

Lektion 3

Bus-Arten

In Ford-Fahrzeugen werden verschiedene Datenbussysteme verwendet:

- CAN (**C**ontroller **A**rea **N**etwork)
- LIN (**L**ocal **I**nterconnect **N**etwork)
- ISO (**I**nternationale **O**rganisation für **S**tandardisierung)
- SCP (**S**tandard **C**orporate **P**rotocol)
- ACP (**A**udio **C**ontrol **P**rotocol)

Steuergeräte, die über **CAN** (controller area network)-, SCP- und/oder ISO-Datenbussysteme kommunizieren, lassen sich über den **DLC** (data link connector) mit **IDS** (Integrated Diagnostic System) überprüfen.

CAN-Datenbussystem

- Das **CAN**-Datenbussystem besteht ebenfalls (wie das SCP-Datenbussystem) aus zwei verdrehten Adern. Es hat jedoch ein anderes Protokoll als das SCP-Datenbussystem und ist schneller.
- **CAN** wurde ursprünglich von der Robert Bosch AG speziell für den Einsatz im Automobilbereich als kostengünstige Netzwerklösung entwickelt.

Aufgrund der unterschiedlichen Anforderungen wird das **CAN**-Datenbussystem in zwei Klassen eingeteilt.

- Klasse B:
 - Bei diesem Datenbussystem beträgt die Übertragungsrate von 5 kbit/s bis zu 125kbit/s. Es findet Anwendung im Bereich der Komfort-/Zentralelektrik.
- Klasse C:
 - Bei diesem Datenbussystem beträgt die Übertragungsrate von 125 kbit/s bis zu 1 Mbit/s. Es wird im Bereich der Antriebs- und Fahrwerkstechnik eingesetzt.

LIN (local interconnect network)-Datenbussystem

- Ist ein Standard, der speziell für die kostengünstige Kommunikation zwischen intelligenten Sensoren und Stellgliedern in Kraftfahrzeugen zum Einsatz kommt. Er wird überall dort eingesetzt, wo die Bandbreite und Vielseitigkeit von **CAN** nicht benötigt wird.
- Beim **LIN**-Datenbussystem handelt es sich um ein Eindraht-Datenbussystem.
- Die Datenübertragungsgeschwindigkeit innerhalb des **LIN**-Datenbussystems beträgt bis zu 20 kbit/s. Je nach Anwendung kann die Datenübertragungsgeschwindigkeit auch geringer sein. Beim 2012.75 B-MAX und 2013.25 Kuga z. B. beträgt die Datenübertragungsgeschwindigkeit ca. 9,6 kbit/s.

ISO-Datenbussystem

- Das ISO-Datenbussystem besteht aus einer einzelnen Kommunikationsleitung (K-Leitung).
- Diese K-Leitung dient nicht der Kommunikation zwischen den Steuergeräten untereinander, sondern ausschließlich zur Diagnose eines einzelnen Steuergeräts.
- Bei neuen Fahrzeugen wird das ISO-Datenbussystem zunehmend durch das **CAN**-Datenbussystem ersetzt.
- Die K-Leitung ist dennoch in den meisten aktuellen Steuergeräten noch vorhanden und wird in der Poduktion zum Ein- und Auslesen von Parametern verwendet.
- Die Datenübertragungsgeschwindigkeit der seriellen Kommunikation hängt vom Modelljahr des Fahrzeugs ab. Sie liegt zwischen 4,8 und 10,4 kBits/s.
- Ist die Leitung durch Unterbrechung, Masse- oder Kurzschluss gestört, ist keine Kommunikation zwischen Modul und Diagnostester möglich.

SCP-Datenbussystem

- Das SCP-Datenbussystem besteht aus zwei verdrehten Adern und wird in aktuellen Fahrzeugen nicht mehr verbaut.
- Ist auch nur eine der beiden Leitungen gestört, ist keine Kommunikation zwischen dem Steuergerät und dem Diagnostester mehr möglich.
- Alle Informationen und Daten werden als Paket (Datenblock) seriell übertragen.
- Die Datenübertragungsgeschwindigkeit beträgt etwa 41,6 kBits/s.
- Alle Knoten, d. h. Anschlusspunkte der Steuergeräte, sind gleichberechtigt. Daher können mehrere Steuergeräte an der Ausführung einer Funktion beteiligt sein.
- Es besteht die Möglichkeit der funktionellen und der physikalischen Adressierung.

- Funktionelle Adressierung bedeutet, dass die Information für alle Steuergeräte bestimmt ist.
- Physikalische Adressierung bedeutet, dass die Information nur für ein bestimmtes Steuergerät bestimmt ist.
- Bei gleichzeitig mehreren Nachrichten, werden diese entsprechend ihrer Wertigkeit nacheinander abgearbeitet.
- Auf jede gesendete Nachricht muss mindestens eine gültige Rückmeldung erfolgen. Geschieht dies nicht, wird ein Fehler im Fehlerspeicher abgelegt.

Hinweis: Dieses Datenbussystem ist nur noch ausschließlich in älteren Modellen im Einsatz.

ACP-Datenbussystem

- ist dem SCP-Datenbussystem ähnlich, hat ein einfacheres Protokoll und gilt ausschließlich für Audioanwendungen und Telefonsysteme.
- Ford-Diagnose-Geräte können nicht auf dieses Datenbussystem zugreifen.

Auch dieses Datenbussystem ist in aktuellen Fahrzeugen nicht mehr verbaut.

Die Merkmale des CAN-Datenbussystems

CAN ist ein sogenanntes Multi-Master-Datenbussystem. Das bedeutet, alle Busteilnehmer (Steuer- und Testgeräte) können Daten verschicken und anfordern.

Im CAN-Datenbussystem gibt es **keine Adressierung** der einzelnen Teilnehmer. Statt dessen werden den zu versendenden Datenpaketen **Identifizier** (Identifizierer) zugeordnet.

Jeder Teilnehmer kann seine Daten auf den Datenbus schicken. Die übrigen Teilnehmer entscheiden anhand des Identifiers, ob sie diese Daten auslesen und weiterverarbeiten oder nicht.

Eine der herausragenden Eigenschaften des CAN-Datenbussystems ist seine **hohe Übertragungssicherheit**. Jeder CAN-Controller eines Teilnehmers registriert einen Übertragungsfehler. Diese werden statistisch erfasst und ausgewertet um entsprechende Maßnahmen treffen zu können. Das kann sogar dazu führen, dass der Datenbusteilnehmer, der die Fehler produziert, vom Datenbussystem abgeschaltet wird.

Ein Datenpaket kann in seinem Datenrahmen aus bis zu acht Bytes bestehen. Umfangreichere Übertragungsdaten, werden auf mehrere Datenrahmen verteilt, gesendet.

Die maximale Übertragungsrate liegt bei 1 Mbit/s, d. h. bis zu 1 Million Pulse in der Sekunde. Auf Grund des Leitungswiderstandes gilt dies für Netzwerke bis 40 Meter Datenbuslänge.

Für längere Distanzen muss die Übertragungsrate reduziert werden:

- Entfernungen bis 500 Meter auf 125 kbit/s
 - Ford: 125/250 kbit/s: max 50m, max 32 Steuergeräte
- Entfernungen bis 1000 Meter auf 50 kbit/s
 - Ford: 500 kbit/s: max 33.5m, max 16 Steuergeräte

Varianten des CAN-Datenbussystem

In Ford-Fahrzeugen werden zur Zeit zwei verschiedene CAN-Datenbussysteme eingesetzt:

- **Klasse C:** Ein Hochgeschwindigkeits-CAN-Datenbussystem (**High-Speed = HS-CAN**)
- **Klasse B:** Ein Mittelgeschwindigkeits-CAN-Datenbussystem (**Mid Speed = MS-CAN**) und das B-CAN-Datenbussystem

Folgende Varianten der Klasse C werden zur Zeit eingesetzt:

- HS-CAN-Datenbussystem – Übertragungsrate 500 kbit/s
- EV HS-CAN-Datenbussystem (Hochvoltkomponenten) – Übertragungsrate 500 kbit/s
- Privater HS- CAN-Datenbussystem
- HS- CAN-Datenbussystem Multimedia

Folgende Varianten der Klasse B werden zur Zeit eingesetzt:

- MS-CAN-Datenbussystem – Übertragungsrate 125 kbit/s.
- B-CAN-Datenbussystem – Übertragungsrate 50kbit/s.
- Privater MS-CAN-Datenbussystem
- MS-CAN-Datenbussystem Multimedia

Die Datenbus-Verbindungen bestehen aus verdrehten Datenbusleitungen.

CAN-Protokolle (Botschaften)

Der Aufbau eines CAN-Protokolls lässt sich am Beispiel des Datenübertragungs-Protokolls erläutern. Das Datenübertragungs-Protokoll ist in sieben Bereiche aufgeteilt:

- Startfeld (besteht aus einem Bit und ist immer dominant),
- Statusfeld (11 Bits),
- Kontrollfeld (6 Bits),
- Datenfeld bis zu 64 Bits (8 Bytes),
- Sicherungsfeld (15 Bits),
- Bestätigungsfeld (mehrere rezessive Bits und ein Begrenzungsbit),
- Endfeld (7 rezessive Bits).

Startbit:

- Das Startbit signalisiert allen CAN-Teilnehmern, dass eine Datenübertragung beginnt.
- Das Startbit ist immer dominant und fordert jeden CAN-Teilnehmer zum "Zuhören" auf.

Statusfeld:

- Im Anschluss an das Startbit folgt das Statusfeld.
- Eine Information, die über den CAN-Bus gesendet werden soll, muss zunächst gekennzeichnet werden.
- Soll zum Beispiel ein Temperaturwert übertragen werden, so muss der Wert mit einer bestimmten Kennung versehen werden.
- Die Kennung für die Temperatur ist zum Beispiel 0815. Somit weiß jeder Teilnehmer, dass 0815 einen Temperaturwert darstellt.
- Eine ID ist einer Botschaft zugeordnet. Eine Botschaft kann bis zu 64 Signal beinhalten (8 Byte Datenlänge).
- Des Weiteren enthält das Statusfeld Informationen über die Priorität des Protokolls (siehe Prioritätenregelung).

Kontrollfeld:

- Als nächstes muss den Teilnehmern mitgeteilt werden, wie viele Temperaturwerte kommen werden. Die Anzahl der zu übertragenden Temperaturwerte wird in das Kontrollfeld geschrieben.

Datenfeld:

- Im Datenfeld befinden sich die eigentlichen Daten (im Beispiel die Temperaturdaten). Jeder Teilnehmer kann auf diese Daten zugreifen, wenn er diese benötigt.

Sicherungsfeld:

- Nachdem jeder Teilnehmer die Daten erhalten hat, werden die Daten auf Vollständigkeit geprüft. Die Prüfdaten sind im Sicherungsfeld abgelegt.

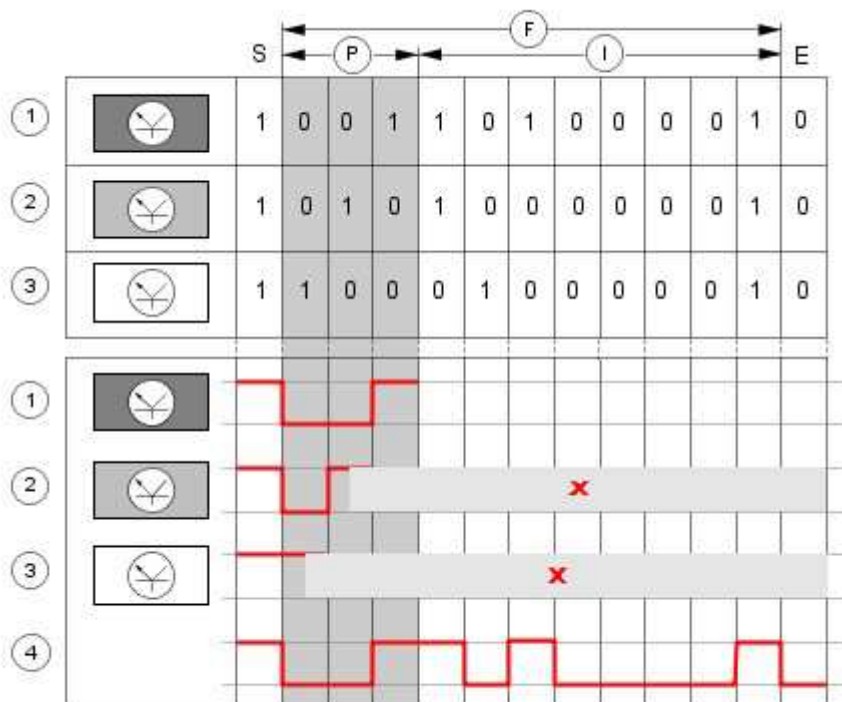
Bestätigungsfeld:

- Sind alle Informationen korrekt angekommen, bestätigen die Teilnehmer das im Bestätigungsfeld. Das Bestätigungsfeld wird vom Sender rezessiv gesendet und vom Empfänger bei korrektem Botschaftempfang dominant überschrieben. Hierdurch wird dem Sender nur signalisiert, dass der Datenübertragung keine Störung aufgetreten ist.

Endfeld:

- Das Endfeld signalisiert den Teilnehmern das Ende des Protokolls.

Prioritätenregelung



E86294

1 Signal vom Modul – ABS (anti-lock brake system)

2 Signal vom PCM (powertrain control module)

3 Signal vom TCM (transmission control module)

4 Resultierendes Bussignal

x Übertragungsabbruch

S Startbit – Statusfeld

E Endbit – Statusfeld

F Statusfeld

I Identifier

P Prioritätenerkennung

Würden alle Steuergeräte gleichzeitig versuchen Daten auf dem Datenbussystem zu versenden, würde es zwangsläufig zu einer Datenkollision kommen. Um dies zu vermeiden, wird folgende Strategie angewendet:

- Jedes aktive Steuergerät beginnt mit dem Sendevorgang.
- Die **Prioritätenerkennung** jedes einzelnen CAN-Protokolls ist im **Statusfeld** festgelegt.
- Dem Statusfeld vorangestellt ist das Startbit. Beendet wird das Statusfeld mit einem Endbit.
- Das Beispiel in der Abbildung zeigt, dass die **ersten drei Statusbits** für die Prioritätenbestimmung festgelegt sind.
- Ein Protokoll von **hoher Priorität** hat automatisch **Vorrang** vor einem Protokoll mit geringerer Priorität.
- Beginnt ein Steuergerät zu senden, verfolgt dieses gleichzeitig Bit für Bit das momentane Geschehen auf dem Datenbus.
- Wird im Identifikationsbereich vom Steuergerät ein rezessives Bit (logisch "1") gesendet, aber ein dominantes (logisch "0") auf dem Datenbus erfasst, erkennt das Steuergerät, dass seine Nachricht durch eine wichtigere überschrieben wurde. Darauf hin bricht das Steuergerät die Übertragung ab.

Hinweis: Beim CAN-Datenbussystem wird ein **dominantes Bit** auch als **höherwertiges bezeichnet**. Somit wird ein **rezessives Bit** auch als **niederwertiges Bit** bezeichnet.

Übertragungssicherheit

Im CAN-Datenbussystem sind alle Teilnehmer parallel zueinander angeschlossen.

Vorteil: Bei Funktionsstörungen (Ausfall) eines Teilnehmers bleibt die Funktionsfähigkeit des Systems erhalten.

Stellt ein Sender eine Störung fest, bricht er die laufende Übertragung mit dem Senden einer Fehlermeldung ab.

Diese besteht aus der Abfolge von sechs dominanten Bits hintereinander und setzt sich gegen jedes andere

Protokoll durch. Im Anschluss kann das Protokoll erneut versandt werden.

In jedem **CAN**-Teilnehmer ist ein Fehlerzählwerk integriert. Dieses sorgt dafür, dass die Leitung nicht durch einen fehlerhaften Datenbusteilnehmer blockiert wird. Ist die zulässige Fehleranzahl überschritten, wird keine weitere Kommunikation mehr zugelassen und ein Fehlercode **DTC** (diagnostic trouble code) gesetzt. Fehlerhafte Teilnehmer werden nicht permanent vom Bus abgetrennt. Es werden zyklisch neue Sendeversuche gestartet.

Störquellen

Störquellen im Fahrzeug sind Bauteile/Systeme, bei deren Betrieb Funken entstehen bzw. Stromkreise geöffnet oder geschlossen werden.

Andere Störquellen sind Stationen, die elektromagnetische Wellen erzeugen, wie zum Beispiel Mobiltelefone und Sendestationen.

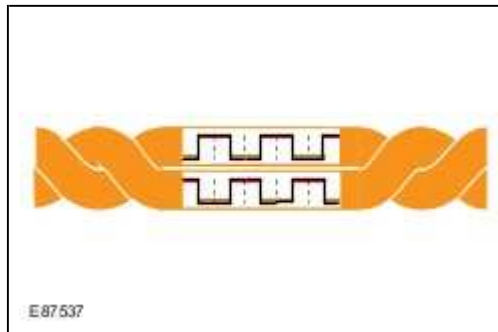
Diese Störquellen können die Datenübertragung auf dem **CAN**-Bus beeinflussen oder verfälschen.

Um Störeinflüsse auf die Datenübertragung zu verhindern, werden die zwei Datenbusleitungen **miteinander verdreht**.

Gleichzeitig werden damit auch Störabstrahlungen von der Datenbusleitung selbst verhindert.

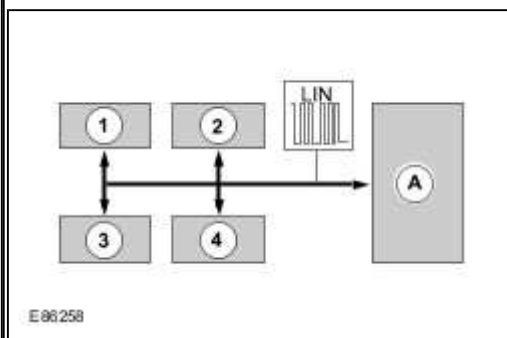
Je nach Übertragungsrage werden die beiden offenen Leitungsenden der Datenbusleitungen mit Abschlusswiderständen verbunden. Hierdurch werden Reflexionen gedämpft, die die Kommunikation stören. Alternativ können die Abschlusswiderstände auch in den Steuergeräten integriert sein. Bei einem **2009 Ford Ka CAN** Klasse B sind keine Abschlusswiderstände verbaut.

Verdrillte **CAN**-Datenbusleitungen



Die Merkmale des **LIN**-Datenbussystem

Beispiel **LIN**-Datenbussystem mit **BCM** (body control module) als Master beim 2011.25 Focus



A **BCM** (Master)

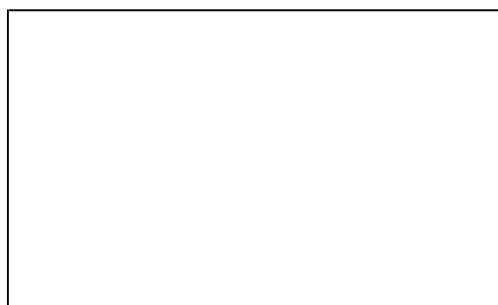
- 1 Spannungsstabilisator (Slave)
- 2 Sensor – Batterieüberwachung (Slave)
- 3 Wischermotor (vorn links) (Slave)
- 4 Regensensor (Slave)

Ein **LIN**-Datenbussystem setzt sich aus einem **LIN-Master**, einem oder mehreren **LIN-Slaves** und der Datenbusleitung zusammen.

Es kommt dort zum Einsatz, wo die Bandbreite und Vielseitigkeit von **CAN** nicht benötigt wird. Die **LIN**-Spezifikation umfasst das **LIN**-Protokoll, ein einheitliches Format zur Beschreibung eines gesamten **LIN**, und die Schnittstelle zwischen einem **LIN** und der Applikation.

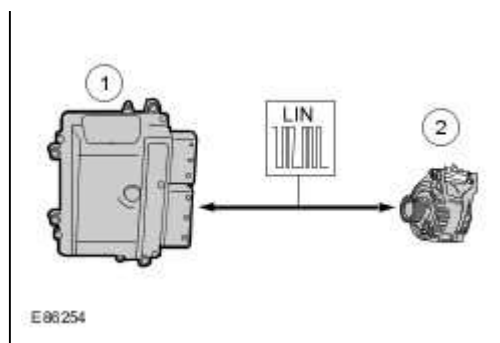
Im **LIN**-Datenbussystem werden **keine**

Beispiel **LIN**-Datenbussystem **PCM** (Master)/und Generator (Slave) beim 2011.25 Focus



Terminierungswiderstände verwendet.

Der Leitungsquerschnitt beträgt $0,35 \text{ mm}^2$. Eine Abschirmung gegen Störquellen ist nicht erforderlich.



1 PCM (Master)

2 Generator (Slave)

LIN-Master

Der LIN-Master fungiert als Gateway (zum Beispiel das PCM oder das BCM) und hat Kenntnis über die zeitliche Reihenfolge aller zu übertragenden Daten. Diese Daten werden von den entsprechenden LIN-Slaves (z. B. Ultraschallsensoren, Lichtschaltereinheit, Generator) übertragen, wenn sie dazu vom LIN-Master aufgefordert werden.

Darüber hinaus hat er folgende weitere Aufgaben:

- Er kontrolliert den Datentransfer und die Datenübertragungsgeschwindigkeit.
- In der Software des LIN-Masters sind für verschiedene Situationen / Ausstattungsvarianten entsprechende Zyklen abgelegt. Die Software wählt daraus den jeweils benötigten aus und fährt diesen Zyklus so lange, bis ein anderer gewählt wird.
- Er übernimmt die Übersetzungsfunktion zwischen den LIN-Steuergeräten des lokalen LIN-Datenbussystems und des CAN-Datenbussystems.
- Er übernimmt die Diagnose der angeschlossenen LIN-Slaves.

LIN-Slaves

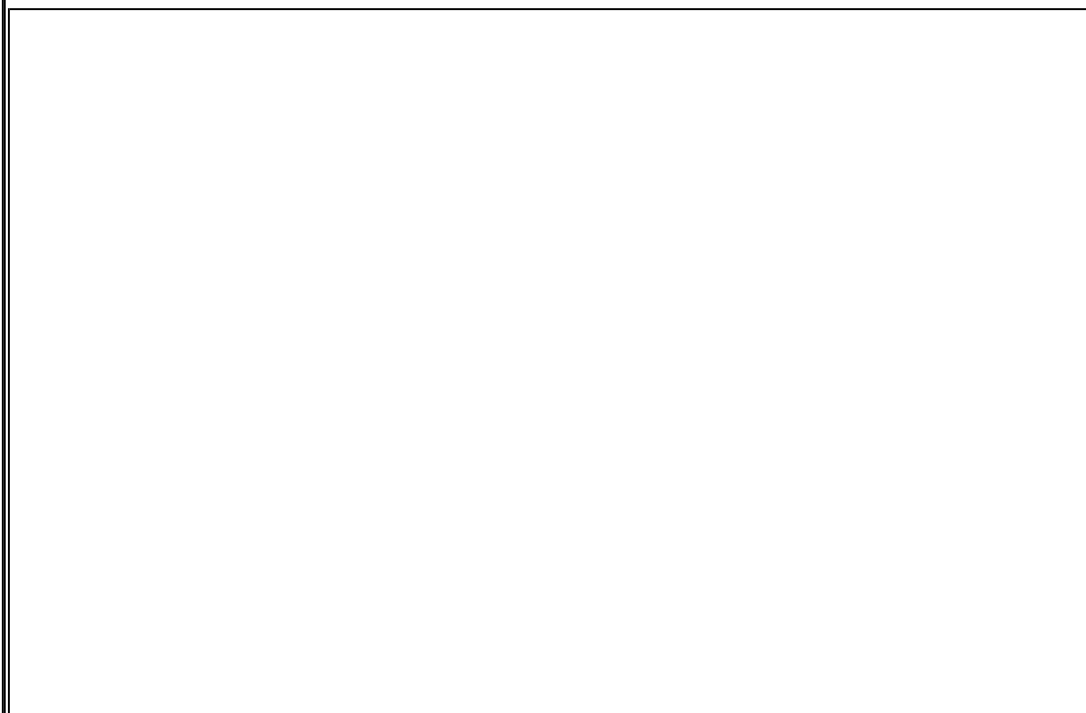
LIN-Slaves können sein:

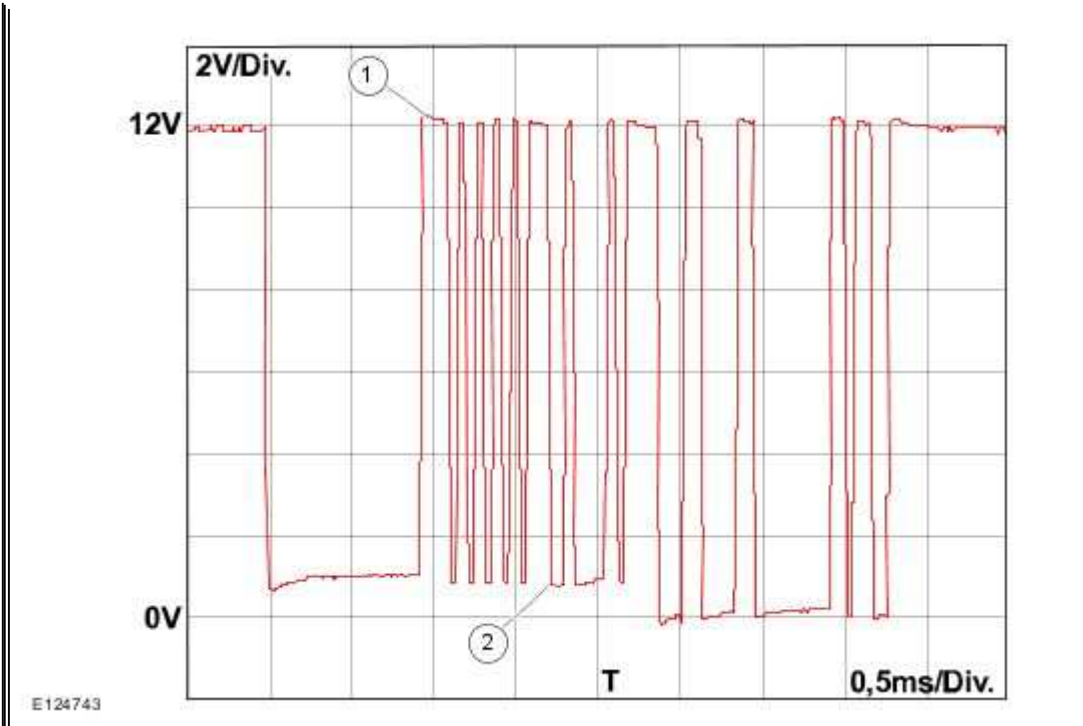
- Stellglieder/Module, wie zum Beispiel Elektronische Lenkschlosseinheit (2011.25 Focus),
- Sensoren, wie zum Beispiel der Licht-/Regensensor (2011.25 Focus),
- Generator (2011.25 Focus).

In den **LIN-Sensoren** ist eine Elektronik integriert, die die gemessenen Werte auswertet. Die Übertragung der Werte erfolgt dann als digitale Signale über das LIN-Datenbussystem.

Die **LIN-Stellglieder/Module** sind intelligente elektronische oder elektromagnetische Baugruppen, die ihre Aufgaben durch das LIN-Datensignal vom LIN-Master erhalten.

LIN-Signale





- 1 Signal – rezessiv
- 2 Signal – dominant

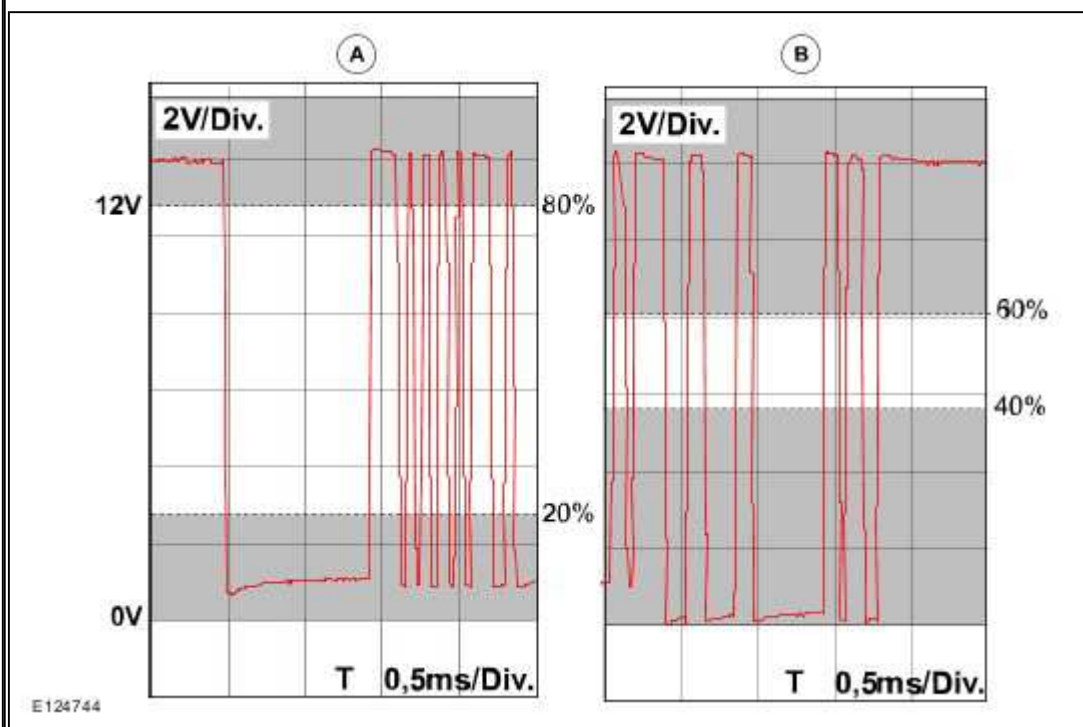
Signal – rezessiv:

- Wird entweder kein Protokoll (keine Botschaft) oder ein rezessives Bit auf dem LIN-Datenbus gesendet, liegt an der Datenbusleitung nahezu Batteriespannung an.

Signal – dominant:

- Um ein dominantes Bit auf dem LIN-Datenbussystem zu übertragen, wird in der Elektronik des sendenden Bauteils die Datenbusleitung durch einen Transceiver auf Masse durchgeschaltet.

Übertragungssicherheit



A Spannungsbereich – Senden

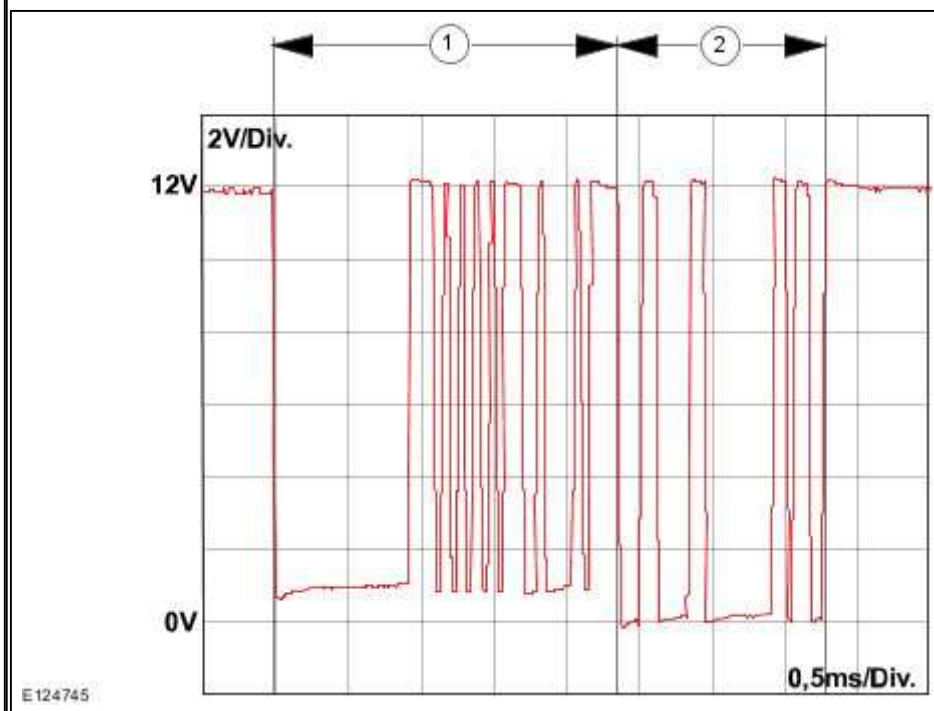
B Spannungsbereich – Empfangen

Durch die Festlegung von Toleranzen beim Senden und Empfangen im Bereich des rezessiven sowie dominanten Signals ist eine stabile Datenübertragung gewährleistet.

Um trotz Störeinstrahlungen noch gültige Signale empfangen zu können, sind die zulässigen Toleranzen

empfangsseitig größer.

LIN-Protokolle (Botschaften)



1 Protokollkopf (Sender: LIN-Master)

2 Protokollinhalt (Sender: LIN-Master oder LIN-Slave)

Ein LIN-Protokoll besteht aus einem Protokollkopf und einem Protokollinhalt.

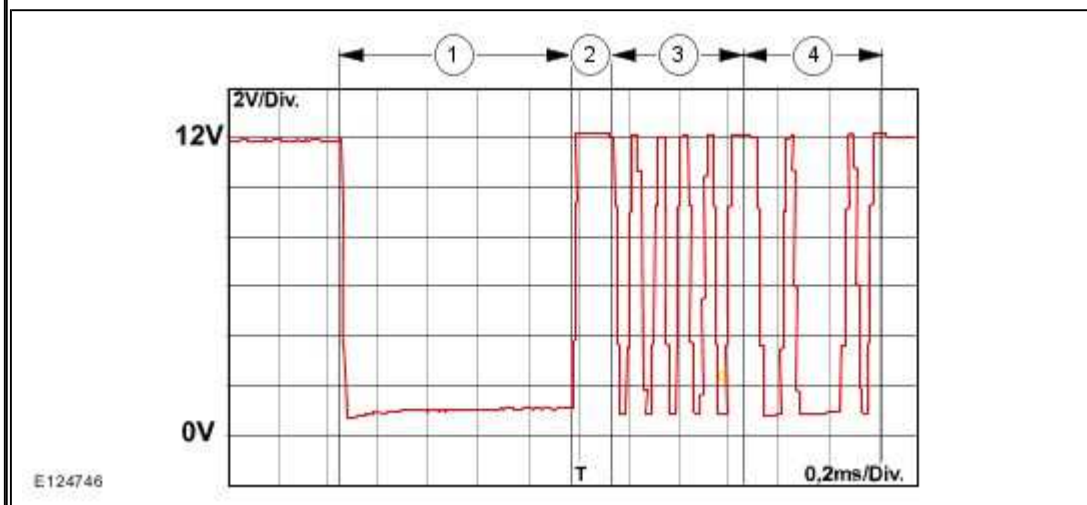
Protokoll mit Slave-Antwort:

- Der LIN-Master fordert durch den Identifier im Protokollkopf einen LIN-Slave auf, Informationen, wie zum Beispiel Schaltzustände oder Messwerte, zu senden.
- Der Protokollinhalt wird vom LIN-Slave gesendet.

Protokoll mit Master-Anweisung:

- Der LIN-Master fordert durch den Identifier im Protokollkopf die entsprechenden LIN-Slaves auf, die Daten des folgenden Protokollinhalts zu verwerfen.
- Das Protokoll wird vom LIN-Master gesendet.

Protokollkopf



1 Synchronisationspause

2 Synchronisationsbegrenzung

3 Synchronisationsfeld

4 Identifier-Feld

Der Protokollkopf wird vom LIN-Master zyklisch gesendet. Er lässt sich in vier Bereiche unterteilen:

- Synchronisationspause,
- Synchronisationsbegrenzung,
- Synchronisationsfeld,
- Identifier-Feld.

Die **Synchronisationspause** ist mindestens 13 Bits lang. Sie wird dominant gesendet.

Die Länge von 13 Bits ist erforderlich, um allen anhängenden **LIN**-Slaves den Start eines Protokolls eindeutig zu signalisieren.

Die **Synchronisationsbegrenzung** ist mindestens 1 Bit lang und rezessiv.

Das **Synchronisationsfeld** besteht aus der Bitfolge 0101010101. Diese Bitfolge veranlasst alle angeschlossenen **LIN**-Slaves sich auf den Systemtakt des **LIN**-Masters einzustellen (zu synchronisieren).

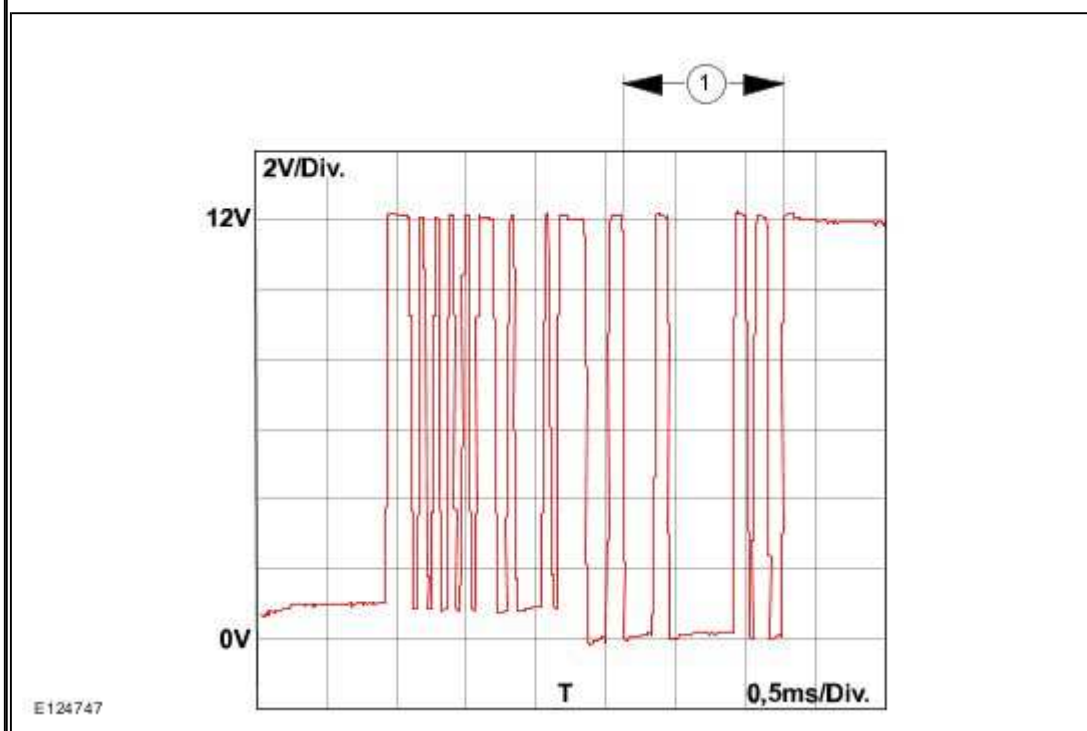
Die Synchronisation aller angeschlossenen **LIN**-Teilnehmer ist für einen fehlerfreien Datenaustausch unbedingt erforderlich. Bei Verlust der Synchronisation würden die Bitwerte beim Empfänger an einer verkehrten Stelle in der Botschaft eingesetzt. Es käme zu Fehlern in der Datenübertragung.

Das **Identifier-Feld** besteht aus 8 Bits. In den ersten 6 Bits ist die Protokollkennung (Identifikation) und die Anzahl der Datenfelder enthalten.

Die letzten beiden Bits enthalten zur Erkennung von Übertragungsfehlern die Prüfsumme der ersten 6 Bits. Dies ist erforderlich, um bei Übertragungsfehlern des Identifiers die Zuordnung eines falschen Protokolls zu verhindern.

Protokollinhalt/Datenfeld

Datenfeld



1 Protokollinhalt (Sender: **LIN**-Master oder **LIN**-Slave)

Im Anschluss an den Protokollkopf folgt der eigentliche Protokollinhalt.

Ein Protokollinhalt kann 1 bis 8 Datenfelder lang sein. Ein Datenfeld besteht aus 10 Bits.

Jedes Datenfeld setzt sich aus einem dominanten Startbit, einem Daten-Byte (8 Bits) und einem rezessiven Endbit zusammen.

Das Start- und das Endbit dienen zur Nachsynchronisation und somit zur Vermeidung von Übertragungsfehlern.