

Technische Universität
Clausthal Institut für Informatik
Prof. G. Kemnitz

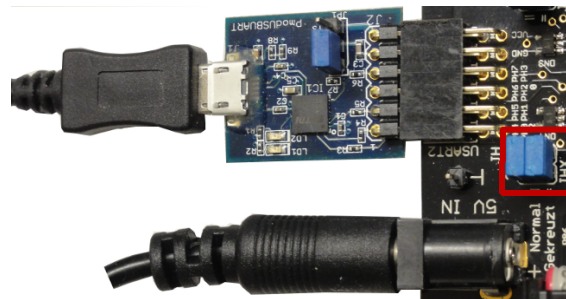
13. November 2018

Rechnerarchitektur: Laborübung 6 (USART, Timer)

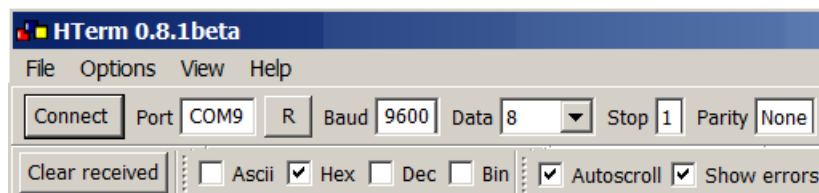
Hinweise: Schreiben Sie die Lösungen, so weit es möglich ist, auf die Aufgabenblätter. Tragen Sie Namen, Matrikelnummer und Studiengang in die nachfolgende Tabelle ein und schreiben Sie auf jedes zusätzlich abgegebene Blatt ihre Matrikelnummer. Lassen Sie für vorgeführte Experimente vom Betreuer die Punkte auf dem Aufgabenblatt eintragen und geben Sie, wenn Sie fertig sind, alle Blätter ab. Für eine Bescheinigung der erfolgreichen Teilnahme sind in jeder bis auf einer Laborübung mindestens 60% der Punkte zu erreichen.

Name	Matrikelnummer	Studiengang	Punkte von 12	≥ 60%

Aufgabe 6.1: Zum ausprobieren der seriellen Schnittstelle sind wie im nachfolgenden Bild ein PModUSBUSART an den Stecker JH zu stecken, dieses mit einem USB-Kabel mit dem PC zu verbinden und Jumper JHX auf »gekreuzt (=)« zu drehen.



Auf dem PC ist das Terminal-Programm HTerm  zu starten, der COM-Port auswählen¹, 9600 Baud, 8 Daten, 1 Stopp- und kein Paritätsbit einstellen und die Verbindung herzustellen (Connect).



Auf den Mikrorechner ist folgendes Echo-Programm zu laden:

```
#include <avr/io.h>           //Projekt: F3-echo
uint8_t daten;              //Datei: echo.c
int main(void){
    UCSR2C=0b110;           //Format 8N1
    UBRR2=51;               //9600 Baud
    UCSR2B=(1<<RXEN2)|(1<<TXEN2); //Empf. + Sender ein
    DDRJ = 0xFF;           //LEDs als Ausgänge
}
```

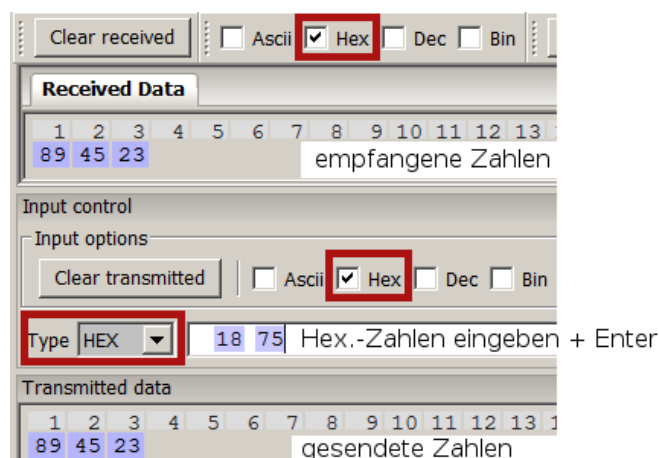
¹Die COM-Schnittstelle, die nach Anstecken des USB-Kabels vom PmodUSBURT und »R« (Refresh Comport List) als neuer Port erscheint.

```

PORTJ= 0; //LED-Ausgabe 0x00
while(1){
  while(!(UCSR2A &(1<<RXC2)));
  daten = UDR2;
  while(!(UCSR2A&(1<<UDRE2)));
  UDR2 = daten;
  PORTJ++;
}
}

```

- a) Senden Sie mit HTerm Zeichen und prüfen Sie, dass dieselben Zeichen empfangen werden. 2P



- b) Bestimmen Sie für jede Zeile der Endlosschleife, welche Aufgabe diese erfüllt und kontrollieren Sie Ihre Vermutungen im Schrittbetrieb. 2P
- c) Ändern Sie das Programm so, dass es jeweils auf zwei Bytes wartet und dann die Summe zurück schickt. 2P

Punkte Aufgabe 6.1	
--------------------	--

Aufgabe 6.2: Das nachfolgende Programm zählt unter Verwendung von Timer 1 im CTC-Modus die LEDs an Port J in gleichmäßigen Zeitschritten hoch:

```

#include <avr/io.h>
int main(void){
  TCCR1A = 0; // WGM1[1:0] = 0
  TCCR1B = 0b1101; // WGM1[3:2] = 1, CS1=0b101
  DDRJ = 0xFF;
  OCR1A = 0x1000; // Vergleichswert
  while(1){
    if (TIFR1 & (1 << OCR1A)){
      PORTJ++;
      TIFR1 = (1 << OCR1A); // Vergleichsbit löschen
    }
  }
}

```

- a) Wie groß ist der Zähltakt bei Voreinstellung »CS1=0b101«? Diese Information finden Sie in der Prozessorbeschreibung »techwww.../site/Dokumentation/ATmega2560/ATmega2560.pdf« in »Table 17-6«. 2P

- b) In welchen Zeitabstand zählen die LEDs hoch? Der Prozessor- und IO-Takt ist 8 MHz. Kontrollieren Sie, ob die LEDs auf der Baugruppe wirklich in diesem Zeitabstand weiterzählen.
2P
- c) Mit welchem Wert ist das Register OCR1A zu initialisieren und welcher Wert ist für CS1 zu verwenden, damit die LEDs die Zeit in Sekunden zählen? Kontrollieren Sie das Ergebnis mit der Baugruppe.
2P

Punkte Aufgabe 6.2	
--------------------	--