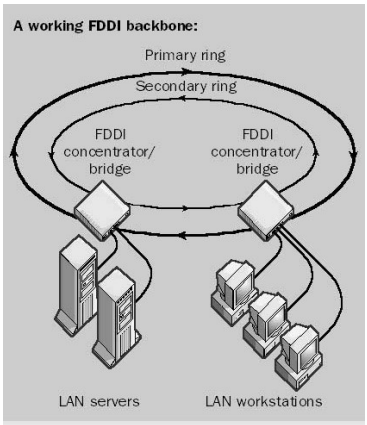
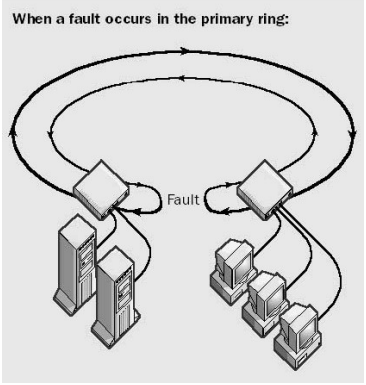
Lecture 9

Fault Tolerance

FDDI is also built for fault tolerance. It is actually two networks, one going counter-clockwise and the other clockwise (as usually drawn). Normally only the primary links, i.e. the counter-clockwise ones, are used, but if a link or even a node goes down, the secondary links will maintain the operation of the network.
For example, consider a four-node network, with nodes A, B, C and D (in counter-clockwise order). Say the link from D to A is accidentally cut. Then the secondaries will automatically activate, so that the network makes "U-turns" at A and D: After a frame goes from B to C to D on the primary link, for instance, it will next go back to C from D on the secondary link, etc. In this manner, we retain the ring topology after the accidental break.

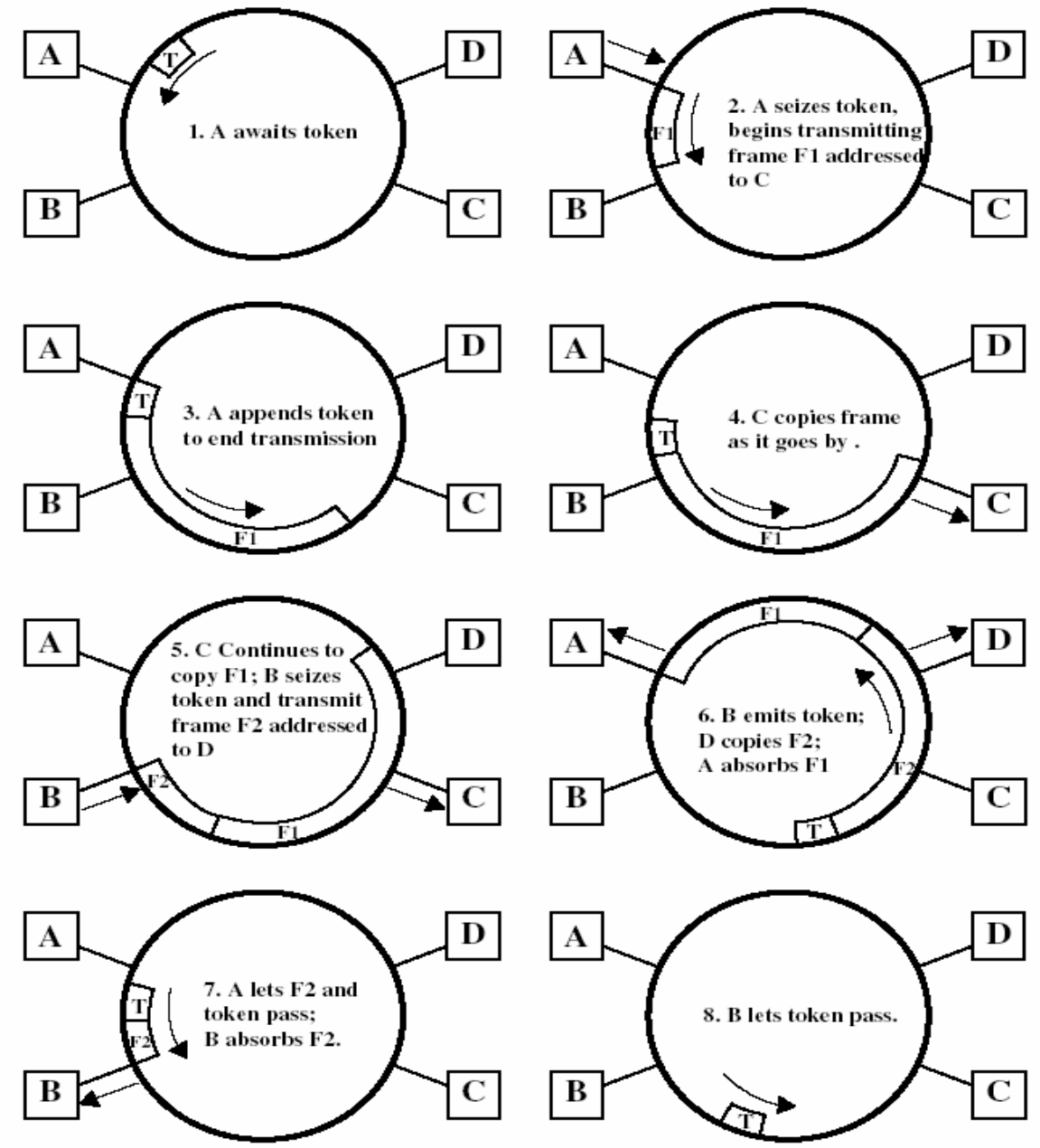




FDDI MAC Protocol

In FDDI, a station waiting for a token seizes the token by grabbing the token as soon as a token passes by. After capturing the token, the station begins transmitting one or more data frames. The 802.5 technique of flipping a bit to convert a token to frame is impractical, as FDDI was designed for considerable high data rate.

FDDI by default supports Early Token Release. Below is the operation of FDDI ring.



Capacity Allocation

The priority scheme used in IEEE 802.5 will not work with FDDI, as a station will often issue a token before its own transmitted frame returns. FDDI capacity allocation scheme seeks to accommodate a mixture of stream and bursty traffic.

Each station is allocated a portion of the total capacity (may be zero); the frames that it transfers are referred to as synchronous frames. Any capacity that is not allocated or not used is available for the transmission of additional asynchronous frames. The allocation must be as following:

DMax + FMax + Token Time + SAi <= TTRT

Where

TTRT = target token rotation time
SA_i = synchronous allocation for station i
DMax = propagation time for one complete circuit of the ring
FMax = time required to transmit a maximum length frame (4500 octets)
Token Time = time required to transmit a token

If a token arrives early, the station can set Token-holding timer (THT) equal to the remaining time. After transmitting synchronous frames, the station can then transmit asynchronous frames up to the time in THT. If the token arrives late, the station will set its Late Counter (LC) to 1 and will only transmit synchronous frames. This scheme is designed to assure that the time between successive sightings of a token is on the order of TTRT or less.

When a station wishes to enter an extended dialogue, it may gain control of all the unallocated (asynchronous) capacity on the ring by using a restricted token. The station captures a non-restricted token, transmits the first frame of the dialogue to the destination station, then issues a restricted token. Only the station that received the last frame may transmit asynchronous frames using the restricted token. The two stations may then exchange data frames and restricted tokens for an extended period. The standard assumes that restricted transmission is predetermined not to violate the TTRT limitation.

• The priority scheme used in 802.5 does not work in FDDI, as a station issues a token before its own transmitted frame returns.
• FDDI uses a capacity allocation scheme which seeks to accommodate a mixture of stream and bursty traffic.
• FDDI defines two types of traffic: synchronous and asynchronous.
• Assume fixed length frame sizes.
• Rather than defining “capacity” as bps, define it as the number of frames that can be transmitted in a given time period.
• Define the total amount of frames in a given time period as the “synchronous traffic”.
• Each station is allocated a certain percentage of the synchronous traffic. “Synchronous Allocation” or “SA”.
• Also define the rate at which the token needs to circulate around the ring as TTRT - “Target Token Rotation Time”.
• Thus, each station i is allocated a specific SA i values such that

$$\sum SA_i \leq \text{TTRT}$$

• However, we have to account for the time to actually transmit the token, propagation time and the time to get at least one frame around the ring:
Total Synchronous Allocation = $\sum SA_i$ + Time to transmit one token + Propagation time around the ring + Time to transmit a frame

Total Synchronous Allocation ≤ TTRT

• If this summation is less than TTRT, then all time left is considered “asynchronous allocation”.
Asynchronous allocation = TTRT - Total Synchronous Allocation
• Operation:
– Each station holds the following variables

TTRT: Fixed constant, same for all stations
SA_i : pre-assigned allocation amount
TRT: Token Rotation Timer - Amount of time before the TTRT time expires
THT: Token Holding Time - Amount of extra time left
LC: Late Counter - Either 0,1, or 2, number of TTRT cycles that have elapsed since last token received.

– TRT is a counter. It continually decrements, unless otherwise stopped, or reset.
– Initialize: TRT ← TTRT; LC ← 0
– While waiting for a token, the TRT continues to decrement. If it hits 0, then it increments the LC from 0 to 1, resets TRT, then continues waiting for token. If LC gets increment to 2, then the token is considered lost
– If it receives a token, and the LC is zero, then TRT represents “extra time”.
THT ← TRT, TRT ← TTRT, enable TRT. Then, the station sends synchronous frames for a time SA_i. After transmitting synchronous frames, or if there were no synchronous frames to transmit, THT is enabled. The station can transmit asynchronous frames as long as THT > 0.
– If it receives a token and the LC is 1, then LC ← 0, TRT continues to decrement. The station can only transmit synchronous frames for a time SA i.

FDDI Physical Layer Specification

Transmission Medium	Optical Fiber	Twisted Pair
Data rate (Mbps)	100	100
Signaling Technique	4B/5B/NRZI	MLT-3
Maximum number of repeaters	100	100
Maximum length between repeaters	2 km	100 m

Some key differences between Token Ring and FDDI are

FDDI	**Token Ring 802.5**
Fiber Optics	Twisted Pair
100Mb/s	16Mb/s
Distributed Clocking	Centralized Clocking
Timed Token Rotation	Priority and Reservation Bits
New Token after Transmit	New Token after Receive