
MPLAB-X

スタートガイド 技術資料

Windows8 環境

(株) マイクロアプリケーションラボラトリー

目次

1. プログラミングの準備・インストール	3
1-1 MPLAB X のインストール	3
1-2 XC8 のインストール	4
1-3 デバッカのインストール	7
■ PICKIT3	7
■ ICD3	7
2. プログラミングのスタート	8
2-1 装置のセットアップ	8
2-2 プロジェクトの作成	9
2-3 プログラムの入力	10
2-4 コンパイルを行う	11
2-5 プログラムを動作させる	11
2-6 プログラムの終了	11
3. プログラムのデバック	12
4. デバック時に便利なウインドウ	13
4-1 Watch ウインドウ	13
4-2 Variable ウインドウ	13
5. デバック時に便利なウインドウ2	14
5-1 Program memory	14
5-2 File Registers	14
5-3 SFRs	15
5-4 Configuration Bits	15
4-5 EE Data Memory	15
5-6 EEPROM Memory	16
5-7 User ID Memory	16
6. プロジェクトウインドウ	17
6-1 プロジェクト	17
6-2 プロジェクト	17

1. プログラミングの準備・インストール

プログラミングをスタートするための環境を整えます。

使用した PC は Windows 8 Intel core i5-3230M CPU 2.6GHz のミドルクラスのノートパソコンです。

OS や PC の仕様により本説明の画面や操作に多少の違いが発生するかもしれません。

MPLAB IDE はマイクロチップが無償提供している PIC マイコン開発用の統合開発環境です。MPLAB は過去 10 年に渡り PIC 開発ユーザに利用され、Version8 まで成長してきましたが、マイクロチップでは 2013 年より MPLAB-X という新しい環境へ一新しました。

MPLAB-X 上では全ての PIC マイコンの開発が実行できます。ここではこの MPLAB-X を利用してプログラミングを行います。

MPLAB-X は WindowsXP, Vista, 7, 8 環境で動作します。さらに Linux や MAC-OS 環境でも利用できるようです。

1-1 MPLAB X のインストール

ここでは、MPLAB-X Ver1.9 をインストールします。

MPLAB-X は基本的に利用するユーザがマイクロチップ社のホームページよりダウンロードして利用しなければなりません。

インストールの時期によりさらに新しい環境が存在し、インストール画面や手順が異なることが予想されます。

MPLAB-X v1.9 のインストールは 「MPLABX-v1.90-windows-installer.exe」 を起動することで開始します。インストールの開始にあたり許可を求めるメッセージが表示される場合があります。

インストールは画面に従い 「Next」 ボタンを押すだけで進めます。

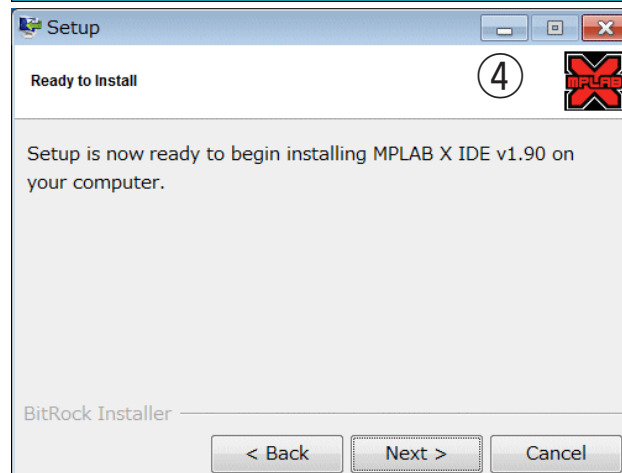
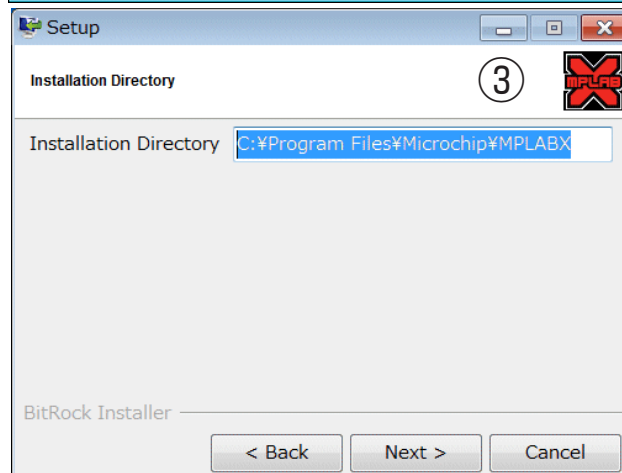
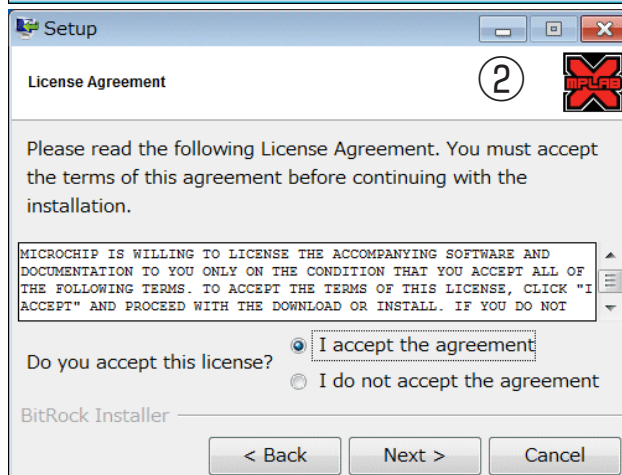
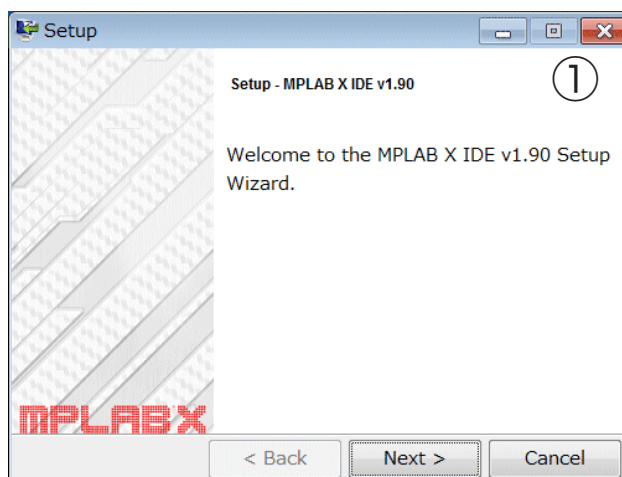
2 番目の画面では 「I accept ……」 を選び次に進みます。インストールするフォルダも自由に選択できますが、説明との食い違いなどで混乱を起こしますから、ここではデフォルトの指定で進めて下さい。

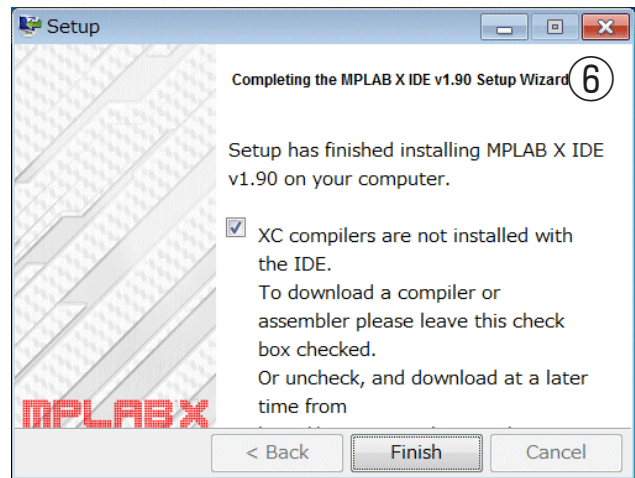
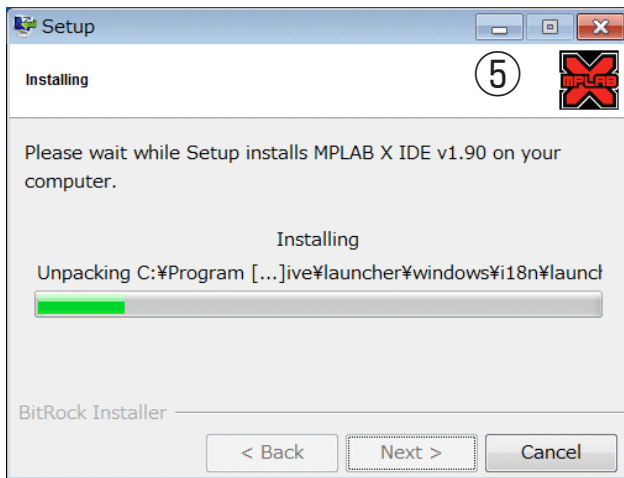
7、8 分でインストールが完了します。



デスクトップのアイコン

MPLAB-X V1.9 インストール画面 →





■旧 MPLAB をご利用の方

旧 MPLAB (～ Ver8) をご利用の場合、MPLAB-X と共存させて利用することは可能ですが、デバッカなどのハードウェア開発環境が双方で自由に利用できない欠点があります。

専用のアプリで装置を切り替える作業が必要です。また、環境によっては切り替えがうまく機能しない場合があります。どちらかの環境に固定した方が結果的に使いやすいのではないのでしょうか？

1-2 XC8 のインストール

MPLAB-X のインストールに続き C コンパイラ XC8 をインストールします。必ず MPLAB をインストールした後に行って下さい。

XC8 コンパイラは PIC16F・PIC18F の 8bit PIC マイコンファミリに対応したコンパイラです。

ANSC 規格に準拠した C 言語記述でオブジェクトファイルを出力します。多くの演算系関数、ペリフェラル対応関数などの組込関数をライブラリとして持っています。

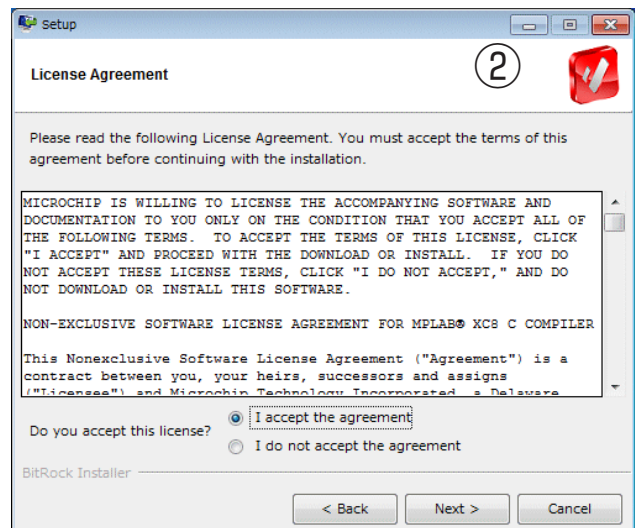
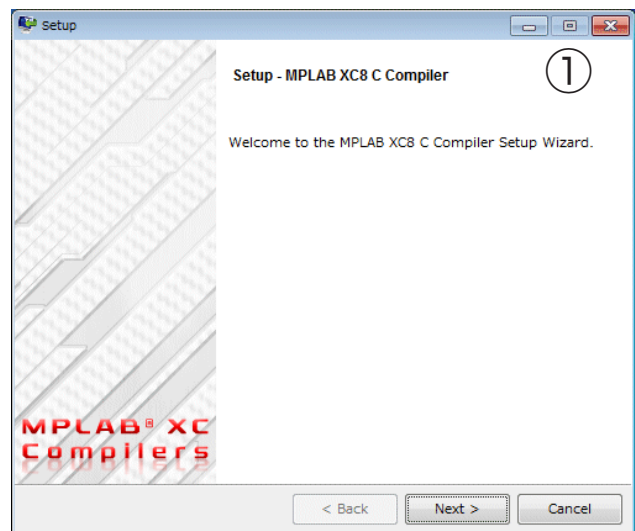
XC8 はプロ版、スタンダード版、無償版の3種類があり、コンパイル効率に相違があります。有償版はメモリ効率が良く、動作速度も速くなります。ここでは XC8 の無償版を利用します。

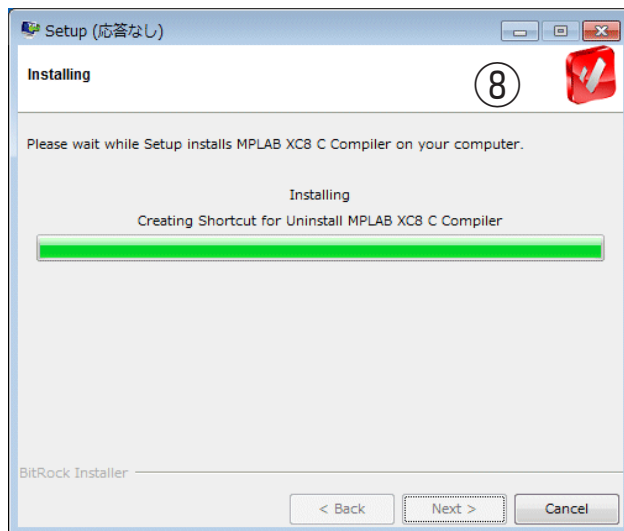
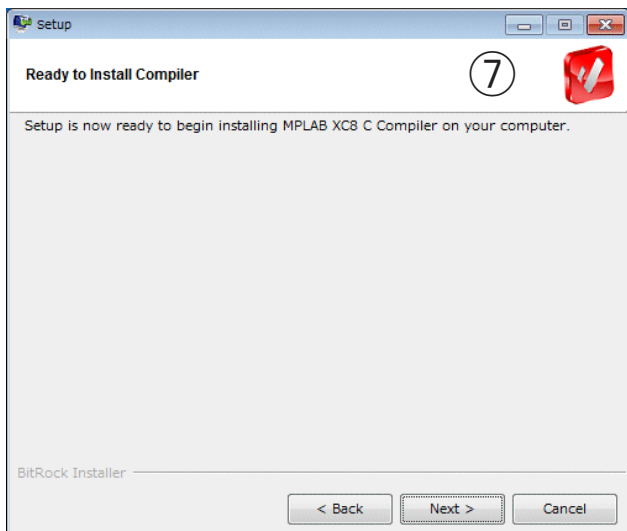
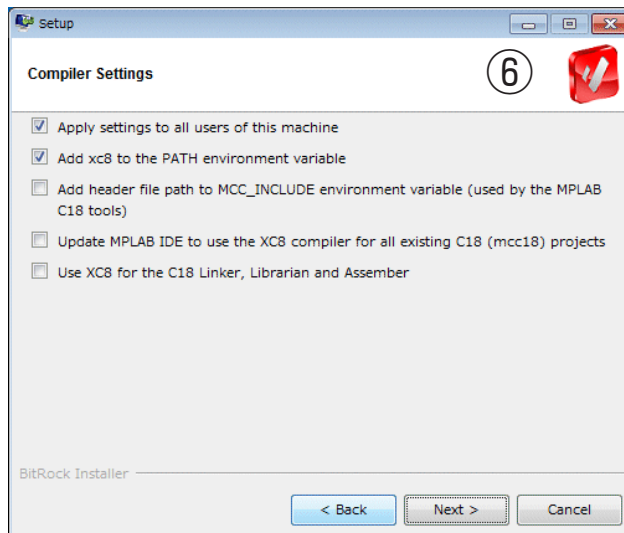
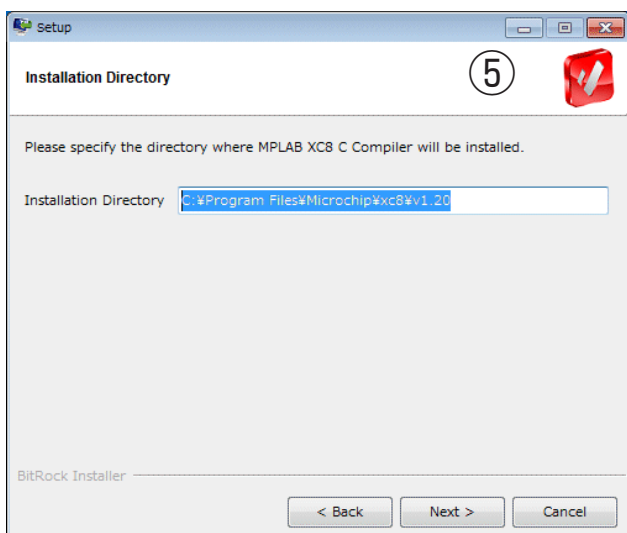
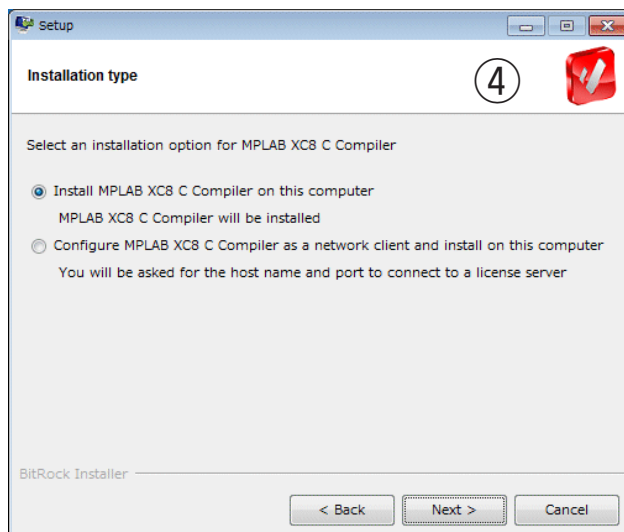
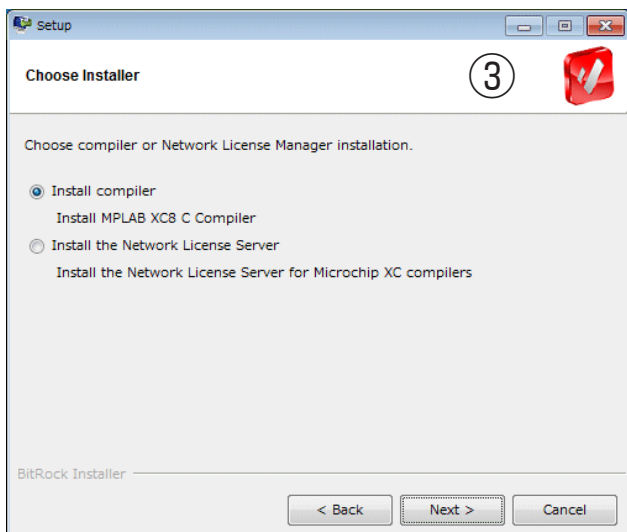
無償版は評価用でもあり、3ヶ月間はプロ版として動作し、3ヶ月後に無償版の機能になります。プロユースで頻繁に利用してゆく場合はスタンダード版・プロ版の XC8 コンパイラをご用意下さい。

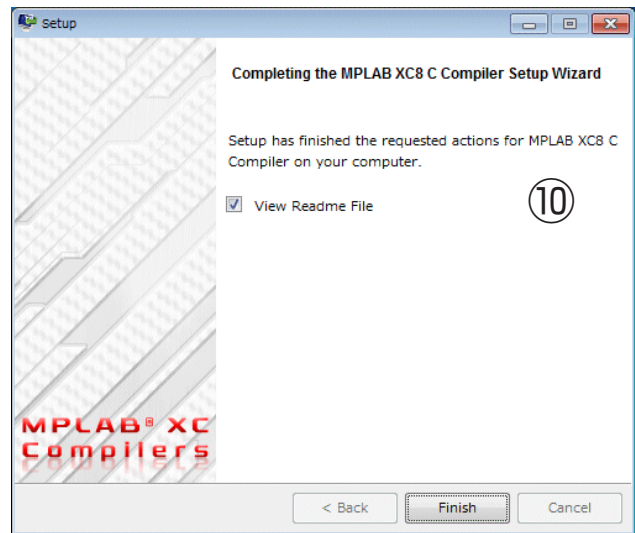
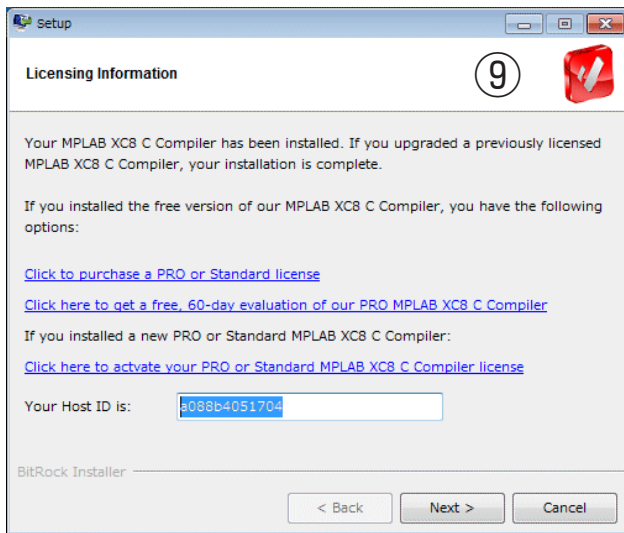
XC8 コンパイラもマイクロチップのホームページに登録してから利用しなければなりません。

マイクロチップ「<http://www.microchip.com/>」に一旦登録を行うようお願いいたします。

XC8 のインストールは「xc8-v1.20-win.exe」を起動します。インストールは画面の指示に従いそのまま進めれば完了します。







1-3 デバッカのインストール

■ PICKIT3

PICKIT3はWindowsのHIDドライバを使用しているため、インストール作業も無くスムーズに導入でき、新旧共存で利用できますが、転送速度やデバック操作の速度が遅い欠点があります。



■ ICD3

MPLAB-ICD3のWindowsデバイスドライバを組み込むのは少し難しい作業です。

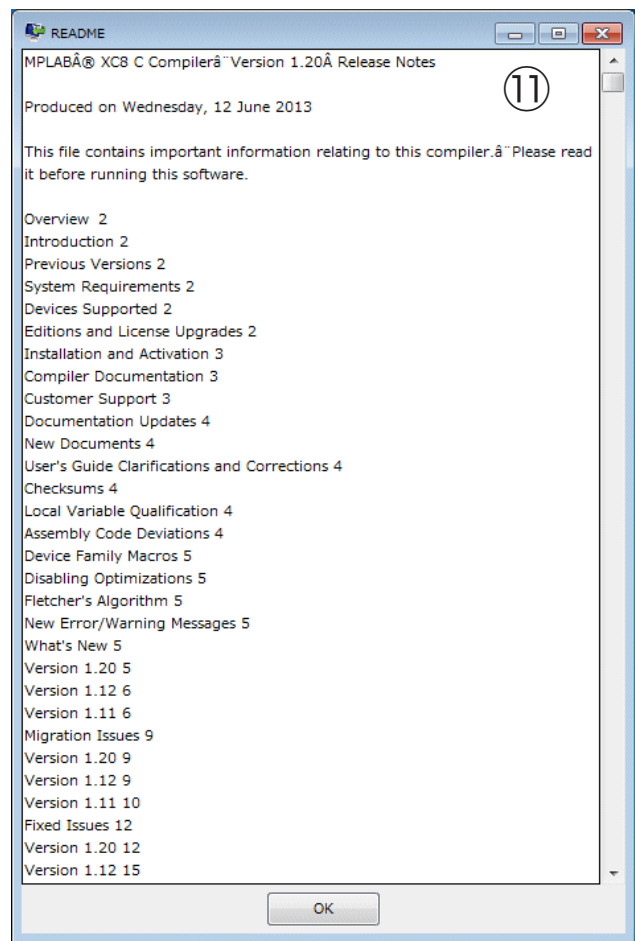
初めてのPCにMPLABを組込む場合や、Windows8ではあまり問題を起こさないと思いますが、すでにMPLABを使用していたWindows-XP・Windows7のPCにMPLAB-Xを新しくインストールした場合、利用していたICD3のドライバが置き換わらない問題が生じます。この問題を解消するためにドライバスイッチャというアプリケーションが同時にインストールされ、このアプリを利用することでICD3のドライバを差し替えるのですが、多くの場合エラーが生じて中止してしまいます。このような時はマニュアル操作でドライバを差し替えた方が簡単で確実です。



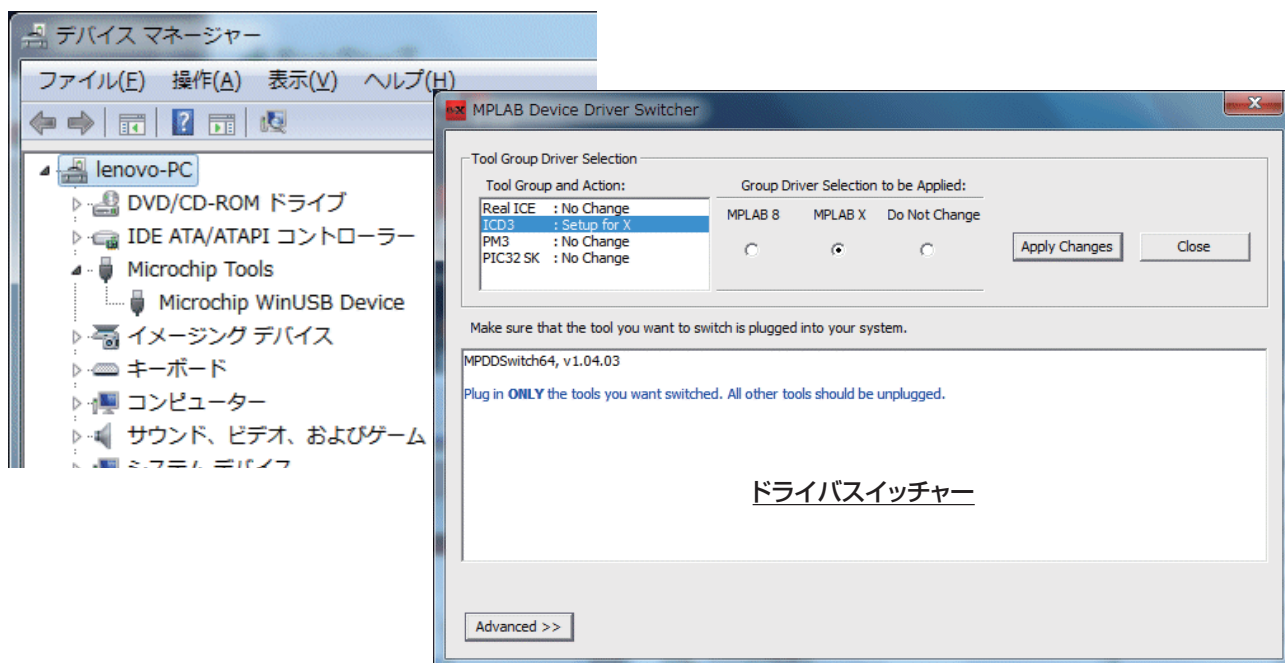
ICD3はすでに古いドライバで認識されているので、デバイスマネージャを開いて、ICD3の項目を右クリックして「ドライバソフトウェアの更新」のメニューから新しいドライバを指定します。ドライバのフォルダは以下の場所になります。ドライバが新しくなるまではMPLAB-Xを起動しないようにします。

ドライバのフォルダ

32-bit Windowsの場合: C:\Program Files\Microchip\MPLAB X IDE\Switcher\32bit\winusb
64-bit Windowsの場合: C:\Program Files (x86)\Microchip\MPLAB X IDE\Switcher\64bit\winusb



古いMPLABを再び使用する場合は同じようにドライバを差し替える必要があるため、新旧のMPLABを共存させて利用することは難しいといえます。



2. プログラミングのスタート

はじめに LED を点灯させるプログラムを作成し、プログラム環境の立ち上げと使い方を理解します。

2-1 装置のセットアップ

はじめにハードウェア環境をセットします。
ここでは MA366 小型評価ボードを使用します。搭載されているデバイスは PIC18F45K22 です。
ポート D に接続されている LED 表示を利用します。



MA366 小型評価ボード PIC18F45K22

MPLAB X と XC8 コンパイラがインストールされたパソコンを起動し、デバッカの MPLAB-ICD3 (PICKIT3) を付属の USB ケーブルで接続します。
ICD3 デバッカはデバイスドライバが必要です。
パソコンのコントロールパネルにあるデバイスマネージャよりデバッカが正しく認識され、接続されたことを確認します。
ICD3 の場合は接続により装置を認識してドライバを準備します。PICKIT3 は標準の HID ドライバを使用するため、自動的に準備されます。

ボードに通電し、ICD3 (PICKIT3) とボードをモジュラコネクタで接続します。PICKIT3 では、コネクタのピンがずれた状態で差し込まないように充分注意して下さい。ずれてしまうと故障の原因になります。



2-2 プロジェクトの作成

接続が完了しましたら MPLAB X を起動します。MPLAB X 起動後に ICD3 (PICKIT3) を接続すると認識されない場合があります。

プログラミングの開始にあたり、プログラムのうつわにあたる「プロジェクト」(Project)を作成します。

プログラムを作るための場所となる作業フォルダを確保します。ここでは「Dev_MPLABX」名のフォルダを作成し、そこに各プログラムを作ってゆきます。

プロジェクトのスタートはツールバーの左から2番目にある New_Project アイコンよりスタートします。

①はじめにスタート画面が表示されます。この画面はそのまま次に送ります。

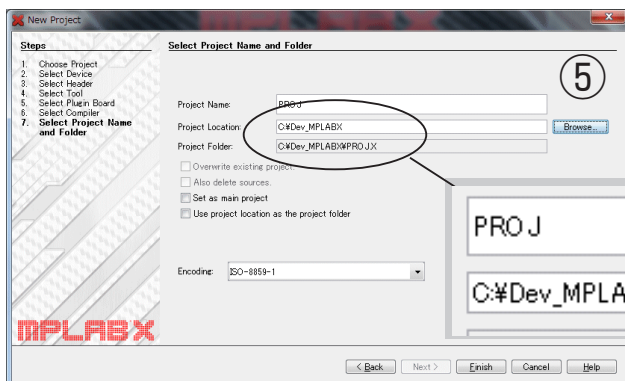
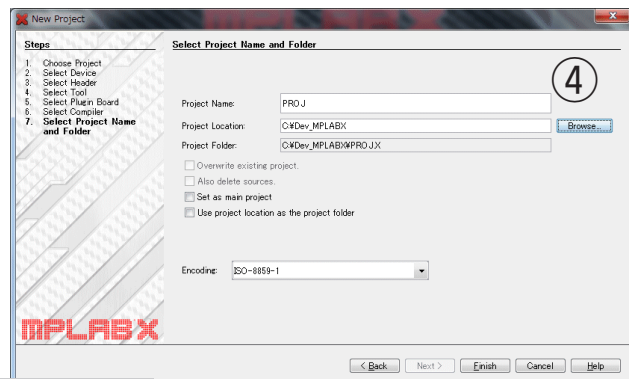
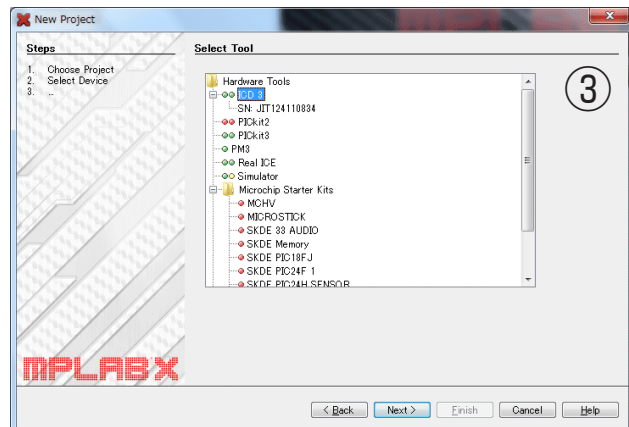
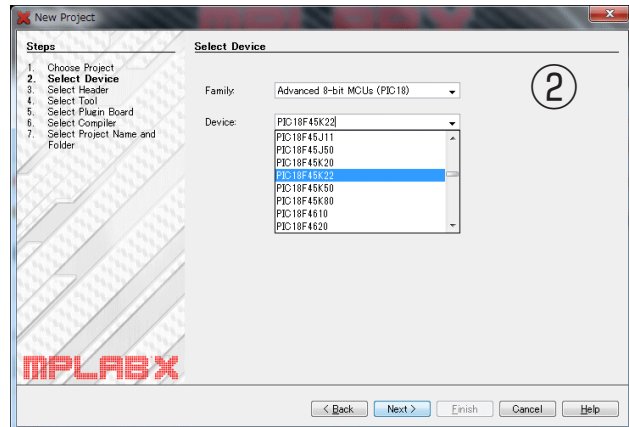
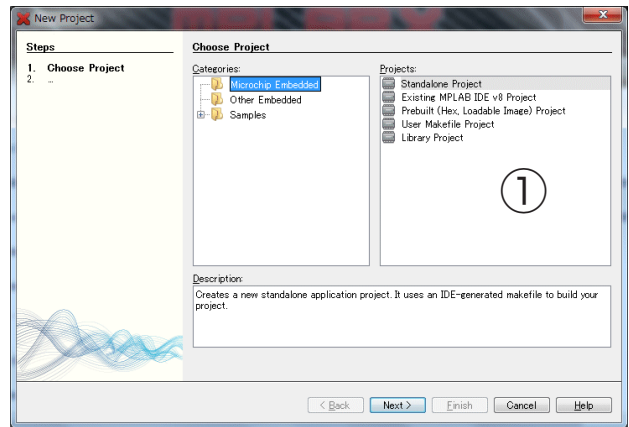
②使用するデバイスを指定します。ここでは PIC18 のファミリーを指定し、PIC18F45K22 を指定します。

③使用するデバックツールを指定します。ここでは ICD3 を指定します。ICD3 が認識されているとそのシリアル番号などが示されます。

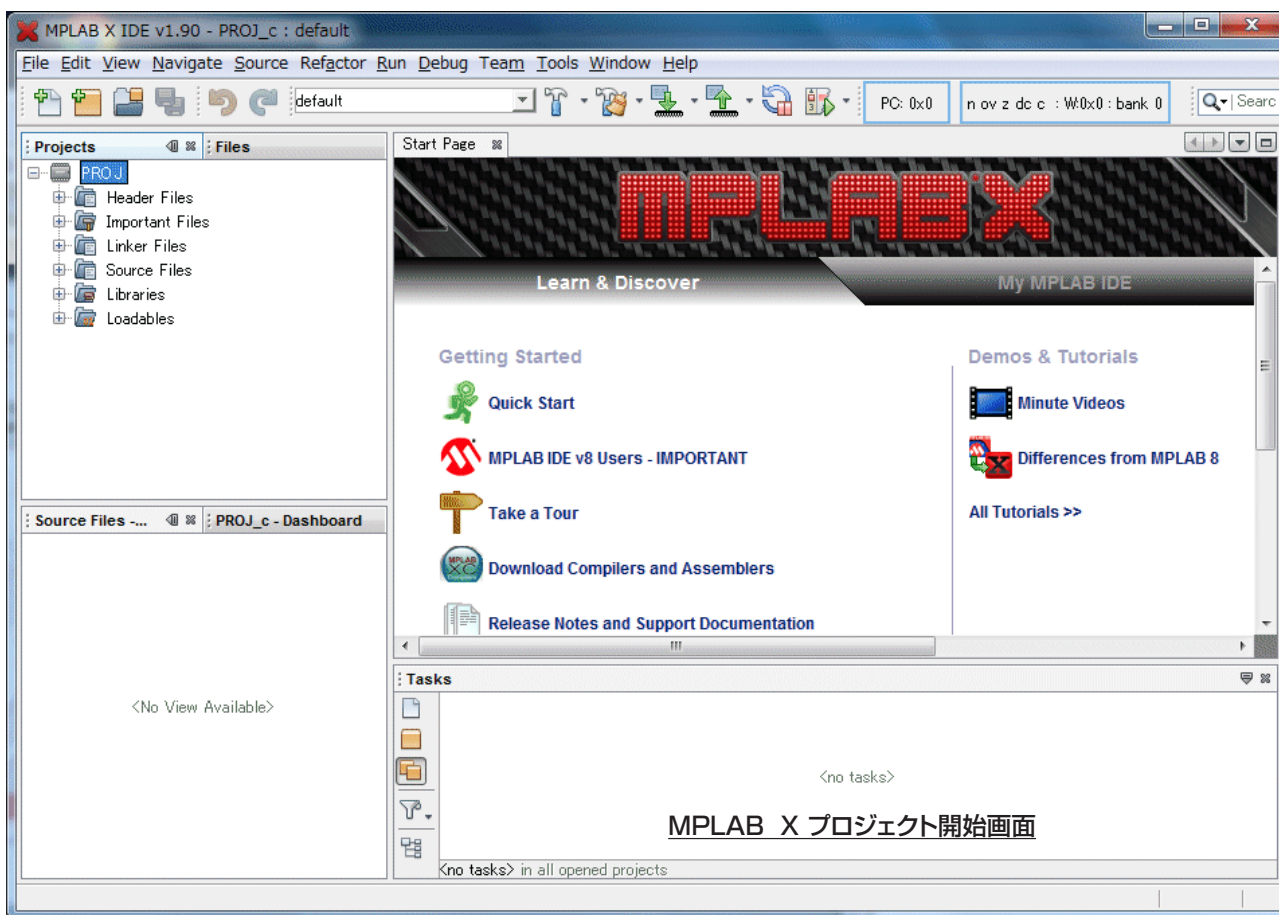
④次に使用するコンパイラを指定します。XC8 コンパイラを選択します。

⑤プロジェクトを格納する場所と名称を指定します。プロジェクト名は好きな名称で良いのですが、ここでは汎用的に利用できる「PROJ」という名称にします。フォルダ位置の指定は「Browse」ボタンより先ほど作成した MA370 フォルダを指定しています。

⑥ Finish を入力して終了します。



プログラム名・フォルダ名などは半角英数文字を使用します。漢字などの全角文字を使用するとツールが認識できない場合があります。



MPLAB X プロジェクト開始画面

2-3 プログラムの入力

⑦次にプロジェクトの“Source_Files”を右クリックロックして“New”の“C_Source_File”を選択します。

⑧画面が開きますので、そこにファイル名称を入力します。ここでは「LED」の名称を指定します。名称には拡張子の“.c”が付加されます。

⑨画面を終了すると“Source_Files”の下に“LED.c”のファイルが加わり、右側のウィンドウにはエディタ画面が開きます。タブ名称を確認して下さい。

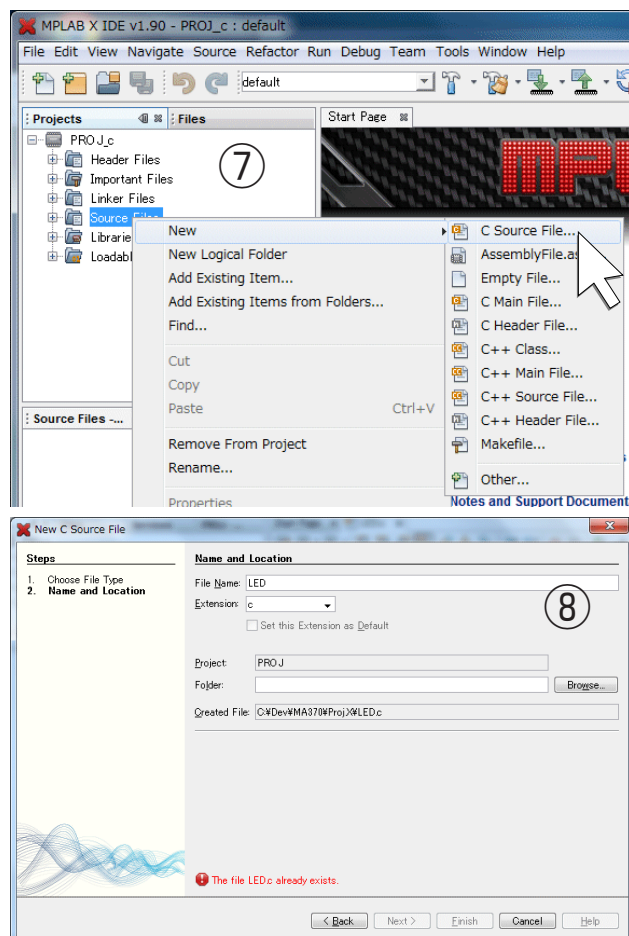
⑩エディタ画面にプログラムを入力して行きます。プログラムはLEDを点灯するだけの簡単なプログラムです。セミコロンやピリオドなどの小さな記号に注意して入力して下さい。

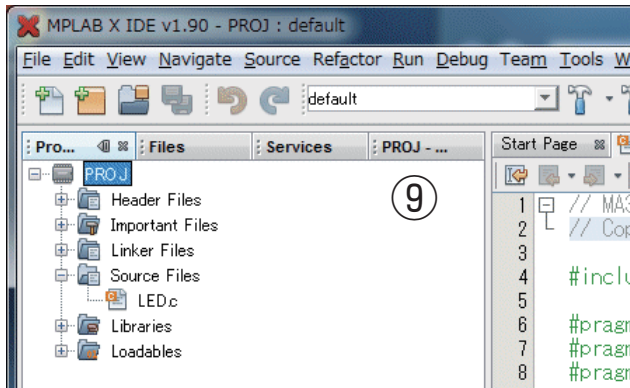
各単語間にはスペースが必要です。この場合はいくつ入力してもかまいません。逆に約束された単語間にスペースを設けることはできません。

小文字や大文字も認識されます。

行の最後には必ずリターン (Enter) を入力します。

⑪プログラムをファイルにセーブします。[File → Save]メニューよりセーブします。





2-4 コンパイルを行う



プログラムをコンパイルする準備が完了しました。コンパイルはツールバーのビルドアイコンより行います。メニュー操作では [Run → Build_Project] になります。コンパイルが開始されると出力ウインドに多くのメッセージが表示され、最後から数行手前に "BUILD_SUCCESSFUL" の表示があればコンパイル成功です。プログラムの入力にエラーがあれば "BUILD_FAILED" のエラー表示になります。この場合はプログラムの入力にミスがないかよく調べて下さい。このコンパイル操作は実行形式ではなく入力に問題がないかどうか確認するための操作です。プログラムの入力エラーがあった場合、エラー行の左側に "!" マークが表示されますので問題を修正します。

入力するプログラム LED.c

```
#include <xc.h>

#pragma config FOSC = HSMP
#pragma config PLLCFG = ON
#pragma config WDTCN = OFF
#pragma config LVP = OFF

void main( void )
{
    unsigned int n = 0;

    ANSELC = 0;
    TRISC4 = 0;

    while(1)
    {
        LATC4 = 1;          //LED on
        while(++n != 0) ;

        LATC4 = 0;          //LED off
        while(++n != 0) ;
    }
}
```

2-5 プログラムを動作させる



それでは実際にプログラムを動作させてみます。プログラム実行のアイコンをクリックします。このアイコンは "Make_and_Program_Device" で、コンパイルとデバイスへの書込を同時に行います。メニューからは [Run] → [Run Project] で起動します。"Programming/verify_Complete" の表示で書込が完了します。完了と同時にプログラムが起動し、MA366 上の LED4 が点滅します。この表示が出ない場合や、コンパイルがスタートしない場合は ICD3 の認識や接続に問題があります。

プログラミングに慣れてきた時はこのアイコンですぐに書き込むことも可能です。書式にエラーがある場合は書込を行わずにエラーを表示して中断します。以上で一連のプログラム作成手順の終了です。

2-6 プログラムの終了

プログラム作業を終了する場合は、開発ボードより ICD3 を取り外してから開発ボードの電源を切断するか、USB との接続を切り離してから開発ボードの電源を OFF にします。MPLAB を終了させます。プロジェクトはコンパイル時にセーブされているようですからそのまま終了しても次回に同じ状態が復元されます。

3. プログラムのデバック



ICD3-PICKIT3はプログラムデバックですから、デバックモードに切り替えるとデバックとして機能します。プログラムのコンパイル時にデバックモードでコンパイルします。Debug Project アイコンのクリックで起動します。メニューからは [Debug] → [Debug Project] で起動します。




書込とデバックの違いは、CPU の中にデバック用のプログラムが送り込まれることからユーザのプログラムエリアを圧迫します。同様にワークメモリ領域も提供する必要があります。


デバックでコンパイルするとデバック用のアイコンが表示され利用できるようになります。

プログラムの開始 (RUN) と停止 (HALT)、ステップ実行ができます。ブレークポイントは実行停止状態でコード内のブレークをかけたいステートメントをダブルクリックするとブレークポイントが設置されます。B のマークで示されます。リセットボタンからプログラムカウンタを初期化して実行を開始すると設定したブレークポイントでプログラムが停止します。ブレーク時は CPU の状態を表示させなければ意味がありません。

メニューより [Window] → [Debugging] よりモニタ用のウインドウである Watch ウィンドウや Variable ウィンドウを開いて変数をモニタします。

■ MPLAB X のツールバー

	コンパイルを行う プロジェクトビルド
	PIC デバイスへのプログラムと実行
	デバックモードでのプログラムと実行



STEP: ステップ実行

RUN: プログラム実行

RESET: デバイスリセット

Pause: プログラム停止

デバックモードの終了

■ 日本語の表示

プログラムのコメント部分には日本語を利用できます。この場合は使用するフォントを MS ゴシックに切り替えます。

Tools メニュー → Options
→ Fonts&Colors

4. デバック時に便利なウィンドウ

4-1 Watch ウィンドウ

メニューの [Window] → [Debugging] → [Watch] よりメモリをモニタするウィンドウである Watch ウィンドウをオープンできます。

画面を右クリックし、New_Watch から表示したい変数を指定します。登録された変数や SFR レジスタを名称で選択します。

名称・タイプ・アドレス・値の項目が表示されていますが、項目を右クリックすると新しい項目を追加できます。

10 進数表示、バイナリー表示などを追加すると便利です。変数値をモニタするだけでなく、その値を書き換えることができます。変更したい値をダブルクリックして数値を入力します。ブレークポイントで停止したときの変数値が期待通りの値でなかった場合、値を変更してプログラムを続行するような場合に使用します。

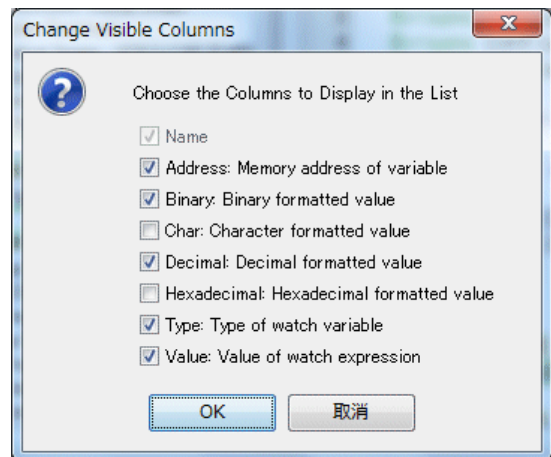


4-2 Variable ウィンドウ

メニューの [Window] → [Debugging] → [Variable] より変数をモニタするウィンドウである Variable ウィンドウをオープンできます。

使用している変数が表示されます。

名称・タイプ・アドレス・値の項目が表示されていますが、項目を右クリックすると新しい項目を追加できます。

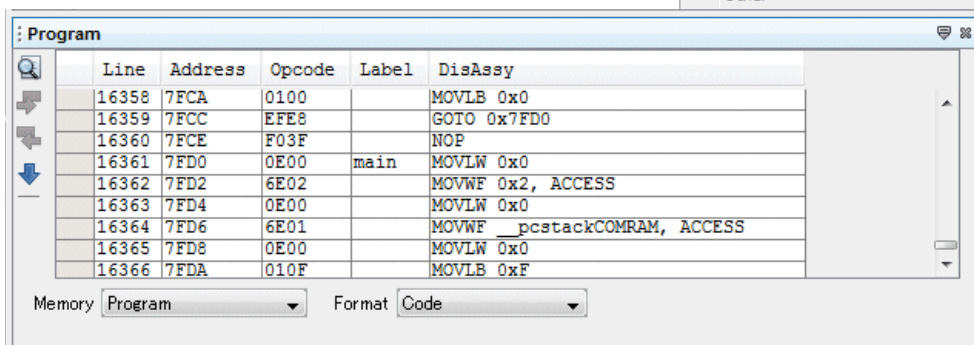


5. デバック時に便利なウインドウ2

メニューの [Window] → [PIC Memory Views] から様々なメモリモニタのためのウインドウをオープンできます。デバック時に便利に利用できます。

5-1 Program memory

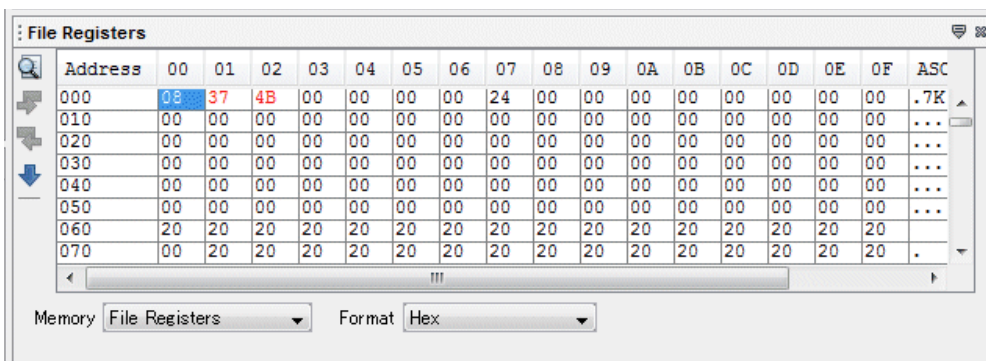
プログラムメモリを見るためのウインドウです。アドレス・オペコード・ラベル・逆アセンブラリストの項目で表示されます。C 言語のプログラムがどのように機械語に展開されたかを調べることができます。コンパイル時に起きたトラブルを追求することができます。



5-2 File Registers

メモリ（ファイルレジスタ）を見るときのウインドウで、1 行に 16 バイトのメモリがマトリックス形式で表示されます。値は 16 進数と ASCII 形式で表現されます。

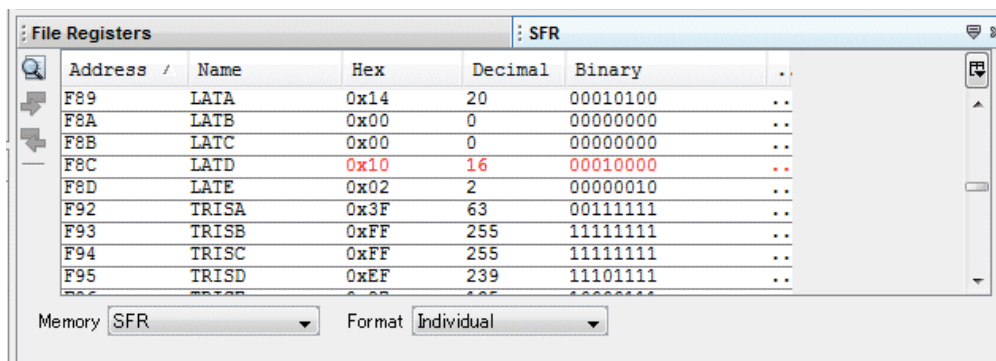
変更があったメモリは赤字で表示されます。表示された値を強制的に変更することができます。変更したいメモリをダブルクリックして数値を入力します。



5-3 SFRs

SFR はスペシャルファイルレジスタのことで、メモリ中の空間に存在する特殊なレジスタ群の専用モニタです。ポートやパリティのためのレジスタが割り当てられてい

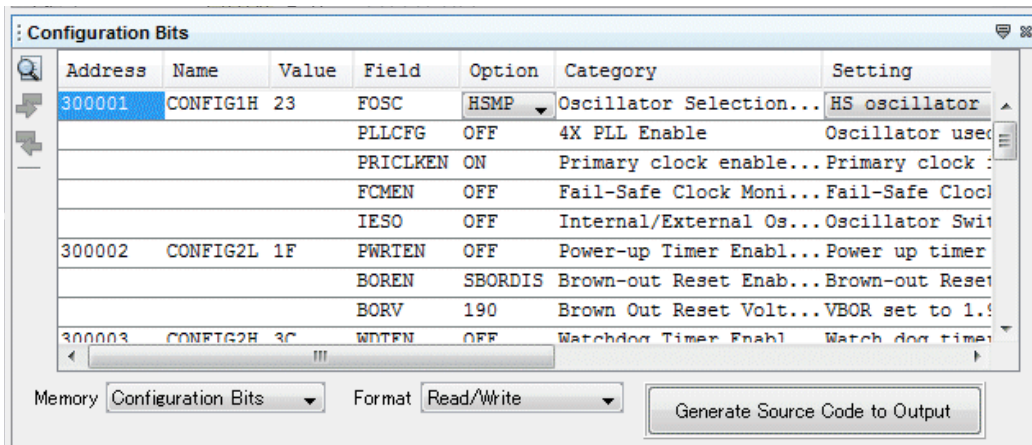
ます。このウインドウではレジスタ名によるメモリ表示が可能です。レジスタへの設定状態の確認や、プログラムによるアクセス状態、設定の変更などに利用できます。



5-4 Configuration Bits

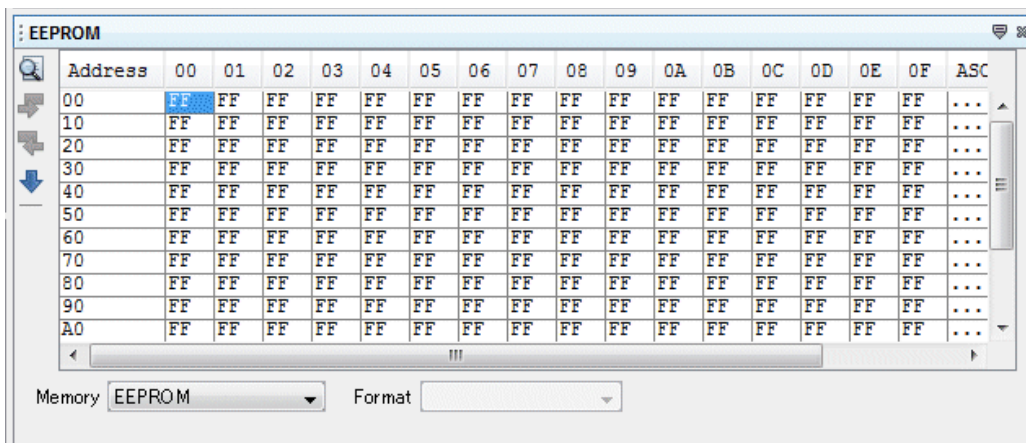
コンフィグレーションレジスタは起動時のハードウェア機能設定です。
最近ではプログラム上に記述する方法で設定しますが、マニュアルによる設定や設定状態の確認に利用できます。

重要な設定としてはクロックソースの選択があります。設定が回路構成と一致しなければ動作しません。リセットや、WDT の設定も重要です。BOR 設定はデバック時に障害が出ることもあり、この時は OFF 設定が良いでしょう。



5-5 EEPROM Memory

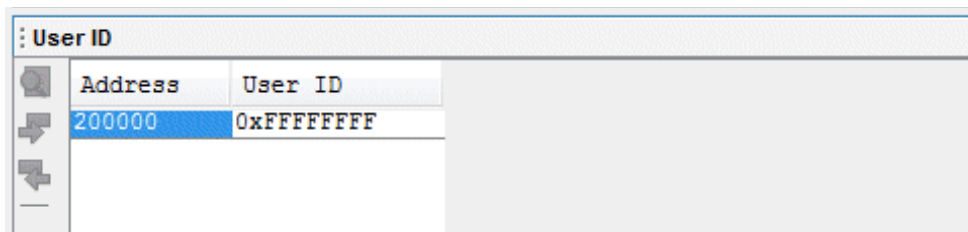
EEPROM メモリの内容を表示するウィンドウです。
EEPROM への書込状態などの確認に利用できます。



5-6 User ID Memory

User が自由に利用できる ID 領域です。シリアル番号やバージョン情報、日付などのプログラム管理に利用できま

す。プログラムメモリにプロテクトをかけた場合、メモリの内容が読めなくなりますが、ID 領域は読取り可能です。



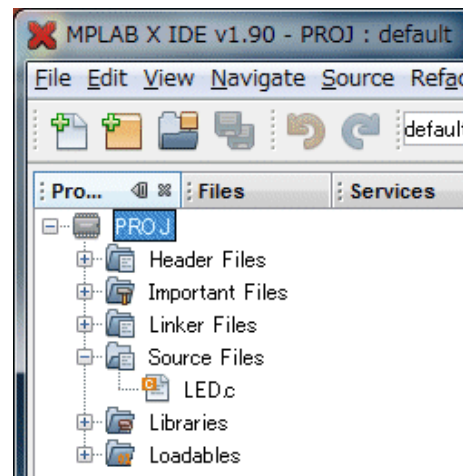
6. プロジェクトウインドウ

6-1 プロジェクト

ウインドウ左上のエリアはプロジェクト関係のウインドウです。

Project ではプロジェクトで利用されるファイル構成をします。プログラムソースファイルはもちろんのこと、ヘッダファイルやライブラリファイルなど同時にコンパイルする必要があるファイルを組み込むことができます。

各ファイルは右クリックで追加したり削除したりの操作ができます。



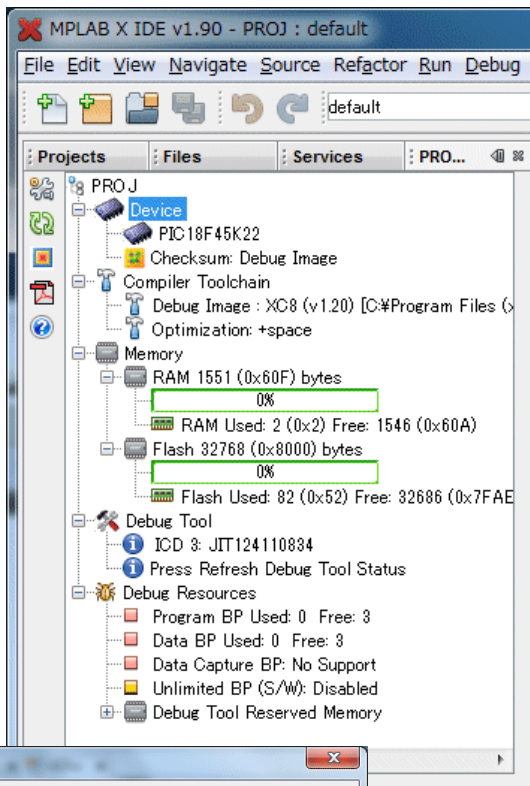
6-2 プロジェクト

Project Dashboard ではプロジェクトがどのような状態で設定されているか知ることができます。

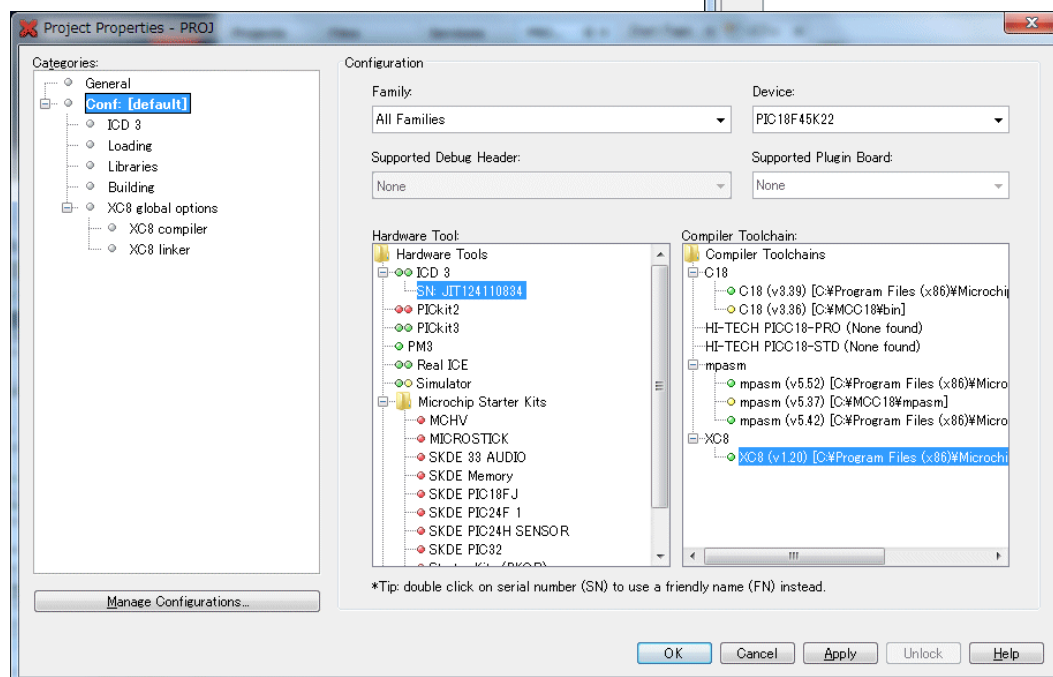
デバイス名やコンパイラの指定、メモリの利用状態、デバッカの設定などの情報が表示されるため、プロジェクト環境の確認に利用できるウインドウです。

よく利用するのはデバッカの確認と切替です。デバッカの接続状態を確認したり、別のデバッカに切り替える操作に利用します。デバック中にデバッカをプログラムシミュレータに切り替えたりすることは、しばしば発生します。

項目をダブルクリックすると設定画面がオープンします。設定の切替はこの画面から行います。左上のアイコンからも起動できます。



プロジェクトプロパティ設定画面



7. XC16 の利用

PIC24F・dsPIC33 などの 16bit アーキテクチャの PIC マイコンでは XC16 コンパイラを使用します。

7-1 XC16 のインストール

はじめに、C コンパイラ XC16 をインストールします。必ず MPLAB をインストールした後に行ってください。

XC16 コンパイラは PIC24F・PIC24H・PIC24E・dsPIC30・dsPIC33・dsPIC33E 用の 16bit PIC マイコンファミリに対応したコンパイラです。

ANSC 規格に準拠した C 言語記述でオブジェクトファイルを出力します。多くの演算系関数、ペリフェラル対応関数などの組込関数をライブラリとして持っています。

XC16 はプロ版、スタンダード版、無償版の3種類があり、

コンパイル効率に相違があります。有償版はメモリ効率が良く、動作速度も速くなります。ここでは XC16 の無償版を利用します。

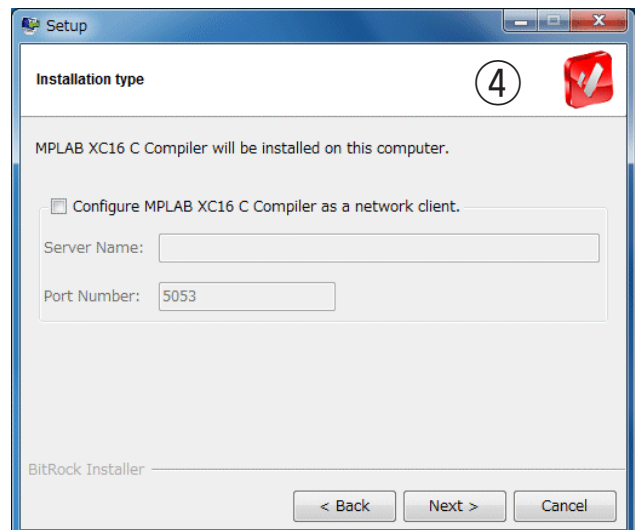
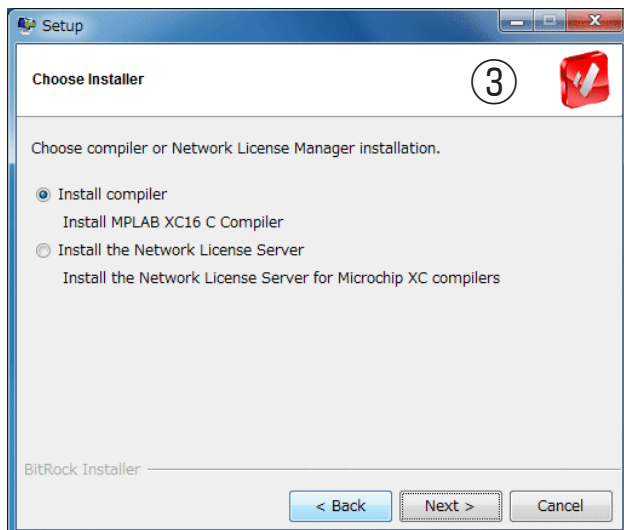
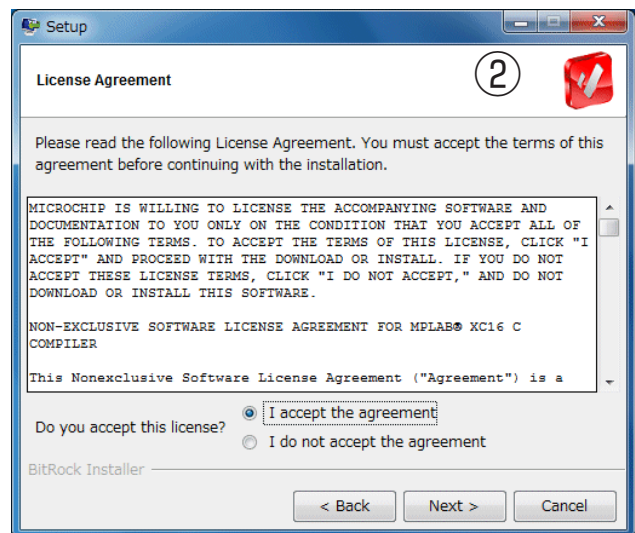
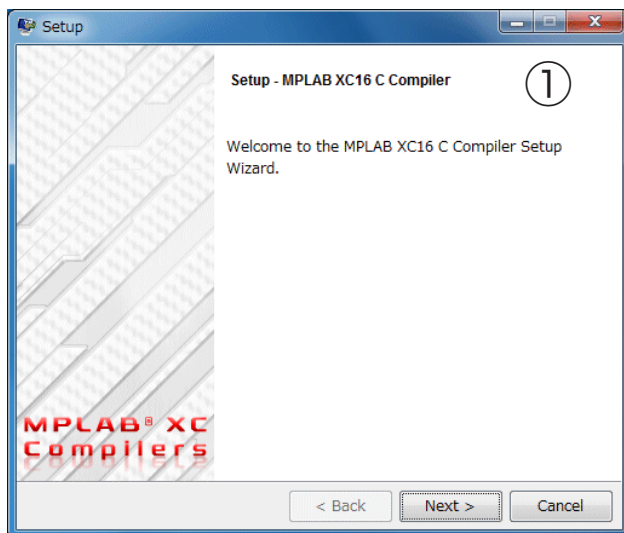
無償版は評価用でもあり、3ヶ月間はプロ版として動作し、3ヶ月後に無償版の機能になります。プロユースで頻繁に利用してゆく場合はスタンダード版・プロ版の XC16 コンパイラをご用意下さい。

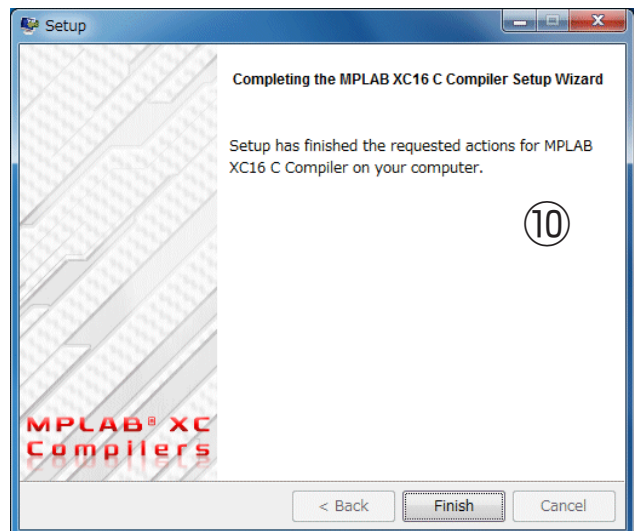
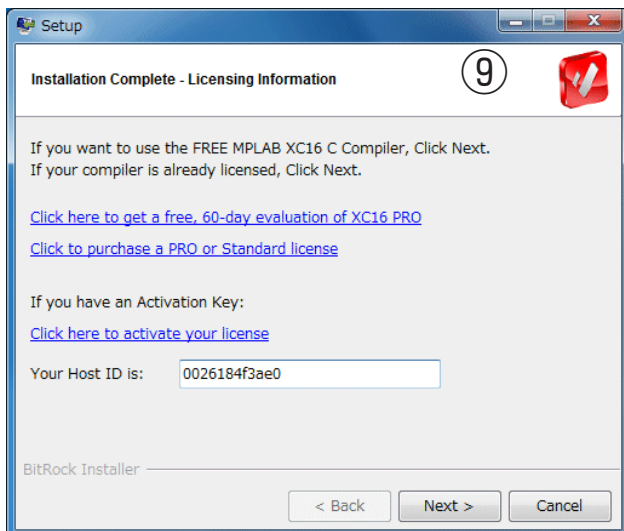
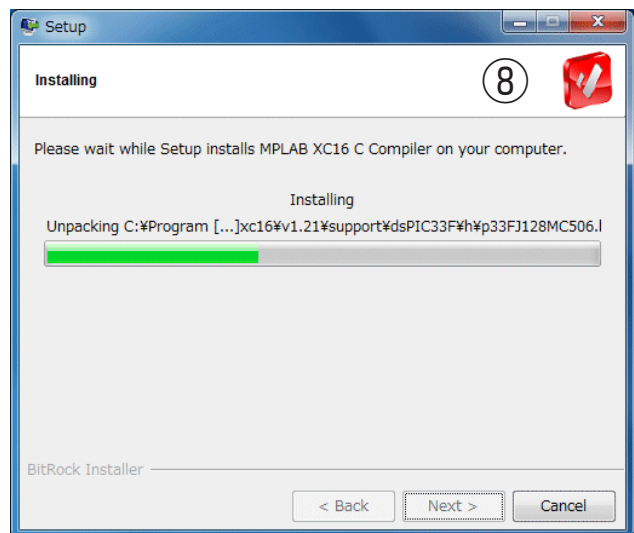
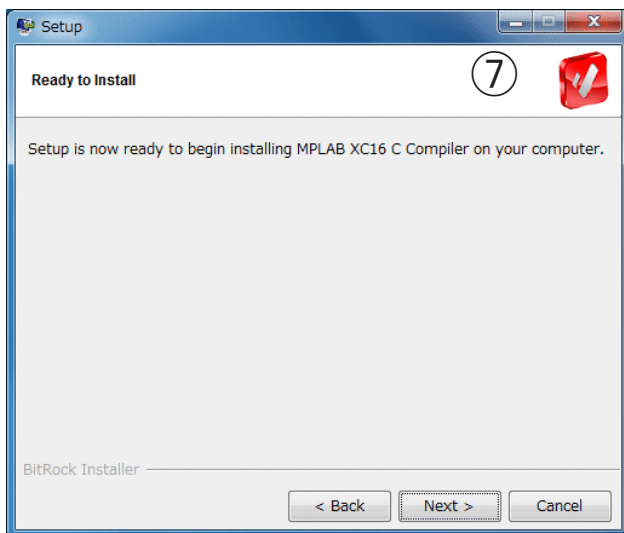
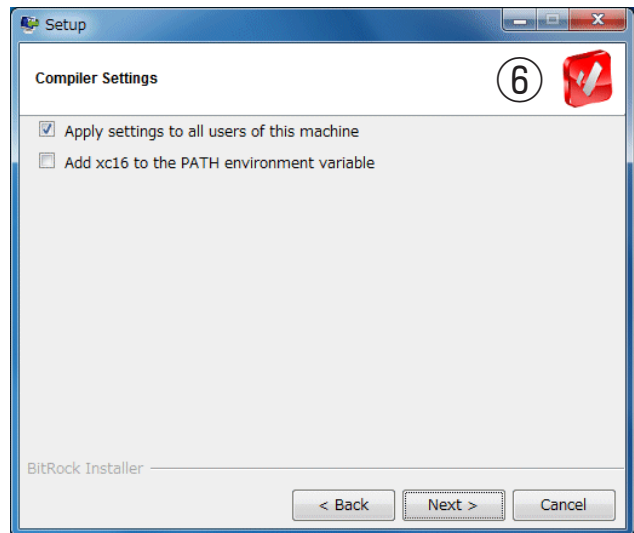
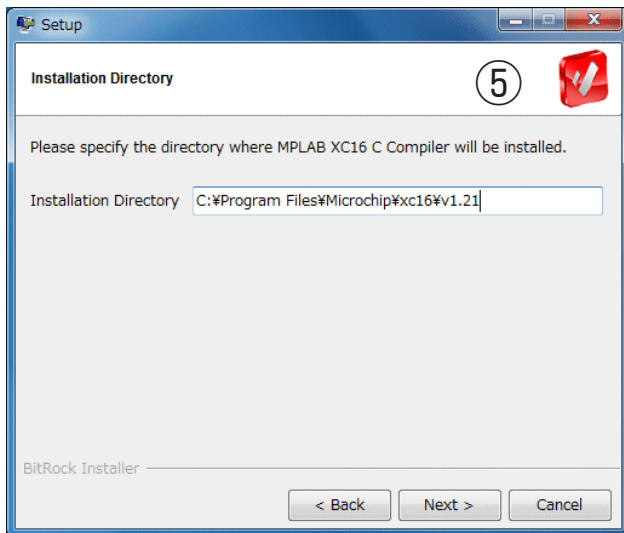
XC16 コンパイラもマイクロチップのホームページに登録してから利用しなければなりません。

マイクロチップ「<http://www.microchip.com/>」に一旦登録を行うようにお願いいたします。

XC16 のインストールは「xc8-v1.20-win.exe」を起動します。(名称は時期により異なる)

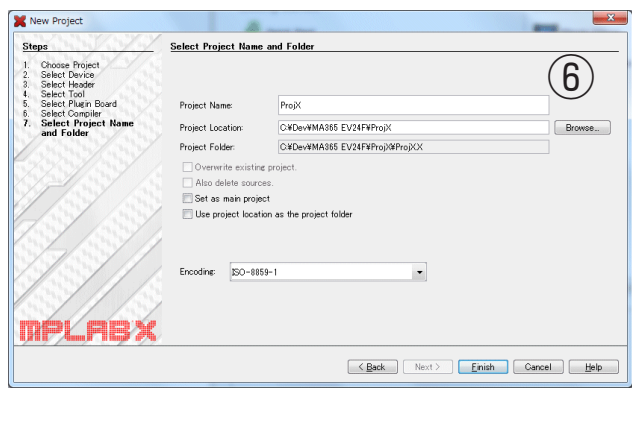
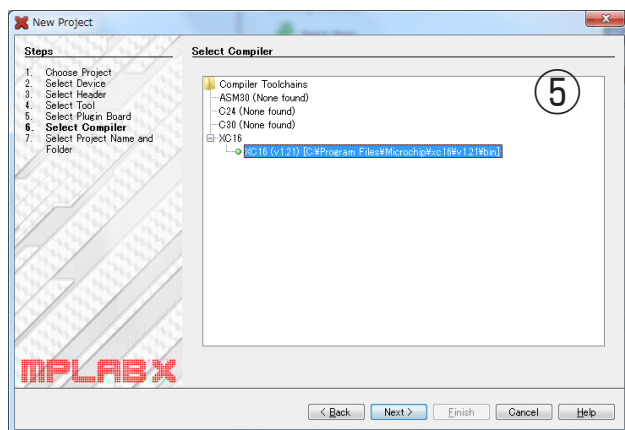
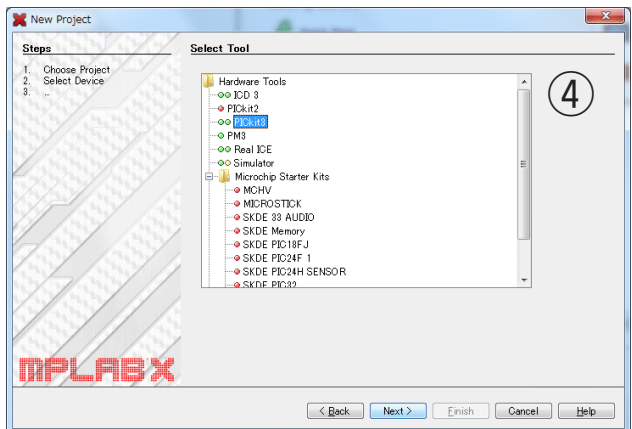
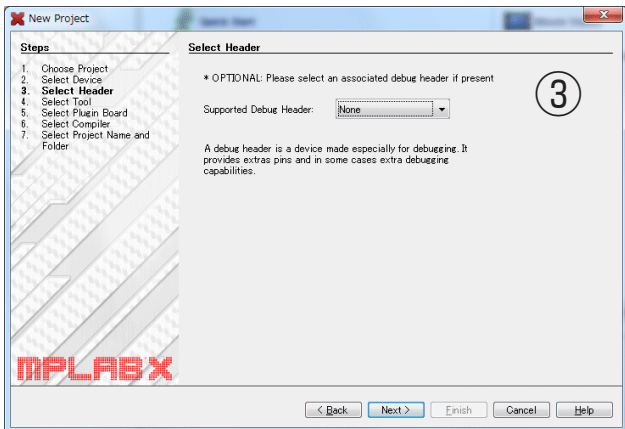
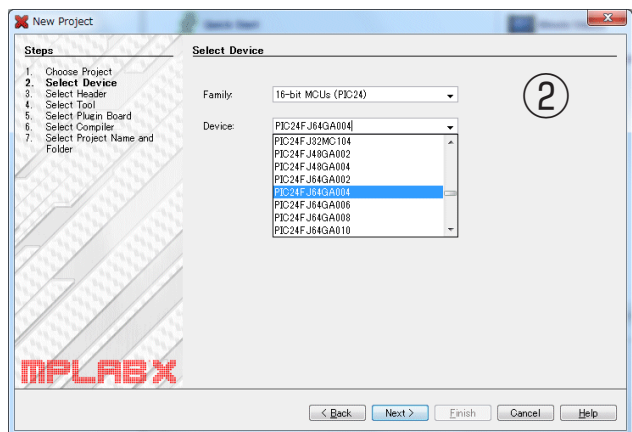
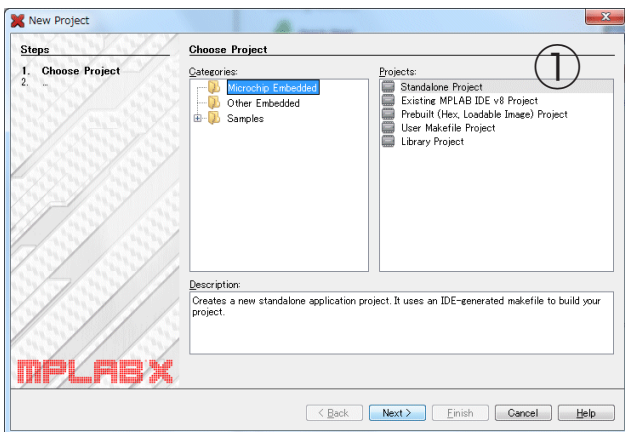
インストールは画面の指示に従いそのまま進めれば完了します。





7-2 XC16 プロジェクト

XC16 を使用したプロジェクトの作成手順を示します。
XC8 と同様の操作です。



インターネットにて製品の
詳細情報を提供しています。

<http://www.mal.jp>

ご注文・カタログ請求は
(株) マイクロアプリケーションラボラトリー
〒211-0053 川崎市中原区上小田中 6-24-10
TEL:044-733-2327 FAX:044-711-8609