Bitübertragungsschicht

Bezeichnungen

Zeichen Element des Zeichenvorrates

Code Abbildung zwischen zwei Zeichenvorräten

Takt Zeiteinheit bei Sender und Empfänger

Digitale Basisband Modulation

Digitale Daten werden durch ein amplituden- und zeitdiskretes Verfahren übertragen.

Beispiel: Die Elemente des Zeichenvorrates $\{0, 1, 2\}$ werden auf Spannungen 0V, 1V, 2V abgebildet und im Zeittakt von 1s mit einem Kupferkabel übertragen.

Mögliche Probleme hierbei sind abhängig vom Übertragungsmedium

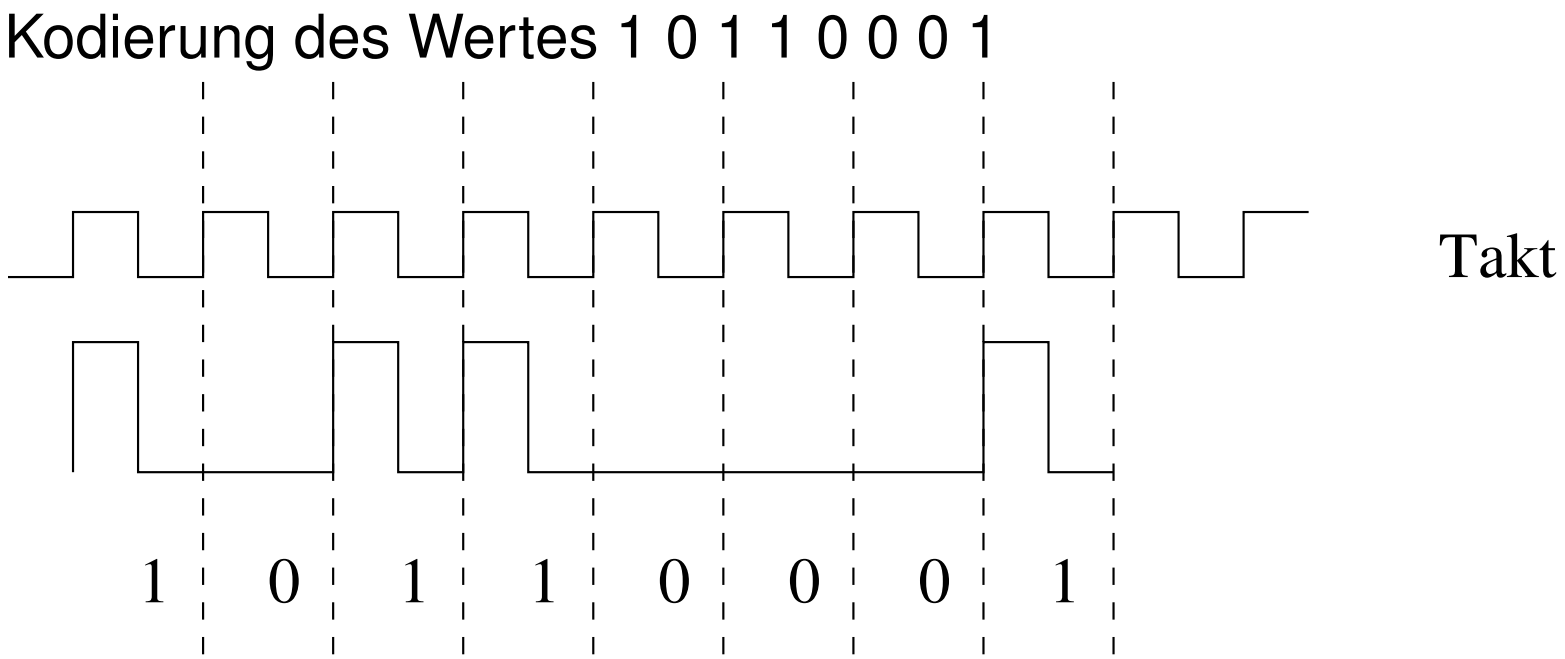
- Synchronisierung von Sender und Empfänger (Taktrückgewinnung)
- Gleichspannungsfreiheit

Synchronisierung

Sender und Empfänger können sich bei jedem Amplitudenwechsel synchronisieren.

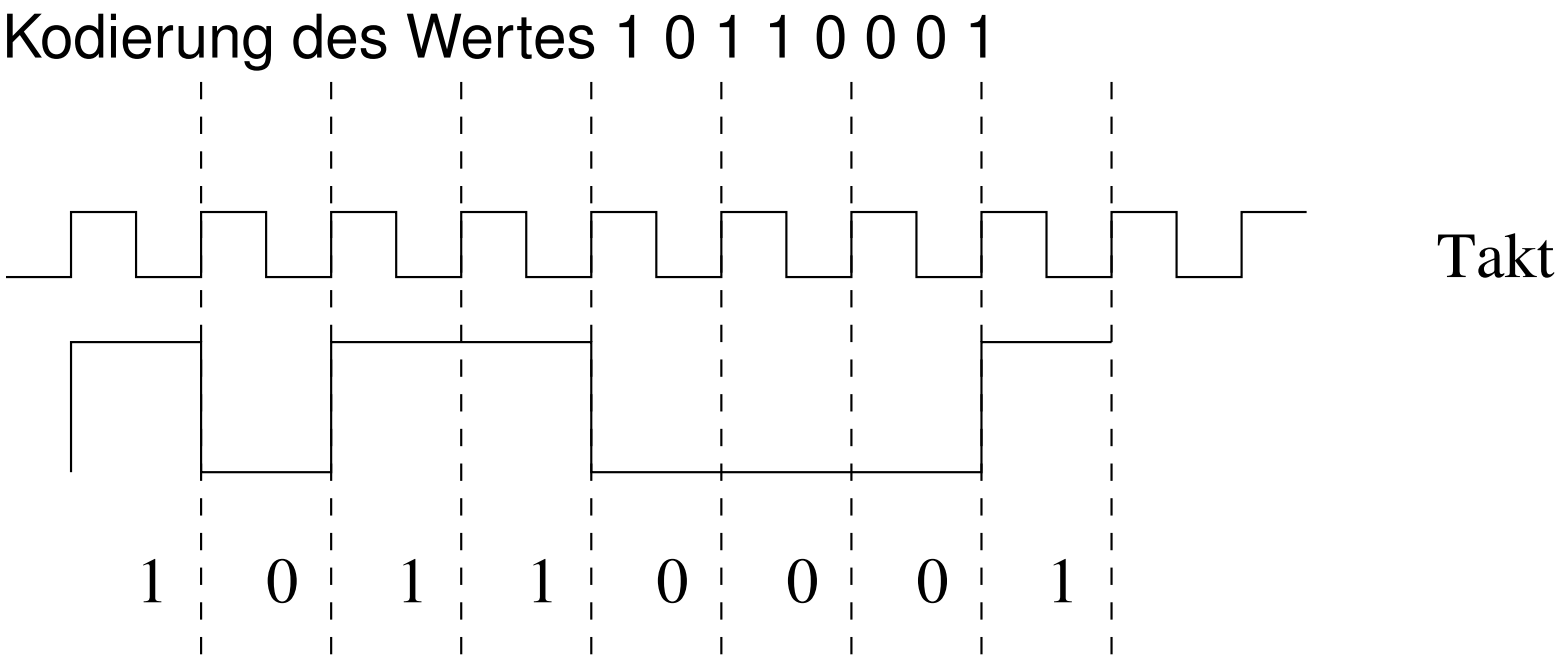
1. **Synchronverfahren:** Sender und Empfänger haben während der gesamten Übertragung synchronen Takt.
2. **Asynchronverfahren:** Sender und Empfänger werden beim Start eines zu übertragenden Datenblocks synchronisiert.
3. **Start/Stop Verfahren:** Spezialfall von (2), jedes Zeichen wird durch ein Start/Stop Signal zur Synchronisierung begrenzt.

Return To Zero (RZ)



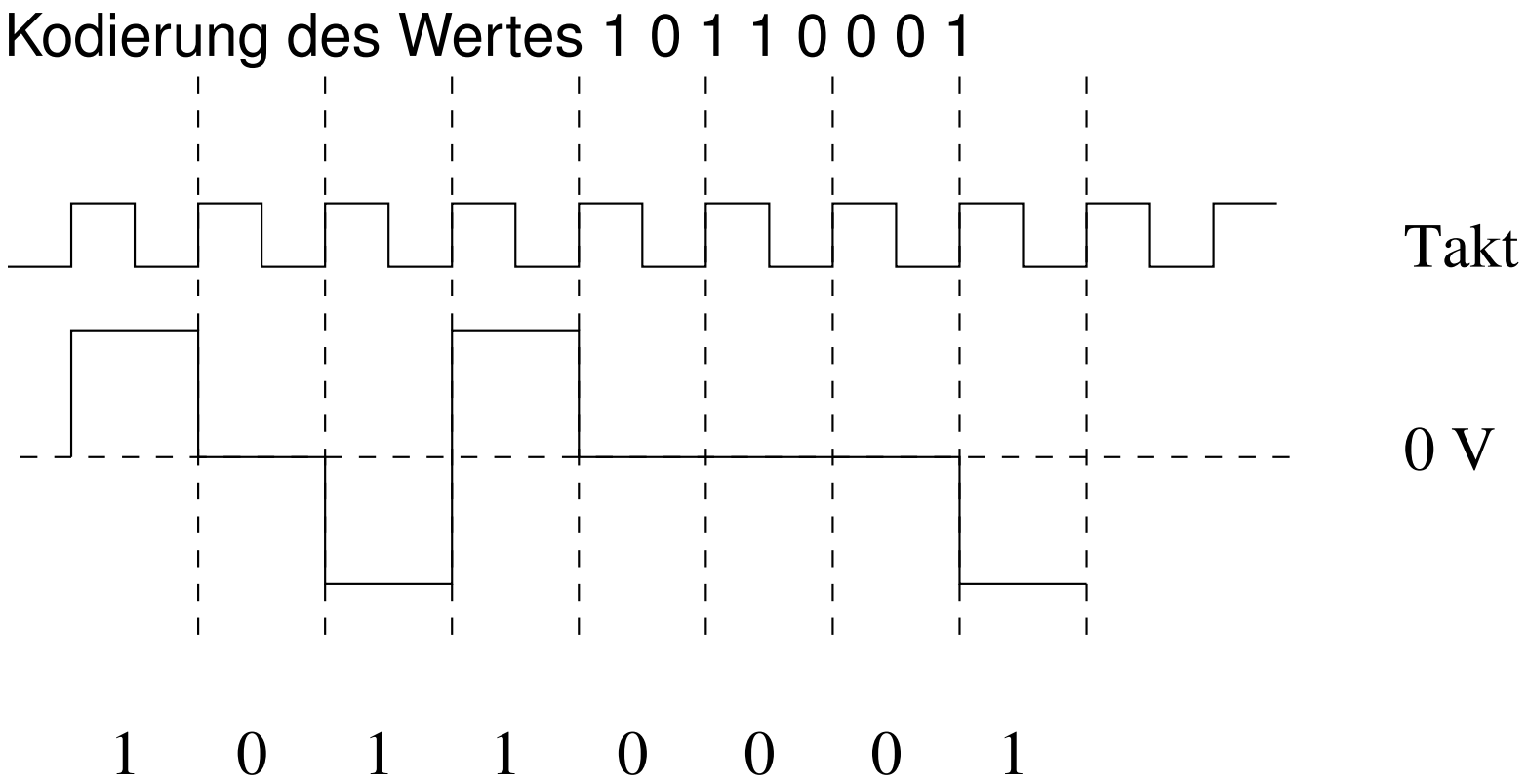
- Keine automatische Taktrückgewinnung, nicht gleichspannungsfrei

No Return To Zero (NRZ)



- Keine automatische Taktrückgewinnung, nicht gleichspannungsfrei

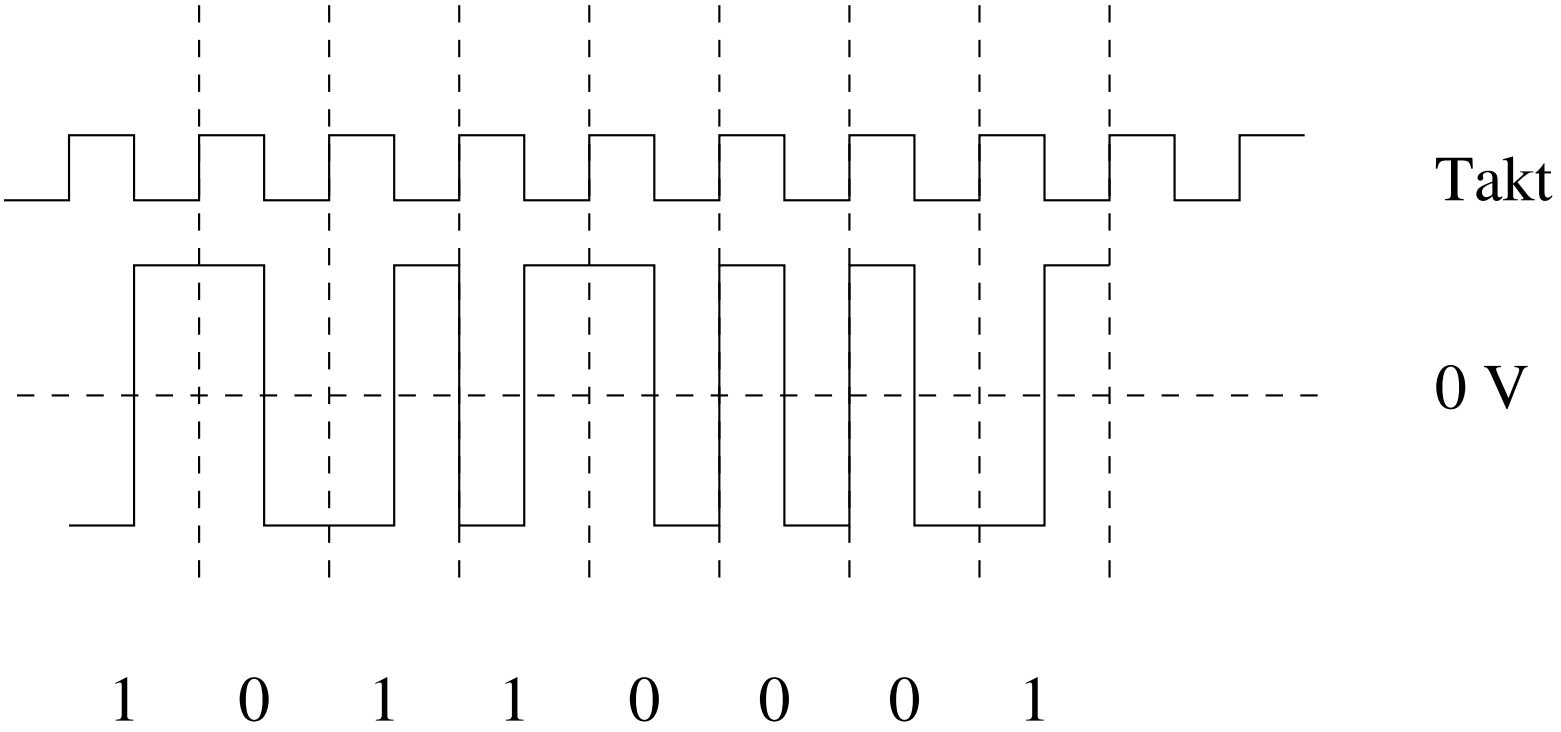
Bipolare Kodierung (AMI Kodierung)



- Keine automatische Taktrückgewinnung, gleichspannungsfrei

Manchester Kodierung

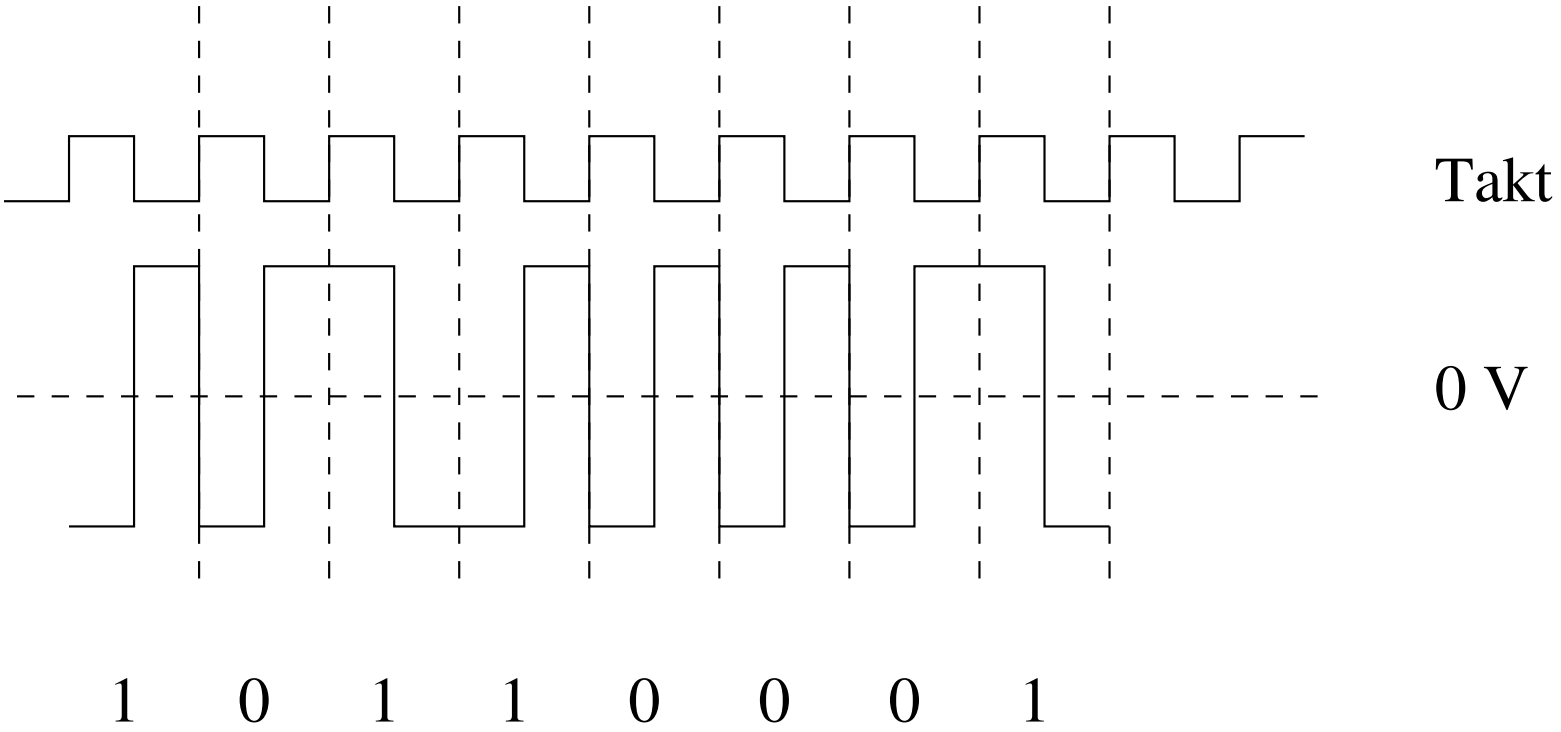
Kodierung des Wertes 1 0 1 1 0 0 0 1,
steigende Flanke kodiert 1



- Automatische Taktrückgewinnung, gleichspannungsfrei

Differentielle Manchester Kodierung

Kodierung des Wertes 1 0 1 1 0 0 0 1,
Flankenwechsel bei Bitanfang bei 0, dadurch sind auch invertierte Signale dekodierbar



- Automatische Taktrückgewinnung, gleichspannungsfrei

Übersicht der Eigenschaften der Kodierverfahren

- **RZ (Return to Zero):** Keine automatische Taktrückgewinnung, nicht gleichspannungsfrei
- **NRZ (No Return zo Zero):** Keine automatische Taktrückgewinnung, nicht gleichspannungsfrei
- **AMI (Bipolare Kodierung):** Keine automatische Taktrückgewinnung, gleichspannungsfrei
- **Manchester Kodierung:** Automatische Taktrückgewinnung, gleichspannungsfrei
- **Differentielle Manchester Kodierung:** Automatische Taktrückgewinnung, gleichspannungsfrei

Kanalkapazität *(Channel Capacity)*

Theorem (Nyquist)

In einem bandbegrenzten, störungsfreien Übertragungskanal mit Bandbreite $B[Hz]$ und L diskreten Signalstufen ist die maximale Datenübertragungsrate C_N

$C_N = 2 \cdot B \cdot \log_2(L)\,[bit/s]$

Theorem (Shannon-Hartley)

In einem bandbegrenzten, gestörten Übertragungskanal mit Bandbreite $B[Hz]$ ist die maximale Datenübertragungsrate C_S

$C_S = B \cdot \log_2(1 + S/N)\,[bit/s]$,

wobei S/N das Signal-Rausch-Verhältnis bezeichnet.

Anwendung

- Die Verfahren RZ, Manchester-Kodierung und Differentielle Manchester Kodierung benötigen die doppelte Bandbreite von AMI und NRZ.
- Bei gegebenem Kanal (Bandbreite) ergibt das eine Halbierung des maximalen Durchsatzes.
- Aber: Taktrückgewinnung muß bei AMI und NRZ auf anderem Wege sichergestellt werden.

4B5B Kodierung

Jeweils 4 Bit des Eingangsdatenstromes werden auf 5 Bit abgebildet, dadurch werden genügend Amplitudenwechsel sichergestellt:

Hex	4B	5B	Hex	4B	5b
0	0000	11110	8	1000	10010
1	0001	01001	9	1001	10011
2	0010	10100	A	1010	10110
3	0011	10101	B	1011	10111
4	0100	01010	C	1100	11010
5	0101	01011	D	1101	11011
6	0110	01110	E	1110	11100
7	0111	01111	F	1111	11101

Serielle Schnittstelle

- Zeichen bestehen aus 5-8 Bit (üblicherweise 7-8 Bit)
- Die Übertragung erfolgt bitweise, LSB (Least Significant Bit) zuerst.
- Spannungen zwischen -3V und -15V werden als 1 dekodiert.
- Spannungen zwischen +3V und +15V werden als 0 dekodiert.
- Jedes Zeichen wird bit einem Startbit (Wert 0) begonnen (Start/Stop Verfahren)
- Jedes Zeichen muß mit 1, 1.5 oder 2 Stopbits (Wert 1) abgeschlossen werden.
- Außerhalb der Übertragung liegt die Spannung für Wert 1 an.

Parität *(Parity)*

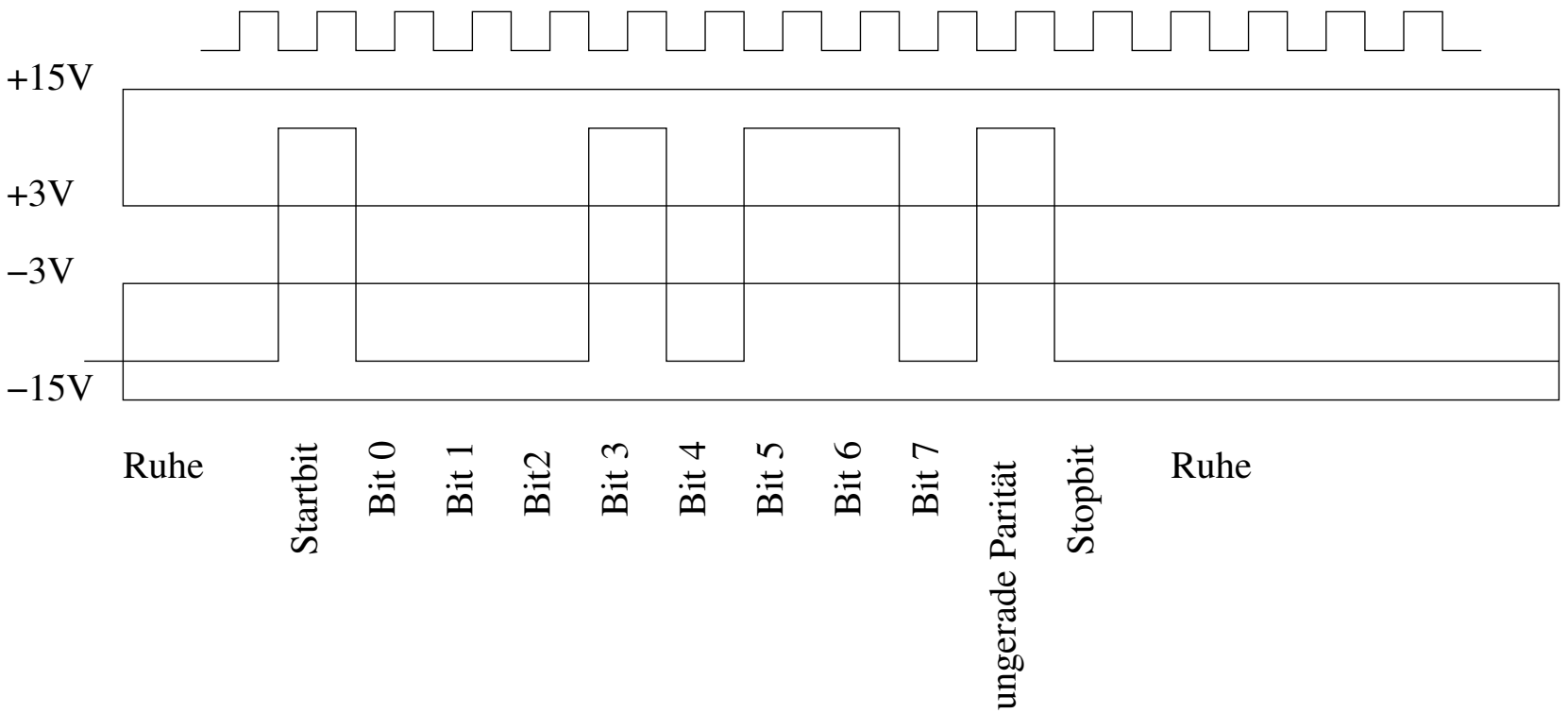
Die Übertragung eines Zeichens kann abgesichert werden durch ein Paritätsbit, das dem Zeichen als weiteres Bit angefügt wird. Man unterscheidet:

- O Ungerade Parität (Odd partiy)
- E Gerade Parität (Even parity)
- N Kein Paritätsbit (No parity)

Das Paritätsbit berechnet sich so, daß die Summe der Bits inklusive Parität ungerade (O) bzw. gerade (E) ist.

Beispiel

Übertragung des Wortes 10010111 mit
8Bit, Ungerader Parität, 1 Stopbit (8O1)



Bemerkungen

- Es folgen maximal 9 gleiche Bit aufeinander, danach können sich Sender und Empfänger neu synchronisieren.
- Das Protokoll ist nicht gleichspannungsfrei
- Es wurde nur das Verhalten auf Rx und Tx Leitungen beschrieben.
- Die maximale Kabellänge hängt von Übertragungsgeschwindigkeit, elektrischem Widerstand, Kapazität und den verwendeten Sendern und Empfängern ab. Der Standard schreibt nur eine Kabelkapazität von weniger als 2500 pF vor.

Sicherungsschicht

Sicherungsschicht *(Data Link Layer)*

Eine Verbindung der Sicherungsschicht bietet die Mittel zum Datenaustausch zwischen Netzwerkknoten, die durch Adressen der Sicherungsschicht identifiziert werden.

- Rahmenbildung für die benutzte Bitübertragungsschicht
- Serialisierung der Rahmen
- Fehlererkennung/Fehlerbehebung bei Übertragungsfehlern der Bitübertragungsschicht
- Fehlerbenachrichtigung an die Vermittlungsschicht bei nicht behebbaren Fehlern
- IEEE sieht hier die Mehrfachzugriffsverfahren (Multiple Access)

Dienste und Methoden der Sicherungsschicht

Dienste der Sicherungsschicht

- Unbestätigter, verbindungsloser Dienst
- Bestätigter, verbindungsloser Dienst
- Bestätigter, verbindungsorientierter Dienst

Methoden in der Sicherungsschicht

- Rahmenbildung
- Fehlererkennung und Fehlerkorrektur
- Flußkontrolle
- Mehrfachzugriff (Multiple Access)

Dienste der Sicherungsschicht (1)

Unbestätigter, verbindungsloser Dienst

- Versendet Rahmen sind unabhängig voneinander
- Es wird keine logische Verbindung aufgebaut
- Keine Detektion fehlerhafter Rahmen (in der Sicherungsschicht)
- Anwendungsgebiete sind z.B. fehlerarme Übertragungsmedien (Glasfaser, Kabel) und real-time Daten (Audio/Video Kommunikation)

Dienste der Sicherungsschicht (2)

Bestätigter, verbindungsloser Dienst

- Versendet Rahmen sind unabhängig voneinander
- Es wird keine logische Verbindung aufgebaut
- Detektion fehlerhafter Rahmen
- Bestätigung (Acknowledgement) für empfangene Rahmen
- Anwendungsgebiete sind z.B. unzuverlässige Übertragungsmedien (Funk,...)

Dienste der Sicherungsschicht (3)

Bestätigter, verbindungsorientierter Dienst

- Es wird eine dedizierte Verbindung zwischen Sender und Empfänger aufgebaut
- Numerierung einzelner Rahmen
- Detektion fehlerhafter Rahmen und falscher Reihenfolge
- Bestätigung (Acknowledgement) für empfangene Rahmen
- Garantie für Vermittlungsschicht für fehlerfreie Rahmen in der richtigen Reihenfolge

Rahmenbildung

Methoden zur Rahmenbegrenzung

- Zeichenzahl
  In der Sicherungsschicht allein nur noch sehr selten verwendet, da Übertragungsfehler der Rahmenlänge nicht ohne andere Verfahren erkannt bzw. korrigiert werden können.
- Flagbytes
- Start- und Endflag
- Kodierregeln der Bitübertragungsschicht
  Nutzung von Redundanz in der Bitübertragungsschicht. Beispiele:
  - Ungültige Übergänge bei der Manchester Codierung
  - 100base-TX verwendet die Bitmuster 11000 und 10001 des 4B5B Codes

Flag Bytes

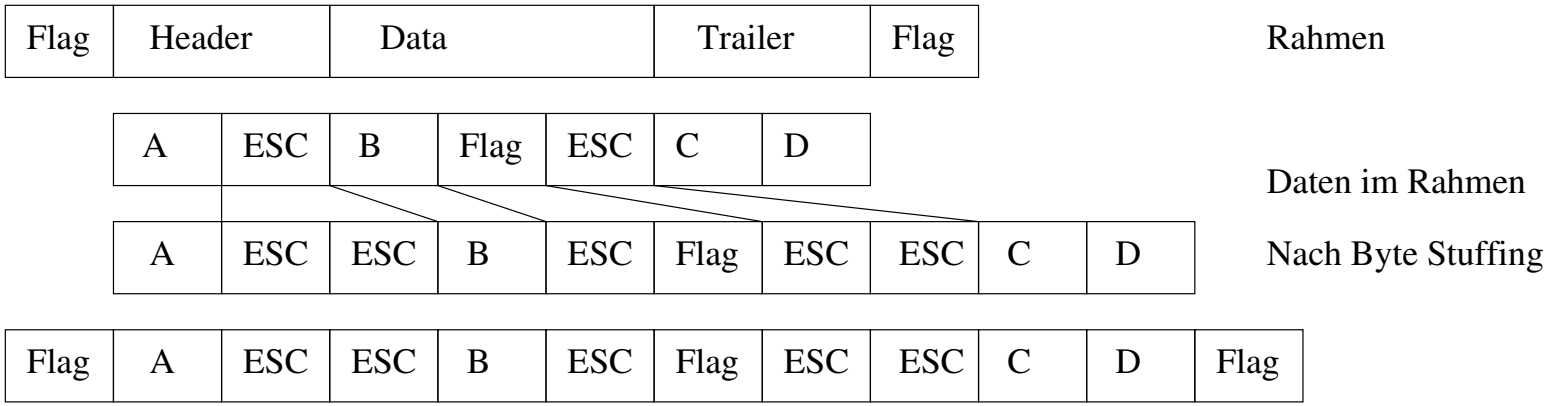
Idee Kennzeichnung von Rahmenanfang und Ende durch spezielle Bitmuster.

Bei zeichenorientierter Bitübertragungsschicht (vgl. z.B. EIA-232) muß die Kennzeichnung durch Zeichen (Flagbytes) erfolgen, da Einzelbits nicht übertragen werden können.

Problem Was passiert, wenn das verwendete Byte im Datenstrom auftaucht?

Byte Stuffing

Ein Byte (nicht das Flagbyte) wird als sogenanntes Escape Zeichen (ESC) gewählt. Dieses Zeichen wird vor der Rahmenbildung vor jedem Flagbyte und vor jedem Escape Zeichen in den Datenstrom eingefügt wird.



Eventuelle Prüfsummen (z.B. im Trailer) werden vor dem Byte Stuffing berechnet und ebenso behandelt.

Start- und Endflag mit Bit Stuffing

- Bei bitorientierter Übertragung kann das Flagbyte Bytegrenzen im Datenstrom überlappen, damit ist keine Erkennung des Rahmenanfangs bei verlorener Synchronisation möglich.
- Daher: Verwende Bitmuster der Form **0111...1110** der Länge n als Flag und stelle sicher, daß in den Daten nie $n-2$ mal **1** gesendet wird.
- Erkennt der Sender das $n-3$ mal **1** gesendet wurde, fügt er vor dem nächsten Bit ein Bit **0** in den Datenstrom ein.
- Analog entfernt der Empfänger die **0**, die auf $n-3$ mal **1** folgt.

Beispiel Bit Stuffing

Flag Bitmuster:	**01110**
Bitfolge Daten:	**0101111101111100110**
Daten nach Bit Stuffing:	**0101101101011010011000**
Gesendeter Rahmen	**01110 0101101101011010011000 01110**

Das Muster **01110** taucht nur am Rahmenanfang und Ende auf. Im Rahmen folgen nie mehr als 2 mal **1** aufeinander. Damit kann der Empfänger den Anfang des nächsten Rahmens sicher finden.

Fehlermanagement

Polynome über $\mathbb{Z}_2$

Die Menge der Polynome mit Koeffizienten aus $\mathbb{Z}_2$ bilden einen kommutativen Ring.
Beispiele: Seien $p(x) = x^4 + x^3 + 1$ und $q(x) = x^3 + x$, dann ist

$$p(x) + q(x) = x^4 + x + 1$$
$$p(x) - q(x) = x^4 + x + 1$$
$$p(x) * q(x) = x^7 + x^6 + x^5 + x^4 + x^3 + x$$

Bemerkung: Die Addition im $\mathbb{Z}_2$ entspricht der logischen Verknüpfung “exklusiv oder” (XOR), die Multiplikation der Operation “und” (AND).

Beispiel Polynomdivision mit Rest

Division $p(x)/q(x)$ mit Rest ergibt: $x+1$ Rest x^2+x+1:

$$
\begin{array}{llll|l}
(x^4 & +x^3 & & +1) & : (x^3+x) = x+1 \text{ Rest } x^2+x+1 \\
x^4 & & +x^2 & & \\
\hline
 & x^3 & +x^2 & +1 & \\
 & x^3 & & +x & \\
\hline
 & & x^2 & +x \quad +1 &
\end{array}
$$

Bemerkung:

Der Grad des Restes ist immer kleiner als der Grad des Divisors.

Anwendung auf Prüfsummen

Sei q ein festes Polynom vom Grad n über $\mathbb{Z}_2$, das sogenannte Generatorpolynom.

Identifiziert man die Bits eines Rahmens mit den Koeffizienten eines Polynoms p über $\mathbb{Z}_2$, dann ist der Rest r der Division $p(x) * x^n / q(x)$ ein Polynom vom Grad $n - 1$. Übertragen werden dann die Koeffizienten von $p(x) * x^n - r(x)$, d.h. die Bits des Rahmens mit angefügten n Prüfbits.

Beispiel

Wir verwenden $n = 3$, $q(x) = x^3 + x$ und wollen **11101** gesichert übertragen.

$$
\begin{array}{cccccccc}
1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\hline
 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
\hline
 & & 0 & 1 & 1 & 0 & 0 & 0 \\
 & & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
 & & & 1 & 1 & 0 & 0 & 0 \\
 & & & 1 & 0 & 1 & 0 & 0 \\
\hline
 & & & & 1 & 1 & 0 & 0 \\
 & & & & 1 & 0 & 1 & 0 \\
\hline
 & & & & & 1 & 1 & 0
\end{array}
\quad : 1010 = 11011 \text{ Rest } 110
$$

Wir übertragen also **11101 110** .

Eigenschaften

Der Empfänger bildet aus den Koeffizienten des empfangenen Rahmens ebenfalls ein Polynom $t(x)$ und berechnet $t(x)/q(x)$. Bei fehlerfreier Übertragung gilt $t(x) = p(x) * x^n - r(x)$.

Sei $e(x) := t(x) - (p(x) * x^n - r(x))$.

1. $p(x) * x^n - r(x)$ wird von $q(x)$ ohne Rest geteilt.
2. Der Rest von $(t(x) - e(x))/q(x)$ ist gleich dem Rest von $e(x)/q(x)$.
3. Ist $x + 1$ ein Faktor von $q(x)$, werden alle Übertragungsfehler mit ungerader Anzahl fehlerhafter Bits erkannt.
4. Mit einem Polynom $q(x)$ vom Grad n und $q(0) = 1$ werden alle Burstfehler bis zu n Bit erkannt.

Beweis zu 3.

Annahme: $e(x)$ hat ungerade Anzahl Terme und $x+1$ teilt $e(x)$.

Bemerkung: Damit der Fehler unerkannt bleibt, muß $e(x)$ von jedem Faktor von $q(x)$ geteilt werden, also insbesondere von $x+1$.

Dann existiert ein Polynom $e'(x)$, so daß $e(x) = (x+1) \cdot e'(x)$ gilt.

Durch Einsetzen erhält man $e(1) = (1+1) \cdot e'(1) = 0$.

Da aber $e(x)$ eine ungerade Zahl an Termen hat, ist $e(1) = 1$, womit die Annahme falsch sein muß.

Beweis zu 4.

Für einen Burstfehler der Länge $k \leq n$ existiert ein $i \geq 0$, so daß gilt:

$e(x) = x^i \cdot (x^{k-1} + \ldots + 1)$.

Da $q(0) = 1$, ist x^i kein Faktor von $q(x)$ für jedes $i > 0$.

Da aber der Grad von $(x^{k-1} + \ldots + 1)$ wegen $k \leq n$ kleiner als der Grad von q ist, wird $(x^{k-1} + \ldots + 1)$ nicht von $q(x)$ geteilt.

Beispiele

Hier sine einige gebräuchliche Generatorpolynome:

- **CRC-16:** Benutzt z.B. von HDLC (high-level data link control, ISO 3309)

$$q(x) = x^{16} + x^{15} + x^2 + 1$$

- **CRC-CCITT:** Benutzt z.B. von PPP (RFC 1662)

$$q(x) = x^{16} + x^{12} + x^5 + 1$$

- **CRC-32:** Benutzt z.B. von Ethernet (IEEE 802.3) und PPP

$$\begin{aligned} q(x) = & x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} \\ & + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1 \end{aligned}$$

Bezeichnungen

$W_n = (b_i, i = 0, \ldots, n-1, b_i \in \{0, 1\})$ ein n-stelliges Binärwort.
Dann heißt $\sum_{i=0}^{n-1} b_i$ das Hamming Gewicht von W_n.

Für zwei n-stelliges Binärworte $V_n = (a_0, \ldots, a_{n-1})$ und
$W_n = (b_0, \ldots, b_{n-1})$ heißt das **Hamming Gewicht** von

$$V_n \oplus W_n := (a_0 \oplus b_0, \ldots, a_{n-1} \oplus b_{n-1})$$

der **Hamming Abstand** von V_n und W_n. Hierbei bezeichnet $\oplus$ die XOR-Verknüpfung.

Das Hamming Gewicht von $(1, 0, 1, 1, 0)$ ist 3.
Der Hamming Abstand von $(1, 0, 1, 1, 1)$ und $(1, 1, 0, 1, 1)$ ist 2.

Bestehe das Quellalphabet aus 2^n n-stelligen Binärworten, die so auf $n + r$-stellige Binärworte abgebildet werden sollen, daß k falsch übertragene Bit korrigiert werden können.

Sei z.B. $n = 1$, $k = 1$. Der Versuch mit $r = 1$ führt z.B. zu

$$(0) \rightarrow (0, 0)$$
$$(1) \rightarrow (1, 1)$$

Wird hier ein Bit falsch übertragen, kann nicht mehr entschieden werden, was das Ursprungswort war.

Der Versuch mit $r = 2$ führt dagegen z.B. zu

$$(0) \rightarrow (0, 0, 0)$$
$$(1) \rightarrow (1, 1, 1)$$

Hier ist eine Dekodierung auch bei einem falschen Bit möglich.

Allgemein gilt: Seien V_n und W_n zwei n-stellige Zeichen des Quellalphabetes.

Seien V_{n+r}, W_{n+r} die zugehörigen $n+r$-stelligen Zeichen des Kodes.

Damit bei Fehlübertragung von k Bit V_{n+r} und W_{n+r} wieder sicher den Zeichen des Quellalphabetes zugeordnet werden können, muß der Hamming Abstand von V_{n+r} und W_{n+r} mindestens $2 \cdot k + 1$ betragen.
Damit eine solche Abbildung existiert muß gelten (Hamming Bedingung, 1950):

$$2^n \cdot \sum_{i=0}^{k} \binom{n+r}{i} \leq 2^{n+r}$$

Korrektur von Einzelfehlern

Für $k = 1$ folgt aus der Hamming Bedingung
$2^n \cdot (1 + n + r) \leq 2^{n+r}$, also $(1 + n + r) \leq 2^r$.

Beispiel: Für $n = 4$ werden wegen

$$1 + 4 + 3 \leq 2^3$$
$$\text{aber } 1 + 4 + 2 \not\leq 2^2$$

mindestens $r = 3$ Prüfbits gebraucht.

Für Einzelfehler hat Hamming ein Kodierverfahren angegeben, daß diese Schranke einhält.

Hamming Code

Betrachte ein Schema mit $n + r$ Spalten $1, \ldots, n + r$. Bezeichne weiter $\odot$ die bitweise AND Verknüpfung. Seien die r Prüfbits in den Spalten $2^0, 2^1, \ldots, 2^{r-1}$ und die n Bit Nutzdaten in den übrigen Spalten, dann bildet Spalte 2^i die Parität für alle Spalten s mit $s \odot 2^i = 2^i$.

Beispiel: Prüfbit 0 in Spalte $2^0 = 1$ bildet die Parität für die Spalten $1, 3, 5, 7, \ldots$.

Prüfbit 1 in Spalte $2^1 = 2$ bildet die Parität für die Spalten $2, 3, 6, 7, 10, 11, \ldots$.

Umgekehrt wird Spalte $25 = 2^4 + 2^3 + 2^0$ abgesichert durch die Prüfbits in den Spalten $1, 8$ und 16.

Beispiel

Seien $n = 4$ und $r = 3$ und wir benutzen gerade Parität.

Spalte(dezimal)		1	2	3	4	5	6	7
Spalte(binär) s		001	010	011	100	101	110	111
Prüfbit		↓	↓		↓			
Bsp. 1:	Daten 1011	0	1	1	0	0	1	1
Bsp. 2:	Daten 0110	1	1	0	0	1	1	0
Bsp. 3:	Daten 0001	1	1	0	1	0	0	1

Bemerkung: Bis auf die Prüfbits sind alle Bits in mindestens zwei Paritätsbits eingeflossen.

Dekodieren der Nachricht

- Der Empfänger berechnet analog zum Sender die Prüfbits.
- Sind alle Prüfbits identisch mit den empfangenen Bits, ist die Nachricht fehlerfrei übertragen worden.
- Im Fall, daß das Bit in Spalte $s = \sum_{i=0}^{r-1} b_i 2^i$ falsch übertragen worden ist, stimmen die empfangenen Paritätsbits in den Spalten $\{b_i 2^i,\ b_i = 1, i = 0, \ldots, r-1\}$ nicht mit den gesendeten Daten überein.
  Beispiel: Fehler in Spalte $s = 5 = 2^0 + 2^2$
  falsche Paritätsbits in Spalten $2^0 = 1$ und $2^2 = 4$.
- Der Empfänger invertiert das Bit in Spalte s und extrahiert die Nutzdaten.

Flußkontrolle (flow control)

Motivation

- Sendet der Sender schneller als der Empfänger die Daten verarbeiten kann, gehen Teile verloren.
- Kann ein Rahmen nicht dekodiert werden, fehlt ebenso ein Stück im Datenstrom.
- Höhere Schichten (Transportschicht) können Fehler auf unteren Schichten bei Bedarf korrigieren.
- Die Sicherungsschicht kann durch Flußkontrolle den Prozeß auf Ebene der Rahmen optimieren. Dadurch müssen nicht komplette Pakete wiederholt übertragen werden.
- Die Flußkontrolle der Sicherungsschicht sollte abschaltbar sein, da für einige Dienste (z.B. Sprachübertragung) Verzögerungen schlimmer sind als Datenverluste.

Stop-and-Wait Protokoll

- Nach jedem übertragenen Rahmen wartet der Sender auf eine Bestätigung (ACK) des Empfängers.
- Vorteil: Extrem einfach zu implementieren, benötigt beim Empfänger wenig Resourcen.
- Nachteil: Das Protokoll ist ineffizient.

Sliding Window Protokoll

- Der Empfänger hat einen Puffer (sog. Fenster) mit Platz für $n-1$ Rahmen.
- Rahmen werden mit $0, 1, \ldots, n-1, 0, 1, \ldots, n-1, 0, \ldots$ modulo n numeriert.
- Der Sender kennt den Wert n und weiß, daß er $n-1$ Rahmen senden darf, ohne daß einer der Rahmen quittiert worden ist.
- Der Empfänger schickt im ACK die Nummer des ersten nicht verarbeiteten Rahmens.

Beispiel mit Puffer für 6 Rahmen,d.h. $n = 7$: