

## 2 Grundlagen

Das erste Kapitel dieser Arbeit soll eine kurze Übersicht über die Grundlagen der EMV geben. Hierzu werden vor allem die EMV-Wirkungskette, die möglichen Kopplungsarten von Störungen sowie die speziellen Anforderungen im Fahrzeug betrachtet. Zudem werden einige in dieser Arbeit notwendige Grundlagen aus der Mikrowellentechnik – die Wellenlängen und Streuparameter – erläutert. Zu Beginn soll jedoch auf das Thema Bussysteme und das in dieser Arbeit untersuchte Bussystem – das LIN-Bussystem – eingegangen werden.

### 2.1 Bussysteme

Sind alle Rechner an einem gemeinsamen Daten-Übertragungsmedium, z.B. einem elektrischen Leiter, angeschlossen, so nennt man dies einen Bus oder ein Netzwerk. Ein solches gemeinsames Übertragungsmedium bietet im Kfz gegenüber der direkten Sensor-/Aktor-Ansteuerung eine große Anzahl von Vorteilen:

**Verringerung der Anzahl von Kabeln und Leitungen** Mit Hilfe eines Bussystems können mehrere Sensorsignale oder Daten über eine gemeinsame Leitung übertragen werden. Dadurch ergibt sich zum einen eine Gewichtsreduzierung des Fahrzeugs, zum anderen wird durch die verringerte Anzahl von Steckern und Verbindungspunkten (z.B. Lötstellen) die Ausfallsicherheit erhöht. Weiterhin vereinfacht sich dadurch auch die konstruktive Verlegung bei der Montage.

**Neue Möglichkeiten des Systemverbundes** Durch die Möglichkeit, Funktionalitäten auf verschiedene Steuergeräte zu verteilen, ergibt sich eine bessere Ausschöpfung des möglichen Funktionspotentials der Steuergeräte. Die durch die Vernetzung der Steuergeräte möglichen verteilten Regelstrategien vermindern die Belastung der einzelnen Steuergeräte. Komplexe Anwendungen bzw. Systeme (z.B. adaptive Dämpfungsregelung) werden dadurch erst möglich. Ein weiterer Vorteil ist die mehrfache Nutzung von Sensorsignalen. Beispielsweise kann der aktuelle Lenkwinkel und die Geschwindigkeit sowohl für das ABS<sup>1</sup>-System als auch für das Navigationssystem genutzt werden.

**Verbesserung der Diagnosemöglichkeiten** Durch gegenseitige Überwachung der Systeme kann eine Fehlererkennung bei Störungen in der Datenverarbeitung eines Steuergerätes detektiert und eventuell sogar unterdrückt werden (Mehrheitsentscheid).

Aus diesen Gründen sind in heutigen modernen Fahrzeugen verschiedene Bussysteme im Einsatz: von einfachen Bussystemen für Sensorsignale, bis hin zu schnellen Bussystemen zur Fahrdynamikregelung.

---

<sup>1</sup>Anti Blockier System

### 2.1.1 Das OSI-Modell

Aus der Normung sind für Kommunikationssysteme die 7 Schichten des OSI-Referenzmodells [13] bekannt, die von Schicht 7 – der Anwendungsschicht (engl. *application layer*) über die Schichten 6 (Darstellungsschicht, engl. *presentation layer*), 5 (Sitzungsschicht, engl. *session layer*), 4 (Transportschicht, engl. *transport layer*), 3 (Vermittlungsschicht, engl. *network layer*) und der Schicht 2 – der Sicherungsschicht (engl. *data link layer*) bis hin zur Schicht 1 – der Bitübertragungsschicht (engl. *physical layer*) reichen [14, 15].

In Abb. 2.1 sind diese 7 Schichten des OSI-Modells dargestellt. Die Bitübertragungsschicht ist die unterste Schicht. Diese Schicht stellt mechanische, elektrische und weitere funktionale Hilfsmittel zur Verfügung, um physikalische Verbindungen zu aktivieren bzw. zu deaktivieren, sie aufrechtzuerhalten und Bits darüber zu übertragen. Das können zum Beispiel elektrische Signale (leitungsgebundene Bussysteme), optische Signale (Lichtleiter, Laser), elektromagnetische Wellen (drahtlose Netze) oder theoretisch auch Schall sein. Geräte und Netzkomponenten, die der Bitübertragungsschicht zugeordnet werden, sind zum Beispiel die Antenne und die Sende- und Empfangsverstärker einer drahtlosen Verbindung. Bei drahtgebundenen Bussystemen fallen neben dem Netzkabel auch der Stecker und die Buchse für das Netzkabel, der Transceiver (ein Kunstwort aus **Transmitter** und **Receiver**) und die zur Erzeugung der Signalform notwendigen Endwiderstände (Terminierungen) unter den Oberbegriff der Bitübertragungsschicht.

Auf der Bitübertragungsschicht wird die Übertragung der digitalen Bits auf einer leitungsgebundenen oder leitungslosen Übertragungsstrecke bewerkstelligt. Die gemeinsame Nutzung eines Übertragungsmediums kann auf dieser Schicht durch frequenz- oder zeitversetzten Zugriff erfolgen. Dies erfordert neben den Spezifikationen bestimmter Übertragungsmedien (zum Beispiel Kupferkabel, Lichtwellenleiter, Stromnetz) und der Definition von Steckverbindungen noch weitere Elemente. Darüber hinaus muss auf dieser Ebene gelöst werden, auf welche Art und Weise ein einzelnes Bit übertragen werden soll. In Rechnernetzen wird heute Information zumeist in Form von Bitfolgen übertragen. Für jedes Medium muss daher eine Codierung der Werte „0“ und „1“ gefunden werden, beispielsweise ein Spannungsimpuls von bestimmter Höhe in einem Kupferkabel oder eine Funkwelle mit bestimmter Frequenz und Amplitude.

Aufgabe der Sicherungsschicht ist es, eine sichere, das heißt weitgehend fehlerfreie, Übertragung zu gewährleisten. Die internationale Elektroingenieursorganisation IEEE sah die Notwendigkeit, für Netze auch den konkurrierenden Zugriff auf ein Übertragungsmedium zu regeln (beispielsweise Arbitrierung), was im OSI-Modell jedoch nicht vorgesehen

Schicht 7	Anwendungsschicht	
Schicht 6	Darstellungsschicht	
Schicht 5	Sitzungsschicht	
Schicht 4	Transportschicht	
Schicht 3	Vermittlungsschicht	
Schicht 2	Sicherungsschicht	LLC
		MAC
Schicht 1	Bitübertragungsschicht	

**Abb. 2.1:** Die 7 Schichten des OSI-Modells. Nach IEEE ist die Schicht 2 nochmals unterteilt in MAC- und LLC-Schicht.

ist. Die IEEE unterteilte deshalb die zweitunterste der 7 Schichten des OSI-Modells in die Unterschichten MAC (Media Access Control) und LLC (Logical Link Control), wobei die MAC-Schicht die untere der beiden ist. Die MAC-Schicht umfasst Netzwerkprotokolle und Beschreibungen, die regeln, wie sich mehrere Rechner das gemeinsam genutzte physikalische Übertragungsmedium teilen. Sie wird benötigt, da ein gemeinsames Medium nicht gleichzeitig von mehreren Rechnern verwendet werden kann, ohne dass es zu Datenkollisionen und damit zu Kommunikationsstörungen oder Datenverlusten kommt. Im ursprünglichen OSI-Modell war eine solche Konkurrenz um das Kommunikationsmedium nicht vorgesehen, weshalb die MAC-Schicht dort nicht enthalten ist.

Bei der Definition eines Busses werden oft Busse auf der Bitübertragungsschicht vermischt mit Bussen der Sicherungsschicht. Busse der Bitübertragungsschicht, wie beispielsweise der LIN- oder CAN-Bus<sup>2</sup>, sind solche, die bereits auf der physikalischen Schicht als Busse funktionieren (jeder Teilnehmer ist an dasselbe Medium – denselben Draht – angeschlossen). Busse, die auf der physikalischen Schicht kein Bussystem darstellen, sondern nur aus einer Kette von Punkt-zu-Punkt-Verbindungen bestehen, werden erst durch die übergeordnete Sicherungsschicht und deren Protokoll zu einem Bus. Ein Beispiel hierfür ist USB<sup>3</sup> oder der Firewire-Bus (IEEE1394).

Die Anforderungen an einen Bus unterscheiden sich in den Datenmengen, die übertragen werden können, in der Schnelligkeit der Übertragung, welche Prioritäten von Daten oder Steuergeräten einzuhalten sind und in den Maßnahmen, die für die Datensicherheit und Fehlererkennung zu ergreifen sind. Weitere Punkte, die typischerweise bei Bussystemen auf der physikalischen Schicht zu beachten sind, sind die Art der Treiberstufen bzw. der Datenübertragung, das Terminierungskonzept sowie auf Protokollebene die Zugriffskontrolle (Busarbitrierung).

### Treiberstufen und Verkabelungsarten

Die einfachste Art der Verkabelung ist eine Eindraht-Leitung mit Rückleitung auf der Masse, wie sie beim LIN-Bus verwendet wird. Um hier zwei mögliche Zustände zu übertragen, kann – wie in Abb. 2.2(a) dargestellt – durch einen einzelnen Transistor der Pegel auf der Busleitung auf Masse gezogen werden (dominant, ein entsprechender Teilnehmer genügt, um den Bus auf Masse zu ziehen) oder – bei hochohmigem Transistor – über einen Pull-up-Widerstand auf das Potential der Betriebsspannung  $U_0$  gelegt werden (rezessiv, alle Teilnehmer müssen gemeinsam den Bus auf die Betriebsspannung schalten). Dies ist ein sogenanntes Single-ended-Bussystem. Jedoch sind die EMV-Eigenschaften durch die unsymmetrische Übertragungstechnik sehr schwierig zu beherrschen.

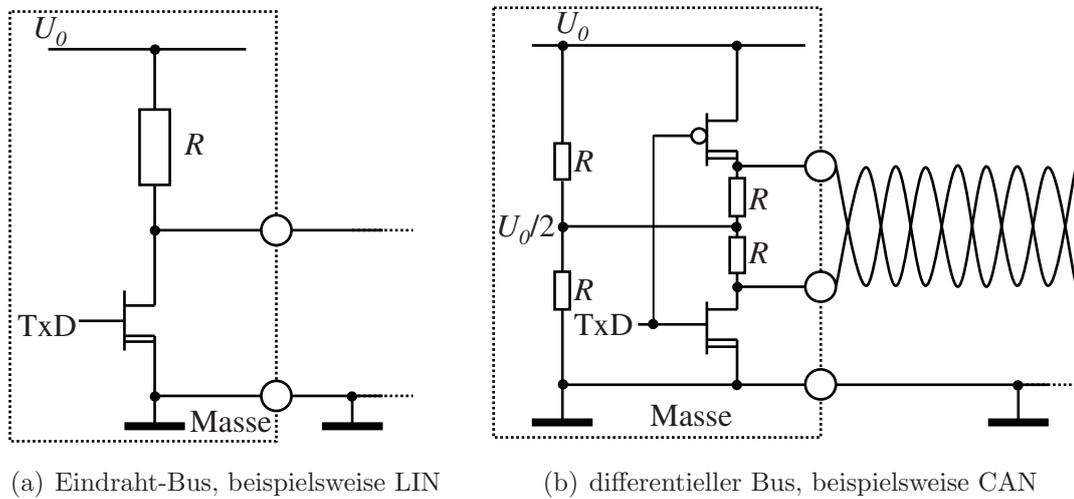
Aus diesem Grund werden für schnellere Busse meist symmetrische Übertragungstechniken verwendet, bei denen die Signale differentiell, also gegenphasig, auf zwei Adern übertragen werden. Hierzu sind Zweidrahtleitungen nötig, die typischerweise verdreht werden (engl. *twisted pair*, TP) und damit unempfindlich sind gegen Gleichtaktstörungen. Die Signalansteuerung über die Treiberstufen kann hierbei unterschiedlich realisiert werden.

---

<sup>2</sup>Controller Area Network, meistverbreitetes Kfz-Bussystem

<sup>3</sup>Universal Serial Bus

Um – wie für den CAN Bus notwendig – einen rezessiven und dominanten Buspegel erzeugen zu können, ist beispielsweise eine Schaltung aus zwei Pull-up-Widerständen und zwei Transistoren notwendig, wie sie in Abb. 2.2(b) dargestellt ist. Hierbei liegt der rezessive Pegel bei beiden Signalleitungen auf dem Mittenpotential  $U_0/2$  und der dominante Pegel auf  $U_0$  bzw. Masse. Für das FlexRay-Bussystem, das keine Bus-Arbitrierung entsprechend dem CAN-Bus benötigt, können andere Schaltungsvarianten verwendet werden, deren Datenzustände „1“ und „0“ beide dominant sind. In diesem Fall sind mehr als zwei Zustände auf dem Bus möglich und der rezessive Zustand beschreibt einen Ruhezustand des Bussystems. Hier ist dann jedoch eine Zugriffskontrolle auf einer höheren Schicht des Bussystems notwendig.



**Abb. 2.2:** Verschiedene Verkabelungsarten und Treiberstufen.

Die für diese differentielle Übertragung verwendeten verdrehten Leitungen sind unempfindlich gegenüber Einstrahlung, da diese auf beiden Adern gleich einkoppelt (engl. *common mode*, Gleichtakt) und nicht im Betriebsmodus des Bussystems (engl. *differential mode*, Gegentakt). Ebenso ist durch die verdrehten Leitungen eine geringe Abstrahlung des Nutzsignals des Bussystems zu erreichen.

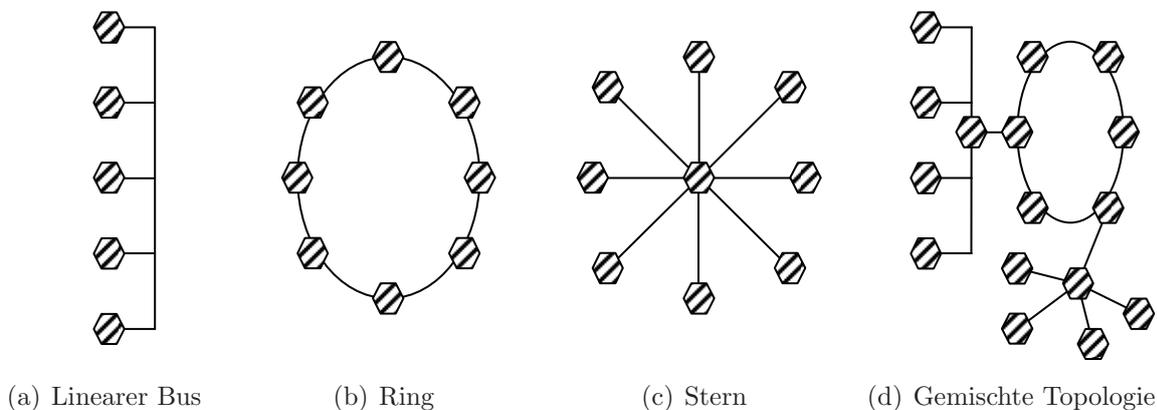
Jedoch fehlt diesen einfachen verdrehten Zweidrahtleitungen eine direkte Bezugsmasse. Dies führt oft zu Problemen, da sich durch Unsymmetrien im Aufbau und in der Ansteuerung typischerweise zum Gegentakt-Datensignal zusätzliche Gleichanteile bilden, deren Bezugsmasse in diesem Fall die Fahrzeugkarosserie ist. Diese Anteile führen bei einer ungeschirmten Zweidrahtleitung zu Emissionen. Deshalb wird für schnellere Datenbussysteme oft eine geschirmte Leitung verwendet. Hierbei sind ein Adernpaar oder mehrere verdrehte Adernpaare von einem Schirm umhüllt, der unter anderem eine definierte Bezugsmasse für das Gleichtaktsignal darstellt und damit die Abstrahlung von Gleichtaktanteilen verhindert.

## Terminierungskonzepte

Um bei elektrischen Übertragungen auf der physikalischen Schicht die spezifizierten Signalformen generieren zu können, ist insbesondere bei schnelleren Bussystemen eine Terminierung erforderlich, die Verformungen des Signals durch Reflexionen an Leitungsenden, Abzweigungen oder Sternpunkten verhindert oder minimiert. Hierzu werden typischerweise an verschiedenen Stellen des Netzwerks die Leitungen mit ihrem entsprechenden Wellenwiderstand abgeschlossen. Um dies erreichen zu können, ist jedoch insbesondere ein definierter Wellenwiderstand der Leitung notwendig, was bei den einfachen Leitungen im Kfz nicht oder nur in eingeschränktem Umfang gegeben ist. Während der LIN-Bus aufgrund der langsamen Signale ohne spezielle Terminierungen arbeitet (die im Leitungsnetz entstehenden Reflexionen sind aufgrund des langsamen Signals nur in den Flanken vorhanden), gibt es für den CAN-Bus verschiedene Varianten von Terminierungen, entsprechend der geplanten Netzwerkauslegung im Fahrzeug. Schnellere Bussysteme (typischerweise  $\geq 50$  MBit/s) werden in der physikalischen Schicht dann meist als reine Punkt-zu-Punkt-Verbindungen mit angepassten Terminierungen ausgeführt.

## Topologien, Busstrukturen

Bei den Busstrukturen unterscheidet man prinzipiell die Sternstruktur, die Ringstruktur und die lineare Struktur (Abb. 2.3). Kombinationen daraus werden auch als gemischte Topologie bezeichnet [16].



**Abb. 2.3:** Verschiedene Bustopologien.

**Linearer Bus** Der lineare Bus, wie er in Abb. 2.3(a) dargestellt ist, ist die einfachste Form des Bussystems. Alle Knoten sind direkt an einer gemeinsamen Busleitung angeschlossen. Meist sind die beiden Enden der linearen Busleitung so beschaltet, dass das elektrische Signal hier nicht reflektiert wird (Abschluss, Terminierung).

**Ring** Eine ringförmige Anordnung, wie in Abb. 2.3(b), erfordert an allen Knoten eine Terminierung und ein protokollgesteuertes Weiterreichen der Daten, da sonst Signale

endlos im Ring laufen. Das heißt, ein Ring-Bus kann nur auf höheren Schichten realisiert werden, wozu ein aufwändigeres Protokoll notwendig ist. Meist wird hierbei ein Zeichen (engl. *token*) weitergegeben, das die Sendeerlaubnis darstellt. Erhält ein Knoten den *token*, schickt er seine Nachricht los und wartet, bis die Nachricht wieder bei ihm ankommt. Der Empfänger der Nachricht setzt während des Umlaufs ein Empfangsbestätigungs-Bit um, damit weiß der Sender, wenn er seine Nachricht wieder erhält, dass diese korrekt übertragen wurde.

**Stern** Im Vergleich zum Ring können Stern-Strukturen entsprechend Abb. 2.3(c) wieder ohne speziell auf die Struktur angepasste Protokolle genutzt werden. Hierbei wird der Bus beispielsweise im Sternpunkt terminiert. Jedoch können auch Sternstrukturen aufgebaut werden, die die Verteilung der Signale im Sternpunkt in der Protokoll-Ebene vornehmen und damit aus physikalischer Sicht eine sternförmige Anordnung von Punkt-zu-Punkt-Verbindungen sind.

**Gemischte Strukturen, Baumstrukturen** Abb. 2.3(d) zeigt eine Kombination von verschiedenen Grundelementen. Hier muss jedoch meist auf der Protokollschicht dafür gesorgt werden, dass die Busstrukturen physikalisch getrennt werden (Terminierungen), denn solche Strukturen sind bei hohen Datenraten auf der physikalischen Schicht aufgrund der entstehenden Signalreflexionen nicht mehr handhabbar.

Oft wird zusätzlich noch der Begriff der Baumtopologie verwendet, welche eine baumartige Vernetzungsstruktur darstellt. Diese Struktur jedoch ist wiederum nichts anderes als eine Verkettung von Sternstrukturen, wobei die außen liegenden Sterne wiederum als Mittelpunkte von Unter-Sternen betrachtet werden können. Meist hängt die dargestellte Topologie nicht nur von der physikalischen Netzstruktur ab, sondern stellt oft auch das darüber liegenden Protokoll dar. Beispielsweise bildet das Firewire-Protokoll in der Protokoll-Schicht eine Baumstruktur mit Wurzel- und Blätter-Knoten aus, wogegen aus physikalischer Sicht (Bitübertragungsschicht) das Firewire-Netzwerk aus einer Reihe von Punkt-zu-Punkt-Verbindungen besteht.

Ein typisches Fahrzeug-Bordnetz besteht heutzutage aus mehreren sogenannten Domänen, die unterschiedliche Teilbereiche der Fahrzeugfunktionen abdecken (Innenraum bzw. Komfortfunktionen, Antriebsstrang bzw. Motorraum und Infotainment/Multimedia). In jeder dieser Domänen gibt es unterschiedliche Bussysteme mit unterschiedlichen Datenraten, beispielsweise für die Übertragung von einfachen Aktor- und Sensordaten (LIN, CAN) oder für die komplexeren Funktionen des Antriebsstranges, wie beispielsweise ESP<sup>4</sup>, die höhere Anforderungen an das Bussystem stellen (CAN, FlexRay). Neben den genannten Systemen hat sich mit dem Infotainment ein weiteres Kommunikationsfeld im Fahrzeug etabliert, das der Unterhaltung der Mitfahrer dienen soll (DVD-Spieler, TV-Empfang, Videodarstellung für die Fond-Passagiere). Auch für diese Anwendungen gibt es spezielle Busse wie LVDS<sup>5</sup>, den D2B<sup>6</sup>-Bus, den MOST<sup>7</sup>-Bus und wiederum FireWire.

---

<sup>4</sup>Elektronisches Stabilitäts-Programm

<sup>5</sup>Low Voltage Digital Signaling; eigentlich ein Überbegriff für differentielle Bussysteme

<sup>6</sup>Domestic Digital Bus

<sup>7</sup>Media Oriented Systems Transport

### 2.1.2 Der LIN-Bus

Das Local Interconnect Network (LIN) wurde vom LIN-Konsortium (Audi, BMW, Daimler, Motorola, VW, Volcano, Volvo) als kostengünstige und einfache Ergänzung zum CAN-Bus konzipiert [17, 18]. Das vordergründige Ziel war es, einfache Kommunikationsaufgaben im Kraftfahrzeug über einen sehr kostengünstigen Bus zu ermöglichen. Beispiele sind die Steuerung von Kleinmotoren oder die Übertragung von langsam veränderlichen Sensordaten (Mechatronik, Smart Sensors). Insbesondere sollte aus Kostengründen ein einheitlicher Standard in Bezug auf Systemkonfiguration, Signalübertragung, Software-Programmierung etc. geschaffen werden. Das Datenformat basiert auf SCI<sup>8</sup>, einem Single-Master/Multiple-Slave-Konzept. Speziell wird hier die UART<sup>9</sup>-Schnittstelle verwendet.

LIN wird einheitlich definiert, das heißt, die Spezifikation umfasst einen kompletten Kommunikationsstandard mit Physical Layer, Anwendungssoftware sowie Interfaces zu den Entwicklungstools. Bei einem einfachen Kommunikationssystem wie LIN sind daher nur die Schichten 1, 2 und 7 des OSI-Referenzmodells implementiert (Physical Layer, Data Link Layer und Application Layer). Wie bereits erwähnt, basiert LIN auf dem genormten UART-Interface, welches in nahezu allen Mikrocontrollern enthalten ist. Aus diesen Gründen ist eine schnelle und kostengünstige Entwicklung und Konfiguration eines LIN-Netzes gewährleistet.

**Bitübertragungsschicht (Physical Layer)** Wie die meisten Feldbusse arbeitet der LIN-Bus, der sich an den Standard ISO 9141 anlehnt, im Master-Slave-Betrieb und überträgt die Daten über ein einadriges Kabel, an das alle LIN-Komponenten – LIN-Master und LIN-Slaves – angeschlossen sind. Die Kabellänge, die Gesamt-Kapazität des Kabels und der Übertragungspegel sind in einer entsprechenden Spezifikation definiert [17]. Die Bustopologie ist meist linear, prinzipiell jedoch beliebig, da der LIN-Bus aufgrund seiner geringen Datenrate sehr einfach zu handhaben ist. Die Datenübertragung erfolgt bitseriell mit maximal 20 kBit/s. Die physikalische Schicht des LIN-Netzwerks ist ein Eindraht-Übertragungssystem, basierend auf der Batteriespannung des Fahrzeugs ( $U_0 = U_{\text{Bat}} = 12 \text{ V}$ ). Eine Skizze des Prinzips der Bitübertragung bei LIN ist bereits in Abb. 2.2(a) dargestellt. Dabei gilt 0 V als der dominante Pegel und demzufolge  $U_{\text{Bat}}$  als rezessiv. Der High-Pegel ist rezessiv, weil er nur erreicht wird, wenn die Sendetransistoren aller LIN-Knoten gesperrt sind. Der Low-Pegel ist dominant, da er sich sofort einstellt, wenn nur einer der Sendetransistoren durchschaltet. Der dominante Pegel überschreibt also den rezessiven Pegel. Heutzutage werden diese Bustreiber von vielen Halbleiterherstellern angeboten [19, 20, 21], wobei sich die Funktionen nur unwesentlich unterscheiden. Immer mehr Halbleiterhersteller gehen auch dazu über, die LIN-Treiberstufe in komplexere Bausteine zu integrieren.

**Protokollschicht (Data Link Layer)** Auch die Protokollschicht des LIN-Systems ist darauf ausgelegt, eine einfache und kostengünstige Kommunikation zu realisieren, an die keine hohen Anforderungen gestellt werden. Aus diesem Grund basiert das LIN-Protokoll auf

<sup>8</sup>Serial Communication Interface

<sup>9</sup>Universal Asynchronous Receiver Transmitter

einer festen Master-Slave-Struktur, die ein Zeitmultiplexverfahren (TDMA) zur Zugriffsverwaltung auf das Kommunikationsmedium verwendet. Beim TDMA-Verfahren wird vom Masterknoten den Slaves periodisch ein Zeitschlitz zugeteilt, in denen die Übertragung des Nachrichtenrahmens stattfindet.

Im Fall des LIN-Protokolls wird vom Master der Header des Datenrahmens gesendet und die jeweiligen Slaves hängen die entsprechenden Daten an ihren Header an. Durch dieses deterministische System kann das Übertragungsraster jeder Nachricht eindeutig berechnet werden und es wird eine maximale Latenzzeit garantiert. Vorteil dieses zentralisierten Protokolls ist die Möglichkeit, eine kostengünstige Konzeption der Peripherieknoten (Slaves) zu ermöglichen, da durch diese Konstellation nur für den Master ein Quarz zur Takterzeugung notwendig ist, während in den Peripherieknoten ein Schwingkreis ausreicht.

**LIN-Datenrahmen** Der Datenrahmen (engl. *frame*) des LIN-Busses ist relativ einfach aufgebaut und besteht aus dem Header, der für den Synchronisationsmechanismus ein Break- und ein Sync-Feld sowie das Identifier-Feld enthält, das die Nachricht exakt identifiziert. Der LIN-Header, mit dem der Master die Nachricht initiiert, dient der Synchronisation der LIN-Slaves. Zu diesem Zweck wird als Start of Frame ein 13 Bit langer „0“-Pegel gesendet, der den Slaves den Beginn eines neuen Datenrahmens anzeigt. Diesem folgt das Sync-Feld mit fünf „1-0“-Folgen, von denen die Slaves die negative Signalfanke für die Taktsynchronisation verwenden.

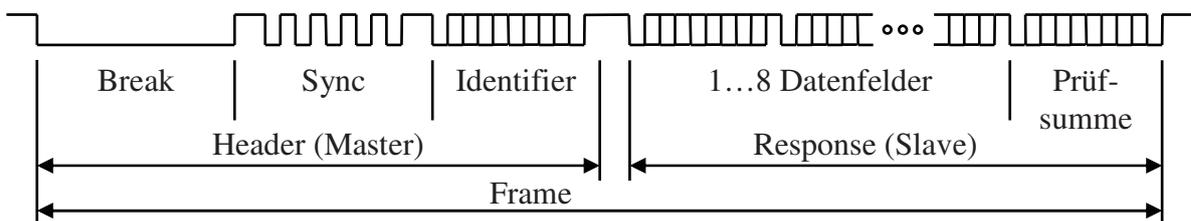


Abb. 2.4: Der LIN-Datenrahmen.

Da das LIN-Protokoll ohne Quell- und Zieladressen arbeitet, werden die Kennungsinformationen der Nachricht im Identifier-Feld übertragen. Das auch als Message-Identifier bezeichnete Datenfeld besteht aus 6 Identifier-Bits und 2 Paritätsbits und ist somit wesentlich kleiner als das des CAN-Buses. Mit dem Identifier-Feld wird den Slaves mitgeteilt, welche Nachricht nun folgt bzw. welcher Slave nun senden muss. Dem Identifier-Feld folgt nach einer kurzen Pause („Interframe-Gap“) das Datenfeld, das aus 1 Byte bis 8 Byte bestehen kann. Das Datenfeld beginnt mit dem geringstwertigen Bit (engl. *least significant bit*, kurz LSB). Das Prüfsummenfeld schließt den LIN-Datenrahmen ab. Als Prüfsumme wird die Modulo-256-Summe über alle Datenbytes gebildet.

Die entscheidenden Vorteile des LIN-Busses sind die Nutzung der standardmäßig bei Mikrokontrollern vorgesehenen SCI(UART)-Schnittstelle – somit entfällt ein separater Buscontroller (wie bei CAN verwendet), die Selbstsynchronisierung ohne Taktoszillatoren in den Slaves und die aus diesen Gründen möglichen kostengünstigen Transceiver. Damit und durch die Verwendung der Eindraht-Schnittstelle ist der LIN-Bus somit eine sehr kostengünstige Bus-Implementierung.