

Zugriffsverfahren zum Kommunikationsmedium

MAC

Netzwerk-Topologien

- Bus
- Ring
- Stern
- Baum
- Gitter
- Vollständig vernetzt

Bewertungskriterien:

Aufwand, Verzögerung, Toleranz von Ausfällen und Netzunterbrechungen

Was beeinträchtigt die Vorhersagbarkeit ?

Last

Fehler

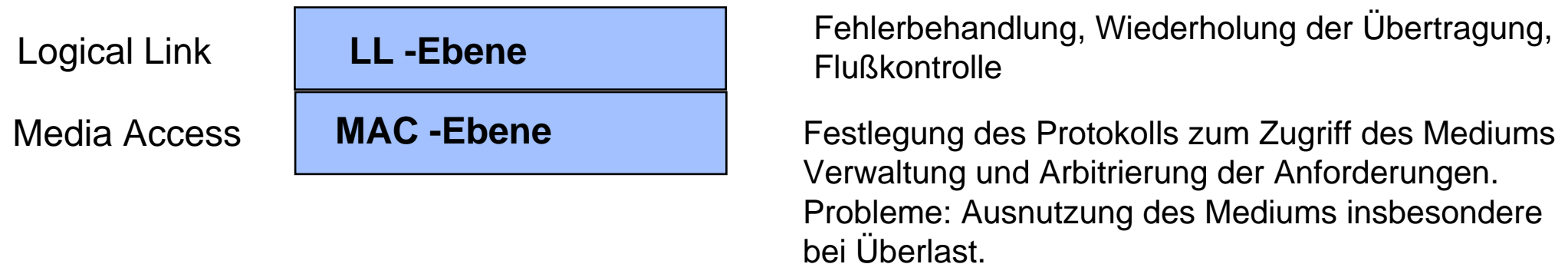


Arbitrierungskonflikte,
Netzwerk-Contention

Übertragungsfehler,
verlorengegangene
Nachrichten

Die Sicherungsschicht

Aufgaben: Transfer von Datenblöcken
Flußkontrolle
Fehlerbehandlung
Nachrichtenerholung



Diskussion:

Wann kann eine Nachricht gesendet werden?

Wie lange kann eine hoch priorisierte Nachricht von einer niedriger priorisierten Nachricht verzögert werden?

Wie verhält sich das Netzwerk unter Last?

Wie verhält sich das Netzwerk unter Fehlern?

MAC-protocols

Kontrollierter Zugriff

Wahlfreier Zugriff

Collision avoidance

Collision resolution

Reservation-based

Token-based

Time-based

Master-Slave

Priority-based

probabilistic

dynamic

static

ATM

TDMA:

TTP,
Maruti

Token-Ring
Token-Bus

Timed
Token
Protocol

CSMA/CA :
Collision Avoidance

IEEE 802.11
P-persistent CSMA

LON, VTCSMA

ProfiBus DP
FIP
CAN-Open

CSMA/CA :
Consistent Arbitration

CAN

CSMA/CD :
Carrier Sense Multiple Access /
Collision Detection

Ethernet

Controlled Access by Collision Exclusion:

Master/Slave

all control information in one place
maximum of control
easy to change

Global Time

Easy temporal co-ordination
Minimal communication overhead

Token-based

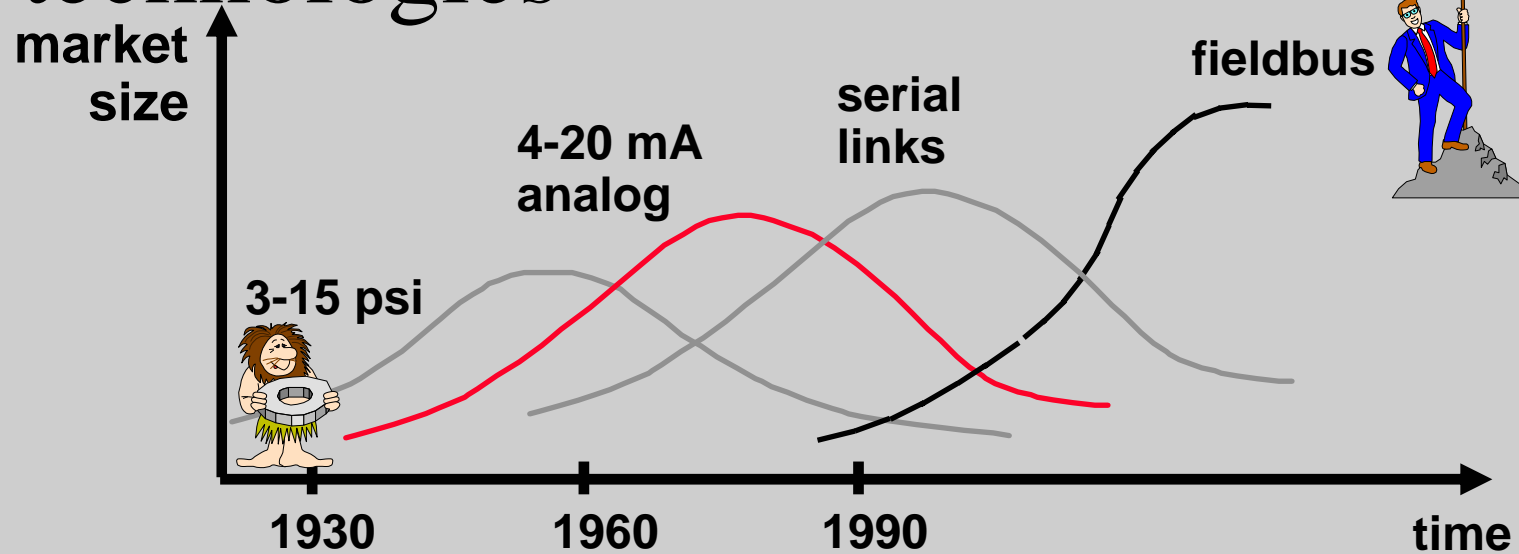
Decentralized mechanism
Integration of critical and non-critical messages

Beispiele aus der industriellen Automatisierung

Your competitors are moving:

Fieldbus technology is advancing

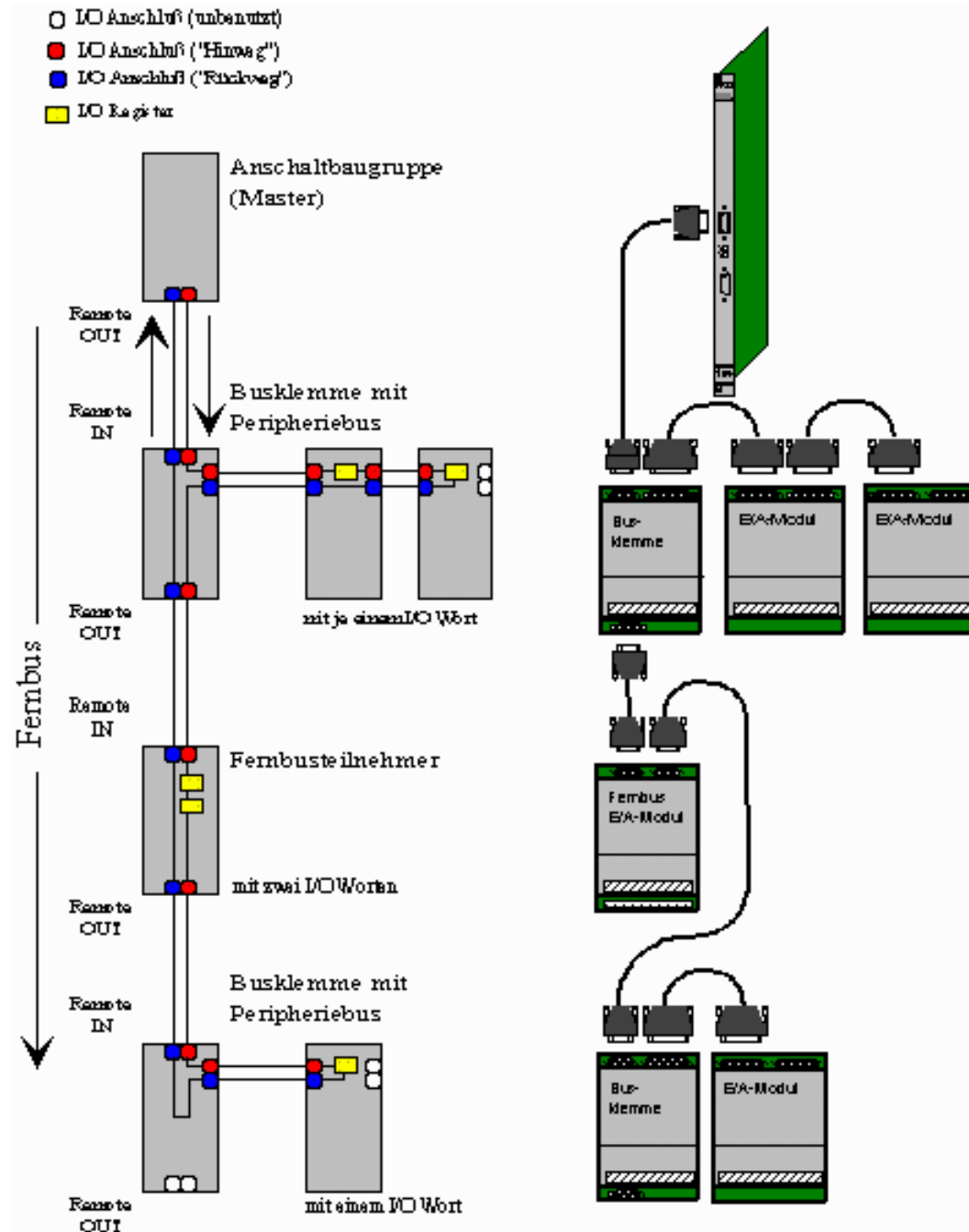
- Life cycles of interconnection technologies



Interbus-S

Phoenix Contact

Prinzip:
synchrones,
erweiterbares
Schieberegister

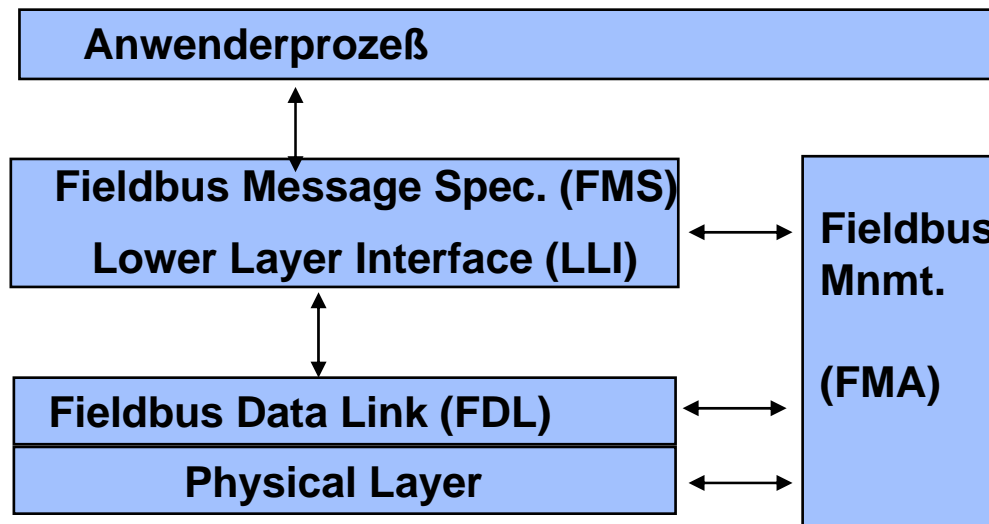


Profibus

Profibus, Basiert auf einem BMFT-Verbundprojekt zwischen Hochschulen und Firmen der Automatisierungstechnik

genormt in: DIN 19245, Teil 1 (für OSI-Schicht 1 und 2) DIN 19245, Teil 2 (für OSI-Schicht 7)

FMS orientiert an
MMS (Manufacturing
Message Specs.)
ISO 9506



Architekturmodell

Generelle Architektur eines Profibus Netzwerks

Fabrikebene

LAN

Bus-Zyklus
< 1 sek.

Profibus-FMS

Zellenebene

Bus-Zyklus
< 100 ms

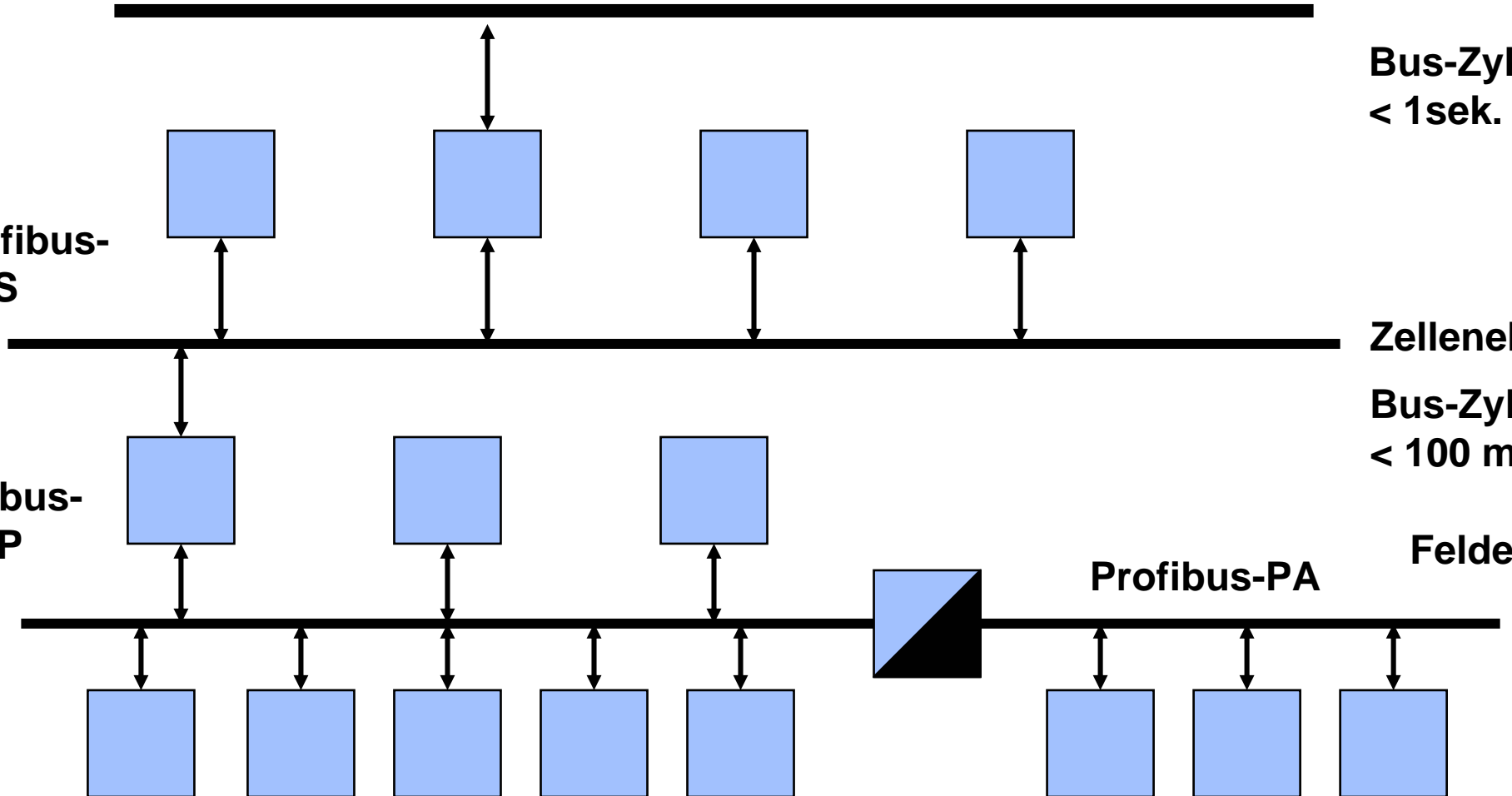
Profibus-DP

Feldebene

Profibus-PA

Bus-Zyklus
< 10 ms

Sensoren/ Aktoren



EN 50170 Volume 2

General Purpose Automation	Factory Automation	Process Automation
PROFIBUS-FMS RS 485 / FO	PROFIBUS-DP RS 485 / FO	PROFIBUS-PA IEC 1158-2
Universal	Fast	Application Oriented
<ul style="list-style-type: none">- Large variety of applications- Multi-master communication	<ul style="list-style-type: none">- Plug and play- Efficient and cost effective	<ul style="list-style-type: none">- Powers over the bus- Intrinsic safety

**Device Profiles
Application Profiles**

Basic PROFIBUS-DP Functions

Transmission Technology

- RS-485, twisted pair, two wire cable or fibre optics
- Baud rates from 9.6 kbit/sec to 12 Mbit/sec

Bus access

- Token passing procedure between masters and master-slave procedure for slaves
- Mono-master or multi-master systems possible
- Master and slave devices, maximum of 126 stations on one bus

Communication

- Peer-to-peer (user data transmission) or Multicast (control commands)
- Cyclic master-slave user data transmission and acyclic master-master data transmission

Operating modes

- Operate: Cyclic transmission of input and output data
- Clear: Inputs are read, and outputs are held in fail-safe status
- Stop: Only master-master data transmission is possible

Synchronization

- Control commands permit synchronization of the inputs and outputs
- Sync mode: Outputs are synchronized
- Freeze mode: Inputs are synchronized

Basic PROFIBUS-DP Functions (cont.)

Functionality

- Cyclic user data transmission between DP master and DP slave(s)
- Dynamic activation or deactivation of individual DP slaves
- Check of DP slave configuration
- Powerful diagnostic functions, 3 hierarchical levels of diagnostic messages
- Synchronization of the inputs and/or the outputs
- Address assignment for the DP slaves over the bus
- Configuration of the DP master (DPM1) over the bus
- Maximum of 244 bytes of input and output data per DP slave

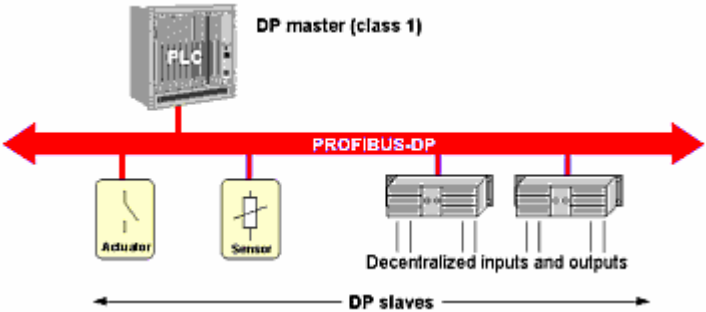
Security and protection functions

- All messages are transmitted with Hamming distance $HD=4$
- Watchdog timer at the DP slave
- Access protection for the inputs/outputs of the DP slaves
- Monitoring of user data transmission with configurable monitoring timer at the master

Types of devices

- Class-2 DP master (DPM2): programming/configuration/diagnostic devices
- Class-1 DP master (DPM1): central programmable controllers such as PLCs, PCs, etc.
- DP slave: device with binary or analog inputs/outputs, drives, valves, etc.

Mono-Master



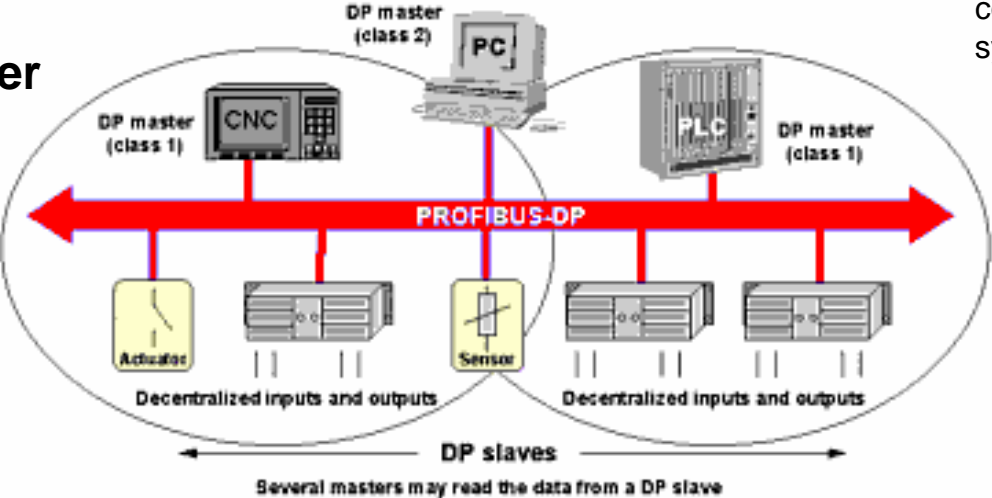
DP Master Class 1 (DPM1)

The class-1 DP master is a central controller which exchanges information with the decentralized stations (i.e., DP slaves) within a specified message cycle. Typical master devices include programmable controllers (PLCs) and PC or VME systems.

DP Master Class 2 (DPM2)

Class-2 DP masters are programmers, configuration devices or operator panels. They are used during commissioning for configuration of the DP system or for operation and monitoring purposes.

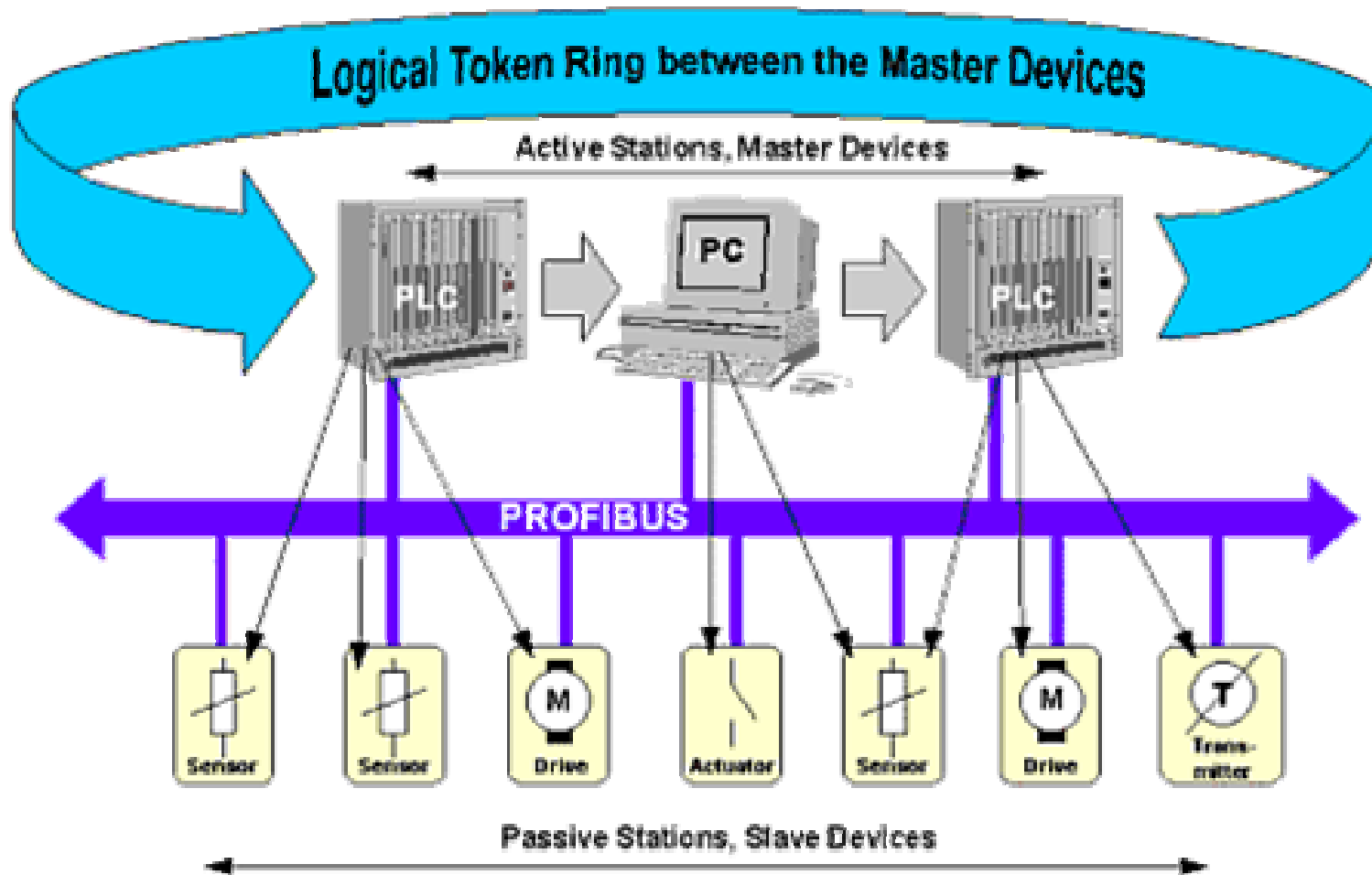
Multi-Master



DP Slave

A DP slave is a peripheral device (I/O devices, drives, HMI, valves, etc.) which collects input information and sends output information to the controller. There are also devices which supply only input or only output information.

Bus Arbitrierung im Profibus



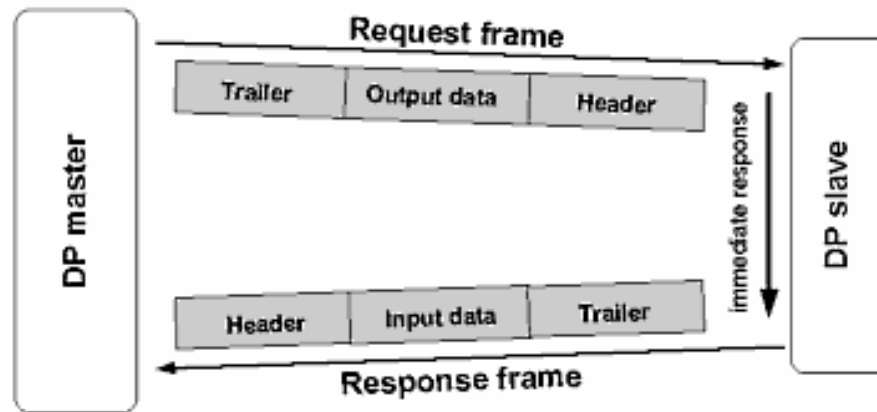
Datentransfer zwischen Master und Slave

Der Datentransfer zwischen dem DPM1 Master und den DP Slaves ist in 3 Phasen unterteilt:

Parametrisierung

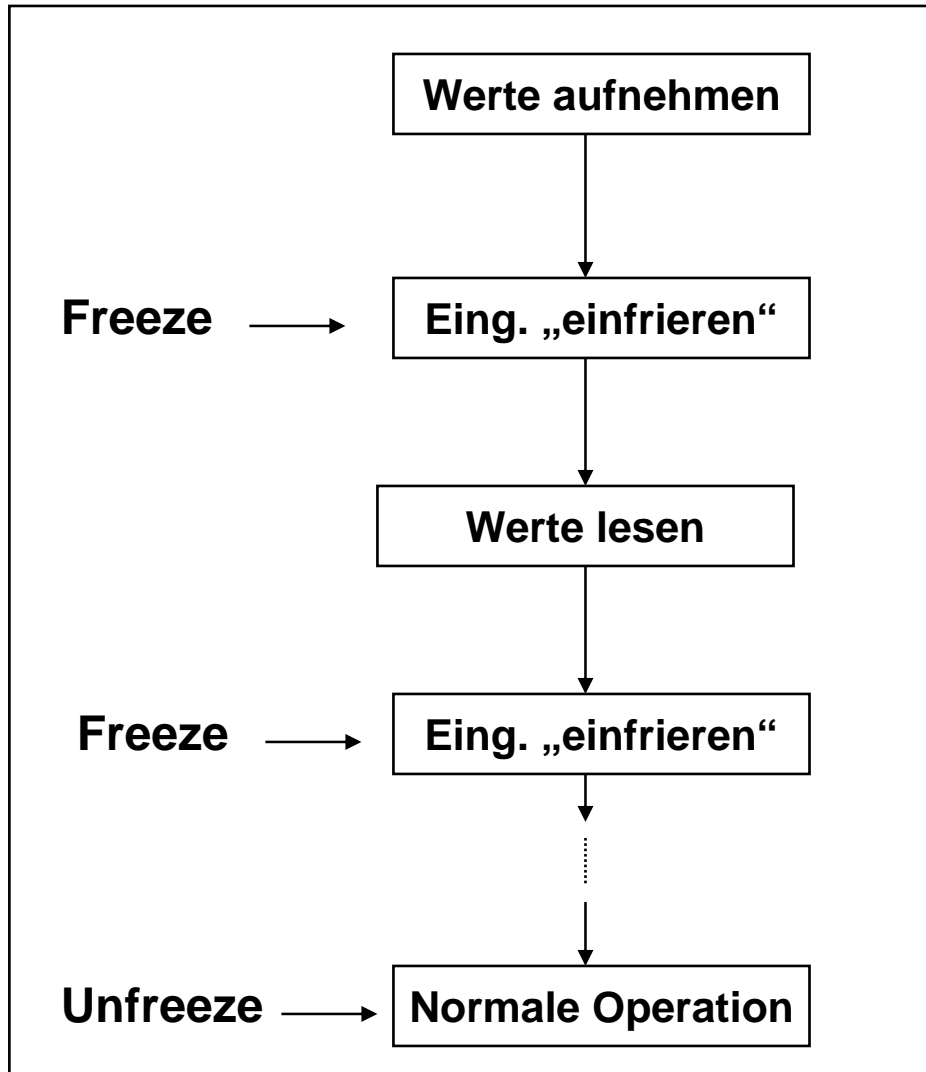
Konfiguration

Datentransfer

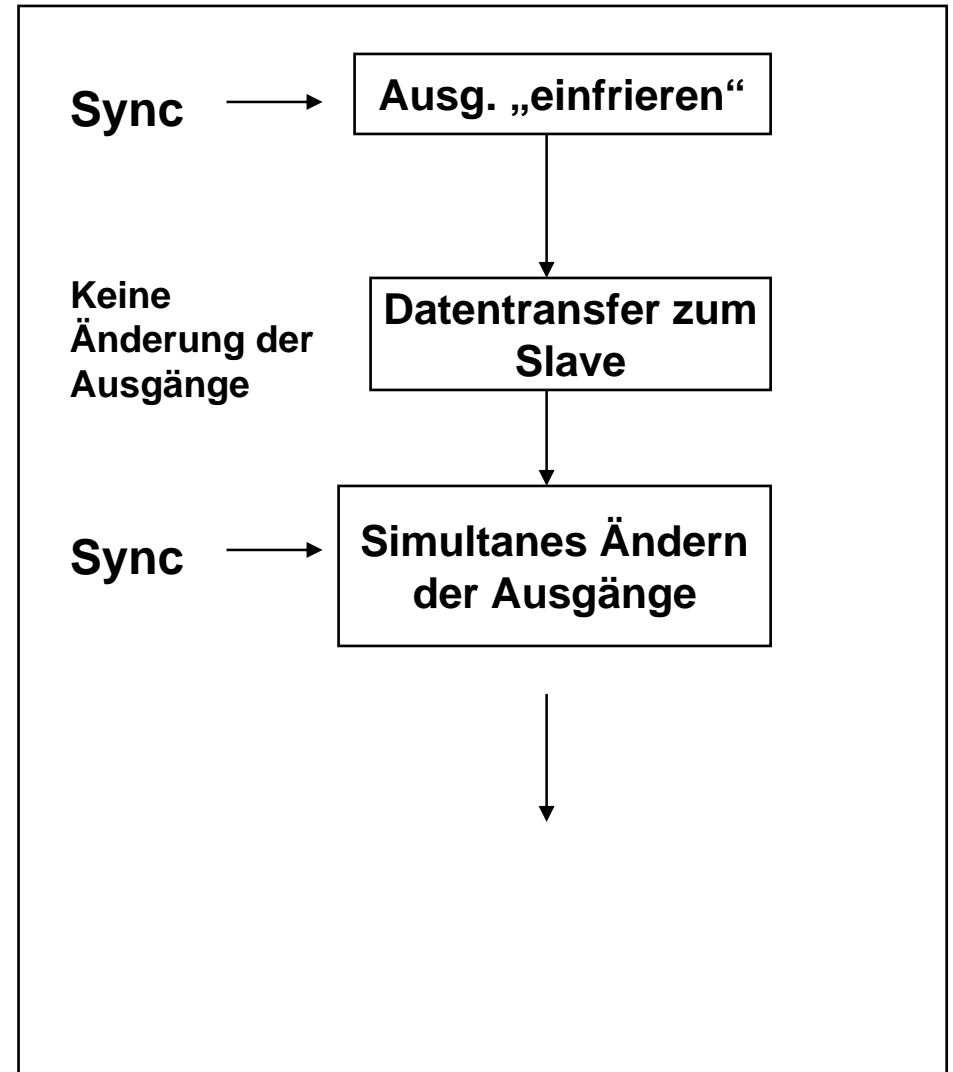


Synchronisierte Aktivitäten der Slaves

Synchronisierte Eingabe



Synchronisiertes Ausgabe



Probleme bei Master-Slave Verfahren?

Master ist der „single point of failure“

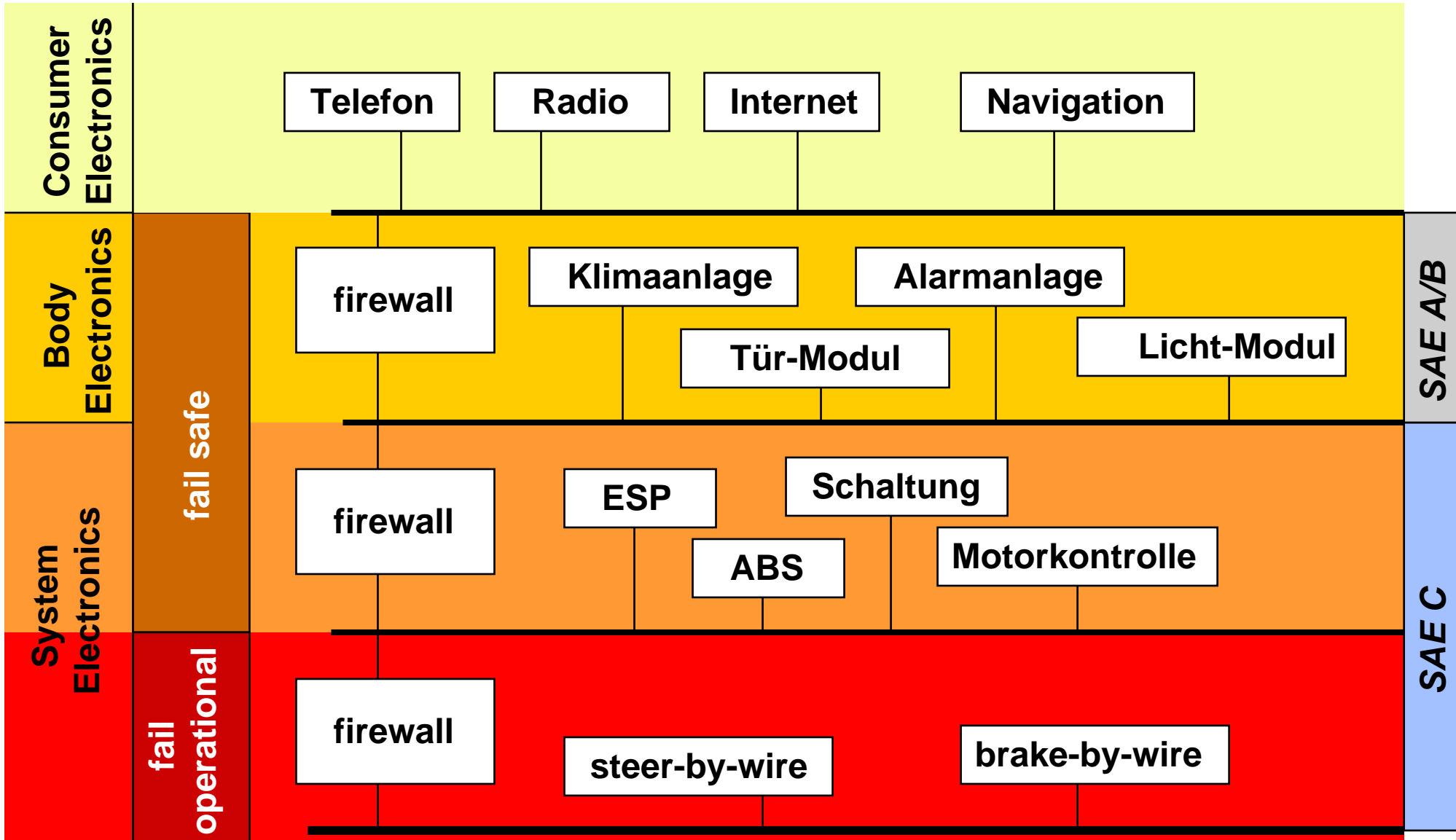
Mehr Nachrichten

Keine unabhängige spontane Erkennung von Ereignissen

Geeignet für zyklische, a priori geplante Kommunikation

Ebenen der Kommunikation im Auto

Nach: T. Führer, B. Müller, W. Dieterle, F. Hartwich, R. Hugel, M. Walther:
 „Time Triggered Communication on CAN“



Time Triggred Protocol (TTP)

TU Wien: H. Kopetz (Lit. Einführung)

Ziele:

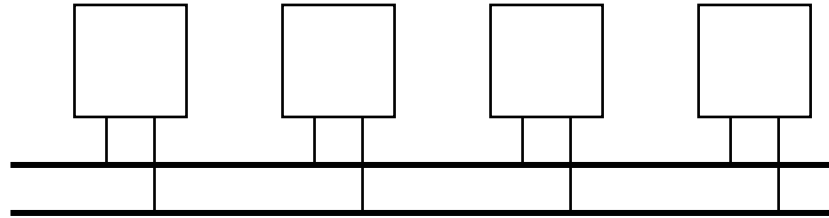
- **Vorhersagbare Verzögerung**
- **Kein Einzelfehler soll zu einem Ausfall der Kommunikation führen**
- **Fehlertoleranz**
 - Fehlererkennung auf Sender- und Empfängerseite
 - Vorwärtsfehlerbehebung
 - Behandlung temporärer globaler Fehler (Black-out)
 - Verteiltes Redundanzmanagement
- **Uhrensynchronisation**
- **Membership-Dienst (Basis für atomaren Multicast)**
- **Unterstützung für schnelle, konsistente Modusänderung**
- **Minimaler Protokolloverhead**
- **Flexibilität ohne Verlust der Vorhersagbarkeit**

Entwurfsprinzipien

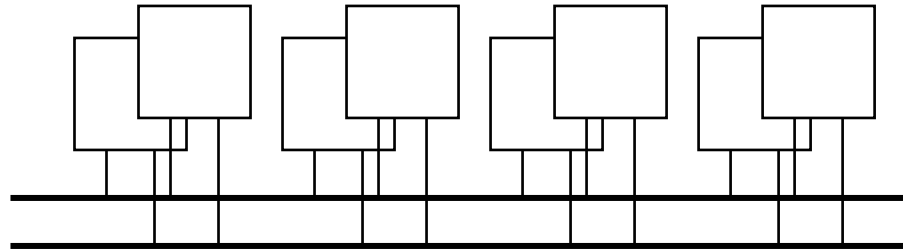
- **Nutzung von a priori Wissen (statisches Planen)**
- **Implizite Flußkontrolle**
- **Fail silence**
- **Kontinuierliche Zustandsübereinstimmung**

Fehlertolerante Netzwerk-Konfigurationen

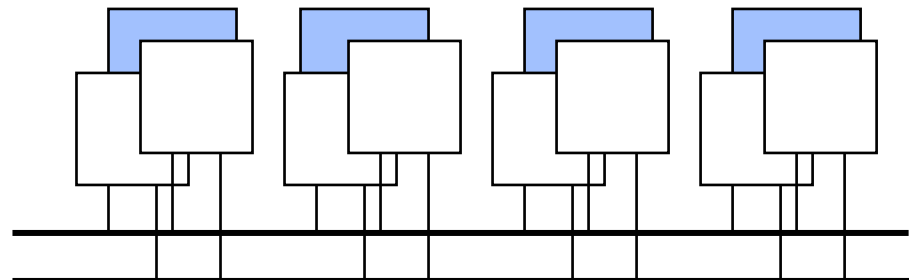
Class 1:
1 Knoten/FTU
2 Frames/FTU



Class 2:
2 aktive Knoten/FTU
2 Frames/FTU



Class 3:
2 aktive Knoten/FTU
4 Frames/FTU



Class 4:
2 aktive Knoten/FTU
+ 1 Ersatz/FTU
4 Frames/FTU



Komponentenredundanz + Zeitredundanz

Fehlerart

Fehlerhäufigkeit

Permanenter Knotenfehler

$10^{-6}/h$

Permanenter Kanalfehler

$10^{-5}/h$

Transienter Knotenfehler

$10^{-4}/h$

Transienter Kanalfehler

$10^{-3}/h$

Wie ist das Verhältnis: Fehlerhafte Nachrichten / Gesamtzahl der Nachrichten ?

Fehlerart

Class 1

Class 2

Class 3

Class 4

Perm.Knotenfehler

0

1

1

2

Perm. Komm.Fehler

1

1

1

1

Trans. Knotenfehler

0

1/Rec.Interv.

1/Rec. Interv.

1/TDMA-Runde

Trans. Komm.Fehler

1 von 2

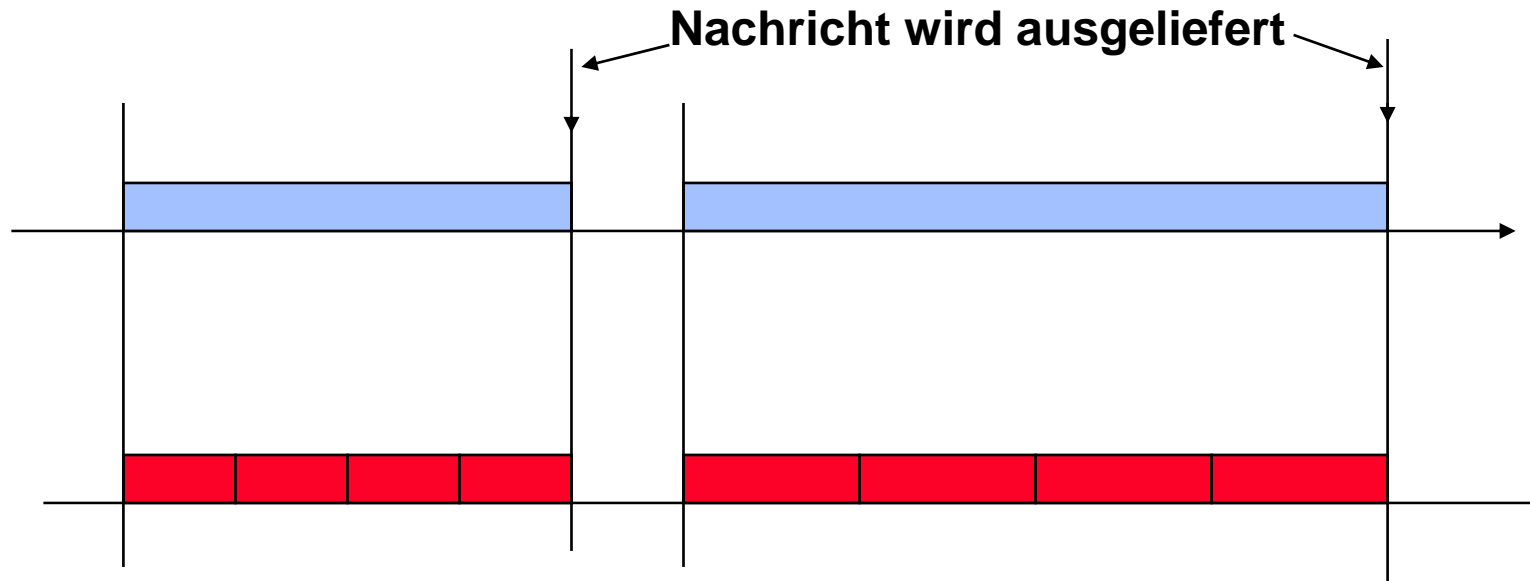
1 von 2

3 von 4

3 von 4

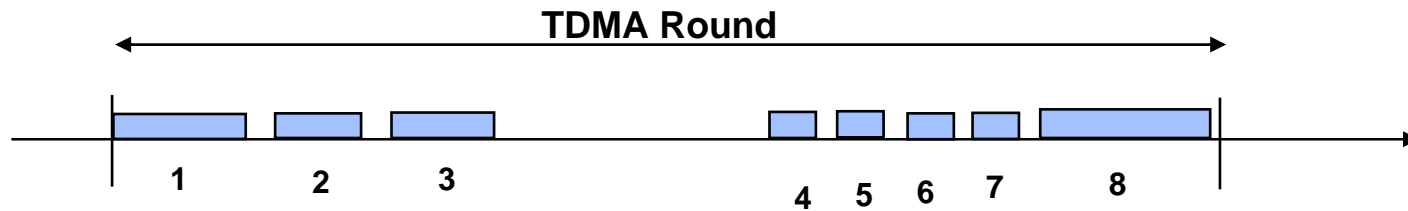
„Vorwärts“-Fehlerbehebung

Vorhersagbarkeit und Fehlertoleranz müssen integriert werden.



Nutzen von Zeitredundanz → Mehrfachsenden einer Nachricht

Nutzung von a priori Wissen: Off-line Scheduling



	Attribute					
	Zeit	Adresse	D	L	I	A
1						
2						
3						
4						
5						
6						
7						
8						

Message
Description
List

MEDL

Zeit: enthält den Zeitpunkt an dem die Nachricht gesendet werden muss

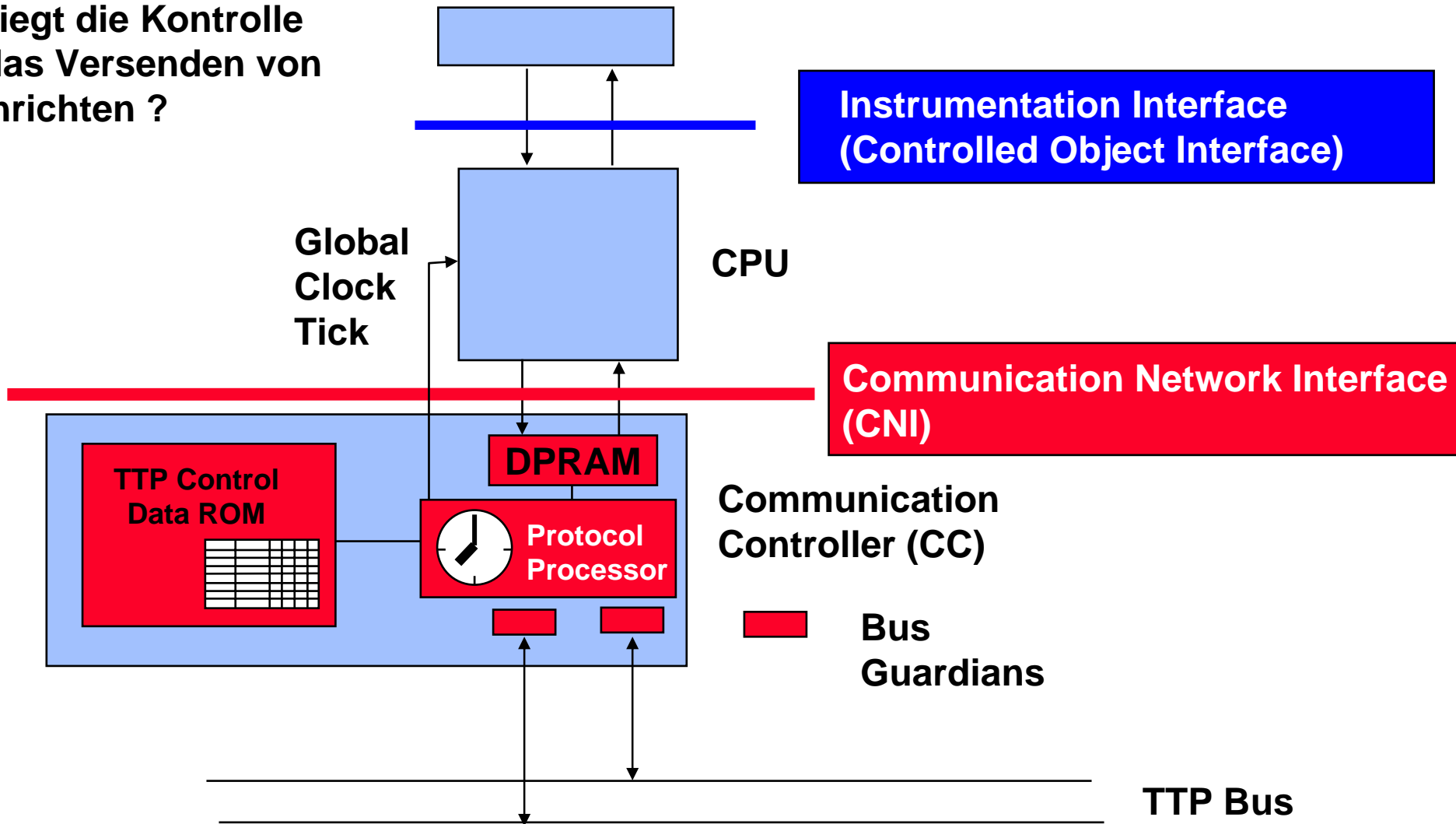
Adresse: Ist die lokale Adresse wo die zu übertragenden Daten gespeichert sind

D: Direction Input oder Output
L: Länge der Nachricht
I: Initialisierung oder normale Nachricht
A: "Additional" Parameter Feld

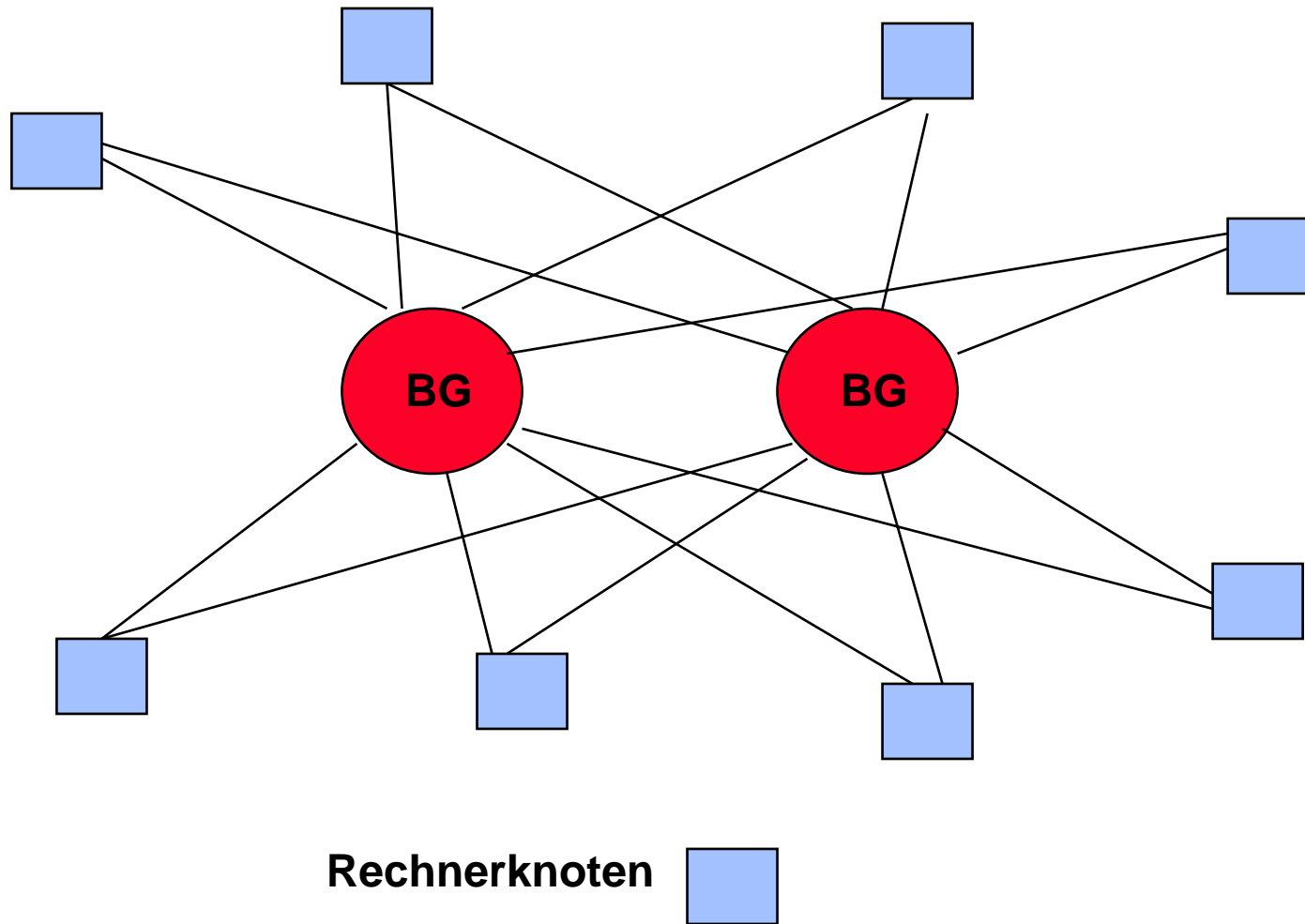
TDMA Round (Cluster Zyklus): Jede FTU hat mindestens ein Mal gesendet

Fail silence und strikte Einhaltung der Sendezeit

Wo liegt die Kontrolle für das Versenden von Nachrichten ?



Auslagern des Bus-Guardians: Stern-Topologie



Format eines TTP Telegramms

89 Bit



Datenfeld

CRC Feld

Start Of Frame
(SOF)

Kontrollfeld

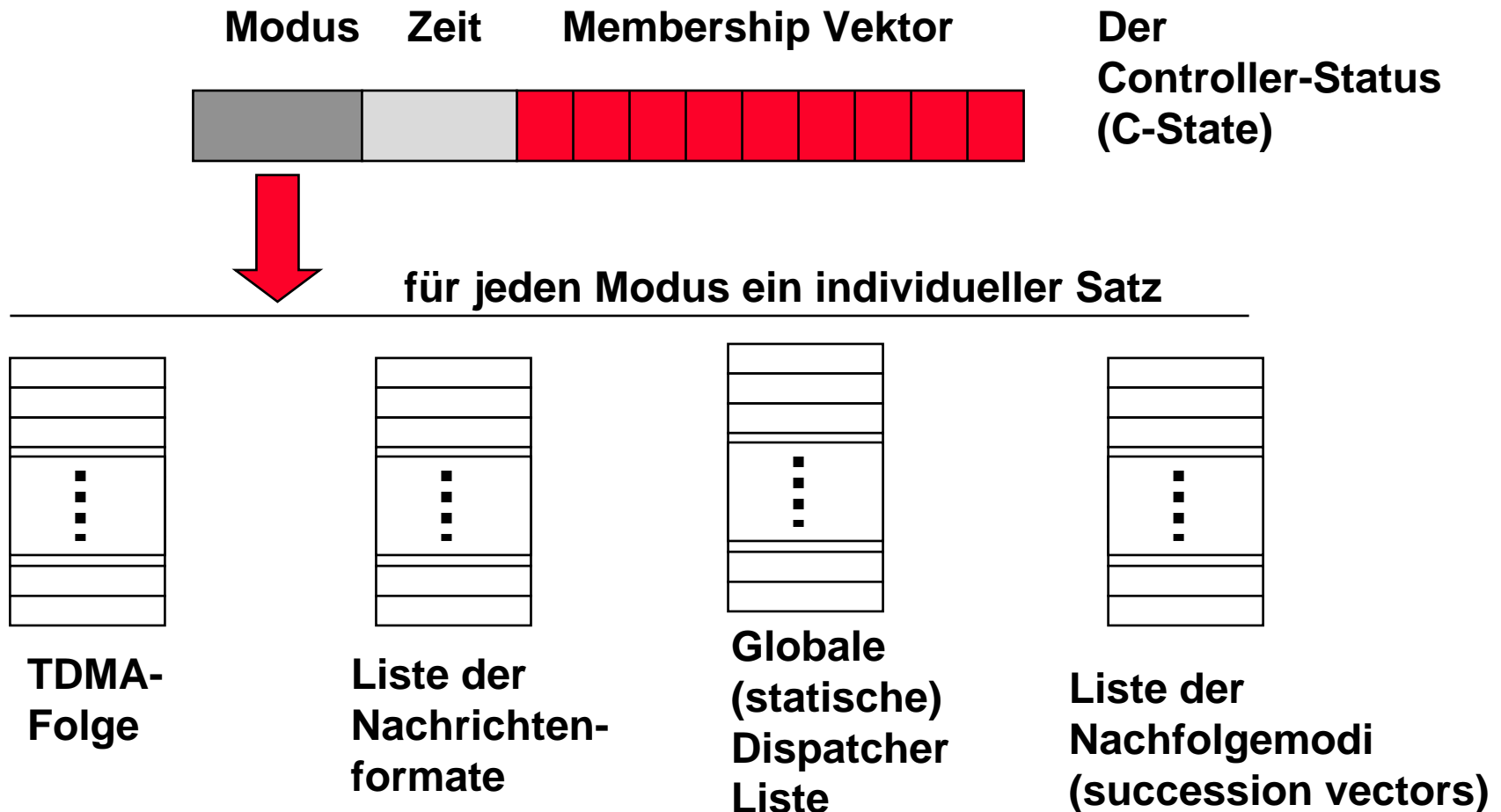


Modus Bits

InitialisierungsBit unterscheidet normale Frames und Initialisierungsframes

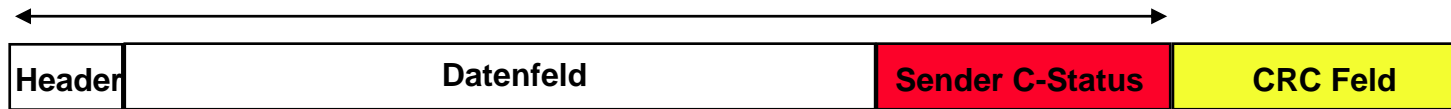
MFM Codierung. Keine datenabhängige Telegrammlänge

Kontinuierliche Überprüfung des globalen Zustands



Kontinuierliche Überprüfung des globalen Zustands durch CRC

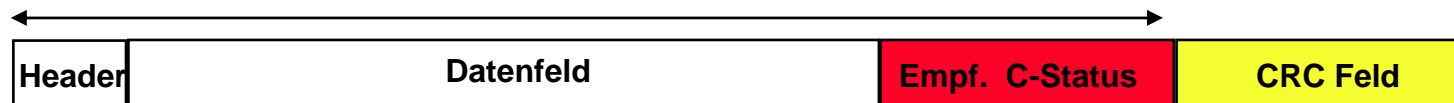
CRC-Generierung im Sender



Nachricht



CRC-Generierung im Empfänger



Behandlung von Modusänderungen

Zu jedem Zeitpunkt sind alle Knoten in einem spezifischen Modus.
→ Konsens notwendig

Modusänderung:

FTU signalisiert Modusänderung im Kontrollfeld durch Angabe der Position des „Succession Vectors“ in der entsprechenden Tabelle.

→ Flexibilität: Succession Vektor kann geändert werden

Kritische Funktionen:

- **Initialisierung**
- **Mitgliedschaft (Membership)**
- **Black-out Behandlung**

Redundanzmanagement und Initialisierung

Jeder Knoten hat einen eindeutigen Namen und dadurch eine Position in der TDMA Runde

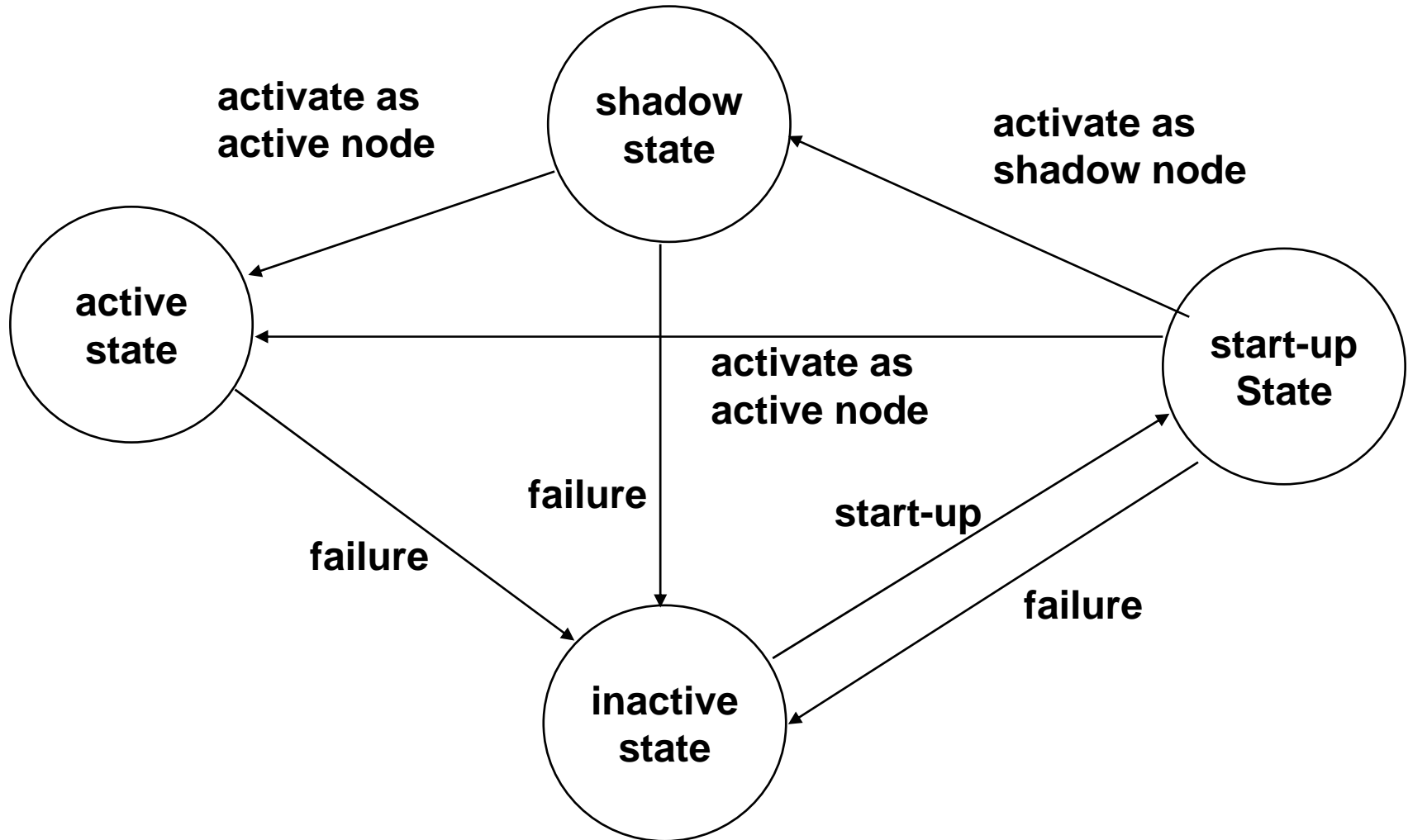
Einige ausgewählte Knoten können einen Initialisierungs-Frame senden.

Initialisierungsframes enthalten den Zustand des Gesamtsystems aus Sicht des Senders „im Klartext“.

Das längste Intervall zwischen zwei I-Frames bestimmt, wie lange ein ausgefallener Knoten auf seine Reintegration warten muss.

Redundanzmanagement und Initialisierung

Lokale Zustände einer FTU:



Redundanzmanagement und Initialisierung

Reset local clock
Monitoring des Busses für I_1 ($I_1 >$ längste TDMA Runde)
Init-Frame wird in diesem Zeitraum gesendet, wenn das Netz initialisiert ist

Falls Nachrichtenverkehr, warte auf I-Frame

Falls kein Nachrichtenverkehr, warte spez. Zeit I_2
(I_2 ist Knotenspez. Verzögerung zur Vermeidung von Kollisionen)

Nach I_2 sende I-Frame mit C-Status im Modus „Init“

Membership Service

Sender setzt Membership Bit (MB) auf 1

Alle Receiver setzen MB auf 1

Wenn kein korrekter Frame empfangen wird, setzen Receiver MB = 0 direkt nach dem TDMA-Slot

Vor dem **Membership-Punkt (a priori bekannter Zeitpunkt, an dem die FTU eine Nachricht sendet) überprüft der sendende Knoten ob er (noch) Mitglied ist**

Membership Service

Ein Knoten ist Mitglied, wenn:

- 1. interne Überprüfung o.k.**
- 2. mindestens einer der Frames die gesendet wurden von einer der nachfolgenden FTUs im ack-Feld bestätigt wurden (d.h. phys.Verbindung ist o.k.)**
- 3. die Anzahl korrekter Frames, welche die FTU in der letzten TDMA-Runde akzeptiert hat größer ist als die Anzahl verworfener Frames.**

Ist dies nicht der Fall, ist der lokale C-Status wahrscheinlich nicht in Übereinstimmung mit den anderen Knoten und der Knoten verliert die Gruppenmitgliedschaft. Dies verhindert die Bildung von Cliquen, die eine jeweils unterschiedliche Sicht der Gruppe haben.

Black-out Behandlung

Black-out bezeichnet eine globale Störung z.B. die Störung des Kommunikationsmediums durch externe elektromagnetische Felder.

Black-out Erkennung:

Knoten überwacht kontinuierlich das Membership-Feld

Bei „starkem“ Rückgang der Membership

-> Change Mode nach Black-out Behandlung

Black-out Mode: Knoten senden nur I-Frames und beobachten den Bus

Wenn die externe Störung verschwindet, stabilisiert sich die Mitgliedschaft

Rückkehr in den Normal-Modus

Diskussion

Synchronität (Jitter, Steadyness, Thightness)

Automatische Uhrensynchronisation

Fehlermaskierung

Monopolisierungs- (Babbling Idiot-) Fehler werden vermieden

Replica Determinismus

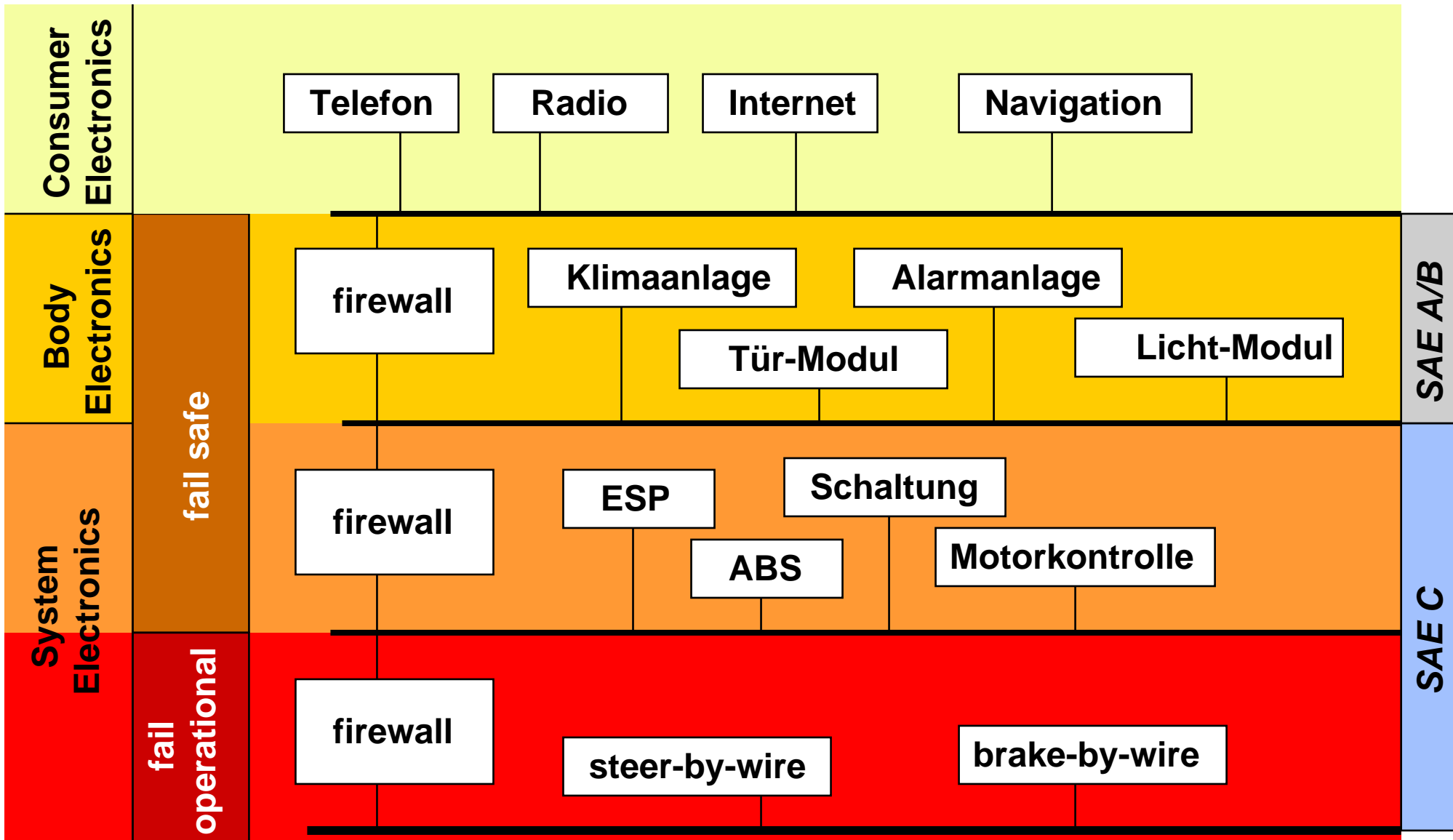
Zusammensetzbarkeit und Erweiterbarkeit

Zusammenfassung TTP

- Die Protokollausführung wird durch den Fortschritt der globalen Zeit initiiert. Der Zeitpunkt, zu dem eine Nachricht verschickt wird ist allen Empfängern a priori bekannt.
- Die maximale Ausführungszeit entspricht ungefähr der mittleren Ausführungszeit.
- Fehlererkennung liegt beim Empfänger. Sie basiert auf seiner a priori Kenntnis darüber, welche Nachricht er zu welchem Zeitpunkt erwarten kann.
- Das Protokoll ist unidirektional. Keine Quittungen erforderlich.
- Implizite Flußkontrolle ist notwendig.
- Es können keine Zugriffskonflikte auftreten.

Ebenen der Kommunikation im Auto

Nach: T. Führer, B. Müller, W. Dieterle, F. Hartwich, R. Hugel, M. Walther:
 „Time Triggered Communication on CAN“



LIN Local Interconnect Network

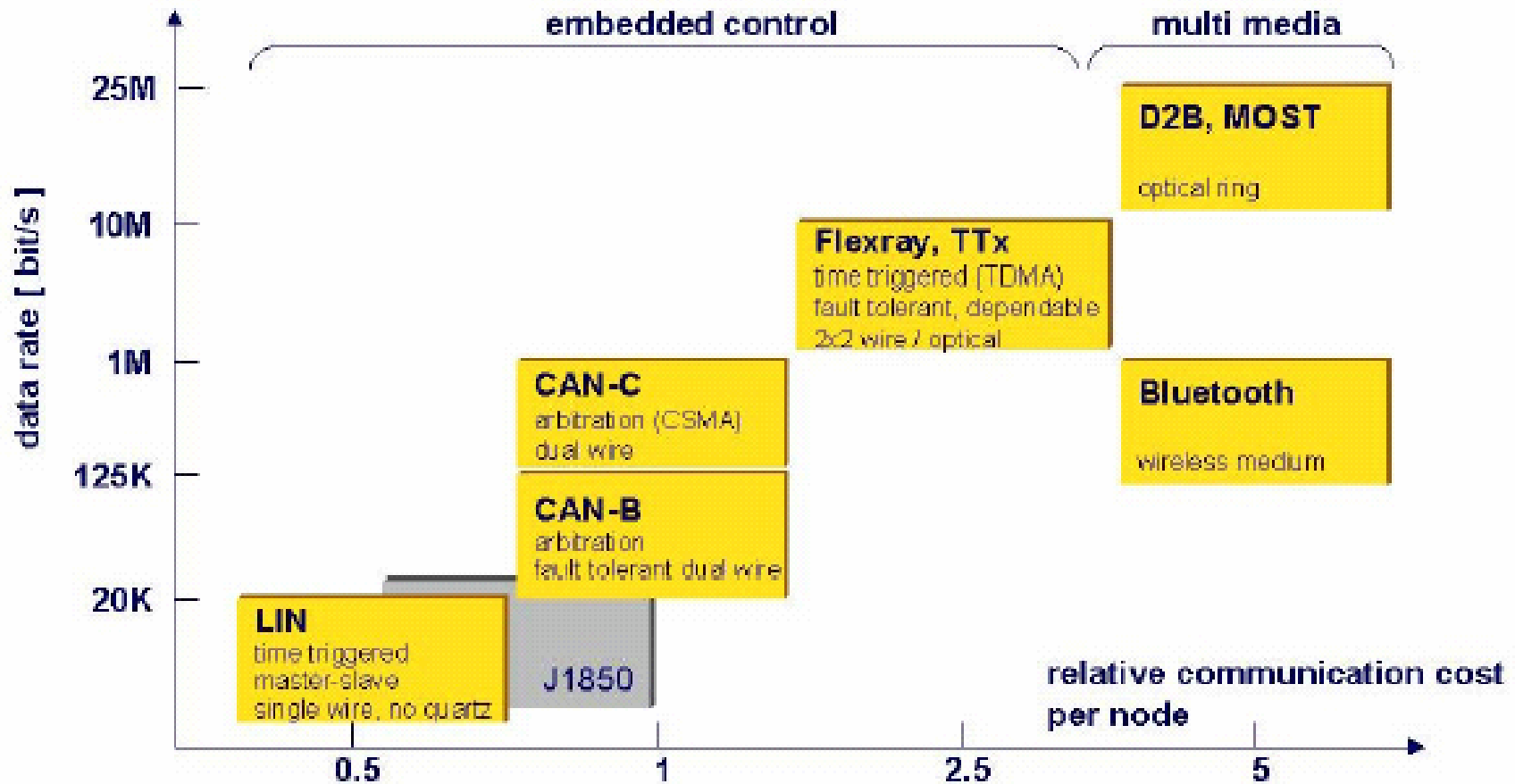


Figure 1: Major Network Protocols in Vehicles

Probleme:

- Exakte globale Zeit wird in jedem Knoten benötigt.
- Reservierte Zeitschlitzte können nicht anderweitig genutzt werden, auch wenn keine Nachrichten gesendet werden müssen.

Lösungen für weniger kritische, einfachere Sensor-Aktor Netzwerke:

LIN (Local Interconnect Network)

TTP/A (Time Triggered Protocol for SAE class A applications)

- **Master/Slave Protokolle**
- **Geringere Zuverlässigkeitsanforderungen**
- **Knoten können mit freilaufenden Oszillatoren betrieben werden**
- **Physische „Ein-Draht-Verbindung“ (asynch. serielle Schnittstelle)**
- **Geringere Anforderungen an die Übertragungsbandbreite**
- **Geringe Kosten**

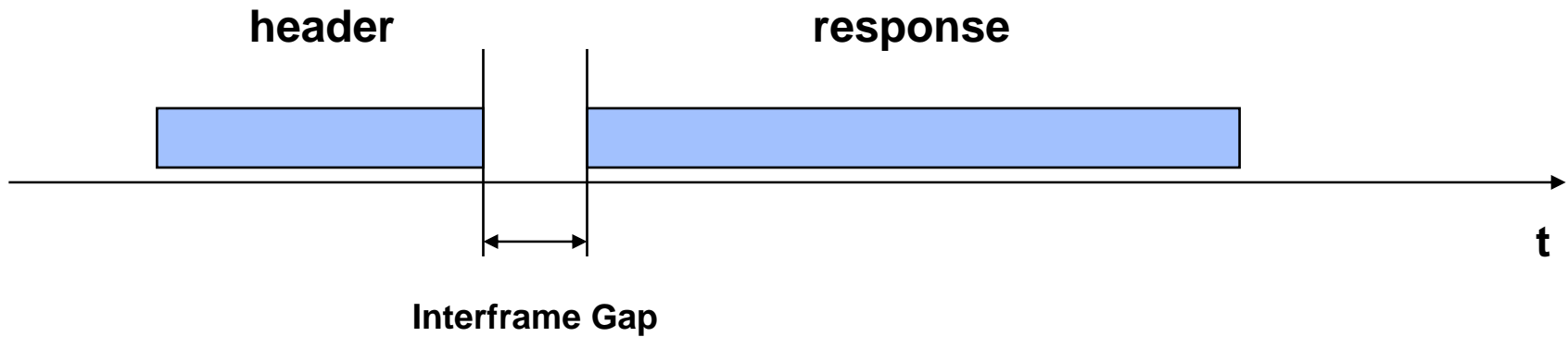
Transmission speed up to	LIN	TTP/A
20 kbits/second	ISO 9141 (ISO-K)	ISO 9141 (ISO-K)
1 Mbit/second	not specified	RS 485 or CAN
above 1 Mbit/second	not specified	fiber optics

Table 4: Transmission speed of LIN and TTP/A

The main properties of the LIN bus are:

- . single-master / multiple-slave concept**
- . low cost silicon implementation based on common UART/SCI interface hardware, an equivalent in software, or as pure state machine.**
- . self synchronization without quartz or ceramics resonator in the slave nodes**
- . guarantee of latency times for signal transmission**
- . low cost single-wire implementation**
- . speed up to 20kbit/s.**

Master-Slave Kommunikation

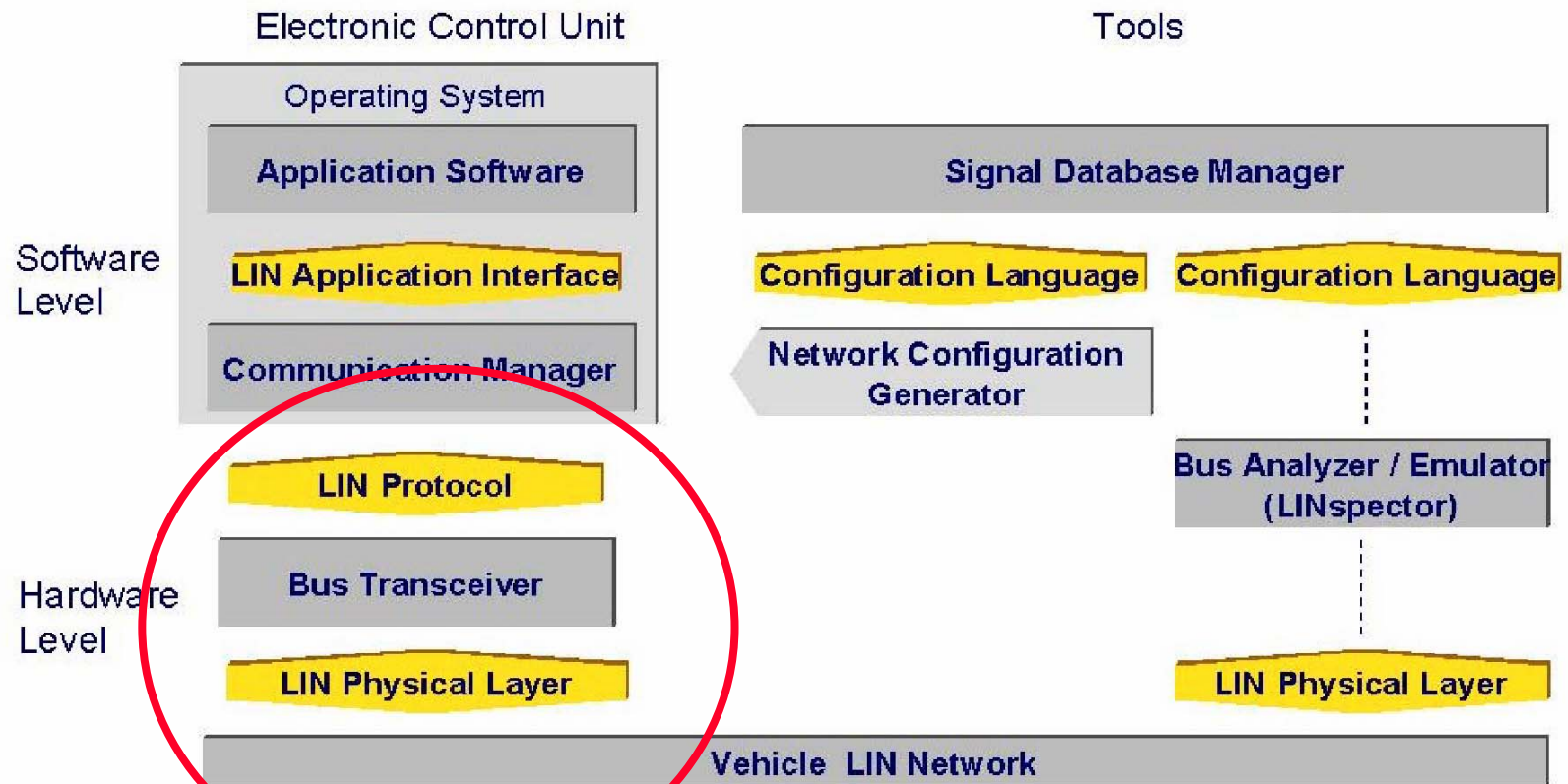


Header:

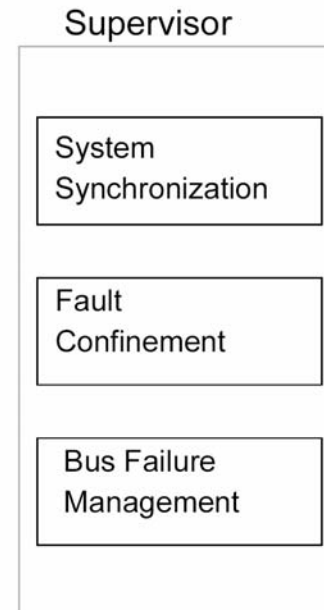
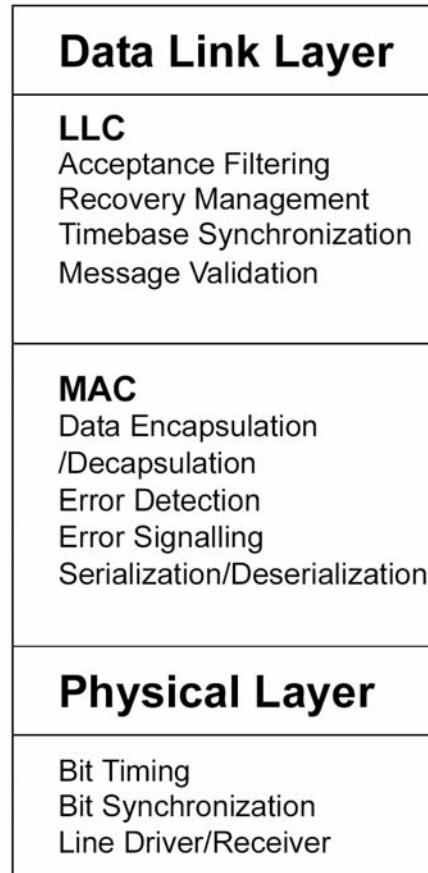
- dient der Synchronisation der Slaves
- legt die Reihenfolge und Länge der Felder im Daten-Telegramm fest

LIN (Local Interconnect Network)

LIN Specification Package, Revision 1.2, Nov. 17, 2000



Ebenen im OSI Referenzmodell

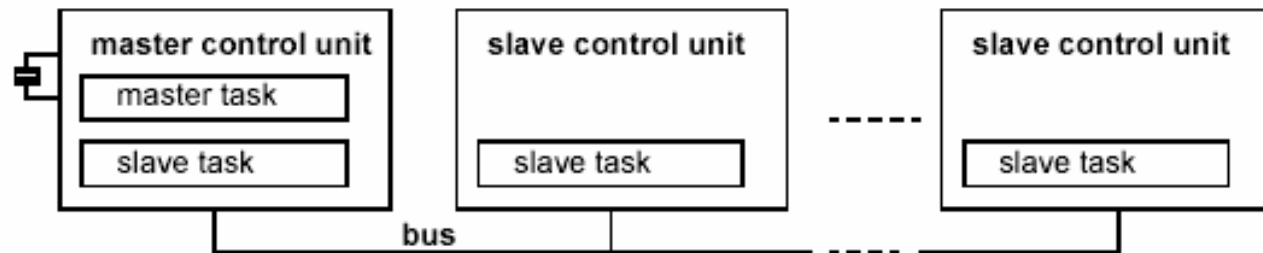


LLC = Logical Link Layer
MAC = Medium Access Control

LIN (Local Interconnect Network)

LIN Specification Package, Revision 1.2, Nov. 17, 2000

Zeitgesteuertes Master/Slave Protokoll



**Nur 1 Slave
darf antworten!
Aber alle Slaves
Empfangen
die Antwort.**

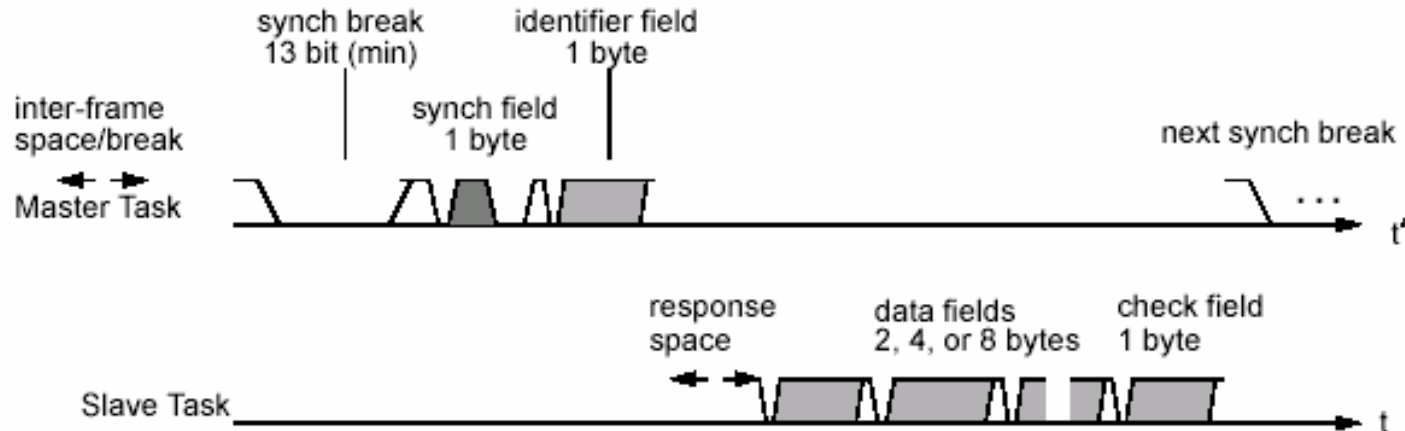


Figure 2.2: Communication Concept of LIN

LIN Specification Package, Revision 1.2, Nov. 17, 2000

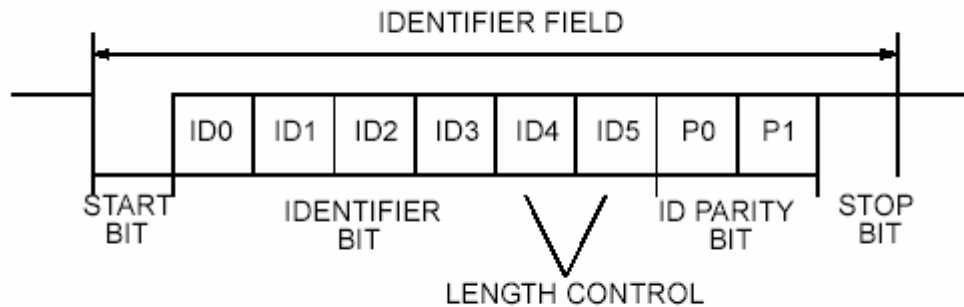


Figure 3.5: IDENTIFIER FIELD

64 Identifier

**in 4 Gruppen der
Länge: 2,4, und 8 Byte**

**Ein ID identifiziert den
Inhalt einer Nachricht,
nicht den Sender oder
Empfänger !**

**Slaves können ohne Änderungen in der Software der anderen Slaves
hinzugefügt oder entfernt werden.**

LIN Nachrichtenformat



inhaltsbezogene Adressierung

max. 8 Byte response frame

16 x

2 byte



2 byte



4 byte



8 byte



**reserved IDs: Master request Frame (0x3C), Slave Response Frame (0x3D)
Extended Frames (User 0x3E, Reserved 0x3F)**

LIN Specification Package, Revision 1.2, Nov. 17, 2000

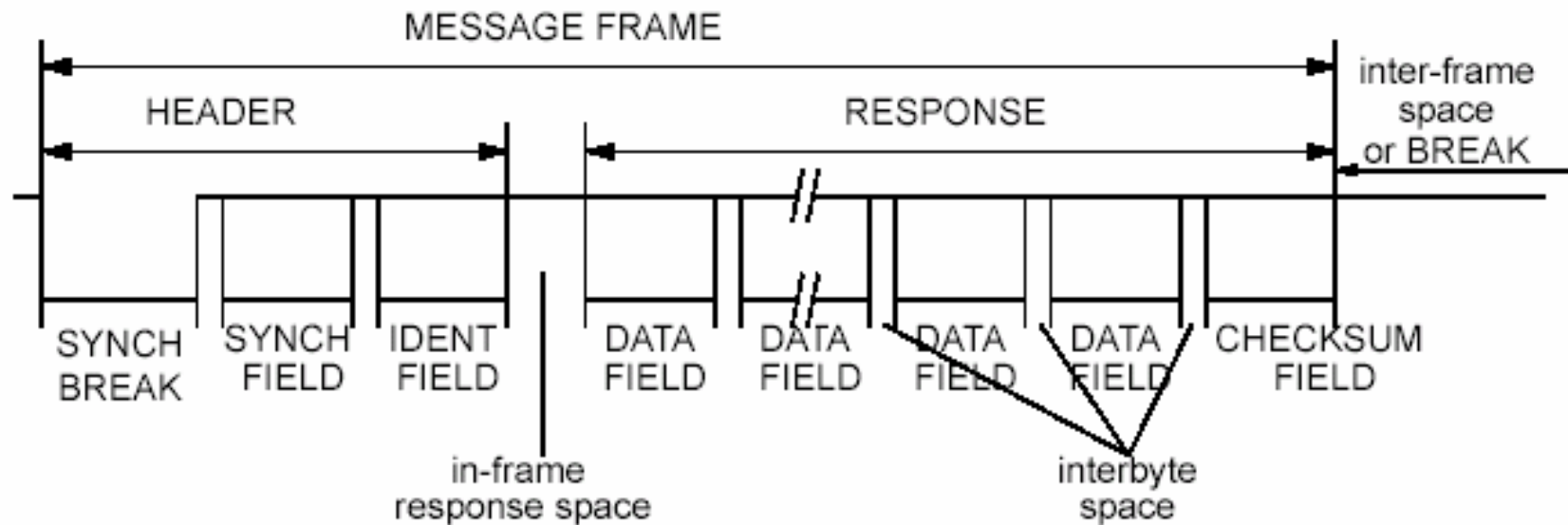
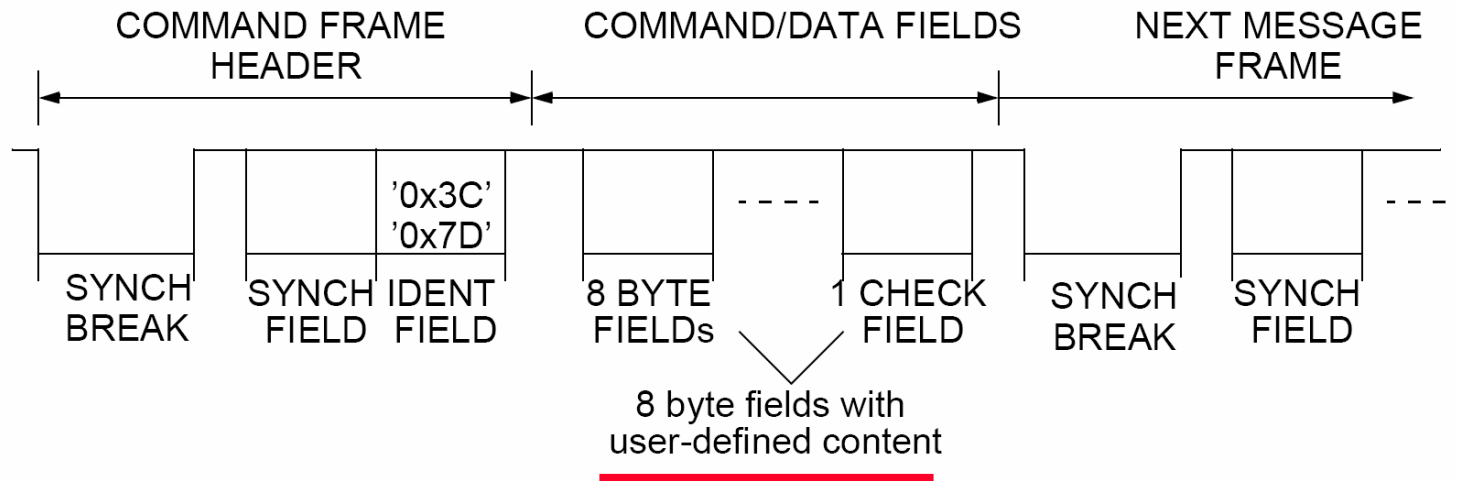


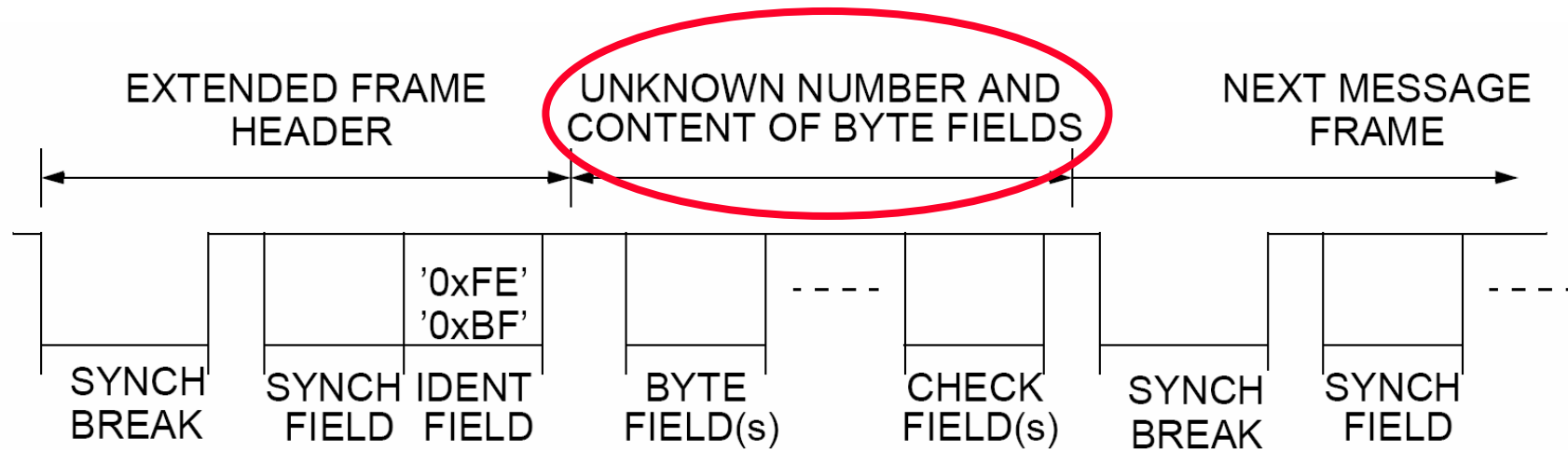
Figure 3.1: LIN MESSAGE FRAME

LIN Master Request Frame



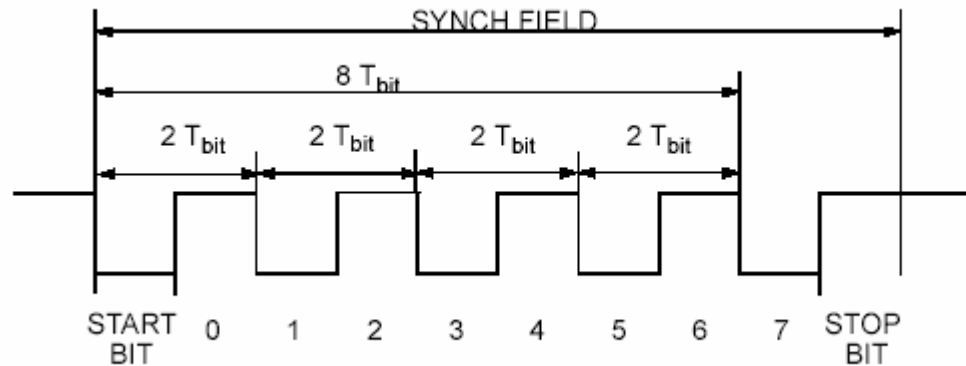
mehrere 8 Byte Felder möglich!
Slave Adresse ist Teil der Command Felder,
Download von Informationen zum Slave,
Anfordern von Informationen vom Slave.

LIN Extended Frame



**slaves, die sich nicht "angesprochen" fühlen
müssen bis zum nächsten SyncBreak warten!**

LIN Specification Package, Revision 1.2, Nov. 17, 2000



**Synch. Feld
0x55**

Figure 9.1: SYNCHRONIZATION FIELD

clock tolerance	Name	$\Delta F / F_{Master}$
master node	$F_{TOL_RES_MASTER}$	$< \pm 0.5\%$
slave node with quartz or ceramic resonator (without the need to synchronize)	$F_{TOL_RES_SLAVE}$	$< \pm 1.5\%$
slave without resonator, lost synchronization	$F_{TOL_UNSYNCH}$	$< \pm 15\%$
slave without resonator, synchronized and for a complete message	F_{TOL_SYNCH}	$< \pm 2\%$

Table 8.1: Oscillator Tolerance

Fehlererkennung durch das LIN Protokoll:

Bit-Error

Checksum-Error

Identifizier-Parity-Error

Slave-Not-Responding-Error

Inconsistent-Synch-Field-Error

No-Bus-Activity

TTP/A

H. Kopetz: Lit. Einführung,

H. Kopetz, W. Elemenreich, C. Mack: A Comparison of LIN and TTP/A, Research report 4/2000,
Institut für Technische Informatik, TU Wien

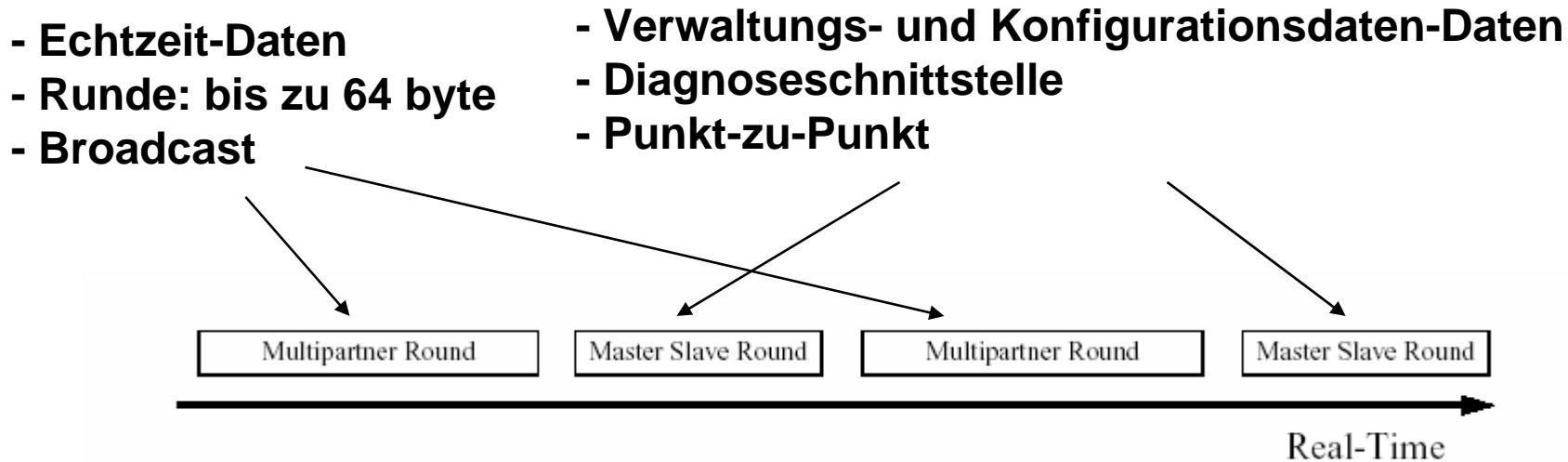
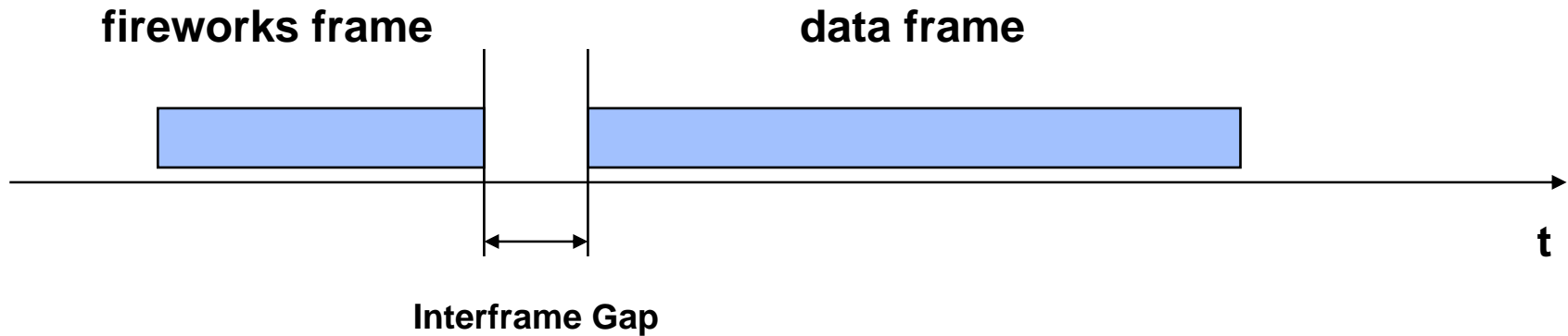


Figure 3: Traffic on the TTP/A Bus

Für Slaves sind 3 Schnittstellen spezifiziert:

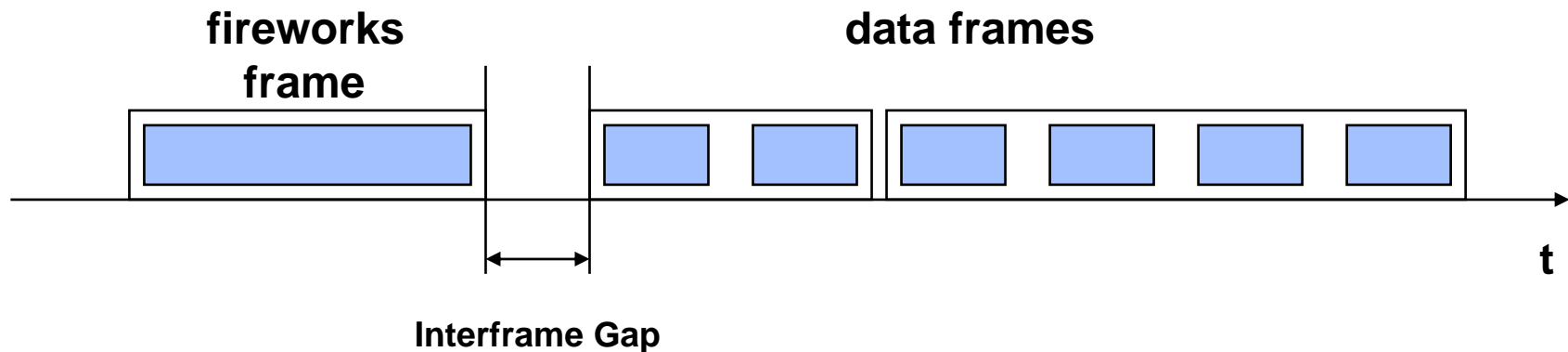
- Echtzeitdaten (RMI : Real-Time message Interface)
- Diagnose (DMI: Diagnostic message Interface)
- Konfiguration (CMI: Configuration Message Interface)

Master-Slave Dialog



<master-slave ("fireworks"), file op and identifier, record number, logical node name, check byte>

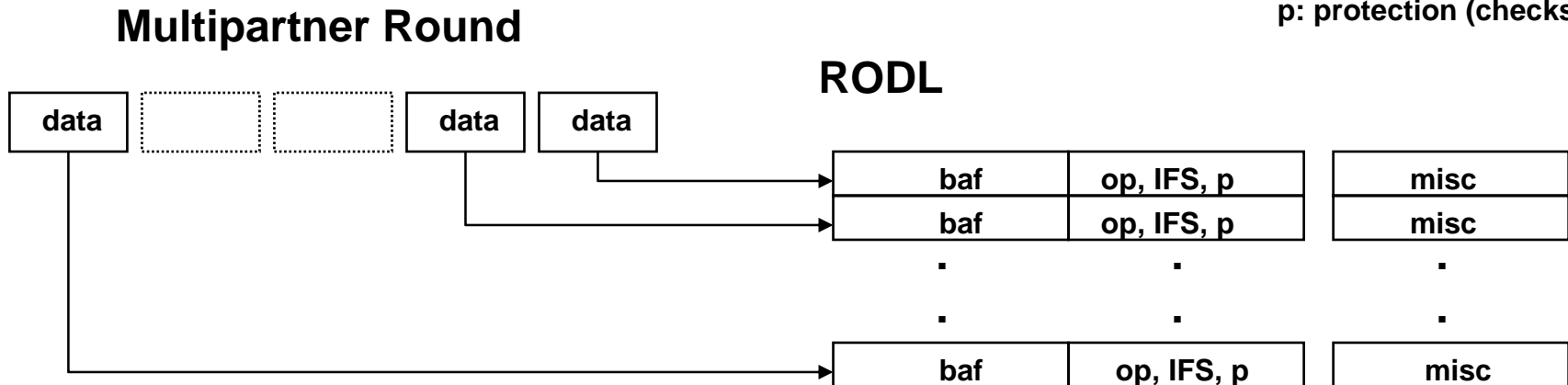
Multipartner Round



Datenzentriertes Kommunikationsmodell

- Echtzeitnachrichten enthalten **NUR Daten !**
- Alle Daten liegen im IFS.
- Die Adressen der Daten werden als IFS-Adressen spezifiziert.
- Die Adressen der Daten sind in der RODL (Round Description List) festgelegt, d.h. der Zeitschlitz, in dem die Daten übertragen werden, wird nach TTP-Manier festgelegt.

baf: byte after fireworks
 op: operation
 IFS: IFS-Adresse
 p: protection (checksum)



Die RODL ist ebenfalls im IFS abgelegt und kann durch das CMI konfiguriert werden.
 Es existieren max. 8 RODLs, die RODL# wird mit einer Hamming Distanz von 4 (hoher Schutz vor Übertr.Fehlern) übertragen.

Verwaltung des Nachrichtenraums durch ein „Interface File System“ IFS

Adresse enthält: < file, record, byte, checksum >
 2^6 2^8 2^2

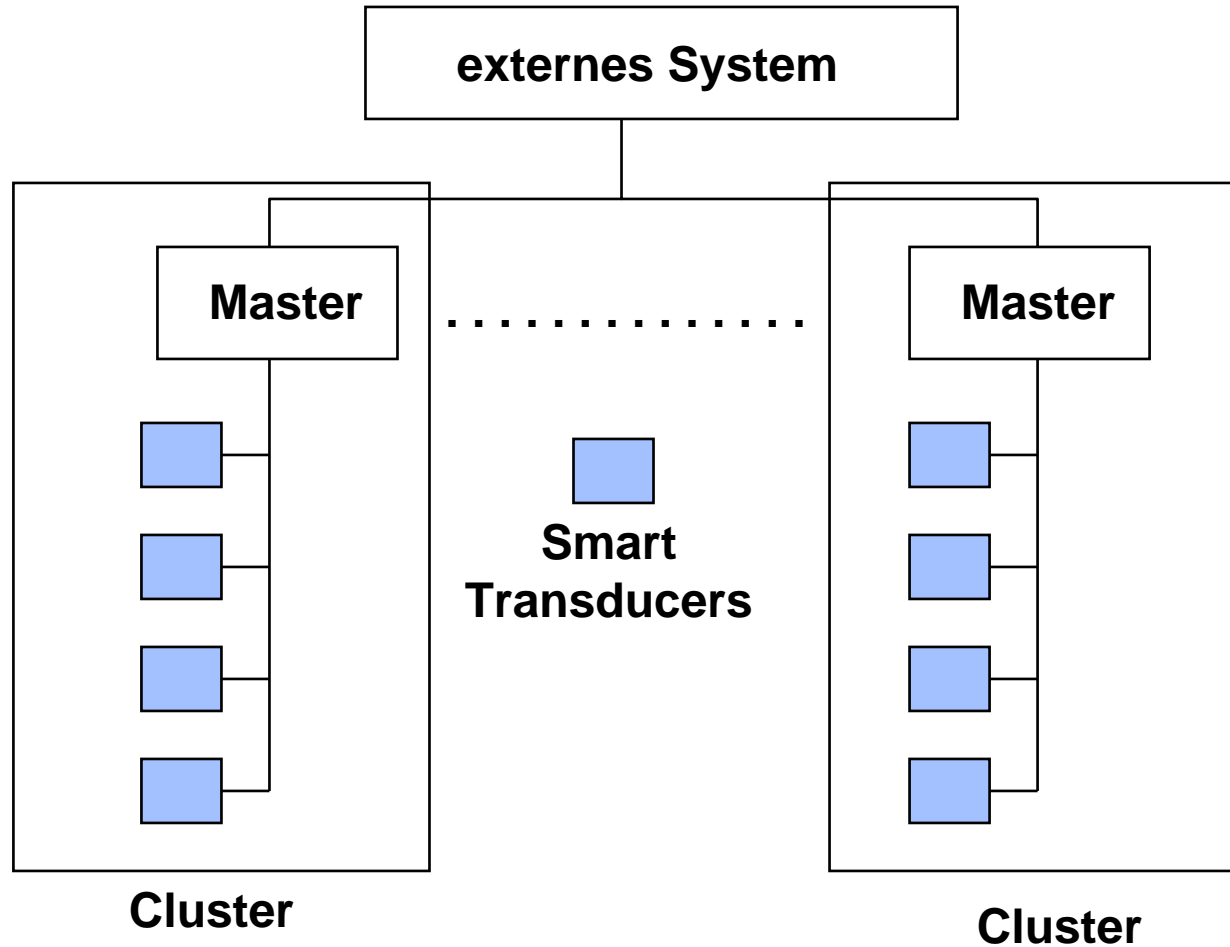
Jeder Knoten im globalen Netzwerk-Filesystem enthält:

bis zu 64 files
zu bis zu 256 records
mit jeweils 4 bytes

d.h. einen Adreßraum von 2^{16} bytes/Knoten.

Und wie werden die Knoten adressiert ?

Globaler Name eines Datums : <cluster name, node name, file name, record name>



Vergleich LIN und TTP/A (Antwortzeiten und Protokolleffizienz)

Antwort-
zeiten

10 nodes, response time in milliseconds on a 20 kbit bus	Minimum LIN	Maximum LIN	Minimum TTP/A	Maximum TTP/A
Every nodes sends four bytes of data	46.75 msec	65.4 msec	35.4 msec	35.6 msec
Every nodes sends two bytes of data	35.75 msec	50.05 msec	22.2 msec	22.3 msec
Every node sends one byte of data	35.75 msec	50.05 msec	15.6 msec	15.7 msec
Every node sends four bits of data	35.75 msec	50.05 msec	9 msec	9.1 msec
Every node sends four bits of data, additional master-slave round for DM service between any two multipartner rounds in TTP/A	not supported	not supported	16.8 msec	16.9 msec

Table 2: Achievable response times of LIN and TTP/A

Protokoll
Overhead

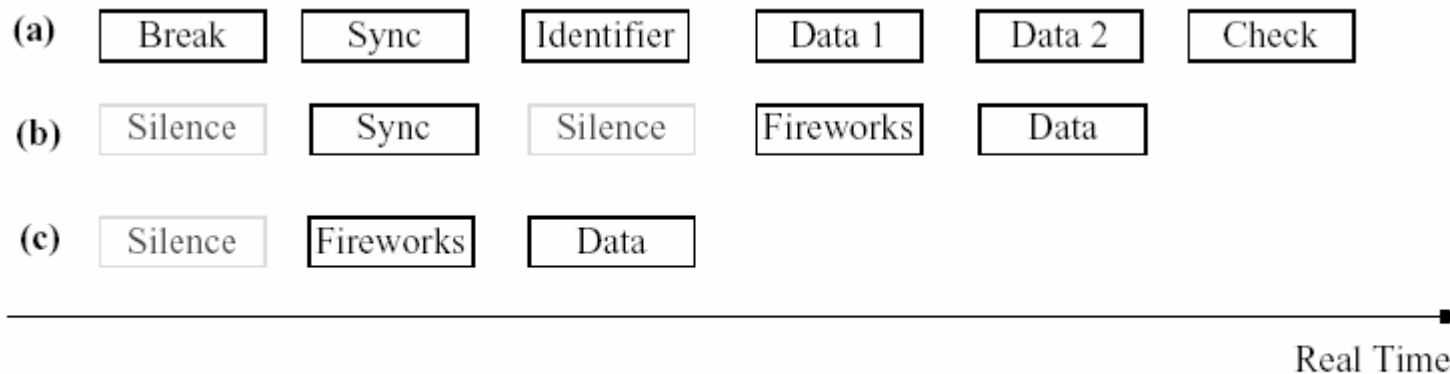


Figure 5: Byte Sequence of the simplest message in LIN (a), in TTP/A with start-up synchronization (b) and in TTP/A without start-up synchronization (c).



Flexibles Protokoll unterstützt synchrone und asynchrone Nachrichten

Für höhere Übertragungsraten geeignet

Integrierte Kommunikations-Controller vorhanden (z.B. Motorola 68HC912BD32)

Integrierter Bestandteil von FlexRay

Prinzipien:

- **Nachrichtenprioritäten sind Knoten-IDs zugeordnet**
- **Zeitschlitz, die Prioritäten zugeordnet sind**
- **Vorrang wird über Wartezeiten durchgesetzt**

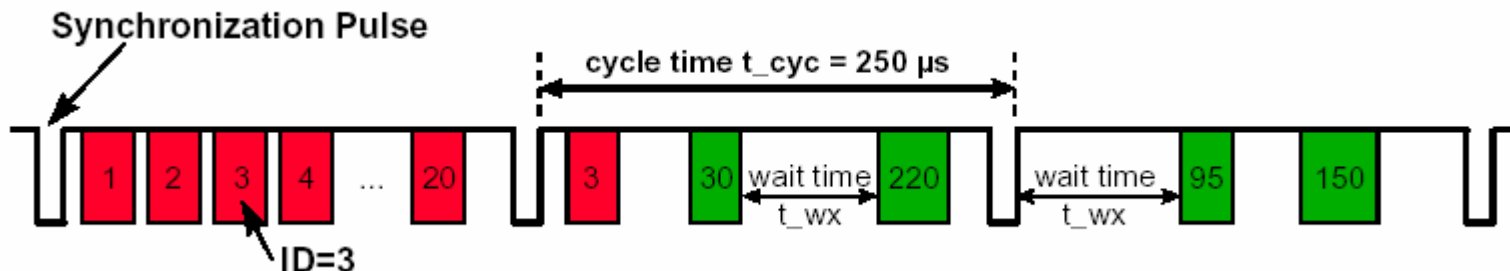


A New High-Performance Data Bus System for Safety-Related Applications

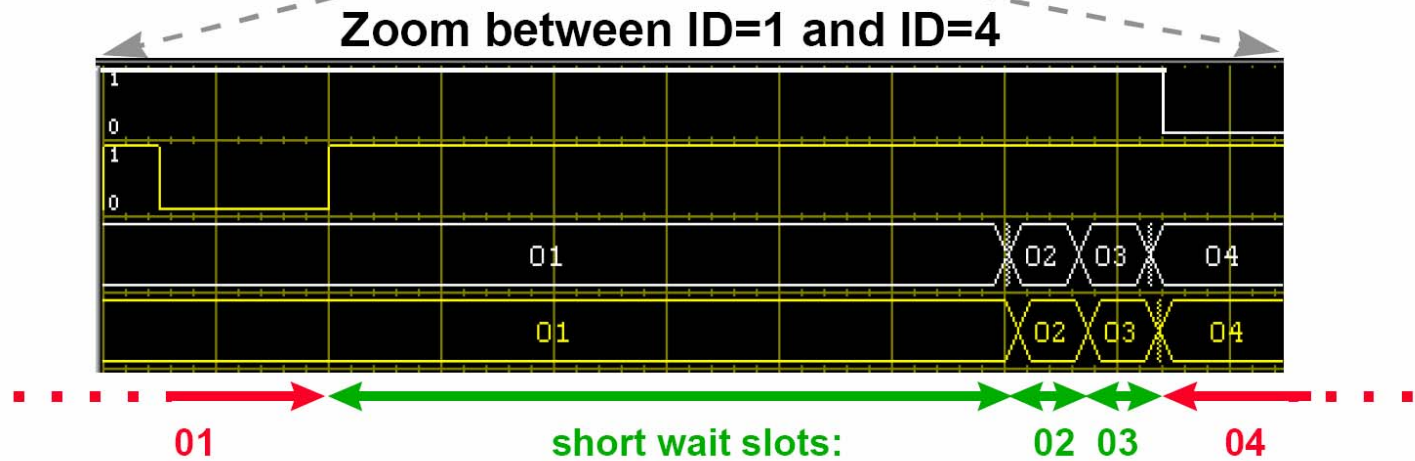
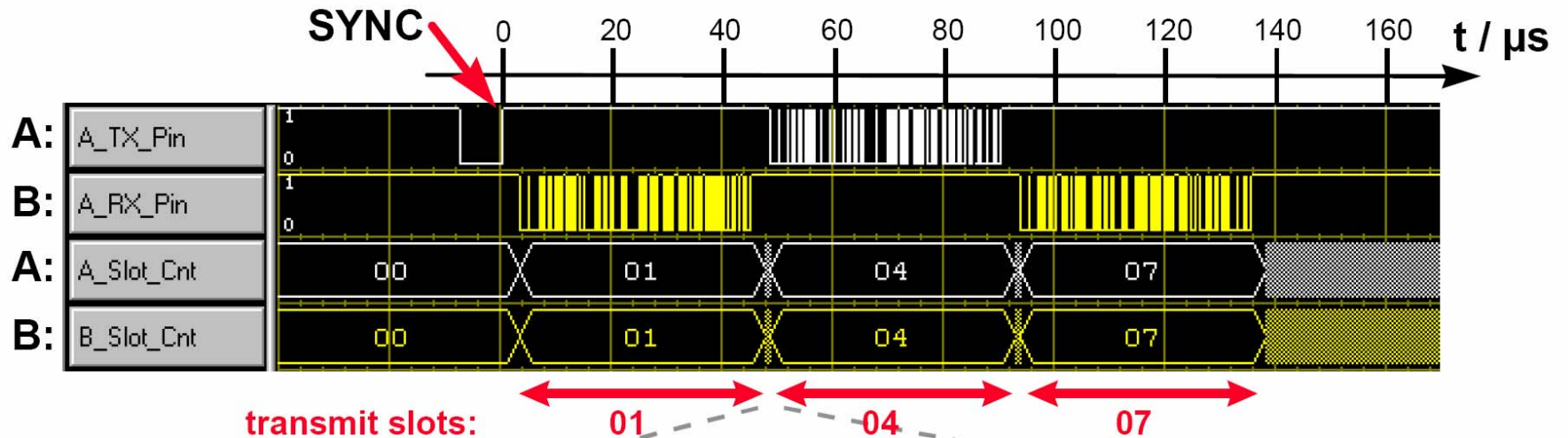
By Josef Berwanger, Martin Peller and Robert Griessbach
BMW AG, EE-211 Development Safety Systems Electronics,
Knorrstrasse 147, 80788 Munich, Germany

Byteflight: Flexible TDMA

- **SyncMaster** sendet einen Synchronisationsimpuls. Die Clock-Synchronisation wird mit **100 ns** angenommen.
- Das Intervall zwischen zwei Synchronisationsimpulsen definiert die Zykluszeit (**250 μ s @ 10 Mbps**)
- Jeder Knoten besitzt eine Menge von Identifiern, welche die Nachrichtenprioritäten festlegen. Es muss sichergestellt sein, dass die ID nur einmal im System vergeben werden.
- Jeder Kommunikations-Controller besitzt einen Zähler, der Zeitschlitz zählt.
- Wird eine Nachricht gesendet, stoppt der Zählvorgang und wird nach Beendigung des Nachrichtentransfers fortgesetzt.
- Erreicht der Zählerstand den Wert der eigenen Priorität, kann eine Nachricht gesendet werden.

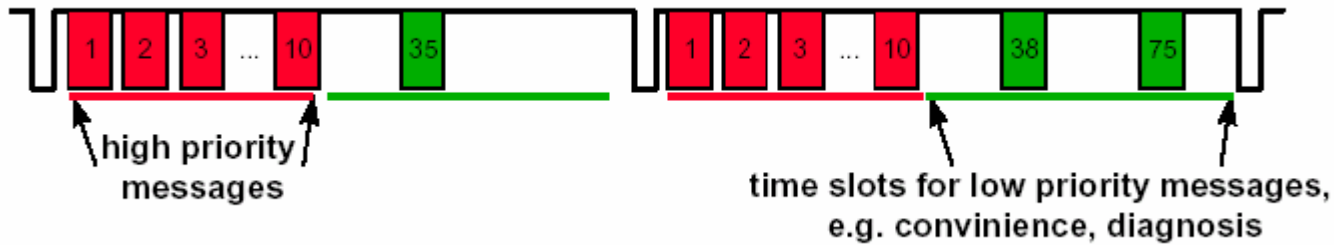


Verteilte synchronisierte "Slot-" Zähler



Waiting period $t_{wait} = t_0 + t_{delta} * (ID - ID_{t-1})$

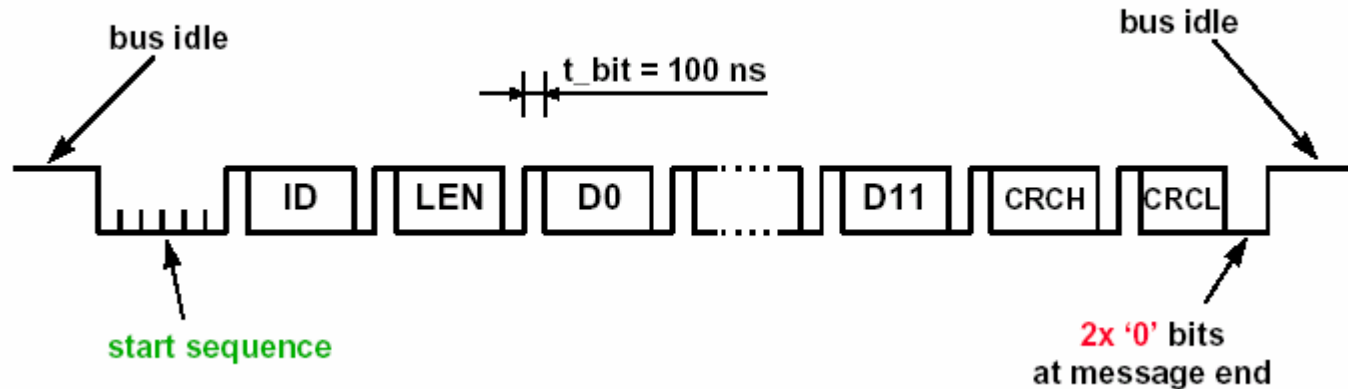
Synchrone und asynchrone Datenübertragung



**Synchronen Nachrichten werden über feste Prioritäten die entsprechenden Slots reserviert. Diese Slots werden in jedem Zyklus belegt (1-10).
Motivation: Möglichkeit einer deterministischen Analyse.**

**Asynchrone Nachrichten erhalten niedrigere Prioritäten, die durch den Slot-Mechanismus entsprechend durchgesetzt werden.
Statistische Analyse**

ByteFlight Nachrichten-Format



Startfolge: 6 Bits
ID: 8 Bits (1 Byte)
Länge: 8 Bits (1 Byte)
Datenteil: 96 Bits (12 Bytes)
CRC: 16 Bits (Hamming Distanz = 6)

Fehlerbehandlung im Byteflight Protokoll

Alarmstatus:

Der Master kann ein spezielles Synchronisationssignal senden, das von allen Stationen erkannt wird. So kann z.B. ein spezieller Zustand vom Master signalisiert werden. Auf das Protokoll hat das spezielle Sync-Signal keinen Einfluss.

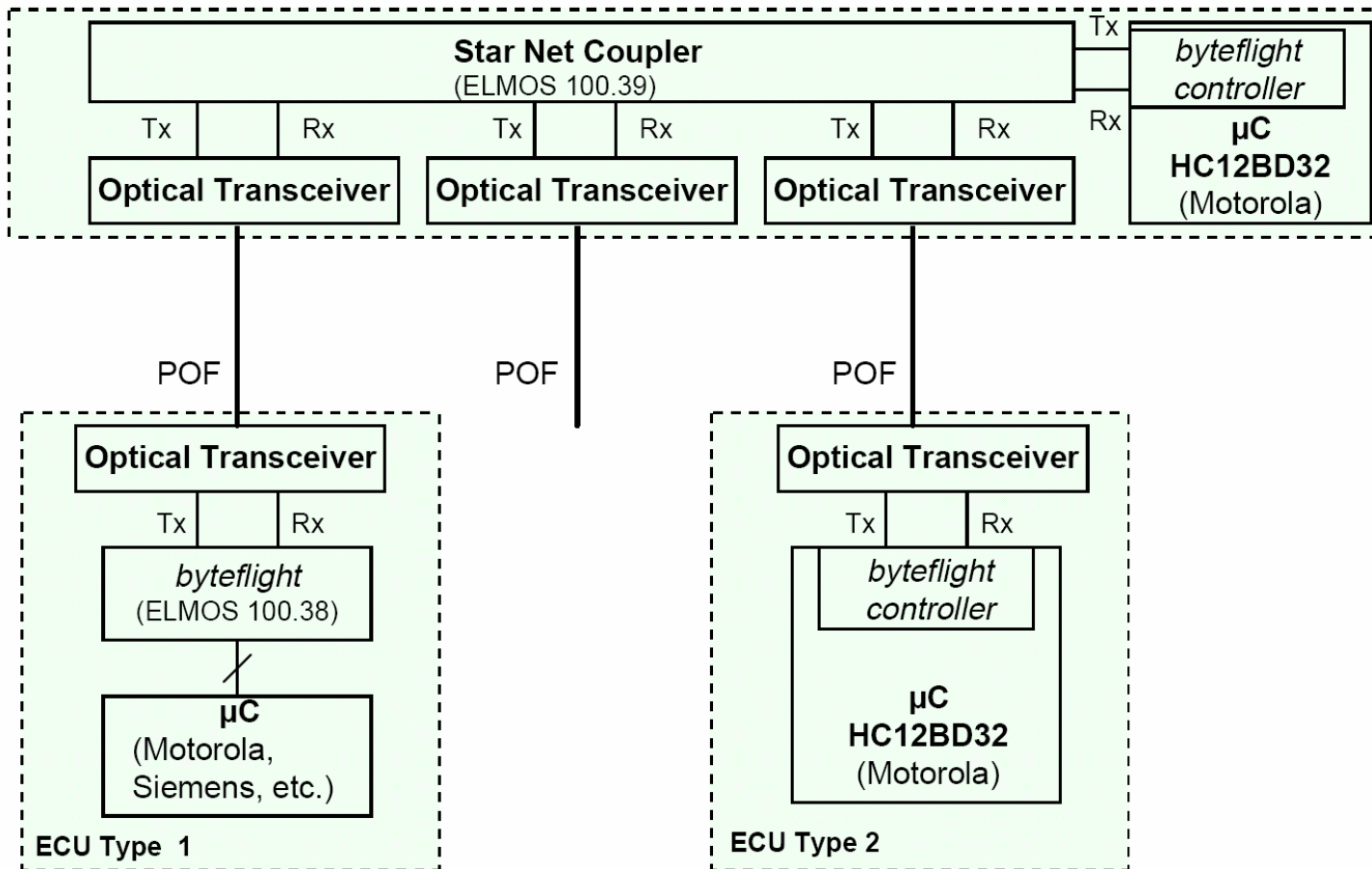
Fehlerbehandlung:

Es wird angenommen, dass sicherheitskritische Nachrichten zyklisch gesendet werden. Sie werden deshalb bei Fehlern nicht wiederholt.

Zeitfehler werden durch den Sternkoppler abgefangen.

In einer Bus-Struktur dienen die BUS-Guardians zur dezentrale Durchsetzung eines fail-silent Verhaltens. Die Zeitsteuerung wird hier ausgenutzt.

Falls ein Master ausfällt, tritt ein vorher bestimmter Konten als Ersatz ein.



Beispiel einer Byteflight Topologie

Vergleich zwischen Byteflight und TTP

byteflight: a new high-performance data bus system for safety related applications,

J. Berwanger, M. Peller, J. Griessbach,
BMW-AG, EE211 Development Safety
Systems Electronic

Feature	TTP [10]	<i>byteflight</i>
Message transmission	synchronous	asynchronous and synchronous
Message identification	time slot	message identifier
Data rate	2 Mbps gross up to 37 % net	10 Mbps gross up to 53 % net
Bit encoding	modified frequency modulation (MFM)	NRZ with start/stop bits
Physical layer	not defined	optical transceiver up to 10 Mbps
Latency jitter	constant for all messages	constant for high priority messages according t_cyc
Clock synchronization	distributed, in μ s range	by master, in 100 ns range
Temporal composability	supported	supported for high priority messages
Error containment (physical layer)	provided with special physical transceiver	provided by optical fiber and transceiver chip
Babbling idiot avoidance	possible by independent bus guardian	provided via star coupler
Extensibility	only if extension planned in original design	extension possible for high priority messages with affect on asynchronous bandwidth
Flexibility	only one message per node and TDMA cycle	flexible bandwidth for each node
Availability of components	microcoded RISC chip available, physical transceiver and independent bus guardian not available	HC12BD32, E100.38 <i>byteflight</i> standalone controller, E100.39 star coupler ASIC, optical transceiver available

Kombination von TDMA und Byteflight



FlexRay Communications System

Protocol Specification

Version 2.0



Registered copy for kaiser@ivs.cs.uni-magdeburg.de

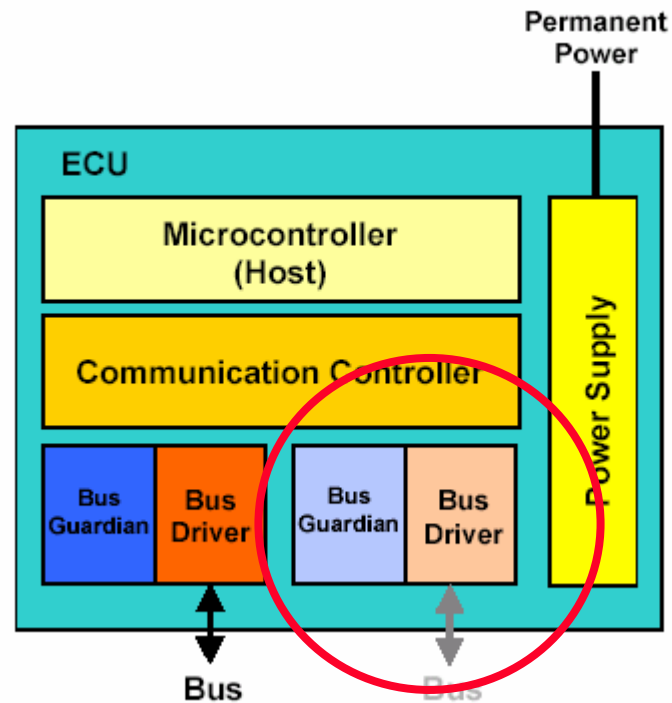


Requirements of the Protocol

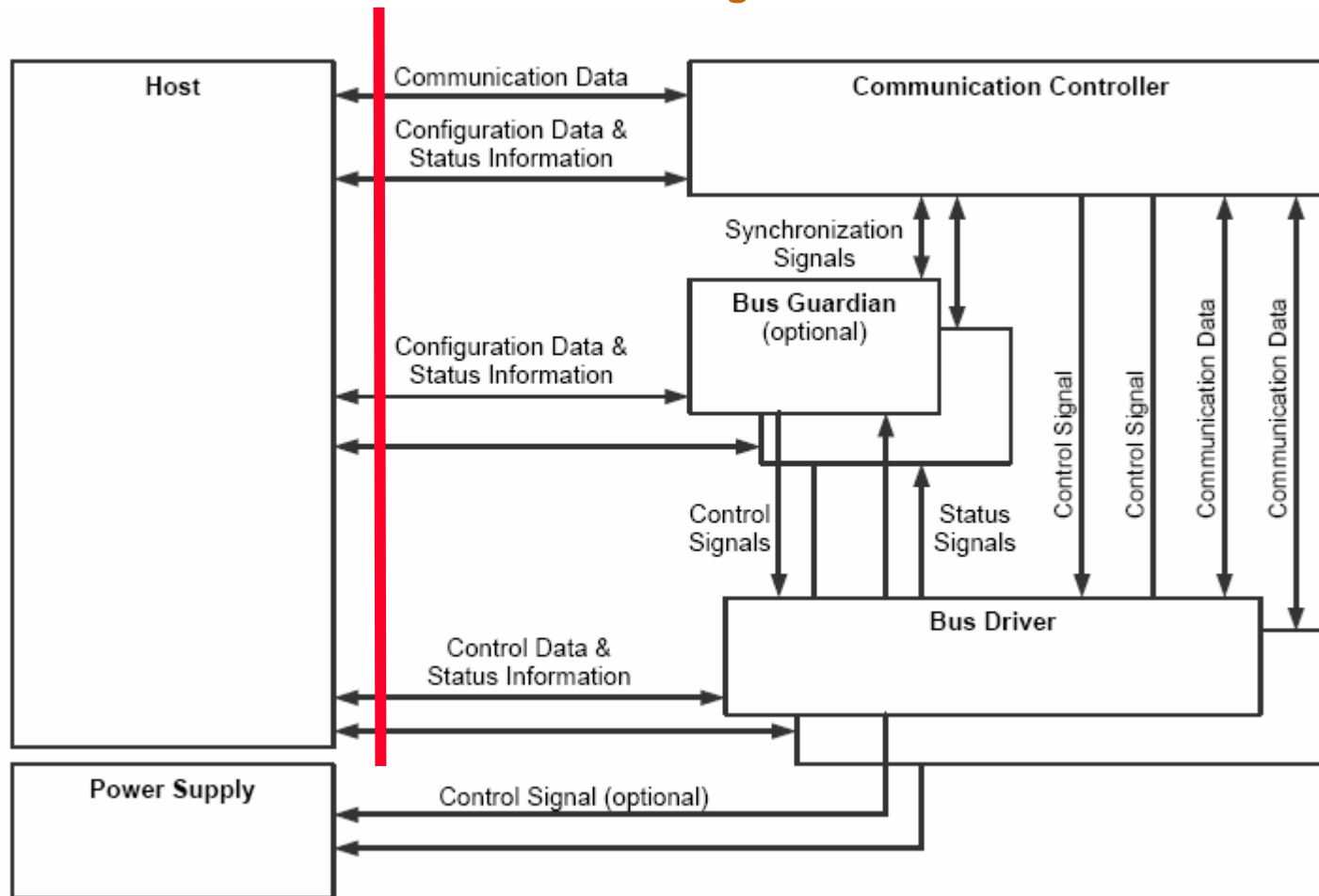
- Synchronous and asynchronous data transmission (scalable)
- Deterministic data transmission, guaranteed message latency
- Fault-tolerant, synchronized global time
- Redundant transmission channels (configurable)
- Flexibility (expandability, bandwidth usage, ...)
- Different topologies (bus, star and multi-star)
- Electrical and optical physical layer
- Communication protocol independent of the baud rate



Architektur eines FlexRay Knotens (ECU: Electronic Control Unit)



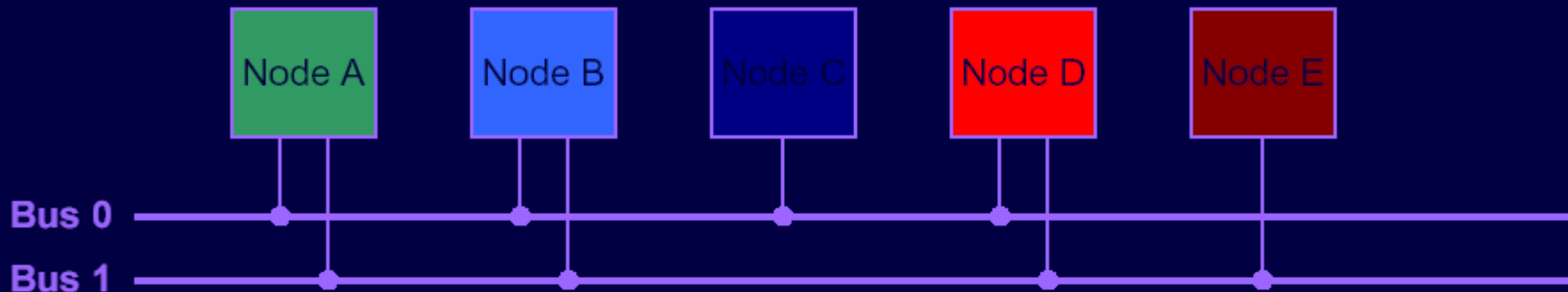
keine Kontrollsignale



Informations- und Kontrollfluss zwischen Host und CC



FlexRay Basic Concepts



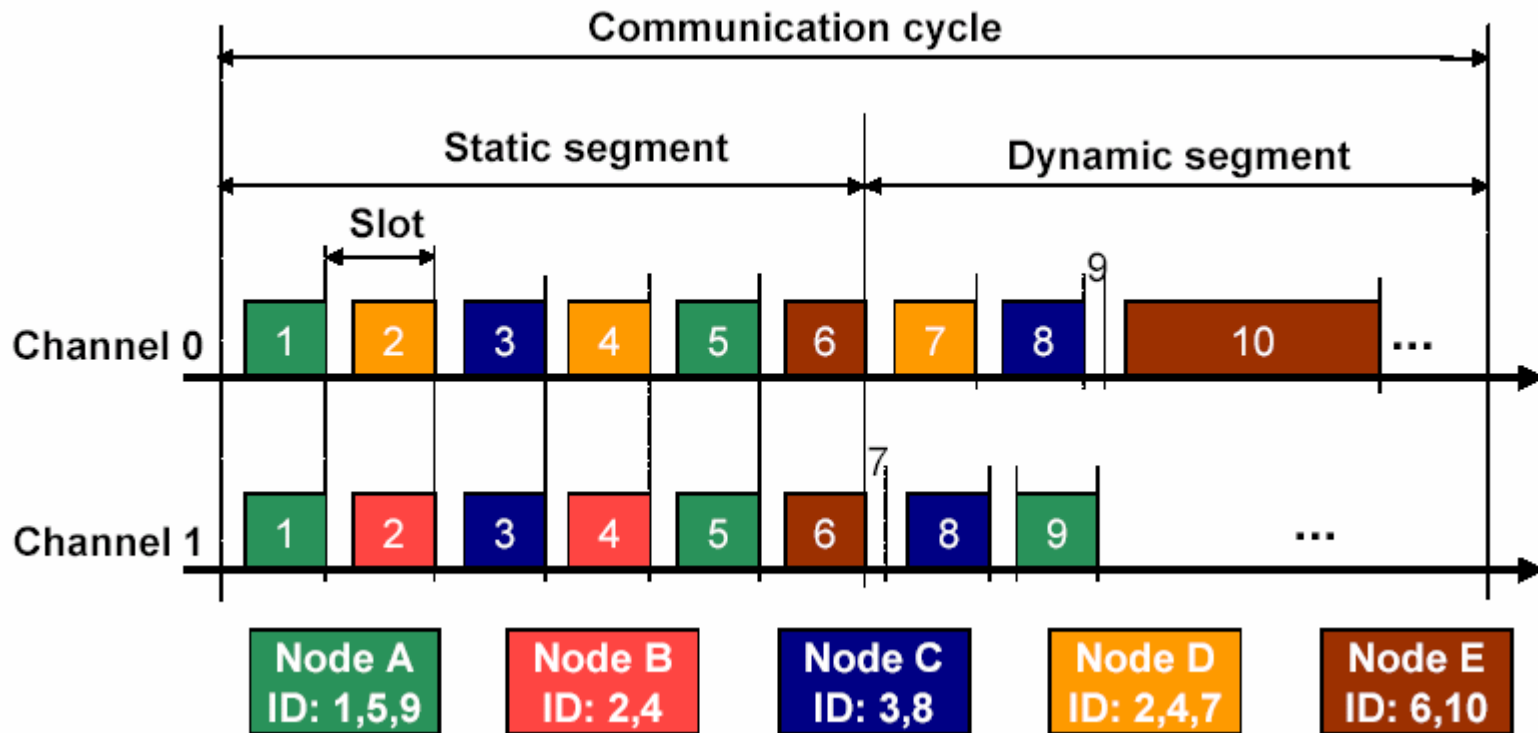
Redundancy

- The protocol supports two serial busses
- A node can either be connected to both or only one of the busses

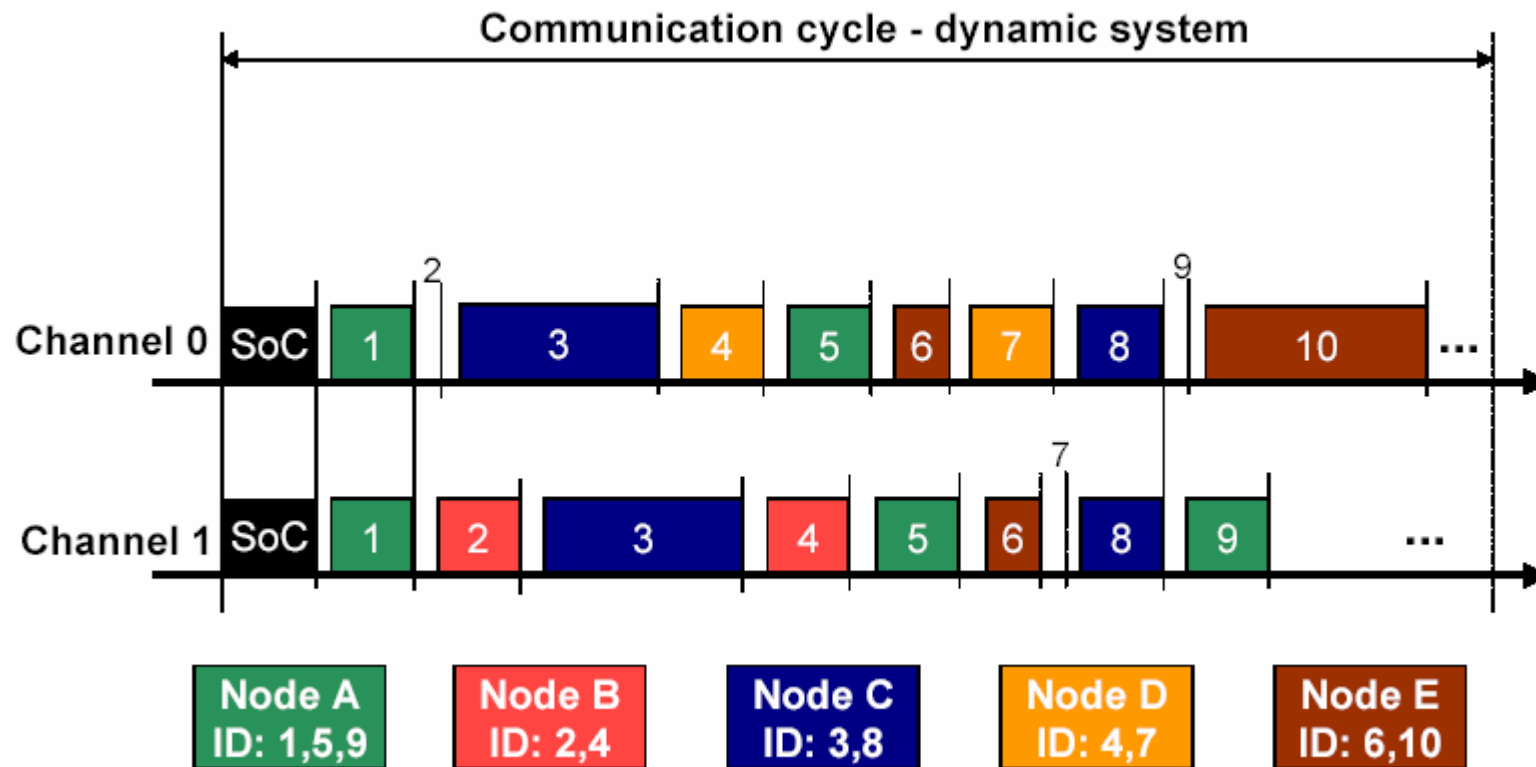
PHY Bit Coding

- transmission speed up to 10 Mbit/s (gross, optical)
- NRZ 8N1 for optical transmission
- Xerxes (MFM extension) coding for electrical transmission



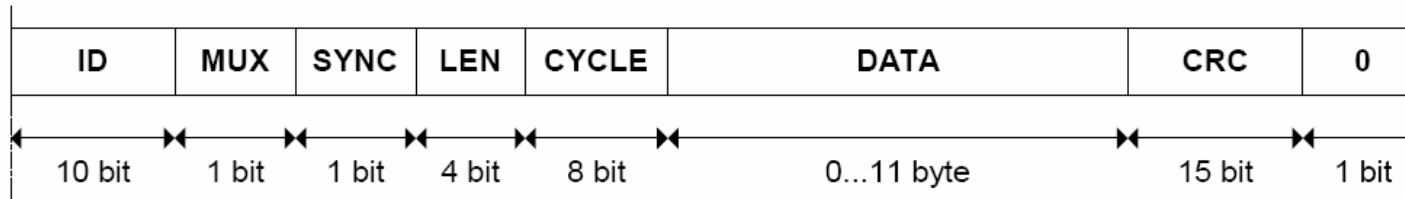


Zyklus mit statischem und dynamischen Segment



Zyklus mit rein dynamischen Segment

Format eines Telegramms



ID: Identifier, 10 Bit, Wertebereich: (110 ... 102310), definiert die Slotposition im statischen Teil und die Priorität im dynamischen Teil. Ein kleinerer Identifier bestimmt eine höhere Priorität. ID = 0 ist für das SYNC-Symbol reserviert. Ein Identifier darf in einem Netzwerk nur einmal verwendet werden. Jeder Knoten kann einen oder mehrere Identifier – sowohl im statischen als auch im dynamischen Teil – verwenden.

MUX: Multiplex-Feld, 1 Bit. Dieses Bit ermöglicht es, unterschiedliche Daten mit dem gleichen Identifier zu senden.

SYNC: SYNC-Feld, 1 Bit. Dieses Bit zeigt an, ob die Nachricht zur Uhrensynchronisierung verwendet wird und ob das erste Datenbyte den Zykluszähler enthält. (SYNC = "1": Botschaft mit Frame-Counter und Uhrensynchronisation, SYNC = "0": Botschaft ohne Frame-Counter)

LEN: Längen-Feld, 4 Bit, Anzahl der Datenbytes (010 ... 1210). Ein Wert größer 12 wird als LEN=12 interpretiert. Bei Nutzung des Zykluszählers (SYNC=1) wird bei einem Wert größer 11 LEN=11 gesetzt.

CYCLE: Das CYCLE-Feld kann als Zykluszähler oder als erstes Datenbyte genutzt werden. Der Zykluszähler wird am Anfang eines jeden Kommunikationszyklus in allen Kommunikationscontrollern synchron hochgezählt.

D0-11: Daten Bytes, 0 – 12 Bytes

CRC: 15 Bit Cyclic Redundancy Check.

Topologie Optionen

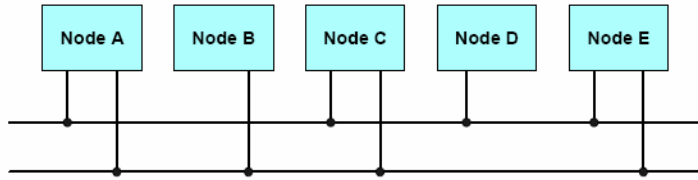


Figure 1-1: Dual channel bus configuration.

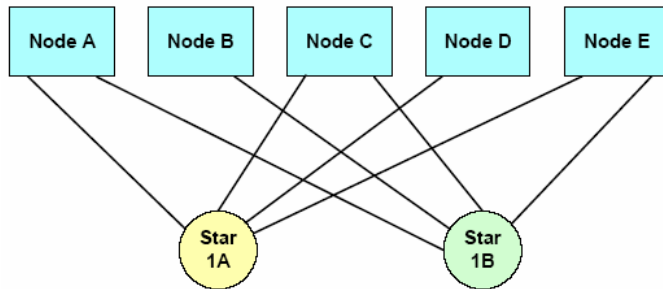


Figure 1-2: Dual channel single star configuration.

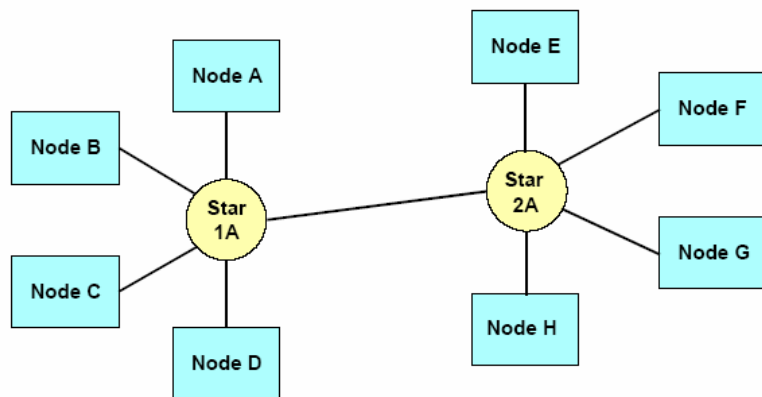


Figure 1-3: Single channel cascaded star configuration.

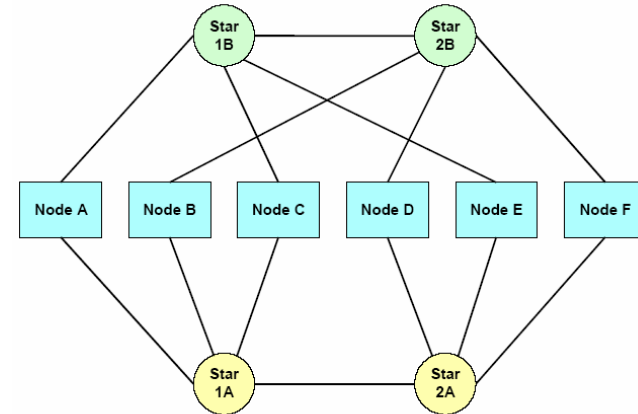


Figure 1-4: Dual channel cascaded star configuration.

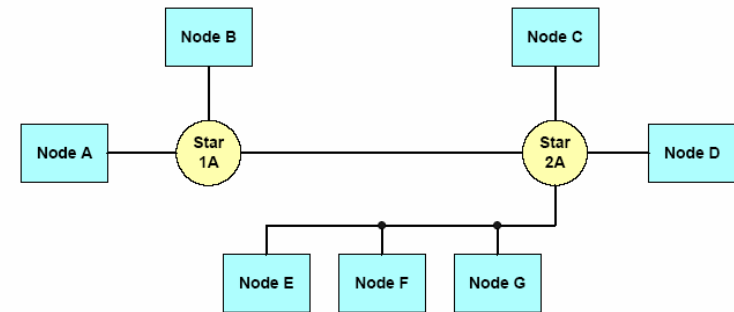


Figure 1-5: Single channel hybrid example.

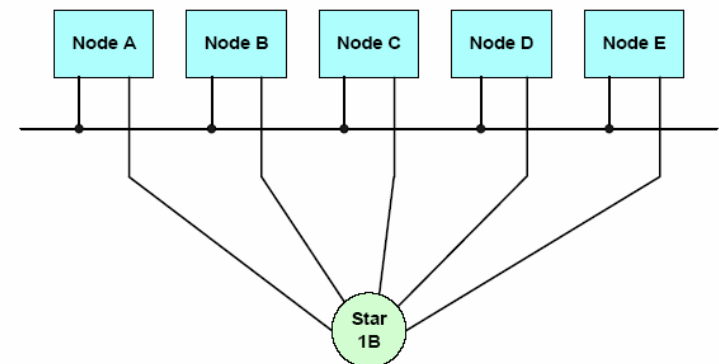


Figure 1-6: Dual channel hybrid example.