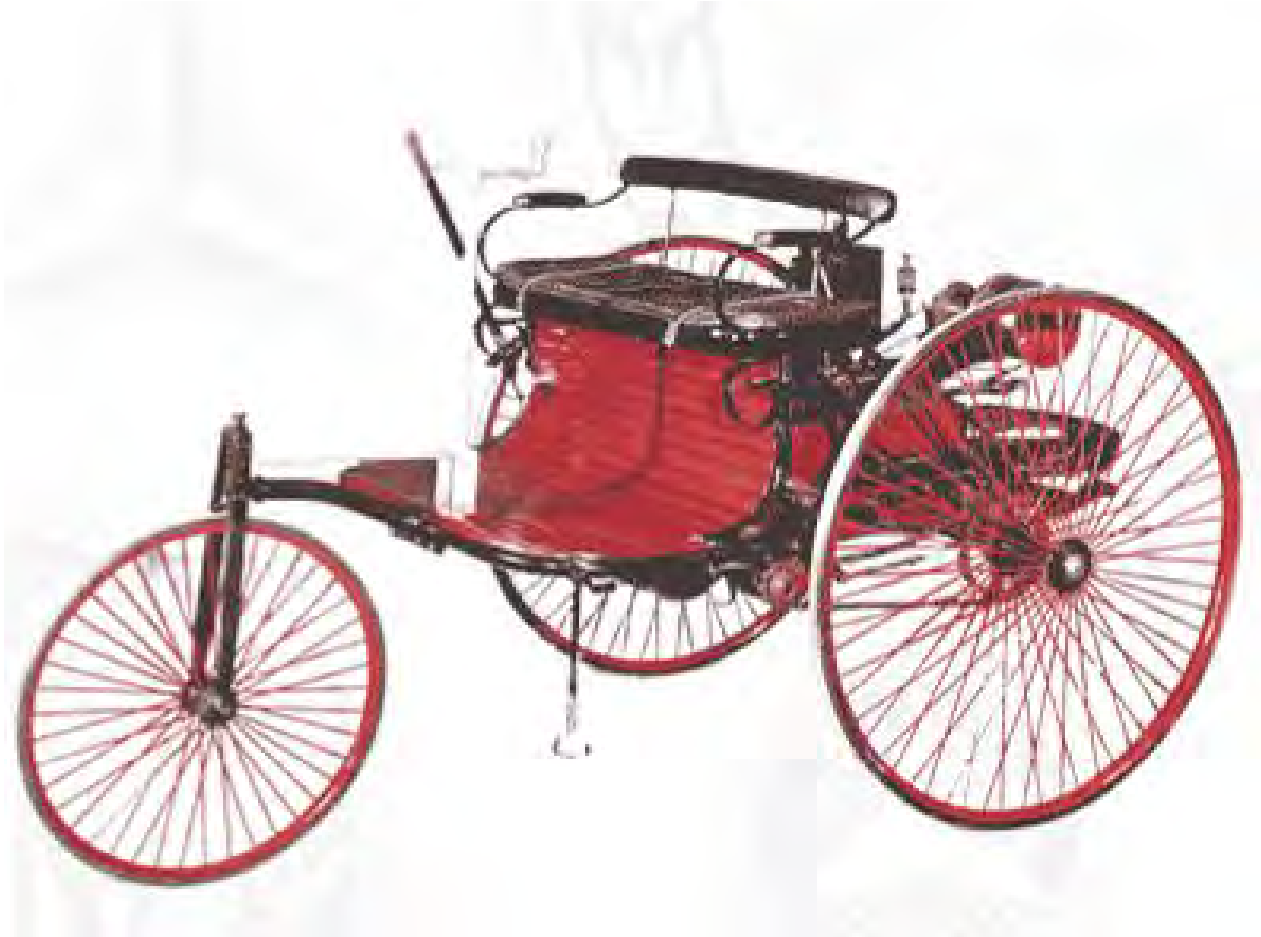
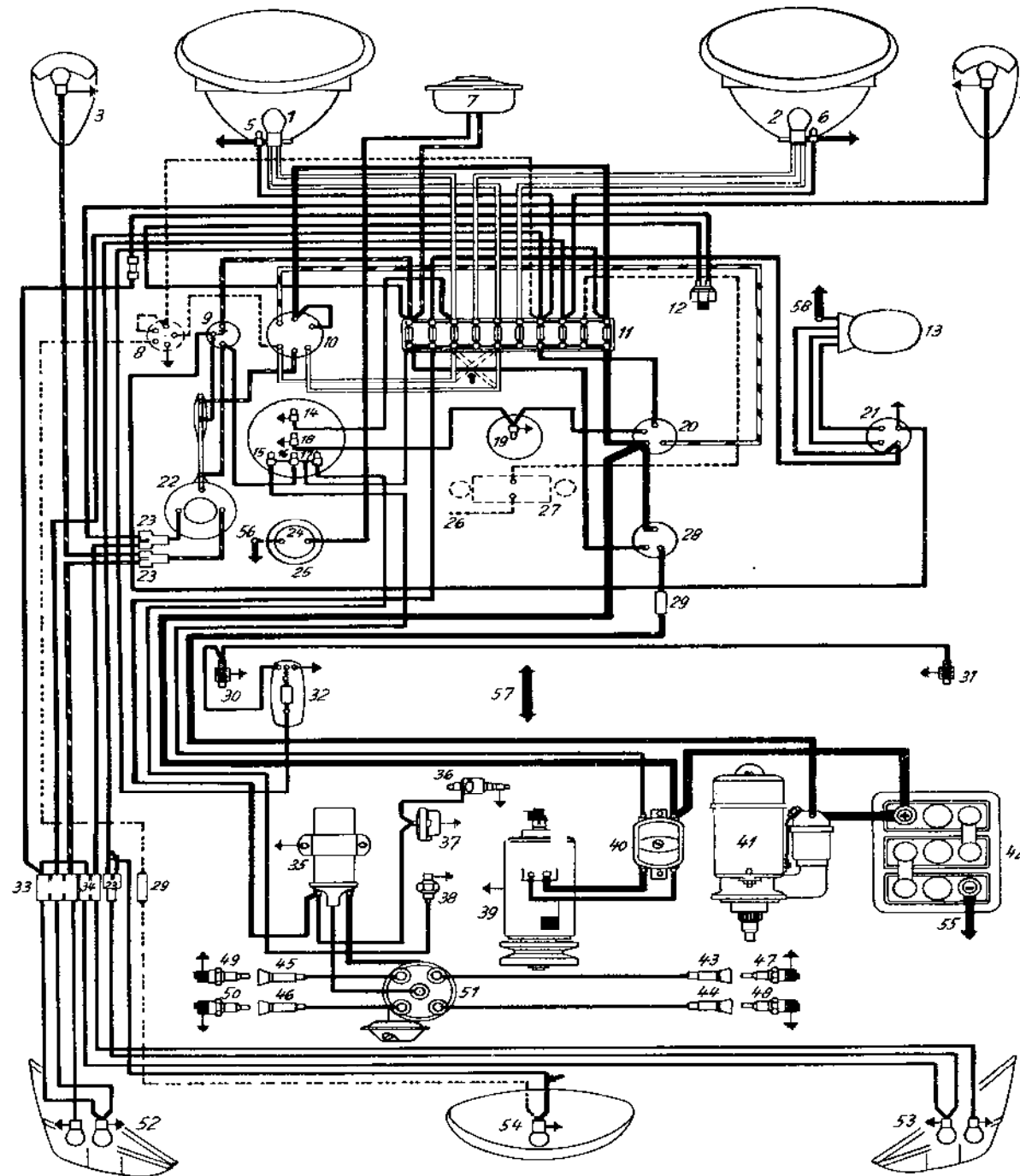


# 1. Einleitung



***„Schön einfach“..... keine Elektrik, keine Probleme.....  
Aber auch kein Komfort, keine Sicherheit usw. !***

Schaltplan eines  
VW Käfer (1964)



## Tendenzielle Vervielfachung der Elektrokomponenten seit 1975

*Elektromotore*



*Glühlampen*



*Relais*



*Steuergeräte*



*Steckverbinder*



*Schalter*







Und was heißt das?

...mehr  
Kabeln und  
Kupfer ist schwer  
und teuer!

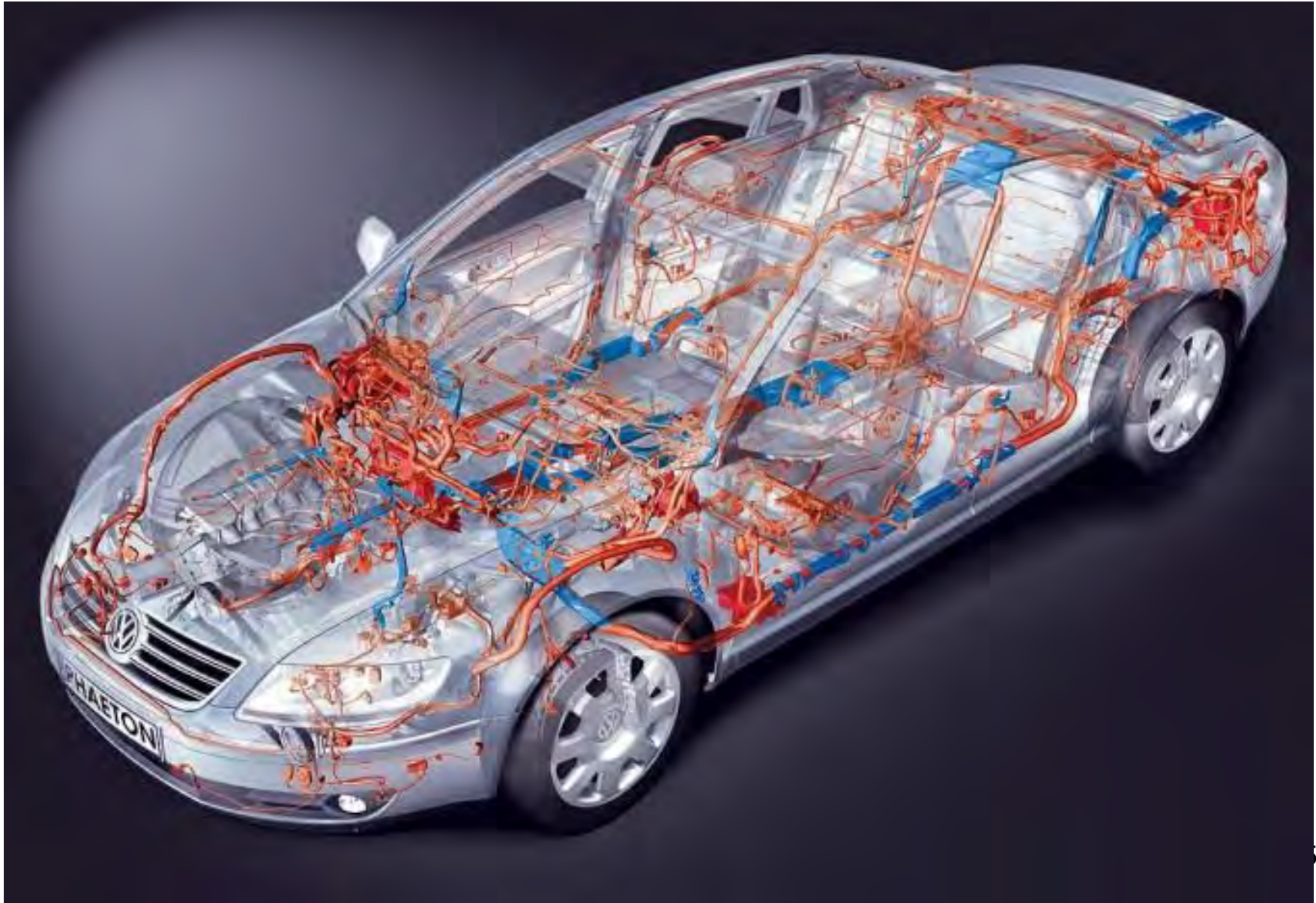


...mehr Kabel  
bedeuten auch  
einen größeren  
Montage-  
aufwand  
- also Kosten!

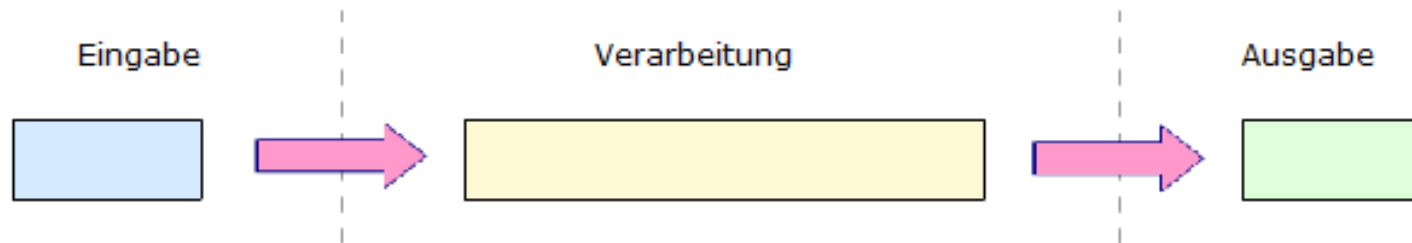
Also müssen wir  
die Menge der  
Leitungen  
reduzieren!

Stark komplexe  
Systeme erschweren  
die Fehlerdiag-  
nose und  
Reparatur!

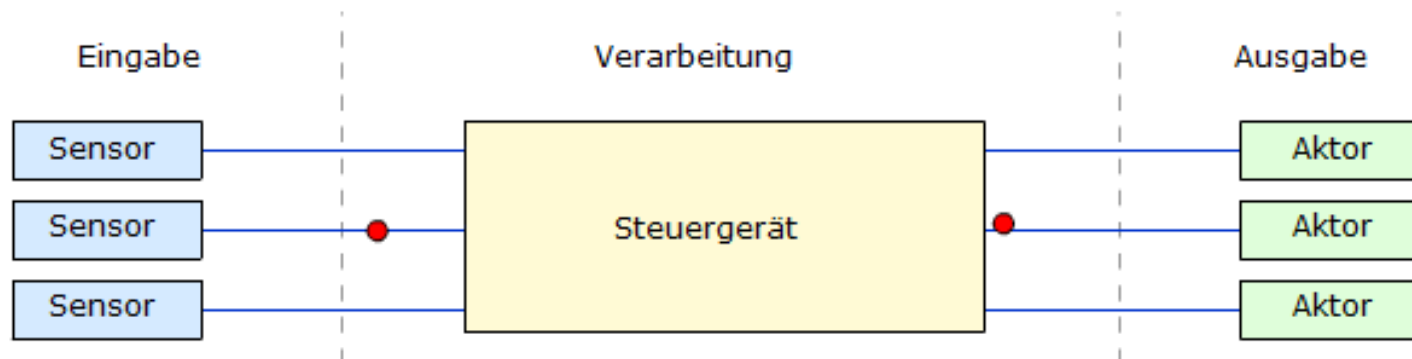
VW Phäton.... CAN-Bus; LIN-Bus; MOST Bus.....



## EVA Prinzip ! Was hat das mit dem CAN-Bus zu tun?



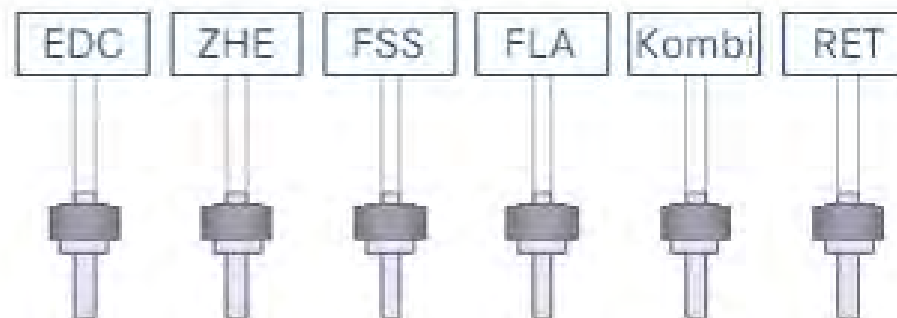
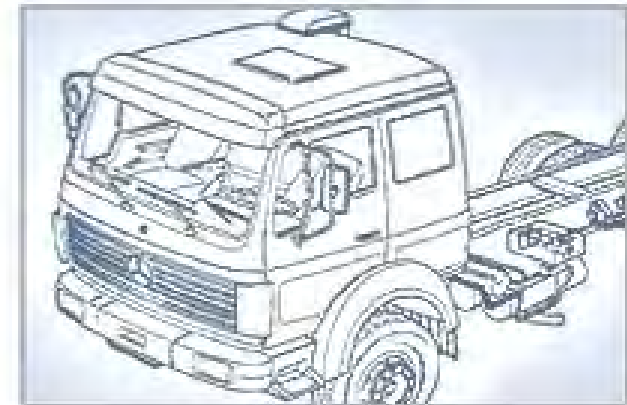
In jedem Kfz befinden sich zahlreiche solcher **informationsverarbeitender Systeme**: Die **Steuergeräte**. Sie weisen den folgenden Aufbau auf:



*Es wurde irgendwann zu viel!*

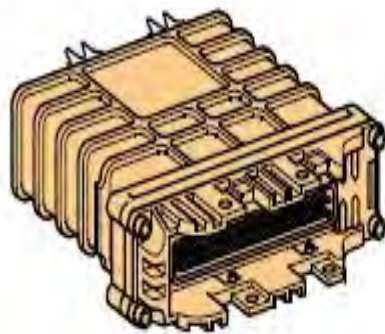
6 Kühlmitteltemperatur Sensoren für  
6 Steuergeräte.....

Heutzutage haben S-Klasse und Co.  
Über 60 STG.....



**Nicht zu vergessen die Verkabelung unter den STG!  
(Beispiel)**

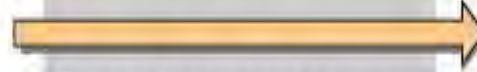
Steuergerät für Motronic



Motordrehzahl



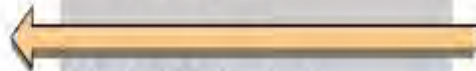
Kraftstoffverbrauch



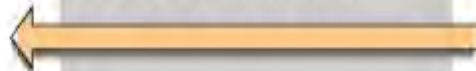
Drosselklappenstellung



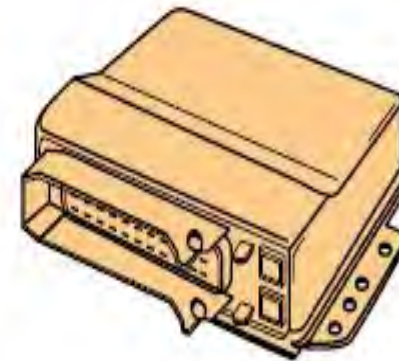
Motoreingriff



Hoch-/Rückschalten



Steuergerät für automatisches

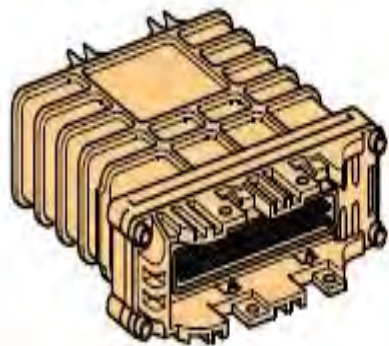


***Ein Sensor für alle!***  
***Digital Datenübertragung macht es möglich!***

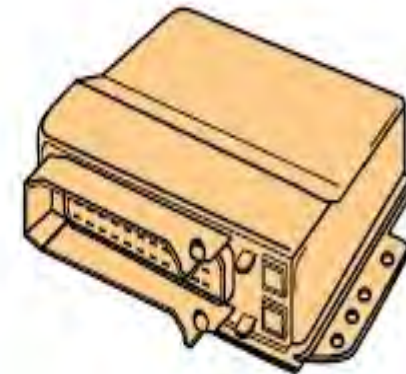




Steuergerät für Motronic



Steuergerät für automatisches



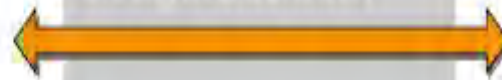
Motordrehzahl

Kraftstoffverbrauch

Drosselklappenstellung

Motoreingriff

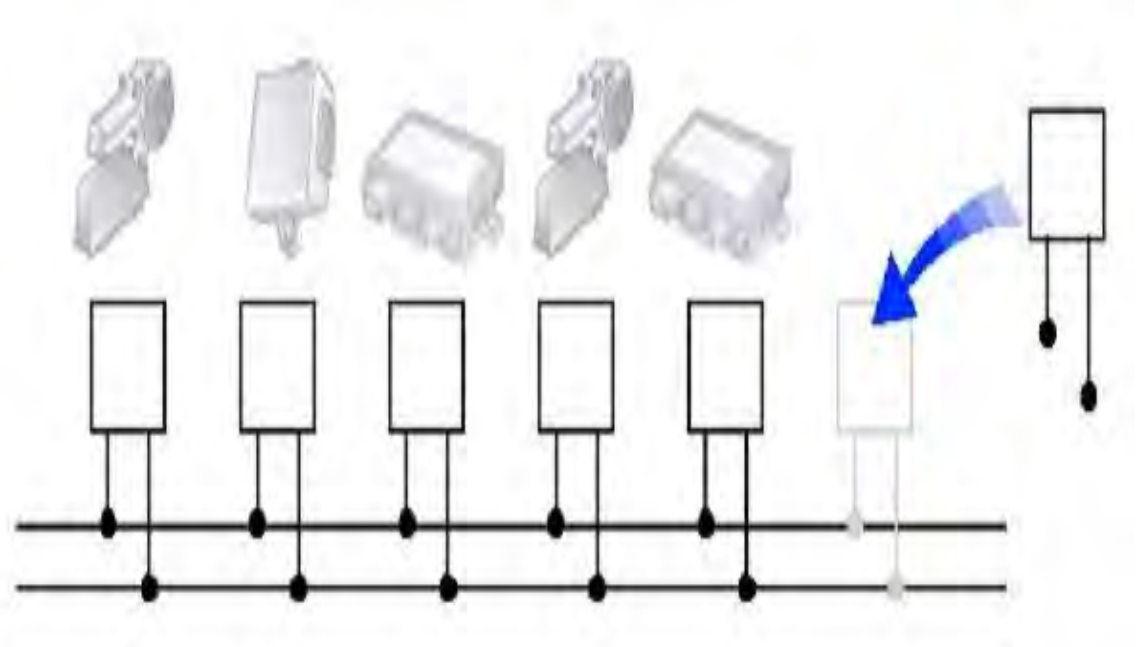
Hoch-/Rückschalten



# Warum also nun CAN-Bus?

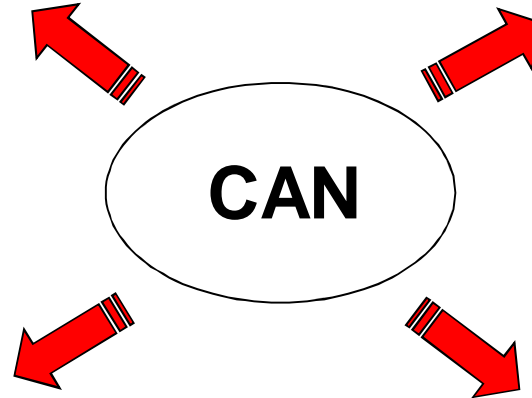
## Gründe für die Entwicklung:

- Gewichtsreduzierung
- Kosten
- Datenübertragung
- Diagnose
- Komfortfunktion
- Nachrüsten von Systemen
- Kaufpreis Fahrzeug





# Einsatzgebiete der Datenübertragungssysteme (CAN-Bus)



# 2. Codierung & Datenübertragung



Alphabet



Morsecode



Binärcode

Code

A, b, c ...

. - . . . . .

bi (Lateinisch) = zwei

01110001 01100010  
01100111

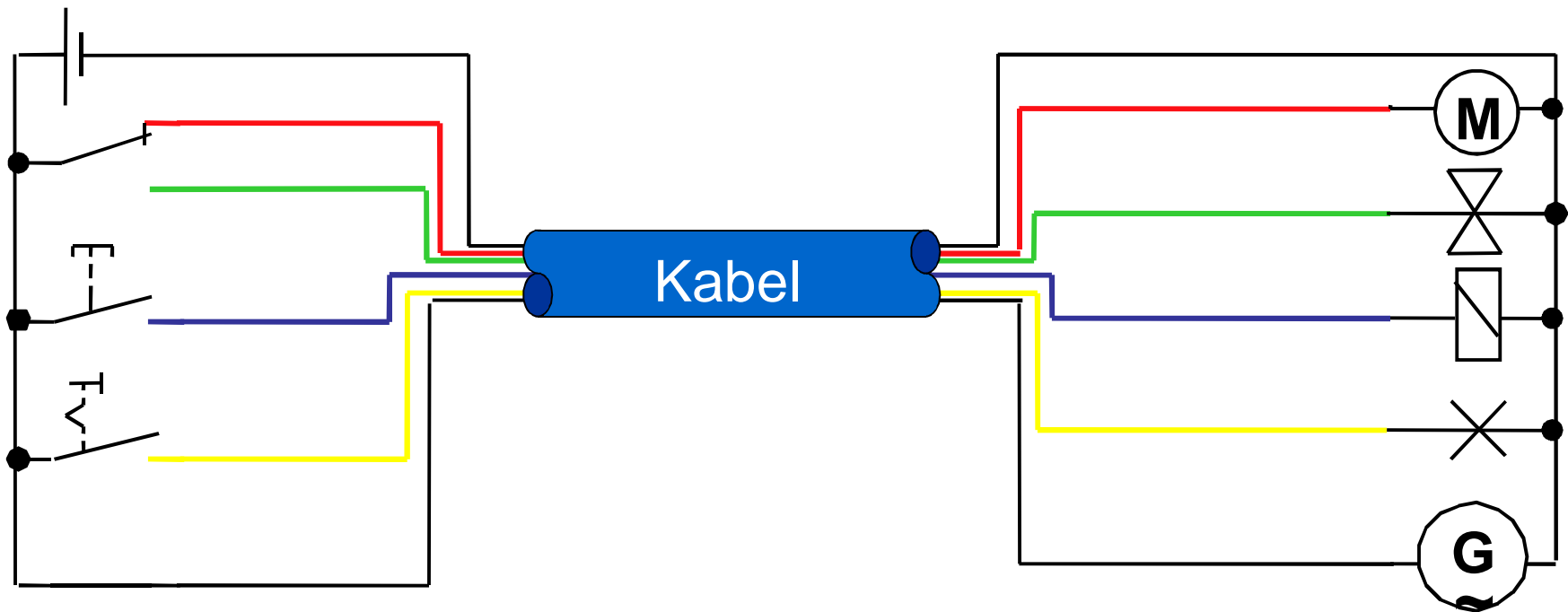
Information Hallo!

..... - . . . . . ---

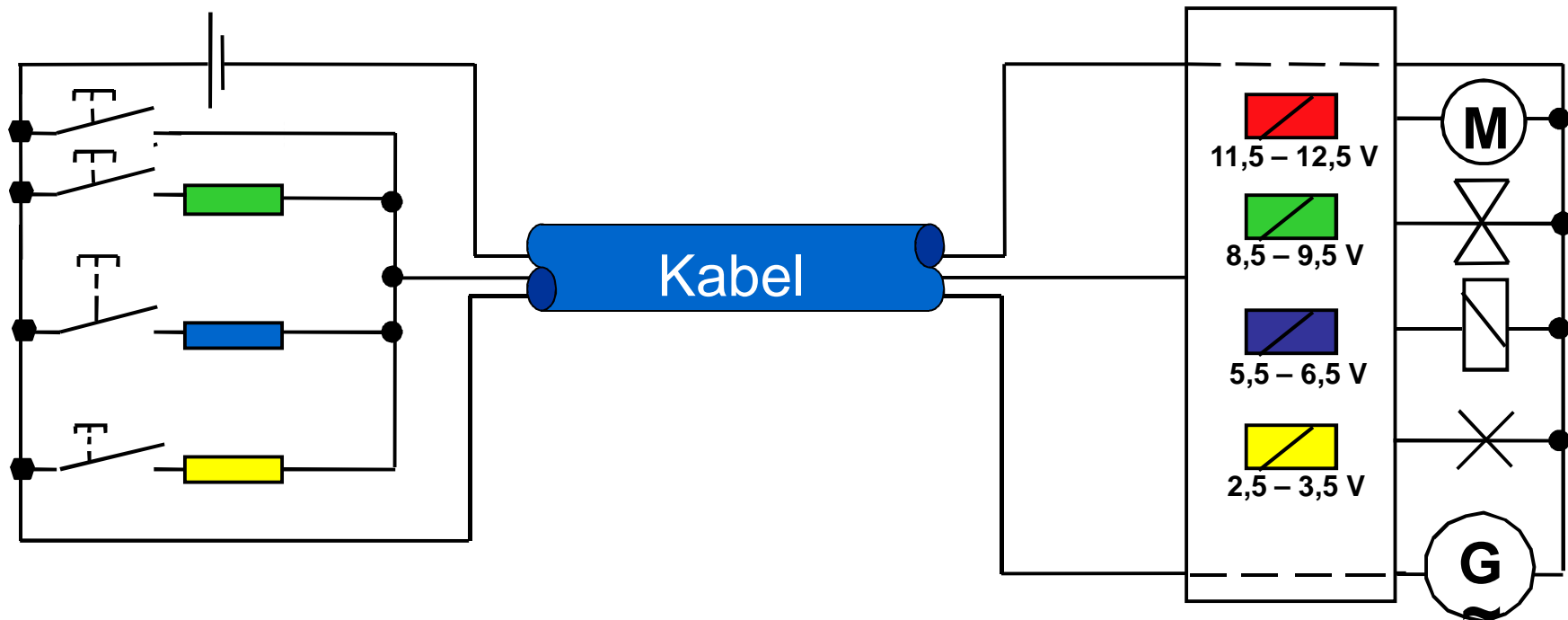
01101000 01100001  
01101100 01101100  
01101111

# 2. Spannungscodierung

Konventionell



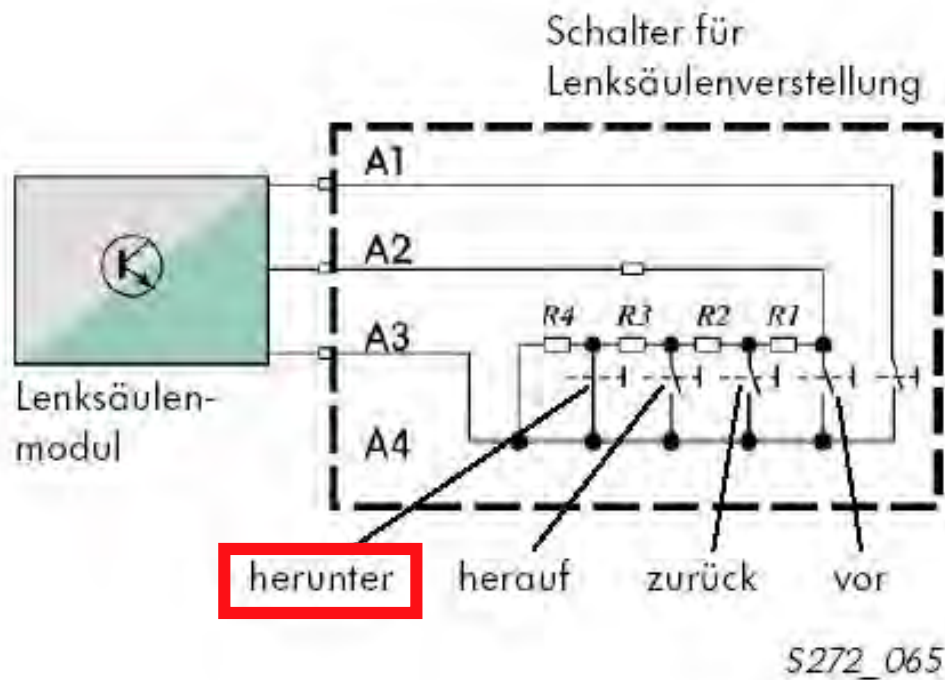
# Spannungscodierung



Die Methode mit dem geringsten Aufwand an elektronischen Bauteilen stellt die Kodierung (Verschlüsselung) der Ein- bzw. Ausschaltinformationen in unterschiedliche Spannungswerte dar.

# Beispiele für Spannungscodierung

## Elektrische Lenksäulenverstellung (VW Phaeton)



## Lenkradfernbedienung (Opel Omega)



# *Datenübertragung und Codierung*

Alphabet:            A bis Z    =>    26 Zeichen

Dezimal:            0 bis 9    =>    10 Zeichen            (Dezimal von 10)

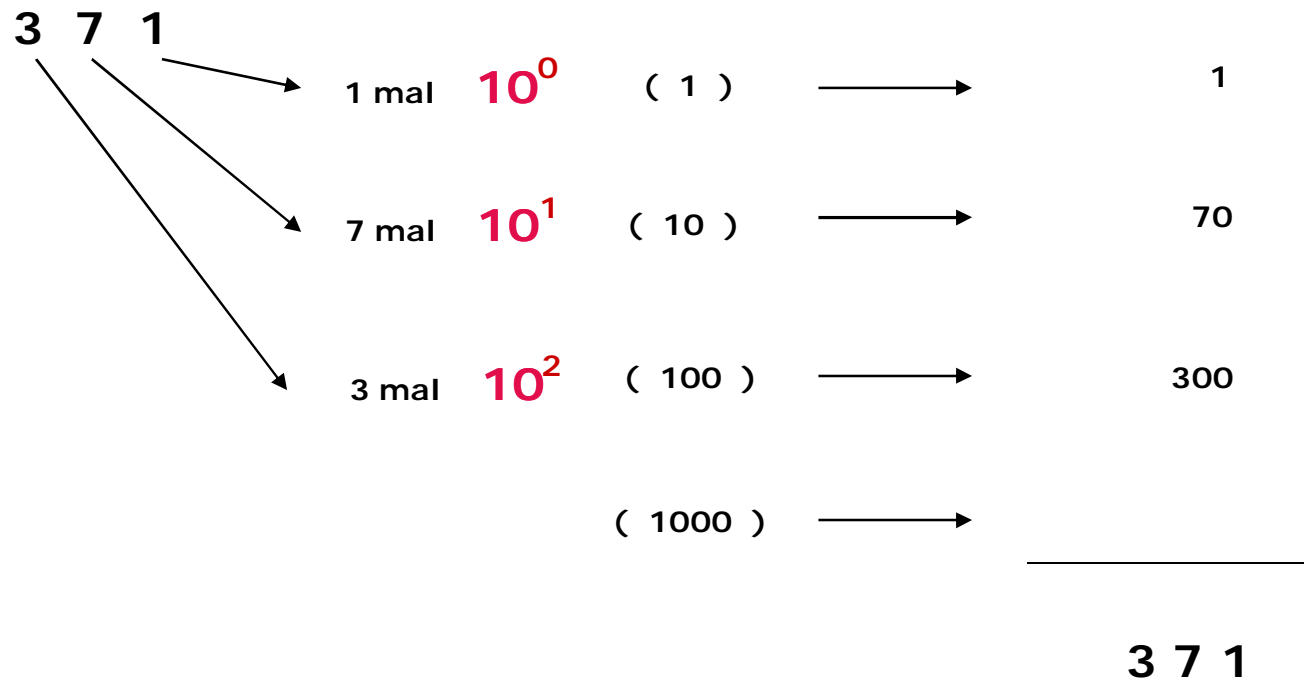
Hexadezimal:        0-9 & A-F =>    16 Zeichen

Binär:                0 & 1            2 Zeichen

# Das Dezimalsystem mit der Basis 10

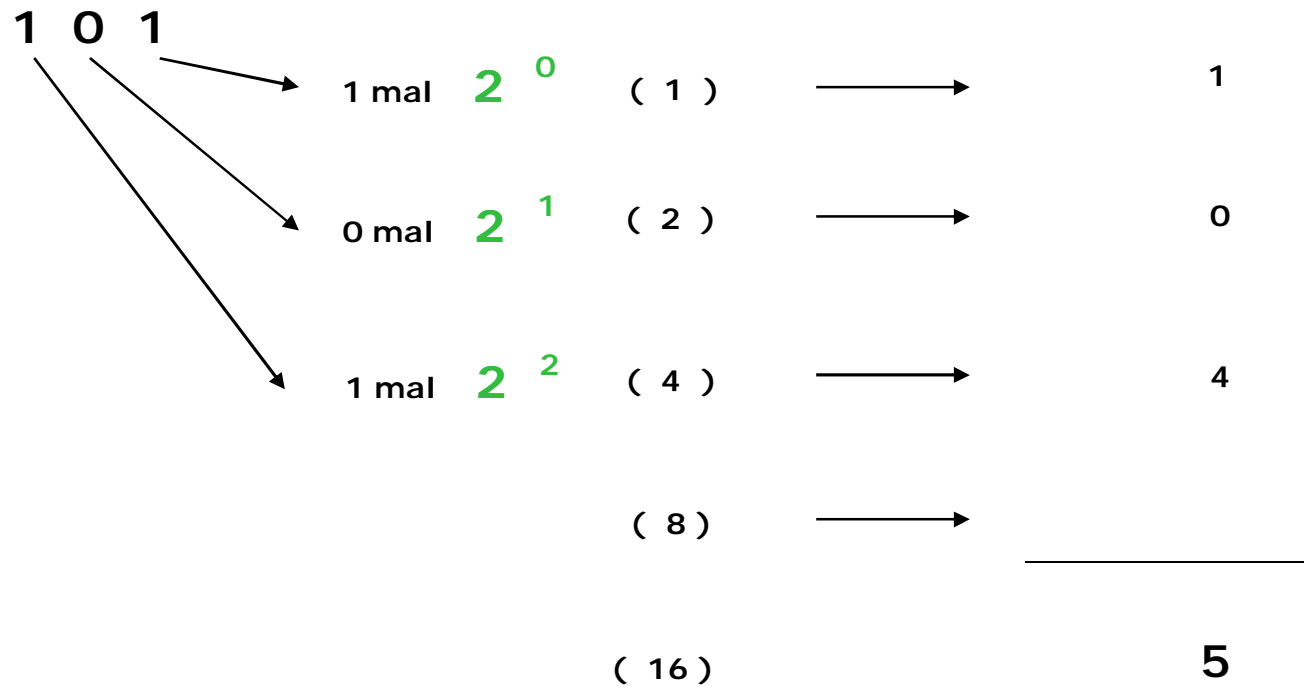
0 1 2 3 4 5 6 7 8 9

Was bedeutet denn eigentlich die uns bekannte Zahl 371



# Das Binärsystem (Dualsystem) mit der Basis 2

0 & 1





# 3. Einführung in die Digitaltechnik



= 1 Bit (Flasche voll entspricht **1** )

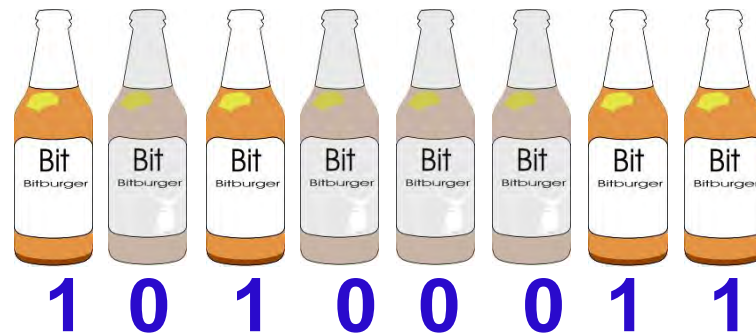
oder



= 1 Bit (Flasche leer entspricht **0** )

# Digitaltechnik: Das Byte

8 **Bit** entsprechen einem  
**Byte**



Somit lautet der  
Informationsge-  
halt dieses  
Bytes ?

Und 4 Bit sind ein ... Nibble!

# Was ist ein Bit?

Ein Bit ist die kleinste mögliche Datenmenge in der Digitalen Übertragungstechnik!

Ein Bit kennt nur 2 Zustände „1“ & „0“ !

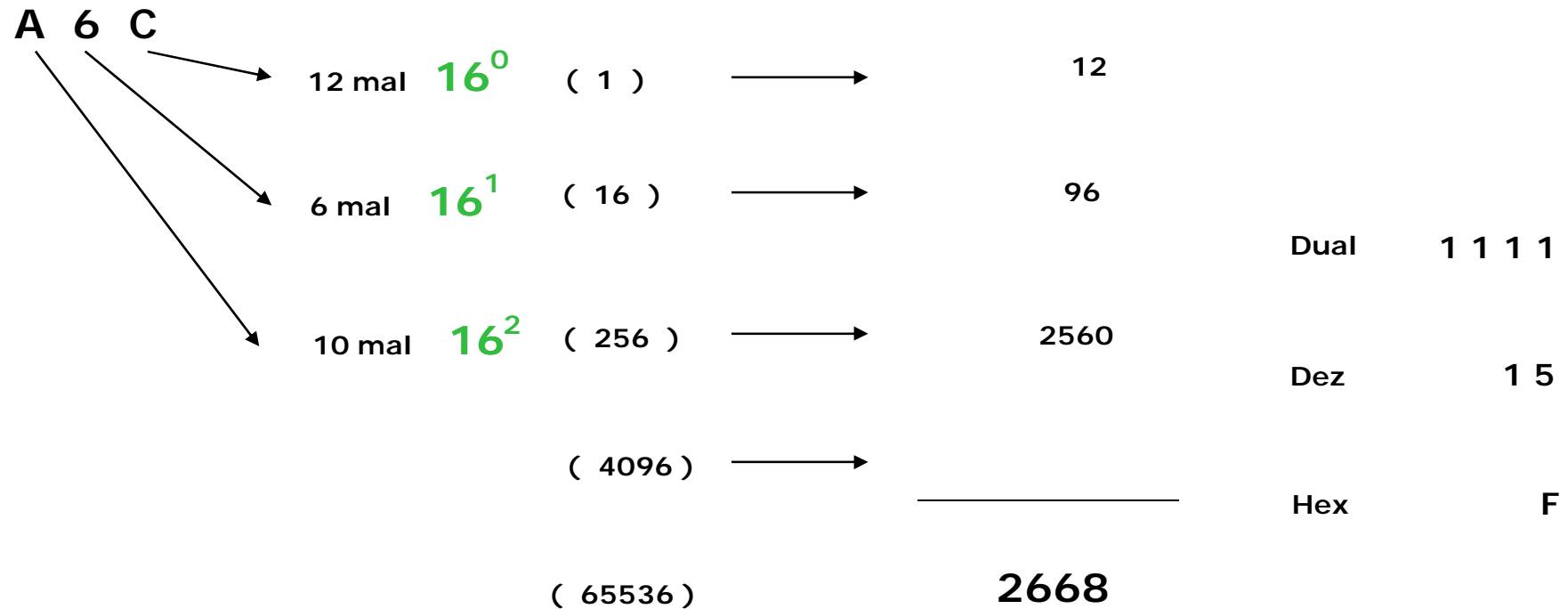
Wie kann man das verstehen?

**Beispielweise:**      **Lichtschalter ein= 1**  
                                 **Lichtschalter aus= 0**

**Transistor niedriger widerstand = 1**  
**Transistor hoher widerstand = 0**

# Das Hexadezimalsystem mit der Basis 16

0 1 2 3 4 5 6 7 8 9  
A B C D E F



# Dezimal, Binär und Hex

**Dezimal:** die Wertigkeit einer Ziffer ist die Zehnerpotenz  $10^i$  !

**Binärsystem:** auch Zweiersystem oder Binärsystem genannt, ist ein Zahlensystem, das nur zwei verschiedene Ziffern zur Darstellung von Zahlen benutzt um Daten zu Übertragen!

**Hexadezimalsystem:** Ein Byte sind 8 Bit und werden z.B. mit 1111 1111 dargestellt. Die gleiche Zahl hexadezimal dargestellt ist FF.

Durch **die hexadezimale Zahlendarstellung** wird nicht der Zahlenumfang des Dualsystems verändert, man erreicht dadurch eine strukturierte Leseart da immer 4 Bit zu einer Einheit zusammengefasst werden. Die hexadezimale Zahlendarstellung dient also auch dem Zweck größere Dualzahlen einfacher darzustellen und damit die Verarbeitungsgeschwindigkeiten zu steigern.

## Dezimal

## Binär

## Hex

0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	a
11	1011	b
12	1100	c
13	1101	d
14	1110	e
15	1111	f

# Übung

<i>Dezimal</i>	<i>Binär</i>	<i>Hexadezimal</i>
209		
	10010001	
		41
249		
	10111010	

# Übung

<i><b>Dezimal</b></i>	<i><b>Binär</b></i>	<i><b>Hexadezimal</b></i>
209	11010001	D1
145	10010001	91
65	1000001	41
249	11111001	F9
186	10111010	BA



## 4. Maßstab für die Geschwindigkeit der Datenübertragung ist die

Für Speicherangaben wird noch das „digitale“ Kilo von 1024 benutzt, das mit einem großen „K“ abgekürzt wird!

übertra

enübertr

bit/s (kb

Aha, dann entsprechen also 62,5 KBit/s genau 64.000 Bit/s!

Und demzufolge wären ein „digitales“ Kilo Mehl genau 1024 Gramm ;-)

Beispiel . Motorola

$$500 \text{ kb/s} = 500.000 \text{ Bit/s}$$



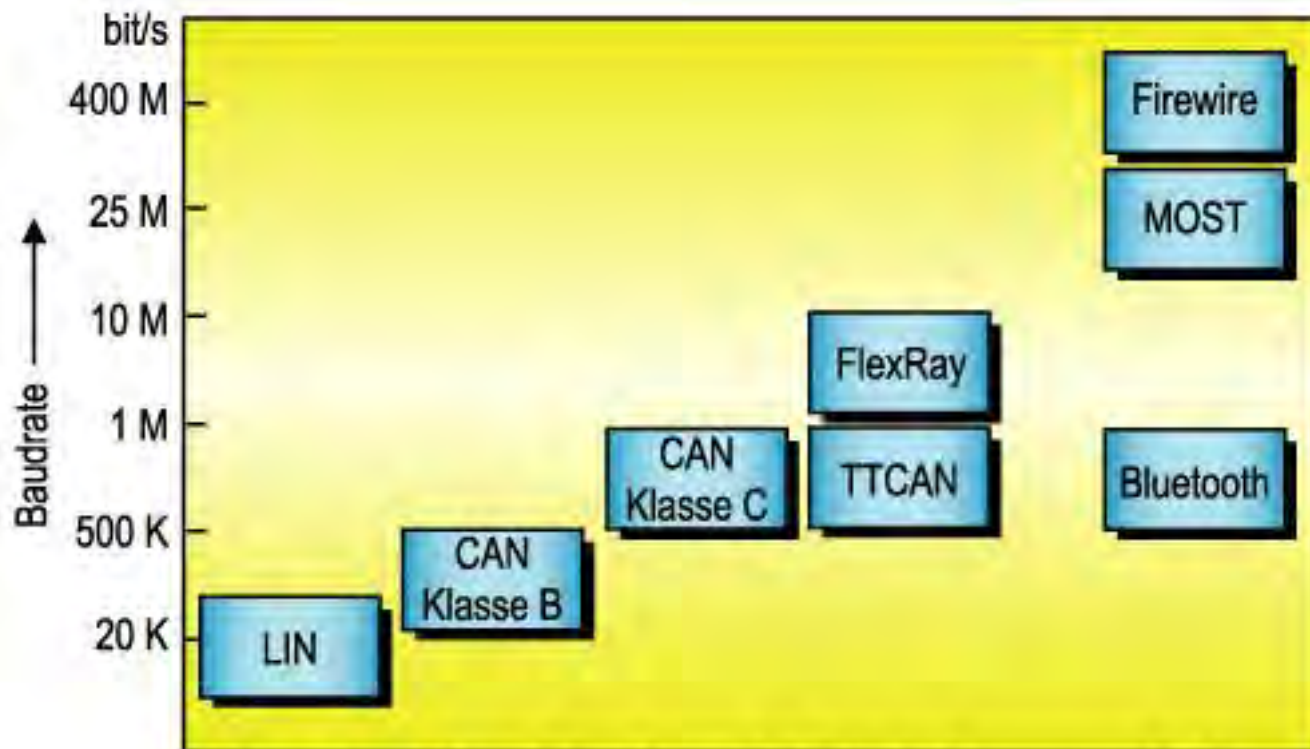
Innenra



$$62,5 \text{ kb/s} = 62.500 \text{ Bit/s}$$



## 4. Maßstab für die Geschwindigkeit der Datenübertragung ist die



# Serielle Datenübertragung

Beispiel :



# Parallele Datenübertragung

Beispiel :



## *Zusammenfassung serielle /parallele Datenübertragung*

### **Serielle Datenübertragung :**

Es werden die Bits nacheinander auf einer Leitung übertragen

- Vorteil : geringer Verkabelungsaufwand
- Nachteil : langsamere Datenübertragung

### **Parallele Datenübertragung :**

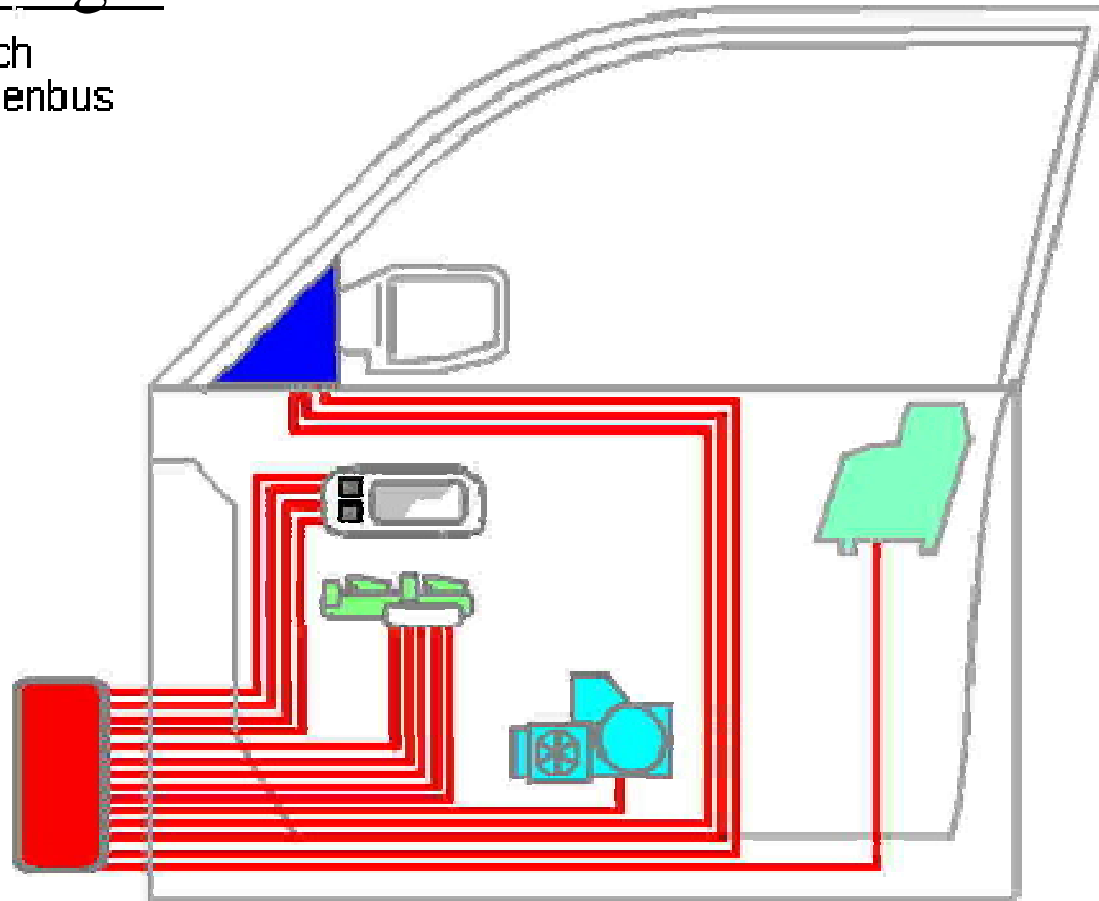
Es werden die Bits parallel auf verschiedenen Leitung übertragen

- Vorteil : schnellere Datenverarbeitung
- Nachteil : hoher Verkabelungsaufwand (für Kfz zu aufwändig)

# 5. Vernetzung im Kfz (Beispiel an der Tür/Spiegel/Fensterheber)

## Alte Technologie

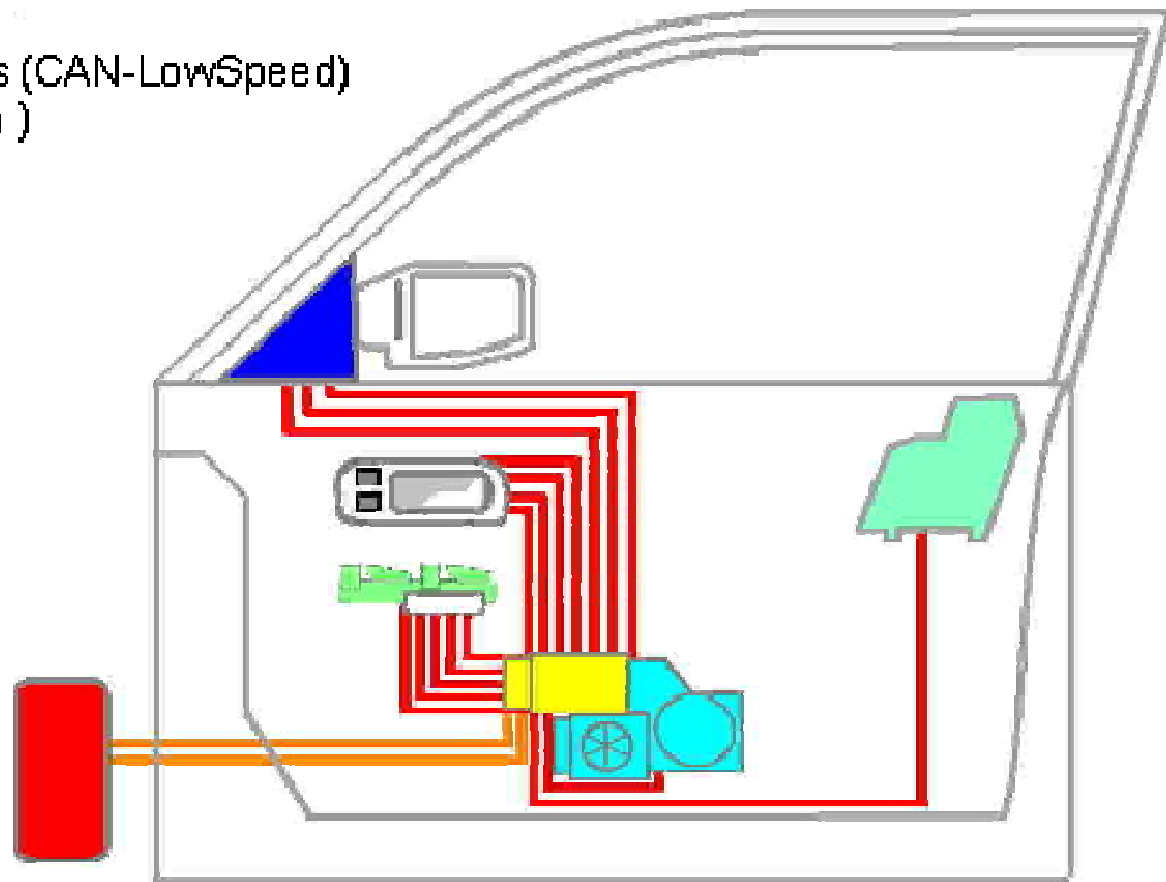
Datenaustausch  
ohne CAN-Datenbus



# *Bussysteme im KFZ*

## Neue Technologie

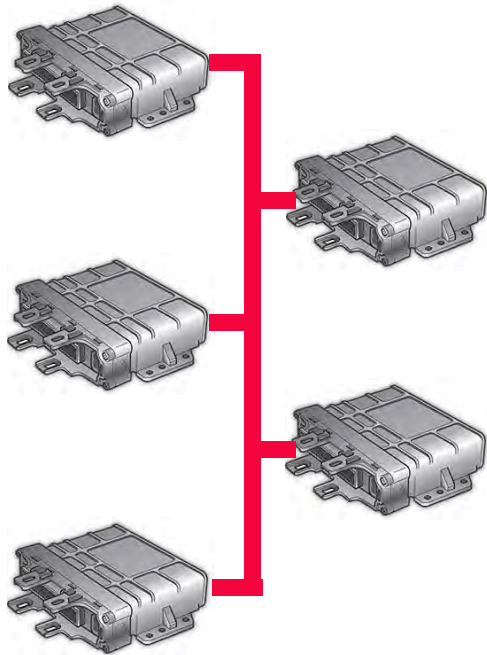
Datenaustausch  
mit CAN Datenbus (CAN-LowSpeed)  
( 2 Datenleitungen )



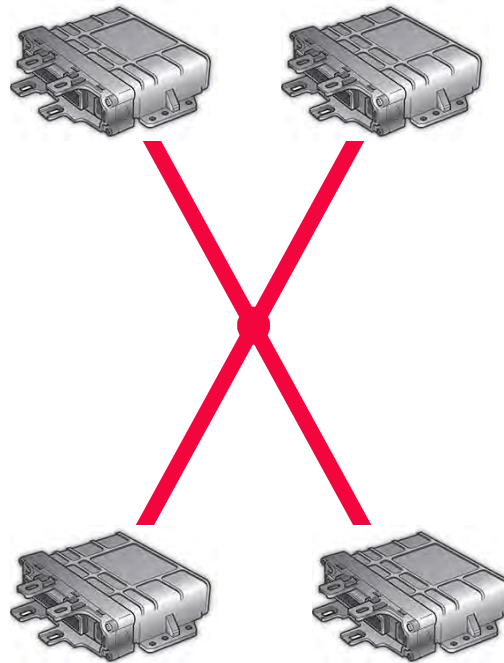
# Topologie der Bussysteme

Netzwerkstrukturen unterteilt man hauptsächlich in 3 (Bus) Topologien:

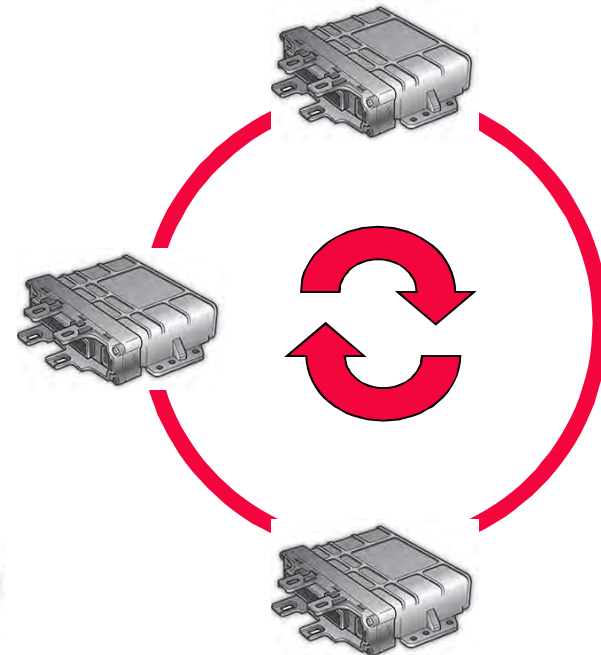
- Sterntopologie
- Ringtopologie
- Lineare Bustopologie



Strang



Stern



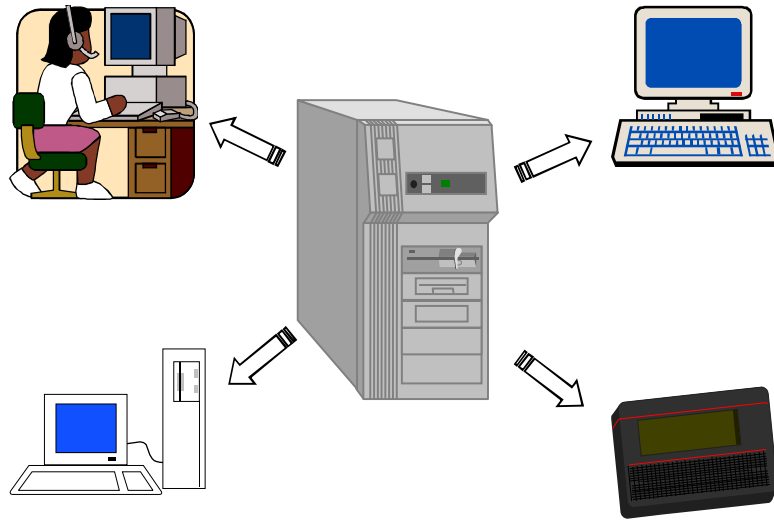
Ring



# Topologien II

## Unterteilung in die wichtigsten Netzwerktopologien

### Die Sterntopologie



**Praktische Anendungen** der Sterntopologie ist die Bürokommunikation, wo mehrere Teilnehmer an einem Master angeschlossen sind.

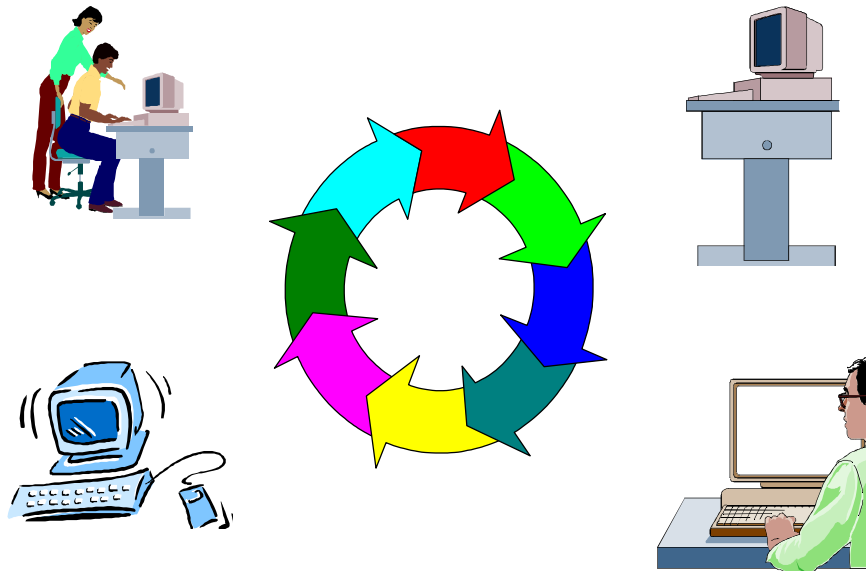
**Der Vorteil** der Sterntopologie ist, wenn ein Slave ausfällt, so treten im Netz keine weiteren Störungen auf. Das System kann hohe Geschwindigkeiten erreichen.

**Der Nachteil** der Sterntopologie ist, wenn der Master ausfallen sollte, dann ist auf dem gesamten Bus keine Kommunikation mehr möglich. Jeder Busteilnehmer muss auch mit dem Master verbunden werden, was einen hohen Kabelaufwand zur Folge hat.

# Topologien III

## Unterteilung in die wichtigsten Netzwerktopologien

### Die Ringtopologie



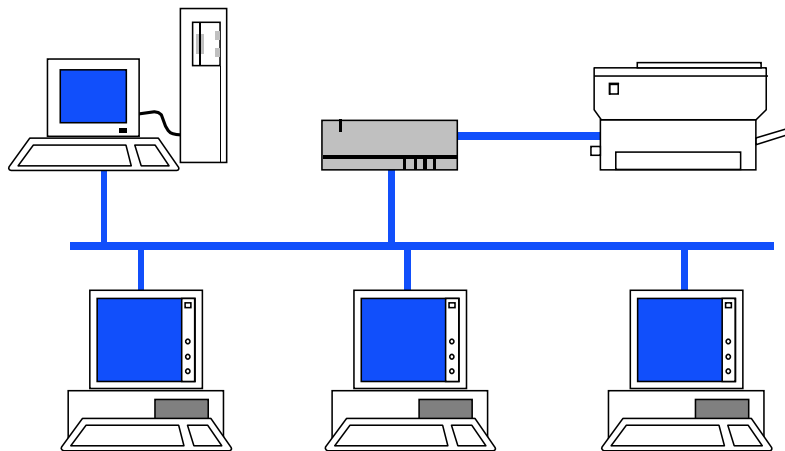
**Der Nachteil** der Ringtopologie ist, wenn ein Teilnehmer ausfällt, dann bricht auch die gesamte Kommunikation auf dem Datenbus zusammen.

**Der Vorteil** der Ringtopologie ist der relativ kleine Kabelaufwand und die damit geringeren Kosten.

# Topologien V

## Unterteilung in die wichtigsten Netzwerktopologien

### Die lineare Bustopologie

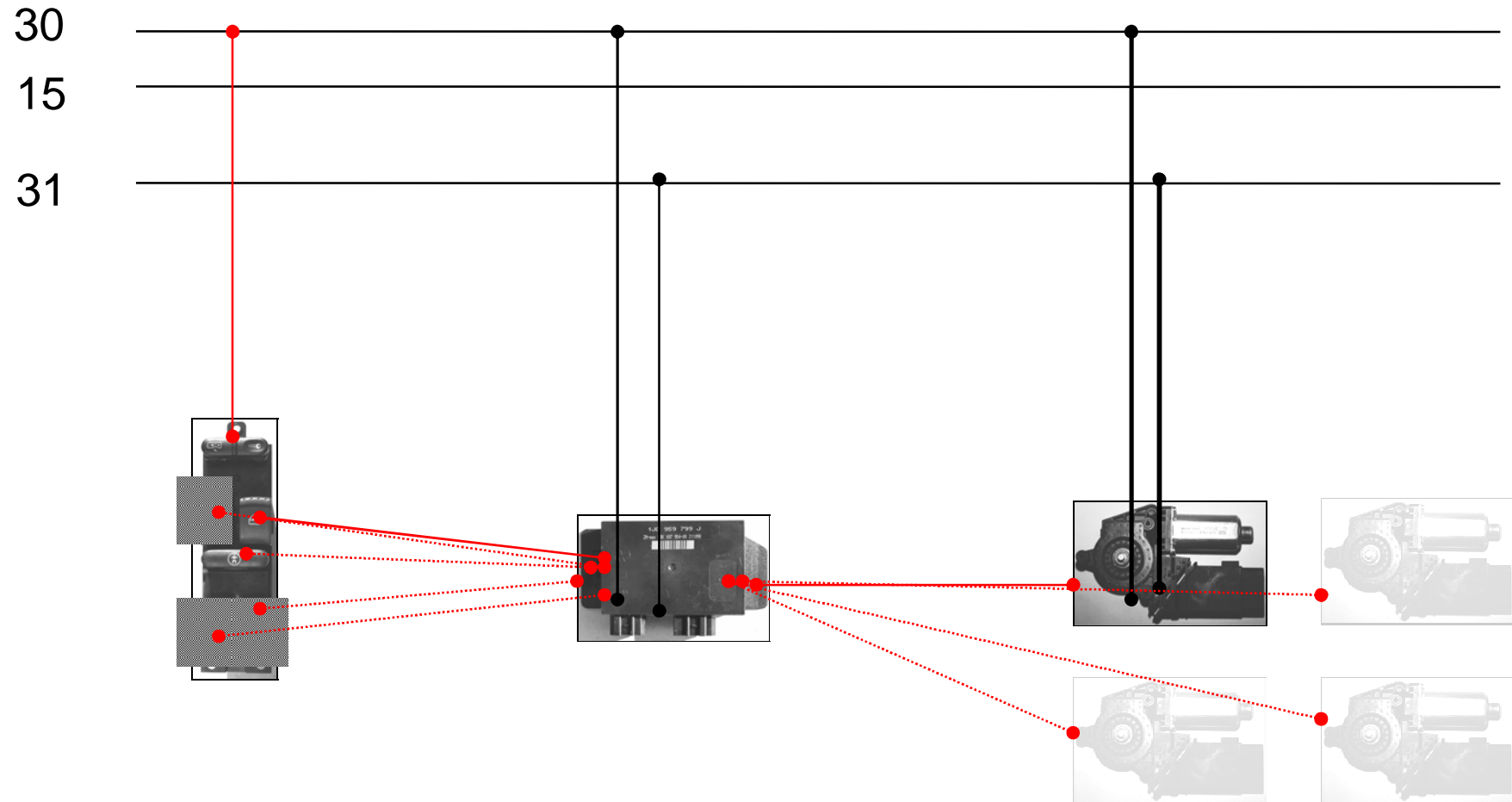


### Anforderungen an das lineare Bussystem im Kfz

- einfacher und kostengünstiger Aufbau durch verdrehte 2-Draht-Leitung
- kurze Übertragungszeiten „Echtzeit“
- hohe Datenübertragungsraten
- Datenkapazität von 0 bis 8 Byte
- hohe Datensicherheit / Störsicherheit
- hohe Verfügbarkeit des Gesamtsystems
- Lokalisation ausgefallener Stationen
- Konfigurationsflexibilität
- prioritätsgesteuerte Kommunikation
- zerstörungsfreies Buszugriffsverfahren
- Multi Master Prinzip

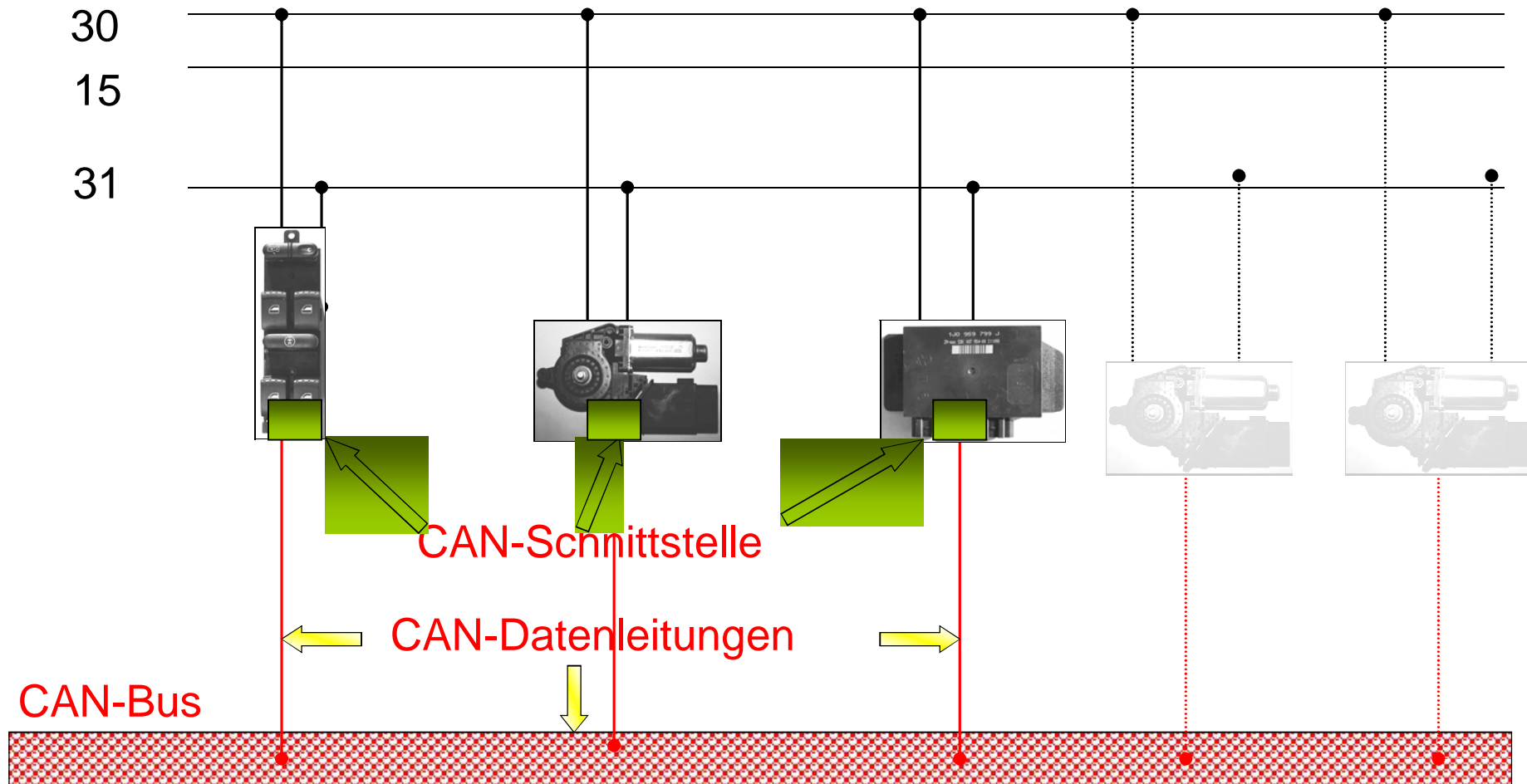
# CAN-Bus-Komponenten

## Alte Technologie



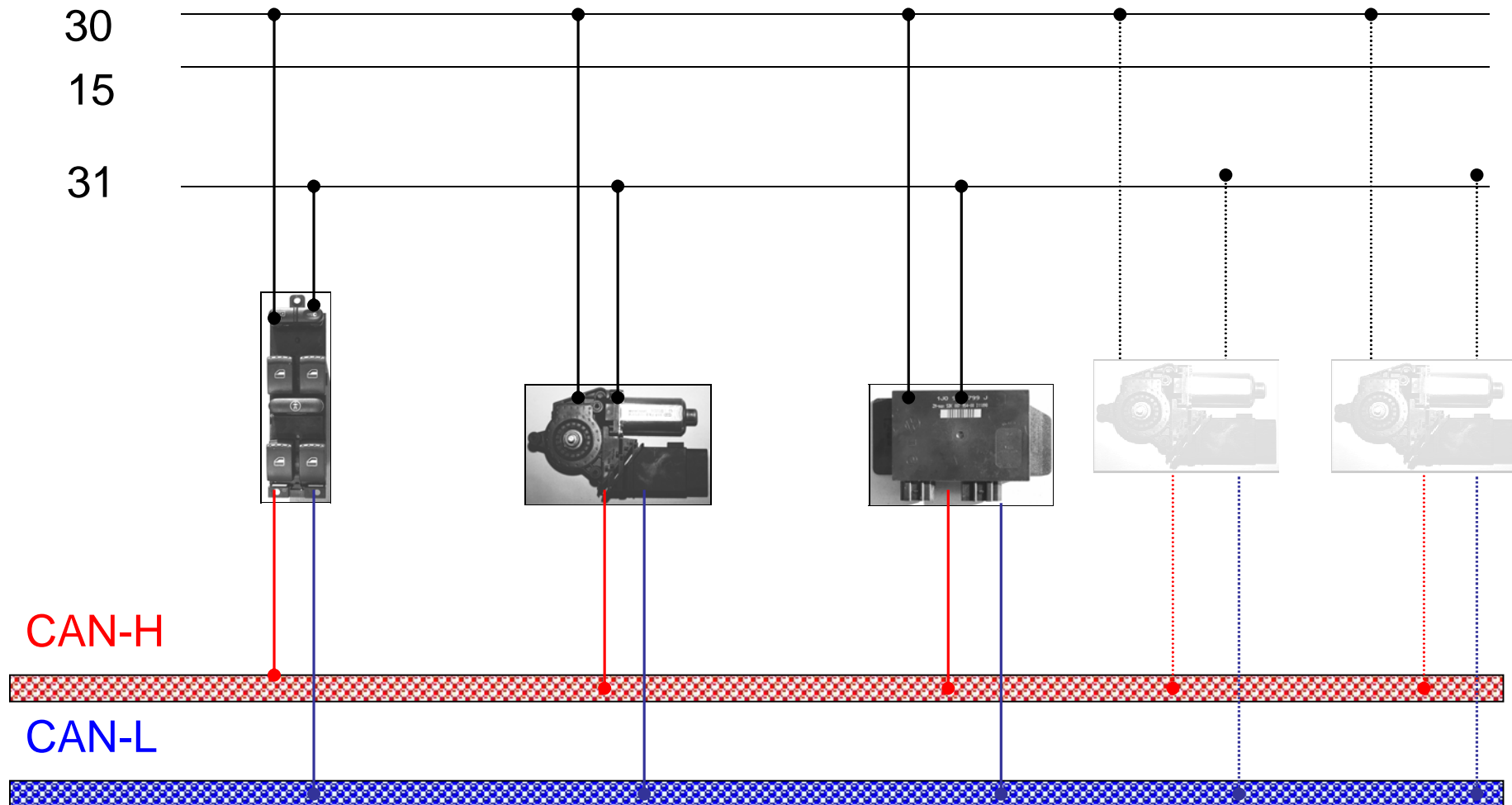
# CAN-Bus-Komponenten

## Neue Technologie

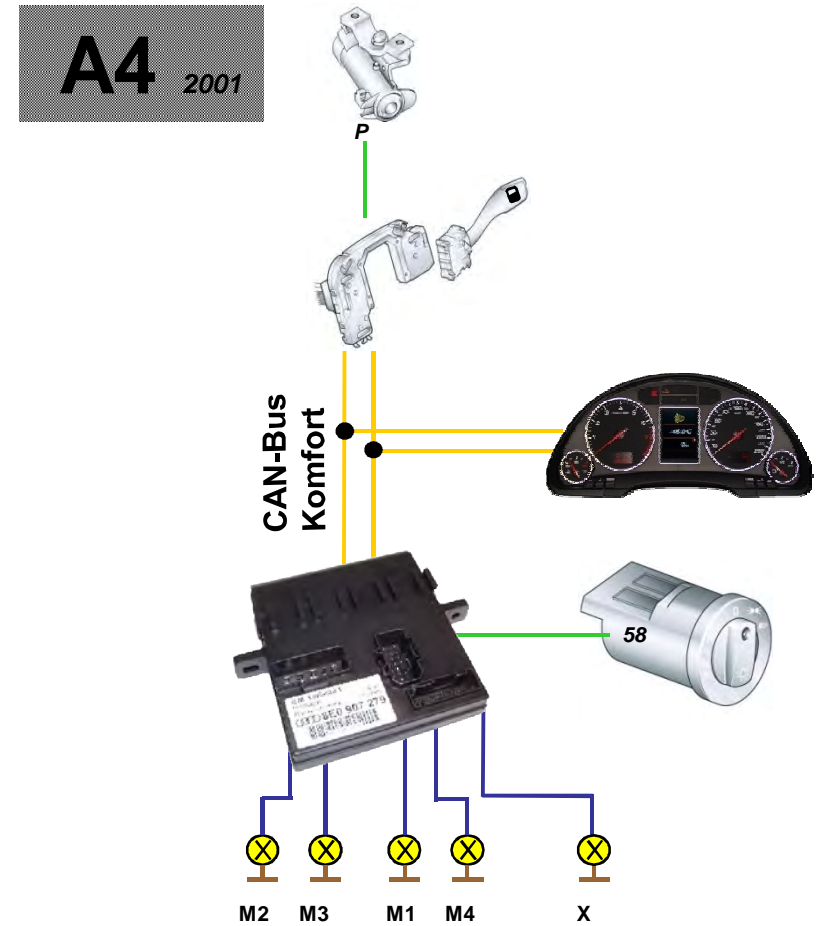
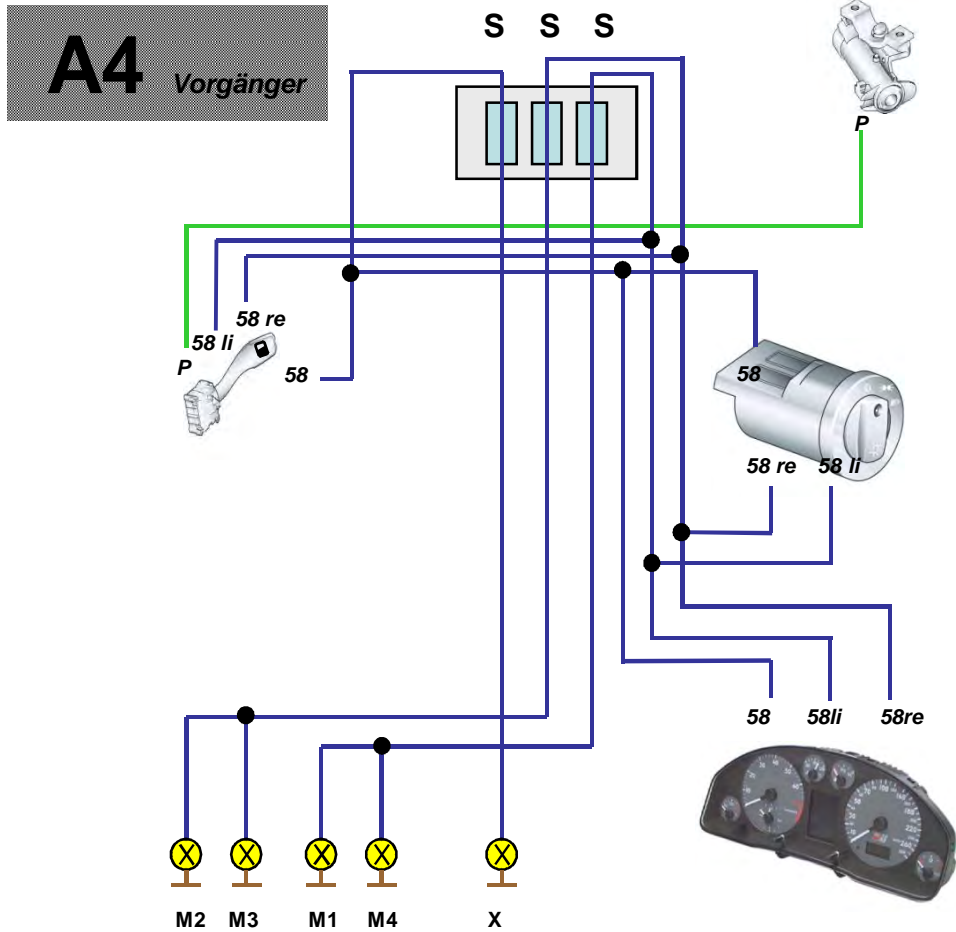


# CAN-Bus-Komponenten

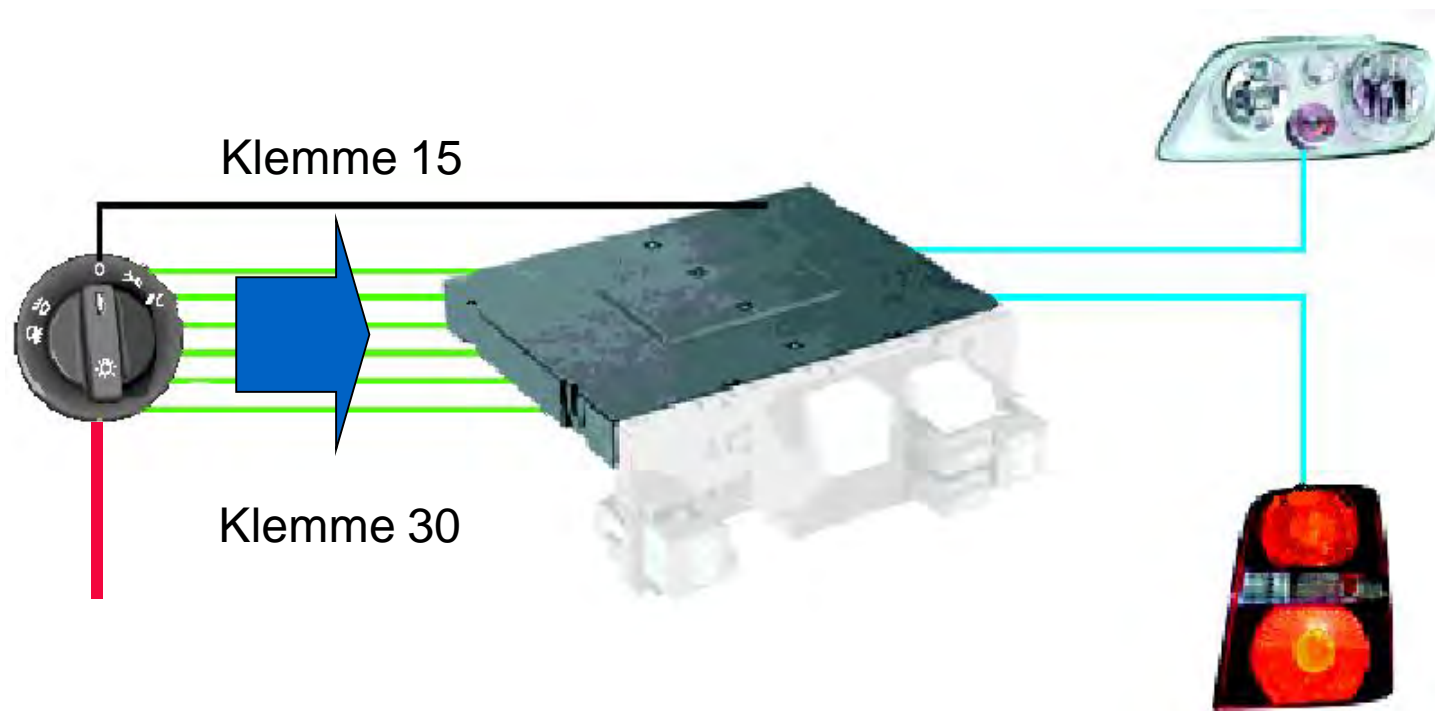
## CAN-Anschlussstechnik



# Vernetzung im Kfz (Beispiel an der Beleuchtung)



# Vernetzte Funktionen Lichtsteuerung Touran

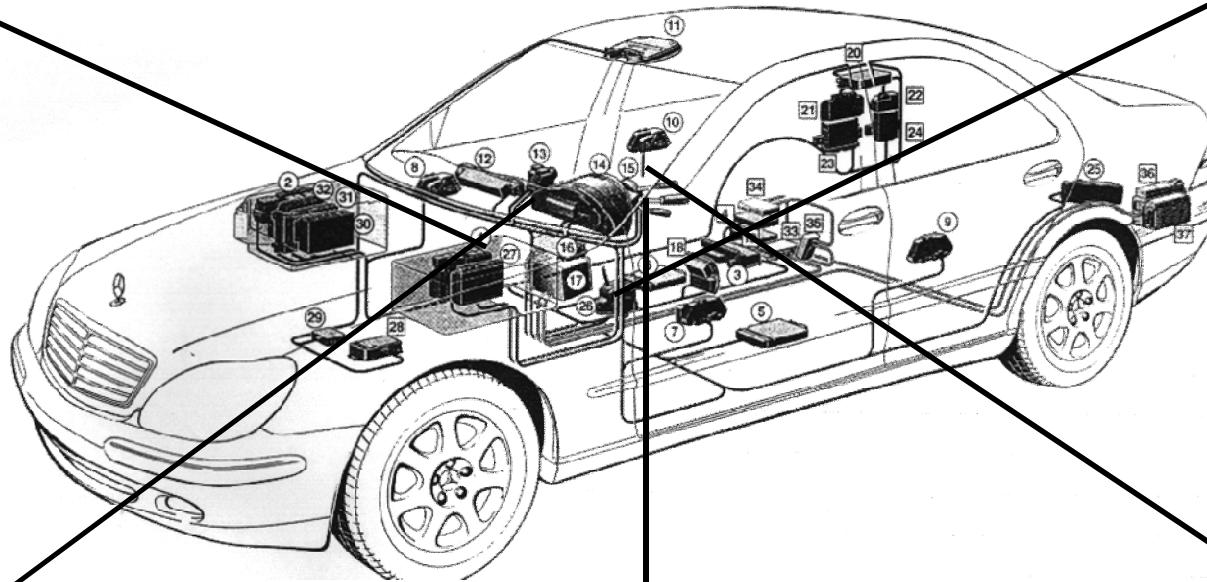




# 5. Vernetzung im Kfz/ Bussysteme

**Motor-Bus  
(High-Speed)**

**MOST-Bus**



**LIN-Bus  
(Klima/ Heizung)**

**Innenraum-Bus  
(Low Speed-Bus)**

**Neuartige-Bus  
(TT-CAN, Flexray, Byteflight)**

# *Einsatzgebiete der verschiedenen Bussysteme*

## **Mechatronische Applikationen/ Smart Sensor und Smart Aktor (Klasse A )**

z.B. Reifendrucksensoren, Regensensors,  
Diebstahlschutzfunktionen, Lüfter...

## **Karosserieelektronik (Klasse B)**

z.B. Klimaanlage, Sitz- und  
Spiegelverstellung, Lampensteuerung und  
Zentralverriegelung ...

## **Antrieb/Fahrwerk (Klasse C)**

z.B. Steuergeräte des Motors, des  
Getriebes, des Fahrwerks und der Bremsen

## **Sicherheitskritische Systeme**

z.B. Sicherheitssysteme (Airbag)

## **Mobile Kommunikation/ Telematik**

Autoradio, Navigation, CD-Wechsler,  
Bedien- und Anzeigeeinheit, Internet,  
E-Mail, DVD- Player, TV, ...

## 6. Multiplex System

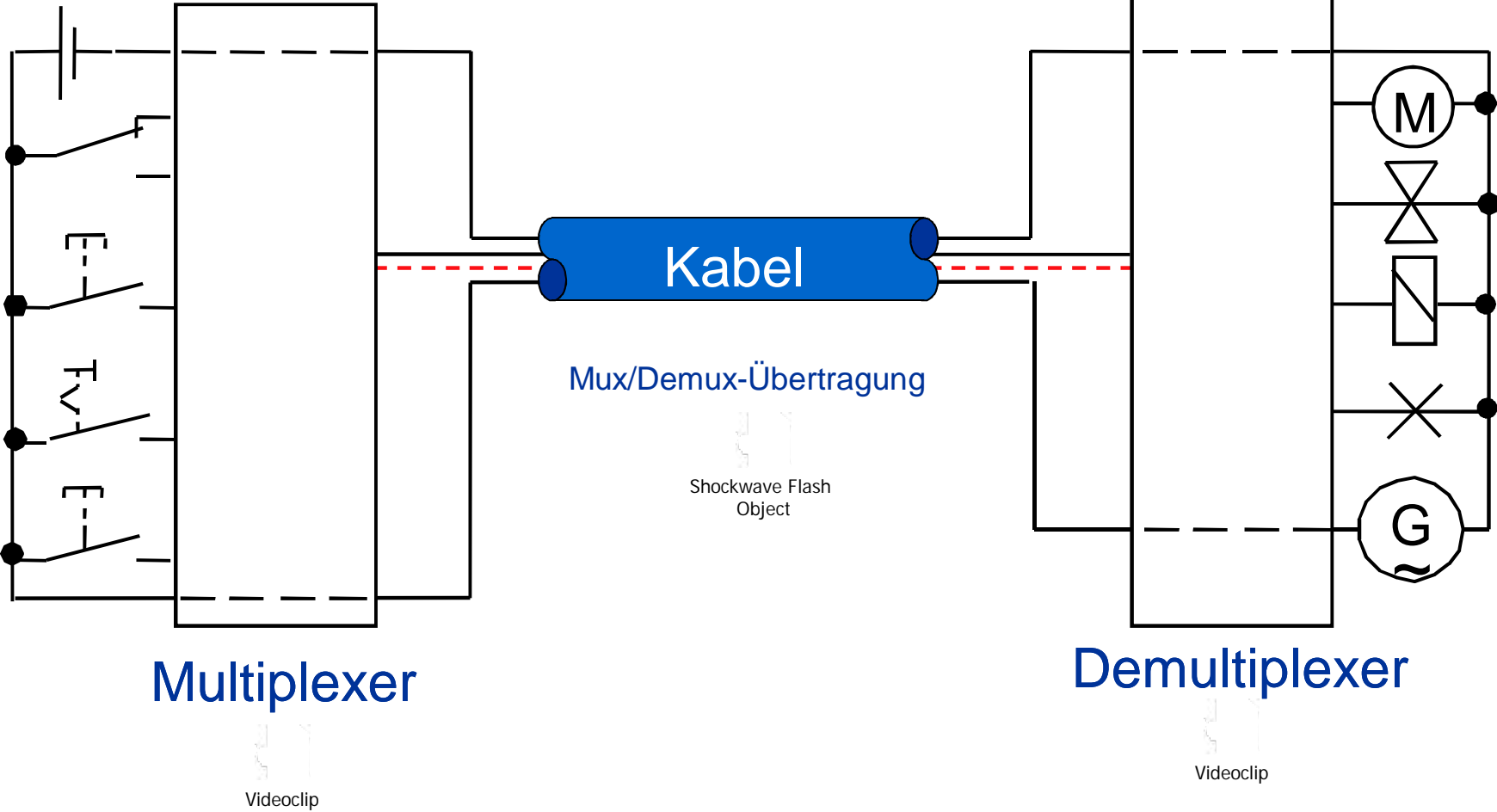
Um eine Vielfalt von Signalen über eine einzige Leitung zeitlich nacheinander zu übertragen, wurde bereits 1990 das Multiplex-System entwickelt.

Es kann 10 bis 100 KBit/s übertragen.

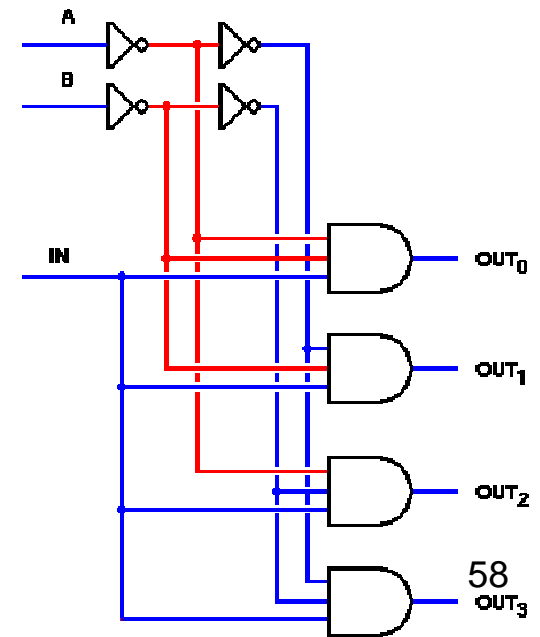
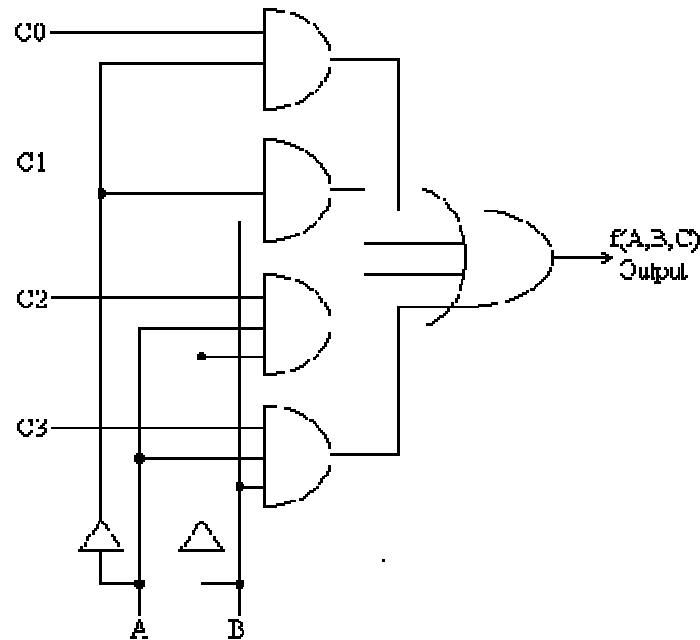
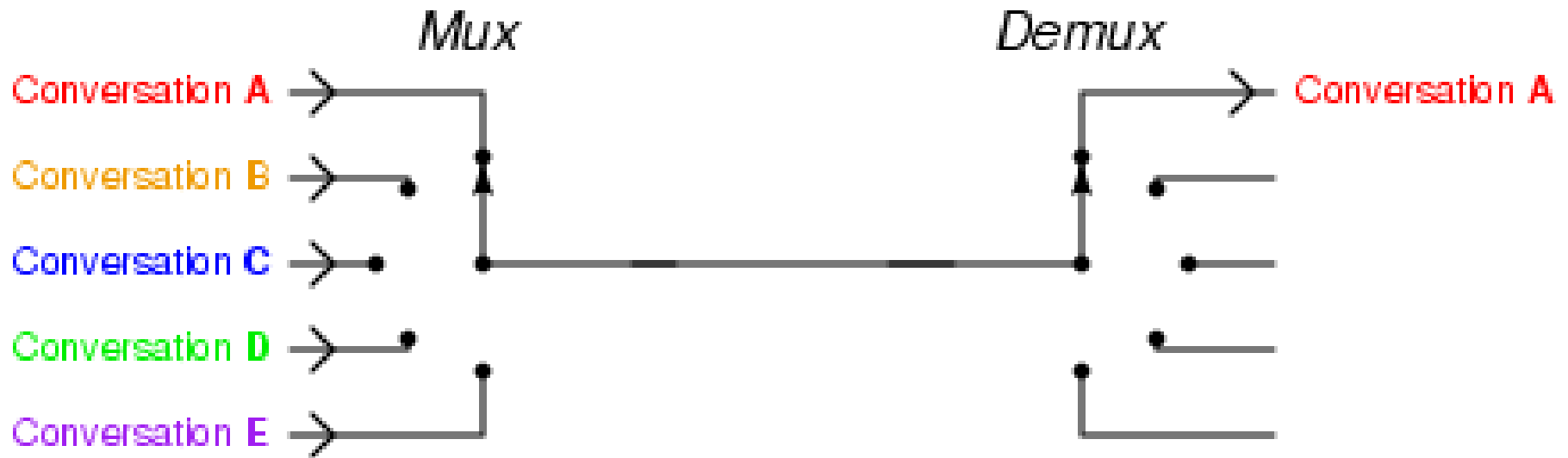
Es kann über eine Energie- und Datenleitung, den sogenannten Datenbus, Informationen digital über eine Leitung übertragen.

Multiplex-Systeme sind sogenannte Niedergeschwindigkeits-Busse

# 6. Multiplex System

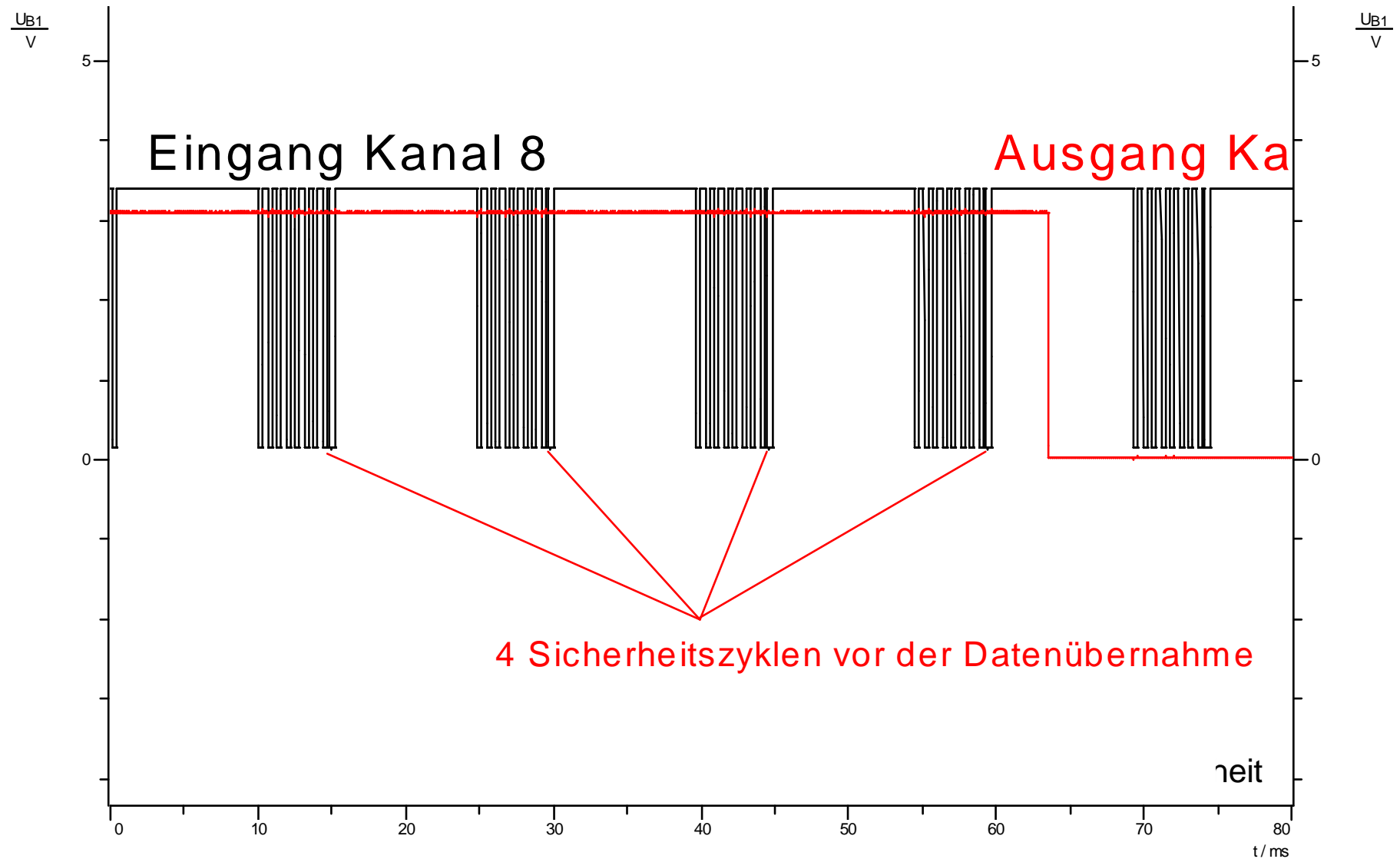


# 6. Multiplex System



# Möglichkeiten der Datenübertragung

Aufbau einer Informationseinheit und die Einbettung der acht Datenbits



## 6. Multiplex System

**Vorteile :** - Kostengünstiges System

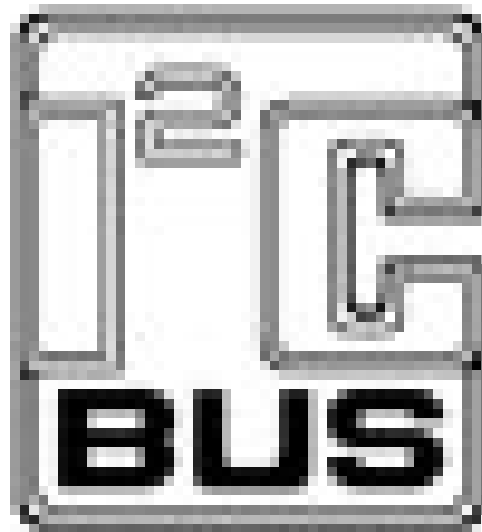
**Nachteile:**

- Jede Botschaft wird 4 mal gesendet zur Datensicherheit
- Folge => niedrige Übertragungsgeschwindigkeit
- Viel Verkabelungsaufwand unter den Systemen
- Aufgrund von nur 8 Bits kein sehr hoher Datenaustausch

**Das Multiplex System wird heute nur noch oft im Bereich der Übersetzung zwischen Fahrzeug und Tester eingesetzt!**

## 7. I<sup>2</sup>C - Bus

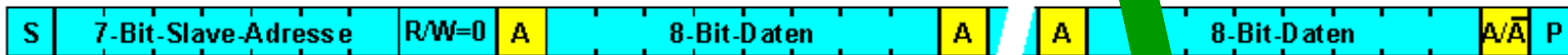
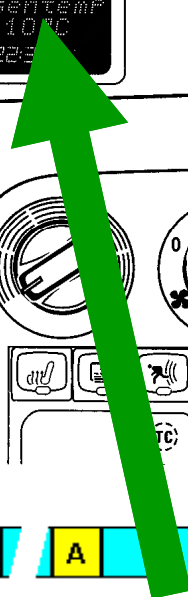
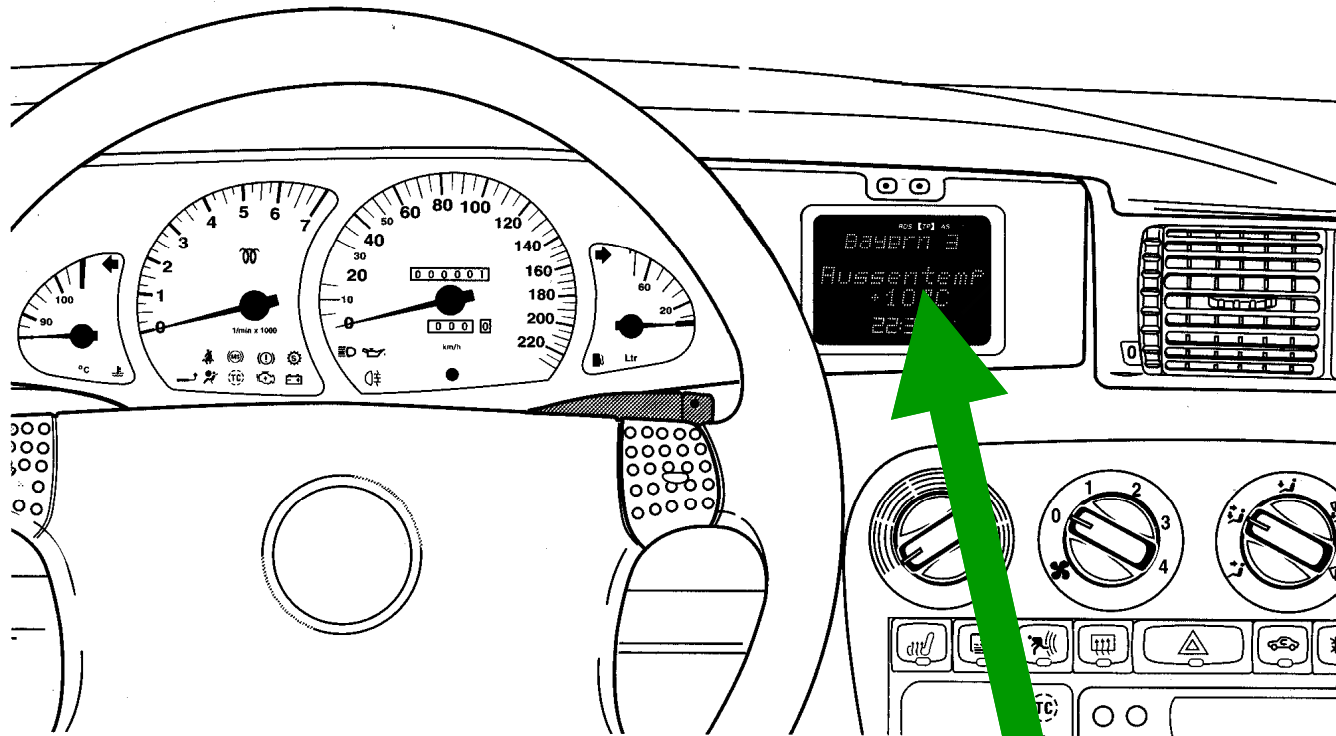
Der I<sup>2</sup>C wird heute nur noch in KFZ-Steuergeräten (intern) zu Steuerungs- und Regelsystemen mit langsamen Komponenten verwendet.



Verwendet wurde dieser Synchroner Serieller Daten-Bus z.B. in der Klimaanlagesteuerung von Renault oder beim Radio von Opel.



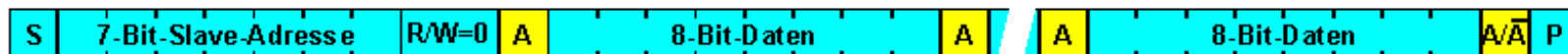
# Der I<sup>2</sup>C-Bus



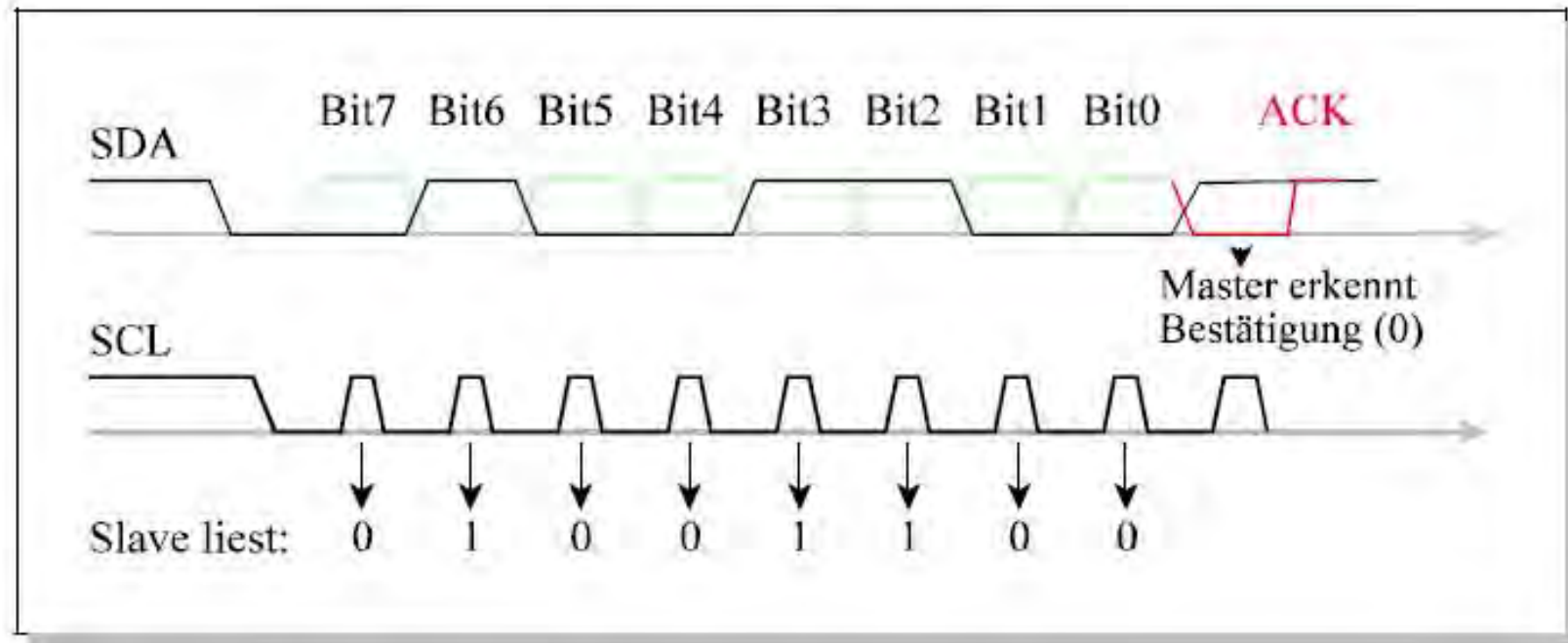
Skip

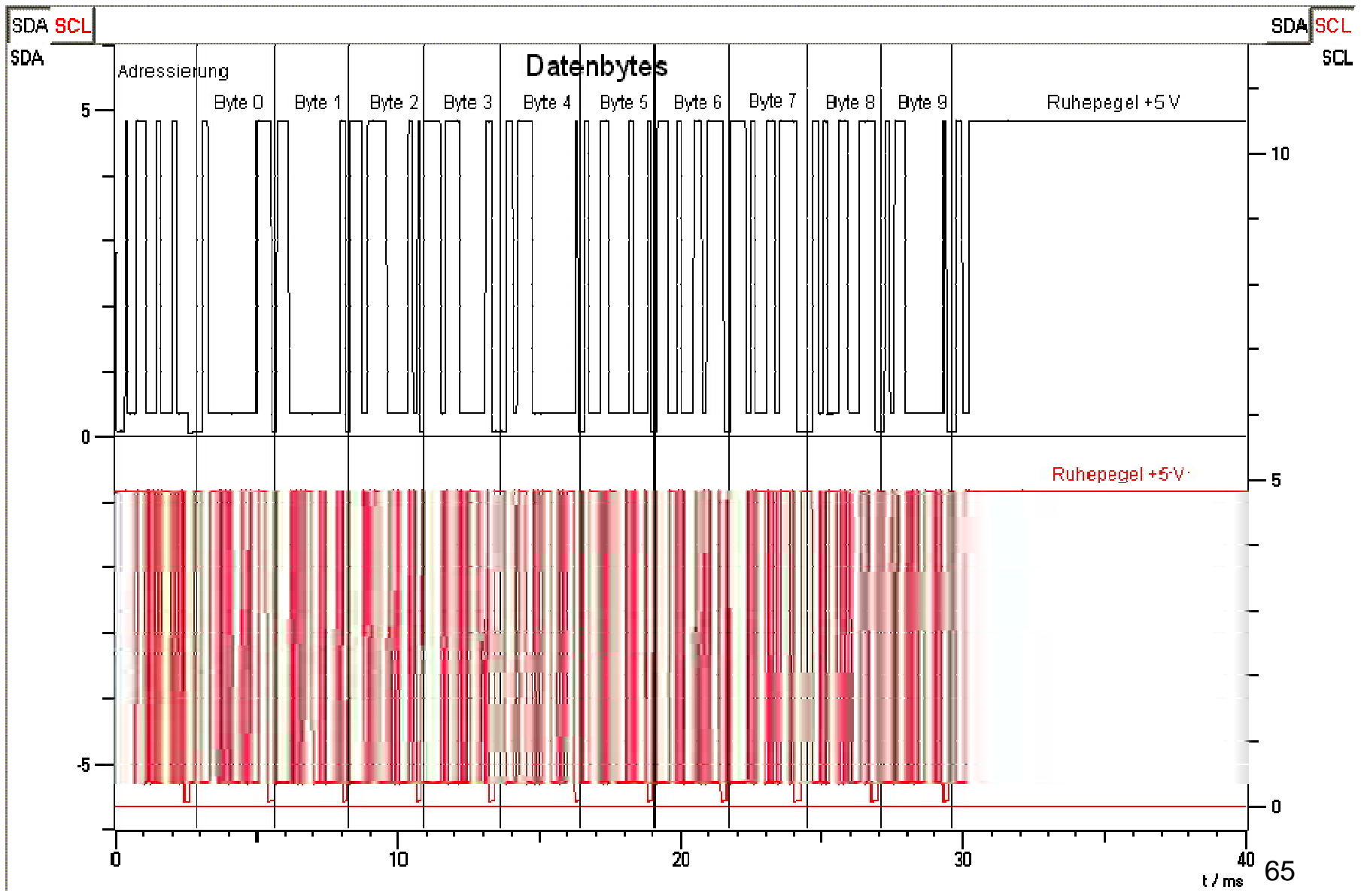
## Der I<sup>2</sup>C-Bus

- Datenübertragung vom Master (Radio) zum Slave (Display)
- Der Master stellt zunächst die STARTBEDINGUNGEN her, indem er den SDA-Pegel von HIGH auf LOW zieht, während der SCL-Pegel HIGH bleibt
- Danach wird der SCL-Pegel vom Master auf LOW gezogen und der Master taktet synchron mit der SCL-Leitung die 7-bit-Adresse des Slaves auf die SDA-Leitung
- Mit dem Takten des achten Bits (LSB), des **R/W-Bits = 0**, teilt er dem Slave mit, dass dieser die folgenden Bytes empfangen soll
- Zwischen achtem und neuntem SCL-Takt zieht der Slave die SDA-Leitung auf LOW-Pegel und bestätigt den Empfang des vom Master gesendeten Bytes (**ACKNOWLEDGE = 0**)
- Nun kann der Master ein weiteres Byte zum Slave durchtakten, welches wiederum im neunten Takt durch einen LOW-Pegel aus SDA vom Slave bestätigt wird usw.
- Die Datenübertragung wird unterbrochen, wenn der Slave die Quittierung verweigert (**ACKNOWLEDGE = 1**) und/oder der Master die STOP-BEDINGUNGEN herstellt



# I<sup>2</sup>C - Bus





# 8. Der Eindraht- Lin- BUS



# Local Interconnect Network

## Einführung

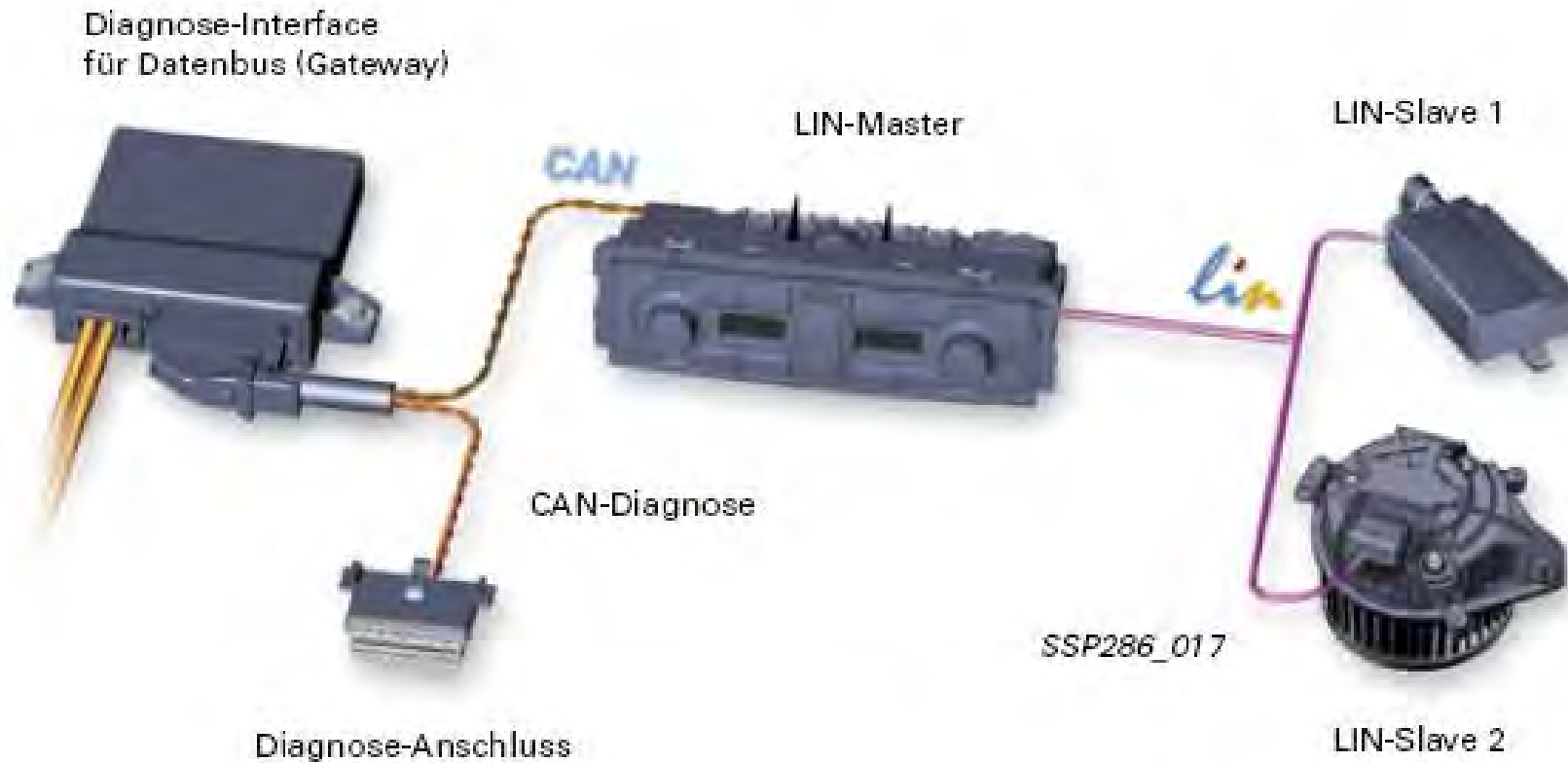
LIN steht für Local Interconnect Network.

Local Interconnect bedeutet, dass sich alle Steuergeräte innerhalb eines begrenzten Bauraums (z. B. Dach) befinden. Dieses wird auch als „lokales Subsystem“ bezeichnet.

Der Datenaustausch zwischen den einzelnen LIN- Bussystemen in einem Fahrzeug erfolgt über jeweils ein Steuergerät durch den CAN Datenbus.



# LIN Master



# *Local Interconnect Network*

## **Anwendungsgebiete:**

- Sitzverstellung
- Klima
- Dachelektronik
- Spiegelverstellung
- Wischerelektronik usw.

**SLAVE:** ist immer passiv! Würde nie alleine was machen ohne Anweisung (in der Art wie ein Lehrling!

**MASTER:** Stellt die fragen und gibt Anweisungen => SLAVE beantwortet und führt aus!

**Wenn ein MASTER oder SLAVE sendet hören alle anderen SLAVE zu!**



# *Local Interconnect Network*

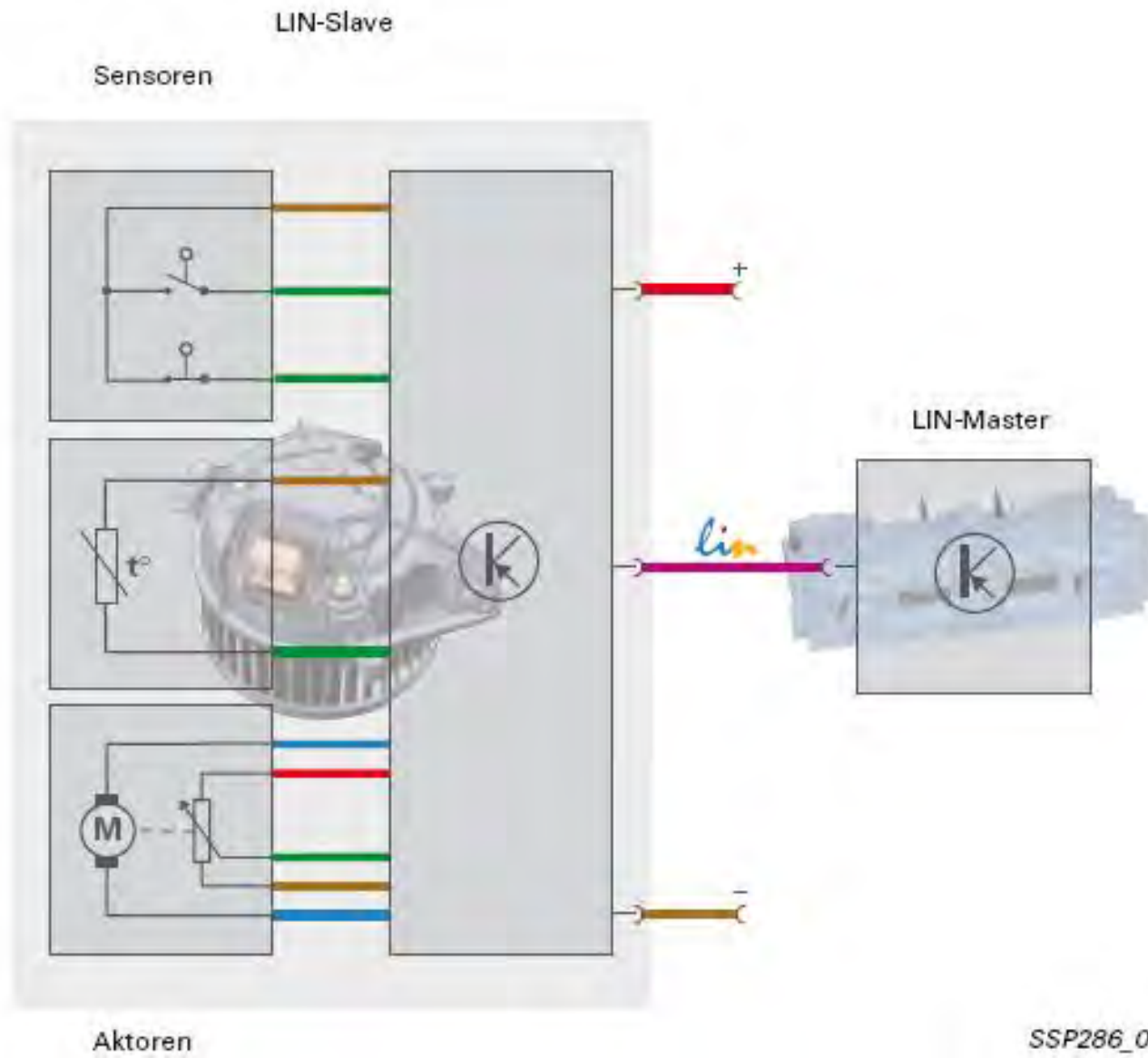
## **Eigenschaften LIN MASTER:**

- Hauptsteuergeräte des Systems ( Bordnetz STG , Klima STG usw.)
- Schnittstelle zwischen CAN und LIN
- Fehler feststellen und analysieren
- CAN Infos aufnehmen und abgeben
- Sendet Daten (HEADER) und empfängt Daten (RESPONSE)

## **Eigenschaften des SLAVE:**

- Kann ein Sensor oder Aktor sein
- Empfängt Daten und verarbeitet diese
- Antwortet auf fragen des Masters
- Muss auch bei Antworten anderer SLAVE Komponenten zuhören

## LIN-Slave-Steuergeräte



SSP286\_070

## Datenübertragung

Die Datenübertragungsrate beträgt 1 - 20 Kbit/s und ist in der Software der LIN-Steuergeräte festgelegt. Dies entspricht maximal einem Fünftel der Datenübertragungsrate des CAN-Komfort.



### Vor und Nachteile des Lin-Bus:

**Vorteile:**

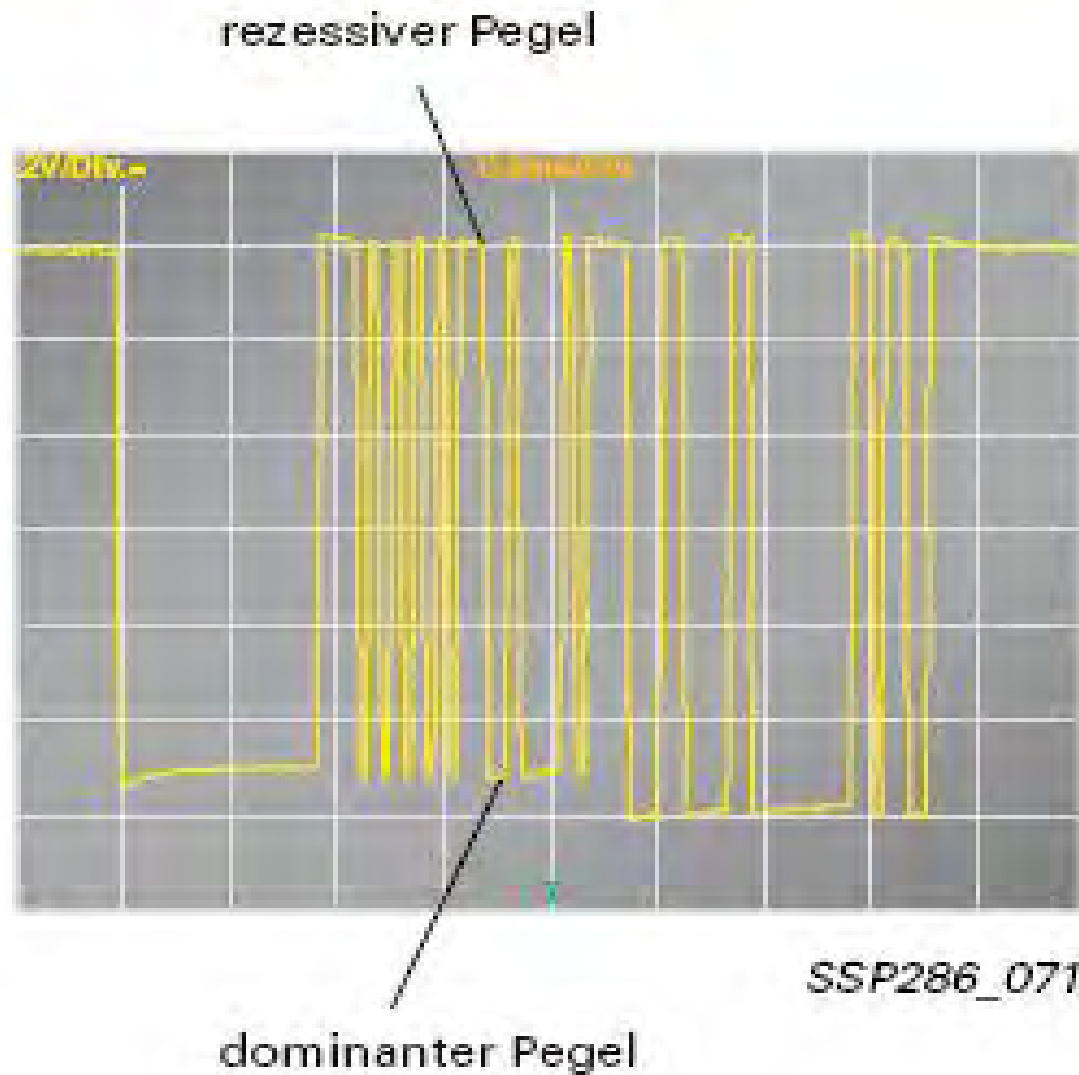
- Weniger Kabel
- Kostengünstige Herstellung

**Nachteile:**

- langsame Datenübertragung

**Eigenschaften:** Arbeitet nach dem Masterprinzip= 1 Master (CAN- Kommunikation) ; bis zu 16 Slave (Sensoren/ Aktoren), Eindraht-Bussystem = Energieversorgung

# Signalbild vom LIN-Bus



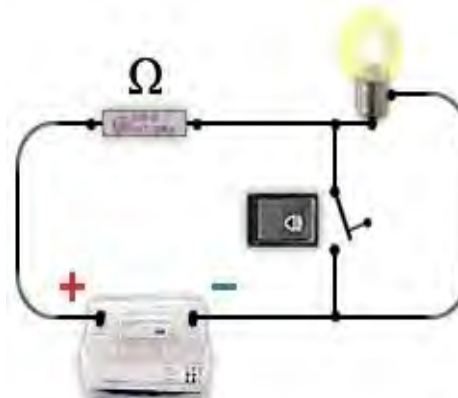
# Dominant und Rezessiv

## Negativlogik (LIN-BUS / CAN-Bus)



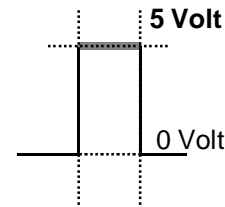
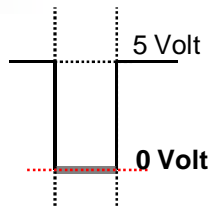
Dominantes Signal

- Schalter geschlossen
- Lampe leuchtet nicht



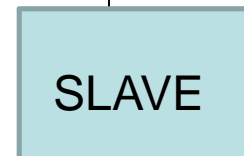
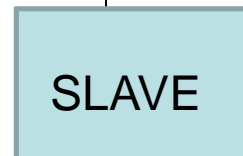
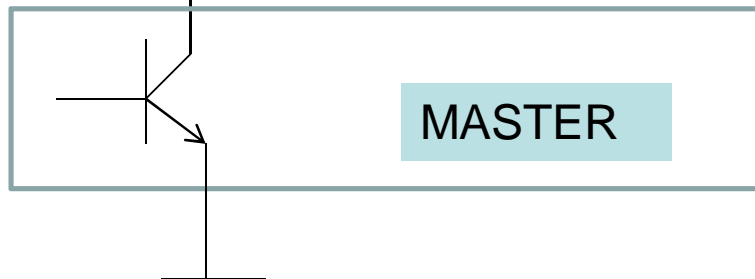
Rezessives Signal

- Schalter geöffnet
- Lampe leuchtet

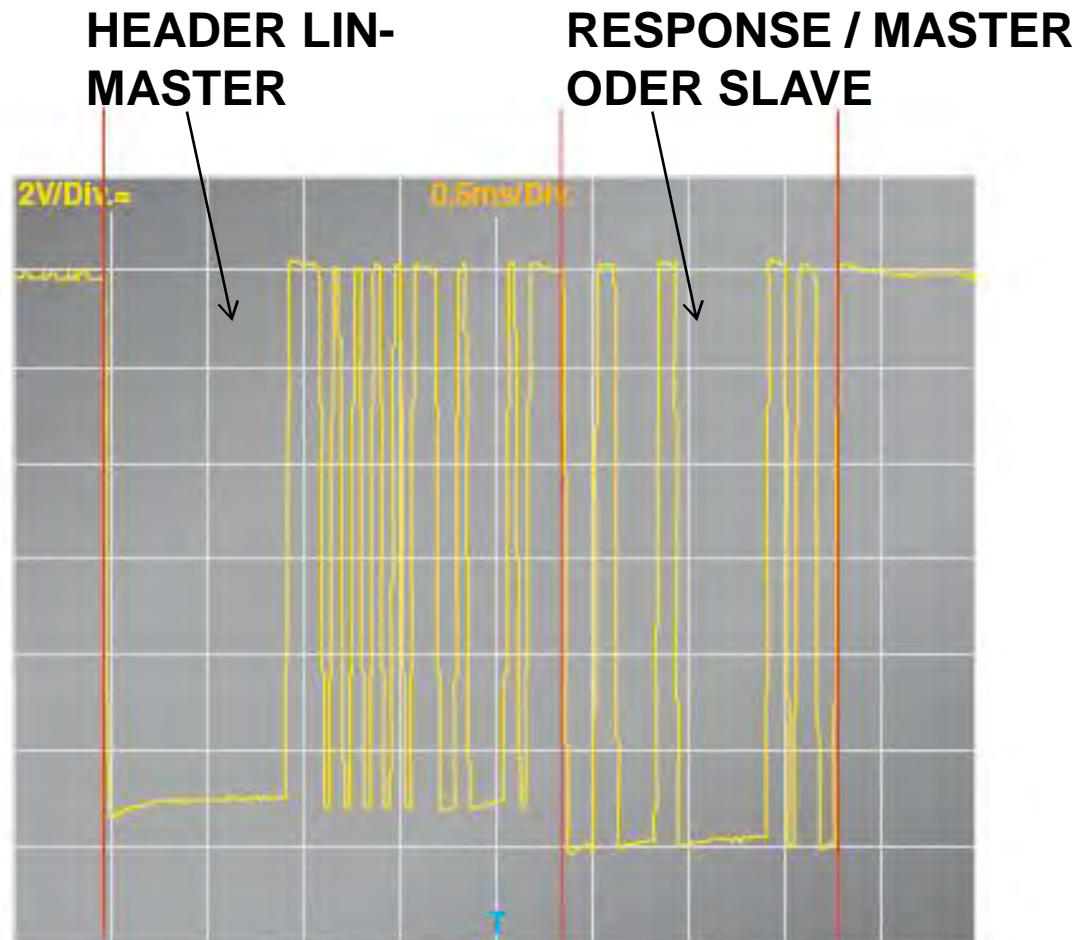


12Volt

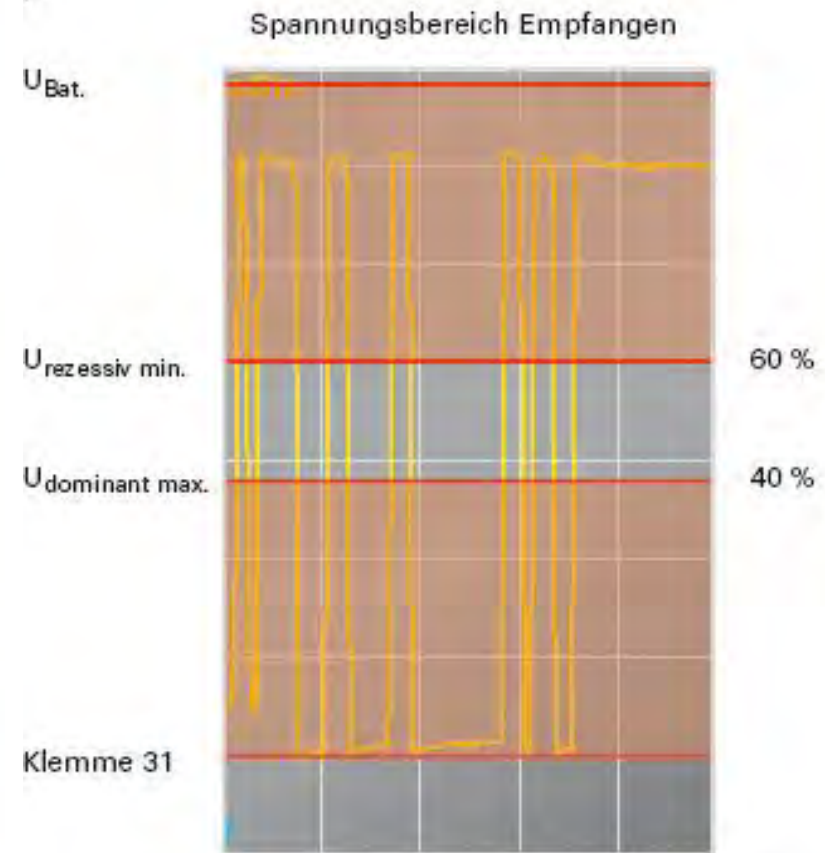
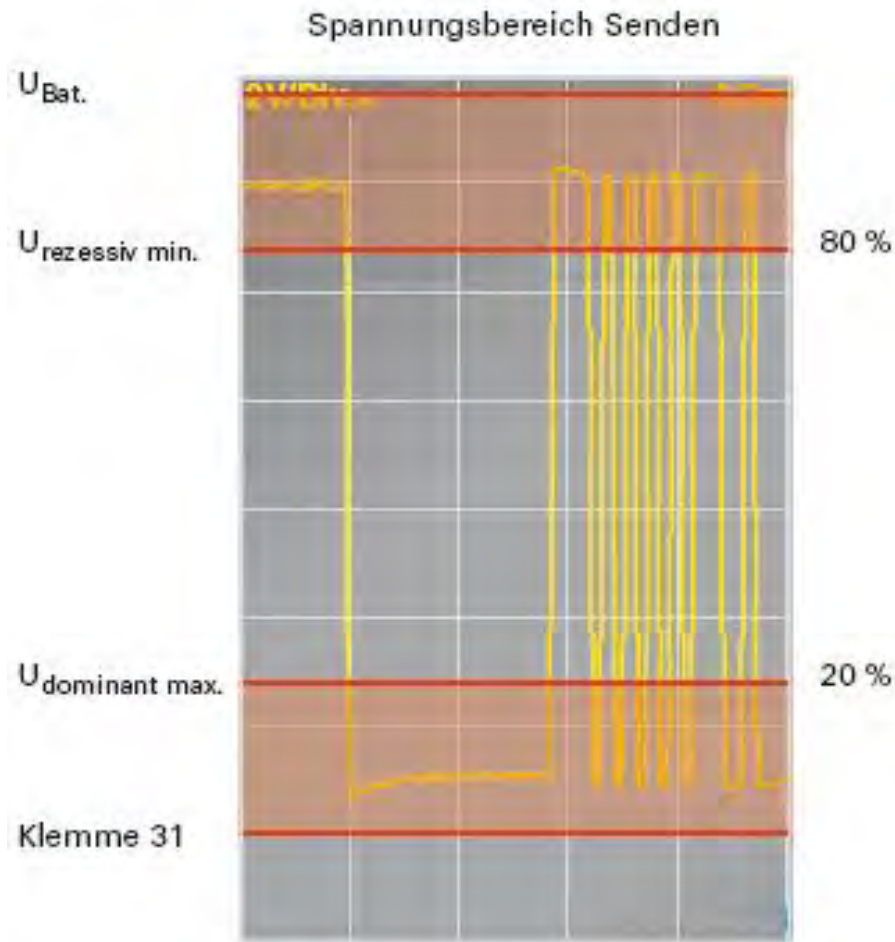
LIN-Bus (lila)



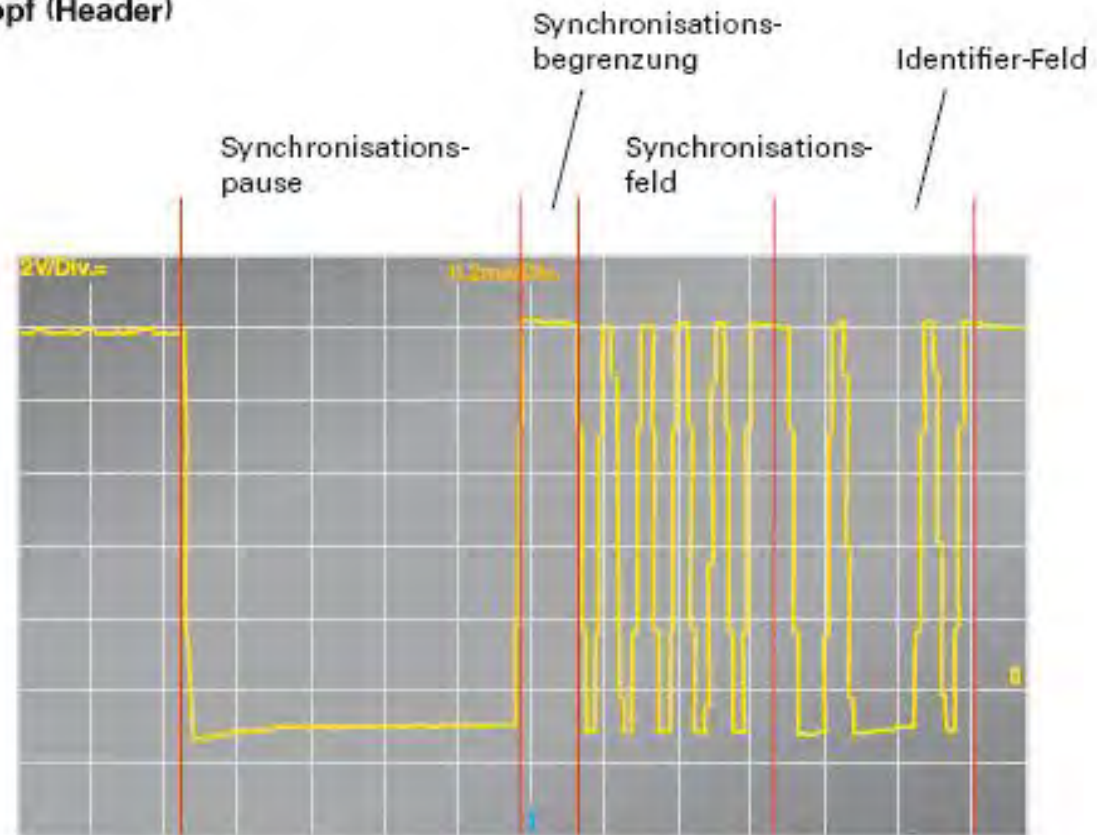
# Nachrichtenaufbau beim LIN-Bus



# Übertragungssicherheit beim LIN-Bus

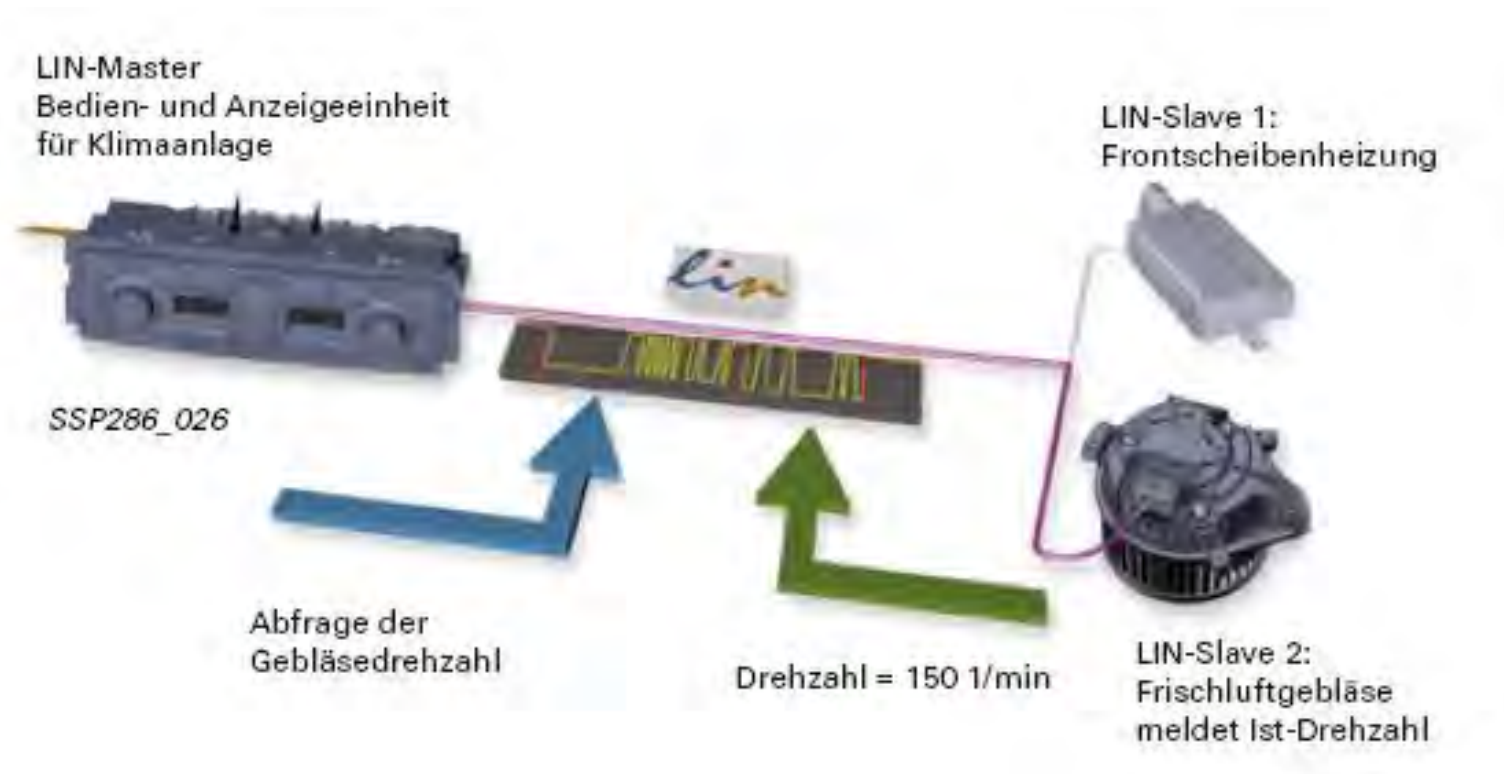


## Botschaftskopf (Header)



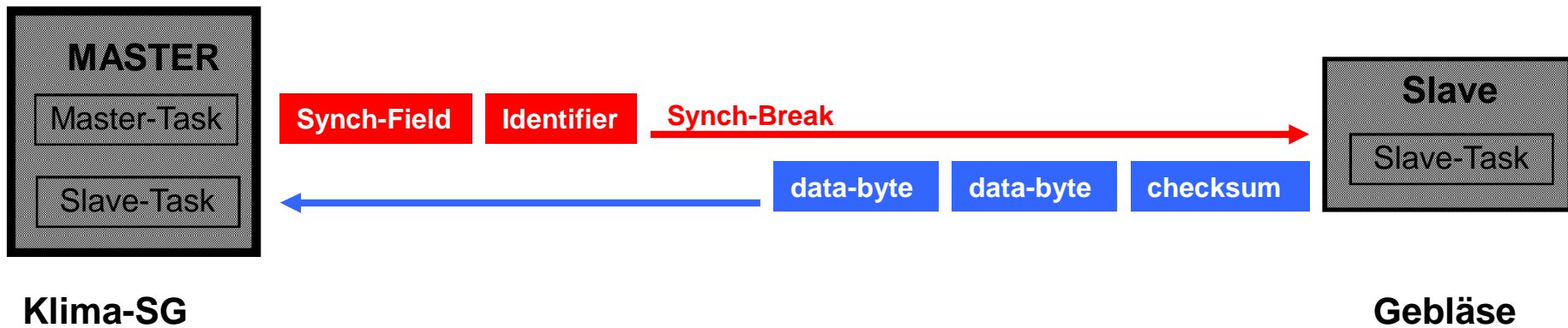


# Beispiele LIN-Bus/ Response



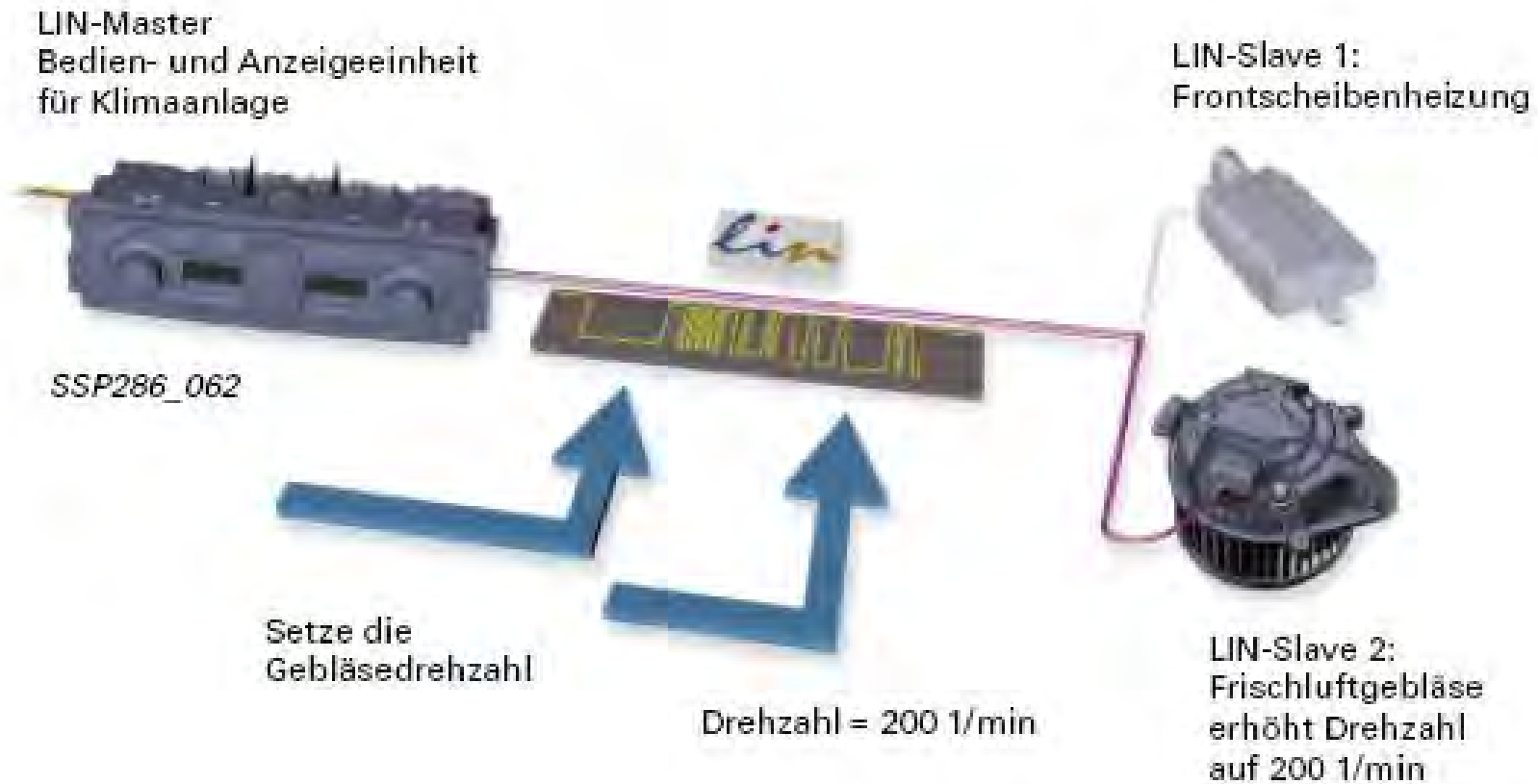
# LIN- Kommunikationsmöglichkeiten

Daten vom Slave zum Master



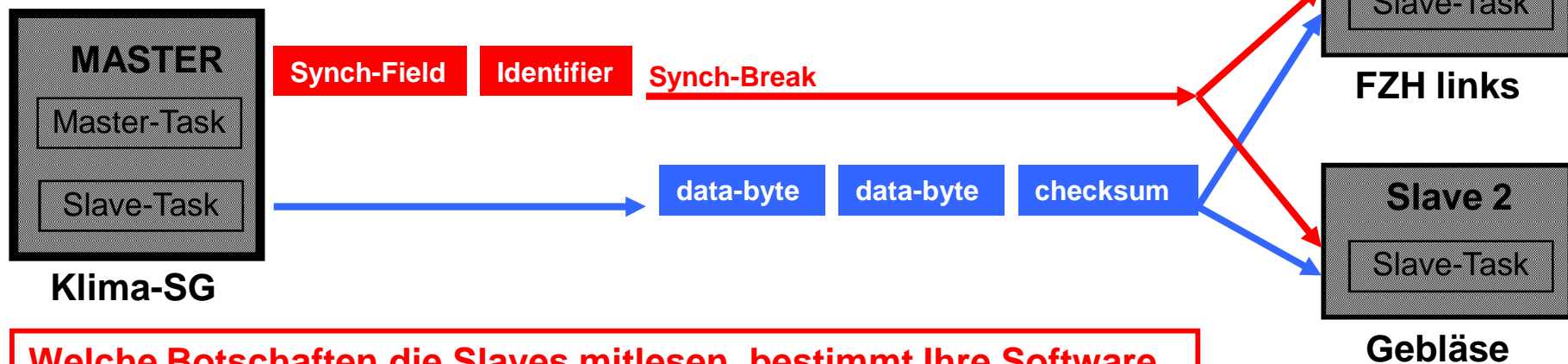
Wann Daten vom Slave zum Master gesendet werden, bestimmt stets der Master!

# Beispiele LIN-Bus/Response



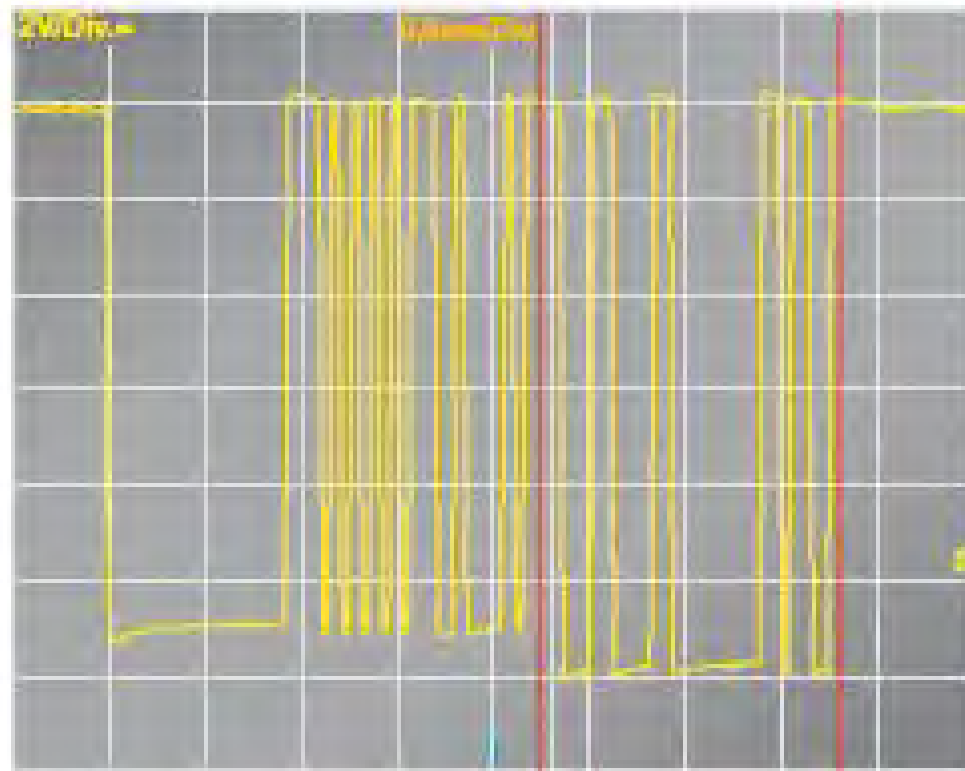
# LIN- Kommunikationsmöglichkeiten

Daten vom Master zu Slave(s)



Welche Botschaften die Slaves mitlesen, bestimmt Ihre Software.

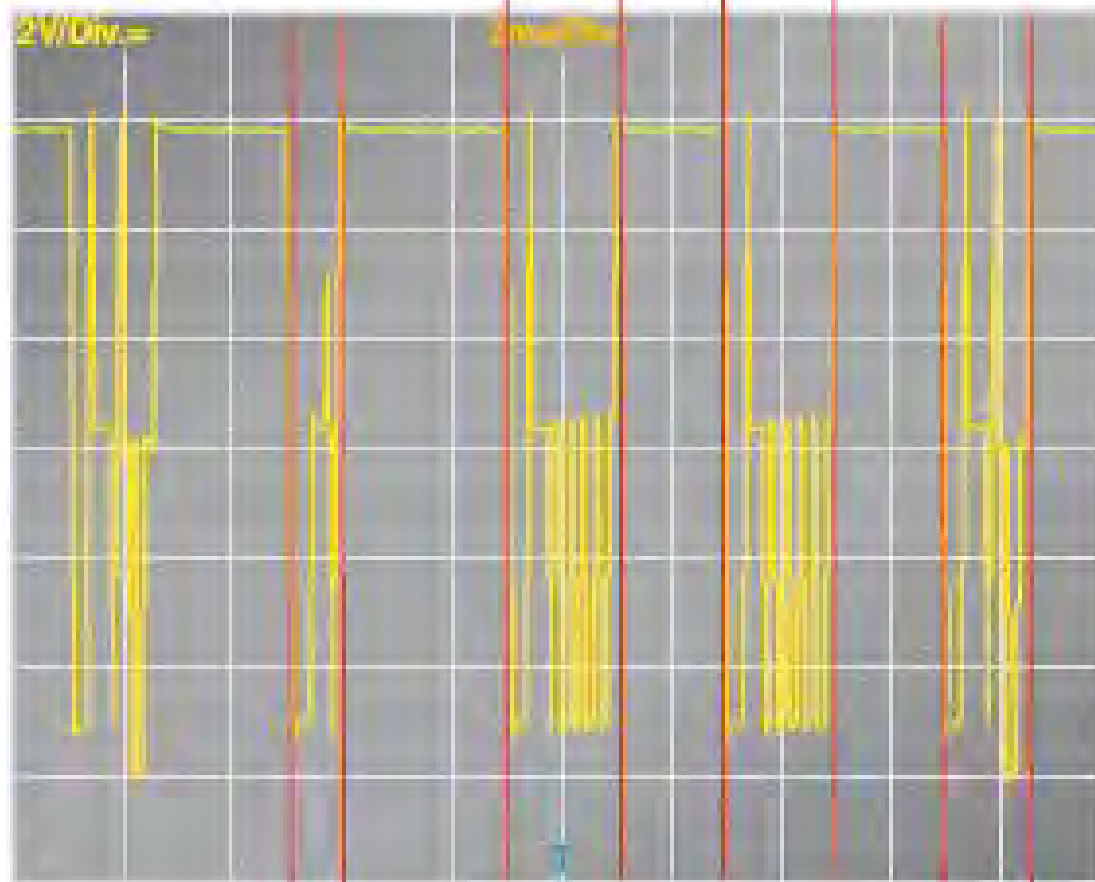
# Response



SSP286\_074

Response

Master-Botschaft



SSP286\_075

Header ohne  
Response

Slave-Botschaft  
(hier erkennbar durch  
unterschiedliche dominante Pegel)

# Zusammenfassung Datenfeld LIN-Bus

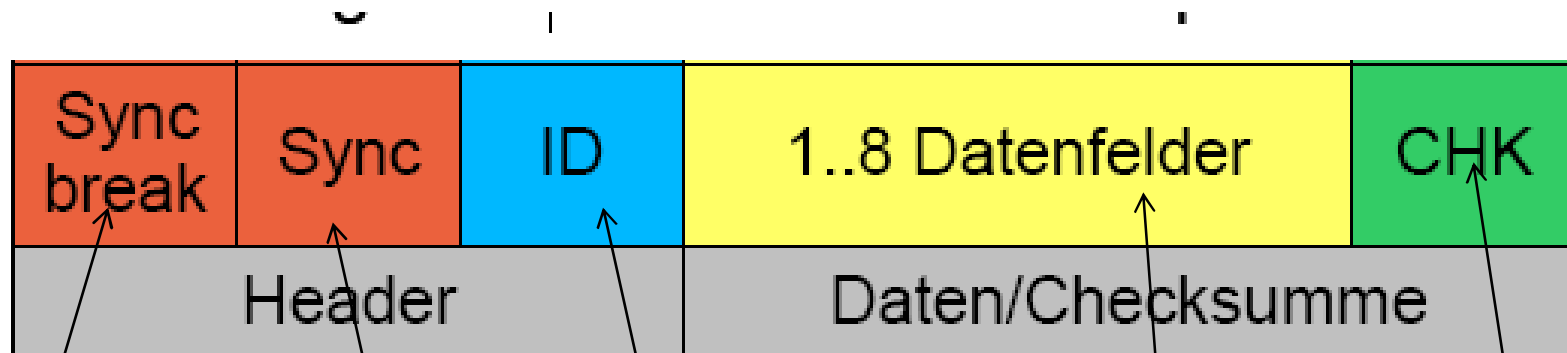


Abb. 2: Datenpaket

$\frac{13 + 1}{13 = \text{Dominat}$   
1 = Stopp-Bit

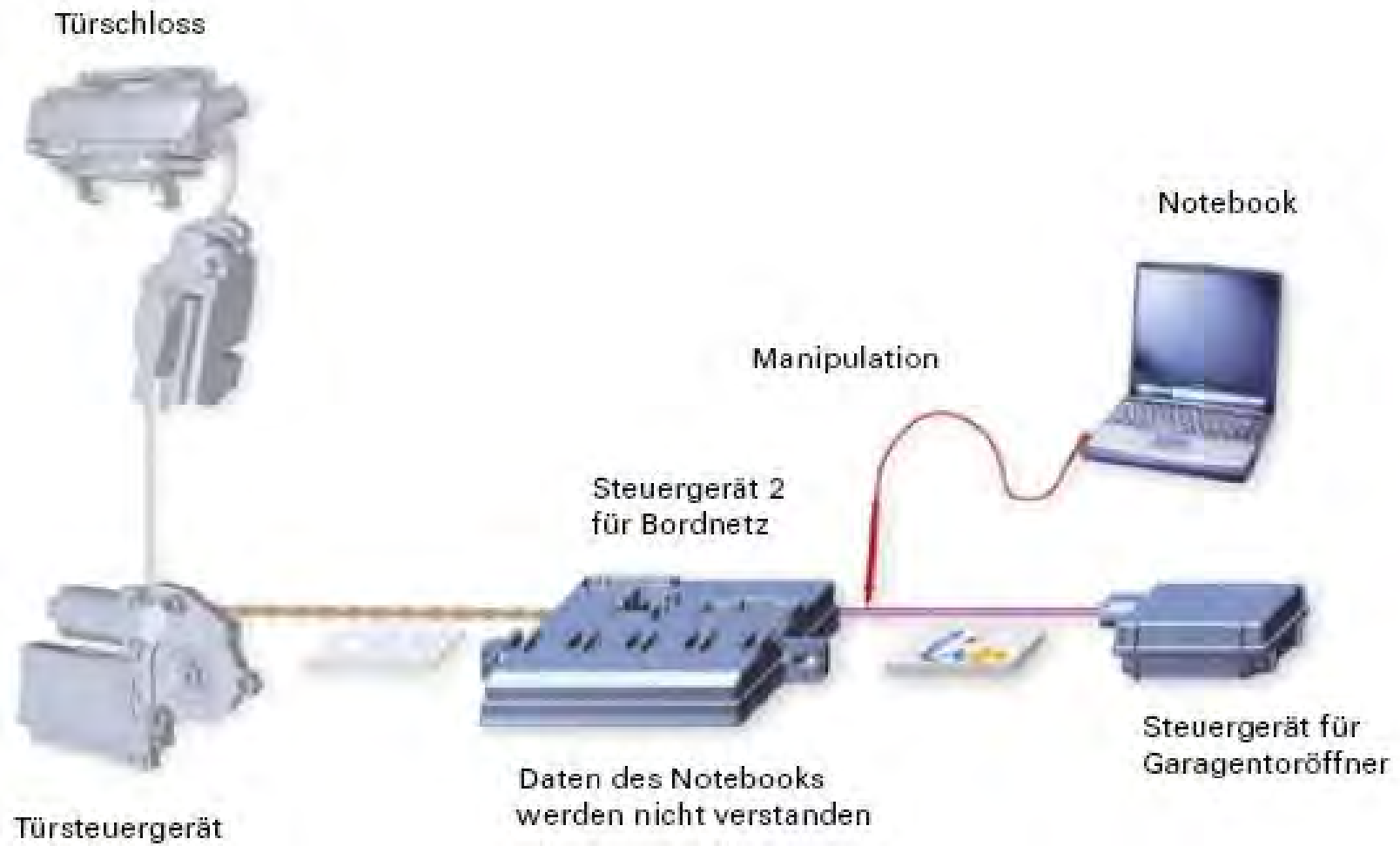
$\frac{1 + 8 + 1}{1 = \text{Dominat}}$   
8 = Daten  
1 = Rezessiv

4 + 4 + 2  
4 = Start-Bits Dominat  
4 = Infos  
2 = DLC

10-80 Bits

1 + 8 + 1  
Checksumm  
Überprüfung

## Diebstahlschutz





# Diagnose des LIN-Bus

Fehlerort	Fehlertext	Ursache für Fehlereintrag
LIN-Slave-Steuergerät, z. B. Gebläseregler	kein Signal/ keine Kommunikation	<p>Ausfall der Datenübertragung vom LIN-Slave-Steuergerät über einen in der LIN-Master-Software festgelegten Zeitraum.</p> <ul style="list-style-type: none"> <li>- Leitungsunterbrechung oder Kurzschluss</li> <li>- Defekte Spannungsversorgung des LIN-Slave-Steuergerätes</li> <li>- Falsche Teilevariante LIN-Slave oder LIN-Master</li> <li>- Defekt des LIN-Slave-Steuergerätes</li> </ul>
LIN-Slave-Steuergerät, z. B. Gebläseregler	Unplausibles Signal	<p>Fehler in der Checksumme. Unvollständige Übertragung der Botschaften.</p> <ul style="list-style-type: none"> <li>- Elektromagnetische Störeinflüsse auf der LIN-Leitung</li> <li>- Kapazitäts- und Widerstandsänderung an der LIN-Leitung (z. B. Feuchtigkeit/Verschmutzung am Steckgehäuse)</li> <li>- Softwareproblem (falsche Teilevarianten)</li> </ul>

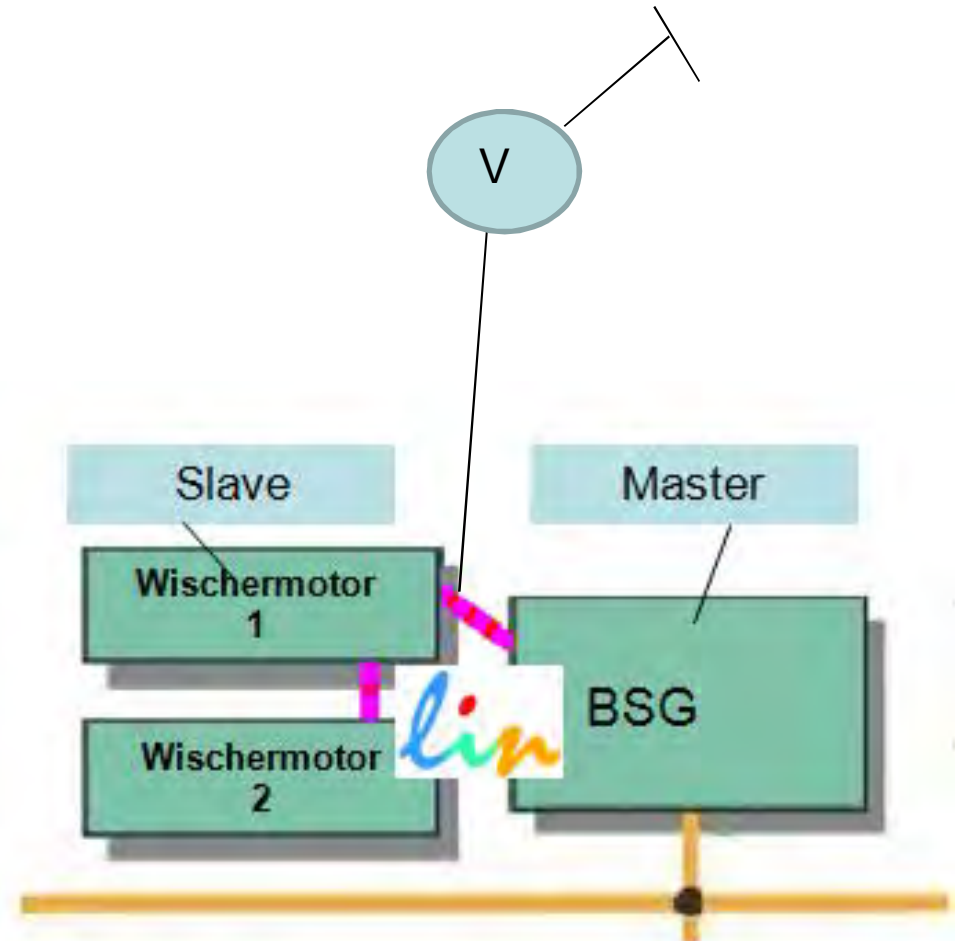
# Diagnose des LIN-Bus

Kann über den Tester keine eindeutige Diagnose gestellt, muss die LIN-Bus Leitung frei gelegt werden und gemessen werden!

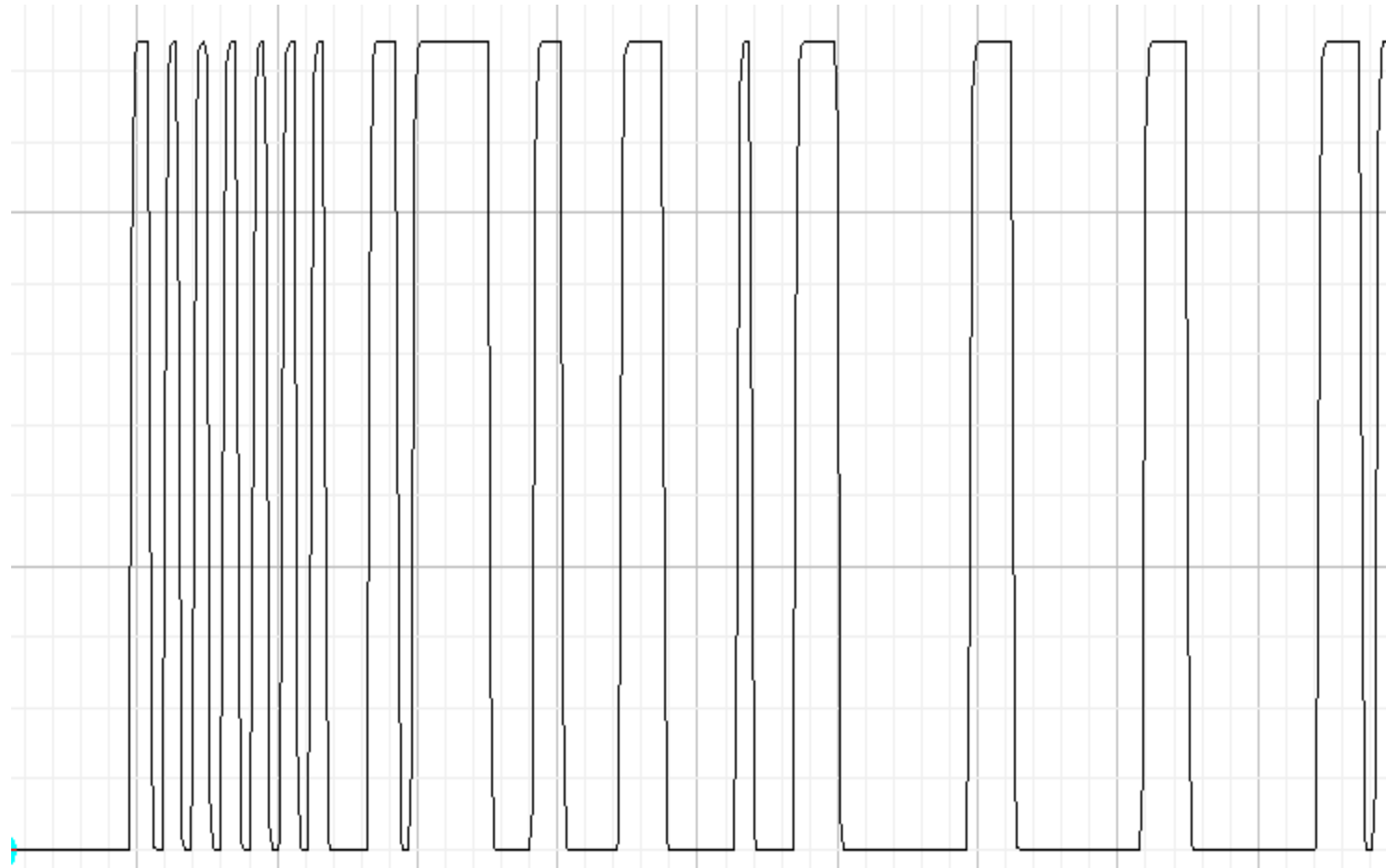
Hier ist am schnellsten und effizientesten eine Spannungsprüfung durch zu führen.

U / LIN-Bus = 12 Volt

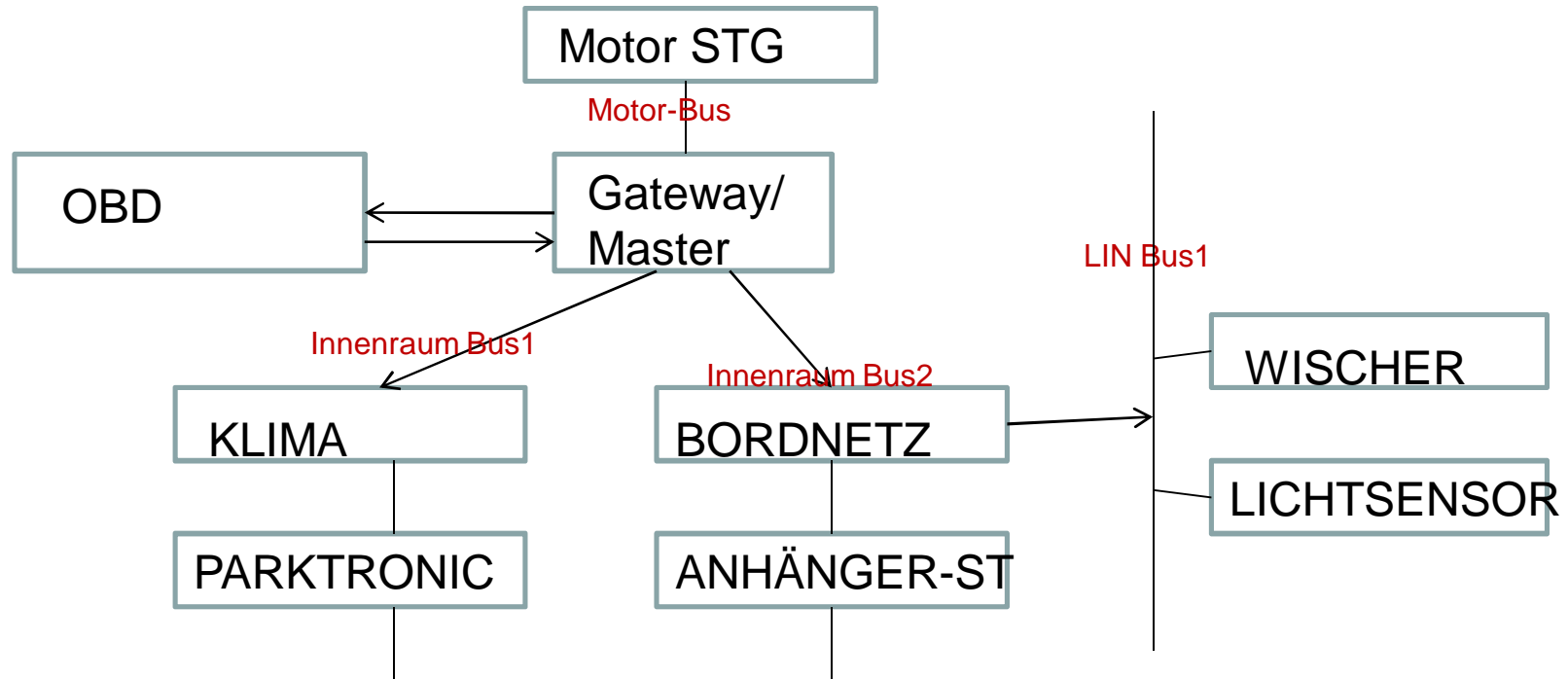
Weiterhin kann eine Überprüfung des LIN-Bus mit dem Oszilloskop statt finden!



**LIN-Bus Wand (HWK-Potsdam..... Spannungspegel 0-12 Volt**



# LIN-Bus Topologie



# Wischersystem

Schalter Frontwischer  
Schalter Heckwischer  
Empf. / Intervall

Schaltmodul  
Lenksäule  
SMLS

Dual-  
Pumpe

Wischer  
vorne

Regen-  
sensor



Bordnetz  
Stg.

Gateway  
Stg.



Heckwischer

Hochdruckpumpe

Komfort  
Stg.

ABS  
Stg.

Motorhaube /Heckklappe



Motorhaube- und Heckklappe

V



## Vernetzte Funktionen Lampenüberwachung

### Kaltüberwachung

Bestromung : Nach Kl. 15 ein, 4 x alle 500ms.

Nach max. 2 s liegt ein Diagnosewert vor.

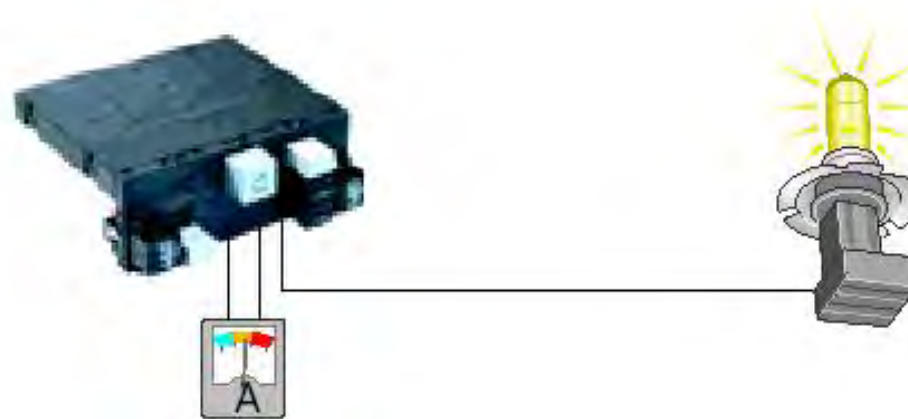
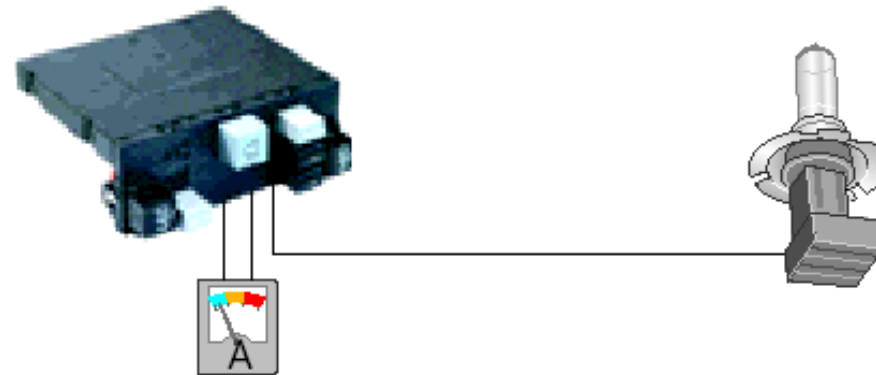
Fehleranzeige über CAN bei Kl. 15 ein. (CAN-Signal)

### Warmüberwachung

Bestromung : Nach Kl. 15 ein, permanent jede Glühlampe.

Nach max. 2 s liegt ein Diagnosewert vor.

Alle Glühlampen



**Glühlampen werden ab 13,2V/  $U_{eff}$  PWM- stabilisiert**

## 9. Der CAN-Bus



# Der CAN-Bus

## Was bedeutet CAN?

**C= Controller**

**A= Area**

**N= Network!**

## Eigenschaften:

- Der Can-Bus ist Asynchrones Serielles Datenübertragungssystem
- Es können in jedem Bus-Netzwerk bis zu Max. 32 STG vorhanden sein
- Die Max. Kabellänge darf 40 Meter nicht übersteigen
- Man unterscheidet: Motor-Bus; Komfort-Bus ; Infotainment-Bus
- Der CAN-Bus arbeitet nach dem sogenannten Broad-Cast Prinzip
- Man unterscheidet (außer Besonderheiten durch Hersteller) 2 verschiedene Bus-Klassen: **Class B = ca.125 K/Bits & Class C = 500-1000K/Bits**
- Guter EMV- Schutz und Übertragungssicherheit



## Überblick Einsatzbereiche am Beispiel VW Phaeton

Dachelektronik  
TSG HR  
TSG BF  
Bordnetz-SG  
Wischermotor  
Klimaanlage  
Zugang und Startberechtigung  
SG und Bedieneinheit Information V  
Schalttafeleinsatz (Gateway)  
Lenksäulenelektronik  
TSG FS  
Bedienung und Anzeige Klima H  
TSG HL  
Reifendrucküberwachung  
Anhängerererkennung  
Heckklappe  
Einparkhilfe  
Zentralsteuergerät Komfort



Datenbus Komfort

Skip

## **Allgemeine Infos zum CAN-Bus**

Wahrscheinlichkeit eines nicht erkannten Fehlers:

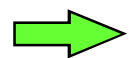
**2000 Betriebsstunden pro Jahr (täglich 5,5 Stunden)**  
**Übertragungsrate 500 kbyte/s**  
**Buslast 25 %**

**Ein unerkannter Fehler in 1000 Jahren!**

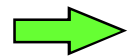
**Der CAN-Bus weist eine hohe Übertragungssicherheit auf**

# Allgemeine Infos zum CAN-Bus

PKW  
Typ A



Gesamtbetriebsdauer eines PKWs ca. 4000 Stunden



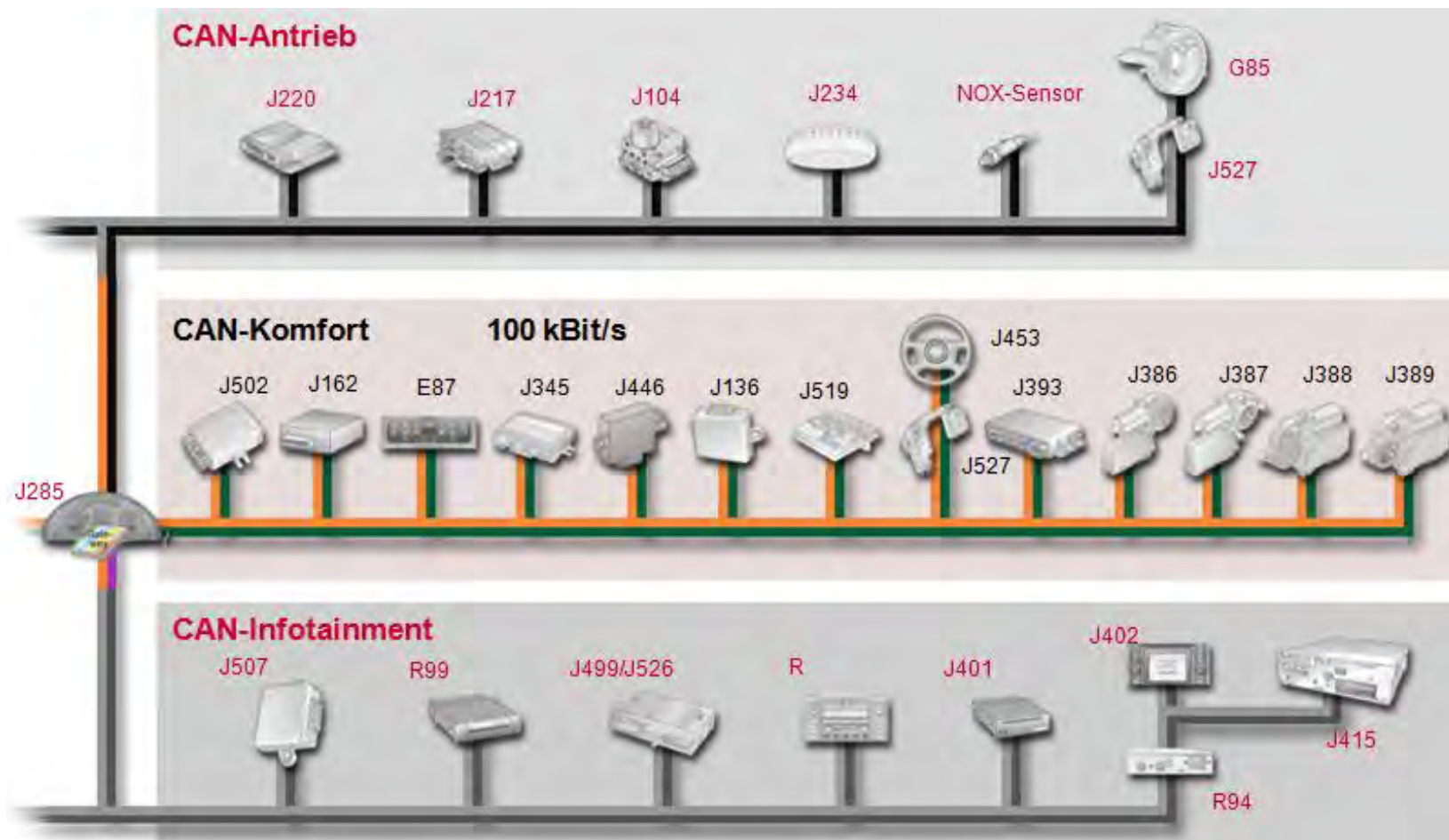
Annahme: Bitfehlerwahrscheinlichkeit  $10^{-6}$

"Plop!"



Bei **100.000.** PKW tritt während der gesamten Betriebsdauer nur bei einem einzigen ein unerkannter Übertragungsfehler auf.

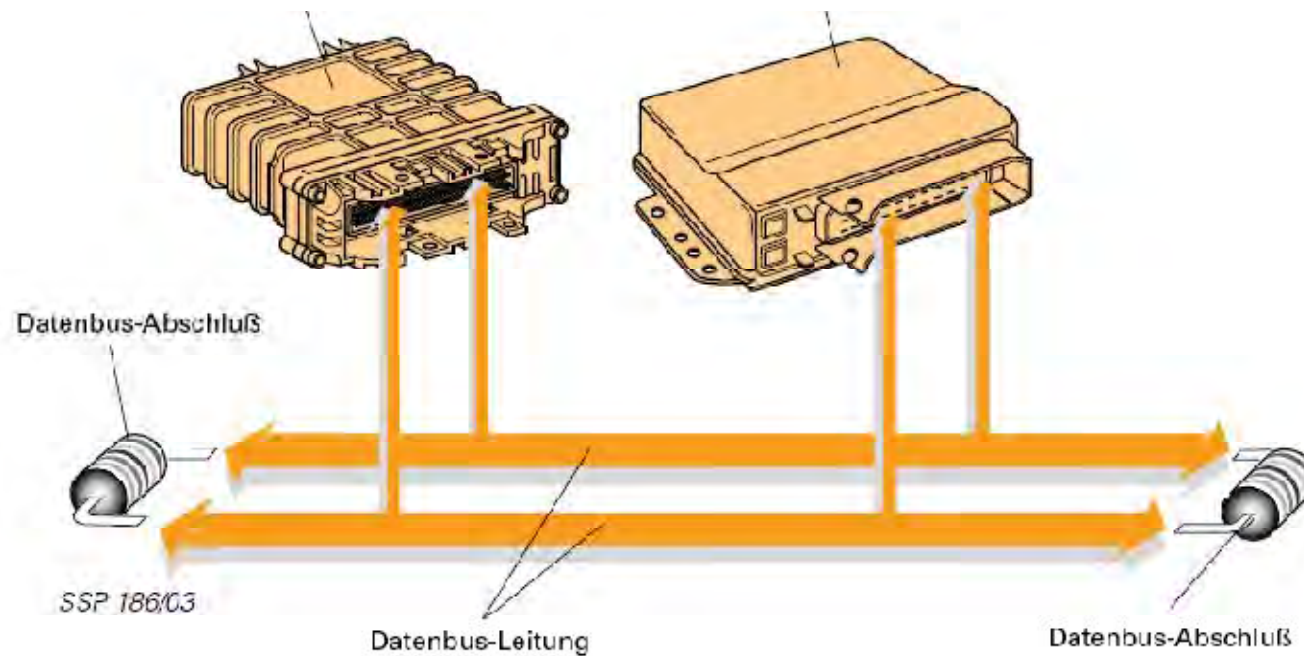
# CAN-Bus die Komponenten (Übersicht)



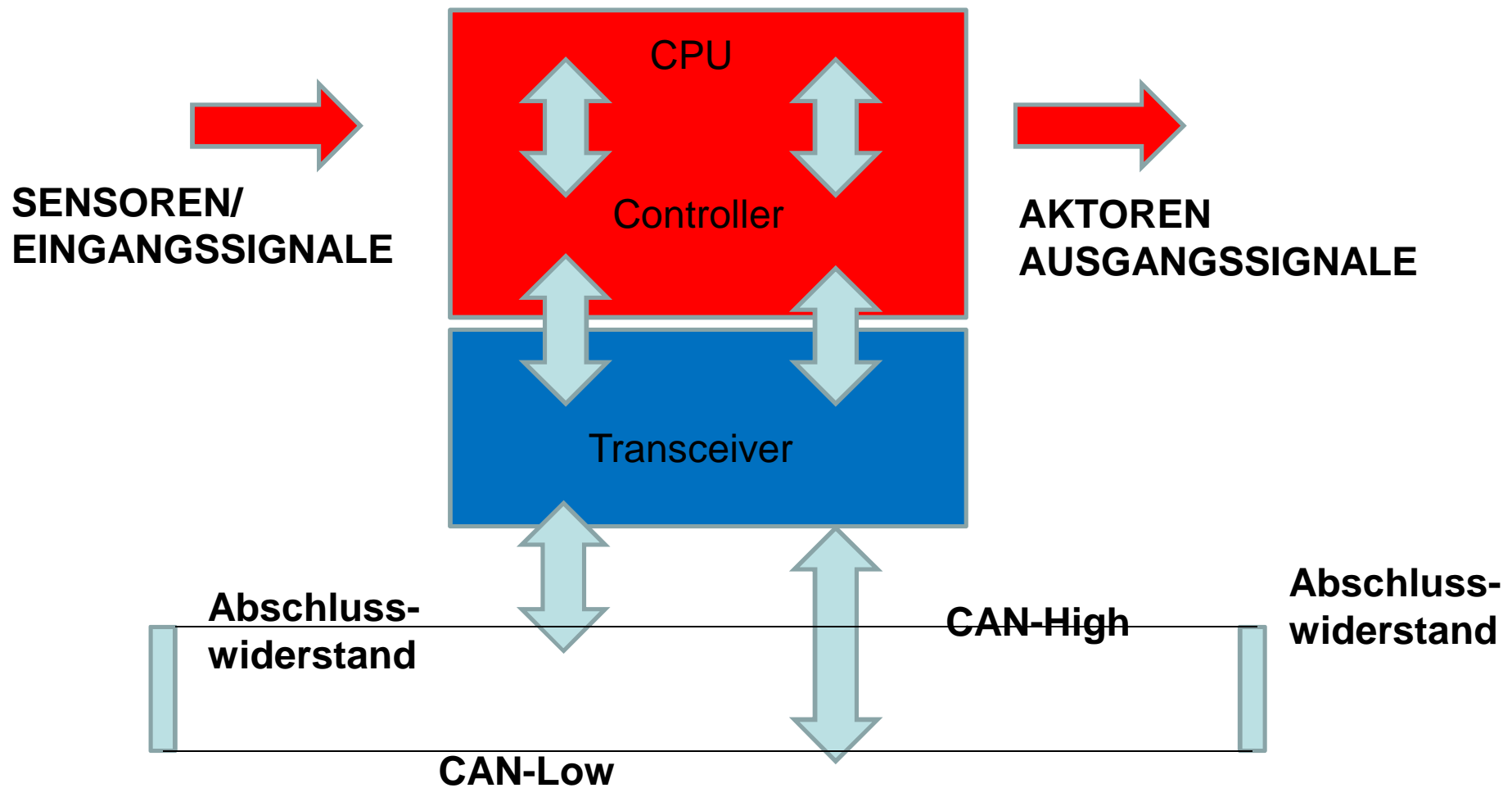
# CAN-Bus die Komponenten

Der CAN-Bus benötigt zu Kommunikation min. 2 Steuergeräte!

**Niemand redet gerne alleine mit sich selber!**

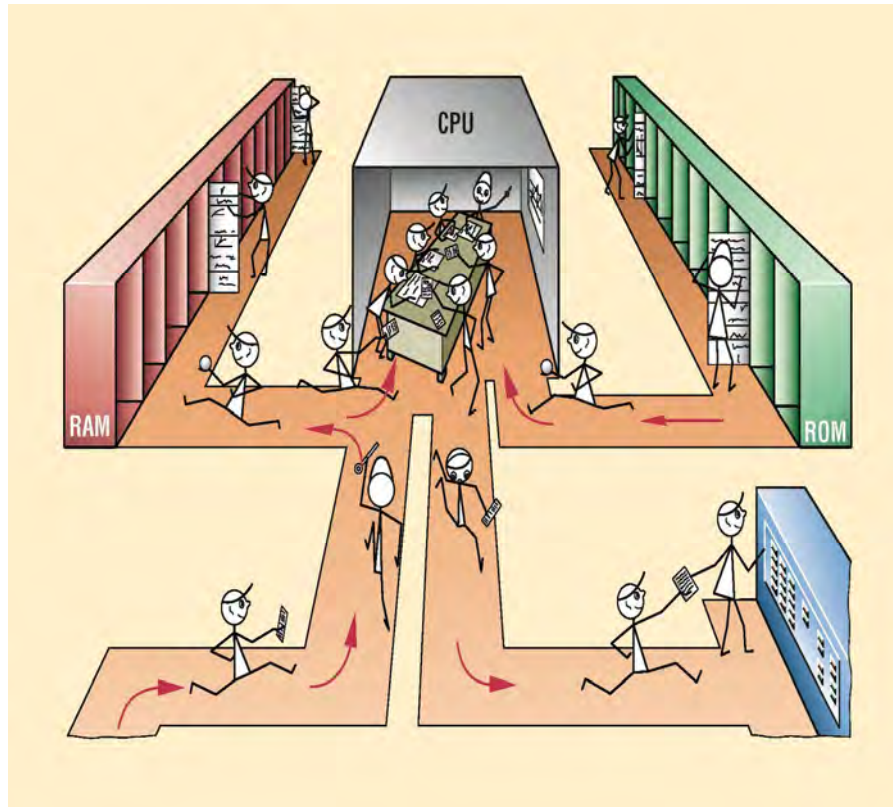


# *CAN-Bus die Komponenten Steuergerät*





# CAN-Bus die Komponenten Steuergerät

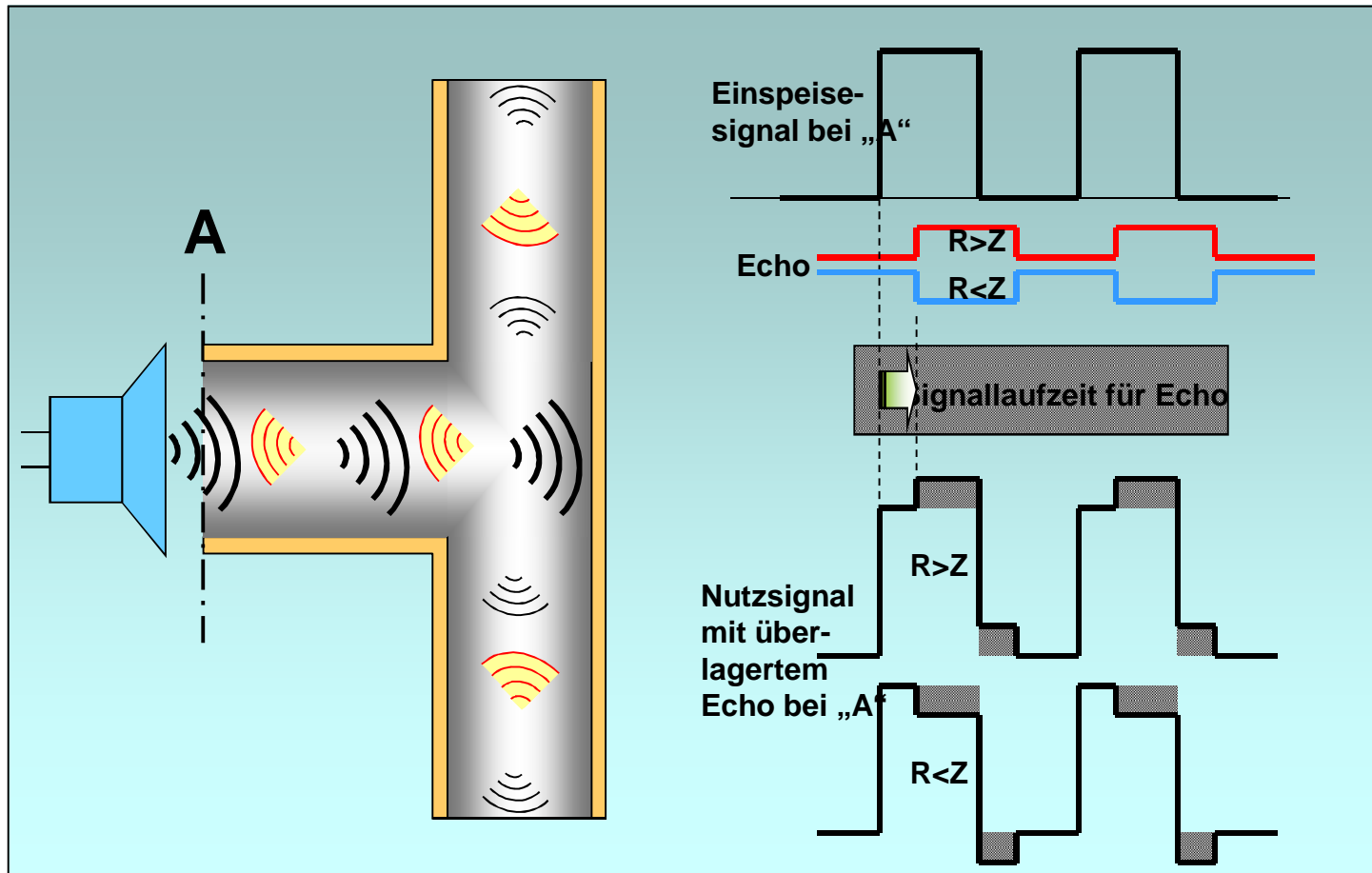


**CPU:** Recheneinheit mit Software, handelt nach festgelegter Matrix des Programmierers

**Controller:** Wie eine Sekretärin zu sehen! Der Controller wertet die Empfangenen Signale vom Receiver aus und bereitet Sie auf für den Transmitter. So gesehen wandelt die Spannungen in „Nullen“ & „Einsen“ (Binärcode) um und umgekehrt!

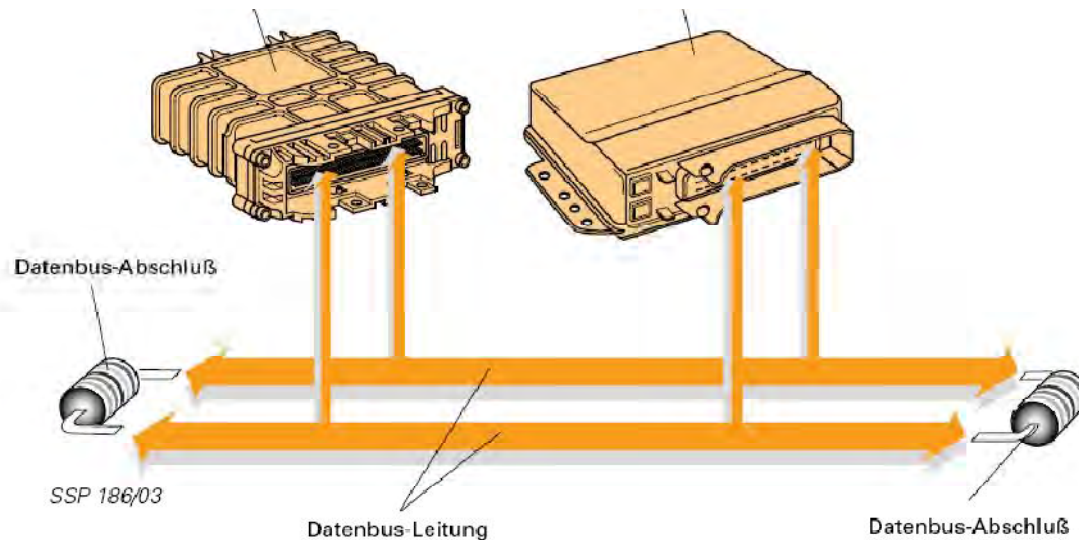
**Transceiver:** ist eine Empfangs- und Übertragungseinrichtung im STG!

# Die Abschlusswiderstände





# Die Abschlusswiderstände



**Die Abschlusswiderstände dienen dazu die gesendeten Signale der STG´s am Ende der Leitung zu vernichten!**

**Sonst:** würden die Signale sich überlagern und somit Störungen auf dem Bus erzeugen!

**Die Abschlusswiderstände sitzen in einem vom Hersteller ausgewähltem STG. Dies ist immer Herstellerabhängig!**

# Die Datenleitungen des CAN-Bus

Grundlagen

## CAN Bus Datenleitung

### **CAN Antrieb**

High or/sw

Low or/br



### **CAN Komfort**

High or/gn

Low or/br



### **CAN Infotainment**

High or/vio

Low or/br



# Kabelfarben der Daten Bus Systeme

## **Antriebs-CAN**

CAN-High = orange/schwarz

CAN-Low = orange/braun

## **Komfort-CAN**

CAN-High = orange/grün

CAN-Low = orange/braun

## **LIN BUS**

nur eine Leitung = violett/

## **Infotainment CAN**

CAN-High orange/lila

CAN-Low = orange/braun

## **Kombi-CAN**

CAN-High = grau/rot

CAN-Low = grau/weiß

## **Diagnose-CAN**

CAN-High = orange/lila

CAN-Low = orange/braun

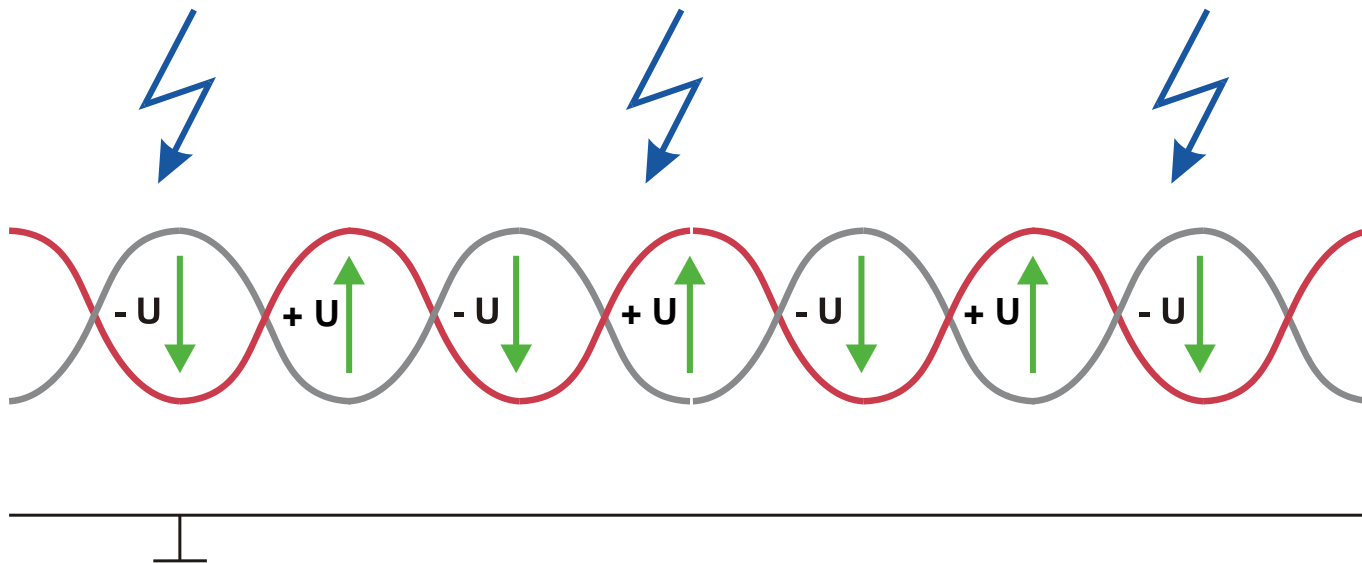
# Die Datenleitungen des CAN-Bus

**Warum 2 Leitungen? Warum Verdrillt? Instandsetzung?**

**Erläuterung Verdrillt:** Magnetfelder in unserer Umgebung stören die Datenübertragung bzw. wirken auf die Spannungspegel ein, die in einer Leitung übertragen werden soll.

Bewegen sich Spannungspegel entgegengesetzt (invertierte Signale) bilden Sie ein gemeinsames Magnetfeld um sich!

Sie schützen sich wie mit einem Schutzschild.



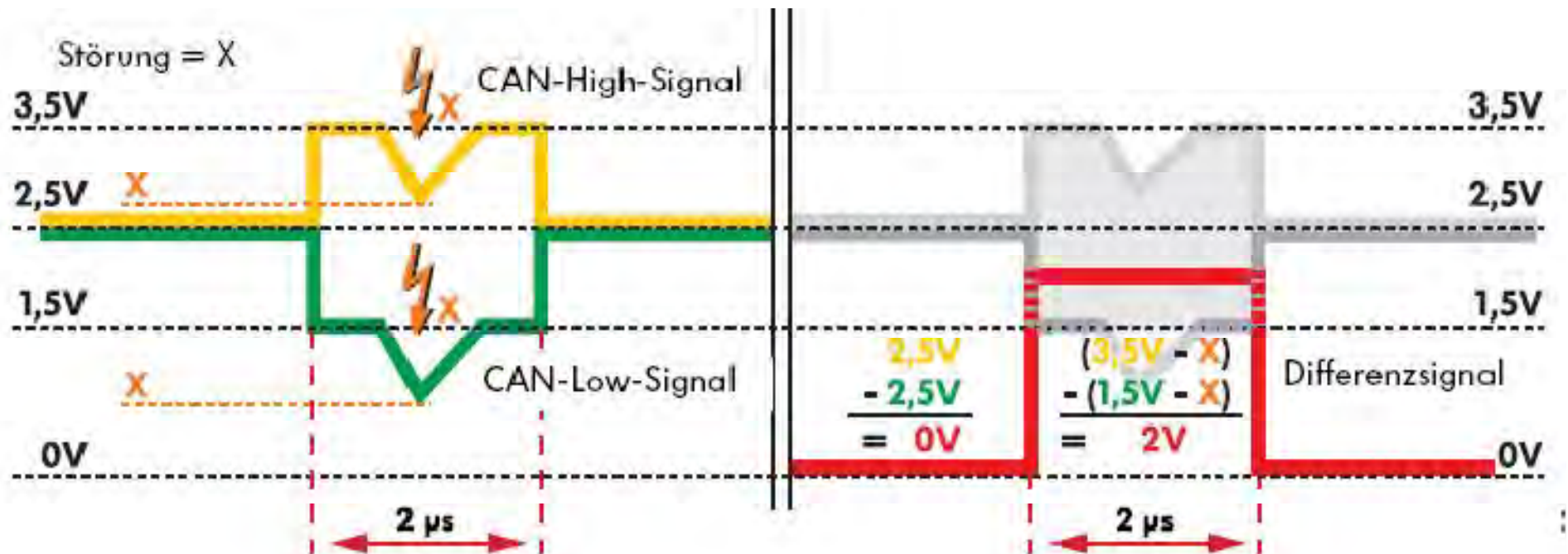
# Die Datenleitungen des CAN-Bus

## Warum 2 Leitungen:

Obwohl der CAN-Bus ein Serielles System ist, besitzt er 2 Datenleitungen.

Diese sogenannte Gleichtaktunterdrückung bzw. Differenzverstärkung trägt zu der hohen Übertragungssicherheit des CAN-Bus bei.

## Beispiel am CAN-Bus Antrieb!



# *Die Datenleitungen des CAN-Bus*

## **Instandsetzung von CAN-Datenleitungen:**

1. Hersteller Angaben beachten (Vorschriften/ Besonderheiten)
2. Schlaglänge: ca. 2cm und gleichmäßig
3. Querschnitt: ca. 0,25 bis 0,5 mm<sup>2</sup>
4. Abschnitt-Instandsetzung



# Begriffserläuterung CAN-Bus

**Redundanz:** dies ist ein Begriff für den Innenraum CAN-Bus bzw. Low-Speed CAN-Bus.

Er wird auch öfters als „**EINDRAHTFÄHIGKEIT**“ beschrieben. **Das bedeutet:** fällt eine Leitung aus, funktioniert der Innenraum CAN-Bus ohne bemerkbare Einschränkung für den Kunden weiter.

**Echtzeitfähigkeit:** der Motor-Bus ist Echtzeitfähig auf Grund seiner hohen Übertragungsgeschwindigkeit. Dieser Bus ist aber nicht Eindrahtfähig! D.h. bei irgendeiner Störung geht alles in den Notlauf.



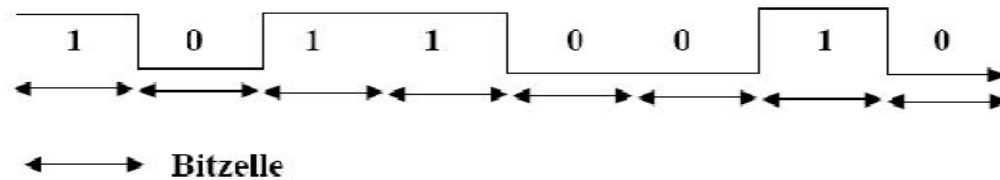
# Begriffserläuterung CAN-Bus

## NRZ Codes

---

NRZ: Non Return to Zero

Beisp.: 1011 0010

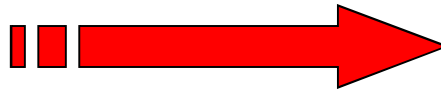




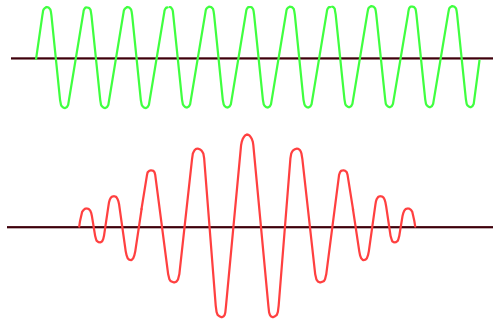
# Begriff Broadcast / Adressierungsverfahren

Adressierungslose Sendung an alle -  
Broadcast

Beispiel



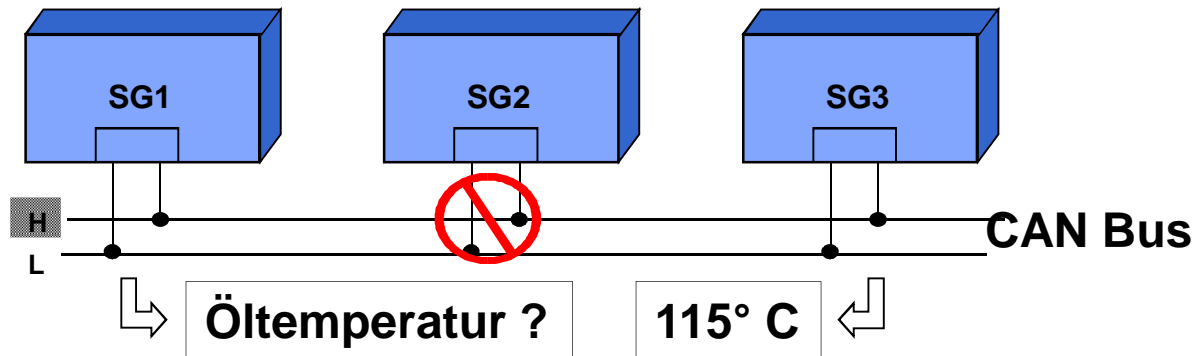
Rundfunk



# Adressierungsverfahren

## Inhaltsbezogene Adressierung - Multicast

## Beispiel CAN



- Botschaften prinzipiell für jeden empfangbar (wie Broadcast)
- Botschaften sind adressiert, das Adressfeld bezeichnet jedoch keine Empfängeradresse sondern den Botschaftsinhalt
- Jeder Empfänger unterscheidet durch individuelle Konfiguration seiner „Mailbox“ welche Inhalte für ihn relevant sind (Akzeptanzfilter)
- Relevante Daten werden automatisch in der Empfängermailbox abgelegt
- Der übrige Datenstrom wird ignoriert

# Adressierungsverfahren

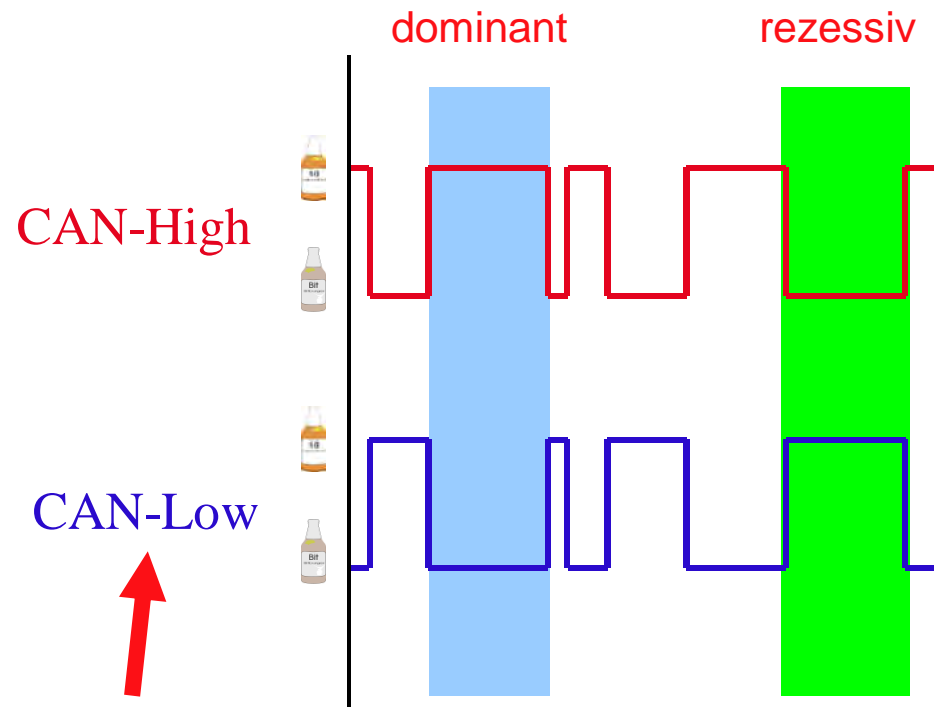
## Inhaltsbezogene Adressierung und die Vorteile

- Aufwand beim Sender gering, ähnlich Broadcast, unabhängig davon ob die Botschaft von einem oder mehreren Empfängern (Multicast) benötigt wird
- Aufwand beim Empfänger gering, ähnlich teilnehmerbezogener Adressierung. Inhaltsbezogene Akzeptanzfilterung liefert die für den Empfänger relevante Daten
- Teilnehmer benötigen kein Wissen über die Netzstruktur. Problemlose Konfiguration möglich
- Teilnehmer können benötigte Informationen ohne Kenntnis der Informationsquelle anfordern (remote frame)

## Unterschied CAN-High zu CAN-Low

Es werden zur selben Zeit auf CAN-High und auf CAN-Low gleiche Botschaften verschickt nur die Pegel sind invertiert. Dabei wird mit NRZ (Non-Return-To-Zero) codiert.

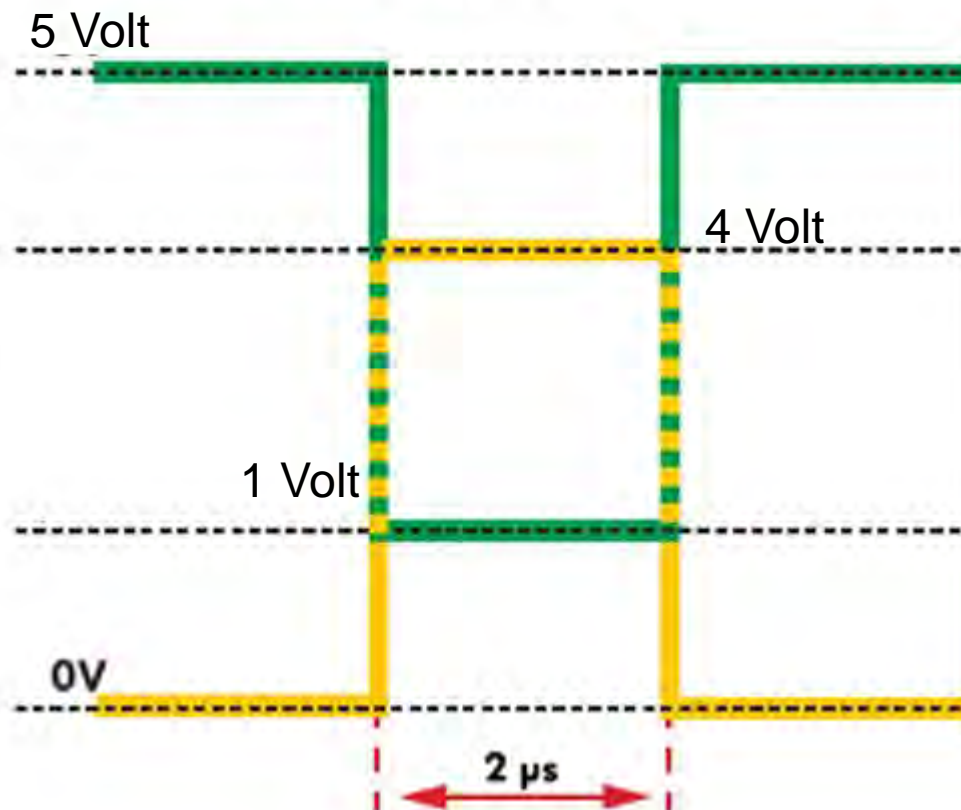
Beispiel :



Das CAN-Low liest sich also in konventioneller Zuordnung für 0 und 1 !

## Unterschied CAN-High zu CAN-Low (Low-Speed)

### Signalverlauf des CAN-Datenbus Komfort/ Infotainment



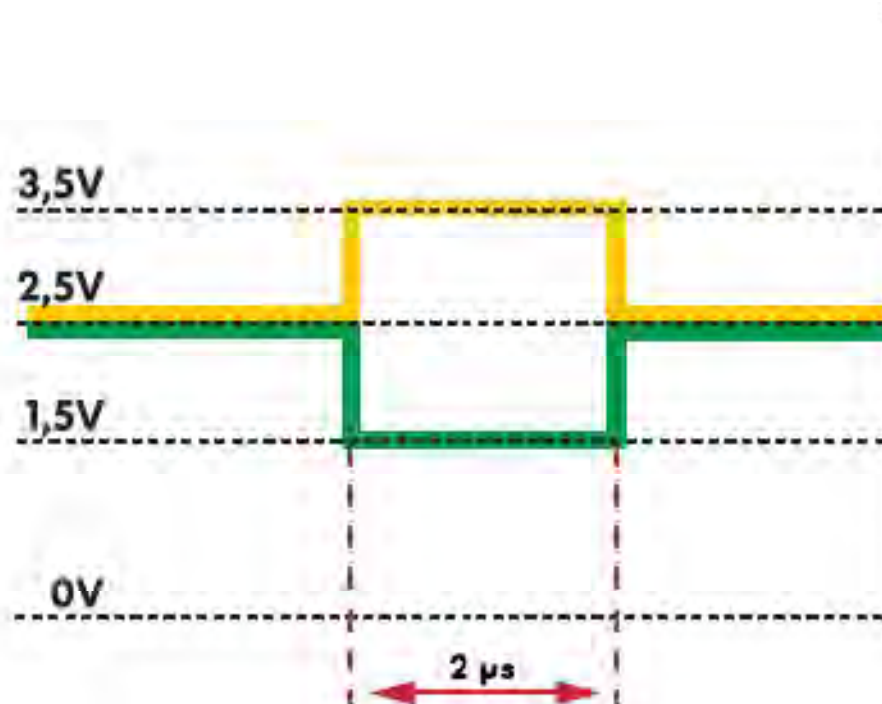
Im **dominanten** Zustand fällt die **CAN-Low-Leitung** auf ca. 1 Volt ab.

Im **rezessiven** Zustand liegt die **CAN-High-Leitung** auf ca. 0V, die **CAN-Low-Leitung** auf ca. 5V.

Im **dominanten** Zustand geht die **CAN-High-Leitung** auf ca. 4 Volt

## Unterschied CAN-High zu CAN-Low (High-Speed)

Signalverlauf auf dem CAN-Datenbus am Beispiel des CAN-Datenbus Antrieb



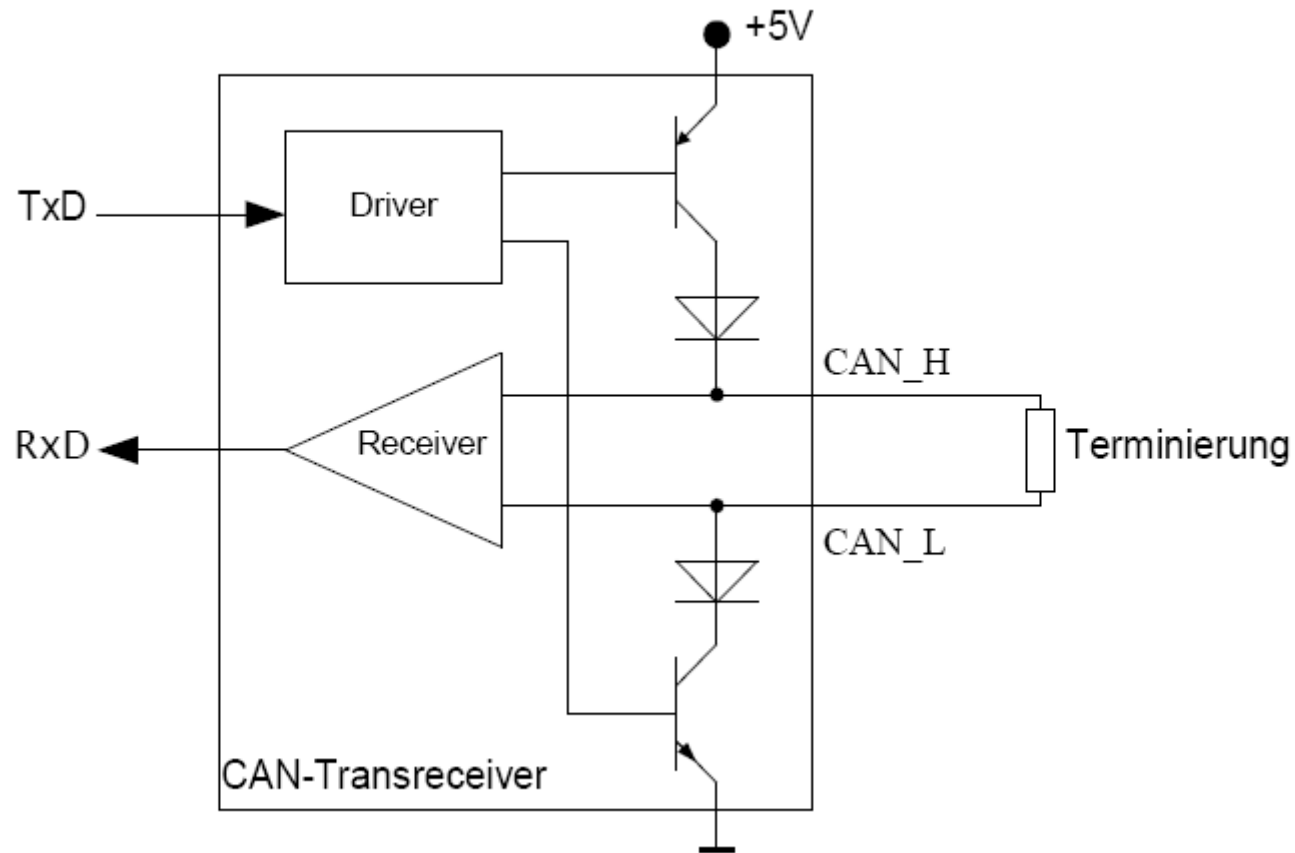
Im **dominanten** Zustand geht die CAN-High-Leitung auf ca. 3,5V

Im **rezessiven** Zustand liegen beide Leitungen bei ca. 2,5V (Ruhepegel)

Im **dominanten** Zustand fällt die CAN-Low-Leitung auf ca. 1,5V

S269\_005

# Wie wird das High & Low Signal geschaltet?



# ***Bildaufnahme mit Gutmanntester/ Innraumbus***



# *Datenrahmen / Datenprotokoll*

## **Einführung und Definition**

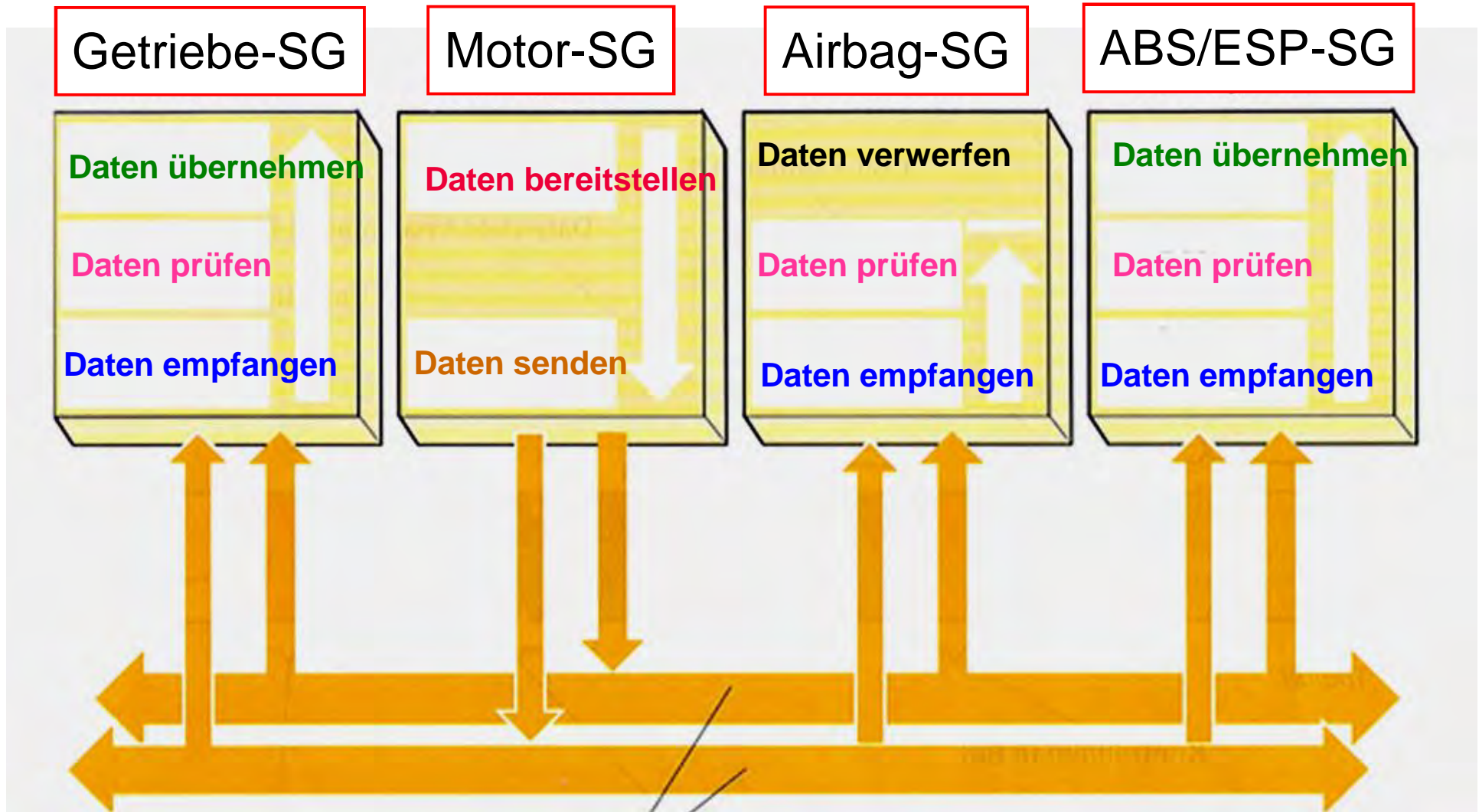
### **CAN Controler Area Network**

Es handelt sich um ein Protokoll für serielle Datenübertragung mit einer hohen Geschwindigkeit und hoher Sicherheit. Bitraten bis zu 1 MBit/s fordern Mechanismen zur korrekten Arbeitsweise des Gesamtsystems.

## **Mechanismen und Eigenschaften des CAN Protokolls**

- Vorrang von Nachrichten > Identifier
- Gewährleistung der vorgegebenen Latenzzeiten > Nachrichtenpause
- Systemweite Datenkonsistenz > das alles STG die Nachricht nahe zu gleich erreichen
- Multi-Masterbetrieb > jeder hört und sagt
- Zerstörungsfreies Buszugriffsverfahren > Multicast prinzip
- Fehlererkennung und Fehleranzeige > Remote Frame
- Flexibilität bei der Konfiguration
- Bei Nachrichtenstörung > Neuübertragung der gesendeten Botschaft
- Unterscheidung von sporadischen Fehlern und Dauerfehlern
- Automatische Abschaltung dauerhaft fehlerhafter Stationen
- Zeitsynchronisation der Stationen

# Datenrahmen/ Datenprotokoll



# *Datenrahmen / Datenprotokoll*

## Frames

**DATA FRAME**  
(Datentelegramm)

Datenübertragung vom Sender zu den Empfängern.

**REMOTE FRAME**  
(Datenanforderung)

Zur Anforderung eines Datentelegramms.

Signalisierung eines Fehlers durch Sender oder Empfänger.

**OVERLOAD FRAME**  
(Überlasttelegramm)

Zur Erzeugung einer Verzögerung zwischen aufeinanderfolgenden Telegrammen.  
*Wird nur noch selten im PKW-Bereich angewendet.*

# *Data-/Remote Frame*

**CAN unterstützt zwei Formate von Botschaftsrahmen:**

## **Standard Data Frame**

11 Bit Identifier => *somit 2048 unterschiedliche Identifier möglich*

Wird im PKW -Bereich eingesetzt

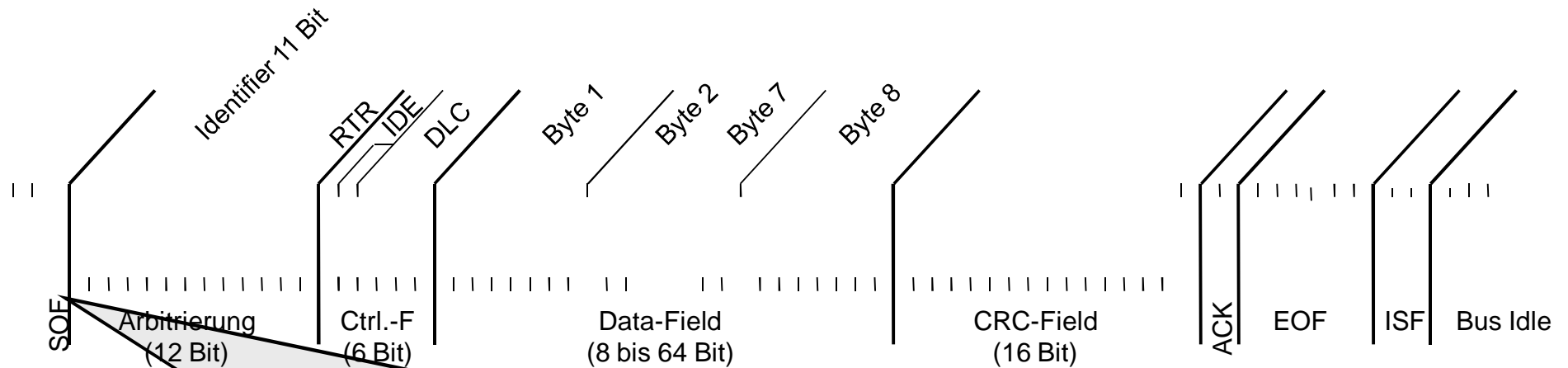
## **Extended Data Frame**

29 Bit Identifier => *somit 536 870 912 unterschiedliche Identifier möglich*

Wird im LKW und Personenkraftverkehr eingesetzt

In deutschen Pkw wird z.Z. nur das Standard Data Frame eingesetzt!

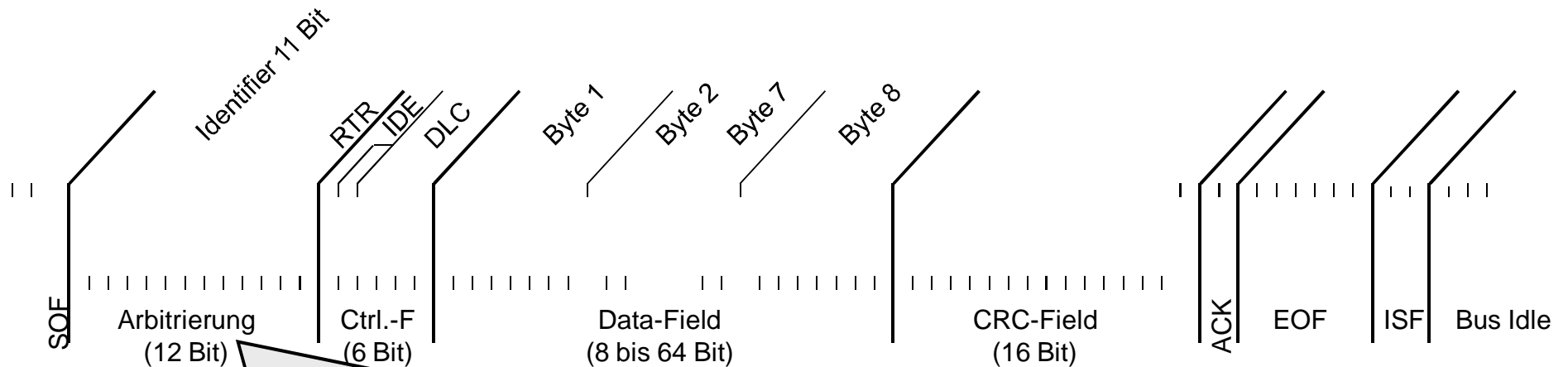
# Standard Data-Frame



## Start of Frame (SOF) Telegrammanfangskennung (Starbit)

- Kennzeichnet den Beginn eines Daten- oder Datenanforderungstelegramms
- Besteht aus einem dominanten Bit (0)
- Über erste Flanke des Start-Bit erfolgt Synchronisation aller Teilnehmer auf den Sender
- Alle Stationen synchronisieren sich auf die Flanke des SOF

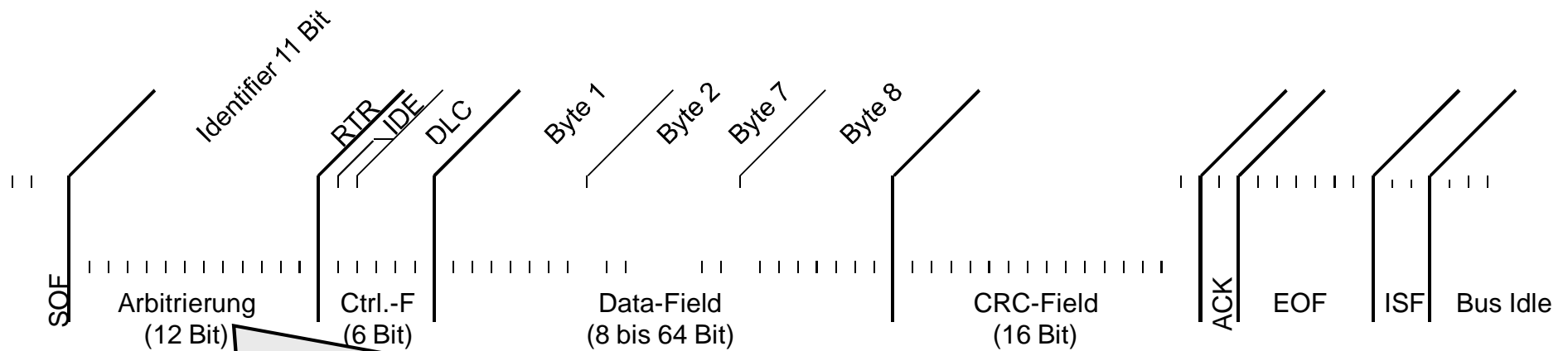
# Standard Data-Frame



## Arbitration Field (Arbitration = Entscheidung; Arbitrer = Schiedsrichter)

- Länge 12 Bit
- Besteht aus dem Identifier Field mit 11 Bit und dem RTR-Bit (Remote Transmission Request Bit)
- Enthält beim Standard Frame den gesamten Identifier und beim Extended Frame die oberen 11 Bit

# Standard Data-Frame



## Identifizier Field

- 11 Bit im Standard-Format
- Unterscheidung von  $2^{11}$  (2048) Nachrichten
- Kennzeichnet Inhalt und Priorität einer Nachricht

## Identifizier 2C1h:

Binär: 0 1 0 1 1 0 0 0 0 0 1

# *Zerstörungsfreie bitweise Arbitrierung*

## **Buszugriffskonflikt:**

- Buszugriffskonflikte treten auf, da der CAN ein asynchroner serieller Multi-Master-Bus ist.
- Buszugriffskonflikte werden zerstörungsfrei durch die bitweise Arbitrierung gelöst.
- Der kleinste hexadezimale Identifier hat automatisch die höchste Priorität. (der mit den meisten Nullen vorn)
- Die Durchführung geschieht nach dem „wired-and“-Schema. (UND-Logik)  
(nur wenn alle rezessives Bit senden, ist der Bus auch rezessiv!)

**Dominantes Bit „überschreibt“ rezessives Bit!**



# Arbitrierung

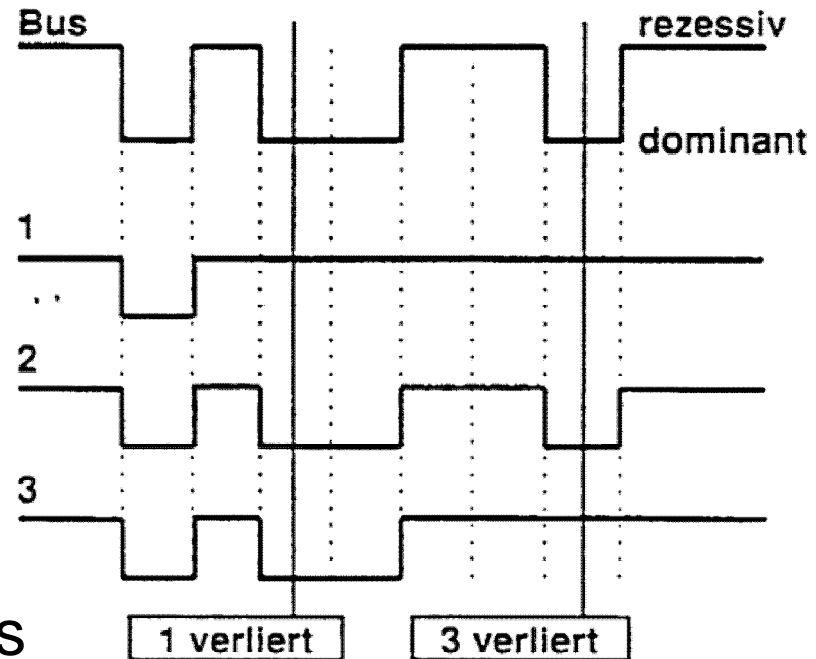
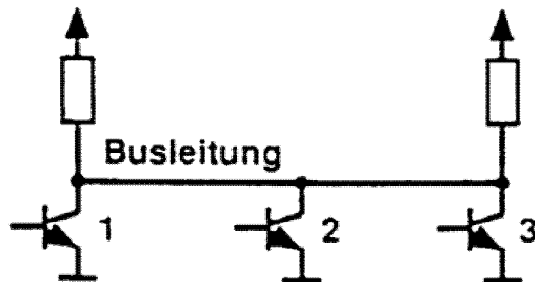
	Identifizier	
<b>SMLS:</b>	<b>2C1 =</b>	<b>01011000001</b>
<b>Kombi:</b>	<b>351 =</b>	<b>01101010001</b>
<b>ILM:</b>	<b>531 =</b>	<b>10100110001</b>
<b>ZKE:</b>	<b>591 =</b>	<b>10110010001</b>

**0 = dominant**  
**1 = rezessive**

***wired-and = dominant überschreibt rezessive***

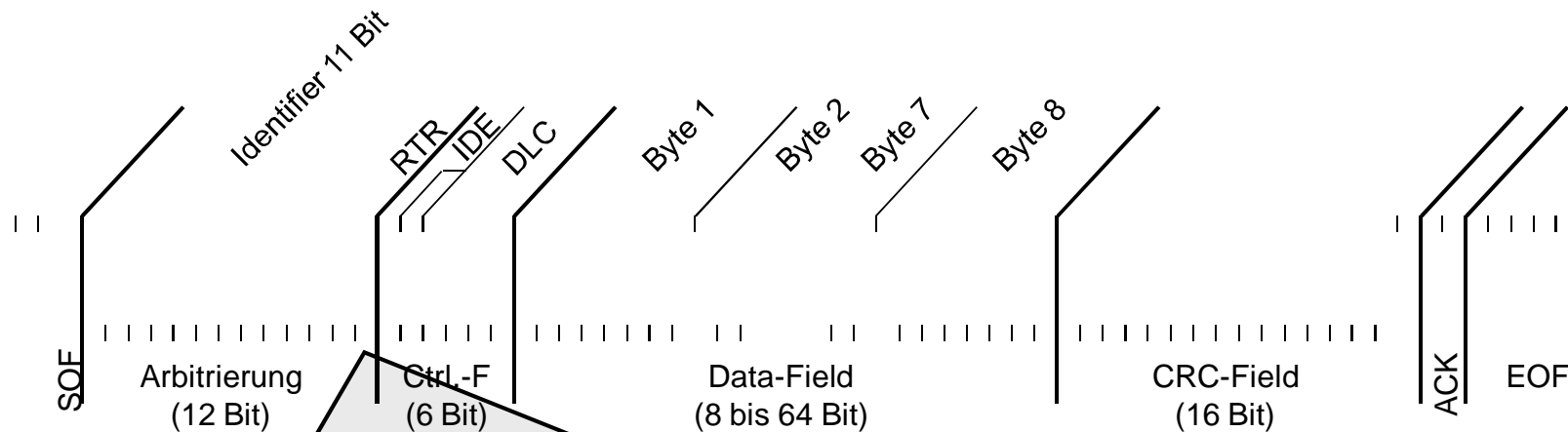
# Zerstörungsfreie bitweise Arbitrierung und Effizienz der Busvergabe

## Buslogik



Nach dem „wired-and“-  
Schema wird ein dominantes  
Bit ein rezessives Bit  
überschreiben.

# Standard Data-Frame



## RTR-Bit (Remote Transmission Request Bit)

Zeigt an, ob es sich um ein Datentelegramm oder eine Sendeaufforderung handelt.

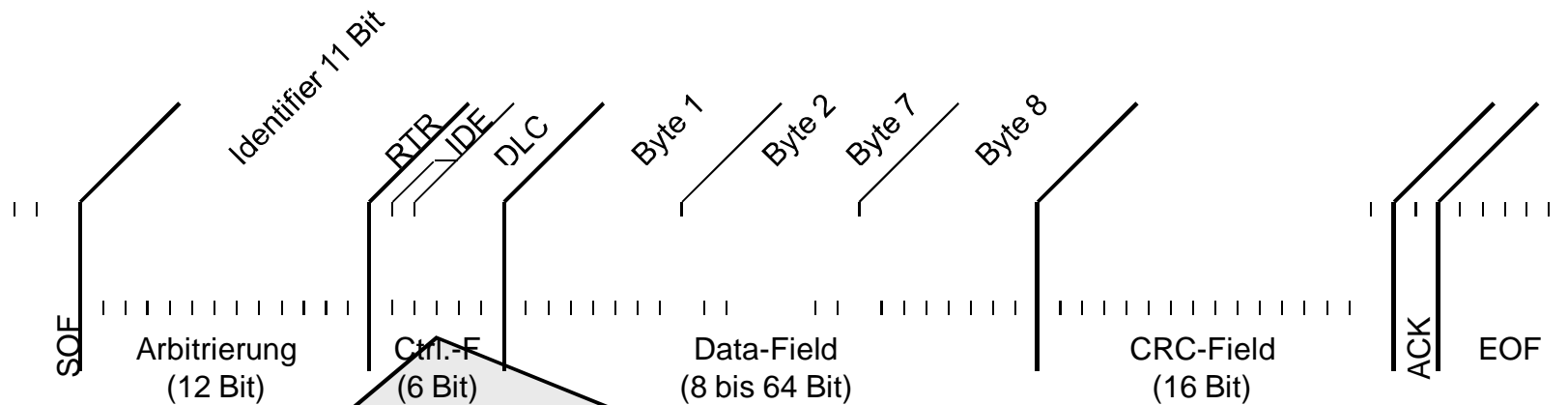
Das RTR-Bit ist bei einem

*Datentelegramm (Data Frame) dominant (0), bei einem*

*Datenanforderungstelegramm (Remote Frame) rezessive (1)*

Wird zeitgleich eine ID von einem Teilnehmer angefordert und von einem anderen dieselbe als Datentelegramm gesendet, so hat das Datentelegramm durch das dominante RTR-Bit Vorrang.

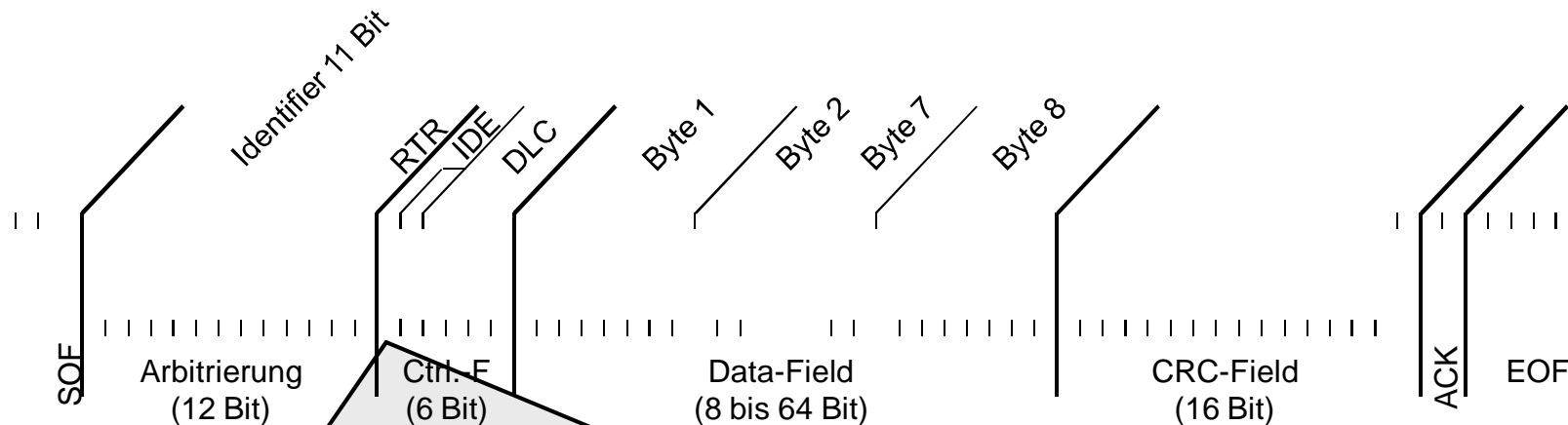
# Standard Data-Frame



## Control Field

- Länge 6 Bit
- Besteht aus dem IDE-Bit einem reserved Bit und dem Data Length Code mit 4 Bit

# Standard Data-Frame



## IDE-Bit (Identifer Extension Bit) (Extension = Ausdehnung)

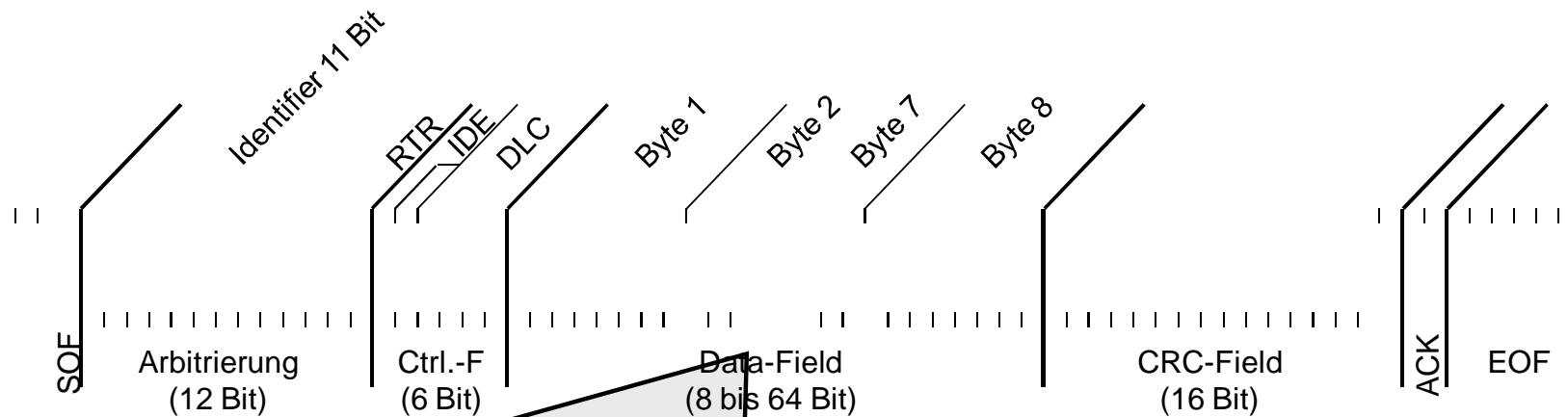
Das IDE-Bit dient der Unterscheidung zwischen Standard und Extended Data Frame

*Standard Data Frame: IDE-Bit dominant (0)*

*Extended Data Frame: IDE-Bit rezessive (1)*

*Auf ein rezessives (1) IDE-Bit folgt beim Extended-Format weitere 18 Identifier-Bits die zusammen mit den 11 ersten Identifier-Bits den 29 Bit langen Extended-Identifier ergeben.*

# Standard Data-Frame



## Data Field

- Enthält die Nutzdaten
- Ist die eigentliche Nutzinformation einer CAN-Nachricht
- 0 bis 8 Byte lang
- Inhalt ist in der jeweiligen K-Matrix des Fahrzeugs festgeschrieben

# K-Matrix

In der Kommunikations-Matrix ist festgelegt, welches Steuergerät Daten auf den CAN-Bus sendet und welche Daten („Objekte“) durch ein Steuergerät weiter verarbeitet werden.

## Grund für Matrix:

- viele Steuergeräte
- verteilte Funktionalität
- hohe Komplexität

BOTSCHAFT	IDENTIFIER	INFORMATIO	AAG_D3	BATTERIEMA	BF_SITZMEM	DACHMODUL_	EINPARKHIL	F_SITZMEMO	FBS_D3	FINGERPRIN	GATEWAY_D3	ILM_HINTEN	ILM_VL_D3	ILM_VR_D3	KLIMA_D3	KLIMA_HL_D	REIFENDRUC	SITZELEKTR	SMLS_D3	STANDHEIZU	TSG_BF_D3	TSG_FD3	TSG_H_L_D3	TSG_H_RD3	Bemerkung
Blinkmodi	396	Blinken_links	E							E	S	E													
Blinkmodi	396	Blinken_rechts	E							E	S	E													
Blinkmodi	396	Blinken_Frequenz	E							E	S	E													
Blinkmodi	396	Blinken_Frequenz_Kombi								E	S	E													
Blinkmodi	396	Blinken_Frequenz_Anhaenger								E	S	E													
SMLS_1	2C2	Blk_links								E	E	E							S						1 = Blinkerhebel Richtung links betätigt
SMLS_1	2C2	Blk_rechts								E	E	E							S						1 = Blinkerhebel Richtung rechts betätigt

# Beispiel an der Botschaft 2C1

2C1 3 01 02 89

ID:	Len:	Daten:	Anzahl:	Periode:
271	1	07 .. .. . . . . . .	959	100
2C1	3	01 02 89 .. .. . . . .	961	100
351	8	44 00 00 00 00 86 86 10	960	100

Byte

1

2

3

Hex

01

02

89

Bit

0 0 0 0 0 0 0 1

0 0 0 0 0 0 1 0

1 0 0 0 1 0 0 1

Wisch-Wasch hinten  
 Intervall Heckwischer ein  
 n.n.

Wisch-Wasch vorne

Wischer Stufe 2  
 Wischer Stufe 1

Intervall ein  
 Tippwischen ein

KD-Fehler  
 Bordcomputer Reset

Bordcomputer Cursor Up  
 Bordcomputer Cursor Down

Intervallstufen  
 1 = 0001 = 1  
 2 = 0101 = 5  
 3 = 1001 = 9  
 4 = 1101 = 13



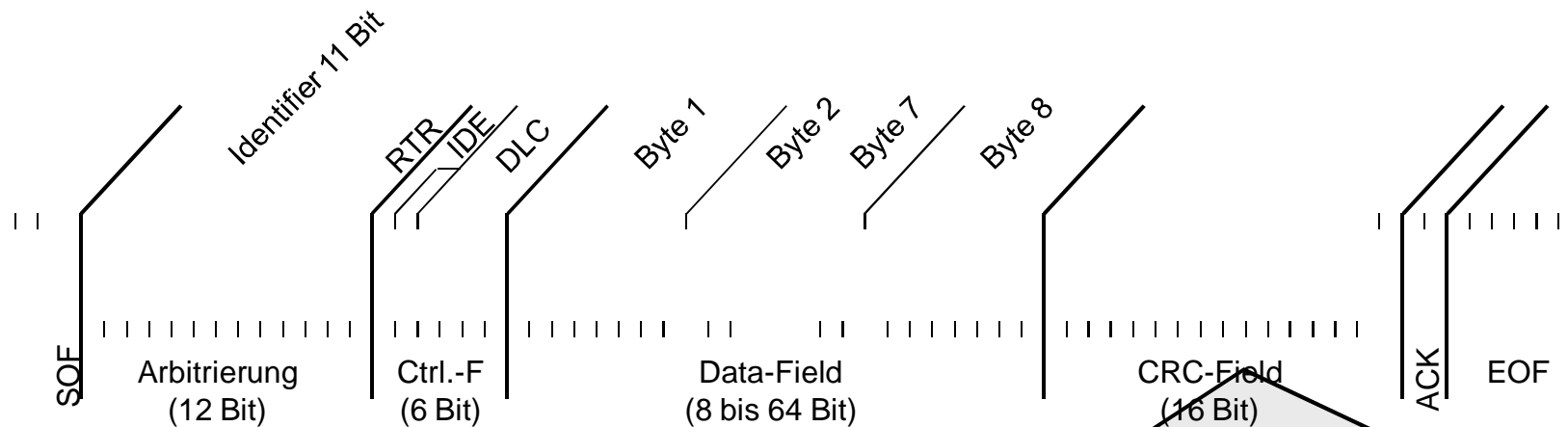
# Botschaften auf dem Bus Nach (1,5Sek. Betriebszeit)

PCANView 32 - Empfangsliste

Empfangen:

ID:	Len:	Daten:	Anzahl:	Periode:	RTR-Anz.:	RTR-Per.:
271	1	07 -- -- -- -- -- --	959	100		
2C1	3	01 02 89 -- -- -- --	961	100		
351	8	44 00 00 00 00 86 86 10	960	100		
353	6	0F 00 00 00 00 00 -- --	960	100		
400	6	01 01 00 00 00 00 -- --	523	190		
401	6	08 01 00 00 00 00 -- --	522	181		
408	6	0A 01 00 00 00 00 -- --	522	191		
40A	6	00 01 00 00 00 00 -- --	522	191		
531	2	00 01 -- -- -- -- --	1966	50		
551	1	00 -- -- -- -- -- --	480	200		
573	3	0B 00 00 -- -- -- --	480	200		
591	4	03 00 0F 00 -- -- --	479	201		
621	3	00 00 80 -- -- -- --	960	100		
623	8	00 00 01 44 18 08 20 00	96	992		
627	8	00 00 00 00 00 00 00 00	192	511		
631	4	08 00 00 00 -- -- --	96	1002		
635	3	0A 00 01 -- -- -- --	479	201		
651	6	C0 14 61 20 11 40 -- --	192	501		
653	3	00 01 00 -- -- -- --	192	501		

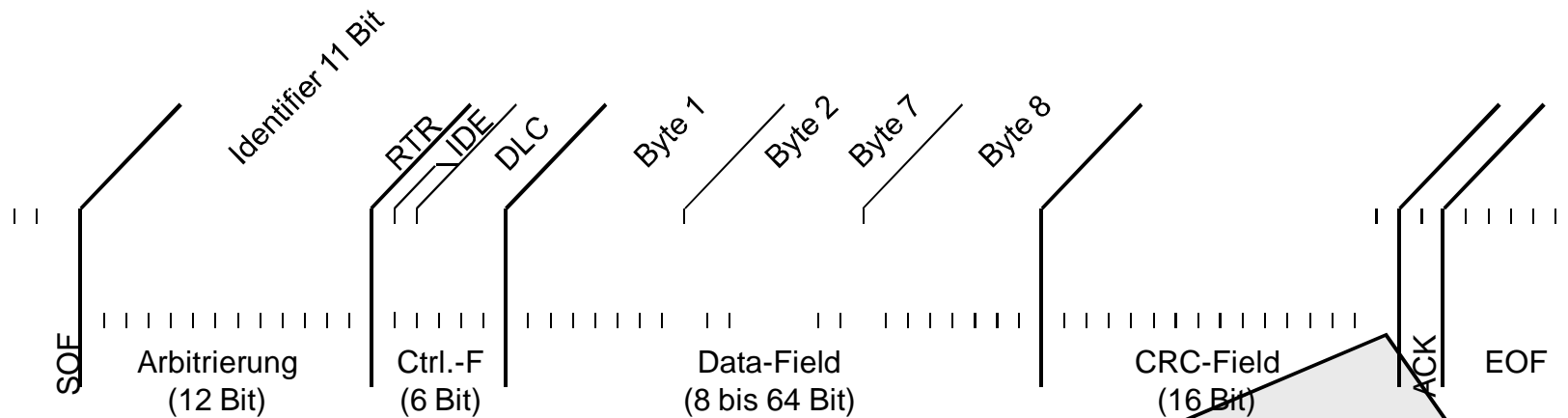
# Standard Data-Frame



## CRC (Cyclic Redundancy Check) Sequence

- 15 Bit Prüfsummenpolynom zur Erkennung von Übertragungsfehlern
- CRC Summe wird vom Start of Frame bis Ende Datenfeld vom Sender und Empfänger berechnet
- Empfänger vergleichen CRC Summe
- ist keine Übereinstimmung zwischen Wert-Sender und Wert-Empfänger werden die Daten als fehlerhaft verworfen

# Standard Data-Frame

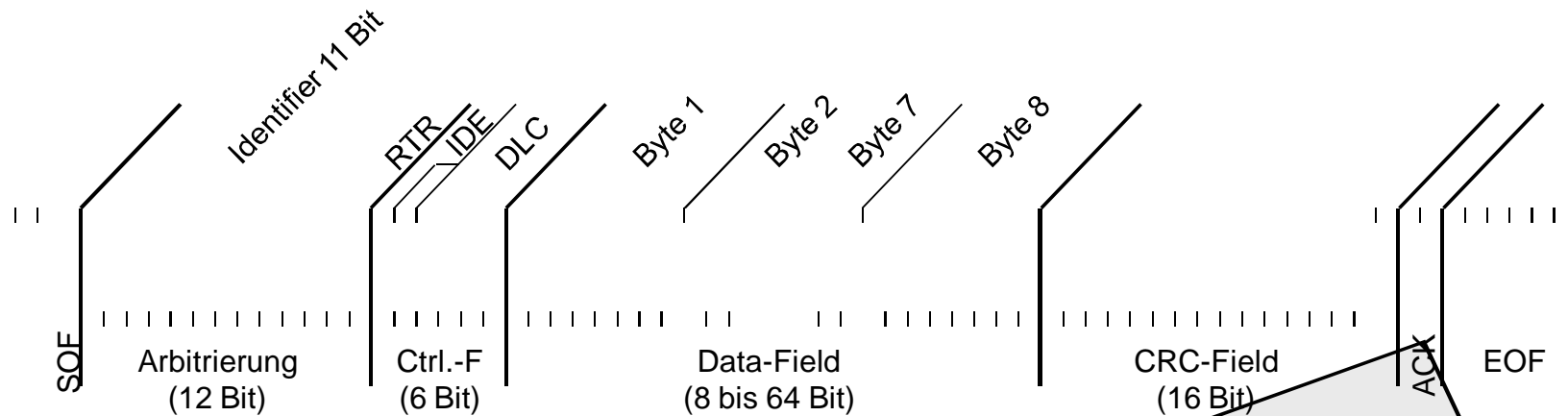


## CRC Delimiter (Feldtrenner; Begrenzer)

1 Bit

- wird immer rezessive (1) gesendet
- schließt die CRC-Sequence ab
- der CRC dient einzig zur Fehlererkennung

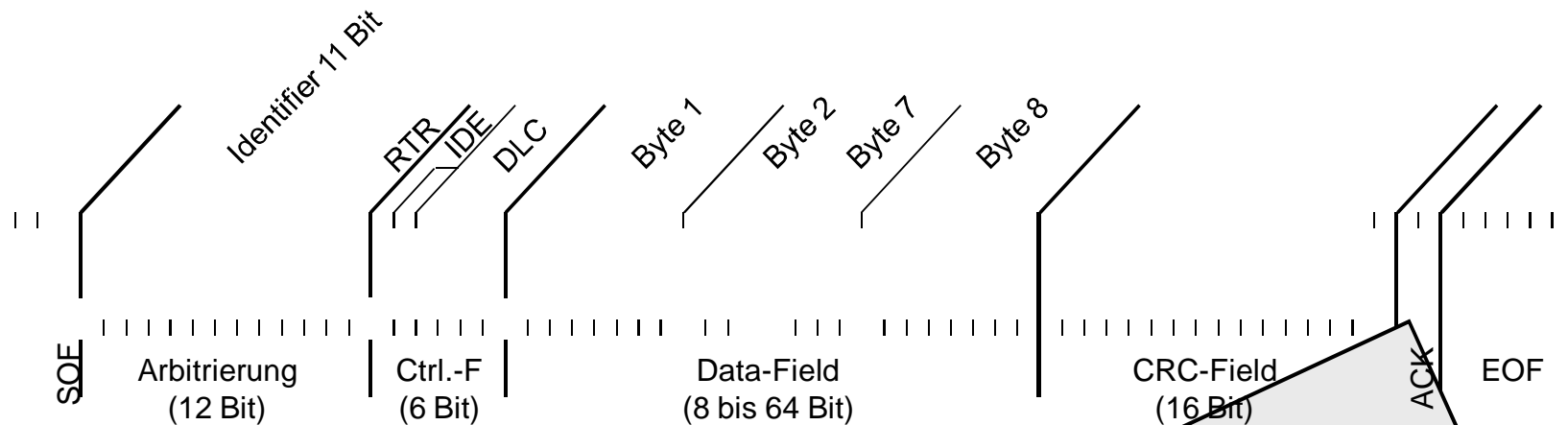
# Standard Data-Frame



## ACK-Field (Bestätigungsfeld)

- 1 Bit Acknowledge Slot
- 1 Bit Acknowledge Delimiter
- stationsneutrale positive Empfangsbestätigung

# Standard Data-Frame

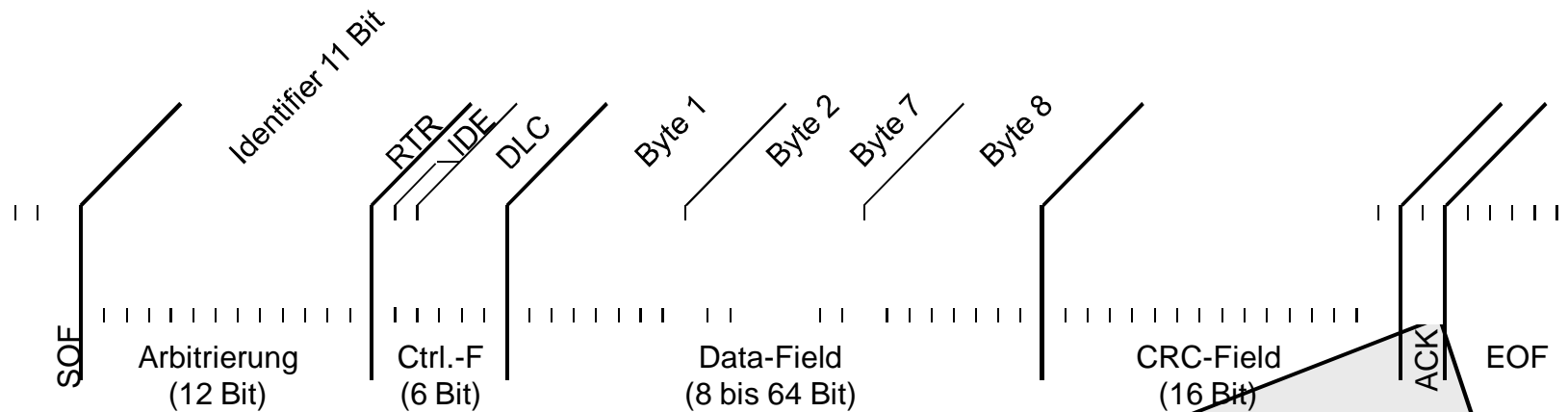


## ACK-Slot

1 Bit Acknowledge Slot

- wird vom Sender der Nachricht rezessive (1) gesendet
- mindestens ein Empfänger, der die Nachricht korrekt empfangen hat, bestätigt in diesem Zeitschlitz mit einem dominanten Bit, auch wenn er die Nachricht nicht weiter verarbeitet
- wird die Nachricht nicht dominant bestätigt hat kein Empfänger die Nachricht empfangen
- es wird bei nicht dominanter Bestätigung kein Error-Frame gesendet

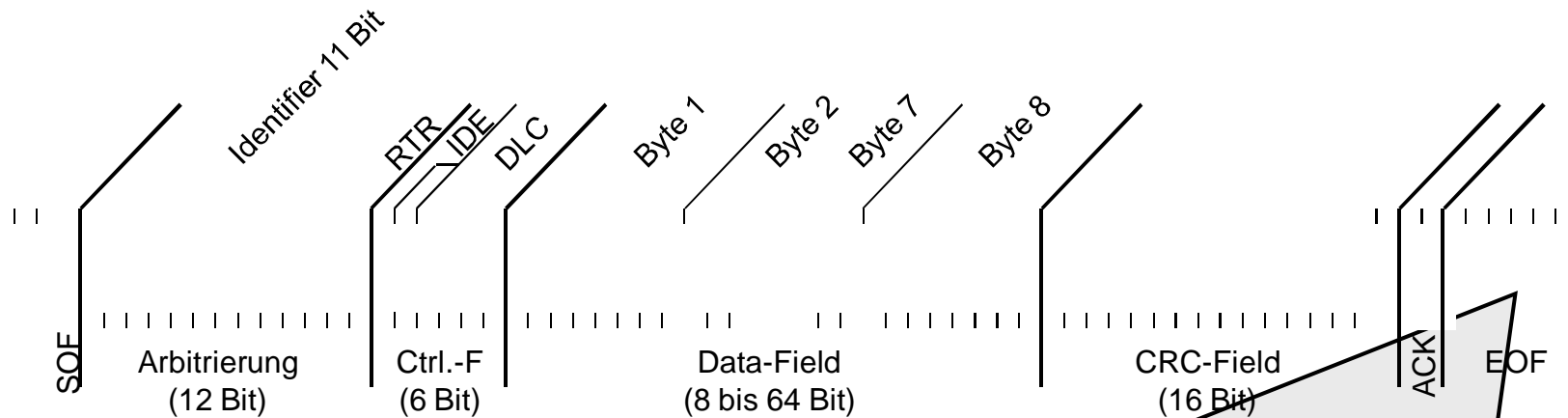
# Standard Data-Frame



## ACK-Delimiter (Begrenzer)

- 1 Bit
- wird vom Sender der Nachricht immer rezessive (1) gesendet
- begrenzt den ACK-Slot
- dient als Laufzeitpuffer für die von den Empfängern gesendeten ACK-Bits

# Standard Data-Frame



## End of Frame EOF (Telegrammendeckennung)

- 7 Bit lang
- immer rezessive (1)
- Kennzeichnet das Ende eines Telegramms
- kein Bit-Stuffing in diesem Bereich

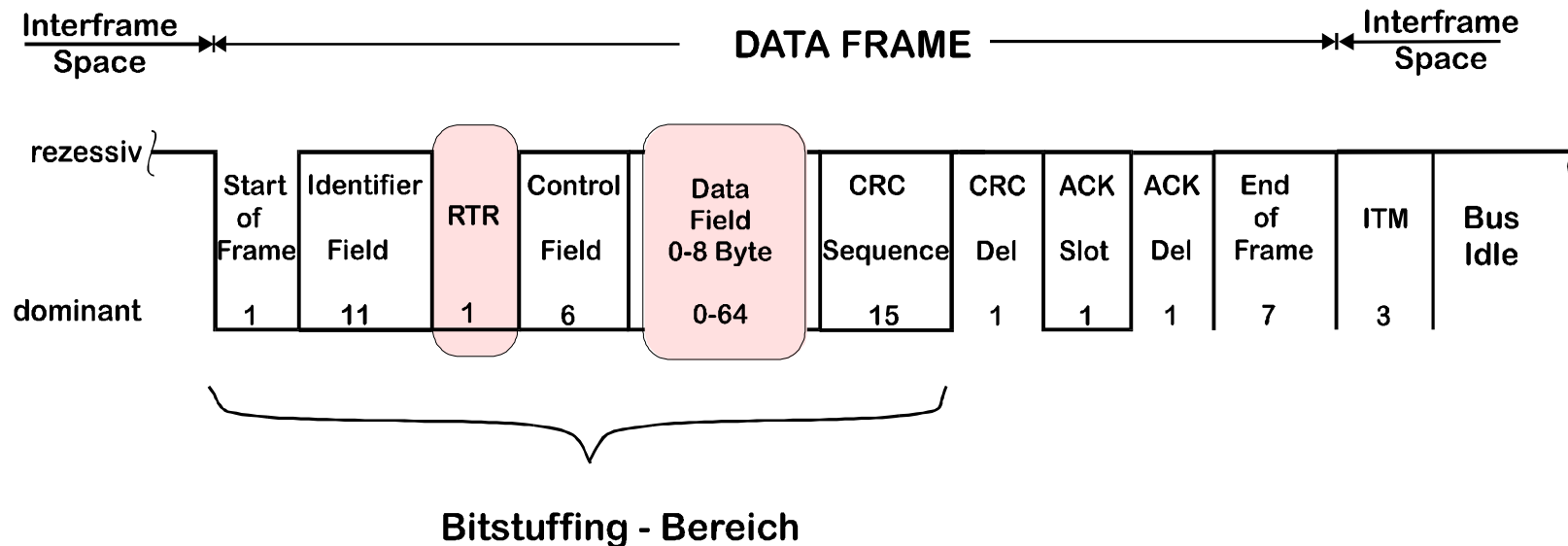
# Bit-Stuffing

**Stuff** = Stopfen > Stuffingregel

**NRZ** > **Vorteil:** geringere Frequenz < 50% als bei anderen Bitcodierungen

**Nachteil:** fehlende Synchronisierung bei vielen gleichen Bits hintereinander

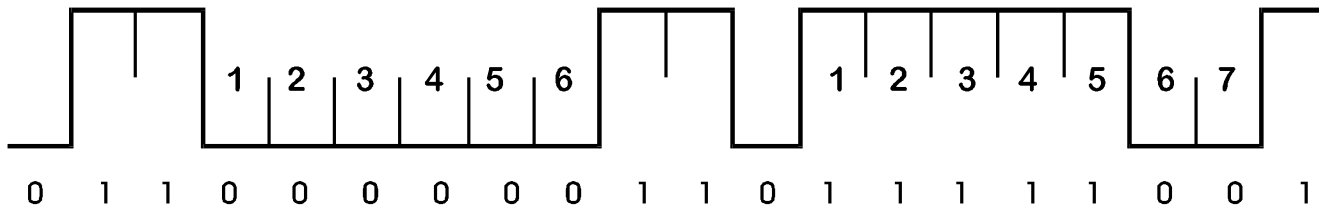
**Abhilfe:** Einstopfen eines invertierten Synchronisierbits nach einem 5. Gleichen Bit beim Empfänger wieder herausfiltern des eingestopften Bits



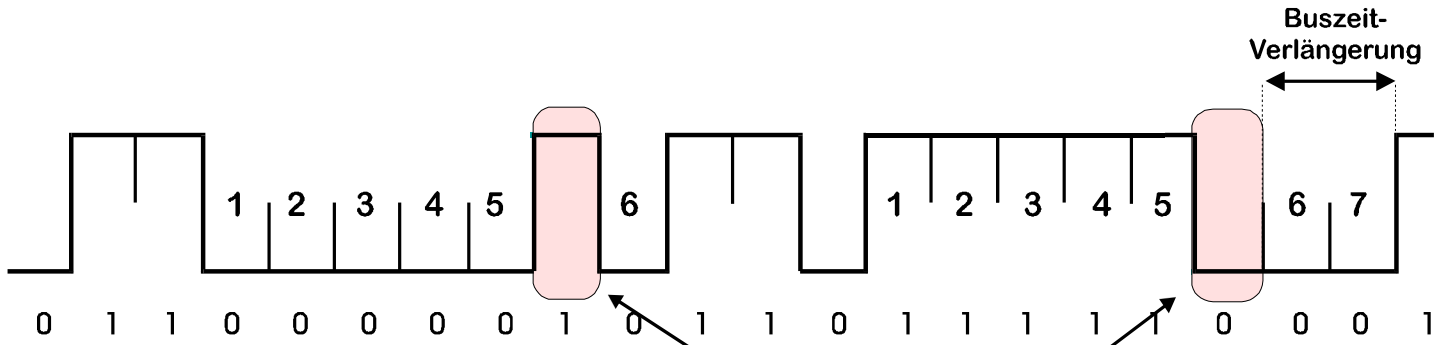


# Bit-Stuffing

**SENDER**  
zu sendende  
Bitfolge  
vor Bitstuffing

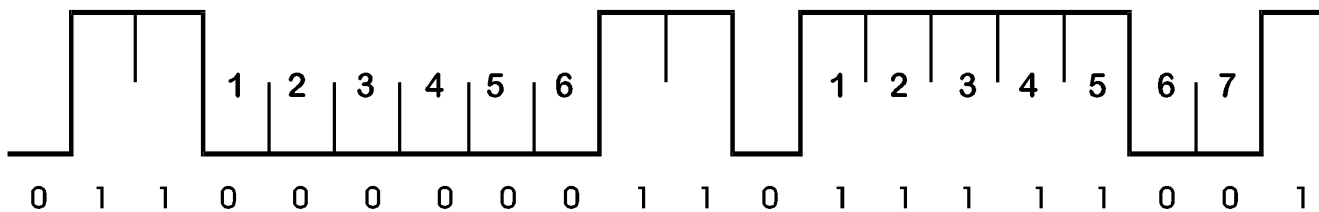


**BUS**  
gesendete  
Bitfolge nach  
Bitstuffing

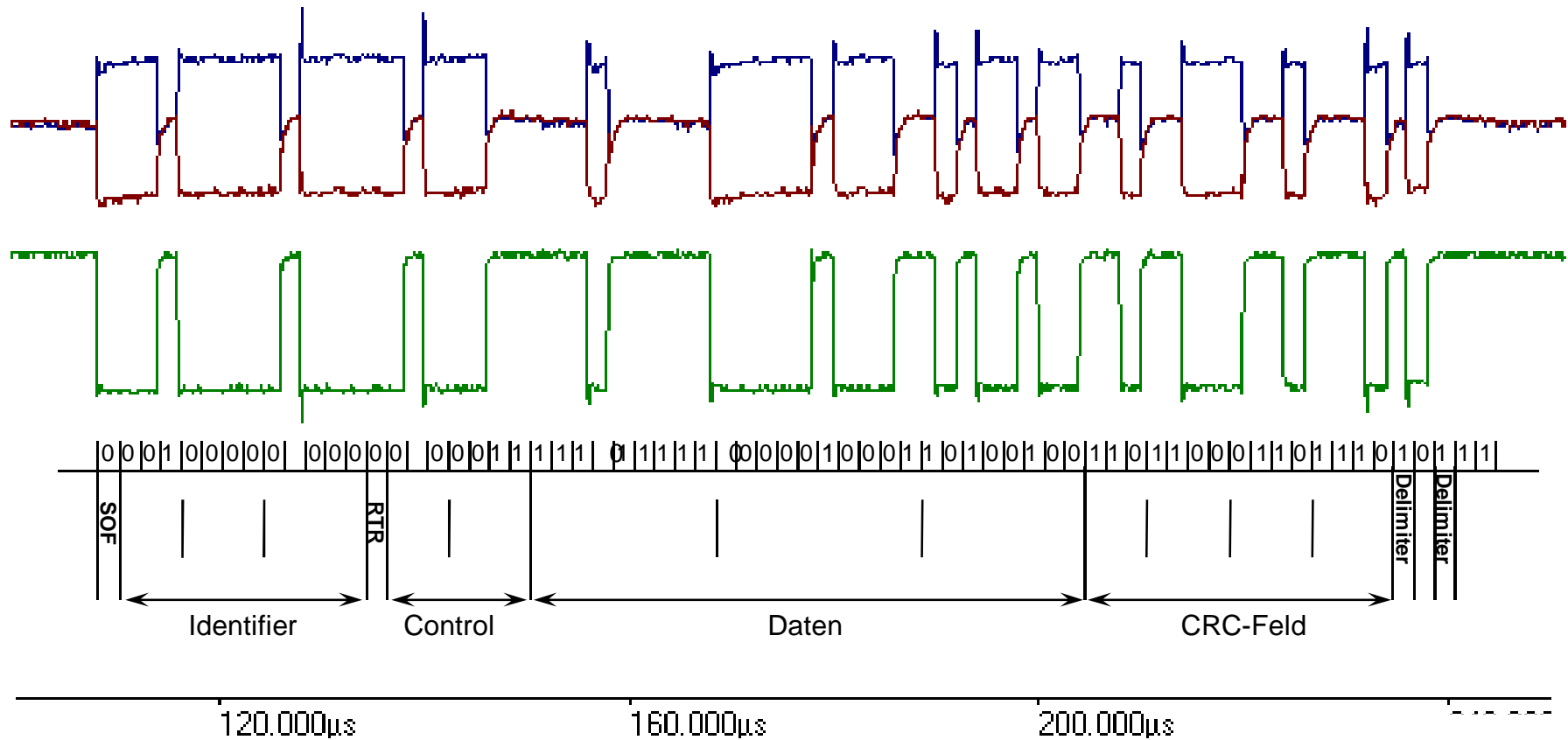


**STUFF BITS**

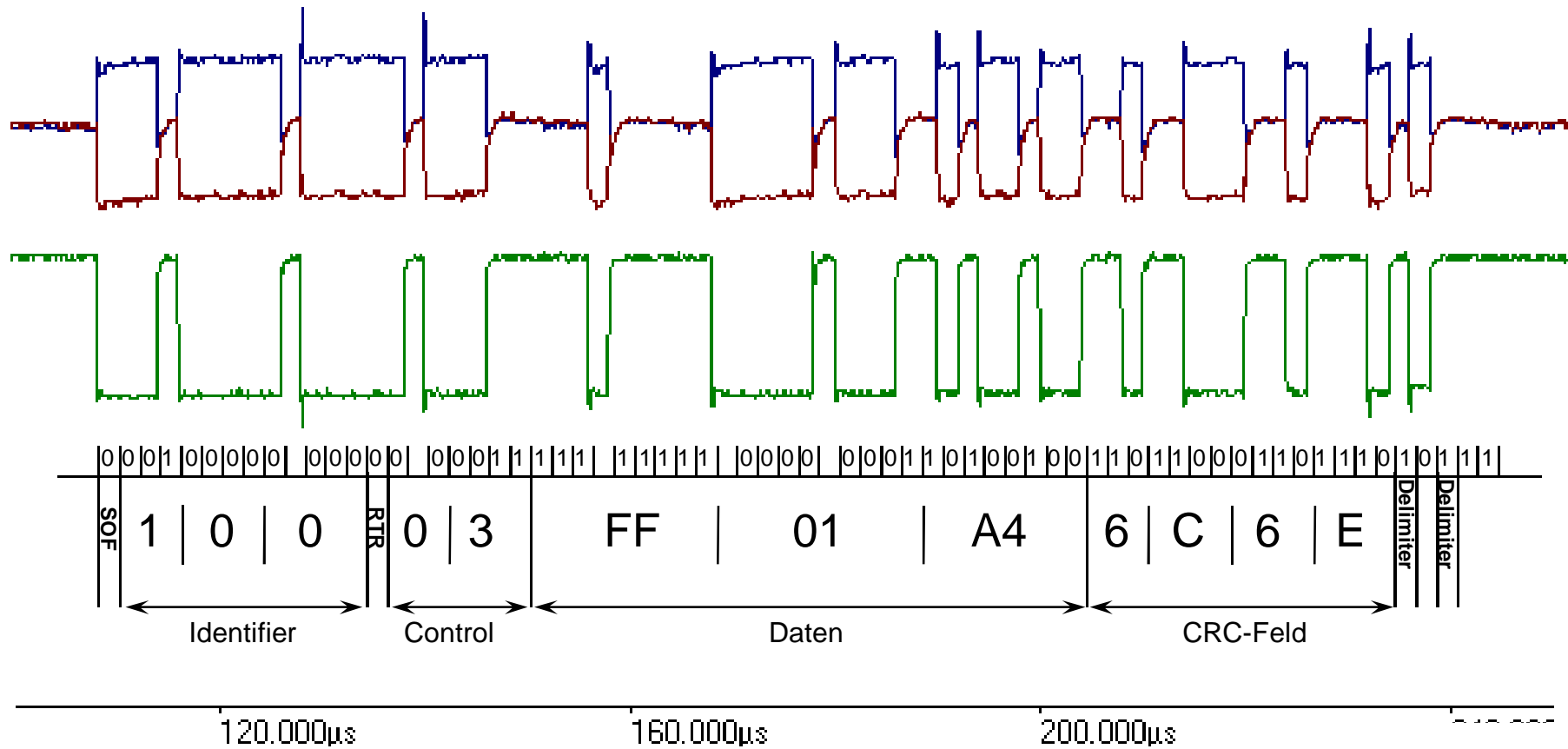
**EMPFÄNGER**  
empfangene  
Bitfolge nach  
Destuffing



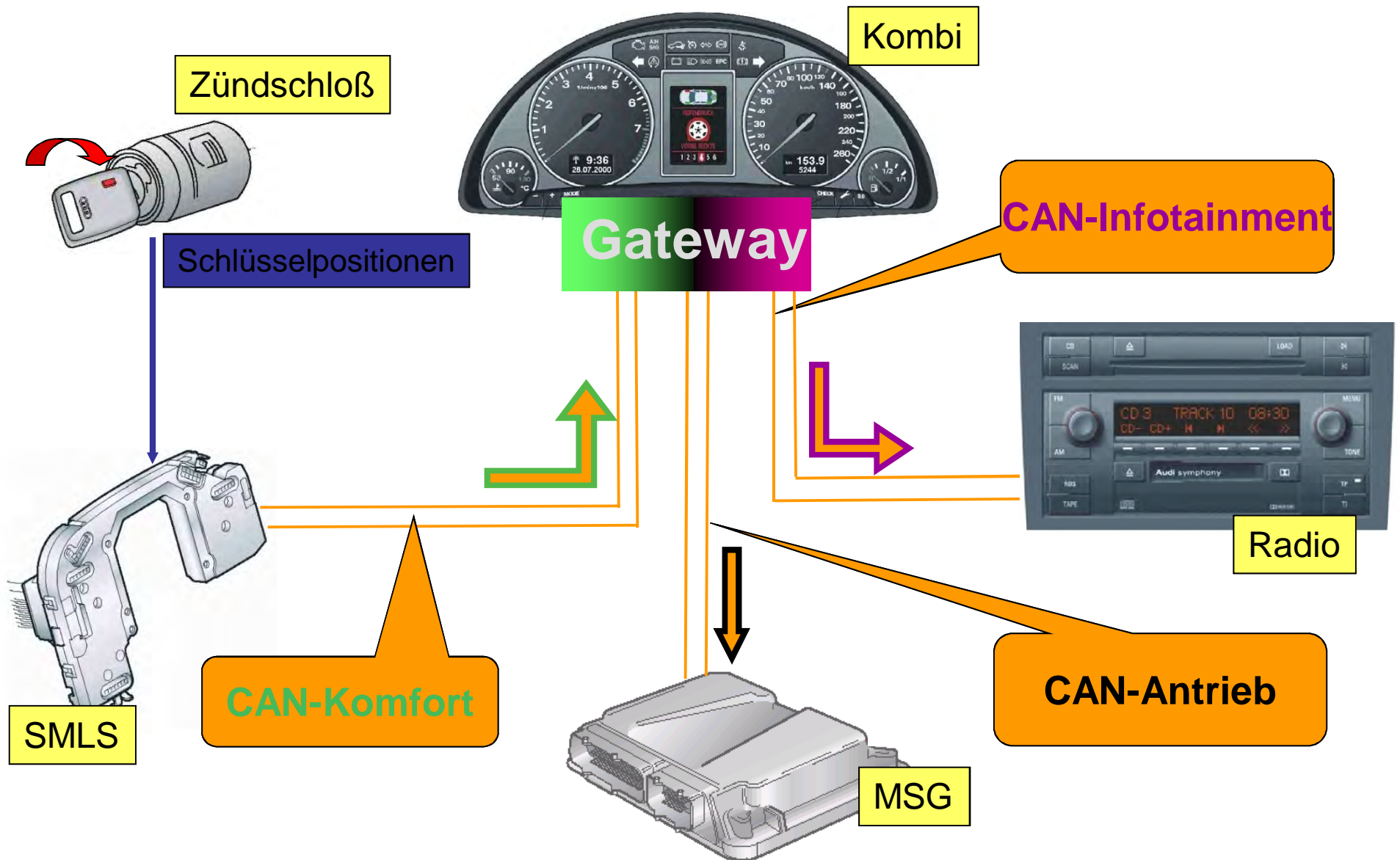
# Aufgabe: Auswerten folgender Botschaft



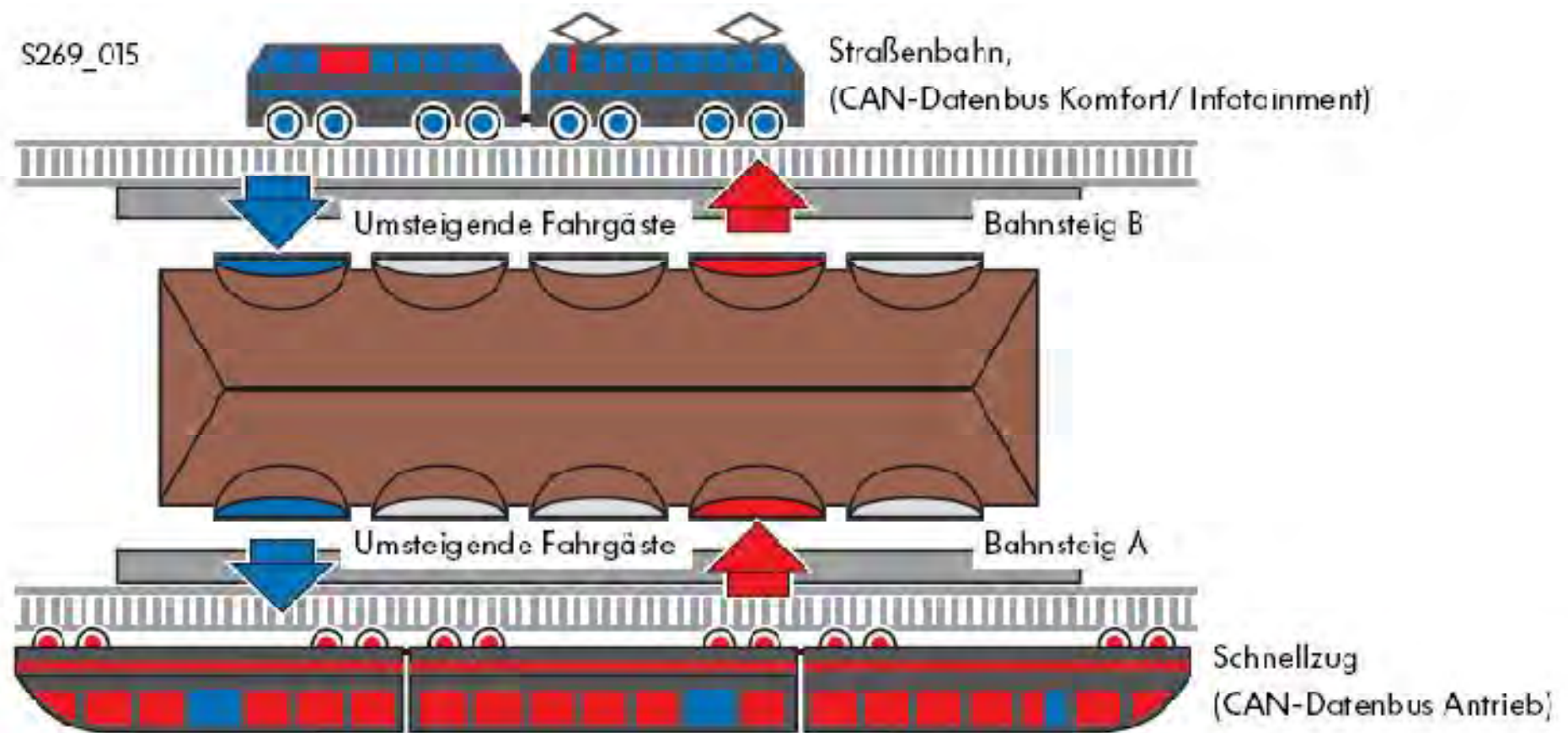
# CAN Datenverkehr



# Diagnose-Interface (Gateway-Modul)

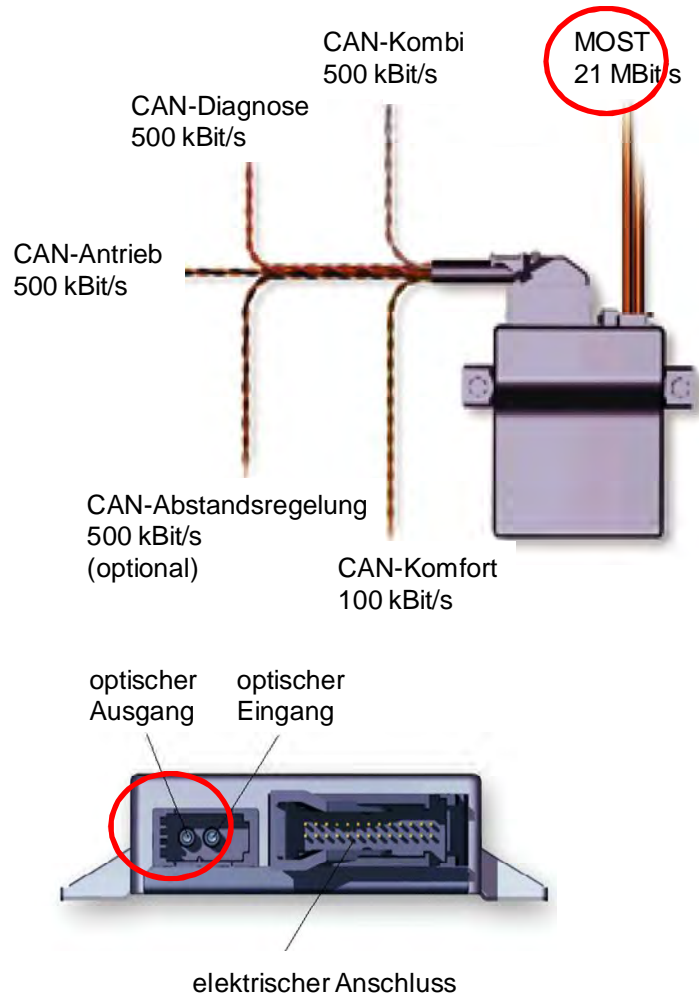


Das Prinzip eines Gateways lässt sich am Beispiel eines Bahnhofs verdeutlichen



# Diagnoseinterface für Datenbus

## Aufgabe/Funktion



## Aufgabe

Es stellt die Schnittstelle zwischen den verschiedenen Bussystemen im Fahrzeug dar, d.h. es ermöglicht eine Kommunikation zwischen den Steuergeräten in den verschiedenen Bussystemen.

## Masterfunktionen

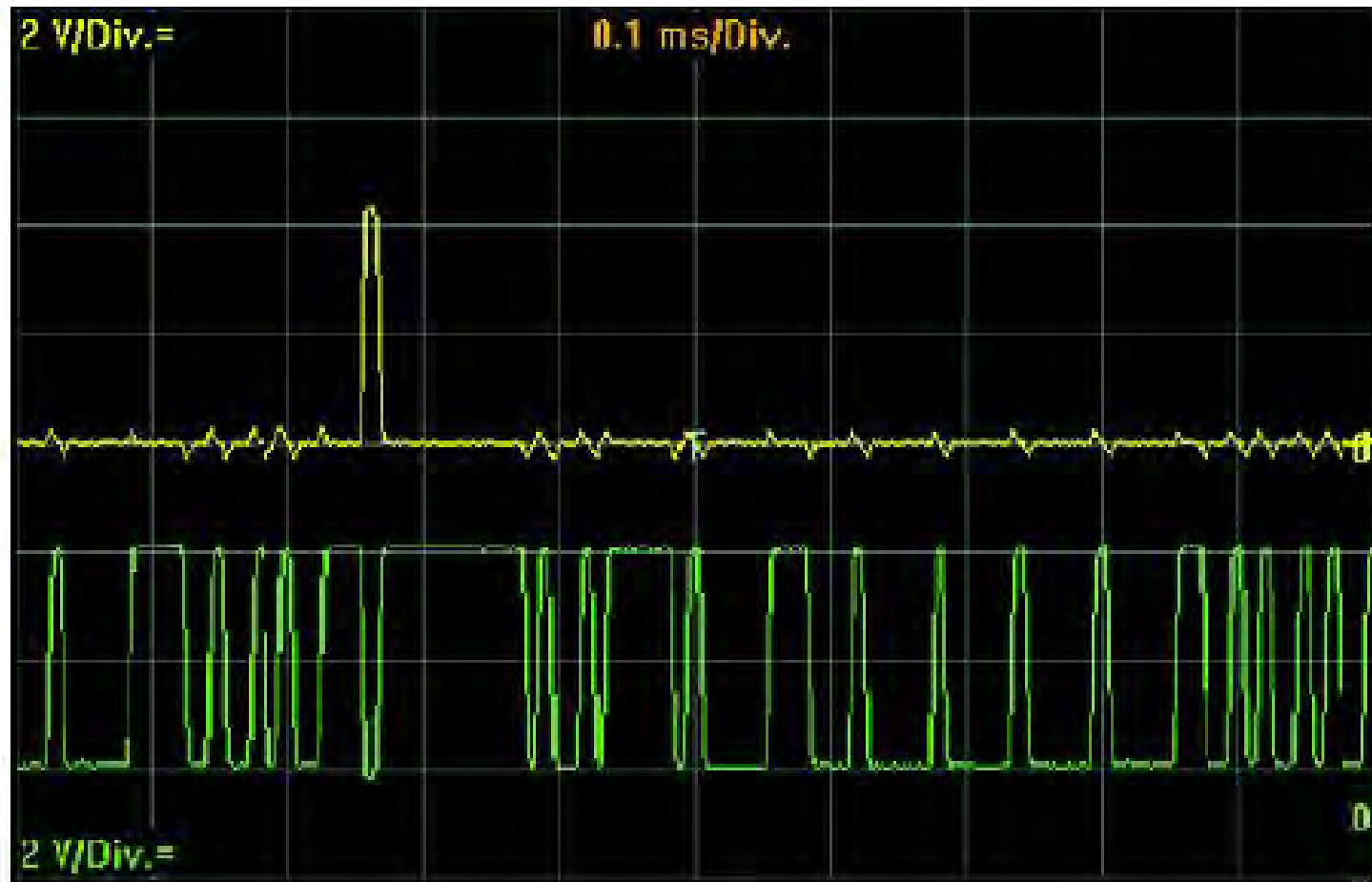
- MOST Ringbruch-Diagnose
- Nachlauf CAN-Antrieb
- Sleep- und Wake-up

## Diagnose CAN-Bus

ISO-Fehlertabelle

ISO	CAN-High	CAN-Low
1		Unterbrechung
2	Unterbrechung	
3		Schluss nach $V_{\text{Batt}}$
4	Schluss nach Masse	
5		Schluss nach Masse
6	Schluss nach $V_{\text{Batt}}$	
7	Schluss nach CAN-Low	Schluss nach CAN-High
8	fehlender $R_{\text{term}}$	fehlender $R_{\text{term}}$

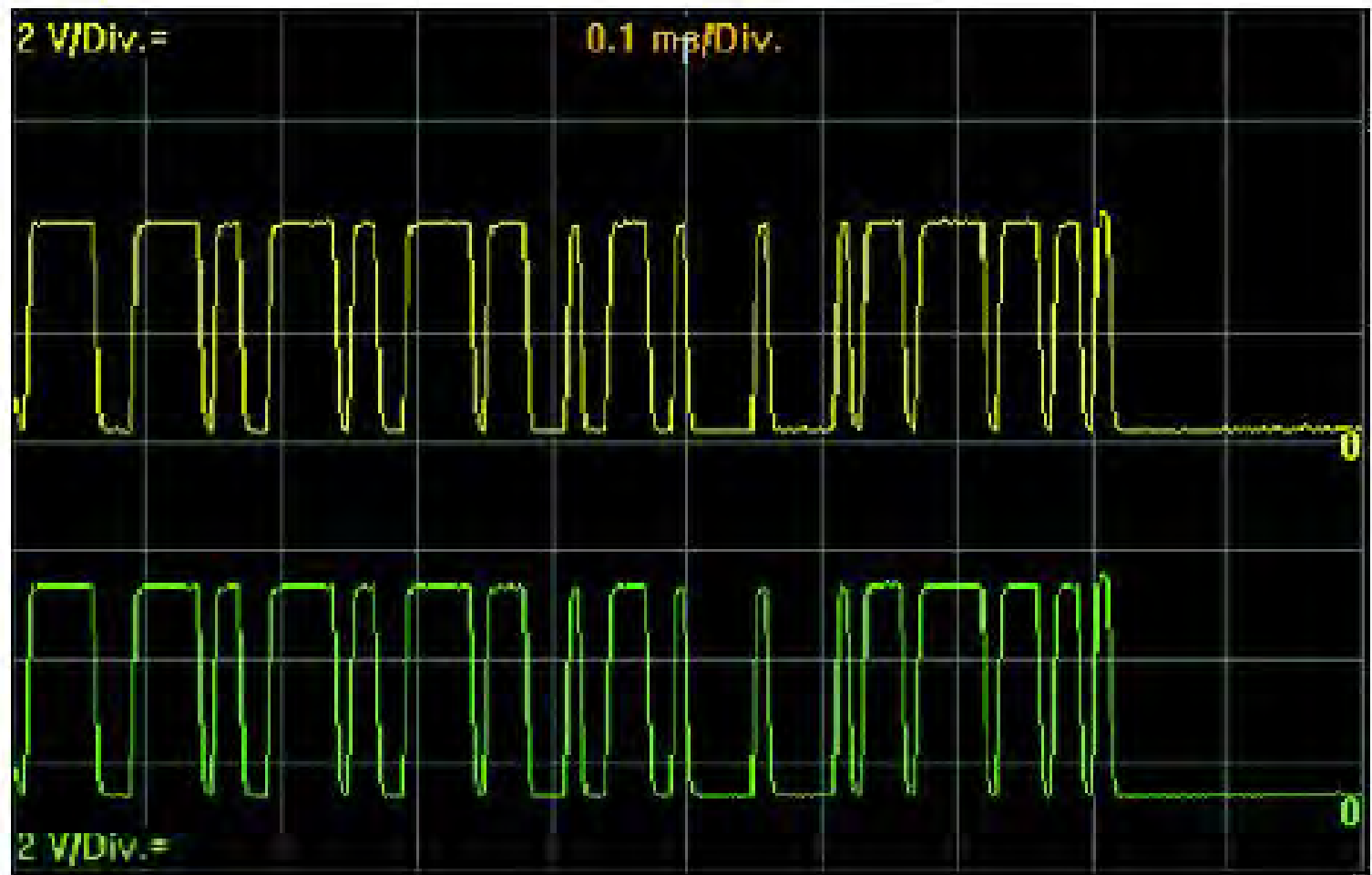
# Darstellung des Signalverlaufs auf dem DSO bei Eindrahtbetrieb (Standbild)



S269\_014

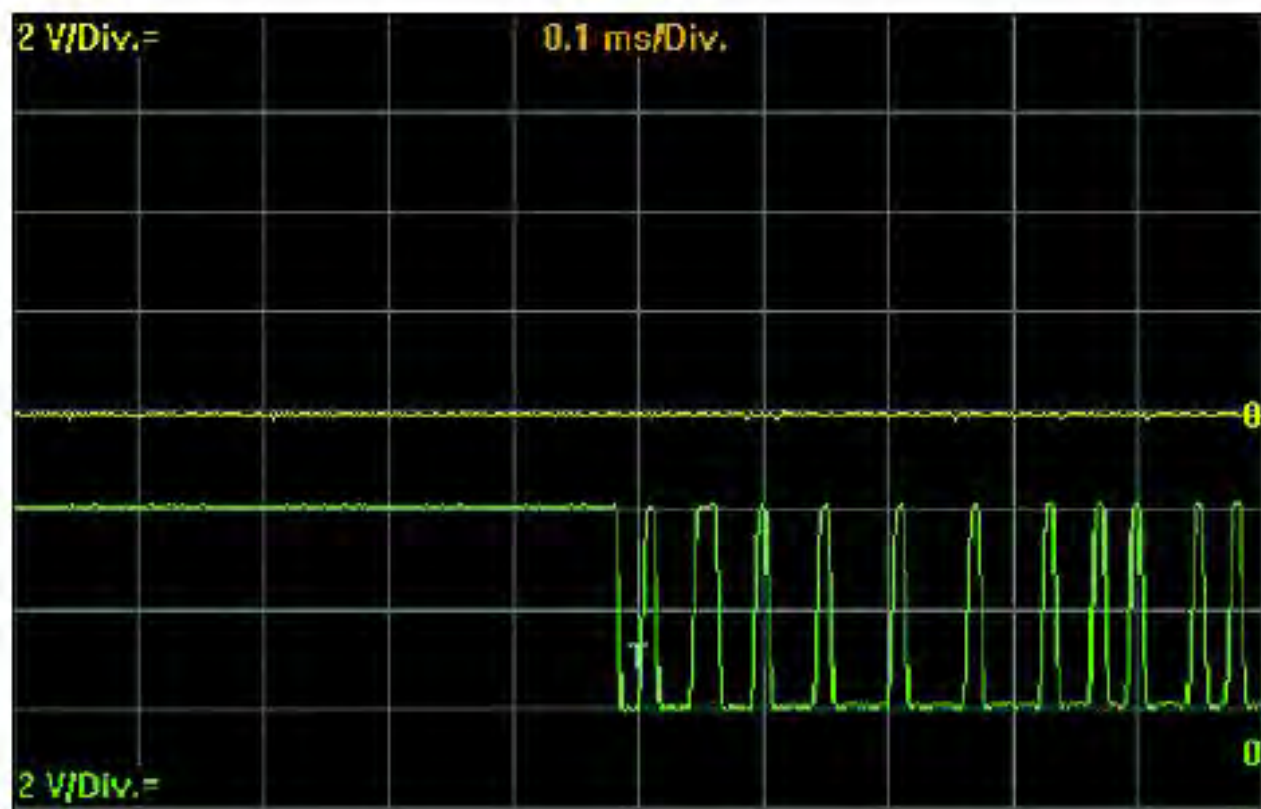


# DSO-Darstellung: Schluss CAN-High-Leitung gegen CAN-Low-Leitung

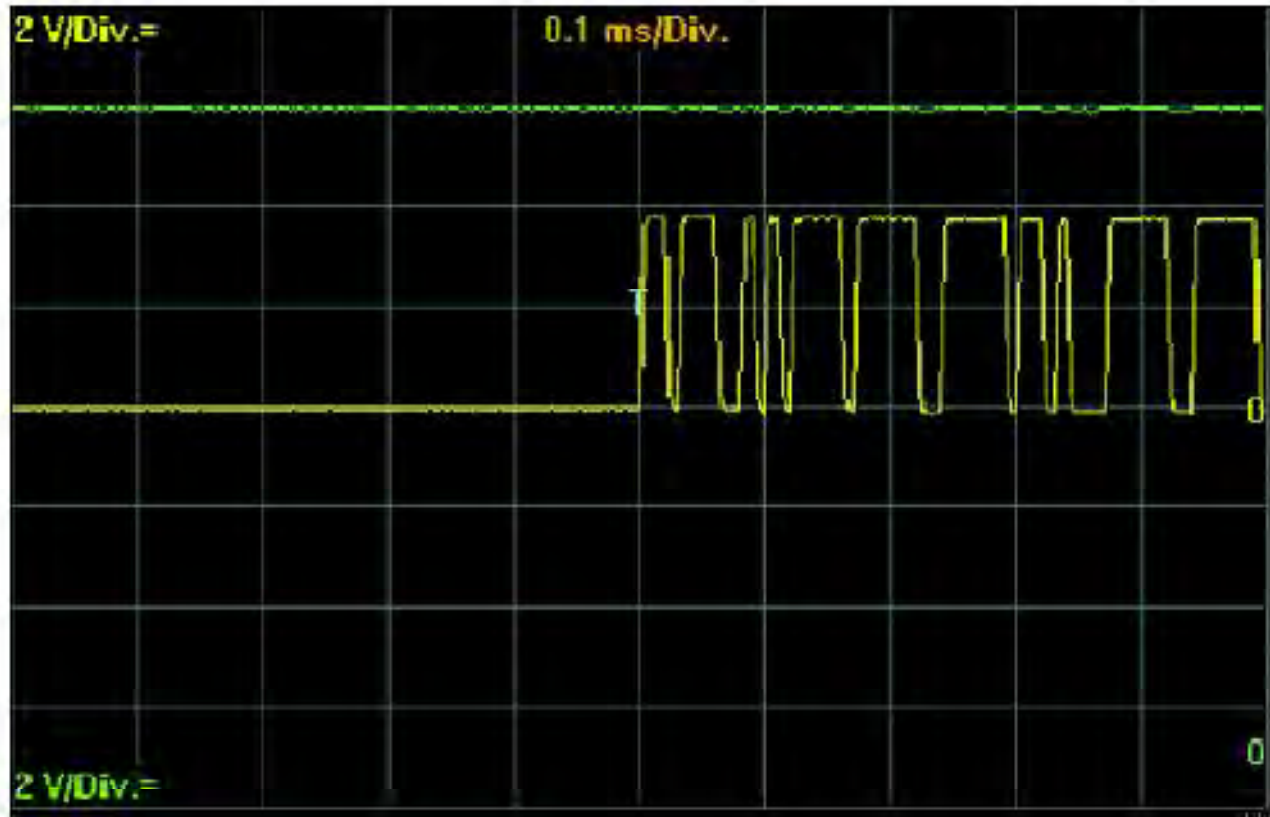


S269\_041

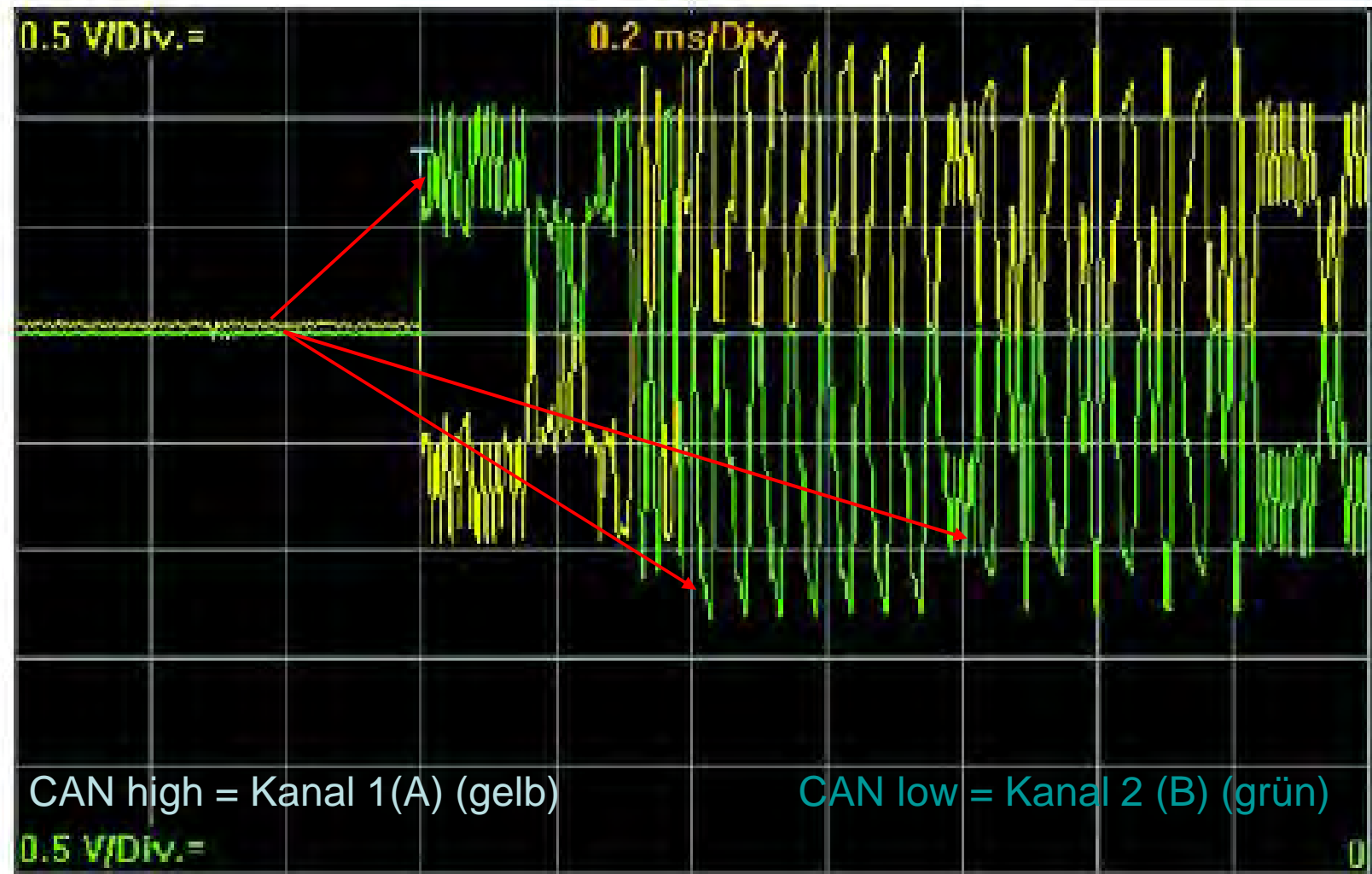
## DSO-Darstellung: CAN-Low-Signal bei Kurzschluss des CAN-High-Signals gegen Masse



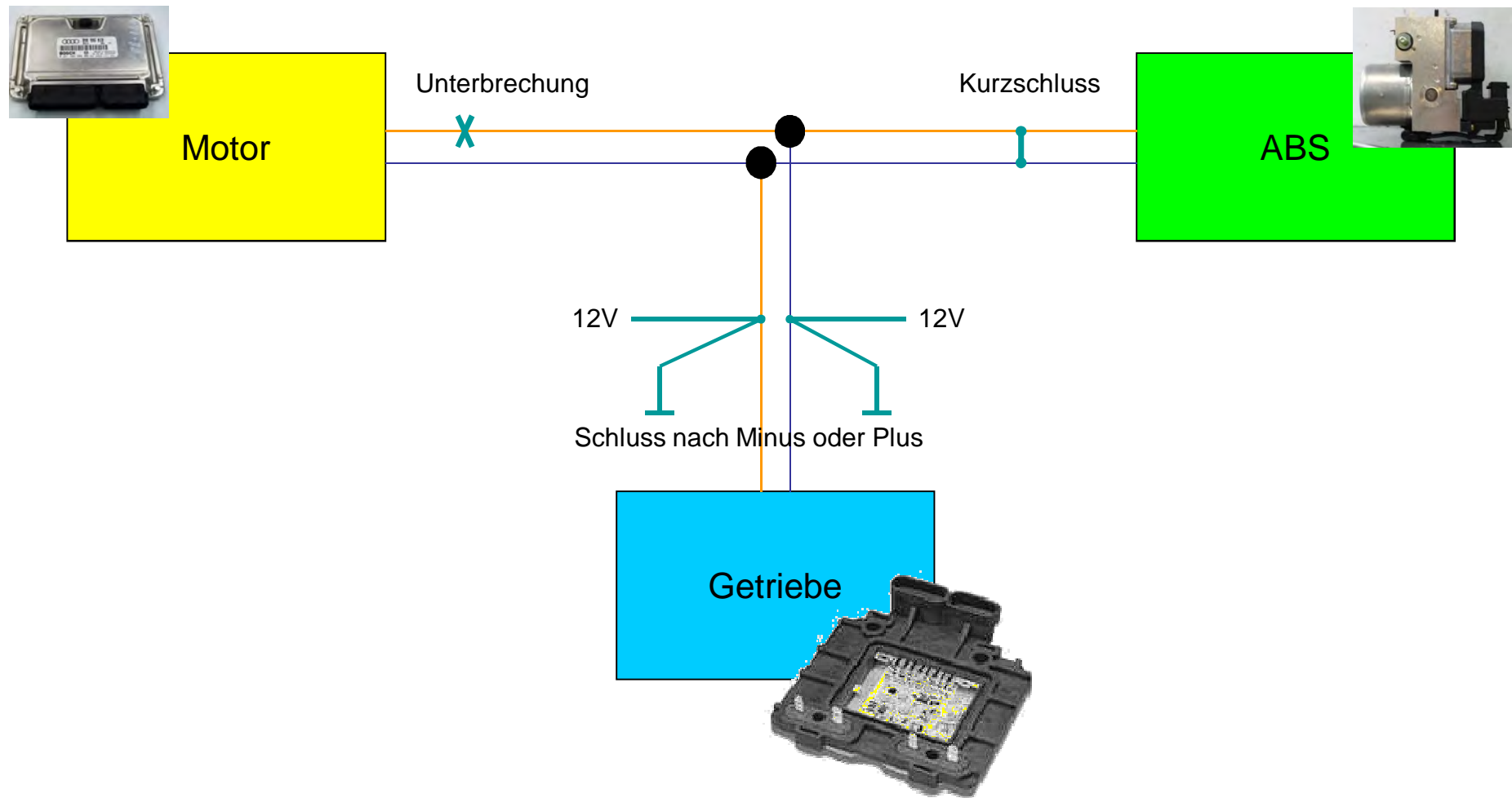
S269\_039



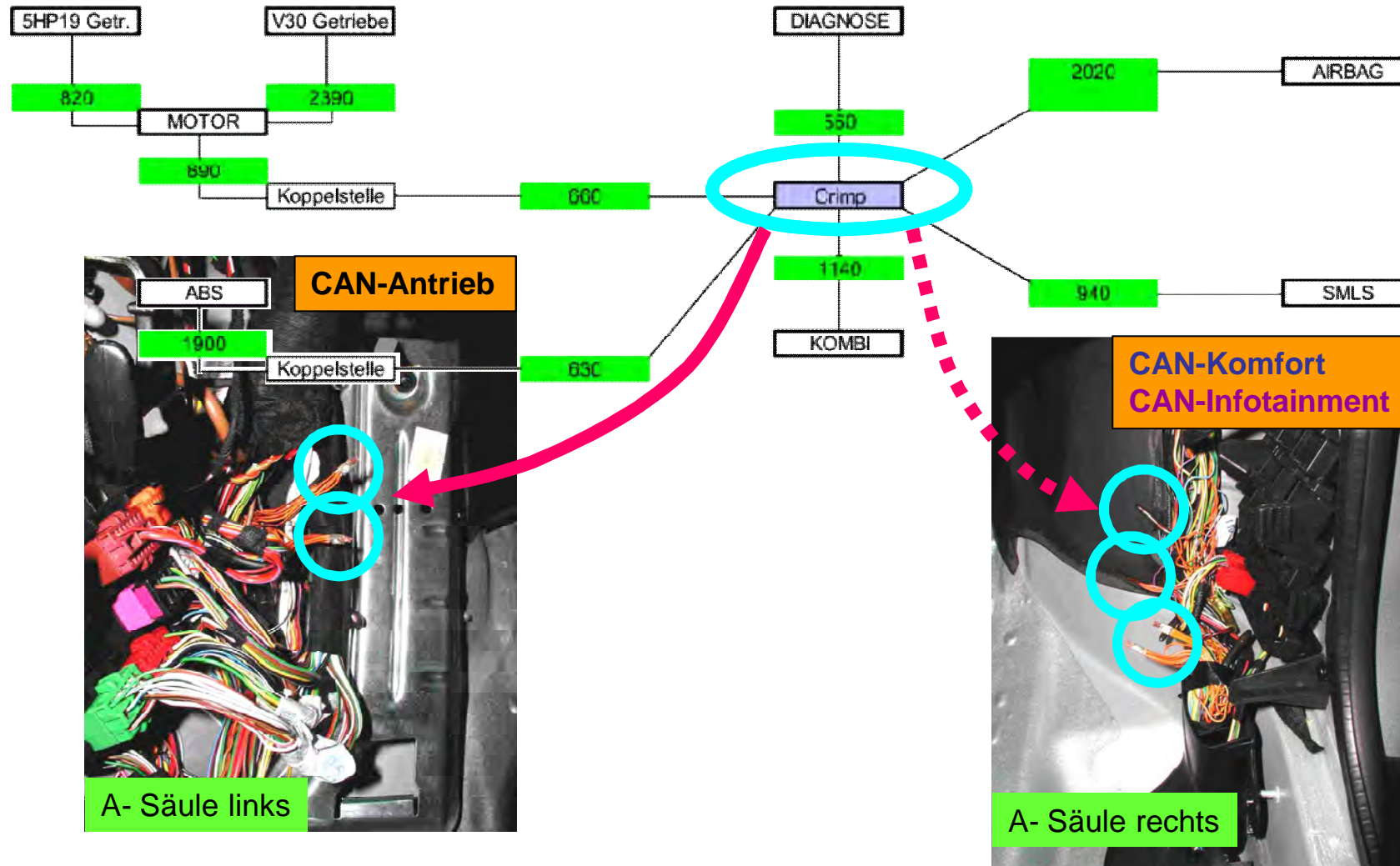
## DSO-Darstellung: Vertauschung CAN-High und CAN-Low



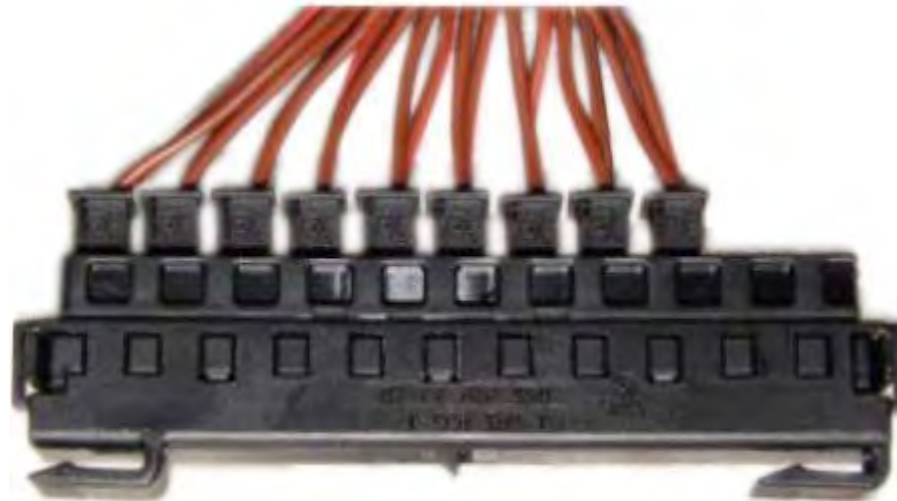
# Elektrische Fehlermöglichkeiten im Fahrzeug




# Zentrale CAN-Knoten Beispiel B6

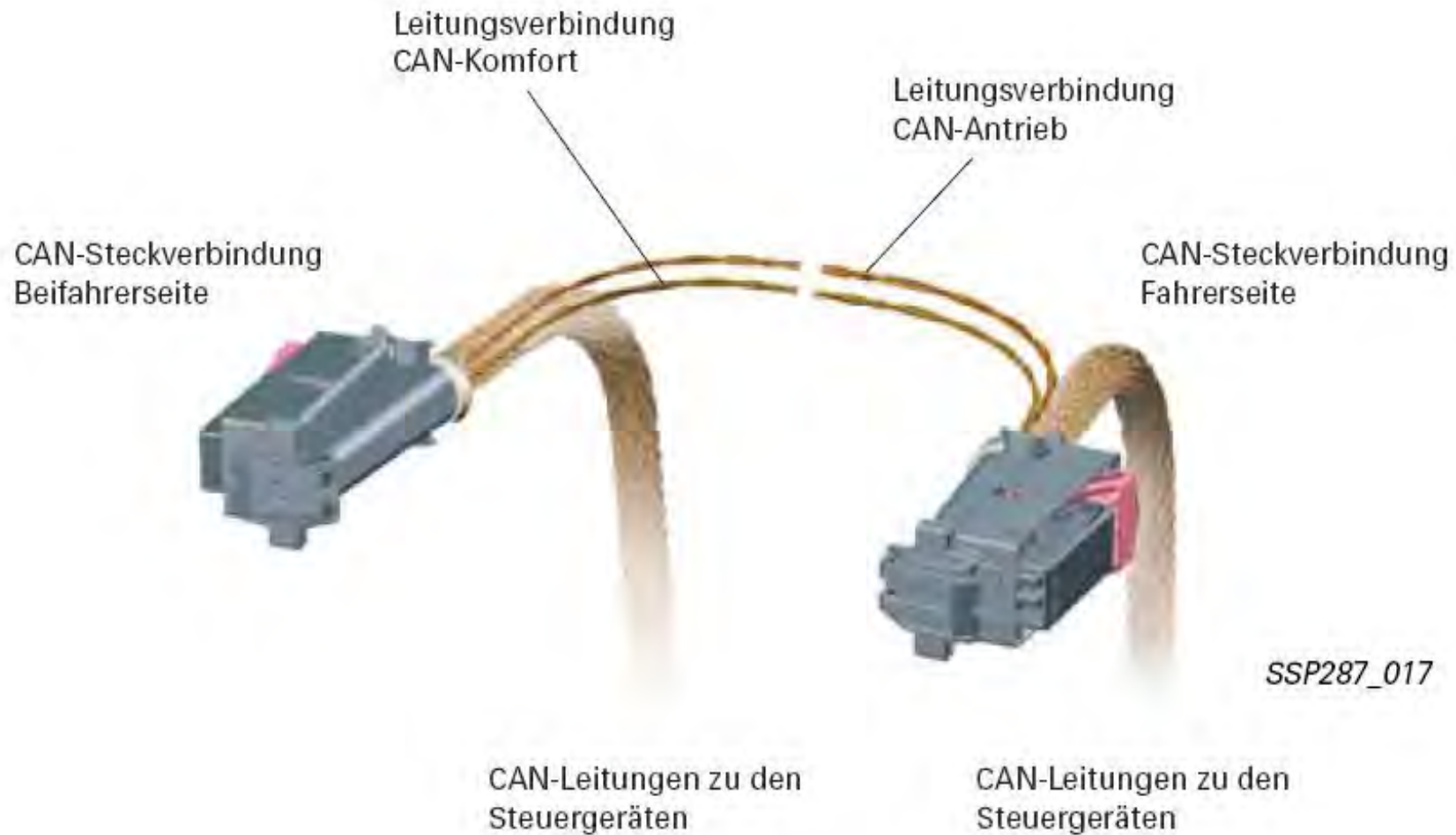


# ***CAN-Bus Verteilerleiste / Potentialverteile (z.B. Mercedes-Benz)***



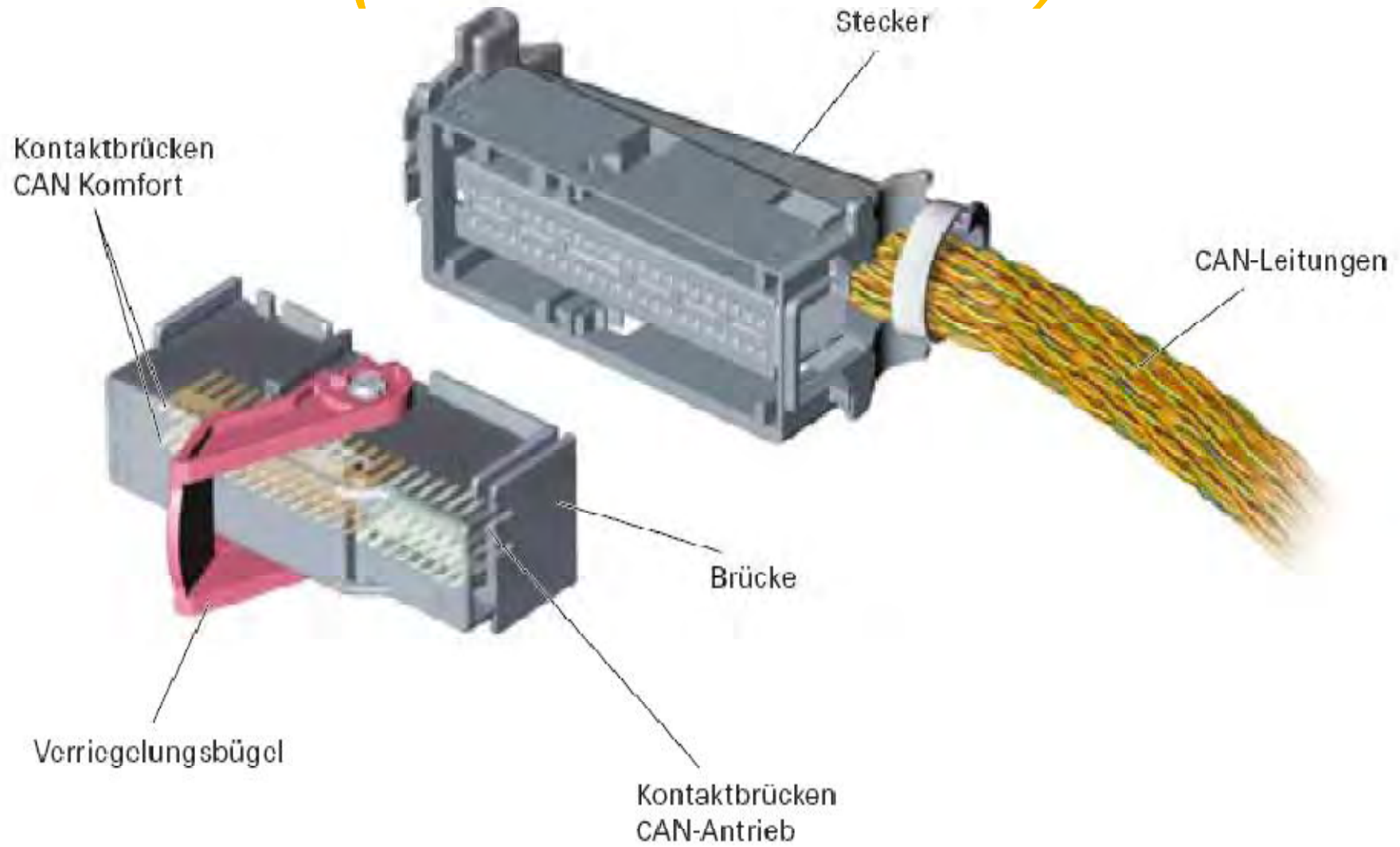
 kfz-tech.de

# ***CAN-Bus Diagnosestecker (z.B. Audi A8 ab 2002)***

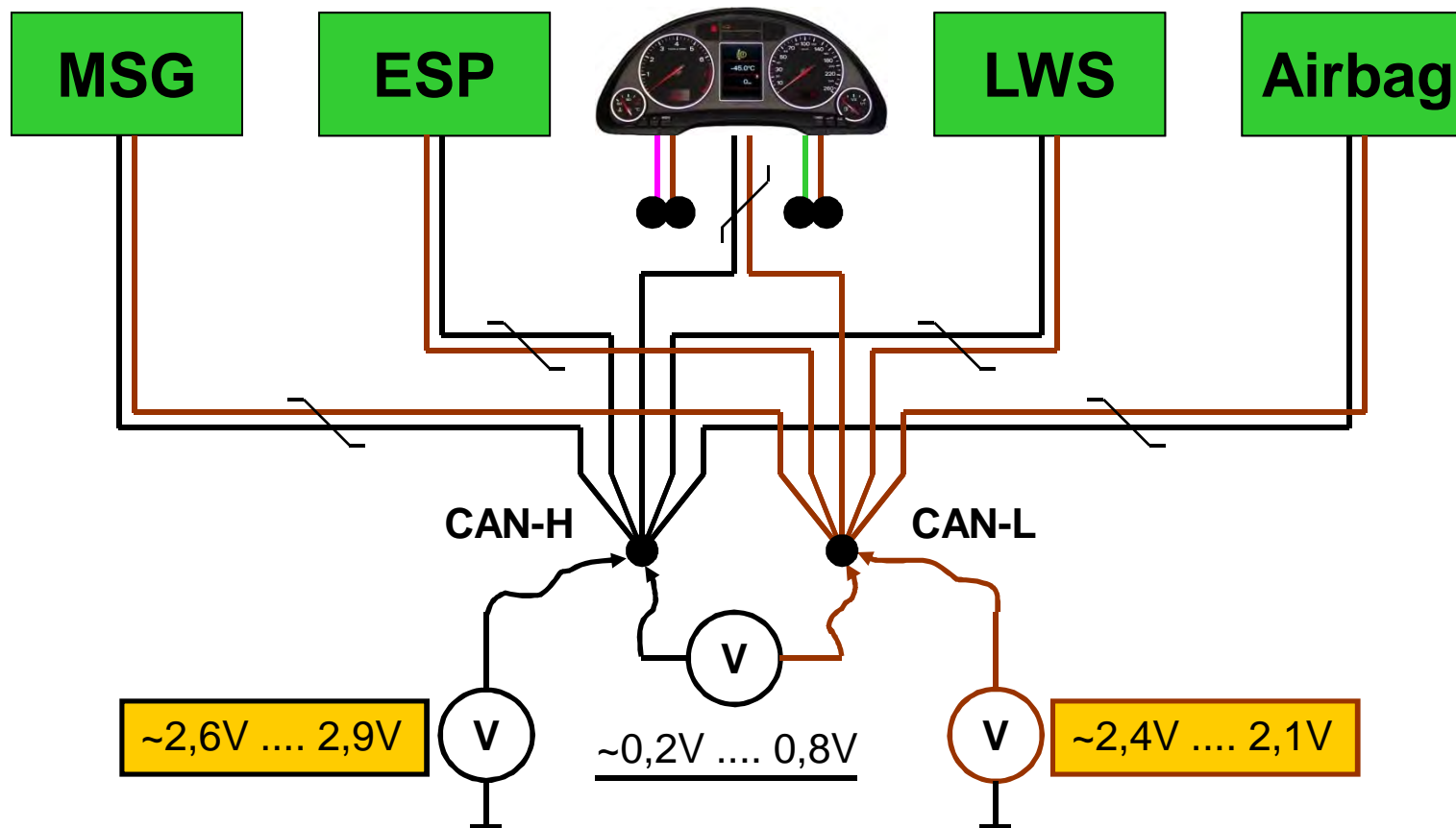




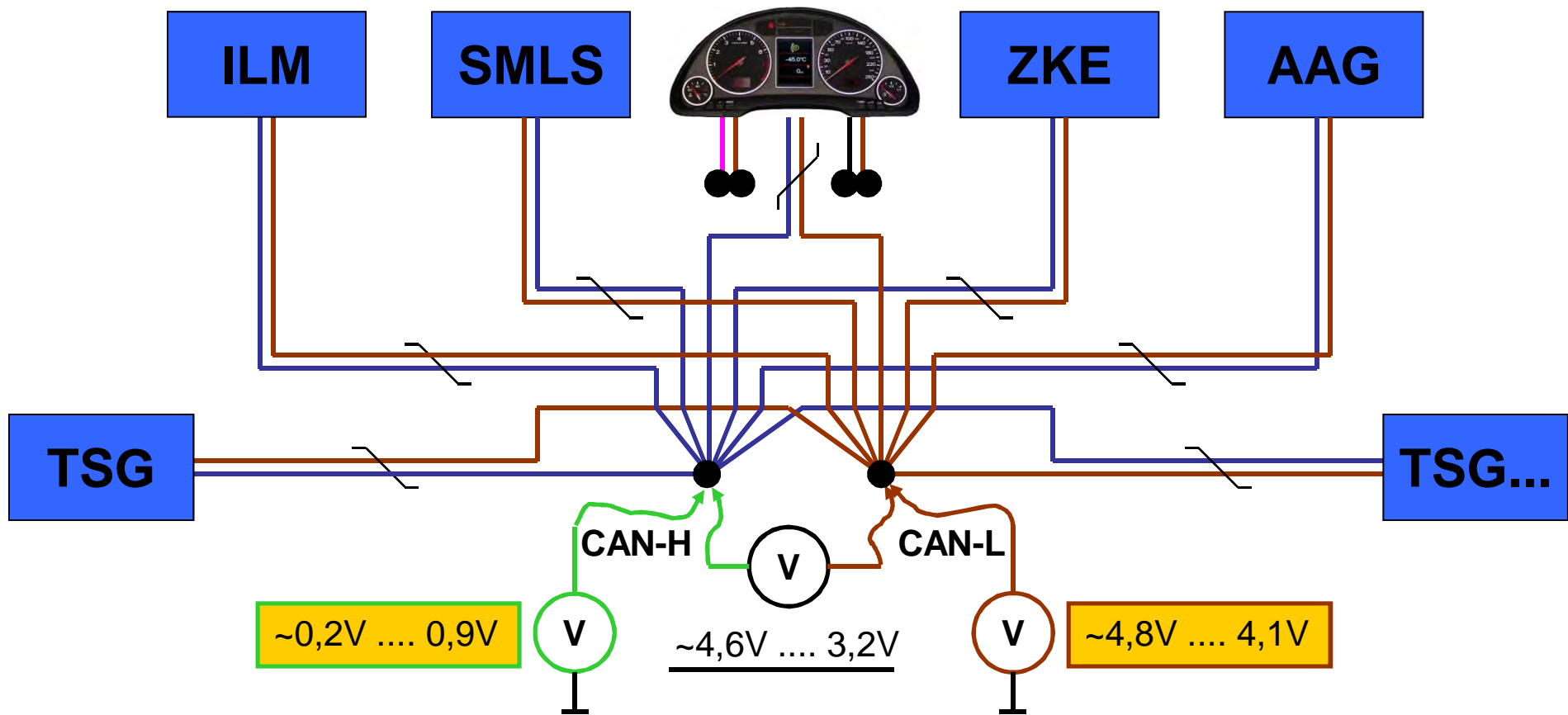
# ***CAN-Bus Diagnosestecker (z.B. Audi A8 ab 2002)***



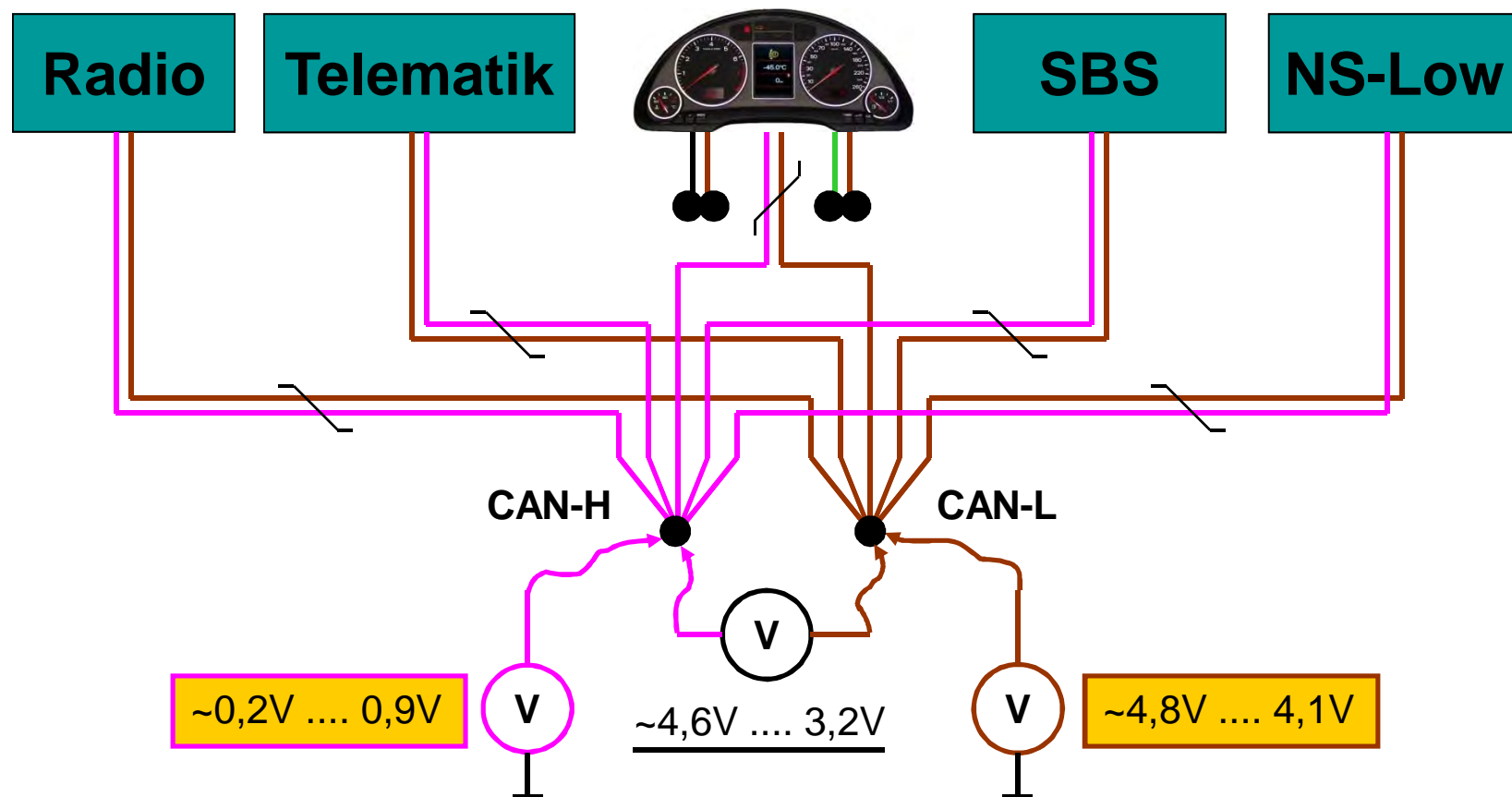
# Messtechnik Antriebs-CAN



# Messtechnik Komfort-CAN



# Messtechnik Infotainment-CAN



# Gateway

## Sleep-, Wakeup-Modus

- Masterfunktion CAN-Antrieb Nachlauf
- passive Steuerung vom Sleepmode, Synchronisation der Bussysteme. => nur wenn alle Stg. Sleepbereit sind.

## Diagnose

- zentrale Überwachung der diagnosefähigen Steuergeräte, auf Kommunikation.
- Vergleich von Soll-, u. Istverbau der Steuergeräte (Codierung)
- Unterstützung der Online-Gateway-Codierhilfe im Feld.
- Zentrale Sammelstelle der Kundendienstfehler.
- Master der MOST-Ringbruchdiagnose.
- Master über Diagnosefehler, Umgebungsdaten wie Verlernzähler, Datum Uhrzeit (=> der vorgekommenen Fehler).
- Masterfunktion „Transportmode“ - TV, ZV-FB, Radio etc. wird abgeschaltet (bisher durch ziehen der Sicherungen). Wird nach 50 km Fahrbetrieb deaktiviert und kann nicht mehr in den Transportmode versetzt werden (außer Werksintern, durch eine bestimmte Codierung).

# Eventgesteuert



**eventgesteuert**

# Zeitgesteuert



**zeitgesteuert**

# Eventgesteuert



## eventgesteuert

Medium wird nur bei Bedarf genutzt

Zeitpunkt zu dem Medium genutzt werden kann hängt von Auslastung ab

Ankunftszeit unbekannt

Eventuell ist Medium überlastet

# Zeitgesteuert



## zeitgesteuert

Zeitpunkt zu dem Medium genutzt werden kann ist festgelegt.

Dauer der Nutzung des Mediums festgelegt

Ankunftszeit bekannt

# ***10.TT-CAN***

***TIME-TRIGGERED – CAN***





# TT-CAN

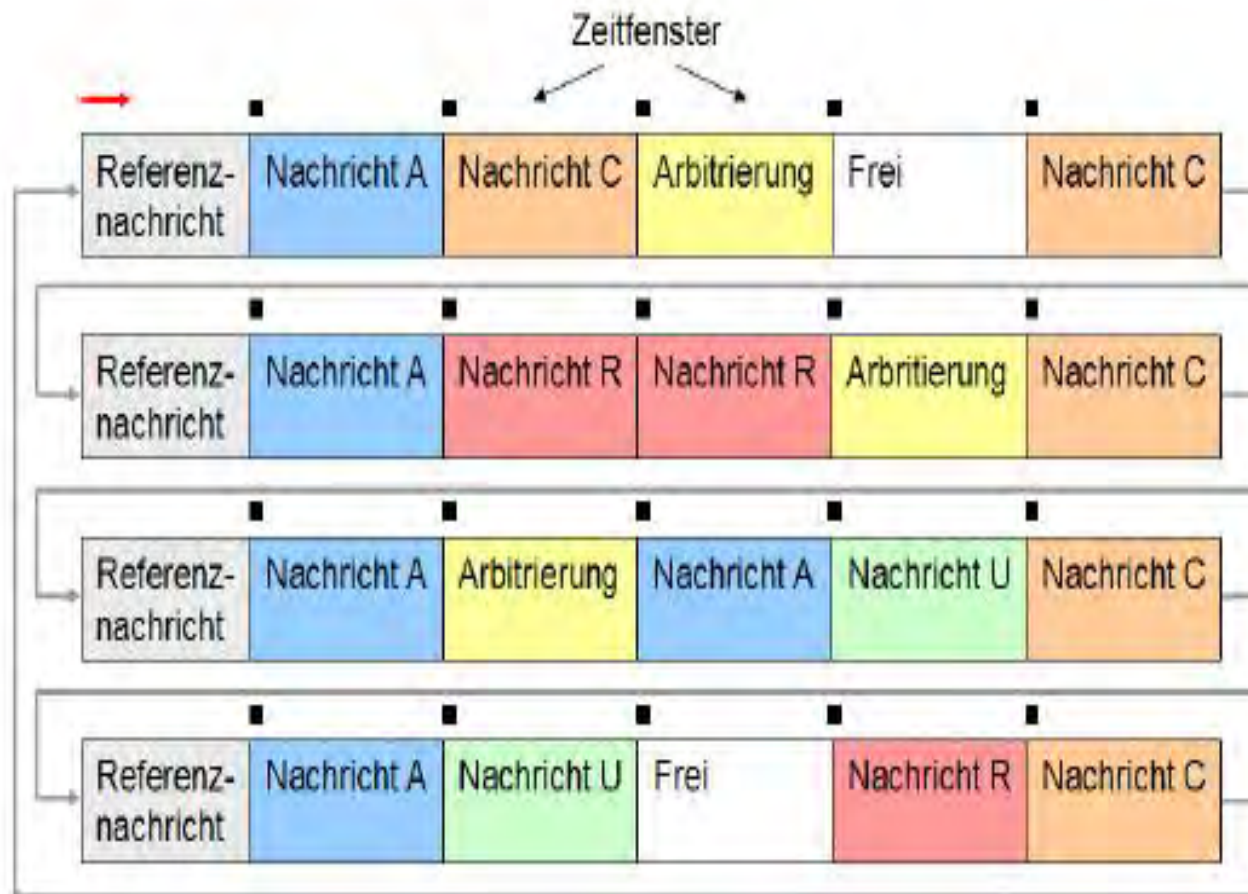
**TT-CAN:** Dieser arbeitet im Grundsatz wie der CAN-Bus.

Ergänzt wird der TT-CAN durch einen Zeitmaster, deswegen TIME TRIGGERED!

Die im CAN-Bus verwendete ereignisgesteuerte Kommunikation verhält sich im Worst-Case-Fall nicht deterministisch. Das heißt, dass über die Sendezeit einer Nachricht keine Aussage getroffen werden kann.

In sicherheitsrelevanten Systemen verwendet man daher den zeitgesteuerten Ansatz. Dabei wird jede Nachricht in einen eigenen Zeitfenster versandt.

# TT-CAN



# TT-CAN

Um Nachrichten zu einem bestimmten Zeitpunkt versenden zu können, muss sichergestellt sein, dass alle Knoten dieselbe Zeit haben. Daher werden in jeder Zeile der Systemmatrix Referenznachrichten versandt, die die Zeit der Knoten auf 0 setzt.

Das Versenden der Referenznachrichten übernimmt ein Zeitmaster. Da bei dessen Ausfall die gesamte Kommunikation gestört ist, werden häufig mehrere Zeitmaster verwendet. In den jeweiligen Knoten werden nur die für sie relevanten Daten der Systemmatrix, also die Sende- Zeitpunkte gespeichert.

Zudem können freie Zeitfenster spezifiziert werden, wodurch eine flexible Erweiterung des Systemes in gewissen Rahmen ermöglicht wird. Ebenso kann in den zeitgesteuerten Ansatz der ereignisgesteuerte Ansatz integriert werden. In diesen Zeitfenstern findet dann wieder die bitweise Arbitrierung statt.

# *11. Byteflight*

***Flexible Time Division  
Multiple Access – FTDMA***

## *Was ist Byteflight?*

Wegen des deterministischen Verhaltens und der hohen Datenrate eignet sich das byteflight Protokoll für Anwendungen aus dem Bereich der passiven Sicherheit.

Dank der hohen Flexibilität aber auch für Karosserie- und Komfortfunktionen. Das Protokoll kann auch in aktiven Sicherheitsanwendungen eingesetzt werden.

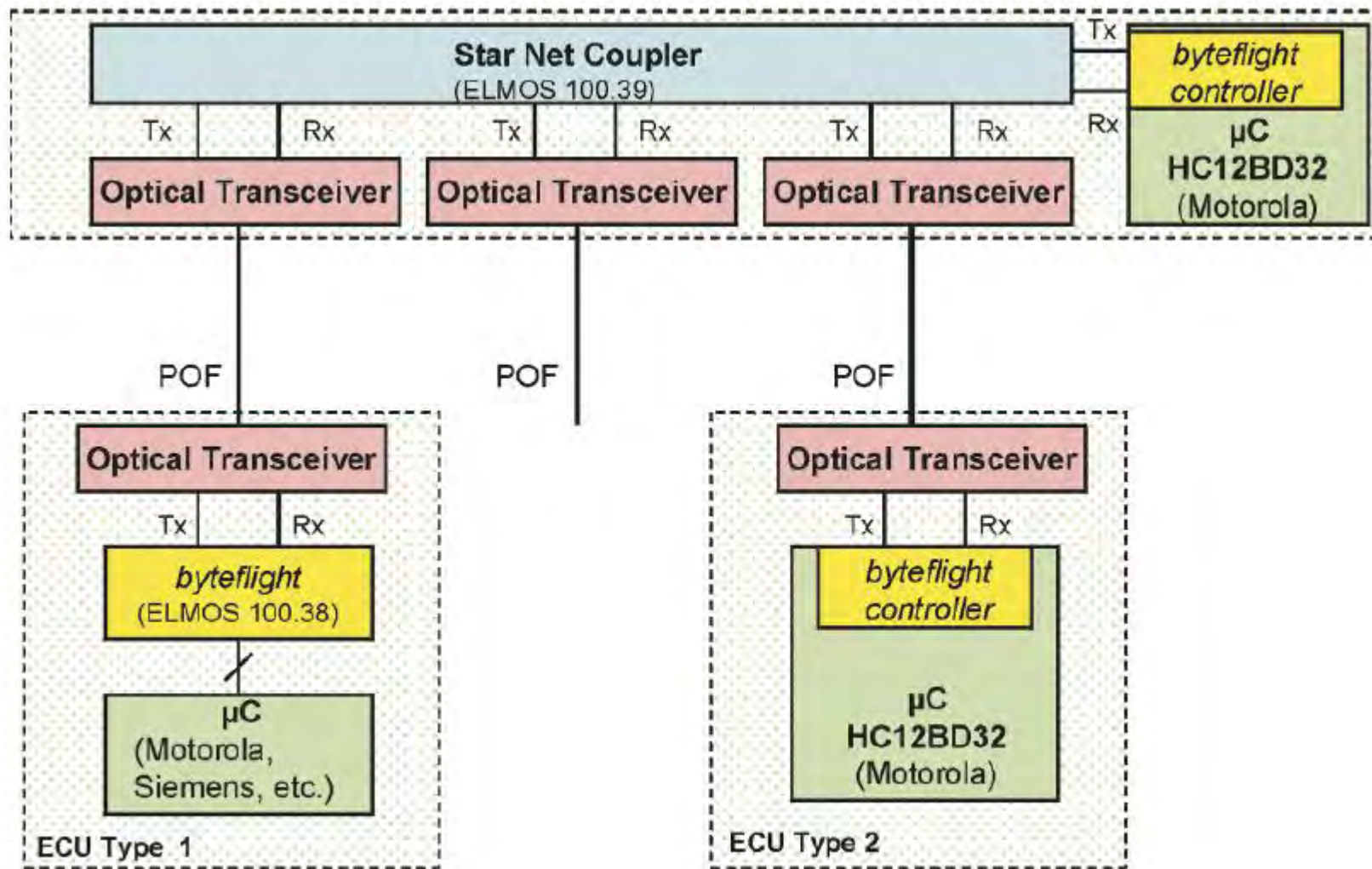
**byteflight** ist ein von [BMW](#) in Zusammenarbeit mit [Motorola](#), [Elmos](#) und [Infineon](#) entwickeltes [Bussystem](#) für sicherheitskritische Anwendungen in der Automotive-Technik. Das Byteflight-System wurde bis 2007 in 7er-, 6er- und 5er-BMWs eingebaut, um zeitkritische Daten des [Airbag](#)-Systems sowie der [Karosserie](#)-Elektronik und des [Chassis](#) zu übertragen. Mit der Modellüberholung der 5er- und 6er-Baureihen (E6x) 2007 wurde ByteFlight wieder aus dem Fahrzeug entfernt und durch einen [Time-Triggered CAN](#) ersetzt. In der künftigen 7er-Baureihe (F01) wird [FlexRay](#) byteflight ersetzen.

Byteflight hat eine [Datenrate](#) von 10 Mbit/s und benutzt als Übertragungsmedium optische [Polymerfasern](#). Byteflight ist ein Netzwerk in [Sterntopologie](#) mit einem intelligenten Sternkoppler.

Neben der erwähnten [Automobiltechnik](#) kann Byteflight in allen [Echtzeitanwendungen](#) mit extrem kurzen [Latenzzeiten](#) und hohem [Datendurchsatz](#) eingesetzt werden und ebenso in Umgebungen mit starker Störbeeinträchtigung.

Wie der [CAN-Bus](#) arbeitet Byteflight auch mit einem nachrichtenorientierten Übertragungsprotokoll: alle Nachrichten werden allen Byteflight-Stationen zur Verfügung gestellt. Die [Datenpakete](#) (Frames) von Byteflight ähneln den CAN-Datenpaketen, die maximale Länge des Datenfeldes liegt bei 12 Byte. Um ein vorhersagbares [Echtzeitverhalten](#) zu erreichen, arbeitet Byteflight mit dem [deterministischen](#) Zugangsverfahren [TDMA](#), bei dem jede angeschlossene Busstation innerhalb einer bestimmten Zeit eine definierte Zeitspanne (Timeslot) für die Datenübertragung erhält.

# Aufbau Byteflight!



# Datenrahmen Byteflight

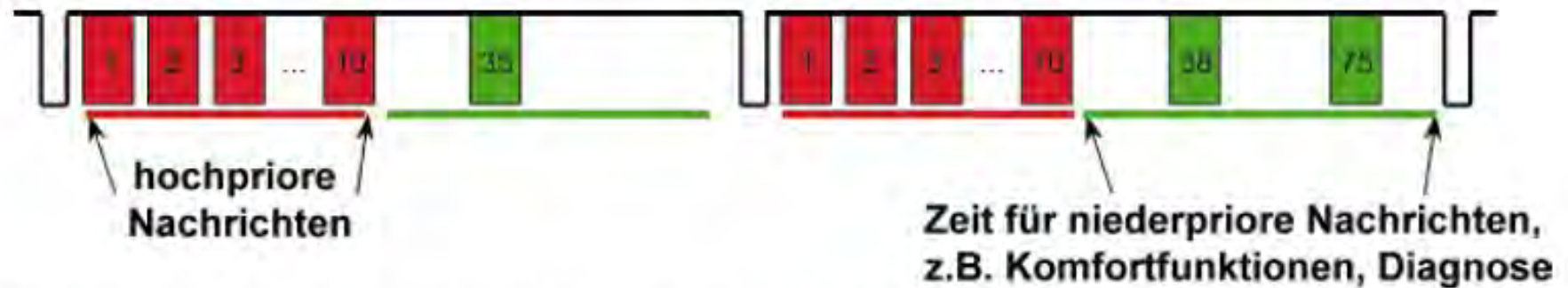
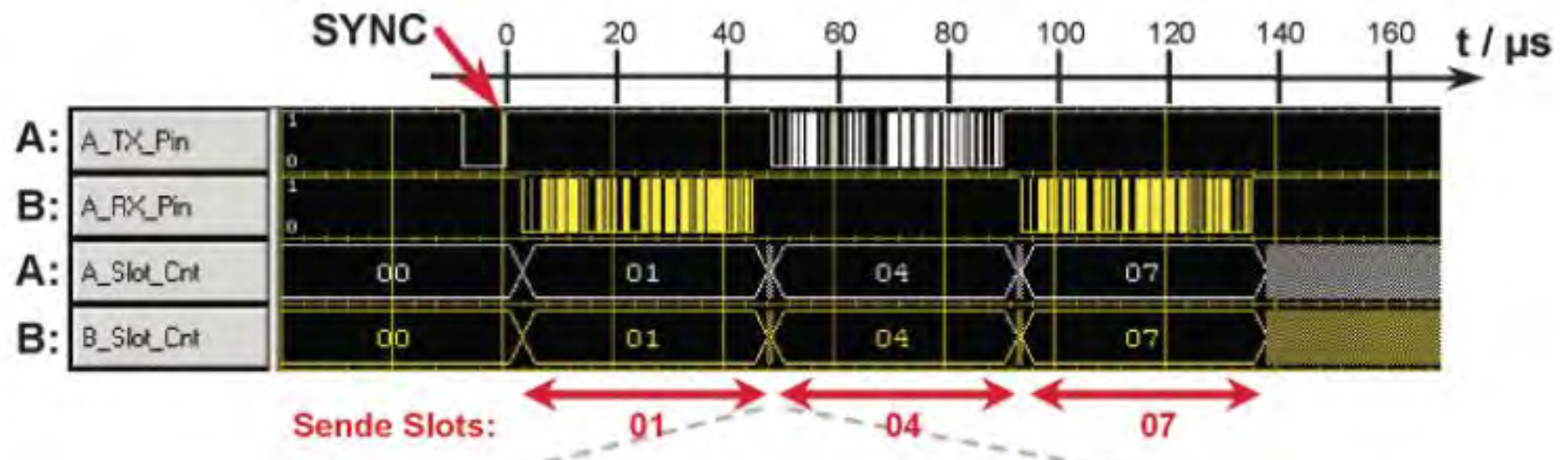
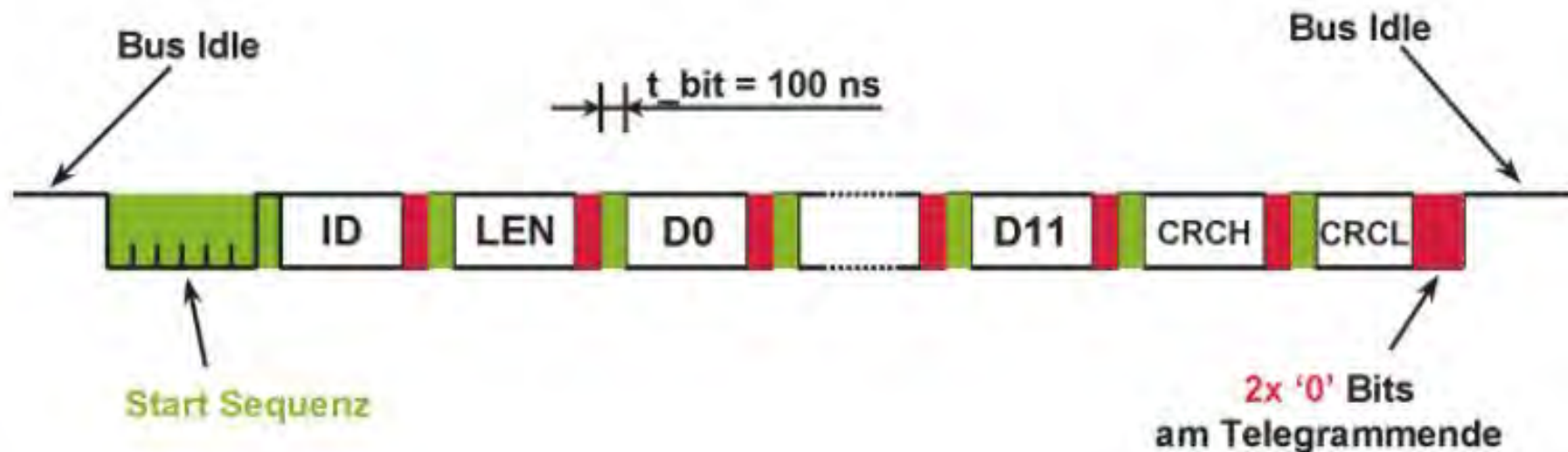


Bild 3: Synchrone und asynchrone Nutzung der Bandbreite

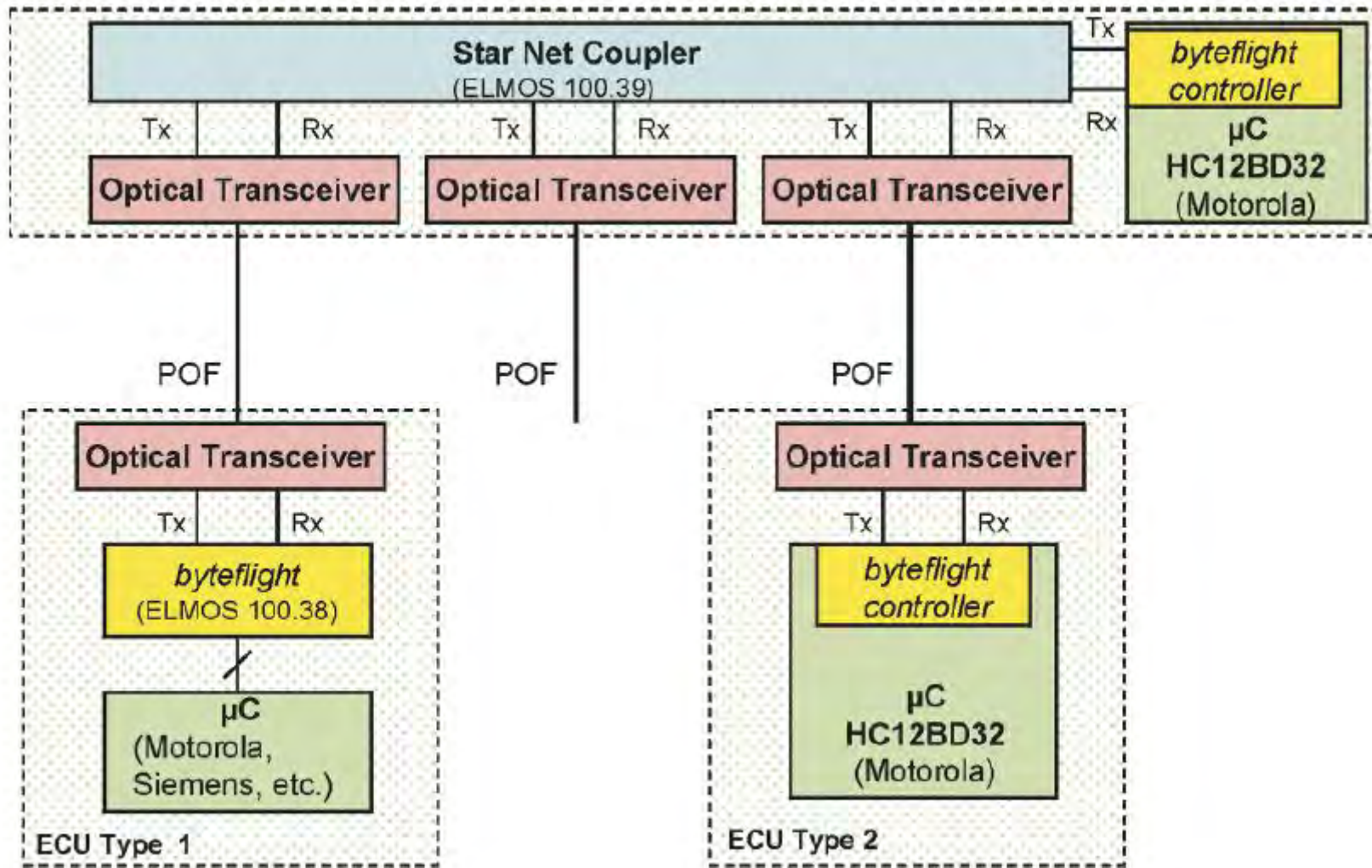






**Telegrammaufbau:** byteflight Nachrichten, **bestehen** aus einer 6 Bit Startsequenz, einem Identifier- und einem Längenbyte, bis zu 12 Datenbytes und zwei CRC (Cyclic Redundancy Check)-Bytes.

Zur Bitsynchronisation ist jedes Byte von je einem Start- und Stop-Bit begrenzt. Die Bitdauer beträgt 100ns bei einer Bruttodatenrate von 10 Mbit/s.





## 12. Flex Ray



459\_025



## **Einführung**

Im Audi A8 '10 setzt mit dem FlexRay ein neues Datenbussystem ein. Was genau ist FlexRay? Das FlexRay-Konsortium, eine Hersteller übergreifende Organisation zur Entwicklung, wurde im Jahr 2000 gegründet und seitdem um weitere Mitglieder erweitert, darunter auch Volkswagen.

Wofür steht FlexRay?

**Flex = Flexibilität**

**Ray = Rochen (im Logo des FlexRay-Konsortiums)**

Ziel des Einsatzes von FlexRay ist es, die erhöhten Anforderungen zukünftiger Vernetzung im Fahrzeug zu erfüllen, insbesondere höhere Datenübertragungsraten, Echtzeitfähigkeit und Ausfallsicherheit.

Er erweitert damit die Einsatzmöglichkeiten z. B. bei der Fahrdynamikregelung, der Abstandsregelung ACC und der Bildverarbeitung.

## **Merkmale**

Der FlexRay im Audi A8 '10 weist folgende Merkmale auf:

- elektrisches Zweidraht-Bussystem
- Datenübertragungsrate: maximal 10 Mbit/s
- Datenübertragung mit drei Signalzuständen
- „Idle“
- „Data 0“
- „Data 1“
- Topologie eines „aktiven“ Sterns
- Echtzeitfähigkeit
- ermöglicht verteilte Regelungen und den Einsatz in sicherheitsrelevanten Systemen

## Grundprinzip

Von den bisher eingesetzten Datenbussystemen, wie CAN, LIN und MOST unterscheidet sich der FlexRay schon durch seine grundsätzliche Arbeitsweise.

Diese lässt sich recht gut mit einer Seilbahn vergleichen. Die Stationen sind dabei die Busteilnehmer, also Sender und Empfänger (Steuergeräte). Die Gondeln der Seilbahn stehen für die Botschaftsrahmen und die Passagiere für die Botschaften.



Der Zeitpunkt, zu dem ein Busteilnehmer Botschaften über den FlexRay senden kann, ist genau festgelegt. Auch die Ankunftszeit einer gesendeten Botschaft beim Empfänger ist genau bekannt. Dies entspricht dem festen „Fahrplan“ einer Seilbahn. Auch wenn ein Busteilnehmer gerade nicht sendet, ist für ihn eine bestimmte Bandbreite reserviert.

Die Seilbahn fährt, egal ob Passagiere an Bord sind. Dadurch ist eine Priorisierung von Botschaften, wie z. B. beim CAN-Bus nicht erforderlich.

Bei Audi würde eine „leere Gondel“ als Fehler des Senders erkannt, d. h. es werden von den Steuergeräten immer Daten gesendet. Neue Inhalte werden über ein so genanntes „Update-Bit“ gekennzeichnet. Stehen keine neuen Daten zur Verfügung, werden die alten Daten erneut versendet

.



# CAN-Bus und FlexRay im Vergleich

Eigenschaft	CAN-Datenbus	FlexRay
-------------	--------------	---------




<b>Verdrahtung</b>	elektrisch, Zweidraht	elektrisch, Zweidraht
<b>Signalzustände</b>	„0“ – dominant, „1“ – rezessiv	„Idle“, „Data 0“, „Data 1“
<b>Datenrate</b>	500 kbit/s	10 Mbit/s
<b>Zugriffsprinzip</b>	ereignisgesteuert	zeitgesteuert
<b>Topologien</b>	Bus, passiver Stern	Aktiver Stern, Punkt zu Punkt, Daisy Chain <sup>1)</sup>
<b>Arbitrierung</b>	Nachricht mit höherer Priorität wird vor einer Nachricht mit niedrigerer Priorität gesendet	keine, Daten werden zu festgelegten Zeiten gesendet
<b>Bestätigungssignal</b>	Empfänger bestätigt den Empfang eines gültigen Datenprotokolls	Sender erhält keine Information, ob ein Datenprotokoll korrekt übertragen wurde
<b>Fehlerprotokoll</b>	Fehler kann im Netzwerk durch ein Fehlerprotokoll kenntlich gemacht werden	Jeder Empfänger prüft für sich, ob das empfangene Datenprotokoll korrekt ist
<b>Länge des Datenprotokolls</b>	maximal 8 Byte Nutzdaten	maximal 256 Byte Nutzdaten

**Nutzen**

• wird nach Bedarf genutzt

• Zeitpunkt, an dem das Datenprotokoll benötigt wird

# FlexRay - Eigenschaften

## Nachteile

### Netzwerkconfiguration

Globale Konfiguration muss in allen Knoten identisch sein.  
Hinzufügen eines Knotens erfordert u. U. Änderung des Netzwerk- in allen Knoten.

Beim Design des Netzwerkes müssen Zugriffszeiten der einzelnen Knoten bereits bekannt sein

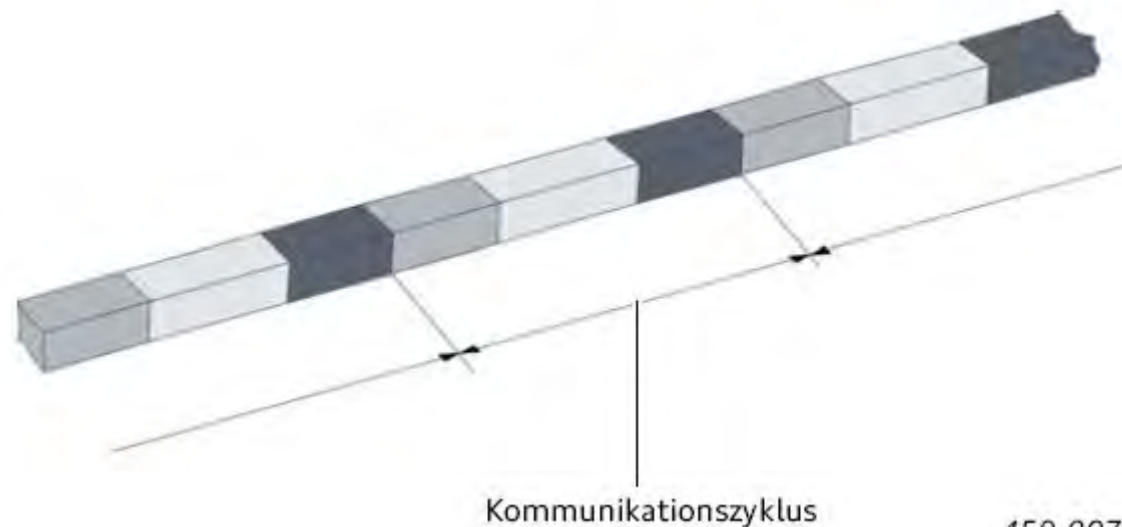
## FlexRay-Protokoll

Auf dem FlexRay werden Botschaften in so genannten Kommunikationszyklen („Communication Cycles“) transportiert. Ein Kommunikationszyklus wiederholt sich ständig, d. h. auf einen folgt unmittelbar der nächste.

Ein Kommunikationszyklus dauert 5 Millisekunden.

Er besteht aus:

- statischem Segment
- dynamischem Segment
- Network Idle Time (Leerlauf)



## Statisches Segment

Das statische Segment dient zur Übertragung der Botschaften zwischen den Busteilnehmern. Es ist in 62 statische Zeitfenster für die Datenübertragung unterteilt, so genannte „Slots“. In einem statischen Zeitfenster darf nur ein bestimmter Busteilnehmer senden.

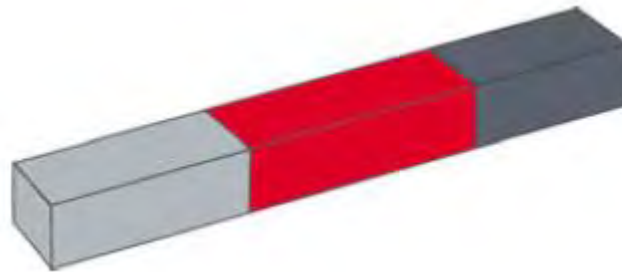
Jedoch können alle Busteilnehmer alle statischen Slots empfangen, also auch die nicht für sie bestimmten. Alle statischen Slots haben exakt die gleiche Länge von 42 Bytes. Die Reihenfolge der Slots ist fest konfiguriert. Von einem zum nächsten Communication Cycle können in den jeweiligen statischen Segmenten unterschiedliche Botschaftsinhalte übertragen werden.

Grundsätzlich wird unabhängig davon, ob alle Slots mit Botschaften „bestückt“ werden, stets die komplette Slot-Struktur übertragen. Bei Audi senden die Busteilnehmer jedoch fortlaufend Botschaften, so genannte „Update Bits“.



## Dynamisches Segment

Das in Minislots unterteilte dynamische Segment wird von allen Busteilnehmern empfangen. Das dynamische Segment ist ein reservierter Platz innerhalb eines Kommunikationszyklus, um ereignisgesteuerte Daten senden zu können.



## Network Idle Time

Network Idle Time bedeutet in etwa „Netzwerk-Stillstandszeit“. Dies ist der Zeitraum, in dem keine Daten auf dem FlexRay übertragen werden. Der Zeitraum wird vom Diagnose-Interface für Datenbus J533 benötigt, um den Ablauf der Datenübertragung auf dem FlexRay synchronisieren zu können. Die Network Idle Time benutzen alle Busteilnehmer, um ihre interne Uhr auf eine globale Zeitbasis zu synchronisieren.

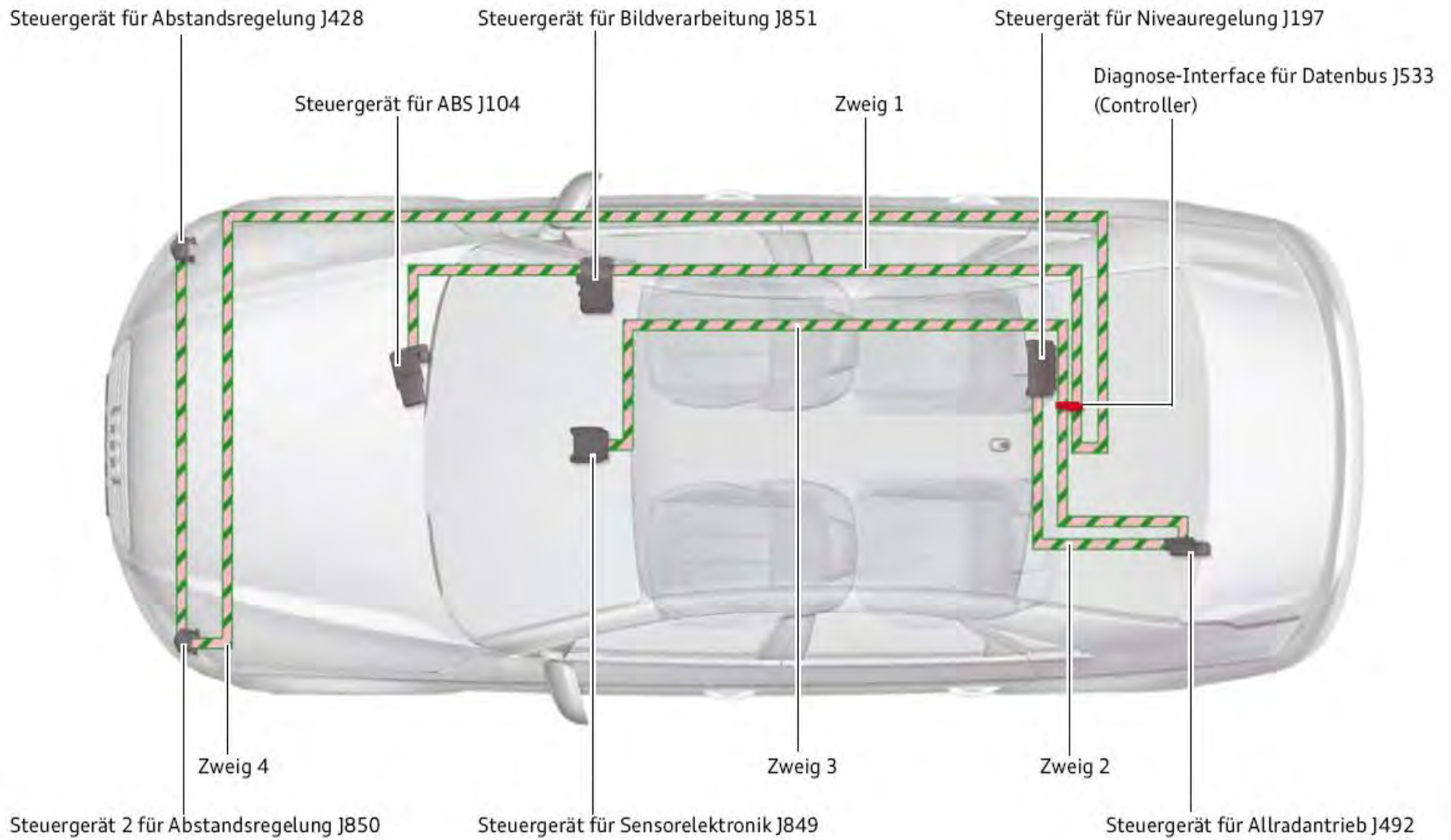


## Aufbau

Die Topologie des FlexRay ist als „aktiver“ Stern mit Punkt-zu-Punkt-Verbindung (Zweig 3) und Daisy Chain (Zweige 1, 2, 4) aufgebaut. Das Diagnose-Interface für Datenbus J533 übernimmt die Funktion des Controllers. Daran befinden sich Anschlüsse für vier Zweige (Branches). Um das Diagnose-Interface für Datenbus J533 herum sind die weiteren Busteilnehmer auf mehreren Zweigen angeordnet.

Im Audi A8 '10 sind dies je Zweig maximal zwei Steuergeräte. Dabei sind der aktive Stern sowie die „Endsteuergeräte“ eines Zweigs niederohmig terminiert (niedriger innerer Widerstand), die „Mittelsteuergeräte“ sind hochohmig terminiert (hoher innerer Widerstand).

# Topologie





## Steuergeräte

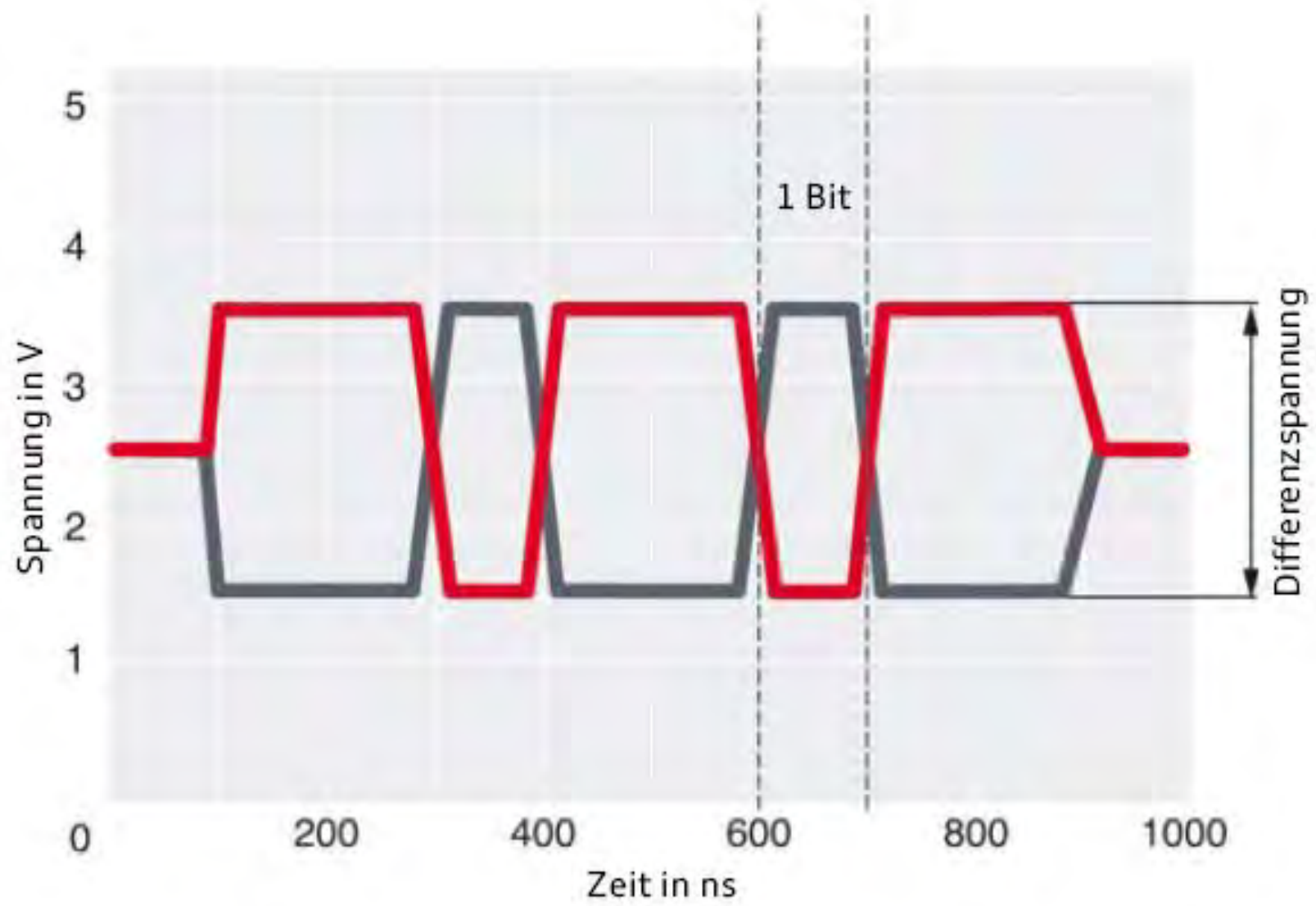
Die „Mittelsteuergeräte“ eines FlexRay-Zweigs verfügen über vier PINs zum Anschluss an den FlexRay, zwei davon zum „Durchleiten“ der Bussignale zu nachfolgenden Steuergeräten. Die anderen beiden dienen zur direkten Kommunikation im Flex-Ray. Die „Endsteuergeräte“, wie z. B. das Steuergerät für ABS J104 (rechte Abbildung), verfügen nur über zwei PINs.



## Signalzustände

Die beiden Leitungen des FlexRay werden mit Busplus und Busminus bezeichnet. Die Spannungspegel beider Leitungen wechseln zwischen minimal 1,5 Volt und maximal 3,5 Volt. Der FlexRay arbeitet mit drei Signalzuständen:

- „Idle“ – die Pegel beider Busleitungen liegen auf 2,5 Volt
- „Data 0“ – die Busplus-Leitung hat einen niedrigen und die Busminus-Leitung einen hohen Spannungspegel
- „Data 1“ – die Busplus-Leitung hat einen hohen und die Busminus-Leitung einen niedrigen Spannungspegel



Ein Bit ist 100 Nanosekunden breit. Die Übertragungszeit ist abhängig von der Leitungslänge und den Übergangszeiten über die Bustreiber. Die Signale werden differenziell übertragen, d. h. es werden zwei Leitungen benötigt.

Im Empfänger wird der eigentliche Bitzustand über die Differenz der beiden Signale ermittelt. Typische Werte sind Differenzspannungen von 1,8 V bis 2,0 V. Direkt am Sender müssen mindestens 1200 mV Differenzspannung anliegen. Beim Empfänger müssen mindestens noch 800 mV Differenzspannung anliegen.

Wenn für 640 – 2660 ms keine Aktivität auf dem Bus stattfindet, geht der FlexRay automatisch in den Sleep-Modus (Idle).

## Verhalten des FlexRay im Fehlerfall

### Kurzschluss einer Busleitung nach Masse

Das Diagnose-Interface für Datenbus J533 erkennt eine dauerhafte Differenzspannung. Der entsprechende Buszweig wird deaktiviert bis wieder „Idle“, d. h. der Spannungspegel des Sleep-Modus, erkannt wird.

### Kurzschluss der Busleitungen zueinander

Das Diagnose-Interface für Datenbus J533 erkennt eine dauerhafte „Idle“-Spannung. Senden und Empfangen ist für die Teilnehmer dieses Buszweigs nicht mehr möglich.

### Ein Steuergerät sendet dauerhaft „Idle“

Das Diagnose-Interface für Datenbus J533 erkennt dieses Verhalten und deaktiviert diesen Buszweig.



## Reparatur einer FlexRay-Leitung

Die FlexRay-Leitungen sind wie die CAN-Leitungen verdreht. Zusätzlich sind sie mit einer Ummantelung versehen. Diese wirkt jedoch nicht als Abschirmung gegen elektromagnetische Störeinflüsse, sondern dient zur Minimierung von äußeren Einwirkungen wie Feuchte und Temperatur auf den Leitungswellenwiderstand. Prinzipiell können FlexRay-Leitungen im Reparaturfall abschnittsweise ersetzt werden. Dabei sind die Entdrilllänge (1) und die Entmantelungslänge (2) zu beachten.

