

Kurzanleitung des seriellen oder USB- ISP-Programmers

H003b / H003c

V 2.2

11. Juli 2007



© 2007 by Peter Küsters

Dieses Dokument ist urheberrechtlich geschützt. Es ist nicht gestattet, dieses Dokument zu verändern und komplett oder Teile daraus ohne schriftliche Genehmigung von uns weiterzugeben, es zu veröffentlichen; es als Download zur Verfügung zu stellen oder den Inhalt anderweitig anderen Personen zur Verfügung zu stellen. Zuwiderhandlungen werden verfolgt.

Sie haben unseren seriellen ISP-Programmer H003b oder die Kombination aus USB-RS232 Konverter und seriellen Programmer (H003c) erworben. Vielen Dank.

Unterstützte Controller

Eines sollten Sie vorab wissen: der serielle Programmer beinhaltet in seinem Gehäuse einen eigenen Mikrocontroller. Dieser kennt nur eine begrenzte Anzahl an Atmel-Controllern. Z.Zt. sind dies die folgenden Controller, d.h. nur diese werden vom seriellen Programmer unterstützt:

AT90:							
AT90S1200	AT90S2313	AT90S2323	AT90S2333	AT90S2343	AT90S4414	AT90S4433	AT90S4434
AT90S8515A	AT90S8534	AT90S8535	AT90S8544	AT90PWM2 ⁵⁾	AT90PWM3 ⁵⁾	AT90CAN128 ⁵⁾	
ATTiny:							
ATTiny10	ATTiny12	ATTiny13 ⁵⁾	ATTiny15	ATTiny24 ⁵⁾	ATTiny25 ⁵⁾	ATTiny26	ATTiny44 ⁵⁾
ATTiny45 ⁵⁾	ATTiny84 ⁵⁾	ATTiny85 ⁵⁾	ATTiny2313 ⁴⁾⁵⁾				
ATMega:							
ATMega8	ATMega16	ATMega32	ATMega48 ⁵⁾	ATMega64 ¹⁾⁵⁾	ATMega88 ⁵⁾	ATMega103	ATMega128
ATMega161	ATMega162 ⁵⁾	ATMega163	ATMega164 ⁵⁾	ATMega168 ⁵⁾	ATMega169 ¹⁾⁵⁾	ATMega324 ⁵⁾	ATMega329 ⁵⁾
ATMega640 ⁵⁾	ATMega649 ⁵⁾	ATMega1280 ⁵⁾	ATMega1281 ⁵⁾	ATMega3290 ⁵⁾	ATMega6490 ⁵⁾	ATMega8515	ATMega8535 ¹⁾⁵⁾
ATMega2560	ATMega2561						
AT89 (8051 kompatibel, kein AVR)							
AT89S53 ²⁾³⁾	AT89S8252 ³⁾						

1) Support durch AVRProg ab V1.40.

2) Bug in AVRprog. Bei Programmierung der Fuse Bits. Typ 89S8252 einstellen.

3) Bug in AVRprog verhindert das Programmieren im Blockmode (Blockmode abschalten, siehe Source NOTE 18)

4) Bei AVRProg statt ATTiny2313 den ATTiny26 einstellen. Für Flash write und AVRProg getestet.

5) Mit Modifikation in avrdude.conf für avrdude zu verwenden.

Die jeweils aktuelle Liste der Controller und der Firmware des Controllers ist in unserem Internet-Shop auf der Produktseite des seriellen Controller abgedruckt:

www.shop.display3000.com

Im Klartext: Wenn Sie einen anderen Controller als hier aufgelistet nutzen möchten, wird dies vermutlich nicht funktionieren.

Die Bedienung ist unproblematisch und bedarf eigentlich keines separaten Manuals. Trotzdem sind gelegentlich bei dem einen oder anderen Punkt Fragen aufgekommen, die wir hiermit sozusagen präventiv beantworten möchten.

Wir möchten hier in diesem Manual auf 5 Aspekte eingehen:

- Bedeutung der LED-Anzeigen
- Baudratenumschaltung
- Problembehandlung
- Nutzung mit Adapter USB-RS232
- Nutzung unter WinAVR (mit AVRDUDE)
- Nutzung unter Bascom

Der serielle Display3000-ISP-Programmer hat drei Besonderheiten, die hier erwähnt werden sollen:

- a) Der Gender Changer ist nicht mehr notwendig – der Programmieradapter kann nun direkt –ohne Zwischenadapter– an den PC gesteckt werden.
- b) Er wird inkl. zwei integrierter Leuchtdioden geliefert. So erkennen Sie mit einem Blick auf den Programmer, ob er korrekt mit Spannung versorgt wird und dass ein Programmiervorgang auch abläuft
- c) Unser Programmer ist mit der neuesten Software ausgestattet, die eine Baudratenumschaltung von der **Standardeinstellung 115.200 Baud** auf 19.200 Baud erlaubt.

Wenn Sie zusätzlich unseren USB-RS232 Adapter einsetzen (diese Kombination ist getestet, es funktionieren NICHT alle am Markt erhältlichen USB-RS232 Adapter), gilt all das nachfolgende ebenfalls – lediglich muss hier die gewünschte Baudrate auch noch in der USB-Software geändert werden.

Die LEDs

Der Programmer erhält seine Spannung aus der Schaltung mit dem Atmel Controller. Nach Anstecken an den PC geschieht daher erstmal nichts – keine LED leuchtet. Sobald der Programmer dann mit dem ISP Stecker in Ihrem Display3000 Modul verbunden wird, erhält er Spannung und führt einen Selbsttest durch: Es leuchtet die rote LED auf, dann leuchtet zusätzlich die grüne LED; die rote LED verlischt dann nach kurzer Zeit und es leuchtet nur noch die grüne LED. Der Programmer ist nun betriebsbereit.

Während eines Programmiervorgangs erlischt die grüne LED und es leuchtet nur die rote LED, nach dem Programmiervorgang leuchtet wieder die grüne LED.

Rot = Programmiervorgang aktiv

Grün = Ready

Test der Verbindung und Baudratenumschaltung:

Standardmäßig wird der Programmierer mit 115.200 Baud geliefert. Möchten Sie diese Verbindung testen oder diese Baudrate des ISP-Programmers ändern, dann hilft Ihnen die folgende Anleitung:



Sie müssen hierzu mithilfe eines Terminalprogramms auf den Programmierer zugreifen. Windows liefert bis zu Vista ein solches Terminalprogramm mit, es nennt sich Hyperterminal und befindet sich meist unter:

Start - Alle Programme – Zubehör – Kommunikation – Hyperterminal

Ab Vista gibt es kein mitgeliefertes Terminalprogramm mehr. Hier sind Sie nun auf andere Anbieter angewiesen. Eine mögliche Alternative ist der Hersteller des Terminalprogramms Hyperterm, welches Microsoft viele Jahre in Windows integriert hat und an das Sie sich evtl. bereits gewöhnt hatten.

Der Link zum Download: <http://www.hilgraeve.com/hpte/download.html>

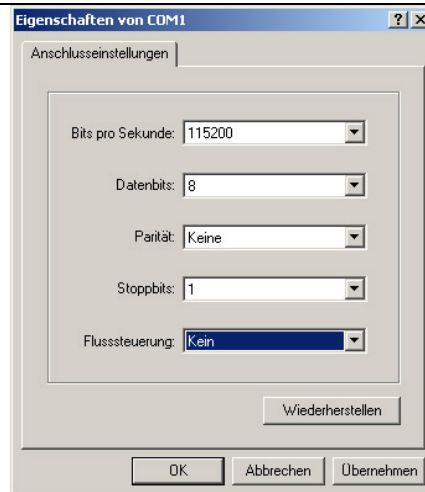
Tipps zur Verbindungsherstellung mit Hyperterm:

<p>Legen Sie eine neue Verbindung an und nennen diese z.B. „AVR910 ISP Programmierer 115200“</p>	
<p>Nun wählen Sie ihre serielle Schnittstelle (meist COM1) aus,</p> <p>Bei Nutzung eines USB-RS232 Adapters kann dies auch COM3 oder COM4 sein. COM5 und höher können die meisten Programme zur Programmierung nicht verarbeiten, daher sollten Sie eine solche Einstellung beim USB-Adapter vermeiden / ändern.</p>	

... und legen im nächsten Menü die Einstellungen fest:
 115.200 Baud
 8 bit
 keine Parität
 1 Stoppbit
 keine Flusststeuerung

Standardmäßig wird der Programmierer mit der Einstellung 115.200 Baud geliefert. Daher müssen Sie beim ersten Mal auch diese Werte angeben.

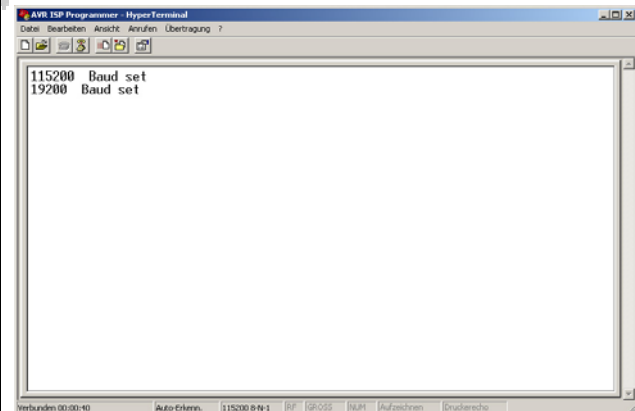
Nach Bestätigen durch OK wird der leere Terminalbildschirm aufgerufen.



Zur **Kontrolle einer korrekten Verbindung** zwischen Computer und Programmieradapter können Sie nun testweise die LEDs umschalten: mittels zweimaligem Drücken der Taste „x“ also „xx“ (grüne LED an) bzw. „y“ (rote LED an). Wenn Sie keine Baudrate verstellen möchten, verlassen Sie das Terminalprogramm wieder.

Durch Eingabe von „-0“ (Minus Null) erscheint auf dem Bildschirm
115200 Baud set

Durch Eingabe von „-1“ (Minus Eins) erscheint auf dem Bildschirm
19200 Baud set



Achtung: Wenn Sie den Programmieradapter von 115.200 Baud (Standardeinstellung) auf 19.200 Baud umgestellt haben, ist nun ein weiteres Ansprechen nicht mehr möglich, denn Hyperterminal arbeitet weiterhin mit 115.200 Baud.

Um nun die korrekte Geschwindigkeit zu prüfen (siehe gelber Kasten weiter oben) und/oder die Baudrate wieder auf 115.200 zurückzusetzen, muss in Hyperterminal eine neue Verbindung geöffnet werden. Nun geben Sie alles wie oben gezeigt ein, allerdings wählen Sie bei der Baudrate die nun aktuelle (da gerade geänderte) Geschwindigkeit des Programmers von 19.200 Baud aus.

Wenn Sie nun in den Hyperterminalbildschirm gelangen, geben Sie zur Kontrolle „-1“ ein und es erscheint **19200 Baud set**. Das Umsetzen auf 115.200 Baud geschieht dann wie oben bereits erwähnt mit „-0“

Der serielle Programmierer unterstützt nur die beiden Baudraten
115.200 Baud (Standardeinstellung) und
19.200 Baud.

Anmerkung: Wenn nicht unbedingt notwendig, sollten Sie die Baudrate auf 115.200 Baud belassen, ansonsten dauert der Programmiervorgang seeeeehr lange.

Kein Ansprechen des Programmieradapters mehr möglich

In sehr selten Fällen kam es in der Vergangenheit vor, dass der Programmieradapter nicht mehr reagiert. Beim Start leuchten zwar die Leuchtdioden - wie oben beschrieben - korrekt auf, aber dann wird der Adapter von der Programmiersoftware nicht gefunden. Der Mikrocontroller im Programmieradapter muss dann zurückgesetzt werden. Dies ist ohne großen Aufwand möglich und kann von Ihnen selbst durchgeführt werden. Hierzu senden wir Ihnen auf Anfrage eine Kurzanleitung zu. Bitte senden Sie uns hierzu eine Mail mit Angabe Ihrer Kundennummer und des ungefähren Kaufdatums des Programmieradapters.

Ab Juli 2007 sollte das Problem Dank einer neuen Firmware im Programmieradapter nicht mehr auftreten.

Nutzung mit USB-Adapter

Wenn Sie einen USB-RS232-Adapter mitbestellt haben, so wird mittels dieses Adapters eine serielle Schnittstelle erstellt. Unter Windows XP ist evtl. keine Treiber-Installation notwendig; wenn doch oder wenn Sie ein andere Betriebssystem nutzen, so müssen Sie von der beiliegenden CD des Adapters den Treiber installieren. Dieser Adapter ist ein *USB 1.1 to RS232-Converter* – wählen Sie dies im Auswahlménü der CD aus (unten Mitte).

Der Adapter funktioniert sowohl an USB 1.1 als auch an USB 2.0 Anschlüssen. Achten Sie darauf, dass Sie eine Com-Schnittstelle 1 bis 4 emulieren, andere werden u.U. von den diversen Programmier-Tools nicht unterstützt. Sie erkennen dies im Gerätemanager von Windows (Start, Systemsteuerung, System, Hardware, Geräte-manager – dort unter „Anschlüsse COM und LPT“). Dann Doppelklick auf „prolific USB to RS232....“, dann *Anschlusseinstellungen* und *Erweitert*. Lesen Sie hierzu auch das Manual auf der Treiber CD des Adapters.

Betrieb unter WinAVR (C-Compiler) mit dem integrierten AVR Dude

Der mitgelieferte Programmierer in WinAVR nennt sich AVR Dude.

Um diesen seriellen Programmierer anzusprechen, ist eine entsprechende Änderung in Makefile notwendig.

Neben der Änderung des Programmieradapters empfiehlt sich zudem die Angabe der Baudrate, denn AVR Dude unterstützt standardmäßig lediglich 19.200 Baud, was doch recht langsam ist.

Folgendermaßen sollten die *Programming Options* in Makefile aussehen; die letzte Option ist sinnvoll um mit maximaler Geschwindigkeit programmieren zu können.

```
.
.
#----- Programming Options (avrdude) -----

# Programming hardware: alf avr910 avrisp bascom bsd
# dt006 pavr picoweb pony-stk200 sp12 stk200 stk500
#
# Type: avrdude -c ?
# to get a full listing.
#
AVRDUDE_PROGRAMMER = avr910

# com1 = serial port. Use lpt1 to connect to parallel port.
AVRDUDE_PORT = com1      # programmer connected to serial device

AVRDUDE_WRITE_FLASH = -U flash:w:$(TARGET).hex
#AVRDUDE_WRITE_EEPROM = -U eeprom:w:$(TARGET).eep

# Uncomment the following if you want avrdude's erase cycle counter.
# Note that this counter needs to be initialized first using -Yn,
# see avrdude manual.
#AVRDUDE_ERASE_COUNTER = -y

# Uncomment the following if you do /not/ wish a verification to be
# performed after programming the device.
AVRDUDE_NO_VERIFY = -V

# Increase verbosity level. Please use this when submitting bug
# reports about avrdude. See <http://savannah.nongnu.org/projects/avrdude>
# to submit bug reports.
#AVRDUDE_VERBOSE = -v -v

AVRDUDE_FLAGS = -p $(MCU) -P $(AVRDUDE_PORT) -c $(AVRDUDE_PROGRAMMER)
AVRDUDE_FLAGS += $(AVRDUDE_NO_VERIFY)
AVRDUDE_FLAGS += $(AVRDUDE_VERBOSE)
AVRDUDE_FLAGS += $(AVRDUDE_ERASE_COUNTER)
AVRDUDE_FLAGS += -b 115200

#----- Debugging Options -----
.
.
```


Betrieb unter Bascom mit dem integrierten Bascom-Programmier-Tool

Bascom und serielle ISP Programmer nach dem AVR910 Standard haben noch nie wirklich miteinander harmoniert.

Angeblich soll ab der Bascom-Version 1.11.8.3 auch dieser Programmer (der die neueste Firmware enthält) nun unterstützt werden. Nun – wir haben es probiert und haben keine zufrieden stellende Lösung gefunden: Abgesehen davon, dass die aktuell eingestellten Fuses nicht korrekt angezeigt wurden (was tragisch enden kann, wenn Sie vermeintlich falsche Fuses ändern), dauerte der Programmiervorgang eine halbe Ewigkeit (ein Code mit 27 KByte benötigte über 40 Minuten bei 19.200 Baud!)

Aber evtl. liegt dies nur an unserer Konfiguration und bei Ihnen klappt dies. Nachfolgend also die Vorgehensweise für Bascom – **jedoch ausdrücklich unter Vorbehalt und ohne jeglichen Support von uns** (besser: separates Programm z.B. OSP II, wie auf der übernächsten Seite beschrieben):

Wählen Sie in Bascom unter Options-Programmer den allerersten Eintrag aus „AVR ISP Programmer“ und wählen nun den korrekten Com-Port sowie die Baudrate aus. Uns liegt die Information vor, dass dies standardmäßig nur mit 19.200 Baud funktioniert, daher empfehlen wir, für einen ersten Test, den ISP Programmieradapter auf 19.200 Baud umzuprogrammieren, wie am Anfang dieses Manuals beschrieben wurde.

Der erste Test unter Bascom

Rufen Sie nun den Programmer von Bascom auf und klicken dann auf Fuses. **ÄNDERN SIE DORT BITTE ERST EINMAL NICHTS!!!!** Schauen Sie, ob die Einstellungen korrekt sein können.

Bei unserem Test wurde z.B. bei einem ATmega128 mit internem 8Mhz Takt angezeigt:

<u>Dargestellt:</u>	<u>richtig wäre:</u>
externer Takt (0000)	interner Takt 8 Mhz
ATmega103-Kompatibilitätsmodus	ATmega128 Modus

Beides wurde also definitiv falsch angezeigt – keine gute Voraussetzung für eine weitere Zusammenarbeit. Ist bei Ihnen jedoch alles OK, haben Sie Glück und Sie können die nächste Stufe testen.

Wenn Sie die gleichen Fehler haben, stoppen Sie hier und beenden das Experiment. Nutzen sie ein separates Tool zum Programmieren (= gehe sofort zu Seite 12). Trösten Sie sich damit, dass Sie dieses Schicksal mit tausenden anderer Anwender teilen und auch wir können Ihnen nicht mehr weiterhelfen. Wenden Sie sich dann an den Support von Bascom unter www.mcselec.com.

Nutzung des Programmieradapters aus Bascom mit 115.200 Baud.

(Dieser Text gilt wie bereits oben erwähnt ohne Erfolgsgarantie!!)

Wenn Bascom die Fuses richtig dargestellt hat (und nur dann!) macht es Sinn, die nächste Stufe zu testen:

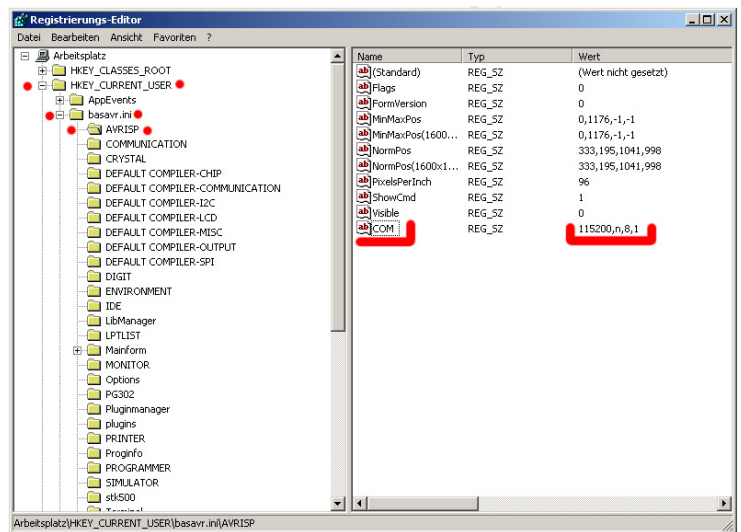
- 1) Beenden Sie Bascom
- 2) Programmieren Sie den Adapter auf 115.200 Baud um wie oben beschrieben und starten Sie dann das Programm *Regedit* aus Windows (*Start-Ausführen* (oder Taste *Windows-R*) und *Regedit* eintippen).

Klicken Sie sich nun nach:
 HKEY_Current_User
 Basavr.ini
 AVR-ISP

Dort wählen Sie nun:
Bearbeiten, Neu, Zeichenfolge

Der Zeichenfolge geben Sie den Wert **COM** und tragen dort ein: **115200 ,n, 8, 1**
 Verlassen Sie nun *Regedit* und starten *Bascom* neu.

Nun, unter *Options-Programmer* wählen Sie den korrekten Com-Port und wählen bei der Baudrate diesmal den Eintrag 115.200 Baud und versuchen erneut, ob Sie eine Verbindung mit dem ISP-Programmer aufnehmen können.

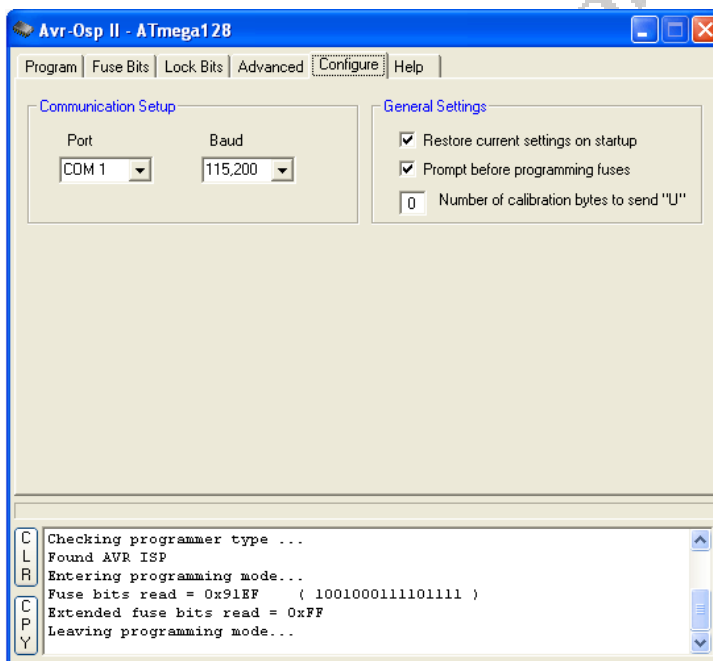
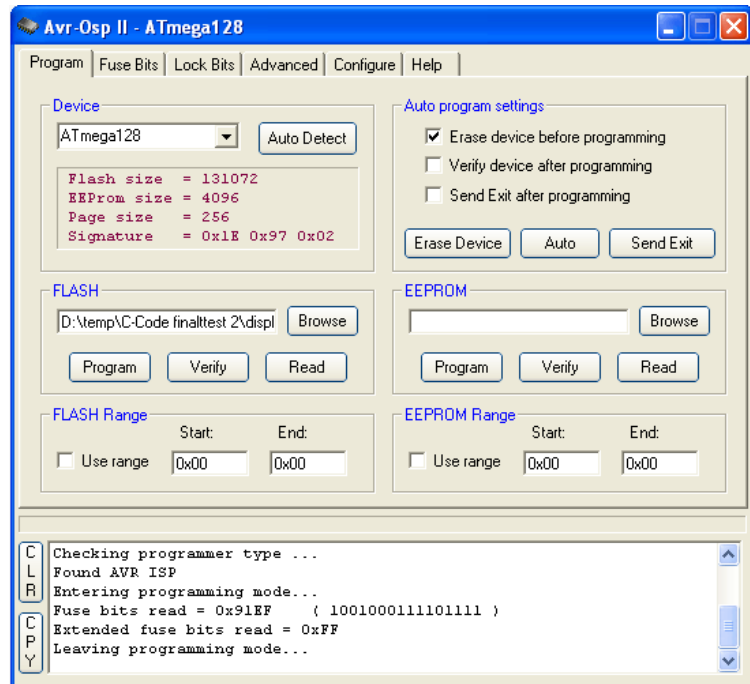


Serieller/USB-Programmer

Nutzen Sie bitte nur das auf der CD mitgelieferte Tool OSP II. Die Vorgängerversionen von OSP II hatten Probleme mit Bascom-Daten sowie dem ATmega2561. Zusammen mit Mike, dem Programmierer haben wir die Probleme eingegrenzt, so dass nach vielem Testen nun endlich (Stand 11.2.2007) sowohl Bascom-Hex-Daten als auch der AT-Mega2561 korrekt unterstützt werden.

Das bislang von uns propagierte AVRProg sollten sie nicht mehr nutzen – OSP II ist weitaus anwenderfreundlicher,

wie Sie anhand des nebenstehenden Bilder erkennen können. Eine Einstellung ist ganz wichtig: Setzen Sie bitte das Häkchen bei „Erase device before programming“. Alle anderen Werte bleiben in der Regel auf den Default-Werten.

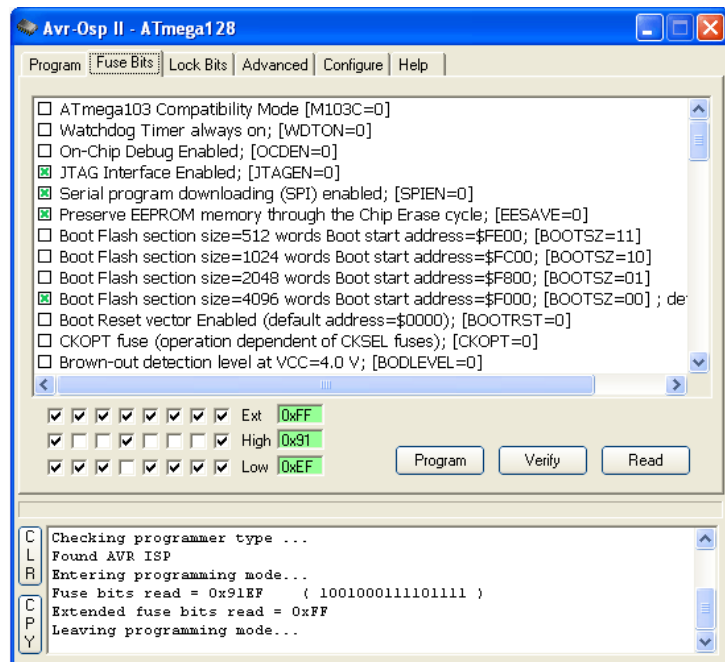


OSP II muss noch auf die von Ihnen genutzte Schnittstelle eingestellt werden. Rufen Sie dazu den nebenstehenden Bildschirm auf und stellen unter Port Ihre Schnittstelle und unter Baud den Wert „115.200“ ein – das ist die von unserem Programmieradapter voreingestellte Geschwindigkeit.

So wie es aussieht, kann man mit OSP II nun endlich auch die Fuses korrekt einstellen.

Wenn der Bildschirm „Fuse Bits“ aufgerufen wird, sind zuerst einmal die zuletzt gebrannten Einstellungen zu sehen. Der Bildschirm zeigt also NICHT die aktuellen Werte. Dazu ist zuerst einmal auf „Read“ zu klicken.

Warnung: Wie immer bei Fuses gilt: Eine falsche Einstellung kann Ihren Controller ins Nirwana schicken. Dann hilft nur noch der Austausch! Also ändern Sie hier bitte nur etwas, wenn Sie genau wissen, was Sie tun. Im Zweifel fragen Sie uns bitte!



Bascom und das mitgelieferte Programmiertool OSP II

Wenn sie einen seriellen Programmieradapter haben, kann leider auch nur weiterhin die Empfehlung bei Bascom sein: Nutzen Sie ein separates Programm zur Programmierung!

Wir können es absolut nicht empfehlen, den Programmer von Bascom zur Programmierung zu nutzen. Was also tun mit Bascom:

Der Ablauf ist dann wie folgt:

- a) Kompilieren in Bascom.... Bascom erzeugt eine Datei *Programmname.Hex*
- b) Wechseln zu OSP II und Einladen der von Bascom erzeugten xx.Hex Datei

Gleiches gilt natürlich für jedes andere Programmierwerkzeug: Wenn dieses Hex-Dateien erzeugt, so können Sie diese mit OSP II brennen.

In OSP II wählen Sie im Feld „Flash“ (siehe Seite 9, das Feld in der Mitte links) die von Bascom erstellte .HEX Datei aus, welche dem compilierten Code für den Controller enthält. Mittels Klick auf „Program“ schreiben Sie diese Datei dann in den Atmel-Controller.

Zwei praktische Effekte:

- 1) Während des Schreibvorgangs (der bei einem großen Programm ja durchaus 2 Minuten und länger dauern kann) können Sie nun wieder zu Bascom wechseln und dort während des Programmiervorganges an Ihrem Programm weiterarbeiten (dies geht mit dem in Bascom integrierten Programmer sonst nicht – da können Sie nur Warten, bis der Vorgang beendet ist).
- 2) OSP II merkt sich die zuletzt programmierte Datei – solange Sie den Namen Ihres Programms nicht ändern (und damit bleibt die Hex-Datei dieselbe und wird beim Neukompilieren jeweils aktualisiert), brauchen Sie hier einfach wieder nur auf „Program“ klicken.

Eeprom-Dateien und Bascom:

Wenn Sie eine Eeprom-Datei aus Bascom heraus, dann ist dieser Datenbereich im Programm mit „\$eeprom“ gekennzeichnet (und wird mit „\$data“ beendet). Bascom erzeugt beim Kompilervorgang eine Datei Namens „*Programmname.eep*“ – diese beinhaltet die Eeprom-Daten. Dies ist jedoch eine Binärdatei und keine Datei im Hex-Format. Da OSP II jedoch nur mit Hex-Dateien arbeitet, müssen Sie Bascom vorher mitteilen, dass die Eeprom-Datei (.eep) als Hex-Datei angelegt werden soll. Dazu fügen Sie bitte im Quellcode hinter die Zeile „\$eeprom“ noch eine neue Zeile mit dem Kommando „\$eepromhex“ hinzu. Nach der Kompilierung lässt sich die so erzeugte eep-Datei auch mit OSP II in das Eeprom schreiben.

Tipp: Wenn Sie die Fuse „preserve Eeprom memory“ aktivieren (dies bedeutet: Eeprom bei der Neuprogrammierung nicht löschen – so liefern wir unsere Controller übrigens auch aus), dann brauchen Sie das Eeprom immer nur dann neu beschreiben, wenn sich etwas geändert hat (in der Regel also nie oder selten).

Sie können OSP II auch als externen Programmer in Bascom eintragen (Options - Programmer, dann unter „Programmer“ den „external Programmer“ auswählen.

Zwei Dinge sind hilfreich zu wissen:

- 1) Die möglichen Parameter erfahren Sie in OSP II unter „Help“
- 2) Wenn OSP II dann nach dem Programmiervorgang nicht geschlossen wurde, dann wird durch erneutes Anwählen von „Run Programmer“ in Bascom versucht, eine neue Instanz von OSP II zu starten – diese könnten sich jedoch gegenseitig stören.

(c) 2005 www.display3000.com