

---

---

## クイックスタート

---

---

### はじめに

本書は MPLAB® IDE バージョン 6.1x のインストールからデバックや PICmicro® MCU に書き込む方法までの概要を説明します。デバック機能についてはサンプルコードを使います。それらに関連する機能についても説明します。

### 概要

本書の最初の章では MPLAB IDE をパソコンにインストールする方法について説明します。次の章ではプロジェクトを作成して、MPLAB のデバック機能に慣れるためのステップごとのチュートリアルです。MPLAB でビルドしたりテストしたりする方法を覚えた後、最後の章では MPLAB でほかの開発ツールを使ったり、MPLAB をカスタマイズします。

- MPLAB IDE を使う準備
- シンプルなプロジェクトのデバック
  - プロジェクトの作成
  - シミュレータの実行
  - デバック
- 次に何を行う？
  - デバイスに書き込み
  - シミュレータのオプション
  - MPLAB ICD 2
  - MPLAB IDE のオンライン・ヘルプ
  - ワークスペースとプロジェクト

### MPLAB IDE を使う準備

MPLAB IDE は Windows® 上で動作する PICmicro マイクロコントローラ (MCU) や dsPIC™ デジタルシグナルコントローラ用の開発環境です。MPLAB IDE を使って次のことを行えます：

- ビルドイン・エディタを使ってソース・コードを作る。
- いろいろな言語ツールを使って、ソース・コードをアセンブル、コンパイル、リンクする。アセンブラ、リンカ、ライブラリアンは MPLAB IDE に入っています。マイクロチップといくつかのサードパーティの C コンパイラもサポートしています。詳しくはリリースノートや README ファイルを参照してください。
- シミュレータや MPLAB ICE 2000 エミュレータや MPLAB ICD 2 インサーキット・デバッガでプログラムの流れを見ながら論理的な動作をデバックできる。サードパーティのエミュレータもサポートするかもしれません。詳しくはリリースノートや README ファイルを参照してください。
- シミュレータやエミュレータを使ってタイミング測定ができる。
- WATCH ウィンドウで変数の内容を見る。

# MPLAB<sup>®</sup> IDE v6.xx Quick Start

- PICSTART<sup>®</sup> Plus や PRO MATE<sup>®</sup> II などを使ってデバイスにファームウェアを書き込める。サードパーティのライターもサポートするかもしれませんが。詳しくはリリースノートや **Readme** ファイルを参照してください。
- MPLAB IDE オンライン・ヘルプで疑問点を調べることができます。

## サポートしているシステム

下記は MPLAB IDE が動作する最小システム構成です：

- PC 互換 Pentium<sup>®</sup> クラスのシステム
- Microsoft Windows 98, Windows 2000 SP2, Windows NT<sup>®</sup> SP6, Windows ME, Windows XP
- RAM 16 MB(32 MB 推奨)
- ハードディスク空き 45MB
- Microsoft Internet Explorer 5.0 以降がインストールされていること

## MPLAB IDE のインストールとアンインストール

MPLAB IDE をインストールするには：

- OS によっては、Administrator 権限でログインする必要があります。
- CD-ROM からのインストールの場合は、ドライブに CD-ROM を入れる。その後はメニューにしたがってください。もしメニューが現れない場合は、エクスプローラを使って MPLAB インストール実行ファイルを探して実行してください。
- マイクロチップの Web サイトからダウンロードした場合は、ダウンロードした実行ファイルをダブルクリックしてインストールしてください。

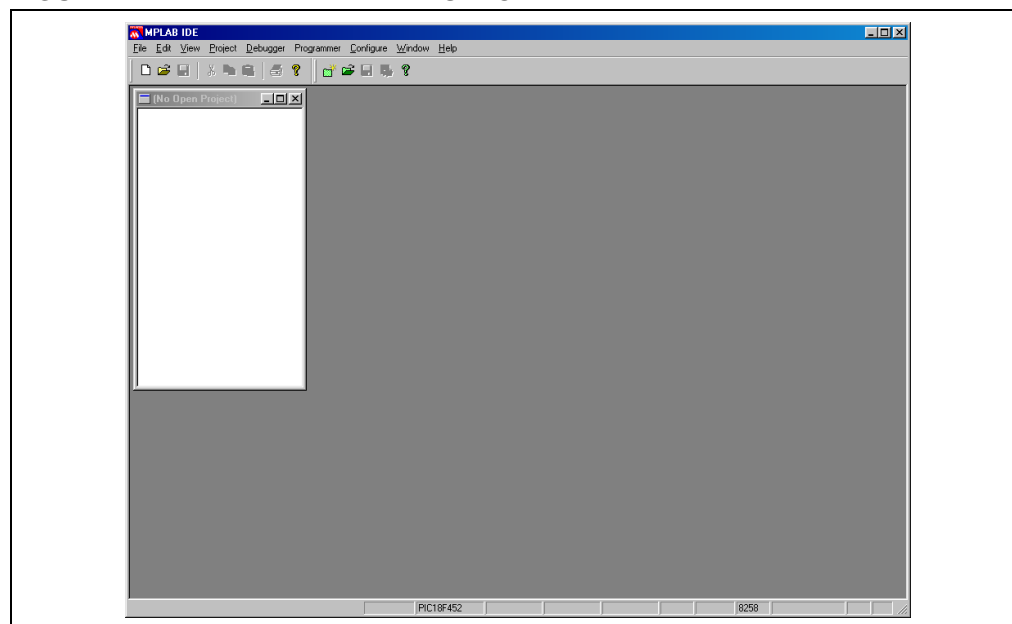
MPLAB IDE をアンインストールするには：

- **Start>Programs>Microchip MPLAB IDE** メニューから UNWISE32 を選択するか
- MPLAB IDE をインストールしたフォルダの中の unwise32.exe を実行する。

## MPLAB IDE を実行する

MPLAB IDE を実行開始するには **Start>Programs>Microchip MPLAB IDE>MPLAB IDE** を選択します。最初にスプラッシュ画面が表示され、その後に MPLAB IDE デスクトップが表示されます。

FIGURE 1: MPLAB IDE DESKTOP



## シンプルなプロジェクトのデバッグ

### プロジェクトの作成

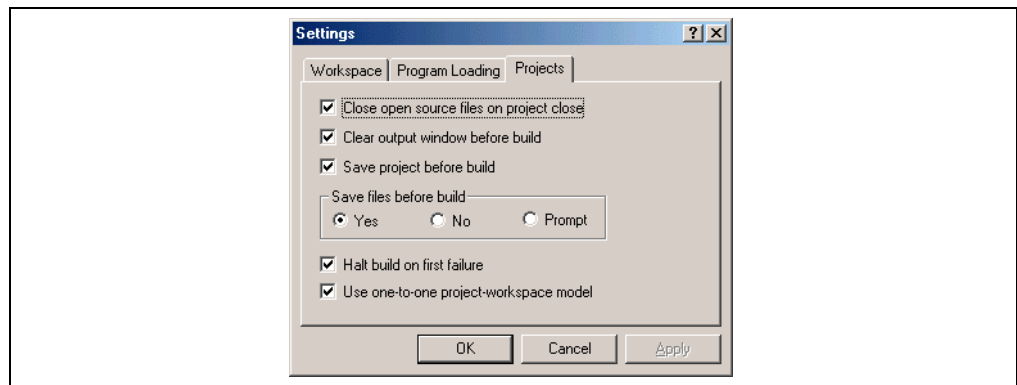
プロジェクトを使ってコード開発ができます。今回はビルドする全ファイルはプロジェクトに入ります。一つのプロジェクトで使える言語ツールは一種類のみです。プロジェクトは HEX ファイルを出力します。デバッグでも HEX ファイルはメモリにロードされて使われます。ほかにもソース・コードでのシンボル定義やメモリ上のアドレスが入ったファイルが作られます。

### プロジェクトの設定

まずワークスペースとプロジェクトの設定を確認します。今回はワークスペースを意識する必要がない **one-to-one project-workspace model** を使います。本書の最終章でワークスペースとプロジェクトについて説明します。

**Configure>Settings** ダイアログの **Projects** タブをクリックします。全項目をチェックして、OK ボタンをクリックします。

**FIGURE 2: SETTINGS: PROJECTS TAB**



# MPLAB® IDE v6.xx Quick Start

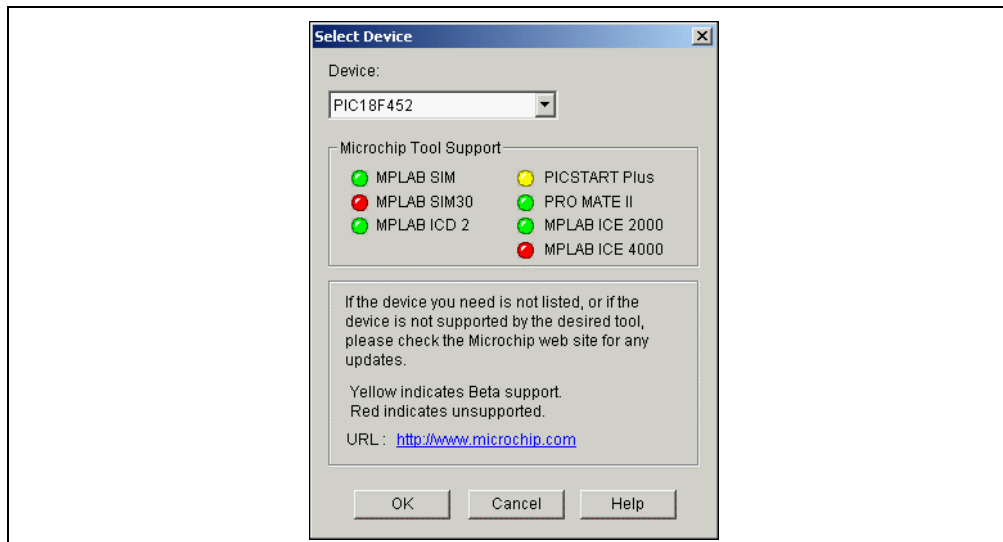
---

## デバイスの選択

MPLAB IDE で使うデバイスを選択します。今回は PIC18F452 を使います。

1. Configure>Select Device を選択します。

**FIGURE 3: SELECT DEVICE DIALOG**



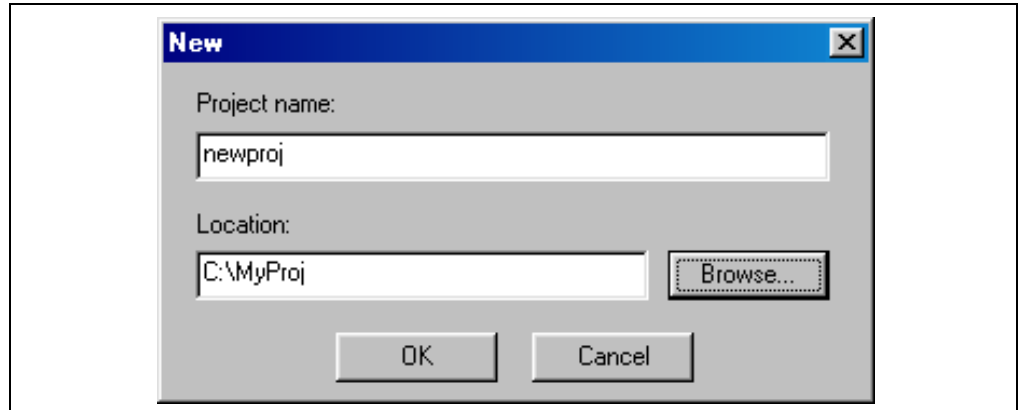
2. Device の選択メニューから **PIC18F452** を選択します。選択したデバイスをサポートしているツールは緑、サポートしていないツールは赤、β 版などでサポートしているツールは黄で表しています。
3. **OK** ボタンをクリックします。

## 新規プロジェクトの作成

次のようにプロジェクトを新規作成できます。

1. **Windows** のエクスプローラを使って、プロジェクトを入れるフォルダを作ります (たとえば **C:\Myproj**)。
2. MPLAB IDE の **Project>New** を選択して、**New** ダイアログを開きます (Figure 4)。
3. プロジェクト名を入力します (たとえば **newproj**)。
4. ステップ1で作ったフォルダを **Browse** ボタンを使うかタイプして入力します。
5. プロジェクト名と場所が正しければ **OK** をクリックします。

**FIGURE 4: NEW DIALOG**



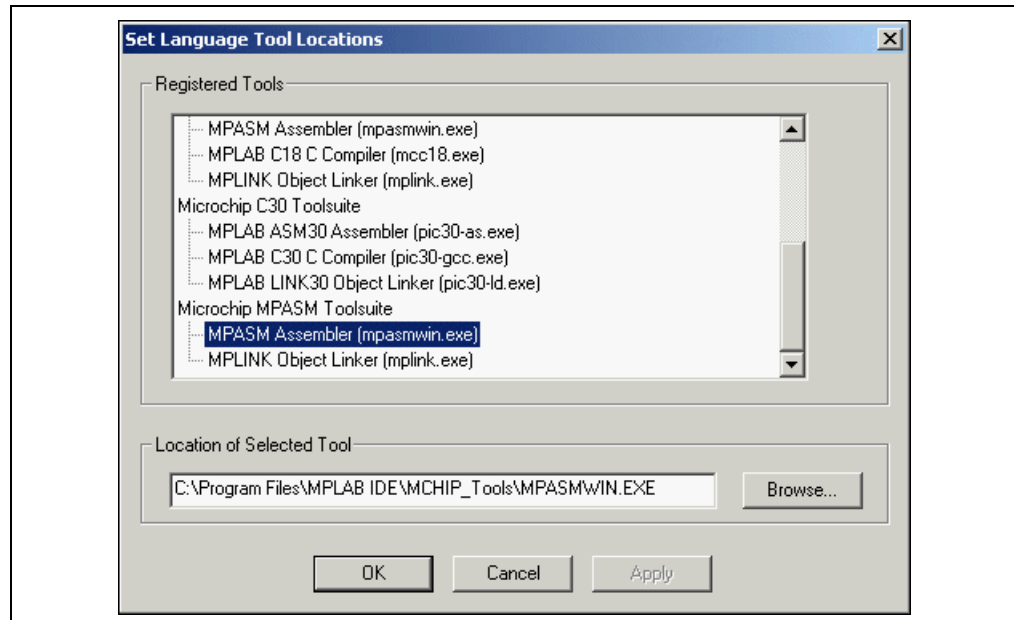
**Note:** ロングファイルネームを MPLAB IDE はサポートしていますが、サポートしていない言語ツールもあります。ソースファイル名やパスでロングファイルネームを使う場合は、言語ツールがサポートしていることをチェックしてください。

# MPLAB® IDE v6.xx Quick Start

言語ツールの場所

**Project > Set Language Tool Locations** で Microchip ツールスイートの場所を確認します。MPASM Assembler (mpasmwin.exe) をクリックして、Location of Selected Tool テキストボックスで実行ファイルのパスが下図のようになっていなければ **Browse** ボタンを使って mpasmwin.exe の場所を指定してください。

**FIGURE 5: SET LANGUAGE TOOL LOCATION DIALOG**



今回必要なのは MPASM アセンブラだけです。他の言語ツールを使うプロジェクトの場合は各言語ツールの場所を確認してください。

必要なツールの設定を確認後 **OK** ボタンをクリックします。

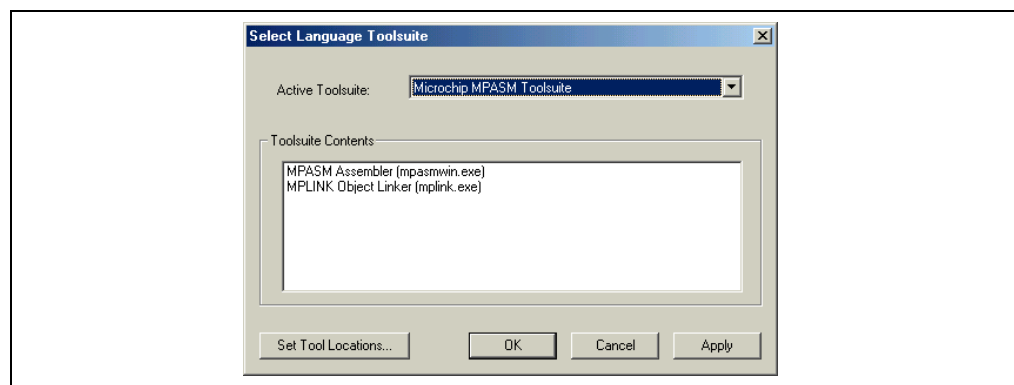
ツールスイートの設定

ソースコードを編集する前に言語ツールスイートを設定してください。

今回は Microchip ツールスイートを選択します：

1. **Project > Set Language Toolsuite** を選択
2. **Active Toolsuite** に Microchip MPASM Toolsuite を選択します。PICmicro の言語ツールが **Toolsuite Contents** に表示されます。
3. **OK** ボタンをクリックします。

**FIGURE 6: SELECT LANGUAGE TOOLSUITE DIALOG**



## ソースコードの作成

MPLAB IDE のエディタを使ってコードを編集できます。

**File>New** を選択後、ワークスペースに編集ウィンドウが開きます。下記のサンプルコードをタイプしたり、本書のファイルからコピーしてペーストしてください。

```
        title "PIC18F452 counting program"
        list p=18f452,f=inhx32
        #include <p18f452.inc>

COUNT    equ    0x00
DVAR      equ    0x01
DVAR2     equ    0x02

        org    00h                ;reset vector
        goto   Start

        org    1Ch
Start
        clrf   WREG                ;clear W register
        movwf  PORTC              ;clear PORTC
        movwf  TRISC              ;config PORTC as outputs

Init
        clrf   COUNT              ;clr count

IncCount
        incf   COUNT,F            ;increment count
        movf   COUNT,W            ;
        movwf  PORTC              ;display on port c

        call   Delay              ;wait
        goto   IncCount           ;loop

Delay
        movlw  0xFF                ;set delay loop
        movwf  DVAR2              ;

D0
        movwf  DVAR                ;reset inner loop

D1
        decfsz DVAR,F
        goto   D1

        decfsz DVAR2,F
        goto   D0
        return
end
```

コードを入力し終わったら、**File>Save** を選択して cnt452.asm というファイル名でプロジェクトフォルダに保存します。

保存した後、コードは色付けされて読みやすくなります。この色付けはカスタマイズできます。詳しくは [Help>MPLAB Editor Help](#) を参照してください。

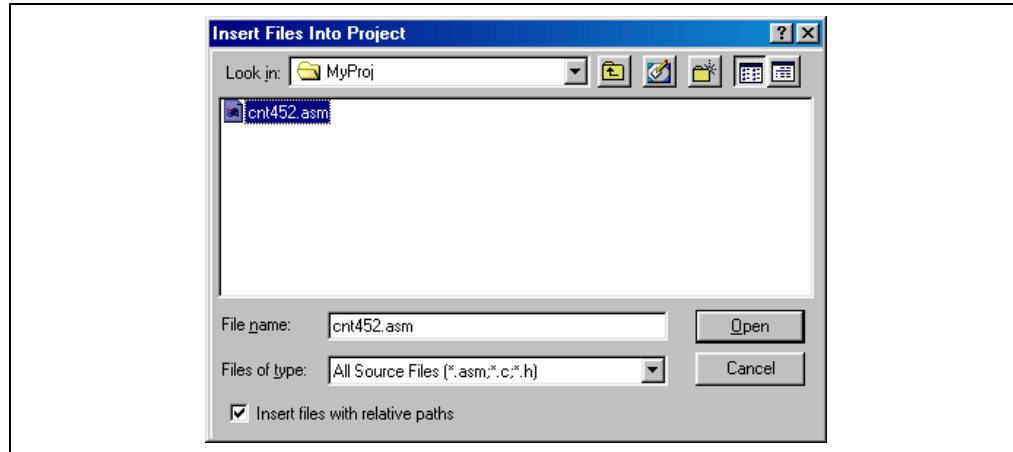
# MPLAB® IDE v6.xx Quick Start

ソースファイルの追加

作成したソースファイルをプロジェクトに追加します。

1. **Project>Add Files to Project** を選択します

**FIGURE 7: SELECT INPUT FILE DIALOG**

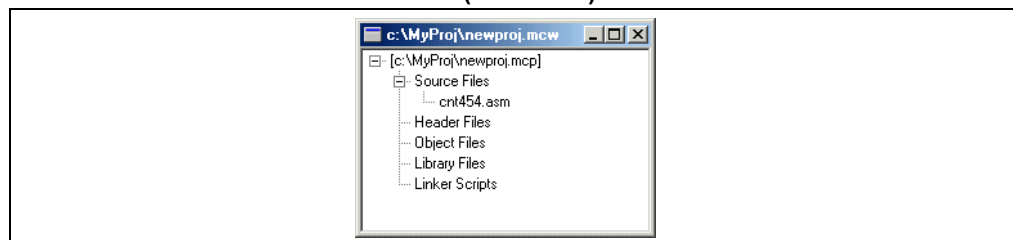


2. 作成したファイル (cnt452.asm) を選択します
3. **Open** ボタンをクリックします

**Note:** 言語ツールによって名前の付け方が違います。たとえば、MPASM アセンブラのソースファイルの場合は拡張子は .asm にしてください。ロングファイルネームをサポートしていない言語ツールもあります。

追加したファイルは **Source Files** の下に表示されます (Figure 8)。もし **Unclassifiable** の下に表示されたら、**Project>Set Language Toolsuite** の **Active Toolsuite** のところで **Microchip Toolsuite** を選択しているかを確認してください。違っていたら、それを選択して **OK** ボタンをクリックすれば **Source Files** の下に表示されるはずです。

**FIGURE 8: PROJECT PANE (WINDOW)**



**Project>Save** を選択してプロジェクトを保存します。

**TIP:** 右クリックでもファイルを追加したり保存したりできます。プロジェクトのパス名やファイル名の上で右クリックする場合と **Source Files** の上で右クリックする場合はメニューが違います。

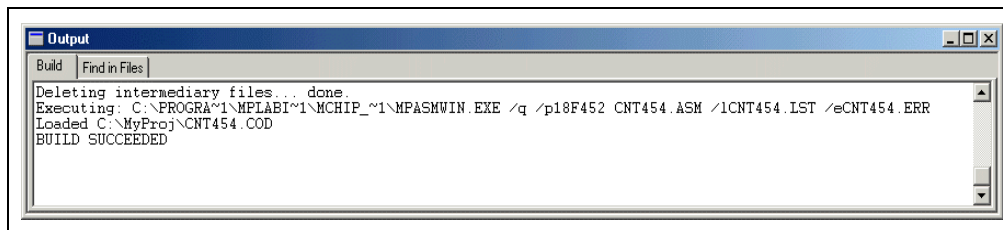


## プロジェクトのビルド

ソースファイルを追加したので、プロジェクトをビルドできます。ビルドすると選択した言語ツールでソースコードがコンパイルされます。

プロジェクトをビルドするには **Project>Build All** を選択します。ソースファイルはエラーなしでアセンブルされるはずですが、

**FIGURE 9: OUTPUT WINDOW**



エラーがあった場合は、下記の項目をチェックして再度ビルドしてください：

- **Editor** ウィンドウで編集したコードのタイプミスなどを確認します。Output ウィンドウでアセンブラがエラーを報告している場合、そのエラーメッセージをダブルクリックすれば、ソースコードウィンドウでエラーが報告された行が黄色の矢印で示されます。
- **MPASM** アセンブラが選択されていることを確認します。Project>Set Language Tool Locations を選択して MPASM Assembler (mpasmwin.exe) をクリックするとその場所が表示されます。場所が正しければ **Cancel** ボタンをクリックします。正しくなければ修正して **OK** ボタンをクリックします。

ビルドしてエラーがなければ、言語ツールで生成されたデバッグファイル (\*.cod または \*.cof) がロードされます。それによってソースコードでデバッグしたり、Watch ウィンドウで変数の内容を見れます。

## プロジェクトの利点

ソースファイルが一つだけの 경우에는、プロジェクトは必要なさそうに見えます。しかし、複数のソースファイルをアセンブル / コンパイルしてリンクする場合は、プロジェクトがないと不便です。

また、Microchip ツールスイートでリンクすると、アセンブラだけを使う場合よりもデバッグ用の情報を多く生成します。たとえば、ローカル変数についての情報などもリンクは生成します。

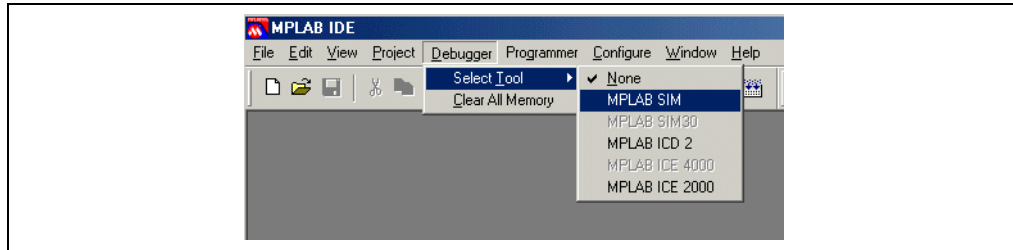
# MPLAB® IDE v6.xx Quick Start

## シミュレータの実行

プロジェクトをビルドしたら、考えたとおりに動作するかをデバッグ・ツールで確認できます。今回はシミュレータを使います。

今回使っている PIC18F452 は MPLAB SIM シミュレータでサポートされています。Debugger>Select Tool>MPLAB SIM で MPLAB SIM simulator を選択してください。

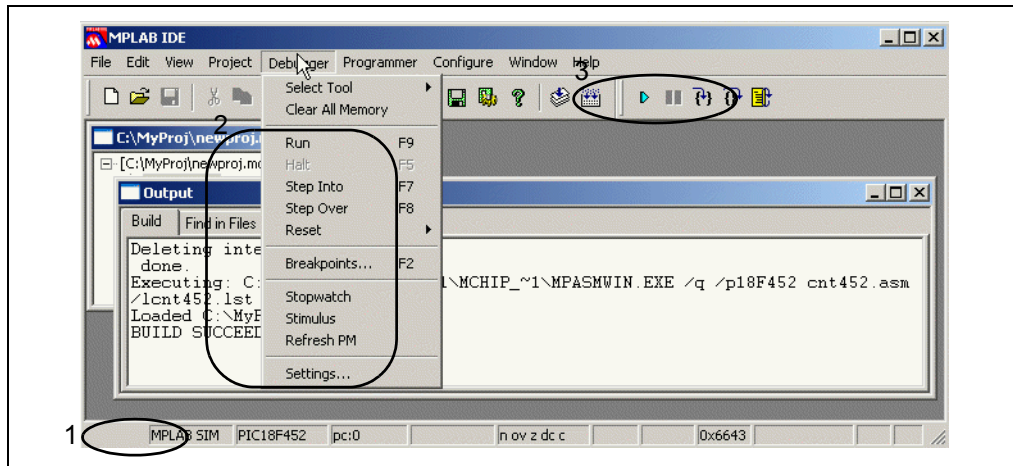
FIGURE 10: SELECT SIMULATOR AS THE DEBUGGER



MPLAB SIM を選択後、次の変化があるはずですが：

1. ステータス・バー (MPLAB IDE ウィンドウの下辺) に MPLAB SIM と表示される
2. Debugger メニューの項目が増える
3. ツールバーにデバッグ用のアイコンが表示される

FIGURE 11: MPLAB IDE DESKTOP WITH MPLAB SIM AS DEBUGGER



**TIP:** カーソルをツールバーのアイコンの上に置くと、短い機能説明が表示されます。

このシミュレータはパソコン上で PICmicro MCU の命令の疑似動作を行うソフトウェアです。実行時間はリアルタイムではなく、パソコンの処理スピードやコードの複雑さや OS のオーバーヘッドや他のタスクをいくつ実行しているかなどによって変わります。しかし、実行時間を表示することはできません。

MPLAB ICE 2000 や MPLAB ICD 2 は外付けのハードウェア・ツールで、実際の PCB 上でリアルタイムにコードを実行できます。

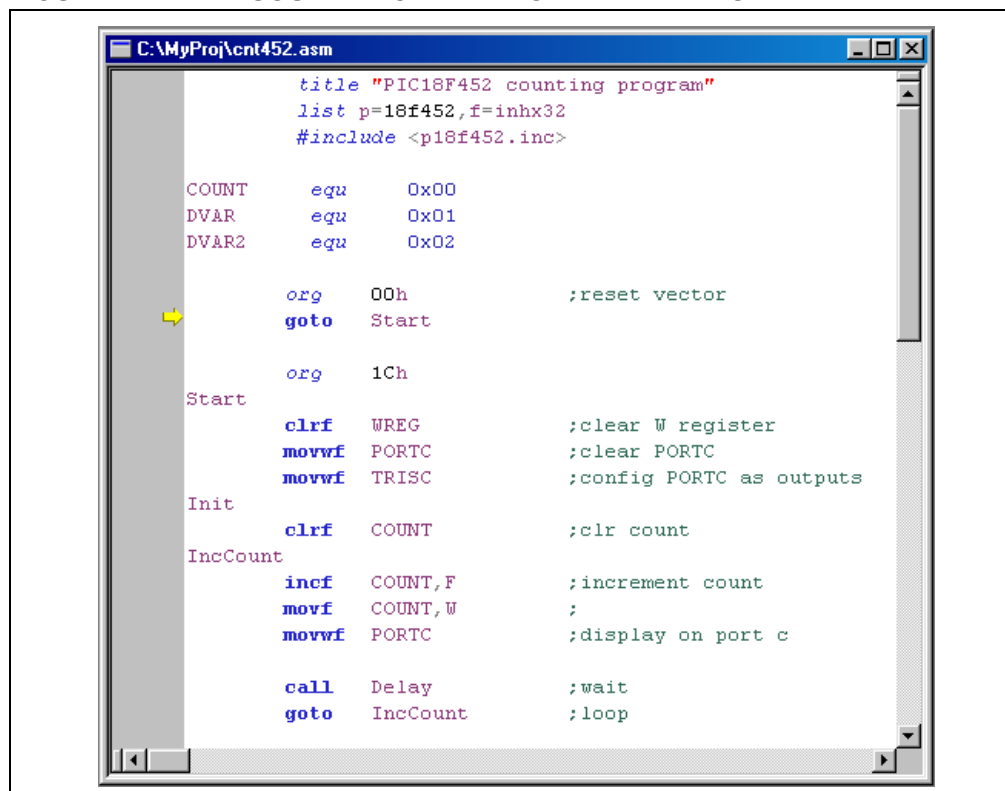
## デバッグ

準備ができれば、コードを実行させてデバッグできます。

コードの実行

まず **Debugger>Reset** を選択します。黄色の矢印が実行する行を指します。

**FIGURE 12: SOURCE CODE WINDOW - AFTER RESET**



コードを実行するには **Debugger>Run** を選択します。ステータスバーに **Running...** と表示されます。

コードの実行を停止するには **Debugger>Halt** を選択します。黄色の矢印は停止した行を指します。

**Debugger>Step Into** を選択すれば、シングルステップを実行できます。実行すると矢印が指している行を実行後、矢印は次に実行される行へ移ります。

**TIP:** メニュー項目を選択する代わりに、ツールバーのアイコンやメニュー項目の隣に書かれているショートカットキーも使えます。ステップを繰り返す場合などで便利です。

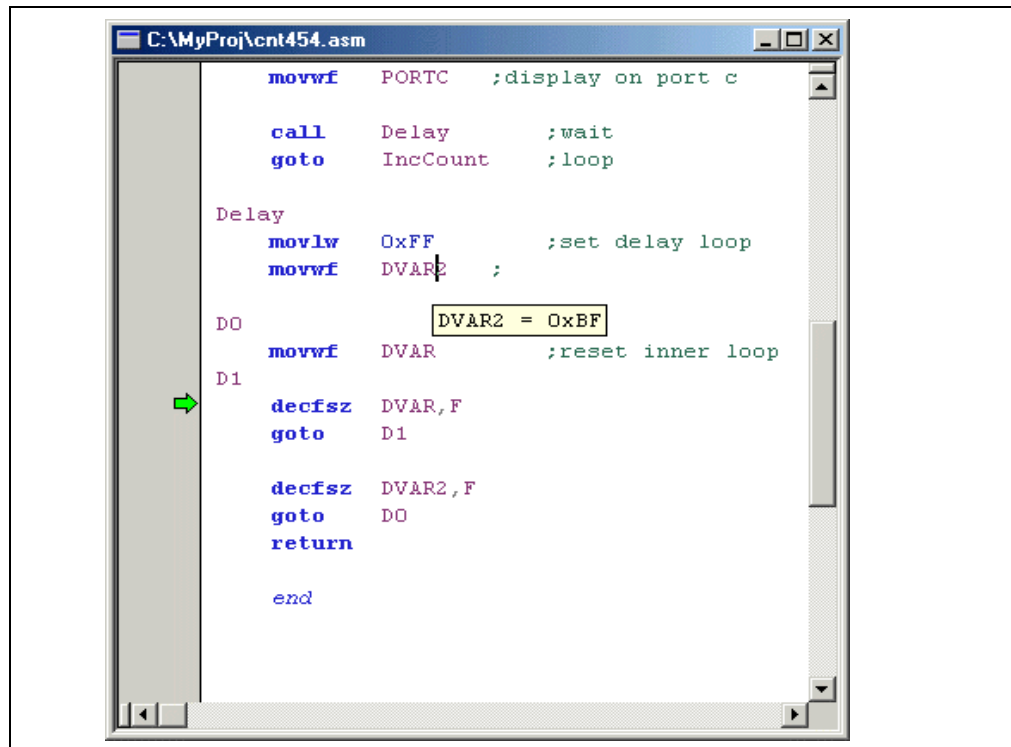
# MPLAB® IDE v6.xx Quick Start

## 変数の内容表示

マウスのカーソルを変数の上に置けば、小さいウィンドウが現れて内容を表示します。

**Note:** Microchip ツールスイートでは、この機能でローカル変数を表示できませんが、ほかのツールスイートでは表示できないかもしれません。

FIGURE 13: MOUSE OVER VARIABLE

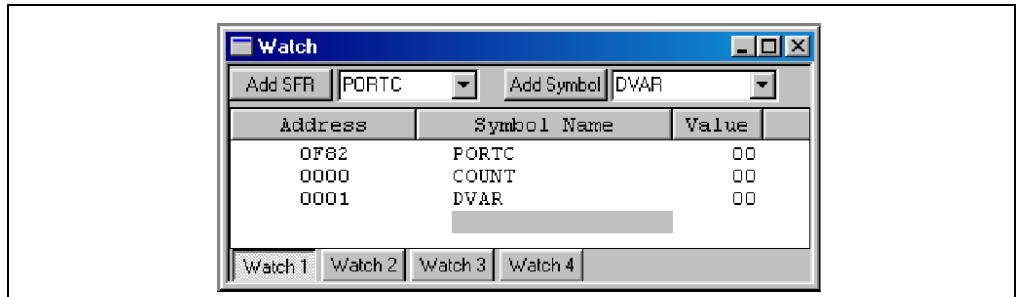


## WATCH ウィンドウ

ある変数の動きをしばらく見ていたい場合は、Watch ウィンドウが便利です。Watch ウィンドウは View メニューのなかにあります。

1. View>Watch を選択して Watch ウィンドウを新規作成します。
2. SFR の選択メニューから PORTC を選択します。プルダウンアイコンを選択後 PORTC と入力したほうが早く入力できます。Add SFR ボタンをクリックして Watch ウィンドウに追加します。
3. シンボルの選択メニューから COUNT を選択します。Add Symbol ボタンをクリックして Watch ウィンドウに追加します。
4. シンボルの選択メニューから DVAR を選択します。Add Symbol ボタンをクリックして Watch ウィンドウに追加します。

FIGURE 14: WATCH WINDOW WITH SYMBOLS

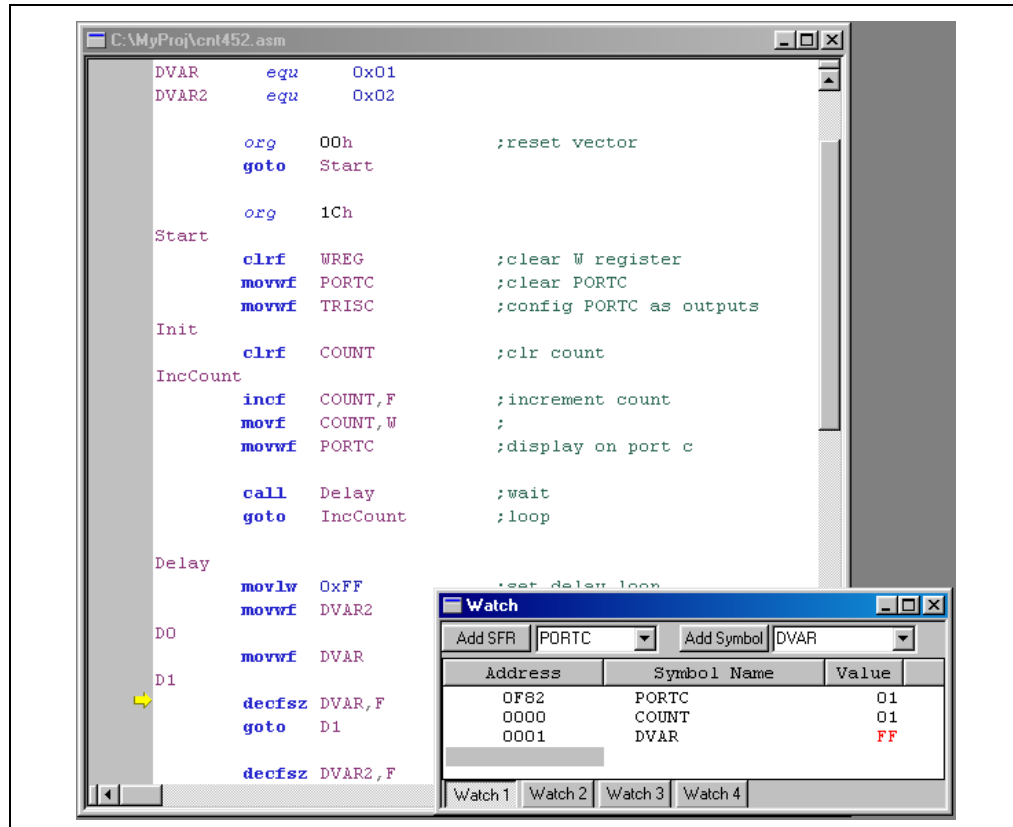


Watch ウィンドウにシンボルが三つ入ったはずですが、左からシンボルのファイルアドレス、シンボル名、シンボルの内容が表示されます。コードをステップさせるとシンボルの内容が変化します。

1. Debugger>Reset してリセットします。
2. 次の行になるまで Debugger>Step Into を選択、またはツールバーの同機能のアイコンをクリックします：

```
incf  COUNT,F          ;increment count
```
3. ステップをもう 1 回実行すると COUNT の内容が 0 から 1 に変わります。
4. ステップをもう 2 回実行すると PORTC の内容が 0 から 1 に変わります。
5. ステップをもう 4 回実行すると DVAR の内容が FF に変わります。Watch ウィンドウのなかの値は、前回停止した状態から変化した場合は赤くなり、変化していない場合は黒くなります。

FIGURE 15: STEPPING THROUGH CODE



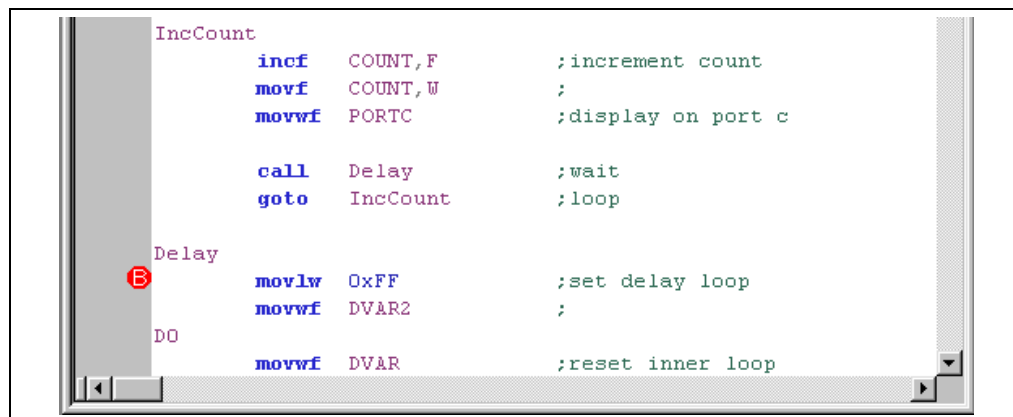
## ブレークポイントの設定

ある番地までコードを実行させて停止したい場合は、ブレークポイントが便利です。ここではブレークポイントの簡単な使い方を説明します。Breakpoint ダイアログについてはオンラインヘルプメニューで説明しています。

1. **Debugger>Reset** を選択してデバイスをリセットします。
2. 次の行をさがして、右クリックします:  

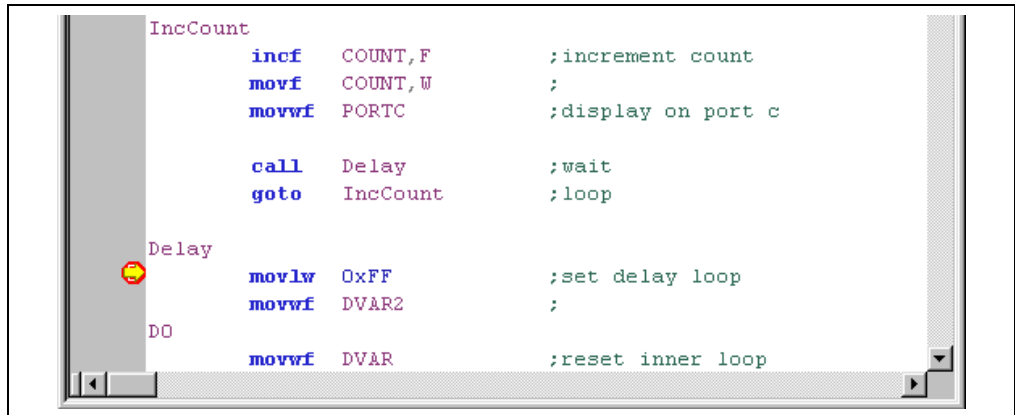
```
movlw 0xFF          ;set delay loop
```
3. ポップアップメニューが表示されます。そのなかの **Set Break Point** を選択します。その行にブレークポイントのマークが表示されます。

FIGURE 16: SOURCE CODE WINDOW - SET BREAKPOINT



4. **Debugger>Run** を選択してコードを実行します。実行後、設定したブレークポイントで停止するはずですが。

**FIGURE 17: SOURCE CODE WINDOW - BREAKPOINT HALT**

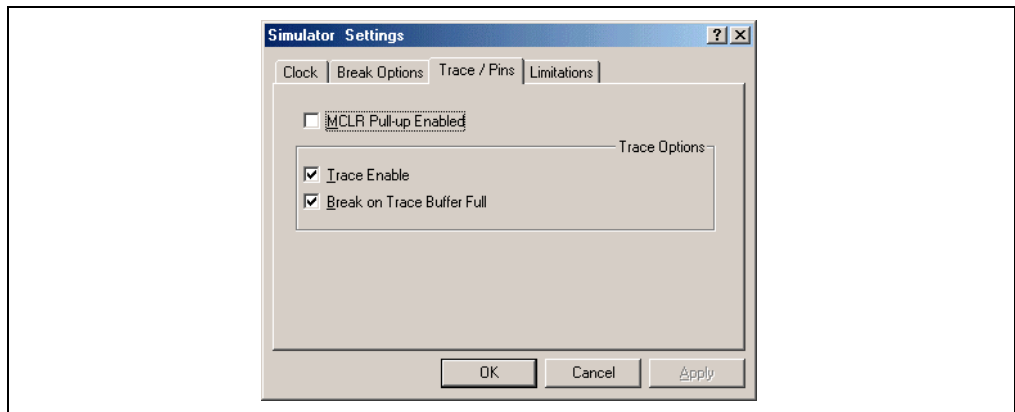


**Note:** ブレークポイントを設定した行が実行されない場合は **Debugger>Halt** などを使って停止するまで実行し続けます。

トレース

トレースでコードの実行を記録できます。 **Debugger>Settings** ダイアログの **Trace/Pins** パネルでトレースをイネーブします。

**FIGURE 18: SIMULATOR TRACE ENABLE**



トレースのデータ収集する方法を決めるチェックボックスが二つあります。上のチェックボックスがチェックされている場合は、シミュレータはコード実行中に停止するまでデータ収集します。つまりブレークポイントや手動でコードを停止するまでデータを収集します。最後に実行したサイクルから数えて **8192** サイクル分のデータが収集されます。これでブレークポイントで停止するまでのコードの実行を見ることができます。

下のチェックボックスもチェックされている場合は、**8192** サイクルのデータを収集したら実行が停止します。これは実行開始してからのコードの実行を見るときに便利です。

**View>Simulator Trace** を選択します。シミュレータ・トレースでは、各サイクルでのタイムスタンプやファイルレジスタにリードやライトされたデータなどの情報を見ることができます。

FIGURE 19: SIMULATOR TRACE DISPLAY

Addr	Op	Label	Instruction	SA	SD	DA	DD	Time
0000	EFOE		GOTO 0x00001c	----	--	----	--	0.000002000:
001C	6AE8	Start	CLRF 0xfe8, 0	----	--	0FE8	00	0.000003000:
001E	6E82		MOVWF 0xf82, 0	----	--	0F82	00	0.000004000:
0020	6E94		MOVWF 0xf94, 0	----	--	0F94	FF	0.000005000:
0022	6A00	Init	CLRF 0, 0	----	--	0000	00	0.000006000:
0024	2A00	IncCount	INCF 0, 0x1, 0	0000	00	0000	00	0.000007000:
0026	5000		MOVF 0, 0, 0	0000	01	0FE8	01	0.000008000:
0028	6E82		MOVWF 0xf82, 0	----	--	0F82	00	0.000009000:
002A	EC19		CALL 0x000032, 0	----	--	----	--	0.000011000:
0032	0EFF	Delay	MOVLW 0xff	----	--	0FE8	FF	0.000012000:
0034	6E02		MOVWF 0x2, 0	----	--	0002	00	0.000013000:
0036	6E01	D0	MOVWF 0x1, 0	----	--	0001	00	0.000014000:
0038	2E01	D1	DECFSZ 0x1, 0x1, 0	0001	FF	0001	FF	0.000015000:
003A	EF1C		GOTO 0x000038	----	--	----	--	0.000017000:
0038	2E01	D1	DECFSZ 0x1, 0x1, 0	0001	FE	0001	FE	0.000018000:

一番左の縦列からプログラムカウンタのアドレス (Addr)、命令のマシン語 (Op)、ラベル (Label)、逆アセンブルされた命令 (Disassembly) が表示されます。その右側にはファイルレジスタにリードやライトされたデータが表示されます:

- SA - Source Address, リードされたレジスタのアドレス
- SD - Source Data, レジスタからリードされたデータ
- DA - Destination Address, ライトされるレジスタのアドレス
- DD - Destination Data, レジスタへライトするデータ

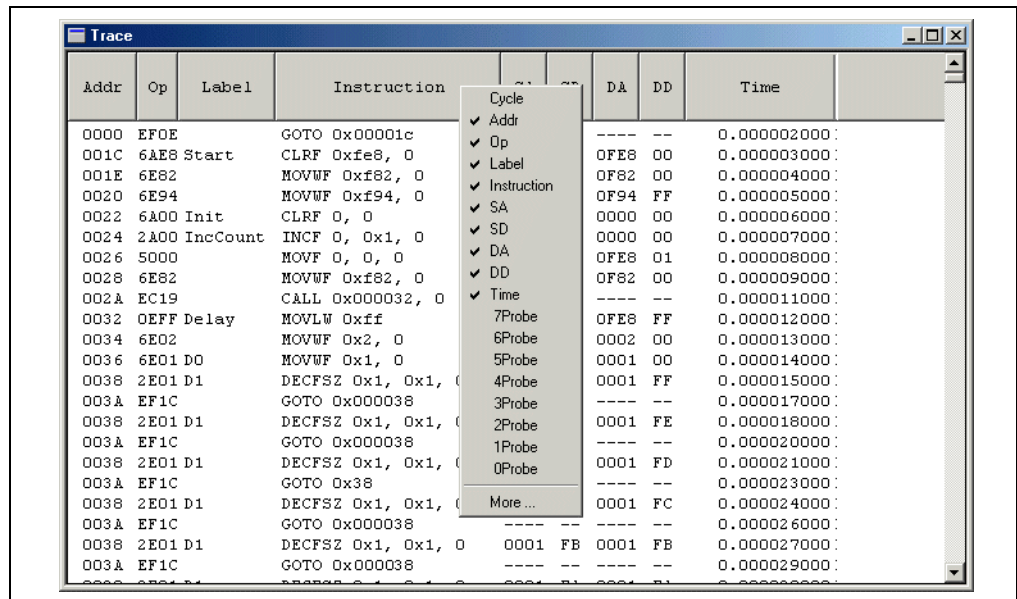
ファイルレジスタをアクセスしない命令ではダッシュ (-) になります。

一番右の縦列はタイムスタンプです。実行時間を見ることができます。実行時間は **Debugger>Settings** ダイアログの **Clock** パネルで入力したクロック周波数で計算しています。



各項目名を表示している横列の上にカーソルを置いて、右クリックするとポップアップメニューが現れます。

**FIGURE 20: SIMULATOR TRACE CONFIGURE**



チェックされた項目はトレースウィンドウに表示されます。チェックを外すと表示されなくなります。一番下に7Probeとか6Probeなどと表示されている項目はシミュレータでは使えません(MPLAB ICE 2000 エミュレータの機能です)。

# MPLAB<sup>®</sup> IDE v6.xx Quick Start

## 次に何を行う？

MPLAB IDE のヘルプメニューにはシミュレータのオンライン・チュートリアルがあります。MPLAB IDE 関連のドキュメントや MPLAB IDE のオンライン・ヘルプでもいろいろな機能について説明しているので参照してください。そのなかでいくつかの機能について説明します。

### デバイスに書き込み

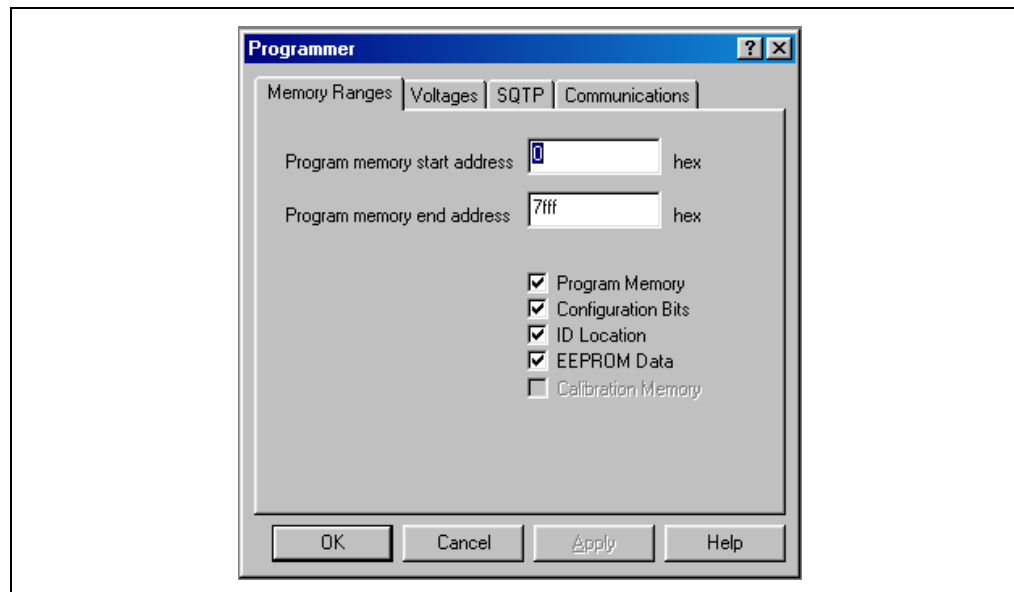
デバッグが終わったら、デバイスにコードを書き込みます。次のライタのどれかと PIC18F452 デバイスを持っていれば、デバイスにコードを書き込めます：

- MPLAB ICD 2
- PICSTART Plus 開発用プログラマー (ファームウェアは v3.00 以降)
- PRO MATE II デバイス・プログラマー

次のようにライタを選択して使う準備をします：

1. **Programmer>Select Programmer** を選択してライタを選びます。Programmer メニューの項目が変わり、ツールバーにアイコンが追加されます。
2. パソコンとの接続を開始します。PICSTART Plus か PRO MATE II の場合は **Programmer>Enable Programmer**、MPLAB ICD 2 の場合は **Programmer>Connect** を選択します。
3. **Programmer>Settings** ダイアログを使ってライタの設定をします。今回はデフォルトの設定のまま使います。

FIGURE 21: PRO MATE II SETTINGS DIALOG



4. コンフィグレーションビットを設定します。今回はデフォルトのまま構いません。ソースコード上にコンフィグレーションビットの設定を記述することもできます (推奨)。また、**Configure>Configuration Bits** を選択して Configuration Bits ウィンドウで手動で設定することもできます。

**TIP:** ソースコード上でコンフィグレーションビットの設定を記述すれば、デバッガの動作に反映します。たとえばソースコードでオシレータの設定を記述すれば、デバッガはそのオシレータの設定を使います。

5. MPLAB IDE にロードされている情報をデバイスに書き込むには **Programmer>Program** を選択します。処理状態はステータスバーに示されます。処理結果は Output ウィンドウの PRO MATE パネルに次のように表示されます。

```
PRO MATE Error Log File
Programming
31-May-2002. 13:06:19
Device Type: PIC18F452
```

Programming/Verification Successful!

書き込みするときには自動的にベリファイも行います。**Programmer>Verify** を選択すれば、再度ベリファイを行えます。

## シミュレータのオプション

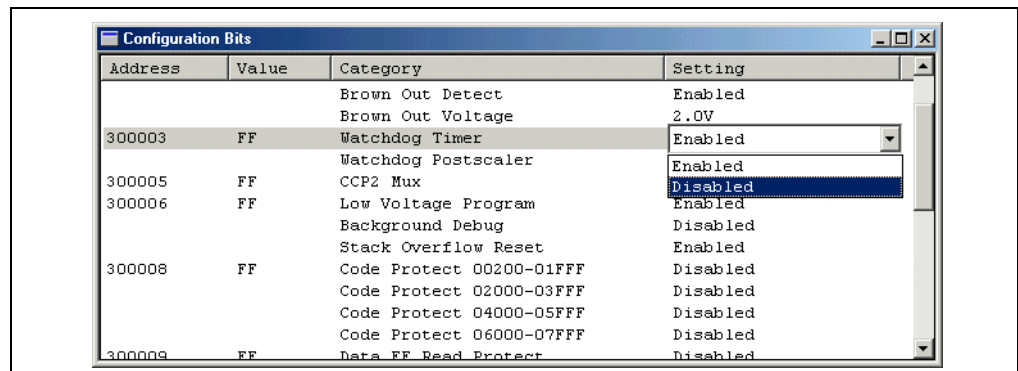
MPLAB IDE のダイアログでは、シミュレータの設定がほかにもあります。

### コンフィグレーションビットの設定

ウォッチドッグタイマ (WDT) はデフォルトではイネーブルになっています。WDT がタイムアウトするとシミュレータでもリセットします。

- **Configure>Configuration Bits** を選択します。
- ウォッチドッグタイマの設定の **Enabled** のところをクリックすれば、選択メニューが表示されます。**Disabled** にすれば WDT ではリセットしなくなります。

**FIGURE 22: DISABLE THE WATCHDOG TIMER**



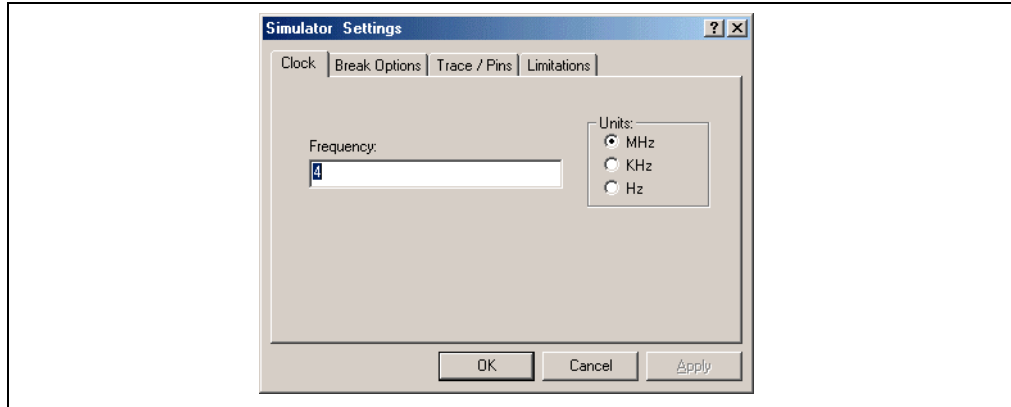
# MPLAB® IDE v6.xx Quick Start

## シミュレータの設定

**Debugger>Settings**を選択してデバッガ(ここではMPLAB SIMシミュレータ)の設定を行うダイアログを表示します。

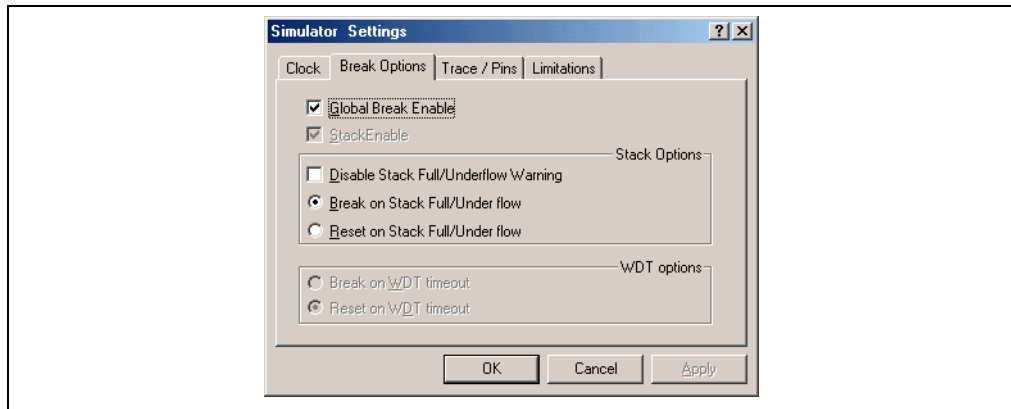
Clock パネルでシミュレータのクロック周波数を設定します。ここでの設定はストップウォッチやトレースで実行時間を計算するために使われます。

**FIGURE 23: DEBUGGER SETTINGS: CLOCK**



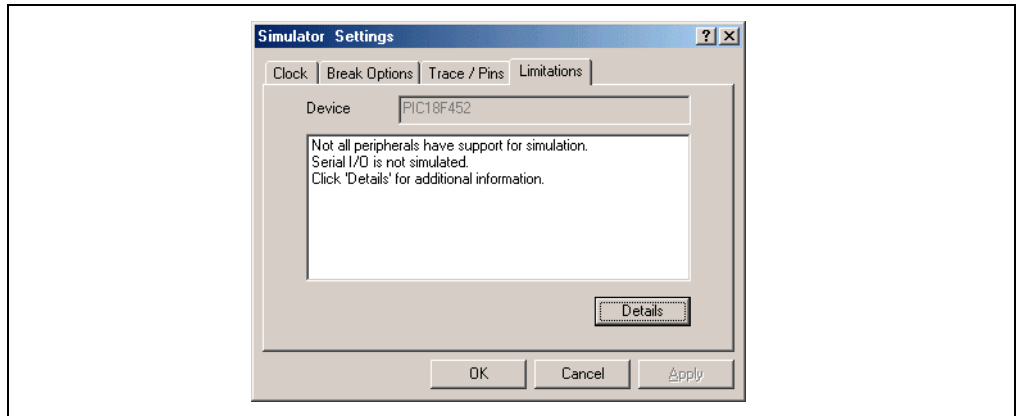
Break Options パネルではブレークポイントの設定ができます。Global Break Enable のチェックを外すと、ブレークポイントは無効になります。これは多数のブレークポイントを設定している場合に、全ブレークポイントを無効にする場合に便利です。Global Break Enable を再度チェックすればブレークポイントを有効にできます。

**FIGURE 24: DEBUGGER SETTINGS: BREAK OPTIONS**



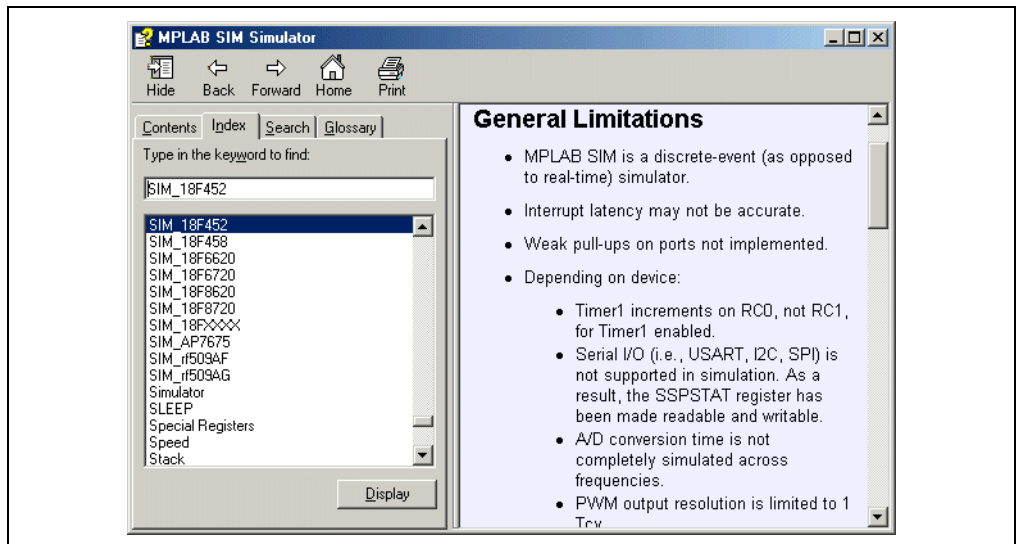
Limitations パネルではシミュレータとデバイスとの相違点を表示します。このパネルでは一般的な相違点を表示しています。

**FIGURE 25: SIMULATOR SETTINGS: LIMITATIONS**



**Details** ボタンをクリックすると、詳細な相違点が表示されます。一般的な相違点も表示できます。

**FIGURE 26: SIMULATOR SETTINGS: LIMITATIONS**



# MPLAB® IDE v6.xx Quick Start

## MPLAB ICD 2

この章では v6.00 から v6.10 にバージョンアップしたときに変更があったダイアログについて、いくつか説明します。また、MPLAB ICD 2 については *MPLAB ICD 2 Quick Start Guide (DS51268)* も参照してください。

MPLAB ICD 2 のドキュメントで推奨されているように電源を接続して、MPLAB ICD 2 をパソコンに接続します。ICD はデバッガかライターとして選択できます。

デバッガとして選択するには *Debugger>Select Tool>MPLAB ICD 2*。

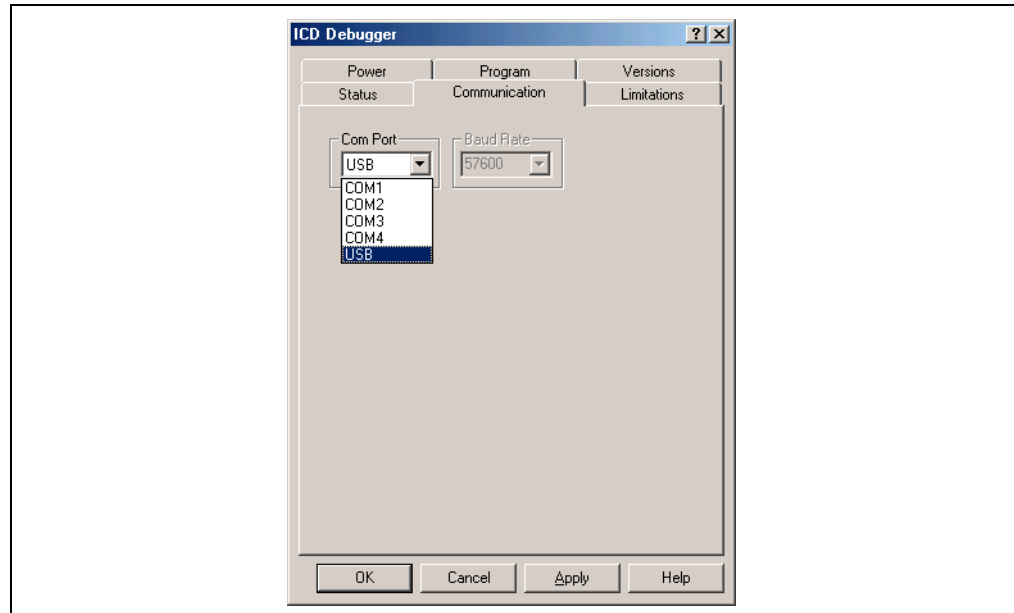
ライターとして選択するには *Programmer>Select Programmer> MPLAB ICD 2*。

**Note:** MPLAB ICD 2 をデバッガとライターの両方では選択しないでください。

MPLAB ICD 2 の通信設定

*Debugger>Settings* の Communication パネルで通信設定ができます。

**FIGURE 27: MPLAB ICD COMMUNICATIONS SETTINGS**



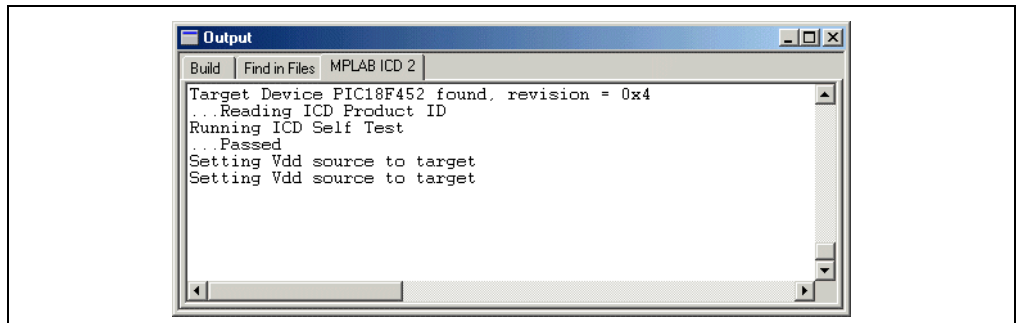
## MPLAB ICD 2 オペレーティングシステムの更新

MPLAB ICD 2 をはじめて使うときや MPLAB IDE が更新されたときは、MPLAB ICD 2 の README ファイルでファームウェアが更新されたかを確認してください。バージョンが正しくないと、正しく動作しないことがあります。

更新されたファームウェアは [Debugger>Download ICD 2 Operating System](#) か [Programmer>Download ICD 2 Operating System](#) でインストールできます。

ダウンロードの実行結果が Output ウィンドウに表示されます。エラーが発生したら、電源やケーブルを確認して再度ダウンロードしてください。問題解決には MPLAB ICD 2 のオンラインヘルプも参照してください。

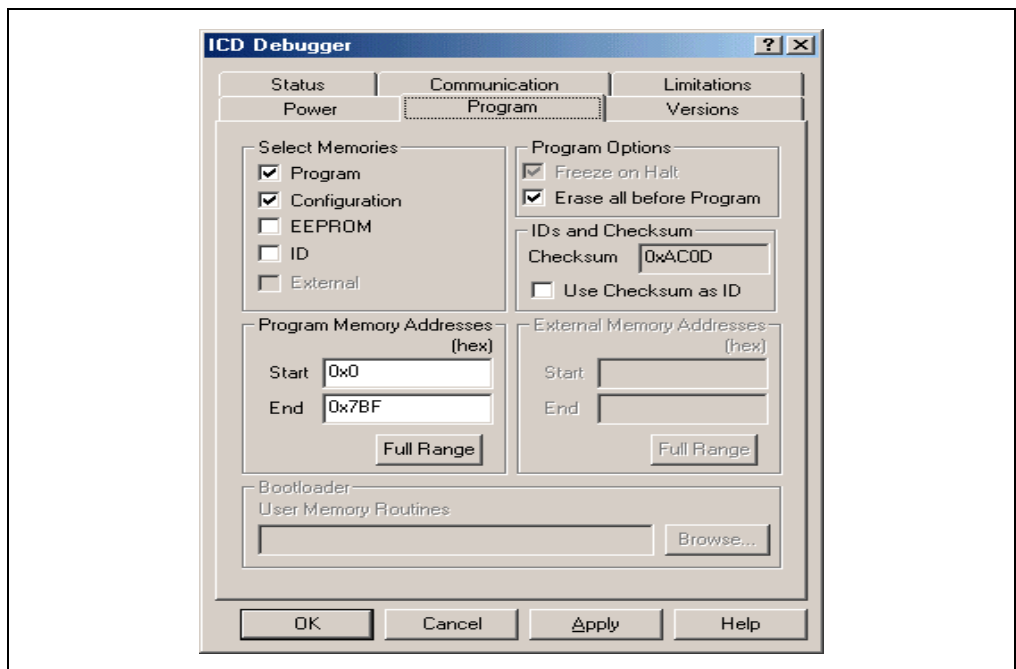
FIGURE 28: MPLAB ICD 2 OUTPUT WINDOW



## MPLAB ICD 2 の書き込みオプション

Settings ダイアログの Program パネルで書き込みの設定ができます。

FIGURE 29: MPLAB ICD 2 PROGRAM SETTINGS



MPLAB ICD 2 をデバッガとして選択しているときは、Program パネルの Configuration は常にチェックされます。これはインサーキットデバッグ機能をイネーブルにするためです。

MPLAB ICD 2 をライターとして選択しているときはチェックを外せます。

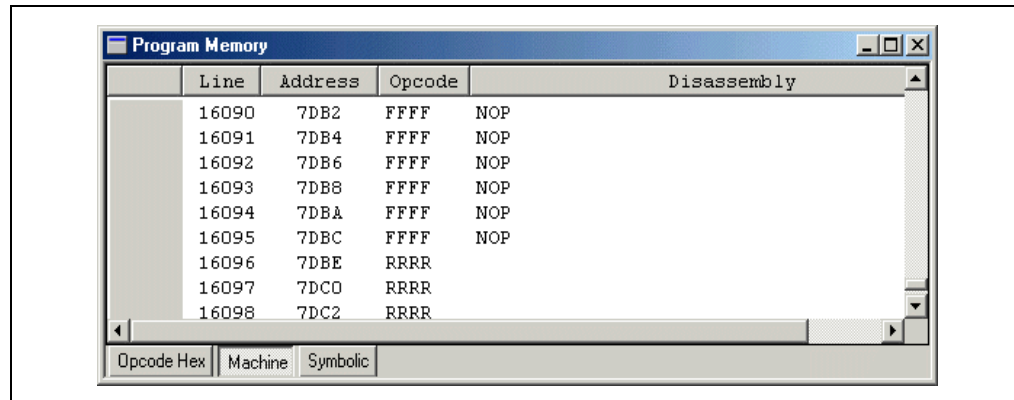
# MPLAB® IDE v6.xx Quick Start

図のように End プログラムアドレスを下位アドレスに設定すれば、書き込みが早く終わります。**Full Range** ボタンで全プログラムメモリに書き込みを行う設定にできます。

MPLAB ICD 2 で使うメモリ領域

MPLAB ICD 2 を使うと MPLAB IDE のいくつかのメモリ・ウィンドウは変わります。RRやRRRRと表示された箇所はMPLAB ICD 2がデバッガの場合に使う領域であることを示しています。

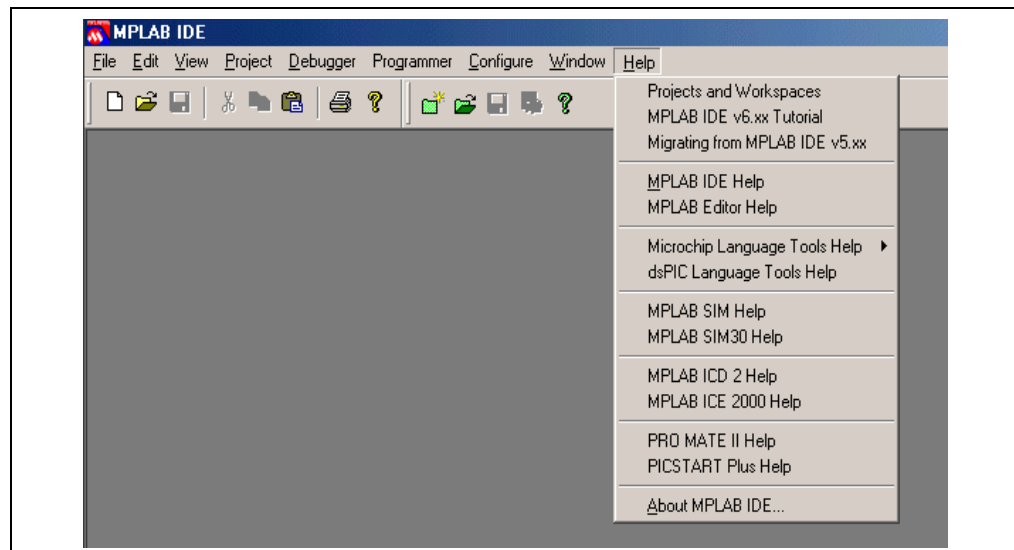
**FIGURE 30: MPLAB ICD 2 RESERVED PROGRAM MEMORY AREAS**



## MPLAB IDE のオンラインヘルプ

MPLAB IDE にはオンラインヘルプがあります。もし MPLAB IDE について知りたいことがあれば、オンラインヘルプを参照してください。選択しているデバイスによって、オンラインヘルプの項目が変わります。特に新しいデバイスとツールの組み合わせで使う場合などではオンラインヘルプを参照してください。

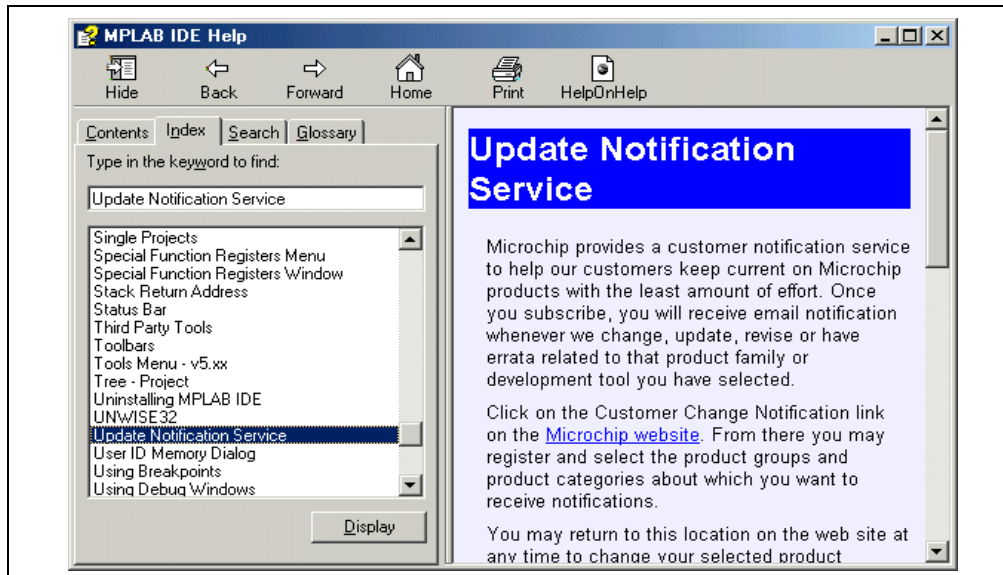
**FIGURE 31: MPLAB IDE HELP MENU**





MPLAB IDE のヘルプは MPLAB IDE やほかのマイクロチップの開発ツールのすべてについて説明しています。また、Microchip Update Notification system というサービスもあります。

FIGURE 32: MPLAB IDE HELP DIALOG



## ワークスペースとプロジェクト

MPLAB にはワークスペースとプロジェクトがあります。

MPLAB のワークスペースは MPLAB IDE ウィンドウのデスクトップ領域のことで、ワークスペースは開いているウィンドウ、選択した PICmicro デバイス、使用しているデバッガやライター、ハードウェアのツールの接続方法を覚えています。ふつうワークスペースはプロジェクトを使う前に設定します。

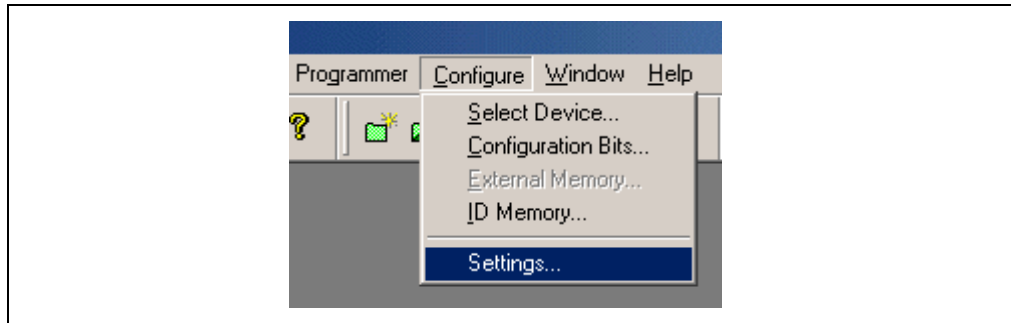
プロジェクトは、MPLAB のワークスペースのなかで開くもので、ソースコードやビルドする方法やビルドするために使ったツールなどの情報を含みます。プロジェクトは違うフォルダや違うパソコンに持ち運びができますが、ワークスペースは持ち運びはできません。

ワークスペースには複数のプロジェクトを入れることができますが、MPLAB IDE は同時には一つのワークスペースしか開けません。

# MPLAB® IDE v6.xx Quick Start

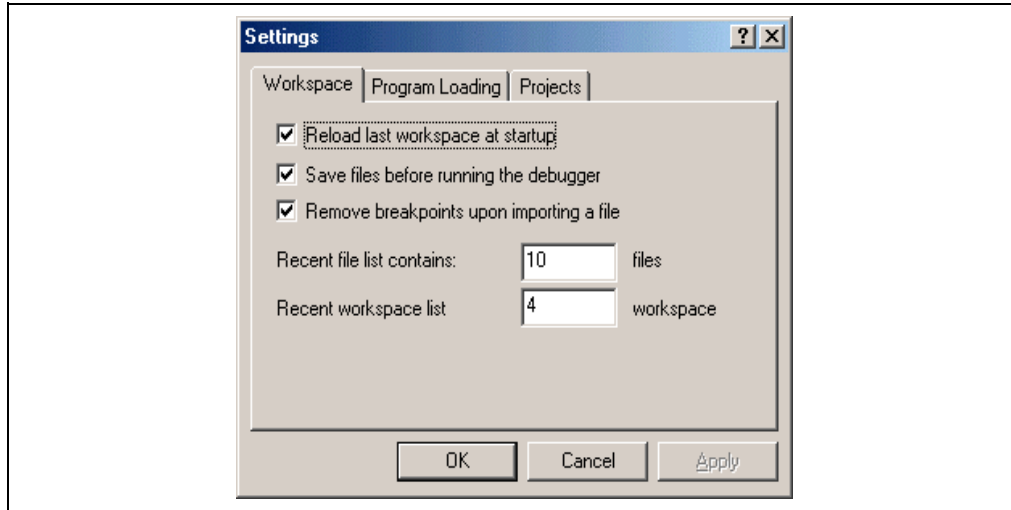
Configure>Settings メニューでワークスペースの設定ができます。本書のような使い方をすることは設定を変える必要はありません。

FIGURE 33: SETTINGS MENU SELECTION



Workspace というパネルがあります。

FIGURE 34: SETTINGS: WORKSPACE TAB



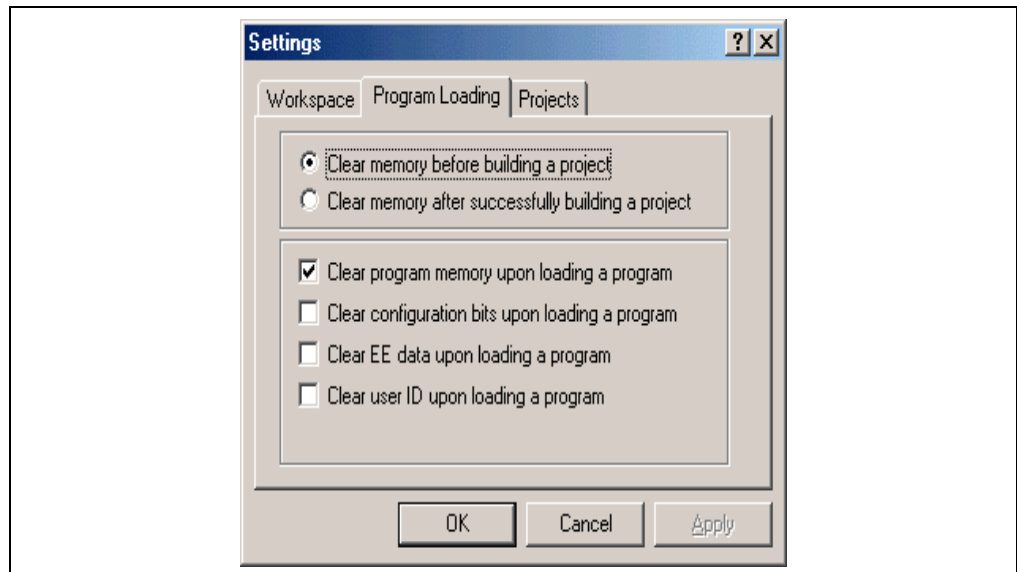
Configure>Settings ダイアログの **Workspace** パネルでは次の設定を行えます：

- MPLAB IDE を実行開始したときに最後に使っていたワークスペースを開く。一つのプロジェクトを継続して使う場合に便利です。
- エミュレータやシミュレータを使用開始する前に全テキストファイルを保存する。これは、ソースファイルなどを変更した後、デバッグを行う前にコンパイルし忘れるのを防ぐための機能です。
- HEX ファイルを読み込む前にブレークポイントを外す。変更が少ない場合で HEX ファイルをロードする場合のために、ブレークポイントを外さないようにもできるようにになっています。

**Note:** HEX ファイルを読み込むのは、おもにコンパイルしたコードをデバイスに書き込むためです。ビルドするたびに HEX ファイルは生成されます。

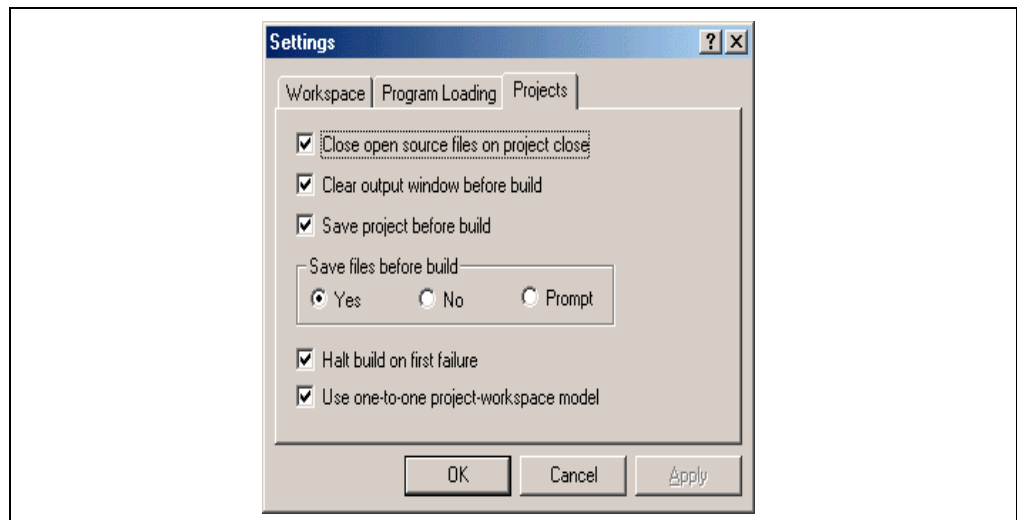
Program Loading パネルで、更新されたコードがロードされる場合に初期化するメモリ領域を選択できます。

**FIGURE 35: SETTINGS: PROGRAM LOADING TAB**



Configure>Settings ダイアログの **Projects** パネルで、プロジェクトをビルドするときの動作をカスタマイズできます。

**FIGURE 36: SETTINGS: PROJECTS TAB**



# MPLAB<sup>®</sup> IDE v6.xx Quick Start

---

このパネルはプロジェクトの設定を行えます。デフォルトのまま使うことを推奨します。不用意にチェックを外すと編集したものがなくなる場合もあるので注意してください。一番下の **Use one-to-one project-workplace model** という項目では MPLAB IDE のプロジェクトの扱いかたを選べます。チェックするとワークスペースは一つのプロジェクトしか持てないようになり、使用上はワークスペースとプロジェクトは同じになります。

**Note:** チェックを外すと、ワークスペースは複数のプロジェクトを持てるようになります。これはメモリブロックごとに違うコードをコンパイルするような用途の場合に便利です。たとえばブートローダーとアプリケーション部を持つことができます。ブートローダーはアプリケーション部と独立していて、コードのダウンロードや更新を行います。同時には一つのプロジェクトしか使えません。複数のプロジェクトを持つワークスペースで、プロジェクトを選ぶには **Project** ウィンドウのなかでそのプロジェクトの上にカーソルを置き、右クリックします。

NOTES: