

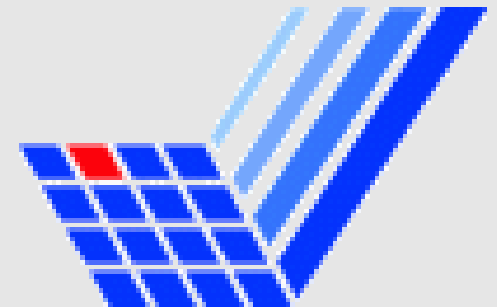
CAN

Sebastian Kosch

sebastian.kosch@gmail.com

PG AutoLab

Seminarwochenende 21.-23. Oktober 2007



Überblick

- Der CAN-Bus
- Das CAN-Protokoll
- CAN-Hardware
- CAN-Software
- Zusammenfassung
- **Fragen**

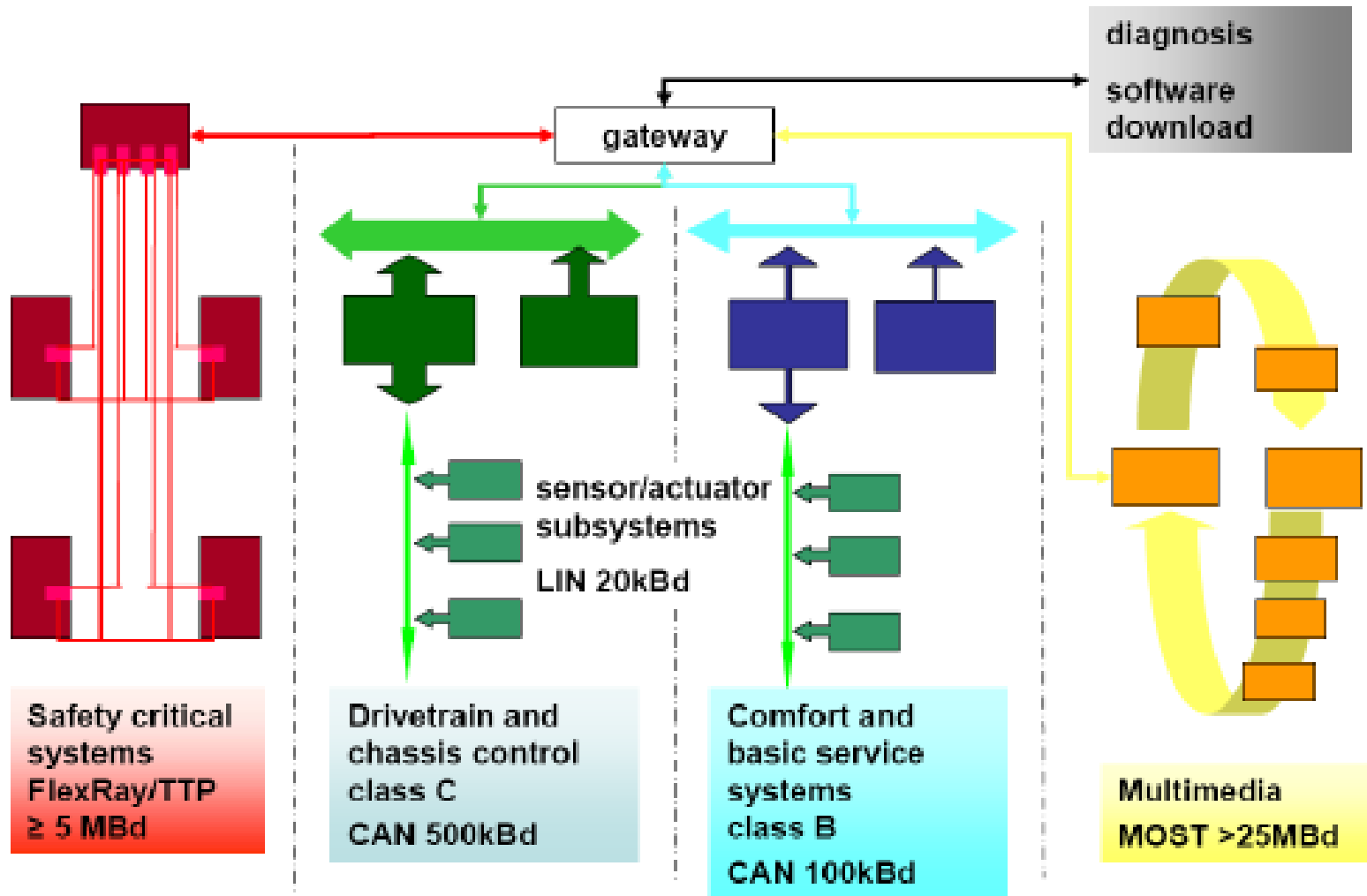


Was ist ein CAN?

- Controller-Area-Network
- Feldbus
- multimaster Buszugriff
- 1983 von Bosch entwickelt um Kabelbäume in Autos zu verkürzen (bis dato: bis zu 2 km pro Fahrzeug)
- Anwendungsgebiete: Automobile, Fahrstuhlsteuerungen, medizinische Geräte, Automatisierungstechnik (z.B. bei Kuka)
- Sorgt durch verschiedene Mechanismen für sichere Datenübertragung



Der CAN-Bus im Auto



Der CAN-Bus im Auto

High-Speed-CAN (125kBit – 1MBit,
max. 20 Knoten)

- Motorelektronik
- Getriebesteuerung
- ABS
- Airbag etc.

Low-Speed-CAN (bis 125kBit, max. 30 Knoten)

- Klimasteuergerät
- Amaturenbrett-Elektronik
- Wischer
- Zentralverriegelung etc.



Der CAN-Bus im Auto

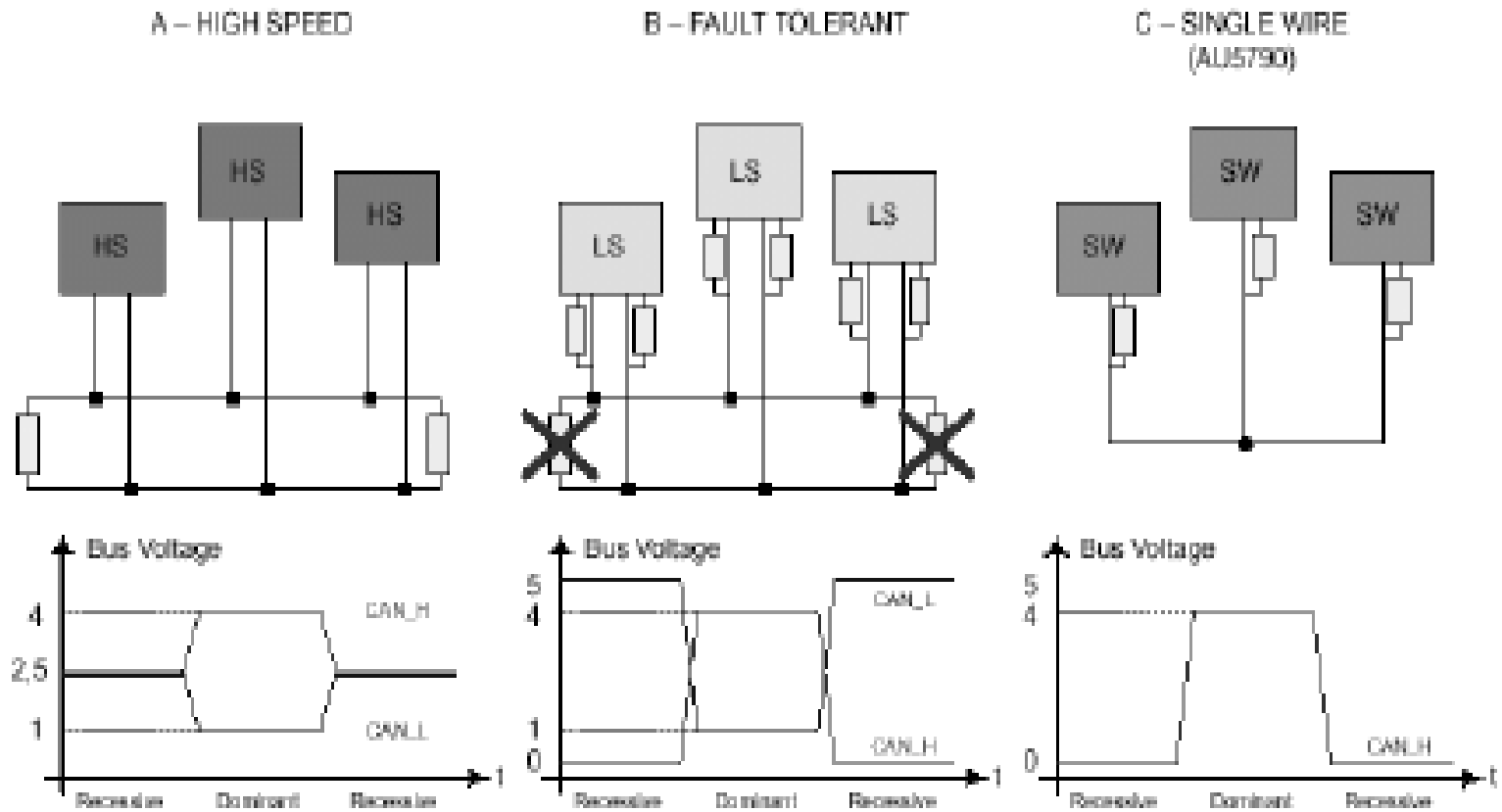


Abb. 6-15 : Physikalische Medien für A: Standard-CAN, B: Low-Speed, Fault-tolerant, C: Single-Wire [2]

Bitrate und Leitungslängen

Bitrate	Kabellänge
10 kbits/s	6,7 km
20 kbits/s	3,3 km
50 kbits/s	1,3 km
125 kbits/s	530 m
250 kbits/s	270 m
500 kbits/s	130 m
1 Mbits/s	40 m

- Bitrate bis zu 1 Mbit/s
 - alle Knoten am Bus müssen Daten gleichzeitig verarbeiten
- ⇒ Bitrate hängt von der Kabellänge des Bus ab.

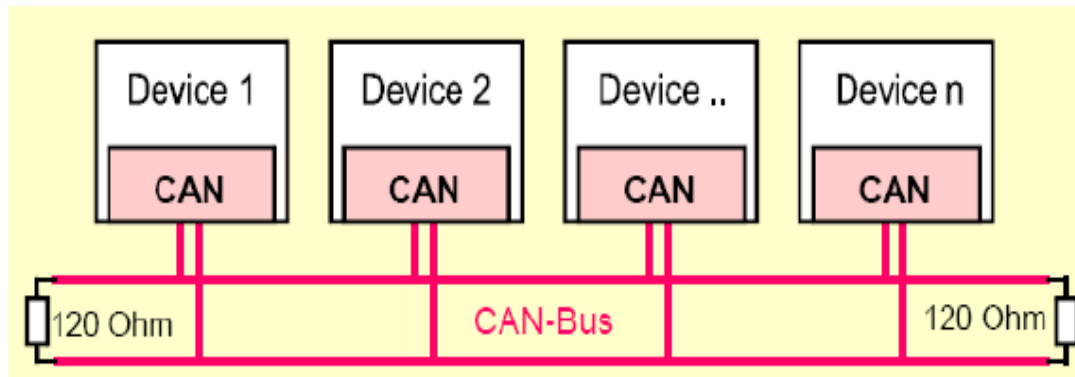


Überblick

- Der CAN-Bus
- Das CAN-Protokoll
- CAN-Hardware
- CAN-Software
- Zusammenfassung
- **Fragen**



Prinzip des Datenaustausches



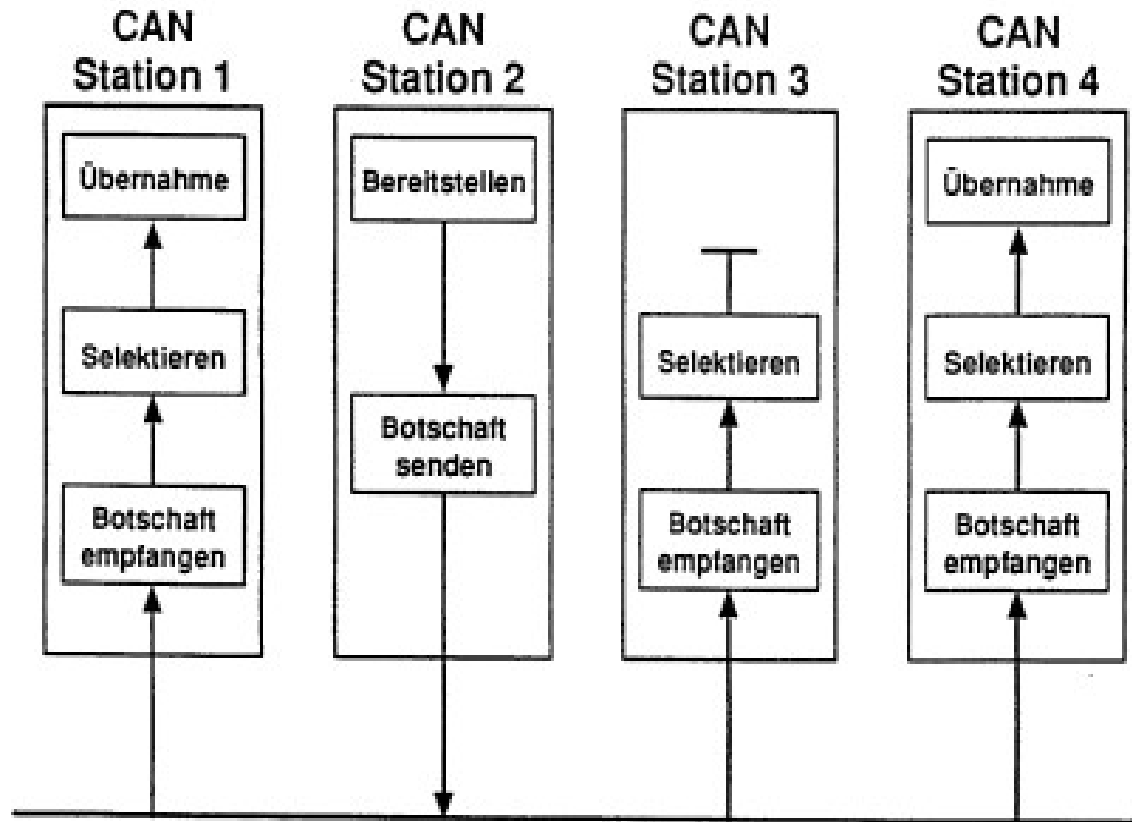
- 2 verdrillte Leitungen: CAN_HIGH, CAN_LOW
- Abschlusswiderstand
- Vielzahl von Steuergeräten können auf CAN-Bus zugreifen
- Jedes Steuergerät hat ein Steuergerät-internes CAN-Interface

Der Nachrichten-Identifizier

- Keine Geräte- oder Systemadressen beim CAN-Bus
- Jede Nachricht hat **eindeutige Identifikationsnummer**
(Nachrichten-Identifizier, ID)
- **Low-Speed-CAN**: 2048 (11 Bit ID-Länge)
- **Standard-CAN** (2.0 A): 2.048 (11 Bit ID-Länge)
- **Extended-CAN** (2.0 B): 29 Bit ID-Länge
- ID legt **Priorität** der Nachricht fest.
- Das höchstsignifikante Bit wird zuerst übertragen



Prinzip des Datenaustausches



Broadcast-Übertragung und Akzeptanzprüfung durch die CAN-Knoten

Prinzip des Datenaustausches

Senden:

- CPU einer Station will senden \Rightarrow CPU übergibt Nachricht, Identifier und Übertragungsaufforderung an zugeordneten CAN-Baustein
- CAN-Baustein übernimmt Bildung und Übertragung der Nachricht. Wenn mehrere Bausteine gleichzeitig auf Bus zugreifen erfolgt eine Bitweise Arbitrierungsphase(ID und RTR)



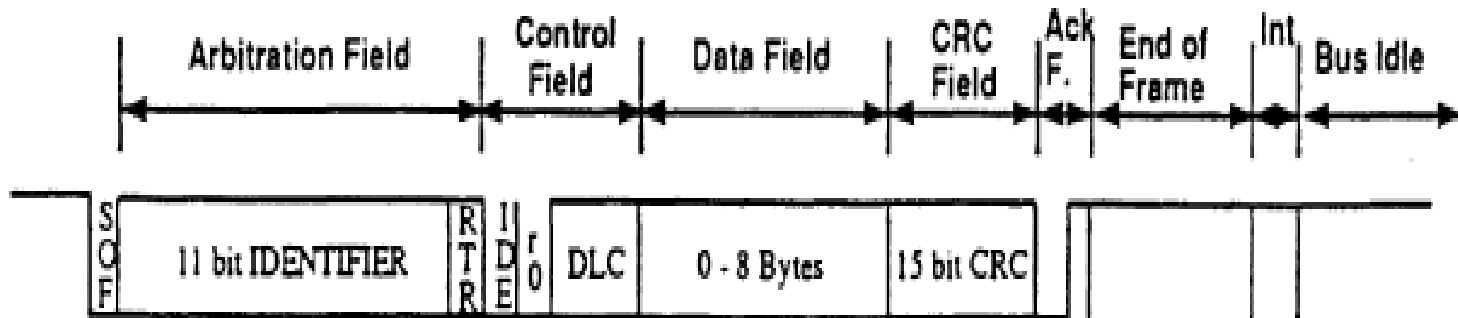
Prinzip des Datenaustausches

Empfangen:

- Wenn ein CAN-Baustein die Buszuteilung hat, werden alle anderen Busteilnehmer zu Empfängern
- Alle Empfänger nehmen nach korrektem Empfang eine Akzeptanzprüfung vor:
 - Verarbeiten der Nachricht
 - Ignorieren der Nachricht



Datentelegramm Standard

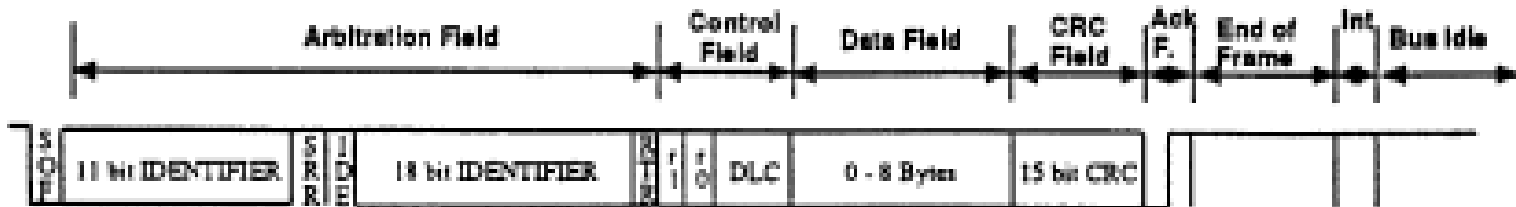


Botschaftsrahmen im Standard-Format (CAN-Spezifikation 2.0A)

- SOS: Start-of-Frame
- Arbitrierung erfolgt über ID und RTR Bits
- IDE: Identifier Control Bit
- DLC: Data Length Code
- r0: reserviert

Datentelegramm Extended

Extended Format



Botschaftsrahmen im Erweiterten-Format (CAN-Spezifikation 2.0B)

- 29-Bit Identifier
 - SRR: Substitute Remote Request Bit
- ⇒ standard Format hat höhere Priorität!

5 Arten der Fehlererkennung

- **Bitüberwachung**
 - Knoten vergleicht soll- und ist-Buspegel
- **Überwachung des Nachrichtenformats**
 - Einhaltung der Frames wird überwacht
- **Zyklische Blockprüfung**
 - gesendete crc checksumme wird mit selbst errechneter verglichen
- **Bit-Stuffing Überprüfung**
 - nach fünf gleichen Bits ein Stuff Bit
- **Überwachung des Acknowledgements durch Sender**
 - ohne ACK wird von Übertragungsfehler ausgegangen

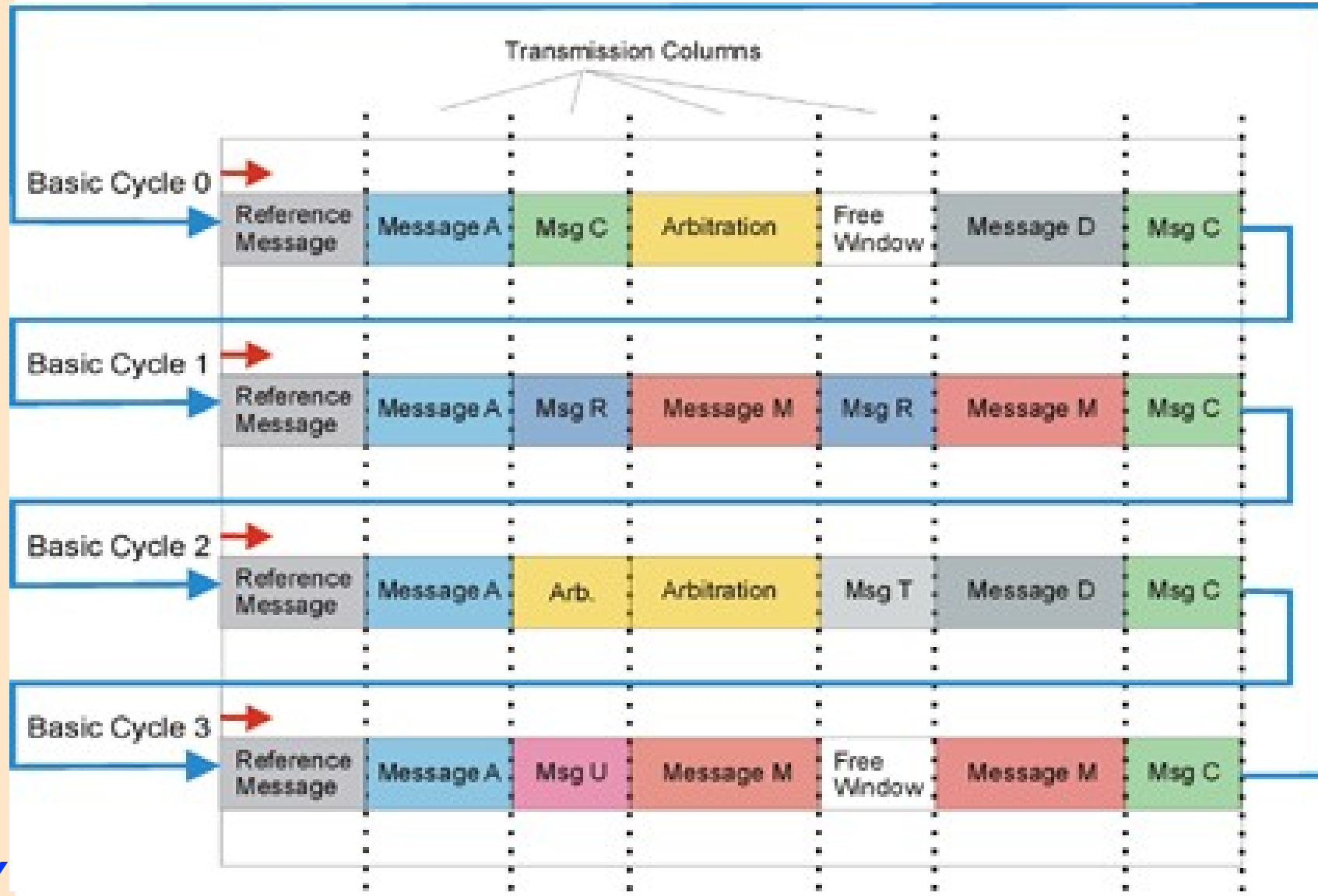


Echtzeitverhalten

- **TTCAN** (time-triggered communication on CAN)
- Nachrichten: „**time-triggered**“ (z.B Status Bremsen) ,
„**event-triggered**“ (z.B Temperaturechsel im
Kühlungsmechanismus)
- „**reference frame**“: Synchronisation
- Knoten „wissen“ wann sie senden müssen
- „Event“ kann Scheduling-Zyklen unterbrechen
- Knoten fällt aus \Rightarrow anderer Knoten übernimmt
Frame
- Bosch hat TTCAN-Modul in Chip implementiert



Echtzeitverhalten



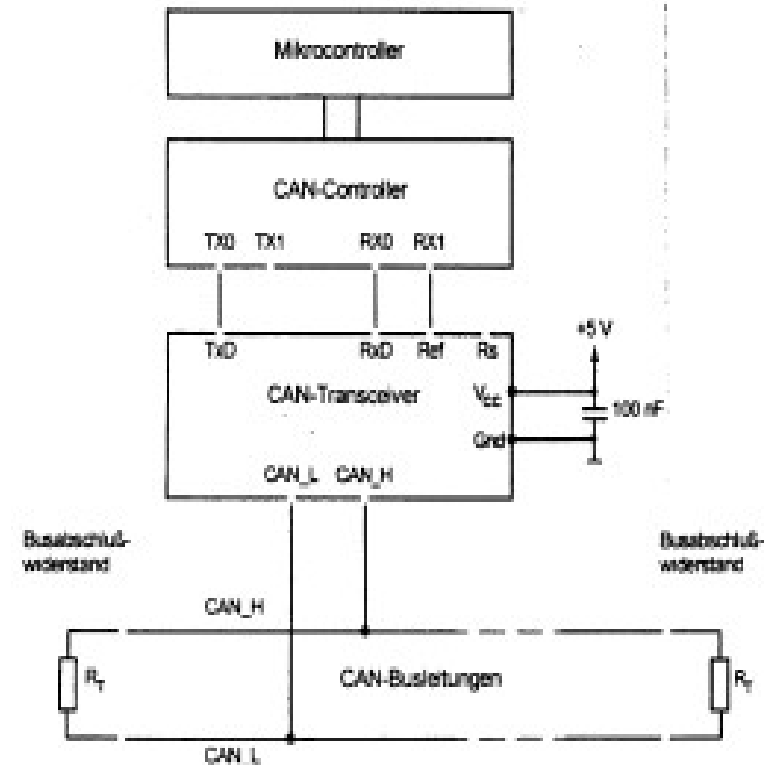
Überblick

- Der CAN-Bus
- Das CAN-Protokoll
- **CAN-Hardware**
- CAN-Software
- Zusammenfassung
- **Fragen**



CAN-Bausteine

- „Stand-alone“
Protokollcontroller-Bausteine
 - erforderliche Schichten für des CAN-Protokolls sind in Hardware implementiert
 - braucht extra „Host-Controller“
 - niedrige Übertragungsraten
- Mikrocontroller mit integriertem Protokollcontroller-Modul
 - hohe Übertragungsraten



Physikalische CAN-Ankopplung nach ISO 11898

CAN-Bausteine

- I/O-Bausteine mit integriertem Protokollcontroller-Modul (SLIO-CAN-Controller)
 - kostengünstige Vernetzung von Sensoren und Aktoren ohne lokale Intelligenz
 - verfügen über digitale und/oder analoge Ein- und Ausgänge
- Transreceiver-Bausteine
 - für hohe Anforderungen in Bezug auf Stör- und Spannungsfestigkeit sind spezielle Technologien nötig



Überblick

- Der CAN-Bus
- Das CAN-Protokoll
- CAN-Hardware
- **CAN-Software**
- Zusammenfassung
- **Fragen**



CAN-Software

- Software wird immer komplexer und umfangreicher
- Viele Automobilhersteller haben ihre eigene spezifische Standardsoftware
⇒ modularer Baukasten
- Unterschiedliche oft Herstellerspezifische Konfigurationsformate (selbst Module gleicher Funktionalität)



CAN-Software

- AUTOSAR Konsortium ⇒ Ziel: industrieweite Standardisierung
- Grundsätzliches Problem: fehlen einer einheitlichen, modulübergreifenden Modellierungsmöglichkeit für alle Anforderungen an die Konfiguration



Zusammenfassung

- CAN-Bus für Klasse B- und C-Systeme im Auto
- Multimaster-System
- Alle Stationen erhalten alle Nachrichten
- Keine Adressen \Rightarrow Identifier
- Sehr gute Fehlererkennung
- Realisiertes Echtzeitverhalten im TTCAN
- keine einheitliche Software



Literatur

- www.can-cia.org
- www.auto.tuwien.ac.at/LVA/DA/DA_04.pdf
- cis.cs.tu-berlin.de/Lehre/SS-03/Sonstiges/technInfo/CAN-Ausarbeitung.pdf
- ivs.cs.uni-magdeburg.de/bs/lehre/wise9900/proro/vortrag/can/CAN.pdf
- www7.informatik.uni-erlangen.de/~dulz/fkom/06/4.pdf
- www.rtos.de/PDF-file/CAN/Deutsch/intro-d.pdf
- www.semiconductors.bosch.de/pdf/can.pdf
- Konrad Etschberger: CAN ; Grundlagen, Protokolle, Bausteine, Anwendungen, Hanser Verlag, 1994



ENDE

DANKE
für die Aufmerksamkeit!

