

### Grundlagen

Die serielle Schnittstelle dient dem Datenaustausch zwischen Computern und Peripheriegeräten. Bei einer seriellen Datenübertragung werden die Bits nacheinander (= seriell) über eine Leitung übertragen.

Die seriellen Schnittstellen werden mit COM1, COM2, ... bezeichnet. Die Kommunikation über die serielle Schnittstelle erfolgt in der Regel **asynchron**, d.h. es gibt keinen gemeinsamen Takt / keine Taktleitung zwischen dem Sender und dem Empfänger.

Ein Datenstrom wird mit einer Baudrate übertragen, die für die Schnittstellenkarte beim Sender und beim Empfänger für eine Übertragung festgelegt werden kann. Die Baudrate beim Sender und beim Empfänger sollte logischerweise gleich eingestellt sein. Typische Werte für die Baudrate sind 9600 Baud und 19200 Baud.

Ein **Datenstrom** (z.B. für die Zeichen eines Textes) wird zu einem Zeitpunkt beim Sender erzeugt und an den Empfänger gesendet. Jedes Zeichen wird einzeln und nacheinander in jeweils einem **Datenwort** von 8 Bit = 1 Byte übertragen. Die Codierung der Zeichen in dem Datenwort erfolgt im bekannten 7-Bit ASCII-Code.

Die Synchronisation der ganzen Übertragung eines Datenwortes (also wann beginnt und endet ein Datenwort) erfolgt durch Ergänzungen der eigentlichen Daten mit **Steuerinformationen**, die voran- und nachgestellt werden. Alles zusammen nennt man den **Datenrahmen**:

Der Datenrahmen besteht immer aus **genau einem Startbit (ist immer logisch 0)**, das den Beginn einer Dateneinheit anzeigt. Das funktioniert deshalb, weil die Empfangsleitung im Ruhezustand auf logisch 1 liegt.

Dann werden die 8 Bit der Daten (das Datenwort) eingesetzt. Bei der Übertragung eines **Datenwortes** werden **zuerst das LSB** und **zuletzt das MSB** übertragen.

Zusätzlich wird eine sog. **Parität** vereinbart. Die Parität ist ein einfacher und damit leider auch wenig leistungsfähiger Schutz gegen Übertragungsfehler. Durch die Parität können nur einfache Bitfehler zuverlässig erkannt werden. Bündelfehler mit mehreren gestörten Bit werden mit 50-prozentiger Wahrscheinlichkeit nicht erfasst. Die Parität eignet sich deshalb nur für kurze und wenig störanfällige Übertragungswege. Der Vorteil der Parität ist der, dass alle seriellen Schnittstellenbausteine die Funktion der Erzeugung und Prüfung von Paritäts-Bit hardwaremäßig unterstützen.

Die Parität zeigt sich im Datenrahmen als das **Paritätsbit**, welches immer dann genau auf das Datenwort folgt, wenn nicht die Möglichkeit 1 (aus den 5 verschiedenen Möglichkeiten) vereinbart wird.

Es gibt 5 Möglichkeiten eine Parität festzulegen:

1. keine Parität: Es wird gar kein Paritätsbit in den Datenrahmen eingefügt (Parameterwert: 0).
2. ungerade Parität („odd“): Das Paritätsbit wird in den Datenrahmen eingefügt und zwar so, dass in den Datenwortbit und dem Paritätsbit zusammen eine **ungerade Anzahl von Einsen** vorkommt (Parameterwert: 1)
3. gerade Parität („even“): Das Paritätsbit wird in den Datenrahmen eingefügt und zwar so, dass in den Datenwortbit und dem Paritätsbit zusammen eine **gerade Anzahl von Einsen** vorkommt (Parameterwert: 2).
4. Mark: Das Paritätsbit wird immer mit Wert 1 in den Datenrahmen eingefügt (Parameterwert: 3)
5. Space: Das Paritätsbit wird immer mit Wert 0 in den Datenrahmen eingefügt (Parameterwert: 4)  
*Mark und Space eignen sich eigentlich nur dazu, Fehler im Paritätsbit selbst zu erkennen.*

*Der angegebene Parameterwert ist für die Programmierung wichtig, weil einer dieser Werte bei der Initialisierung einer seriellen Schnittstelle angegeben werden muss.*

Zum Abschluss besteht ein Datenrahmen aus **einem oder zwei Stoppbit (die sind immer logisch 1)**, welche das Ende einer Dateneinheit kennzeichnen. Die Anzahl Stoppbit (Eins oder Zwei) ist genau wie die Parität einstellbar und sollte beim Sender und Empfänger logischerweise wieder gleich eingestellt sein.

Zusammenfassung:

Die folgenden Einstellungen müssen bei der seriellen Übertragung zwischen einem Sender und Empfänger sowohl auf Sender- als auch auf Empfängerseite vereinbart werden:

- COM-Portnummer
- Anzahl der Datenbits (Datenwort)
- Baudrate
- Anzahl Stoppbit
- Parität

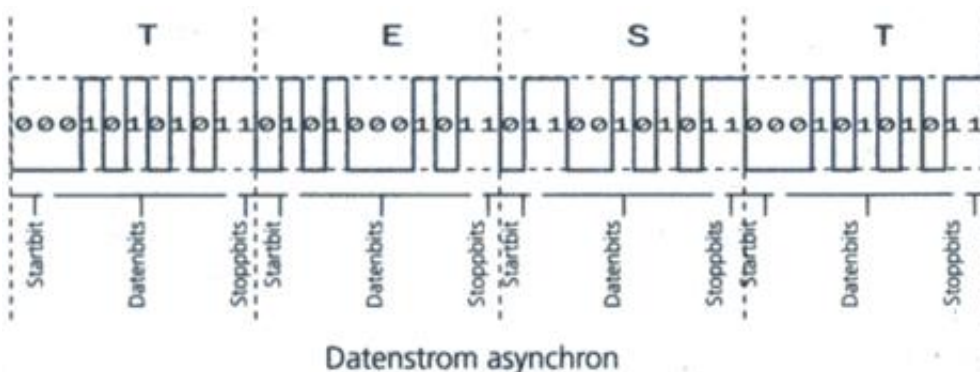
Hier ist beispielhaft die Darstellung von 4 Datenrahmen für die Übertragung des Textes „TEST“ abgebildet. Die Einstellungen zwischen Sender und Empfänger sind:

- Anzahl der Datenbits: 8
- Anzahl Stoppbit: 2
- Parität: 0 (also kein Paritätsbit)

← Übertragungsrichtung

$b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7$  sind die übertragenen Datenbits für S

1 1 0 0 1 0 1 0

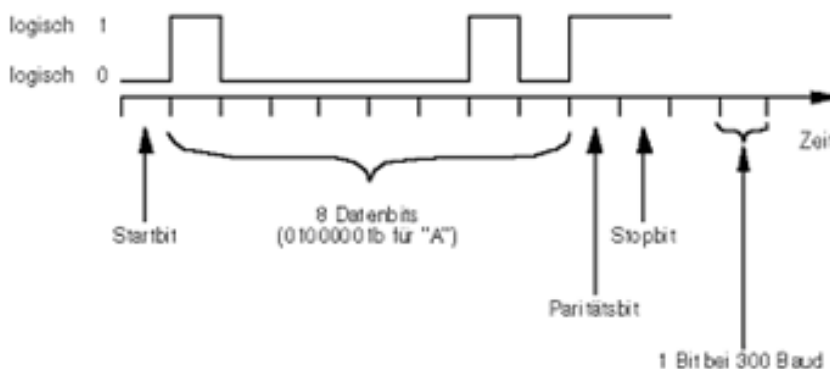


Datenbits für S, in der Reihenfolge von  $b_7$  bis  $b_0$ : 01010011 ergibt die ASCII-Codierung für S (83<sub>dez</sub>)

Ein weiteres Beispiel für die Übertragung des Buchstaben „A“. Die Einstellungen zwischen Sender und Empfänger sind:

- Anzahl der Datenbits: 8
- Anzahl Stoppbit: 1
- ungerade Parität

← Übertragungsrichtung



### Baudrate

Die **Baudrate** gibt die Übertragung einer **Anzahl Bit pro Sekunde** an. Da auch die Bit der Steuerinformationen mit übertragen werden müssen, ist die Anzahl der übertragenen Zeichen pro Sekunde nicht der Quotient aus Baudrate und Datenbit, da man immer einen gesamten Datenrahmen für die Übertragung eines Zeichens einbeziehen muss.

**Berechnen Sie:** Wie viele Zeichen werden bei den folgenden Einstellungen zwischen Sender und Empfänger in 10 Minuten übertragen?

#### System 1

- COM1
- 8 Bit
- 19200
- 2
- gerade Parität

*Ergebnis (Kontrolle) [960000 Z./min]*

#### System 2

- COM1
- 8 Bit
- 19200
- 1
- keine Parität

*Ergebnis (Kontrolle) [1152000 Z./min]*

### Standardprotokoll RS-232

RS-232 (Recommended Standard 232) ist ein **Standard für eine serielle Schnittstelle**, der **1962** vom US-amerikanischen Standardisierungsgremium *Electronic Industries Association* (EIA) erarbeitet wurde und bis in die 2010er Jahre häufig bei Computern vorhanden war. Statt der Bezeichnung RS-232 findet man deshalb auch die Bezeichnung EIA-232. Beide Bezeichnungen beziehen sich auf die gleiche Beschreibung.

RS-232 ist ein Standardkommunikationsprotokoll, das von seriellen Schnittstellen zum Verbinden eines Computers und seines Peripheriegeräts verwendet wird. Diese Norm beschreibt den Prozess des Datenaustauschs zwischen einem Telekommunikationsgerät wie einem Modem und einem Computerterminal. Der RS-232-Standard definiert die elektrischen Eigenschaften von Signalen, ihren Zweck, ihre Dauer sowie die Größe der Steckverbinder und ihre Pinbelegung und den Datenrahmen (Dataframe) mit ihren möglichen Konfigurationen (Baudrate, Stoppbit, Parität, etc.)

Es sollte auch beachtet werden, dass der **RS-232-Standard das Protokoll auf der physischen Ebene** ist und **nicht die Transportprotokolle** definiert, die für die Datenübertragung verwendet werden. Die Transportprotokolle können je nach verwendeter Kommunikationsausrüstung und Software variieren.

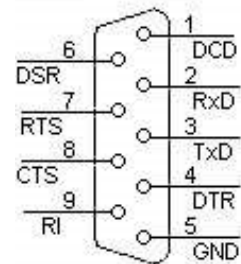
Die Transportprotokolle müssen je nach Projektanforderungen selbst definiert und implementiert werden. Sehr wichtig ist vor allem der kontrollierte Kommunikationsstart der beteiligten Geräte. Hier muss sichergestellt werden, dass alle Geräte bereit sind und dass klar ist, welches Geräte mit der Übertragung der gewünschten Nutzdaten startet und welches Gerät Nutzdaten einliest. Der Beginn und das Ende der Nutzdatenübertragung muss dabei ebenso sichergestellt sein, wie das korrekte Empfangen der Nutzdaten. Ggf. sind hier mehrfache „Hin- und Her-Versendungen“ (Stichworte: Software-Handshake, Hardware-Handshake oder auch eine Kombination von Beidem). Das Ergebnis dieser Vorarbeit ist ein Protokoll (Transportprotokoll).

Der Entwurf / die Modellierung des Protokolls kann / sollte mit einem UML-Sequenzdiagramm ausgeführt werden.

### RS-232 - der technische Anschluss

Als Anschluss für die serielle Schnittstelle, auch als V24- und RS 232C-Schnittstelle bezeichnet, sind i.d.R. **9-polige Sub-D-Steckverbindungen** gebräuchlich.

Leider sind Stand 02/2018 in den meisten IT-Systemen keine RS 232C-Schnittstellen (karten) verbaut. Eine Nachbildung kann über sog. USB-Serial-Adapter (z.B. vom Hersteller Digitus) und einer zugehörigen Treibersoftware 1:1 nachgebildet werden.



RS232

In der nachfolgenden **Tabelle sind die Leitungen nach RS-232** und deren Bedeutung aus Sicht des PCs beschrieben:

*(fett gedruckt sind die Signalleitungen, die über die Java-Klasse `Serial` angesprochen werden können)*

PIN	Leitung	Kürzel	Bedeutung
1	Data Carrier Detect	DCD	Eingang: Mit einem High-Pegel an diesem Eingang signalisiert die Gegenstelle, dass sie einlaufende Daten auf der Leitung erkennt.
2	<b>Receive Data</b>	<b>RxD (RD)</b>	<b>Eingang: Datenempfangsleitung</b>
3	<b>Transmit Data</b>	<b>TxD (TD)</b>	<b>Ausgang: Datensendeleitung</b>
4	<b>Data Terminal Ready</b>	<b>DTR</b>	<b>Ausgang: PC signalisiert der Gegenstelle Betriebsbereitschaft des Gerätes</b>
5	Signal Ground	GND	Massepotential 0 V
6	<b>Data Set Ready</b>	<b>DSR</b>	<b>Eingang: PC empfängt die Betriebsbereitschaft des verbundenen Gerätes (Gegenstelle)</b>
7	<b>Request To Send</b>	<b>RTS</b>	<b>Ausgang: PC signalisiert der Gegenstelle, dass Daten gesendet werden sollen / können.</b>
8	<b>Clear To Send</b>	<b>CTS</b>	<b>Eingang: PC empfängt von der Gegenstelle, dass er selbst Daten senden soll / kann.</b>
9	Ring Indicator	RI	Eingang: Ein High-Pegel an diesem Eingang signalisiert dem Empfangs-Gerät, dass ein Anruf ankommt, d.h. dass jemand eine Datenverbindung aufzubauen wünscht.

Tabelle 1: Leitungen und deren Bedeutung aus Sicht des PCs

### Das universelle Nullmodemkabel<sup>1</sup>

Als **Nullmodem-Kabel** wird ein Kabel bezeichnet, das für Datenaustausch zwei Computer über die serielle Schnittstelle (RS-232 bzw. EIA-232) miteinander verbindet.



Der Name **Null-Modem** (engl. für **kein Modem** im Sinne von „Modem entfallen“) rührt daher, dass Computer (Datenendeinrichtungen, DEE) auf lange Distanz oft über Modems miteinander verbunden werden (bzw. wurden), die traditionellerweise jeweils über eine serielle Schnittstelle mit dem Computer verbunden sind. Bei einer direkten Verbindung, bei der die Computer nur

<sup>1</sup> Text- und Bildquelle: <http://de.wikipedia.org/wiki/Nullmodem-Kabel>

wenige Meter voneinander entfernt stehen, können die Modems entfallen und durch ein Nullmodemkabel ersetzt werden. Ein Nullmodem-Kabel hat beidseitig weibliche (female) D-Sub-Steckverbinder.

Im **Unterschied zu einem seriellen Standardkabel**, das einen Computer mit einem Modem verbindet, müssen hier **einige Leitungen gekreuzt** werden.

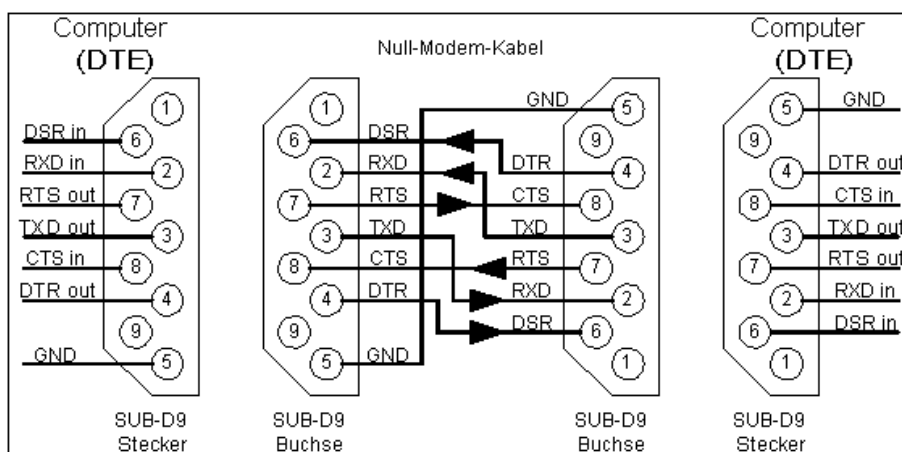
Im Bereich der Ethernet-Netzwerktechnik nennt man das Pendant **Crosskabel**. Für die Anwendung über USB-Kabel nennt man das Pendant **USB-Link-Kabel** (synonym auch zu finden: USB-Datenkabel).

Leider gibt es **viele Varianten von Nullmodemkabeln**. Es sind jedoch IMMER mindestens 3 Leitungen verbunden:

- Auf Pin 5 jeweils die gemeinsame Masseleitung (1:1 durchverbunden).
- Auf Pin 3 (TxD) zum Pin 2 (Rx) die gekreuzte Verbindung Tx auf Rx.
- Auf Pin 2 (Rx) zum Pin 3 (Tx) die gekreuzte Verbindung Rx auf Tx.

### Aufbau eines Standard-Nullmodem-Kabels (Full-Null-Cabel):

Im Bild ist ein vollbeschaltetes Nullmodemkabel (nach RS-232) abgebildet, mit dem auch Hardware-Handshakes (Hardwaresignalisierungen) über die Leitungspaare DTR / DSR und RTS / CTS durchgeführt werden können.



**ACHTUNG:** Man sieht es seriellen Schnittstellenkabel NICHT an, welche Verdrahtung enthalten ist. Im Zweifelsfall ist das auszumessen oder über eine Referenzprogramm zwischen 2 Rechnern auszuprobieren.

Getestete Nullmodemkabel sollten mit Permanentmarker mit einen großen X markiert werden.

Eine andere Darstellung als Tabelle (nur die für DK relevanten PIN / Leitungen):

