

K8048 PIC PROGRAMMER BOARD

Velleman Kits

Welcome to the exciting world of Velleman Kits.

Velleman Kit is known all over the world for our High Quality electronic kits. Our range goes from easy to build Mini Kits to more advanced Kits such as High-End audio circuits, domotic systems and computer interface kits.

Because of the huge success of our sophisticated measuring instruments like the PC- and hand-held oscilloscopes, we've founded the Velleman Instruments division.

The Velleman Kit team

Table of Contents

Allgemeines	2
FCC informationen	2
Spezifikationen	3
Verbindung	3
Hardware installieren	4
Programmierungsverfahren	5
Program schreiben/ändern	5
Kompilieren	5
Eigentliches Programmieren des Mikrocontrollers	7
Programmieren	8
Pic-Gerät löschen	10
Experimente	11
ICSP	12

1 Allgemeines

Der K8048 ist ein multifunktionaler Programmierkasten für Microchip ® PIC™ FLASH Mikrocontrollern*. Diese 'Controllers' können verschiedene Male aufs neue programmiert werden und deshalb können Sie die Software eines Geräts, in dem diese implementiert sind, einfach upgraden und nach Herzenslust experimentieren. Mithilfe des K8048 können verschiedene Typen von IC's programmiert werden. Es gibt auch LEDs und Drückknöpfe auf der Platine um kleine Schaltungen zu prüfen. Es gibt 2 wichtige Stufen bei der Anwendung dieses Programmierkastens: Erstens wird der Programmcode mit einem ASCII Textautomaten, z.B. NotePAD, der als Standard bei Microsoft Windows mitgeliefert ist, geschrieben, oder Sie können das Entwicklungspaket von Microchip MPLAB, das man auf der Website www.microchip.com finden kann, verwenden. Zweitens wird nach der Kompilationsstufe mithilfe des K8048 und der PICPROG2 Software das Programm im Prozessor programmiert.

Auf dem K8048 sind Stellen für die unmittelbare Montage verschiedener 'Footprints' vorgesehen: 8-Pins, 14-Pins, 18-Pins und 28-Pins. Andere Typen von 'Footprints' sind dank der ICSP-Schnittstelle auch verwendbar. Mit dieser Erweiterung ist es möglich, andere Anschlüsse oder Controller, die sich in einem Gerät befinden, zu programmieren, ohne dass sie aus der Halterung entfernt werden müssen (siehe ICSP).

Mit dem mitgelieferten Controller können Sie sofort mit der Programmierung anfangen.

* The Microchip name and logo, PIC, and PICmicro are registered trademarks of Microchip Technology Inc. in the USA and other countries.

2 FCC Informationen

Dieses Gerät ist konform mit Teil 15 der FCC Normen, unter der Bedingung, dass es in Übereinstimmung mit dieser mitgelieferten Bauanleitung zusammengesetzt wurde. Für den Betrieb gelten die folgenden Bedingungen (1) das Gerät darf keine schädliche Interferenz verursachen, und (2) der Betrieb darf nicht durch ungewünschte Interferenz beeinflusst werden.

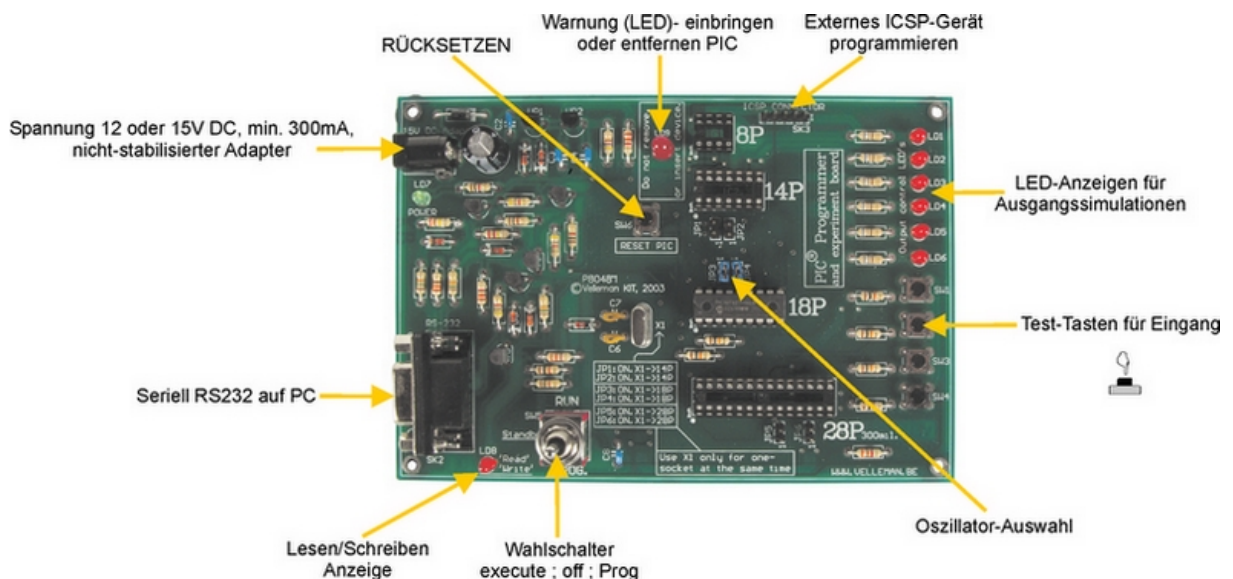
Mehr Informationen über FCC auf <http://www.fcc.gov>

3 Spezifikationen

- Geeignet zum Programmieren von Microchip® FLASH PIC(tm) Mikrocontrollerns.
- Unterstützt 4 verschiedene 300 mil. PICs: 8p, 14p, 18p und 28p.
- Test-Tasten und LED-Anzeigen um Bildungsexperimente anzustellen, wie bei den mitgelieferten Programmbeispielen.
- Einfach an einen PC anzuschließen dank dem seriellen Port.
- Der Flash Mikrocontroller (PIC16F627), der bis 1000 Male umprogrammiert werden kann, ist im Lieferumfang enthalten.
- Software zum Programmieren Ihres Quellcodes ist mitgeliefert.
- Spannungsversorgung: 12 oder 15V DC, min. 300mA, nicht-stabilisierter Adapter
 - (PS1205 / PS1208/PS1508 (230Vac))
 - (PS1208USA (115Vac))
- IBM-kompatibler PC, Pentium oder höher, mit Windows™ 95/98/ME/NT/2000/XP, CDRom und unbenutztem seriellen RS232-Port. (Kabel nicht mitgeliefert, z.B. CW014).
- Unterstützt diese FLASH Mikrocontrollerns:
 - PIC12F629, PIC12F675, PIC16F83, PIC16F84(A), PIC16F871, PIC16F872,
 - PIC16F873, PIC16F874, PIC16F876, PIC16F627(A), PIC16F628(A), PIC16F630,*
- Abmessungen: 145 mm x 100 mm.

(*)An updated list and software updates can be found on our website: www.velleman.be

4 Verbindung



5 Hardware installieren

- Verbinden Sie den K8048 mit Ihrem PC mithilfe eines 9-poligen seriellen Kabels. Verwenden Sie hierzu einen unbenutzten seriellen Port und sorgen Sie dafür, dass installierte Software während der Anwendung des K8048 diesen Port nicht benutzt. (interrupt conflict), z:B. Steuerprogramme eines anderen Programmers, Messgeräte, PDAs usw.
 - Setzen Sie den Wahlschalter (SW5) auf die mittlere Position (Standby).
 - Schließen Sie einen 12 bis 15V DC, min. 300mA, nicht-stabilisierten Adapter über SK1 an den K8048 an. (Empfohlener Velleman-Adapter: [PS1205](#) / [PS1208](#) / [PS1508](#) (230Vac) oder [PS1208USA](#) (115Vac))
 - Die 'POWER' LED (LD7) soll jetzt als einzige LED aufleuchten, als Anzeige, dass die Spannungsversorgung des K8048 in Ordnung ist.
-

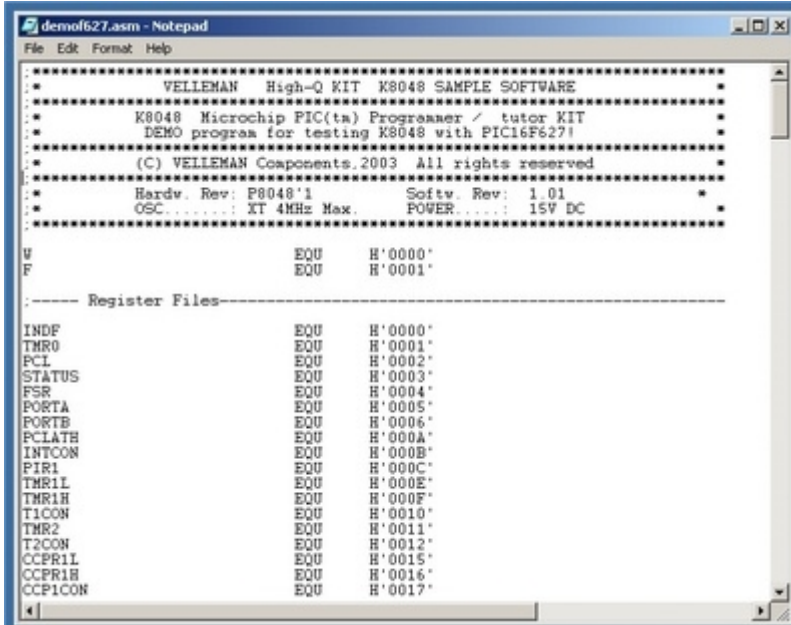
6 Programmierungsverfahren

Mikrocontroller programmieren: Vorbereitungen

- **Das Schreiben oder Ändern des Quellcodes** ist meistens im PIC Assembler programmiert. Siehe Beispiele in dem Velleman\K8048\Examples\ - Ordner.
- **Kompilieren:** hier wird der Code, den Sie in 'Normalsprache' geschrieben haben, in Computersprache umgesetzt. Das geschieht mithilfe der Microchip PIC Assembler Software, MPASMWIN.EXE.
- **Das Programmieren:** hier wird der Maschinencode über Ihren PC und den K8048, mit der PROGPI2.EXE Software im Prozessor programmiert.
- **Pic-Gerät löschen:** Inhalt eines Controllers löschen.

6.1 Program schreiben/ändern

Sie schreiben oder ändern das Programm: \Velleman\K8048\Examples\DEMOF627.ASM



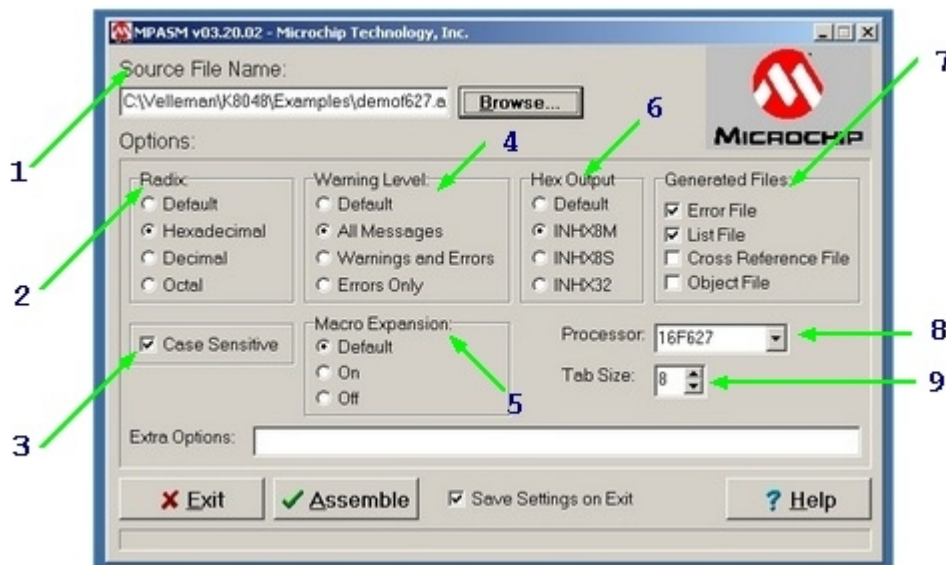
```
demof627.asm - Notepad
File Edit Format Help
*****
* VELLEMAN High-Q KIT K8048 SAMPLE SOFTWARE
*****
* K8048 Microchip PIC(tm) Programmer / tutor KIT
* DEMO program for testing K8048 with PIC16F627!
*****
* (C) VELLEMAN Components,2003 All rights reserved
*****
* Hardw. Rev: P8048'1          Softw. Rev: 1.01
* OSC.....: XT 4MHz Max.     POWER.....: 15V DC
*****
W          EQU   H'0000'
F          EQU   H'0001'

----- Register Files -----
INDF      EQU   H'0000'
TMR0      EQU   H'0001'
PC1        EQU   H'0002'
STATUS    EQU   H'0003'
FSR       EQU   H'0004'
PORTA     EQU   H'0005'
PORTB     EQU   H'0006'
PCLATH    EQU   H'000A'
INTCON    EQU   H'000B'
PIR1      EQU   H'000C'
TMR1L     EQU   H'000E'
TMR1H     EQU   H'000F'
T1CON     EQU   H'0010'
TMR2      EQU   H'0011'
T2CON     EQU   H'0012'
CCPR1L    EQU   H'0015'
CCPR1H    EQU   H'0016'
CCP1CON   EQU   H'0017'
```

Beispiel eines Programms, das mit 'NOTEPAD' bearbeitet wird.

6.2 Kompilieren

Lassen Sie MPASMWIN.EXE im Velleman\K8048-Ordner ablaufen
Sie können folgenden Schirm mit den Einstellungen sehen (für weitere Informationen, bitte klicken Sie auf die Nummer)



- 1 Der Name und die Speicherstelle der Datei z.B. C:\TEST\DEMOF267.ASM
- 2 Hier wählen Sie das Format des herzustellenden Maschinencodes. Für den K8048 wählen Sie immer 'HEXADEMICAL'
- 3 Wenn diese Option abgehakt ist, dann werden die Variablen auf Ihre Schreibweise kontrolliert (groß oder klein geschrieben).
- 4 Hier bestimmen Sie das Maß, in dem Fehlermeldungen und Warnungen in der Fehlerdatei generiert werden. Am besten wird ALL MESSAGES oder ERRORS ONLY gewählt.
- 5 Lassen Sie diese Option auf 'Default' (Standardeinstellung) stehen. In der EIN-Position werden die Makro völlig in der LIST-Datei ausgeschrieben (xxx.LST).
- 6 Dies ist die Wortlänge der Bytes in der Maschinencode-Datei für die PROG PIC2 Software, die zu dem K8048 gehört. Wählen Sie immer 'INHX8M'. Andere Wahlen werden einen unlesbaren oder nicht-funktionierenden Maschinencode zur Folge haben.
- 7 Hier können Sie anklicken welche Dateien Sie zusätzlich auf den HEX Maschinencode generieren möchten.

Error file: Hier werden die Fehlermeldungen gespeichert.

List file: Datei enthält Ihren Quellcode + Fehlermeldungen und zusätzliche Information.

Die 'CROSS REFERENCE & OBJECT'-Dateien brauchen Sie nicht zu generieren, mit dem K8048 können Sie diese sogar nicht verwenden.

8. Hier sollen Sie den richtigen Prozessor, den Sie verwenden, wählen. Manche PIC's sind nach ihren Nummern, auch mit einem Erzeugungscode versehen; auch diese sind sehr wichtig. z.B.: PIC16F84A.
9. Geben Sie die Spaltenbreite in Zeichenanzahl in den generierten Textdateien wieder..

Wenn alles korrekt eingestellt worden ist, klicken Sie auf **<ASSEMBLE>**

Wenn keine Fehler im Quellcode vorgefunden werden, erscheint der Schirm wie unten abgebildet.

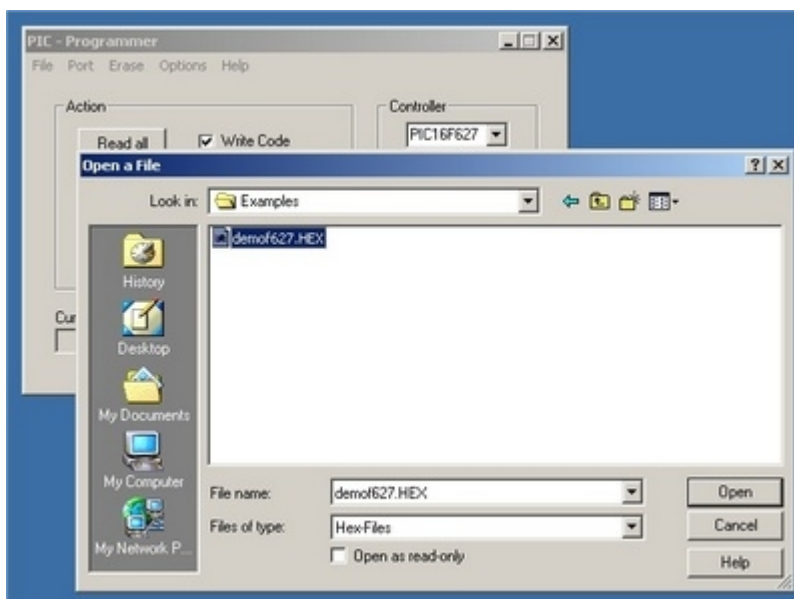


Der Assembler erzeugt jetzt die folgenden Dateien:

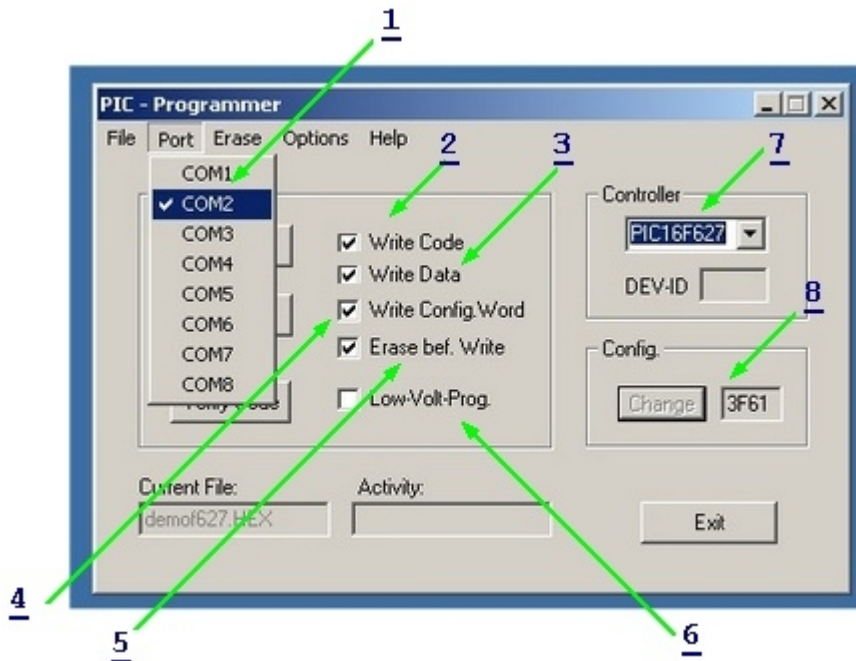
- **xxx.ERR** => Fehlerdatei, enthält alle Fehler und/oder Warnungen, die der MPASM Assembler in Ihrem Quellcode gefunden hat.
- **xxx.Hex** => Ist die eigentliche Maschinencode-Datei, die durch die Programmiersoftware zum effizienten Programmieren des Controllers verwendet werden wird.
- **xxx.LST** => Listing-Datei, enthält Ihren Quellcode mit der Anmerkung, die der Assembler generiert.
- **xxx.COD** => Code-Datei, wird nur von der MPLAB Programmierumgebung verwendet und nicht von der K8048 Software.

6.3 Eigentliches Programmieren des Mikrocontrollers

- Verbinden Sie den K8048 mit Ihrem Computer über einen freien seriellen Port.
- Die IRQ-Linie dieses seriellen Ports (IRQ 3 of 4) darf auch nicht von einem anderen Gerät, z.B. von einem internen Modem verwendet werden.
- Schließen Sie den passenden Adapter an und überprüfen Sie ob die LD7 "Power LEDS" aufleuchtet.
- Lassen Sie das PROG PIC2, PIC Programmer-Programm ablaufen. (Befindet sich im C:\Velleman\K8048\ - Ordner).
- Klicken Sie auf 'FILE' -> 'OPEN' und wählen Sie das kompilierte Programm (Hex 8M-bestand), das Sie programmieren wollen.



- Überprüfen Sie die Einstellungen der Software

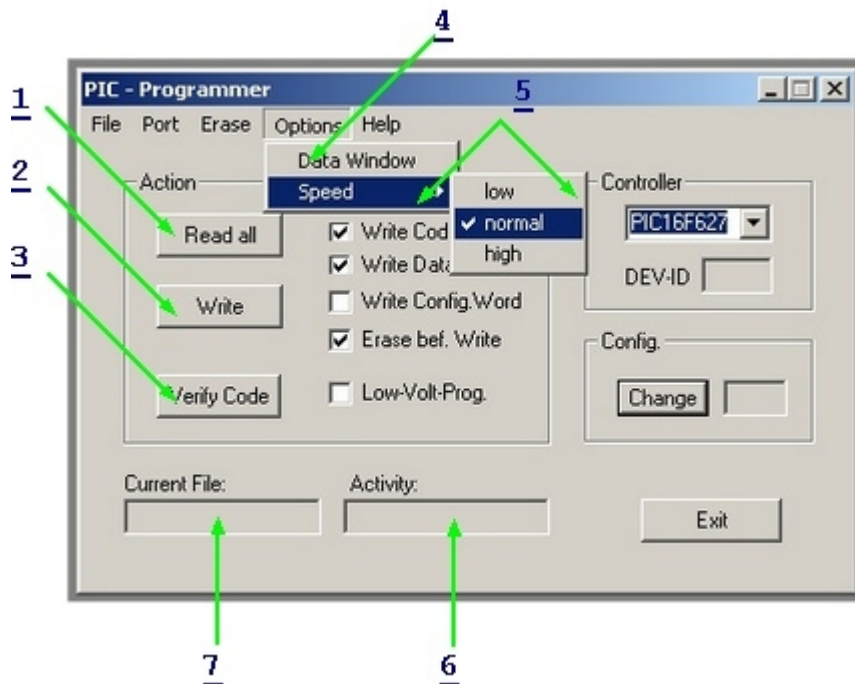


1. Stellen Sie die Nummer des seriellen Ports, an dem der K8048 angeschlossen ist, ein..
2. Programmieren des CODES, EIN ist abgehakt (Standardeinstellung = EIN).
3. Programmieren der EEPROM-Daten, EIN ist abgehakt (Standardeinstellung = EIN).
4. Programmieren der Konfigurationseinstellungen, EIN ist abgehakt (Standardeinstellung = EIN).
5. ERASE before WRITE (LÖSCHEN vor SCHREIBEN): EIN ist abgehakt (Standardeinstellung = EIN). Das bedeutet, dass vor dem Programmieren, der Controller gelöscht wird.
6. LOW VOLTAGE (NIEDERSpannung) Programmieren: ist nicht zutreffend für den K8048. Alle Controllers werden, mit einer Programmierspannung (VPP) von 13V programmiert und nicht mit 5V.
7. Hier wählen Sie den benutzten Mikrocontroller. Der mitgelieferte Mikrocontroller ist des Typs PIC16F627 oder PIC16F627A. Achten Sie darauf, dass Sie die richtige Typbezeichnung selektiert haben.
8. Die Konfiguration des Controllers wird normalerweise schon im Quellcode mit der `__CONFIG` Compiler-Anweisung bestimmt. Nur der fortgeschrittene Anwender kann diese Optionen manuell einstellen oder anpassen.

6.3.1 Programmieren

- Vergewissern Sie sich, dass der SW5 auf der mittleren Position (Standby) steht.
- Montieren Sie den Controller in der entsprechenden IC-Fassung
 - o 08 -Pins PIC montieren Sie in der IC1-Fassung
 - o 14 -Pins PIC montieren Sie in der IC2-Fassung
 - o 18 -Pins PIC montieren Sie in der IC3-Fassung
 - o 28 -Pins PIC montieren Sie in der IC4-Fassung
- Setzen Sie jetzt den SW5 auf 'PROG'. Die Warnungs-LED LD9 wird jetzt blinken. Das bedeutet, dass wenn SW5 in dieser Position geschaltet ist, kein Controller in die Fassung eingebracht oder aus der Fassung entfernt werden darf.
- Klicken Sie jetzt auf WRITE, der Controller wird programmiert. Sie können die Adressen im Fenster 'Activity' sehen.'READ/WRITE' - Anzeige, LED LD8 leuchtet auf.
- Wenn das Programmieren beendet ist, dann können Sie SW5 erneut in die mittlere Position

setzen STAND-BY) bevor Sie den Controller entfernen oder auf den Test-Modus umschalten (siehe Test-Tasten und Diagnose-LEDs LED's unter 'Experimente'). Zum Testen des Programms (nur wenn das Programm so geschrieben ist, dass die Test-Tasten und LEDs verwendet werden) müssen Sie den SW5 in die 'RUN'-Position schalten.



1. Mit diesem Knopf können Sie ein Programm und Daten aus einem Controller ablesen, aber nur wenn dieser Controller nicht 'code-protected' (Code-geschützt) ist. PICs, deren Codeschutz eingestellt ist, können nicht mehr gelesen, sondern nur noch gelöscht werden und dann werden auch alle Codes gelöscht. Der hexadezimale Maschinencode wird in einer Textdatei gespeichert.

* **Achtung:** Um diese Funktion verwenden zu können, muss SW5 in die 'PROG'-Position gesetzt werden!

2. Der Controller wird programmiert, Sie können die Adresse im Fenster 'Activity' sehen. Jetzt brennt LD8, die 'READ/WRITE'- Anzeige.
3. Mit dieser Taste können Sie überprüfen, ob der PIC das Programm, das Sie soeben geschrieben haben, auch richtig gespeichert hat. Achtung, der Controller darf nicht Code-geschützt sein. Während des Programmierens wird jedes Byte automatisch geprüft ob es richtig programmiert wurde, bevor der Codeschutz eingeschaltet wird.

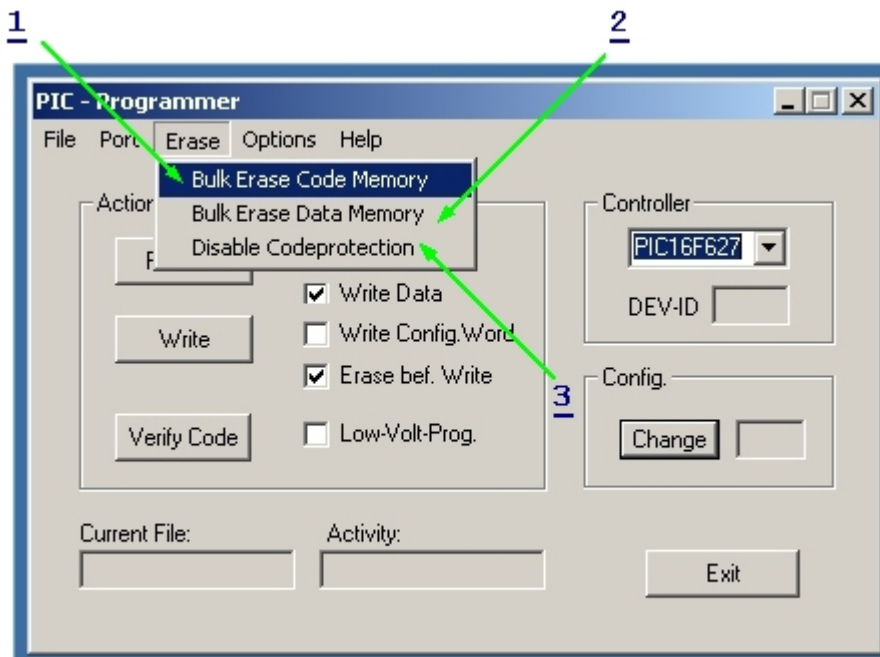
* **Achtung:** Um diese Funktion verwenden zu können, muss SW5 in die 'PROG'-Position gesetzt werden!

4. Ermöglicht Ihnen, die Daten, die sich im EEPROM-Speicher eines Controllers befinden, nachzuprüfen und zu ändern. * **Achtung:** Um diese Funktion verwenden zu können, muss SW5 in die 'PROG'-Position gesetzt werden!
5. Normalerweise lassen Sie diese Einstellung auf 'NORMAL' stehen. Wenn Sie bei der Programmierung Problemen begegnen, können Sie bei langsameren Computern, die Geschwindigkeit auf 'HIGH' einstellen. Bei sehr schnellen Computern können Sie die Geschwindigkeit auf 'LOW' einstellen um den Geschwindigkeitsunterschied zu kompensieren.
6. Hier kann der Anwender sehen, welche Adresse oder welchen Adressenbereich die Software benutzt zum Lesen oder Schreiben.
7. Name der Datei, die in den Speicher geladen ist und zum Programmieren verwendet werden wird.

6.4 Pic-Gerät löschen

Inhalt eines Controllers löschen:

1. Dient zum Löschen des Gesamtinhalts eines Programms aus einem Controller.
2. Dient nur zum Löschen der EEPROM-Daten.
3. Damit können Sie den Codeschutz ausschalten, aber der Inhalt kann nicht mehr gelesen und das Programm kann nicht mehr verwendet werden.



* **Achtung:** Wenn Sie diese Funktion verwenden möchten, müssen Sie den SW5 in die 'PROG'-Position schalten!

1. Dient zum Löschen des gesamten Inhalts eines Programms aus dem Controller.
2. Dient nur zum Löschen der EEPROM-Daten.
3. Dient zum Ausschalten des Codeschutzes, aber der Inhalt kann nicht mehr gelesen werden und das Programm wird unbenutzbar.

Sie können Programmbeispiele in dem C:\Velleman\K8048\Examples Ordner finden.
Updates der Programmsoftware und zusätzliche Beispiele können Sie über unsere Website www.velleman.be herunterladen.

7 Experimente

Anwendung von den Test-Tasten und den LEDs

Der K8048 verfügt über max. 4 Test-Eingänge, die Sie manuell mit SW1, SW2, SW3 und SW4 bedienen können.

Er verfügt auch über max. 6 Diagnose-LEDs, mit denen Sie die Ausgänge simulieren können.

Indem Sie auf den Link 'cross-reference'-Tabelle klicken, können Sie für jede IC-Fassung nachschlagen welche I/O- Linie mit welcher Taste oder LED verbunden ist.

	8 pin PIC Fassung	14 pin PIC Fassung	18 pin PIC Fassung	28 pin PIC Fassung
LD1	GP2 (pin 5)	RC0 (pin 10)	RB0 (pin 6)	RA0 (pin 2)
LD2	GP4 (pin 3)	RC1 (pin 9)	RB1 (pin 7)	RA1 (pin 3)
LD3	NC	RC2 (pin 8)	RB2 (pin 8)	RA2 (pin 4)
LD4	NC	RC3 (pin 7)	RB3 (pin 9)	RA3 (pin 5)
LD5	NC	RC4 (pin 6)	RB4 (pin 10)	RA4 (pin 6)
LD6	NC	NC	RB5 (pin 11)	RA5 (pin 7)
SW1	GP5 (pin 2)	RC5 (pin 5)	RA0 (pin 17)	RB0 (pin 21)
SW2	NC	RA2 (pin 11)	RA1 (pin 18)	RB1 (pin 22)
SW3	NC	NC	RA2 (pin 1)	RB2 (pin 23)
SW4	NC	NC	RA3 (pin 2)	RB4 (pin 25)

Anmerkungen: Die I/O-Linie eines Druckknopfes sollen in der Quelle als Eingang konfiguriert sein (1 im TRIS-Register für diesen bestimmten Eingang). Die I/O-Linien einer LED sollen in der Quelle als Ausgang konfiguriert sein (0 im TRIS-Register für diesen bestimmten Eingang).

8 ICSP

Mit ICSP wurde beabsichtigt, Controllers "in-circuit" zu programmieren.

Dieses Verfahren wird auch 'code downloading' genannt, dies heißt, dass der einschlägige Chip einfach mit einem Bus mit 4 oder 5 Leitungen, dem ICSP-Bus, verbunden ist. Verschiedene Mitglieder der PIC-Familie können programmiert und umprogrammiert werden, während sie noch immer auf der Platine befestigt sind und die Versorgung sogar nicht abgekuppelt werden muss.

Diese IC-Fassungen werden auf den Programmier-Modus umgeschaltet, indem Sie eine Spannung (normalerweise) zwischen 12V und 14V auf "/MCLR" applizieren. Ab diesem Moment wird RB6 als CLOCK-Eingang funktionieren, während RB7 in einen DATA-Eingang/Ausgang geändert wird.

