

Diplomarbeit



„Einsatz eines zuverlässigen Transportdienstes zur Simulation des zeitgesteuerten Echtzeitprotokolls TTP/C“

Nawfel Chaieb

Betreuer : Prof. Dr. Walter F. Tichy
Betreuende Mitarbeiter : Dipl.-Inform. Marc Schanne

Inhalt

- **Motivation**
- **Ereignisgesteuerte /zeitgesteuerte Architektur**
- **Das zeitgesteuerte Protokoll TTP/C**
- **Architektur des Simulators**
- **Beispiel Anwendung: Brake-by-Wire**
- **Fehlerinjektion**

Motivation

- Die dauerhaft zunehmende Kompliziertheit von Fahrzeugenelektronik und von schnell wachsenden Menge der Sensoren/Aktoren und der elektronischen Steuereinheiten.
 - Deterministische Protokolle mit fehlertolerantem Verhalten werden benötigt
 - hohe Datenrate für kritische Systeme
- Das TTP/C Protokoll genügt die neuen Anforderungen
- Ein TTP/C Controller kostet zwischen 50 und 200 Euro, d.h. ein TTP-Netz (Hardware und Software) bis zu mehrere tausend Euro
- Simulator mit einer Testumgebung und Beispiel-Anwendung

Kommunikationssysteme

LIN-Bus:

- Einsatz in kostensensitiven Bereichen
- Lokale Ankopplung von Sensoren und Aktoren
- Datenrate von 20kbit/s

CAN-Bus:

- Einsatz seit den 90er Jahren im Fahrwerks- und Antriebsbereich
- Datenrate von max. 500kbit/s in Fahrzeugen

FlexRay / TTP Klasse C:

- Gefordert von einem Konsortium mehrerer Automobilhersteller
- Höhere Übertragungsraten
- Echtzeitfähig und fehlertolerant
- Unterstützung verteilter Steuer- und Regelungssysteme
- Flexibilität (Erweiterungen, Bandbreitennutzung, ...)

Ereignisgesteuert (ET) vs. zeitgesteuert (TT)

Ein Echtzeitsystem ist **ereignisgesteuert**, wenn diese Steuersignale nur vom Auftreten von Ereignissen abgeleitet werden, wie :

- Auftreten einer externen Unterbrechung (Interrupt)
- Beendigung eines Tasks
- Empfang einer Nachricht

Ein Echtzeitsystem ist **zeitgesteuert**, wenn die Steuersignale, welche :

- die Beobachtung eines echtzeitfähigen Teilsystems
- das Senden und Empfangen von Nachrichten
- die Startzeit eines Tasks festlegen, vom Fortschreiten einer globalen Zeit ableiten.

Vergleich

Event-Triggered Architektur	Time-Triggered Architektur
<ul style="list-style-type: none">- Nachrichten kommunizieren Ereignisse - Verhalten unvorhersehbar und von den Ereignissen abhängig - Grosse Jitter - Gut für unregelmäßige Datentransfer - Membership-Dienst schwierig - Zufälliger Buszugriff	<ul style="list-style-type: none">- Nachrichten kommunizieren Zustände - Verhalten von der Zeit abhängig und vorhersehbar - Minimales Jitter. - eignet sich für regulären Datenfluss - Membership-Dienst einfach - Deterministisches Verhalten - Vorhersagbare Netzlast

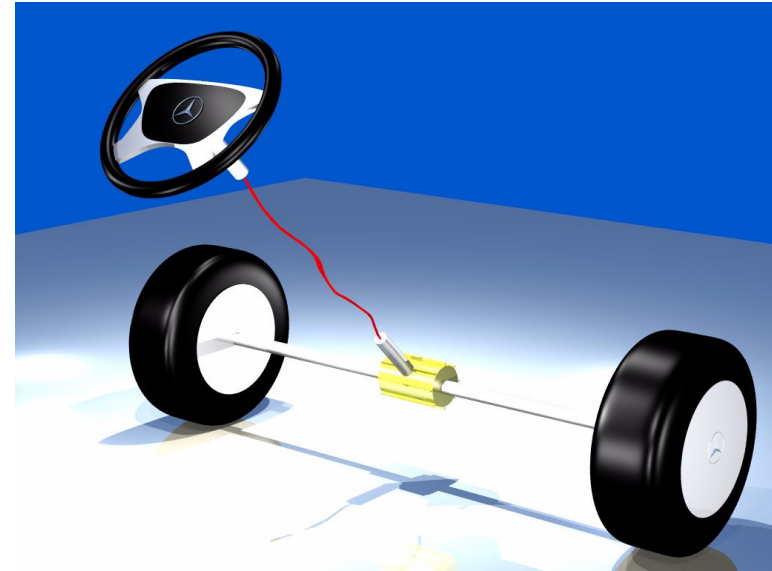
Die zeitgesteuerte Architektur (TTA)

Die zeitgesteuerte Architektur wurde für verteilte harte Echtzeitsysteme entwickelt.

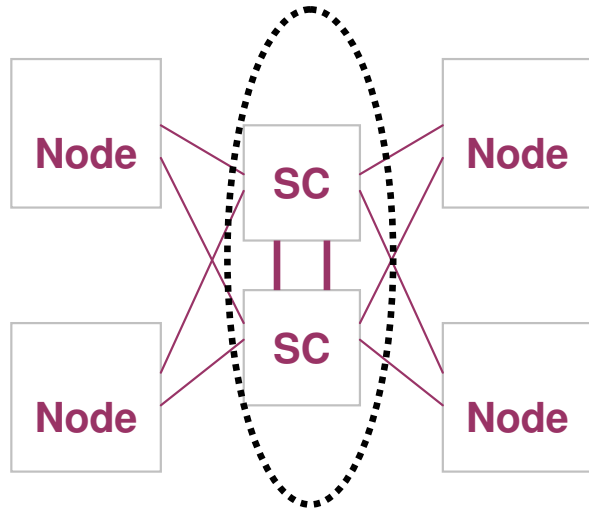
TTA reduziert die Komplexität, die in ereignisgesteuerten Systemen durch die steigende Zahl an miteinander kommunizierenden Funktionen entsteht.

Vorteile der neuen Bus-Technologien:

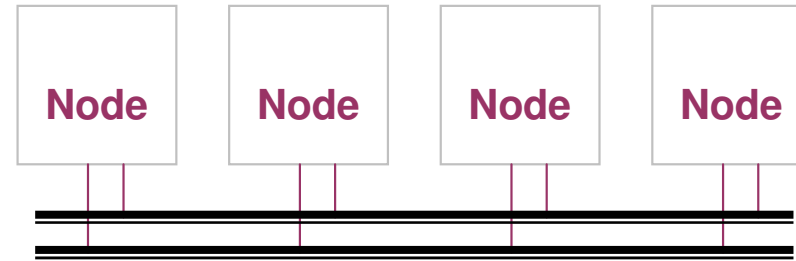
- Flexible Gestaltungsmöglichkeiten
- Gewichtsersparnis
- Platz sparend
- Erhöhung der aktiven und passiven Sicherheit
- Neue Innenraumkonzepte möglich



Das zeitgesteuerte Protokoll TTP/C



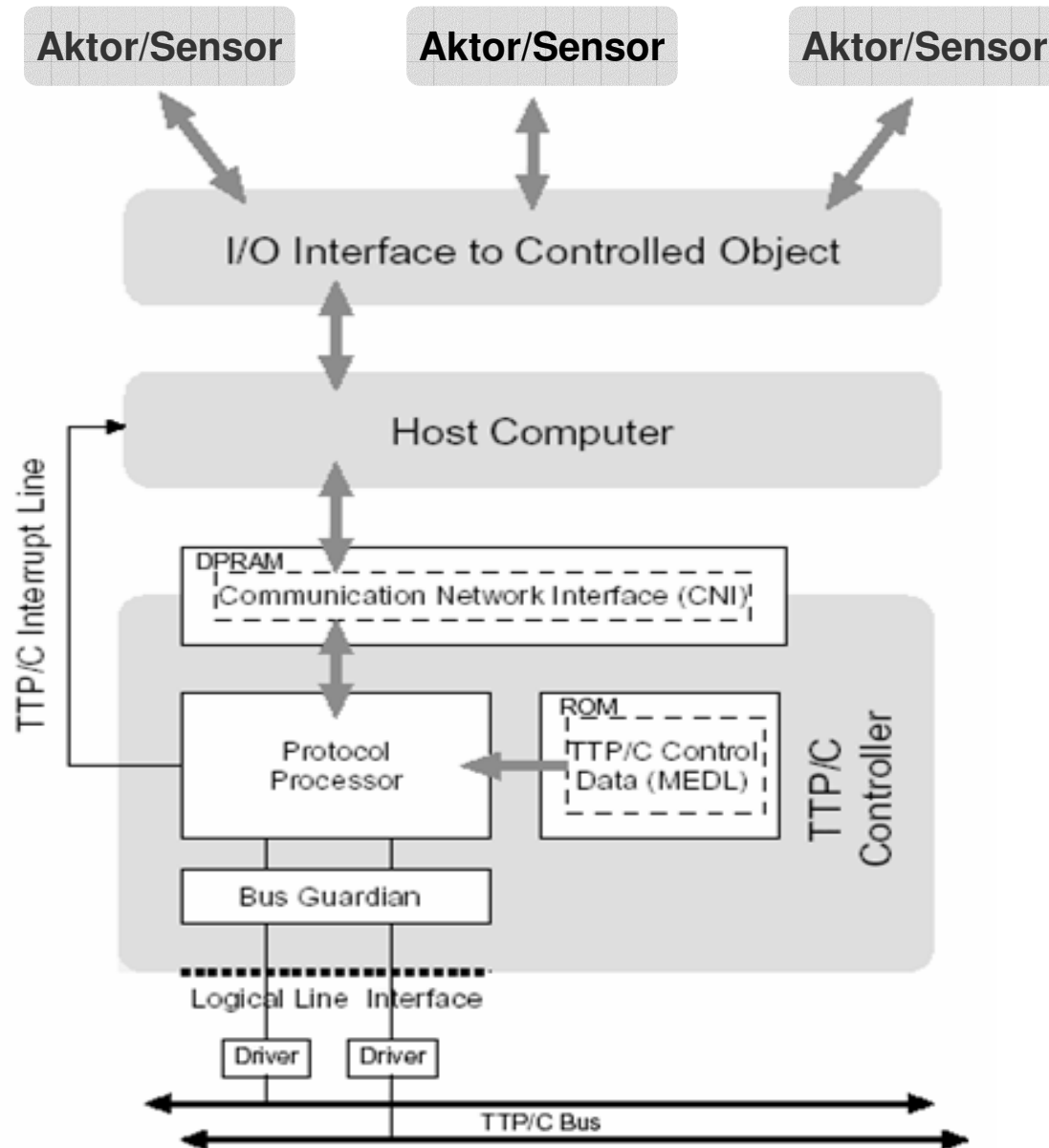
Star Netzwerktopologie



Bus Netzwerktopologie

- Medien-Zugriffs-Strategie beruht auf einem statischen TDMA-Schema
- für alle Knoten eine einheitliche Zeit und ein Zeitmodell.

TTP/C Knoten Architektur



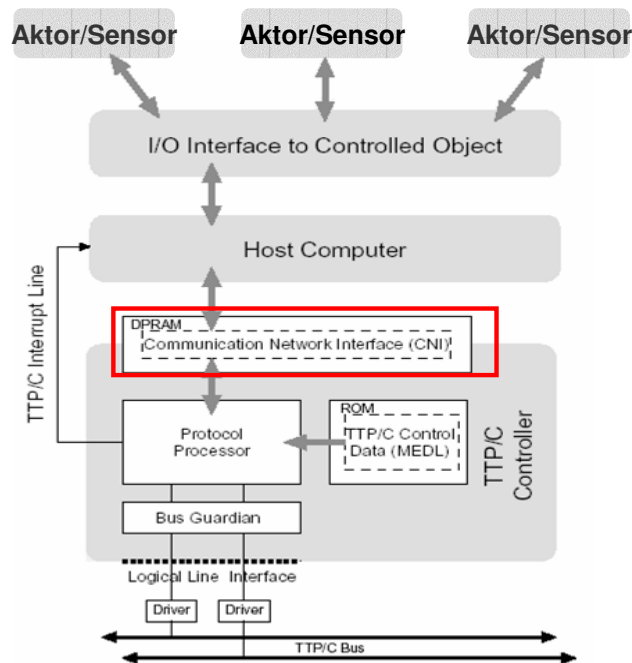
Die Kommunikationsnetzschnittstelle (CNI)

→ befindet sich zwischen dem autonomen Controller und der Host-CPU in einem DPRAM.

→ ist eine Datenschnittstelle und versteckt das Kommunikations-Subsystem hinter der Speicherabstraktion

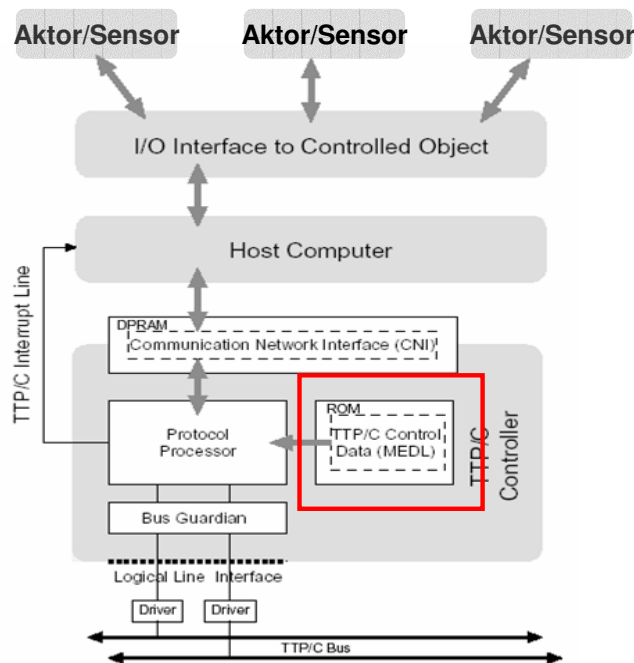
Das CNI besteht aus 3 Komponenten:

- Status Bereich: Globale Zeit, Mitgliedschaft, Protokoll Zustand
- Steuerungsbereich: Controller ein/ausschalten
Steuerung des Zeitinterrupts
- Datenbereich: Aktuelle Version der input/output Nachrichten



Message Descriptor List (MeDL)

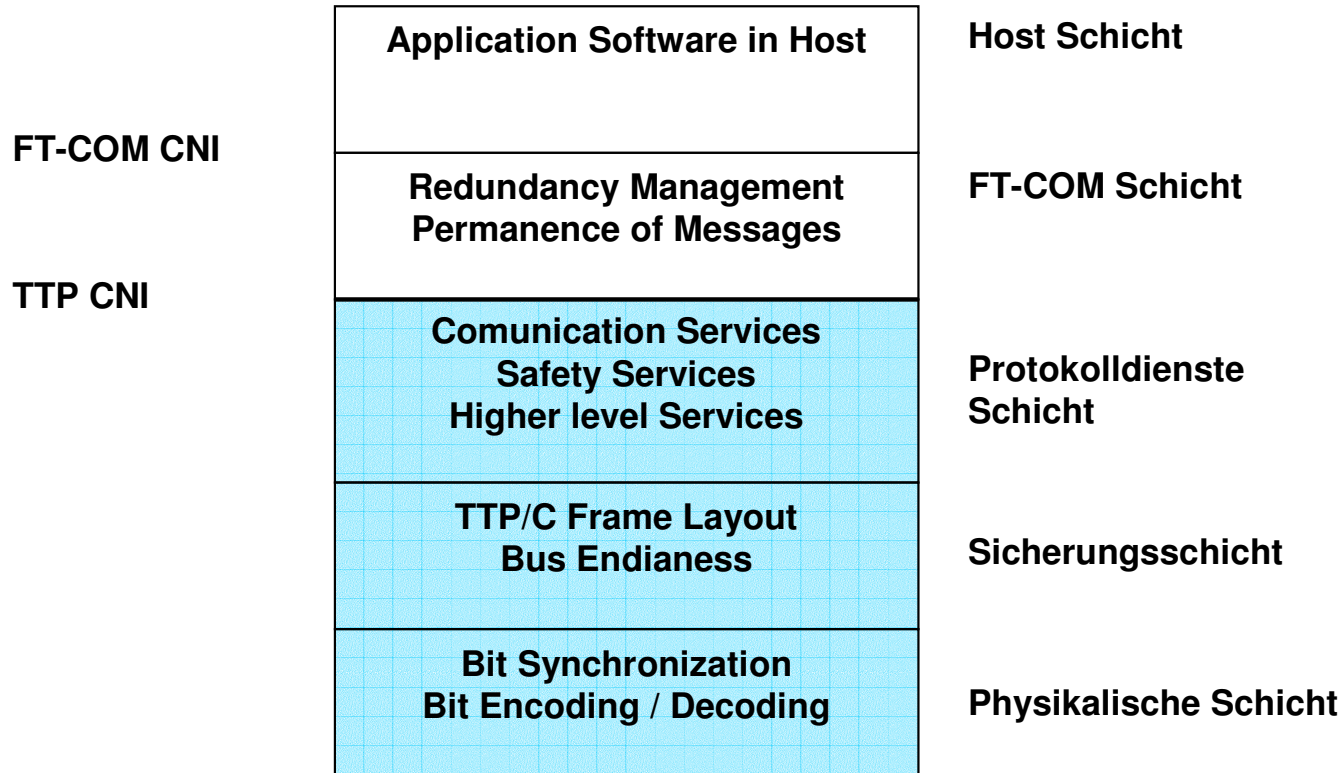
- wird in einem ROM gespeichert
- die genaue Zuordnung der Zeitscheiben an die einzelnen Knoten
- die notwendigen Kontrollinformationen z.B. Identifikation des Knotens, Dauer der TDMA-Runden Modus...



→ für zu sendende Nachrichten ist der Sendezeitpunkt und die Adresse in der CNI enthalten, wo die Daten abgeholt werden müssen.

→ für zu empfangende Nachrichten ist der Empfangszeitpunkt und die Adresse in der CNI enthalten, wo die Daten abgelegt werden müssen.

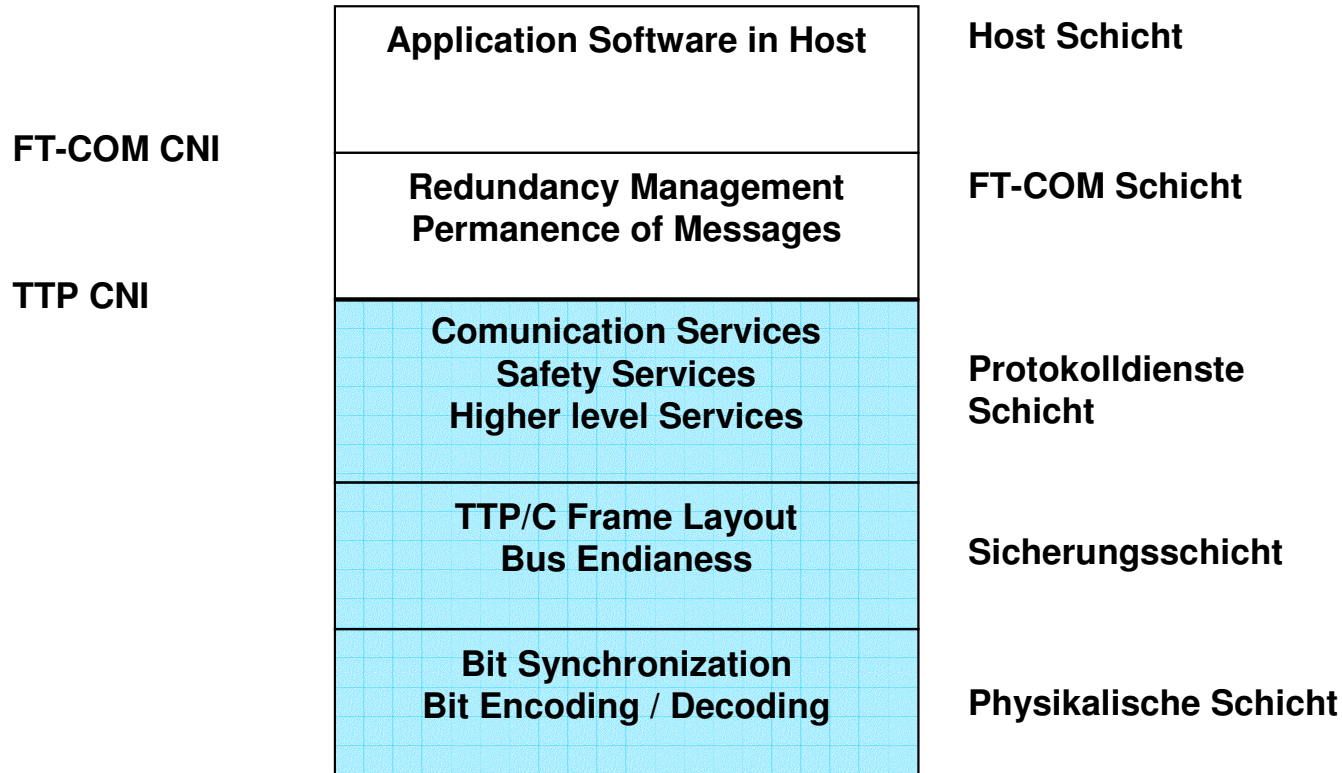
TTA Schichtenarchitektur



Stellen Sie die Mittel zur Verfügung, Rahmen zwischen den Knoten auszutauschen



- Schichten implementiert im Host-Applikation
- Schichten des TTP/C Protokolls

TTA Schichtenarchitektur

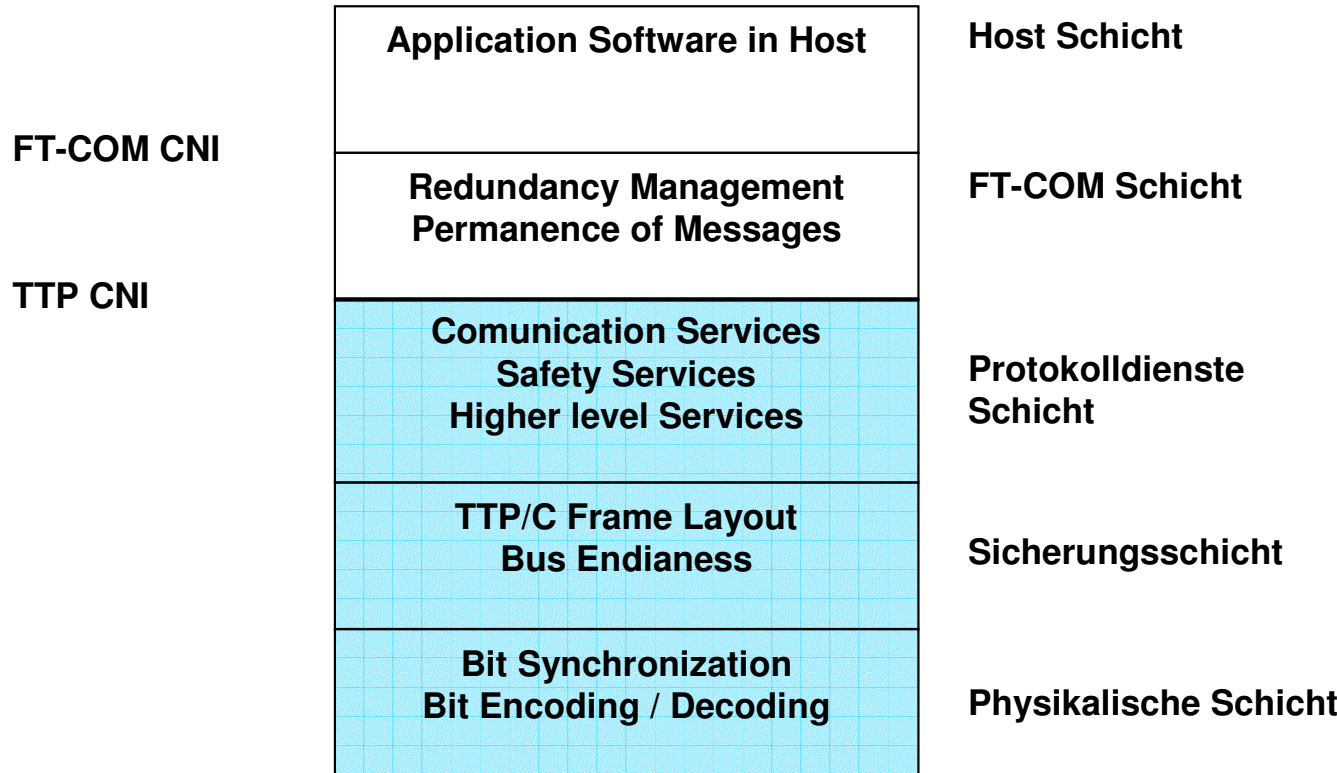


Die Protokollservice Schicht liefert den Protokollbetrieb einschließlich aller Services von TTP/C

Rahmen Erkennung und Austausch zwischen den Knoten

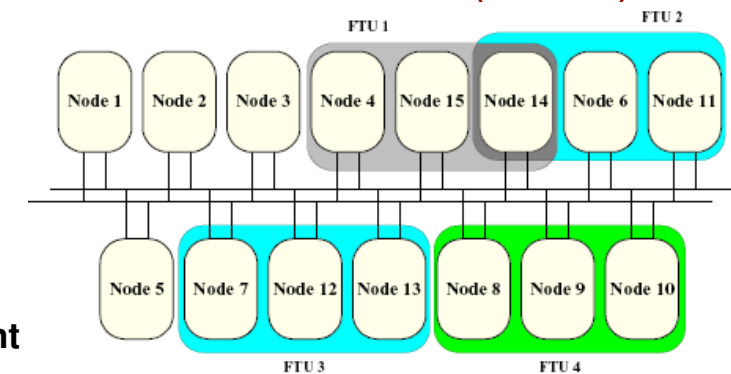
-  Schichten implementiert im Host-Applikation
-  Schichten des TTP/C Protokolls

TTA Schichtenarchitektur



die Ausführung der knotenlokalen Applikation, also der Funktionen dieses Knotens.

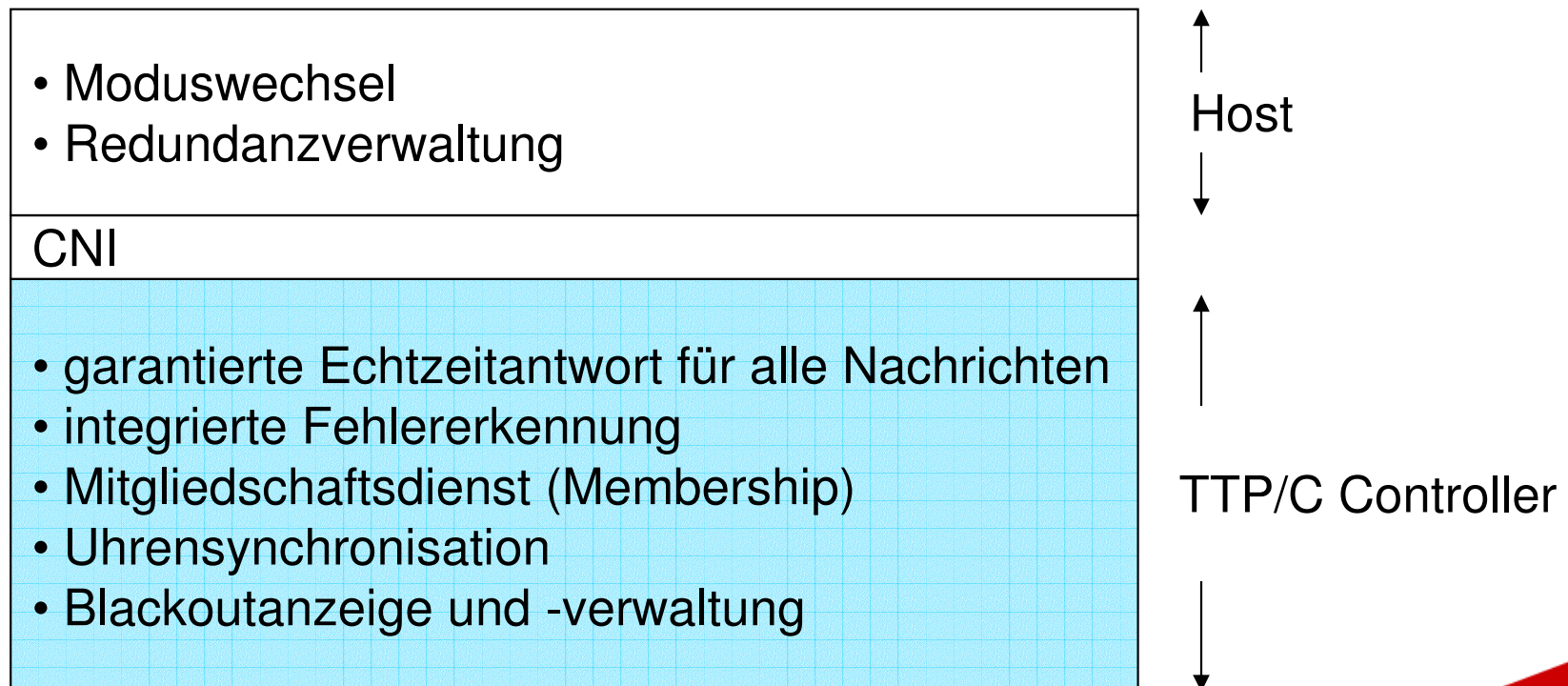
Gruppierung von zwei oder mehr Knoten in Fail Tolerant Units (FTUs)



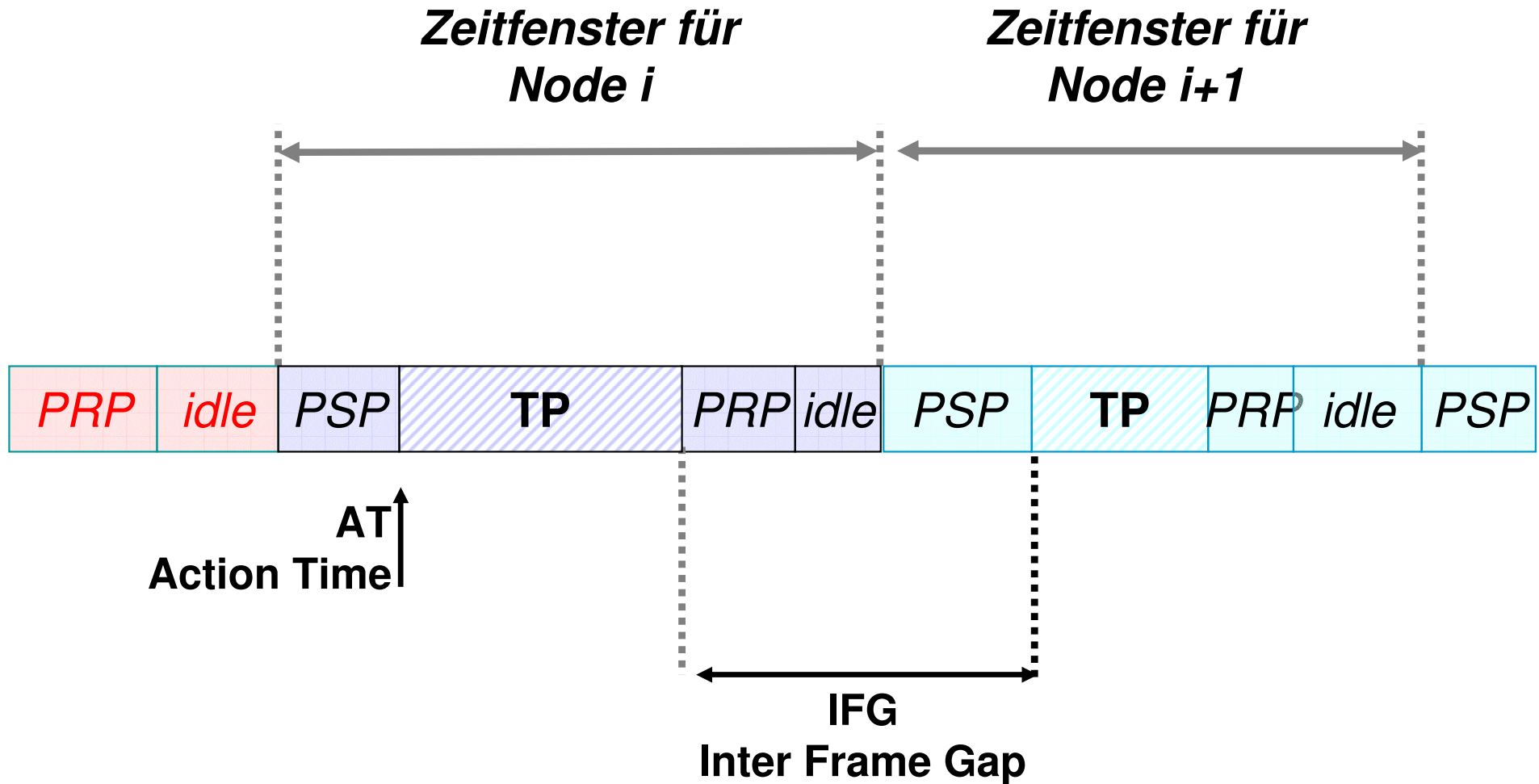
- Schichten implementiert im Host-Applikation
- Schichten des TTP/C Protokolls

Dienste von TTP/C

TTP/C unterstützt alle Dienste, die für den Entwurf von sicherheitskritischen, fehlertoleranten Klasse-C Systemen gebraucht werden:

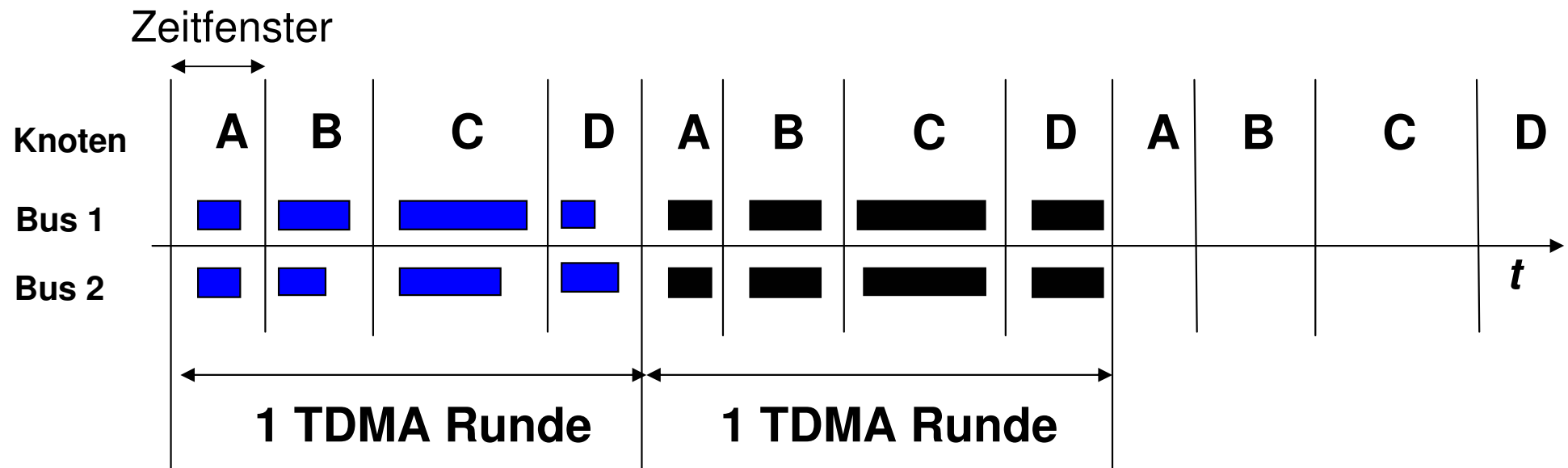


TTP/C: Phasen der Übertragung



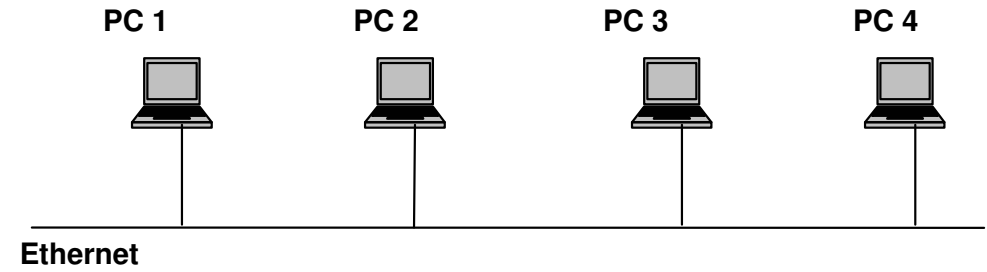
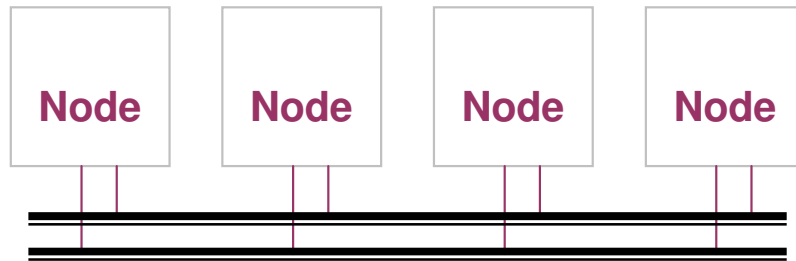
PSP (Pre Send Phase) – TP (Transmission Phase) – PRP (Post Receive Phase)

Beispiel mit 4 Kommunikationspartnern

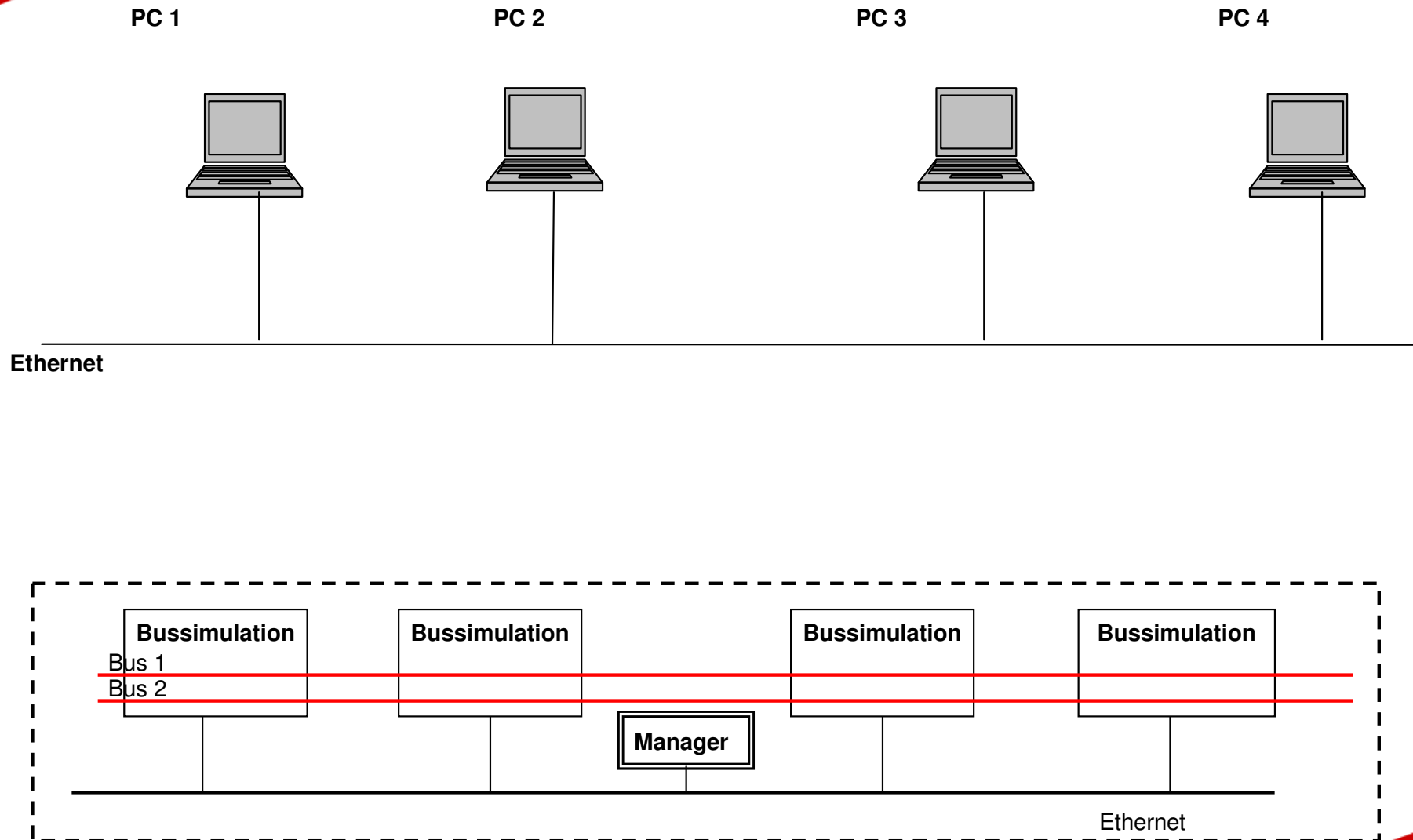


PSP (Pre Send Phase) – TP (Transmission Phase) – PRP (Post Receive Phase)

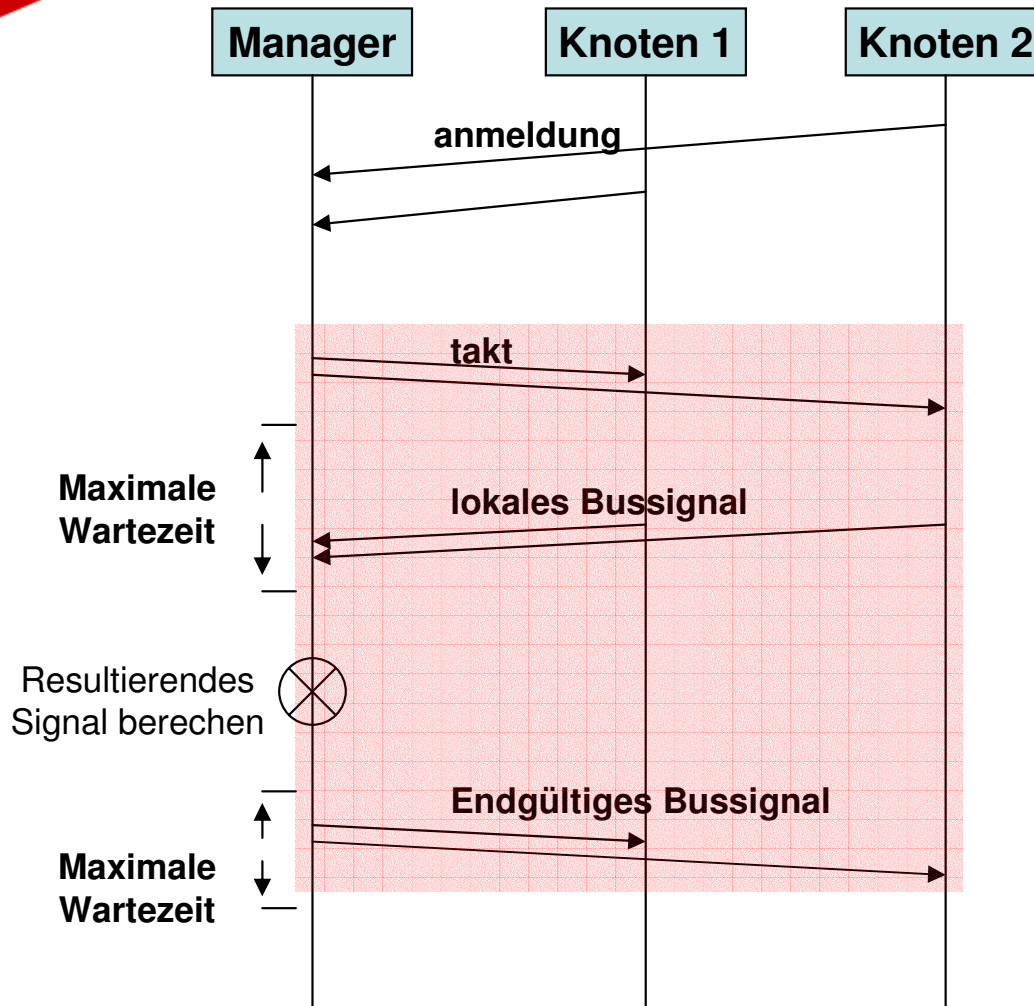
Architektur des Simulators



Architektur des Simulators



Protokoll der Bus-Simulation



Anmeldung:

- Jeder Knoten muss sich beim Manager anmelden bevor er in die Kommunikation teilnehmen kann

1.Phase:

- Takt-Daten senden
- Auf Antwort der Knoten warten

2.Phase:

- Bussignal berechnen
- Der Manager sendet das Bussignal zu den Knoten.

Bussimulation

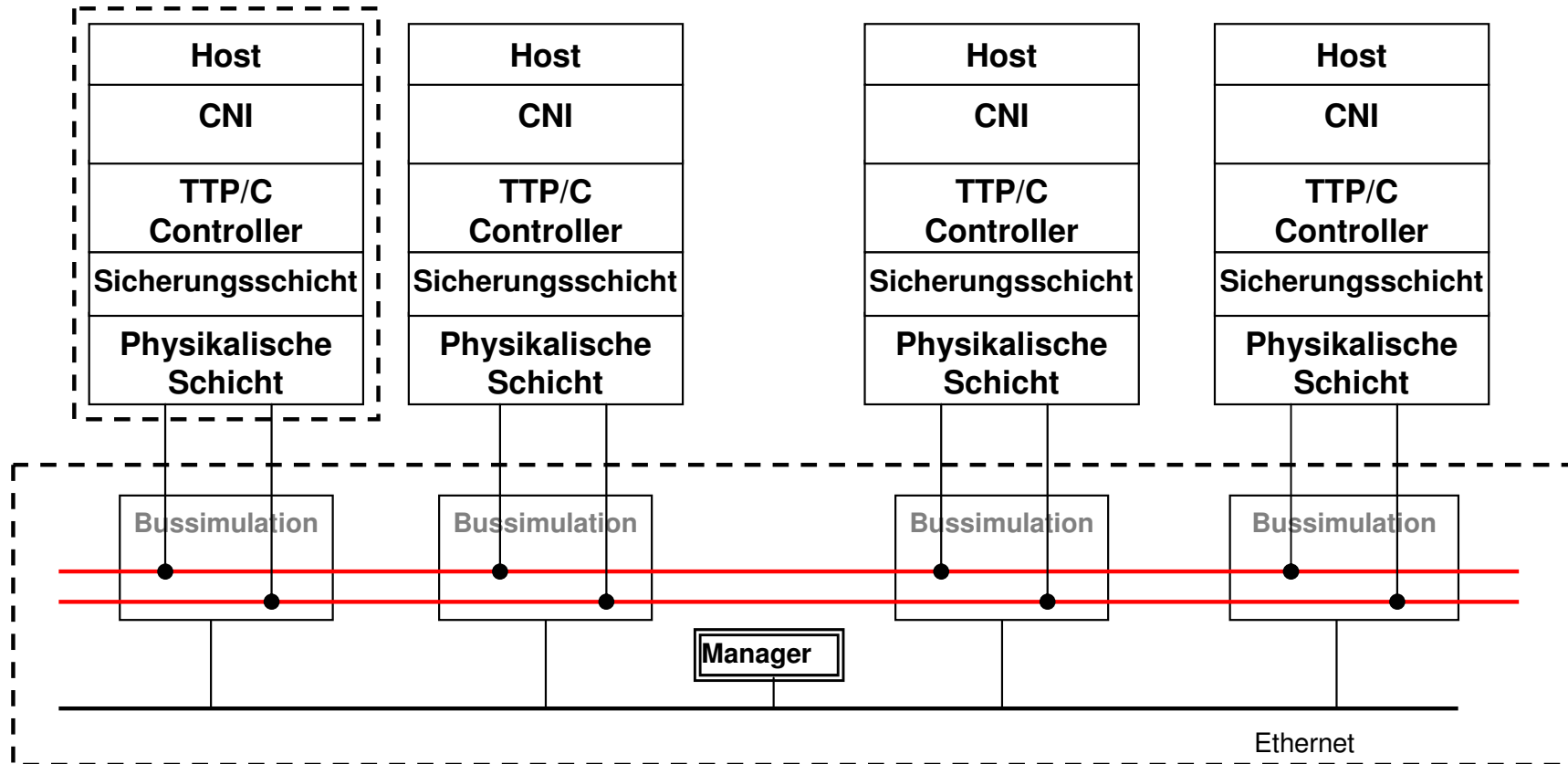
- Wired-And Mechanismus: der dominante Zustand (logisch 0) überschreibt den rezessiven Zustand (logisch 1).
- Ein Knoten im System sendet eine logische 0, eine logische 1, oder kein Signal (tristate) auf dem Bus.
- Broadcast-Bus :
 - * es liefert eine Nachricht an alle Knoten, die intakt sind,
 - * alle Knoten erhalten alle Nachrichten in derselben Reihenfolge (keine Überholung).

sender Knoten

	0	1	Tristate
anderer Knoten	0	0	0
1	0	1	1
Tristate	0	1	Tristate

Entscheidungstabelle
des Managers

Simulator Architektur

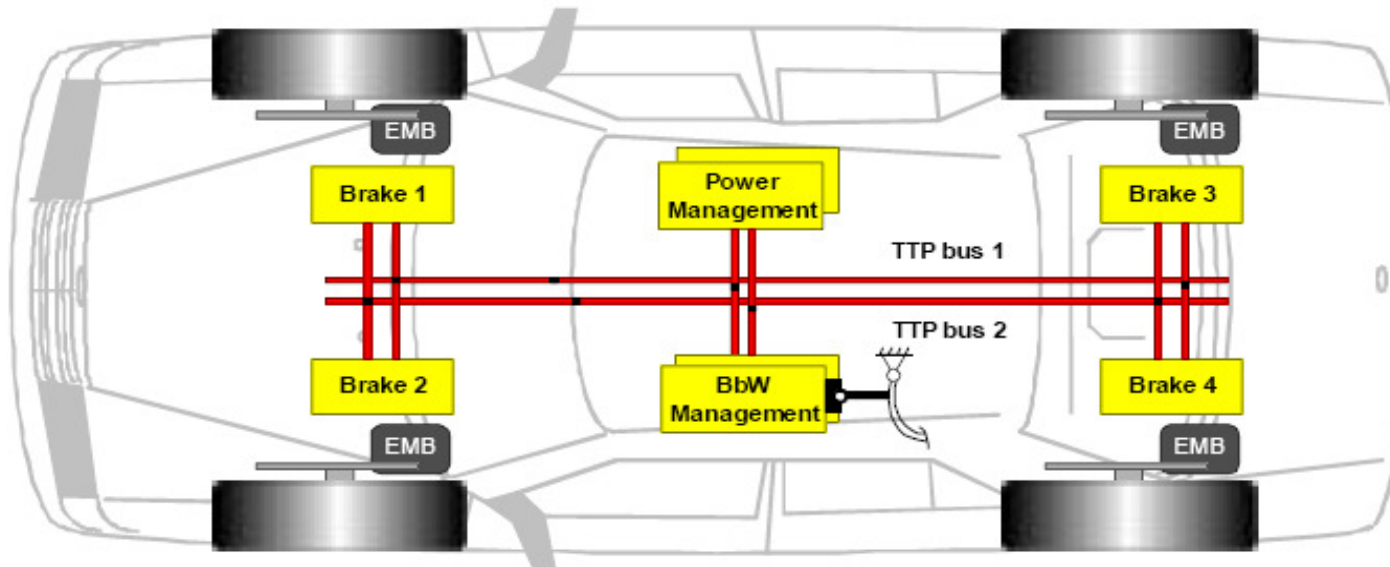


X-by-wire

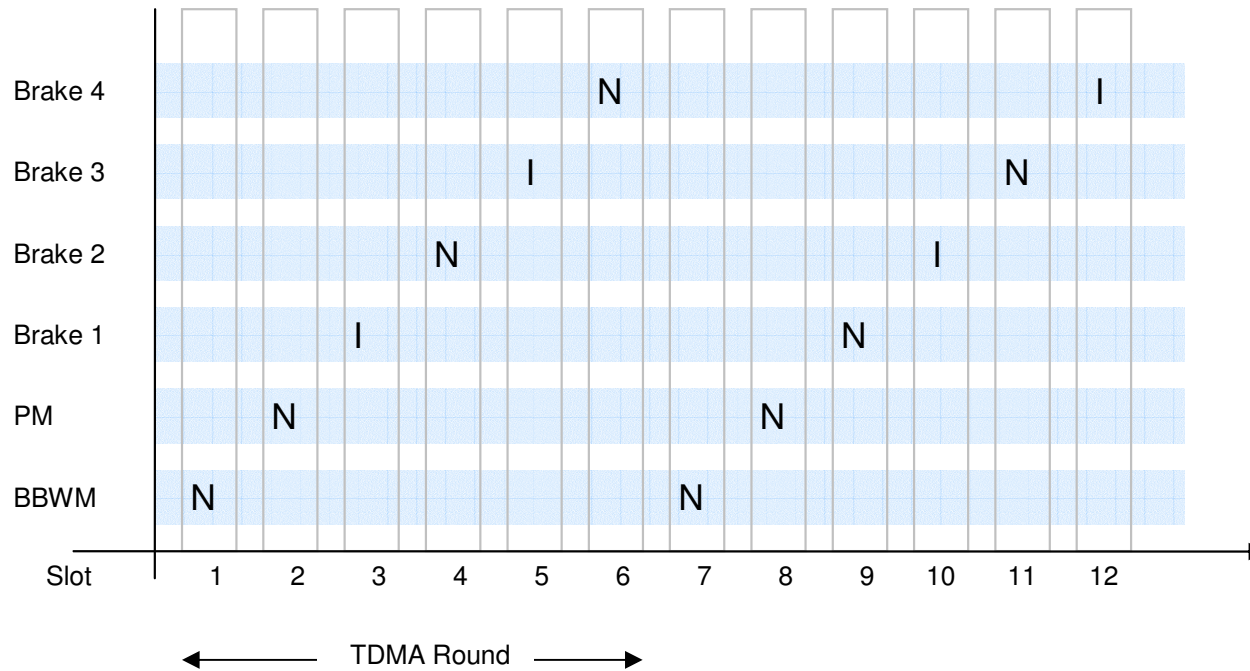
X-by-wire Systeme sind z.B. „Drive-by-wire“, „Brake-by-wire“, „fly-by-wire“...
Mögliche Assistenzsysteme (X-by-Wire System):

- Spurhalte-/Spurwechselsysteme
- Notbremsassistent
- Automatisches Einparken
- Abstandshalter („Autonomous Cruise Control“ ACC)
- Zukunft : Autopilot?

Brake-by-wire mit 6.Komponenten



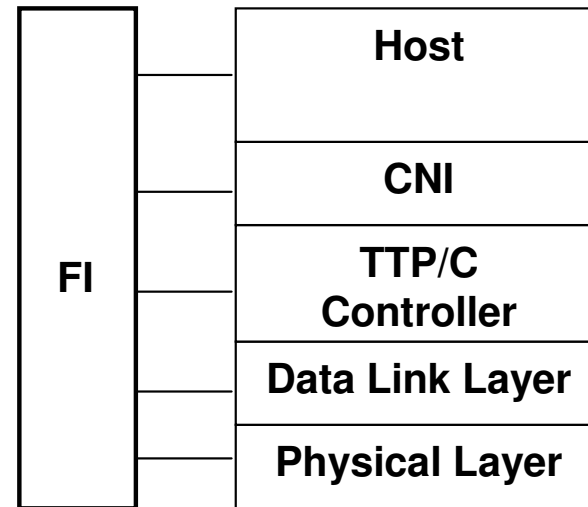
Beispiel Applikation : Brake-by-wire System



N : N-Rahmen : zum Senden der Daten, nur implizite Informationen über den Systemzustand

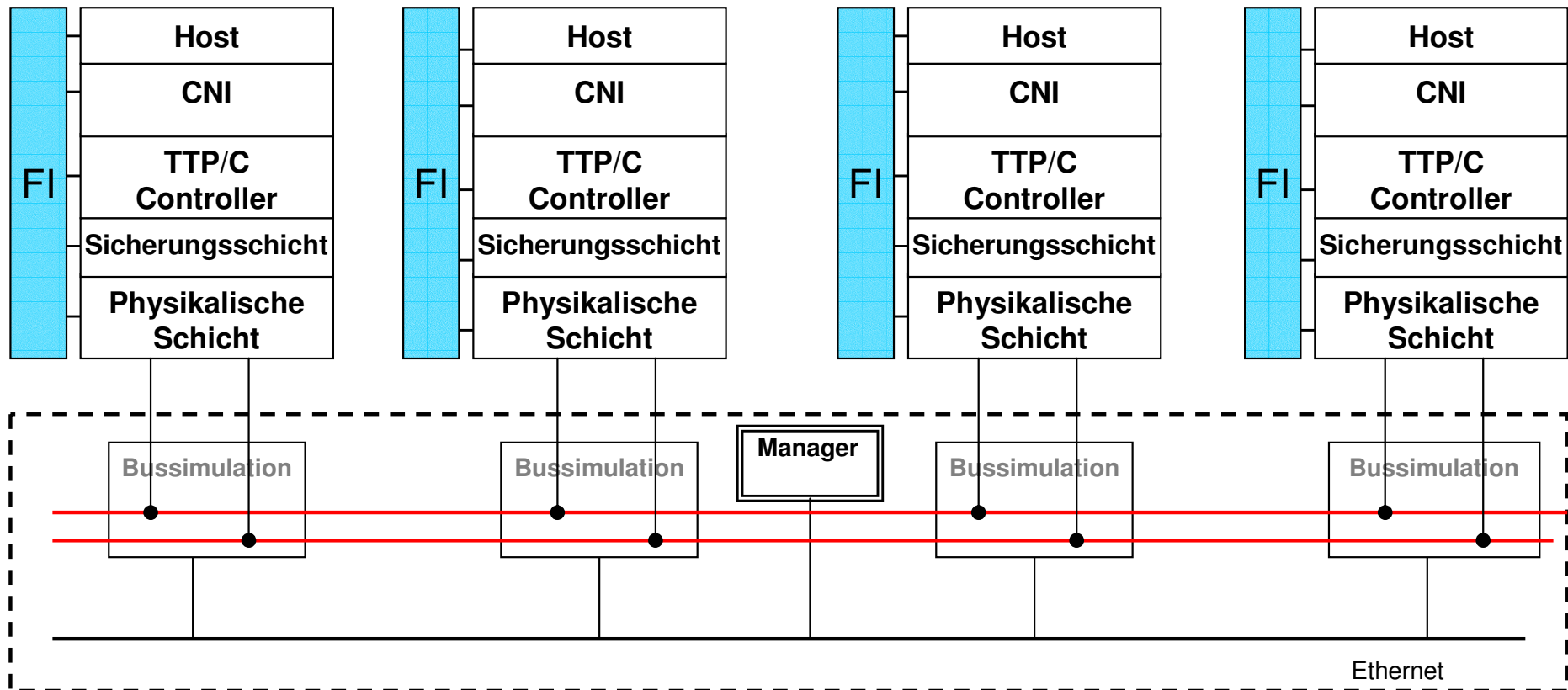
I : I-Rahmen : wird zur Initialisierung und Synchronisation von Controllern verwendet. Explizite Informationen über die globale Zeit und aktuelle Busteilnehmer

Fehlerinjektion



- 3. Komponente in der Simulatorarchitektur
- Fehlerinjektion auf alle Schichten möglich
- Systemzustand und Protokollablauf beeinflussen
- Testen der Fehlertoleranz im System
- Vergleichen der Simulatorreaktion auf eines auftretenden Fehlers mit der TTP/C-Spezifikation

Fehlerinjektion



Fehlerinjektion & Tests

Mögliche Probleme in einer TTP-Netz, die durch das Protokoll gelöst werden:

- Babbling Idiot
- Zeit /Synchronisationsprobleme
- Byzantine Fehlern
- Umkippen von einzelnen Bits

Node	n ₁	n ₂	n ₃	n ₄	n ₅
n ₁	1	1	1	0	0
n ₂	1	1	1	0	0
n ₃	1	1	1	0	0
n ₄	0	0	0	1	1
n ₅	0	0	0	1	1

Clique

Clique

Testen des Simulators

- System Start up
 - ✓ Dauer der Startphase und ausgetauschte Daten
- Neuer Knoten im System
 - ✓ Test der Integrationsabläufe, System erkennt den neuen Knoten
- Unbekannter Busteilnehmer
 - ✓ Daten werden verworfen, Fail-Silent-Prinzip
- Fehlerhafte Daten auf dem Bus
 - ✓ Fehlererkennung und Fehlertoleranz
- Synchronisation
 - ✓ Abweichungen der lokalen Uhren ausgleichen

Danke für Ihre Aufmerksamkeit!



Fragen ?

Transportprotokolle

Transportprotokolle sorgen für die zuverlässige Datenübertragung zwischen zwei Computern. Die verbreitetsten Protokolle sind:

- **APPC - Advanced Program-to-Program Communication** : ein Peer-to-Peer Protokoll in IBM SNA, hauptsächlich eingesetzt bei AS/400
- **TCP - Transmission Control Protocol** : das TCP/IP Programm für die gesicherte Übertragung sequentieller Daten
- **SPX - Sequential Packet Exchange** : Teil von Novells IPX/SPX-Protokollsammlung für sequentielle Daten
- **NWLink - Die Microsoft-Implementierung des IPX/SPX Protokolls**
- **NetBEUI - Network Basic Input/Output System Extended User Interface** : Stellt zwischen Computern eine Kommunikationssitzung her (NetBIOS) und bietet Transportdienste für die Daten (NetBEUI)
- **ATP - AppleTalk Transaction Protocol, NBP - Name Binding Protocol** : Ein Kommunikationssteuerungs- & Datenübermittlungsprotokoll von Apple.

Zuverlässige Transport Protokolle

ISO/OSI:

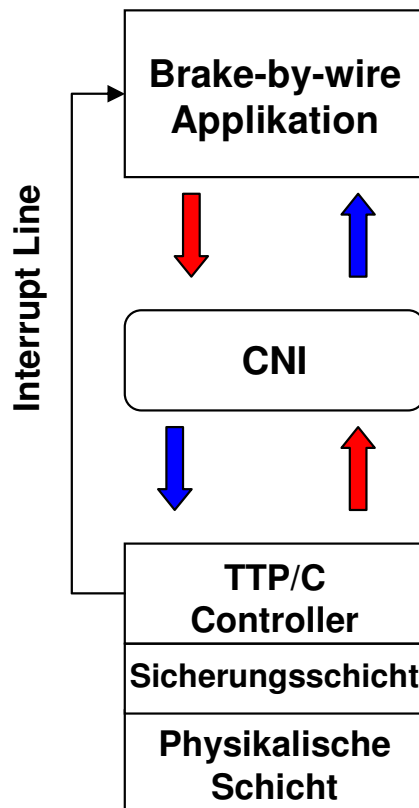
TP0 bis TP4 (TP4 ~ TCP)

ATM (B-ISDN)

Derzeit befindet sich eine Reihe von Transportprotokollen in der Entwicklung:

- MTP (Multicast Transport Protocol),
- PGM (Pragmatic General Multicast) und
- RMP (Reliable Multicast Protocol).

Brake-by-wire



Das CNI ist eine reine Datenschnittstelle, über die keine Steuersignale gesendet werden: sie wirkt wie eine „**temporäre Firewall**“. Der Hostprozessor hat keine Möglichkeit, das zeitliche Verhalten des Kommunikationssystems zu beeinflussen.

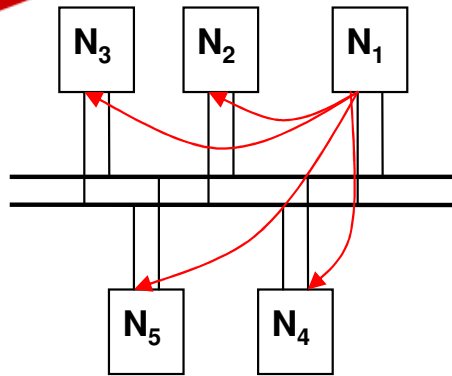
Übertragung der Daten:

1. Vom Host zum TTP/C Controller

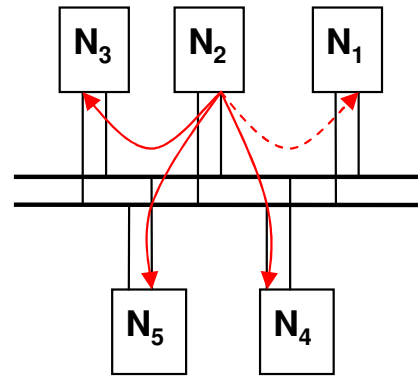
- Der Host kennt die **aktuelle globale Zeit** und weiß daher wann der Controller einen Lesezugriff auf dem CNI durchführt.
- Der Host muss es vermeiden, in das CNI zu schreiben, wenn der Controller einen Lesevorgang durchführt.

2. Vom Controller zum Host

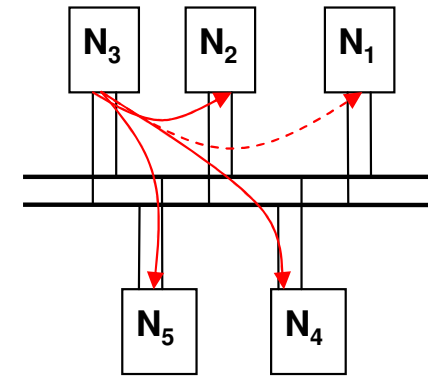
- Wird durch das MeDL geregelt.
- Der Non-Blocking-Write (NBW) Algorithmus koordiniert die konkurrierenden Zugriffe



N₁ sendet fehlerhafte Daten.

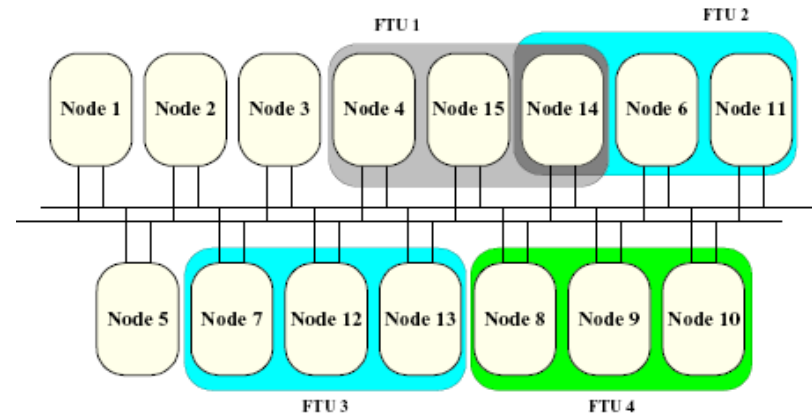
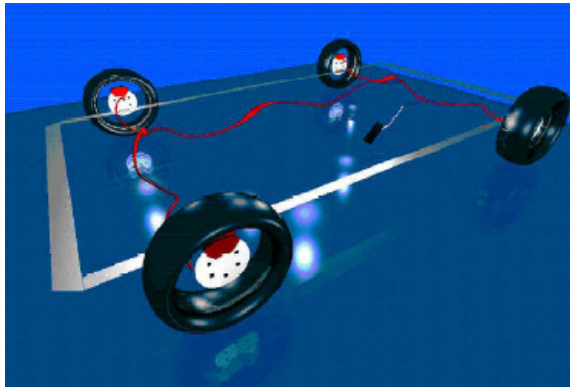


Alle Nodes erkennen den Fehler. N₁ wird aus der Mitgliedschaftsliste gelöscht.
N₂ ist der 1. Successor von N₁

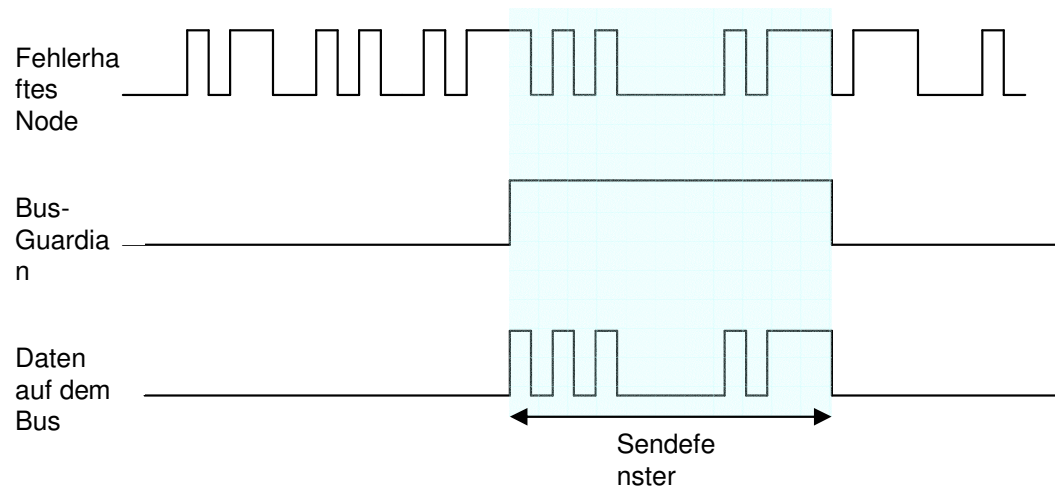


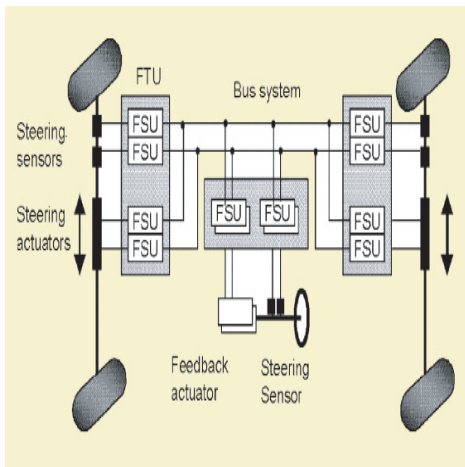
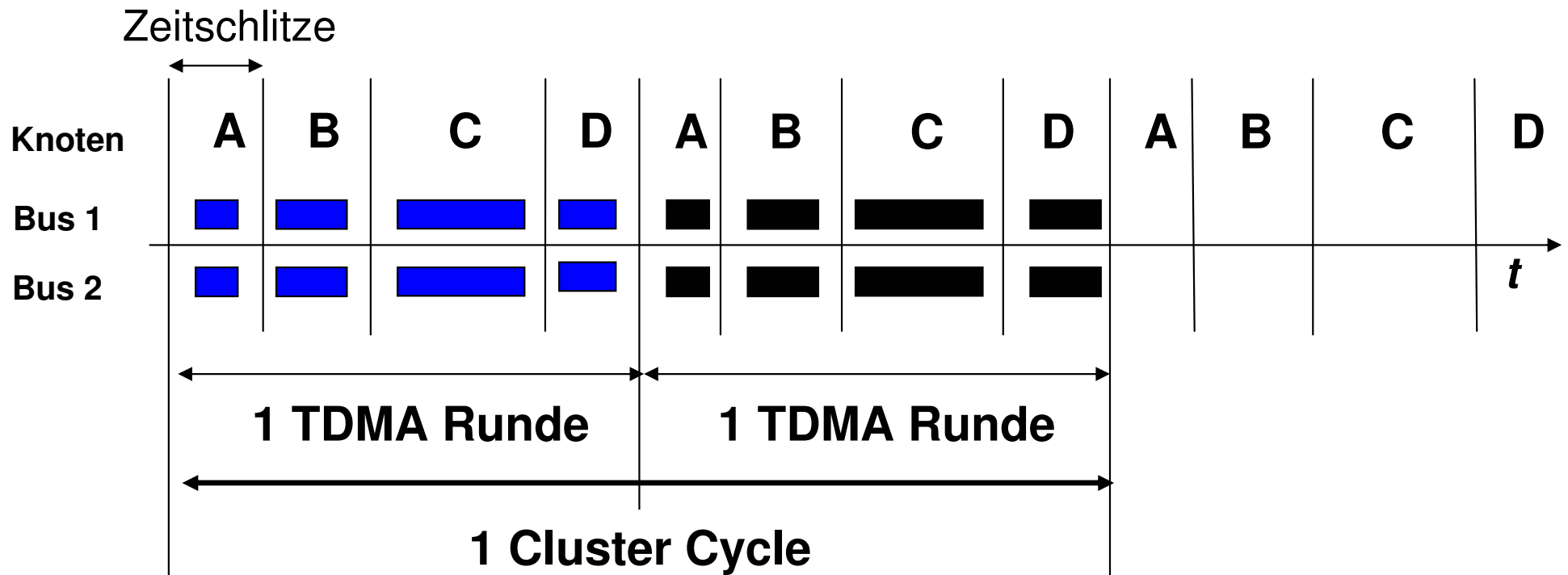
N₃ ist der 2. Successor von N₁.
Acknowledgment abgeschlossen.
N₁ erkennt seinen Membership-Verlust.

1. ein Datentransfer ist konsistent (Rahmen wird von allen Knoten entweder als Korrekt oder als fehlerhaft).
2. ein Node sendet keine Daten außerhalb seiner Schlitze (Slot).
3. ein Node sendet nie einen korrekten Frame außerhalb ihres Slot.
4. ein Node versteckt nie seine Identität, wenn es einen Rahmen sendet.



Danke für Ihre Aufmerksamkeit!





Leistung

TTP/C

- 25 MBit/sec Übertragungsrate (1 GBit/sec im Labor)
- Dateneffizienz:
95,8% bei 10 MBit/sec
78% bei 100 MBit/sec
→ gute Skalierbarkeit
- Datenübertragungsrate durch das Protokoll nicht beschränkt

FlexRay

- Voraussichtliche "Übertragungsrate für Controller der 1. Generation:
10 MBit/sec
- Dateneffizienz:
45,7% bei 10 MBit/sec
14,5% bei 100 MBit/sec

→ TTP/C ist in Geschwindigkeit, Dateneffizienz und Skalierbarkeit überlegen

Flexibilität

TTP/C

- vorausberechnete message descriptor list
- maximal ein slot pro Knoten
- variabel in der Länge einer Nachricht

FlexRay

- jeder Knoten lernt die MEDL mit der Zeit (Identifikation)
- umdisponieren der slots möglich
- statische slot grösse (im 1. Abschnitt der Runde)
- mehrere slots pro Knoten möglich

→ FlexRay ist flexibler

Zusammenfassung

- gestiegene Anforderungen im Bereich:
 - Sicherheit
 - Leistung
- herkömmliche Bussysteme sind nicht mehr ausreichend
- Zeitgesteuerte Protokolle bieten Vorteile gegenüber ereignisgesteuerten Protokollen
- TTP/C kann durch Sicherheit und Leistung überzeugen und wird z.B. im neuen Airbus A380 eingesetzt.