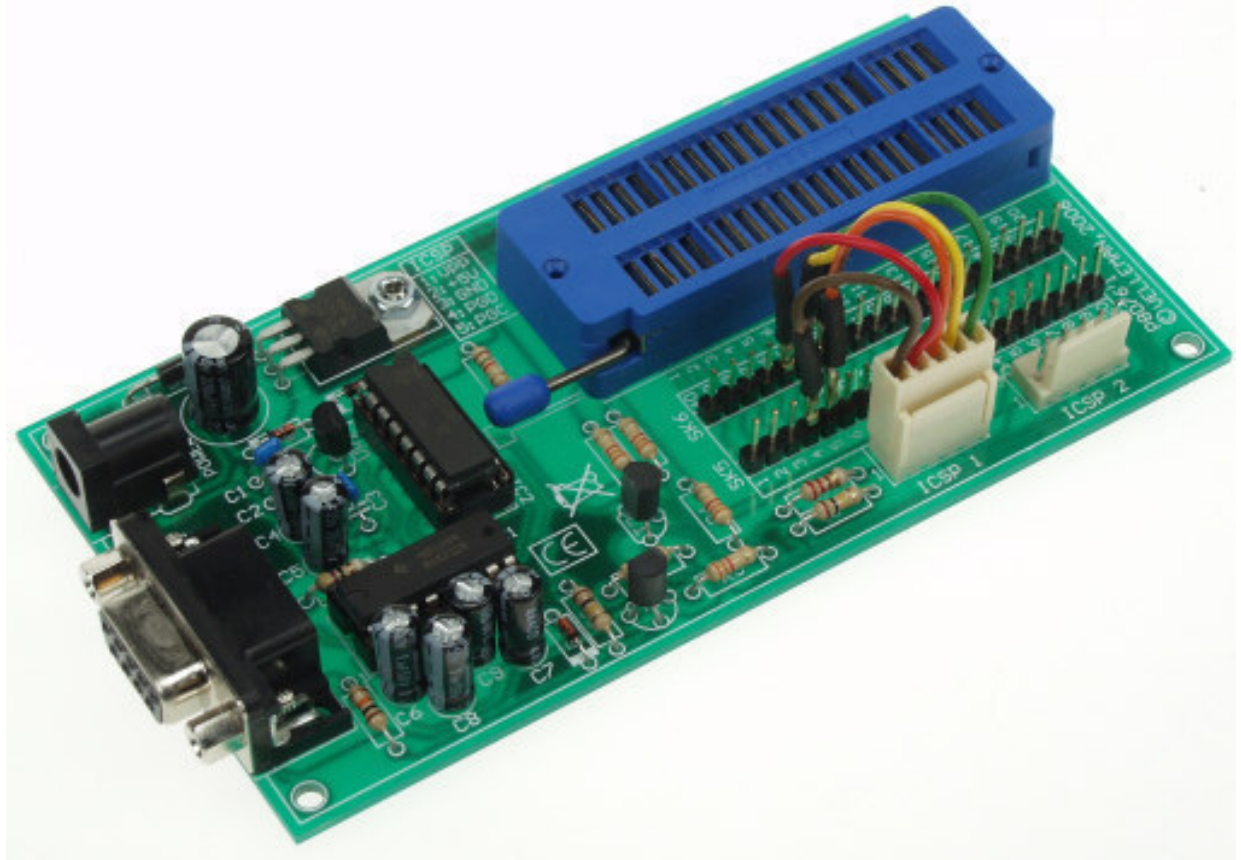




VM134
(K8076)

QUICK GUIDE

1 Allgemeine Information

1.1 Einführung

Wir bedanken uns für den Kauf der Velleman-Produkte. (Bausatz-Version = K8076) VM134 ist ein multifunktionaler und erzieherischer Programmierer für das Programmieren einer Auswahl von [Microchip® PIC™ FLASH microcontrollers](http://www.microchip.com). Diese FLASH-Controller können viele Male erneut programmiert werden, wodurch sie sehr geeignet sind für das Lehren einer PIC-Programmiersprache. Ein anderer Vorteil der reprogrammierbaren Controller ist, dass die Software eines Gerätes, in dem dieser Controller eingebaut ist, einfach verbessert werden kann.

Der VM134 PIC Programmer ist die zusammengebaute Version des Programmierers K8076. Deshalb kann der VM134 in dieser Bedienungsanleitung und in der Software auch K8076 genannt werden.

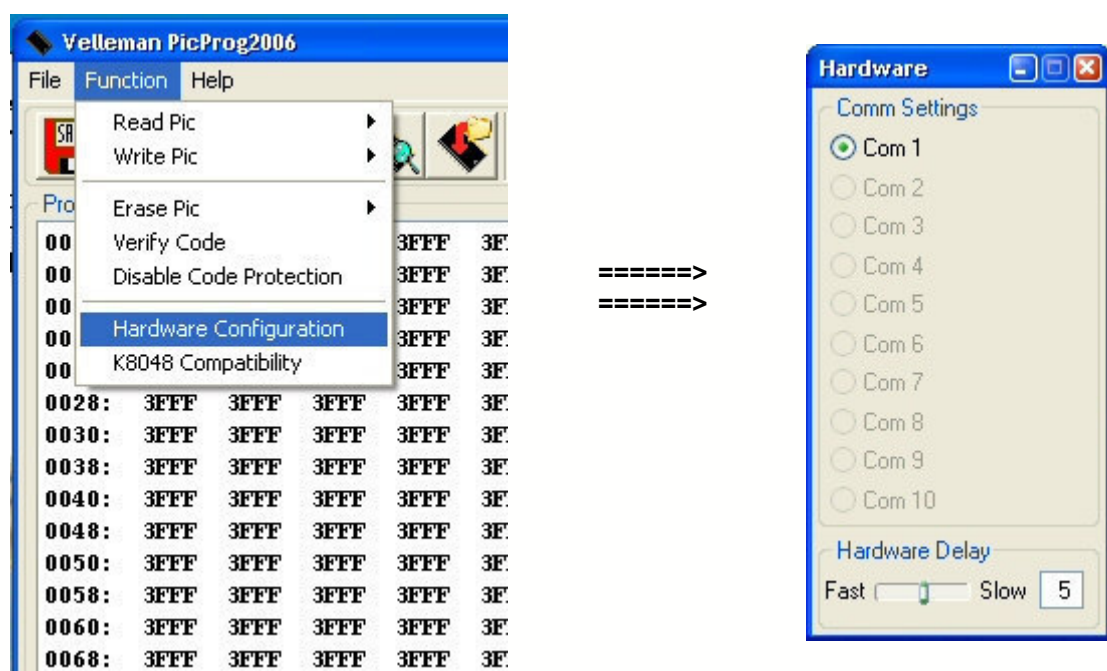
Es gibt zwei wichtige Phasen bei der Anwendung dieses Programmiers. Zuerst muss der Programmcode geschrieben werden. Meistens geschieht das in einer graphischen Umgebung (IDE). Für die PICs ist das MPLAB(™) von Microchip selbst. Sie können dieses Softwarepaket kostenlos auf www.microchip.com herunterladen. Es kann auch einfacher: Schreiben sie das Programm in einem ASCII-Textverarbeitungsprogramm wie z.B. Notepad, das bei jeder Windowsinstallation auf Ihrem PC installiert wird.

Informationen über die Befehle, die in der Assemblersprache verwendet werden können Sie für jeden Typ auf der Microchip-Website finden. Alle Informationen stehen in den Datenblättern. Ein Beispielsprogramm wird auf dieser Cd mitgeliefert.

1.2 Verbindungskabel

Der Anschluss des Programmiers geschieht an einen freien seriellen RS232-Port Ihres Computers. Dieser serielle Port muss 100% IBM kompatibel sein, was bedeutet, dass er über einen 16550 kompatiblen UART verfügen muss. Der Bausatz selbst verwendet kein RS232-Protokoll, sondern ein emuliertes I2C-Protokoll über "handshaking" Linien.

Ein **UART**, *universal asynchronous receiver / transmitter* ist verantwortlich für die Ausführung der Hauptaufgaben bei der seriellen Kommunikation mit Computern. Der Chip wandelt eingehende Parallelinformation in serielle Daten, die über eine Kommunikationslinie verschickt werden können, um. Ein zweiter **UART** wird verwendet um die Informationen wieder zu empfangen. Der **UART** erledigt alle Aufgaben wie Timing, Paritätskontrolle, usw. die für die Kommunikation erforderlich sind. Die einzigen erforderlichen Chips sind die Line Treiber, die die Signale von TTL Niveau in Line Spannungen umwandeln und umgekehrt.



1.3 Anschluss und Test

- Sorgen Sie dafür, dass sich kein Controller in der ZIF-Fassung befindet.
- Schließen Sie den seriellen Steckverbinder an den seriellen Port an.
- Schließen Sie eine Stromversorgung von 15VDC an. Diese Spannung darf stabilisiert sein, aber es ist nicht unbedingt notwendig, denn die Leiterplatte hat einen Spannungsregler (12VDC unstabiliertes Netzgerät funktioniert in vielen Fällen auch weil die Klemmspannung beträgt immer ungefähr 15 bis 16V).
- Beim Einschalten der Stromversorgung sollte die LD1 aufleuchten. Sie ist die "Power Good" LED. Diese LED zeigt, dass der Programmer unter Spannung steht und dass der Controller mit +5V versorgt wird.



- Starten Sie jetzt die "PICprog2006" Software und klicken Sie das Piktogramm rechts oben "Hardware Connections" (**Abb.1.0**).



FIG. 1.0

- Wenn Sie mit der Maus auf LEDs LD3, LD2 und LD4 klicken, müssen Sie diese Leiterplatte aufleuchten lassen (**Abb.2.0**)

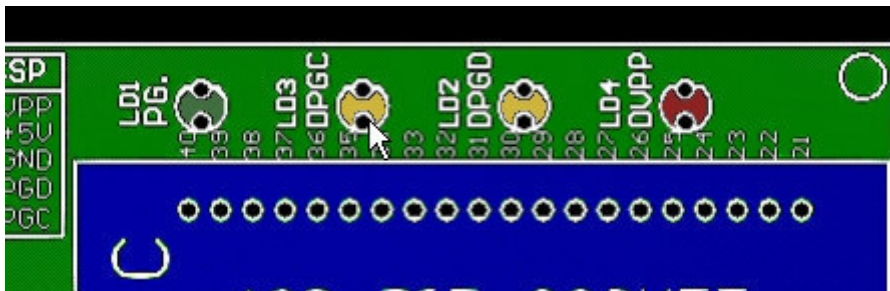
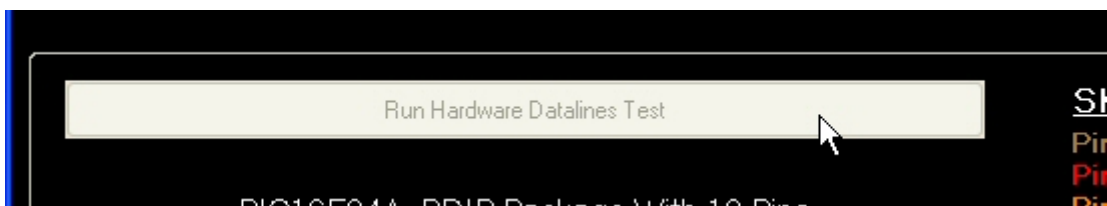


FIG. 2.0

- Wenn Sie "Run hardware datalines test" drücken, starten Sie eine automatische LED-Sequenz. Diese LEDs muss synchron mit dem Schirm aufleuchten (**Abb. 3.0**). Klicken Sie erneut diese Taste um den Test zu stoppen. Achtung ! Während dieser Tests darf sich kein PIC in der Fassung befinden !

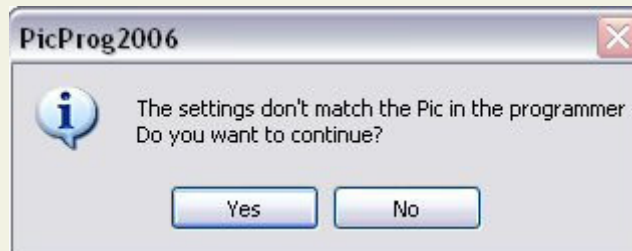


- Wenn die Tests gut verlaufen sind, sind Sie mit dem Test des Programmers fertig und können Sie mit dem ersten Projekt anfangen. Wenn Sie nicht gut verlaufen sind, müssen Sie nach einem Hardware-Fehler auf der Leiterplatte oder der Nicht-Kompatibilität mit dem Computer suchen. Wenn Sie das nicht machen, sind die Chancen gut, dass Sie die PIC-Controller permanent beschädigen.

ACHTUNG: Wenn es ein Kommunikationsproblem zwischen dem PC und dem VM134 gibt, oder es ergibt sich ein ernsthaftes Hardwareproblem mit dem VM134, dann wird man nicht auf die LEDs oder die Testtaste klicken können. Auch kann die folgende Mitteilung erscheinen:



Wenn es eine Verbindung zwischen dem VM134 und dem PIC Controller im Sockel des Programmierers gibt, erscheint nachfolgende Meldung.



1.4 Haftungsausschluss

Velleman Components NV und der Softwarehersteller sind nicht haftbar für das Versagen der Hardware und/oder Software oder den verursachten Schaden.

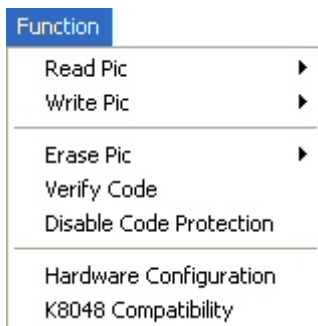
2 Menüleiste

2.1 File

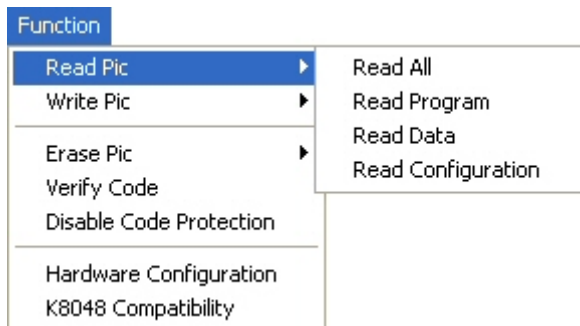


"Load File" : Hex-Datei laden
 "Save File" : Hex-Datei speichern
 "End" : Programmende

2.2 Function

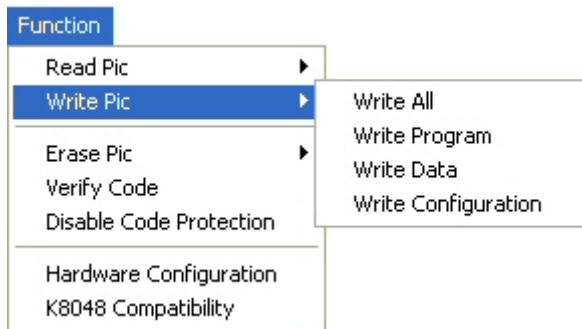


"Read PIC"



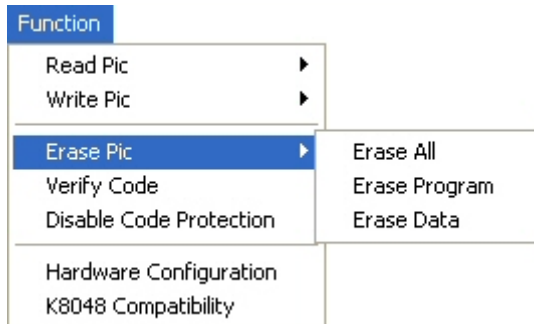
"Read All" : Alle verfügbaren Daten lesen
 "Read Program" : Nur Programmdateien lesen
 "Read Data" : Nur EEPROM-Daten lesen
 "Read Configuration" : Die Konfigurationsbits lesen

"Write PIC"



"Write All" : Alle verfügbaren Daten schreiben
 "Write Program" : Programmdateien schreiben
 "Write Data" : EEPROM-Daten schreiben
 "Write Configuration" : Konfiguration schreiben

"Erase PIC"



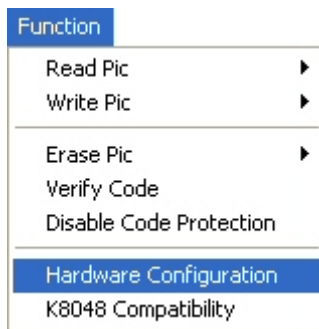
"Erase All" : Alle verfügbaren Daten löschen
 "Erase Program" : Nur Programmdateien löschen
 "Erase Data" : Nur EEPROM-Daten löschen

"Disable Code Protection"



Machen Sie den PIC wieder verfügbar nachdem er mit einem Leseschutz gesichert war, alle Daten werden gelöscht

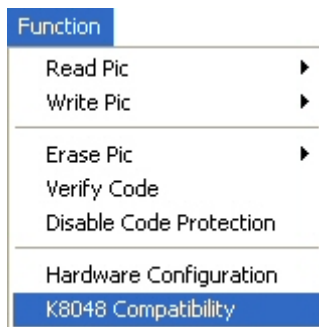
"Hardware Configuratie"



"Comm Settings" : Alle verfügbaren RS232-Ports selektieren lassen

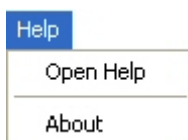
"Hardware Delay" : wenn die Kommunikationsgeschwindigkeit zu niedrig ist.

"K8048 Compatibility"



Ermöglicht es (wo möglich), K8048 zu verwenden. Nicht bei allen PICs möglich.

2.3 Hilfe-Menü

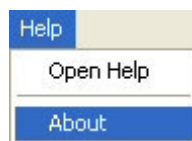


"Open Help"



So können Sie die Hilfe-Datei dieses Bausatzes heranziehen

"About"



Zum Abrufen der Programmversion



3 Tastenleiste

3.1 Tastenleiste



1. HEX-Datei speichern.
Speichern Sie den Inhalt eines gelesenen Controllers als INHX8M-Datei auf der Festplatte. Achtung: Controller, deren "code protect"-Bit aktiviert wurde, können nicht gelesen werden. Das wird meistens von den Herstellern zum Copyrightschutz gemacht. Hauptspeicherauszug: HEX-Dateien werden als 18Fxxx-Familie werden als INHX32-Datei extern gespeichert.
2. HEX-Datei laden
Liest eine Datei von einem Speichermedium in den Speicher der Software ein. Achtung: diese Datei soll ein INHX8M, INHX16 oder INHX32 Format haben. Der Compiler (z.B. MPASM) muss so eingestellt werden, dass er eine INHX8M-Datei generiert.
3. Mpasm-Editor laden.
Starten Sie den mitgelieferten Assembler von Microchip. Aktualisierungen finden Sie immer zusammen mit dem gesamten Compilerpack von Microchip "MPLAB" auf Ihrer Website: www.microchip.com.
4. Alle Daten zum PIC schreiben
Speichern Sie sofort die HEX-Datei auf dem Controller, der sich in der ZIF-Fassung befindet.
5. Alle Daten aus dem PIC laden
Liest den Inhalt eines Controllers und bringt ihn in Pufferspeicher der Software unter. Achtung: Controller, deren "code protect"-Bit aktiviert wurde, können nicht gelesen werden. Das wird meistens von den Herstellern zum Copyrightschutz gemacht.
6. Alle Daten unmittelbar von einer HEX-Datei zum PIC schreiben
Mit dieser Taste können Sie unmittelbar eine HEX-Datei schreiben, ohne dass Sie sie zuerst in den Pufferspeicher laden müssen.
7. Hilfedatei abfragen
Starten Sie die online Bedienungsanleitung der jetzigen PICprog2006 Software.
8. Leiste für PIC-Auswahl



Wählen Sie die gewünschte Controller-Familie. Die Einstellung der Familie und des Typs wurde aufgeteilt um die Auswahlliste zu verkürzen und zu vereinfachen.

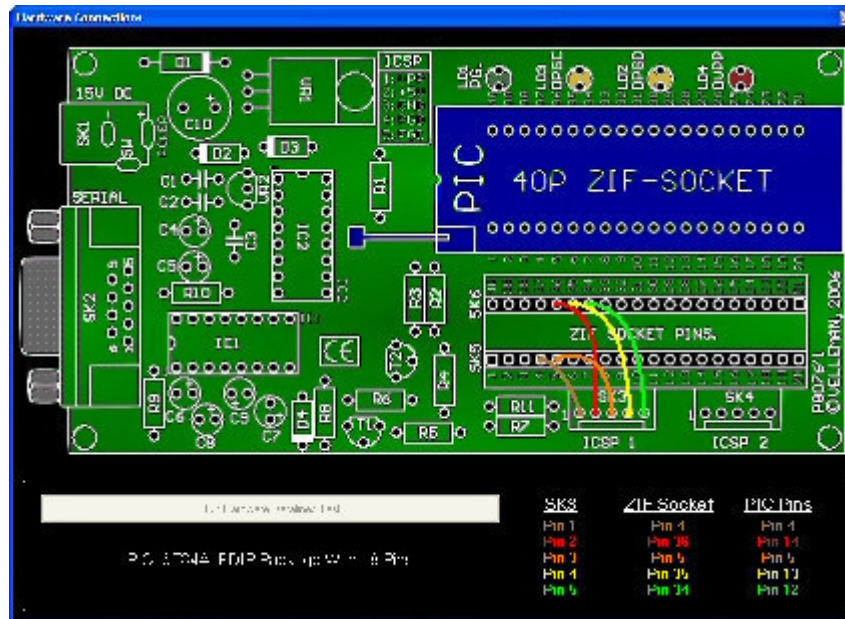


Wählen Sie hier den gewünschten Controller, der zu der in "Punkt 8" gewählten Familie gehört. Controller mit einem grünen "V"-Zeichen sind effektiv von Velleman mit diesem PIC-Programmer getestet worden, während die Controller mit einem gelben "X"-Zeichen in der jetzigen Software eingeschlossen, aber nicht mit diesem Programmer getestet. Sollten Sie Probleme haben, dann können Sie einen Controller mit Begleitschreiben an unsere Support-Abteilung der Hauptgeschäftsstelle schicken.

Wir werden alles Mögliche machen um dieses Problem zu lösen. Wir können keine Garantie bieten, denn wir sind von den von Microchip freigegebenen Protokollaten abhängig.

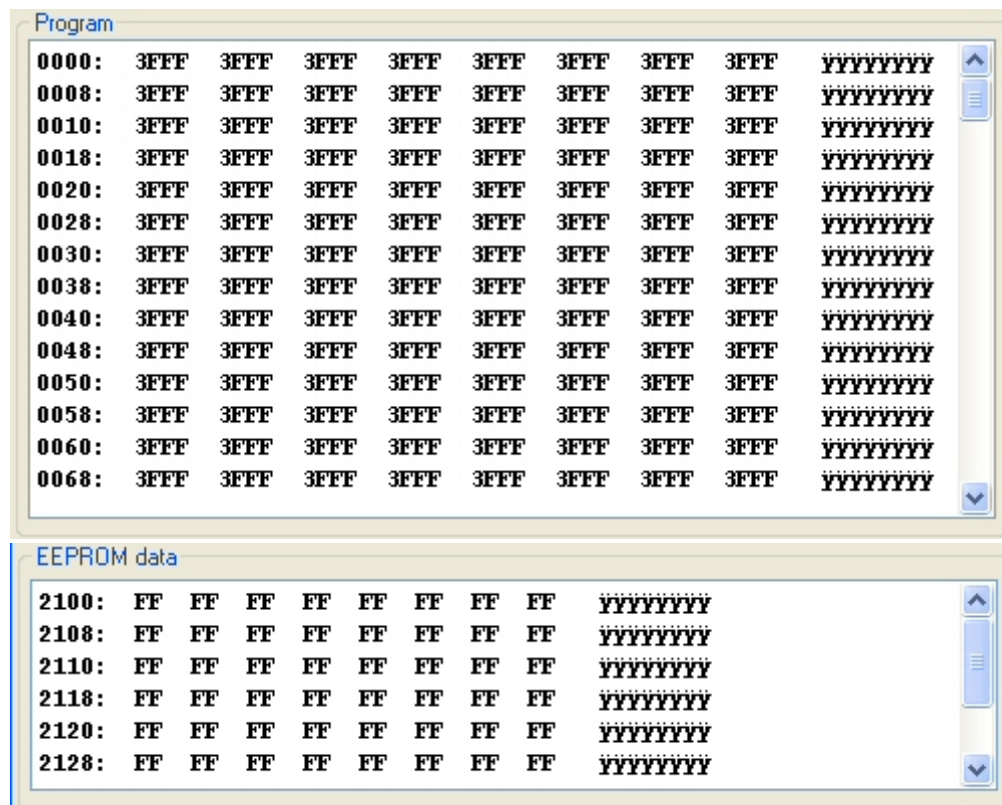
9. Hardwarekonfiguration

Klicken Sie hier für eine visuelle Darstellung der Weise, wie das PIC-kabel angeschlossen werden muss, sodass Sie den jetzt gewählten Controller programmieren können.



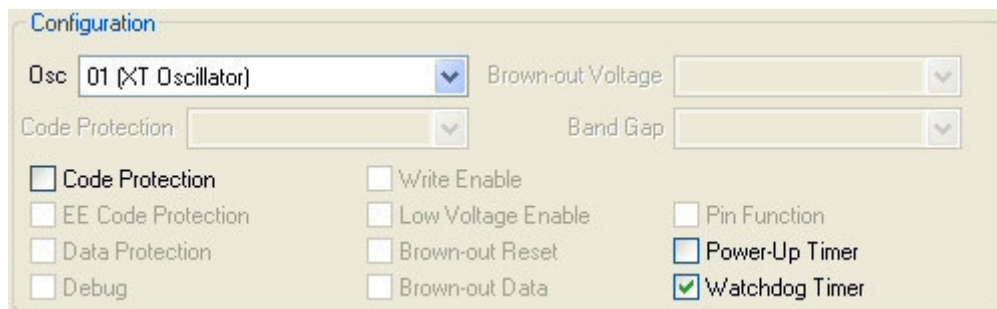
4 Windows

4.1 Programmcode

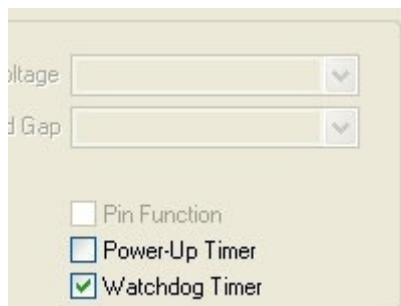


Hier haben Sie einerseits den Programmcode. Er ist ein hexadezimaler Upcode, den der Controller ausführen wird. Andererseits sehen Sie auch den Datacode. Es sind die Werte, die im EEPROM-Speicher des Controllers gespeichert sind oder werden. Dieses Fenster erscheint nur bei Controllern, die einen EEPROM-Speicher haben (z.B. PIC16F627)

4.2 Konfiguration



Hier können die Programmieroptionen nachgeschlagen und nötigenfalls auch angepasst werden. Es ist empfehlenswert, diese Einstellungen unmittelbar in dem Assemblerprogramm mittels der `__CONFIG` Compiler Anweisung auszuführen. Beispiel: ziehe "BLINKLED.ASM".



Diese sind die PIC-Controlleroptionen, die ein- und ausgeschaltet werden können. Die meisten sind auch über die `__CONFIG` Compiler Anweisung im Assemblerprogramm einzustellen. Zur weiteren Erklärung, siehe Datenblätter des verwendeten Controllers auf der Microchip-Website www.microchip.com.

5 Gebrauch

5.1 PIC wählen

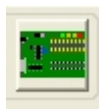
1. Wählen Sie rechts oben zuerst die richtige PIC-Familie, z.B. "PIC10F", "PIC16F",...



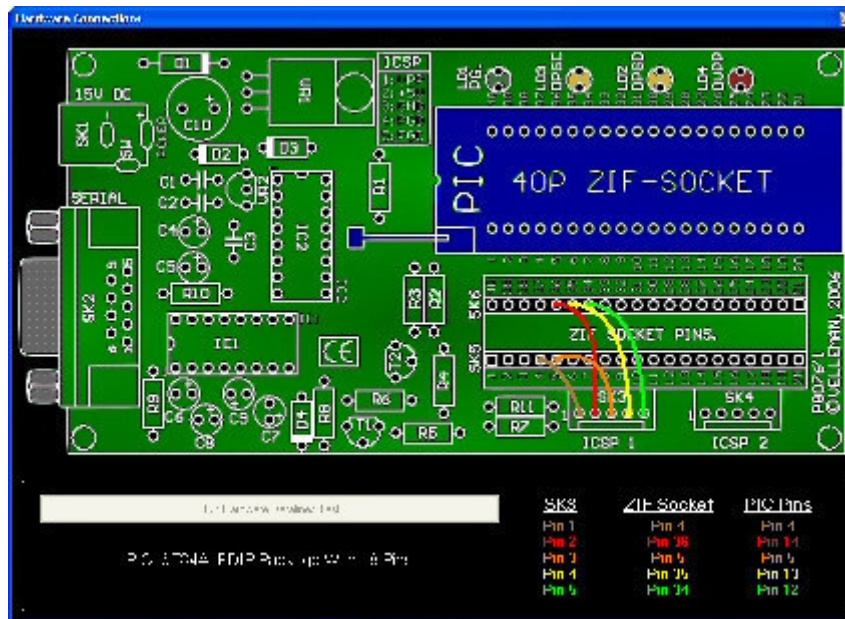
2. Wählen Sie den richtigen Typ im Menu daneben, z.B. "PIC10F200",...



3. Klicken Sie das Symbol "Hardware connections"

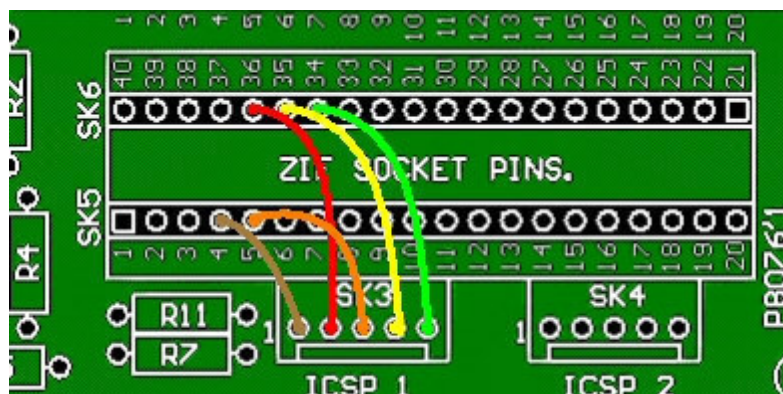


- *Sie sehen jetzt ein Foto von wie Sie das Anschlusskabel für die PIC-Konfiguration mit den Stiften der ZIF-Fassung verbinden müssen. Wenn Sie die mitgelieferten Drähte verwenden, stimmt der Farbcode dieser Drähte mit den Farben auf Ihrem Schirm.*



Anschlusskabel für die PIC-Konfiguration

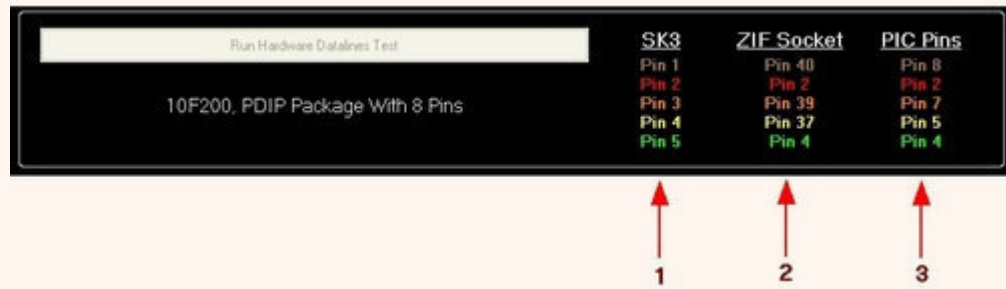
- Die Stifte des ICSP1 oder ICSP2-Steckverbinden müssen korrekt angeschlossen sein, bevor ein PIC-Controller in die ZIF-Fassung eingebracht werden darf.



- Die Steckverbinder ICSP1 und ICSP2 dürfen durcheinander verwendet werden. Diese sind, was die Hardware betrifft, gleich.

Hinweis: Die ICSP-Steckverbinder können auch verwendet werden um einen Controller, der sich extern auf einer Leiterplatte befindet, zu programmieren. Die Drähte zu dieser Leiterplatte müssen möglich kurz sein (+/- 20cm).

Hinweis: Im Foto sehen Sie unten rechts auch Informationen über wie die Verbindung der ICSP-Steckverbinder SK3 und SK4 (1) im Verhältnis zu der PIN-Belegung der ZIF-Fassung (2) und dem PIC-Controller selbst (3) steht.



- Wenn alle Verbindungen richtig hergestellt sind, darf der Controller in die ZIF-Fassung eingebracht werden und darf der Hebel nach unten gedrückt werden.

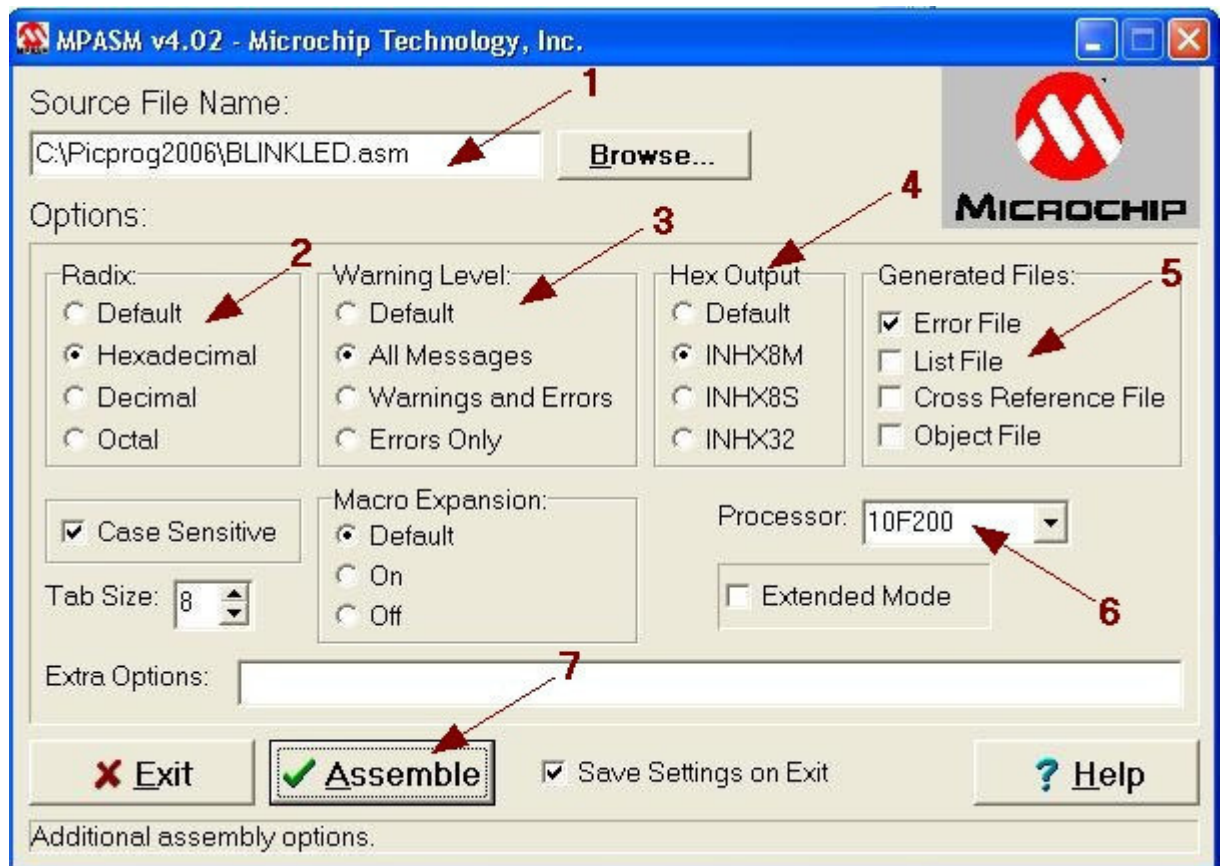
Hinweis: Wenn LD1 erlischt beim Einbringen eines PIC in die ZIF-Fassung, bedeutet das, dass es einen internen Kurzschluss in der Komponente gibt oder dass das PIC-Kabel nicht richtig angeschlossen ist und deshalb ein Kurzschluss entsteht. Der VM134 selbst ist kurzfristig vor Kurzschluss geschützt mittels R10.

5.2 PIC-Controller programmieren

Die Arbeitsweise zum Programmieren und Testen eines PIC wird unten anhand einer einfachen Beispielschaltung erklärt. Die Schaltung ist einfach eine blinkende LED. Der verwendete Controller im Beispiel ist ein PIC10F200.

STAP 1: Kompilieren des U-Codes

- Starten Sie PICprog2006
- Klicken Sie "MPASM" Symbol
- Lesen Sie die ".ASM"- Datei ein.



1. Die zu kompilierende Datei ist meistens ein ".ASM"-Typ.
2. Stellen Sie die standard anzunehmende Radix (Radix eingegeben in .ASM-Datei wird Vorrang vor dieser Einstellung haben).
3. Sie lassen am besten diese Einstellung auf "All Messages", sodass Fehlermeldungen und Warnungen in die .ERR oder .LST-Dateien aufgenommen werden können.
4. Hier können Sie das Ausgangsformat wählen. Diese Einstellung MUSS auf INHX8M, INHX16, INHX32 eingestellt sein, sonst kann die Programmiersoftware die HEX-Datei nicht richtig interpretieren.
5. Hier wählen Sie welche Dateien zur Information zusätzlich generiert werden durch MPASM, z.B. Datei mit Fehlermeldungen,...
6. Hier müssen Sie den PIC-Controllertyp, der programmiert werden muss, wählen.

* Für weitere Informationen, siehe MPASM-Hilfe oder die Dokumentation auf der Website von [Microchip](http://www.microchip.com).

- Wenn alle Einstellungen korrekt eingegeben wurden, drücken Sie die "ASSEMBLE" (7)-Taste.
- Wenn sich keine Fehler ergeben während der Zusammenstellung des Codes, werden Sie den folgenden Schirm sehen:



Bevor Sie zur Programmierung des Controllers übergehen, darf sich kein einziger Fehler mehr im Assemblercode befinden. (Fehler => 0).

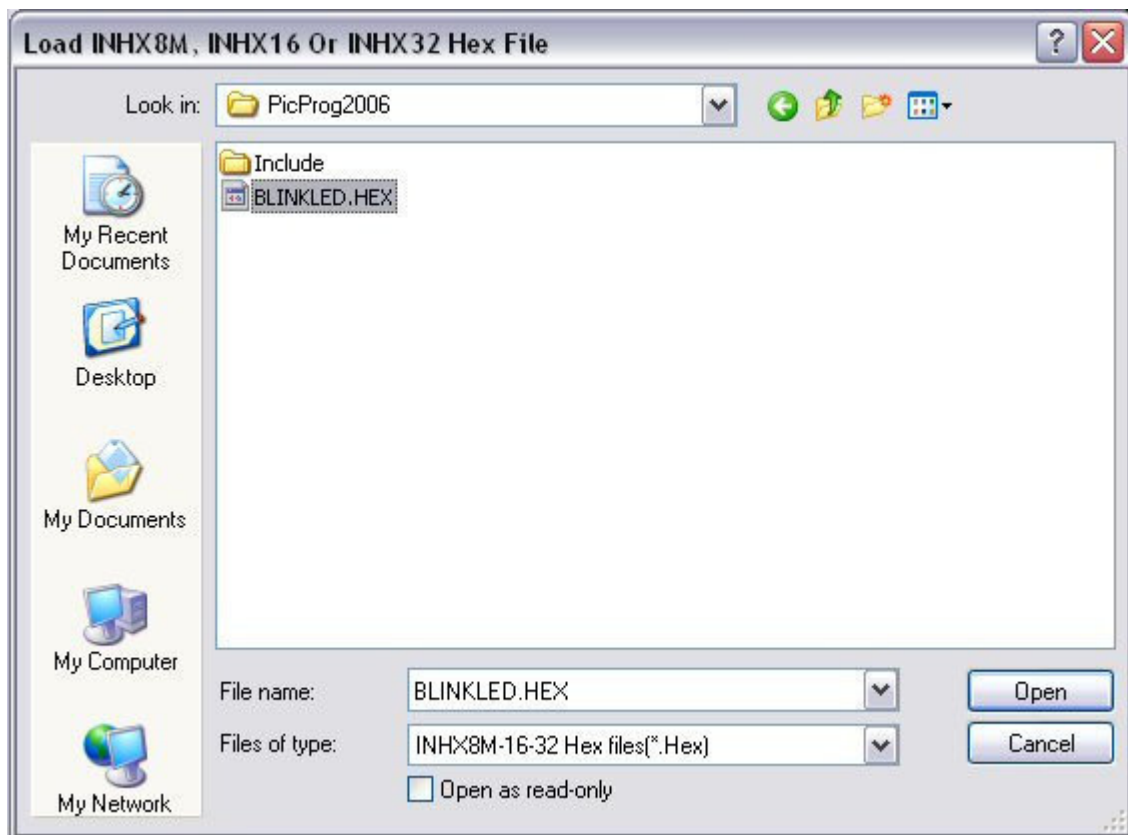
Ursachen einer Kommunikationsstörung:

- Der PIC Controller-Typ stimmt nicht mit dem ausgewählten Softwaretyp überein
- Die Stromversorgung des VM134 ist zu niedrig (15V)
- Die PIC-Auswahl durch Patchkabel ist nicht korrekt
- Der PIC Controller ist defekt
- Der PIC Controller befindet sich in einem Status, der nicht in den Programmiermodus gestellt werden kann

Bemerkung: Dieser PIC Programmer kann keine Controller programmieren die gleichzeitig den internen Oszillator und den MCLR-Pol als Eingang verwenden. Wenn Sie einen Controller so programmieren, besteht die Möglichkeit, dass Sie ihn nicht wieder verwenden können.

STAP 2: Programmieren des Controllers

- Starten Sie PICprog2006
- Klicken Sie das Piktogramm "LOAD HEX FILE". Sie werden den folgenden Schirm bekommen:



- Klicken Sie die gewünschte HEX-Datei (z.B. BLINKLED.HEX)

Sollte sich ein Hardwareproblem ergeben, dann kann die folgende Mitteilung erscheinen. Prüfen Sie den VM134 und/oder die Auswahl des Controllers.



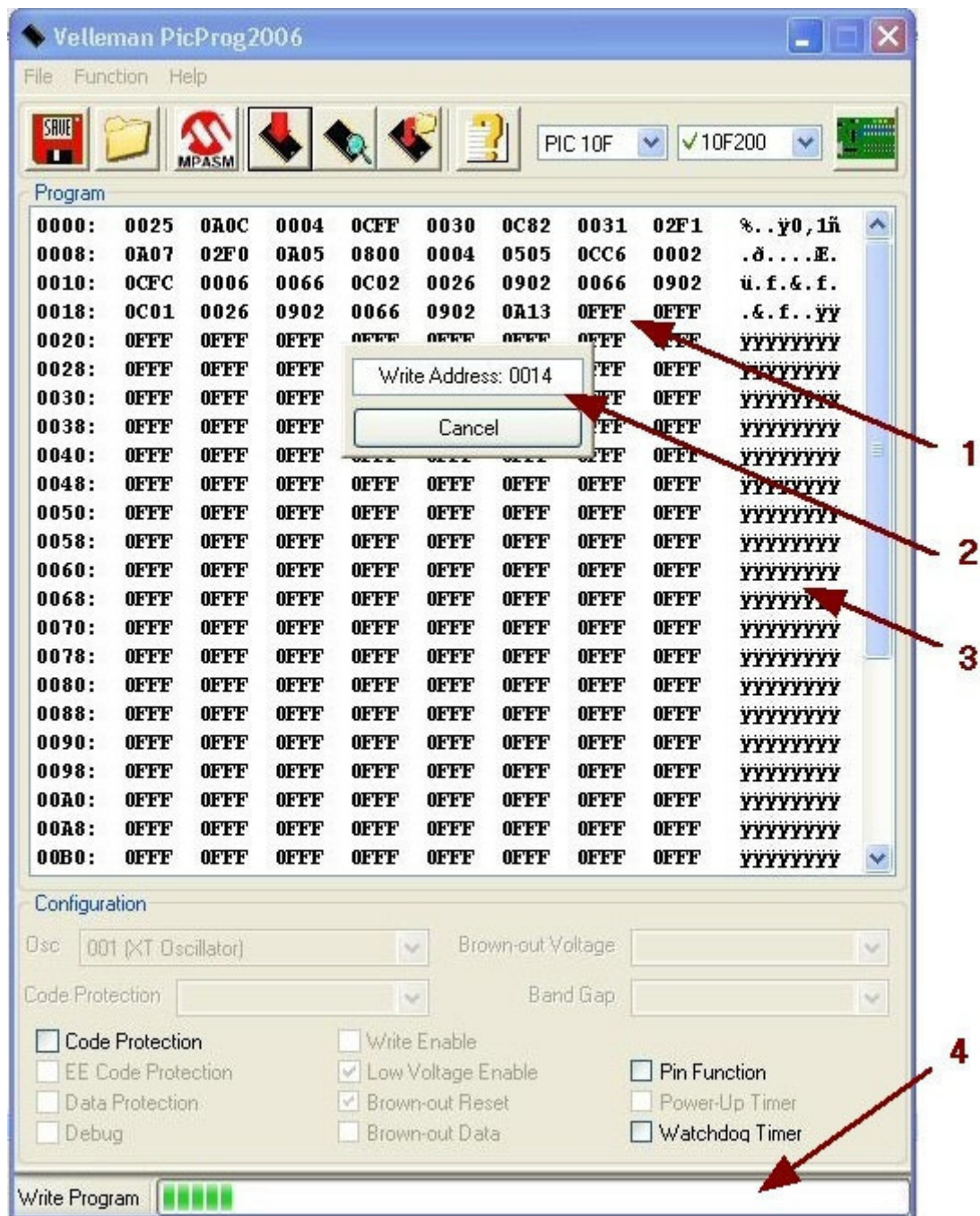
- Klicken Sie das "WRITE ALL DATA TO PIC" Piktogramm.

Das Programm wird Sie um eine Bestätigung bitten:

Klicken Sie "YES" wenn Sie sich sicher sind, dass Sie den Controller überschreiben wollen.



- Sie Sehen den Verlauf von allerhand Aktionen wie Löschen, Programmieren, Prüfen und das Einstellen verschiedener Parameter im Controller.



1. HEX-Code, oder im Controller gespeichert werden wird.
2. ASCII-Variante des Codes.
3. Adressenzähler: so wissen Sie an welcher Speicherstelle im Moment gelesen oder geschrieben wird.
4. Verlaufeiste: Hier können Sie visuelle den Prozentsatz des Programmier- oder Lehrverfahren sehen.

- Wenn die Programmierung beendet ist, kann der Hebel der ZIF-Fassung aufgehoben werden und kann der Controller entfernt werden.

Wenn während des Programmierverfahrens etwas misslingt (z.B. Verbindung wird abgebrochen, Interrupt anderer Software auf dem seriellen Port usw.), kann der Controller permanent beschädigt werden. Für mehr Information, siehe die Datenblätter des verwendeten Controllers.