

## Zusammenfassung des Vortrags

Hinführung	Die UART-Schnittstelle (Universal Asynchronous Receiver and Transmitter) bietet eine serielle Datenübertragung in asynchronen Schaltkreisen. Asynchrone Schaltkreise haben das Potential schneller und effizienter zu arbeiten als synchrone, weil erstere aus selbständigen Einheiten bestehen, die sich nur zu Kommunikationszwecken synchronisieren. Wenn die Kommunikation ruht, können einzelne Teilnehmer sich in den Sleep-mode versetzen und bei Bedarf durch einen sog. Handshake wieder geweckt werden. Die Kommunikationsschnittstelle ermöglicht eine dezentrale, effiziente und vielseitige Entwicklung heterogener Endgeräte.
Hardware (vgl. Abb.2)	<ul style="list-style-type: none"> <li>• UART-spezifischen IC: MAX232(Pegelwandler) und RS232(Handshake)</li> <li>• Kondensatoren zur Entkopplung der Versorgungsspannung</li> <li>• SUB D9 Verbinder für 9-poliges Verlängerungskabel (Modem-Kabel)</li> </ul>
Software	Für das UART-Protokoll sind von Softwareseite einige wichtige Dinge zu leisten. Im Wesentlichen müssen die Frequenz, mit der die Datenwörter übertragen werden, in die spezifischen Register (UBRR) eingeschrieben werden und Dinge wie Start/Stopp-bits abgesprochen werden. Die Baudrate ist der wichtigste Parameter, der bei der Konfiguration festzulegen ist. Er muss sich ganzzahlig aus einem Systemtakt gewinnen lassen, damit beim Datentransfer die Abtastung auf Empfängerseite keinen "missing Code" generiert.
Handshake	Der Handshake bietet die Möglichkeit, entweder Hardware basiert oder Software basiert, eine Kommunikation zwischen logischen Einheiten zu koordinieren. Basiert der Handshake auf Hardware, wie das heute noch bei herkömmlichen Telefonleitungen realisiert wird, benötigt man zusätzlich zum DTE das DCE (vgl. Tab. 2), welche die Sende- und Empfangsbereitschaft absprechen. In moderner Architektur benutzt man gekreuzte Datenleitungen, die komplementäre Ports miteinander verknüpfen (vgl. Abb. 1). Man benötigt im Bussystem somit zwei zusätzliche Leitungen: die CTS- und die RTS-Leitung. Den praktischen Aufbau mit der benötigten Hardware für die UART-Schnittstelle lässt sich an Abb. 2 nachvollziehen. Software basierte Handshakes findet man zum Beispiel in Routern. Hier werden die CTS/RTS-Leitung mit einem Datenwort (XON/XOFF) ersetzt.

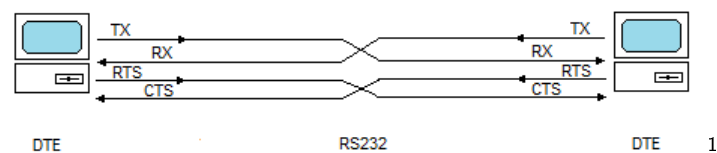
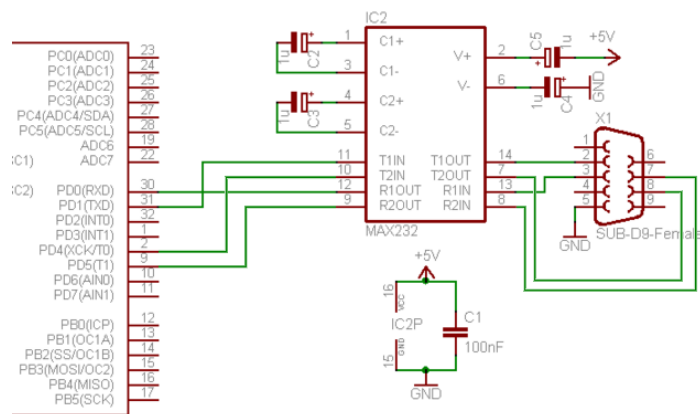


Abbildung 1: moderner Hardwarehandshake

<sup>1</sup>Quelle: mikrocontroller.net/articles/AVR-Tutorial:\_UART



2

Abbildung 2: UART-Aufbau mit Handshake

## Quellen

DCE	Data Carrier Equipment
DTE	Data Terminal Equipment
CTS	Clear to Send
RTS	Request to Send
RXD	receive Data
TXD	transmit Data
UART	Universal Asynchronous Receiver and Transmitter
SPI	serial peripheral interface
ISP	in system programming
SCK	serial clock

Tabelle 2: Abkürzungsverzeichnis

- Inhaltsquellen:

1. mikrocontroller.net/articles/AVR-Tutorial:\_UART (abgerufen 13.5.14)
2. mikrocontroller.net/articles/Baudratenquarz (abgerufen 13.5.14)
3. wikipedia.org/wiki/Émile\_Baudot (abgerufen 8.5.14)

- Bildquellen:

1. Emile-Baudot: wikipedia.org/wiki/Émile\_Baudot (abgerufen 8.5.14)
2. mikrocontroller.net/articles/AVR-Tutorial:\_UART (abgerufen 8.5.14)
3. ASCII-Tabelle: asciitable.com (abgerufen 8.5.14)
4. Handshake: wikipedia.org/wiki/Handshake (abgerufen 11.5.14)

<sup>2</sup>Quelle: mikrocontroller.net/articles/AVR-Tutorial:\_UART