

Schichten in Echtzeitsystemen

- Die Nachrichtenübertragungszeit setzt sich aus folgenden Komponenten zusammen:
 - Umsetzung der Protokolle der einzelnen Schichten durch den Sender
 - Wartezeit auf Medienzugang
 - Übertragungszeit auf Medium
 - Entpacken der Nachricht in den einzelnen Schichten durch den Empfänger
- ⇒ Jede zu durchlaufende Schicht verlängert die Übertragungszeit und vergrößert die zu sendenden Daten.
- ⇒ in Echtzeitsystemen wird die Anzahl der Schichten zumeist reduziert auf:
- Anwendungsschicht
 - Sicherungsschicht
 - Physikalische Schicht



Echtzeitfähige Kommunikation

Medienzugriffsverfahren

Problemstellung

- Zugriffsverfahren regeln die Vergabe des Kommunikationsmediums an die einzelnen Einheiten.
- Das Kommunikationsmedium kann in den meisten Fällen nur exklusiv genutzt werden, Kollisionen müssen zumindest erkannt werden um Verfälschungen zu verhindern.
- Zugriffsverfahren können dabei in unterschiedliche Klassen aufgeteilt werden:
 - Erkennen von Kollisionen, Beispiel: CSMA/CD
 - Vermeiden von Kollisionen, Beispiel: CSMA/CA
 - Ausschluss von Kollisionen, Beispiel: token-basiert, TDMA



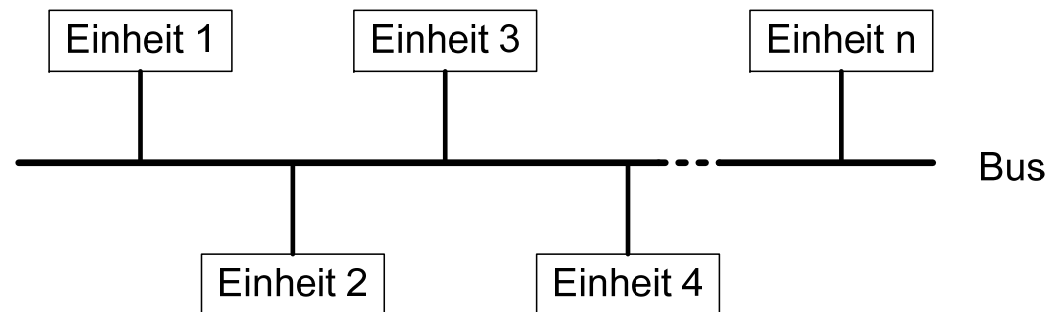
Echtzeitfähige Kommunikation

Carrier Sense Multiple Access/Collision Detection (CSMA/CD)

Vertreter: Ethernet (nicht echtzeitfähig!)

CSMA/CD

- CSMA/CD: Carrier Sense Multiple Access - Collision Detection
 - alle am Bus angeschlossenen Einheiten können die aktuell versendeten Daten lesen (**Carrier Sense**).
 - mehrere Einheiten dürfen Daten auf den Bus schreiben (**Multiple Access**).



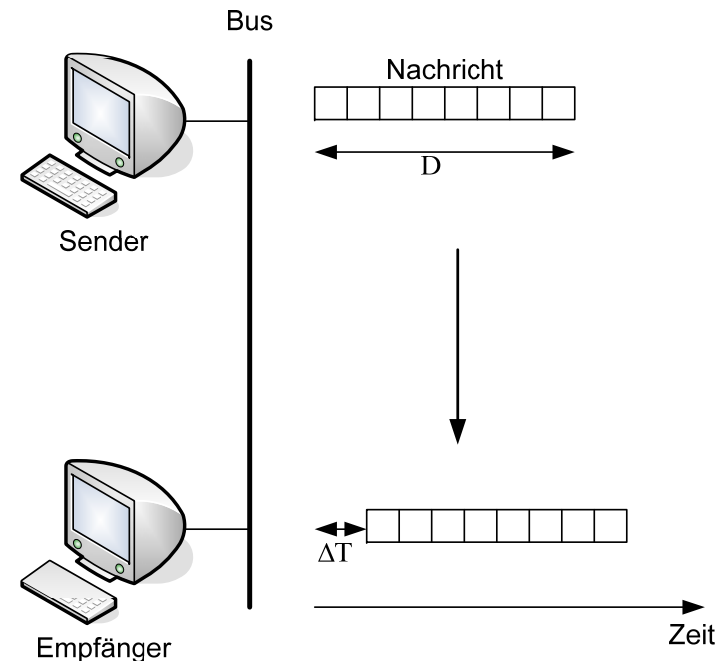
- Während der Übertragung überprüft der sendende Knoten gleichzeitig das Resultat auf dem Bus, ergibt sich eine Abweichung, so wird eine Kollision angenommen (**Collision Detection**)

CSMA/CD: Ablauf

- Beschrieben wird im Folgenden das 1-persistente CSMA/CD- Verfahren (Spezifikation in der Norm IEEE 802.3)
- Ablauf zum Senden eines Paketes:
 1. Test, ob Leitung frei ist (**carrier sense**)
 2. Falls Leitung für die Zeitdauer eines IFS (**inter frame spacing**) frei ist, wird die Übertragung gestartet, ansonsten Fortfahren mit Schritt 5.
 3. Übertragung der Daten inklusive Überwachung der Leitung. Im Fall einer Kollision: senden eines **JAM**-Signals, fortfahren mit Schritt 5.
 4. Übertragung erfolgreich beendet: Benachrichtige höhere Schicht, Beendigung
 5. Warten bis Leitung frei ist
 6. Sobald Leitung frei: weitere zufälliges Warten (z.B. **Backoff-Verfahren**) und Neustarten mit Schritt 1, falls maximale Sendeversuchszahl noch nicht erreicht.
 7. Maximale Anzahl an Sendeversuchen erreicht: Fehlermeldung an höhere Schicht.

Kollisionen

- Um Kollisionen rechtzeitig zu erkennen muss die Signallaufzeit ΔT deutlich kleiner als die Nachrichtenübertragungsdauer D sein.
- Das Störsignal (JAM) wird geschickt um alle anderen Nachrichten auf die Kollision aufmerksam zu machen \Rightarrow Verkürzung der Zeit zur Kollisionserkennung
- Würden die Rechner nach einer Kollision nicht eine zufällige Zeit warten, käme es sofort zu einer erneuten Kollision.
- Lösung im Ethernet: Die Sender wählen eine zufällige Zahl d aus dem Intervall $[0 \dots 2^i]$, mit i = Anzahl der bisherigen Kollisionen (Backoff-Verfahren).
 \Rightarrow Mit ansteigendem i wird eine Kollision immer unwahrscheinlicher.
 \Rightarrow Bei $i = 16$ wird die Übertragung abgebrochen und ein Systemfehler vermutet.



TCP vs. UDP

- TCP (Transmission Control Protocol) ist ein zuverlässiges, verbindungsorientiertes Transportprotokoll:
 - Vor der Übertragung der Daten wird zunächst eine Verbindung zwischen Sender und Empfänger aufgebaut (Handshake).
 - Datenverluste werden erkannt und automatisch behoben durch Neuversenden des entsprechenden Datenpakets.
 - ⇒ Aufgrund von unvorhersehbaren Verzögerungen (Backoff-Verfahren) und hohem Overhead ist TCP nicht für den Einsatz in Echtzeitsystemen geeignet.
 - Weiteres Problem: Slow Start der Congestion Control Strategie von TCP/IP ⇒ zu Beginn der Übertragung wird nicht die volle Bandbreite ausgenutzt
- UDP (User Datagram Protocol) ist ein minimales, verbindungsloses Netzprotokoll:
 - Verwendung vor allem bei Anwendungen mit kleinen Datenpaketen (Overhead zum Verbindungsaufbau entfällt)
 - UDP ist nicht-zuverlässig: Pakete können verloren gehen und in unterschiedlicher Reihenfolge beim Empfänger ankommen.
 - ⇒ Einsatz in weichen Echtzeitsystemen, in denen der Verlust einzelner Nachrichten toleriert werden kann (z.B. Multimedia-Protokollen wie z.B. VoIP, VoD) möglich.

RTP, RTSP: Motivation

- Problem von UDP/IP in Multimediasystemen:
 - keine Möglichkeit zur Synchronisation
 - verschiedene Multimediasströme können kollidieren (z.B. in VoD)
 - Qualitätskontrolle ist wünschenswert

⇒ in Multimediasystemen werden zusätzliche Protokolle (RTP, RTCP) verwendet.
- Multimediaverbindung mit RTP/RTCP
 - Zur Übertragung der **Steuerungsnachrichten** (in der Regel nicht zeitkritisch) werden zuverlässige Protokolle eingesetzt (z.B. TCP/IP)
 - Zur **Datenübertragung** wird ein **RTP (Real-Time Transport Protocol)**-Kanal eingesetzt.
 - Jeder RTP-Kanal wird mit einem **RTCP (Real-Time Control Protocol)**-Kanal zur Überwachung der Qualität verknüpft.
 - RTP/RTCP setzen in der Regel auf UDP/IP auf und sind End-zu-End-Protokolle

RTP, RTCP

- RTP:
 - Multicasting
 - Bestimmung des Datenformats (PT)
 - Zeitgebend durch Zeitstempel, die Berechnung des Jitters wird dadurch möglich
 - Möglichkeit zur Ordnung der Pakete und zum Erkennen von verlorenen Paketen durch Sequenznummer

Byte 0				Byte 1				Byte 2				Byte 3																			
Bit 0	1	2	3	4	5	6	7	Bit 0	1	2	3	4	5	6	7	Bit 0	1	2	3	4	5	6	7	Bit 0	1	2	3	4	5	6	7
V=2	P	X	CC	M	PT			sequence number																							
timestamp (in sample rate units)																															
synchronization source (SSRC) identifier																															
contributing source (CSRC) identifiers (optional)																															
Header Extension (optional)																															

RTP Header

- RTCP:
 - Überwachung der Qualität der Datenkanal: versandte Daten/Pakete, verlorene Pakete, Jitter, Round trip delay
 - Unterschiedliche Pakete stehen zur Verfügung: Sender report, receiver report, source description und anwendungsspezifische Pakete

Zusammenfassung Ethernet

- Ethernet ist aufgrund des CSMA/CD Zugriffsverfahrens für harte Echtzeitsysteme nicht geeignet:
 - unbestimmte Verzögerungen durch Backoff-Verfahren
 - keine Priorisierung von Nachrichten möglich
- Aufgrund der starken Verbreitung (\Rightarrow niedrige Kosten, gute Unterstützung) wird Ethernet dennoch häufig in Echtzeitsystemen eingesetzt:
 - Durch Verwendung von echtzeitfähigen Protokollen in weichen Echtzeitsystemen (z.B. Multimedialkontrolle).
 - Durch Verringerung der Kollisionswahrscheinlichkeit durch Aufteilung des Netzes in verschiedene Kollisionsdomänen (z.B. switched ethernet).
- Mittlerweile werden auch diverse Implementierungen von Real-Time Ethernet eingesetzt, allerdings gibt es noch keinen allgemein anerkannten Standard (siehe Zusammenfassung/Trends).

Echtzeitfähige Kommunikation

Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA*)

Vertreter: CAN

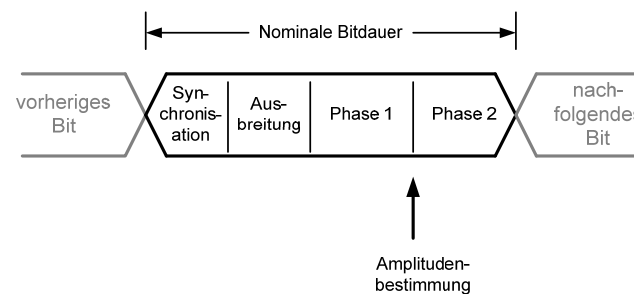
Teilweise wird die hier vorgestellte Methode auch CSMA/CR (Collision Resolution) genannt.

CAN-Protokoll

- Grundidee von Collision Avoidance:
 - Kollisionen werden rechtzeitig erkannt, bevor Nachrichten unbrauchbar werden
 - Wichtigere Nachrichten werden bevorzugt \Rightarrow Priorisierung der Nachrichten
- Daten:
 - CAN (Controller Area Network) wurde 1981 von Intel und Bosch entwickelt.
 - Einsatzbereich: vor allem Automobilbereich, Automatisierungstechnik
 - Datenübertragungsraten von bis zu 1Mbit/s, Reichweite 1km
 - Implementierung der Schichten 1,2 und 7 des ISO/OSI-Modells

CAN: Schicht 1

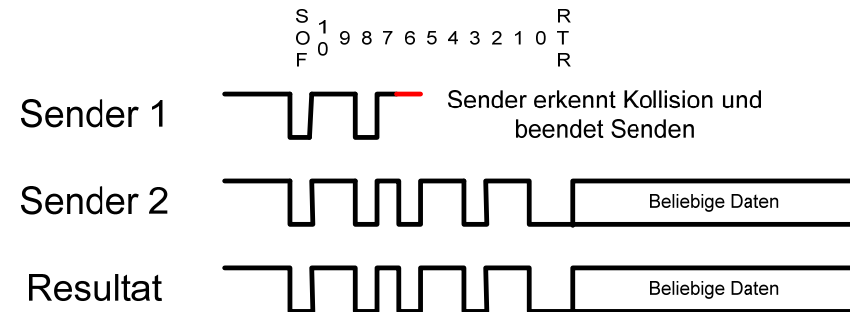
- Busmedium:
 - Kupfer oder Glasfaser
 - Empfehlung Twisted Pair: Möglichkeit zur differentiellen Übertragung (robuster gegenüber Störungen)
- Codierung: NRZ-L (Non-Return-to-Zero-Level)
 - Problem mit NRZ-L: lange Sequenzen monotone Sequenzen von 0 oder 1 können zu Problemen bei der Synchronisation führen, in CAN wird deshalb nach fünf gleichen Bits ein inverses Bit eingefügt (**Bitstuffing**)
- Daten werden **bitsynchron** übertragen:
 - ⇒ Datenübertragungsrate und maximale Kabellänge sind miteinander verknüpft.
 - Konfigurationsmöglichkeiten:
 - 1 MBit/s, maximale Länge: 40m
 - 500 kBit/s, maximale Länge: 100m
 - 125 kBit/s, maximale Länge: 500m
 - Maximale Teilnehmerzahl: 32-128



http://www.port.de/pdf/CAN_Bit_Timing.pdf

CAN: Schicht 2

- Realisierung eines CSMA/CA-Verfahrens:
 - Bei der Übertragung wirken Bits je nach Wert entweder **dominant** (typischerweise 0) oder **rezessiv** (1).
 - Dominante Bits überschreiben rezessive Bits, falls sie gleichzeitig gesendet werden.
 - Jedem Nachrichtentyp (z.B. Sensorwert, Kontrollnachricht) wird ein Identifikator zugewiesen, der die Wichtigkeit des Typs festlegt.
 - Jeder Identifikator sollte nur einem Sender zugewiesen werden.
 - Wie bei Ethernet wartet der Sender bis der Kanal frei ist und startet dann die Versendung der Nachricht.



- Beim gleichzeitigen Senden zweier Nachrichten, dominiert der Identifikator des wichtigeren Nachrichtentyps, den Sender der unwichtigeren Nachricht beendet das Senden.
- ⇒ Verzögerung von hochpriorigen Nachrichten auf die maximale Nachrichtenlänge begrenzt (in Übertragung befindliche Nachrichten werden nicht unterbrochen)

CAN: Framearten

- Datenframe:
 - Versand von maximal 64bit Daten
- Remoteframe:
 - Verwendung zur Anforderung von Daten
 - Wie Datenframe, nur RTR-Feld auf 1 gesetzt
- Fehlerframe:
 - Signalisierung von erkannten Fehlerbedingungen
- Überlastframe:
 - Zwangspause zwischen Remoteframe und Datenframe

Länge in Bit	1	11	1	1	1	4	0..64	15	1	1	1	7	3
Zweck	Start of frame	Identifier (Extended CAN 27bit)	Remote Transmission Bit	Identifier Extension Bit	reserviert	Datenlängenfeld	Datenfeld	CRC-Prüfsumme	CRC Delimiter	Bestätigungsslot	Bestätigungsdelimiter	End of Frame	Intermission

CAN: Schicht 7

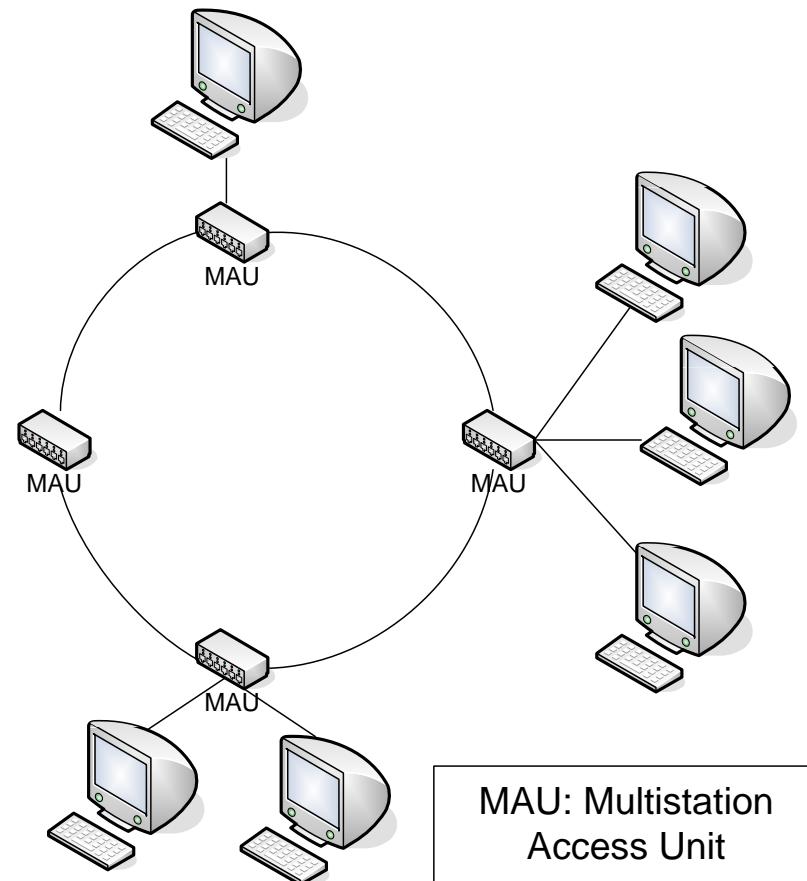
- Im Gegensatz zu Schicht 1 und 2 ist die Schicht 7 nicht in einer internationalen Norm spezifiziert.
- Es existieren jedoch diverse Implementierungen (z.B. CANOpen) für Dienste der Schichten 3-7 zur Realisierung von:
 - Flusskontrolle
 - Geräteadressierung
 - Übertragung größerer Datenmengen
 - Grunddienste für Anwendungen (Request, Indication, Response, Confirmation)
- Zudem gibt es Versuche eine Norm CAL (CAN Application Layer) einzuführen.
- Ziele:
 - Einheitliche Sprache zur Entwicklung von verteilten Anwendungen
 - Ermöglichung der Interaktion von CAN-Modulen unterschiedlicher Hersteller

Echtzeitfähige Kommunikation

Tokenbasierte Verfahren
Vertreter: Token Ring

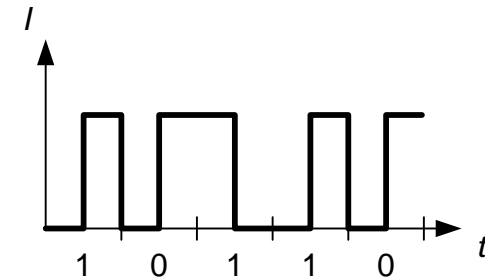
Tokenbasierte Verfahren

- Nachteil von CSMA/CA: Begrenzung der Datenrate und der Netzlänge durch Bitsynchronität
- Tokenbasierter Ansatz: Eine Einheit darf nur dann senden, wenn sie eine Berechtigung (Token) besitzt.
- Die Berechtigung wird zumeist zyklisch weitergegeben \Rightarrow Token Ring.
- Die Berechtigung / das Token ist dabei eine spezielle Bitsequenz.



Token Ring: Schicht 1

- Token Ring wird im Standard IEEE 802.5 spezifiziert.
- Erreichbare Geschwindigkeiten: 4 bzw. 16 MBit/s
⇒ aufgrund der Kollisionsfreiheit mit den effektiven Datenübertragungsraten von 10 bzw. 100 MBit/s Ethernet vergleichbar
- Codierung:
 - differentieller Manchester-Code
 - somit selbstsynchronisierend
- Topologie:
 - Ring
 - aufgrund der möglichen Verwendung von MAUs auch sternförmige Verkabelung möglich



Differentieller Manchester-Code

Token Ring: Zugriffsverfahren

1. Die Station, die das Token besitzt, darf Daten versenden.
2. Das Datenpaket wird von Station zu Station übertragen.
3. Die einzelnen Stationen empfangen die Daten und regenerieren sie zur Weitersendung an den nächsten Nachbarn.
4. Der Empfänger einer Nachricht kopiert die Nachricht und leitet die Nachricht mit dem gesetzten C-Bit (siehe Nachrichtenaufbau) zur Empfangsbestätigung weiter.
5. Empfängt der Sender seine eigene Nachricht, so entfernt er diese aus dem Netz.
6. Nach Ende der Übertragung wird auch das Token weitergesendet (maximale Token-Wartezeit wird vorher definiert, Standardwert: 10ms)
7. Im 16 MBit/s Modus wird das Token direkt im Anschluß an das Nachrichtenpaket versendet (**early release**) \Rightarrow es können sich gleichzeitig mehrere Token im Netz befinden

Token Ring: Prioritäten

- Token Ring unterstützt Prioritäten:
 - Insgesamt gibt es 8 Prioritätsstufen (3 Bit)
 - Jeder Station wird eine Priorität zugewiesen.
 - Der Datenrahmen besitzt ebenfalls einen Speicherplatz für die Priorität.
 - Eine Station kann in die Priorität in dem Prioritätsfeld von Nachrichten vormerken, allerdings darf die Priorität nur erhöht werden.
 - Stationen dürfen Tokens nur dann annehmen, wenn ihre Priorität mindestens so hoch ist, wie die Priorität des Tokens.
 - Applet zum Ablauf:
<http://www.nt.fh-koeln.de/vogt/mm/tokenring/tokenring.html>

Token Ring: Token Paket

- Das Token besteht aus:
 - Startsequenz (1 Byte, JK0JK000)
 - J, K: Codeverletzungen entsprechend Manchester-Code (kein Übergang in Taktmitte)
 - Zugriffskontrolle (1 Byte, PPPTMRRR)
 - P: Zugriffspriorität
 - T: Tokenbit (0: freies Token, 1: Daten)
 - M: Monitorbit
 - R: Reservierungspriorität
 - Endsequenz (1 Byte, JK1JK1IE)
 - I: Zwischenrahmenbit (0: letztes Paket, 1: weitere Pakete folgen)
 - E: Fehlerbit (0: fehlerfrei, 1: Fehler entdeckt)

Token Ring: Tokenrahmen

- Der Datenrahmen besteht aus:
 - Startsequenz wie Token
 - Zugriffskontrolle wie Token
 - Rahmenkontrolle (1 Byte, FFrrZZZZ)
 - FF: Paketart (00: Protokollsteuerpaket, 01: Paket mit Anwenderdaten)
 - rr: reserviert für zukünftige Anwendungen
 - ZZZZ: Informationen zur Paketpufferung
 - Zieladresse (6 Byte): Adresse eines spezifischen Geräts oder Multicast-Adresse
 - Quelladresse (6 Byte)
 - Routing Informationen (0-30 Bytes): optional
 - Daten
 - Prüfsumme FCS (4 Byte): Berechnung auf Basis der Daten zwischen Start- und Endsequenz
 - Endsequenz wie Token
 - Paketstatus (1 Byte ACrrACrr)
 - A: Paket wurde vom Empfänger als an in adressiert erkannt
 - C: Paket wurde vom Empfänger erfolgreich empfangen

Token Ring: Monitor

- Für den fehlerfreien Ablauf des Protokolls existiert im Token Ring ein Monitor.
- Aufgaben:
 - Entfernung von fehlerhaften Rahmen
 - Neugenerierung eines Tokens bei Verlust des Tokens (nach Ablauf einer Kontrollzeit)
 - Entfernung endlos kreisender Nachrichten bei Ausfall der Senderstation (Markierung der Nachricht beim Passieren des Monitors, Löschen der Nachricht beim 2. Passieren)
 - Signalisierung der Existenz des Monitors (durch Active Monitor Present Nachricht)

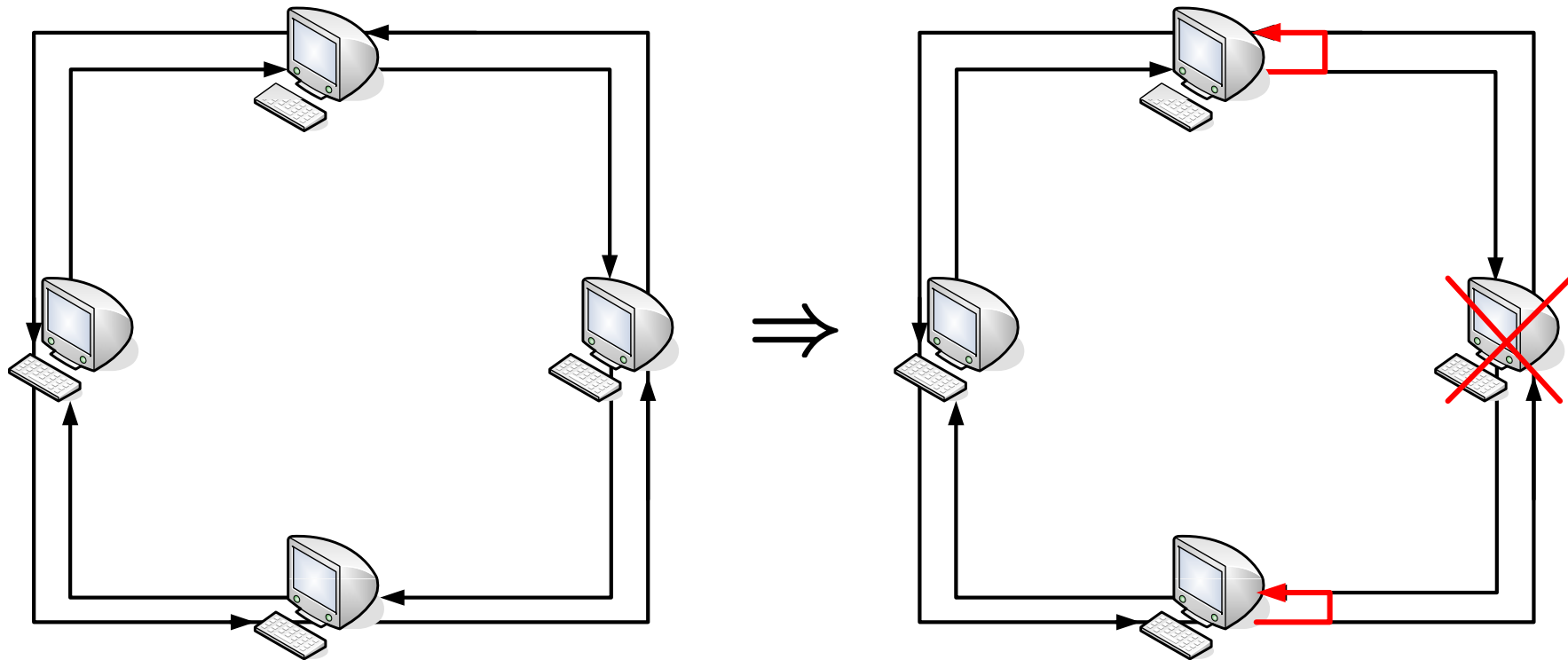
Token Ring: Initialisierung / Rekonfigurierung

- Bei der Initialisierung bzw. dem Ablauf des Standby Monitor Timer (Mechanismus zur Tolerierung des Ausfalls des Monitors)
 1. Senden eines Claim Token Paketes
 2. Überprüfung, ob weitere Pakete die Station passieren
 3. Falls nein \Rightarrow Station wird zum Monitor
 4. Generierung eines Tokens
 5. Jede Station überprüft mittels des Duplicate Adress Test Paketes, ob die eigene Adresse bereits im Netzwerk vorhanden ist.
- Der Ausfall einer Station kann durch das Netzwerk erkannt werden und evtl. durch Überbrückung kompensiert werden.

FDDI

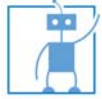
- Fiber Distributed Data Interface (FDDI) ist eine Weiterentwicklung von Token Ring
- Medium: Glasfaserkabel
- doppelter gegenläufiger Ring (aktiver Ring, Reservering) mit Token-Mechanismus
- Datenrate: 100 MBit/s, 1000 MBit/s
- Codierung: 4B5B (wie in FastEthernet)
- maximal 1000 Einheiten
- Ringlänge: max. 200 km
- Maximaler Abstand zwischen zwei Einheiten: 2 km
- Fehlertoleranz (maximal eine Station)
- Nachrichten können hintereinander gelegt werden (early release)
- Weitere Entwicklungen FDDI-2

Fehlerkonfiguration in FDDI



MAP / Token Bus

- **MAP: Manufacturing Automation Protocol** (Entwicklung ab 1982 von General Motors)
- Einsatz hauptsächlich im Produktionsbereich
- Schicht 1: anstelle von Ring-Topologie nun beliebige Topologie durch den Einsatz von Bridges, Gateways und Routern
- Medienzugriffsverfahren:
 - Token Bus, spezifiziert in IEEE 802.4
 - ähnlich Token-Ring, die benachbarte Station zur Weiterleitung des Tokens wird anhand einer Adresse bestimmt.
- In MAP werden zudem alle sieben Schichten des ISO/OSI-Modells spezifiziert.
- Aufgrund des Umfangs und der Komplexität konnte sich MAP nicht durchsetzen.
- Maximale Übertragungsrate: 10 MBit/s



Echtzeitfähige Kommunikation

Zeitgesteuerte Verfahren

Vertreter: TTP

Zugriffsverfahren: TDMA

- **TDMA (Time Division Multiple Access)** bezeichnet ein Verfahren, bei dem der Zugriff auf das Medium in Zeitscheiben (slots) eingeteilt wird.
- Die Zeitscheiben werden für jeweils einen Sender zur Verfügung gestellt.
- Vorteile:
 - Kollisionen sind per Design ausgeschlossen
 - Einzelnen Sendern kann eine Bandbreite garantiert werden.
 - Das zeitliche Verhalten ist vollkommen deterministisch.
 - Synchronisationsalgorithmen können direkt im Protokoll spezifiziert und durch Hardware implementiert werden.
- Nachteil:
 - keine dynamische Zuteilung bei reinem TDMA-Verfahren möglich
- Bekannte Vertreter: TTP, Flexray (kombiniert zeitgesteuert und dynamische Kommunikation)