

Synchronisation

- ◆ Es ist so das bei jedem Char-sendung 3-mal 11100000 oder 11100000 sendet und empfängt abhängig von der Position. Es sieht wie folgendes aus:
- ◆ Natürlich werden Masken zum Senden und Empfangen verwendet
- ◆ PC1 sendet 00000111 an PC2 und empfängt 11100000 und as wird 3-mal gemacht um Die Stabilität der Verbindung möglichst zu sicheren.
- ◆ PC2 sendet 11100000 an PC1 und empfängt 00000111 und das wird 3-mal gemacht um Die Stabilität der Verbindung möglichst zu sicheren
- ◆ Wir haben auch die Gleiche Delay an Jede Stelle der Sendung und des Empfangs eingestellt und es werden die gleiche Schritte an jedem PC ausgeführt.
„Programm-Spiegelung“
- ◆ Es hat geklappt weil die Bearbeitungszeit von der Char ungefähr gleich ist!

Char-sendung

Setting Adresses:

PC1

Richtung-Mask : 00001111

Sendung-Mask : 00000111

Empfang-Mask : 11100000

R	R	R	R	W	W	W	W
---	---	---	---	---	---	---	---

Beispiel: Sending 'x'

ASCII = 01111000

PC2

RichtungsMask : 11110000

Sendung-Mask : 11100000

Empfang-Mask : 00000111

W	W	W	W	R	R	R	R
---	---	---	---	---	---	---	---

Beispiel: Sending 'y' a in

b in ASCII = 01111001

Wir senden 'x' und 'y' in 3-Bits aufgetret auf 3-Male

Erste Sendung :

Sending: 00000011

Empfangen : 01100000

Sending: 01100000

Empfangen: 00000011

Dan warden die 3 bits in Vector gespeichert.

Char-sendung

Beispiel: Sending 'x'
ASCII = 01111000

Beispiel: Sending 'y' a in
b in ASCII = 01111001

Zweite Sendung :

Sending: 00000110

Empfangen : 11000000

Dan warden die 3 bits inVector gespeichert.

Sending: 11000000

Empfangen : 00000110

Beispiel: Sending 'x'
ASCII = 01111000

Beispiel: Sending 'y' a in
b in ASCII = 01111001

Ende der Sendung byte

Dritte Sendung :

Sending: 00000001

Empfangen : 01100000

Dan warden die 3 bits inVector gespeichert.

Sending: 01100000

Empfangen : 00000001

Char-sendung

Aufbau des gesendeten Bytes nach Shifting

01100000 | 00011000 | 00000000 = 01111000 = 'x'

01100000 | 00011000 | 00000001 = 01111001 = 'y'

Die Aufteilung des Bytes mit:

bitset<8> DevidePC2Byte(bitset<8> bit, int part);

bitset<8> DevidePC1Byte(bitset<8> bit, int part);

Der Aufbau des Bytes mit:

bitset<8> BuildPC2Byte(vector <bitset<8> > halter);

bitset<8> BuildPC1Byte(vector <bitset<8>> halter);



Datei und Nachricht Übertragung

Die Übertragung von Dateien erfolgt zeichenweise. Die Zeichen in einer Datei werden in Gruppen von 10 Zeichen aufgeteilt und anschließend einzeln übertragen. Gleichzeitig wird jedes empfangene Zeichen in einer Datei gespeichert.

Nachricht Übertragung wird wie Single Char gesendet, nur diesmal zahlreiche Zeichen je nach wie lang die Nachricht ist. Nach der Sendung wird den Text in einen Textfile gespeichert.



Geschwindigkeitsmessung

Die Größe der Datei ist 49 Byte

Die Geschwindigkeit unserer Übertragung ist 19,6 ms

Also die Geschwindigkeit = $49 / 19,6 = 2,5$ Byte/ms

Die Geschwindigkeit der Unix scp bei 49 Byte Datei ist 0,224

Also Die Geschwindigkeit = 218,75 Byte/ms

Das heißt Die Unix scp Übertragung ist ungefähr 9fache schneller!



Fotos

Char-sendung

```
[nhussain@ada5 V7Habibi]$ ./PC1.o 0
[B15F] Verwende Adapter: /dev/ttyUSB0
[B15F] Stelle Verbindung mit Adapter her... OK
[B15F] Teste Verbindung... OK
[B15F] AVR Firmware Version: Dec 3 2019 um 12:51:25 Uhr (boardinfo.h)
[nhussain@ada5 V7Habibi]$ ./PC1.o 1
[B15F] Verwende Adapter: /dev/ttyUSB0
[B15F] Stelle Verbindung mit Adapter her... OK
[B15F] Teste Verbindung... OK
[B15F] AVR Firmware Version: Dec 3 2019 um 12:51:25 Uhr (boardinfo.h)
Gebe einen Character zu senden: a
Establishing connection...
Establishing connection...
Establishing connection...
a zu binär: 01100001
Pc1 empfaengt : 01100010
Character Empfangen: b
[nhussain@ada5 V7Habibi]$ echo "Hi Fadi"|./PC1.o 2
```

Nachrichte-sendung

```
[nhussain@ada5 V7Habibi]$ echo "Hi Fadi"|./PC1.o 2
[B15F] Verwende Adapter: /dev/ttyUSB0
[B15F] Stelle Verbindung mit Adapter her... OK
[B15F] Teste Verbindung... OK
[B15F] AVR Firmware Version: Dec 3 2019 um 12:51:25 Uhr (boardinfo.h)
Establishing connection...
Establishing connection...
Establishing connection...
Pc1 empfaengt : 01001000
Character Empfangen: H
Pc1 empfaengt : 01101001
Character Empfangen: i
Pc1 empfaengt : 00100000
Character Empfangen:
Pc1 empfaengt : 01001110
Character Empfangen: N
Pc1 empfaengt : 01101111
Character Empfangen: o
Pc1 empfaengt : 01110101
Character Empfangen: u
Pc1 empfaengt : 01110010
Character Empfangen: r
Pc1 empfaengt : 00010111
Character Empfangen:
end sending
[nhussain@ada5 V7Habibi]$
```

```
[fmkhalal@ada6 V7Habibi]$ echo "Hi Nour"|./PC2.o 2
[B15F] Verwende Adapter: /dev/ttyUSB0
[B15F] Stelle Verbindung mit Adapter her... OK
[B15F] Teste Verbindung... OK
[B15F] AVR Firmware Version: Mar 10 2022 um 09:44:02 Uhr (boardinfo.h)
Establishing connection...
Establishing connection...
Establishing connection...
PC2 empfaengt: 01001000
Character Empfangen: H
PC2 empfaengt: 01101001
Character Empfangen: i
PC2 empfaengt: 00100000
Character Empfangen:
PC2 empfaengt: 01000110
Character Empfangen: F
PC2 empfaengt: 01100001
Character Empfangen: a
PC2 empfaengt: 01100100
Character Empfangen: d
PC2 empfaengt: 01101001
Character Empfangen: i
PC2 empfaengt: 00010111
Character Empfangen:
end sending
[fmkhalal@ada6 V7Habibi]$
```

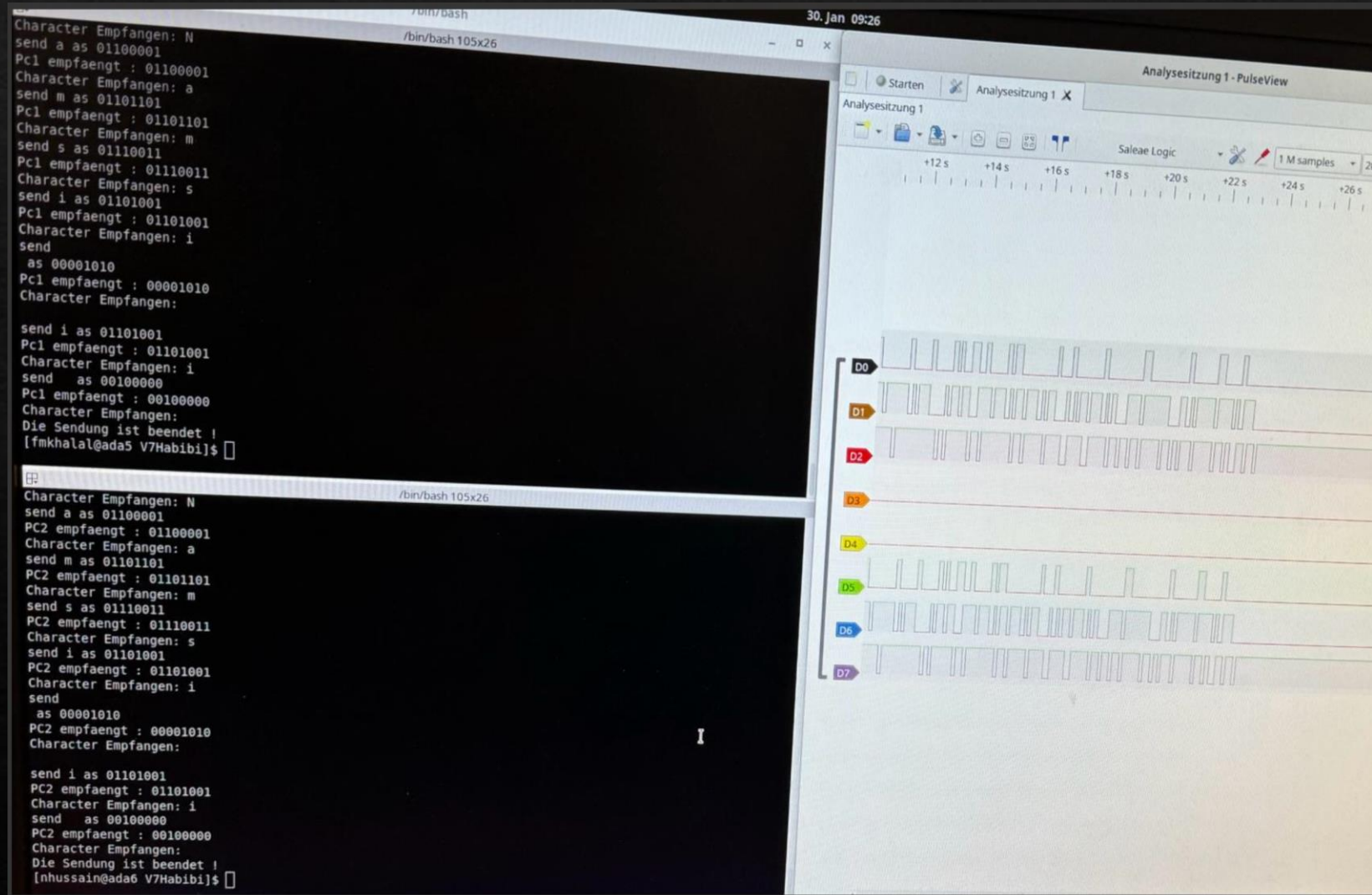
Datei-sendung

```
send 0 as 10010000
Pc1 empfaengt : 00000000
Character Empfangen:
send 3 as 00110011
Pc1 empfaengt : 00000000
Character Empfangen:
send 0 as 10010011
Pc1 empfaengt : 00000000
Character Empfangen:
send 0 as 10000010
Pc1 empfaengt : 00000000
Character Empfangen:
send as 00010100
Pc1 empfaengt : 00000000
Character Empfangen:
send 0 as 11011110
Pc1 empfaengt : 00000000
Character Empfangen:
send 0 as 00011010
Pc1 empfaengt : 00000000
Character Empfangen:
send 0 as 10101100
Pc1 empfaengt : 00000000
Character Empfangen:
Die Sendung ist beendet !
[fmkhalal@ada5 V7Habibi]$
```

```
send 0 as 10010000
PC2 empfaengt : 00000000
Character Empfangen:
send 3 as 00110011
PC2 empfaengt : 00000000
Character Empfangen:
send 0 as 10010011
PC2 empfaengt : 00000000
Character Empfangen:
send 0 as 10000010
PC2 empfaengt : 00000000
Character Empfangen:
send as 00010100
PC2 empfaengt : 00000000
Character Empfangen:
send 0 as 11011110
PC2 empfaengt : 00000000
Character Empfangen:
send 0 as 00011010
PC2 empfaengt : 00000000
Character Empfangen:
send 0 as 10101100
PC2 empfaengt : 00000000
Character Empfangen:
Die Sendung ist beendet !
[nhussain@ada5 V7Habibi]$
```


Fotos

Nutzung der Logik-analyser



Time der Unixscp

```
Transferred: sent 3432, received
Bytes per second: sent 35195.5,
debug1: Exit status 0

real    0m0,224s
user    0m0,023s
sys     0m0,019s
```