

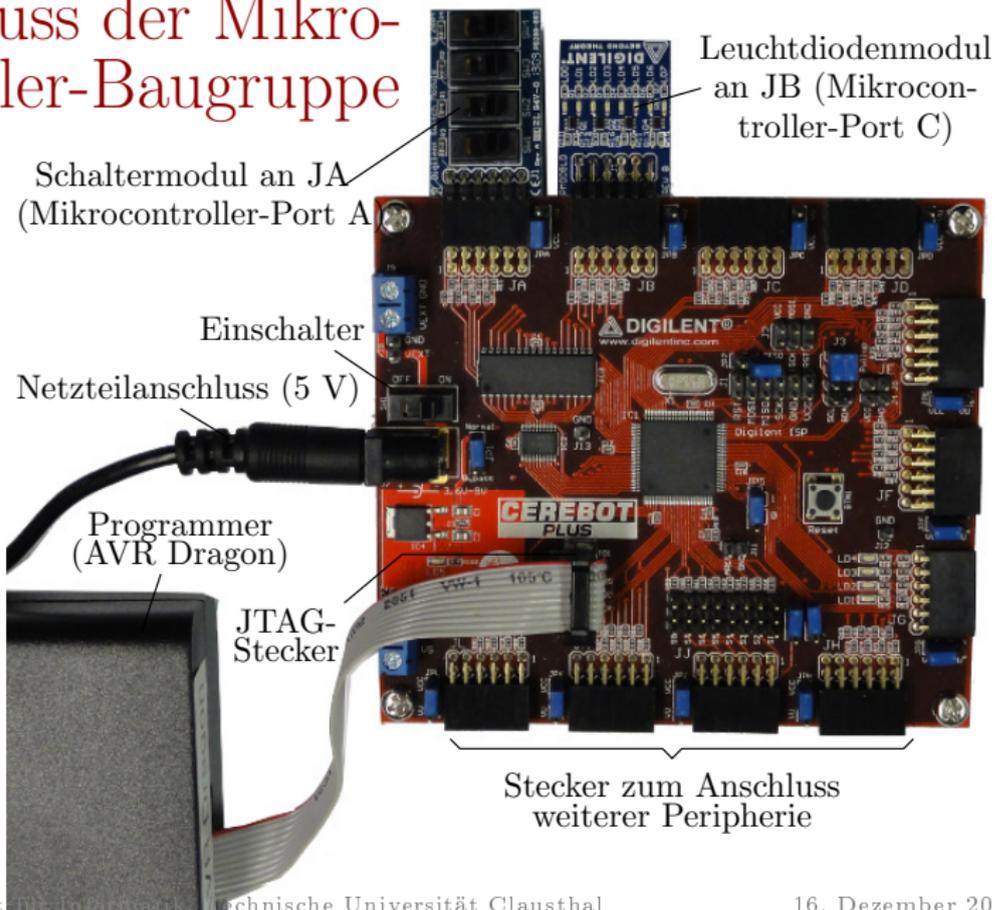


Rechnerarchitektur, Einführung in die Laborübungen

G. Kemnitz

Institut für Informatik, Technische Universität Clausthal
16. Dezember 2014

Anschluss der Mikrocontroller-Baugruppe



Verbindung mit dem PC

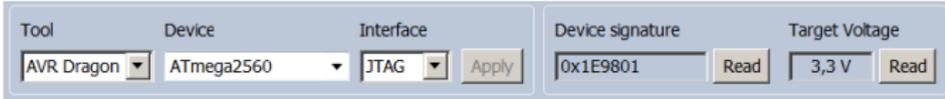
- Rechner unter Windows starten
- Web-Browser öffnen. Foliensatz zum Mitlesen öffnen:

`techwww.in.tu-clausthal.de/site/Lehre/Rechnerarchitektur/`

- Atmel Studio starten 

Zur Kontrolle, ob der Prozessor richtig angeschlossen und vom System erkannt wird, in Atmel Studio:

- Tools > Device Programming



The screenshot shows the 'Device Programming' dialog box in Atmel Studio. It is divided into two main sections. The left section contains three dropdown menus: 'Tool' set to 'AVR Dragon', 'Device' set to 'ATmega2560', and 'Interface' set to 'JTAG'. There is an 'Apply' button to the right of these dropdowns. The right section contains two input fields: 'Device signature' with the value '0x1E9801' and a 'Read' button to its right; and 'Target Voltage' with the value '3,3 V' and a 'Read' button to its right.

- Tool, Device, Interface einstellen; Apply, Read, Read: Device Signature und Target Voltage sollten richtig angezeigt werden.

- Kontrolle der Sicherungsbits (Fuses, Grundeinstellungen):

| Interface settings | Fuse Name | Value |
|--------------------|---|-------------------------------------|
| Tool information | <input checked="" type="checkbox"/> BODLEVEL | DISABLED ▾ |
| Device information | <input checked="" type="checkbox"/> OCDEN | <input type="checkbox"/> |
| Memories | <input checked="" type="checkbox"/> JTAGEN | <input checked="" type="checkbox"/> |
| Fuses | <input checked="" type="checkbox"/> SPIEN | <input checked="" type="checkbox"/> |
| Lock bits | <input checked="" type="checkbox"/> WDTON | <input type="checkbox"/> |
| Production file | <input checked="" type="checkbox"/> EESAVE | <input type="checkbox"/> |
| | <input checked="" type="checkbox"/> BOOTSZ | 4096W_1F000 ▾ |
| | <input checked="" type="checkbox"/> BOOTRST | <input type="checkbox"/> |
| | <input checked="" type="checkbox"/> CKDIV8 | <input type="checkbox"/> |
| | <input checked="" type="checkbox"/> CKOUT | <input type="checkbox"/> |
| | <input checked="" type="checkbox"/> SUT_CKSEL | EXTXOSC_3MHZ_8MHZ_1KCK_0MS ▾ |

Für Praktika sollten JTAGEN (JTAG Enabled), SPIEN (SPI Enabled) und OCDEN (On-Chip Debugging Enabled) gesetzt und WDTEN (WatchDog Timer Enabled) nicht gesetzt sein.

Das erste Programm

```
9  #include <avr/io.h>
10 void main(){
11     |   DDRA = 0;
12     |   DDRC = 0xFF;
13     |       while(1){
14     |           PORTC = PINA;
15     |       }
16 }
```

Projekt anlegen:

- Projekt anlegen: File > New > Project > Auswahl: GCC Executable Project, Name: Sw2Led, OK > Device: rechts oben ATmega2560 eingeben und dann links auswählen, OK
- Programm wie oben eingeben.
- Übersetzen: Build > Build Solution (F7). Wenn Fehler angezeigt werden, diese beseitigen.

- Programmer auswählen: Project > SW2Led Properties (Alt+F7) > Tools > Select debugger ...: AVR Dragon ..., Interface: JTAG; Speichern (Strg+S).
- Programm im Debugger-Modus starten: Debug > Start Debugging and Break (Alt+F5).
- Disassembliertes Programm anzeigen: Debug > Windows > Disassembly (Alt+8).

| Address | Hex | Assembly | Comment |
|----------|-------|----------------|----------------------|
| 0000007D | 17.b8 | OUT 0x07,R1 | Out to I/O location |
| 16: | | PORTC = PINA; | |
| 0000007E | 80.b1 | IN R24,0x00 | In from I/O location |
| 0000007F | 88.b9 | OUT 0x08,R24 | Out to I/O location |
| 00000080 | fd.cf | RJMP PC-0x0002 | Relative jump |

| | | | |
|--|---------------------|--|--|
| | Befehlszähler | | Einzelschritt |
| | Unterbrechungspunkt | | Start bis zum nächsten Unterbrechungspunkt |

Der Befehlszähler zeigt auf die Startadresse von main()(0x7D).

Beobachtung der Register und Ports

Im folgenden Test im Schrittbetrieb sind das Register R24 und die Ports A und C zu beobachten.

The screenshot shows two windows from a debugger. The 'IO View' window on the left displays a list of I/O components. The 'Processor' window on the right shows the state of various processor registers and status flags.

| Name | Value |
|--------------|-------|
| JTAG | |
| PORTA | |
| PORTB | |
| PORTC | |
| PORTD | |
| PORTE | |

| Name | Address | Value | Bits |
|-------|---------|-------|-----------------|
| PINA | 0x20 | 0x00 | □ □ □ □ □ □ □ □ |
| DDRA | 0x21 | 0x00 | □ □ □ □ □ □ □ □ |
| PORTA | 0x22 | 0x00 | □ □ □ □ □ □ □ □ |

| Name | Value |
|------------------|-------------|
| Program Counter | 0x0000007D |
| Stack Pointer | 0x21FC |
| X Register | 0x0000 |
| Y Register | 0x21FF |
| Z Register | 0x01FF |
| Status Register | I T H S V F |
| Cycle Counter | 0 |
| Frequency | |
| Stop Watch | |
| Registers | |
| R00 | 0x00 |
| R01 | 0x00 |

- Register anzeigen: Debug > Windows > Processor View.
- EA-Register anzeigen: Debug > Windows > IO-View.

Test des Programms

Disassembly x SW2Led SW2Led.c

| | | | |
|------------|-------|----------------|----------------------|
| → 0000007D | 17.b8 | OUT 0x07,R1 | Out to I/O location |
| 16: | | PORTC = PINA; | |
| 0000007E | 80.b1 | IN R24,0x00 | In from I/O location |
| 0000007F | 88.b9 | OUT 0x08,R24 | Out to I/O location |
| ● 00000080 | fd.cf | RJMP PC-0x0002 | Relative jump |

| | | | |
|---|---------------------|---|--|
| → | Befehlszähler | ↵ | Einzelschritt |
| ● | Unterbrechungspunkt | ▶ | Start bis zum nächsten Unterbrechungspunkt |

- Mit den Schaltern unterschiedliche Werte einstellen.
- Im Schrittbetrieb beobachten, wie die Schalterwerte zur Ausgabe transferiert werden.
- Es ist auch möglich, wenn das Programm hält, die angezeigten Registerinhalte im Prozessor zu ändern.



Fakt 1

Prägen Sie sich die einzelnen Arbeitsschritte für die nachfolgenden Aufgaben ein.