



MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ご注意：この日本語版ドキュメントは、参考資料としてご使用の上、最新情報につきましても、必ず英語版オリジナルをご参照いただきますようお願いいたします。

マイクロチップテクノロジー社(以下、マイクロチップ社)デバイスのコード保護機能に関する以下の点にご留意ください。

- マイクロチップ社製品は、その該当するマイクロチップ社データシートに記載の仕様を満たしています。
- マイクロチップ社では、通常の条件ならびに仕様どおりの方法で使用した場合、マイクロチップ社製品は現在市場に流通している同種製品としては最もセキュリティの高い部類に入る製品であると考えております。
- コード保護機能を解除するための不正かつ違法な方法が存在します。マイクロチップ社の確認している範囲では、このような方法のいずれにおいても、マイクロチップ社製品をマイクロチップ社データシートの動作仕様外の方法で使用する必要があります。このような行為は、知的所有権の侵害に該当する可能性が非常に高いと言えます。
- マイクロチップ社は、コードの保全について懸念を抱いているお客様と連携し、対応策に取り組んでいきます。
- マイクロチップ社を含むすべての半導体メーカーの中で、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、マイクロチップ社が製品を「解読不能」として保証しているものではありません。

コード保護機能は常に進歩しています。マイクロチップ社では、製品のコード保護機能の改善に継続的に取り組んでいます。マイクロチップ社のコード保護機能を解除しようとする行為は、デジタルミレニアム著作権法に抵触する可能性があります。そのような行為によってソフトウェアまたはその他の著作物に不正なアクセスを受けた場合は、デジタルミレニアム著作権法の定めるところにより損害賠償訴訟を起こす権利があります。

本書に記載されているデバイス アプリケーションなどに関する情報は、ユーザーの便宜のためにのみ提供されているものであり、更新によって無効とされることがあります。アプリケーションと仕様の整合性を保証することは、お客様の責任において行ってください。マイクロチップ社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、商品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。マイクロチップ社は、本書の情報およびその使用に起因する一切の責任を否認します。マイクロチップ社デバイスを生命維持および/または保安のアプリケーションに使用することはデバイス購入者の全責任において行うものとし、デバイス購入者は、デバイスの使用に起因するすべての損害、請求、訴訟、および出費に関してマイクロチップ社を弁護、免責し、同社に不利益が及ばないようにすることに同意するものとし、暗黙的あるいは明示的を問わず、マイクロチップ社が知的財産権を保有しているライセンスは一切譲渡されません。

商標

Microchip の社名とロゴ、Microchip ロゴ、Accuron、dsPIC、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、rPIC、SmartShunt、UNI/O は、米国およびその他の国における Microchip Technology Incorporated の登録商標です。


FilterLab、Hampshire、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL、SmartSensor、The Embedded Control Solutions Company は、米国における Microchip Technology Incorporated の登録商標です。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、In-Circuit Serial Programming、ICSP、ICEPIC、Mindi、MiWi、MPASM、MPLAB Certified ロゴ、MPLIB、MPLINK、mTouch、nanoWatt XLP、PICkit、PICDEM、PICDEM.net、Pictail、PIC³² logo、PowerCal、PowerInfo、PowerMate、PowerTool、Real ICE、rFLAB、Select Mode、Total Endurance、TSHARC、WiperLock、ZENA は、米国およびその他の国における Microchip Technology Incorporated の商標です。

SQTP は米国における Microchip Technology Incorporated のサービスマークです。

その他、本書に記載されている商標は、各社に帰属します。

© 2009, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 再生紙を使用しています。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

マイクロチップ社では、Chandler および Tempe (アリゾナ州)、Gresham (オレゴン州) の本部、設計部およびウエハ製造工場そしてカリフォルニア州とインドのデザインセンターが ISO/TS-16949:2002 認証を取得しています。マイクロチップ社の品質システムプロセスおよび手順は、PIC[®] MCU および dsPIC[®] DSC、KEELOQ[®] コードホッピングデバイス、シリアルEEPROM、マイクロペリフェラル、不揮発性メモリ、アナログ製品に採用されています。また、マイクロチップ社の開発システムの設計および製造に関する品質システムは、ISO 9001:2000 の認証を受けています。

目次

序章	1
第 1 部 — 入門編	
第 1 章 概要	
1.1 はじめに	9
1.2 MPLAB ICD 3 インサーキット デバッガとは	9
1.3 MPLAB ICD 3 インサーキット デバッガの利点	10
1.4 MPLAB ICD 3 インサーキット デバッガのキット内容	10
1.5 デバイスと機能のサポート	11
第 2 章 動作原理	
2.1 はじめに	13
2.2 MPLAB ICD 3 インサーキット デバッガと MPLAB ICE 2000/4000 インサーキット エミュレータの違い	13
2.3 MPLAB ICD 3 インサーキット デバッガと MPLAB ICD 2 デバッガの 違い	13
2.4 デバッガとターゲットの通信	14
2.5 通信の接続	15
2.6 ICD 3 インサーキット デバッガを使用したデバッグ	17
2.7 デバッグを行うための条件	18
2.8 ICD 3 インサーキット デバッガを使用したプログラミング	20
2.9 デバッガが使用するリソース	20
第 3 章 インストール	
3.1 はじめに	21
3.2 ソフトウェアのインストール	21
3.3 USB デバイス ドライバのインストール	21
3.4 ターゲットの接続	22
3.5 ターゲット ボードのセットアップ	22
3.6 MPLAB IDE のセットアップ	23
第 4 章 全般的なセットアップ	
4.1 はじめに	25
4.2 MPLAB IDE ソフトウェアの起動	25
4.3 プロジェクトの作成	26
4.4 プロジェクトの表示	26
4.5 プロジェクトのビルド	26

4.6	コンフィギュレーション ビットの設定	27
4.7	MPLAB ICD 3 インサーキット デバッガをデバッガまたはプログラマ として設定	27
4.8	デバッガ/プログラマの制約	27
第 5 章	チュートリアル	
5.1	はじめに	29
5.2	環境のセットアップとデバイスの選択	30
5.3	アプリケーション コードの作成	30
5.4	プロジェクト ウィザードの実行	33
5.5	プロジェクトの表示	35
5.6	デバッグ オプションの確認	36
5.7	Hex ファイルの作成	37
5.8	デモ ボードのセットアップ	39
5.9	デバッグ用プログラム コードのロード	39
5.10	デバッグ コードの実行	40
5.11	ブレークポイントを使用したコードのデバッグ	40
5.12	アプリケーションのプログラミング	45
第 2 部	トラブルシューティング	
第 6 章	よく寄せられる質問 (FAQ)	
6.1	はじめに	49
6.2	動作のしくみに関する FAQ	49
6.3	不具合に関する FAQ	50
第 7 章	エラー メッセージ	
7.1	はじめに	53
7.2	特定の状況で表示されるエラー メッセージ	53
7.3	一般的な対処方法	58
第 3 部	リファレンス	
第 8 章	基本的なデバッグ機能	
8.1	はじめに	63
8.2	ブレークポイント	63
8.3	ストップウォッチ	63
第 9 章	デバッガ機能一覧	
9.1	はじめに	65
9.2	デバッグに関する機能	65
9.3	デバッグに関するダイアログ/ウィンドウ	68
9.4	プログラミングに関する機能	73
9.5	[Settings] ダイアログ	74

第 10 章	ハードウェア仕様	
10.1	はじめに	79
10.2	ハイライト	79
10.3	適合宣言書	79
10.4	USB ポート / 電源	80
10.5	MPLAB ICD 3 デバッガ	80
10.6	標準通信ハードウェア	81
10.7	ICD 3 テスト インターフェース ボード	83
10.8	ターゲット ボードに関する注意事項	83
用語集	85
索引	99
世界各国での販売およびサービス	102

MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

序章

顧客の皆様への注意

ドキュメントはすべて古くなります。本書も例外ではありません。マイクロチップ社のツールおよびマニュアルはユーザーのニーズを満たすために改良を重ねており、実際のダイアログやツールの内容が本書に記載されているものと異なる場合があります。最新のドキュメントを入手するには、弊社のウェブサイト (www.microchip.com) をご覧ください。

ドキュメントは「DS」番号で識別されています。この識別番号は、各ページのフッタ部分、ページ番号の前に記載されています。DS 番号の表記規則は「DSXXXXXA」で、「XXXXX」が文書番号、「A」が文書のリビジョンレベルを表しています。

開発ツールについての最新情報は、MPLAB® IDE のオンラインヘルプをご覧ください。[ヘルプ]メニューを選択して、次に[トピック]を選択すると、利用できるオンラインヘルプファイルのリストが表示されます。

はじめに

ここでは、MPLAB ICD 3 インサーキットデバッガの説明に入る前に、一般的な参考情報について説明します。この章の内容は次のとおりです。

- 本書の構成
- 本書で使用される表記
- 保証登録
- 推奨参考資料
- マイクロチップ社のウェブサイト
- 開発システムのお客様変更通知サービス
- 顧客サービス
- 改版履歴

本書の構成

本書では、MPLAB ICD 3 インサーキットデバッガを開発用ツールとして使用し、ターゲットボードのファームウェアのエミュレーションとデバッグ、およびデバイスへのプログラミングを行う方法を説明します。本書の構成は次のとおりです。

第 1 部 — 入門編

- **第 1 章 概要** — MPLAB ICD 3 インサーキットデバッガの概要、およびアプリケーション開発におけるその利点を紹介します。
- **第 2 章 動作原理** — MPLAB ICD 3 インサーキットデバッガの動作原理を説明します。構成オプションについても説明します。
- **第 3 章 インストール** — デバッガのソフトウェアおよびハードウェアのインストール方法を説明します。
- **第 4 章 一般的なセットアップ** — デバッガの使用に必要な MPLAB IDE のセットアップ方法を説明します。

- **第5章 チュートリアル**— デバッガの使用法をチュートリアル形式で簡単に説明します。

第2部 — トラブルシューティング

- **第6章 よく寄せられる質問 (FAQ)**— トラブルシューティングの参考として、よく寄せられる質問を紹介します。
- **第7章 エラーメッセージ**— エラーメッセージの一覧と、その解決方法を紹介합니다。

第3部 — リファレンス

- **第8章 基本的なデバッグ機能**— MPLAB IDE で MPLAB ICD 3 インサーキット デバッガをデバッガ/プログラマとして選択した場合に利用できる基本的なデバッグ機能を説明します。ここでは、ブレークポイント、ストップウォッチ、トリガ、リアルタイム ウォッチなどのデバッグ機能を紹介합니다。
- **第9章 デバッガ機能一覧**— MPLAB IDE で MPLAB ICD 3 デバッガをデバッガ/プログラマとして選択した場合に利用できるデバッグ機能を一覧形式にまとめます。
- **第10章 ハードウェア仕様**— MPLAB ICD 3 デバッガ システムのハードウェアおよび電氣的仕様を説明します。

本書で使用される表記

本書では以下の表記上の規則を使用しています。

表記上の規則

説明	意味	例
明朝フォント:		
斜体文字	参考資料	<i>MPLAB[®] IDE User's Guide</i>
	強調文字	... は 唯一 のコンパイラです ...
角括弧: []	ウィンドウ	[Output] ウィンドウ
	ダイアログ	[Settings] ダイアログ
	メニューの選択肢	[Enable Programmer] を選択
かぎ括弧: 「」	ウィンドウまたはダイアログのフィールド名	「Save project before build」
右山括弧 (>) を使用し、角括弧で囲まれた下線付きイタリックテキスト	メニューパス	<i>[File] > [Save]</i>
太字で角括弧に囲まれたテキスト	ダイアログのボタン	[OK] をクリックします。
	タブ	[Power] タブを選択します。
N ⁿ Rnnnn	Verilog 形式の数です。N が合計桁数、R が基数、n が桁を表します。	4 ^b 0010, 2 ^h F1
山括弧で囲まれたテキスト: <>	キーボードのキー	<Enter>, <F1> を押します。
クーリエフォント:		
通常のクーリエ	サンプルソースコード	#define START
	ファイル名	autoexec.bat
	ファイルパス	c:\mcc18\h
	キーワード	_asm, _endasm, static
	コマンド行オプション	-Opa+, -Opa-
	ビット値	0, 1
	定数	0xFF, 'A'
イタリッククーリエ	変数の引数	<i>file.o: file</i> は任意の有効なファイル名
角括弧: []	任意の引数	mcc18 [オプション] <i>file</i> [オプション]
中括弧とパイプ文字: {}	いずれかの引数を選択する場合 (OR 選択)	errorlevel {0 1}
省略記号: ...	繰り返されるテキスト	var_name [, var_name...]
	ユーザーが定義するコード	void main (void) { ... }

保証登録

同封の保証登録カードにご記入いただき、お早めにご郵送ください。この保証登録カードを送付されたお客様は、製品のアップデート版を受け取ることができるようになります。暫定ソフトウェアリリースはマイクロチップ社のウェブサイトで入手できます。

推奨参考資料

本書は、MPLAB ICD 3 インサーキット デバッガの使用方法を説明したものです。他にも役に立つ文書がありますので、以下に一覧でご紹介します。マイクロチップ社作成の以下のような文書がありますので、参考資料としておすすめします。

まず、この文書を最初にお読みください。この文書には、MPLAB ICD 3 を実際のターゲット デザインに使用する際の重要な注意事項が記載されています。

MPLAB ICD 3 インサーキット デバッガのリリース ノート

MPLAB ICD 3 インサーキット デバッガを使用する際は、MPLAB IDE のインストール先ディレクトリの Readmes サブディレクトリにある「Readme for MPLAB ICD 3 Debugger.htm」ファイル (HTML 形式ファイル) に記載されている最新情報をお読みください。リリース ノート (Readme ファイル) には、本書には記載されていない最新情報および既知の問題が記述されています。

Using MPLAB ICD 3 In-Circuit Debugger Poster (DS51765)

標準通信を利用して MPLAB ICD 3 インサーキット デバッガとターゲット ボードを接続する際のソフトウェアのインストールおよびハードウェアの接続の方法を記載したポスターです。

MPLAB ICD 3 インサーキット デバッガのオンライン ヘルプ ファイル

MPLAB ICD 3 インサーキット デバッガのすべてを解説したヘルプ ファイルが MPLAB IDE に含まれています。このファイルには、使用法、トラブルシューティング、ハードウェア仕様などが記載されています。印刷されたマニュアルよりも新しい内容が記載されている場合があります。また、各種デバイスのデバッグ予約リソースおよび制約についても記載しています。

Header Board Specification (DS51292)

この文書では、MPLAB ICD 3 インサーキット デバッガのヘッダの取り付けおよび使用方法を説明しています。ヘッダとは、一部デバイスで製品名の末尾が「-ICE」で終わる特別なデバイスを使用して、ピンやリソースの無駄なくデバッグを行えるようにするものです。

Transition Socket Specification (DS51194)

MPLAB ICE 2000/4000 デバイスのアダプタ、MPLAB ICD 2 のヘッダ、および MPLAB ICD 3 インサーキット デバッガのヘッダで使用可能なトランジションソケットの情報は、この文書を参照してください。

マイクロチップ社のウェブサイト

マイクロチップ社は同社のウェブサイト (www.microchip.com) でオンライン サポートを行っています。ウェブサイトにはご使用のブラウザでアクセスでき、下記の情報が含まれます。

- **製品サポート** – データシートと正誤表、アプリケーション ノートとサンプル プログラム、設計リソース、ユーザーガイド、ハードウェア サポート文書、最新リリース ソフトウェア、ソフトウェア アーカイブ
- **一般的なテクニカル サポート** – よく寄せられる質問 (FAQ)、テクニカル サポートの依頼、オンライン ディスカッション グループ、マイクロチップ社 コンサルタント プログラム メンバーのリスト
- **マイクロチップのビジネス** – 製品選択と注文ガイド、マイクロチップ社の最新プレスリリース、セミナーとイベントのリスト、マイクロチップの営業オフィス、代理店、工場代理人のリスト

開発システムのお客様変更通知サービス

マイクロチップ社のお客様変更通知サービスは、お客様がマイクロチップ社製品の最新情報を入手できるようにします。加入者は、指定した製品ファミリーや興味のある開発ツールに関する変更、更新、リビジョン、正誤表があるときは常に E メール通知を受け取ることができます。

登録には、マイクロチップのウェブサイト (www.microchip.com) にアクセスして、[お客様変更通知] をクリックし、登録指示に従うだけです。

開発システム製品のグループ カテゴリは、以下のとおりです。

- **コンパイラ** – マイクロチップ社製 C コンパイラ、アセンブラ、リンカおよび他の言語ツールの最新情報です。これには、MPLAB C コンパイラ全製品、MPLAB アセンブラ全製品 (MPASM™ アセンブラを含む)、MPLAB リンカ全製品 (MPLINK™ オブジェクトリンカを含む)、MPLAB ライブラリアン全製品 (MPLIB™ オブジェクトライブラリアンを含む) が含まれます。
- **エミュレータ** – マイクロチップ社製インサーキットエミュレータの最新情報です。これには、MPLAB REAL ICE™、MPLAB ICE 2000、MPLAB ICE 4000 インサーキットエミュレータが含まれます。
- **インサーキット デバッガ** – マイクロチップ社製インサーキット デバッガの最新情報です。これには、MPLAB ICD 2 インサーキット デバッガと PICKit™ 2 Debug Express が含まれます。
- **MPLAB® IDE** – 開発システムツール向け Windows® 統合開発環境であるマイクロチップ社製 MPLAB IDE の最新情報です。このリストでは、MPLAB IDE、MPLAB IDE Project Manager、MPLAB Editor、MPLAB SIM シミュレータ、および一般的な編集およびデバッグ機能が取り上げられています。
- **プログラマ** – マイクロチップ社製プログラマの最新情報です。これには、MPLAB PM3 および PRO MATE II デバイス プログラマ、PICSTART® Plus、PICKit 1 および 2 開発プログラマが含まれます。

顧客サービス

マイクロチップ製品のユーザーは、いくつかのチャンネルを介してサポートを受けられます。

- 販売代理店または販売担当者
- 地域の営業所
- フィールドアプリケーション エンジニア (FAE)
- 技術サポート

技術サポートを得るには、販売代理店か販売担当者、フィールドアプリケーション エンジニア (FAE) に連絡してください。地域の営業所でもお客様の手助けをします。営業所と所在地のリストが本文書の最後に記載されています。

技術サポートは弊社ウェブサイト (<http://support.microchip.com>) を通して受けることができます。

改版履歴

リビジョン A (2008 年 9 月)

本文書の初版リリース



MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

第 1 部 — 入門編

第 1 章 概要	9
第 2 章 動作原理	13
第 3 章 インストール	21
第 4 章 全般的なセットアップ	25
第 5 章 チュートリアル	29

MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

第 1 章 概要

1.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガ システムの概要を説明します。

- MPLAB ICD 3 インサーキット デバッガとは
- MPLAB ICD 3 インサーキット デバッガの利点
- MPLAB ICD 3 インサーキット デバッガのキット内容
- デバイスと機能のサポート

1.2 MPLAB ICD 3 インサーキット デバッガとは

MPLAB ICD 3 インサーキット デバッガは、Windows® PC 上で動作する MPLAB IDE (v8.15 以降) によって制御されるインサーキット デバッガです。MPLAB ICD 3 インサーキット デバッガは、開発エンジニアのツールスイートに欠かせない重要な要素です。ソフトウェア開発からハードウェア インテグレーションまで幅広い用途に対応します。

MPLAB ICD 3 インサーキット デバッガは、ICSP™ (In-Circuit Serial Programming™) および Enhanced ICSP の 2 線式シリアル インターフェースを使用してマイクロチップ社の PIC® マイクロコントローラ (MCU) および dsPIC® デジタルシグナルコントローラ (DSC) のハードウェアおよびソフトウェア開発を行える多機能デバッガ システムです。

このデバッガ システムは専用のデバッガ チップではなくエミュレーション回路を内蔵したデバイスを使用してエミュレーションを行うので、実際のデバイスと同じようにコードが実行されます。ターゲット デバイスで利用できる機能はすべてインタラクティブに利用でき、MPLAB IDE インターフェースを利用して設定や変更が行えます。

MPLAB ICD 3 デバッガは、多彩なデバッグ機能を内蔵した組み込みプロセッサのエミュレーション用として開発されています。これらのプロセッサは、従来のシステムプロセッサと比べて次のような違いがあります。

- プロセッサの最大動作速度でデバッグが行える
- I/O ポートのデータ入力を取り込むことができる

デバッガとして使用する以外に、MPLAB ICD 3 インサーキット デバッガ システムは開発プログラマとしても使用できます。

1.3 MPLAB ICD 3 インサーキット デバッガの利点

MPLAB ICD 3 インサーキット デバッガには次の利点があります。

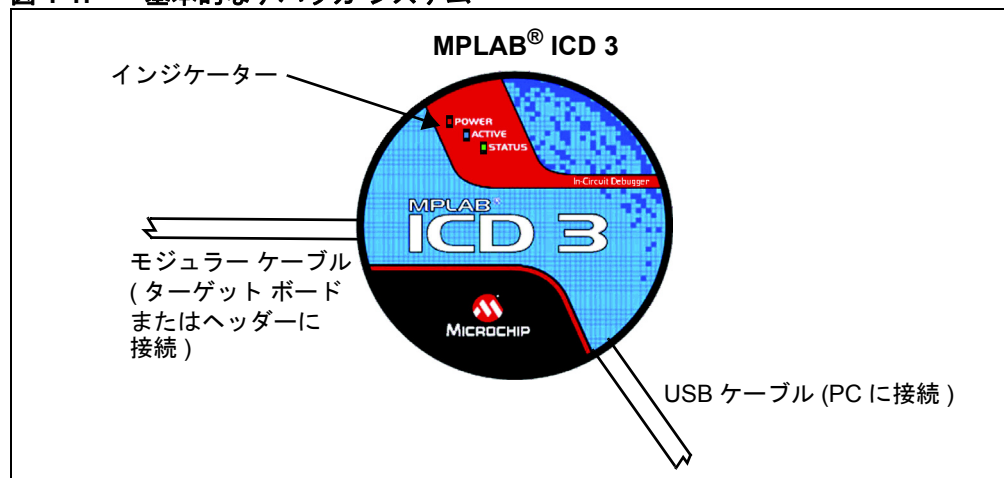
- 実際のハードウェア上でアプリケーション デバッグをリアルタイムに行える
- ハードウェア ブレークポイントを利用したデバッグ
- ソフトウェア ブレークポイントを利用したデバッグ
- 内部イベントに基づいたブレークポイントの設定
- 内部ファイル レジスタの観察
- フルスピードでのエミュレーション
- デバイスへの書き込み

1.4 MPLAB ICD 3 インサーキット デバッガのキット内容

MPLAB ICD 3 インサーキット デバッガ システムのキットには次のものが含まれます。

1. MPLAB ICD 3 (インジケータ付き)
2. USB ケーブル (デバッガと PC の通信、およびデバッガへの電源供給用)
3. MPLAB ICD 3 とヘッダ モジュールまたはターゲット ボードを接続するためのケーブル (MPLAB ICD 2 と同等品)
4. MPLAB IDE クイックスタートガイド (DS51281)
5. MPLAB IDE ソフトウェアおよびオンライン ドキュメントを収録した CD-ROM
6. ICD 3 テスト インターフェース ボード

図 1-1: 基本的なデバッグ システム



その他、以下の別売りオプションのハードウェアもあります。

- トランジション ソケット
- ICD ヘッダ
- MPLAB プロセッサ拡張キット

1.5 デバイスと機能のサポート

各デバイスとその機能のサポート状況 (予定を含む) を、表 1-1 と表 1-2 に示します。

表 1-1: 32 ビットおよび 16 ビット (データ メモリ) デバイス

Feature	PIC32MX	dsPIC33F, PIC24F/H	dsPIC30F SMPS ⁽¹⁾	dsPIC30F
Reset application	C	C	C	C
Run, Halt	C	C	C	C
Single Step	C	C	C	C
Animate	C	C	C	C
Full Speed Emulation	C	C	C	C
Hardware Breakpoints	C	C	C	C
Advanced Breakpoints	C	C	C	C
Software Breakpoints	N	C	C	C
Peripheral Freeze ⁽²⁾	C	C	C	C
Break on data fetch or write	C	C	C	C
Break on Stack overflow	C	C	C	C
Stopwatch	C	C	C	N
Pass Counter	C	C	C	C
WDT overflow	C	C	C	N
Standard Speed Comm.	C	C	C	C
Processor Pak	N	F	F	N

記号の説明:

C = サポート済み

D = デバイスによってはサポート

F = 現在は未サポート、しかし将来サポート予定

N = サポートなし

注 1: 現在の SMPS (スイッチング電源) デバイスには、dsPIC30F1010/2020/2023 があります。

2: この機能は、選択したデバイスによって動作が異なります。

MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

表 1-2: 8 ビット (データ メモリ) デバイス

Feature	PIC18FXXJ	PIC18F, PIC18F Enh, PIC18FXXK	PIC12F, PIC16F
Reset application	C	C	C
Run, Halt	C	C	C
Single Step	C	C	C
Animate	C	C	C
Full Speed Emulation	C	C	C
Hardware Breakpoints	C	C	C
Advanced Breakpoints	C	C	N
Software Breakpoints	C	C	N
Peripheral Freeze ⁽¹⁾	C	C	C
Break on data fetch or write	C	C	N
Break on Stack overflow	C	C	N
Stopwatch	C	N	N
Pass Counter	C	C	N
WDT overflow	C	N	N
Standard Speed Comm.	C	C	C
Processor Pak	F	F	F

記号の説明:

C = サポート済み

F = 現在は未サポート、しかし将来サポート予定

N = サポートなし

注 1: この機能は、選択したデバイスによって動作が異なります。

第 2 章 動作原理

2.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガ システムの動作のしくみについて簡単に説明します。ここでの目的は、ICD 3 インサーキット デバッガを使用してエミュレーションとプログラミングを行えるよう、ターゲット ボードの設計に必要な情報を提供することにあります。また、万一問題が発生してもすぐに解決できるように、インサーキット エミュレーションおよびプログラミングの基本的な動作原理を説明します。

- MPLAB ICD 3 インサーキット デバッガと MPLAB ICE 2000/4000 インサーキット エミュレータの違い
- MPLAB ICD 3 インサーキット デバッガと MPLAB ICD 2 デバッガの違い
- デバッガとターゲットの通信
- 通信の接続
- ICD 3 インサーキット デバッガを使用したデバッグ
- デバッグを行うための条件
- ICD 3 インサーキット デバッガを使用したプログラミング
- デバッガが使用するリソース

2.2 MPLAB ICD 3 インサーキット デバッガと MPLAB ICE 2000/4000 インサーキット エミュレータの違い

MPLAB ICD 3 インサーキット デバッガ システムは次世代のインサーキット デバッガ (ICD) システムです。MPLAB ICD 3 インサーキット デバッガ システムには、従来のインサーキット エミュレータ システム (MPLAB ICE 2000/4000 など) と大きく異なる点が 1 つあります。それは、標準の量産デバイスをエミュレーション デバイスと同じように使用できるという点です。

これは、量産品のシリコンとエミュレーション シリコンの違い (エラータ) が排除されるという意味で大きな利点となります。また、従来のエミュレータ システムでは内部バスをオフチップの外部メモリに転送する必要があり、フル スピードのエミュレーションが行えないため、高速化が進んだ最近のデバイスでは大きなボトルネックとなっていました。

この他、量産品のシリコンが発売されてからエミュレーション シリコンが提供されるまで待つ必要がないのも重要な利点の 1 つです。更に、量産用ボードで問題が発生しても、トランジション ソケットを装着して複雑なケーブリングやセットアップを行う必要がなく、簡単にアプリケーションにアクセスしてデバッグが行えるという利点もあります。

2.3 MPLAB ICD 3 インサーキット デバッガと MPLAB ICD 2 デバッガの違い

MPLAB ICD 3 インサーキット デバッガ システムは、MPLAB ICD 2 インサーキット デバッガ システムと基本的な動作は同じですが、動作速度と機能が改善されています。その他、MPLAB ICD 3 の主な特長は次のとおりです。

- Hi-Speed USB に対応
- USB バス パワー対応
- ハードウェア アクセラレータ

- プログラマブルな電源電圧
- RS-232 ポートを廃止
- 診断用のセルフテスト インターフェース ボードが添付

2.4 デバッガとターゲットの通信

ここからは、デバッガ システムの構成について説明します。

注意

ソフトウェアおよびUSBドライバのインストールが完了するまでハードウェアを接続しないでください。また、ポッドまたはターゲットに電源を供給している間はハードウェアの接続を変更しないでください。

ICSP によるデバイスとの標準通信

デバッガ システムは、標準の ICSP 通信を使用してプログラミングおよびデバッグを行うように構成できます。この 6 ピン接続は、MPLAB ICD 2 インサーキット デバッガと同じものです。

モジュラ ケーブルは、(1) ターゲット デバイスがターゲット ボードに直接実装されている場合はターゲットの対応ソケットに挿入し (図 2-1)、(2) ターゲット ボードに標準アダプタ / ヘッダ ボードのコンボ (プロセッサ パックとして供給) を接続している場合は、標準アダプタの対応ソケットに挿入します (図 2-2)。

注: 旧式のヘッダ ボードで、8 ピンコネクタではなく 6 ピン (RJ-11) コネクタを採用しているものはデバッガにそのまま接続できます。

標準通信の詳細は、第 10 章「ハードウェア仕様」を参照してください。

図 2-1: 標準のデバッガ システム — オンボードの ICE 回路内蔵デバイスの場合

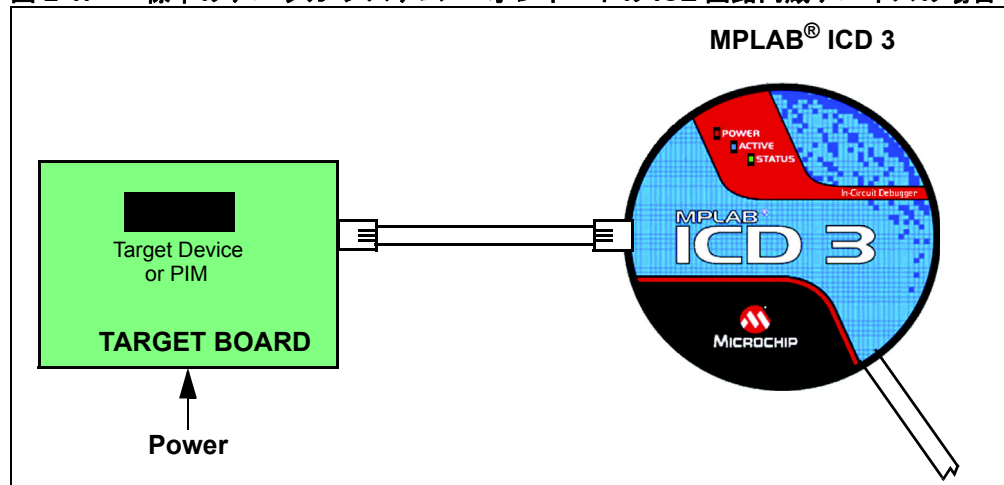
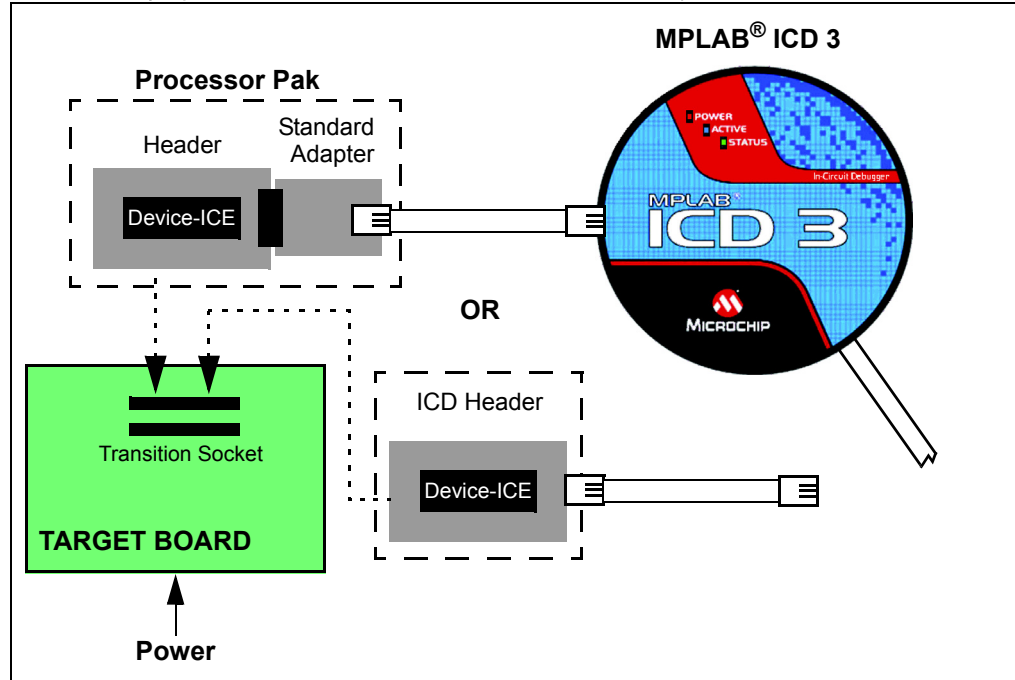


図 2-2: 標準のデバッガ システム — ICE デバイスの場合



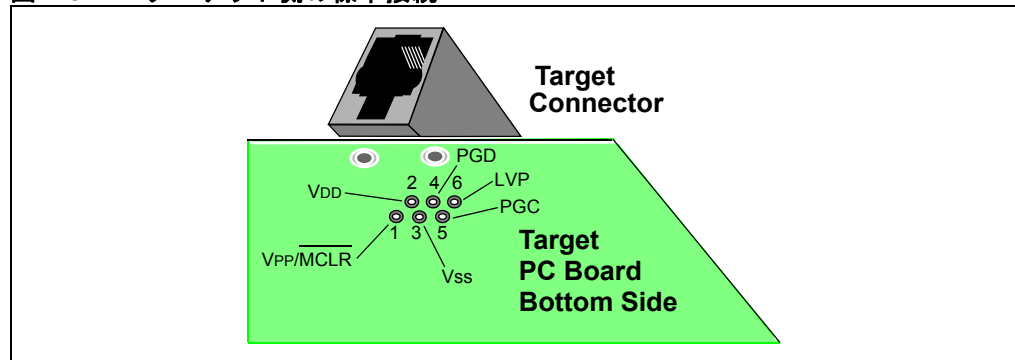
2.5 通信の接続

2.5.1 ターゲットとの標準通信の接続

MPLAB ICD 3 インサーキット デバッガとターゲット デバイスは、RJ-11 コネクタを使用して 6 線式モジュラ インターフェイス ケーブルで接続します。ターゲット ボードのはんだ面から見たコネクタのピン番号を図 2-3 に示します。

注: デバッガとターゲットをケーブルで接続すると、反対のピン番号どうしが接続されることとなります。つまり、ケーブルの片方のピン 1 は、もう片方のピン 6 に接続されます。**10.6.2.3 項「モジュラ ケーブルの仕様」**を参照してください。

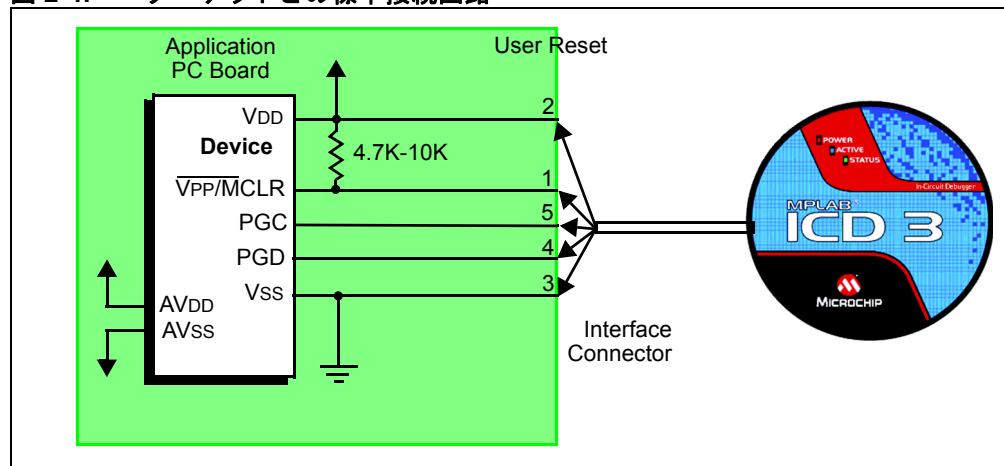
図 2-3: ターゲット側の標準接続



2.5.2 ターゲットとの接続回路

MPLAB ICD 3 インサーキット デバッガとターゲット ボードを接続する際の配線を図 2-4 に示します。この図には、コネクタからターゲットのプリント基板への配線も示されています。V_{PP}/MCLR ラインから V_{DD} へプルアップ抵抗 (通常、約 10 kΩ) を接続しておくことを推奨します。こうすると、このラインを Low にするとデバイスをリセットできます。

図 2-4: ターゲットとの標準接続回路



2.5.3 ターゲットの自己給電

以下の説明では、ピン 1 (V_{PP}/MCLR)、ピン 5 (PGC)、ピン 4 (PGD) の 3 つのみがアクティブで、これ以外はデバッガの動作には関係しません。ただし参考までに図 2-4 にはピン 2 (V_{DD}) とピン 3 (V_{SS}) も示しています。MPLAB ICD 3 では、ターゲット デバイスの電源をデバッガから供給する構成と、ターゲット側に外付けの電源を使用する構成の 2 種類が可能です。

このうち、推奨されるのはターゲット アプリケーション側に外付けの電源を使用する構成です。この構成では、デバッガがターゲットの V_{DD} を検出して、ターゲットが低電圧動作の場合にレベル変換を実行します。デバッガが V_{DD} ライン (インターフェース コネクタのピン 2) の電圧を検出できない場合、デバッガは動作しません。

2.5.4 デバッガから電源を供給する場合

デバッガの内部電源には、(1) 電圧範囲が狭い (3V ~ 5V)、(2) 供給できる電流量が最大 100 mA までという 2 つの制約があります。この電源供給方法は、デバイスの V_{DD} が他のアプリケーション回路から分離されて独立プログラミングを行えるようなきわめて小規模のアプリケーションには適していますが、USB バス パワーとして PC から流れる電流が大きくなるため、一般的な用途には推奨できません。

デバッガがターゲットの V_{DD} を検出して、ターゲットが低電圧動作の場合にレベル変換を行うことに注意してください。デバッガが V_{DD} ライン (インターフェース コネクタのピン 2) の電圧を検出できない場合、デバッガは動作しません。

すべてのデバイスに AV_{DD} および AV_{SS} ラインがあるわけではありませんが、ある場合には、これらのラインをすべて適切なレベルに接続しないとデバッガは正しく動作しません。

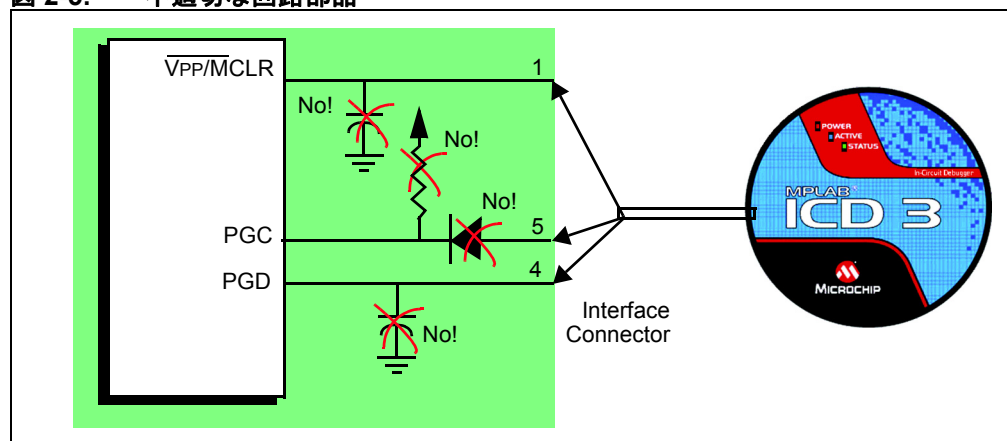
一般に、VDD/AVDD および VSS/AVSS ラインはすべて適切なレベルに接続することを推奨します。また、VCAP ラインのあるデバイス (PIC18FXXJ など) では、このラインを適切なコンデンサまたはレベルに接続するようにしてください。

注： 配線は非常にシンプルです。問題が発生する場合は、これらの重要なラインに他の接続や部品を取り付けているために MPLAB ICD 3 インサーキット デバッガ システムの正常な動作が妨げられているのが主な原因です。これについては次項で説明します。

2.5.5 デバッガの正常動作を妨げる回路

図 2-5 は、デバッガのアクティブなラインに MPLAB ICD 3 インサーキット デバッガ システムの正常動作を妨げる部品を取り付けた様子を示したものです。

図 2-5: 不適切な回路部品



具体的には、次のガイドラインに従う必要があります。

- PGC/PGD にはプルアップ抵抗を使用しない — これらのラインはデバッガ内部で 4.7 kΩ の抵抗でプルダウンしているため、電圧レベルが低下してしまいます。
- PGC/PGD にはコンデンサを使用しない — プログラミングおよびデバッグの通信 (の際の) データおよびクロック ラインの高速遷移が遅れてしまいます。
- MCLR にはコンデンサを使用しない — VPP の高速遷移が遅れてしまいます。通常、シンプルなプルアップ抵抗で十分です。
- PGC/PGD にダイオードを使用しない — デバッガとターゲットデバイスの双方向通信が行えなくなってしまいます。

2.6 ICD 3 インサーキット デバッガを使用したデバッグ

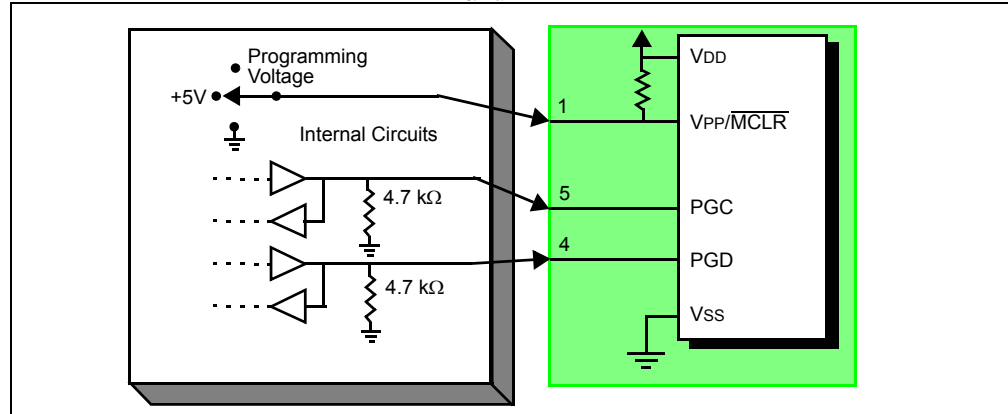
MPLAB ICD 3 インサーキット デバッガ システムでのデバッグには、2 段階の手順があります。まず、アプリケーションをターゲットデバイスにプログラミングします。次に、ターゲットのフラッシュ デバイスに内蔵されたインサーキット デバッグ ハードウェアを使用してアプリケーションプログラムを実行、テストします。これら 2 つの手順を実行するには、MPLAB IDE の操作に直接関連します。

1. コードをターゲット デバイスにプログラミングして、専用のデバッグ機能を有効にする (詳細は次項を参照)。
2. デバッガを使用してブレークポイントを設定し、実行する。

ターゲット デバイスに正しくプログラミングが行えないと、MPLAB ICD 3 インサーキット デバッガではデバッグが行えません。

図 2-6 に、プログラミングに必要な配線を示します。これは図 2-4 と同じものですが、分かりやすくするためにデバッガの VDD ラインと VSS ラインは省略しています。

図 2-6: プログラミング時の正しい接続



この図は、MPLAB ICD 3 インサーキット デバッガの内部インターフェース回路の一部を簡単に示したものです。プログラミング時はターゲット デバイスにクロックは必要ありませんが、電源を供給する必要があります。プログラミングを行う際、デバッガは V_{PP}/\overline{MCLR} をプログラミング レベルにし、PGC にクロック パルス、PGD にシリアル データを送出します。そしてターゲット デバイスに正しくプログラミングされたことを確認するために、PGC にクロックを送出し、PGD からデータを読み出します。これは、ターゲット デバイスの ICSP プロトコルに従って行われます。

2.7 デバッグを行うための条件

MPLAB ICD 3 インサーキット デバッガ システムでデバッグ (ブレイクポイントの設定、レジスタ内容の表示など) を行うには、以下の条件をすべて満たす必要があります。

- デバッガが PC に接続されていること。USB ケーブルを利用して PC からデバッガに電源が供給されていること、および USB ケーブルを利用して MPLAB IDE ソフトウェアとデバッガが通信を行えることが必要です。詳細は、第 3 章「インストール」を参照してください。
- モジュラ インターフェース ケーブル (または同等品) を利用してデバッガがターゲット デバイスの V_{PP} 、PGC、PGD ピンに接続されていること (詳細は前述)。VSS と VDD もデバッガとターゲット デバイスの間で接続されている必要があります。
- ターゲット デバイスに電源が供給されており、オシレータが正しく動作していること。何らかの理由でターゲット デバイスが動作しない場合、MPLAB ICD 3 インサーキット デバッガでデバッグを行うことはできません。
- ターゲット デバイスのコンフィギュレーションワードが正しくプログラミングされていること。
 - ターゲットの設計に応じてオシレータのコンフィギュレーション ビットを RC、XT などに正しく設定しておく必要があります。
 - ウォッチドッグ タイマがデフォルトで有効になっているデバイスでは、無効にしておく必要があります。
 - ターゲット デバイスのコード保護は無効にしておく必要があります。
 - ターゲット デバイスのテーブル読み出し保護は無効にしておく必要があります。
- LVP は無効にしておくこと。

上記の条件が満たされていることを確認したら、次の項に進んでください。

- デバッグ開始までの操作手順
- デバッグの詳細

2.7.1 デバッグ開始までの操作手順

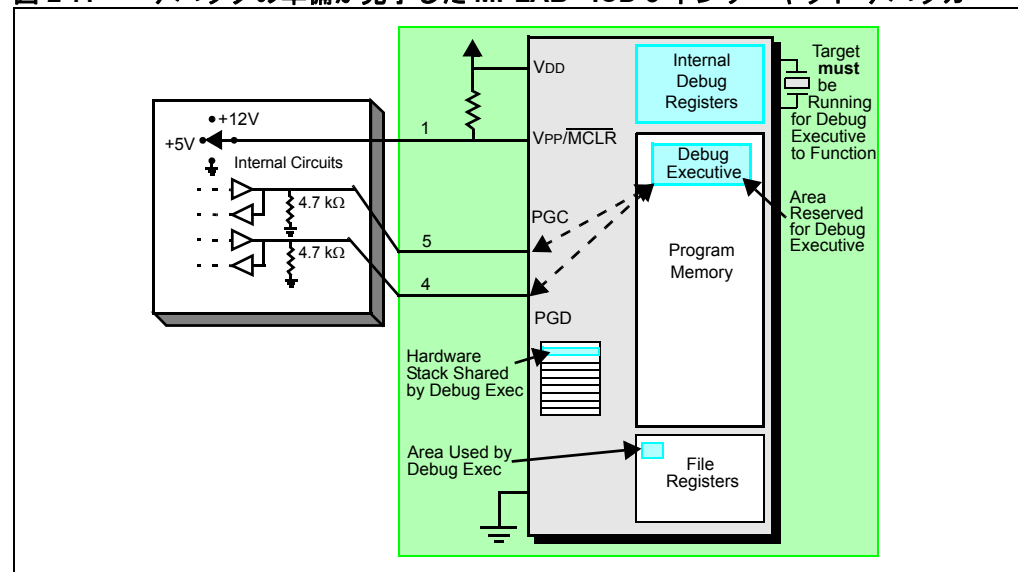
デバッグを行うための条件に記載した条件がすべて満たされていれば、MPLAB IDE のメニューで **[Debugger] > [Select Tool] > [MPLAB ICD 3]** の順にクリックして MPLAB ICD 3 インサーキット デバッガをデバッガに指定すると、次の操作を実行できるようになります。

- **[Project] > [Build Configuration] > [Debug]** の順にクリックすると、アプリケーション コードのコンパイルとアセンブルが行えます。
- **[Debugger] > [Program]** の順にクリックすると、前述の ICSP プロトコルを利用してアプリケーション コードをデバイスのメモリにプログラミングできます。
- サイズの小さい「デバッグ実行プログラム」がターゲットデバイスのプログラム メモリの上位アドレスにロードされます。デバッグ実行プログラムをプログラム メモリにロードしておく必要があるため、この予約領域をアプリケーション プログラムで使用することはできません。一部のデバイスには、デバッグ実行プログラム専用のメモリ領域が用意されたものもあります。詳細はデバイスのデータシートで確認してください。
- ターゲットデバイスの「インサーキット デバッグ」専用レジスタが有効になります。これにより、デバッガからデバッグ実行プログラムを有効にできるようになります。
- $V_{PP}/MCLR$ ラインを Low にしておくこと、ターゲット デバイスはリセット状態に保持されます。

2.7.2 デバッグの詳細

図 2-7 に、デバッグ準備が完了した MPLAB ICD 3 インサーキット デバッガ システムを示します。

図 2-7: デバッグの準備が完了した MPLAB® ICD 3 インサーキット デバッガ



一般に、アプリケーションプログラムが正しく動作するかどうかを確認するには、プログラムコードの最初の方にブレークポイントを設定します。MPLAB IDE のユーザー インターフェイスでブレークポイントを設定すると、ブレークポイントのアドレスがターゲット デバイス内部のデバッグ専用レジスタに保存されます。PGC および PGD 上のコマンドは、これらレジスタと直接通信してブレークポイントのアドレスを設定します。

通常は、次に MPLAB IDE で **[Debugger] > [Run]** の順にクリックするか、または **[Run]** アイコン (右向きの三角形) をクリックします。すると、デバッガからの命令によってデバッグ実行プログラムが実行されます。ターゲットはリセット ベクタか

ら動作を開始し、プログラムカウンタが内部デバッグレジスタに保存されたブレークポイントのアドレスに達すると実行が停止します。

ブレークポイントのアドレスの命令が実行されると、割り込みと同じようなメカニズムでターゲットデバイスのインサーキット-デバッグメカニズムが「起動」してデバイスのプログラムカウンタをデバッグ実行プログラムに転送し、これによってユーザーのアプリケーションが停止します。デバッガはPGCとPGDを利用してデバッグ実行プログラムと通信を行い、ブレークポイントの状態に関する情報を取得してMPLAB IDEに送信します。この後、MPLAB IDEは一連の問い合わせをデバッガに送信し、ファイルレジスタの内容やCPUの状態など、ターゲットデバイスに関する情報を取得します。これらのクエリは、最終的にはデバッグ実行プログラムによって実行されます。

デバッグ実行プログラムは、プログラムメモリ内のアプリケーションとまったく同じように動作します。一時変数を保存するためにスタックの一部を使用します。オシレータがない、電源の接続が不正、ターゲットボードの短絡など、何らかの理由でデバイスが動作しない場合は、デバッグ実行プログラムとMPLAB ICD 3 インサーキットデバッガの通信が行えず、MPLAB IDEにエラーメッセージが表示されます。

プログラム実行を停止するもう1つの手段として、MPLAB IDEの[Halt]ボタン([Run]ボタンの右にある一時停止のアイコン)をクリックするという方法もあります。するとPGCラインとPGDラインが切り替わり、ターゲットデバイスのインサーキット-デバッグメカニズムによってプログラムカウンタがプログラムメモリ内のユーザーコードからデバッグ実行プログラムに切り替わります。この場合もターゲットアプリケーションのプログラム実行が停止し、MPLAB IDEはデバッガとデバッグ実行プログラムの間で行われる通信を利用してターゲットデバイスの状態を問い合わせます。

2.8 ICD 3 インサーキット デバッガを使用したプログラミング

MPLAB ICD 3 インサーキットデバッガをプログラマとして使用して、実際のデバイス(すなわち、ヘッダボードに装着された-ICE/-ICD以外のデバイス)に書き込みを行うことができます。[\[Programmer\]](#) > [\[Select Programmer\]](#) の順にクリックして「MPLAB ICD 3」を選択した後、MPLAB IDEのツールバーで[\[Build Configuration\]](#) リストボックスから「Release」を選択してアプリケーションコードのコンパイル/アセンブルを実行します。または、[\[Project\]](#) > [\[Build Configuration\]](#) > [\[Release\]](#) の順にクリックしても「Release」に設定できます。

デバッガをプログラマとして使用する場合は、デバッグ機能はすべてオフになるか切り離されます。[\[Programmer\]](#) > [\[Program\]](#) の順にクリックしてデバイスへのプログラミングを行うと、MPLAB IDEによってインサーキットデバッグレジスタが無効にされるため、MPLAB ICD 3 インサーキットデバッガはターゲットアプリケーションコードとコンフィギュレーションビット(および一部デバイスで選択した場合はEEPROMデータ)のみをターゲットデバイスにプログラミングします。デバッグ実行プログラムはロードされません。プログラマとして使用する場合、デバッガはMCLRラインをトグルしてターゲットをリセットおよび開始することしか行えません。ブレークポイントを設定することはできず、レジスタの内容を確認したり変更したりすることもできません。

MPLAB ICD 3 インサーキットデバッガシステムは、ICSPを利用してターゲットをプログラミングします。Vpp、PGC、PGDラインは、前述したとおりに接続してください。プログラミング時にはクロックは不要です。また、コード保護、ウォッチドッグタイマの有効化、テーブル読み出し保護を含め、プロセッサのすべてのモードをプログラミングできます。

2.9 デバッガが使用するリソース

各デバイスでデバッガが使用するリソースの一覧は、MPLAB IDEにあるMPLAB ICD 3 インサーキットデバッガのオンラインヘルプファイルを参照してください。

第 3 章 インストール

3.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガ システムのインストール方法を説明します。

- ソフトウェアのインストール
- USB デバイス ドライバのインストール
- ターゲットの接続
- ターゲット ボードのセットアップ
- MPLAB IDE のセットアップ

3.2 ソフトウェアのインストール

MPLAB IDE ソフトウェアをインストールするには、まず最新の MPLAB IDE インストール ファイル (MPxxxxxx.exe。xxxxxx は MPLAB IDE のバージョン) をマイクロチップ社のウェブ サイト (www.microchip.com) からダウンロードするか、MPLAB IDE の CD-ROM (DS51123) を入手してください。次にこのインストール ファイルを実行し、画面の指示に従って MPLAB IDE をインストールします。

注： MPLAB ICD 3 インサーキット デバッガを使用するには、MPLAB IDE v8.15 以降が必要です。

3.3 USB デバイス ドライバのインストール

MPLAB IDE をインストールすると MPLAB ICD 3 インサーキット デバッガ用の USB デバイス ドライバもインストールされます。このため、MPLAB IDE をインストール後に MPLAB ICD 3 インサーキット デバッガを USB ケーブルで PC に接続すると、Windows の「新しいハードウェアの検出ウィザード」が開始し、ドライバが自動的にインストールされます。

USB デバイス ドライバのインストールの詳細は、次のファイルを参照してください。

MPLAB IDE のインストール先ディレクトリ\ICD 3\Drivers\ddri.htm

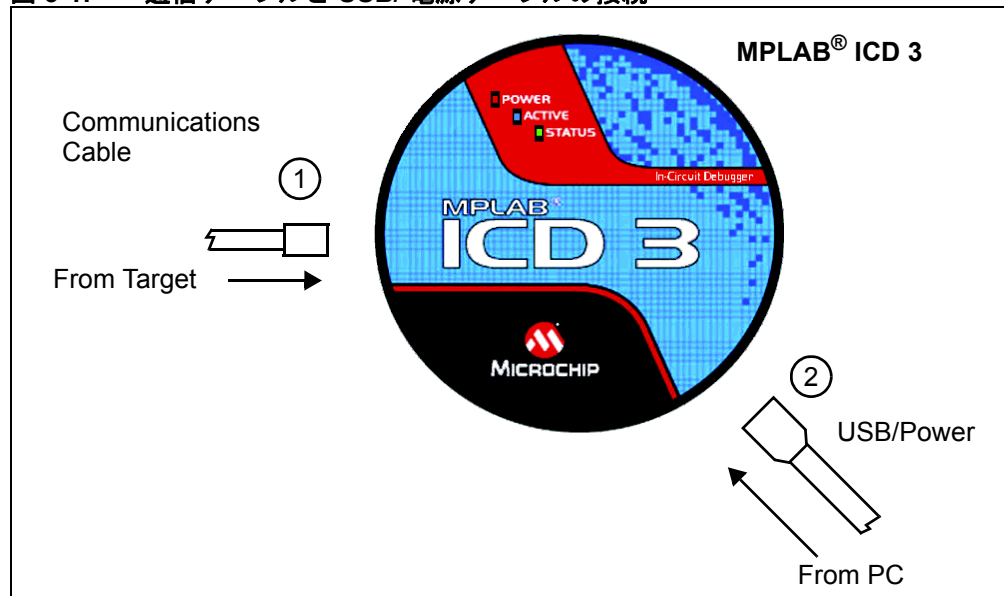
注： 新しい MPLAB ICD 3 を PC に接続した場合は、ドライバの再インストールが必要になります。

3.4 ターゲットの接続

MPLAB ICD 3 インサーキット デバッガには、ターゲットとの通信方法を選択するための接続回路が内蔵されています。詳細および接続図は、2.4 項「デバッガとターゲットの通信」を参照してください。

1. デバッガとターゲットを通信ケーブルで接続します。
2. まだ接続していない場合は、USB/電源ケーブルを接続します。

図 3-1: 通信ケーブルと USB/電源ケーブルの接続



3.5 ターゲット ボードのセットアップ

使用するターゲット デバイスの種類に合わせてターゲットをセットアップする必要があります。

3.5.1 通常のデバイスを使用する場合

通常のデバイスを使用する場合は、デバッガをターゲット ボードに直接接続します。MPLAB ICD 3 インサーキット デバッガでエミュレーションを行うには、ターゲット ボード上のデバイスにデバッグ回路が内蔵されている必要があります。必要なデバッグ回路がデバイスに内蔵されているかどうかは、各デバイスのデータシートで「Background Debugger Enable」コンフィギュレーション ビットがあるかどうかで確認できます。

注: ICD に対応する回路を内蔵したデバイスは、将来的にサポートされる予定です。

ターゲット ボードには、デバッガで使用する通信方法に対応したコネクタが必要です。接続の詳細は、2.4 項「デバッガとターゲットの通信」の「ICSP によるデバイスとの標準通信」を参照してください。

3.5.2 ICE デバイスを使用する場合

ICE デバイスを使用する場合は、ICE ヘッドボードが必要です。ヘッドボードには、特定のデバイスまたはデバイスファミリのエミュレーションに必要なハードウェアが含まれています。ICE ヘッドの詳細は、『*Header Board Specification*』(DS51292)を参照してください。

注： ICD デバイス (デバイス名 -ICD) を搭載した ICD ヘッドボードは、将来的にサポートされる予定です。

ICE ヘッドをターゲットボードに接続するには、トランジションソケットを使用します。共通のヘッドをさまざまな種類の表面実装パッケージに接続できるようにするため、トランジションソケットには数多くの種類が用意されています。トランジションソケットの詳細は、『*Transition Socket Specification*』(DS51194)を参照してください。

ヘッドボードのレイアウトは、ヘッドまたはプロセッサエクステンションパックごとに異なります。接続の詳細は、2.4 項「デバッガとターゲットの通信」の「ICSP によるデバイスとの標準通信」を参照してください。

3.5.3 ターゲットへの電源供給

MPLAB ICD 3 とターゲットへの電源の供給方法は 2 種類あります。

以下に、電源供給方法に関する要点を記します。

- USB 接続を使用する場合、MPLAB ICD 3 の電源は PC から供給することができますが、MPLAB ICD 3 から供給できる電流には制限があり、小型のターゲットボードに最大 100 mA (3V ~ 5V の VDD) しか供給できません。
- ターゲットで電源 VDD を自給すれば 2V ~ 5V の広い電圧範囲が得られるので、こちらの方法を推奨します。もう 1 つの利点として、プラグ & プレイによるターゲット検出機能が継承される点が挙げられます。つまり、MPLAB IDE がターゲットおよびデバイスを検出すると、その内容が [Output] ウィンドウに表示されます。

注： ターゲット電圧は、ICSP インターフェースのドライバに対してのみ電源を供給します。MPLAB ICD 3 には電源供給を行いません。MPLAB ICD 3 システムの電源はすべて USB ポートから供給されます。

まだ MPLAB ICD 3 とターゲットを接続していない場合は、適切なケーブルで接続してください (3.4 項「ターゲットの接続」参照)。次に、ターゲットの電源を投入します。MPLAB ICD 3 からターゲットに電源を供給する場合は、9.5.8 項「[Settings] ダイアログの [Power] タブ」の説明を参照してください。

3.6 MPLAB IDE のセットアップ

ハードウェアを接続して電源を投入したら、MPLAB IDE で MPLAB ICD 3 インサーキットデバッガを使用できるように準備を行います。

一部のデバイスでは、コンフィギュレーションビットで通信チャンネルを選択する必要があります (PGC1/EMUC1、PGD1/EMUD1 など)。ここで選択したピンが、デバイスに物理的に接続されているピンと同じであることを確認してください。

プロジェクトのセットアップなど、MPLAB ICD 3 で作業を始めるための準備方法は、第 4 章「全般的なセットアップ」を参照してください。

MPLAB ICD 3 を使用してデバイスのプログラミングとデバッグを行う手順は、第 5 章「チュートリアル」を参照してください。

MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

第 4 章 全般的なセットアップ

4.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガで作業を始めるまでの準備方法を説明します。

- MPLAB IDE ソフトウェアの起動
- プロジェクトの作成
- プロジェクトの表示
- プロジェクトのビルド
- コンフィギュレーション ビットの設定
- MPLAB ICD 3 インサーキット デバッガをデバッガまたはプログラマとして設定
- デバッガ / プログラマの制約

4.2 MPLAB IDE ソフトウェアの起動

MPLAB IDE ソフトウェアのインストールが完了したら (3.2 項「ソフトウェアのインストール」)、次のいずれかの方法で起動してください。

- [スタート] > [すべてのプログラム] > [Microchip] > [MPLAB IDE vx.xx] > [MPLAB IDE] の順にクリックする (「vx.xx」はバージョン番号)。
- デスクトップにある MPLAB IDE のアイコンをダブルクリックする。
- MPLAB IDE のインストール先ディレクトリの \core サブディレクトリにある mplab.exe を実行する。

MPLAB IDE ソフトウェアの使用法の詳細は次のドキュメントを参照してください。

- 『MPLAB IDE User's Guide』 (DS51519) — MPLAB IDE の使用法を詳しく解説したガイド。
- 『MPLAB IDE Quick Start Guide』 (DS51281) — 上記ユーザーズガイドの第 1 章と第 2 章を抜粋したもの。
- オンラインヘルプファイル — MPLAB IDE および MPLAB ICD 3 インサーキット デバッガに関する最新情報を記載。
- Readme ファイル — 各リリースの直前に発生した情報を Readme for MPLAB IDE.txt および Readme for MPLAB ICD 3 Debugger.txt に記載。これらのファイルは、どちらも MPLAB IDE のインストール先ディレクトリの Readmes サブディレクトリにあります。

4.3 プロジェクトの作成

新規プロジェクトを簡単に作成するには、[Project] > [Project Wizard] の順にクリックします。プロジェクトウィザードの指示に従って操作すると、新規プロジェクトを作成し、そのプロジェクトのビルドに使用する言語ツールを選択できます。ウィザードの手順に従っていくと、ソースファイル、ライブラリ、リンカスクリプトなどが、プロジェクトウィンドウの各種「ノード」に追加されます。ウィザードの使用の詳細は MPLAB IDE のマニュアルを参照してください。基本的な手順は次のとおりです。

- デバイスを選択する (PIC24FJ128GA010 など)
- 言語ツールスイートを選択する (Microchip C30 Toolsuite など)
- プロジェクト名を指定する
- アプリケーションファイルを追加する (program.c, support.s, counter.asm など)

注: プロジェクトにカスタムリンカスクリプトがない場合は、プロジェクトマネージャによって適切なリンカスクリプトが自動的に選択されます。

4.4 プロジェクトの表示

プロジェクトウィザードを使ってプロジェクトを作成したら、プロジェクトおよび関連ファイルがプロジェクトウィンドウに表示されます。プロジェクトにファイルを追加するには、プロジェクトウィンドウを使用します。プロジェクトウィンドウでツリーの任意の行を右クリックするとポップアップメニューが表示され、ファイルの追加や削除を行えます。

プロジェクトウィンドウの使用の詳細は MPLAB IDE のマニュアルを参照してください。

4.5 プロジェクトのビルド

プロジェクトを作成したら、アプリケーションをビルドする必要があります。ビルドを行うと、アプリケーションのオブジェクト (Hex) コードが作成されます。このオブジェクトコードを、MPLAB ICD 3 インサーキットデバッガを使ってターゲットに書き込みます。

ビルドのオプションを設定するには、[Project] > [Build Options] > [Project] の順にクリックします。

注: MPLAB ICD 3 をデバッガとして使用する場合は、プロジェクトマネージャツールバー ([View] > [Toolbars] > [Project Manager]) のドロップダウンリストで「Debug」を選択し、プログラマとして使用する場合は「Release」を選択します。

設定が完了したら、[Project] > [Build All] の順にクリックしてプロジェクトをビルドします。

4.6 コンフィギュレーション ビットの設定

デバイスのコンフィギュレーション ビットはコード中でも設定できますが、MPLAB IDE の [Configuration Bits] ウィンドウで設定することもできます。[[Configure](#)] > [[Configuration Bits](#)] の順にクリックします。[Settings] 欄の文字列をクリックすると、設定を変更できます。

ここでは、いくつかのコンフィギュレーション ビットについて簡単に説明します。

- **Watchdog Timer Enable** — ほとんどのデバイスでウォッチドッグ タイマはデフォルトで有効に設定されています。通常は、このビットを無効にしておくといでしょう。
- **Comm Channel Select** — 一部のデバイスでは、デバイスの通信チャネルを選択する必要があります (PGC1/EMUC1、PGD1/EMUD1 など)。ここで選択したピンが、デバイスに物理的に接続されているピンと同じであることを確認してください。
- **Oscillator** — ターゲットのオシレータと同じ設定を選択してください。

4.7 MPLAB ICD 3 インサーキット デバッガをデバッガまたはプログラマとして設定

MPLAB ICD 3 インサーキット デバッガをデバッガとして選択するには、[[Debugger](#)] > [[Select Tool](#)] > [[MPLAB ICD 3](#)] の順にクリックします。デバッガを選択すると、[Debugger] メニューと MPLAB IDE のツールバーには選択したデバッガの機能が表示されます。また、[Output] ウィンドウが開き、[MPLAB ICD 3] タブに MPLAB ICD 3 のステータスおよび通信に関するメッセージが表示されます。詳細は、9.2 項「デバッガに関する機能」および 9.3 項「デバッグに関するダイアログ/ウィンドウ」を参照してください。

MPLAB ICD 3 インサーキット デバッガをプログラマとして選択するには、[[Programmer](#)] > [[Select Programmer](#)] > [[MPLAB ICD 3](#)] の順にクリックします。プログラマを選択すると、[Programmer] メニューと MPLAB IDE のツールバーには選択したプログラマの機能が表示されます。また、[Output] ウィンドウが開き、[MPLAB ICD 3] タブに ICE のステータスおよび通信に関するメッセージが表示されます。詳細は、9.4 項「プログラミングに関する機能」を参照してください。

オプションの設定は、[[Debugger](#)] > [[Settings](#)] または [[Programmer](#)] > [[Settings](#)] の順にクリックして [Settings] ダイアログ (9.5 項「[Settings] ダイアログ」) で行います。

エラーが発生した場合は、以下の項を参照してください。

- 第 7 章「エラー メッセージ」
- 第 6 章「よく寄せられる質問 (FAQ)」
- 10.7 項「ICD 3 テスト インターフェース ボード」

4.8 デバッガ/プログラマの制約

各デバイスにおけるデバッガの制約については、MPLAB IDE の MPLAB ICD 3 オンライン ヘルプ ファイルを参照してください ([[Help](#)] > [[Topics](#)] > [[MPLAB ICD 3](#)] の順にクリックして、[OK] をクリック)。

MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

第 5 章 チュートリアル

5.1 はじめに

このチュートリアルでは、サンプルプログラムの `counter.c` と `timer.c` を使用した簡単なプロジェクトの開発手順を説明します。ここで紹介するのは、PIC24FJ128GA010 と Explorer 16 デモ ボード (DM240001) を使用した実装です。`counter.c` は、簡単なカウンタ プログラムです。このカウンタは、タイマ 1 (`timer.c`) の間隔でインクリメントし、その内容をポート A 経由でデモ ボードの LED に表示します。

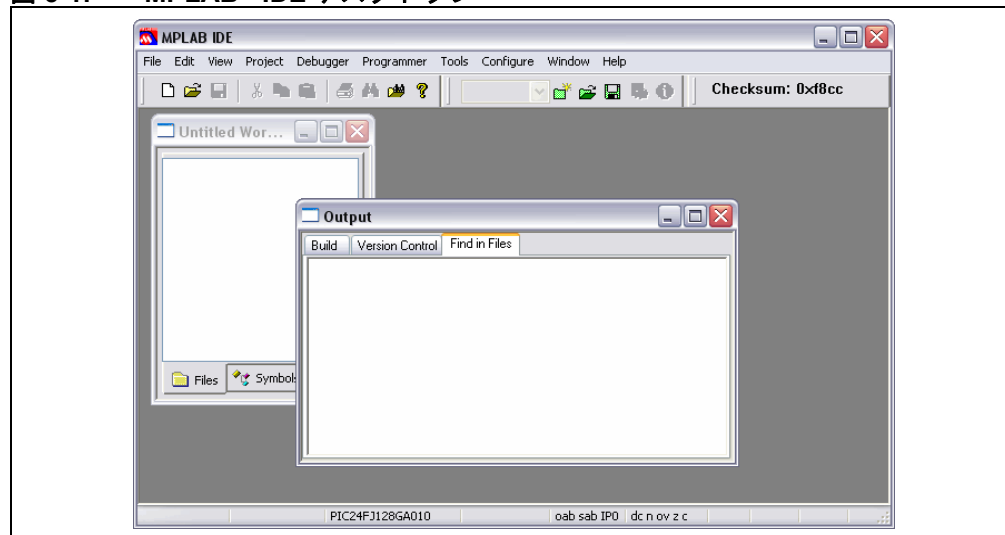
この章の内容は次のとおりです。

- 環境のセットアップとデバイスの選択
- アプリケーション コードの作成
- プロジェクト ウィザードの実行
- プロジェクトの表示
- デバッグ オプションの確認
- デモ ボードのセットアップ
- デバッグ用プログラム コードのロード
- デバッグ コードの実行
- ブレークポイントを使用したコードのデバッグ
- アプリケーションのプログラミング

5.2 環境のセットアップとデバイスの選択

チュートリアルを始める前に、第3章「インストール」の手順に従って MPLAB IDE ソフトウェアと MPLAB ICD 3 システム ハードウェアをセットアップしてください。MPLAB IDE のアイコンをダブルクリックするとアプリケーションが起動します。アプリケーションが起動すると、MPLAB IDE デスクトップが表示されます。

図 5-1: MPLAB[®] IDE デスクトップ



デバイスの選択

このチュートリアルで使用するデバイスを選択するには：

1. **[Configure] > [Select Device]** の順にクリックします。
2. **[Device Selection]** ダイアログの **[Device]** リストボックスで「PIC24FJ128GA010」を選択します。**[Microchip Tool Support]** の **[Programmers]** セクションと **[Debuggers]** セクションで「MPLAB ICD 3」に緑のアイコンが表示されていることを確認してください。
3. **[OK]** をクリックします。

5.3 アプリケーションコードの作成

このチュートリアルでは2つのCプログラム (counter.c と timer.c) を使用します。各コードは次ページ以降に記載しています。

1. Windows[®] エクスプローラを使用して、プロジェクト用のフォルダとサブディレクトリを作成します (C:\Projects\ICD3Tut など)。
2. **[File] > [New]** の順にクリックして、エディタ ウィンドウを開きます。このウィンドウに counter.c のコードを入力し、プロジェクト用フォルダのサブディレクトリに保存します。
3. **[File] > [New]** の順にクリックして、エディタ ウィンドウをもう1つ開きます。このウィンドウに timer.c のコードを入力し、プロジェクト用フォルダのサブディレクトリに保存します。

counter.c

```
/*
 * MPLAB ICD 3 In-Circuit Debugger Tutorial
 * Counting program
 *
 *
 * Demo Board:      Explorer 16
 * Processor:       PIC24FJ128GA010
 * Compiler:        MPLAB C30
 * Linker:          MPLAB LINK30
 * Company:         Microchip Technology Incorporated
 *
 */

#include "p24FJ128GA010.h"

// Set up configuration bits
_CONFIG1( JTAGEN_OFF & GCP_OFF & GWRP_OFF & COE_OFF & FWDTEN_OFF & ICS_PGx2)
_CONFIG2( FCKSM_CSDCMD & OSCIOFNC_ON & POSCMOD_HS & FNOSC_PRI )

void TimerInit(void);
unsigned char TimerIsOverflowEvent(void);

// Set up user-defined variables
#define INIT_COUNT 0
unsigned int counter;

int main(void)
{
    // Set up PortA IOs as digital output
    AD1PCFG = 0xffff;
    TRISA = 0x0000;

    // Set up Timer1
    TimerInit();

    // Initialize variables
    counter = INIT_COUNT;

    while (1) {
        // Wait for Timer1 overflow
        if (TimerIsOverflowEvent()){
            counter++; //increment counter
            PORTA = counter; //display on port LEDs
        } // End of if...

        } // End of while loop...
    } // End of main()...
```

MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

timer.c

```
/*
 * MPLAB ICD 3 In-Circuit Debugger Tutorial
 * Timer program
 *
 * Demo Board:      Explorer 16
 * Processor:       PIC24FJ128GA010
 * Compiler:        MPLAB C30
 * Linker:          MPLAB LINK30
 * Company:         Microchip Technology Incorporated
 *
 */
#include "p24FJ128GA010.h"

//declare functions
extern void TimerInit(void);
extern unsigned char TimerIsOverflowEvent(void);

/*
 * Function:         TimerInit
 *
 * PreCondition:    None.
 *
 * Input:           None.
 *
 * Output:          None.
 *
 * Overview:        Initializes Timer1 for use.
 *
 */
void TimerInit(void)
{
    PR1 = 0xFFFF;

    IPC0bits.T1IP = 5;
    T1CON = 0b1000000000010000;
    IFS0bits.T1IF = 0;
}

/*
 * Function:         TimerIsOverflowEvent
 *
 * PreCondition:    None.
 *
 * Input:           None.
 *
 * Output:          Status.
 *
 * Overview:        Checks for an overflow event, returns TRUE if
 *                  an overflow occurred.
 *
 * Note:            This function should be checked at least twice
 *                  per overflow period.
 *
 */
unsigned char TimerIsOverflowEvent(void)
{
    if (IFS0bits.T1IF)
    {

```

```

    IFS0bits.T1IF = 0;
    TMR1 = 0;
    return(1);
}
return(0);
}

/*****
 * EOF
 *****/

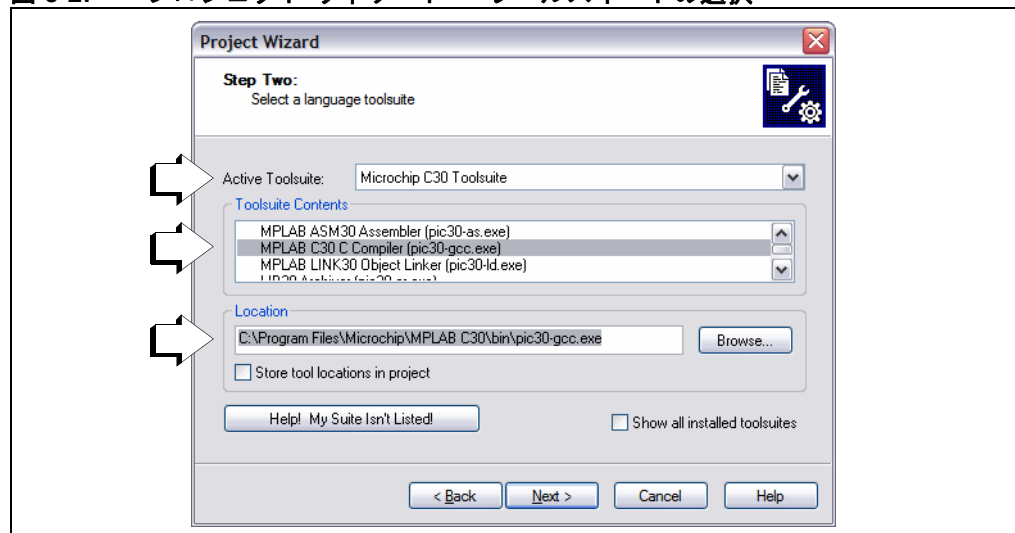
```

5.4 プロジェクト ウィザードの実行

このプロジェクトでは MPLAB C30 C コンパイラを使用します。完全版のコンパイラを購入するか、マイクロチップ社のウェブサイトで無償の Student Edition をダウンロードしてください。

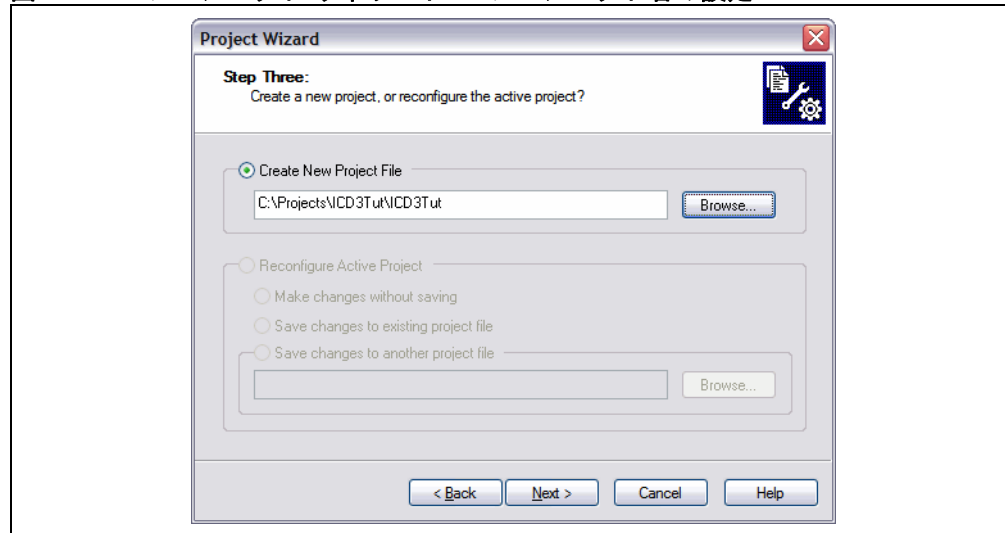
1. プロジェクトのセットアップを行うには、**[Project] > [Project Wizard]** の順にクリックします。ウィザードの開始画面が表示されます。
2. 次の画面に進みます。PIC24FJ128GA010 を選択してください。
3. 次の画面に進み、言語ツールのセットアップを行います。**[Active Toolsuite]** プルダウンメニューで「Microchip C30 Toolsuite」を選択します。各ツールに適切な実行ファイルが指定されていることを確認してください。これら実行ファイルは、デフォルトでは C:\Program Files\Microchip\MPLAB C30\bin にあります。MPLAB C30 の実行ファイルは pic30-gcc.exe、MPLAB LINK30 の実行ファイルは pic30-ld.exe です。

図 5-2: プロジェクトウィザードー ツールスイートの選択



4. 次の画面に進み、プロジェクトの名前と保存場所を指定します。**[Browse]** をクリックして保存場所を選択することもできます。

図 5-3: プロジェクトウィザードープロジェクト名の設定

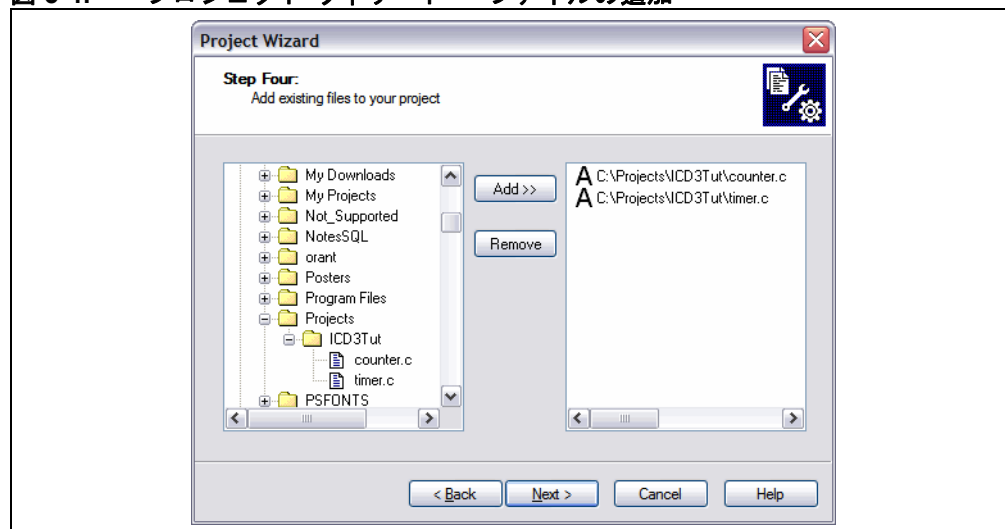


5. 次の画面に進み、プロジェクトファイルを追加します。ファイルは、必要に応じて後から追加することもできます。

このチュートリアルでは、プロジェクト用ディレクトリに2つのファイルが表示されます。counter.c をクリックして選択し、[ADD>>] をクリックして右の枠に追加します。次に timer.c をクリックして選択し、[ADD>>] をクリックして右の列に追加します。

ファイル名の先頭にある「A」はそのままにしておいてください。ここに表示されるアルファベットの意味は、ウィザードの [ヘルプ] ボタンをクリックしてください。

図 5-4: プロジェクトウィザードーファイルの追加

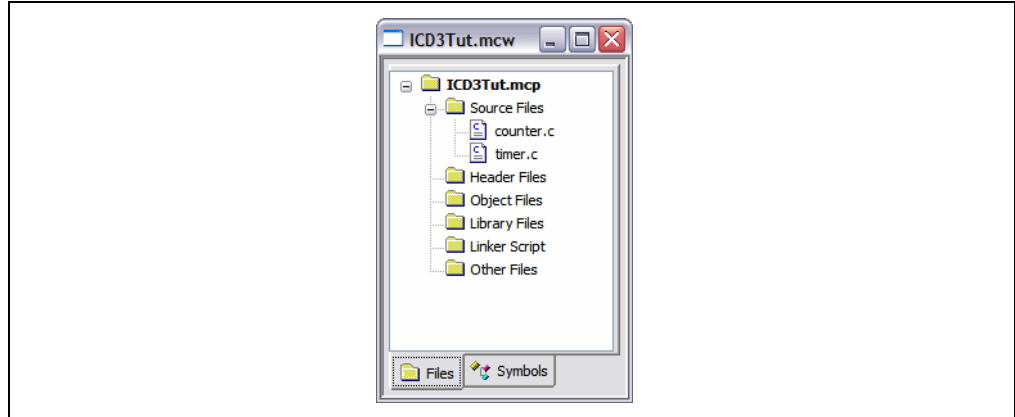


6. 次の画面に進むと、サマリが表示されます。内容を訂正したい場合は、[< 戻る] をクリックして前の画面に戻ってください。内容が正しければ、[完了] をクリックします。

5.5 プロジェクトの表示

ウィザードを終了すると、MPLAB IDE デスクトップが再び表示されます。プロジェクトウィンドウが表示されていない場合は、[\[View\]/\[Project\]](#) の順にクリックしてプロジェクトウィンドウを表示してください。

図 5-5: プロジェクトウィンドウ



プロジェクトにファイルを追加するには、プロジェクトウィンドウを使用します。プロジェクトウィンドウでツリーの任意の行を右クリックするとポップアップメニューが表示され、ファイルの追加や削除を行えます。

注: プロジェクトではヘッダファイル p24FJ128GA010.h およびリンクスクリプトファイルも使用しますが、これらは手動でプロジェクトに追加しなくても MPLAB IDE によって自動的に選択されます。

5.6 デバッグ オプションの確認

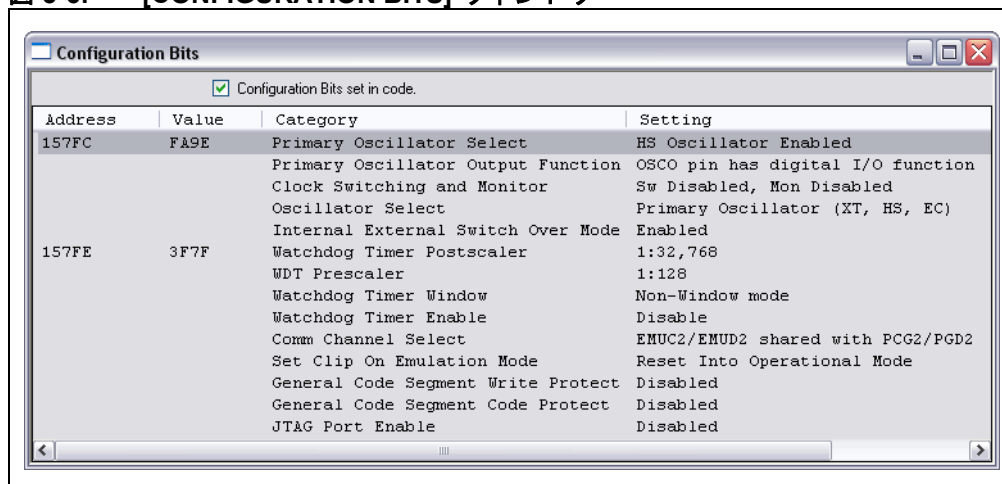
コードのデバッグを始める前に、いくつかの項目のデフォルト設定を確認しておきます。実際のプロジェクトでは、このチュートリアルとは異なる設定となる場合があります。

5.6.1 コンフィギュレーション ビット

このチュートリアルでは、counter.c のコードで `_CONFIG1` および `_CONFIG2` ディレクティブを使用してデバイスのコンフィギュレーション ビットを設定しています。PIC24FJ128GA010 のコンフィギュレーション レジスタ ビットの動作の詳細は『PIC24FJ128GA Family Data Sheet』(DS39747) を参照してください。

[\[Configure>Configuration Bits\]](#) をクリックし、[\[Configuration Bits set in code\]](#) のチェックを解除すると、Configuration Bits ウィンドウでコンフィギュレーション ビットを設定することもできます。このチュートリアルでは、コンフィギュレーション ビットの値は変更しないでください。

図 5-6: [\[CONFIGURATION BITS\]](#) ウィンドウ



5.6.2 MPLAB ICD 3 インサーキット デバッガをデバッガとして選択

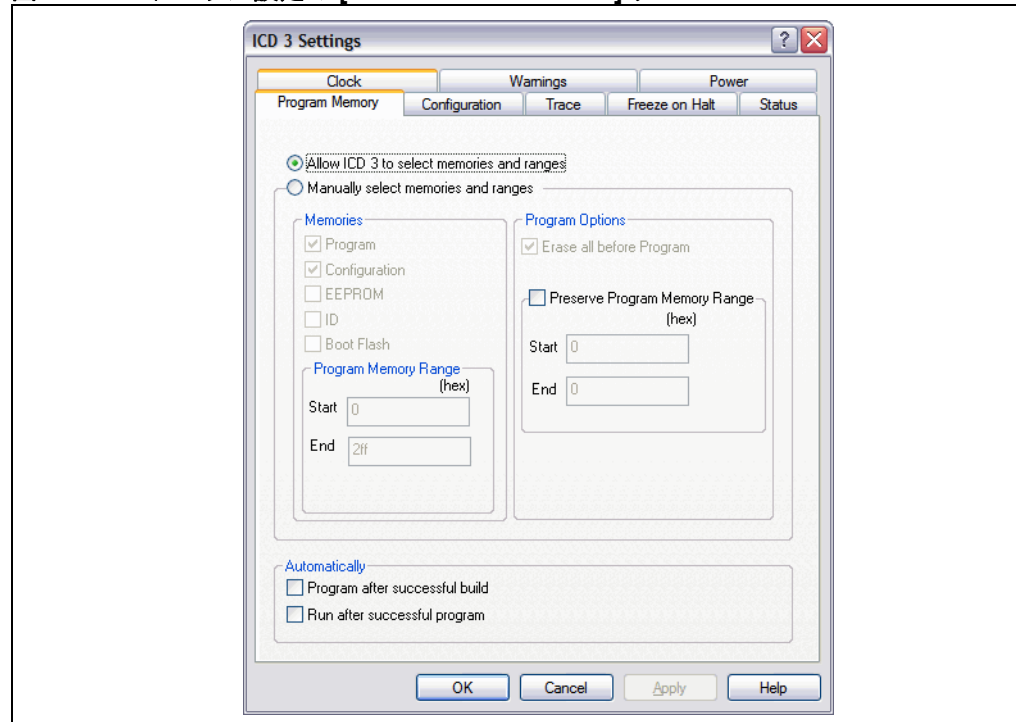
デバッガに MPLAB ICD 3 インサーキット デバッガを選択するには、[\[Debugger\]](#) > [\[Select Tool\]](#) > [\[ICD 3\]](#) の順にクリックします。次に、以下の操作を行います。

1. [\[Output\]](#) ウィンドウが開き、接続情報が表示されます。MPLAB IDE のバージョンや選択したデバイスの種類によっては、ファームウェアの更新を促すメッセージボックスが表示される場合があります。その場合はメッセージボックスで **[OK]** をクリックして、新しいファームウェアをインストールします。MPLAB ICD 3 のファームウェアはデバイス ファミリごとに種類が異なるため、デバイスを変更した場合にもこのメッセージボックスが表示されることがあります。
2. [\[Output\]](#) ウィンドウにはファームウェアの更新に関する情報が表示され、MPLAB ICD 3 をターゲットに接続すると接続状況が表示されます。
3. [\[Debugger\]](#) メニューには、利用可能なデバッガのデバッグ オプションが表示されます。
4. デバッグ ツールバーが表示されます。ボタンの上にマウス ポインタを置くと、機能の説明がポップアップ表示されます。

5.6.3 プログラミングのオプション

プログラミングのオプションを設定するには、**[Debugger]** > **[Settings]** の順にクリックして、**[Program Memory]** タブをクリックします。

図 5-7: デバッガ設定の **[PROGRAM MEMORY]** タブ



ここでは、プログラミングの範囲をデバッガで自動選択する（推奨）か、ユーザーが範囲を設定するかを選択します。

- **[Memories]** セクションの **[Program]** にチェックを入れ、**[EEPROM]** と **[ID]** のチェックを外します。デバッガとして MPLAB ICD 3 インサーキット デバッガを使用する場合は、コンフィギュレーション ビットは常にプログラミングされるので、**[Configuration]** にチェックが入った状態で淡色表示されます。
- PIC24FJ デバイスの場合、チップにプログラミングを行う前に毎回すべてのメモリが消去されます。このため、**[Program Options]** セクションの **[Erase all before Program]** も設定を変更することはできません。
- **[Program Memory]** のアドレス (**[Start]** と **[End]**) で、読み出し、書き込み、ベリファイを行うプログラムメモリの範囲を指定します。

なお、コードのデバッグ時には、編集、再ビルド、再プログラム、実行の一連の操作を頻繁に繰り返すことになります。この操作を自動化するには、**[Program after successful build]** と **[Run after successful program]** のチェックボックスを選択します。今は、これらのチェックボックスは選択しないでください。

5.7 HEX ファイルの作成

デバッグ用の Hex ファイルを作成するには：

- プロジェクト ツールバーで、**[Build Configuration]** ドロップダウン リストから「Debug」を選択します。
- **[Project]** > **[Build All]** の順にクリックするか、またはプロジェクト ウィンドウでプロジェクト名を右クリックし、ポップアップメニューから **[Build All]** をクリックします。

MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

プロジェクトがビルドされ (図 5-8)、プロジェクトと同じ名前の .hex ファイルが生成されます (図 5-9)。この Hex ファイルをターゲット デバイスにプログラミングします。

注: ビルド オプション (*[Project] > [Build Options] > [Project]* の **[MPLAB C30]** タブおよび **[MPLAB LINK30]** タブ) の設定によっては、**[Output]** ウィンドウの表示が図 5-8 と異なる場合があります。

図 5-8: **[OUTPUT]** ウィンドウ

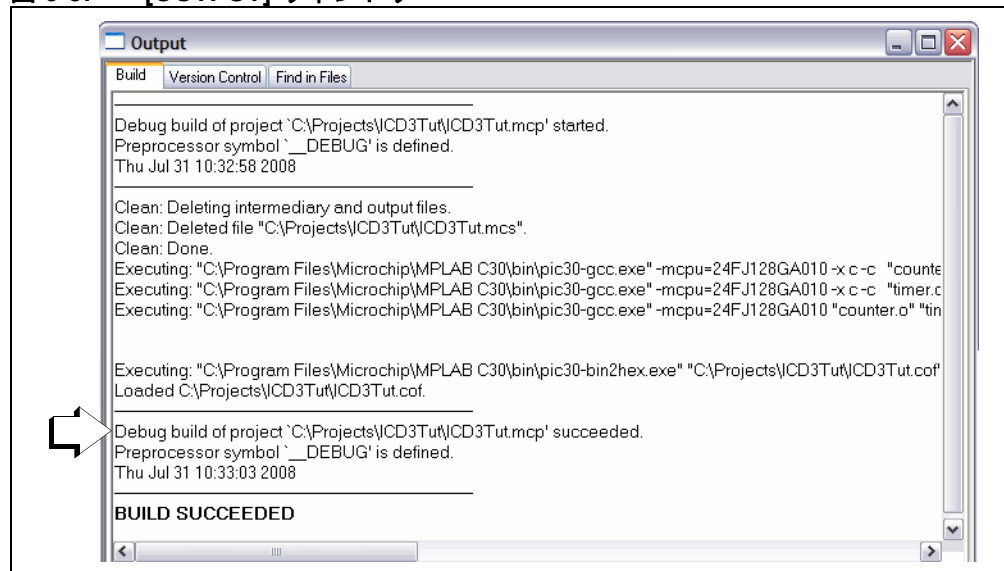
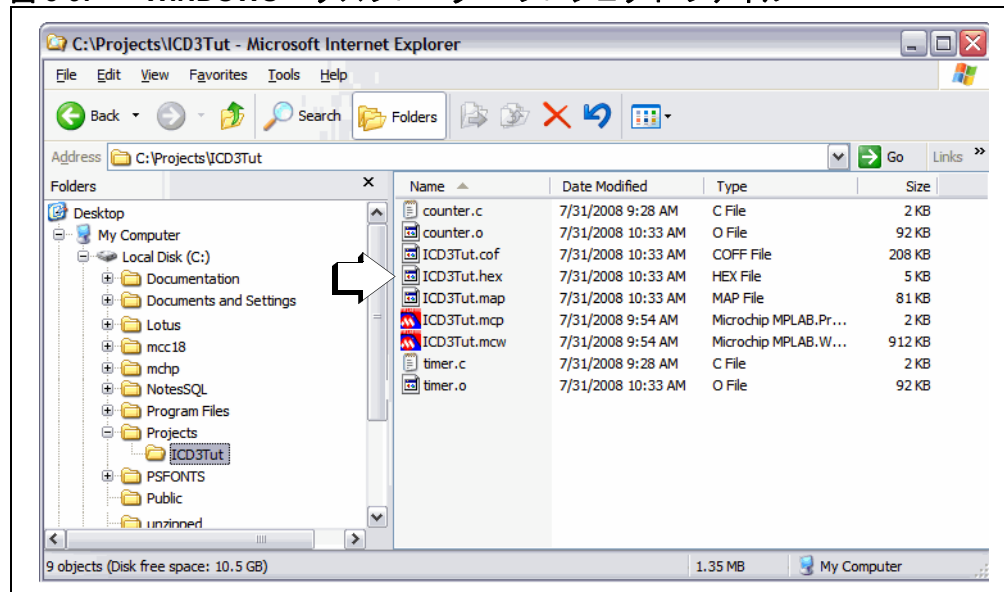


図 5-9: **WINDOWS** エクスプローラー - プロジェクト ファイル



5.8 デモボードのセットアップ

デバッグを始める前に、Explorer 16 デモボードが正しくセットアップされていることを確認してください。詳細は、『*Explorer 16 Development Board User's Guide*』(DS51589)を参照してください。

このチュートリアルでは、次のとおり設定します。

- PIC24FJ128GA010 PIM (プラグインモジュール) をボードに接続します。
- S2: 「PIM」を選択します。「PIC」はデバイスがボードに実装されている場合に選択します。
- J7: 「PIC24」を選択します。これにより、デバッガはオンボードの PIC18LF4550 USB デバイスではなく PIC24FJ128GA010 と直接通信を行います。
- JP2: ジャンパ2 を接続すると、LED が有効になります。
- D1 点灯: ボードに電源が供給されていることを示します。

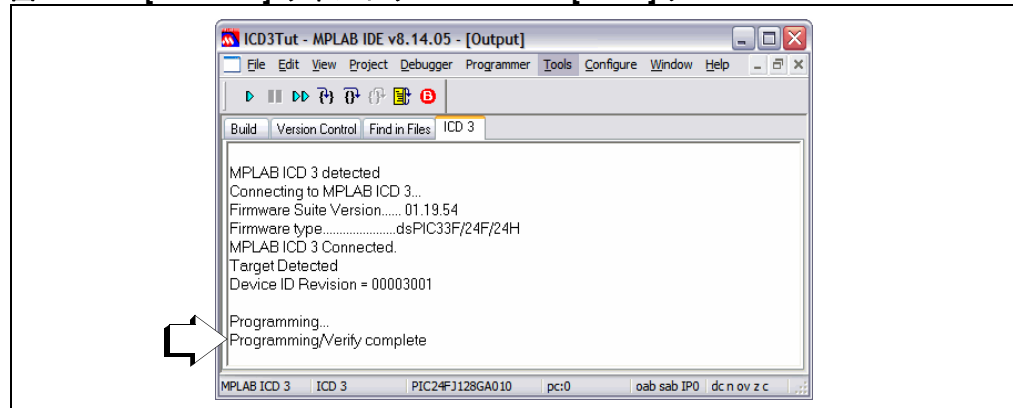
5.9 デバッグ用プログラムコードのロード

[Debugger] > **[Program]** の順にクリックして RITut.hex を Explorer 16 デモボード上の PIC24FJ128GA010 にプログラミングします。

注: MPLAB ICD 3 のデバッグ機能を実行するデバッグ実行コードは、プログラムメモリの上位アドレスに自動的に書き込まれます。MPLAB ICD 3 インサーキットデバッガのイン-サーキットデバッグ機能を利用するには、デバッグコードをターゲットデバイスに書き込む必要があります。

書き込み中、[Output] ウィンドウの **[ICD 3]** タブに現在の処理状況が表示されます。書き込みが完了すると、図 5-10 のような画面が表示されます。

図 5-10: [OUTPUT] ウィンドウ — MPLAB® [ICD 3] タブ










注: デバイスへの書き込みやデバッガとの通信が正常に行えない場合は、Explorer 16 ボードを取り外し、テストインターフェースボード (10.7 項「ICD 3 テストインターフェースボード」) を使用して通信が正しく行えるかどうか確認してください。詳細は、第 6 章「よく寄せられる質問 (FAQ)」を参照してください。

5.10 デバッグコードの実行

MPLAB ICD 3 インサーキット デバッガは、リアルタイム モードまたはステップ モードのいずれかで実行できます。

- デバイスを MPLAB IDE の Run モードに設定すると、リアルタイム モードで実行されます。
- ステップ モード実行は、いったんプロセッサを停止してから実行します。

次に示すツールバーのボタンで、よく使うデバッグ コマンドを手軽に使用できます。

Debugger Menu	Run	Halt	Animate	Step Into	Step Over	Step Out	Reset
Toolbar Buttons							

リアルタイム モードでデバッグを開始するには：

1. ソース ファイルの counter.c と timer.c を開きます (プロジェクト ウィンドウでファイル名をダブルクリックするか、**[File] > [Open]** の順にクリックして開きます)。
2. **[Debugger] > [Run]** の順にクリック (またはツールバーの **[Run]** ボタンをクリック) します。
3. LED の状態を観察します。2 進数でインクリメントしていることを確認します。
4. **[Debugger] > [Halt]** の順にクリック (またはツールバーの **[Halt]** ボタンをクリック) してプログラムの実行を停止します。
5. デバッガが停止したら、先ほど開いたソース コード ウィンドウの 1 つが前面に表示され、プログラムの停止位置が緑の矢印で表示されます。

ステップ モードを使用するには：

1. **[Debugger] > [Step Into]** の順にクリック (またはツールバーの **[Step Into]** ボタンをクリック) すると、デバッガは 1 命令を実行して停止します。これに合わせて、コード行の先頭にある緑の矢印も移動します。
2. 必要に応じてこの操作を繰り返します。

ステップ機能の **[Step Over]** と **[Step Out]** は関数呼び出しで使用します。詳細は、MPLAB IDE のマニュアルを参照してください。

5.11 ブレークポイントを使用したコードのデバッグ

このチュートリアルサンプル コードはデバッグ済みで、正常に動作します。しかし、ここではこのコードを使用して MPLAB ICD 3 インサーキット デバッガのデバッグ機能を実際に見てみることにします。最初に紹介するデバッグ機能は、ブレークポイントです。ブレークポイントとは、コード内の指定した行で実行を停止する機能です。

- ソフトウェア ブレークポイントの設定

5.11.1 ブレークポイントの種類を選択

このチュートリアルで使用するデバイスでは、ハードウェア ブレークポイントまたはソフトウェア ブレークポイントのいずれかを選択できます。

ブレークポイントのオプションを設定するには、**[Debugger] > [Settings]** の順にクリックして、**[Configuration]** タブをクリックします。ブレークポイントの種類は、アプリケーションのニーズに応じて選択してください。このチュートリアルでは、まずデフォルトのブレークポイント (ハードウェア ブレークポイント) から使用していきます。

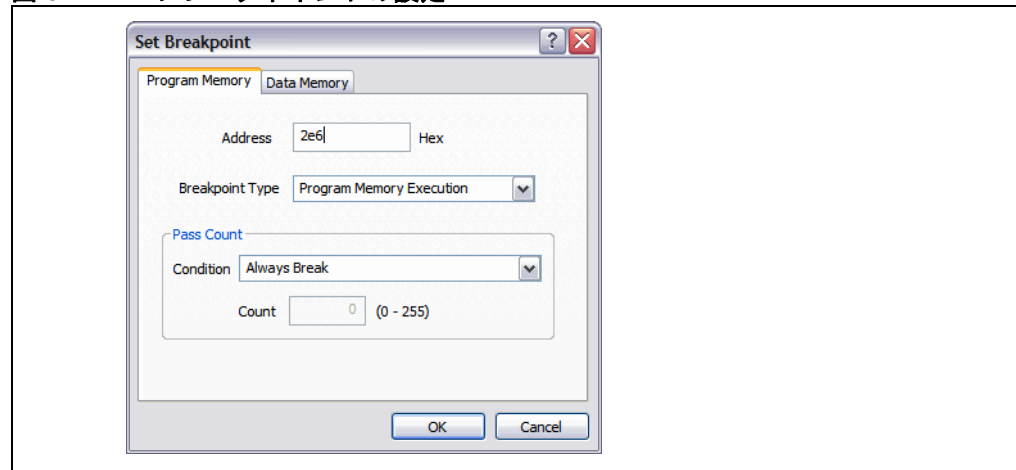
5.11.2 単一のハードウェア ブレークポイントの設定

ブレークポイントを1つずつ設定するには:

1. *[Debugger]* > *[Reset]* > *[Processor Reset]* の順にクリック (またはツールバーの **[Reset]** ボタンをクリック) して、サンプルプログラムをリセットします。
2. counter.c のコードで、次の行を選択するか、またはこの行にカーソルを置きます。

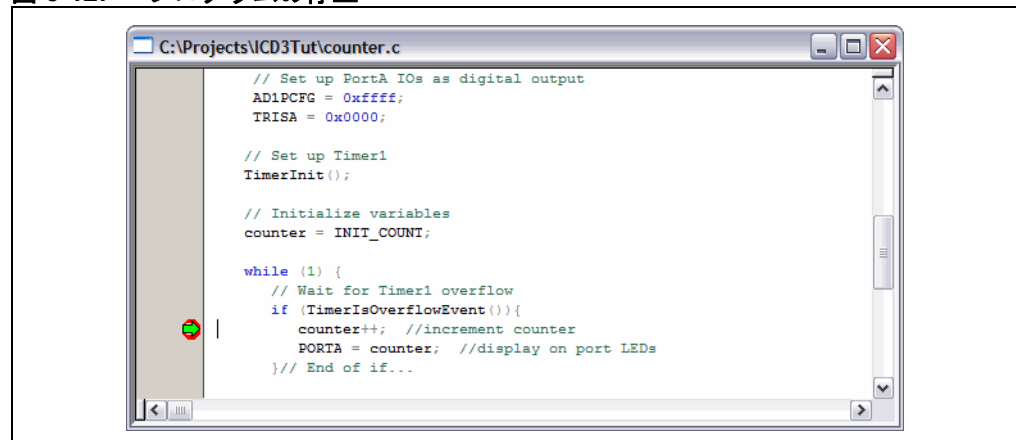
```
counter++; //increment counter
```
3. この行をダブルクリックするか、または右クリックしてショートカットメニューの *[Set Breakpoint]* を選択します。この行にブレークポイント (赤の「B」アイコン) が設定されます (図 5-11)。

図 5-11: ブレークポイントの設定



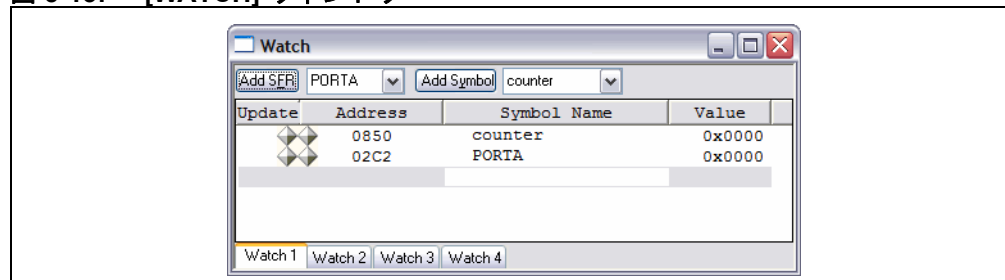
4. *[Debugger]* > *[Run]* の順にクリック (またはツールバーの **[Run]** ボタンをクリック) すると、プログラムは再びリアルタイム モードで実行されます。プログラムはブレークポイントが設定された行で停止し、ブレークポイントのアイコンの上に緑の矢印が表示されます。

図 5-12: プログラムの停止



- 新規 [Watch] ウィンドウを開き、プログラムの実行時に変数 counter の値がどのように変化するかを観察します。[View] > [Watch] の順にクリックします。[Watch] ウィンドウが開き、[Watch 1] タブが表示されます。[Add Symbol] ボタンの右のリストから「counter」を選択してボタンをクリックします。counter が [Watch] ウィンドウに追加されます。[Add SFR] ボタンの右のリストから「PORTA」を選択してボタンをクリックします。PORTA が [Watch] ウィンドウに追加されます。選択したシンボルが [Watch] ウィンドウに表示された様子を図 5-13 に示します。

図 5-13: [WATCH] ウィンドウ



- [Debugger] > [Run] の順にクリック (またはツールバーの [Run] ボタンをクリック) して、プログラムを再度実行します。プログラムがブレークポイントで停止すると、2つの変数の値が1つインクリメントしたことが分かります。
- プログラムを実行するたびに変数の値が増えることを確認してください。以上を確認できたら、[Debugger] > [Reset] > [Processor Reset] の順にクリック (またはツールバーの [Reset] ボタンをクリック) してプロセッサをリセットします。

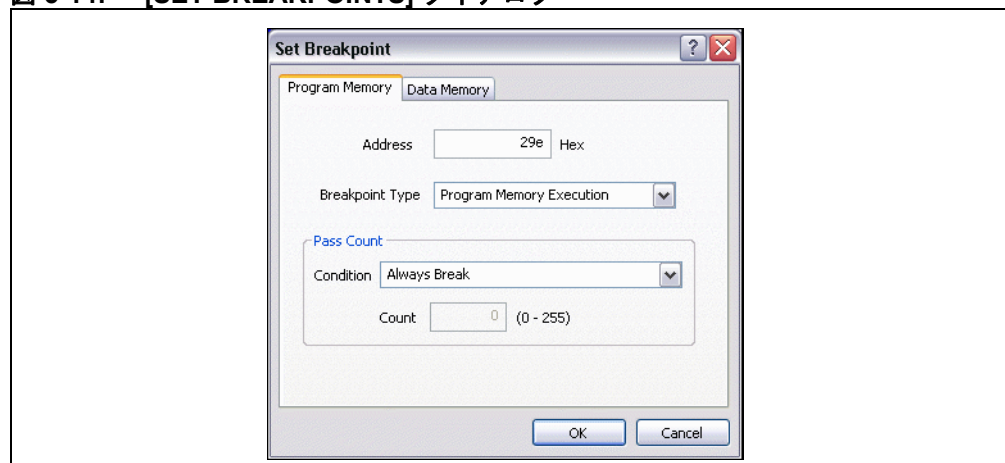
5.11.3 複数のハードウェア ブレークポイントの設定

複数のブレークポイントを設定するには、前の項で説明した方法でブレークポイントを1つずつ設定するか、または [Breakpoints] ダイアログ (9.3.1 項「[Breakpoints] ダイアログ」) を使用します。[Breakpoints] ダイアログでは、ブレークポイントどうしの連携を制御することもできます。

注: デバイスで設定可能なブレークポイント数の上限を超えると、MPLAB IDE で警告が表示されます。

- [Debugger] > [Breakpoints] の順にクリックして [Breakpoints] ダイアログを開きます。このダイアログには、前の項で設定したブレークポイントが表示されます。他のブレークポイントを追加するには、[Add Breakpoint] ボタンをクリックします。
- [Set Breakpoint] ダイアログの [Program Memory] タブで、[Address] 欄に「2e6」(Hex) と入力し、[OK] をクリックします。

図 5-14: [SET BREAKPOINTS] ダイアログ

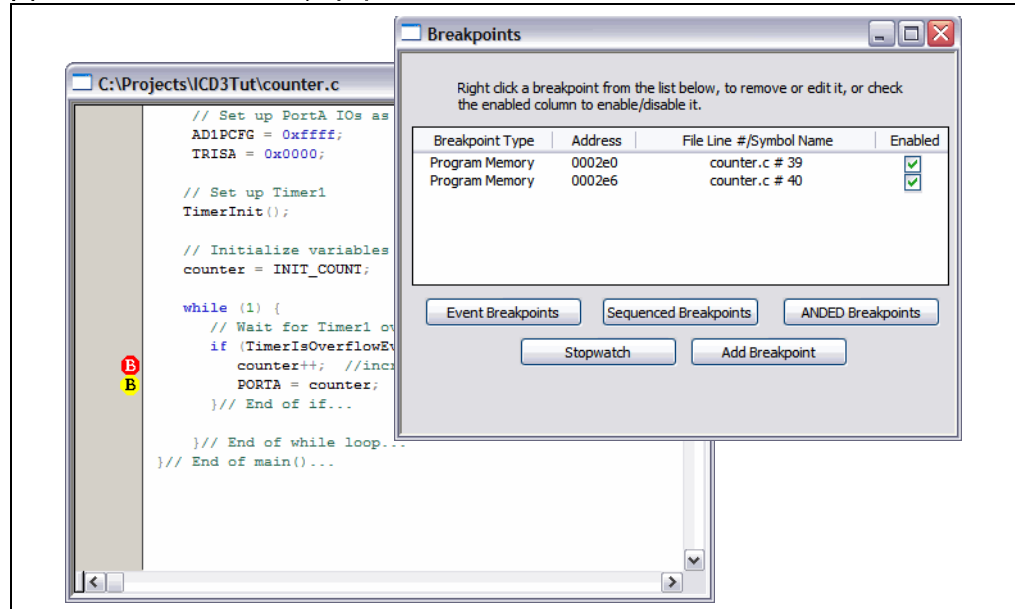


[Breakpoints] ダイアログのブレークポイント一覧の一番下に新しいブレークポイントが追加され、次に示すコード行の先頭にブレークポイントのアイコンが表示されます。

```
PORTA = counter; //display on port LEDs
```

このブレークポイントはアドレスで設定しているので、黄色のアイコンで表示されます。

図 5-15: 2つのブレークポイント



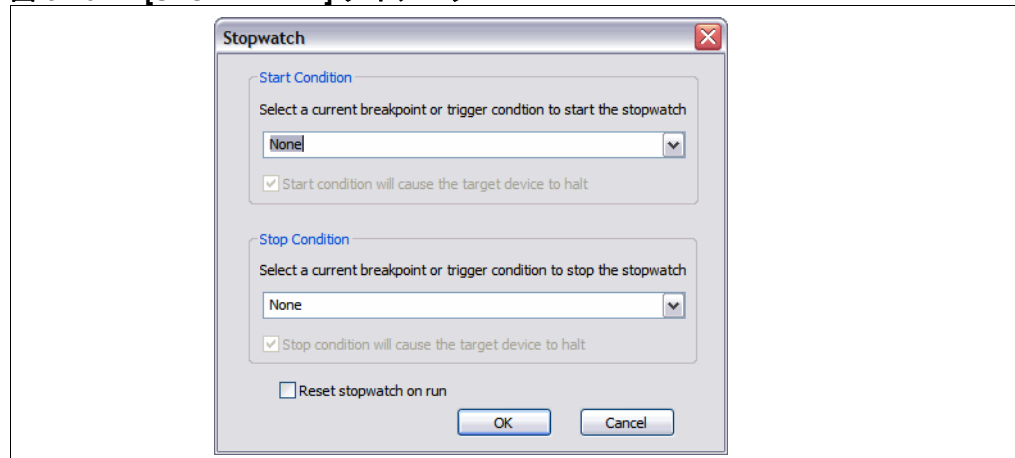
- プログラムを実行すると、最初のブレークポイントで停止します。[Watch] ウィンドウの値は変化しません。もう一度プログラムを実行すると、2番目のブレークポイントで停止します。(ブレークポイントを越えていくつかの命令が実行される場合もあります。)今度は、[Watch] ウィンドウの値が変化します。

5.11.4 ブレークポイントとストップウォッチの併用

2つのブレークポイント間の時間を計測するには、ストップウォッチを使用します。

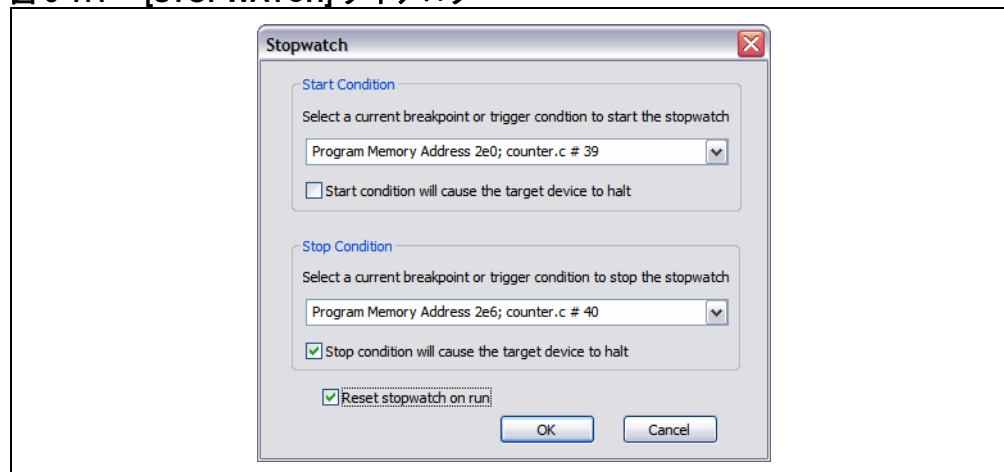
- [Breakpoints] ダイアログで [Stopwatch] をクリックすると、[Stopwatch] ダイアログが開きます。

図 5-16: [STOPWATCH] ダイアログ



2. [Start Condition] セクションのドロップダウン リストから最初のブレークポイントを選択します。[Start condition will cause the target device to halt] のチェックを外します。
3. [Stop Condition] セクションのドロップダウン リストから 2 番目のブレークポイントを選択します。[Stop condition will cause the target device to halt] にチェックを入れます。
4. [Reset stopwatch on run] にチェックを入れます。
5. [OK] をクリックします。

図 5-17: [STOPWATCH] ダイアログ



6. プログラムを実行し、停止するまで待ちます。[Output] ウィンドウの [ICD 3] タブに、2 つの命令間のサイクル数が次のように表示されます。
Stopwatch cycle count = 4(decimal)
7. コードから 2 つのブレークポイントを削除します。削除するには、[Breakpoints] ダイアログで削除するか、ブレークポイントを設定している行をダブルクリックするか、またはその行を右クリックして [Remove Breakpoint] をクリックします。あるいは、右クリック メニューで [Breakpoints] > [Remove All Breakpoints] の順にクリックすると、2 つのブレークポイントを一度に削除できます。

5.11.5 ソフトウェア ブレークポイントの設定

ブレークポイントの種類をハードウェアからソフトウェアに変更するには：

- [Debugger] > [Settings] の順にクリックして、**[Configuration]** タブをクリックします。
- [Use Software Breakpoints] のラジオボタンをクリックします。
- [OK] をクリックします。

これで、今までのハードウェアブレークポイントではなく、ソフトウェアブレークポイントを利用できるようになります。

注： ソフトウェアブレークポイントを使用してデバッグを行うと、デバイスの耐久性が低下します。したがって、製品用デバイスにはソフトウェアブレークポイントによるデバッグは行わないようにしてください。

1. ソフトウェア ブレークポイントを1つだけ設定するには、**5.11.2 項「単一のハードウェア ブレークポイントの設定」**の説明に従ってください。
 - ソフトウェア ブレークポイントを設定すると、[Output] ウィンドウには次の内容が表示されます。
Programming software breakpoint(s)...
Software breakpoint(s) set.
 - このチュートリアルで既にハードウェア ブレークポイントを設定している場合は、変数が [Watch] ウィンドウに追加されており、ソフトウェア ブレークポイントでそのまま使用できます。
2. ソフトウェア ブレークポイントを複数設定するには、**5.11.3 項「複数のハードウェア ブレークポイントの設定」**の説明に従ってください。
 - ソフトウェア ブレークポイントでは、ブレークポイントを越えて命令が実行されること (スキッド) はなく、プログラムはブレークポイントで必ず停止します。このため、[Watch] ウィンドウでの値の変化がハードウェア ブレークポイントの場合と異なることがあります。
 - ソフトウェア ブレークポイントには、設定可能な数の上限があります。このチュートリアルでは2つしか使用していませんが、ソフトウェア ブレークポイントは999個までに制限されています。
3. ストップウォッチはハードウェア ブレークポイントと組み合わせて使用します。ソフトウェア ブレークポイントでストップウォッチを使用することもできますが、選択したソフトウェア ブレークポイントはハードウェア ブレークポイントに変換されます。これは、[Output] ウィンドウで次のように表示されます。
Converting breakpoint types...
Breakpoint type conversion complete.
5.11.4 項「ブレークポイントとストップウォッチの併用」で説明した手順に従ってください。
4. このチュートリアルを進める前に、ブレークポイントの種類を再びハードウェアに戻してください。[Debugger] > [Settings] の順にクリックして、[Configuration] タブをクリックします。[Use Hardware Breakpoints] のラジオボタンをクリックして [OK] をクリックします。

5.12 アプリケーションのプログラミング

プログラムのデバッグが完了して正常な動作が確認できたら、次はデバイスへのプログラミングを実行して最終デザインでデバイスが単独動作できるようにします。このプログラミング時にはデバッグ用の予約リソースが解放され、アプリケーションで使用できるようになります。

アプリケーションのプログラミングを行うには：

1. [Debugger] > [Select Tool] > [None] の順にクリックして、MPLAB ICD 3 インサーキット デバッガをデバッガとしての指定から解除します。
2. [Programmer] > [Select Programmer] > [ICD 3] の順にクリックして、MPLAB ICD 3 インサーキット デバッガをプログラマとして指定します。
3. オプション: ID メモリをサポートしたデバイスの場合、[Configure] > [ID Memory] の順にクリックして、ID を設定します。
4. [Programmer] > [Settings] の順にクリックして、[Program Memory] タブでプログラミングのパラメータを設定します。
5. プロジェクト ツールバーで、[Build Configuration] ドロップダウンリストから「Release」を選択します。次に、[Project] > [Build All] の順にクリックします。
6. [Programmer] > [Program] の順にクリックします。

これで、アプリケーションは単独で動作するはずですが、デモボードのリセット (MCLR) ボタンを押すと、カウントが再スタートします。

ボタンを押すとプログラムが開始または終了するようにプログラム コードを変更することもできます。プログラムを変更するには、MPLAB ICD 3 インサーキット デバッガをデバッガとして指定する必要があります。

1. [Programmer] > [Select Programmer] > [None] の順にクリックして、MPLAB ICD 3 インサーキット デバッガをプログラマとしての指定から解除します。
2. [Debugger] > [Select Tool] > [ICD 3] の順にクリックして、MPLAB ICD 3 インサーキット デバッガをデバッガとして指定します。
3. counter.c のコードを必要に応じて書き換えます (コード例は示しません。演習として考えてみてください)。
4. プロジェクト マネージャ ツールバーで、[Build Configuration] ドロップダウン リストから「Debug」を選択します。次に、[Project] > [Build All] の順にクリックします。
5. [Debugger] > [Program] の順にクリックします。
6. 必要に応じて、実行、ステップ実行、デバッグを行ってください。



MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

第 2 部 — トラブルシューティング

第 6 章 よく寄せられる質問 (FAQ).....	49
第 7 章 エラー メッセージ	53

MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

第 6 章 よく寄せられる質問 (FAQ)

6.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガ システムについてよく寄せられる質問とその回答を紹介します。

- 動作のしくみに関する FAQ
- 不具合に関する FAQ

6.2 動作のしくみに関する FAQ

- **デバイスのシリコンには、MPLAB ICD 3 インサーキット デバッガと通信を行うためにどのような機能が内蔵されていますか。**

MPLAB ICD 3 インサーキット デバッガは、ISCP インターフェースを利用してフラッシュ シリコンと通信を行います。そして、テスト メモリにロードされたデバッグ実行プログラムを使用します。

- **デバッグ実行プログラムを動作させるとプロセッサのスループットは低下しますか。**

Run モードではデバッグ実行プログラムは動作しないので、コード実行のスループットには一切影響しません。つまり、ターゲット デバイスの実行サイクルがデバッガによって占有されることはありません。

- **MPLAB ICD 3 インサーキット デバッガと他のインサーキット エミュレータ / デバッガにはどのような違いがありますか。**

2.2 項「MPLAB ICD 3 インサーキット デバッガと MPLAB ICE 2000/4000 インサーキット エミュレータの違い」および 2.3 項「MPLAB ICD 3 インサーキット デバッガと MPLAB ICD 2 デバッガの違い」を参照してください。

- **MPLAB ICD 3 インサーキット デバッガが MPLAB ICD 2 よりも多くの機能を実行できるのは、MPLAB IDE とどのようなインターフェースで接続されているためですか。**

MPLAB ICD 3 インサーキット デバッガはテスト領域にロードされたデバッグ実行プログラムを利用して通信を行います。デバッグ実行ファイルは、より効率的な通信が行えるように最適化されています。デバッガには FPGA、大容量 SRAM バッファ (1Mx8)、High-Speed USB インターフェースが実装されています。SRAM にプログラム メモリのイメージをダウンロードして保存することで、プログラミングの高速化が図られています。また、デバッガの FPGA はデバイスのインサーキット デバッガ モジュールとの通信を高速化するアクセラレータの役割を果たします。

- **MPLAB ICE 2000/4000 デバッガでは、データに対して複雑なトリガを実行するにはそのデータをバスに転送する必要がありました。MPLAB ICD 3 インサーキット デバッガでもその必要がありますか。例えば、フラグが High になったときに停止することはできますか。**

MPLAB ICE 2000/4000 デバッガでは、専用のデバッガ チップ (-ME) を使用してモニタリングを行っています。MPLAB ICD 3 インサーキット デバッガには -ME チップはないので、外部からバスをモニタリングすることはありません。MPLAB ICD 3 インサーキット デバッガでは、外部ブレイクポイントを使用する

のではなく、デバッグ エンジンに内蔵されたブレークポイント回路を使用します。つまりバスとブレークポイントはデバイス内部でモニタリングされます。

- **MPLAB ICD 3 インサーキット デバッガでは MPLAB ICE 2000/4000 のように複合ブレークポイントは利用できますか。**

はい。データ メモリ内の値に基づいてブレークできます。また、指定した複数のイベントが発生した場合のみプログラム実行を停止するシーケンス ブレークポイントも設定できます。ただし MPLAB ICE 2000 では 4 つのシーケンスを設定できますが、MPLAB ICD 3 インサーキット デバッガでは 2 つまでしか設定できません。AND 条件でブレークポイントを設定したり、PASS カウントを実行することもできます。詳細は、9.3.1 項「[Breakpoints] ダイアログ」を参照してください。

- **ドライバ ボードは光学的または電氣的に絶縁されていますか。**

DC は光学絶縁されていますが、AC は光学絶縁されていません。現在のシステムにフロート電圧または高電圧 (120V) を印加することはできません。

- **標準ケーブルには何か制約がありますか。**

標準の ICSP RJ-11 ケーブルで可能な最大クロック速度は約 15 Mbps です。フルスピードの場合、dsPIC33F DSC の動作速度は 15 Mbps を上回ります。

- **MPLAB ICD 3 インサーキット デバッガによってプログラムの実行速度は低下しますか。**

MPLAB ICD 3 インサーキット デバッガによって実行サイクルが占有されることはありません。データの出力はシリコン内部のステート マシンによって行われます。

- **dsPIC DSC を任意の速度で動作させてデバッグを行うことはできますか。**

MPLAB ICD 3 は、デバイスのデータシートで規定された任意の速度でデバイスを動作させてデバッグを実行できます。

- **ピン 6 (LVP ピン) はどのような働きをしますか。**

ピン 6 は、LVP (Low-Voltage Programming) 接続用に予約されています。

6.3 不具合に関する FAQ

- **PC が一度省電力 / 休止モードになった後、デバッガが動作しなくなりました。どうしたのでしょうか。**

デバッガを (特にデバッガとして) 長時間使用する際は、Windows コントロールパネルの [電源オプション] で休止モードを無効にしておくようにしてください ([休止状態] タブで [休止状態を有効にする] のチェックを外します)。こうすると、すべての USB サブシステム コンポーネントで完全な通信を維持できます。

- **周辺モジュールに [Freeze on Halt] を設定していないのに突然フリーズしてしまいます。なぜでしょうか。**

dsPIC30F/33F および PIC24F/H デバイスの場合、デバッガは周辺制御レジスタの予約ビット (通常、ビット 14 または 5) を Freeze ビットとして使用します。レジスタ全体に書き込みを実行したときに、このビットが上書きされた可能性があります (このビットはデバッグ モードではユーザーからアクセス可能です)。

この問題を防ぐには、レジスタ全体を書き換える命令 (MOV) ではなく、アプリケーションで変更が必要なビットだけを書き換える命令 (BTS、BTC) を使用するようにします。

- **16 ビット デバイスを使用中に、予期しないリセットが発生しました。どのようにして原因を調べればよいでしょうか。**

以下の点を確認してみてください。

- RCON レジスタを確認してリセット要因を調べる。
- 割り込みサービスルーチン (ISR) でトラップ/割り込みを処理する。
例えば、下記の trap.c のようなコードを挿入します。

```
void __attribute__((__interrupt__)) _OscillatorFail(void);
:
void __attribute__((__interrupt__))
_AltOscillatorFail(void);
:
void __attribute__((__interrupt__)) _OscillatorFail(void)
{
    INTCON1bits.OSCFAIL = 0;          //Clear the trap flag
    while (1);
}
:
void __attribute__((__interrupt__))
_AltOscillatorFail(void)
{
    INTCON1bits.OSCFAIL = 0;
    while (1);
}
:
```

- ASSERT を使用する。

- **デバッガが完了したコードをデバイスに書き込みましたが、動作しません。何が原因でしょうか。**

以下の点を確認してみてください。

- デバッガをプログラマとして選択してヘッダボードへの書き込みを行いませんでしたか。ヘッダボードに搭載されているのは -ICE/-ICD バージョンのデバイスで、実際のデバイスとは動作が異なることがあります。デバッガをプログラマとして使用して書き込めるのは、通常のデバイスのみです。通常のデバイスにもオンボードに ICE/ICD 回路を内蔵したデバイスがありますが、ヘッダボードに搭載されている特殊な -ICE/-ICD デバイスとは別のものです。
- デバッガをデバッガとして選択したままデバイスにプログラミングを行いませんでしたか。デバッガをデバッガとして選択してプログラミングを行うと、デバッガはプログラムメモリにデバッグ実行プログラムを書き込み、デバイスが持つデバッグ用の機能のセットアップを行います (2.7.1 項「デバッグ開始までの操作手順」を参照)。完成した (リリース用) コードを書き込むには、デバッガをプログラマとして指定する必要があります。
- [Build Configuration] ドロップダウンリストまたは [Project] メニューで「Release」を選択しましたか。最終版の (リリース用) コードをプログラミングするには、この操作が必要です。プロジェクトを再ビルドしてデバイスにもう一度書き込みを行い、コードが動作するかどうか試してみてください。

- **ソフトウェアブレークポイントを設定していないのに、コードに 1 つ設定されています。なぜでしょうか。**

これはファントムブレークポイントと呼ばれるものです。ごくまれに、設定していないブレークポイントが有効になってしまうことがあります。このようなブレークポイントは無効にするか削除してください。

- この FAQ に記載されていない問題が発生した場合はどうすればよいでしょうか。
次のリソースを参照してください。
 - 第 10 章「ハードウェア仕様」
 - 2.9 項「デバッガが使用するリソース」
 - 7.2 項「特定の状況で表示されるエラー メッセージ」
 - 7.3 項「一般的な対処方法」

第 7 章 エラー メッセージ

7.1 はじめに

MPLAB ICD 3 インサーキット デバッガで表示されるエラー メッセージにはさまざまな種類があります。そのいくつかは特定の状況で表示されるエラー メッセージであり、それ以外は、一般的な対処法で解決できるものです。

- 特定の状況で表示されるエラー メッセージ
- 一般的な対処方法

7.2 特定の状況で表示されるエラー メッセージ

以下に、MPLAB ICD 3 インサーキット デバッガのエラー メッセージをエラー コードの番号順に示します。

注： 現時点では、エラー メッセージにエラー コードが表示されない場合があります。ヘルプの [検索] タブでメッセージを検索して、強調表示された項目を参照してください。

以下のエラー メッセージで「%x」（変数）と表記された部分は、実際のエラー メッセージでは状況に応じた文字列が表示されます。

ICD3Err0001: Failed while writing to program memory.

(プログラム メモリへの書き込みに失敗しました。)

ICD3Err0002: Failed while writing to EEPROM.

(EEPROM への書き込みに失敗しました。)

ICD3Err0003: Failed while writing to configuration memory.

(コンフィギュレーション メモリへの書き込みに失敗しました。)

7.3.1 項「読み出し / 書き込みエラーの対処方法」を参照してください。

ICD3Err0005: ICD 3 is currently busy and cannot be unloaded at this time.

(ICD 3 は現在ビジーで選択を解除できません。)

MPLAB ICD 3 インサーキット デバッガをデバッガまたはプログラマとしての指定から解除しようとしてこのエラーが表示された場合は、次の解決方法を試してください。

1. デバッガが現在実行中のタスクが終了するまで待つ。その後、もう一度デバッガの選択を解除してください。
2. [Halt] をクリックして実行中のアプリケーションを停止する。その後、もう一度デバッガの選択を解除してください。
3. デバッガのケーブルを抜き、PC から取り外す。その後、もう一度デバッガの選択を解除してください。
4. MPLAB IDE を終了する。

ICD3Err0006: Failed while writing to user ID memory.

(ユーザー ID メモリへの書き込みに失敗しました。)

ICD3Err0007: Failed while reading program memory.

(プログラム メモリの読み出しに失敗しました。)

ICD3Err0008: Failed while reading EEPROM.

(EEPROM の読み出しに失敗しました。)

ICD3Err0009: Failed while reading configuration memory.

(コンフィギュレーション メモリの読み出しに失敗しました。)

ICD3Err0010: Failed while reading user ID memory.

(ユーザー ID メモリの読み出しに失敗しました。)

7.3.1 項「読み出し / 書き込みエラーの対処方法」を参照してください。

ICD3Err0011: Bulk erase failed. (一括消去に失敗しました。)

7.3.1 項「読み出し / 書き込みエラーの対処方法」を参照してください。

それでも解決しない場合は、別のデバイスで試してください。

ICD3Err0012: Download debug exec failed.

(デバッグ実行プログラムのダウンロードに失敗しました。)

[Debugger] メニューから書き込みを実行しようとしてこのエラーが表示された場合は、次の解決方法を試してください。

1. MPLAB ICD 3 インサーキット デバッガをデバッガとしての指定から解除する。
2. プロジェクトを閉じて MPLAB IDE を終了する。
3. MPLAB IDE を再起動してもう一度プロジェクトを開く。
4. もう一度デバッガをデバッガとして指定し、ターゲットデバイスへの書き込みを実行する。

それでも解決しない場合は、7.3.4 項「インストール ファイルが破損している場合の対処方法」を参照してください。

ICD3Err0013: NMMR register write failed.

(NMMR レジスタへの書き込みに失敗しました。)

ICD3Err0014: File register write failed.

(ファイル レジスタへの書き込みに失敗しました。)

7.3.2 項「デバッガとターゲットの通信エラーの対処方法」を参照してください。

ICD3Err0015: Data transfer was unsuccessful. %d byte(s) expected, %d byte(s) transferred. (データ転送に失敗しました。%d バイトのうち %d バイトしか転送されていません。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0016: Cannot transmit. ICD 3 not found.

(送信できません。ICD 3 が見つかりません。)

デバッガが PC に正しく接続されていません。

ICD3Err0017: File register read failed.

(ファイル レジスタの読み出しに失敗しました。)

ICD3Err0018: NMMR register read failed.

(NMMR レジスタの読み出しに失敗しました。)

ICD3Err0019: Failed while reading emulation registers.

(エミュレーション レジスタの読み出しに失敗しました。)

ICD3Err0020: Failed while writing emulation registers.

(エミュレーション レジスタの書き込みに失敗しました。)

7.3.2 項「デバッガとターゲットの通信エラーの対処方法」を参照してください。

ICD3Err0021: Command not echoed properly. Sent %x, received %x.

(コマンドのエコーが正しくありません。%x を送信して %x を受信しました。)

ICD3Err0022: Failed to get ICD 3 version information.

(ICD 3 のバージョン情報の取得に失敗しました。)

ICD3Err0023: Download FPGA failed. (FPGA のダウンロードに失敗しました。)

ICD3Err0024: Download RS failed. (RS のダウンロードに失敗しました。)

ICD3Err0025: Download AP failed. (AP のダウンロードに失敗しました。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0026: Download program exec failed.

(プログラム実行ファイルのダウンロードに失敗しました。)

[Debugger] メニューから書き込みを実行しようとしてこのエラーが表示された場合は、次の解決方法を試してください。

1. MPLAB ICD 3 インサーキット デバッガをデバッガとしての指定から解除する。
2. プロジェクトを閉じて MPLAB IDE を終了する。
3. MPLAB IDE を再起動してもう一度プロジェクトを開く。
4. もう一度デバッガをデバッガとして指定し、ターゲットデバイスへの書き込みを実行する。

それでも解決しない場合は、7.3.4 項「インストールファイルが破損している場合の対処方法」を参照してください。

ICD3Err0027: Bulk transfer failed due to invalid checksum.

(チェックサム エラーにより一括転送に失敗しました。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

また、使用しているケーブルの長さが適切であることを確認してください。

ICD3Err0028: Download device database failed.

(デバイス データベースのダウンロードに失敗しました。)

このエラーが表示された場合は、次の解決方法を試してください。

1. もう一度ダウンロードしてください。一過性のエラーの場合があります。
2. 手動でダウンロードしてください。[Debugger] > [Settings] の順にクリックして、[Configuration] タブをクリックし、[Manual Download] をクリックします。最も数の大きい .jam ファイルを選択して [Open] をクリックします。

ICD3Err0029: Communication failure. Unexpected command echo response %x received from ICD 3. (通信エラー。ICD 3 から予期しないコマンド エコー %x が返されました。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0030: Unable to read/find firmware File %s.

(ファームウェア ファイル %s が見つからないか、読み出せません。)

Hex ファイルが存在する場合は、次の解決方法を試してください。

- 再接続してもう一度試してみる。
- それでも解決しない場合は、ファイルが破損している場合があります。MPLAB IDE を再インストールしてください。

Hex ファイルが存在しない場合は、次の解決方法を試してください。

- MPLAB IDE を再インストールする。

ICD3Err0031: Failed to get PC. (PC の取得に失敗しました。)

ICD3Err0032: Failed to set PC. (PC のセットに失敗しました。)

7.3.2 項「デバッガとターゲットの通信エラーの対処方法」を参照してください。

ICD3Err0033: %d bytes expected, %d bytes received.

(%d バイトのうち %d バイトしか受信できませんでした。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0034: This version of MPLAB IDE does not support hardware revision %06x. Please upgrade to the latest version of MPLAB IDE before continuing.

(このバージョンの MPLAB IDE はハードウェア リビジョン %06x をサポートしていません。最新バージョンの MPLAB IDE にアップグレードしてから作業を続けてください。)

最新の MPLAB IDE を www.microchip.com から入手してください。

ICD3Err0035: Failed to get Device ID. (デバイス ID の取得に失敗しました。)

7.3.1 項「読み出し / 書き込みエラーの対処方法」を参照してください。

ICD3Err0036: MPLAB IDE has lost communication with ICD 3.

(MPLAB IDE と ICD 3 の通信が切断されました。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0037: Timed out waiting for response from ICD 3.

(タイムアウト エラー。ICD 3 からの応答がありません。)

ICD3Err0038: Failed to initialize ICD 3. (ICD 3 の初期化に失敗しました。)

ICD3Err0039: ICD 3 self-test failed. (ICD 3 のセルフテストに失敗しました。)

これらのエラーは、デバッガが応答していない場合に表示されます。

1. デバッガのケーブルを抜き、PC から一旦取り外してから差し直します。
2. MPLAB IDE 上でデバッガに再接続します。
3. それでも解決しない場合はマイクロチップ社までお問い合わせください。

ICD3Err0040: The target device is not ready for debugging. Please check your configuration bit settings and program the device before proceeding.

(ターゲット デバイスでデバッグの準備ができていません。コンフィギュレーションビットの設定を確認してデバイスのプログラミングを行ってから続行してください。)

このメッセージが表示されるのは、デバイスに最初にプログラミングせずに実行しようとした場合です。このような場合、またはデバイスのプログラミング直後にこのメッセージが表示された場合は、7.3.6 項「デバッグ エラーの対処方法」を参照してください。

ICD3Err0041: While receiving streaming data, ICD 3 has gotten out-of-sync with MPLAB IDE. To correct this you must reset the target device.

(ストリーミング データ受信中に ICD 3 と MPLAB IDE の同期がずれました。)

この問題を解決するにはターゲット デバイスのリセットが必要です。)

[Halt]、[Reset] を実行してから再度 [Run] を実行してください。それでも解決しない場合は、次の解決方法を試してください。

1. デバッガのケーブルを抜き、PC から一旦取り外してから差し直します。
2. MPLAB IDE 上でデバッガに再接続します。
3. ターゲットの動作速度が、[Settings] ダイアログの [Clock] タブに入力した値と同じであることを確認します。
4. 再度 [Run] を実行します。

ICD3Err0045: You must connect to a target device to use MPLAB ICD 3.

(MPLAB ICD 3 を使用するには、ターゲット デバイスに接続する必要があります。)

電源が検出されていません。

1. デバッガとターゲットの間で VDD と GND が接続されていることを確認してください。
2. ターゲットに電源が供給されていることを確認してください。
3. デバッガが検出できる十分なレベルの電源がターゲットに供給されていることを確認してください (第 10 章「ハードウェア仕様」参照)。

ICD3Err0046: An error occurred while trying to read the stopwatch count. The stopwatch count may not be accurate.

(ストップウォッチのカウンtr読み出し中にエラーが発生しました。ストップウォッチのカウンtr値が正確でない可能性があります。)

7.3.2 項「デバッガとターゲットの通信エラーの対処方法」を参照してください。

ICD3Err0047: Bootloader download failed.

(ブートローダのダウンロードに失敗しました。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0052: The current ICD 3 hardware version %x, is out of date. This version of MPLAB IDE will support only version %x or higher.

(現在の ICD 3 ハードウェア バージョン %x は旧式です。このバージョンの MPLAB IDE ではバージョン %x 以降しかサポートされません。)

最新ファームウェアのダウンロードを確認するダイアログで **[Cancel]** をクリックしませんでしたか。もしそうなら、最新ファームウェアをすぐにダウンロードしてください。 **[Debugger] > [Settings]** の順にクリックして、 **[Configuration]** タブをクリックし、 **[Manual Download]** をクリックします。最も数の大きい .jam ファイルを選択して **[Open]** をクリックします。

ダウンロードするファイルが見つからない場合や、上記手順でも解決しない場合 (ファイルが破損している可能性があります) は、最新バージョンの MPLAB IDE を入手してインストールしてください。最新の MPLAB IDE を www.microchip.com から入手してください。

ICD3Err0053: Unable to get ICD 3 protocol versions.

(ICD 3 のプロトコル バージョンを取得できません。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0054: MPLAB IDE's ICD 3 protocol definitions are out of date. You must upgrade MPLAB IDE to continue. (MPLAB IDE の ICD 3 プロトコル定義が最新ではありません。MPLAB IDE をアップグレードしてから作業を継続してください。)

最新の MPLAB IDE を www.microchip.com から入手してください。

ICD3Err0055: Unable to set firmware suite version.

(ファームウェアスイートのバージョンを設定できません。)

ICD3Err0056: Unable to get voltages from ICD 3.

(ICD 3 から電圧を取得できません。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0057: Self-test could not be completed.

(セルフテストを完了できませんでした。)

ICD3 テスト インターフェイス ボードを使用していることを確認してください。

7.3.2 項「デバッガとターゲットの通信エラーの対処方法」も参照してください。

ICD3Err0063: Test interface clock write failure. Please ensure that the tester is properly connected. (テスト インターフェイス クロックの書き込みに失敗しました。テストが正しく接続されていることを確認してください。)

ICD3Err0064: Test interface data write failure.

(テスト インターフェイスのデータ書き込みに失敗しました。)

ICD3Err0065: Test interface clock read failure.

(テスト インターフェイスのクロック読み出しに失敗しました。)

ICD3Err0066: Test interface data read failure.

(テスト インターフェイスのデータ読み出しに失敗しました。)

クロック / データがデバッガから出力されていません。接続を確認してもう一度試してください。

ICD3Err0067: Failed to set/clear software breakpoint.

(ソフトウェア ブレークポイントの設定 / 解除に失敗しました。)

再プログラミングして、もう一度試してください。

ICD3Err0068: Failed while writing to boot FLASH memory.

(ブート フラッシュ メモリへの書き込みに失敗しました。)

ICD3Err0069: Failed while reading boot FLASH memory.

(ブート フラッシュ メモリの読み出しに失敗しました。)

ICD3Err0070: Failed while writing peripheral memory.

(周辺メモリへの書き込みに失敗しました。)

ICD3Err0071: Failed while reading peripheral memory.

(周辺メモリの読み出しに失敗しました。)

7.3.1 項「読み出し / 書き込みエラーの対処方法」を参照してください。

ICD3Err0072: Unable to send freeze peripheral information.

(周辺モジュールのフリーズ情報を送信できません。)

7.3.3 項「デバッガと PC の通信エラーの対処方法」を参照してください。

ICD3Err0073: Device is code protected. (デバイスはコード保護されています。)

デバイスに読み出し、書き込み、ブランク チェックまたはベリファイの操作を行おうとしましたが、デバイスのコード保護が有効になっており、コードの読み出しや変更が行えません。コード保護に関するコンフィギュレーションビットの設定を確認してください。

コード保護を無効にするには、デバイスのデータシートを参照して該当するコンフィギュレーションビットをコード内または [Configuration Bits] ウィンドウ ([\[Configure\]](#) > [\[Configuration Bits\]](#)) でセットまたはクリアします。その後で、デバイス全体を消去そして再プログラミングしてください。

ICD3Err0082: Test interface LVP failure. (テスト インターフェースの LVP エラー。)

ICD3Err0083: Test interface MCLR failure.

(テスト インターフェースの MCLR エラー。)

7.3 一般的な対処方法

以下、問題が発生した場合の一般的な対処方法を紹介します。

- 読み出し / 書き込みエラーの対処方法
- デバッガとターゲットの通信エラーの対処方法
- デバッガと PC の通信エラーの対処方法
- インストール ファイルが破損している場合の対処方法
- USB ポート通信エラーの対処方法
- デバッグ エラーの対処方法
- 内部エラーの対処方法

7.3.1 読み出し / 書き込みエラーの対処方法

読み出し / 書き込みエラーが発生した場合は、次の事項を確認してください。

1. [Abort] をクリックすると、読み出し / 書き込みエラーが発生することがあります。
2. 同じ操作を再試行してください。一過性のエラーの場合があります。
3. ターゲットに電源が供給されており、デバイスの電圧レベルが適正であることを確認してください。デバイスの適正電圧は、各デバイスのデータシートを参照してください。
4. デバッガとターゲットの接続が正しいこと (PGC と PGD が接続されていること) を確認してください。
5. 書き込みエラーの場合は、[Settings] ダイアログの [Program Memory] タブで [Erase all before Program] にチェックが入っていることを確認してください。
6. 使用しているケーブルの長さが適切であることを確認してください。

7.3.2 デバッガとターゲットの通信エラーの対処方法

MPLAB ICD3 インサーキット デバッガとターゲット デバイスの同期がとれていません。

1. [Reset] をクリックしてから同じ操作を再試行してください。
2. 使用しているケーブルの長さが適切であることを確認してください。

7.3.3 デバッガと PC の通信エラーの対処方法

MPLAB ICD 3 インサーキット デバッガと MPLAB IDE の同期がとれていません。

1. デバッガのケーブルを抜き、PC から一旦取り外してから差し直してください。
2. デバッガに再接続してください。
3. 同じ操作を再実行してください。一過性のグリッチによるエラーの場合があります。
4. インストールされている MPLAB IDE のバージョンが、MPLAB ICD 3 インサーキット デバッガにロードされているファームウェアのバージョンに対応していない可能性があります。7.3.4 項「インストール ファイルが破損している場合の対処方法」の手順に従ってください。

7.3.4 インストール ファイルが破損している場合の対処方法

MPLAB IDE が完全にインストールされていないか、MPLAB IDE のファイルが破損している可能性があります。

1. すべてのバージョンの MPLAB IDE を PC から完全にアンインストールしてください。
2. 使用するバージョンの MPLAB IDE のみを再インストールします。
3. それでも解決しない場合はマイクロチップ社までお問い合わせください。

7.3.5 USB ポート通信エラーの対処方法

通信ポートに問題があるか、存在しない通信ポートを指定している可能性があります。

1. MPLAB ICD 3 インサーキット デバッガに再接続してください。
2. デバッガが物理的に PC の適切な USB ポートに接続されていることを確認してください。
3. デバッガの [Settings] ダイアログで正しい USB ポートが選択されていることを確認してください。
4. USB ポートを他のデバイスが使用していないことを確認してください。
5. USB ハブを使用する場合は、電源付きのものを使用してください。
6. USB ドライバがロードされていることを確認してください。

7.3.6 デバッグ エラーの対処方法

MPLAB ICD 3 インサーキット デバッガがデバッグを実行できませんでした。これには多くの原因が考えられます。

デバッグに失敗する主な理由

1. オシレータが動作していない。オシレータに関するコンフィギュレーション ビットの設定を確認してください。
2. ターゲット ボードに電源が供給されていない。電源ケーブルの接続を確認してください。
3. MPLAB ICD 3 インサーキット デバッガと PC の物理的な接続に何らかの問題がある。USB 通信ケーブルの接続を確認してください。
4. デバッガとターゲット ボードの物理的な接続に何らかの問題がある。通信ケーブルの接続を確認してください。
5. デバイスのコード保護が有効になっている。コード保護に関するコンフィギュレーション ビットの設定を確認してください。
6. Release モードでプロジェクトの再ビルドを行おうとしている。プロジェクト ツールバーの [Build Configuration] ドロップダウン リストで「Debug」を選択してからプロジェクトを再ビルドしてください。

7. MPLAB IDE で MPLAB ICD 3 インサーキット デバッガがデバッガではなくプログラマとして選択されている。
8. デバッガと PC の通信が何らかの理由で中断された。MPLAB IDE 上でデバッガを再接続してください。
9. ターゲット アプリケーションに何らかの破損またはエラーがある。例えば、プロジェクトでデバッガ バージョンのリンカ スクリプト (例 : 18F8722i.lkr) でなく通常のリンカ スクリプト (例 : 18F8722.lkr) を使用した場合などです。ターゲット アプリケーションを再ビルドしてプログラミングし直してください。その後で、ターゲットでパワーオンリセットを実行します。
10. 他のコンフィギュレーション ビットの設定が干渉してデバッグが行えない。ターゲットがコードを実行できないようなコンフィギュレーション ビットの設定が行われていると、デバッガはコードをデバッグ モードにすることができません。
11. デバッガは、要求された動作を常に実行できるわけではありません。例えばターゲット アプリケーションが実行中の場合、デバッガはブレークポイントを設定できません。

その他の確認事項

1. 一過性のグリッチによるエラーの場合があります。同じ操作を再試行してください。
2. プログラミングそのものに問題があることも考えられます。プログラマ モードに切り替えて、なるべくシンプルなアプリケーション (LED 点滅プログラムなど) でターゲットへの書き込みテストを行ってください。このプログラムが動作しない場合は、ターゲットのセットアップに問題があることとなります。
3. ターゲット デバイスが何らかの損傷 (過電流など) を受けた可能性があります。開発環境では、電子部品に悪影響が及ぶことがよくあります。ターゲット デバイスを別のものに交換してみてください。
4. マイクロチップ社では、ほとんどのマイクロコントローラをサポートしたデモボードを多数用意しています。MPLAB ICD 3 インサーキット デバッガが正しく動作するかどうかを検証するには、正常動作が確認されたこれらのアプリケーションを使用するのも 1 つの方法です。または、テスト インターフェース ボードを使用してデバッガのセルフテストを行ってください (10.7 項「ICD 3 テスト インターフェース ボード」参照)。
5. デバッガの動作原理を確認して、アプリケーションを正しくセットアップしてください (第 2 章「動作原理」参照)。
6. それでも解決しない場合はマイクロチップ社までお問い合わせください。

7.3.7 内部エラーの対処方法

内部エラーは、通常なら表示されることはありません。これらは主にマイクロチップ社の社内開発に使用します。

主な原因として、インストール ファイルの破損が考えられます (7.3.4 項「インストール ファイルが破損している場合の対処方法」参照)。

また、システム リソースの減少が原因のこともあります。

1. システムを再起動してメモリを解放してください。
2. ハードディスク ドライブに十分な空き容量があること (および適度にデフラグされていること) を確認してください。

それでも解決しない場合はマイクロチップ社までお問い合わせください。



MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

第 3 部 — リファレンス

第 8 章 基本的なデバッグ機能	63
第 9 章 デバッガ機能一覧	65
第 10 章 ハードウェア仕様	79

ノート:

第 8 章 基本的なデバッグ機能

8.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガの基本的なデバッグ機能であるブレークポイントとストップウォッチについて説明します。

8.2 ブレークポイント

ブレークポイントを使用すると、コードの指定行でコード実行を停止できます。

ブレークポイントとトリガは同じリソースを使用します。このため、トリガを設定していると、その分だけブレークポイントを設定できる数が少なくなります。

ハードウェアブレークポイントまたはソフトウェアブレークポイントを選択するには：

1. **[Debugger]** > **[Settings]** の順にクリックして、**[Configuration]** タブをクリックします。
2. アプリケーションに設定したいブレークポイントの種類を選択します。ハードウェアブレークポイントとソフトウェアブレークポイントの欄に、それぞれの特長が表示されます (詳細は、**9.5.2 項「[Settings] ダイアログの [Configuration] タブ」**を参照してください)。

注： ソフトウェアブレークポイントを使用してデバッグを行うと、デバイスの耐久性が低下します。したがって、製品用デバイスにはソフトウェアブレークポイントによるデバッグは行わないようにしてください。

8.3 ストップウォッチ

ブレークポイントとストップウォッチを併用すると、コード実行の時間を測定できます。

コードにブレークポイントを設定するには、次のいずれかの操作を行います。

- コードの任意の行をダブルクリックまたは右クリックして、個別にブレークポイントを設定します。
- **[Debugger]** > **[Breakpoints]** の順にクリックして **[Breakpoints]** ダイアログを開き、複数のブレークポイントとその条件を設定します。詳細は、**9.3.1 項「[Breakpoints] ダイアログ」**を参照してください。

ストップウォッチを使用してブレークポイント間の時間を測定するには：

1. **[Breakpoints]** ダイアログを開きます (**[Debugger]** > **[Breakpoints]**)。
2. **[Breakpoints]** ダイアログで **[Stopwatch]** をクリックすると、**[Stopwatch]** ダイアログが開きます。
3. **[Start Condition]** のドロップダウンリストから、ストップウォッチ開始点となるブレークポイントを選択します。必要に応じて **[Start condition will cause the target device to halt]** のチェックを選択します。
4. **[Stop Condition]** のドロップダウンリストから、ストップウォッチ終了点となるブレークポイントを選択します。必要に応じて **[Stop condition will cause the target device to halt]** のチェックを選択します。

5. 必要に応じて [Reset stopwatch on run] のチェックを選択します。
6. [OK] をクリックします。

第 9 章 デバッガ機能一覧

9.1 はじめに

ここでは、MPLAB IDE のメニュー、ウィンドウ、ダイアログで利用できる MPLAB ICD 3 インサーキット デバッガの機能をまとめます。

- デバッグに関する機能
- デバッグに関するダイアログ / ウィンドウ
- プログラミングに関する機能
- [Settings] ダイアログ

9.2 デバッグに関する機能

[Debugger] メニューで MPLAB ICD 3 インサーキット デバッガをデバッガに選択すると、MPLAB IDE に次の機能が追加されます。

- [Debugger] メニュー — ドロップダウン メニューに項目が追加されます。
- デバッグに関する右クリック メニュー — このメニューに項目が追加されます。
- ツールバー / ステータス バー — メニュー バーの下にツールバーが表示されます。ステータス バーにも追加の情報が表示されます。

9.2.1 [Debugger] メニュー

[Run] (F9 キー)

ブレークポイントまで、または [Halt] が選択されるまでプログラム コードを実行します。

コード実行は、現在のプログラム カウンタ (ステータス バーに表示) から開始します。現在のプログラム カウンタ位置は、[Program Memory] ウィンドウのポインタでも表示されます。プログラム実行中は、一部の機能が無効になります。

[Animate]

[Animate] を実行すると、デバッガが 1 命令ずつ実行し、画面に表示されるレジスタの値もそれに合わせて更新されます。

Animate は Run よりも実行速度は遅くなりますが、[Special Function Register] ウィンドウや [Watch] ウィンドウでレジスタの値の変化を確認できます。

Animate を停止するには、メニューで **[Debugger] > [Halt]** の順にクリックするか、ツールバーの [Halt] をクリックするか、<F5> キーを押します。

[Halt] (F5 キー)

[Halt] を実行すると、プログラム コードの実行が停止します。[Halt] をクリックすると、ステータス情報が更新されます。

[Step Into] (F7 キー)

プログラム コードを 1 命令ずつ実行して停止します。

アセンブリ コードの場合は、1 命令 (1 サイクル命令または複数サイクル命令) を実行して停止します。1 命令を実行するたびに、すべてのウィンドウが更新されます。

C コードの場合は、コードを 1 行 (アセンブリ命令レベルでは複数の命令に相当することもあります) ずつ実行して停止します。実行が完了するたびに、すべてのウィンドウが更新されます。

注: SLEEP 命令には [Step Into] を実行しないでください。

[Step Over] (F8 キー)

現在のプログラム カウンタ位置の命令を実行します。CALL 命令があると、呼び出されたサブルーチンをすべて実行後、CALL 命令の次のアドレスで停止します。Step Over がなかなか終了せず、ハングしたと思われる場合は [Halt] をクリックしてください。

[Step Out]

利用できません。

[Reset] (F6 キー)

ターゲットプロセッサにリセットシーケンスを送ります。 $\overline{\text{MCLR}}$ が発行され、プログラム カウンタはリセットベクタに戻ります。

[Breakpoints]

[Breakpoints] ダイアログが開きます (9.3.1 項「[Breakpoints] ダイアログ」参照)。このダイアログでは複数のブレークポイントを設定できます。

注: コード行を右クリックまたはダブルクリックすると、その行にシンプルなブレークポイントを設定できます。

[Program]

コードをターゲット デバイスにダウンロードします。

[Read]

ターゲットのメモリを読み出します。情報は MPLAB IDE にアップロードされます。

[Erase Flash Device]

すべてのフラッシュ メモリを消去します。

[Debug Read]

デバッグ実行プログラムを利用してプログラム メモリを読み出します。

[Abort Operation]

プログラミング操作 (プログラム書き込み、読み出しなど) を中止します。操作を中断すると、デバイスの状態は不明となります。

[Reconnect]

PC と MPLAB ICD 3 インサーキット デバッガを再接続します。接続の状況は、[Output] ダイアログの [ICD 3] タブに表示されます。

[Settings]

プログラムの設定ダイアログが開きます (9.5 項「[Settings] ダイアログ」参照)。プログラムとファームウェアのオプションを設定します。

9.2.2 デバッガに関する右クリック メニュー

プログラム メモリやソース コード ファイルなどのコード表示画面を右クリックすると、デバッガに関する以下のメニュー項目が表示されます。ここに記載されていないメニュー項目は、MPLAB IDE ヘルプまたは MPLAB Editor ヘルプを参照してください。

[Set Breakpoint]

現在選択している行にブレークポイントを設定 (または解除) します。

[Breakpoints]

すべてのブレークポイントを一括して削除、有効化、無効化します。

[Run To Cursor]

現在のカーソル位置までプログラムを実行します。以前の [Run to Here] と同じです。

[Set PC at Cursor]

現在のカーソル位置にプログラム カウンタ (PC) をセットします。

[Center Debug Location]

現在 PC がセットされている行をウィンドウの中央に表示します。

9.2.3 ツールバー / ステータス バー

MPLAB ICD 3 インサーキット デバッガをデバッガとして指定すると、MPLAB IDE に以下のツールバーが表示されます。

- 基本的なデバッグ ツールバー ([Run]、[Halt]、[Animate]、[Step Into]、[Step Over]、[Step Out]、[Reset])。
- シンプルなプログラム ツールバー ([Read]、[Program]、[Erase Flash Device])。

MPLAB IDE デスクトップの一番下にあるステータス バーには、現在選択されているデバッガ (MPLAB ICD 3) やその他開発に関する情報が表示されます。ステータス バーの表示内容は、MPLAB IDE のオンライン ヘルプを参照してください。

9.3 デバッグに関するダイアログ/ウィンドウ

9.2 項「デバッグに関する機能」に記載したメニュー項目をクリックすると、デバッグに関する次のダイアログやウィンドウが開きます。

- [Breakpoints] ダイアログ
 - [Set Breakpoint] ダイアログ
 - [Stopwatch] ダイアログ
 - [Event Breakpoints] ダイアログ
 - [Sequenced Breakpoints] ダイアログ
 - [ANDed Breakpoints] ダイアログ

9.3.1 [Breakpoints] ダイアログ

ブレークポイントを設定するには、[Debugger] > [Breakpoints] の順にクリックします。

このダイアログでは、さまざまな種類のブレークポイントを設定できます。**[Add Breakpoint]** をクリックすると、ブレークポイントの一覧に新規ブレークポイントを追加できます。選択したデバイスの種類によっては、高度なブレークポイントに関するボタンが表示されることがあります。

9.3.1.1 [BREAKPOINTS] ダイアログ ウィンドウ

このウィンドウには、各ブレークポイントに関する情報が表示されます。

表 9-1: [BREAKPOINTS] ダイアログ ウィンドウ

Control	Function
Breakpoint Type	Type of breakpoint – program or data
Address	Hex address of breakpoint location
File Line #	File name and line number of breakpoint location
Enabled	Check to enable a breakpoint

この window に追加されたブレークポイントを右クリックすると、次のメニュー項目が表示されます。

- [Delete] — 選択したブレークポイントを削除します。
- [Edit/View] — [Set Breakpoint] ダイアログを開きます。
- [Delete All] — ウィンドウに表示されたブレークポイントをすべて削除します。
- [Disable All] — ウィンドウに表示されたブレークポイントをすべて無効にします。

9.3.1.2 [BREAKPOINTS] ダイアログのボタン

これらのボタンを使ってブレークポイントを追加したり、ブレーク条件を詳しく設定したりできます。また、ブレークポイントやトリガにストップウォッチを組み合わせ使用することもできます。

注： 表示されるボタンは、選択したデバイスの種類によって異なります。

表 9-2: [BREAKPOINTS] ダイアログのボタン

Control	Function	Related Dialog
Add Breakpoint	Add a breakpoint	Section9.3.2 “Set Breakpoint Dialog”
Stopwatch	Set up the stopwatch	Section9.3.3 “Stopwatch Dia-log”
Event Breakpoints	Set up break on an event	Section9.3.4 “Event Break-points Dialog”
Sequenced Breakpoints	Set up a sequence until break	Section9.3.5 “Sequenced Breakpoints Dialog”
ANDed Breakpoints	Set up ANDed condition until break	Section9.3.6 “ANDed Break-points Dialog”

9.3.2 [Set Breakpoint] ダイアログ

[Breakpoints] ダイアログで [Add Breakpoint] ボタンをクリックすると、このダイアログが表示されます。

[Breakpoints] ダイアログに追加するブレークポイントをここで設定します。

9.3.2.1 [PROGRAM MEMORY] タブ

プログラムメモリのブレークポイントをここで設定します。

表 9-3: プログラムメモリのブレークポイント

Control	Function
Address	Location of breakpoint in hex
Breakpoint Type	The type of program memory breakpoint. See the device data sheet for more information on table reads/writes. <i>Program Memory Execution</i> – break on execution of above address <i>TBLRD Program Memory</i> – break on table read of above address <i>TBLWT Program Memory</i> – break on table write to above address
Pass Count	Break on pass count condition. <i>Always break</i> – always break as specified in “Breakpoint type” <i>Break occurs Count instructions after Event</i> – wait Count (0-255) instructions before breaking after event specified in “Breakpoint type” <i>Event must occur Count times</i> – break only after event specified in “Breakpoint type” occurs Count (0-255) times

9.3.2.2 [DATA MEMORY] タブ

データメモリのブレークポイントをここで設定します。

表 9-4: データメモリのブレークポイント

Control	Function
Address	Location of breakpoint in hex
Breakpoint Type	The type of data memory breakpoint. See the device data sheet for more information on X Bus reads/writes. <i>X Bus Read</i> – break on an X bus read of above address <i>X Bus Read Specific Byte</i> – break on an X bus read of above address for the specific byte value in “Specific Value” <i>X Bus Read Specific Word</i> – break on an X bus read of above address for the specific word value in “Specific Value” <i>X Bus Write</i> – break on an X bus write of above address <i>X Bus Write Specific Byte</i> – break on an X bus write of above address for the specific byte value in “Specific Value” <i>X Bus Write Specific Word</i> – break on an X bus write of above address for the specific word value in “Specific Value”
Pass Count	Break on pass count condition. <i>Always break</i> – always break as specified in “Breakpoint type” <i>Break occurs Count instructions after Event</i> – wait Count (0-255) instructions before breaking after event specified in “Breakpoint type” <i>Event must occur Count times</i> – break only after event specified in “Breakpoint type” occurs Count (0-255) times

9.3.3 [Stopwatch] ダイアログ

[Breakpoints] ダイアログで [Stopwatch] ボタンをクリックすると、このダイアログが表示されます。

ストップウォッチを使用すると、2つのブレークポイント/トリガ条件の間の経過時間を計測できます。ストップウォッチの値は10進数です。

表 9-5: ストップウォッチの設定

Control	Function
Start Condition	Select an available breakpoint or trigger condition to start the stopwatch. Available breakpoints/triggers are those previously added to the breakpoint dialog. Select None to clear the start condition. To halt the program run on this condition, check the check box next to “Start condition will cause the target device to halt”.
Stop Condition	Select an available breakpoint or trigger condition to stop the stopwatch. Available breakpoints/triggers are those previously added to the breakpoint dialog. Select None to clear the stop condition. To halt the program run on this condition, check the check box next to “Stop condition will cause the target device to halt”.
Reset stopwatch on run	Reset the stopwatch values to zero every time the program is run.

9.3.4 [Event Breakpoints] ダイアログ

[Breakpoints] ダイアログで [Event Breakpoints] ボタンをクリックすると、このダイアログが表示されます。

プログラム実行を必ず停止する条件をここで選択します。

- [Break on Watchdog Timer] — ウォッチドッグ タイマがタイムアウトになると必ずプログラム実行を停止します。コンフィギュレーション ビットでウォッチドッグ タイマが有効に設定されていることを確認してください。
- [Break on SLEEP instruction] — プログラム内の SLEEP 命令で実行を停止します。

9.3.5 [Sequenced Breakpoints] ダイアログ

[Breakpoints] ダイアログで [Sequenced Breakpoints] ボタンをクリックすると、このダイアログが表示されます。

ここではブレークポイントの発生順を設定します。ブレークポイントのシーケンス実行はボトムアップ方式で行われます。つまり、シーケンスの最後のブレークポイントが最初に発生します。

シーケンスにブレークポイントを追加するには：

- [Available Breakpoints] リストでブレークポイントをクリックします。[Available Breakpoints] リストには、[Breakpoints] ダイアログに追加済みのブレークポイント / トリガが表示されます。
- [Sequences] リストでシーケンスをクリックします。
- [Add] をクリックします。

シーケンス内でのブレークポイントの順番を変更するには、[Sequences] リストでブレークポイントをドラッグアンドドロップします。

シーケンスからブレークポイントを削除するには：

- [Sequences] リストでブレークポイントをクリックします。
- [Remove] をクリックします。

9.3.6 [ANDed Breakpoints] ダイアログ

[Breakpoints] ダイアログで [ANDed Breakpoints] ボタンをクリックすると、このダイアログが表示されます。

プログラム実行を停止する AND 条件 (ブレークポイント 1 とブレークポイント 2 が同時に発生した場合のみプログラム実行を停止する) を設定します。AND 条件で実行が停止するのは、データメモリのブレークポイントとプログラムメモリのブレークポイントが同時に発生した場合のみです。

ブレークポイントを AND 条件に追加するには：

- [Available Breakpoints] リストでブレークポイントをクリックします。[Available Breakpoints] リストには、[Breakpoints] ダイアログに追加済みのブレークポイント / トリガが表示されます。
- [Add] をクリックします。

AND 条件からブレークポイントを削除するには：

- [ANDed Breakpoints] リストでブレークポイントをクリックします。
- [Remove] をクリックします。

[Close]

このウィンドウを閉じます。

[Find]

[Find] ダイアログを開きます。[Find What] フィールドに検索文字列を入力するか、またはドロップダウンリストから選択します。また、エディタ ウィンドウで文字列を選択するか、文字列の上にカーソルを置いて [Find] ダイアログを開くと、その文字列が [Find what] フィールドに入った状態で開きます。

[Find] ダイアログでは、検索オプションと検索の方向を設定できます。[Up] を選択するとカーソル位置から上方向、[Down] を選択すると下方向に検索が実行されます。

[Find Next]

[Find] ダイアログの検索文字列を更に検索します。

<F3> キーを押しても [Find Next] と同じ動作となります。

<Shift> + <F3> キーを押すと、反対方向に [Find Next] が実行されます。

[Go To]

選択した場所へジャンプします。

- [Trigger] — トリガの位置にジャンプします。
- [Top] — ウィンドウの先頭にジャンプします。
- [Bottom] — ウィンドウの最後にジャンプします。

[Show Source]

ウィンドウの下部にソース コードを表示するかどうかを切り替えます。

[Refresh]

ウィンドウの表示内容を更新します。

[Properties]

ウィンドウのプロパティを設定します。

9.4 プログラミングに関する機能

[Programmer] メニューで MPLAB ICD 3 インサーキット デバッガをプログラマに選択すると、MPLAB IDE に次の機能が追加されます。

- [Programmer] メニュー
- ツールバー / ステータス バー

9.4.1 [Programmer] メニュー

[Program]

指定したメモリ領域 (プログラム メモリ、コンフィギュレーション ビット、ID ロケーション、EEPROM データ) にプログラミングを実行します。プログラミングのオプションは [Settings] ダイアログを参照してください。

[Verify]

指定したメモリ領域 (プログラム メモリ、コンフィギュレーション ビット、ID ロケーション、EEPROM データ) に書き込んだ内容をベリファイします。

[Read]

指定したメモリ領域 (プログラム メモリ、コンフィギュレーション ビット、ID ロケーション、EEPROM データ) から読み出しを実行します。読み出しのオプションは [Settings] ダイアログを参照してください。

[Blank Check All]

デバイスのメモリがすべて消去されてブランクであることを確認します。

[Erase Flash Device]

すべてのフラッシュ メモリを消去します。

[Settings]

プログラマ ダイアログが開きます (9.5 項「[Settings] ダイアログ」参照)。プログラマとファームウェアのオプションを設定します。

9.4.2 ツールバー / ステータス バー

MPLAB ICD 3 インサーキット デバッガをプログラマとして指定すると、MPLAB IDE に以下のツールバーが表示されます。

- 基本的なプログラム ツールバー ([Blank Check All]、[Read]、[Program]、[Verify]、[Erase Flash Device])。

MPLAB IDE デスクトップの一番下にあるステータス バーには、現在選択されているプログラマ (MPLAB ICD 3) やその他プログラミングに関する情報が表示されます。ステータス バーの表示内容は、MPLAB IDE のオンラインヘルプを参照してください。

9.5 [SETTINGS] ダイアログ

[Debugger] > [Settings] または [Programmer] > [Settings] の順にクリックすると [Settings] ダイアログが開き、MPLAB ICD 3 インサーキット デバッガを設定できます。

注： 表示されるタブは、選択したデバイスの種類によって異なります。

- [Settings] ダイアログの [Program Memory] タブ
- [Settings] ダイアログの [Configuration] タブ
- [Settings] ダイアログの [Freeze on Halt] タブ
- [Settings] ダイアログの [Status] タブ
- [Settings] ダイアログの [Clock] タブ
- [Settings] ダイアログの [Secure Segment] タブ
- [Settings] ダイアログの [Warning] タブ
- [Settings] ダイアログの [Power] タブ

9.5.1 [Settings] ダイアログの [Program Memory] タブ

このタブでは、デバッグ/プログラミングのオプションを設定します。

- [Allow MPLAB ICD 3 to select memories and ranges] — 選択したデバイスとデフォルト設定に基づいて、書き込むメモリとアドレス範囲をデバッガが自動で設定します。
- [Manually select memories and ranges] — プログラミングを行うメモリの種類とアドレス範囲をユーザーが設定します。

表 9-6: 手動で設定する場合のオプション

Memories	
Program	Check to program Program Memory into target.
Configuration	Check to program Configuration bits into target. Note: This memory is always programmed when debugger selected as a debugger.
EEPROM	Check to erase and then program EEPROM memory on target, if available. Uncheck to erase EEPROM memory on target.
ID	Check to program ID Memory into target.
Program Options	
Erase all before Program	Check to erase all memory before programming begins. Unless programming new or already erased devices, it is important to have this box checked. If not checked, the device is not erased and program code will be merged with the code already in the device.
Program Memory	
Start, End	The starting and ending hex address range in program memory for programming, reading, or verification. If you receive a programming error due to an incorrect end address, you need to perform a reconnect, correct the end address and program again. Note: The address range does not apply to the Erase function. The Erase function will erase all data on the device.

- 自動オプションの設定 — ビルド完了後にプログラミングを行う場合は [Program after successful build] に、プログラミング完了後に実行する場合は [Run after successful program] にチェックを入れます。

9.5.2 [Settings] ダイアログの [Configuration] タブ

このタブでは、デバッガの動作を設定します。

表 9-7: [CONFIGURATION] タブの項目

Download Firmware	Set up firmware download options.
Auto Download Latest Firmware	Check to allow automatic download of the latest version of firmware for the target device (recommended).
Manual Download	Manually select a firmware file to download to the target device.
Breakpoints	Depending on your selected device, you may be able to use software breakpoints. Review the text beneath each type of breakpoint to determine which is best for your current needs.
Use Hardware Breakpoints	This is the default/classic mode for breakpoint behavior. Using hardware breakpoints means: <ul style="list-style-type: none"> • Number of breakpoints: limited • Breakpoints are written to debug registers • Time to set breakpoints: minimal • Skidding: yes
Use Software Breakpoints	Using software breakpoints means: <ul style="list-style-type: none"> • Number of breakpoints: unlimited • Breakpoints are written to program memory • Time to set breakpoints: oscillator speed dependent – can take minutes • Skidding: no <p>Note: Using software breakpoints for debug impacts device endurance. Therefore, it is recommended that devices used in this manner not be used as production parts.</p>

9.5.3 [Settings] ダイアログの [Freeze on Halt] タブ

このタブでは、Halt 時にフリーズする周辺モジュールを選択します。

PIC18 MCU デバイス

[Freeze on Halt] チェックボックスで、デバイスのすべての周辺モジュールに対してフリーズの有効/無効を設定します。チェックを入れても周辺モジュールが停止しない場合があります。これは、その周辺モジュールが Freeze on Halt 機能をサポートしておらず、デバッガで機能を設定できないためです。

dsPIC30F/33F、PIC24F/H、PIC32MX デバイス

[Peripherals to Freeze on Halt] リストで、Halt 時にフリーズしたい周辺モジュールにチェックを入れます。チェックを外した周辺モジュールは、プログラム実行が停止しても動作を継続します。目的の周辺モジュールがリストにない場合は、[All Other Peripherals] にチェックを入れてください。チェックを入れても周辺モジュールが停止しない場合があります。これは、その周辺モジュールが Freeze on Halt 機能をサポートしておらず、デバッガで機能を設定できないためです。

[All Other Peripherals] を含め、すべての周辺モジュールを選択するには、[Check All] をクリックします。[All Other Peripherals] を含め、すべての周辺モジュールの選択を解除するには、[Uncheck All] をクリックします。

9.5.4 [Settings] ダイアログの [Status] タブ

このタブでは、MPLAB ICD 3 システムの状態を確認できます。

表 9-8: [STATUS] タブの項目

Versions	
Firmware Suite Version	Debugger firmware suite version. The firmware suite consists of the three items specified below.
FPGA Version	Internal FPGA chip firmware version.
Algorithm Plug-in Version	Debugger algorithm plug-in version. For your selected device, an algorithm is used to support the device plugged into the target.
OS Version	Debugger operating system version.
Voltages	
ICD 3 VPP	Debugger VPP.
ICD 3 VDD	Debugger VDD.
Target VDD	VDD sensed at target.
Refresh Voltages	Sensing of Status tab items occurs when the tab is achieved. To see updates otherwise, click this button.

9.5.5 [Settings] ダイアログの [Clock] タブ

このタブでは、動作クロック (命令) 速度を入力します。これは動作速度を設定するものではなく、デバッガに値を知らせるためのものです。

表 9-9: [CLOCK] タブのオプション

Target Run-Time Instruction Speed	
Speed value	Enter a value for the “Speed unit” selected. Example 1: For a PIC24 MCU and a target clock oscillator at 32 MHz (HS), instruction speed = 32 MHz/2 = 16 MIPS. Example 2: For a PIC18F8722 MCU and a target clock oscillator at 10 MHz (HS) making use of the PLL (x4 = 40 MHz), instruction speed = 40 MHz/4 = 10 MIPS.
Speed unit	Select either: KIPS – Thousands (10^3) of instructions per second MIPS – Millions (10^6) of instructions per second
Debug Mode Clock	
Use FRC in Debug mode	Select the checkbox if you want to use the fast internal oscillator on the device. If selected any non-frozen peripherals will operate at the FRC during debugging (see Section 9.5.3 “Settings Dialog, Freeze on Halt Tab”).

9.5.6 [Settings] ダイアログの [Secure Segment] タブ

CodeGuard™ セキュリティ対応デバイスは、このタブでセキュア セグメントのプロパティを設定します。

CodeGuard セキュリティ機能の詳細は、マイクロチップ社ウェブ サイトで 16 ビット デバイス向け CodeGuard セキュリティ リファレンス マニュアル (DS70180)、dsPIC33F/PIC24H および dsPIC30F デバイス プログラミング 仕様を参照してください。

表 9-10: [SECURE SEGMENT] タブのオプション

Full Chip Programming	Click to select to program all program memory segments.
Segment Programming	Click to select segment programming. Select from: - Boot, secure and general segments - Secure and general segments - General segment only

9.5.7 [Settings] ダイアログの [Warning] タブ

このタブには、MPLAB ICD 3 インサーキット デバッガのすべての警告が一覧表示されます。

- 警告の項目にチェックを入れると、その警告が有効になります。警告は [Output] ウィンドウに表示されます。
- 警告の項目のチェックを外すと、その警告は無効になります。

警告はエラーではないので、発生してもプロジェクトのビルドは中止されません。エラーメッセージは、第 7 章「エラーメッセージ」を参照してください。

9.5.8 [Settings] ダイアログの [Power] タブ

このタブでは、電源オプションを設定します。

ターゲット回路の電源を MPLAB ICD 3 から供給する場合は [Power target circuit from MPLAB ICD 3] にチェックを入れます。

スライドバーを調整して電圧を設定します。スライドバーを動かすと、フィールドの数値が変化します。

MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

第 10 章 ハードウェア仕様

10.1 はじめに

ここでは、MPLAB ICD 3 インサーキット デバッガ システムのハードウェアおよび電氣的仕様について詳しく説明します。

10.2 ハイライト

この章の内容は次のとおりです。

- 適合宣言書
- USB ポート / 電源
- MPLAB ICD 3 デバッガ
- 標準通信ハードウェア
- ICD 3 テスト インターフェース ボード
- ターゲット ボードに関する注意事項

10.3 適合宣言書

当社

Microchip Technology Inc.
2355 W. Chandler Blvd.
Chandler, Arizona 85224-6199
USA

は、以下の製品

MPLAB® ICD 3 インサーキット デバッガ

が、操作マニュアルに記載された制限事項を遵守した場合に以下の規格に適合することを宣言します。

規格 : EN61010-1 試験所用電気機器
Microchip Technology, Inc.

日付 : 2006 年 8 月

MPLAB ICD 3 インサーキット デバッガの使用に関する重要な注意事項

MPLAB ICD 3 インサーキット デバッガは、その特殊な性質により、通常より高いレベルの電磁放射を発生して無線機器などあらゆる種類の機器の動作に影響を与えることがあるので注意してください。

したがって、上記欧州規格に適合するためには、次の制限事項を遵守する必要があります。

1. 本開発システムの使用は工業地域 (または同等地域) のみに限定すること。
2. 電磁波の影響を受ける可能性のある機器 (無線受信機やテレビなど) から 20 メートル以上離れた場所で本システムを使用すること。

10.4 USB ポート / 電源

MPLAB ICD 3 インサーキット デバッガは、ホスト PC の USB (Universal Serial Bus) 2.0 ポートに接続します。USB コネクタはポッドの側面にあります。

システムは USB インターフェースを利用してファームウェアを再ロードできます。

システムの電源は USB インターフェースから供給されます。このデバッガは USB 仕様のハイパワーシステムに分類され、デバッガモードとプログラマモードのいずれの動作においても USB から 300 mA の電力供給を受ける必要があります。

注： MPLAB ICD 3 インサーキット デバッガの電源は USB 接続から供給されます。ターゲットボードの電源は自己給電とします。ただし、ターゲットの消費電流が 100 mA 未満の場合に限り、MPLAB ICD 3 から電源を供給することもできます。

ケーブルの長さ — PC とデバッガの接続ケーブルは、正常な動作に適した長さのものがデバッガキットに同梱されています。

電源付きハブ — USB ハブを使用する場合は、セルフパワー型のものを使用してください。PC のキーボードに付いている USB ポートではデバッガの動作に十分な電力が得られません。

PC の省電力モード (休止など) — PC とデバッガの USB 通信が正常に行えなくなるため、休止などの省電力モードは使用しないでください。

10.5 MPLAB ICD 3 デバッガ

デバッガにはメインボードが内蔵されており、本体には USB コネクタと RJ-11 コネクタが付いています。デバッガ本体にはインジケータライト (LED) もあります。

10.5.1 メインボード

メインボードには、インターフェースプロセッサ (dsPIC DSC)、480 Mbps 対応の USB 2.0 インターフェース、システム全体の制御および高速通信用の FPGA (Field Programmable Gate Array) オンボードフラッシュにエミュレーションデバイスへ書き込むプログラムコードイメージを保持するための SRAM、LED インジケータが実装されています。

10.5.2 インジケータ LED

インジケータ LED の状態と意味は次のとおりです。

LED	Color	Description
Active	Blue	Lit when power is first applied or when target is connected.
Status	Green	Lit when the debugger is operating normally – standby.
	Red	Lit when an operation has failed.
	Orange	Lit when the debugger is busy.

10.6 標準通信ハードウェア

デバッグとターゲットの標準通信 (2.4 項「デバッグとターゲットの通信」の「ICSP によるデバイスとの標準通信」) には、RJ-11 コネクタを使用します。

ヘッダ ボード使用時に標準通信を行うには、デバイス専用のプロセッサ パック (-ICE/ICD デバイスを装着した 8 ピン コネクタのヘッダ ボードと標準アダプタ ボードのセット) が必要になることがあります。

注: 旧式のヘッダ ボードで、8 ピン コネクタではなく 6 ピン (RJ-11) コネクタを採用しているものはデバッグにそのまま接続できます。

利用可能なヘッダ ボードの詳細は、『*Header Board Specification*』 (DS51292) を参照してください。

10.6.1 標準通信

標準通信は、ターゲット プロセッサとのメイン インターフェースとなるものです。この中には、ターゲット デバイスへの接続およびプログラミングに必要な高電圧 (VPP)、VDD 検出ライン、クロックおよびデータ接続が含まれます。

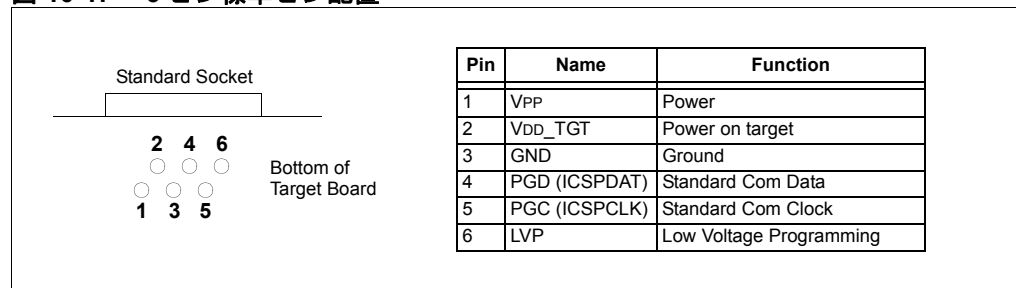
VPP の高電圧ラインは、個々のエミュレーション プロセッサの電圧要件に応じて 0V ~ 14V の振幅の可変電圧を生成できます。

VDD 検出ラインには、ターゲット プロセッサからごく微量の電流が流れます。VDD 検出ラインはターゲットの電圧を追跡するためのリファレンスとして使用されるだけで、実際の電源は MPLAB ICD 3 インサーキット デバッグ システムから供給されます。VDD 接続はオプティカル スイッチで絶縁されています。

クロックおよびデータ接続のインターフェースには、次の特長があります。

- 高インピーダンス モードのクロックおよびデータ信号 (MPLAB ICD 3 インサーキット デバッグ システムに電源が供給されていない場合を含む)
- ターゲット システムの不良または不適切な接続による過電圧からクロックおよびデータ信号を保護
- ターゲット システムの不良箇所での電氣的短絡による過電流からクロックおよびデータ信号を保護

図 10-1: 6 ピン標準ピン配置



10.6.2 モジュラ ケーブルとコネクタ

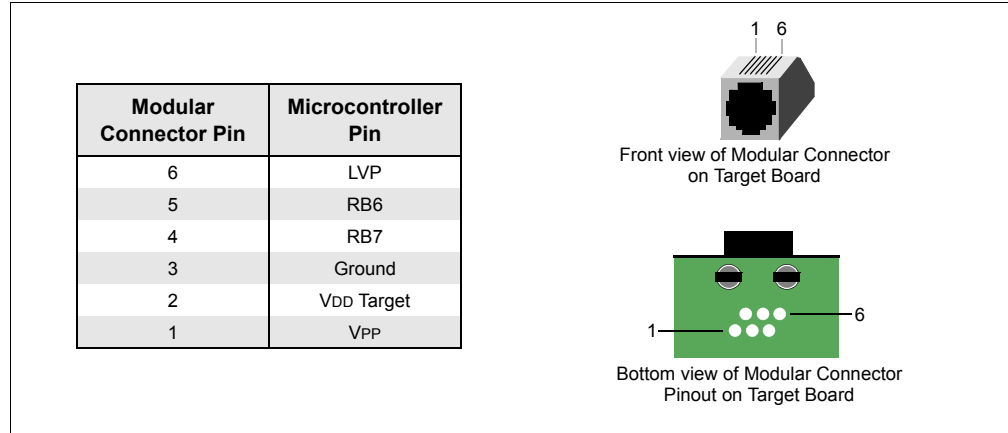
標準通信では、デバッガとターゲットアプリケーションをモジュラ ケーブルで接続します。以下に、このケーブルとコネクタの仕様を示します。

10.6.2.1 モジュラ コネクタの仕様

- メーカー名、製品番号 — AMP Incorporated、555165-1
- 販売代理店名、製品番号 — Digi-Key、A9031ND

アプリケーション側のモジュラ コネクタとマイクロコントローラのピンの対応を、次の表に示します。この構成では、ICD の機能をすべて利用できます。

図 10-2: ターゲット ボードのモジュラ コネクタのピン配置

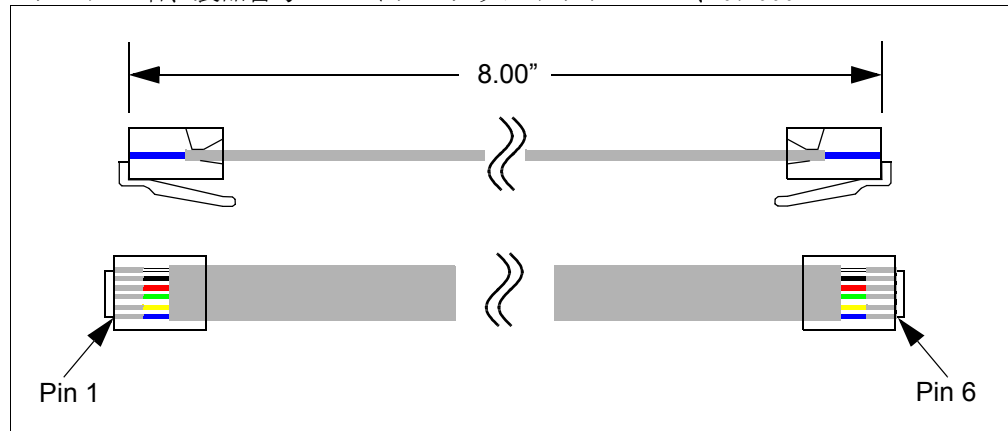


10.6.2.2 モジュラ プラグの仕様

- メーカー名、製品番号 — AMP Incorporated、5-554710-3
- 販売代理店名、製品番号 — Digi-Key、A9117ND

10.6.2.3 モジュラ ケーブルの仕様

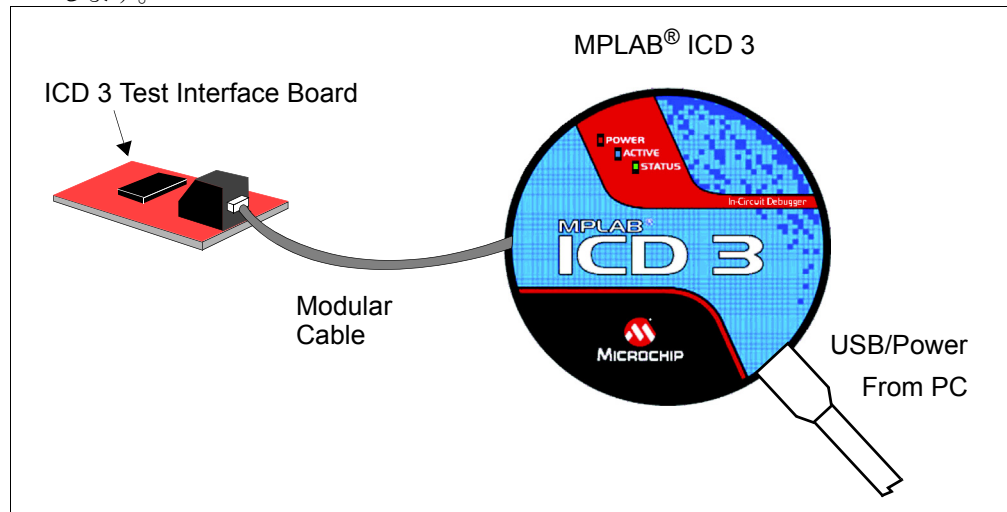
- メーカー名、製品番号 — マイクロチップ テクノロジー、07-00024



10.7 ICD 3 テスト インターフェース ボード

このボードは、デバッガが正しく動作していることを確認するために使用します。使用法は次のとおりです。

1. デバッガのケーブルを抜いて、ターゲットと PC からデバッガを切り離します。
2. デバッガと ICD 3 テスト インターフェース ボードをモジュラ ケーブルで接続します。



3. デバッガと PC を接続します。
4. MPLAB IDE で、デバッガまたはプログラムのいずれかに MPLAB ICD 3 インサーキット デバッガを指定します。
5. すると MPLAB IDE が完全なセルフテストを実行し、そのステータス (合格 / 失敗) が表示されます。

10.8 ターゲット ボードに関する注意事項

ターゲット ボードには、選択したデバイス (2.0V ~ 5.5V) およびアプリケーションの要件に合わせて電源を供給してください。

デバッガはターゲットの電源を検出します。VDD_TGT には 10 KΩ の負荷があります。

デバッガとターゲットの通信の方法によっては、ターゲット ボードの回路に関していくつかの注意事項があります。詳しくは次の項を参照してください。

- 2.5.2 項「ターゲットとの接続回路」
- 2.5.5 項「デバッガの正常動作を妨げる回路」

MPLAB[®] ICD 3 インサーキット デバッガ ユーザーズ ガイド

ノート:

用語集

1 対 1 のプロジェクト/ワークスペース モデル

1 つのワークスペースで 1 つのプロジェクトを扱う。MPLAB IDE におけるアプリケーション開発の最も一般的な構成。[\[Configure\]](#) > [\[Settings\]](#) の順にクリックして **[Projects]** タブで **[Use one-to-one project-workspace model]** にチェックを入れて指定する。

ANSI

American National Standards Institute (米国規格協会) の略。米国における標準規格の策定と承認を行う団体。

ASCII

American Standard Code for Information Interchange の略。7 桁の 2 進数で 1 つの文字を表現する文字セット エンコード方式。大文字、小文字、数字、記号、制御文字などが含まれる。

C

簡潔な表現、現代的な制御フローとデータ構造、豊富に用意された演算子などを特長とする汎用プログラミング言語。

Clean

MPLAB IDE の **[Project]** メニューの項目。アクティブなプロジェクトのオブジェクトファイル、Hex ファイル、デバッグ ファイルなどすべての中間ファイルが削除される。これらのファイルは、プロジェクトのビルド時に他のファイルから再構築される。

COFF

Common Object File Format の略。このフォーマットのオブジェクトファイルには、マシンコードのほか、デバッグなどに関する情報が含まれる。

DSC

「デジタル シグナル コントローラ」を参照。

EEPROM

Electrically Erasable Programmable Read Only Memory の略。電氣的に消去可能なタイプの PROM。データの書き込みと消去はバイト単位で行われる。EEPROM の内容は電源をオフにしても保持される。

EPROM

Erasable Programmable Read Only Memory の略。再書き込みが行えるタイプの ROM で、消去は紫外線照射によって行うものが主流。

FNOP

Forced No Operation の略。Forced NOP サイクルは、2 サイクル命令の 2 サイクル目で発生する。PIC マイクロコントローラのアーキテクチャはパイプライン構造となっており、現在の命令を実行中に物理アドレス領域の次の命令がプリフェッチされる。しかし、現在の命令によってプログラム カウンタが変化した場合、プリフェッチした命令は明示的に無視され、Forced NOP サイクルが発生する。

GPR

General Purpose Register (汎用レジスタ) の略。デバイスのデータ メモリ (RAM) のうち、汎用目的に使用できる部分。

Halt

プログラム実行を停止すること。Halt を実行することは、ブレークポイントで停止することと同じ。

Hex コード

実行可能な命令を 16 進数形式のコードで保存したもの。Hex コードは Hex ファイルに保存される。

Hex ファイル

デバイスに書き込み可能な 16 進数形式のアドレスと値 (Hex コード) を記述した ASCII ファイル。

ICD

In-Circuit Debugger (インサーキット デバッガ) の略。マイクロチップ社の MPLAB ICD 2 など。

ICE

In-Circuit Emulator (インサーキット エミュレータ) の略。マイクロチップ社の ICE には MPLAB ICE 2000、MPLAB ICE 4000、MPLAB ICD 3 システムなどがある。

ICSP

In-Circuit Serial Programming (インサーキット シリアルプログラミング) の略。マイクロチップ社製の組み込みデバイスをシリアル通信を利用して最小限のデバイスピンでプログラミングする方法。

IDE

Integrated Development Environment (統合開発環境) の略。マイクロチップ社の MPLAB IDE など。

IRQ

「割り込み要求」を参照。

ISO

「国際標準化機構 (International Organization for Standardization)」を参照。

ISR

「割り込みサービスルーチン」を参照。

Makefile

プロジェクトの Make に関する指示をファイルにエクスポートしたもの。このファイルは、MPLAB IDE 以外の環境で make コマンドを実行してプロジェクトをビルドする際に使用する。

Makefile をエクスポートするには、*[Project] > [Build Options] > [Project]* の順にクリックして、**[Directories]** タブの *[Build Directory Policy]* で *[Assemble/Compile/Link in the project directory]* を選択しておく必要がある。

Make Project

アプリケーションを再ビルドするコマンド。前回の完全なコンパイル後に変更されたソースファイルのみを再コンパイルする。

MCU

Microcontroller Unit の略。マイクロコントローラのこと。μC と表記されることもある。

MPASM™ アセンブラ

PIC マイクロコントローラ デバイス、KeeLoq® デバイス、マイクロチップ社のメモリデバイスに対応したマイクロチップ社の再配置可能なマクロアセンブラ。

MPLAB ASM30

dsPIC デジタル シグナル コントローラ および PIC24 PIC デバイスに対応したマイクロチップ社の再配置可能なマクロ アセンブラ。

MPLAB C17

PIC17 MCU デバイスの生産終了に伴い、提供を中止。「MPLAB C18」を参照。

マイクロチップ社の MPLAB C17 C コンパイラのこと。PIC17 MCU デバイスに対応した C コンパイラ。

MPLAB C18

マイクロチップ社の MPLAB C18 C コンパイラのこと。PIC18 MCU デバイスに対応した C コンパイラ。

MPLAB C30

dsPIC デジタル シグナル コントローラ および PIC24 PIC デバイスに対応したマイクロチップ社の C コンパイラ。

MPLAB ICD 2

MPLAB IDE と組み合わせて使用するマイクロチップ社のインサーキット デバッガ。ICD では、デバッグ回路を内蔵したフラッシュ デバイスがサポートされる。ICD の中心的なコンポーネントはポッドと呼ばれる。ポッド、ヘッダ ボード(「デバイス名-ICD」を搭載)、ターゲット ボード、ケーブル、MPLAB IDE ソフトウェアを組み合わせて完全なシステムが構成される。

MPLAB ICD 3 インサーキット デバッガ

MPLAB IDE と組み合わせて使用するマイクロチップ社のインサーキット デバッガ。MPLAB ICD デバッガでは PIC18F と PIC24 MCU、および dsPIC DSC がサポートされる。ICE の中心的なコンポーネントはポッドと呼ばれる。ポッド、ドライバ(およびレシーバ)カード、ケーブル、MPLAB IDE ソフトウェアを組み合わせて完全なシステムが構成される。

MPLAB ICE 2000

MPLAB IDE と組み合わせて使用するマイクロチップ社のインサーキット デバッガ。MPLAB ICE 2000 では 8 ビット PIC MCU がサポートされる。ICE の中心的なコンポーネントはポッドと呼ばれる。ポッド、プロセッサ モジュール、ケーブル、MPLAB IDE ソフトウェアを組み合わせて完全なシステムが構成される。

MPLAB ICE 4000

新規デザインには推奨しません。「MPLAB ICD 3 インサーキット デバッガ」を参照してください。

MPLAB IDE と組み合わせて使用するマイクロチップ社のインサーキット デバッガ。MPLAB ICE 4000 では PIC18F と PIC24 MCU、および dsPIC DSC がサポートされる。ICE の中心的なコンポーネントはポッドと呼ばれる。ポッド、プロセッサ モジュール、ケーブル、MPLAB IDE ソフトウェアを組み合わせて完全なシステムが構成される。

MPLAB IDE

マイクロチップ社の統合開発環境。

MPLAB LIB30

MPLAB LIB30アーカイバ/ライブラリアンは、MPLAB ASM30またはMPLAB C30 C コンパイラで作成したオブジェクト モジュールに使用するオブジェクト ライブラリアン。

MPLAB LINK30

マイクロチップ社の MPLAB ASM30 アセンブラおよび MPLAB C30 C コンパイラに対応したオブジェクト リンカ。

MPLAB PM3

マイクロチップ社提供のデバイス プログラマ。PIC18 マイクロコントローラおよび dsPIC デジタル シグナル コントローラ の書き込みに対応。MPLAB IDE との併用も、単体での使用も可能。PRO MATE II の後継。

MPLAB PM3 Environment

デバイスのプログラミングに関する設定ファイルを保存したフォルダ。このフォルダを SD/MMC カードに転送できる。

MPLAB SIM

MPLAB IDE と組み合わせて使用するマイクロチップ社のシミュレータで、PIC MCU および dsPIC DSC デバイスに対応する。

MPLIB™ オブジェクト ライブラリアン

MPLAB IDE と組み合わせて使用するマイクロチップ社のライブラリアン。MPLIB ライブラリアンは、MPASM アセンブラ (mpasm または mpasmwin v2.0) または MPLAB C17/C18 C コンパイラで作成した COFF オブジェクト モジュールに使用するオブジェクト ライブラリアン。

MPLINK™ オブジェクト リンカ

マイクロチップ社の MPASM アセンブラ および MPLAB C17/C18 C コンパイラに対応したオブジェクト リンカ。マイクロチップ社の MPLIB ライブラリアンとの併用も可能。MPLAB IDE に統合して使用できるように設計されているが、MPLAB IDE 以外の環境でも使用できる。

MRU

Most Recently Used の略。最近使用したファイルおよびウィンドウのこと。MPLAB IDE のメインメニューで選択できる。

NOP

No Operation の略。実行してもプログラム カウンタが進むだけで何も動作を行わない命令。

OTP

One Time Programmable の略。パッケージに窓のない EPROM デバイス。EPROM を消去するには紫外線照射が必要なため、パッケージに窓のあるデバイスしか消去できない。

PC

パーソナル コンピュータ または プログラム カウンタ の略。

PIC MCU

マイクロチップ社のすべてのマイクロコントローラ ファミリの総称。

PICSTART Plus

マイクロチップ社提供の開発用デバイス プログラマ。8 ピン、14 ピン、28 ピン 40 ピンの PIC マイクロコントローラ のプログラミングに対応。書き込みには必ず MPLAB IDE ソフトウェアを使用する。

PRO MATE II

生産終了品です。「MPLAB PM3」を参照してください。

マイクロチップ社提供のデバイス プログラマ。ほとんどの PIC マイクロコントローラ、メモリ、KEELOQ デバイスへの書き込みに対応。MPLAB IDE との併用も、単体での使用も可能。

PWM 信号

パルス幅変調 (Pulse-Width Modulation) 信号。一部の PIC MCU には周辺モジュールとして PWM が内蔵されている。

RAM

Random Access Memory の略。データメモリ。任意の順にメモリ内の情報にアクセスできる。

ROM

Read Only Memory の略。プログラムメモリ。メモリの内容を変更できない。

Run

デバッガを Halt から解放するコマンド。デバッガはアプリケーションコードを実行し、I/O に対してリアルタイムに変更、応答を行う。

SFR

「特殊機能レジスタ」を参照。

Single Step

コードを 1 命令ずつ実行するコマンド。1 命令を実行するたびに、MPLAB IDE のレジスタウィンドウ、ウォッチ変数、ステータスディスプレイの表示が更新されるので、命令実行を解析してデバッグできる。C コンパイラのソースコードもシングルステップ実行できるが、その場合は 1 命令ずつ実行されるのではなく、高級言語の C で記述されたコードの 1 行から生成されるすべてのアセンブリレベル命令がシングルステップで実行される。

Step Into

Single Step と同じコマンド。Step Over とは異なり、Step Into では CALL 命令によって呼び出されるサブルーチンもステップ実行される。

Step Out

現在ステップ実行中のサブルーチンから抜け出すためのコマンド。このコマンドを実行すると、サブルーチンの残りのコードがすべて実行され、サブルーチンのリターンアドレスで実行が停止する。

Step Over

Step Over を実行すると、サブルーチン内はステップ実行せずにデバッグできる。Step Over では、CALL 命令があると CALL の次の命令にブレークポイントが設定される。何らかの理由により、サブルーチンが無限ループになるなど正しくリターンしない場合は、次のブレークポイントには到達しない。CALL 命令の処理以外は、Step Over コマンドと Single Step コマンドは同じ。

USB

Universal Serial Bus の略。2本のシリアル伝送線でPCと外部周辺機器の通信を行う外部周辺インターフェース規格。USB 1.0/1.1 でサポートされるデータ転送レートは 12 Mbps。USB 2.0 (Hi-Speed USB) は最大 480 Mbps のデータレートをサポートしている。

[Watch] ウィンドウ

ウォッチ変数の一覧が表示され、ブレークポイントで毎回表示が更新されるウィンドウ。

WDT

「ウォッチドッグタイマ」を参照。

アーカイバ

ライブラリを作成、操作するためのツール。

アーカイブ

再配置可能なオブジェクトモジュールを集めたもの。複数のソースファイルオブジェクトファイルにアセンブルした後、アーカイバを使用してこれらオブジェクトファイルを 1 つのライブラリファイルにまとめると生成される。ライブラリをオブジェクトモジュールや他のライブラリとリンクすると、実行コードが生成される。

アクセス メモリ (PIC18 のみ)

PIC18 デバイスにおいて、バンク選択レジスタ (BSR) の設定にかかわらずアクセスできる特別なレジスタ。

アセンブラ

アセンブリ言語のソース コードをマシン コードに変換する言語ツール。

アセンブリ言語

2 進数のマシン コードをシンボル表現で記述したプログラミング言語。

アップロード

デバッガやプログラマなどのツールからホスト PC へ、またはターゲット ボードからデバッガへデータを転送すること。

アドレス

メモリ内の位置を一意に特定する値。

アプリケーション

PIC マイクロコントローラで制御されるソフトウェアとハードウェアを組み合わせたもの。

アルファベット文字

アルファベットの小文字と大文字の総称 (a, b, ..., z, A, B, ..., Z)。

イベント

アドレス、データ、バス カウント、外部入力、サイクルタイプ (フェッチ、R/W)、タイム スタンプなど、バス サイクルを記述したもの。トリガ、ブレイクポイント、割り込みを記述するために使用する。

入れ子の深さ

マクロに他のマクロを含めることのできる階層の数。

インポート

Hex ファイルなどの外部ソースから MPLAB IDE にデータを取り込むこと。

ウォッチドッグ タイマ

PIC マイクロコントローラに内蔵されたタイマの 1 つで、ユーザーが設定した期間が経過するとプロセッサをリセットする。WDT の有効化 / 無効化、および設定はコンフィギュレーション ビットで行う。

ウォッチ変数

デバッグセッション中に [Watch] ウィンドウで観察できる変数。

英数字

アルファベット文字と 0 ~ 9 の 10 進数の数字の総称。

エクスポート

MPLAB IDE のデータを標準フォーマットで外部に出力すること。

エミュレーション

エミュレーション メモリにロードされたソフトウェアを、あたかもマイクロコントローラにロードされたファームウェアのように実行すること。

エミュレーション メモリ

デバッガに内蔵されたプログラム メモリ。

演算子

定義可能な式を構成する際に使用される「+」や「-」などの記号。各演算子に割り当てられた優先順位に基づいて式が評価される。

オフチップメモリ

PIC17 または PIC18 デバイスで選択できるメモリ オプション。ターゲットボード上のメモリを使用するか、またはすべてのプログラムメモリをデバッガから供給する。
[\[Options\]](#) > [\[Development Mode\]](#) の順にクリックして **[Memory]** タブでオフチップメモリの選択を行う。

オブジェクトコード

アセンブラまたはコンパイラによって生成されるマシンコード。

オブジェクトファイル

マシンコードを含むファイル。デバッグ情報を含むこともある。そのまま実行できるものと、他のオブジェクトファイル(ライブラリなど)とリンクしてから完全な実行プログラムを生成する再配置可能形式のものがある。

オブジェクトファイルディレクティブ

オブジェクトファイル作成時にのみ使用するディレクティブ。

オペコード

Operational Code の略。「ニーモニック」を参照。

拡張マイクロコントローラモード

拡張マイクロコントローラモードでは、オンチップのプログラムメモリと外部メモリの両方が利用できる。プログラムメモリのアドレスが PIC17 または PIC18 デバイスの内部メモリ領域より大きくなると、自動的に外部メモリの実行に切り替わる。

環境 — IDE

アプリケーション開発に特化したデスクトップ環境。

外部RAM

オフチップの読み書き可能なメモリ。

外部シンボル

外部リンケージを持つ識別子のシンボル。参照の場合と定義の場合がある。

外部シンボル解決

リンカがすべての入力モジュールの外部シンボル定義を 1 つにまとめ、すべての外部シンボル参照を解決しようとするプロセス。外部シンボル参照に対応する定義が存在しない場合は、リンカエラーとなる。

外部入力ライン

外部信号に基づいてイベントを設定するための外部入力信号ロジックプローブライン (TRIGIN)。

外部ラベル

外部リンケージを持つラベル。

外部リンケージ

関数や変数が、それ自身が定義されたモジュールの外部から参照できる場合、外部リンケージを持つという。

基数

アドレスを指定する際の記数法 (16 進法、10 進法) の底。

警告

デバイス、ソフトウェアファイル、装置に物理的な損傷を与える可能性のある状況で、ユーザーに注意を促すために表示されるメッセージ。

高級言語

プログラムを記述するための言語で、プロセッサから見てアセンブリよりも遠い位置関係にあるもの。

校正メモリ

PIC マイクロコントローラのオンボード RC オシレータやその他の周辺モジュールの校正値を格納するための特殊機能レジスタ。

国際標準化機構 (International Organization for Standardization)

コンピューティングや通信をはじめ、多くのテクノロジーおよびビジネス関連の標準規格の策定を行っている団体。

コマンドライン インターフェース

プログラムとユーザーのやり取りをテキストの入出力だけで行う方法。

コンパイラ

高級言語で記述されたソース ファイルをマシン コードに変換するプログラム。

コンフィギュレーション ビット

PIC マイクロコントローラの動作モードを設定するために書き込む専用ビット。コンフィギュレーション ビットは事前プログラミングされている場合とされていない場合がある。

再帰

定義した関数やマクロがそれ自身を呼び出すこと。再帰マクロを作成する際は、再帰から抜けずに無限ループとなりやすいので注意が必要。

シェル

MPASM アセンブラにおいて、マクロ アセンブラへの入力を行うためのプロンプト インターフェース。MPASM アセンブラには DOS 用シェルと Windows 用シェルの 2 種類がある。

システム ウィンドウ コントロール

ウィンドウや一部のダイアログの左上隅にあるコントロール。通常、このコントロールをクリックすると、[最小化]、[最大化]、[閉じる] などのメニュー項目がポップアップ表示される。

シナリオ

MPLAB SIM シミュレータにおいて、スティミュラス制御を具体的に設定したもの。

シミュレータ

デバイスの動作をモデリングするソフトウェア プログラム。

修飾子

パス カウンタで使用する、または複合トリガにおける次の動作前のイベントとして使用するアドレスまたはアドレス範囲。

シンボル

プログラムを構成するさまざまな要素を記述する汎用のメカニズム。これら要素には、関数名、変数名、セクション名、ファイル名、struct/enum/union タグ名などがある。MPLAB IDE では、主に変数名、関数名、アセンブリ ラベルをシンボルと呼ぶ。リンク実行後は、シンボルの値はメモリ内の値となる。

スキッド

ハードウェア ブレークポイントを使用してプロセッサを停止する場合、ブレークポイントから更に 1 つ以上の命令を実行してプロセッサが停止することがある。ブレークポイントの後に実行される命令の数をスキッドと呼ぶ。

スキュー

命令実行に関する情報は、さまざまなタイミングでプロセッサ バスに現れる。例えば、実行されるオペコードは直前の命令の実行時にフェッチとしてバスに現れる。ソース データのアドレスと値、およびデスティネーションデータのアドレスは、オペコードが実際に実行されるときにバスに現れる。デスティネーションデータの値

は次の命令の実行時にバスに現れる。トレース バッファには、1 インスタンスでバス上に存在する情報がキャプチャされる。したがって、トレース バッファの 1 エントリには 3 つの命令の実行情報が含まれる。1 つの命令実行である情報から次の情報までにキャプチャされるサイクル数をスキューと呼ぶ。

スタック (ソフトウェア スタック)

アプリケーションがリターンアドレス、関数パラメータ、ローカル変数を保存するのに使用するメモリ。高級言語でコードを開発する場合、このメモリは通常コンパイラによって管理される。

スタック (ハードウェア スタック)

PIC マイクロコントローラで関数呼び出しを行うときにリターンアドレスを格納する場所。

スタティック RAM (SRAM)

Static Random Access Memory の略。ターゲット ボード上の読み書き可能なプログラムメモリ。頻繁に書き換える必要のないプログラムを書き込む。

ステータス バー

MPLAB IDE ウィンドウの一番下にあるバーで、カーソル位置、開発モードおよびデバイス、アクティブなツールバーなどに関する情報が表示される。

スティミュラス

シミュレータへの入力。すなわち、外部信号に対する応答をシミュレートするために生成されるデータ。通常、このデータはテキスト ファイルにアクションのリストとして記述される。スティミュラスの種類には、非同期、同期 (ピン)、クロック動作、レジスタがある。

ストップウォッチ

実行サイクルを測定するためのカウンタ。

制御ディレクティブ

アセンブリ言語コード内で使用するディレクティブで、指定した式のアセンブル時の値に基づいてコードを含めるか除外するかを決定する。

絶対セクション

リンカによって変更されない固定 (絶対) アドレスを持つセクション。

ソース コード

プログラマが記述したコンピュータ プログラム。プログラミング言語で記述されたソース コードは、マシン コードに変換して実行するか、またはインタープリタによって実行される。

ソース ファイル

ソース コードを記述した ASCII テキスト ファイル。

相互参照ファイル

シンボルテーブルおよびそのシンボルを参照するファイル リストを参照するファイル。シンボルが定義されている場合は、リストの最初のファイルがシンボル定義の位置となる。残りのファイルにはシンボルへの参照が含まれる。

ターゲット

ユーザー ハードウェアのこと。

ターゲット アプリケーション

ターゲット ボードにロードしたソフトウェア。

ターゲット プロセッサ

ターゲット アプリケーション ボードで使用されているマイクロコントローラ デバイス。

ターゲット ボード

ターゲット アプリケーションを構成する回路とプログラマブルなデバイス。

ダウンロード

ホストから別のデバイス (デバッガ、プログラマ、ターゲット ボードなど) にデータを送信すること。

ツールバー

MPLAB IDE の機能を実行するためのボタン (アイコン) を縦または横に並べたもの。

テンプレート

後でファイルに挿入するために作成するテキスト行。MPLAB Editor では、テンプレートはテンプレート ファイルに保存される。

データ ディレクティブ

アセンブラによって行われるプログラム メモリまたはデータ メモリの割り当てを制御するディレクティブ。データ項目をシンボル (意味のある名前) を使って参照する手段としても使われる。

データ メモリ

マイクロチップ社の MCU および DSC デバイスでは、データ メモリ (RAM) は汎用レジスタ (GPR) と特殊機能レジスタ (SFR) で構成される。EEPROM データ メモリを内蔵したデバイスもある。

ディレクティブ

言語ツールの動作を制御するためにソース コードに記述するステートメント。

デジタル シグナル コントローラ

デジタル信号処理機能を搭載したマイクロコントローラ。マイクロチップ社の dsPIC DSC デバイスなど。

デバイス プログラマ

マイクロコントローラなど、電氣的に書き込み可能な半導体デバイスにプログラミングを行うためのツール。

デバッガ

エミュレーションを実行するハードウェア。

デバッガ システム

MPLAB ICE 2000 および MPLAB ICE 4000 の場合、デバッガ システムはポッド、プロセッサ モジュール、デバイス アダプタ、ターゲット ボード、ケーブル、MPLAB IDE ソフトウェアで構成される。MPLAB ICD 3 の場合、デバッガ システムはポッド、ドライバ (およびレシーバ) カード、ターゲット ボード、ケーブル、MPLAB IDE ソフトウェアで構成される。

デバッグ情報

コンパイラおよびアセンブラでこのオプションを選択すると、アプリケーション コードのデバッグに使用できるさまざまなレベルの情報が出力される。デバッグ オプションの選択の詳細はコンパイラまたはアセンブラのマニュアルを参照。

特殊機能レジスタ

I/O プロセッサ機能、I/O ステータス、タイマなど、さまざまなモードや周辺モジュールを制御するレジスタ専用使用するデータ メモリ (RAM) 領域。

トリガ出力

任意のアドレスまたはアドレス範囲で生成でき、トレースおよびブレイクポイント設定から独立したデバッガ出力信号のこと。トリガ出力ポイントはいくつでも設定できる。

トレース

プログラム実行のログを記録するデバッガまたはシミュレータの機能。デバッガはプログラム実行のログをトレース バッファに記録し、これを MPLAB IDE のトレース ウィンドウにアップロードする。

トレース メモリ

デバッガに内蔵されたトレース用のメモリ。トレース バッファとも呼ばれる。

内部リンク

関数や変数が、それ自身が定義されたモジュールの外部からアクセスできない場合、内部リンケージを持つという。

ニーモニック

マシン コードと 1 対 1 で対応したテキスト命令。オペコードとも呼ばれる。

ノード

MPLAB IDE のプロジェクトを構成するコンポーネント。

パス カウンタ

イベント(特定のアドレスの命令を実行するなど)が発生するたびに値が減少するカウンタ。パス カウンタの値がゼロになると、イベントの条件が満たされる。パス カウンタはブレイクログジックやトレース ログジック、および複合トリガ ダイアログの任意のシーケンシャル イベントに割り当てられる。

パワーオン リセット エミュレーション

データ RAM 領域にランダムな値を書き込んで、初回電源投入時の RAM の非初期化値をシミュレートするソフトウェア ランダム化処理。

非初期化データ

初期値なしで定義されたデータ。C では、

```
int myVar;
```

として定義した変数は非初期化データ セクションに格納される。

非同期ステイミュラス

シミュレータ デバイスへの外部入力をシミュレートするために生成されるデータ。

非リアルタイム

ブレイクポイントで停止中、またはシングルステップ実行中のプロセッサ、あるいはシミュレータ モードで動作中の MPLAB IDE を指す。

ビルド

すべてのソース ファイルをコンパイルおよびリンクしてアプリケーションを作成すること。

ファイル レジスタ

オンチップのデータ メモリ。汎用レジスタ (GPR) と特殊機能レジスタ (SFR) を含む。

フィルタ

トレース ディスプレイまたはデータ ファイルにどのデータを含めるか/除外するかを選択するもの。

不揮発性ストレージ

電源をオフにしても内容が失われないストレージデバイス。

フラッシュ

データの書き込みと消去をバイト単位ではなくブロック単位で行えるタイプの EEPROM。

ブックマーク

ファイル内の特定の行に簡単な操作でアクセスできるようにする機能。

ブックマークの有効 / 無効の切り替え、前後のブックマークへの移動、すべてのブックマークのクリアなど、ブックマークの管理は [Edit] メニューで [Bookmarks] をクリックして行う。

ブレイクポイント (ソフトウェア ブレイクポイント)

ファームウェアの実行が停止するアドレス。通常、特別な Break 命令によって実行が停止される。

ブレイクポイント (ハードウェア ブレイクポイント)

実行するとファームウェアの実行が停止するイベント。

プラグイン

MPLAB IDE では、標準コンポーネントにプラグイン モジュールを追加することによって、多様なソフトウェア / ハードウェア ツールに対応する。一部のプラグイン ツールは、[Tools] メニューから利用できる。

プログラム カウンタ

現在実行中の命令のアドレスを格納した場所。

プログラム メモリ

デバイス内で命令が保存されるメモリ領域。また、デバッガまたはシミュレータにダウンロードしたターゲット アプリケーションのファームウェアを格納するメモリ領域もプログラム メモリと呼ばれる。

プロジェクト

アプリケーションのビルドに必要なファイル (ソース コードやリンカ スクリプト ファイルなど) 一式と、各種ビルド ツールやビルド オプションとの関連付けをまとめたもの。

プロトタイプ システム

ユーザーのターゲット アプリケーションまたはターゲット ボードのこと。

プロファイル

MPLAB SIM シミュレータにおいて、実行したステイミュラスをレジスタ別に一覧表示したものの。

ホスト PC

サポートされた Windows オペレーティング システムが動作する PC。

ポッド

MPLAB ICD 3 システムの場合 : ヘッド ボードまたはターゲット ボードに装着された ICE デバイス用のエミュレーション制御回路を内蔵。ICE デバイスとは、ICE 回路を内蔵した標準デバイス、または標準デバイスの特別な ICE バージョン (デバイス名 -ICE) を指す。

MPLAB ICD 2 の場合 : ヘッド ボードまたはターゲット ボードに装着された ICD デバイス用のデバッグ制御回路を内蔵。ICD デバイスとは、ICD 回路を内蔵した標準デバイス、または標準デバイスの特別な ICD バージョン (デバイス名 -ICD) を指す。

MPLAB ICE 2000/4000 の場合 : エミュレーション メモリ、トレース メモリ、イベントおよびサイクル タイマ、トレース / ブレイクポイント ロジックを含む外付けのハードウェア デバッガ。

マイクロコントローラ

CPU、RAM、プログラム メモリ、I/O ポート、タイマなど多くの機能が統合されたチップ。

マイクロコントローラ モード

PIC17およびPIC18マイクロコントローラで設定可能なプログラムメモリ構成の1つ。マイクロコントローラモードでは、内部実行のみが許可される。つまり、マイクロコントローラモードではオンチップのプログラムメモリしか利用できない。

マイクロプロセッサ モード

PIC17およびPIC18マイクロコントローラで設定可能なプログラムメモリ構成の1つ。マイクロプロセッサモードでは、オンチップのプログラムメモリは使用されない。プログラムメモリ全体が外部にマッピングされる。

マクロ

マクロ命令。一連の命令シーケンスを短い名前では表現した命令。

マクロ ディレクティブ

マクロ定義の中で実行とデータ割り当てを制御するディレクティブ。

マシンコード

コンピュータプログラムをプロセッサが実際に読み出して解釈できる形式で表現したもの。2進数のマシンコードで記述されたプログラムは、マシン命令のシーケンス(命令間にデータを挟むこともある)で構成される。ある特定のプロセッサで使用できるすべての命令の集合を「命令セット」という。

マシン語

あるCPUが翻訳を必要とせず実行できる命令の集合。

命令

CPUに対して特定の演算を実行するように指示するビット列。演算の対象となるデータを含めることもできる。

命令セット

特定のプロセッサが理解できるマシン語命令の集合。

メッセージ

言語ツールの動作に問題が発生したことを知らせる文字列。メッセージが表示されても処理は停止しない。

読み出し専用メモリ

恒久的に保存されているデータへの高速アクセスが可能なメモリハードウェア。ただし、データの追加や変更は不可。

ライブラリ

「アーカイバ」を参照。

ライブラリアン

「アーカイバ」を参照。

リアルタイム

インサーキットデバッガまたはデバッガがHalt状態から解放されると、プロセッサの実行はリアルタイムモードとなり、通常のチップとまったく同様に動作する。リアルタイムモードでは、デバッガのリアルタイムトレースバッファが有効になり、選択したすべてのサイクルを常時キャプチャする。また、すべてのブレイクロジックが有効になる。インサーキットデバッガまたはデバッガでは、有効なブレイクポイントで停止するか、またはユーザーによって実行が停止されるまでプロセッサはリアルタイムで動作する。

シミュレータでは、ホストCPUでシミュレート可能な最大速度でマイクロコントローラの命令を実行することをリアルタイムと呼ぶ。

リアルタイム ウォッチ

アプリケーション実行中に [Watch] ウィンドウで変数の値がリアルタイムに変化すること。リアルタイム ウォッチの設定方法は、各ツールのマニュアルを参照。リアルタイム ウォッチをサポートしていないツールもある。

リスティング ディレクティブ

アセンブラのリスティング ファイルのフォーマットを制御するディレクティブ。タイトルや改ページ指示など、リスティング ファイルに関するさまざまな設定を行う。

リスティング ファイル

ソース ファイルにある各 C ソース ステートメント、アセンブリ命令、アセンブラ ディレクティブ、マクロに対して生成されたマシン コードを記述した ASCII テキスト ファイル。

リンカ

オブジェクト ファイルとライブラリを結合し、モジュール間の参照を解決して実行形式のコードを生成する言語ツール。

リンカ スクリプト ファイル

リンカのコマンド ファイル。リンカのオプションを定義し、ターゲット プラットフォームで利用可能なメモリを記述する。

ロー (raw) データ

あるセクションに関連付けられたコードまたはデータを 2 進数で表現したもの。

ローカル ラベル

マクロ内で LOCAL ディレクティブで定義されたラベル。ローカル ラベルは、マクロの同一インスタンス内でのみ有効。すなわち、LOCAL として宣言されたシンボルやラベルには、ENDM マクロ以降はアクセスできない。

ロジック プローブ

マイクロチップ社製デバッガには、最大 14 のロジック プローブを接続できるものがある。ロジック プローブは、外部トレース入力、トリガ出力信号、+5V、共通グラウンドを提供する。

ワークスペース

MPLAB IDE において、デバイスやデバッグ ツール/プログラムの選択内容、起動中のウィンドウとその位置など、IDE の各種設定を保存したもの。

ワークブック

MPLAB SIM シミュレータにおいて、SCL スティミュラスの生成に関する設定を保存したもの。

割り込み

CPU に対する信号の一種。この信号が発生すると、現在動作中のアプリケーションの実行を一時停止し、制御を割り込みサービス ルーチン (ISR) に渡してイベントを処理する。ISR の実行が完了すると、通常のアプリケーションの実行が再開される。

割り込みサービス ルーチン

割り込みが発生すると実行されるユーザー作成コード。通常、発生した割り込みの種類によってプログラム メモリ内の異なる位置のコードが実行される。

割り込みハンドラ

割り込み発生時に専用のコードを実行するルーチン。

割り込み要求

プロセッサの通常の命令実行を一時的に停止し、割り込みハンドラ ルーチンの実行開始を要求するイベント。プロセッサによっては複数の割り込み要求イベントを持ち、優先順位の異なる割り込みを処理できるものもある。

索引

A		ICD3Err0014	54
[Abort Operation]	66	ICD3Err0015	54
[Animate]	65	ICD3Err0016	54
AVdd	17	ICD3Err0017	54
AVss	17	ICD3Err0018	54
B		ICD3Err0019	54
[Blank Check All]	73	ICD3Err0020	54
[Breakpoints]		ICD3Err0021	54
セットアップ	66	ICD3Err0022	54
ダイアログ	68	ICD3Err0023	54
ハードウェア	75	ICD3Err0024	54
有効化	67	ICD3Err0025	54
[Build Configuration]	20, 59	ICD3Err0026	55
C		ICD3Err0027	55
CD-ROM	10	ICD3Err0028	55
CodeGuard セキュリティ	77	ICD3Err0029	55
D		ICD3Err0030	55
[Debug Read]	66	ICD3Err0031	55
[Debugger] メニュー	40	ICD3Err0032	55
[Download Firmware]	75	ICD3Err0033	55
E		ICD3Err0034	55
[Erase All Before Programming]	37	ICD3Err0035	56
[Erase Flash Device]	66, 73	ICD3Err0036	56
Explorer 16 デモ ボード	29	ICD3Err0037	56
F		ICD3Err0038	56
[Freeze on Halt]	50	ICD3Err0039	56
H		ICD3Err0040	56
[Halt]	65	ICD3Err0041	56
Hex ファイル	37	ICD3Err0045	56
Hex ファイルの作成	37	ICD3Err0046	56
I		ICD3Err0047	56
ICD 3 テスト インターフェース ボード	10, 83	ICD3Err0052	57
ICD3Err0001	53	ICD3Err0053	57
ICD3Err0002	53	ICD3Err0054	57
ICD3Err0003	53	ICD3Err0055	57
ICD3Err0005	53	ICD3Err0056	57
ICD3Err0006	53	ICD3Err0057	57
ICD3Err0007	53	ICD3Err0063	57
ICD3Err0008	53	ICD3Err0064	57
ICD3Err0009	53	ICD3Err0065	57
ICD3Err0010	54	ICD3Err0066	57
ICD3Err0011	54	ICD3Err0067	57
ICD3Err0012	54	ICD3Err0068	57
ICD3Err0013	54	ICD3Err0069	57
		ICD3Err0070	57
		ICD3Err0071	58
		ICD3Err0072	58
		ICD3Err0073	58
		ICD ヘッダ	10

MPLAB® ICD 3 インサーキット デバッガ ユーザーズ ガイド

ICSP	18, 19, 20, 81
ICSPCLK	81
ICSPDAT	81
ICSP によるデバイスとの標準通信	14
L	
LED	29, 80
M	
MPLAB C30	33
MPLAB ICD 3 とは	9
MPLAB IDE	21
P	
PC、省電力モード	50, 51, 80
PGC	15, 16, 17, 18, 19
PGD	15, 16, 17, 18, 19
PIC24FJ128GA010、チュートリアル	29
PIM	14
[Program]	66, 73
[Program after successful build]	74
[Program Memory] タブ	37
R	
[Read]	66, 73
Readme	4
[Reconnect]	66
[Reset]	
プロセッサ	66
[Run]	65
[Run after successful program]	74
S	
SMPS	11
U	
USB	80, 89
ケーブル	10
デバイス ドライバ	21
ハブ	80
V	
Vcap	17
Vdd	15, 16, 17, 18
[Verify]	73
Vpp	15, 16, 17, 18, 19
Vss	15, 16, 17, 18
W	
[Watch] ウィンドウ	42
WWW アドレス	5
い	
一般的な対処方法	58
インジケータ LED	80
インターネット アドレス	5
う	
ウォッチドッグ タイマ	18, 90
お	
お客様変更通知サービス	5
き	
キット内容	10
機能のサポート	11
休止モード	50, 51, 80
け	
ケーブル	
長さ	80, 82
こ	
コードの実行	40
コード保護	18
顧客サービス	6
コンデンサ	17
コンフィギュレーション ビット	18, 27, 36
さ	
参考資料、推奨	4
し	
省電力モード	50, 51, 80
す	
ステップ	65
ストップウォッチ	43, 63
せ	
セキュア セグメント	77
そ	
ソフトウェア ブレークポイント	44
た	
ターゲット デバイス	18
ターゲットとの接続	
回路	16
標準	15
不適切な回路	17
耐久性、カード ガイド	80
タイマ 1	29
ち	
チュートリアル	29
つ	
ツールバーのボタン	40
て	
抵抗	17
テーブル読み出し保護	18
デバイスと開発モードの選択	30
デバイスと機能のサポート	11
デバイスによる予約リソース	20
デバッガの正常動作を妨げる回路	17
デバッグ	40
実行プログラム	20
レジスタ	20
デバッグ モード	
操作手順	19

デモ ボード	39
と	
ドライバ ボード	
標準	81
トランジション ソケット	10
仕様	4, 23
は	
ハードウェア ブレークポイント	41
ハードウェアおよびソフトウェアの セットアップ	30
ハブ、USB	80
ひ	
標準通信	
接続	15
ドライバ ボード	81
ふ	
ファームウェアのダウンロード	75
ブルアップ抵抗	17
ブレークポイント	
セットアップ	41, 63
ソフトウェア	44, 63, 75
ハードウェア	41, 63
ブレークポイントの設定	41
プログラミング	45
プログラミングとデバッグのオプションの設定	38
プログラミングのオプション	37
プログラムおよびデバッグ コードのロード	39
プロジェクトウィザード	26, 33
プロセッサ拡張キット	10
へ	
ヘッダ ボード	
仕様	4
ほ	
ポート A	29
保証登録	4
本書	
構成	1
表記	3
ま	
マイクロチップ社のインターネット ウェブサイト	5
み	
緑のアイコン	30
も	
モジュラ インターフェース ケーブル	18

世界各国での販売およびサービス

北米

本社

2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
テクニカル サポート :
http://support.microchip.com
ウェブ アドレス :
www.microchip.com

アトランタ

Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

ボストン

Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

シカゴ

Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

クリーブランド

Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

ダラス

Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

デトロイト

Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

ココモ

Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

ロサンゼルス

Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

サンタクララ

Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

トロント

Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

アジア / 太平洋

アジア太平洋支社

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

オーストラリア - シドニー

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

中国 - 北京

Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 - 成都

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 香港 SAR

Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 南京

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青島

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 瀋陽

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深川

Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 - 武漢

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 厦門

Tel: 86-592-2388138
Fax: 86-592-2388130

中国 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 珠海

Tel: 86-756-3210040
Fax: 86-756-3210049

アジア / 太平洋

インド - バンガロール

Tel: 91-80-3090-4444
Fax: 91-80-3090-4080

インド - ニューデリー

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

インド - プネ

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 - 横浜

Tel: 81-45-471-6166
Fax: 81-45-471-6122

韓国 - 大邱

Tel: 82-53-744-4301
Fax: 82-53-744-4302

韓国 - ソウル

Tel: 82-2-554-7200
Fax: 82-2-558-5932 または
82-2-558-5934

マレーシア - クアラルンプール

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

マレーシア - ペナン

Tel: 60-4-227-8870
Fax: 60-4-227-4068

フィリピン - マニラ

Tel: 63-2-634-9065
Fax: 63-2-634-9069

シンガポール

Tel: 65-6334-8870
Fax: 65-6334-8850

台湾 - 新竹

Tel: 886-3-6578-300
Fax: 886-3-6578-370

台湾 - 高雄

Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

タイ - バンコク

Tel: 66-2-694-1351
Fax: 66-2-694-1350

ヨーロッパ

オーストリア - ヴェルス

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

デンマーク - コペンハーゲン

Tel: 45-4450-2828
Fax: 45-4485-2829

フランス - パリ

Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

ドイツ - ミュンヘン

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

イタリア - ミラノ

Tel: 39-0331-742611
Fax: 39-0331-466781

オランダ - ドリユーン

Tel: 31-416-690399
Fax: 31-416-690340

スペイン - マドリッド

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 - ウォーキングム

Tel: 44-118-921-5869
Fax: 44-118-921-5820