

AVR-ChipBasic8: Bedienungsanleitung

V0.50 (c) 2006-2009 Jörg Wolfram



1 Rechtliches

Das Programm unterliegt der GPL (GNU General Public Licence) Version 3 oder höher, jede Nutzung der Software/Informationen nonkonform zur GPL oder ausserhalb des Geltungsbereiches der GPL ist untersagt!

Die Veröffentlichung dieses Projekts erfolgt in der Hoffnung, daß es Ihnen von Nutzen sein wird, aber OHNE IRGEND-EINE GARANTIE, auch ohne die implizite Garantie der MARKTREIFE oder der VERWENDBARKEIT FÜR EINEN BESTIMMTEN ZWECK.

Alle im Text genannten Marken sind Eigentum des entsprechenden Inhabers.

2 Features

Als Eingabegerät dient eine normale PS2-Computertastatur, als Ausgabegerät ein Fernsehgerät mit SCART- oder BAS-Eingang (B/W).

- 20 Programmzeilen mit maximal 25 nutzbaren Zeichen, Fullscreen-Editor
- 1 Programm im internen Flash
- Tiny-Basic ähnliche Programmiersprache
- 26 Variablen zu jeweils 1 Byte
- 30x23 Zeichen normal und invertiert, Pseudografik
- Schnelles Scrolling
- PS2-Tastatur zur Eingabe
- 1-Kanalige Audioausgabe (Noten in 1 Klangfarbe, Rauschen) mit Hüllkurve
- serielle RS232-Schnittstelle mit 1200 Baud und optionaler Ladungspumpe
- 4 Portpins als I/O und Analogeingänge nutzbar
- Schnittstelle für EEPROM (24C16) mit 4 Programmen
- Up/Download von Programmen über die serielle Schnittstelle
- automatischer Start des Programmes nach Einschalten
- Tastenkombinationen für Abbruch, Neustart, Screenshot
- Versionen für Mega8, Mega16 auf ChipBasic Board und Mega88

3 Autostart

Das System ist mit einem simplen Autostart-Mechanismus ausgestattet. Um in den normalen Mode zu kommen ist es notwendig eine Taste (ausgenommen Shift, CTRL und ALT) zu betätigen, während der Intro-Screen angezeigt wird. Wird keine Taste betätigt oder ist keine Tastatur angeschlossen, startet das Programm automatisch. Damit ist es auch möglich, einen Chip mit fertigem Programm ohne Tastatur einzusetzen.

4 Die Tastatur

Die vorliegende Version benötigt je nach Assembleroption eine Tastatur mit deutschem oder US-Layout. Bei Bedarf lässt sich das Programm auch an andere Tastenbelegungen anpassen.

Zur einfacheren Bedienung gibt es am unteren Bildrand 4 Felder mit Funktionen, die den Tasten **F1...F4** entsprechen. Daneben gibt es noch ein paar Tastenkombinationen, die zum Teil auch vom PC her bekannt sein dürften:

- CTRL+ALT+DEL startet komplett neu, dabei werden alle Variablen und ungespeicherte Programme gelöscht
- CTRL+C unterbricht das Programm zum nächstmöglichen Zeitpunkt
- CTRL+P sendet einen Screenshot an die serielle Schnittstelle

Weiterhin gibt es zu beachten, dass der Klammeraffe nur mittels **SHIFT+3** erreichbar ist.

5 Der Editor

A screenshot of a text editor window titled "Program: Hello". The editor shows a program with the following code:

```
END in 21:1
01 ? 010,9:"Hello World!"
02 VID 0:SV 10
03 VID 1:SV 10
04 INP K
05 ? K
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
```

The cursor is at the end of line 20. At the bottom of the window, there are four buttons labeled "Load", "Name", "Disk", and "Run".

Die Programme werden mit einem einfachen Fullscreen-Editor geschrieben und bearbeitet. Ganz oben steht der Programmname. Darunter befindet sich eine Statuszeile, in der z.B. Fehler angezeigt werden. Normalerweise lädt der Editor beim Start das Programm aus dem Flash, ist jedoch beim Start die **ALT** Taste gedrückt, wird der Editor mit einem leeren Programm gestartet.

Die Position des Cursors wird durch ein invers blinkendes Zeichen dargestellt. Es wird immer eingefügt, Zeichen unter/rechts vom Cursor wandern nach rechts. In der untersten Zeile sind 4 Felder zu sehen. Diese entsprechen den Tasten **F1...F4** und haben folgende Funktionen:

F1	F2	F3	F4
Load	Name	Disk	Run
Programm aus dem Flash neu in den Editor laden	Änderung des Programmnamens	Zeigt die Programmselektorbox an, wenn EEPROM angesteckt	Speichert und startet das aktuelle Programm

Die Taste F1 hat eine Zweitfunktion, wenn zusätzlich die **CTRL** oder **STRG** Taste gedrückt wird. Dann kann ein Programm über die serielle Schnittstelle eingelesen werden. Ausserdem haben die folgenden Tasten / Tastenkombinationen spezielle Funktionen:

DEL	Löscht das Zeichen unter dem Cursor, von rechts rücken Zeichen nach.
Backspace	Löscht Zeichen links neben dem Cursor, von rechts rücken Zeichen nach.
ENTER	Es wird an den Anfang der nächsten Zeile gesprungen.
ALT+DEL	Die aktuelle Zeile wird gelöscht bzw. mit Leerzeichen gefüllt.
F11	Gibt das aktuelle Programm auf die serielle Schnittstelle aus.



Wenn die Programmselektorbox angezeigt wird, kann das zu lesende oder zu schreibende Programm mit den Cursortasten ausgewählt werden. Mit der Taste **L** wird das Programm in den Editor geladen, mit der Taste **S** wird das im editor befindliche Programm im EEPROM gespeichert. Dies dauert allerdings 20-30 Sekunden da nur Byteweise geschrieben wird. Mit **ESC** wird ohne Laden/Speichern zum Editor zurückgekehrt.

6 Serieller Programmtransfer

6.1 Kommunikationsparameter

Für die Funktionen der seriellen Schnittstelle sollte am PC folgendes eingestellt werden

8 Bit — 1200 Baud — no Parity — 1 Stopp-Bit

Für alle Funktionen wird normale Textübertragung verwendet. Unter Windows kann man z.B. das Hyperterminal verwenden (Text senden/aufzeichnen) verwenden, unter Linux minicom oder das Programm **chiptrans8.pl** im Examples-Ordner.

6.2 Senden zum AVR

1. **LOAD** (CTRL+F1) im Editor auswählen
2. Transfer am PC mit **./chiptrans8.pl -w Dateiname** starten
3. am AVR wird das Programm, am PC der Status angezeigt

6.3 Empfangen vom AVR

1. Transfer am PC mit **./chiptrans8.pl -r Dateiname** starten
2. **SEND**(F11) im EDITOR betätigen
3. am PC wird der Status angezeigt

6.4 Nutzung der Screenshot-Funktion

Für den Screenshot kann das Programm **screenshot.pl** im tools-Verzeichnis benutzt werden. Es benötigt Einen Perl-Interpreter sowie ImageMagick und wird mit „./screenshot8.pl dateiname.endung“ aufgerufen. Danach kann dann der Screenshot vom Controllerboard mittels **CTRL+P** ausgelöst werden. Die Schnittstelle ist im Programm festgelegt (/dev/ttyS0), kann aber leicht geändert werden.

6.5 Zeilenenden

Ab Version 0.50 ist es möglich, die Art der zu sendenden Zeilenenden zu konfigurieren. Dazu gibt es den Befehl **CFG n**. Von N werden dabei derzeit nur die untersten beiden Bits ausgewertet.

Wert (binär)	Beispiel	Zeilenende	PC-Plattform
00	0	kein Zeilenende	für Testzwecke
01	1	nur CR	MAC
10	2	nur LF	Unix/Linux
11	3	CR und LF	Windows

Die Einstellung wird im EEPROM gespeichert und braucht normalerweise nur einmal durchgeführt zu werden.