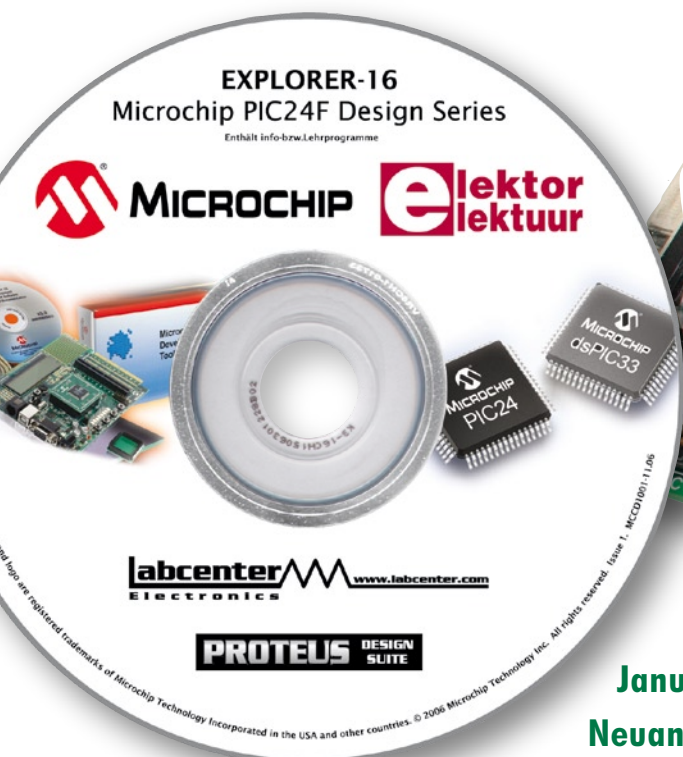


# Explorer-16

Von Jan Buiting & Luc Lemmens, in Kooperation mit  
Microchip UK & Labcenter Electronics



✓ **C30 Compiler**  
 ✓ **MPLAB IDE**  
 ✓ **Proteus VSM**

Januar ist der Monat der  
Neuanfänge – und diesen wollen  
wir nutzen, Ihnen den Umstieg auf moderne 16-bit-

Mikrocontroller zu erleichtern. Das *Explorer-16*-Projekt entstand in  
Zusammenarbeit mit Microchip und Labcenter. Auf der beigelegten CD-ROM finden  
Sie Microchip MPLAB, C30 und last not least Proteus VSM. Und im nächsten Monat  
warten hochinteressante Hardware-Angebote auf Sie!

Das Projekt *Explorer-16* stellt einen Schritt von 8- auf 16-bit-Mikrocontroller dar, richtet sich aber nicht nur an Controller-Experten. Das Projekt eignet sich sogar sehr gut für Anfänger: Man kann ohne weiteres direkt mit 16-bit-Mikrocontrollern loslegen, denn die dafür zur Verfügung stehenden Tools sind außergewöhnlich leistungsfähig. Wir werden also sozusagen dennoch auf relativ hohem Niveau einsteigen – denn schließlich ist alles, was Sie für die Programmierung Ihres eigenen 16-bit-PIC®-Controllers benötigen, Teil dieses Projekts. Für dieses Projekt fiel die Wahl auf die PIC-Serie, da hierfür hochqualitative kostenlose Software zur Verfügung steht und die beschriebene Hardware oft sehr günstig gehalten werden kann.

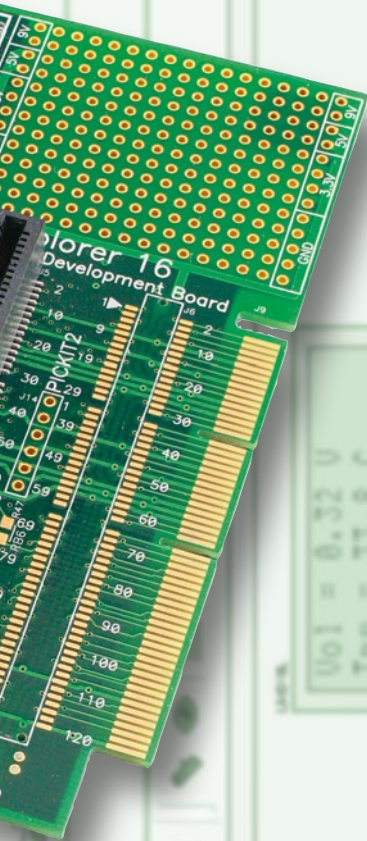
## Projekt-Überblick

Die zu diesem Projekt gehörende Artikelserie steht auf drei Säulen:

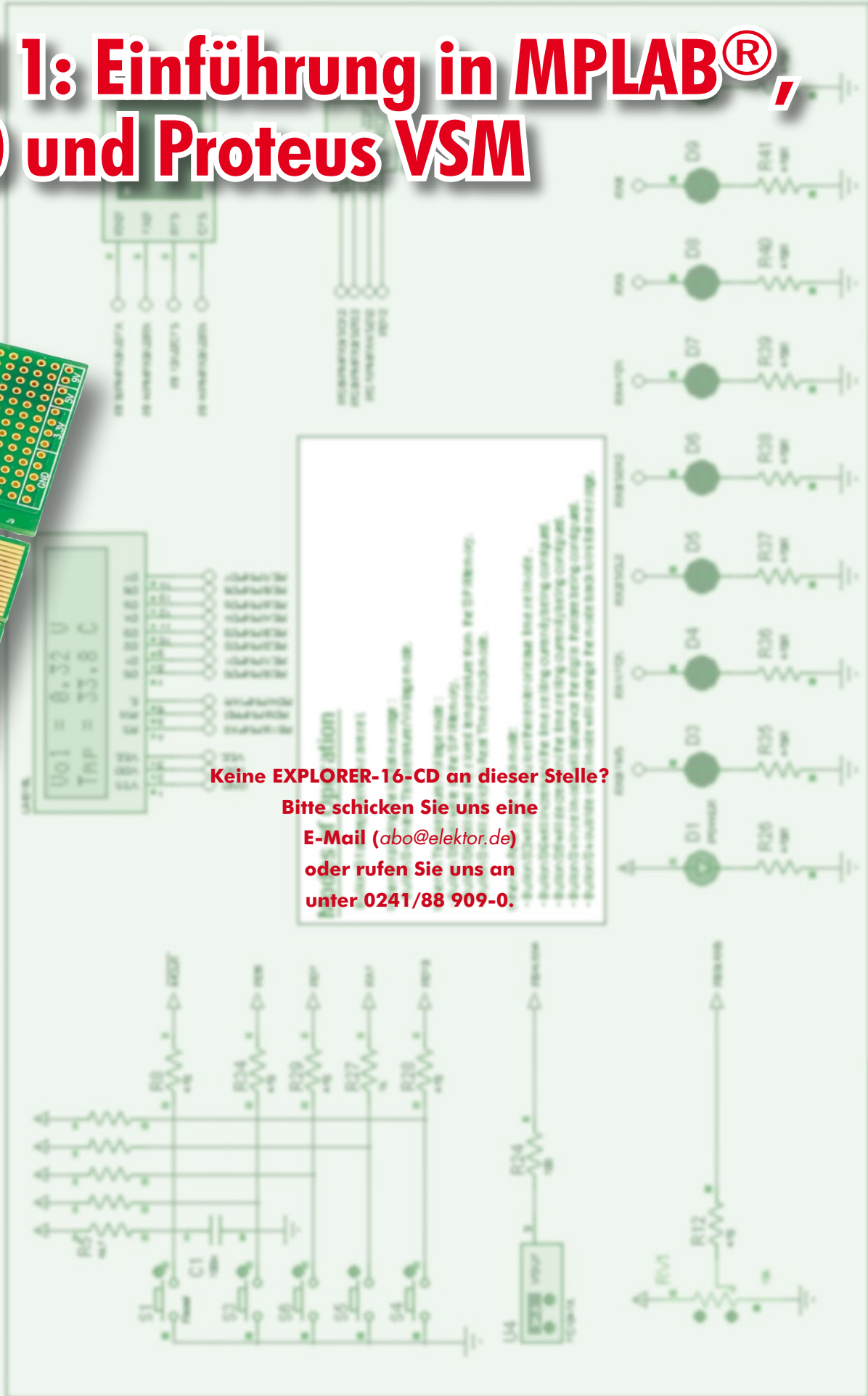
### 1. Software

Die in dieser Januar-Ausgabe enthaltene kostenlose CD-ROM enthält drei wichtige Software-Pakete, die sich nicht nur für den Typ PIC24F (ein 16-bit-Mikrocontroller von Microchip) eignen, sondern auch prima das Grundkonzept der Hardware-Simulation veranschaulichen. Alle drei Komponenten sind sehr benutzerfreundliche, schlanke Programme, die auf einem halbwegs aktuellen PC unter Windows laufen:

# Teil 1: Einführung in MPLAB<sup>®</sup>, C30 und Proteus VSM



Keine EXPLORER-16-CD an dieser Stelle?  
Bitte schicken Sie uns eine  
E-Mail ([abo@elektor.de](mailto:abo@elektor.de))  
oder rufen Sie uns an  
unter 0241/88 909-0.



Explorer 16 Virtual Evaluation Board

Inter active Elements

**PROTEUS**  
The Complete Electronics Design System

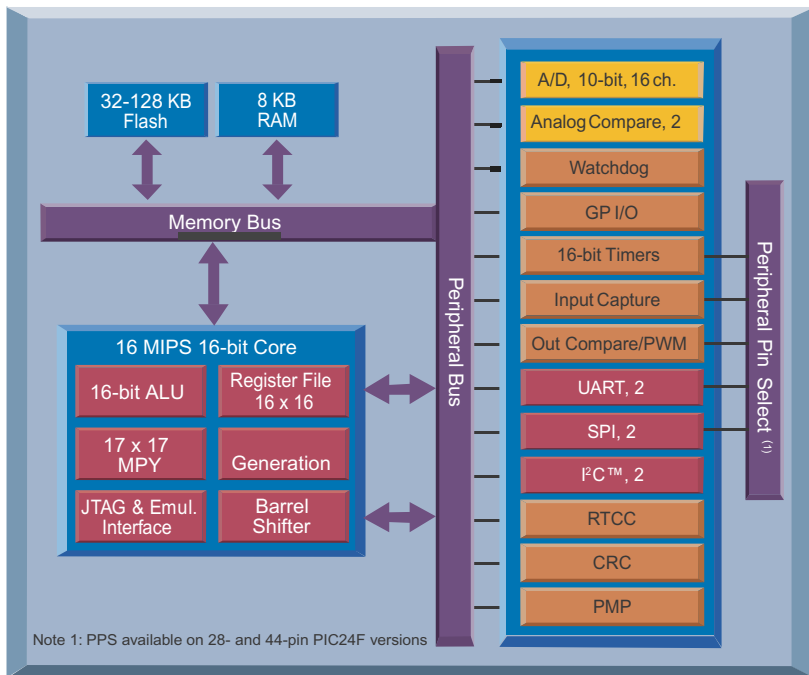


Bild 1. Blockdiagramm des PIC24F-Controllers.

- MPLAB IDE — Version 7.50 der bekannten Entwicklungs-Plattform für PIC-Controller.
- MPLAB C30 — Version 2.05 (Studentenversion) des C-Compilers für PIC-Controller, komplettiert durch eine Sammlung von Utilities.
- Proteus VSM — Version 7.00. Es handelt sich um eine Spezialausgabe der Labcenter Simulations-Software, die so schnell ist, dass sie eine komplette Mikrocontroller-Platine auf einem normalen PC simulieren kann, und dies sogar in 16-bit-Ausführung. Als Premiere gibt's dabei ein kostenloses PIC24F-Simulations-Plugin, das die Verbindung mit der Explorer-16-Platine herstellt.

Mit diesen Software-Paketen kann man sogar ganz ohne Hardware - rein virtuell - eigene Mikrocontroller-Projekte entwickeln. Einzige Voraussetzung ist ein gewöhnlicher PC!

**2. Hardware**

Mit der Februar-Ausgabe 2007 wird ELEKTOR eine **Explorer-16-Entwicklungsplatine** mit einer speziellen Zusatzkonfiguration zu unglaublich günstigen Konditionen anbieten (siehe Foto).

**3. Internet-Support**

Alle für das Projekt erforderlichen zusätzlichen Dateien und Infos finden sich auf einer speziellen Explorer-16-Seite unter [www.elektor.de](http://www.elektor.de). Außerdem werden Ihnen Experten von Labcenter und Microchip in einem - englischsprachigen - Forum mit Rat und Tipps zur Seite stehen ([www.elektor-electronics.co.uk](http://www.elektor-electronics.co.uk)).

**PIC24F-Architektur**

Der grundsätzliche Aufbau eines PIC24F-Controllers erschließt sich durch das Block-Diagramm des Controllers in **Bild 1** und des CPU-Kerns in **Bild 2**. Unsere Entwicklungs-Platine nutzt das leistungsfähigste Mitglied der Controller-Familie: den Typ **PIC24FJ128GA010**. Dieser Chip verfügt über 128 kB Flash-Speicher – daher ist ein Gehäuse mit 100 Pins notwendig! Seine „Visitenkarte“ findet sich im Kasten **PIC24F128-Features**. Der Chip prahlt mit A/D, UART, SPI, I<sup>2</sup>C, RTCC, WDT ICE, ISP und POR. Erweiterte Stromsparmöglichkeiten und ein Parallel-Port mit Master/Slave-Funktionalität kommen noch hinzu. Zwei der verlinkten Datenblätter, die PDF-Dokumente [1] und [2], sollten Sie am besten jetzt gleich runterladen und in einem Ordner namens „Explorer-16“ auf Ihrer Festplatte ablegen. Die Datei zum PIC24F ist mit ihren 230 Seiten schon etwas mehr als ein Datenblatt. Es handelt sich glücklicherweise um ein ordentliches PDF mit Tabs zu den einzelnen Kapiteln. Falls Sie von der populären PIC18F-Serie auf die neue 24F-Familie umsteigen, ist vermutlich auch [3] für Sie von Interesse.

**Software-Installation**

Nach Einlegen der beigelegten CD sollte ein Begrüßungs-Fenster erscheinen (**Bild 3**). Falls nicht, klicken Sie bitte auf Start → Ausführen... und dann via „Durchsuchen“ zur Datei „*index.htm*“ auf der CD. Den Installations-Bildschirm, der nach Verlassen der Begrüßung zu sehen ist, zeigt **Bild 4**. Die Installation verläuft ohne Haken und Ösen. Es wird empfohlen, alle drei Software-Komponenten zu installieren. Die Installations-Software selbst bedient sich des als Standard etablierten Browsers. Falls Sie einen Exoten installiert haben, mit dem es Schwierigkeiten gibt: Mit dem IE in der Version 6.0.2900 klappt es auf alle Fälle.

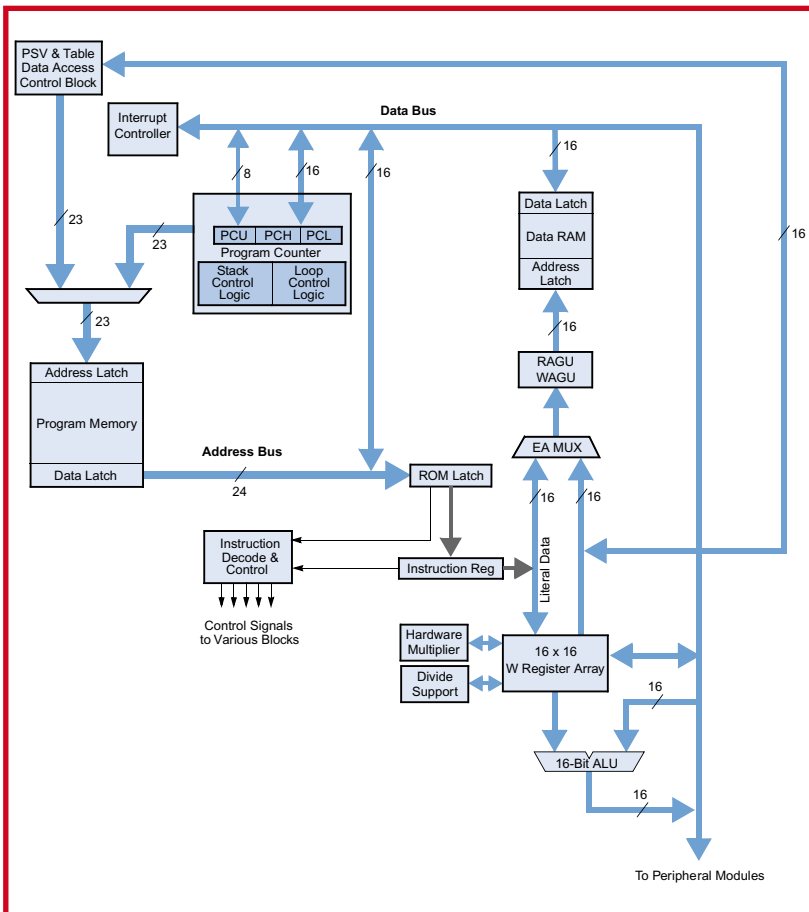


Bild 2. Blockdiagramm des PIC24F-CPU-Kerns.

**MPLAB IDE** wird wie ein konventionelles Windows-Programm installiert. An einem bestimmten Punkt erscheint die Meldung: „Der Herausgeber konnte nicht verifiziert werden. Wollen Sie die Software wirklich ausführen?“ Dann einfach auf die Schaltfläche „Ausführen“ klicken, um die Installation fortzusetzen.

**MPLAB C30** verhält sich ganz ähnlich. Das Dokument mit den *Release Notes*, das Sie zum Schluss angeboten bekommen, ist lesenswert und sollte gesichert werden.

**Proteus VSM 7.00 Demo** ist sehr einfach zu installieren, indem einfach alle vorgeschlagenen Optionen akzeptiert werden. In einer „Guided Tour“ zum MPLAB-Viewer am Schluss der Installation finden sich einige interessante Informationen.

## MPLAB IDE

Angesichts des Umfangs und der Leistungsfähigkeit ist es geradezu paradox, dass man nur wenige Worte zur neuesten PIC-Entwicklungsumgebung von Microchip verlieren muss. MPLAB ist schon viele Jahre auf dem Markt und hat sich nicht ohne Grund zum De-Facto-Standard für PIC-Controller entwickelt. Das Programm - oder präziser

**EXPLORER-16** wird exklusiv von Microchip Technology, Labcenter Electronics und ELEKTOR angeboten. Als Teil des Projekts enthält die Januar-Ausgabe 2007 von ELEKTOR eine CD als kostenlose Beilage. Die Explorer-16-Entwicklungs-Platine und eine Aufsteck-Platine werden in der Februar-Ausgabe beschrieben und zu besonders günstigen Konditionen angeboten.

gesagt der Satz an Utilities - ist sehr leicht verständlich, intuitiv verwendbar und mit einem guten Support versehen. Zudem sind noch ausgeklügelte Hilfe-Dateien vorhanden, die von Microchips Forum und dem technischen Support ergänzt werden. Die integrierte Entwicklungsumgebung MPLAB ist für Microchips PICmicro-Controller und die Chips der dsPIC®-Familie geeignet. Die IDE enthält folgende Komponenten: MPASM™ (Assembler), MPSIM™ (Software-Simulator), MPLINK™ (Linker), einen Source-Level-Debugger, eine Online-Hilfe, Projekt- und Setup-Wizards, einen Projekt-Manager, einen Visual-Device-Initialisierer, einen Editor und Treiber für diverse Hardware-Erweiterungen. Spezielle Versionen einiger Utilities wie Assembler und Linker unterstützen auch Chips der dsPIC3-Serie.

## PIC24FJ128-Features

### CPU

- Modifizierte Harvard-Architektur
- Bis zu 16 MIPS @ 32 MHz Takt
- 8 MHz interner Taktgenerator:
  - 4x-PLL-Option
  - Viele Teilerfaktoren
- 17-bit x 17-bit Einzeltakt-Operation
- Fließkomma- und Integer Multiplikation
- 32-bit / 16-bit Hardware-Division
- 16 x 16-bit Register-Array
- C-Compiler-optimierter Befehlssatz:
  - 76 Basis-Befehle
  - Flexible Adressiermöglichkeiten
- Linearer Adressraum für Programmspeicher bis zu 12 MByte
- Linearer Adressraum für Daten bis zu 64 kByte
- Zwei Adressgeneratoren für separaten Lese- und Schreibzugriff auf Daten

### Spezielle Mikrocontroller-Features

- Betriebsspannung: 2 V bis 3,6 V
- Flash-Speicher:
  - ca. 1000 Lösch/Schreib-Zyklen
  - ca. 20 Jahre Datenstabilität
- Auto-Reprogrammierfähigkeit via Software
- Wählbare Stromspar-Modi:
  - Sleep, Idle und alternative Taktung
- Ausfallsicherer Takt:
  - Entdeckt Takt-Fehler und schaltet ggfs. auf den internen Low-Power-RC-Oszillator um
- On-Chip-LDO-Regler
- JTAG-Scan und Programmier-Support
- Power-On-Reset (POR), Power-Up-Timer (PWRT) und Oszillator-Start-Up-Timer (OST)

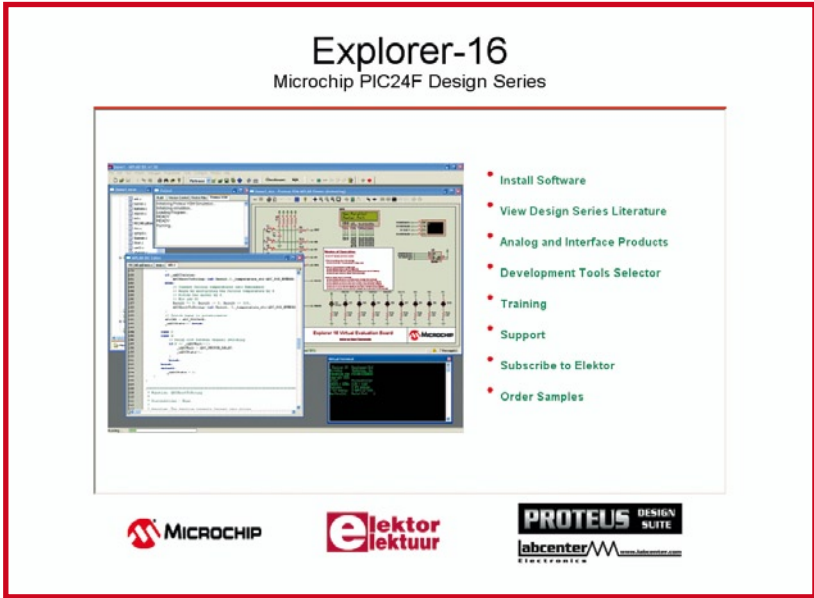
- Flexibler Watchdog-Timer (WDT) mit internem, stabilem Low-Power-RC-Oszillator
- In-Circuit-Serial-Programming (ICSP™) und In-Circuit-Emulation (ICE) mit 2 Pins

### Analoge Features

- 10-bit, bis zu 16-Kanal-Analog/Digital-Konverter (A/D)
  - 500.000 Samples/s
  - A/D-Konversion auch in Sleep- und Idle-Modi
- Doppelte programmierbare Analog-Komparatoren

### Periphere Features

- Zwei 3-Wire/4-Wire-SPI-Module mit Unterstützung von vier Frame-Modi und vierstufigem FIFO-Buffer
- Zwei I<sup>2</sup>C™-Module mit Multi-Master/Slave-Modus und 7-bit/10-bit-Adressierung
- Zwei UART-Module:
  - Kompatibel mit RS-232, RS-485 und LIN 1.2
  - Kompatibel mit IrDA®, interner Hardware-Encoder/Decoder
  - Auto-Wake-Up-On-Start-Bit
  - Auto-Baudrate-Detektion
  - vierstufiger FIFO-Buffer
- Parallel Master Slave Port (PMP/PSP):
  - Unterstützt 8-bit- und 16-bit-Daten
  - Unterstützt 16 Adressleitungen
- Hardware Real-Time Uhr/Kalender (RTCC):
  - Uhr-, Kalender- und Alarmfunktionen
- Fünf 16-bit-Timer/Counter mit programmierbaren Vorteilern
- Fünf 16-bit-Eingänge
- Fünf 16-bit-Compare/PWM Ausgänge
- Hochstrom-I/O-Pins: Sink/Source 18 mA/18 mA
- Konfigurierbare Open-Drain-Ausgänge und digitale I/O-Pins
- Bis zu fünf externe Interrupt-Quellen

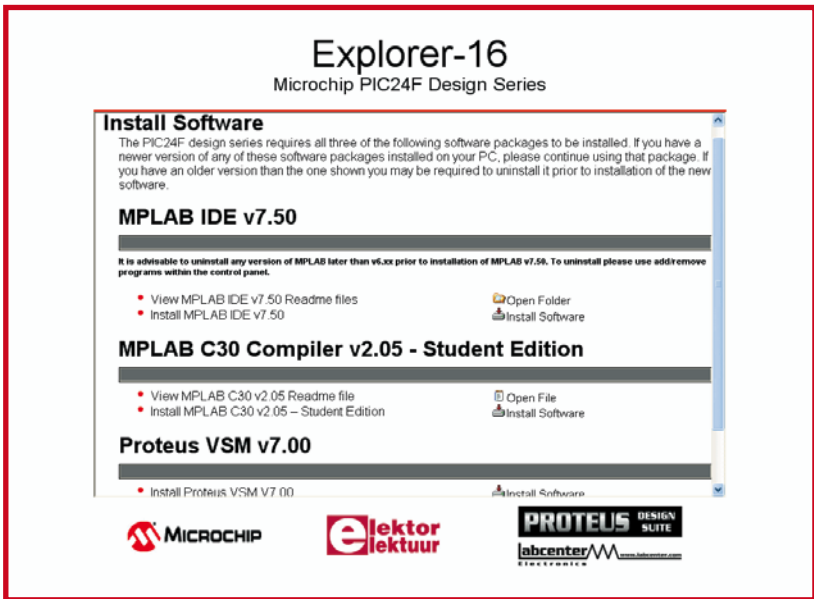


**Bild 3.** Die Startseite führt nicht nur zu den drei Installationsprogrammen, sie enthält auch Links zu einer ganzen Sammlung an Dokumenten zum Explorer-1-Projekt.

Mit MPLAB kann man:

- Eigene Quelltexte schreiben, editieren und debuggen;
- Fehler, die ein Compiler etc. findet, automatisch im Quelltext anzeigen lassen;
- Breakpoints für einfacheres Debugging setzen;
- Programme im Einzelschritt-Modus ablaufen lassen und dabei via Software-Simulation Werte von Variablen beobachten;
- Erstellten Code via Programmer in Mikrocontroller übertragen;
- Probleme auf einfache Weise und schnell lösen, da in der MPLAB IDE auch ein Online-Hilfesystem integriert ist.

MPLAB sieht etwas tot aus, wenn man es zum ersten Mal startet. Um etwas Action auf den Bildschirm zu zaubern,



**Bild 4.** Das Installations-Fenster der beigelegten Projekt-CD.

klicke man auf File → Open Workspace und navigiere zum Ordner „examples“. Hier finden sich etliche interessante Projekte, anhand derer besonders Einsteiger einen Überblick über die Vielzahl an Funktionen finden, ohne Hardware-Schäden zu riskieren. Die Release-Notes aller MPLAB-Utilities finden sich in Start → All Programs → Microchip → MPLAB IDE v. 7.50 → Documentation. Dieses Fenster listet auch alle „Install-and-Repair“-Utilities auf. Eine auch nur in Ansätzen grundlegende Diskussion der Features von MPLAB oder die Beschreibung der besten Art und Weise, MPLAB für eigene Projekte einzusetzen, würde mehr als eine komplette ELEKTOR-Ausgabe füllen. Ein komplettes „MPLAB Users Guide“ und etliche andere PDF-Dokumente befinden sich allerdings auf der CD. Zugriff erhalten Sie via „View Design Series Literature“. Außerdem gibt es mit „MPLAB Getting Started“ eine extra Einführung.

### MPLAB C30

Die Explorer-16-CD enthält die Studenten-Version des C30-Compilers von Microchip in der Version 2.05. Dieser C-Compiler ist für 60 Tage ab Installation voll funktionsfähig. Anschließend werden nur einige Optimierungs-Features gesperrt. Alles Weitere entspricht der Vollversion. Es gibt also insbesondere keine Beschränkung beim Speicher, von dem der Typ PIC24F128 ja eine Menge bietet. Selbstverständlich kann man MPLAB C30 auch mit kleineren 16-bit-PICs und dsPIC@s einsetzen, wenn man nicht so viel Leistung benötigt. Wer aber gerne auf die Vollversion upgraden möchte, der kann dies einfach tun, indem er einen Lizenzschlüssel von Microchip erwirbt. Und wer sich die in der Februar-Ausgabe vorgestellte Platine besorgt, der erhält sogar noch einmal einen Rabatt auf die Lizenz. Ein Gutschein hierfür liegt jeder Platine bei.

Der C30-Compiler scheint ein weniger bekanntes Plugin für MPLAB zu sein, sodass eine Kurzeinführung angemessen erscheint. Vorausgeschickt wird, dass jedes Projekt mit dem Erstellen des Quelltextes für das passende C-Programm beginnt und mit den Schritten Kompilierung, Fehlerbeseitigung und schließlich der Übertragung des Codes in den realen Ziel-Chip abgeschlossen wird. MPLAB C30 ist ein C-Compiler nach dem ANSI-Standard mit den entsprechenden Libraries für alle gebräuchlichen PIC-Controller inklusive der neuen dsPIC®-Serie. Der Compiler ist voll in die MPLAB-Umgebung integriert, was so genanntes „High-Level-Debugging“ auf Quelltextebene ermöglicht. Der Compiler bringt seinen eigenen Assembler, Linker und die Libraries zur Verarbeitung von hybridem Code aus C und Assembler mit. Auch aus hybridem Code wird schließlich nur eine einzige ausführbare Datei erstellt. Die MPLAB-C30-Libraries enthalten die Funktionen String-Manipulation, dynamische Speicher-Allokation, Daten-Konversion sowie Zeit- und Mathematik-Funktionen (trigonometrische, Exponential- und hyperbolische Funktionen). Das so genannte „Small Code Model“ des C30-Compilers nutzt eine besonders effiziente Form des Aufrufs von Instruktionen. Das entsprechende „Small Data Model“ erlaubt eine ähnliche Effizienz beim Zugriff auf Daten im SFR-Raum.

Obwohl der C30-Compiler in die MPLAB-Umgebung integriert ist, kann man ihn auch einzeln per Kommandozeile steuern. Passende Beispiele finden sich unter dem Menü-Eintrag „Open Workspace“ in MPLAB. Von dort navigiert

man zu MPLAB C30 → Examples → MPLAB Link30.  
Eine komplette C30-Anleitung ist auf der beigelegten CD  
im Ordner „Documentation“ enthalten.

### Proteus VSM

Der Simulator des Explorer-16-Projekts stammt von Labcenter Electronics. VSM (steht für **V**irtual **S**ystems **M**odeling) ist Teil der Proteus-Design-Suite, die außerdem noch die Pakete:

- ISIS (schematic capture);
- ProSPICE (mixed mode simulation);
- ARES PCB (design)

enthält.

Speziell für das Explorer-16-Projekt stellt Labcenter eine große Zahl von Designs für die Chips der Serien PIC10, PIC12, PIC14, PIC16, PIC18 und PIC24F kostenlos zur Verfügung.

Auf diesen Designs lässt sich jeder Code zum Laufen bringen, will man jedoch mit eigener Hardware experimentieren, so kommt man nicht ohne eine Proteus VSM Vollversion aus. Für alle ELEKTOR-Leser gibt's aber eine günstige Alternative – siehe dazu unter [www.labcenter.co.uk/products/elektoroffer.htm](http://www.labcenter.co.uk/products/elektoroffer.htm).

Nach Start von ISIS kommt man via Start → Labcenter → Proteus 7 Demonstration → ISIS 7 Demo zu Beispielen, mit denen man sich die Arbeitsweise der Software erschließen kann. Weitere Beispiele finden sich im Tutorials-Ordner.

Wie C30 ist auch VSM voll in MPLAB integriert. Man greift darauf via Debugger → Tool zu. Wenn Sie es nicht erwarten können, diese Software in Aktion zu sehen, dann folgen Sie den acht Schritten der Kurzanleitung im Kasten. Für diese eindrucksvolle Demonstration muss allerdings zuvor eine Datei mit der Bezeichnung „Demo1.zip“ aus dem Explorer-16-Bereich der ELEKTOR-Webseite herunter geladen werden. Genauso funktioniert das auch mit den vorinstallierten Virtual Evaluation Boards. Mehr darüber unter [4].

### Ausblick

Wir unterstellen einmal vorsichtig, dass Sie mit dem Ausprobieren und der Einarbeitung in die auf der beigelegten CD enthaltenen Software locker einen Monat keine Längeweile mehr haben werden ;)

In der Februar-Ausgabe geht es dann mit der Explorer-16-Hardware weiter. Eine erste Anwendung wird vorgestellt, die etwas mit Sprachreproduktion zu tun hat – soviel sei verraten. Diese Anwendung zeigt, wie man mit C30 und MPLAB zur funktionierenden Software kommt. Und anhand von VSM erkennt man, dass Simulation heute zu einem nahezu unverzichtbaren Teil der Programmierung von Mikrocontrollern geworden ist.

(060280-1)

## Weblinks

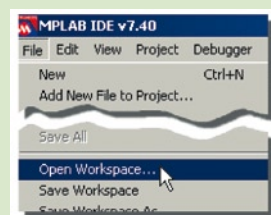
- [1] Microchip PIC24FJ128GA Family Data Sheet:  
[ww1.microchip.com/downloads/en/DeviceDoc/39747C.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/39747C.pdf)
- [2] Microchip High-Performance PIC24 Microcontroller Übersicht:  
[ww1.microchip.com/downloads/en/DeviceDoc/39754b.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/39754b.pdf)
- [3] PIC18F to PIC24F Migration, eine Übersicht:  
[ww1.microchip.com/downloads/en/DeviceDoc/39764a.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/39764a.pdf)
- [4] Proteus VSM:  
[www.labcenter.co.uk/products/vsm\\_overview.htm](http://www.labcenter.co.uk/products/vsm_overview.htm)

## Kurzanleitung zu Proteus VSM unter MPLAB

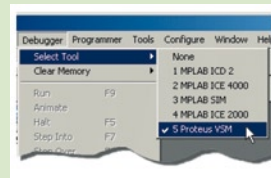
Diese kurze Anleitung zeigt, wie man eine VSM-Simulation unter der Kontrolle von MPLAB vornimmt:

1. Download der Datei „Demo1.zip“ von der Explorer-16-Seite unter [www.elektor.de](http://www.elektor.de). Den entpackten Demo1-Ordner im Pfad: c:\program files\microchip\mplab c30\examples speichern.

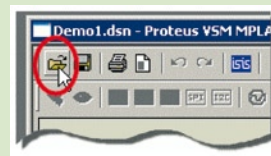
2. MPLAB IDE starten, in das File-Menü gehen, den Eintrag „Open Workspace“ auswählen, in den Demo1-Ordner wechseln und „Demo1.mcw“ öffnen.



3. Im Debugger-Menü von MPLAB den Eintrag „Select Tool“ auswählen und dann Proteus VSM. MPLAB ist nun konfiguriert, Proteus als Standard-Debugger zu verwenden.

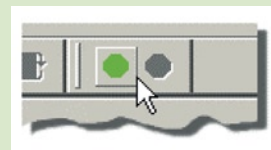


4. Nun sollte sich Proteus VSM innerhalb von MPLAB zeigen. Mit dem Open-Icon wird die Schaltung „Demo1“ via Datei-Auswahlbox ausgewählt.

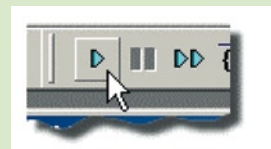


5. Nun den Eintrag „Build All“ aus dem Debugger-Menü auswählen.

6. Da jetzt Schaltung und Projekt zur Verfügung stehen, kann simuliert werden: Die grüne Schaltfläche oben in der MPLAB IDE verbindet die Proteus-Simulation mit MPLAB.



7. An diesem Punkt verharrt die Simulation bei der Ablaufzeit Null. Die Simulation wird durch einen Klick auf „Play“ (rechts oben in MPLAB) gestartet. Der Code des Programms wird simulierend ausgeführt und der VSM-Viewer zeigt die Resultate der simulierten Schaltung.



8. Die rote Schaltfläche oben in MPLAB trennt den VSM-Viewer wieder von MPLAB und stoppt die Simulation. Eine detailliertere Demonstration folgt in einem späteren Artikel.

