

Grundlagen der Bussysteme in Industrie und Gebäudetechnik

R. Tuzinski, Berlin

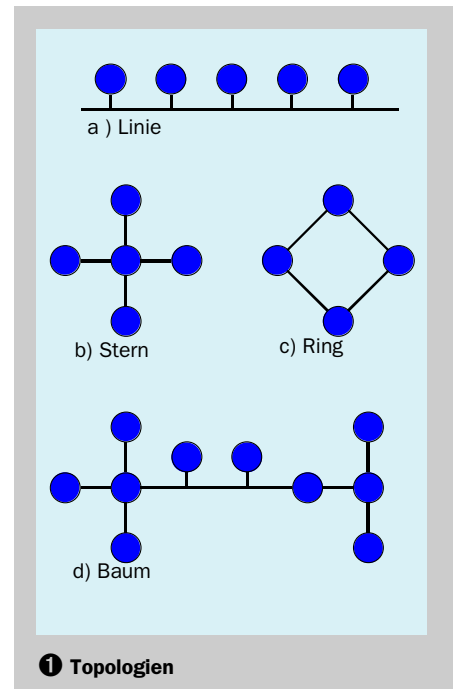
Ob Feldbusse, Ethernet, EIB, LON oder andere – längst haben sich die unterschiedlichsten Bussysteme in Industrie und Gebäudetechnik etabliert. Bei dieser Vielfalt wird oft übersehen, dass die Mechanismen vieler Systeme ähnlich sind. Der Beitrag beschreibt die Grundlagen der Bustechnik in Hinblick auf physikalische Eigenschaften und verwendete Zugriffsverfahren.

1 ISO/OSI-Schichtenmodell

Mit steigendem Automatisierungsgrad einer Maschine, Anlage oder eines Gebäudes wächst die Anzahl der Ein-/Ausgabepunkte (E/A) und damit der Verkabelungsaufwand bei paralleler Verdrahtung. Gleichzeitig erhöht sich der Aufwand bei Projektierung, Installation, Inbetriebnahme und Wartung. Ab einer gewissen Zahl von Komponenten und der zu realisierenden Aufgaben ist deshalb die serielle Vernetzung der Komponenten wirtschaftlicher und praktikabler. Der Bus ersetzt die parallelen Leitungsbündel durch ein einziges Buskabel und verbindet alle Geräte, z. T. vom einfachen binären Sensor bis hin in die Leitebene.

Da Bussysteme in den unterschiedlichsten Anwendungen eingesetzt werden, lässt sich nicht pauschal sagen, dass ein bestimmter Bus grundsätzlich am besten geeignet ist. So kommt es beispielsweise in der Leitebene darauf an, große Datenmengen zu beherrschen, während direkt an Sensoren und Aktoren oft nur einzelne Bits zu übertragen sind. Darüber hinaus sind – vor allem im industriellen Bereich – oft zeitkritische Probleme oder große Störeinflüsse zu berücksichtigen. Hier muss gewährleistet sein, dass alle prozesskritischen Daten zu einem vorhersagbaren Zeitpunkt

und unverfälscht zur Verfügung stehen. Zur Einteilung der Bussysteme kann zunächst das ISO/OSI-Referenzmodell (OSI = Open System Interconnection) herangezogen werden (Tafel 1). In ihm hat die Internationale Organisation für Standardisierung (ISO) sieben Schichten festgelegt als Basis der weltweiten Normung von Kommunikationsprotokollen. Definiert wird lediglich das externe Verhalten eines Teilnehmers, nicht aber die konkrete Realisierung. Eine Schicht fasst jeweils zusammengehörige Funktionen zusammen und hat Schnittstellen zu den Nachbarschichten. Bei den Bussystemen in der Industrie sind in der Regel nur die Schichten 1, 2 und 7 vorhanden. Der EIB hat zusätzlich die Schichten 3 und 4 implementiert und LON alle sieben. Bei besonders hohen zeitlichen Anforderungen wird der Austausch der Daten in Schicht 2 vorgenommen. Der Verzicht auf die Schichten 3 bis 6 in der Industrie hat einerseits funktionale Gründe (die Bussysteme haben beispielsweise einen fest vorgegebenen Weg zwischen den einzelnen Teilnehmern und benötigen keine Vermittlungsschicht), zum anderen bedeutet jede zusätzliche Schicht einen erhöhten Zeitbedarf. Darüber hinaus kommen typischerweise Automatisierungsgeräte derselben Art zum Einsatz, sodass dadurch viele Fest-



legungen schon prinzipiell vorhanden sind. Schicht 1, die Bitübertragungsschicht, beschreibt den grundlegenden physikalischen Aufbau von Bussystemen.

2 Topologie

Der Begriff „Topologie“ stammt ursprünglich aus der Geometrie und bezeichnet dort die Lehre von der Lage und Anordnung geometrischer Gebilde im Raum. In der Datenverarbeitung ist damit die Verbindungsstruktur zwischen Knoten in einem Netzwerk gemeint, also die Netzwerk-Topologie. Zu unterscheiden sind zwei grundsätzliche Varianten, die Linien- und die Sternstruktur.

2.1 Linie

In dieser Topologie sind die Komponenten (Busteilnehmer) in einer Reihe miteinander verbunden (Bild 1a). Eine solche Anordnung wird in der EDV eigentlich als „Bus“ bezeichnet, beispielsweise der ISA- oder der PCI-Bus in Computern. Um Verwechslungen zu vermeiden, ist bei den Bussystemen in Industrie und Gebäudetechnik der Begriff „Linie“ üblicher. Die maximale Länge der Linien ist vom Übertragungsmedium und von der Anzahl der angeschlossenen Geräte abhängig. Sie kann durch den Einsatz von so genannten Repeatern vergrößert werden. Solche Repeater sind nichts anderes als Verstärker, die die Signale vom Sender zum Empfänger und umgekehrt anheben.

2.2 Stern

Bei der Sternstruktur (Bild 1b) sind alle Busteilnehmer direkt mit einem zentralen Knoten verbunden, der auch als Hub be-

Tafel 1 ISO/OSI-Schichtenmodell

Schicht	Bezeichnung	Funktion
7	Anwendungsschicht (Application Layer)	Schnittstelle zum Anwenderprogramm mit anwendungsorientierten Befehlen
6	Darstellungsschicht (Presentation Layer)	Darstellung (Kodierung) der Daten für Auswertung- und Interpretation in der nächsten Schicht
5	Sitzungsschicht (Steuerung) (Session Layer)	Auf- und Abbau von temporären Teilnehmerverbindungen; Synchronisation kommunizierender Prozesse
4	Transportschicht (Transport Layer)	Kontrolle der Datenübertragung für Schicht 5 (Transportfehler, Zerlegung in Teilpakete)
3	Netzschicht (Vermittlung) (Network Layer)	Auf- und Abbau von Verbindungen; Vermeidung von Staus im Netz
2	Verbindungsschicht (Data Link Layer)	Beschreibung des Buszugriffsverfahren (MAC = Medium Access Control) einschließlich Datensicherung
1	Bitübertragungsschicht (Physical Layer)	Definition von Hardware, Kodierung, Geschwindigkeit der Datenübertragung

zeichnet wird. Der „einfache“ Hub schaltet alle Teilnehmer datentechnisch parallel zusammen, während ein Switch auch noch unterscheiden kann, welcher Teilnehmer in der Regel mit welchem anderen in Verbindung treten möchte. Ist der Knoten darüber hinaus noch in der Lage, unterschiedliche Protokolle von verschiedenen Teilnehmern zu übersetzen und anzupassen, wird er als Router bezeichnet. Ein typisches Beispiel ist der DSL-Router, der das auf Ethernet basierende Heim- oder Büro-LAN (local area network) und das DSL-Netz der Telekom verbindet.

Eine Sonderform der Linie, bei der Anfang und Ende miteinander verbunden sind, ist der **Ring** (Bild 1c).

Aus der Kombination von Linien und Sternen entsteht ein **Baum** (Bild 1d). Switches und Router kommen in Baumstrukturen vor allem zum Einsatz, wenn eigenständige Teilnetze gebildet (Segmentierung) oder vorhandene Netzwerke gekoppelt werden sollen.

3 Physikalische Eigenschaften

3.1 Leitungen

In Schicht 1 des ISO/OSI-Modells wird neben der Topologie und dem mechanischen Aufbau der Steckverbinder auch definiert, welche Leitungen zum Einsatz kommen:

Verdrillte Zweidrahtleitungen (Twisted Pair) sind die am weitesten verbreitete Verdrahtungsart bei Sensoren- und Aktoren. **Koaxialkabel** erlauben große Reichweiten, sind einfach zu installieren und EMV-gerecht.

Lichtwellenleiter (LWL) gewährleisten hohe Datensicherheit und Datenraten. Es müssen aber Einschränkungen bei der Topologie (meist sternförmig) und Handhabung in Kauf genommen werden. Die EMV-Festigkeit ist sehr gut.

3.2 Kodierung und Schnittstellen

Kodierung. Ferner werden die Spannungspegel und die Kodierungen für die binären Signale festgelegt (Bild 2), entweder unipolar, bipolar ohne Rückkehr zum Spannungsnullpunkt (NRZ = non-return-to-zero), bipolar mit Rückkehr zum Spannungsnullpunkt (RZ = return-to-zero) oder nach dem Manchester-Verfahren.

Beim Manchester-Verfahren wird die binäre Informationen durch einen Spannungswechsel innerhalb der Dauer eines Bits (Bitzeit) dargestellt. Dadurch können Sender und Empfänger leichter synchronisiert werden, denn der Übergang in der Mitte der Bitzeit ergibt einen zuverlässigen Takt. Die erste Hälfte der Bitzeit enthält den invertierten, die zweite Hälfte den eigentlichen Bitwert. Die Kodierung stellt meistens eine XOR-Verknüpfung aus Takt und (Bit-)Signal dar.

Von den unterschiedlichen Kodierungsverfahren werden bei den Bussystemen in Industrie und Gebäudetechnik hauptsächlich die bipolare NRZ-Kodierung, aber auch das Manchester-Verfahren (z. B. bei AS-Interface, Profibus) eingesetzt.

Schnittstellen. Die Kodierung der einzelnen Bits ist eng verwandt mit der Schnittstellen-Definition. Eine solche Schnittstelle besteht zwischen allen Geräten und dem Bus und legt physikalisch fest, in welchem Format und mit welchen Spannungen die Daten auf den Bus gegeben werden.

Beim seriellen Datentransfer müssen die einzelnen Bits, aus denen jedes Datenbyte besteht, nacheinander auf den Bus gegeben werden. Die Übertragung eines Bytes beginnt in der Regel mit einem vorangestellten Startbit. Anschließend folgen nacheinander 5 bis 8 Datenbits. Dem letzten Datenbit kann ein Paritätsbit folgen, das zur Erkennung von Übertragungsfehlern dient. Das Paritätsbit bewirkt, dass bei gerader Parität (Even) immer eine gerade bzw. bei ungerader Parität (Odd) eine ungerade Anzahl von "1"-Bits übertragen wird. Das Ende des Zeichens bilden 1 oder 2 Stoppbits. Aus der Computerwelt sehr bekannt ist die RS232-Schnittstelle zur seriellen Kommunikation. Sie war früher als COM-Port an jedem PC vorhanden, wird dort aber inzwischen von der USB-Schnittstelle (Universal Serial Bus) verdrängt.

Die erzielbare Entfernung zwischen zwei RS232-Geräten ist wie bei allen seriellen Übertragungsverfahren vom verwendeten Kabel und der Baudrate abhängig. Als Richtmaß sollte hier bei einer Übertragungsrate von 9600 bit/s eine Distanz von 15 bis 30 m nicht überschritten werden. RS232-Schnittstellen besitzen neben den Datenleitungen eine Vielzahl von zusätzlichen, so genannten Handshake-Verbindungen, die jedoch in ihrer Gesamtheit fast nur zur Verbindung zwischen Modem und Computer benötigt werden. Der weitaus häufigere Fall der Verbindung zweier Geräte untereinander lässt sich in der Regel mit einer reduzierten Anzahl von Leitungen realisieren. Während die klassische RS232 den Datenverkehr zwischen zwei Geräten spezifiziert, ist bei Bussen der Zugriff mehrerer Geräte auf das gemeinsame Übertragungsmedium nötig. Dies wird durch die Schnittstellen RS422/423 und RS485 ermöglicht, wobei letztere bei zahlreichen Bussystemen zum Einsatz kommt.

Da mehrere Sender auf einer gemeinsamen Leitung arbeiten, muss durch ein Protokoll sichergestellt werden, dass zu jedem Zeitpunkt maximal ein Sender aktiv ist (s. u.: Zugriffsverfahren). Alle anderen Sender müssen sich zu dieser Zeit in hochohmigem Zustand befinden. Die Aktivierung der Senderbausteine kann durch Schalten einer Handshake-Leitung oder datenflussgesteuert erfolgen. Eine Terminierung des Kabels

ist bei RS422-Leitungen nur bei hohen Baudraten und großen Kabellängen, bei RS485-Verbindungen dagegen grundsätzlich nötig. Obwohl für große Entfernungen bestimmt, zwischen denen Potentialverschiebungen unvermeidbar sind, schreibt die Spezifikation für keine der beiden Schnittstellen eine galvanische Trennung vor. Da die Empfängerbausteine empfindlich auf Verschiebung des Massepotentials reagieren, ist für zuverlässige Installationen eine galvanische Trennung empfehlenswert.

Die Daten werden ohne Massebezug als Spannungsdifferenz zwischen zwei korrespondierenden Leitungen übertragen. Für jedes zu übertragende Signal existiert ein Aderpaar, das aus einer invertierten und einer nicht invertierten Signalleitung besteht. Die invertierte Leitung wird in der Regel durch den Index „A“ oder „-“ gekennzeichnet, während die nicht invertierte Leitung mit „B“ oder „+“ bezeichnet wird.

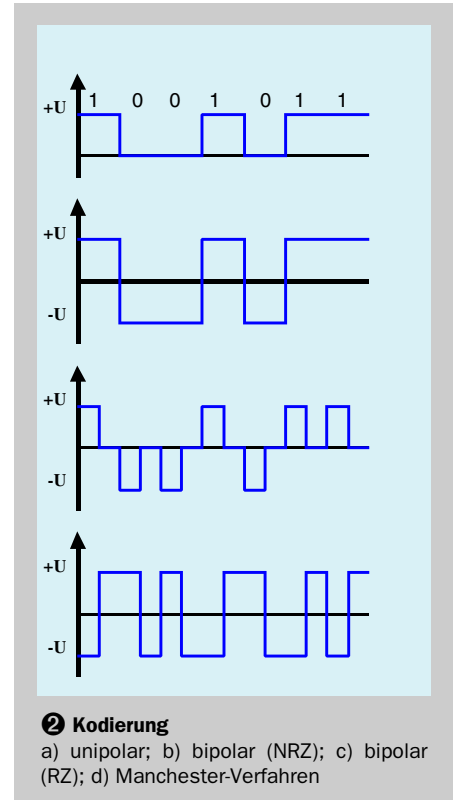
Die RS485-Schnittstelle stellt eine Erweiterung der RS422-Definition dar. Während die RS422 lediglich den unidirektionalen Anschluss von bis zu zehn Empfängern an einen Sendebaustein zulässt, ist die RS485 als bidirektionales Bussystem mit bis zu 32 Teilnehmern konzipiert. Physikalisch unterscheiden sich beide Schnittstellen nur unwesentlich. Der Empfänger wertet lediglich die Differenz zwischen beiden Leitungen aus, sodass Gleichtakt-Störungen auf der Übertragungsleitung nicht zu einer Verfälschung des Nutzsignals führen. Durch die Verwendung von abgeschirmtem, paarig verseiltem Level-5-Kabel lassen sich Datenübertragungen über Distanzen von bis zu 1200 m bei einer Geschwindigkeit von bis zu 100 kbit/s realisieren (100 m bei 12 Mbit/s). RS422-Sender stellen unter Last Ausgangspegel von ± 2 V zwischen den beiden Ausgängen zur Verfügung. Die Empfängerbausteine erkennen Pegel von ± 200 mV noch als gültiges Signal.

4 Zugriffsverfahren

Durch den Bus besteht zwischen allen Teilnehmern eine Verbindung, die wie beschrieben als elektrische Leitung, als Lichtwellenleiter oder auch als Funkkanal ausgeführt sein kann.

Relativ unproblematisch ist der Datenempfang, da prinzipiell alle Teilnehmer den gesamten Datenverkehr auf dem Bus jederzeit „mithören“. Wollen aber mehrere Teilnehmer gleichzeitig senden, so sind dafür Übertragungsregeln nötig. Diese Protokolle stellen sicher, dass zu jedem Zeitpunkt immer nur ein Sender aktiv ist, damit sich keine Daten auf dem Kanal überlagern.

Echtzeit. Bei den so genannten Echtzeit- oder deterministischen Systemen kommt hinzu, dass die Wartezeit eines sendewilligen Teilnehmers bis zum Senden und dem



Abarbeiten der Meldung in einer eindeutig vorhersehbaren Zeitspanne liegt. Schnelle, genau definierte Reaktionszeiten sind insbesondere bei Alarm- und Störmeldungen wichtig, wodurch der Begriff „Echtzeit“ oft als eine Art „Schnelligkeit“ angesehen wird, was aber falsch ist. Ein Bussystem, das beispielsweise immer nach maximal einer Stunde reagiert, arbeitet genauso in Echtzeit wie Systeme in Antriebsregelungen oder in der Sicherheitstechnik, die Aufgaben im Mikrosekundenbereich bewerkstelligen. Für die Echtzeitfähigkeit kommt es nicht darauf an, wie schnell reagiert wird, sondern dass die Reaktion innerhalb einer vorhersehbaren (berechenbaren) Zeit stattfindet.

4.1 Frequenzmultiplex

Diese Technik eröffnet prinzipiell die Möglichkeit, dass pro Kanal immer nur ein Teilnehmer sendet. Jedem Teilnehmer kann ein eigener Frequenzbereich zugewiesen werden und jeder, der „zuhören“ will, muss einen Empfänger für diese Frequenz haben. Alternativ können alle Teilnehmer auch auf einem gemeinsamen breiteren Frequenzband kommunizieren (Spread-Spectrum). Bei den üblichen Bussystemen in Industrie und Gebäudetechnik kommt das Frequenzmultiplex kaum zum Einsatz. Sie arbeiten zeitgesteuert, wie nachstehend ausgeführt.

4.2 Zeitmultiplex

Bei den Zeitmultiplexverfahren steht der Bus den einzelnen Teilnehmern zeitlich gestaffelt zur Verfügung. Unterschiede ergeben sich durch die Art des Zeitrasters:

Beim festen Zeitraster wird jedem Teilnehmer der Bus zugeordnet, unabhängig davon, ob er senden will oder nicht. Effektiver sind Zeitmultiplexverfahren mit einem bedarfsabhängigen Raster. Hier erfolgt die Buszuteilung nur dann, wenn auch wirklich eine Übertragungsanforderung vorliegt. Der Datentransfer wird so insgesamt schneller abgewickelt, jedoch ist bei diesem Verfahren eine übergeordnete Instanz, ein so genannter Master nötig, der die Buszuteilung übernimmt.

4.3 Master/Slave-Verfahren

Der Master legt die Regeln fest, nach denen die Kommunikation erfolgen soll und steuert den gesamten Kommunikationsablauf. Die Teilnehmer, die der Master anspricht, werden als Slaves bezeichnet. Sie müssen den Nachrichtentransfer exakt so durchführen, wie es vom Master festgelegt wird. Zu jedem Zeitpunkt darf immer nur ein einziger Master für die Buszuteilung zuständig sein. Der große Nachteil dieses Verfahrens ist, dass beim Ausfall des Masters auch das gesamte Bussystem ausfällt. Es gibt allerdings auch die Möglichkeit eines Mastertransfers, also die Übergabe des Masterrechts von einem Teilnehmer zu einem anderen (Multi-Master-Systeme).

Beim Datenverkehr in Master/Slave-Systemen gibt es prinzipiell zwei Varianten: Zum einen kann der Master Daten auf den Bus geben, die der Slave übernimmt und weiterverarbeitet. Zum anderen kann der Master Daten anfordern, die der Slave daraufhin sendet. Konkret spielt sich auf dem Bus folgender Ablauf ab: In einem Kommandoteil (Command) seiner Nachricht legt der Master die Funktion und die Bedeutung der Nachricht fest, d. h. will er senden oder erwartet er Daten. Im Antwortteil (Answer) kann der Slave entweder die geforderten Daten senden oder aber eine Empfangsbestätigung/Quittung (Acknowledge) auf den Bus geben, um anzuzeigen, dass er die Nachricht korrekt erhalten hat.

In der Regel werden in der Nachricht sowohl Sender- als auch Empfängeradressen mitgeliefert. Anhand der Empfängeradresse kann der Slave erkennen, dass die Daten für ihn bestimmt sind. Alle anderen Teilnehmer, die den Busverkehr ja ständig „mithören“, ignorieren diese Nachricht. Für Rundrufe, also für Nachrichten, die sich an mehrere Teilnehmer richten, können bestimmte, gemeinsame Empfängeradressen festgelegt werden. Das Mitführen der Senderadresse in der Nachricht kommt vor allem in Multi-Master-Systemen zum Tragen, damit der Slave weiß, an wen er seine Daten oder die Quittung senden soll.

Polling. Um über den aktuellen Zustand am Bus auf dem Laufenden zu sein, muss der Master natürlich kontinuierlich mit seinen Slaves Kontakt aufnehmen, und sei es nur, um zu erfahren, dass derzeit keine we-

sentlichen Nachrichten vorliegen. Diese zyklische Abfrage, das Pollen, der Teilnehmer hat eine ständige Grundbelastung des Busses zur Folge. Aber nur so lässt sich beispielsweise feststellen, ob ein Teilnehmer überhaupt noch funktionstüchtig ist. Zeitkritische Teilnehmer können auch mehrmals innerhalb des Polling-Zyklus aufgerufen werden, um Reaktionszeiten zu verkürzen. Befehle an einen oder mehrere Slaves schiebt der Masters bei Bedarf ein.

4.4 Token-Verfahren

Diese Technik – auch Token-Passing genannt – kommt ohne Master aus. Ein Token (= Sendeberechtigung) wird als möglichst kurzes Telegramm von Teilnehmer zu Teilnehmer weitergereicht. Es darf nur der senden, der den Token besitzt. Dabei bilden die einzelnen Teilnehmer einen meistens durch ihre Adresse festgelegten logischen oder physikalischen Ring (Tokenbus / Tokenring). Hat ein Teilnehmer einen Sendewunsch, so sendet er zunächst statt des Tokens das entsprechende Datentelegramm, das je nach Dienst vom Empfänger quittiert wird oder nicht. Erst danach wird der Token an den nächsten Teilnehmer übergeben.

Das Token-Verfahren kann erweitert werden. Dabei ist jeder Teilnehmer mit einem dezentralen Entscheider ausgestattet. Dieser kann bewirken, dass eine eigene Nachricht als wichtiger empfunden wird als die momentan aktuelle Nachricht auf dem Bus. In einem lokalen Speicher wird die aktuelle Busnachricht zwischengespeichert und die eigene Nachricht eingeschoben.

4.4 CSMA-Verfahren (Carrier Sense Multiple Access)

Auch bei diesem Verfahren, das durch seine Verwendung bei Ethernet weit verbreitet ist, gibt es keinen Master. Die Zugriffsberechtigung ist dezentral geregelt, alle Teilnehmer sind gleichberechtigt. Hat ein Teilnehmer einen Sendewunsch, so prüft er, ob gerade eine Übertragung läuft (Carrier Sense). Ist dies nicht der Fall, legt er sein Datentelegramm auf den Bus. Nun kann es aber – beispielsweise durch Signallaufzeiten – dazu kommen, dass andere Teilnehmer fast gleichzeitig senden. Dies führt zur Kollision der verschiedenen Datentelegramme auf dem Bus. Die Daten werden verfälscht.

Die einzelnen Varianten des Verfahrens unterscheiden sich darin, wie in einem solchen Fall weiter verfahren wird. Normalerweise stellt der Sender die Verfälschung erst nach der Übertragung fest, indem er die gesendete Prüfinformation des Telegramms mit der empfangenen kontrolliert. Als Prüfinformation sind im Allgemeinen CRC (cyclic redundancy checksum) bzw. eine Paritätsprüfung gebräuchlich. Gibt es Fehler, wird nach einer gewissen, zufallsge-

steuerten Zeit eine erneute Übertragung eingeleitet. Der Faktor „Zufall“ ist nötig, damit sich zwei konkurrierende Sender nicht fortwährend gegenseitig stören und den Bus ständig besetzen.

CSMA/CD. Beim CSMA/CD-Verfahren (CD = Collision Detection) erkennt der sendende Teilnehmer durch Mithören des eigenen Telegrammes, dass das gesendete und empfangene Signal nicht übereinstimmen. Er bricht die Übertragung unverzüglich ab, gibt somit den Bus frei und startet später einen erneuten Sendeversuch.

TSMA/CD (Transition Sense Multiple Access / Collision Detection), das beim LON zum Einsatz kommt, funktioniert im Prinzip ähnlich wie CSMA/CD. Bei steigender Buslast steigt jedoch die zufällige Wartezeit stark an. Dadurch wird ein verbessertes Hochlastverhalten erzielt.

CSMA/CA. Das Verfahren (CA = Collision Avoidance) verwenden beispielsweise CAN und EIB. Senden zwei Teilnehmer gleichzeitig ein Datentelegramm, so setzt sich der Teilnehmer mit der (mitgesendeten) höheren Priorität durch. Der andere Teilnehmer bricht seine Übertragung ab

und wiederholt sein Senden erst nach Freierwerden des Busses. Eine bitweise Arbitrierung der Nachrichtennummer funktioniert nach folgendem Prinzip: Auf dem Bus gibt es einen dominanten Pegel („0“) und einen rezessiven („1“). Die sendewilligen Teilnehmer legen nun parallel ihre Nachrichtennummer an und überprüfen das Signal auf dem Bus. Beispiel: Zwei Teilnehmer legen zuerst beide „0“ und „1“ an und empfangen ebenfalls „0“ und „1“. Beim 2. Takt legen beide Teilnehmer „1“ an. Beim 3. Takt legt Teilnehmer 1 eine „0“ und Teilnehmer 2 eine „1“ an. Die „0“ ist dominant, Teilnehmer 2 bricht daraufhin seinen Sendeversuch ab. Eine weitere Möglichkeit, um eine Kollision zu vermeiden, besteht darin, jedem Teilnehmer eine bestimmte Wartezeit zuzuweisen, die absolut oder relativ zur letzten Benutzung des Busses sein kann.

4.5 TDMA-Verfahren (Time Division Multiple Access)

Hierbei wird jedem Teilnehmer ein bestimmter Zeitschlitz innerhalb der Gesamtzykluszeit fest zugeordnet. Innerhalb dieses Zeitschlitzes sendet der Slave seine Daten

an einen Master, der wiederum einen Zeitschlitz zum Senden von Steuerdaten an die verschiedenen Slaves hat. Haben die Teilnehmer unterschiedliche Nachrichtenlängen, so können die Zeitschlitzte auch unterschiedlich lang gemacht werden. Beispielsweise kommt ein Endschalter mit nur einem Bit aus und benötigt daher eine geringere Bandbreite als ein A/D Wandler. Darüber hinaus können beim Interbus in einem zusätzlichen Zeitschlitz bei Bedarf Parameter, Steuerdaten und Messdaten übertragen werden, wobei die Gesamtnachricht auf mehrere Zyklen aufgeteilt wird. Die Vergabe des Zugriffsrechts erfolgt hierbei entsprechend dem augenblicklichen Bedarf. Hauptvorteil dieses Verfahrens ist die garantierte Gesamtzykluszeit, auch ist der Overhead vergleichsweise gering. Ein weiterer Vertreter, der dieses Verfahren mit einem zusätzlichen Master-Synchronisationstelegramm verwirklicht, ist der Sercos-Bus. Über die Mastersynchronisationstelegramme werden die Teilnehmer zu Beginn eines Zeitrahmens synchronisiert. Danach folgen die Telegramme der einzelnen Teilnehmer und das Mastertelegramm. ■

Tafel 2 Typische Kenndaten von Bussystemen (Auswahl)

Bezeichnung	Topologie	Anzahl Busteilnehmer	Übertragungsrates Leitungslänge	Zugriffsverfahren	Schnittstelle/Kodierung
Arcnet	Linie Baum	256 (davon 1...255 Master)	30 bit/s...10 Mbit/s divers	Tokenring	RS485 (NRZ)
AS-i (Aktor-Sensor-Interface)	Linie Baum	1 Master 31 Slaves (erw: 62)	167 kbit/s <100 m	Master/Slave (zyklisches Polling)	Manchester (RZ)
Bitbus	Linie Baum	1 Master 250 Slaves	62,5...1500 kbit/s <1200 m	Master/Slave (zyklisches Polling)	RS485
CAN (Controller Area Network)	Linie	32 (chipabhängig)	50...1000 kbit/s 1000...25 m	Multi-Master (CSMA/CA)	RS485 (NRZ) ISO 11898
CANopen	Linie	127	50...1000 kbit/s 1000...10 m	Master/Slave Multi-Master (CSMA/CA)	
DeviceNet	Linie	64	125...500 kbit/s 500...100 m	Master/Slave Multi-Master Peer-to-Peer	
DIN-Messbus	Linie	31	0,11...1000 kbit/s <500 m	Polling	RS485
EIB (European Installation Bus)	Linie Baum	64x12x15	9,6 kbit/s <700 m	CSMA/CA bitweise Arbitrierung	Busankoppler
FIP/WorldFIP (Factory Information Protokoll)	Linie	256	31,25...2000 kbit/s <2000 m	Delegated Token (TDMA)	Manchester
IEC-Feldbus	Linie	32	31,25...2500 kbit/s <1900 m	Delegated Token	eigene
Interbus	Doppel-Ring	512 Fernbus: 254	500 kbit/s <12800 m, 400 m / Segm.	Master/Slave (TDMA, verteiltes Schieberegister)	RS485 (NRZ)
LON (Local Operating Network)	frei	127x255	0,6...1250 kbit/s <1300 m (<2000 m)	TSMA (CSMA/CD)	eigener Transceiver RS485
P-Net	Ring Linie	128 (davon <32 Master)	76,8 kbit/s <1200 m	Master/Slave Multi-Master (Token)	RS485 IS16 (Ex)
Profibus-DP (Process Field Bus - Decentralised Peripherals)	Linie	126	9,6...12000 kbit/s 1200...100 m	Master/Slave Token passing	RS485 (NRZ)
Profibus-FMS (Fieldbus Messaging Service)	Linie	127	93,75...500 kbit/s 1200...200 m		RS485 (NRZ)
Profibus-PA (Process Automation)	Linie Baum	32	31,25 kbit/s <1900 m	Master/Slave	MBP (Manchester Coded, Bus Powered)
Sercos	LWL-Ring	254	2...16 Mbit/s 50/250 m pro Teilnehmer	Zeitraumen (TDMA)	NRZI-kodierte HDLC-Protokolle