous transmission time. This timer is modeled by the places $P30$ and $P33$, and the transition $STXt$ as shown in Figure 7.

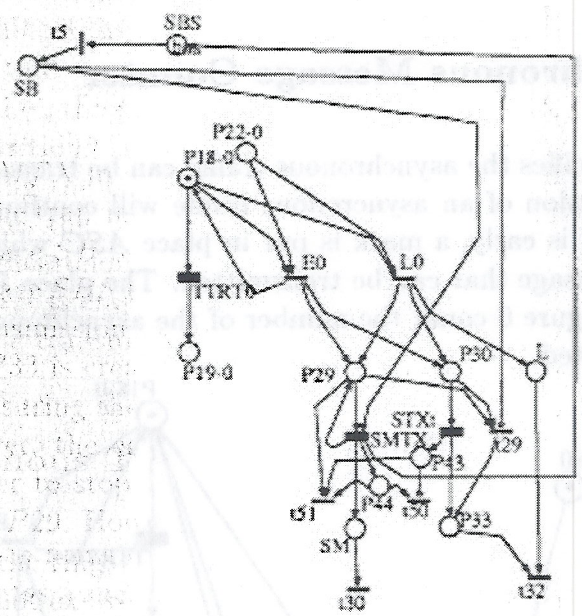

Figure 7: Synchronous Transmission.

The transition $SMTX$ fires when there is synchronous traffic to transmit. The time of $SMTX$ is equal to the transmission time of a message. Each time $SMTX$ fires, a mark is removed from the place SB indicating that a synchronous message is transmitted, and a mark is created in place SBS to free a place in the synchronous buffer. When the timer of the synchronous transmission expires, the mark of the place $P30$ is removed and a mark in places $P43$ and $P33$ is created. The place $P43$ helps to solve the following problem: when the timer measuring the synchronous transmission expires, then the transmission should stop. However, the entire frame which was on the ring should continue to be transmitted. In order to stop the transmission of the synchronous frames we remove a mark from the place $P29$. However, this would also stop the transmission of the message which is already on the ring. Therefore, after each transmission we add a mark to place $P44$. If the timer is not expired, i.e., the place $P43$ is empty, then the transition $t50$ fires under the condition that the place $P44$ has a mark, i.e., the transmission is continued. If the timer expires, i.e., the place $P43$ contains a mark, the arrival of the mark at place $P44$ triggers the transition $t51$ to fire which will remove marks from the places $P43$, $P44$, and $P29$, i.e., stopping the transmission of the synchronous messages. The arrival of a mark at $P33$ activates the passing of the token to the next station if it is late or the beginning of the asynchronous message transmission if it is early. Furthermore, if the synchronous buffer SB is empty, the transition $t29$ fires which removes the marks from $P29$ and $P30$, i.e., the transmission of the synchronous messages is stopped.

2.9 Asynchronous Transmission

If the token is late, the firing of the transition L_j dependent on the current timer Tj at the token arrival creates a mark in the place L as shown in Figure 8.

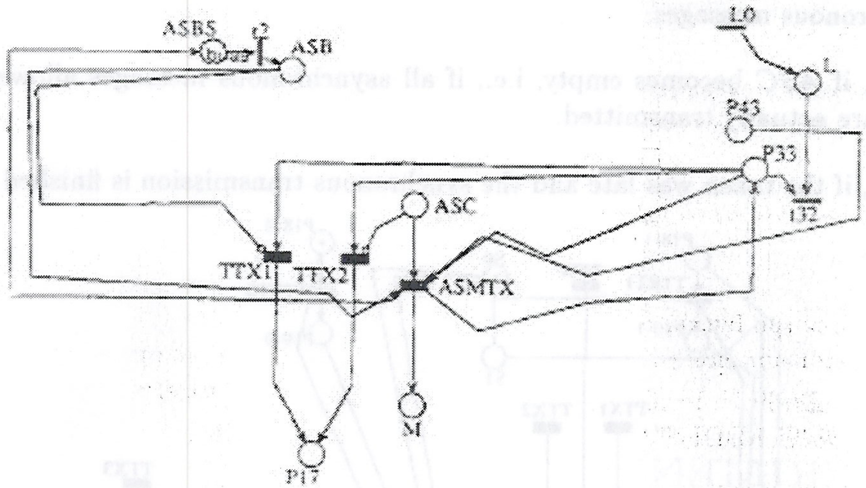

Figure 8: Asynchronous Transmission.

A mark in place L indicates that the token is late. If the synchronous transmission time expires or the synchronous buffer SB is empty, and the token is late, the transition $t32$ fires and as a result the token passing procedure is called. As mentioned before the asynchronous transmission is not allowed due to the late token. No mark in the place L indicates that the token is early. In this case the transition $ASMTX$ may fire under the condition that the place ASB is not empty, i.e., there are asynchronous messages to transmit. Since the token is early, the place ASC will contain at least a mark. The firing time of the transition $ASMTX$ is equal to the transmission time of a message. When $ASMTX$ fires, i.e., the message is transmitted, a mark is put into place M as in the case of the synchronous transmission. On the other hand, a mark is created at place $ASBS$ to free a place in the asynchronous buffer. The inhibitor arc between the place $P43$ and the transition $ASMTX$ indicates that the asynchronous transmission can only start when the last synchronous message has been transmitted completely. The transition $TTX1$ may only fire when the asynchronous buffer (place ASB) is empty. The firing time of $TTX1$ is equal to the token transmission time. Removing a mark from $P33$ indicates that the asynchronous transmission is stopped. Creating a mark in $P17$ means that the token passing procedure is activated. The transition $TTX2$ fires when the place ASC is empty, i.e., all asynchronous messages have been transmitted within the specified transmission time. The time for $TTX2$ is equal to the token transmission time. Its firing creates a mark in $P17$ and removes a mark from $P33$.

2.10 Token Passing

As shown in Figure 9 the token is passed to the next station by creating a mark in place $P17$ which is realized by the following transitions:

a) TFW, if there is nothing to transmit when the token arrives.

b) $TTX1$, if the asynchronous buffer ASB becomes empty during the transmission of the asynchronous messages.

c) $TTX2$, if ASC becomes empty, i.e., if all asynchronous messages allowed to be transmitted are actually transmitted.

d) $TTX3$, if the token was late and the synchronous transmission is finished.

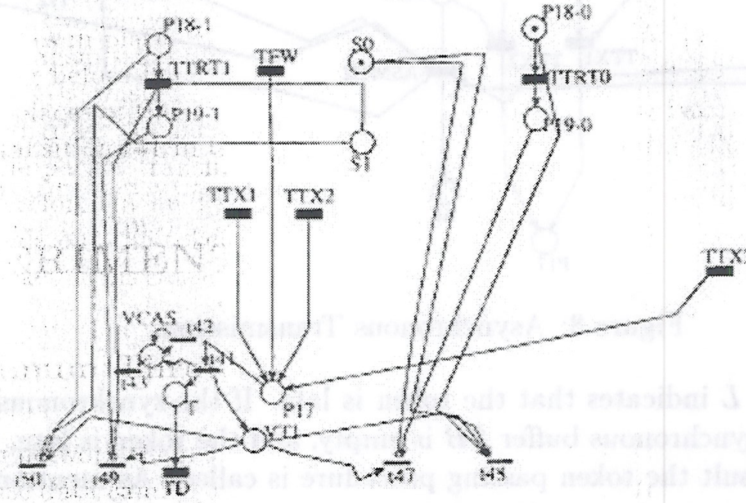

Figure 9: Token Passing.

A mark resides in place $P17$ for a time period equal to the propagation time on the optical fiber ring before being sent to the place TAR of the next station. In order to reinitialize the places of the station waiting for the next token arrival, a mark is created by firing the transition $t41$ in the places $VCAS$, T and VTI during this period.

3 EXPERIMENTS

3.1 Maximum Throughput

In a token ring network the maximum throughput is strictly less than the capacity of the network because data cannot be transmitted during the token transmission, propagation and identification.

Let A and B two successive stations. It is clear that the longer the time is between the end of the transmission by A and the beginning of the transmission by B, the more network capacity is wasted. In low speed networks the wasted part of the network capacity is negligible. In fact, in those networks the propagation delay is negligible according to the packet transmission time. However, this is not the case in high speed networks. Therefore, in an FDDI network a station may transmit several packets in a cycle and sends out the token immediately without waiting for the return of the last transmitted packet. This causes an increase of the maximum throughput as demonstrated below.

The *maximum throughput* is computed by

$$\gamma_{max} = U \ C \tag{1}$$

where C is the network capacity (100 Mbps) and

$$U = \frac{Ttp}{Ttp + Tpj} \tag{2}$$

with

Ttp is the *mean total time* during which all stations have transmitted packets in a particular cycle and

$$Tpj = ttx \ + \ delay \ + \ tdx \tag{3}$$

where ttx is the token transmission time, $delay$ is the propagation delay between two station and tdx is the token detection time.

Equation (2) can also be expressed as

$$U = \frac{1}{1+a} \tag{4}$$

with

$$a = \frac{Ttj}{Ttp} \tag{5}$$

where Ttj is the time used for token passing during a particular cycle and is computed from:

$$Ttj = N \ delay + N \ ttx + N \ tdx \tag{6}$$

where N is the total number of stations.

Given that the FDDI is conceived for a maximum distance of 200 km [1], it is possible that the parameter U (equations (2) and (4)) will be much smaller than 1 for certain implementations of the network. That is why we are interested to study the maximum throughput for various cases.

We analyze an FDDI model with $N = 4$ symmetrical stations and with the parameters given in Table 1. In Table 1 $TTRT$ is the target token rotation time, $ttxs$ is the synchronous transmission time (a percentage of TTRT), mtx is the message transmission time and pfw is the packet forward time.

$TTRT$	0.36 *msec.*
ttx	0.0011 *msec.*
tdx	0.0008 *msec.*
$ttxs$	0.04 *msec.*
mtx	0.01 *msec.*
pfw	0.0006 *msec*
infinite buffers	

Table 1: Case *A*

Figure 10 shows the value for U dependent on the the propagation delay between two adjacent stations.

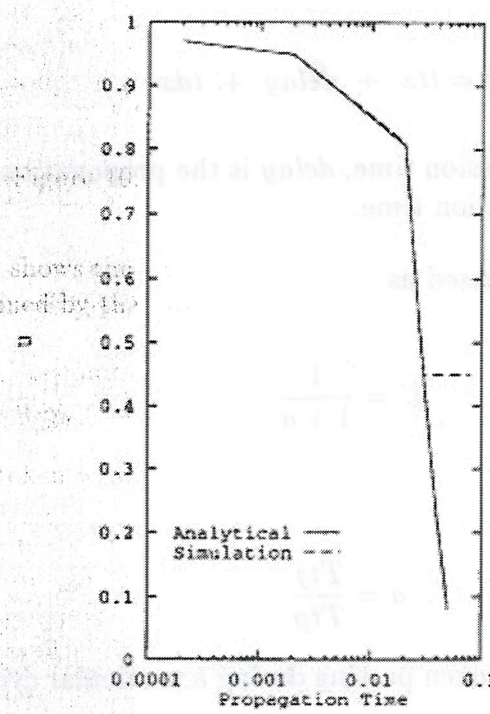

Figure 10: U dependent on Propagation Time.

The dotted line shows the simulation results whereas the solid line contains the analytical results obtained by the formula:

$$\gamma_{max} = \frac{TTRT + ttxs - Tpj}{TTRT + ttxs + Tpj / N} C. \quad (7)$$

This formula is obtained by the same method as in [3]. The difference between their formula and ours is the fact that we take the synchronous transmission time into account (they allow only the asyachronous traffic in their model), and we neglect the residual time for the transmission of a packet after the expiration of the THT, (which is 3 % of the TTRT). In comparing the analytical with the simulation results we see that they match exactly up to a certain point beyond which the throughput becomes constant. In fact, at this point $TTRT$ is equal to the synchronous transmission time plus the token passing time. If we increase the propagation delay, the value of TTRT becomes insufficient for the transmission of the synchronous messages and the network does not function correctly. In this case the actual network is reinitialized. In the model, however, only the synchronous traffic is transmitted. That is why the throughput becomes constant in Figure 10, but obviously the mean cycle time becomes greater than the $TTRT$ causing the deviation of the analytical results from the simulation.

Figure 11 shows the simulation results for the same set of parameters given above, (Case A), and for the set of parameters (Case B) given in Table 2.

$TTRT$	0.20 $msec$
ttx	0.0011 $msec.$
tdx	0.0008 $msec.$
$ttxs$	0.008 $msec.$
mtx	0.0002 $msec.$
pfw	0.0006 $msec$
infinite buffers	

Table 2: Case B

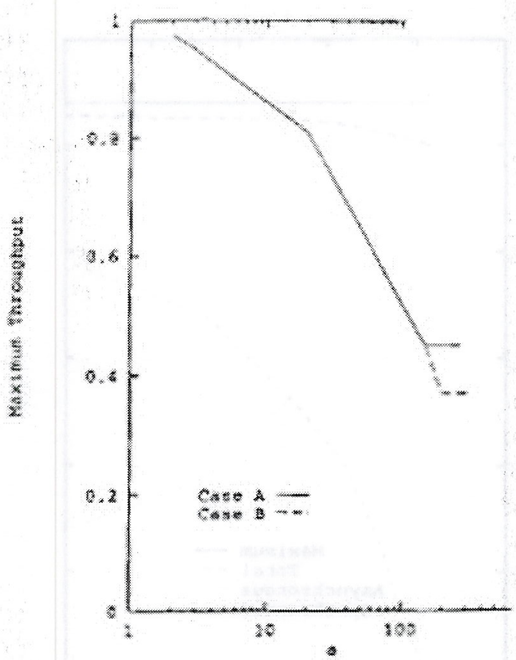

Figure 11: Throughput γ dependent on a (in fact 100 times a).

Note that the synchronous transmission time in case B is less than in case A.

We conclude that the parameter α must be taken into account in the implementation of the FDDI network, since the network capacity may be misused for certain cases considered in the standard [1]. In fact, the standard [1] suggests a maximum ring length of 200 km which gives a total delay of 66.6 msec. Note that the delay time in the curves in Figure 10 and 11 does not go beyond this value.

3.2 The Influence of the TTRT

Here we show the influence of TTRT on the capacity allocation between asynchronous and synchronous traffic. The network is analyzed in heavy load for both asynchronous and synchronous traffic using the parameters of case C.

ttx	0.0011 $msec.$
tdx	0.0008 $msec.$
$ttxs$	0.04 $msec.$
mtx	0.01 $msec.$
pfw	0.0006 $msec.$
$delay$	0.0015 $msec.$
infinite buffers	

Table 3: Case C

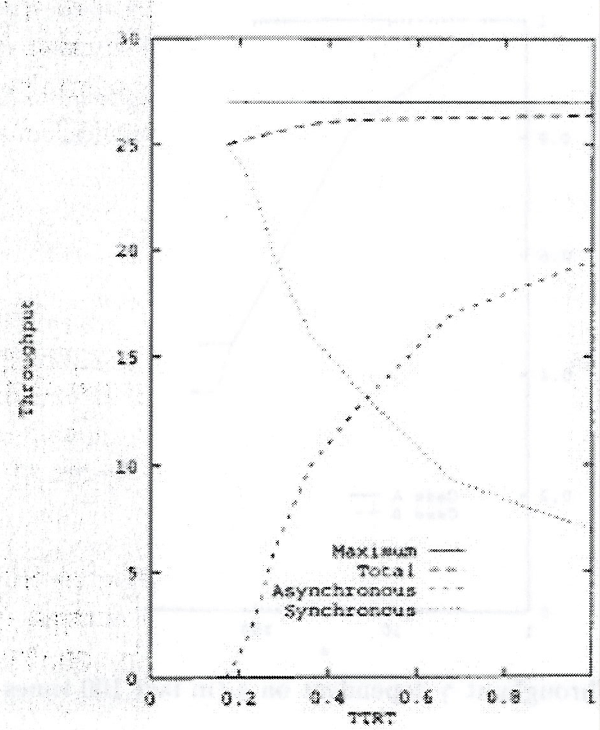

Figure 12: Total Traffic

The curves in Figure 12 depict the synchronous, asynchronous, total traffic and the capacity for a station (a quarter of the total capacity due to the symmetry of the network since there are $N = 4$ total number of stations).

The larger the TTRT the more asynchronous traffic absorbs the capacity of the network. On the other hand, the total traffic is increasing with the TTRT which is a consequence of the fact which we discussed in section 3.1.

4 CONCLUSIONS

The Petri net model of the FDDI protocol allowed us to compute the performance measures of the network. We conclude that the access protocol effectively insures the percentage of the synchronous traffic bandwidth. Figure 12 shows the distribution of the bandwidth in terms of the TTRT parameters for a heavy load case. Similarly, the maximum throughput is studied for various load configurations. From our model, it is easy to see that the capacity is not well utilized for certain network configurations allowed by the standard [1]. In fact, we have shown in section 3.1 a case where the maximum throughput was less than 40% of the network capacity. This observation leads to the fact that the slotted protocols such as the DQDB protocol [10] could be more adequate for high speed networks than the FDDI protocol. Finally, it is clear that if the capacity on the fiber increases, the parameter a introduced in section 3.1 will increase accordingly and the network throughput will decrease. This also shows that the FDDI access protocol is not well-suited for very high speed networks of the future.

REFERENCES

[1] *Information processing systems-Fibre Distributed Data Interface (FDDI)*, International standard, ISO 9314-2. 1989.

[2] G. Chiola; "GreatSPN 1.5 Software Architecture", *Proc. of the Fifth Int. Conf. Modeling Techniques and Tools for Computer Performance Evaluation*, Torino, Italy, February 1991, pp. 117-133.

[3] D. Dykeman and W. Bux; "Analysis and Tuning of the FDDI Media Access Control Protocol", *IEEE Journal on Selected Areas in Communications*, Vol. 6, No. 6, pp. 997-1010, July 1988.

[4] W. L. Genter and K. S. Vastola, "Delay Analysis of the FDDI Synchronous Data Class", *Proc. of INFOCOM 90*, June 1990, pp. 766-773.

[5] R. M. Grow, "A Timed Token Protocol for Local Area Networks", *Electro '82*, Token Access Protocols, May 1982.

[6] M. J. Johnson, "Reliability Mechanisms of the FDDI High Bandwidth Token Ring Protocol", *Computer Networks and ISDN Journal*, Vol. 11, pp. 121-131, February 1986.

[7] M. J. Johnson, "Proof that Timing Requirements of the FDDI Token Ring Protocol are Satisfied", *IEEE Transactions on Communications*, Vol 35, No. 6, pp. 620-625, June 1987.

[8] D. Karvelas and A. Leon Garcia, "Performance Analysis of the Medium Access Control Protocol of the FDDI Token Ring Network", *Proc. of the GLOBECOM 88*, Nov. 1988, pp. 1119-1123.

[9] R. LaMaire and E. Spiegel, "FDDI Performance Analysis: Delay Approximations", *GLOBECOM'90*, December 1990.

[10] J. Liebeherr, I. F. Akyildiz and A. N. Tantawi, "Modeling of the DQDB Access Protocol with Stochastic Petri Nets", *Proc. of the Fourth Int. Workshop on MAN, Captiva Island, Florida*, November 1990.

[11] G. M. Lundy and I. F. Akyildiz, "A Formal Model of the FDDI Network Protocol", *Proc. of the EFOC/LAN Int. Conference, London, England*, June 1991.

[12] J. Pang and F. A. Tobagi, "Throughput Analysis of a Timer Controlled Token Passing Protocol under Heavy Load", *IEEE Transactions on Communications*, Vol. COM-37, July 1989, pp. 694-702.

[13] F. E. Ross, "An Overview of FDDI: The Fiber Distributed Data Interface", *IEEE Journal on Selected Areas in Communications*, Vol. 7, No. 7, pp. 1043-1051, September 1989.

[14] A. Schill and M. Zieher, "Performance Analysis of the FDDI 100 Mbitps Optical Token Ring", *Int. Conf. on High Speed Local Area Networks*, North Holland, 1987, pp. 53-74.

[15] K. C. Sevcik and M. J. Johnson, "Cycle Time Properties of the FDDI Token Ring Protocol", *IEEE Transactions on Software Engineering*, Vol. SE-13, No.3, pp. 376-385, March 1987.

[16] M. Tangemann and K. Sauer, "Performance Analysis of the FDDI Media Access Control Protocol", *Fourth Int. Conference on Data Communication Systems and their Performance*, North Holland, June 1990, pp. 32-43.

[17] J. N. Ulm, "A Timed Token Ring Local Area Network and its Performance Characteristics", *Proc. of the 7th Conference on Local Computer Networks*, IEEE, pp. 50-56, February 1982.

[18] P. N. Werahara and A. P. Jayasumana, "Throughput Evaluation of an Asymmetrical FDDI Token Ring Network with Multiple Classes of Traffic", *Proc. of INFOCOM 90*, June 1990, pp. 997-1004.

APPENDIX: THE COMPLETE MODEL OF AN FDDI STATION