

RS-232 - DIE SERIELLE SCHNITTSTELLE

Inhaltsverzeichnis

1. SERIELLE SCHNITTSTELLEN

1.1. ALLGEMEINES

Serielle Schnittstellen dienen zum physischen Austausch von Daten zwischen Computer und Peripheriegeräten. Die Datenbits werden hierbei nacheinander (also seriell) übertragen.

Die RS-232-Schnittstelle (oft auch COM-Schnittstelle oder V.24 genannt) wurde in den 1960er Jahren hauptsächlich für Einsatzbereiche in der Telekommunikation (z.B. Fernschreiber, später Modems) und EDV (zur Anbindung von Terminals an Mainframes) entwickelt. Die letzte Aktualisierung der Spezifikationen der RS-232-Schnittstelle erfolgte 1997.

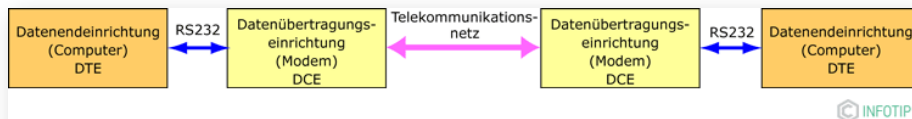


Abb. 1: Prinzip der Kommunikation über Modems

Mit der zunehmenden Digitalisierung von elektronischen Geräten stieg auch der Bedarf diese rechnergestützt auszulesen und zu verwalten. Aufgrund der einfachen Implementierbarkeit und Zuverlässigkeit fand die RS-232 schnell Anwendung in der allgemeinen PC-Technik und sogar in der Unterhaltungselektronik. Die RS-232 eignet sich zum Anschluss von Kassenterminals, Druckern und Messgeräten, steuert Plasmabildschirme und man kann Firmware auf DVD-Player und Satellitenreceiver übertragen. Bis zum Erscheinen des USB (Universal Serial Bus) war praktisch jeder PC mit mindestens einer RS-232-Schnittstelle ausgestattet.

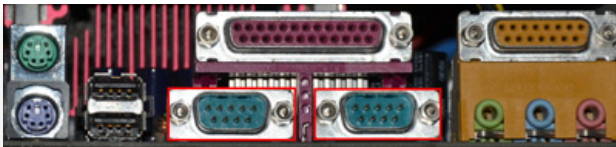


Abb. 2: RS-232-Schnittstellen auf einem PC-Mainboard (von 2005)

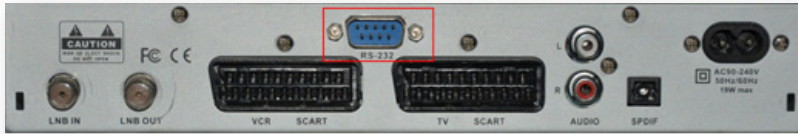


Abb. 3: RS-232-Schnittstelle an einem Satellitenreceiver

Die Bezeichnung "RS" ist ursprünglich auf die *Radio Section* der EIA (Electronic Industries Alliance: eine US-amerikanische Standard- und Handelsorganisation (bis 2011)) zurückzuführen, wird heutzutage allerdings als *Recommended Standard* (= Empfohlener Standard) interpretiert.

2. ÜBERTRAGUNGSVERFAHREN

2.1. PRINZIP DER DATENÜBERTRAGUNG

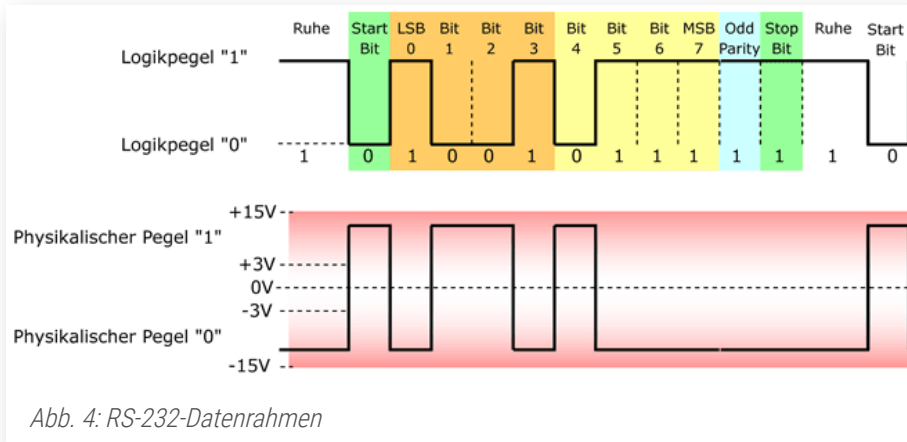
Die Verwendung des Prinzips der seriellen Datenübertragung bedeutet, dass die Bits der zu übertragenden Bytes nacheinander über die Verbindungsleitung übertragen werden. Bei der RS-232-Schnittstelle erfolgt dieses wortweise. Jedes Datenwort besteht, abhängig von der eingestellten Konfiguration, aus fünf ("Fernschreibcode") bis neun Bits. Üblich sind allerdings sieben oder acht Datenbits (Nutzdaten). Die Datenbits sind zu einzelnen Zeichen, meistens hexadezimal nach ASCII kodiert.

Bei der RS-232-Schnittstelle erfolgt die Datenübertragung asynchron, d.h. es wird kein separates Taktsignal, das die einzelnen Bits markiert und den Empfänger mit dem Sender synchronisiert, mit übertragen. Der Empfänger muss also aus dem Datenfluss das Taktsignal zurückgewinnen. Pegeländerungen (= Flanken) werden durch Abtasten ("sampling") des empfangenen Signals mit einem Vielfachen der Datenrate erkannt. Die Synchronisation des Empfängers wird dann mit markierenden Pegeln im Datenstrom, den sogenannten Start- und Stopp-Bits, vorgenommen.

Vor dem Start der Übertragung auf der Datenleitung führt diese im Ruhezustand den Pegel der logischen "1". Mit dem Beginn der Übertragung wird nächst eine logische "0" gesendet, was den (invertierten) Pegel des Start-Bits darstellt. Es folgen dann die Datenbits der Nutzdaten. Abgeschlossen wird die Übertragung des Datenwortes mit einem Stopp-Bit, das durch eine logische "1" markiert ist.

Die Dauer der Übertragungszeit eines einzelnen Bits ist abhängig von der eingestellten Bitrate.

Bei der Übertragung synchronisiert sich der Empfänger auf die Mitte der einzelnen Datenbits und tastet die folgenden Bits des Datenwortes mit seiner eigenen Bitrate ab. Damit das funktioniert, dürfen die Bitraten von Sender und Empfänger nur einige Prozent voneinander abweichen. Übliche Bitraten und deren Bitdauer zeigt Tabelle 1.



Bitrate (Bit/s)	Bitdauer
50	20,0 ms
110	9,09 ms
150	6,67 ms
300	3,33 ms
1.200	833 µs
2.400	417 µs
4.800	208 µs
9.600	104 µs
19.200	52,1 µs
38.400	26,0 µs
57.600	17,4 µs
115.200	8,68 µs
230.400	4,34 µs
460.800	2,17 µs
500.000	2,00 µs

Tabelle 1: Übliche RS-232-Übertragungsraten

Der RS-232-Standard spezifiziert nicht wie die eigentlichen Daten übertragen werden. Kommunikationsparameter wie Datenrate, Zeichenkodierung, Datenrahmen, Datenkompression, Fehlererkennung und Übertragungsprotokoll müssen vor der Datenübertragung eingestellt bzw. von Sender- und Empfangsgerät ausgehandelt werden.

Die serielle Schnittstelle eines PCs verwendet beispielsweise sieben oder acht Bit lange ASCII-kodierte Worte als Nutzdaten. Dem folgt ein Paritätsbit (Parity Bit) zur Fehlererkennung. Das Paritätsbit kennzeichnet entweder eine gerade (even) Parität (Quersumme) oder eine ungerade Parität (odd parity). Wahlweise kann auf die Übertragung eines Paritätsbit auch verzichtet werden (no parity) oder ständig gesetzt (mark

parity) oder gelöscht (space parity) sein. Eingerahmt sind Datenwort und Paritätsbit vom Start- und einem oder zwei Stopp-Bits. Die Ruhezeit darf beliebig lang sein.

Die über eine RS-232-Schnittstelle erreichbare maximale Datenrate hängt von mehreren Faktoren ab, wobei die Konfiguration natürlich auch eine Rolle spielt. Im wesentlichen bestimmen aber die Schnelligkeit des Schnittstellenbausteins

und die Länge und Qualität des Übertragungskabel wie schnell und sicher die Datenübertragung stattfindet.

3. DIE RS-232-SCHNITTSTELLE

3.1. STANDARDSCHNITTSTELLEN UND SIGNALE



Abb. 5: 25-poliger
Geräteanschluss

Die ursprüngliche RS-232-Spezifikation definierte die Verbindung zwischen einem Terminal als Datenendeinrichtung (DEE (engl. Data Terminal Equipment (DTE)) und einem Modem (Datenübertragungseinrichtung, DÜE (engl. Data Communication Equipment, DCE).

Als DTEs werden heute Computer, Drucker, Plotter usw. betrachtet. Diesen Geräten ist gemeinsam, dass sie Daten senden oder empfangen können, diese aber nicht weiterleiten. Der Schnittstellenanschluss eines DTEs ist immer als Stiftleiste ("male") ausgelegt.



Abb. 6: 9-poliger
Geräteanschluss

DCEs sind Geräte, die Daten empfangen und sie ohne Verarbeitung an andere Geräte weiterleiten. Der Schnittstellenanschluss eines DCE ist immer als Buchse ("female") ausgelegt.

Die "klassische" RS-232-Schnittstelle ist die 25-polige Sub-D-Verbinder (auch als DB25 oder DE25 bezeichnet, Abb. 5). Da in der EDV aber nicht alle Signale, die der Standard vorsieht, benötigt werden, hat sich der Anschluss der Geräte über 9-polige Sub-D-Verbinder durchgesetzt.

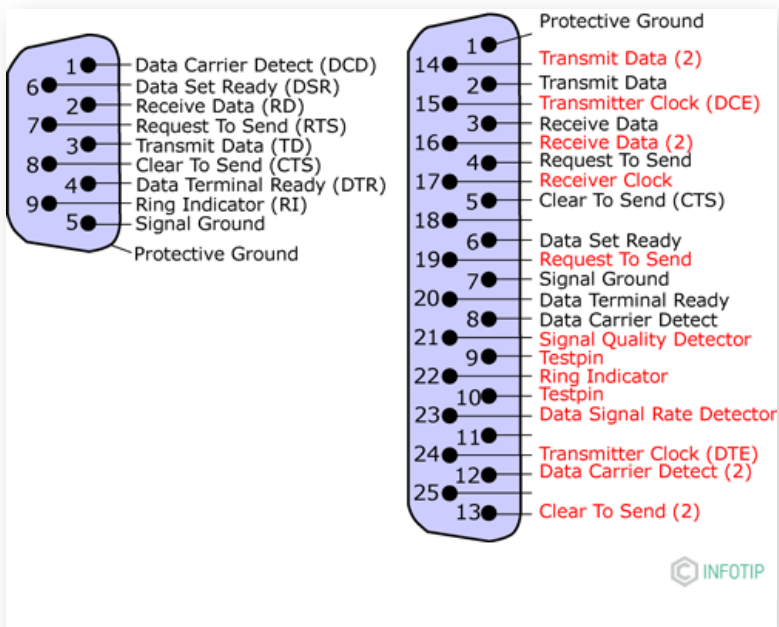


Abb. 7: Pinbelegung der Geräteanschlussstecker ("Male")

Der Hauptunterschied zwischen dem 25-poligen und dem 9-poligen Verbinder ist, dass der 25-polige zwei Datenkanäle zur Verfügung stellt und der 9-polige nur einen.

Signal	Abk.	Verbinder		Beschreibung
		25-pol.	9-pol.	
Transmitted Data	TxD	Pin 2	Pin 3	Sendedaten (Datenübertragung vom Computer zum Peripheriegerät)
Received Data	RxD	Pin 3	Pin 2	Empfangsdaten (Datenübertragung von Peripheriegerät zum Computer)
Common Ground	GND	Pin 7	Pin 5	Gemeinsame Masseleitung für alle Signale
Request To Send	RTS	Pin 4	Pin 7	Request to Send - der Computer signalisiert, dass er Daten an das Peripheriegerät senden möchte.
Clear To Send	CTS	Pin 5	Pin 8	Clear to Send - das Peripheriegerät signalisiert, dass es Daten entgegennehmen kann.
Data Set Ready	DSR	Pin 6	Pin 6	Data Set Ready - das Peripheriegerät signalisiert, dass es bereit (eingeschaltet) ist.
Data Carrier Detect	DCD	Pin 8	Pin 1	Data Carrier Detect - das Peripheriegerät (Modem) signalisiert, dass eine Verbindung über Telefonleitung hergestellt ist
Data Terminal Ready	DTR	Pin 20	Pin 4	Data Terminal Ready - der Computer signalisiert, dass er verfügbar (eingeschaltet) ist.
Ring Indicator	RI	Pin 22	Pin 9	Ring Indicator - das Peripheriegerät (Modem) zeigt einen ankommenden Telefonanruf an.

Tabelle 2: Pinbelegung und Signale von RS-232-Schnittstellen

3.2. FLUSSSTEUERUNG (HANDSHAKE DTE-DCE)

Von den neun Verbindungsleitungen, die die ursprüngliche Spezifizierung bietet, werden in modernen Anwendungen maximal fünf Leitungen verwendet. Verzichtet man auf eine hardwaremäßige Flusssteuerung der Daten ("Handshaking" = Handschütteln) reichen sogar nur drei Leitungen, nämlich TxD, RxD und GND, aus. Hiermit ist eine bidirektionale Kommunikation möglich, die auch einen Vollduplexbetrieb erlaubt.

Die Flusssteuerung soll Datenverluste vermeiden wenn der Empfänger nicht schnell genug ist um die ihm gesendeten Daten rechtzeitig zu verarbeiten. Der Datenverlust entsteht dann, wenn der Eingangspufferspeicher des Empfängers überläuft.

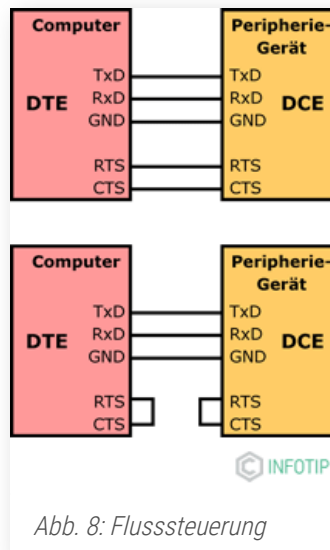


Abb. 8: Flusssteuerung

Man unterscheidet zwei Arten von Flusssteuerung:

Beim Hardwarehandshake informieren sich die Geräte gegenseitig über die Steuerleitungen RTS und CTS über ihren momentanen Status. (Abb. 8 oben)

Beim Softwarehandshake schickt der Empfänger zur Signalisierung dem Sender über die Datenleitungen TxD und RxD spezielle Steuerzeichen, die den Status des Eingangspufferspeicher aufzeigen. (Abb. 8 unten)

3.3. SIGNALPEGEL

Die physikalischen Pegel, die den logischen Pegeln entsprechen, werden vom RS-232-Standard definiert. Gültige Signalpegel liegen in den Bereichen von +3V und +15V und -3V und -15V. Pegel zwischen -3V und +3V sind nicht zulässig. Der Standard sieht einen maximalen Pegelhub von ± 25 V vor. Üblich sind ± 5 V, ± 10 V, ± 12 V und ± 15 V. Die tatsächlich anliegenden Pegel sind abhängig von der Treiberschaltung.

Für die Datenleitungen (TxD und RxD) ist eine logische "0" als negativer Pegel und eine logische "1" als positiver Pegel definiert (siehe Abb. 4). Steuerleitungen (RTS, CTS) haben umgekehrte Pegel: Ein aktives Signal hat einen positiven Pegel, ein inaktives Signal einen negativen Pegel.

Da die RS-232-Signalpegel höher sind als die logischen Pegel von Logikbausteinen (z.B. TTL 5V) können RS-232-Daten nicht direkt verarbeitet werden. Hier müssen zur Anpassung spezielle Pegelwandlerbausteine (siehe Abb. 14) zwischengeschaltet werden, die auch gleichzeitig die Logikbausteine vor Überspannung schützen.

Die RS-232-Signalspannungen sind asymmetrisch und sind auf Masse = 0V bezogen. Beim Verbinden mit anderen Geräten deren Masse nicht 0V ist können daher Probleme durch Erdschleifen auftreten, die die Datenübertragung stören könnten. Diese Erdschleifen werden meist durch die mit Schutz Erde (Schuko, Schutzklasse I) versehenen EDV-Geräte zusammen mit den Außenleitern der abgeschirmten Kabel zwischen den Geräten verursacht.

3.4. KABELLÄNGEN UND DATENÜBERTRAGUNGSRATEN

Kabellänge und die mögliche Datenübertragungsrate stehen in einem direktem Zusammenhang. Mit steigender Kabellänge steigt auch dessen ohmscher Widerstand und die Kapazität zwischen den einzelnen Leitern. Dieses führt zu einer Dämpfung der Signalamplitude und die Signalfanken werden verschliffen. Da der Standard auch keine Abschlusswiderstände an den Leitungsenden vorsieht, können mit steigender Kabellänge und Datenrate Reflexionen wirksam werden.

Der RS-232-Standard definiert keine maximale Länge des Verbindungskabels, aber dessen maximale Kapazität (2500pF). Diese wird mit Standardkabel bei ca. 15 m erreicht. Mit speziellen kapazitätsarmen Kabeln können auch noch bei Kabellängen von 300 m hohe Datenraten erreicht werden.

Baudrate	Kabellänge
2.400	900 m
4.800	300 m
9.600	152 m
19.200	15 m
57.600	5 m
115.200	<2 m

Tabelle 3: Datenraten abhängig von der Kabellänge

4. VERBINDUNGSKABEL

Damit zwei über ein RS-232-Kabel miteinander verbundene Geräte kommunizieren können, müssen die sendenden Anschlüsse des einen Gerätes mit den empfangenden Anschlüssen des anderen und umgekehrt verbunden sein.

4.1. VERBINDUNG ZWISCHEN DATENENDGERÄT UND PERIPHERIEGERÄT (DTE-DCE)

Die Verbindung zwischen einem Datenendgerät (Computer) und einem Peripheriegerät ist die Standardsituation für die RS-232 geschaffen wurde: Der Computer sendet die Daten und das Peripheriegerät empfängt sie. Über die Handshakeleitungen wird der Datenfluss gesteuert. Die Belegung der Anschlüsse von Computer und Gerät sind so ausgelegt, dass ein 1:1 Verbindungskabel eingesetzt wird. Bei diesem Verbindungskabel sind jeweils gleiche Pins miteinander verbunden.

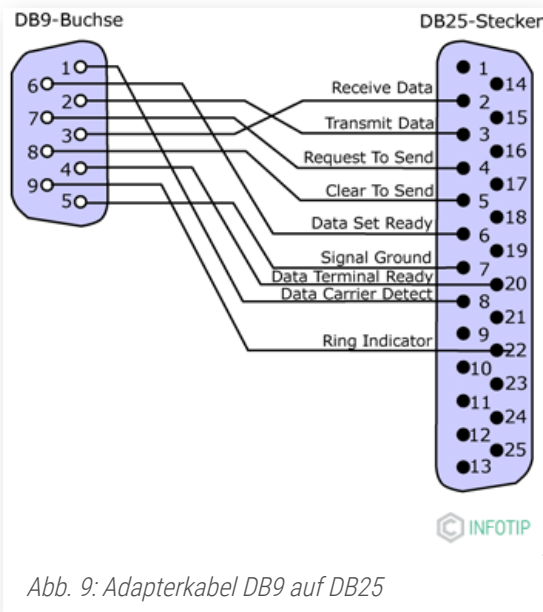


Abb. 9: Adapterkabel DB9 auf DB25

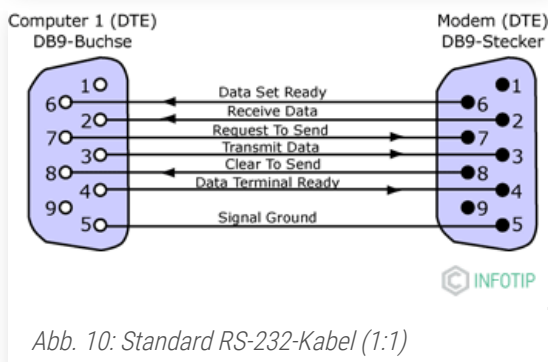


Abb. 10: Standard RS-232-Kabel (1:1)

4.2. VERBINDUNG ZWISCHEN ZWEI GLEICHEN GERÄTEN (DTE-DTE)

Die RS-232 wurde geschaffen um Geräte wie Fernschreiber usw. über Modems miteinander kommunizieren zu lassen. Sollen aber die Endgeräte direkt miteinander verbunden werden, scheitert dies zuerst einmal daran, dass beide Geräteanschlüsse Stecker sind und ein Standard-RS-232-Kabel einen Stecker- und einen Buchsenanschluss aufweist. Zudem würden bei einer Verbindung der beiden Geräte mit einem 1:1-Kabel die Sendesignale des einen Gerätes auf die Sendeleitungen des anderen geschaltet und die Empfangsleitungen auf die Empfangsleitungen des anderen. Eine Kommunikation ist also nicht möglich. Um eine Kommunikation zu ermöglichen müssen daher die Leitungen gekreuzt werden damit Senden auf Empfangen und Empfangen auf Senden geschaltet werden. Eine solcher Verbinder wird als Nullmodem bezeichnet, da er eine Verbindung zwischen zwei Endgeräten ohne Modems (also null Modems) erlaubt. In der Anfangszeit der PC-Ära, als die Netzwerktechnik noch nicht allgemein verfügbar war, waren Nullmodem-Kabel weit verbreitet um zwischen Computern Daten auszutauschen.



Abb. 11: Nullmodem-Adapter [1]

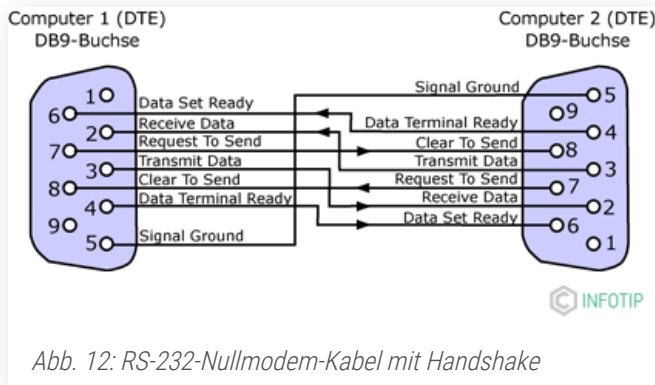


Abb. 12: RS-232-Nullmodem-Kabel mit Handshake

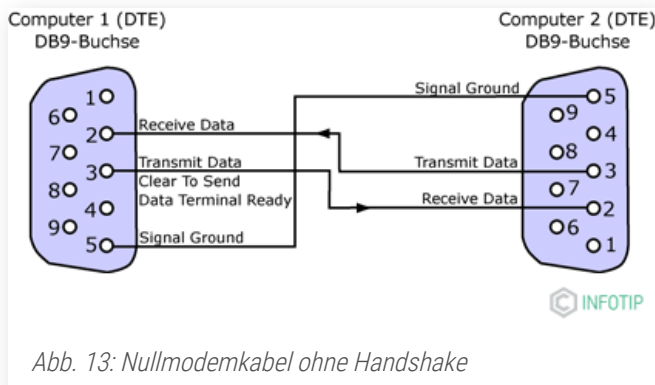
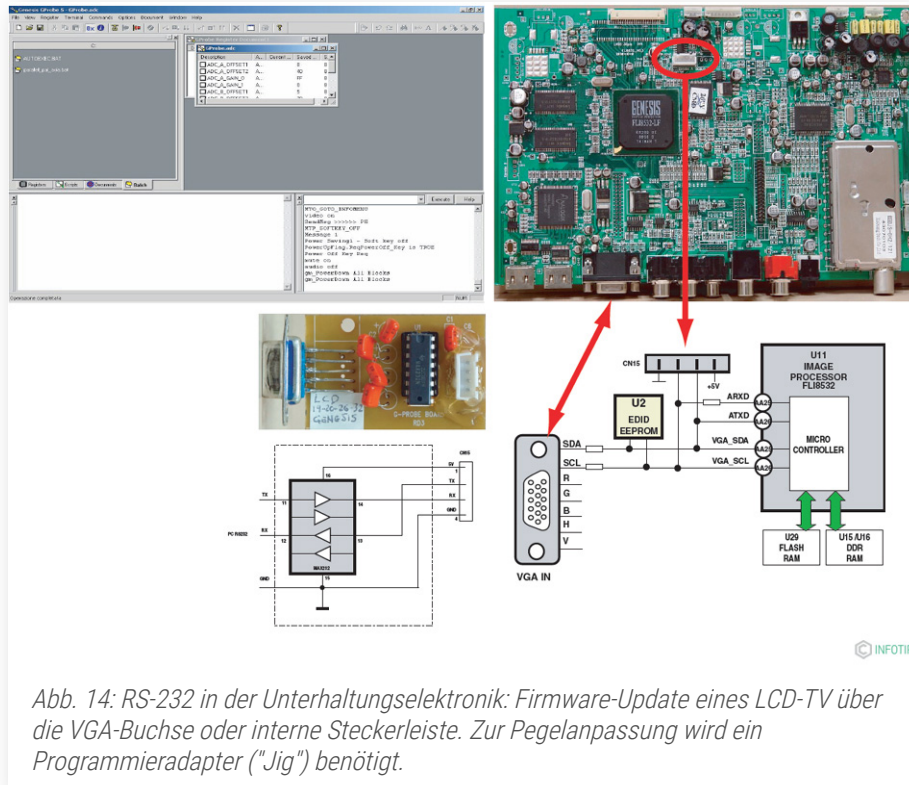


Abb. 13: Nullmodemkabel ohne Handshake

5. DIE RS-232-SCHNITTSTELLE HEUTE

In den meisten Anwendungen verliert die RS-232-Schnittstelle immer mehr an Bedeutung. Ethernet, USB und IEEE 1394 (Firewire) sind schneller, erlauben größere Reichweiten und sind, da die Signalübertragung symmetrisch erfolgt, weit aus weniger störanfällig. Seit etwa 2005 werden praktisch keine PCs mehr serienmäßig mit eingebauter RS-232 ausgestattet. Auch in der Unterhaltungselektronik, wo die RS-232-Schnittstelle häufig zum Durchführen von Firmware-Updates verwendet wurde (siehe Abb. 14), wird diese überflüssig, weil immer mehr Geräte mit USB- und Netzwerkanschluss ausgestattet sind.



In einigen Bereichen, so in der Laborautomation, der Maschinensteuerung und -programmierung und vielen EDV-Anwendungen (z.B. Verwaltung von Servern oder Switches über Konsole) wird die RS-232-Schnittstelle aufgrund ihrer Einfachheit und leichten Implantierbarkeit noch lange in Verwendung bleiben.

REFERENZEN

Abbildungen

[1] Foto "Abb. 11: Nullmodem-Adapter" Lizenz: Public domain via Wikimedia Commons, Quelle: "Null_modem.jpg" by Dori (Own work): http://commons.wikimedia.org/wiki/File%3ANull_modem.jpg

Rechtshinweis

Sofern auf dieser Seite markenrechtlich geschützte Begriffe, geschützte (Wort- und/oder Bild-) Marken oder geschützte Produktnamen genannt werden, weisen wir ausdrücklich darauf hin, dass die Nennung dieser Marken, Namen und Begriffe hier ausschließlich der redaktionellen