Ausführliches Manual zum Microchip PIC-Programmer Version 7.33

Dokument enversion 2.0

Steven Wetzel, M.Sc.

10. Februar 2006

Inhaltsverzeichnis

1	1 Hardware	2
	1.1 Der Programmer	
2	2 Software	3
	2.1 IC-Prog	
	$2.1.1 \text{Installation} \dots \dots \dots \dots \dots \dots \dots \dots \dots $	
	2.1.2 Konfiguration	
	2.2 FPP	5
	2.2.1 Die Konfiguration	6
	2.2.2 Definition der Pins am Port	7
	2.2.3 Programmieren mit dem FPP	7
3	3 Software für die Programmierung	8
	3.1 Assembler	
	3.2 C	
	3.3 BASIC	
	3.4 Programmiersoftware	
4	4 Troubleshooting	11
	4.1 Vorbemerkung	
	4.2 Der Hardwaretest	

Hardware

1.1 Der Programmer

Wie die Stromversorgung anzuschließen ist, kann auf der Platine abgelesen werden. Die Versorgungsspannung kann im Bereich von 16V...18V liegen.

Der Controller wird so eingesetzt, dass die kerbe in Abbildung 1.1 nach rechts zeigt, also entgegengesetzt zum Stecker.



Abbildung 1.1: PIC-Programmer

J6 ist ein Jumper, der im Normalfall gesteckt bleibt. Wird MCLR als Input programmiert, kann der PIC nicht wieder gelöscht werden, da über diesen Pin die Programmierspannung angelegt wird. Um den PIC dennoch zu löschen, muss die Programmierspannung vor der Versorgung anliegen. Dazu wird der Jumper gezogen und die Spannung hinzugeschalten. Diese Funktion ist noch experimentell und kann deshalb nicht garantiert werden.

14-pol. PICs wie z.B. der 16F630, werden wie 18 pol, Controller behandelt, d.h. Pin 1 vom 14-pol. PIC ist an der selben Stelle wie beim 18-pol.

Software

2.1 IC-Prog

2.1.1 Installation

Zur Ansteuerung des Programmers existiert zahlreiche Freeware, IC-Prog und FPP. Das wohl vielfältigste Programm ist der IC-Prog.

Wird eines der NT-Derivate benutzt, muss der Treiber eingebunden werden. Nach dem Start meldet sich IC-Prog mit einer Fehlermeldung.



Abbildung 2.1: Fehlermeldung beim Start ohne Treiber

Der Treiber kann auf derselben Seite geladen werden. Er muss sich im selben Verzeichnis wie das Hauptprogramm befinden.

Unter Settings \rightarrow Options \rightarrow Misc kann mit Enable NT/2000/XP Driver der Treiber eingebunden werden. IC-Prog startet neu und der Treiber wird installiert.

0	otions						X
	Confirmation	Notification	n rd	I²C Langi	Progr uage	amming Shell	Shortcuts Misc
Options : Process Priority Normal High Realtime		vl ibi)					
					ок		<u>C</u> ancel

Abbildung 2.2: Installation des Treibers

2.1.2 Konfiguration

Der Programmer wird unter **Settings** \rightarrow **Hardware**(F3) konfiguriert. Einfach COM-Port einstellen, der Rest bleibt. Mit Delay wird die Zeit zwischen den Zyklen eingestellt.Diese kann bei diesem Programmer auf 10 bleiben.

lardware settings		
Programmer:	Interface © Direct I/O © Windows API	
Ports	Communication	
Com 1	🔲 Invert Data Out	
C Com 2	🔲 Invert Data In	
C Com 3	Invert Clock	
C Com 4	Invert MCLR	
Ι/Ο Delaγ (10)	Invert VCC	
	Invert VPP	
	OK <u>C</u> ancel	

Abbildung 2.3: Hardwarekonfiguration

Einige User haben gemeldet, dass der Programmer bei ihnen nur im Windows API Mode funktioniert. Ein Blick auf die Hilfe von IC-Prog verrät, dass dieser für Windows NT verwendet werden soll. Tests haben bestätigt, dass unter Windows XP der Programmer im Direct I/O Modus läuft, unter Windows 2000 dagegen im Windows API Modus.

Im Hauptmenü wird der Controllertyp sowie die Fuses (Config-Word) eingestellt. Nun kann der Controller programmiert werden. Einige Controller unterstützen kein **Verify during Program**-

🗞 IC-Prog 1.05A - Prototype Programmer	
<u>File Edit Buffer Settings Command Tools Vie</u>	ew <u>H</u> elp
🖙 • 🔒 🕼 🖀 🍫 🐝 🔇	😓 📚 🗐 🗐 🛛 PIC 16F84A. 🔽 🔗
Address - Program Code	Configuration ()
0000: 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF ÿÿÿÿÿÿÿ Oscillator:
0008: 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF YYYYYYY
0010: 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF <u>ÿÿÿÿÿÿÿ</u>
0018: 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF <u>yyyyyyy</u>
0020: 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF <u>yyyyyyy</u>
UU28: 3FFF 3FFF 3FFF 3FFF 3FFF	SFFF SFFF SFFF YYYYYYY
0030: SFFF SFFF SFFF SFFF SFFF	SFFF SFFF SFFF YYYYYYY
0040, SEFE SEFE SEFE SEFE SEFE	STIT STIT STIT YYYYYYY
0040: SFFF SFFF SFFF SFFF SFFF	SFFF SFFF SFFF WWWWWWW
0050 · SFFF SFFF SFFF SFFF SFFF	SFFF SFFF SFFF WWWWWWW
0058: 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF VVVVVVV
Address Former Date	
Address - Eeprom Data	
0000: FF FF FF FF FF FF FF	<u> </u>
UUU8: FF FF FF FF FF FF FF FF	YYYYYYY
0010: FF FF FF FF FF FF FF FF	YYYYYYY
0010; FF FF FF FF FF FF FF FF	XXXXXXXX
0020. FF FF FF FF FF FF FF FF	Checksum ID Value
0020. FF FF FF FF FF FF FF FF	
0038. FF FF FF FF FF FF FF FF	
	Config word : 3FF1h
Buffer 1 Buffer 2 Buffer 3 Buffer 4 Buffer 5	
	JDM Programmer on Com1 Device: PIC 16F84A (104)

Abbildung 2.4: Hauptmenü

ming. Das sollte dann unter Settings \rightarrow Options \rightarrow Programming ausgeschalten werden. Weiteres steht in den Datenblättern der einzelnen Controller sowie auf der Seite von IC-Prog.

2.2 FPP

Der FPP ist ein einfaches aber sehr effektives Tool für die Programmierung. Er ist sehr klein und in weiten Bereichen frei konfigurierbar. Beim FPP werden die Pins am RS232- bzw. Parallelport der Programmierung entsprechend konfiguriert. Nach dem Start erscheint das Hauptfenster.

🚔 FPP - [16F84]	
Flash PIC Programmer	-Buffer
	Load Save
Blank	Rel <u>o</u> ad <u>B</u> lank
	Device
	<u>R</u> ead <u>P</u> rogram
	Verify Erase
	✓ Code 1024 ▼
	🗹 Data 🛛 🗲 💌
Run Setup Code • Hex •	✓ ID #####
Quit About Data ASCII O	Config ####

Abbildung 2.5: FPP Hauptmenü

2.2.1 Die Konfiguration

Ein Klick auf **Setup** öffnet ein Fenster, in dem der Port sowie der Controller festgelegt werden.

FPP Setup 🔀
Hardware
Use current
on port: COM1 (03F8H) 💌
Device
16F84 💌
_ Timing
Prog cycle delay: 20 ms
Power up delay: 25 ms
1/0 delay: 6 ticks
OK Cancel <u>D</u> efine/Test

Abbildung 2.6: FPP Setup für Controller und Port

2.2.2 Definition der Pins am Port

Die Pins werden entsprechend ihrer Funktion konfiguriert. Für den vorliegenden Programmer sieht das folgendermaßen aus: Wichtig ist, dass alle Änderungen mit Apply bestätigt werden.

FPP Define/Test 🛛 🗙			
Define Hardware	Apply Test Enable Low High		
OUT 4 - CLK 7 - VDD - MCLR 3 - PGM - READ - IN 8 -	9 9 9 9 9 9 0 0 0 0 0 9 1		
<u>OK</u>	Cancel		

Abbildung 2.7: FPP Definition der Port-Pins

2.2.3 Programmieren mit dem FPP

Die Programmierung gestaltet sich relativ einfach. Hierzu muss man wissen, dass Flash- Bausteine vor der Programmierung gelöscht werden müssen. Ein Klick auf Erase (Abbildung 2.5) erledigt das. Als nächstes ist mit Load das HEX-File zu laden und wichtig, vor der Programmierung das CONFIG-Word einzutragen. Wie das Config-Word aufgebaut ist, steht in den Datenblättern. Mit **Program** wird der Controller dann programmiert.

Software für die Programmierung

3.1 Assembler

Microchip selber bietet mit MPLAB eine ausgereifte IDE an. Diese beinhaltet einen Simulator, Debugger und Assembler. Ab Version 6 wird Syntax Highlighting unterstützt. Unter der Adresse http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1475&category=devSoftware kann die Software geladen werden.

🙀 MPASM v02.70 - Microchip Technology, Inc.						
Source File Name:						
	Browse					
Options:	Options: MICROCHIP					
Radix: © Default © Hexadecimal © Decimal © Octal	Warning Level: © Default © All Messages © Warnings and Errors © Errors Only	Hex Output © Default © INHX8M © INHX8S © INHX32	Generated Files:			
I Case Sensitive	© Default © On © Off	Processor: Tab Size:	Default			
Extra Options:						
★ Exit ✓ Assemble ✓ Save Settings on Exit ? Help						
Override source file macro expansion.						

Abbildung 3.1: Microchip Assembler

3.2 C

Als C-Compiler stehen neben den gcc-Tools verschiedene kommerzielle und Shareware- Programme zur Verfügung. Hier eine kleine Auflistung:

- PICC http://www.htsoft.com/downloads/demos.php
- C2C http://www.picant.com/c2c/c.html
- CCS http://www.ccsinfo.com/picc-referall.shtml
- GNUPIC http://www.gnupic.org/
- und natürlich die Microchip-eigenen C-Compiler

Der vermutlich geeignetste Compiler ist der PICC von HT-Soft. Dieser hochoptimierende Compiler steht für die 1k-PIC-Controller frei zur Verfügung (PICClite). Er lässt sich sehr einfach in MPLAB integrieren. Erfahrungen haben hier gezeigt, dass der älteren Version von MPLAB (5.70) der Vorzug gegeben werden sollte, da die Befehlslänge der 6.x-Version mit 63 Zeichen zu gering ist und somit das Projekt im Root-Verzeichnis liegen muss. Wenn man auf Syntax-Highlighting verzichten kann, ist die 5.7-Version gut geeignet.

3.3 BASIC

Auch BASIC-Compiler für den PIC gibt es.

- PicBASIC http://www.melabs.com/products/pbc.htm
- PROTON BASIC http://www.picbasic.org/about.php
- BASIC18 http://www.midwest-software.com/Basic18/basic18.htm

3.4 Programmiersoftware

Und zum Schluss noch die Links der Programmiersoftware.

- IC-Prog http://www.ic-prog.com/
- FPP http://people.man.ac.uk/ mbhstdj/piclinks.html
- WinPIC http://www.qsl.net/dl4yhf/winpicpr.html

Troubleshooting

4.1 Vorbemerkung

Der Programmer wird immer getestet ausgeliefert. Es kann jedoch trotzdem vorkommen, dass es zu Problemen mit der Arbeitsweise kommt. Die Funktionsweise des programmers lässt sich sehr einfach testen. Alles was man dazu benötigt, ist ein Voltmeter. Vor dem Test müssen folgende Punkte sichergestellt sein:

- richtiger COM-Port gewählt (kann im Hardware-Test getestet werden)
- Treiber installiert (es hat sich gezeigt, dass es in seltenen Fällen zu Störungen des Treibers kommt. Dies äußert sich oft darin, dass beim Auslesen die erste Zeile fehlt.)
- korrektes Kabel gewählt, also eine 1:1-Verlängerung, kein Null-Modemkabel

4.2 Der Hardwaretest

Der Hardwaretest kann unter dem Menüpunkt Settings (Hardware Check).

Hardware Check	×
Signals Out	Signals In
Enable Data Out	🗖 Data In
Enable Clock	
Enable MCLR	
Enable VCC	
Enable VPP	
	ок

Abbildung 4.1: Hardware Check

Der Programmer wird eingeschalten, es befindet sich kein PIC-Controller im Slot. Alle Signale können am 6pol. Header für die ICSP abgegriffen werden.

- Enable MCLR anklicken. Das schaltet die Programmierspannung ein.
 - Vpp geht von 0 V auf ca. 13.5 V
- Enable MCLR aus, Enable Clock an
 - Clk (Clock) geht von 0 V auf 5 V
- Enable Clock aus, Enable Data Out an
 - Data geht von 0 V auf 5 V

Sollten die Signale nicht messbar sein, sollten die Test noch einmal in einem anderen Modus (Direct I/O oder Windows API) durchgeführt werden. Es hat sich gezeigt, dass unter WIndows2000 nur der Windows-API-Modus funktioniert, unter den Win95/98/ME/XP-Systeme dagegen der DirectIO-Modus. WinPIC funktioniert nur mit letzteren Systemen.