

BUSSYSTEME

Bus: Gemeinsames Medium zur Datenübertragung zwischen mehreren Teilnehmern

Anwendung:

- Lokale Busse
- Backplane Busse
- Serielle Verbindung von Peripheriegeräten (USB, FireWire, etc.)
- Feldbusse

Parallel vs Serial

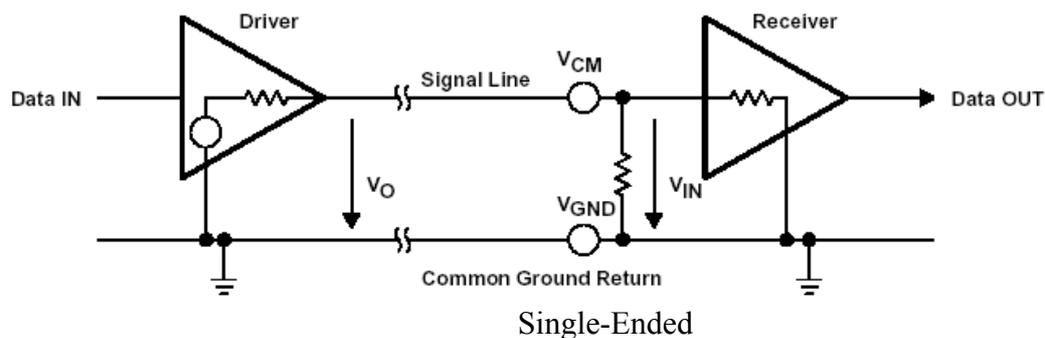
Vorteile der parallelen Übertragung:

- Kein Zeitverlust durch Umsetzung parallel/seriell
- Einzelne Leitungen können zu Kontrollzwecken hinzugefügt werden -> kürzere Reaktionszeiten
- Weite Verbreitung als sog. Backplanebusse

Vorteile der seriellen Übertragung:

- Keine Probleme mit line to line skew
- Geringere Kabelkosten, längere Kabel möglich
- Geringe Leiterplattenfläche

Single-Ended vs Differential



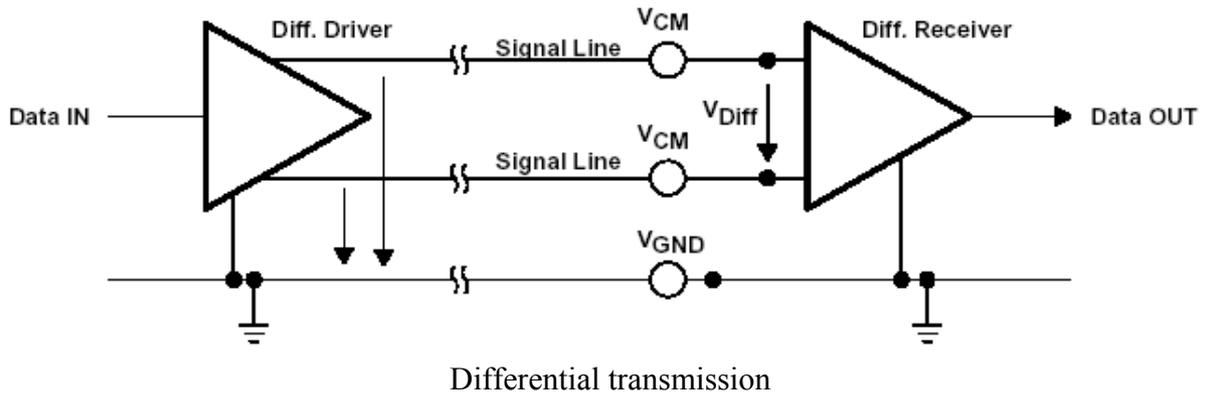
Vorteile:

- Geringe Kosten (Leitungen, Stecker)

Nachteile:

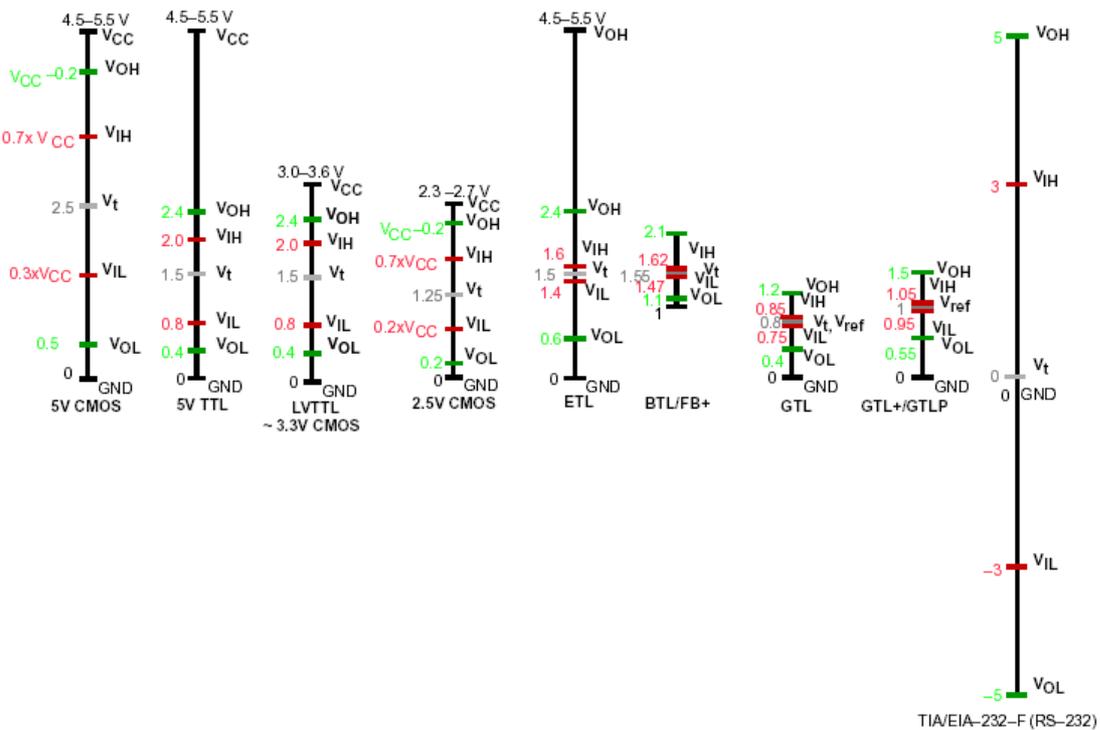
- Geringe Störimmunität (Signalleitung und Rückleitung bilden Schleife mit relativ großer Fläche)
- Empfindlich gegen Potentialverschiebung zwischen Sender und Empfänger
- Wirkt als Störquelle
- Relativ großes Übersprechen zwischen mehreren Signalleitungen

Anwendung: Backplanebusse, lokale Busse (TTL, LVTTTL); RS232

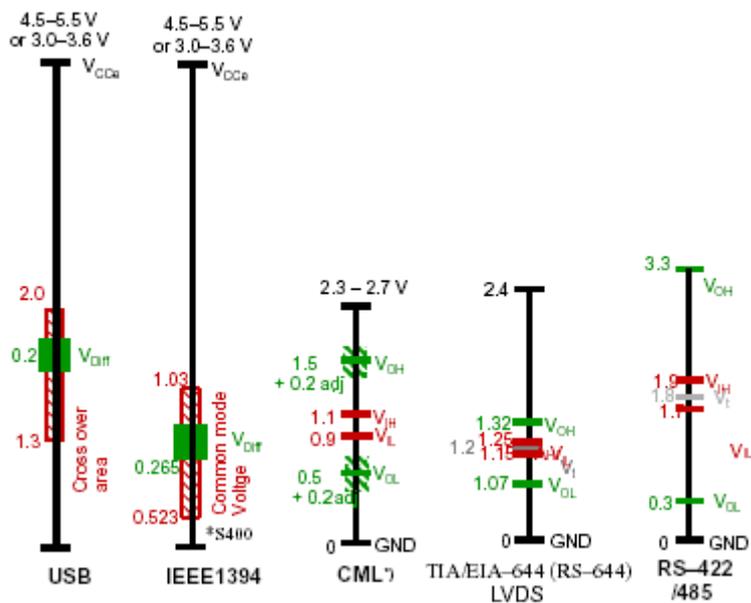


Vorteile:

- Geringe Stömpfindlichkeit
- Unempfindlich gegen Potentialverschiebung zwischen Sender und Empfänger
- Geringe Abstrahlung

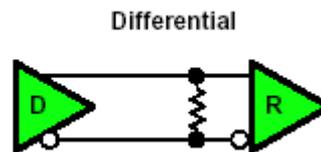
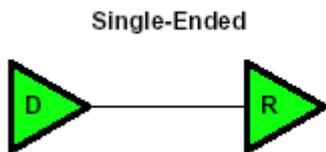


Schaltpegel verschiedener Single-Ended Transmission Standards

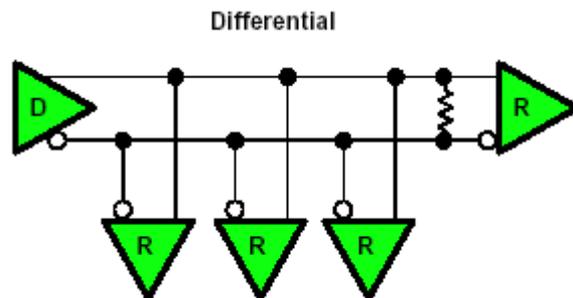
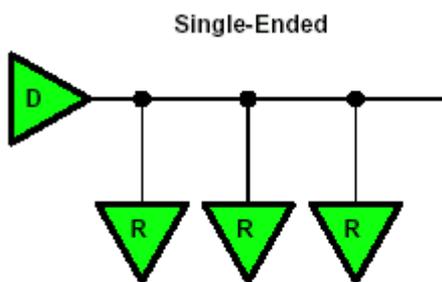


Schaltpegel verschiedener Differential Transmission Standards

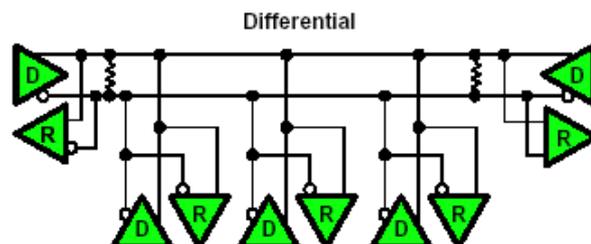
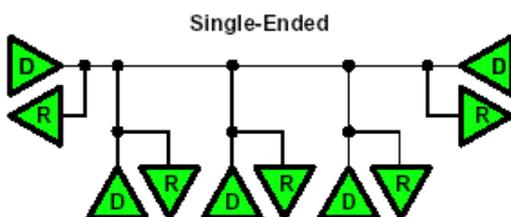
Topologien



Point to point

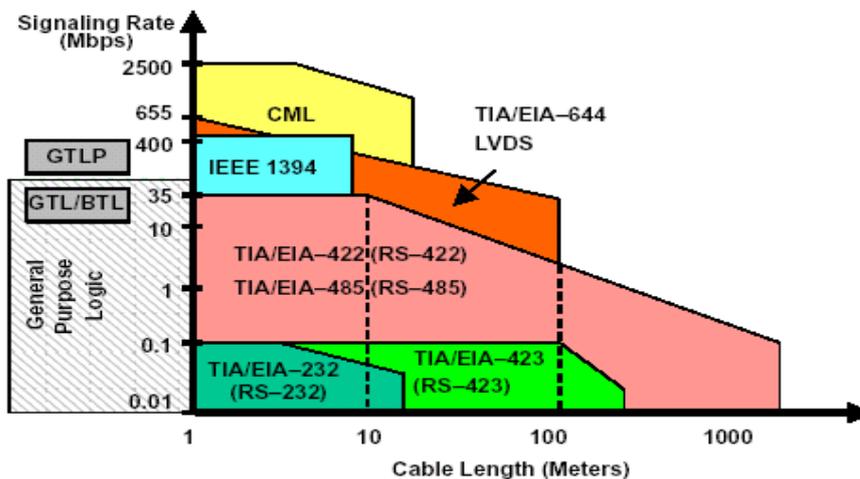


Multidrop

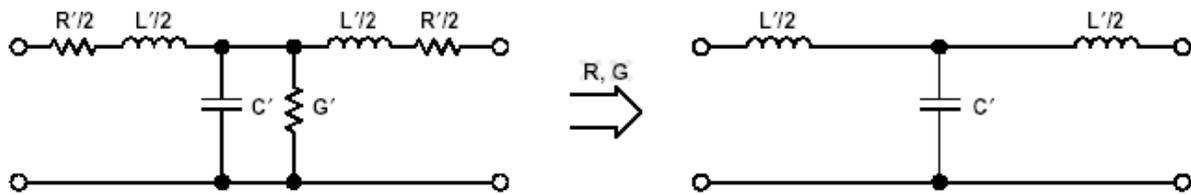


Multipoint

TYPE	TRANSMISSION MODE	DATA RATE PER LINE	DATA RATE PER DEVICE	DISTANCE	STANDARD	PRODUCT FAMILY	WEB PAGE http://www.ti.com/sc
Serial	Multipoint	25/50 Mbps		1.5 m	IEEE1394-1995	IEEE1394 Backplane	/1394
		100 – 400 Mbps		4.5 m	IEEE1394-1995/P1394.a	IEEE1394 Cable	
		12 Mbps		5 m	USB 1.1	USB	/usb
		35 Mbps		10 m (1200 m)	TIA/EIA 485 (ISO8482)	TIA/EIA 485	/docs/products/msp/datatran/index.htm
		200 Mbps		0.5 m	In definition	LVDM	
	Multidrop	10 MBps		10 m (1200 m)	TIA/EIA 422 (ITU-T V.11)	TIA/EIA-422	/docs/products/msp/datatran/index.htm
		200/100 Mbps	4 ch: 800/400 Mbps	0.5 m/10 m	TIA/EIA-644/TIA/EIA-644 (LVDS) /in definition	LVDS/LVDM	
Point-to-point	512 Kbps		20 m	TIA/EIA-232 (ITU-T V.28)	TIA/EIA-232	/docs/products/msp/datatran/index.htm	
Serial	Point-to-point	400/200 Mbps	4 ch: 1600/800 Mbps	1 m/ 10 m	TIA/EIA-644/TIA/EIA-644 (LVDS)	LVDS	/docs/products/msp/datatran/index.htm
Parallel-to-serial Serial-to-parallel	Point-to-point	455 Mbps	4 ch: 1.83 Gbps	< 10 m	TIA/EIA-644/TIA/EIA-644 (LVDS)	LVDS SerDes/FlatLink	/serdes
		1.25 Gbps	1.25 Gbps full duplex	< 10 m	IEEE P802.3z	Gigabit Ethernet	
		2.5 Gbps	2.5 Gbps full duplex	< 10 m	IEEE P802.3z extended 2 Gbps	Serial Gigabit CMOS	
Parallel	Multipoint	35 Mbps		10 m (1200 m)	TIA/EIA-485 (ISO8482)	TIA/EIA-485	/docs/products/msp/datatran/index.htm
		40/20 MHz	9 ch: 360/180 Mbps	12 m/25 m	SCSI (ISO/IEC9316)	SCSI	
		40 Mbps	9 ch: 360 Mbps	12 m	LVD-SCSI (1142-D SPI-2)	LVD-SCSI	
		200 s/100 Mbps		0.5 m/10 m	In definition	LVDM	
		33/66 MHz		0.2 m	PCI Compact	PCI	/pci
		33/66 MHz		0.2 m	PCI	PCI	
		4 MHz CLK	16 ch: 64 Mbps	10 m	IEEE Std 1284-1994	AC1284, LVC161284, LV161284	/logic
		20 MHz CLK	32 ch: 640 Mbps	0.5 m	CMOS, JESD20, TTL IEEE 1014-1987	AC, AHC, ABT	/logic
		33 MHz CLK	20 ch: 660 Mbps	0.5 m	LVTTTL as stated in JEDS8-A, June 1994, IEEE 1014-1987	LVTH, ALVT	/alvt /lvt
		40 MHz CLK	16 ch: 640 Mbps	0.5 m	VME64 Standard ANSI/ITA1-1991	ABTE	/logic
		60 MHz CLK	16 ch: 960 Mbps	0.5 m	IEEE Std 1194.1-1991	BTL/FB+	
		60 MHz CLK	18 ch: 1.08 Gbps	0.5 m	JESD8-3	GTL/GTL+	/gtl
		100 MHz CLK	18 ch: 1.8 Gbps	0.5 m	JESD8-3	GTLF	/gtlf
200 MHz CLK	18 ch: 3.6 Gbps	0.1 m	EIA/JESD8-8, EIA/JESD8-9	SSTL	/logic		



Physikalische Eigenschaften von Busleitungen



L' = Characteristic Inductance Per Unit Length (nH/cm)
 C' = Characteristic Capacitance Per Unit Length (pF/cm)
 R' = Characteristic Resistance Per Unit Length (Ω /cm)
 G' = Characteristic Conductance Per Unit Length (S/cm)

Complex Line Impedance

$$\vec{Z}_0 = \sqrt{\frac{j\omega L' + R'}{j\omega C' + G'}}$$

With $R' \ll j\omega L'$ and $G' \ll j\omega C'$:

Line Impedance $Z_0 = \sqrt{\frac{L'}{C'}}$ (real number !)

Propagation Time $\tau = \sqrt{L' \times C'}$

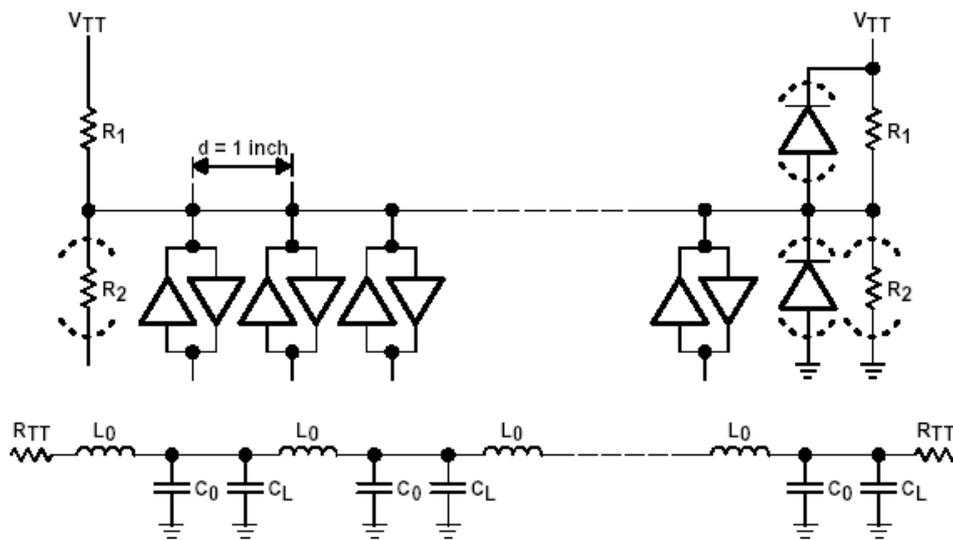
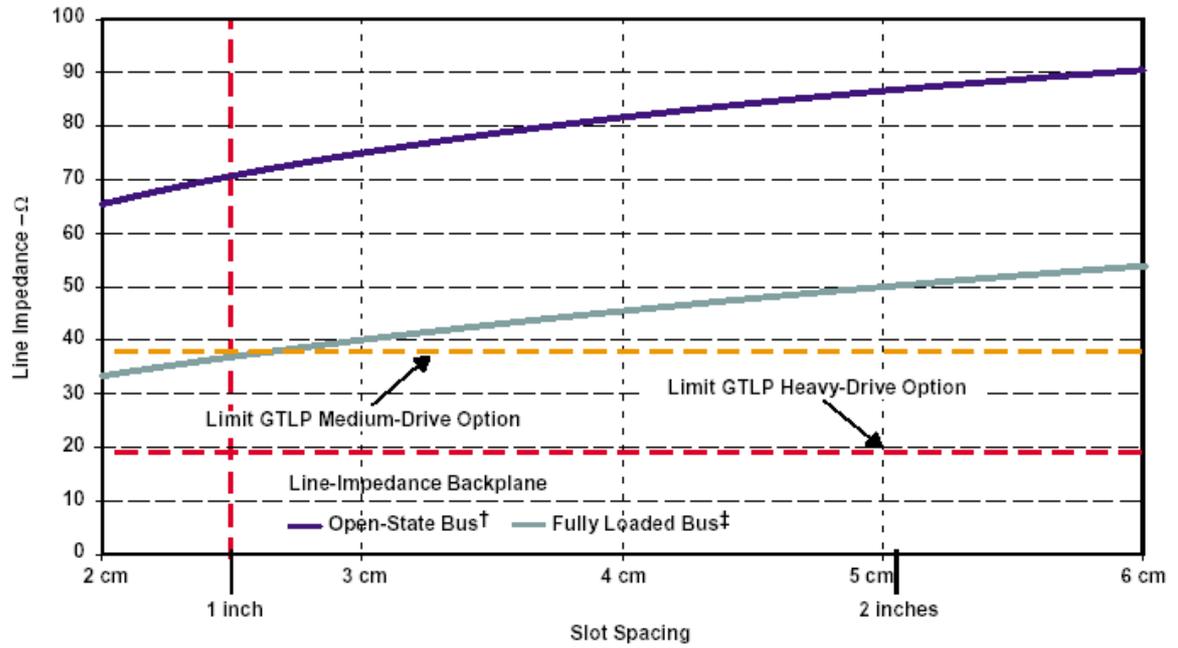


Bild : Zusätzliche kapazitive Last durch Busmodule

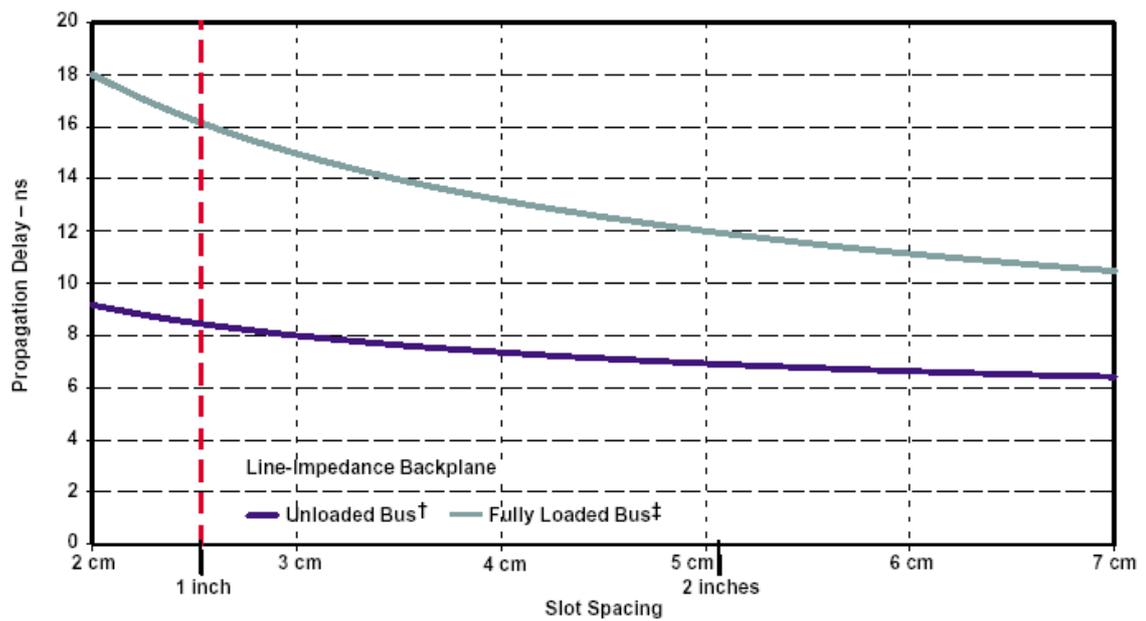
Table 1. Comparison of PCB Line and Bus Line (Slot Distance = 1 Inch)

LINE PARAMETER	OPEN LINE		LOADED LINE (BUS)
	PCB	BUS	
Inductive layer, L_0	6.5 nH/cm	6.5 nH/cm	6.5 nH/cm
Capacitive layer, C_0	0.4 pF/cm	0.4 pF/cm	0.4 pF/cm
Connector	Not applicable	2 pF per slide-in module	2 pF per slide-in module
Feed line to backplane side of connector	Not applicable	$\ll 1$ pF	$\ll 1$ pF
Feed line to module side of connector	Not applicable	Not applicable	~ 1 pF
Input capacity of receiver	Not applicable	Not applicable	9 pF
Additional capacity load, C_L	Not applicable	2 pF/2.54 cm	10 pF/2.54 cm
Capacitive load per cm	0.4 pF	1.2 pF	4.73 pF
Line impedance, Z_0	127 Ω	74 Ω	37 Ω
Propagation delay time of signal, τ	5.1 ns/m	8.8 ns/m	17.5 ns/m



† Assuming a capacitive load of 2 pF per slot (connector only)
 ‡ Assuming a capacitive load of 10 pF per slot (complete module)

Bild : Einfluß der Steckplatzdistanz auf die Leitungsimpedanz



† Assuming a capacitive load of 2 pF per slot (connector only)
 ‡ Assuming a capacitive load of 10 pF per slot (complete module)

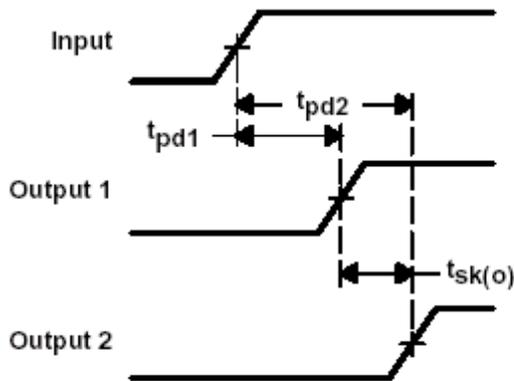
Bild : Einfluß der Steckplatzdistanz auf die Laufzeit

Eigenschaften von Sender- und Empfängerbausteinen

Skew

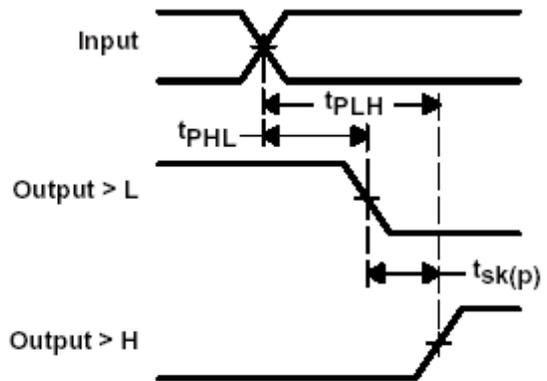
Output-to-Output Skew

The maximum difference of the delay between the fastest and slowest output driver within one circuit [$t_{sk(o)} \sim 0.5 \text{ ns}$]



Pulse Skew

The maximum difference of the delay between rising- and falling-edge transition of one single output pin of one circuit [$t_{sk(p)} \sim 0.8 \text{ ns}$]



Process Skew

The maximum skew between outputs of various circuits of the same function type [$t_{sk(pr)} \sim 1 \text{ ns}$]

Bild : Skew Definition

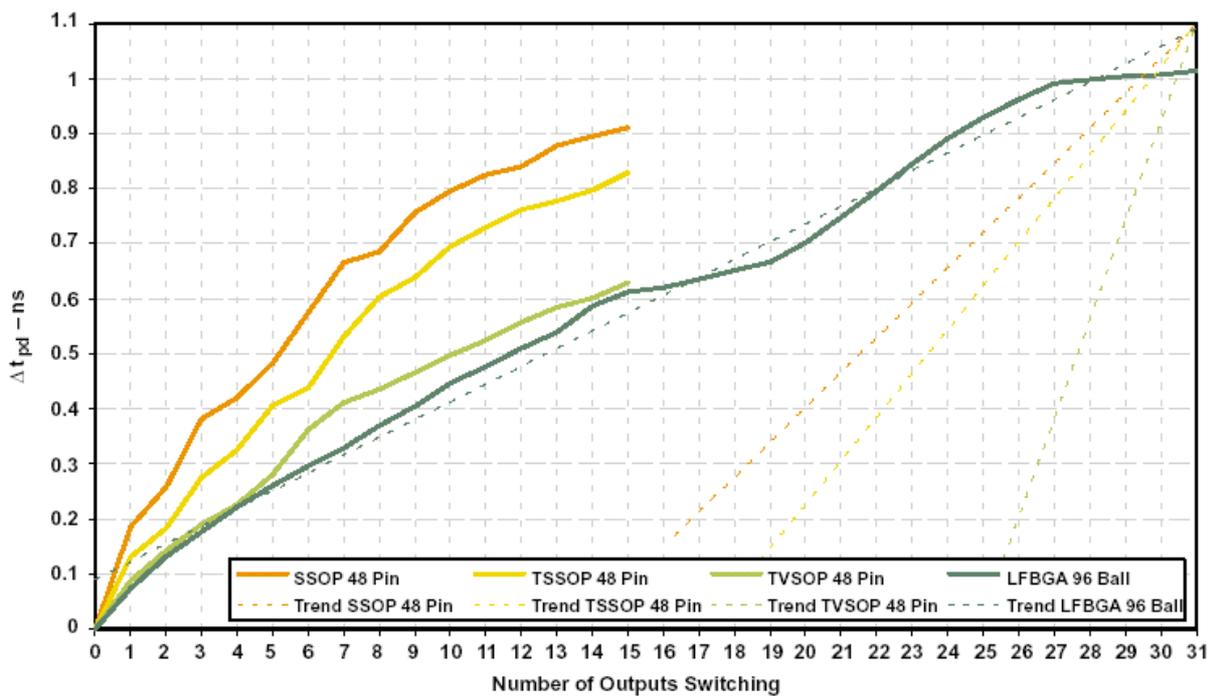


Bild : Vergrößerung von t_{pD} durch gleichzeitiges Schalten mehrerer Ausgänge

Setup und Hold Time

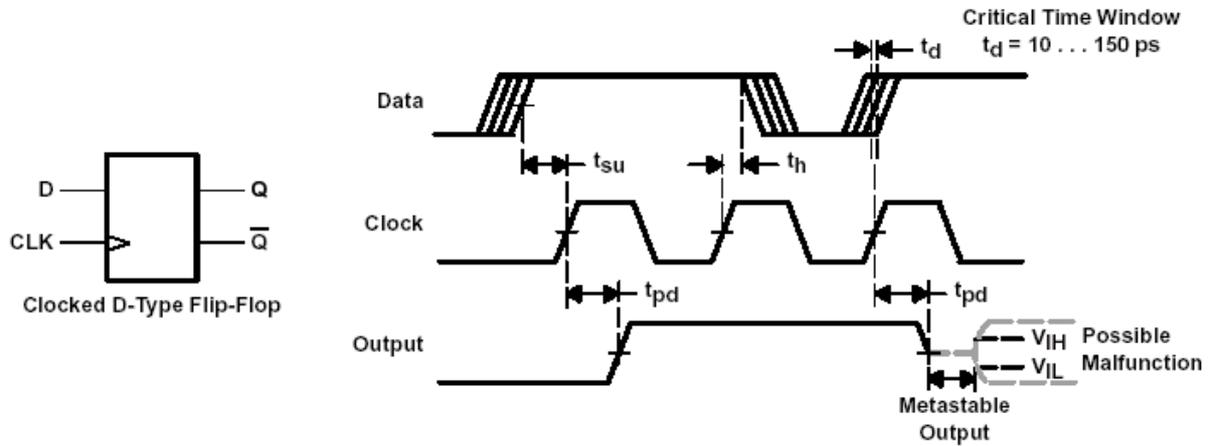


Bild : Setup und Hold Time

Central Synchronous Clock Distribution

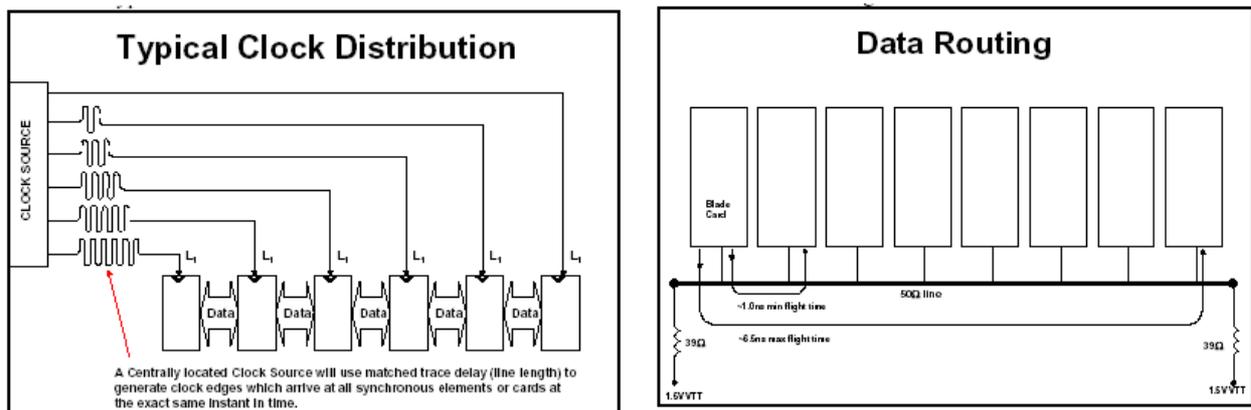
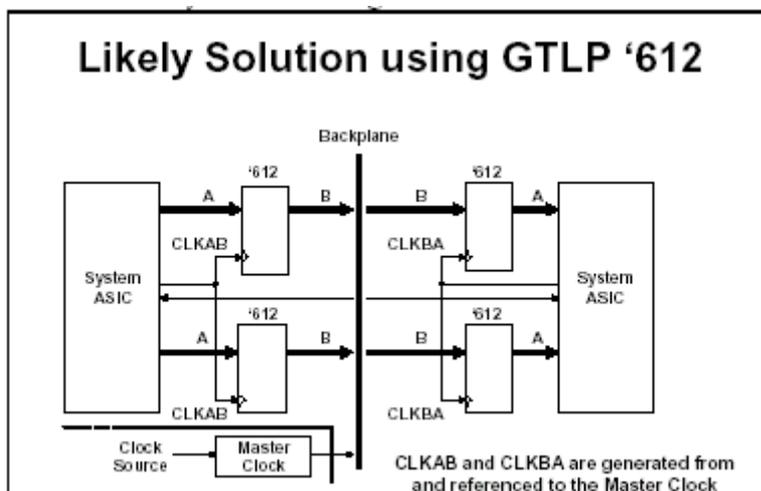


Bild : Central Synchronous Clock Distribution

Die Taktflanken müssen zum selben Zeitpunkt an allen Teilnehmern ankommen.



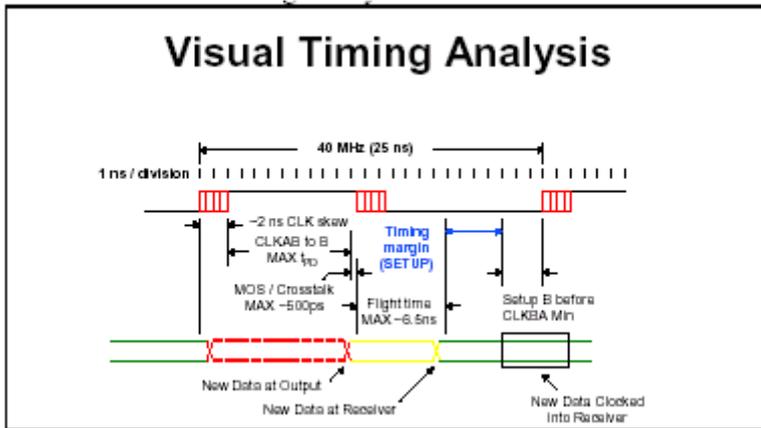


Bild : Setup Time

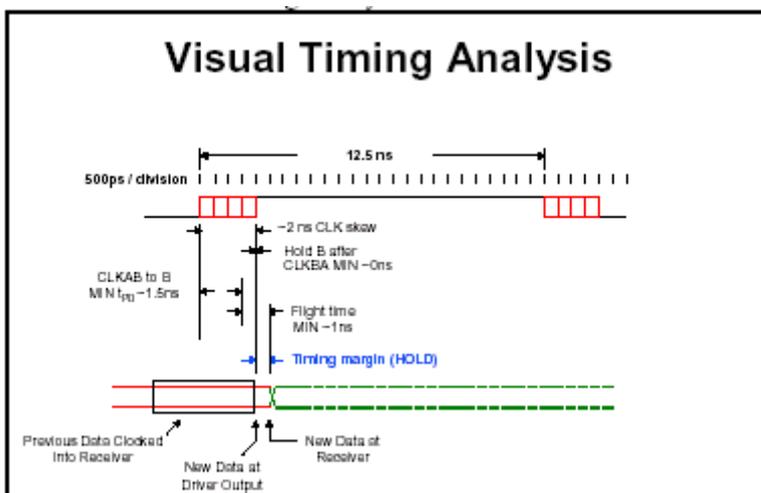
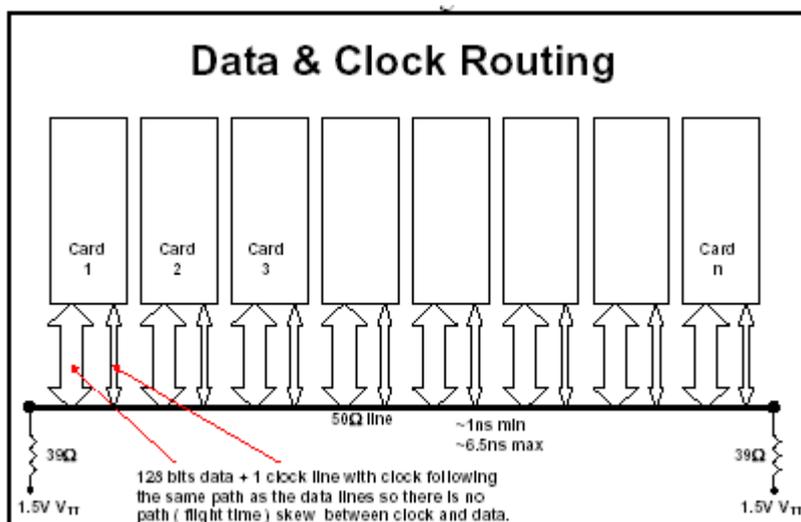
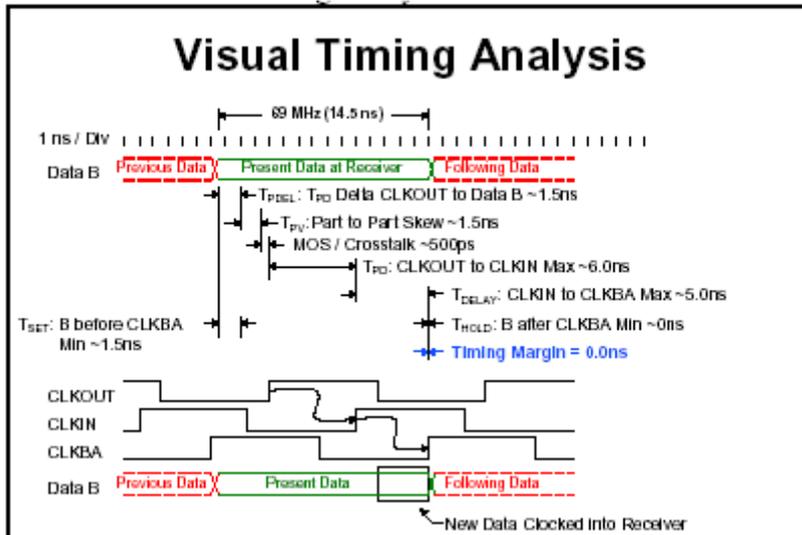


Bild : Hold Time

Source Synchronous Clock Distribution





Bus Hold Circuits

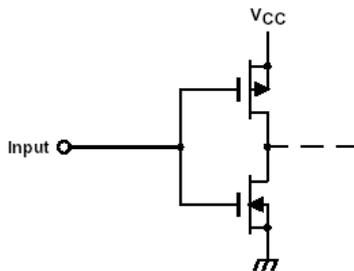


Figure 1. Input Stage of a CMOS Circuit

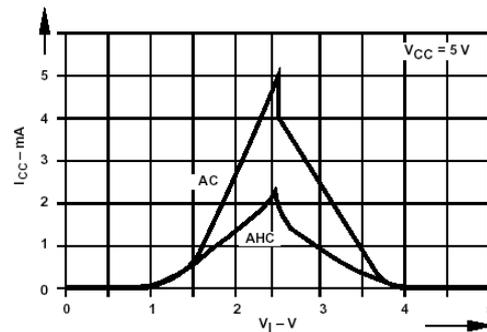


Figure 2. Power Consumption of CMOS Input Stages vs Input Voltage

16Bit Bustreiber in AC Technologie mit $V_{cc}=5\text{V}$; $I_{cc}=16*5\text{mA}=80\text{mA} \rightarrow 400\text{mW}$

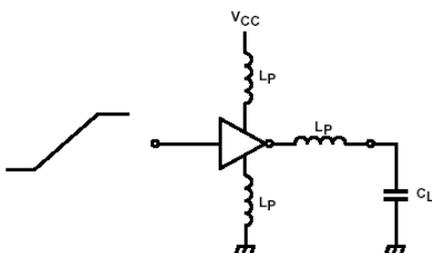


Figure 3. Parasitic Components Causing Circuit Oscillation

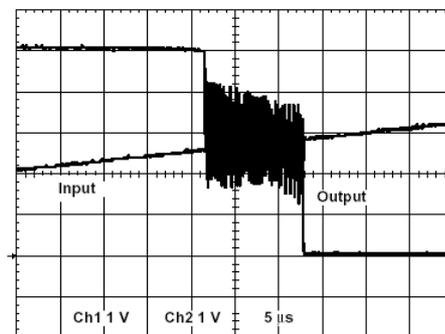
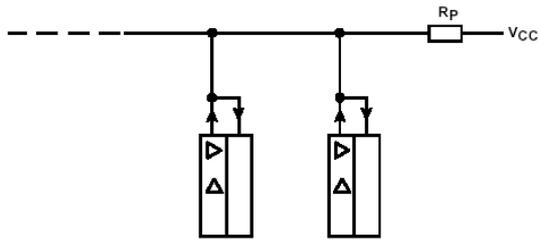


Figure 4. Oscillation at the Output of a CMOS Circuit

Schaltet ein Ausgang in den hochohmigen Zustand, ergibt sich mit $I_{oz}=10\mu\text{A}$ und $C_s=20\text{pF}$ eine Spannungsdrift von $dV/dt=I_{oz}/C_s=10\mu\text{A}/20\text{pF}=0.5\text{V}/\mu\text{s}$; $dV_{max}=1\text{V} \rightarrow t_{max}=2\mu\text{s}$
 Liegen mehrere Bausteine an der Busleitung ändert sich an der Spannungsdrift nichts, wenn man annimmt, dass alle I_{oz} und C_s gleichen Wert haben.

Pull-up Widerstände



$$R_p \leq \frac{t_r}{2.2 \cdot C_s \cdot n}$$

n: Anzahl der angeschlossenen Bausteine

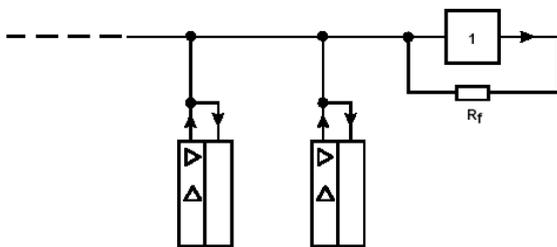
Für $\Delta t / \Delta V < 10 \text{ ns/V}$ ($t_r = 50 \text{ ns}$), $n = 10$, $C_s = 20 \text{ pF}$ ist

$$R_p \leq 110 \Omega$$

Nachteil: hoher Stromverbrauch, großer Treiberstrom notwendig.

Figure 7. Creating a Defined Level Using Pullup Resistors

Bus Hold Circuit



R_f ist so zu dimensionieren, dass ein ausreichender Spannungspegel auf der Leitung entsteht.

$$R_f \leq \frac{V_r}{I_{oz} \cdot n}$$

Mit $I_{oz} = 10 \mu\text{A}$, $n = 10$ und $V_r = 1 \text{ V}$ ergibt sich

$$R_f \leq 10 \text{ k}\Omega$$

Figure 8. Bus With Bus-Hold Circuit

Businterface mit integriertem Bus-Hold

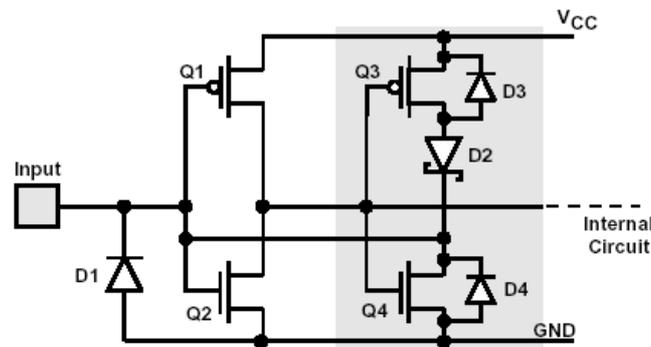
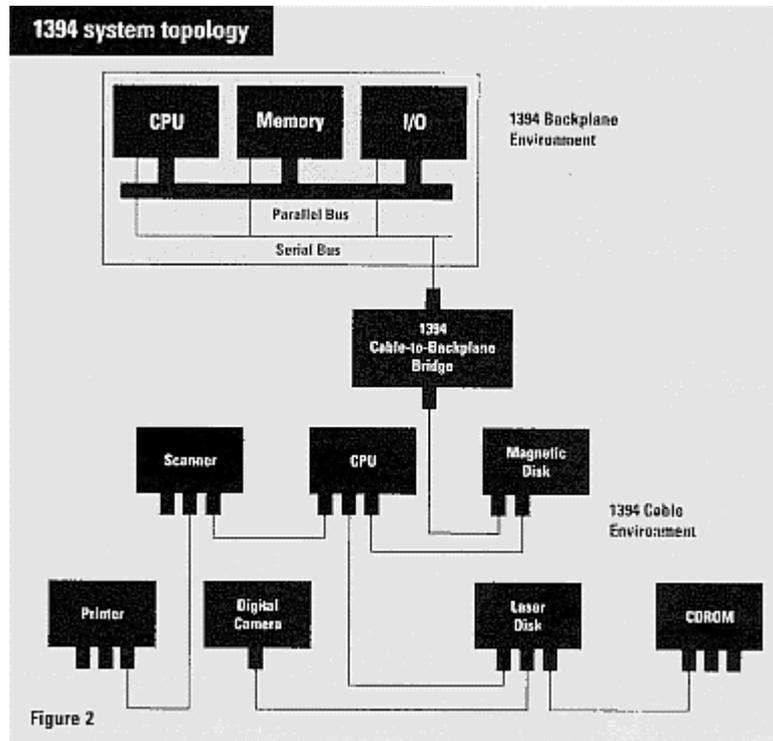


Figure 9. Simplified Circuit Diagram of Bus-Hold Circuits

IEEE 1394 (FireWire, iLINK)

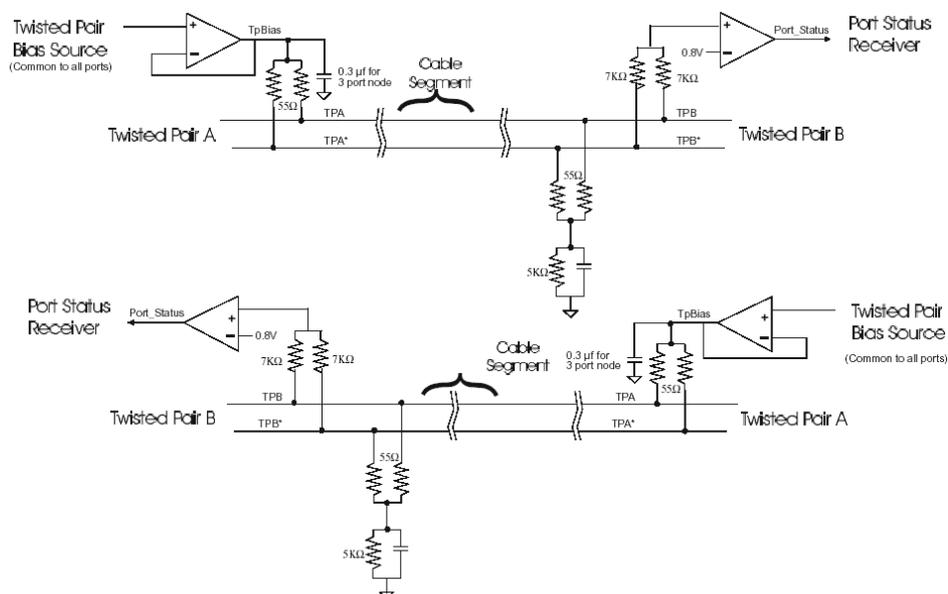
Anwendung:

- Unterstützung und Erweiterung zu parallelen Backplane-Bussen (z.B.: VME, FB+)
- Punkt-zu-Punkt-Verbindung (virtueller Bus) von Peripherie- und A/V-Geräten

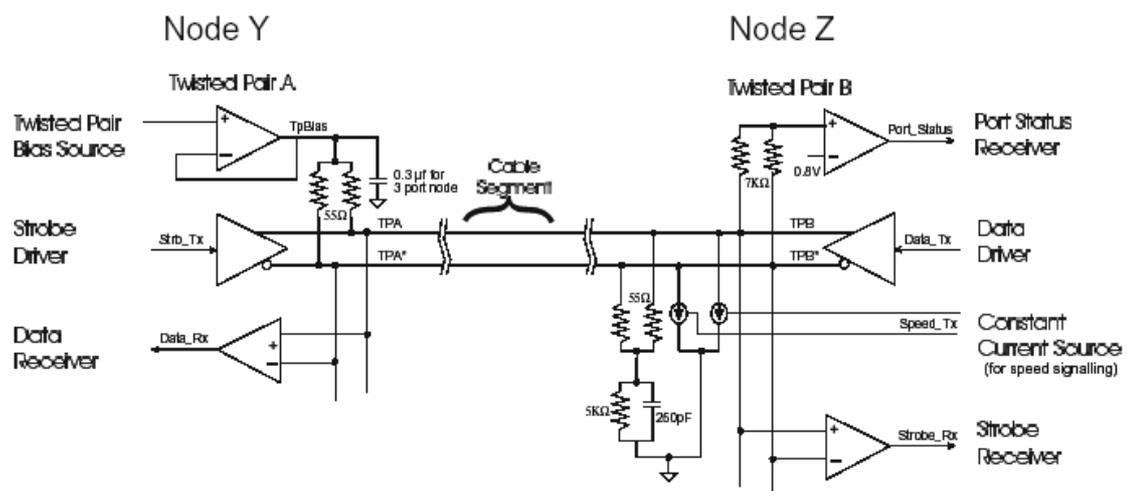
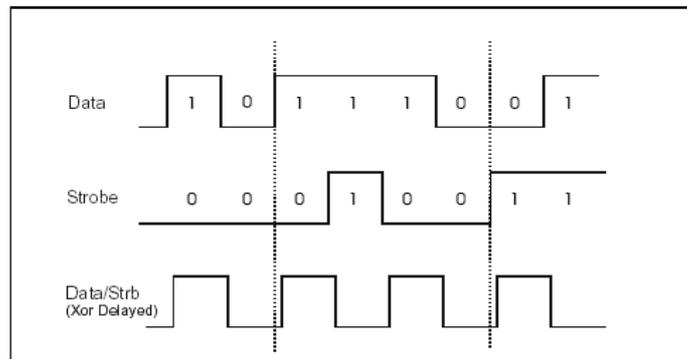


Signaldarstellung:

- Common mode signaling; $V_{CM} > 1V$: Gerät vorhanden



- Differential mode signaling; Arbitrierung und Datentransfer; Signalhub 175-265mV; NRZ, Data/Strobe encoding.



Eigenschaften :

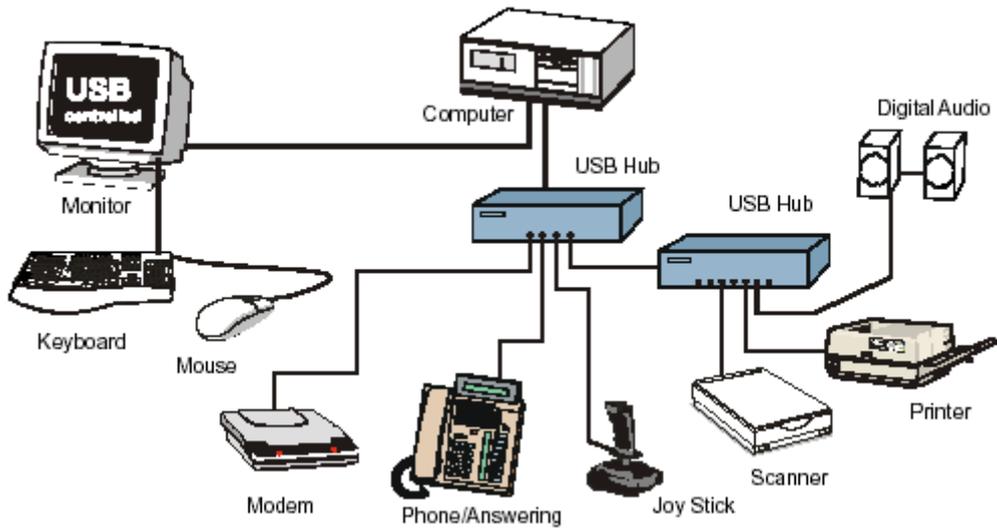
- 25 und 50 Mbps Datenrate für Backplane-Anwendungen
- 100/200/400 Mbps Datenrate für I/O-Anwendungen
- Hinzufügen und Entfernen von Geräten während des Betriebes möglich
- Garantierte Bandbreite für Echtzeitanwendungen (z.B. Video)
- Kabel: 2-fach twisted pair, geschirmt; Stromversorgung wird optional mitgeführt.

Universal Serial Bus (USB)

Serieller Bus zum Anschluß von Peripheriegeräten an den PC als Ersatz für die verschiedenen traditionellen Schnittstellen (COM, Centronics, Game/Midi-port, etc.)

Eigenschaften:

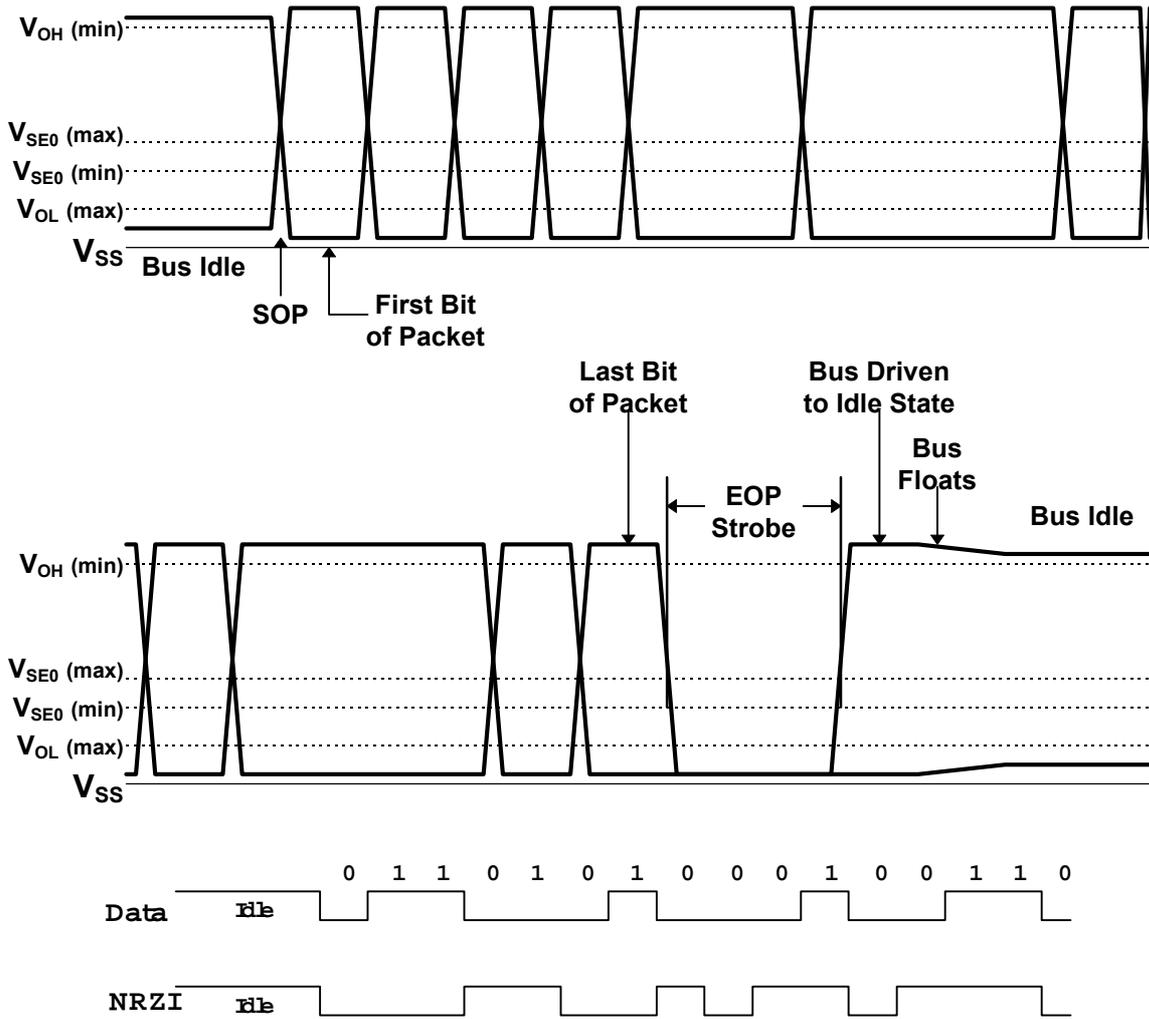
- Topologie: Tiered Star
- Datenrate: 1.5Mbps (low speed) und 12Mbps (full speed)
- Hinzufügen und Entfernen von Geräten während des Betriebes möglich
- Kabel: twisted pair; Stromversorgung (5V) wird mitgeführt.



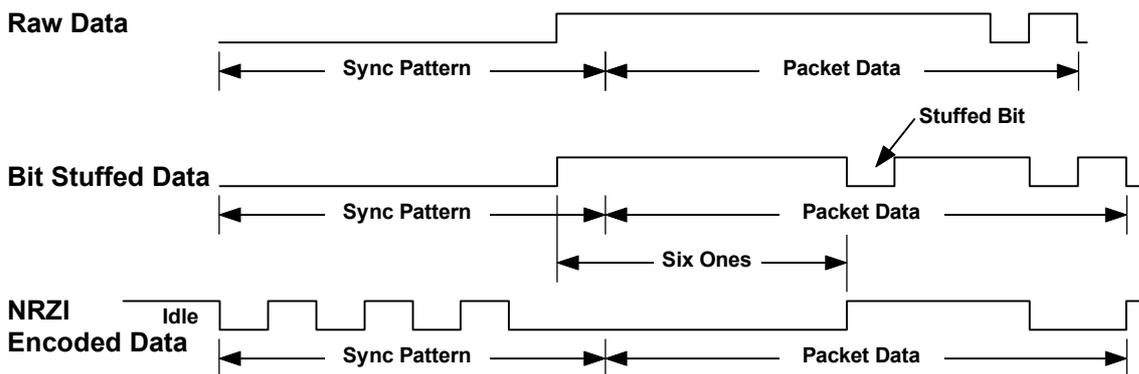
Signaldarstellung:

Differential mode signaling; NZRI, Bit Stuffing (nach sechs "1"bits wird ein "0"bit eingefügt; zur Sicherstellung der Taktrückgewinnung).

Bus State	Signaling Levels	
	From Originating Driver	At Receiver
Differential "1"	$(D+) - (D-) > 200 \text{ mV}$ and $D+$ or $D- > V_{se0}$	
Differential "0"	$(D+) - (D-) < -200 \text{ mV}$ and $D+$ or $D- > V_{se0}$	
Idle ('J' State): Low Speed Full Speed	Differential "0" Differential "1"	
Non-idle ('K' State): Low Speed Full Speed	Differential "1" Differential "0"	
Start of Packet (SOP)	Data lines switch from 'J' to 'K' State	
End of Packet (EOP)	$D+$ and $D- < V_{se0}$ for 2 bit times ¹ followed by a 'J' for 1 bit time	$D+$ and $D- < V_{se0}$ for > 0.6 bit times ² followed by a 'J' State
Disconnect (Upstream only)	(n.a.)	$D+$ and $D- < V_{se0}$ for $\geq 2.5 \mu\text{s}$
Connect (Upstream only)	(n.a.)	$D+$ or $D- > V_{se0}$ for $\geq 2.5 \mu\text{s}$
Reset (Downstream only)	$D+$ and $D- < V_{se0}$ for $\geq 10 \text{ ms}$	$D+$ and $D- < V_{se0}$ for $\geq 2.5 \mu\text{s}$ (must be recognized by $5.5 \mu\text{s}$)
Resume	From a suspended state, data lines switch from 'J' to static 'K' State. Host ends resume by sending a low speed EOP and returning the data lines to the 'J' state	



Data Encoding Sequence:



Protokoll

USB ist ein "polled bus". Übertragungen finden immer zwischen dem Host und einem USB-Teilnehmer statt; Initiator der Übertragung ist immer der Host-Controller.

TIA/EIA-232 (RS232)

Traditionelle serielle Schnittstelle zwischen Computer und Peripherie.

Eigenschaften:

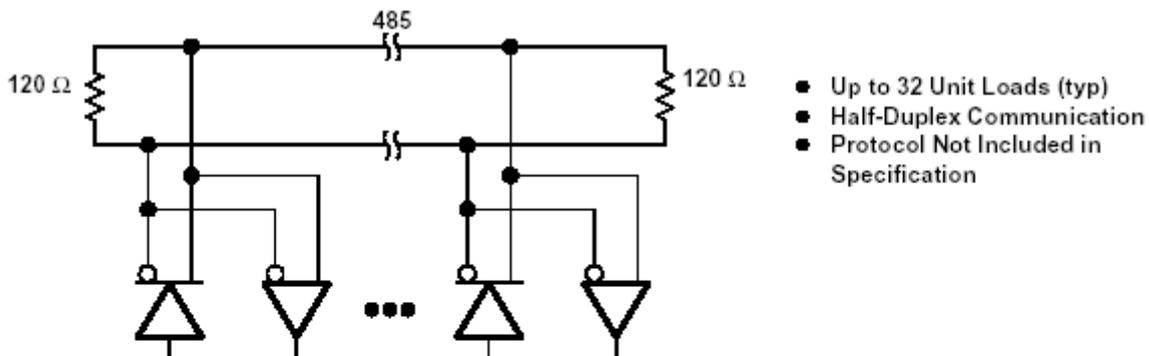
- Signaldarstellung: Single ended; $\pm(5V\div 15V)$
- Leitungslänge durch max. zulässige Kapazität von 2500pF auf ca. 20m begrenzt
- Max. Signalanstiegsgeschwindigkeit $30V/\mu s \rightarrow$ Reflexionen vernachlässigbar
- Zusätzlich zu den Datensignalen (RxD und TxD) sind Steuer- und Handshake-signale definiert (data carrier detect DCD, data set ready DSR, request to send RTS, clear to send CTS, data terminal ready DTR, ring indicator RI).
- Neben den elektrischen Spezifikationen sind auch mechanische Eigenschaften und Steckerbelegung sowie das Protokoll (ITU-T V.24) festgelegt.

TIA/EIA-422 (RS422)

Serielle Schnittstelle mit differentieller Signaldarstellung für Multidrop-Topologie. Es sind nur die elektrischen Eigenschaften spezifiziert. Mechanische Eigenschaften sowie ein Protokoll sind nicht festgelegt.

TIA/EIA-485 (RS485)

Sehr ähnlich TIA/EIA-422, jedoch für Multipoint-Topologie (Treiber kann hochohmigen Zustand annehmen). Sehr störfest, daher Standard in industriellen Anwendungen (Feldbusse).



KEY PARAMETERS	SPECIFICATION LIMITS
Maximum common-mode voltage	-7 V to 12 V
Receiver input resistance	12 k Ω minimum
Receiver sensitivity	± 200 mV
Driver load	60 Ω
Driver output short-circuit limit	250 mA to -7 V to 12 V

SCSI (small computer systems interface)

Computerperipheriebus mit hohem Datendurchsatz. 8-bit Datenbus plus 1bit parity (Wide SCSI 16-bit Datenbus plus je 1bit parity pro byte). Zusätzlich 9 bit breiter Steuerbus zur

Synchronisierung, Arbitrierung und Adressierung. Maximale Teilnehmerzahl inclusive Host-Controller ist acht.

- Erste Version mit Single-ended-Signalen mit TTL-Pegel; Datenrate bis 10Mbps.
- Nächste Generation HVD-SCSI (high voltage differential) mit TIA/EIA-485; Datenrate bis 40Mbps bei max. 12m Leitungslänge.
- Weiterentwicklung LVD-SCSI (low voltage differential) verwendet TIA/EIA-644

Das Protokoll ist in ISO/IEC 9316 festgelegt.

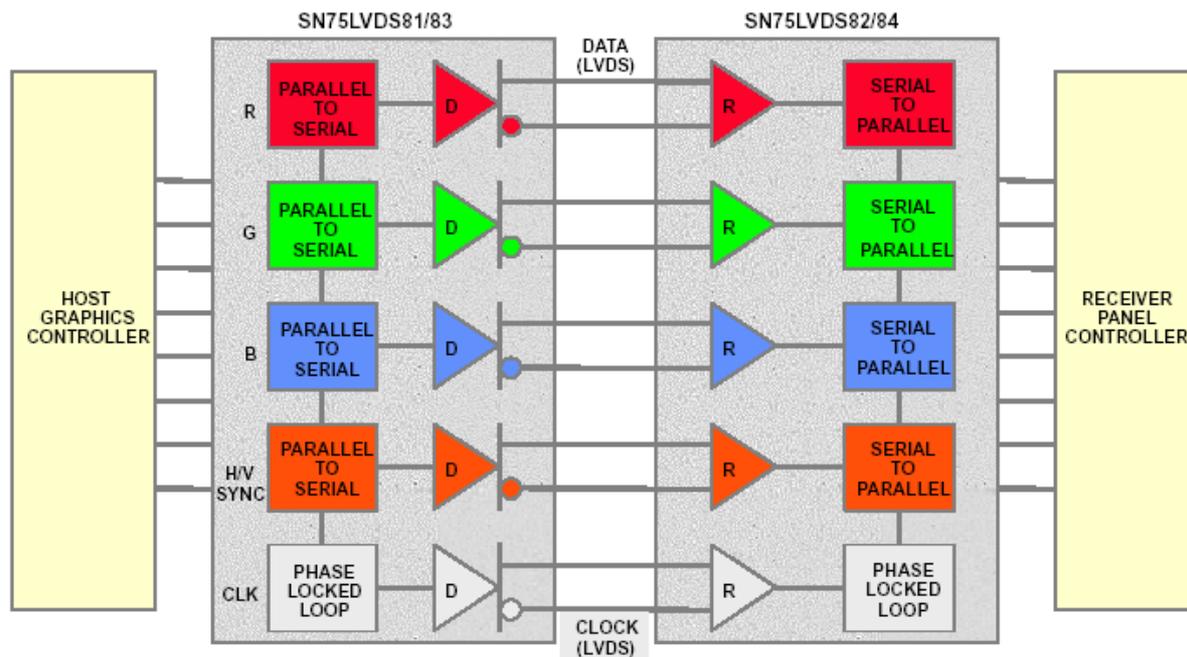
TIA/EIA-644 (LVDS low voltage differential signaling)

Upgrade für TIA/EIA-422. Durch Verkleinerung des Spannungshubes auf ca. 300mV (Empfindlichkeit des Empfängers 50mV) an einem 100Ohm Abschlusswiderstand werden eine wesentlich höhere Datenrate (bei gleicher Flankensteilheit kürzere Signalübergangszeiten) sowie geringere Störabstrahlung erreicht. Weiters ist die Stromaufnahme wesentlich geringer. Die Treiber arbeiten im Gegensatz zu TIA/EIA-422 als Stromquellen, sodaß die Stromaufnahme gleichmäßiger ist und daher durch die Schaltvorgänge weniger Störungen verursacht werden.

LVDM (LVDS Multipoint): Erweiterung von LVDS für Multipointtopologie; Upgrade für TIA/EIA-485. Die Stromergiebigkeit der Treiber ist gegenüber LVDS verdoppelt, da hier die Leitung an beiden Seiten abgeschlossen werden muß und der Lastwiderstand sich daher auf 50Ohm halbiert.

LVDS Serdes (serializer deserializer)

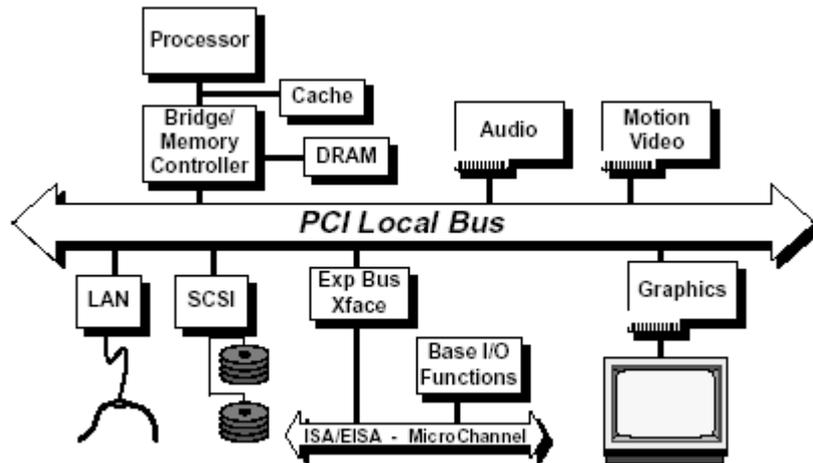
Verbindung zweier paralleler Busse über serielle Datenpfade in LVDS-Technologie.



Der Takt des parallelen Busses wird mit übertragen. Die PLL bedingt Grenzen dieses Taktes (z.B. 31MHz bis 65MHz).

PCI Local Bus

Derzeit Standard Local Bus in PCs. Als CompactPCI (anderer Formfaktor) auch in industriellen Anwendungen häufig eingesetzt.



- 32bit Adressraum
- 32bit/64bit Datenbus
- 33MHz/66MHz Bustakt → 4Gbps
- CMOS Treiber; TTL Pegel; 5V, 3.3V
- Reflected wave switching (gültiger Signalpegel entsteht erst durch Reflexion am offenen Busende) → kein Busabschluß, geringe Buslänge
- Prozessorunabhängig
- Multimaster; Central Arbitration (Request, Grant)
- Burst mode, variable Länge
- Speicher-, I/O- und Konfigurationsadressraum (Speicher- und I/O-Adressraum sowie IRQs werden erst beim Hochfahren (PCI-BIOS) festgelegt).

General Purpose Interface Logic

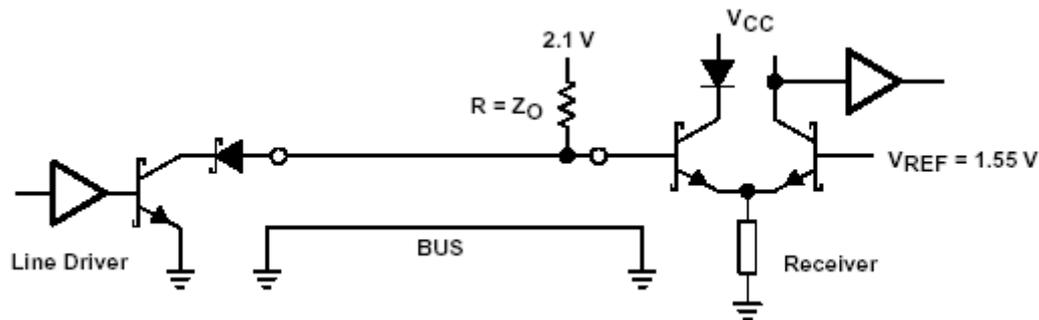
Traditionelle Technologie für Backplane Busse.

FAMILY	RECOMMENDED SUPPLY VOLTAGE	DRIVE (mA)	LEVELS	INTERFACING TO (LV) TTL	LIVE INSERTION, REMOVAL HOT INSERTION
AHC	5 V	±8	5 V CMOS	Use level shifter	
AC	5 V	±24	5 V CMOS	Use level shifter	
ABT	5 V	-32/ +64	TTL	Yes	I _{OFF} , PU/D-3-state
ABT25	5 V	-80 (32)/+188 (64)			
ABTE	5 V	-60 (12)/ +90 (12)	ETL	Yes	I _{OFF} , PU/D-3-state and precharge
LVC	3.3 V	±24	LVTTTL	Yes	I _{OFF} , (LVCZ: PU/D-3-state)
ALVC	3.3 V	±24	LVTTTL	Yes	
LVT	3.3 V	-32/ +64	LVTTTL	Yes	I _{OFF} , PU/D-3-state
ALVT	3.3 V	-32/ +64	LVTTTL	Yes	I _{OFF} , PU/D-3-state

Probleme bei Anschaltung einer Baugruppe während des Betriebs:

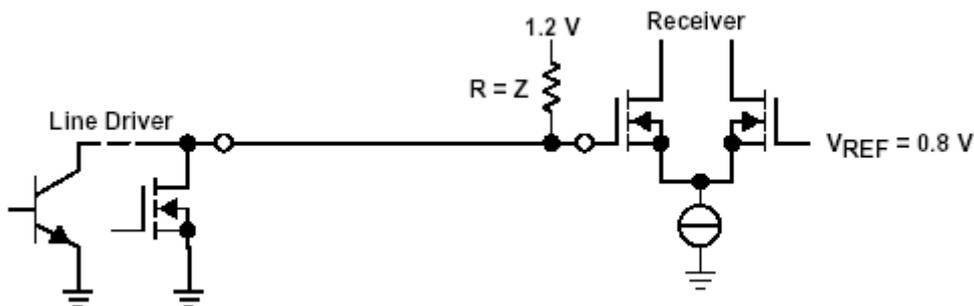
- Zerstörung der Schnittstellenbausteine durch Kurzschlussstrom → voreilende Kontakte für die Versorgung.
- Beeinflussung des Signalpegels → Powerup-3-state, precharge
- Störung auf der Stromversorgung (Ladestrom aller Blockkondensatoren).

Backplane Transceiver Logic



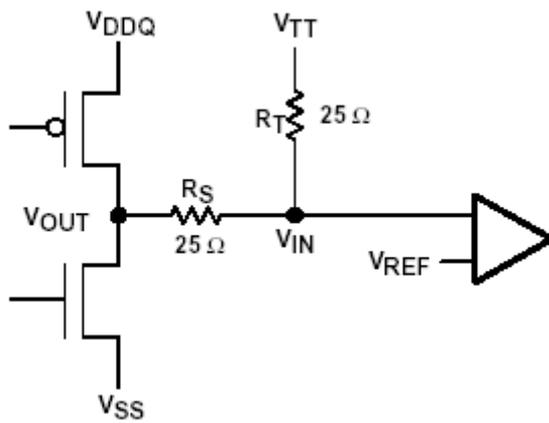
- Physical Layer des Futurebus
- Signalhub 1.1V
- Diode reduziert Ausgangskapazität auf $< 5\text{pF}$
- Incident wave switching
- Live insertion

GTL (Gunning Transceiver Logic)



- Entwickelt für Busse auf Leiterplattenebene z.B. Prozessor – Speicher
- Signalhub 0.8V
- Treiberfähigkeit 40mA → $R_{\text{min}} = 20\text{Ohm}$ → $Z_{0\text{min}} = 40\text{Ohm}$
- Verlustleistung pro Pin 16mW → in ASICs integrierbar
- GTLP mit Treiberfähigkeit bis 100mA; für Backplanebusse; incident wave switching

SSTL (Stub Series Terminated Logic)



- Interface zu SDRAMs (DDR)
- Bustakt bis 200MHz
- Anpassung am Anfang und am Ende der Busleitung
- $V_{DDQ} \leq V_{DD}$ um Verlustleistung in der Ausgangsstufe klein zu halten

FELDBUSSE

1. Allgemeines

Einsatz von Digitalrechnern zur Führung von techn. Prozessen in den 60er Jahren. -> Prozessrechner; streng zentrale Strukturen; multitaskfähiges Echtzeitbetriebssystem.

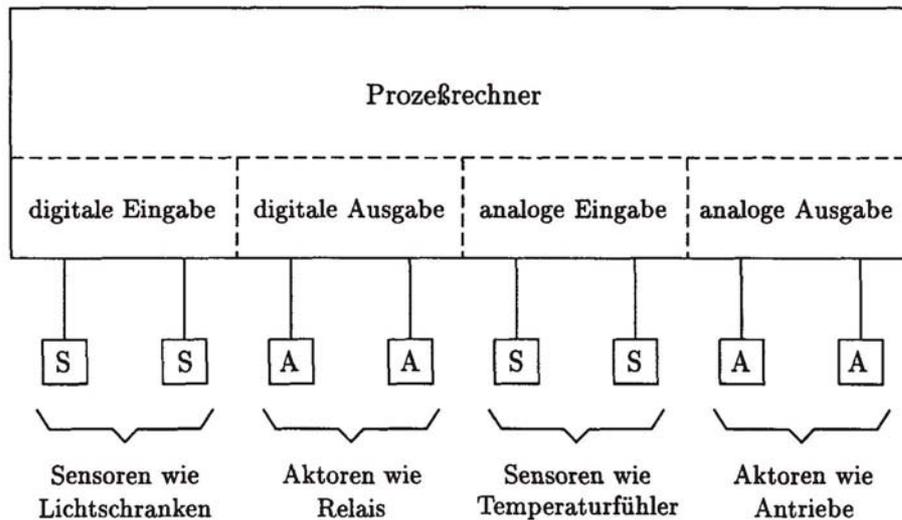
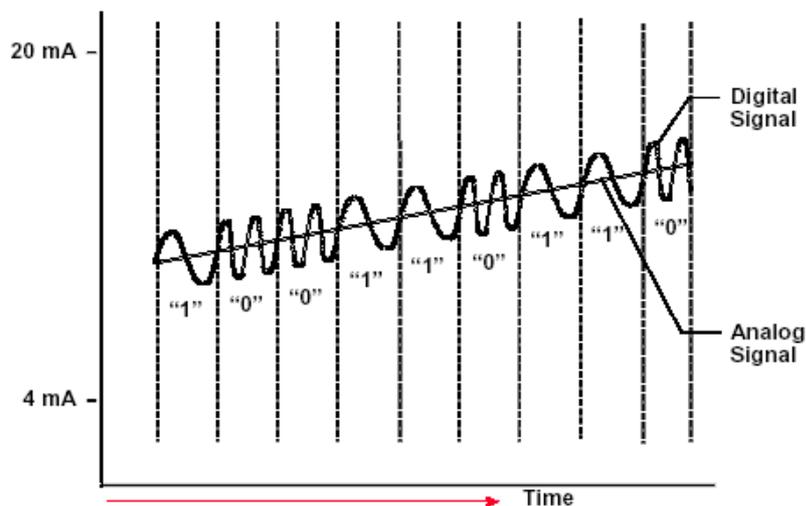


Bild 1: Klassischer Prozessrechner

Extremer Preisverfall der Hardware -> zunehmende Dezentralisierung, SPS. Durch den Einsatz von Mikrocontrollern übernehmen Feldgeräte immer mehr Aufgaben wie Messwertaufbereitung, Messwertüberwachung etc. -> Parametrisierung notwendig -> Einstellmöglichkeit vor Ort, zusätzliche Punkt zu Punktverbindung oder (Übergangslösung) HART-Protokoll (Highway Addressable Remote Transducer).



Das HART-Protokoll basiert auf dem Bell 202 telephone communication standard und verwendet *frequency shift keying* (1200Hz und 2200Hz) zur Übertragung der zusätzlichen Information. Das analoge 4-20mA Signal zur Übertragung der eigentlichen Prozessgröße bleibt ungestört erhalten.

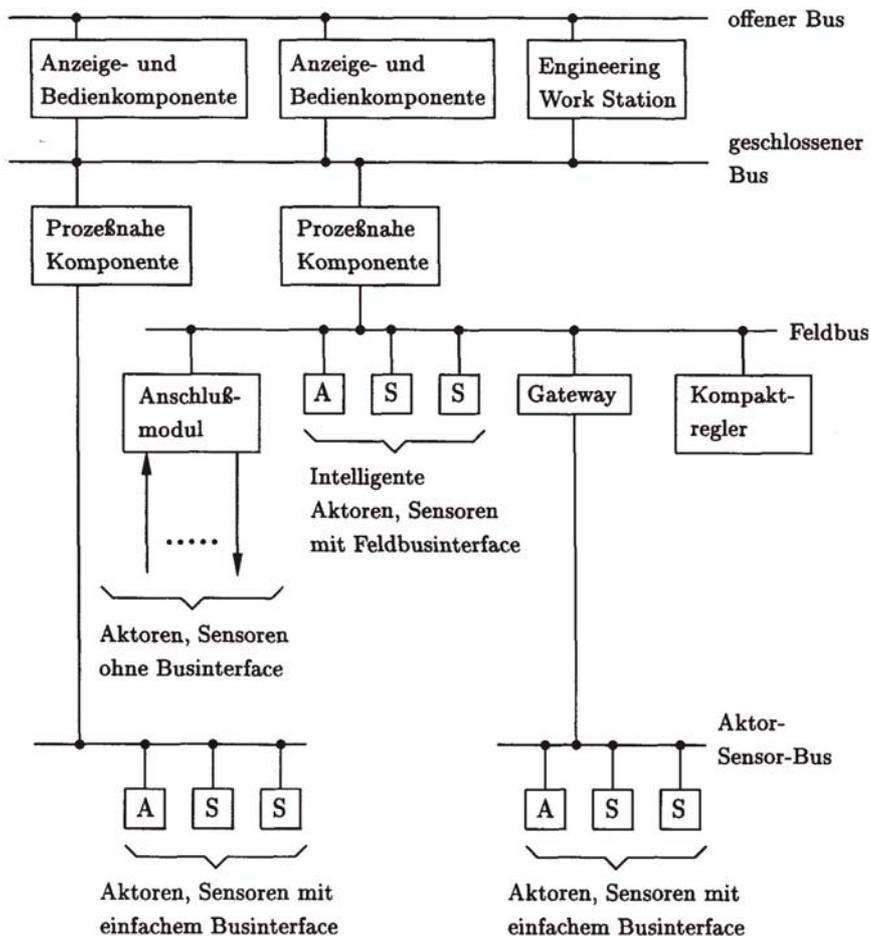


Bild 2: Struktur moderner Prozessleitsysteme

- Offener Bus (Backbone): Prozessleitebene und Verbindung zur Betriebsleitebene; Anforderungen wie an LAN -> Ethernet, TCP/IP
- Geschlossener Bus(Zellenbus): Verbindet die verschiedenen Fertigungszellen miteinander und stellt die Verbindung zur Prozessleitebene her. MAP (Manufacturing Automation Protocol) wurde Anfang der 80er von GM entwickelt; Profibus FMS.
- Feldbus: Verbindet die Feldebene (Sensoren und Aktoren) mit den prozessnahen Komponenten (SPS, Industrie-PC) sowie diese oft auch miteinander.
- Aktor-Sensor-Bus: Besonders einfacher und preisgünstiger Feldbus (z.B. ASI).

Einsatzgebiete für Feldbusse:

- Fertigungsautomatisierung: Mittelhohe Übertragungsraten, mittelgroße Ausdehnung.
- Prozessautomatisierung (Verfahrenstechnik, chem. Industrie): Eigensicherheit, geringe Anforderung an die Übertragungsraten, da Prozesse vergleichsweise langsam.
- Gebäudeautomatisierung (Klimatisierung, Beleuchtung etc.): Sehr große räumliche Ausdehnung, sehr viele Geräte, Geräte sollen während des Betriebes angeschlossen oder abgeklemmt werden können.
- Fahrzeugtechnik: Kleine räumliche Ausdehnung, sehr kurze Reaktionszeiten erforderlich, extrem hohe Anforderungen an die Übertragungssicherheit.

Bussystem	Hersteller	Hauptanwendungsgebiet
Ethernet	Siemens	Lokales Büronetz (LAN), offener und geschlossener Systembus
PROFIBUS-FMS	Siemens	Fertigungsautomatisierung
PROFIBUS-DP	Siemens	Fertigungsautomatisierung
PROFIBUS-PA	Siemens	Prozeßautomatisierung
P-NET	(dänisches Produkt)	Fertigungsautomatisierung (Prozeßautomatisierung)
WorldFIP	Telemecanique	Fertigungsautomatisierung (Prozeßautomatisierung)
Foundation- Feldbus	Fieldbus-Foundation	Prozeßautomatisierung
CAN	Bosch Chip: Intel, Motorola, Philips	Fahrzeugtechnik
LON	Echelon Chip: Motorola, Toshiba	Gebäudeautomatisierung
Interbus-S	Phoenix Contact	Fertigungsautomatisierung
ASI	Siemens Pepperl+Fuchs	Aktor-Sensor-Bus für die Fertigungsautomatisierung

Bild 3: Übersicht Bussysteme

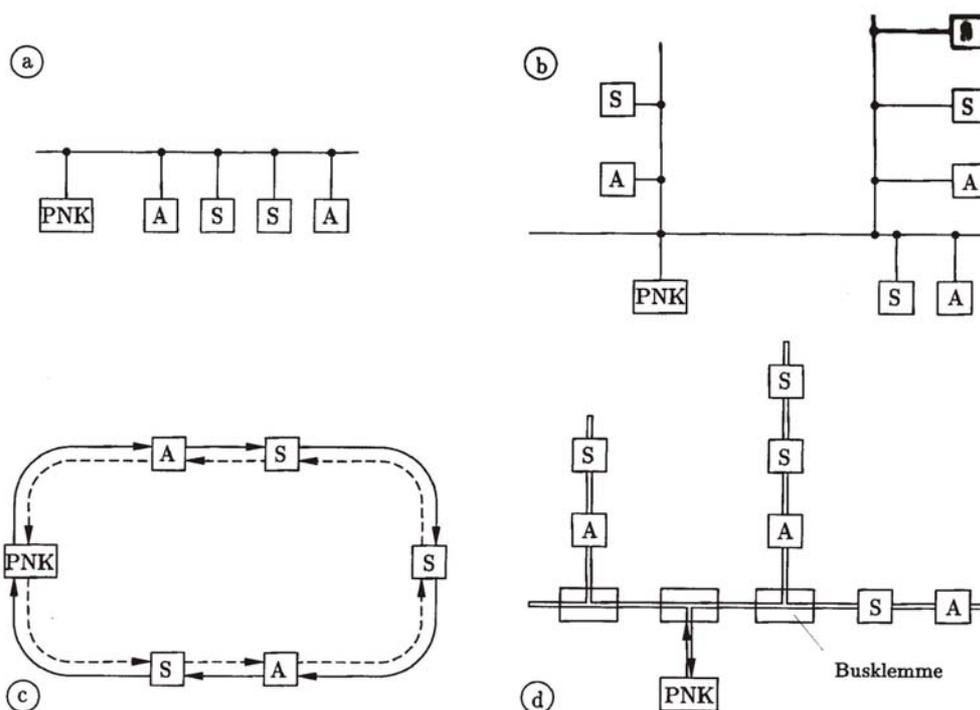


Bild 4: Busstrukturen

Im allgemeinen ist die Datenübertragung zwischen PNK und Feldgerät (z.B. Ausgabewert an Aktor, Status an PNK) oder zwischen zwei PNKs bidirektional.

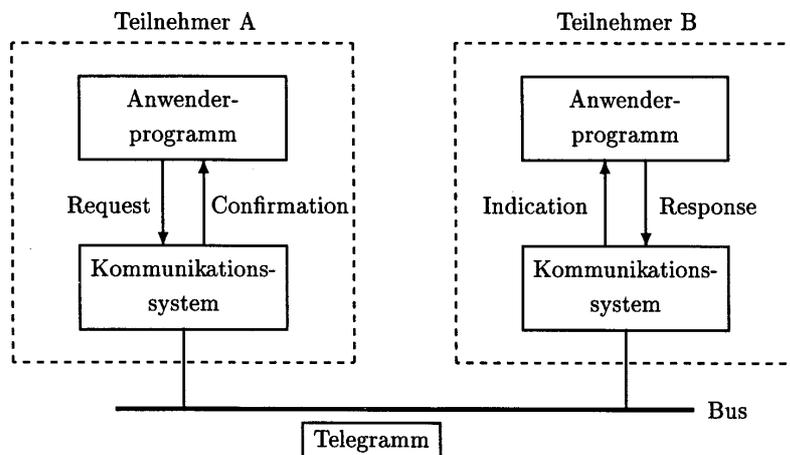


Bild 5: Datenübertragung zwischen 2 Teilnehmern

Ziel der Normungsbestrebungen ist die völlige Unabhängigkeit des Anwenderprogrammes von der Realisierung der Datenübertragung über den Bus. Das Anwenderprogramm benutzt das Kommunikationssystem mit Hilfe sogenannter Dienste (z.B. „send and request data with reply“ mit den Dienstprimitiven request, indication, response und confirmation (PROFIBUS)).

Verbindungslose Übertragung: Ein Sender übermittelt seine Information an mehrere (Multicast) oder an alle Teilnehmer (Broadcast) -> Producer-Consumer-Modell.

Verbindungsorientierte Übertragung: Der Sender spricht nur einen bestimmten Teilnehmer an -> Client-Server-Modell.

Bei allen Feldbussen ist das Kommunikationssystem nach dem sogenannten ISO-OSI-Schichtenmodell aufgebaut. Dabei sind mit Ausnahme von P-NET und LON die Schichten 3 bis 6 nicht realisiert.

7	Anwendungsschicht (Application Layer)		
6	Darstellungsschicht (Presentation Layer)		
5	Sitzungsschicht (Session Layer)		
4	Transportschicht (Transport Layer)		
3	Netzwerkschicht (Network Layer)		
2	Datenübertragungsschicht (Data Link Layer)	2b	Rahmenbildung (Framing)
		2a	Buszugriff (MAC)
1	Physikalische Schicht (Physical Layer)		

Bild 6: ISO-OSI-Schichtenmodell

Aufgaben der einzelnen Dienste:

- Während der Prozessführung müssen Sensor- und Aktordaten sowie kurze Statusinformationen (ob ein Feldgerät korrekt arbeitet) in möglichst gleich bleibenden Zeitabständen zyklisch übertragen werden (Datentransferphase). Möglichst effizient mit minimalem Overhead -> Dienste der Schicht 2.

- Vor der Datentransferphase müssen die einzelnen Stationen initialisiert und parametriert werden. Ebenso muß die Möglichkeit bestehen, während der Datentransferphase ausführliche Diagnoseinformation zu übertragen sowie ggf. Parametersätze in einer Station zu ändern. Es sind daher zusätzliche azyklische Übertragungen einzelner Nachrichten nötig.-> Schicht 7.

Bussystem	genormte Schicht	Norm
Ethernet	1, 2	IEEE 802.3, IEEE 802.2
PROFIBUS-FMS	1, 2 7	EN 50170 (und DIN 19245, Teil 1) EN 50170 (und DIN 19245, Teil 2)
PROFIBUS-DP	1, 2, 7	EN 50170 (und DIN 19245, Teil 3)
PROFIBUS-PA	1 2 7	IEC 1158-2 DIN 19245, Teil 1 und 4 (Teil 4 enthält Anpassung von Teil 1 an IEC 1158-2) DIN 19245, Teil 3
P-NET	1, 2, 7	EN 50170 (und dänische Norm)
WorldFIP	1, 2, 7	EN 50170 (und französische Norm)
Foundation-Feldbus		(Aufnahme in EN 50170 beantragt)
CAN	1, 2	ISO 11898 (und auch ISO 11519-1)
LON		(Protokoll offengelegt)
Interbus-S	1, 2, 7	DIN 19258
ASI		zur Normung eingereicht

Bild 7: Normungsstand 1988

2. Physikalisches Medium (Schicht 1)

Grundsätzlich sind die Realisierung des Übertragungsmediums und die Art der Kodierung nicht festgelegt. Es müssen lediglich die der Schicht 2 zur Verfügung gestellten Dienste gleich bleiben.

Bussystem	Leitung	Übertragungsstandard
Ethernet	Koaxialkabel mit doppelter Schirmung	IEEE 802.3
PROFIBUS-FMS	Zweidrahtleitung, verdrillt, geschirmt	RS 485
PROFIBUS-DP	Zweidrahtleitung, verdrillt, geschirmt	RS 485
PROFIBUS-PA	Zweidrahtleitung, verdrillt, geschirmt oder ungeschirmt	IEC 1158-2
P-NET	Zweidrahtleitung, verdrillt, geschirmt	RS 485
WorldFIP	Zweidrahtleitung, verdrillt, geschirmt oder ungeschirmt	IEC 1158-2
Foundation-Feldbus	Zweidrahtleitung, verdrillt, geschirmt oder ungeschirmt	IEC 1158-2
CAN	Zweidrahtleitung, verdrillt, geschirmt	RS 485, modifiziert
LON	z.B. Zweidrahtleitung, verdrillt	z.B. RS 485
Interbus-S	5-adrig, paarweise verdrillt	RS 485
ASI	Zweidrahtleitung, ungeschirmt	spezielles Verfahren

Bild 8: Vorzugsweise verwendete Übertragungsmedien

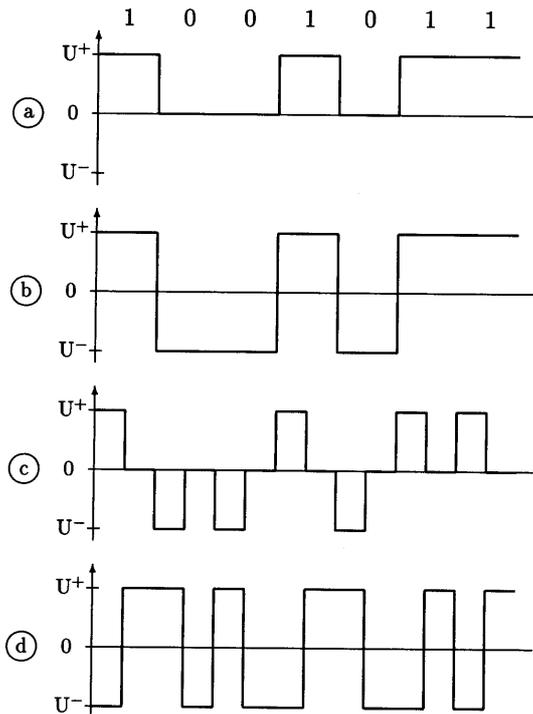


Bild 9: Übliche Kodierungsverfahren; a) unipolar NRZ b) bipolar NRZ c) bipolar RZ d) Manchester

3. Buszugriffsverfahren

Legt fest, welche Sendestation zu einem bestimmten Zeitpunkt den Bus belegen darf. Hat wesentlichen Einfluß auf die zu erzielende Datenrate, die maximale Leitungslänge und die Echtzeitfähigkeit des Bussystems.

3.1. Buszugriff nach Zuteilung:

Master-Slave-Verfahren: Ein Teilnehmer (Master) spricht die ihm zugeordneten anderen Teilnehmer (Slaves) der Reihe nach an. -> „Kommando-Antwort-Schema“.

Vorteile:

- Slaves können einfach aufgebaut werden
- Die Zeit innerhalb der ein bestimmtes Telegramm übertragen wird ist bestimmbar - >echtzeitfähig.

Nachteile:

- Ausfall des Masters blockiert das gesamte (Teil)system.
- Übertragung von Slave zu Slave ist umständlich, da Kommunikation immer über Master läuft.
- Bei zyklischen Abfragen werden oft dieselben Daten übertragen, da sich in der Zwischenzeit nichts geändert hat.

Token-passing-Verfahren: Bei mehreren Teilnehmern mit Masterfunktion an einem Bus wird das Zugriffsrecht mittels eines speziellen Telegramms (Token) reihum weitergegeben. Jeder

dieser Teilnehmer kann wiederum Master einer Master-Slave-Konfiguration sein (PROFIBUS).

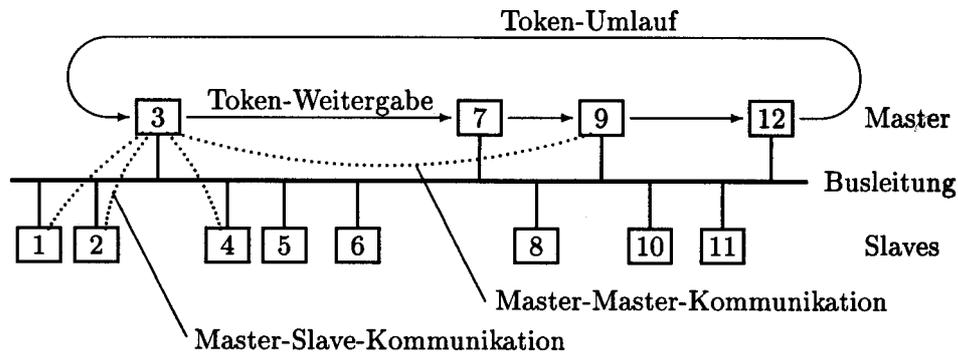


Bild : Token-passing und Master-Slave

Vorteile:

- Teilnehmer können direkt miteinander kommunizieren
- Teilnehmer der gerade nichts zu tun hat, kann seine Zeitscheibe einem anderen Teilnehmer zur Verfügung stellen.
- Wird ein Master defekt, kann das erkannt werden und er wird vom Tokenbesitz ausgeschlossen. Das restliche System ist weiter betriebsfähig.
- Die maximale Tokenumlaufzeit ist bestimmbar -> echtzeitfähig.

Nachteile:

- Die Busverwaltung ist komplex
- Tokenumlaufzeit erhöht sich mit steigender Teilnehmerzahl
- Wird eine neue Station angeschlossen oder eine vorhandene weggenommen, muß das gesamte Bussystem neu konfiguriert werden.

Arbitrator-Producer-Consumer-Verfahren (WorldFIP): Der Arbitrator hat die einzige Aufgabe, den Bus zu verwalten. Er stößt reihum einen Teilnehmer nach dem anderen an, sein Telegramm über den Bus zu versenden (Producer). Alle daran interessierten Teilnehmer (Consumer) gehen dabei auf Empfang. Alle für die Prozessführung notwendigen zyklischen Übertragungen werden bei der Konfigurierung in eine Tabelle im Arbitrator eingetragen und in der Produktivphase der Reihe nach zyklisch angestoßen. Darüber hinaus kann ein Producer bei Bedarf in seinem Telegramm beim Arbitrator auch einen azyklischen Übertragungswunsch bekannt geben.

Vorteile:

- Direkte Kommunikation zwischen den Teilnehmern
- Echtzeitfähigkeit

Nachteile:

- Ist der Arbitrator defekt, fällt der gesamte Bus aus. Daher ist ein zweiter Arbitrator im Standby-Betrieb vorgesehen.
- Zykluszeit vergrößert sich mit der Teilnehmerzahl.

Summierrahmen-Verfahren (Interbus-S): Spezielle Variante eines Master-Slave-Verfahrens.

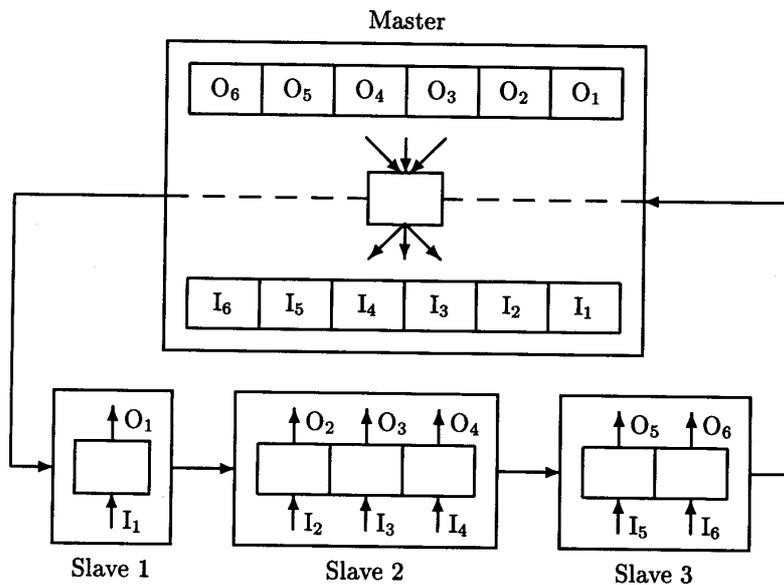


Bild : Summierrahmen-Verfahren

Vorteile:

- Hohe Protokolleffizienz (wenig Overhead)
- Die Daten sind automatisch konsistent
- Die Echtzeitfähigkeit ist erfüllt

Nachteile:

- Der gesamte Bus fällt aus, wenn auch nur ein Teilnehmer ausfällt.
- Das Hinzufügen oder Entfernen eines Teilnehmers bedingt eine Neukonfiguration.
- Eine direkte Übertragung von Slave zu Slave ist nicht möglich
- Geringe Flexibilität der Struktur

3.2. Buszugriff nach Bedarf

CSMA-Verfahren (Carrier Sense Multiple Access): Keine zentrale Busverwaltung. Möchte ein Teilnehmer senden, wartet er bis der Bus frei ist und beginnt dann zu senden. Versuchen das mehrere Teilnehmer gleichzeitig, kommt es zu einer Kollision am Bus. Durch Mitlesen des eigenen Telegrammes wird die Kollision erkannt. Je nach Reaktion darauf verschiedene Verfahren:

CSMA/CD-Verfahren (Collision Detection): Das Telegramm wird sofort abgebrochen und der Bus freigegeben. Nach einer zufälligen Zeit startet ein neuer Versuch. Wird bei Ethernet verwendet.

Vorteile:

- Für sehr viele Teilnehmer geeignet, da eine Übertragung nur bei Bedarf stattfindet.

- Ein Teilnehmer kann hinzugefügt oder entfernt werden, ohne dass der Bus rekonfiguriert werden muß.

Nachteile:

- Wenig Effizienz im Hochlastbetrieb, da viele Telegramme abgebrochen werden müssen.
- Nicht Echtzeitfähig.

CSMA-Verfahren bei LON: Hier beginnt ein Teilnehmer nach Freiwerden des Busses erst nach einer von der momentanen Busbelastung und einer Zufallskomponente abhängigen Zeit zu senden. Ebenso ist es möglich, Prioritäten zu vergeben (hohe Priorität -> kurze Wartezeit). Es gibt eine Variante mit und ohne CD. Wird ohne Kollisionserkennung gearbeitet, wartet der Sender eine bestimmte Zeit auf eine Antwort des angesprochenen Teilnehmers. Wird diese nicht korrekt empfangen, wird eine Kollision angenommen und ein neuer Versuch gestartet.

Vorteile:

- Verbesserung der Echtzeitfähigkeit
- Verbesserung des Datendurchsatzes bei hoher Busbelastung

CSMA/CA-Verfahren (Collision Avoidance) bei CAN-Bus: Modifizierte RS485-Schnittstelle mit einem dominanten (logisch 0) und einem rezessiven (logisch 1) Zustand. Jedes Telegramm beginnt mit einem dominanten Startbit und anschließender Adressinformation. Diese kennzeichnet nicht eine Station, sondern das zu übertragende Objekt (z.B. Temperaturmesswert). Senden zwei oder mehrere Teilnehmer gleichzeitig, setzen sich die dominanten Zustände durch (das Telegramm bleibt unzerstört) und die unterlegenen Sender ziehen sich zurück. Die Adresse stellt somit gleichzeitig eine Priorität dar.

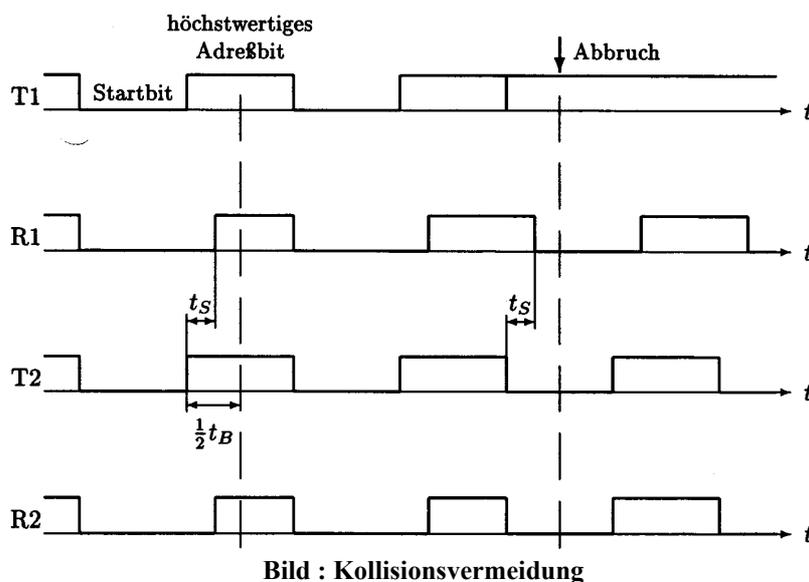


Bild : Kollisionsvermeidung

Vorteile:

- Viele Teilnehmer möglich

- Ein Teilnehmer kann hinzugefügt oder entfernt werden, ohne dass der Bus rekonfiguriert werden muß.
- Kein Einbruch der Effizienz im Hochlastfall
- Echtzeitfähigkeit ist für hochpriorige Telegramme gegeben

Nachteile:

- Die maximale Übertragungsrate hängt sehr stark von der räumlichen Ausdehnung des Busse ab (Laufzeit).

4. Telegrammaufbau

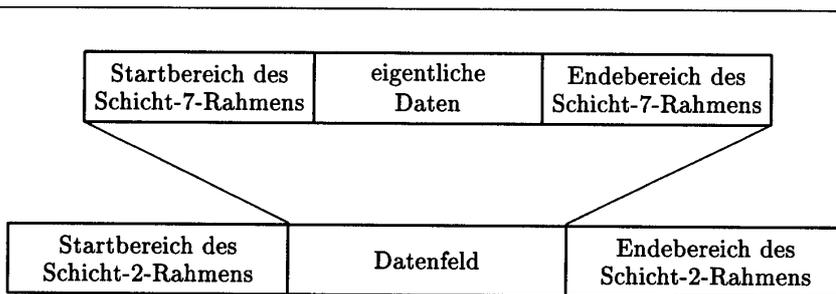


Bild: Telegrammaufbau

5. Datensicherung

Bussystem	Prüfmethode	Übertragung
Ethernet	CRC	synchron
PROFIBUS-FMS	Paritybit plus Prüfsumme	asynchron
PROFIBUS-DP	Paritybit plus Prüfsumme	asynchron
PROFIBUS-PA	CRC	synchron
P-NET	Prüfsumme	asynchron
WorldFIP	CRC	synchron
Foundation-Feldbus	CRC	synchron
CAN	CRC	synchron
LON	CRC	synchron
Interbus-S	CRC	asynchron
ASI	Paritybit	asynchron

Bild : Datensicherung und Übertragung

6. Schnittstelle Kommunikation – Anwendung

In einer konkreten Feldbusstation ist es schwierig eine klare Trennlinie zwischen Kommunikationssystem und Anwendung zu ziehen. Über den Application Layer (Schicht 7 des ISO-OSI-Modells) werden noch die Schnittstellenbegriffe Application-Layer-Interface (ALI) und User-Layer (Schicht 8) verwendet.

6.1. Datenorientierte Schnittstelle:

Hier erfolgt der Datenaustausch zwischen Applikation und Kommunikationssystem über einen gemeinsamen Speicherbereich in den die Anwendung die bereitzustellenden Daten

schreibt und aus dem sie die für sie bestimmten Daten entnimmt. In einer PNK stellt dieser Speicherbereich somit ein Prozessabbild der gesamten Applikation dar.

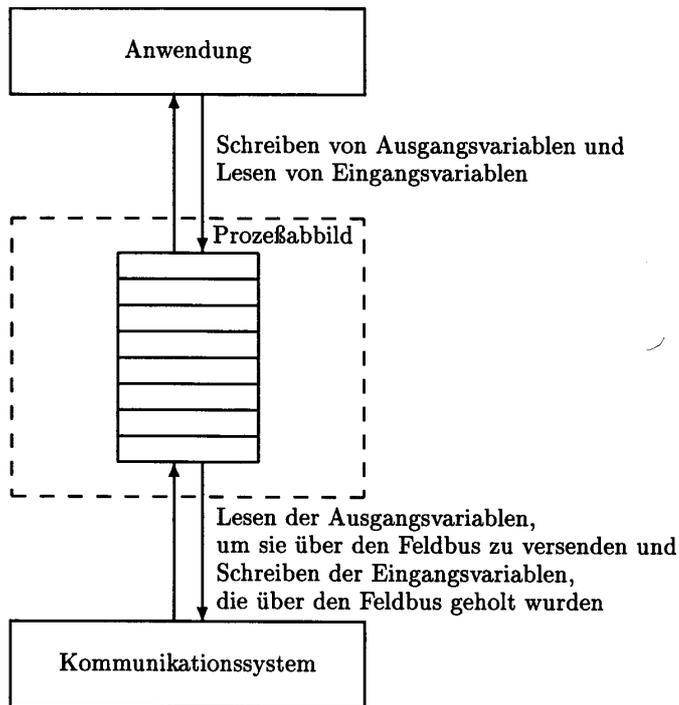


Bild : Datenorientierte Schnittstelle

Diese Schnittstelle wird für die zyklische Datenübertragung bei PROFIBUS-DP, INTERBUS-S und ASI verwendet. Der Zugriff des Kommunikationssystems auf den Speicher geschieht asynchron zur Anwendung und es können daher merkliche Verzögerungen zwischen Erfassung eines Messwertes durch einen Sensor und eine daraus resultierende Reaktion entstehen.

Die datenorientierte Schnittstelle setzt primär auf die Schicht 2 auf, die Aufgaben der Schicht 7 beschränken sich auf ein Minimum. Man erhält damit eine sehr effiziente Schnittstelle zum schnellen Übertragen von Prozessdaten.

6.2. Nachrichtenorientierte Schnittstelle:

Bei Feldbusanwendungen ist es oft notwendig, über den zyklischen Austausch von Eingangs- und Ausgangsgrößen hinaus fallweise komplexe Datenstrukturen zu übertragen, Events bzw. Alarmer zu übermitteln sowie Programme zu laden und zu starten. Daher ist es sinnvoll, einen leistungsfähigen Application Layer (Schicht 7) zu definieren, der dem Anwenderprogramm entsprechende Dienste anbietet. Die dzt. verbreitetsten Protokolle wie FMS (Fieldbus Message Specification) bei PROFIBUS, SUB-MMS bei WorldFIP, CMS (CAN-based Message Specification) bei CAN und PMS (Peripherals Message Specification) bei Interbus-S sind alle eine Untermenge der in den 80er Jahren von General Motors im Rahmen von MAP entwickelten Schnittstelle MMS (Manufacturing Message Specification).

Bei FMS basiert das Protokoll auf den drei Elementen:

- Kommunikationsobjekte: Sensor- und Aktordaten, Parametersätze und Programme die zwischen Teilnehmern übertragen werden sollen werden mit ihren wichtigsten Eigenschaften zu Objekten verbunden. Jeder Teilnehmer besitzt ein Objektverzeichnis, in dem alle von ihm angebotenen Kommunikationsobjekte

aufgelistet sind. Mit Hilfe eines entsprechenden Dienstes kann dieses Verzeichnis von einem anderen Teilnehmer angefordert werden.

- Kommunikationsbeziehungen: Bevor eine Datenübertragung stattfindet muß eine Kommunikationsbeziehung definiert werden. Bei FMS werden diese Beziehungen ebenfalls in einer Liste, der sog. Kommunikationsbeziehungsliste eingetragen. Das hat den Vorteil, dass neben der Adresse des anzusprechenden Teilnehmers auch noch andere Parameter dieser Verbindung vermerkt werden können.
- Kommunikationsdienste: Bei FMS sind insgesamt 39 Dienste zum Aufbau einer Verbindung, zum Übertragen von Daten sowie zum Laden und Steuern von Programmen beschrieben.

Index 0	Header
Index 1 - i	Statisches Typverzeichnis
Index k - n	Statisches Objektverzeichnis
Index p - t	Dynamisches Variablenlisten-Verzeichnis
Index v - z	Dynamisches Program-Invocation-Verzeichnis

Aufbau eines Objektverzeichnisses

Index	Objektcode	Bedeutung
1	Datentyp	Boolean
2	Datentyp	Integer8
3	Datentyp	Unsigned32

Index	Objekt-code	Anzahl Elemente	Daten-typ 1	Länge 1		
4	Datentyp-Strukturbeschreibung	4	1	1		
	Daten-typ 2	Länge 2	Daten-typ 3	Länge 3	Daten-typ 4	Länge 4
	2	1	2	1	3	4

Statisches Typenverzeichnis

Index	Objekt-code	Datentyp-Index	Länge	lokale Adresse	Name
8	Simple	3	4	4020H	Fehlercode

Index	Objekt-code	Datentyp-Index	Name	lokale Adresse 1	lokale Adresse 2
9	Record	4	MultiSensor	4032H	4064H
				lokale Adresse 3	lokale Adresse 4
				4065H	4018H

Index	Objekt-code	Max-Octets	lokale Adresse	Domain-Status	Name
14	Domain	1024	5000H	1	Code
15	Domain	100	5400H	1	Daten

Index	Objekt-code	Index-Event-Daten	Länge	Enabled	Name
18	Event	8	4	true	SensorWartung

Statisches Objektverzeichnis

Der überwiegende Anteil der Busbelastung wird durch die zyklische Übertragung der Prozessdaten verursacht. Daher ist für diese Aufgabe eine datenorientierte Schnittstelle (Prozessabbild) ideal. Die restlichen Übertragungen vor und während der Datentransferphase kommen vergleichsweise selten vor, sodaß hier auf eine effiziente Übertragung zugunsten einer komfortablen Schnittstelle zur Anwendung (nachrichtenorientierte Schnittstelle) verzichtet werden kann. Bei WorldFIP, CAN und Interbus-S sind daher beide Schnittstellenformen parallel verfügbar.

Bei PROFIBUS war zunächst mit der Variante FMS nur eine nachrichtenorientierte Schnittstelle verfügbar. Die fehlende Akzeptanz auf der Feldebene führte zur Entwicklung der Variante PROFIBUS-DP (Dezentrale Peripherie), die primär auf einer datenorientierter Schnittstelle basiert. 1996 wurde PROFIBUS-DP um einige Elemente einer nachrichtenorientierten Schnittstelle erweitert (DPV1). Zur Abdeckung der besonderen Anforderungen in der Prozessautomatisierung (chem. Industrie) wurde weiters, basierend auf dem Protokoll DPV1, unter Verwendung einer eigensicheren Übertragungsphysik die Variante PROFIBUS-PA (Prozess Automatisierung) entwickelt. Die Variante FMS wird heute meist auf Zellenebene eingesetzt.

6.3. Application Layer Interface

In den einzelnen Feldbusnormen sind die verfügbaren Dienste samt der zu übergebenden Parameter, der Telegrammaufbau usw. genau festgelegt. Wie diese Dienste vom Anwenderprogramm aufgerufen werden hängt u.a. von der verwendeten Programmiersprache und dem Betriebssystem ab und ist nicht festgelegt. Es gibt z.Zt. eine Vielzahl von Bestrebungen, hier eine Standardisierung zu erreichen.

6.4. User Layer

Über die Feldbusnorm hinaus ist es wünschenswert, dass sich Feldgeräte gleichen Typs (z.B. Temperatursensoren) gleich verhalten, sodass man Geräte verschiedener Hersteller untereinander austauschen kann und auch eine Bedienung (z.B. Parametrierung) über den Bus mit einem einzigen Bediengerät und einer einheitlichen Oberfläche möglich ist.

Profile:

- Kommunikationsprofil: Festlegung (Einschränkung) der bereitzustellenden Kommunikationsdienste
- Anwenderprofil: Festlegung der Gerätefunktionen

Mehrere Geräteprofile für Geräte einer Branche → Branchenprofil

Die Profilbildung wird von den Nutzerorganisationen Betrieben und die Profile sind daher Feldbusspezifisch.

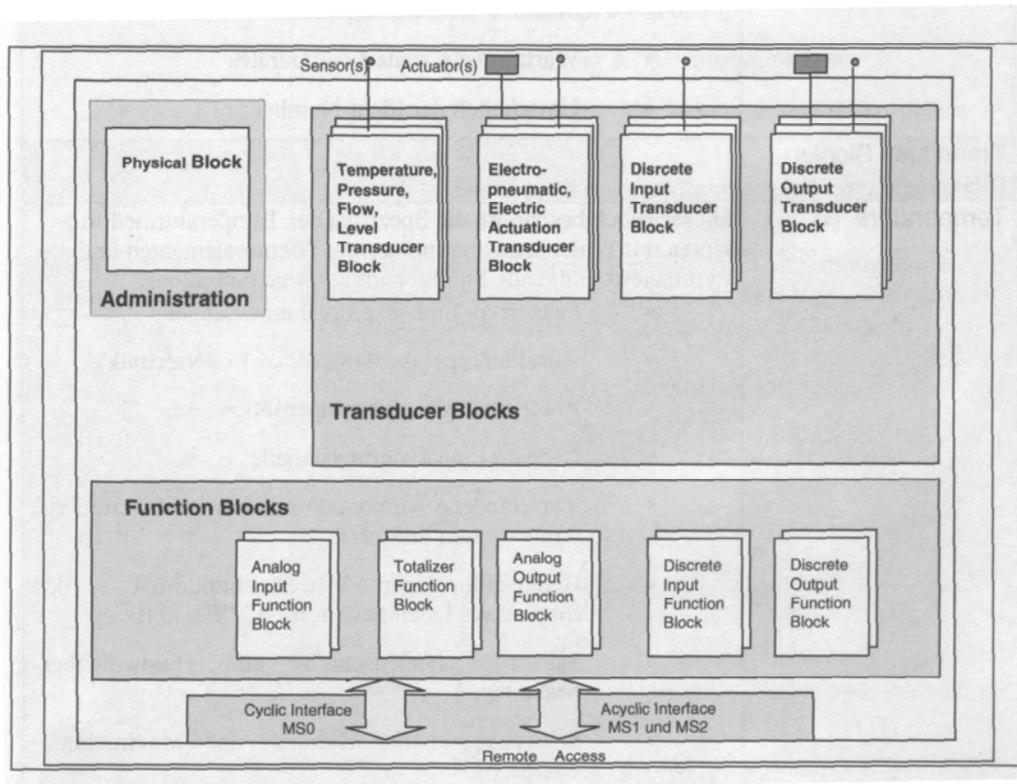
Beispiel für die Festlegung einer Gerätefunktion: Der Messwertes soll als 16bit Integer übertragen werden. Liefert der Sensor einen Wert geringerer Auflösung (z.B. 12 Bit) muß festgelegt sein, an welchen Bitstellen der Wert steht und wie die verbleibenden Stellen zu interpretieren sind (z.B. Wert linksbündig, die restlichen Stellen sind auf „0“ zu setzen).

Für die Erstellung und Beschreibung eines Anwenderprofils wird zunehmend das sog. Function-Block-Modell verwendet.

Function-Block-Modell

Die gesamte Funktionalität eines Gerätes wird in Grundfunktionen (Blöcke) aufgeteilt. Jedes Gerät enthält einen **Physical Block** (Diagnosedaten, elektronisches Typenschild, etc.), **Function Blocks** (die eigentliche Funktionalität wie z.B. „Analog Input“, „Analog Output“, „PID Regler“, etc.) und **Transducer Blocks** (sensorspezifische Funktionen wie z.B. Kalibrierung, Linearisierung, etc.)

Es wird nicht die Realisierung sondern nur das nach außen sichtbare Funktionsverhalten beschrieben. Dieses ist immer mit Kommunikationsobjekten verbunden (z.B. Messwert, Minimal- und Maximalwert, Stützstellen für eine Linearisierungsfunktion, etc.).



Nr.	Variablen-Name	Objekt-typ	Datentyp	Anzahl Bytes	Zugriff	Art
10	out	Record	DS-33	5	r	zyk
11	pv_scale	Record	DS-36	11	r,w	a
16	pv_ftime	Simple	Float	4	r,w	a
19	alarm_hys	Simple	Float	4	r,w	a
21	hi_hi_lim	Simple	Float	4	r,w	a
23	hi_lim	Simple	Float	4	r,w	a
25	lo_lim	Simple	Float	4	r,w	a
27	lo_lo_lim	Simple	Float	4	r,w	a
30	hi_hi_alm	Record	DS-39	14	r	a
31	hi_alm	Record	DS-39	14	r	a
32	lo_alm	Record	DS-39	14	r	a
33	lo_lo_alm	Record	DS-39	14	r	a

Individuelle Parameter für Function-Block „Analog Input“

Name	Bedeutung
out	aktueller Meßwert und Status
pv_scale	Skalierung mit unterer und oberer Meßbereichsgrenze, Einheiten, Anzahl Ziffern nach dem Dezimalpunkt
pv_ftime	Anstiegszeit in s
alarm_hys	Hysterese der Alarmfunktion in % des Meßbereiches
hi_hi_lim	obere Alarmgrenze
hi_lim	obere Warnungsgrenze
lo_lim	untere Warnungsgrenze
lo_lo_lim	untere Alarmgrenze
hi_hi_alm	Status der oberen Alarmgrenze mit Zeitstempel
hi_alm	Status der oberen Warnungsgrenze mit Zeitstempel
lo_alm	Status der unteren Warnungsgrenze mit Zeitstempel
lo_lo_alm	Status der unteren Alarmgrenze mit Zeitstempel

Bedeutung der individuellen Parameter für „Analog Input“

Um eine komplette Gerätebeschreibung in einer maschinenlesbaren Form zu erstellen, wurde eine standardisierte Device Description Language (DDL) entwickelt. Die Gerätebeschreibung kann dann in kodierter Form in der Firmware des Feldgerätes gespeichert sein und über den Feldbus von der PNK oder der Bedieneinheit abgerufen werden.

Eine der DDL vergleichbare Beschreibungsform ist die sog. Gerätestammdatei (GSD) bei PROFIBUS – Feldgeräten. Sie ist eine ASCII-Datei und muß mit jedem Feldgerät mitgeliefert werden. Bei der Projektierung des Feldbusses werden alle Gerätestammdateien in das Projektierungstool eingelesen.

Beispiele für Feldbusrealisierungen

PROFIBUS-DP

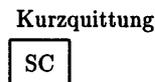
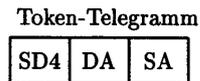
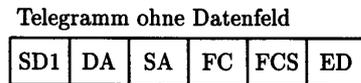
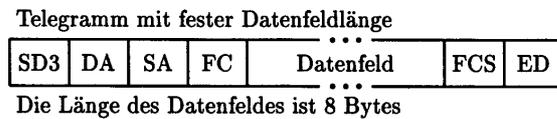
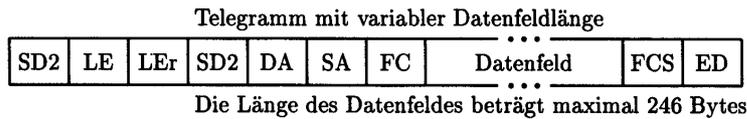
PROFIBUS-FMS	PROFIBUS-DP	PROFIBUS-PA
Profile für die Fertigungsautomatisierung	Profile für die Fertigungsautomatisierung	Profile für die Prozeßautomatisierung
Application Layer Interface (ALI)		
Schicht 7: Fieldbus Message Specification (FMS)	Schicht 7: Direct Data Link Mapper (DDL M)	
Schicht 3 bis 6: leer		
Schicht 2: Fieldbus Data Link (FDL)		
Schicht 1: DIN 19245, Teil 1 (RS485)		IEC 1158-2

Bild : Varianten des PROFIBUS

Physikalisches Medium: Linie an beiden Enden abgeschlossen (Standard: geschirmte verdrehte Zweidrahtleitung); RS485; Übertragungsgeschwindigkeit von 9.6kbit/s (1200m) bis 12Mbit/s (100m).

Buszugriffsverfahren: Token-Passing kombiniert mit Master-Slave-Verfahren.

Telegrammaufbau: Asynchrone Übertragung, jedes Byte (Oktett) wird als UART-Zeichen übertragen. Fünf verschiedene Telegrammformate:



- SD1 bis SD4 Startbyte (Start Delimiter)
(gleichzeitig Kennung des Telegrammformates)
- LE Längenbyte (Length),
(Anzahl Bytes in DA, SA, FC und Datenfeld, Wert 4 bis 249)
- LEr Wiederholung (repeated) des Längenbytes zur Sicherheit
- DA Zieladrefbyte (Destination Address)
- SA Quelladrefbyte (Source Address)
- FC Controlbyte (Frame Control)
(Festlegung des Telegrammtyps wie Senden,
Anfordern, Quittieren und anderes)
- FCS Prüfbyte (Frame Check Sequence)
- ED Endebyte (End Delimiter), Code: 16H
- SC Einzelzeichen, Code: E5H

Bild : Telegrammaufbau bei PROFIBUS

Datensicherung

Durch Paritybit und Prüfsumme wird eine Hammingdistanz von $d=4$ erreicht.

Dienste der Schicht 2 (FDL)

Von den vier möglichen Diensten

- SDN (Send Data with No Acknowledge)
- SDA (Send Data with Acknowledge)
- SRD (Send and Request Data with Reply)
- CSRD (Cyclic SRD)

werden von PROFIBUS-DP nur die Dienste SDN und SRD genutzt.

Typen von Teilnehmern

- DP-Slave: Feldgerät wie Sensor oder Aktor
- DP-Master, Klasse 1 (DPM1): Prozessnahe Komponente (PNK) wie z.B. SPS oder Industrie-PC.
- DP-Master, Klasse 2 (DPM2): Für Inbetriebnahme, Diagnose und Wartung des Feldbussystems, z.B. HMI (Human Machine Interface). Ein DPM2 dient nicht der unmittelbaren Prozessführung und muß nicht dauernd angeschlossen sein.

Anwendungsschicht

Der in der Norm definierte Direct Data Link Mapper (DDLMM) erfüllt die Aufgabe der Anwenderschicht (Schicht 7), obwohl formale Widersprüche bestehen. Die zyklisch zu übertragenden Daten werden ohne Overhead im Datenfeld des Schicht2-Rahmens übertragen. Bei einer azyklischen Übertragung werden zur Kennzeichnung des Dienstes der die Daten sendet sowie des Dienstes für den die Daten bestimmt sind, sog. Dienstzugangspunkte (definierte Kennzahlen) verwendet. Diese Kennungen werden als Adresserweiterung aufgefasst und stehen vor den zu übertragenden Daten im Datenfeld des Schicht2-Telegramms.

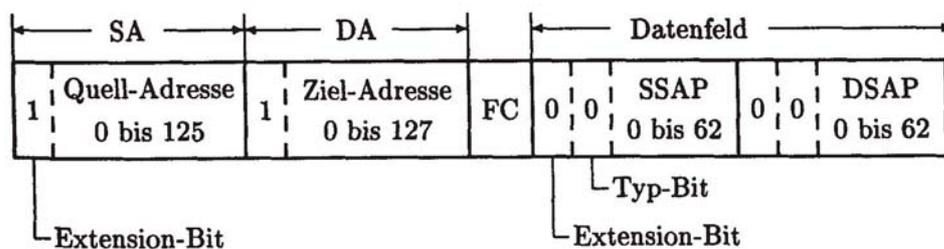


Bild : Adresserweiterung

Im jeweiligen Adressbyte werden nur die niederwertigsten 7Bit für die Stationsadresse verwendet (Adresse 127 für Broadcast, Adresse 126 reserviert -> max. 126 Teilnehmer). Das höchstwertige Bit dient einer Adresserweiterung. Es gibt zwei Typen von Adresserweiterung:

- Region/Segment-Adresse: In größeren Systemen wird der Bus über Segmentkoppler (Repeater) segmentiert, da aus physikalischen Gründen (RS485) eine Linie nur 32 Teilnehmer haben darf. Die Koppler belegen keine der 126 möglichen Teilnehmeradressen.
- Dienstzugangspunkt (SAP Service Access Point): Für den zyklischen Datenaustausch wird kein SAP angegeben, um eine möglichst effiziente Übertragung zu erhalten.

Dienst DDLML_ ...	DP-Slave		DPM1		DPM2		SSAP	DSAP
	Req	Res	Req	Res	Req	Res		
Data_Exchange	-	M	M	-	-	-	NIL	NIL
Chk_Cfg	-	M	M	-	-	-	62	62
Set_Prm	-	M	M	-	-	-	62	61
Slave_Diag	-	M	M	-	O	-	62	60
Get_Cfg	-	M	-	-	O	-	62	59
Global_Control	-	M	M	-	-	-	62	58
RD_Outp	-	M	-	-	O	-	62	57
RD_Inp	-	M	-	-	O	-	62	56
Set_Slave_Add	-	O	-	-	O	-	62	55
Get_Master_Diag	-	-	-	M	O	-	54	54
Start_Seq	-	-	-	O	O	-	54	54
Download	-	-	-	O	O	-	54	54
Upload	-	-	-	O	O	-	54	54
End_Seq	-	-	-	O	O	-	54	54
Act_Param	-	-	-	O	O	-	54	54

Bild: DDLML Dienste

Dienste für Datentransfer:

DDLML_Data_Exchange

- Parameter der Request-Primitive
 - Stationsadresse des angesprochenen Slaves
 - Ausgabedaten an den Slave
- Parameter der Confirmation-Primitive
 - Meldung, ob Übertragung erfolgreich war
 - Eingabedaten vom Slave
 - Diagnoseflag (Details werden bei *Slave_Diag* übertragen)

DDLML_Global_Control

- Parameter der Request-Primitive
 - Stationsadresse des angesprochenen Slaves oder Multicast
 - Kommando-Oktet (im Datenfeld) für die Kommandos „Sync“, „Freeze“ und „Clear“.
 - Oktet mit Gruppenauswahl (im Datenfeld)

Dienste für die Inbetriebnahme:

Während der Startphase wird eine bestimmte Sequenz von Diensten durchlaufen:

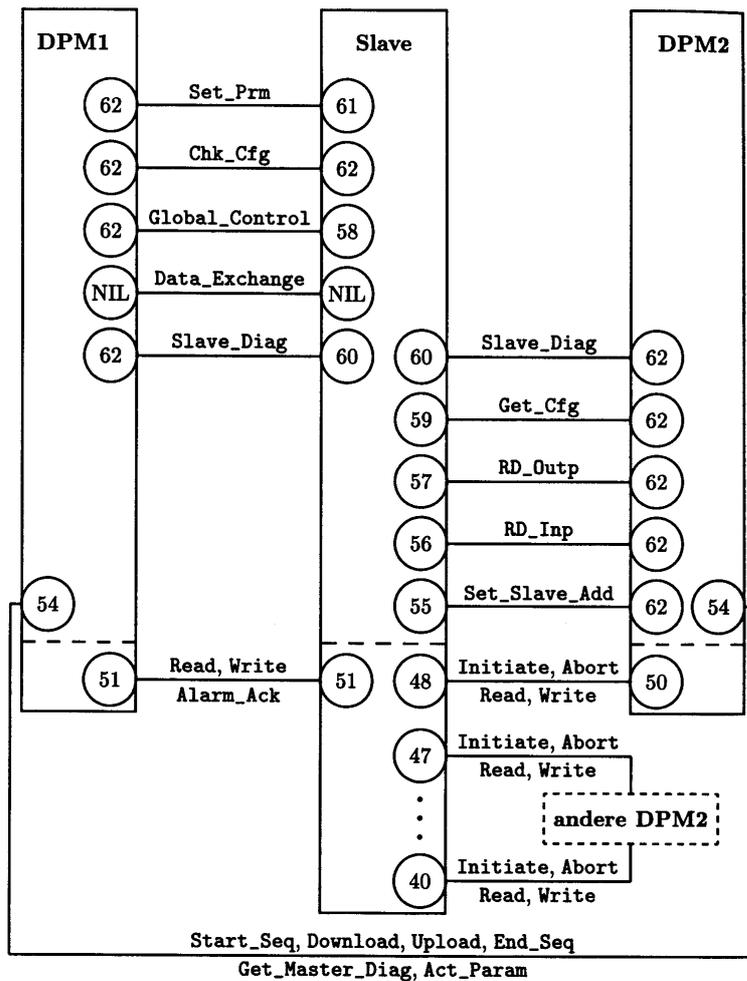
- 1.) *DDLMSlaveDiag*; Der Master fordert den Slave auf, seine Diagnosedaten zu senden. Hier ist nur relevant, ob der Slave bereit ist, mit dem Master zu kommunizieren.
- 2.) Mit *DDLMSetPrm* werden die Parameter (7 bis 244 Oktets) an den Slave übertragen und dort eingestellt. Die ersten 7 Oktets sind obligatorisch und betreffen
 - die Zugriffsberechtigung in Multi-Master-Systemen,
 - die Einstellung eines allfälligen Watchdog-Intervalls,
 - die minimale Reaktionszeit für das Antworttelegramm,
 - eine Überprüfung der Ident-Nummer sowie
 - eine etwaige Gruppenzugehörigkeit für den *DDLMSGlobalControl* Dienst.Die restlichen Parameter sind anwendungsspezifisch, ihre Interpretation obliegt dem Anwenderprogramm.
- 3.) Mit dem Dienst *DDLMSChkCfg* wird die Länge der zu übertragenden Prozessdaten vereinbart. In einem Slave können mehrere Gruppen von digitalen und analogen Ein- und Ausgängen vorhanden sein. Das Gerät ist dann i.a. modular aufgebaut. Die Konfigurationsdaten bestehen aus je einem Oktet pro Modul. Darin ist festgelegt
 - die Länge der Daten
 - die Struktur der Daten (Bytes oder 16bit-Wörter)
 - die Übertragungsrichtung (Input oder Output)
 - eine allfällige Forderung nach Konsistenz der Daten
- 4.) Mit einem nochmaligen *DDLMSlaveDiag* Dienst fordert der Master den Slave auf, zu bestätigen, dass er mit den übertragenen Parametern und Konfigurationsdaten einverstanden ist.

Damit ist die Startphase beendet und die Datentransferphase (zyklischer Datenaustausch) beginnt.

Sollen während der Datentransferphase Einstellungen im Slave, wie z.B. Parameterwerte für Kennlinien, Regelalgorithmen etc., geändert werden, besteht nur die Möglichkeit, diese im Datenfeld des Dienstes *DDLMSDataExchange* zusätzlich zu den Prozessdaten zu übertragen. Der dafür zu reservierende Platz im Datenfeld bleibt die meiste Zeit ungenutzt und verringert daher die Effizienz der Datenübertragung. Außerdem muß der Anwender das Protokoll für die Übertragung dieser Parameter selbst entwickeln. Mitte 1996 wurden daher die DDLMS-Dienste erweitert (DPV1).

Erweiterung des Dienstumfanges

Die Dienste *DDLMSRead* und *DDLMSWrite* versehen ihre Daten mit einem Schicht-7-Rahmen. Da nach wie vor keine Objekte und keine Objektlisten existieren, wird die Kennzeichnung der Daten in anderer Weise durchgeführt. Man geht davon aus, dass der Slave modular aufgebaut ist und bildet den Rahmen aus der Funktionskennung, der Modulnummer, einer Kennziffer zur Unterscheidung innerhalb des Moduls und der Länge der zu übertragenden Daten.



WorldFIP (Factory Information Protocol)

Französischer Beitrag zur europäischen Feldbusnorm EN50170

Physikalisches Medium: IEC 1158-2, synchrone Übertragung, Manchesterkodierung

Buszugriffsverfahren: Arbitrator-Producer-Consumer-Verfahren

Telegrammaufbau: Präambel zur Synchronisierung; Violationbits \underline{V} und \overline{V} (abweichend vom Manchestercode wird während der ganzen Bitzeit konstanter Pegel gesendet) garantieren eindeutige Erkennbarkeit der Delimiter; kein explizites Adressfeld, gegebenenfalls wird eine Adresse im Datenfeld untergebracht.

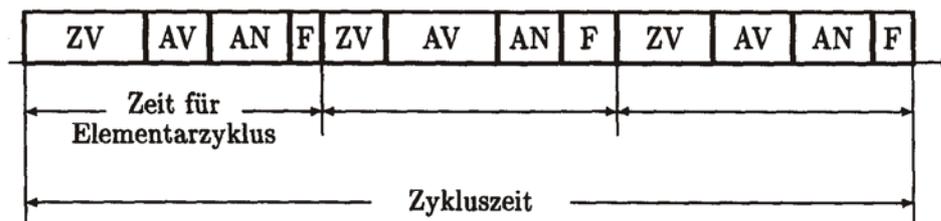
Präambel 10101010	Start Delimiter 1 \overline{V} \overline{V} 10 \overline{V} \overline{V} 0	Control 1 Byte	Datenfeld n Bytes	CRC 2 Bytes	End Delimiter 1 \overline{V} \overline{V} \overline{V} \overline{V} 101
-----------------------------	--	--------------------------	-----------------------------	-----------------------	---

Telegrammaufbau WorldFIP

Dienste der Schicht 2:

Es werden sowohl eine datenorientierte Schnittstelle zur zyklischen und azyklischen Übertragung der Prozessdaten als auch eine nachrichtenorientierte Schnittstelle (Diagnose, Neuparametrierung usw.) bereitgestellt.

Bei der Systemkonfigurierung wird im Arbitrator eine sog. *Scanning Table* erstellt, in der jeder Transfer, der während einer Zykluszeit vorgenommen werden soll, eingetragen ist. Ein Zyklus wird in mehrere Elementarzyklen aufgeteilt. Damit ist es möglich verschiedene Variablen innerhalb eines Zyklus verschieden oft zu übertragen (schnelle und langsame Teilprozesse).



Bedeutung der Abkürzungen:

- ZV** Zeit für zyklische Übertragung von Variablen,
- AV** Zeit für azyklische Übertragung von Variablen,
- AN** Zeit für azyklische Übertragung von Nachrichten,
- F** Füllzeit.

Jeder zyklische Transfer wird vom Arbitrator durch ein Telegramm initiiert, welches im Datenfeld einen 16bit Identifier enthält. Dieser Identifier bestimmt keinen Teilnehmer, sondern ist einem Kommunikationsobjekt (z.B. ein Temperaturwert) zugeordnet (es sind 65536 verschiedene Objekte möglich). Derjenige Teilnehmer (Producer), der dieses Objekt bereitstellt, sendet als Antwort ein Telegramm mit dem zugehörigen Variablenwert im Datenfeld. Da alle Teilnehmer wissen, um welches Kommunikationsobjekt es sich handelt,

können sie (Consumer) diesen Variablenwert dem Antworttelegramm entnehmen, wenn sie ihn benötigen. Es ist kein Quittierungstelegramm vorgesehen.

Möchte ein Producer weiter Information azyklisch anderen Teilnehmern übersenden oder von diesen erhalten, vermerkt er dies in seinem zyklischen Antworttelegramm. In der Zeit AV spricht der Arbitrator diesen Producer wieder an (gleiche ID) und erhält im Datenfeld des Antworttelegramms eine Liste der gewünschten Übertragungen. Der Arbitrator arbeitet diese Liste dann einmal ab. Dabei wird zwischen einer Übertragung von Variablen und Nachrichten unterschieden. Eine azyklische Variablenübertragung wird mit den gleichen Telegrammen abgewickelt wie eine zyklische. Die Übertragung von Nachrichten ist etwas aufwändiger (Sende- und Empfängeradressen, Quittierung usw.).

Die aktionslose Füllzeit dient der Sicherstellung einer konstanten Elementarzykluszeit.

Foundation Fieldbus

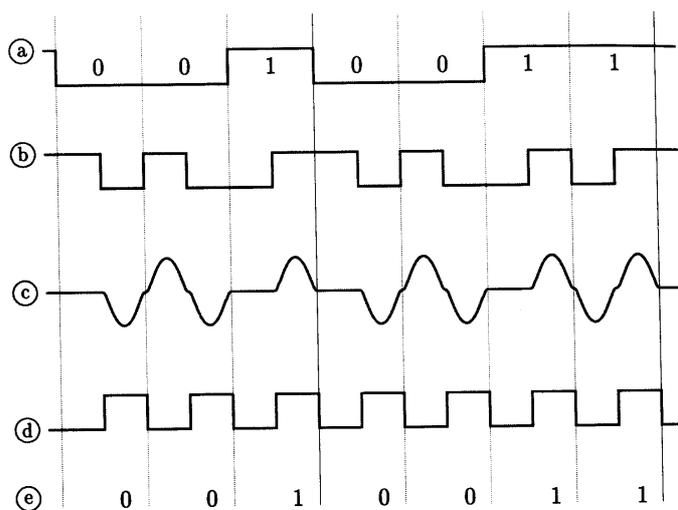
Siemens, Yokogawa, Rosemount und Fisher Control haben 1992 das *Interoperable Systems Project* (ISP) gegründet. Es sollte ein einziger internationaler interoperabler Feldbus für die Prozessautomatisierung entwickelt werden. Noch vor Abschluss der Arbeiten vereinigten sich 1994 die ISP-Foundation und die nordamerikanische Sektion von WorldFIP zur Fieldbus-Foundation (FF). Daraus entstand der **Foundation Fieldbus**.

Die Schicht 1 ist nach IEC 1158-2 ausgelegt. Schicht 2 lehnt sich sehr stark an WorldFIP an, während die Schicht 7 identisch mit PROFIBUS-FMS ist. Darüber hinaus werden das *Function-Block-Modell* und die *Device Description Language* stark forciert.

Aktor Sensor Interface (ASI)

In den Jahren 1990 bis 1993 für den Anschluss einfacher Sensore und Aktoren mit dem Ziel entwickelt, besonders einfach und preisgünstig zu sein.

Physikalisches Medium



Ungeschirmte Zweidrahtleitung mit 1.5mm² Querschnitt. Neben dem Signal wird auch eine Hilfsenergie mit bis zu 8A bei 24VDC mit übertragen.

Das Signal wird Manchestercodiert. Durch geeignete Filterung entstehen sin²förmige Pulse entsprechender Polarität. Die Signalspannung bewegt sich zwischen -2V und +2V und wird

der Versorgungsspannung von 24V überlagert. Das Signal ist Gleichstromfrei, niederfrequent und enthält die Taktinformation (synchrone Übertragung).

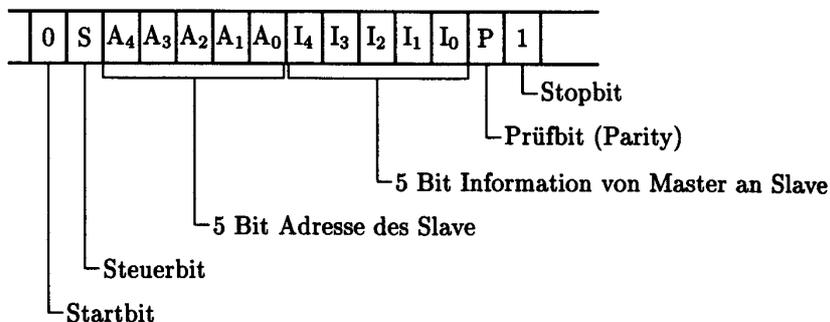
- Übertragungsrate: 167 kbit/s
- Max. Leitungslänge: 100m

Buszugriffsverfahren: Master-Slave-Verfahren

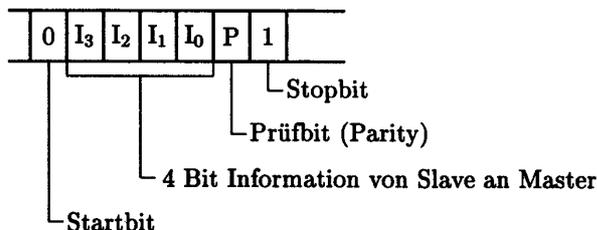
Telegrammaufbau

- Start- und Stoppenkung bestehen aus je einem Bit
- Steuerbit zur Unterscheidung zwischen Kommandos und Daten
- Adressfeld 5 Bit, Adresse 0 hat Sonderfunktion → 31 Slaves adressierbar
- Datenfeld für Daten (4 Bit), Parameter (4 Bit) und Kommandos (5 Bit)
- Paritätsbit gerade Parität

Telegramm des Masters



Antwort des Slaves



S	I ₄	I ₃ I ₂ I ₁ I ₀	Bedeutung
0	0	0000 ... 1111	4 Bit Daten I ₃ bis I ₀
0	1	0000 ... 1111	4 Bit Parameter I ₃ bis I ₀
0		00001 ... 11111 (neue Adresse)	neue Adresse einstellen (Stationsadresse A ₄ ... A ₀ muß dabei gleich 0 sein)
1	0	0000	Kommando „Stationsadresse auf 0 setzen“
1	1	1100	Kommando „Reset des Slave“
1	1	0000	Kommando „E/A-Konfigurationscode lesen“
1	1	0001	Kommando „Identifikationscode lesen“
1	1	1110	Kommando „Status lesen“
1	1	1111	Kommando „Status lesen und löschen“

Schnittstelle zwischen Kommunikationssystem und Anwendung

Datenorientierte Schnittstelle (Prozessabbild im Master). Es existiert eine Inbetriebnahmephase, ähnlich der bei PROFIBUS-DP, allerdings wesentlich einfacher. Jeder Slave besitzt zwei 4Bit-Kenngrößen zur Identifizierung seines Typs.

- E/A-Konfigurationscode: Die 4 Informationsbits I_0 bis I_3 können einem 4 Bit Port im Slave zugeordnet werden. Jedes Bit dieses Ports kann entweder Eingang, Ausgang oder bidirektional sein. Der Konfigurationscode beschreibt den Aufbau diese Ports. Von den $3^4 = 81$ Möglichkeiten werden 16 ausgewählt und den Werten des Konfigurationscodes zugeordnet.
- Identifikationscode: Beschreibt, wie die einzelnen Datenbits zu interpretieren sind. In Profilen wird festgelegt, dass beispielsweise der Identifikationscode „0“ einen Slave mit 4 einzelnen Sensoren beschreibt und der Identifikationscode „1“ einen Slave, bei dem I_0 das Messsignal, I_1 eine Warnmeldung, I_2 eine Bereitschaftsanzeige und I_3 eine Selbsttestauslösung (Ausgang) bedeuten.

Zwischen den einzelnen Datenaustauschzyklen werden immer anstehende azyklische Kommandos eingefügt um den Status zu lesen, Parameter zu ändern. Auch können neue Slaves hinzugefügt oder defekte Slaves ausgetauscht werden, ohne den Betrieb zu unterbrechen.