

PROTOKOLL

zur Laborübung

RS232 - COM

HTL
St. Pölten

EL

Gruppe / Klasse 5 / 3BHEL	Protokollführer HOFSTÄTTER A.	Unterschrift
Übungs- / Abgabedatum 5. Mai 2014 5. Mai 2014	Mitarbeiter	Unterschrift
Lehrer [Lehrer]	Mitarbeiter	Unterschrift
Note	Mitarbeiter	Unterschrift

RS232 - COM

Verwendete Geräte

Nr.	Gerät	Hersteller	Typ	Platznummer
1.	Oszilloskope			-

Verwendete Programme

Nr.	Name	Version
1.	Putty	0.63

1 Inhaltsverzeichnis

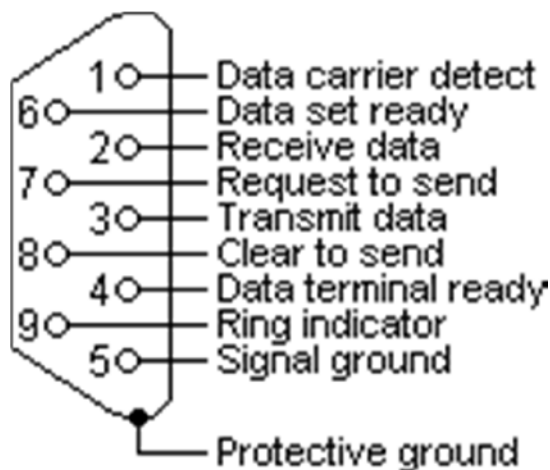
<u>1</u>	<u>INHALTSVERZEICHNIS</u>	<u>2</u>
<u>2</u>	<u>AUFGABENSTELLUNG</u>	<u>3</u>
<u>3</u>	<u>ANSCHLÜSSE</u>	<u>3</u>
<u>4</u>	<u>DATEN ÜBER PUTTY SENDEN</u>	<u>3</u>
<u>5</u>	<u>DEKODIEREN DER NUTZDATEN UND SIGNALFORMATIONEN AUS DER EXE "COMPORT2.EXE"</u>	<u>6</u>
<u>6</u>	<u>WEITERES WURDEN ÜBER EINEN NULLMODEN ADAPTER DATEN GESENDET</u>	<u>7</u>
<u>7</u>	<u>ZULETZT WURDE EIN C-PROGRAMM ZUR SERIELLEN DATENÜBERMITTLUNG GESCHRIEBEN</u>	<u>7</u>

2 Aufgabenstellung

1. Löte die Verbindungen an einem 9-pin Sub-D Stecker zu einen Seriellen Loopback Adapter mit Anschlusslaschen für das Oszi.
2. Sende seriell Daten mit dem Programm Putty an die Serielle Schnittstelle. Zeichne diese mit dem Oszi auf entschlüsse die Baudrate Parität Stopbit und das gesendete Zeichen.
3. Rufe das Programm `\\s1-wald\export\info\l-boeh\Comport\Comport2.exe` und entschlüsse das gesendete Zeichen analog zum Punkt 2.
4. Löte einen Nullmodem Adapter. Wiederhole den Punkt 2 in Zusammenarbeit mit deinem Platznachbar und dem Nullmodem Adapter. (Einer sendet der Andere empfängt die Daten.)
5. Schreibe ein C++ Programm welches über die Com1 laufen deinen Namen sendet (9600,8N1).

3 Anschlüsse

Als erstes wurde ein passender RS232 Adapter für die Serielle PC-Schnittstelle gelötet. Damit wurden am Oszilloskope die Pins 2,3 und 5 abgegriffen. Somit konnten die Datenleitungen (RX, TX) sowie die Signalmasse (GND, Pin 5) abgegriffen werden. s

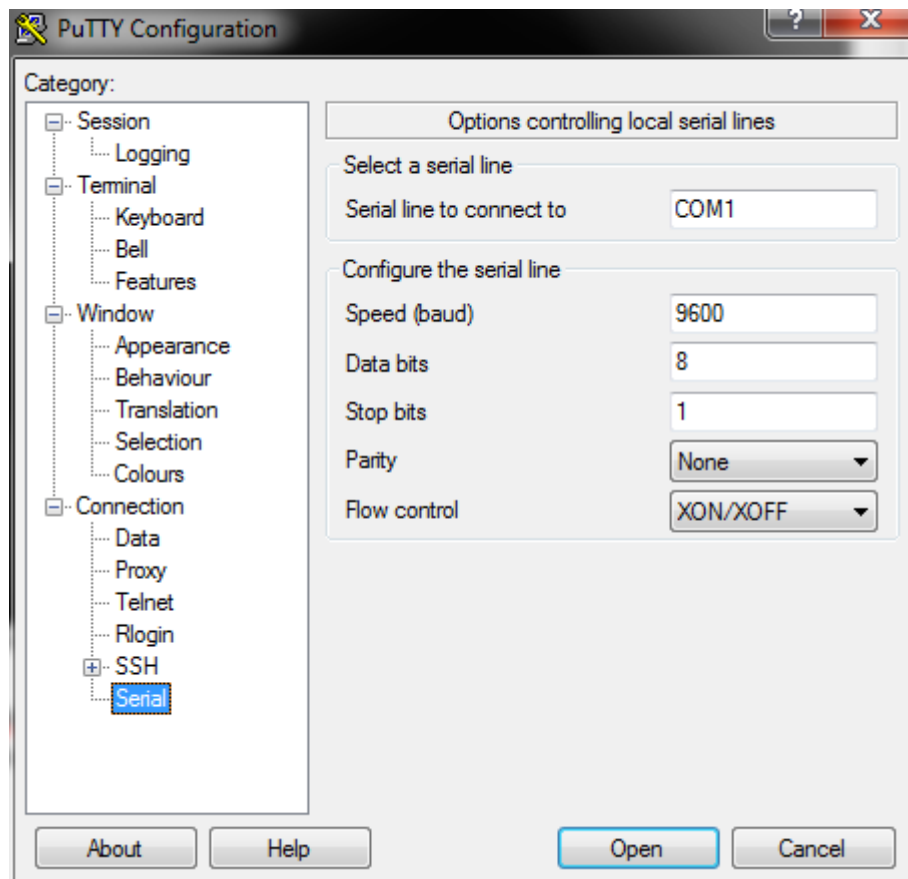


4 Daten über Putty senden

Mit dem Terminal Emulationsprogramm Putty 0.63 wurden Daten an die serielle Schnittstelle des PCs (COM1) gesendet. Die Daten wurden

1. Per Loopback Interface direkt am PC
2. Manuell durch messen der Datenleitung (RX) und GND

gemessen.

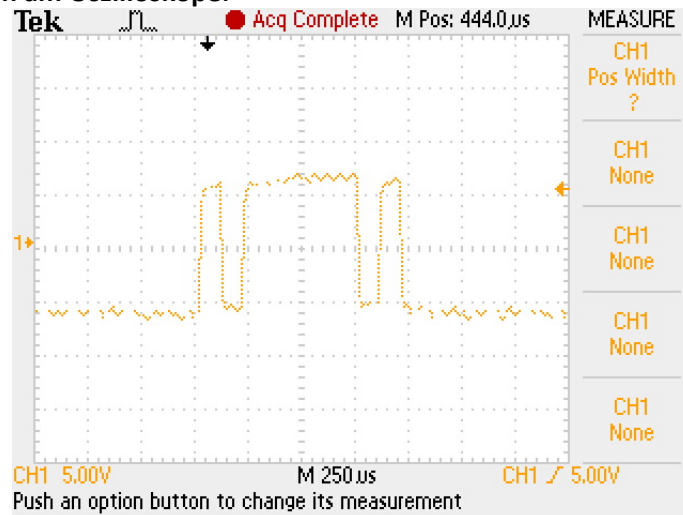


Anschließend wurden mit der initialisierten Sessione zwei Zeichen gesendet:

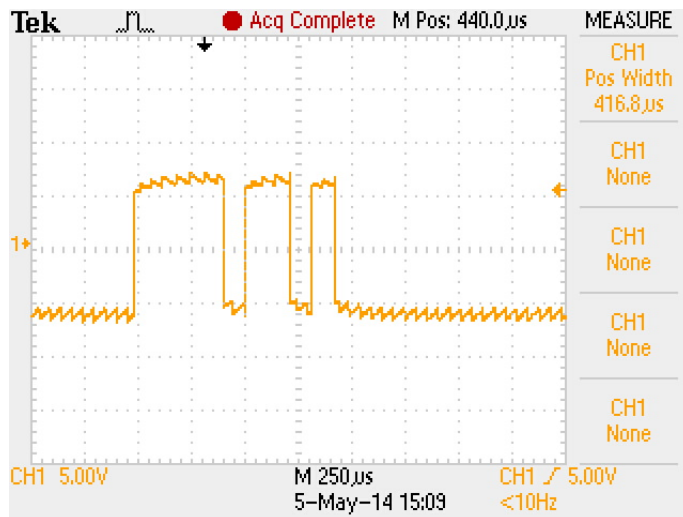
„AH“

➔ Steht für Alexander Hofstätter

Buchstabe „A“ gemessen am Oszilloskope:



Buchstabe „H“ gemessen am Oszilloskope:



Per seriellen loopback adapter am pc wurden zusätzlich diverse signalinformationen abgegriffen:

IOCTL_SERIAL_GET_BAUD_RATE: Retrieve Baud Rate

Baud Rate=9600

IOCTL_SERIAL_GET_LINE_CONTROL: Retrieve line control

WordLength=8

StopBits=1 stop bit

Parity=No parity

IOCTL_SERIAL_GET_CHARS: Retrieve special characters

EofChar=0x0

ErrorChar=0x0

BreakChar=0x0

EventChar=0x0

XonChar=0x11

XoffChar=0x13

IOCTL_SERIAL_GET_HANDFLOW: Retrieve handshake information

ControlHandShake=1

FlowReplace=67

XonLimit=2048

XoffLimit=512

000035: Write Request (DOWN), 05.05.2014 09:20:22.430 +3.030 (1. Device: Prolific USB-to-Serial Comm Port (COM1))

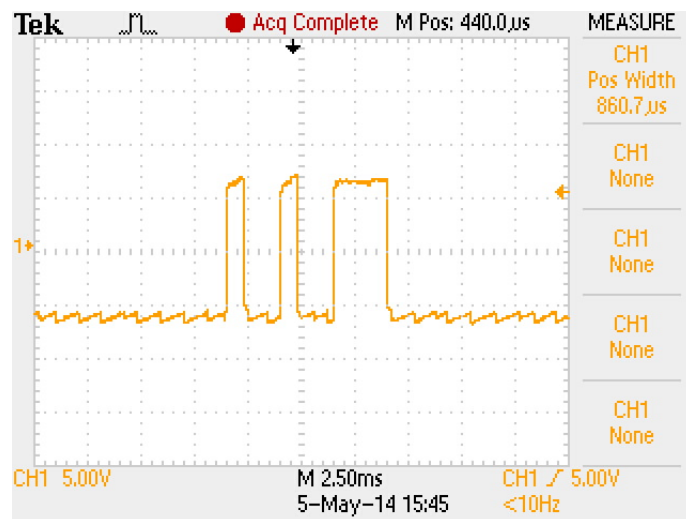
Buffer size: 0x2 bytes –

Die 2 bit sind in diesem Fall die reinen Nutzdaten (=Buchstaben) in der Ascii Kodierung. Aus dieser Bitfolge wurden die zwei Buchstaben A und H extrahiert.

A -> Ascii Code: 41

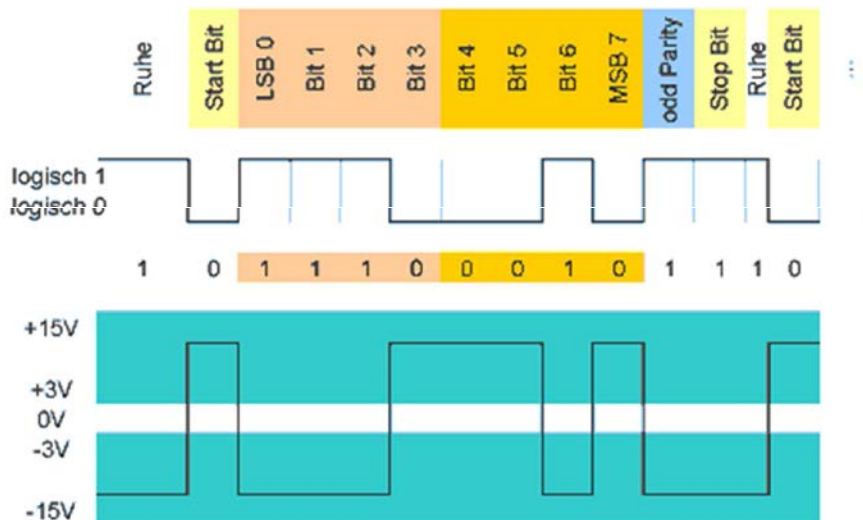
H -> Ascii Code: 48

5 Dekodieren der Nutzdaten und Signalinformationen aus der exe "Comport2.exe"



Synchronisation
Daten low & high
Check

9600 8O1 = 9600 Baud; 8 Datenbits; odd Parity; 1 Stopbit
ASCII "G" = \$47 = 0100 0111



Bitfolge: (LSB0-....-MSB6)
001 1011

Hex: 1 B

Das ergibt den Ordinalwert 27, dies ist der Wert für die Escape Sequenz ESC.

Baudrate: 1200, Parity: even, 7 daten bits, 1 stop bit

