PICkit™ 3
MPLAB® X IDE 用
インサーキット デバッガ / プログラマ
ユーザガイド

注意： この日本語版文書は参考資料としてご利用ください。最新情報は必ずオリジナルの英語版をご参照願います。

© 2014 Microchip Technology Inc.

Microchip 社製デバイスのコード保護機能に関して以下の点にご注意ください。

- Microchip 社製品は、該当する Microchip 社データシートに記載の仕様を満たしています。
- Microchip 社では、通常の条件ならびに仕様に従って使用した場合、Microchip 社製品のセキュリティ レベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- しかし、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する事もまた事実です。弊社の理解では、こうした手法は Microchip 社データシートにある動作仕様書以外の方法で Microchip 社製品を使用する事になります。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。
- Microchip 社は、コードの保全性に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。
- Microchip 社を含む全ての半導体メーカーで、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、Microchip 社が製品を「解読不能」として保証す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。Microchip 社では、常に製品のコード保護機能の改善に取り組んでいます。Microchip 社のコード保護機能の侵害は、デジタル ミレニアム著作権法に違反します。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、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 アプリケーション等に関する情報は、ユーザ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事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。Microchip 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。Microchip 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。生命維持装置あるいは生命安全用途に Microchip 社の製品を使用する事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、Microchip 社は擁護され、免責され、損害を受けない事に同意するものとします。暗黙的あるいは明示的を問わず、Microchip 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 社の名称とロゴ、Microchip ロゴ、dsPIC、FlashFlex、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、PIC32 ロゴ、rfPIC、SST、SST ロゴ、SuperFlash、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、HI-TECH C、Linear Active Thermistor、MTP、SEEVAL、Embedded Control Solutions Company は、米国における Microchip Technology Incorporated の登録商標です。

Silicon Storage Technology は、他の国における Microchip Technology Inc. の登録商標です。

Analog-for-the-Digital Age、Application Maestro、BodyCom、chipKIT、chipKIT ロゴ、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPF、MPLAB Certified ロゴ、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICtail、REAL ICE、rfLAB、Select Mode、SQI、Serial Quad I/O、Total Endurance、TSHARC、UniWinDriver、WiperLock、ZENA および Z-Scale は、米国およびその他の 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。

GestIC および ULPP は、Microchip Technology Inc. の子会社である Microchip Technology Germany II GmbH & Co. & KG 社の他の国における登録商標です。

その他、本書に記載されている商標は各社に帰属します。

© 2014, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-62077-560-8

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

Microchip 社では、Chandler および Tempe (アリゾナ州)、Gresham (オレゴン州) の本部、設計部およびウェハー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: 2009 認証を取得しています。Microchip 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。さらに、開発システムの設計と製造に関する Microchip 社の品質システムは ISO 9001:2000 認証を取得しています。

© 2014 Microchip Technology Inc.

宣言の対象： PICkit™3 インサーキット デバッガ / プログラマ

EU Declaration of Conformity

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

This declaration of conformity is issued by the manufacturer.

The development/evaluation tool is designed to be used for research and development in a laboratory environment. This development/evaluation tool is not intended to be a finished appliance, nor is it intended for incorporation into finished appliances that are made commercially available as single functional units to end users. This development/evaluation tool complies with EU EMC Directive 2004/108/EC and as supported by the European Commission's Guide for the EMC Directive 2004/108/EC (8th February 2010).

This development/evaluation tool complies with EU RoHS2 Directive 2011/65/EU.

For information regarding the exclusive, limited warranties applicable to Microchip products, please see Microchip's standard terms and conditions of sale, which are printed on our sales documentation and available at www.microchip.com.

Signed for and on behalf of Microchip Technology Inc. at Chandler, Arizona, USA

Derek Carlson
VP Development Tools

05-DEC-2011
Date

NOTES:

© 2014 Microchip Technology Inc.

目次

まえがき 7

第 1 部 - 入門編

Chapter 1. インサーキット デバッガ / プログラマについて
1.1 はじめに 13
1.2 PICkit 3 インサーキット デバッガ / プログラマについて 13
1.3 PICkit 3 インサーキット デバッガ / プログラマの機能 16
1.4 PICkit 3 インサーキット デバッガ / プログラマの構成要素 16

Chapter 2. 動作
2.1 はじめに 17
2.2 ツールの比較 18
2.3 PICkit 3 と PICkit 2 18
2.4 デバッガとターゲットの通信 19
2.5 通信の接続 21
2.6 デバッグ 24
2.7 デバッグのための要件 25
2.8 プログラミング 27
2.9 デバッガが使うリソース 27

Chapter 3. デバッガの使い方
3.1 はじめに 29
3.2 インストールと設定 29
3.3 一般的なデバッグ機能 30
3.4 ターゲットの接続 30
3.5 ターゲットボードの設定 31
3.6 MPLAB X IDE の設定 33
3.7 デバッグの開始と停止 33
3.8 プロセッサメモリとファイルの表示 33

Chapter 4. PICkit 3 Debug Express
4.1 はじめに 35
4.2 PICkit 3 Debug Express キットの内容 35
4.3 ハードウェアとソフトウェアのインストール 35

Chapter 5. PICkit 3 Programmer-To-Go

5.1 はじめに 37
5.2 PICkit 3 Programmer-To-Go の USB 電源 37
5.3 PICkit 3 Programmer-To-Go 対応デバイス 38
5.4 PICkit 3 で Programmer-To-Go 機能を使うための設定 39
5.5 PICkit 3 Programmer-To-Go の使用 42
5.6 Programmer-To-Go モードの終了 43

第 2 部 - トラブルシュート

Chapter 6. トラブルシュートの最初のステップ

6.1 はじめに 47
6.2 最初に確認する 5 つの項目 47
6.3 デバッグに失敗する主な理由 47
6.4 その他の確認事項 48

Chapter 7. よく寄せられる質問 (FAQ)

7.1 はじめに 49
7.2 動作に関する FAQ 49
7.3 不具合に関する FAQ 50

Chapter 8. エラーメッセージ

8.1 はじめに 53
8.2 特定の状況で表示されるエラーメッセージ 53
8.3 一般的な対処方法 54

Chapter 9. エンジニアリング技術ノート (ETN)

第 3 部 - リファレンス

補遺 A. ハードウェア仕様

A.1 はじめに 61
A.2 ハイライト 61
A.3 適合宣言書 61
A.4 USB ポート / 電源 62
A.5 PICkit 3 インサーキット デバッガ / プログラマ 62
A.6 標準通信ハードウェア 63
A.7 ターゲットボードに関する注意事項 65

補遺 B. PICkit 3 の回路図

補遺 C. 改訂履歴

用語集 71

索引 91

各国の営業所とサービス 94

© 2014 Microchip Technology Inc.

まえがき

お客様へのご注意

全ての文書の内容は時間と共に古くなります。本書も例外ではありません。Microchip 社のツールとマニュアルはお客様のニーズを満たすために常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。最新の文書は弊社ウェブサイト (www.microchip.com) をご覧ください。

文書は「DS」番号で識別します。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しています。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。

開発ツールの最新情報は、MPLAB® X IDE のオンラインヘルプをご覧ください。[Help] メニューから [Topics] を選択すると、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

本章では、PICkit 3™ インサーキット デバッガ / プログラマを使い始める前に知っておくと便利な一般情報を提供します。主な内容は下記の通りです。

- 本書の構成
- 本書の表記規則
- 推奨参考資料

本書の構成

本書では、PICkit 3 を開発ツールとして使い、ターゲットボード上のファームウェアのエミュレーションとデバッグを実行する方法、並びにデバイスのプログラム方法を説明します。以下に本書の構成を示します。

第 1 部 - 入門編

- **Chapter 1. インサーキット デバッガ / プログラマについて**
  PICkit 3 を紹介し、これをアプリケーションの開発に役立てる方法を説明します。
- **Chapter 2. 動作**
  PICkit 3 の動作原理を説明します。設定オプションも説明します。
- **Chapter 3. デバッガの使い方**
  インストールと設定、一般的なデバッグ機能、ターゲットの使用、MPLAB® 統合開発環境 (IDE) の設定、関連するデバッガのトピックについて説明します。
- **Chapter 4.PICkit 3 Debug Express**
  PICkit 3 Debug Express の基本的な使い方を説明します。
- **Chapter 5.PICkit 3 Programmer-To-Go**
  PC に接続せずに PICkit 3 でデバイスをプログラムする手順を説明します。

第 2 部 - トラブルシュート

- **Chapter 6. トラブルシュートの最初のステップ** - デバッガの動作に問題がある場合に最初に試すべき項目を説明します。
- **Chapter 7. よく寄せられる質問 (FAQ)** - トラブルシュートに役立つ、よく寄せられる質問の一覧を示します。
- **Chapter 8. エラーメッセージ** - エラーメッセージと、推奨している解決方法の一覧を示します。
- **Chapter 9. エンジニアリング技術ノート (ETN)**

第 3 部 - リファレンス

- **補遺A.ハードウェア仕様** - デバッグシステムのハードウェア仕様と電気的仕様を示します。
- **補遺 B.PICkit 3 の回路図**
- **補遺 C. 改訂履歴**

© 2014 Microchip Technology Inc.

本書の表記規則

本書では以下の表記規則を適用します。

本書の表記規則

表記	意味	例
Arial、MS ゴシックフォント :		
二重かぎカッコ : 『』	参考資料	『MPLAB® IDE ユーザガイド』
	テキストの強調	... は唯一のコンパイラです ...
角カッコ : [ ]	ウィンドウ名	[Output] ウィンドウ
	ダイアログ名	[Settings] ダイアログ
	メニューの選択肢	[Enable Programmer] を選択
かぎカッコ : 「 」	ウィンドウまたはダイアログのフィールド名	「Save project before build」
右山カッコ (>) を使い、角カッコで囲まれた下線付きテキスト	メニューパス	[File]>[Save]
角カッコで囲んだ太字のテキスト	ダイアログのボタン	**[OK]** をクリックする
	タブ	**[Power]** タブをクリックする
N'Rnnnn	Verilog 形式の数値 : N は総桁数、R は基数、n は各桁の値	4'b0010, 2'hF1
山カッコで囲まれたテキスト : < >	キーボードのキー	<Enter>、<F1> を押す
Courier New フォント :		
標準の Courier New	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`
	コマンドライン オプション	`-Opa+, -Opa-`
	ビット値	`0, 1`
	定数	`0xFF, 'A'`
斜体	変数の引数	*file.o* (*file* は有効な任意のファイル名)
角カッコ : [ ]	オプションの引数	`mcc18 [options] file [options]`
中カッコとパイプ文字 : { | }	どちら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...]`
	ユーザが定義するコード	`void main (void) { ... }`

推奨参考資料

本書は、PICkit 3 の使い方を説明したものです。その他の便利な文書を、以下に一覧でご紹介します。以下に記載した Microchip 社の文書を推奨します。

Multi-Tool Design Advisory (DS51764)

まず、この文書を最初にお読みください。この文書には、PICkit を使う際に注意すべき重要な情報を記載しています。

PICkit 3 のリリースノート

PICkit 3 の使い方に関する最新情報は、[Start Page] の [Release Notes and Support Documentation] のリリースノートをお読みください。リリースノートには、ユーザガイドには反映されていない最新情報と既知の問題を記載しています。

MPLAB X - Using PICkit 3 In-Circuit Debugger Poster (DS52010)

このシートには、標準通信とターゲットボードを使って PICkit 3 プログラマ / デバッガを使うためのハードウェアの接続方法とソフトウェアのインストール方法を記載しています。

PICkit 3 インサーキット デバッガ / プログラマ オンラインヘルプ ファイル

MPLAB X IDE はデバッガに関するヘルプファイルを備えています。使い方、トラブルシュート、ハードウェア仕様を記載しています。印刷された文書よりも新しい情報を含んでいる可能性があります。さらに、各種デバイスの制約事項の一覧も収めています。

Processor Extension Pak and Header Specification (DS51292)

この文書では、ピンまたはリソースを無駄に専有せずにデバイスを適切にデバッグするための、デバッグヘッダの取り付け方法と使い方を説明しています。プロセッサ拡張パックとヘッダのオンラインヘルプ ファイルも参照してください。

Transition Socket Specification (DS51194)

ヘッダと一緒に使う変換ソケットに関する情報は、この資料を参照してください。

© 2014 Microchip Technology Inc.

第 1 部 – 入門編

Chapter 1. インサーキット デバッガ / プログラマについて 13
Chapter 2. 動作 17
Chapter 3. デバッガの使い方 29
Chapter 4. PICkit 3 Debug Express 35
Chapter 5. PICkit 3 Programmer-To-Go 37

NOTES:

© 2014 Microchip Technology Inc.

Chapter 1. インサーキット デバッガ / プログラマについて

1.1 はじめに

ここでは、PICkit™ 3 インサーキット デバッガ / プログラマの概要を説明します。

- PICkit 3 インサーキット デバッガ / プログラマについて
- PICkit 3 インサーキット デバッガ / プログラマの機能
- PICkit 3 インサーキット デバッガ / プログラマの構成要素

1.2 PICkit 3 インサーキット デバッガ / プログラマについて

PICkit 3 インサーキット デバッガ / プログラマ (図 1-1 参照) はシンプルかつ低コストなインサーキット デバッガで、Windows® で動作する MPLABX IDE ソフトウェアを使って操作します。PICkit 3 インサーキット デバッガ / プログラマは、開発エンジニアにとって欠かせないツールの 1 つです。ソフトウェアの開発からハードウェアの統合まで、幅広い用途に対応します。

PICkit 3 インサーキット デバッガ / プログラマは、インサーキット シリアル プログラミング ™ (ICSP™) と拡張 ICSP の 2 線式シリアル インターフェイスを使って、Microchip 社の PIC® MCU と dsPIC® DSC 向けのハードウェア / ソフトウェアの開発を行えるデバッガシステムです。

PICkit 3 インサーキット デバッガ / プログラマは、デバッガとしてだけでなく開発プログラマとしても使えます。

Note: PICkit 3 インサーキット デバッガ / プログラマは、量産プログラマではありません。開発専用です。

PICkit 3 デバッガは、デバッグ機能を内蔵した組み込みプロセッサのプログラムとデバッグ用に開発しました。PICkit 3 の主な特長は以下の通りです。

- Windows 標準ドライバでフルスピード USB をサポート
- リアルタイム実行
- プロセッサが最大速度で動作可能
- 過電圧 / 短絡監視機能を内蔵
- 低電圧～ 5 V (レンジ : 1.8 ～ 5 V)
- LED インジケータ (POWER、ACTIVE、STATUS)
- マイクロコントローラのプログラムメモリとデータメモリの読み書き
- 全種類のメモリ (EEPROM、ID、コンフィグレーション、プログラム) の消去とベリファイ
- ブレークポイントでの周辺モジュールのフリーズ

図 1-1: PICkit™ 3 MCU インサーキット デバッガ / プログラマ

凡例 :

1 – ストラップ用ループ
2 – USB ポート
3 – 1 ピンマーカー
4 – プログラミング コネクタ
5 – インジケータ LED
6 – プッシュボタン

1.2.1 ストラップ用ループ

ストラップ用ループを取り付け具として使い、PICkit 3 を吊るしたり身に着けたりできます。

1.2.2 USB ポート

USB mini-B コネクタタイプの USB ポートです。付属の USB ケーブルを使って PICkit 3 を PC に接続します。

1.2.3 1 ピンマーカー

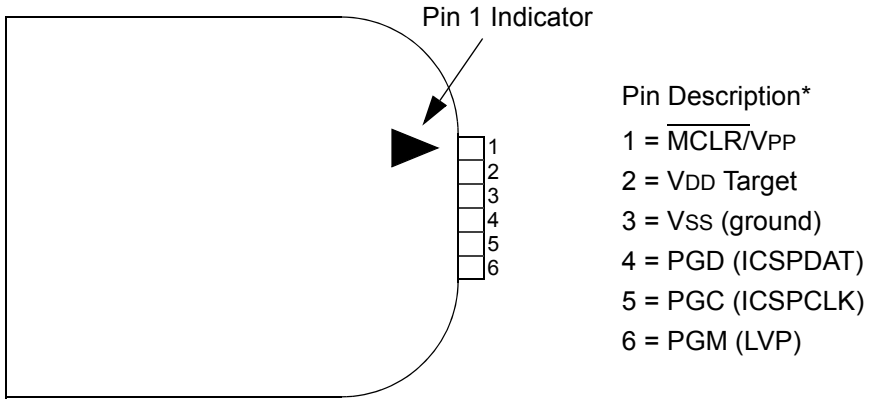
コネクタの向きを合わせる目印として、1 ピンの位置を示しています。

© 2014 Microchip Technology Inc.

1.2.4 プログラミング コネクタ

ターゲット デバイスと接続するための 6 ピンヘッダ (0.100 インチ間隔) です。ピン配置は図 1-2 を参照してください。

図 1-2: PICkit™ 3 プログラマのコネクタのピン配置



* 6 ピンヘッダ (0.100 インチ間隔) は 0.025 インチ角型ピンに適合します。

Note: シリアル EEPROM デバイスのプログラミング時は、異なるプログラミング コネクタのピン配置が必要です。このタイプのデバイスのピン配置は、MPLAB X IDE ソフトウェアに付属する PICkit 3 の ReadMe ファイルに記載しています (MPLAB X IDE の [Start Page] で [Release Notes and Support Documentation] をクリック)。

1.2.5 インジケータ LED

インジケータ LED は、PICkit 3 の動作ステータスを表示します。

1. **POWER** (緑) - PICkit 3 に USB ポートから電源が供給されています。
2. **ACTIVE** (青) - PC の USB ポートに接続されて通信リンクがアクティブです。
3. **STATUS** (3 色のいずれか)

   正常 (緑) - 開始する準備が整ったか、正常に完了しました。

   ビジー (オレンジ) - プログラミング等の動作を実行中です。

   エラー (赤) - エラーが発生しました。

Note: 点滅する LED は、その他の情報を示します。詳細は表 5-2 を参照してください。

1.2.6 プッシュボタン

このプッシュボタンは、Programmer-To-Go 機能で使います。**Chapter 5.「PICkit 3 Programmer-To-Go」**を参照してください。

1.3 PICkit 3 インサーキット デバッガ / プログラマの機能

PICkit 3 インサーキット デバッガ / プログラマで以下が可能です。

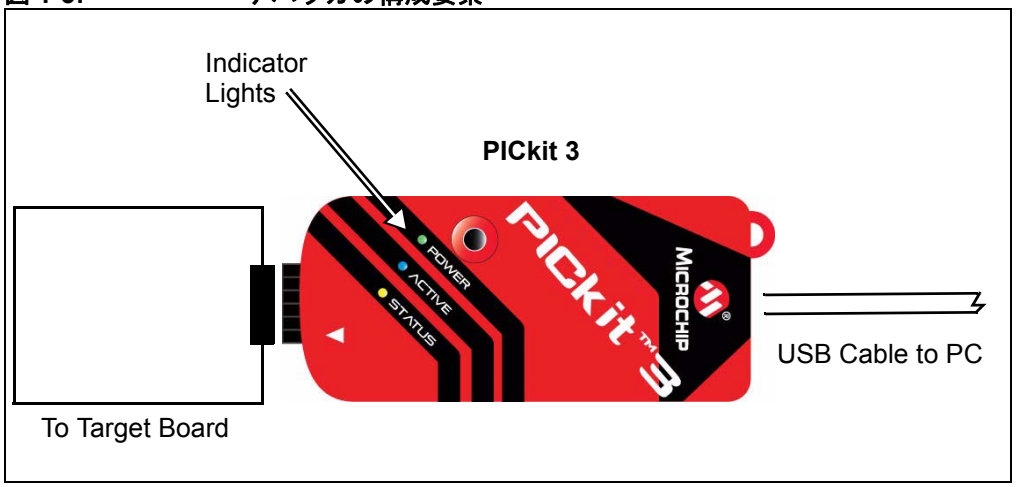
- 実際のハードウェアでのアプリケーションのリアルタイム デバッグ
- ハードウェア ブレークポイント使ったデバッグ
- 内部イベントに基づいたブレークポイント設定
- 内部ファイルレジスタの監視
- フルスピードでのエミュレート
- デバイスのプログラム

1.4 PICkit 3 インサーキット デバッガ / プログラマの構成要素

PICkit 3 インサーキット デバッガ / プログラマは、以下の構成要素を含みます。

1. PICkit 3 (POWER、ACTIVE、STATUS インジケータ付き)
2. USB ケーブル (デバッガと PC 間の通信と、デバッガへの電源供給用)

図 1-3: デバッガの構成要素



その他の製品も別途ご注文頂けます。

- 以下を含む PICkit 3 Debug Express Kit:
  - PIC18F45K20 MCU を実装した 44 ピン デモボード
  - PIC18 MCU 用 MPLAB C コンパイラ (Free Version)
  - 分かりやすいレッスンとチュートリアル
  - その他のソフトウェア ユーティリティ、サンプルコード (ソースコード付き)、文書一式
- 変換ソケット
- ICD ヘッダ
- MPLAB X IDE プロセッサ拡張キット

© 2014 Microchip Technology Inc.

Chapter 2. 動作

2.1 はじめに

ここでは、PICkit 3 インサーキット デバッガ / プログラマの動作を簡単に説明します。ここでの目的は、PICkit 3 デバッガ / プログラマを使ってデバッグ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事にあります。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 デバッグとプログラミングの基本的な動作原理を説明します。

- ツールの比較
- PICkit 3 と PICkit 2
- デバッガとターゲットの通信
- 通信の接続
- デバッグ
- デバッグのための要件
- プログラミング
- デバッガが使うリソース

2.2 ツールの比較

下表に示す通り、PICkit 3 インサーキット デバッガ / プログラマは、Microchip 社の他のデバッグツールとは物理的、動作的に異なります。機能はデバイスによって異なる場合があります (オンラインヘルプ ファイルの **[Device and Feature Support]** を参照してください)。

表 2-1: デバッグツールの比較

機能	PICkit 3 プログラマ / デバッガ	MPLAB ICD 3 インサーキット デバッガ	MPLAB REAL ICE インサーキット エミュレータ
USB スピード	フルスピードのみ	ハイスピード / フルスピード	ハイスピード / フルスピード
USB ドライバ	HID	Microchip	Microchip
USB バスパワー動作	Yes	Yes	Yes
ターゲットへの給電	Yes	Yes	No
プログラマブル V_{PP} と V_{DD}	Yes	Yes	Yes
ターゲット Vdd からの ドレイン電流	20 mA	<1 mA	<1 mA
過電圧 / 過電流保護	Yes (SW)	Yes (HW)	Yes (HW)
デバイス エミュレーション	フルスピード	フルスピード	フルスピード
ハードウェア ブレーク ポイント	単純	複合	複合
ストップウォッチ	Yes	Yes	Yes
ソフトウェア ブレーク ポイント	No	Yes	Yes
プログラム イメージ	512 KB	No	No
シリアライズド USB	Yes	Yes	Yes
トレース	No	No	Yes
データキャプチャ	No	No	Yes
ロジックプローブ トリガ	No	No	Yes
高速 /LVDS 接続	No	No	Yes
量産プログラマ	No	Yes	Yes

2.3 PICkit 3 と PICkit 2

PICkit 3 インサーキット デバッガ / プログラマは PICkit 2 インサーキット デバッガと似ています。

これら 2 つのデバッガの似ている点は以下の通りです。

- PC から USB ケーブルで電源を供給する
- 供給する電源電圧値を設定できる

PICkit 2 にはなく、PICkit 3 で追加された機能は以下の通りです。

- 拡張 EE プログラム イメージ空間 (512 KB)
- 正確な参照電圧
- 電圧レンジが広くなった (1.8 ～ 5 V V_{DD}、1.8 ～ 14 V V_{PP})

© 2014 Microchip Technology Inc.

2.4 デバッガとターゲットの通信

ここでは、デバッガの構成を説明します。

CAUTION

ソフトウェアをインストールしてからハードウェアを接続します。ソフトウェアと USB ドライバをインストールする前に、ハードウェアを接続しないでください。

PICkit 3 またはターゲットに電源を供給している時にハードウェア接続を変更しないでください。

標準 ICSP による通信

PICkit 3 デバッガは、標準 ICSP 通信によるプログラミングとデバッグが可能です。この 6 ピン接続は、従来の PICkit 2 開発用プログラマ / デバッガで使っていたものと同じです。

モジュラケーブルは、以下のどちらかに挿入できます。

- デバイスがターゲットボードに実装済みの場合はターゲットボードの対応ソケット (図 2-1)
- ターゲットボードに標準のアダプタ / ヘッダボード コンボ (プロセッサパックとして提供) を接続している場合はアダプタ / ヘッダボード (図 2-2)

Note: 旧式のヘッダボードは 6 ピンシングル インライン コネクタではなく 6 ピンのモジュラコネクタを使っているため、AC164110 ICSP アダプタを使ってデバッガと接続します。

標準通信の詳細は、**補遺 A.「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 — ICE 回路内蔵デバイスの場合

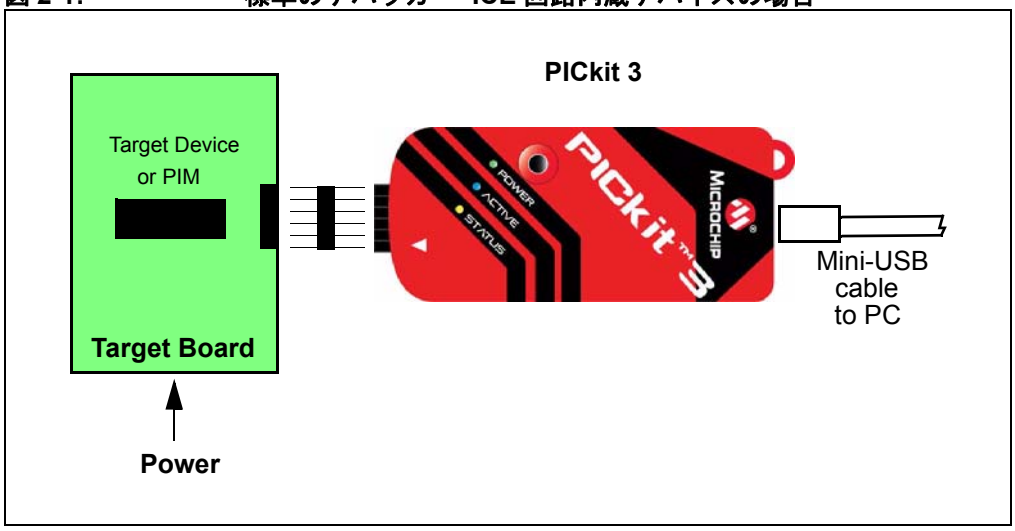
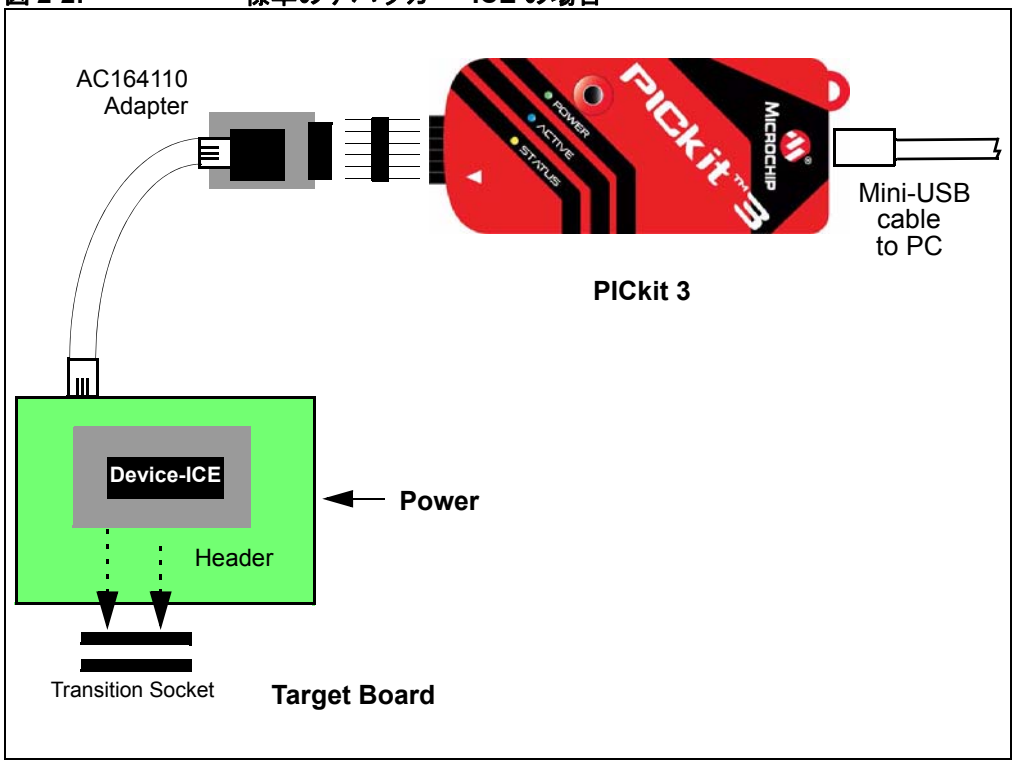


図 2-2: 標準のデバッガ — ICE の場合



© 2014 Microchip Technology Inc.

2.5 通信の接続

2.5.1 標準通信ターゲットとの接続

2.5.1.1 インライン コネクタを使う場合

6 ピンのインライン コネクタを使って PICkit 3 インサーキット デバッガ / プログラマとターゲットボードのコネクタを接続します (図 2-1 参照)。表 2-2 と**セクション A.6「標準通信ハードウェア」**も参照してください。

表 2-2: ターゲット側コネクタのピン配置

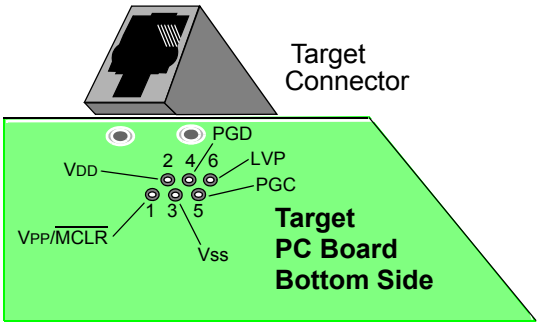
コネクタのピン番号	マイクロコントローラのピン
1	$\overline{\text{MCLR}}$/VPP
2	VDD
3	グランド
4	PGD (ICSPDAT)
5	PGC (ICSPCLK)
6	PGM (LVP)

2.5.1.2 アダプタを使う場合

PICkit 3インサーキット デバッガ/プログラマとターゲット デバイスの間にAC164110アダプタを使って、6 芯のモジュラ インターフェイス ケーブルで接続します。図 2-3 に、ターゲット PC ボードの裏面から見たコネクタのピン番号を示します。

> **Note:** 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結線されます。つまり、ケーブルの片方の 1 ピンは、もう片方の 6 ピンに接続されます。**セクション A.6.2.3「モジュラケーブルの仕様」**を参照してください。

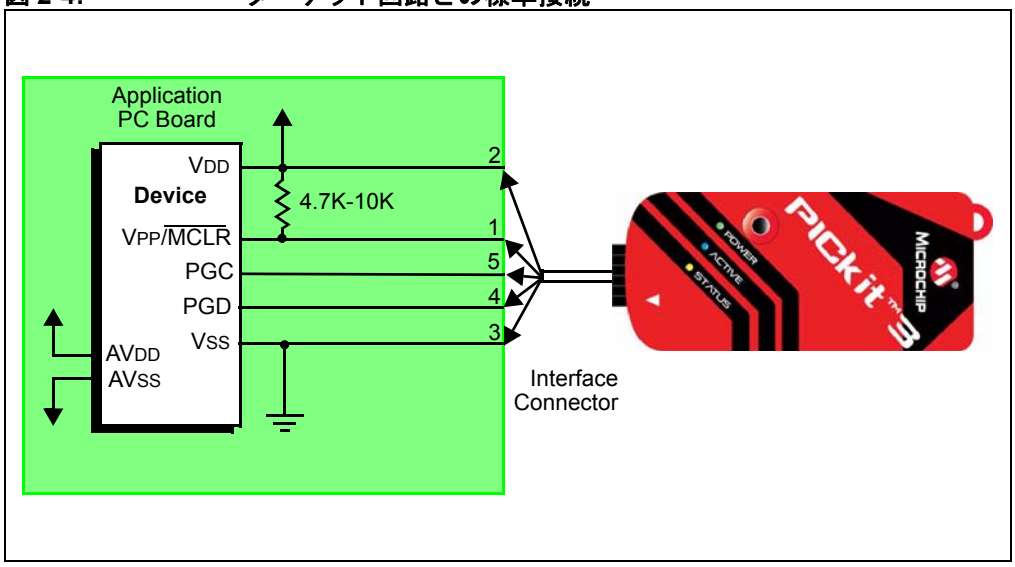
図 2-3: ターゲット側の標準 RJ-11 接続部



2.5.2 ターゲット回路との接続

図 2-4 に、PICkit 3 インサーキット デバッガ / プログラマとターゲットボード上のコネクタの配線を示します。この図では、コネクタからターゲットのプリント基板上のデバイスまでの配線も示しています。VPP/MCLR ラインから VDD へのプルアップ抵抗 (約 10 kΩ typ.) を接続する事を推奨します。これによって、このラインを Low にストローブしてデバイスをリセットできます。

図 2-4: ターゲット回路との標準接続



2.5.3 ターゲットの自己給電

以下の説明では、1 ピン (VPP/MCLR)、5 ピン (PGC)、4 ピン (PGD) の 3 つのラインのみがアクティブで、基本的なデバッガの動作に関係します。参考までに図 2-4 には、2 ピン (VDD) と 3 ピン (VSS) も示しています。PICkit 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う構成の 2 種類が可能です。

ターゲット アプリケーションの外付け電源を使う方法を推奨します。この構成では、デバッガがターゲットの VDD 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 VDD ライン (インターフェイス コネクタの 2 ピン) 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は 30 mA までに制限されています。この電源供給方法は、デバイスの VDD 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非常に小規模のアプリケーションには適しています。しかし、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しません。

デバイスによっては AVDD ラインと AVSS ラインがない場合もありますが、これらのラインがターゲット デバイスにある場合、デバッガが動作するにはこれらを全て適切なレベルに接続する必要があります。フローティングのままにする事はできません。

また、VCAP ラインのあるデバイス (例 : PIC18FXXJ MCU) は、適切なコンデンサまたはレベルに接続する必要があります。

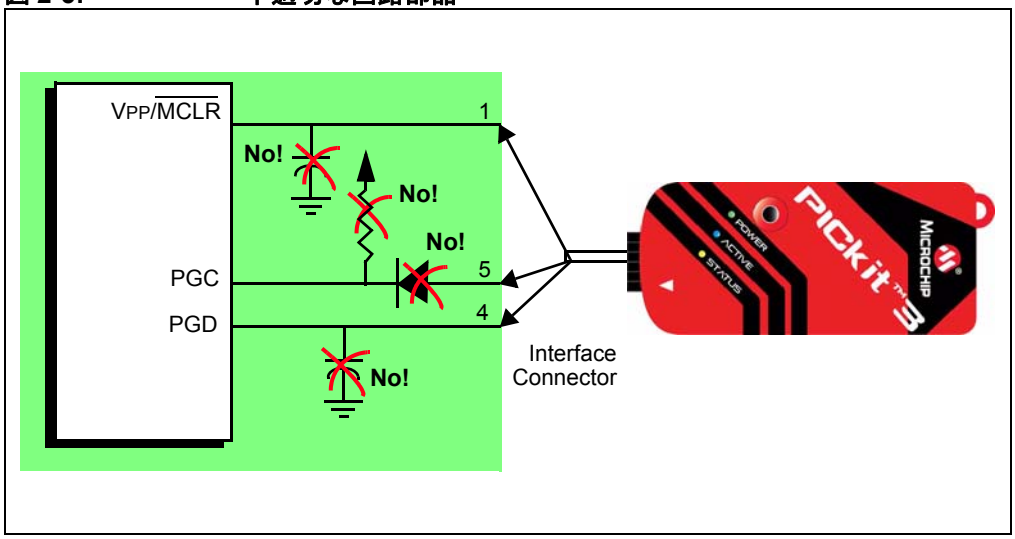
Note:	配線は非常にシンプルです。問題が発生する場合、これらの重要なラインに他の接続または部品を取り付けているために PICkit 3 インサーキット デバッガ/プログラマの正常な動作が妨げられている場合がほとんどです。

© 2014 Microchip Technology Inc.

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 PICkit 3 デバッガ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、以下のガイドラインに従う必要があります。

- PGC/PGD にプルアップ抵抗を使わない - これらのラインはデバッガ内部に 4.7 kΩ のプルダウン抵抗があるため、プルアップ抵抗を使うと電圧レベルが低下します。
- PGC/PGD にコンデンサを使わない - コンデンサを使うと、プログラミングとデバッグの通信の際にデータ / クロックラインの高速遷移を妨げます。
- $\overline{\text{MCLR}}$ にコンデンサを使わない - VPP の高速遷移を妨げます。通常、シンプルなプルアップ抵抗で十分です。
- PGC/PGD にダイオードを使わない - ダイオードを使うと、デバッガとターゲットデバイスの双方向通信ができません。

2.6 デバッグ

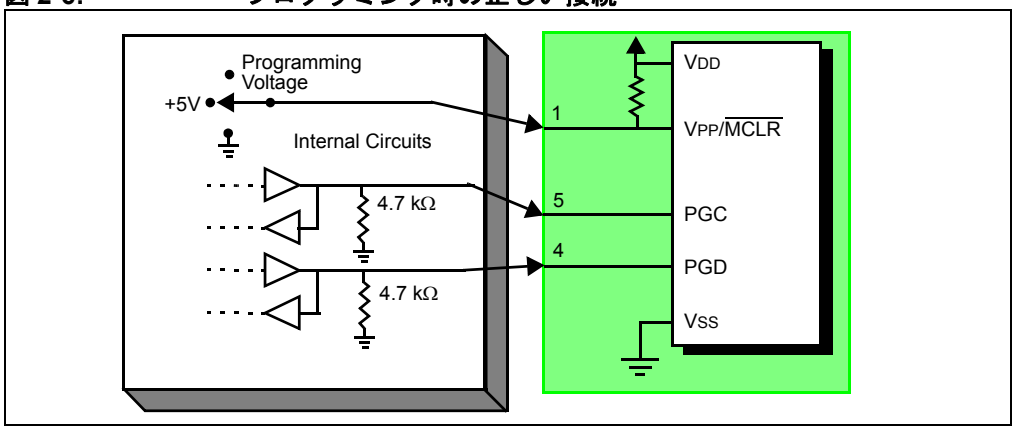
PICkit 3 インサーキット デバッガ / プログラマをデバッガとして使うには、2 段階の手順があります。まず、アプリケーションをターゲット デバイスにプログラムします (通常は PICkit 3 をプログラマとして使います)。次に、ターゲットのフラッシュ デバイスに内蔵されたインサーキット デバッグ ハードウェアを使ってアプリケーション プログラムを実行、テストします。これら 2 つの手順は、MPLAB X IDE での以下の操作に対応しています。

1. コードをターゲットにプログラムし、専用のデバッグ機能を有効にする (次セクション参照)。
2. デバッガを使っ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ムできないと、PICkit 3 インサーキット デバッガ / プログラマでデバッグを行えません。

図 2-6 に、プログラミングに必要な基本の配線を示します。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、PICkit 3 インサーキット デバッガ / プログラマの内部インターフェイ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不要ですが、電源の供給は必要です。プログラミングを行う際、デバッガは VPP/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。デバイスが正しくプログラムされた事を確認するために、PGC にクロックを送出して PGD からデータを読み出します。これは、デバイスの ICSP プロトコルに準拠しています。

© 2014 Microchip Technology Inc.

2.7 デバッグのための要件

PICkit 3 インサーキット デバッガ / プログラマでデバッグ (例 : ブレークポイントの設定、レジスタの参照) を行うために、正しく実施する必要のある重要な項目は以下の通りです。

- デバッガが PC に接続されている必要があります。USB ケーブルを介して PC からデバッガに電源を供給する事と、USB ケーブルを介して MPLAB X IDE ソフトウェアとデバッガが通信を行う事が必要です。詳細は **Chapter 3.「デバッガの使い方」**を参照してください。
- 図 2-6 に示したように、モジュラ インターフェイス ケーブル (または同等品) を使ってデバッガをターゲット デバイスの VPP、PGC、PGD ピンに接続する必要があります。VSS と VDD もデバッガとターゲット デバイスの間で接続しておく必要があります。
- ターゲット デバイスに電源を供給している事と、オシレータが正しく動作している事が必要です。何らかの理由でターゲット デバイスが動作しない場合、PICkit 3 インサーキット デバッガ / プログラマでデバッグを行う事はできません。
- ターゲット デバイスのコンフィグレーション ワードが以下に示すように正しくプログラムされている必要があります。
  - ターゲットの設計に応じてオシレータのコンフィグレーション ビットを RC、XT 等に対応する値に設定しておく必要があります。
  - 一部のデバイスではウォッチドッグ タイマが既定値で有効であるため、無効にする必要があります。
  - ターゲット デバイスのコード保護を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を無効にしておく必要があります。
- PGM (LVP) は無効にしておく必要があります。

上記の条件が満たされている事を確認したら、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

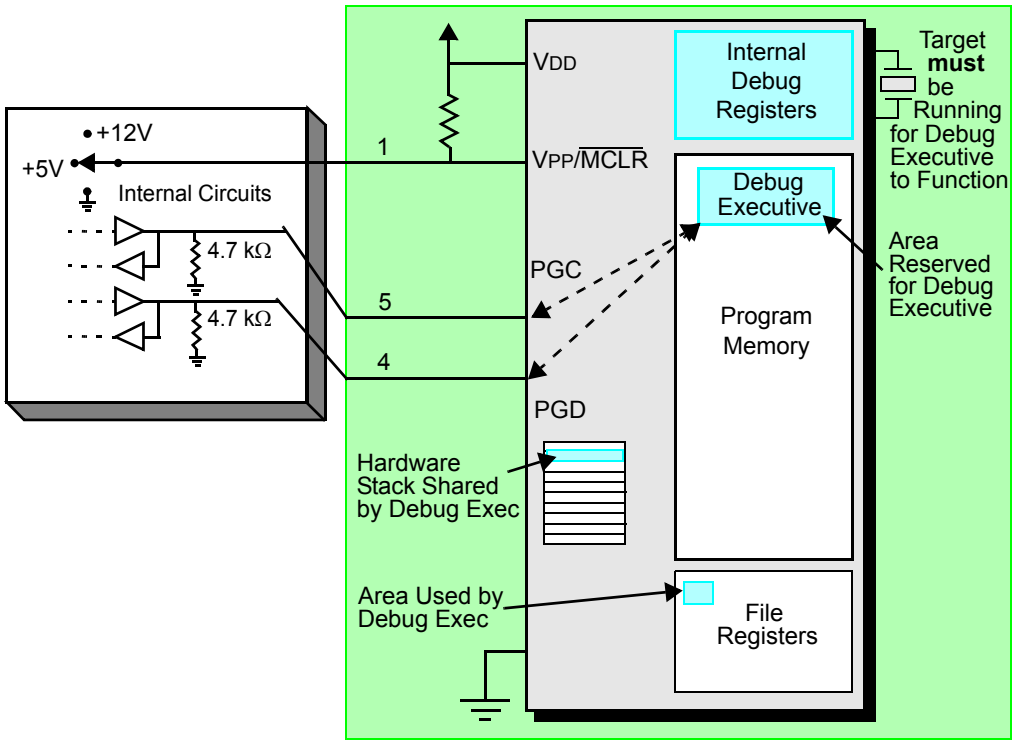
デバッグのための要件が満たされている事を前提に、以下の操作は MPLAB X IDE のメニューから PICkit 3 インサーキット デバッガ / プログラマを現在のツールとして設定している場合 ([Edit]>[Project Properties]、Hardware Tool) に実行できます。

- [Debug]>[Debug Project] の順に選択し、このセクションの最初で説明した 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書き込みます。
- ターゲット デバイスのプログラムメモリの上位空間に、小さな「デバッグ実行」プログラムを読み込みます。デバッグ実行プログラムはプログラムメモリに常駐する必要があるため、アプリケーション プログラムはこの予約済み空間を使えません。デバイスによっては、デバッグ実行プログラム専用の特別なメモリ空間を装備しています。詳細はデバイスのデータシートで確認してください。
- MPLAB X IDE がターゲット デバイスの「インサーキット デバッグ」専用レジスタを有効にします。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する事ができます。デバイスの予約済みリソースの詳細は、**セクション 2.9「デバッガが使うリソース」**を参照してください。
- ターゲット デバイスがデバッグモードで動作します。

2.7.2 デバッグの詳細

図 2-7 は、デバッグを実行できる状態になった PICkit 3 インサーキット デバッガ / プログラマを示しています。

図 2-7: デバッグの準備が完了した PICkit™ 3 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 X IDE のユーザインターフェイスでブレークポイントを設定すると、そ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 と PGD に対するコマンドは、これらの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。

次に MPLAB X IDE で [Debug]>[Debug Project] を選択します。デバッガが、デバッグ実行プログラムの実行を命令します。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、ユーザのアプリケーションは停止します。デバッガは PGC と PGD を介してデバッグ実行プログラムと通信し、ブレークポイントのステータス情報を取得して MPLAB X IDE に返します。この後、MPLAB X 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 の状態等、ターゲット デバイスに関する情報を取得します。これらの問い合わせは、最終的にはデバッグ実行プログラムが実行します。

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います。オシレータがない、電源接続が不良、ターゲットボードの短絡等、何らかの理由でデバイスが動作しない場合、デバッグ実行プログラムと PICkit 3 インサーキット デバッガ / プログラマが通信できず、MPLAB X IDE はエラーメッセージを表示します。

© 2014 Microchip Technology Inc.

ブレークポイントを取得するもう 1 つの方法は、[Debug]>[Pause] の選択です。すると、PGC ラインと PGD ラインが切り換わり、ターゲット デバイスのインサーキット デバッグ メカニズムの働きにより、プログラム カウンタがプログラムメモリ内のユーザコードからデバッグ実行プログラムに切り換わります。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 X IDE はデバッガとデバッグ実行プログラムの間の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 プログラミング

PICkit 3 を使ってデバイスをプログラムするには、以下の 3 つの方法があります。

- PICkit 3 と PC を接続し、MPLAB X IDE を使う。
- MPLAB X IDE で設定後、PICkit 3 の Programmer-To-Go 機能を使う (詳細は **Chapter 5.「PICkit 3 Programmer-To-Go」**参照)。
- PICkit 3 プログラマ アプリケーション (MPLAB X IDE を使わずに PICkit 3 でデバイスをプログラムするためのソフトウェア) を利用する (詳細は『PICkit 3 Programmer Application User's Guide』参照)。

PICkit 3 をプログラマとして使うと、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) に対してプログラムできます。PICkit 3 を現在のツールとして設定し ([Edit]>[Project Properties]、Hardware Tool)、以下の操作を実行します。

- [Run]>[Run Project] の順に選択し、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書き込む。プログラミング中に、クロックは必要ありません。また、コードの保護、ウォッチドッグ タイマの有効化、テーブル読み出しの保護等、プロセッサの全てのモードをプログラムできます。
- 一部のターゲット デバイスでは、プログラムメモリの上位空間に、小さな「プログラム実行」プログラムを読み込みます。これによって、大きなメモリを備えたデバイスのプログラミング速度が向上します。
- MPLAB X IDE はターゲット デバイスの「インサーキット デバッグ」専用レジスタを、全てのデバッグ機能と共に無効にします。これは、ブレークポイントの設定、レジスタの内容の参照または変更ができなくなる事を意味します。
- ターゲット デバイスはリリースモードで動作します。プログラマとして使う場合、デバッガは $\overline{\text{MCLR}}$ ラインをトグルしてターゲットをリセット / 起動する事しか行えません。

PICkit 3 インサーキット デバッガ / プログラマは、ICSP を利用してターゲットをプログラムします。VPP、PGC、PGD ラインは、前述した通りに接続してください。プログラミング中に、クロックは必要ありません。また、コード保護、ウォッチドッグ タイマ、テーブル読み出し保護等、プロセッサの全てのモードをプログラミングできます。

2.9 デバッガが使うリソース

各デバイスでデバッガが使うリソースを見るには、MPLAB X IDE の [Start Page] で [Release Notes and Support Documentation]>[Reserved Resources for PICkit 3] をクリックします。

NOTES:

© 2014 Microchip Technology Inc.

Chapter 3. デバッガの使い方

3.1 はじめに

ここでは、PICkit 3 インサーキット デバッガ / プログラマのインストール方法と使い方に関する以下のトピックについて説明します。

- インストールと設定
- 一般的なデバッグ機能
- ターゲットの接続
- ターゲットボードの設定
- MPLAB X IDE の設定
- デバッグの開始と停止
- プロセッサメモリとファイルの表示
- エディタに関する詳細は、NetBeans ヘルプの [IDE Basics]>[Basic File Features] を参照してください。

3.2 インストールと設定

IDE のインストールとデバッガの設定に関する詳細は、ヘルプファイルの [Getting Started with MPLAB X IDE] を参照してください。

作業手順は以下の通りです。

1. MPLAB X IDE をインストールする。
2. PICkit 3 を PC に接続し、既定値のドライバをインストールする。ターゲットの接続の詳細は、**Chapter 2.「動作」**を参照してください。

   Note: デバッガがターゲットボードに供給できる最大電流は 100 mA です。

3. 開発に使う言語ツールスイート / コンパイラをインストールする。
4. MPLAB X IDE を起動する。
5. [New Project] ウィザードを使い ([File]>[New Project])、「PICkit 3」をプロジェクトに追加する。
6. [Project Properties] ダイアログ ([File]>[Project Properties]) を使ってオプションを設定する。
7. [Project Properties] ダイアログ ([File]>[Project Properties]、[Hardware tool]) を使ってプログラミングのツールオプションを設定する。
8. [Run]>[Run Project] を選択してプロジェクトを実行 (ビルドと実行) する。

注意事項 :

1. 各デバッガには一意の識別子が付けられています。OS はこの識別子を認識します。
2. MPLAB X IDE は実行時 (Run または Debug Run) のみハードウェア ツールに接続します。MPLAB IDE v8 のように常時ハードウェア ツールに接続するには、[Tools]>[Options] を選択し、**[Embedded]** ボタンをクリックし、**[Generic Settings]** タブで [Keep hardware tool connected] にチェックを入れます。
3. コンフィグレーション ビットを表示できるのは、[Configuration Bits] ウィンドウのみです。これらをコードに設定するには、[Window]>[PIC Memory Views] の順に選択します。次に、[Memory] ドロップダウン リストから [Configuration Bits] を選択し、[Format] ドロップダウン リストから [Read/Write] を選択して、設定へのアクセスを有効にします。

3.3 一般的なデバッグ機能

デバッグ機能の詳細は、ヘルプファイルの [Getting Started with MPLAB X IDE] の [Debugging Code] のセクションを参照してください。このセクションは、以下の内容を含みます。

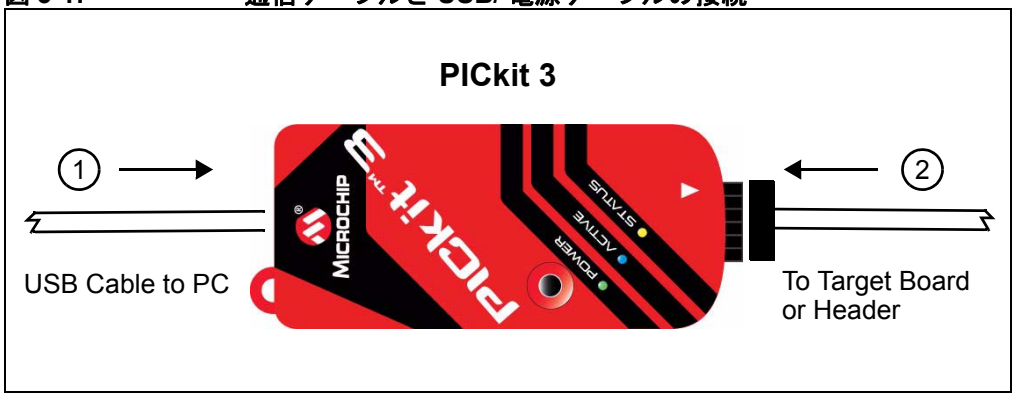
1. [Debug]>[Debug Project] の選択によるプロジェクトのデバッグ実行 (ビルド、プログラム、実行)
2. ブレークポイントの使い方
3. コードのステップ実行
4. [Watches] ウィンドウの使い方
5. メモリ、変数、コールスタックの表示
6. コールグラフの使い方

3.4 ターゲットの接続

PICkit 3 プログラマ / 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。詳細と接続図は、**セクション 2.4「デバッガとターゲットの通信」**を参照してください。

1. USB/ 電源ケーブルを接続する。
2. デバッガとターゲットを通信ケーブルで接続する (RJ11 プラグを使う場合) か、6 ピン インラインヘッダに直接接続する。

図 3-1: 通信ケーブルと USB/ 電源ケーブルの接続



© 2014 Microchip Technology Inc.

3.5 ターゲットボードの設定

3.5.1 量産デバイスを使う場合

量産デバイスを使う場合、デバッガをターゲットボードに直接接続します。PICkit 3 インサーキット デバッガ / プログラマでデバッグするには、ターゲットボード上のデバイスがデバッグ回路を内蔵している必要があります。必要なデバッグ回路を内蔵しているかどうかは、各デバイスのデータシートで確認してください (デバッグ回路を内蔵するデバイスには「バックグラウンド デバッガ イネーブル」コンフィグレーション ビットがあります)。

Note: 将来的には、ICD をサポートする回路を内蔵したデバイスも使用可能になる予定です。

ターゲットボードには、デバッガで使う通信方法に対応したコネクタが必要です。接続の詳細は、**セクション 2.4「デバッガとターゲットの通信」**の「標準 ICSP による通信」を参照してください。

3.5.2 ICE デバイスを使う場合

ICE デバイスを使う場合、ICE ヘッダボードが必要です。このヘッダボードは、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を実装しています。ICE ヘッダの詳細は、『Processor Extension Pak and Header Specification』(DS51292) を参照してください。

Note: 将来的には、ICD デバイス (デバイス名 -ICD) を搭載した ICD ヘッダボードも使用可能になる予定です。

ICE ヘッダをターゲットボードに接続するには、変換ソケットを使います。共通のヘッダを各種の表面実装パッケージに接続できるように、各種変換ソケットを用意しています。変換ソケットの詳細は、『Transition Socket Specification』(DS51194) を参照してください。

ヘッダボードのレイアウトは、ヘッダまたはプロセッサ拡張パックごとに異なります。接続の詳細は、**セクション 2.4「デバッガとターゲットの通信」**の「標準 ICSP による通信」を参照してください。

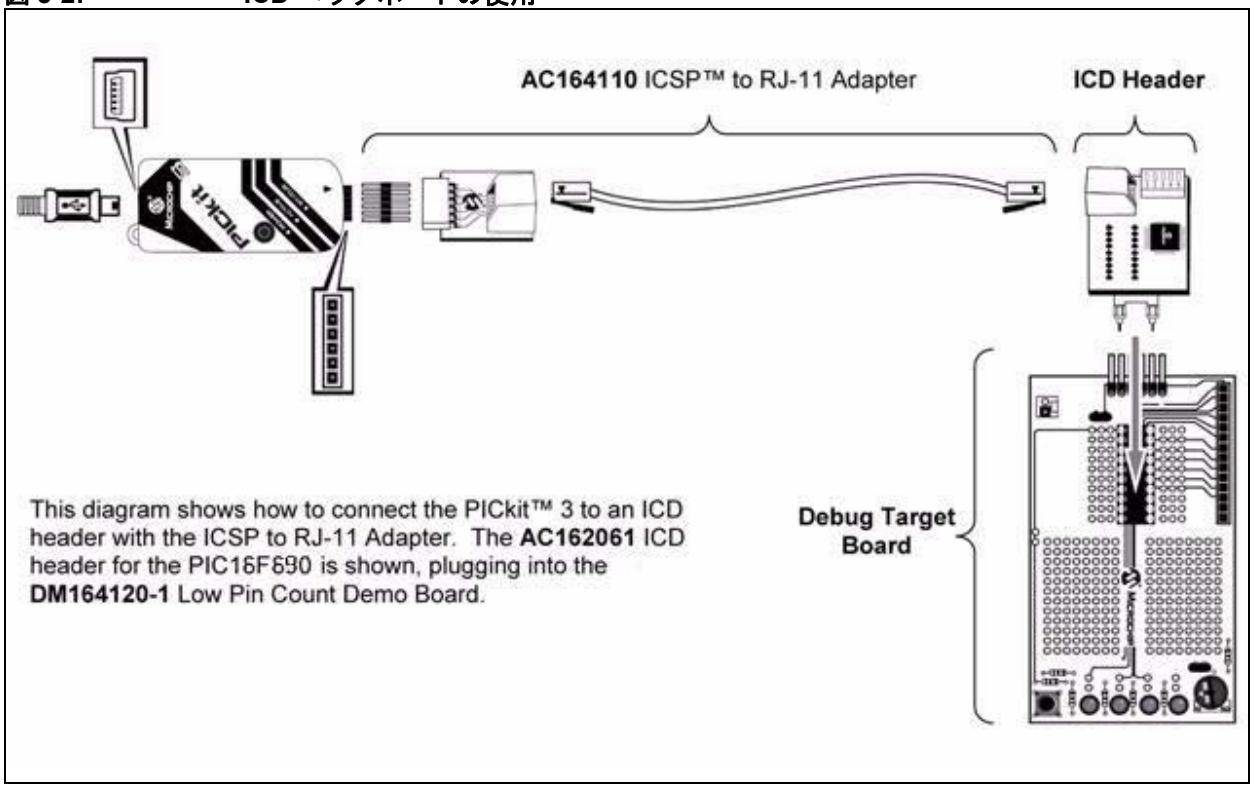
3.5.3 ICD ヘッダを使う場合

PIC MCU のベースライン全製品と一部のミッドレンジ製品の場合、デバッグ機能を利用するには特別な「–ICD」デバイスを実装したデバッグ ヘッダボードが必要です。これらのデバイスの種類と必要な ICD ヘッダボードの製品番号は、『Processor Extension Pak and Header Specification』(DS51292) を参照してください。『Processor Extension Pak and Header Specification』は、弊社ウェブサイト (www.microchip.com) で公開しています。

各 ICD ヘッダボードは「–ICD」デバイスを実装しており、量産デバイスの代わりにターゲットボードで使います。しかし、ほとんどのヘッダボードにはデバッグ用の RJ-11 コネクタが付いており、RJ-11/ICSP アダプタキット (AC164110) を使って PICkit 3 に接続します。図 3-2 に、PIC18F45K20 の ICD ヘッダ (AC162061) と AC164110 アダプタキット、少ピン デモボードを使った場合の接続を示します。

多くのミッドレンジ製品、PIC18、16 ビット製品には ICD ヘッダは不要です。ICSP プログラミング接続で直接デバッグが行えます。

図 3-2: ICD ヘッダボードの使用



3.5.4 ターゲットへの電源供給

以下に、電源供給方法に関する要点を記します。

- USB 接続を使う場合、PICkit 3 の電源は PC から供給できますが、電流量に制限があり、小型のターゲットボードに最大 30 mA (VDD 1.8 ～ 5 V) しか供給できません。
- 推奨するのは、ターゲットで VDD を自給する方法です。この方法だと、より大きな電流が得られます。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 X IDE がターゲットと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。

> **Note:** ターゲット電圧は、ICSP インターフェイスのドライバに対してのみ電源を供給します。PICkit 3 には電源供給を行いません。PICkit 3 の電源は USB ポートからのみ供給されます。

PICkit 3 とターゲットを適切なケーブルで接続します (セクション 3.4「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。PICkit 3 からターゲットに電源を供給する事もできます。

© 2014 Microchip Technology Inc.

3.6 MPLAB X IDE の設定

ハードウェアを接続して電源を投入したら、MPLAB X IDE で PICkit 3 インサーキット デバッガを使えるように設定します。

一部のデバイスでは、コンフィグレーション ビットで通信チャンネルを選択する必要があります (例 : PGC1/EMUC1、PGD1/EMUD1)。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事を確認します。

ソフトウェアのインストールとデバッガの設定に関する詳細は、「MPLAB X IDE Help」を参照してください。

3.7 デバッグの開始と停止

MPLAB X IDE 内でアプリケーションをデバッグするには、ソースコードを含むプロジェクトを作成する必要があります。プロジェクトを作成すると、以下に示す方法に従い、コードのビルド、デバイスへのプログラム、実行ができます。

- コードを実行するには、[Debug]>[Debug Project] または [Run] ツールバーの **[Debug Project]** のどちらかを選択します。
- コードを停止するには、[Debug]>[Pause] または [Debug] ツールバーの **[Pause]** のどちらかを選択します。
- コードの実行を再開するには、[Debug]>[Continue] または [Debug] ツールバーの **[Continue]** のどちらかを選択します。
- コードをステップ実行するには、[Debug]>[Step Into] または [Debug] ツールバーの **[Step Into]** のどちらかを選択します。スリープ命令をステップ実行しないように注意してください。エミュレーションを再開するのにプロセッサ リセットを実行する必要が生じます。
- コード行をステップオーバーするには、[Debug]>[Step Over] または [Debug] ツールバーの **[Step Over]** のどちらかを選択します。
- コード実行を終了するには、[Debug]>[Finish Debugger Session] または [Debug] ツールバーの **[Finish Debugger Session]** のどちらかを選択します。
- コードに対するプロセッサ リセットを実行するには、[Debug]>[Reset] または [Debug] ツールバーの **[Reset]** のどちらかを選択します。デバイスによっては POR/BOR、MCLR、システムリセット等、上記以外のリセットも実行できます。

3.8 プロセッサメモリとファイルの表示

MPLAB X IDEには、デバッグ情報や各種プロセッサメモリの情報を表示する複数のウィンドウがあり、[Window] メニューから選択できます。これらのウィンドウの使い方に関する詳細は、MPLAB X IDE のオンラインヘルプを参照してください。

- [Window]>[PIC Memory Views] - データ (RAM) とコード (ROM) のデバイスメモリを表示する。RAM、フラッシュ、特殊機能レジスタ (SFR)、CPU、コンフィグレーション ビットから選択する。
- [Window]>[Debugging] - デバッグ情報を表示する。変数、ウォッチ、コールスタック、ブレークポイント、ストップウォッチから選択する。

ソースコードを表示するには、[Projects] ウィンドウ内でソースコード ファイルを探してダブルクリックします。[Files] ウィンドウにソースコードが表示されます。このウィンドウには、選択したプロセッサとビルドツールに応じて、色分けされたコードが表示されます。色分けのスタイルを変更するには、[Tools]>[Options]、**[Fonts & Colors]**、**[Syntax]** タブを選択します。

エディタに関する詳細は、NetBeans ヘルプの [IDE Basics]>[Basic File Features] を参照してください。

NOTES:

© 2014 Microchip Technology Inc.

Chapter 4. PICkit 3 Debug Express

4.1 はじめに

PICkit 3 Debug Express キットを MPLAB X IDE アプリケーションと組み合わせると、プログラムの実行、停止、シングルステップ実行を行えます。複数のブレークポイントの設定、プロセッサのリセットも可能です。プロセッサを停止すると、レジスタの内容を観測、変更できます。

MPLAB X IDE の使い方は、以下のマニュアルを参照してください。

- 『MPLAB® X IDE User's Guide』(DS51519)
- MPLAB® X IDE オンラインヘルプ

4.2 PICkit 3 Debug Express キットの内容

PICkit 3 Debug Express キット (DV164131) には、以下のものが含まれます。

1. PICkit 3 インサーキット デバッガ / プログラマ
2. USB ケーブル
3. デバイス実装済みの 44 ピンデモボード

4.3 ハードウェアとソフトウェアのインストール

PICkit 3 のハードウェアとソフトウェアをインストールします。

> **Note:** PICkit 3 Debug Express を使うには、MPLAB IDE バージョン 1.20 以上が必要です。

4.3.1 予約済みリソース

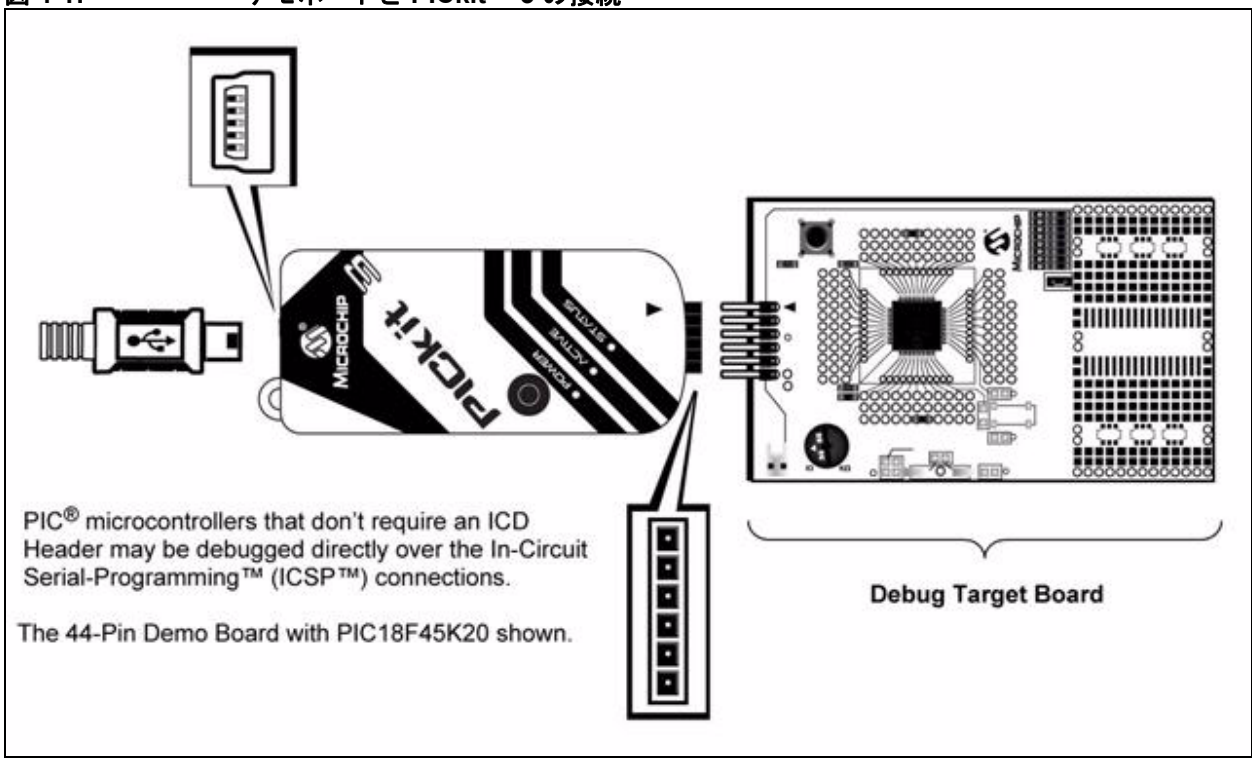
ICD デバイス内蔵のインサーキット デバッグ機能とデバッガの ICSP 機能を使うため、PICkit 3 Debug Express はデバッグ実行時、内部リソースの一部を使います。

各デバイスでデバッガが使うリソースの一覧を表示するには、MPLAB X IDE の [Start Page] で [Release Notes and Support Documentation]>[Reserved Resources for PICkit 3] をクリックします。

4.3.2 デモボードの接続

44 ピン デモボードに実装されている PIC18F45K20 は、図 4-1 に示すようにデモボードを PICkit 3 に接続するだけでデバッグできます (Explorer 16 ボードを使ったデバッグも可能です)。

図 4-1: デモボードと PICkit™ 3 の接続



4.3.3 コンフィグレーション ビットと Debug Express

ICD ヘッダなしで直接デバッグが可能な PIC マイクロコントローラには、コンフィグレーション ワードに $\overline{\text{DEBUG}}$ ビットがあり、このビットで PIC マイクロコントローラのデバッグモードの有効 / 無効を切り換えます。

MPLAB X IDE で PICkit 3 Debug Express を使う場合、このビットは自動的にセットされるため、ソースコードのコンフィグレーション設定では明示的に指定しないでください。

CAUTION

通常は、$\overline{\text{DEBUG}}$ コンフィグレーション ビットの値をソースコードのコンフィグレーション設定で指定しないでください。指定すると、このデバッガ以外でデバイスをプログラムする際にこのビットがアサートされる可能性があります。このビットがアサートされるとアプリケーション回路でデバイスが正しく動作しないか、または全く動作しなくなります。

PIC24 や dsPIC33 ファミリ等、多くの 16 ビットデバイスには、複数の ICSP プログラミング / デバッグポート ピン (例 : PGC1/EMUC1 と PGD1/EMUD1、PGC2/EMUC2 と PGD2/EMUD2) があります。プログラミングの場合は任意の ICSP ポートを使えますが、デバッグの場合は一度に 1 つのポートだけがアクティブになります。このアクティブな EMU ポートはデバイスのコンフィグレーション ビットで設定します。アクティブポートの設定が、PICkit 3 が接続されている EMU ポートに一致していないと、デバイスはデバッグモードに移行できません。MPLAB X IDE の [Configuration Bits] ダイアログでは、これらのビットを [Comm Channel Select] ビットと表示します。

© 2014 Microchip Technology Inc.

Chapter 5. PICkit 3 Programmer-To-Go

5.1 はじめに

PICkit 3 Programmer-To-Go を使うと、PIC MCU のメモリイメージを PICkit 3 本体にダウンロードして、後で PIC MCU にプログラムできます。つまり、この機能を使うとソフトウェアまたは PC がなくてもデバイスをプログラムできます。必要なのは PICkit 3 の USB 電源のみです。

Note: PICkit 3 にはプログラマとデバッガの機能がありますが、Programming-To-Go 機能を利用する場合はプログラマの機能のみが利用できます。Programming-To-Go ではデバッグ機能は利用できません。

本セクションでは、以下の内容について説明します。

- PICkit 3 Programmer-To-Go の USB 電源
- PICkit 3 Programmer-To-Go 対応デバイス
- PICkit 3 で Programmer-To-Go 機能を使うための設定
- PICkit 3 Programmer-To-Go の使用
- Programmer-To-Go モードの終了

5.2 PICkit 3 Programmer-To-Go の USB 電源

PICkit 3 プログラマ ハードウェアは、ターゲットの ICSP コネクタの VDD ピンから供給される電源だけでは動作しません。従って、PICkit 3 本体上部の USB mini-B ポートから 5 V 電源を供給する必要があります。これには、以下のようにいくつかの方法があります。

- PC の USB ポートまたは USB ハブのポート (通信ではなく電源としてのみ使用)
- 携帯型機器の USB ホストポート
- USB mini-B コネクタ対応の USB 電源アダプタ / 充電器 (自動車のシガレット ライター電源または家庭用コンセントから給電)
- 携帯電話 / モバイル機器用の USB mini-B コネクタ出力付きポータブル充電器 / 電源
- レギュレートされた 5 V を PICkit 3 の USB ポートに供給するカスタム バッテリパック

5.2.1 電源要件

USB 電源は以下の条件を満たす必要があります。

- PICkit 3 に 100 mA 以上の電流を供給する事
- レギュレートされた 4.5 ～ 5.5 V の電圧を安定的に出力できる事

Note 1: ほとんどのバッテリ内蔵型のポータブル充電器 / 電源は、内部のバッテリ電圧が低下して出力電圧が 4.5 V を下回った時にその事を示す手段がありません。従ってバッテリの残り容量には十分注意し、PICkit 3 に必ず 4.5 V 以上の電圧が供給されるようにしてください。

2: バッテリ駆動型の電源は、使わない時は PICkit 3 から外しておきます。PICkit 3 に接続したままにしておくとバッテリが消耗します。

5.3 PICkit 3 Programmer-To-Go 対応デバイス

PICkit 3 と MPLAB X IDE がサポートする下記のファミリのデバイスは全て Programmer-To-Go 機能にも対応しています。表 5-1 に、サポートされるデバイスファミリとそれぞれのプログラムメモリの制約を示します。

表 5-1: Programmer-To-Go 対応デバイス

デバイスファミリ	対応デバイス
ベースライン	全て [1]
ミッドレンジ	全て [1]
PIC18F	全て [1]
PIC18 J シリーズ	全て [1]
PIC18 K シリーズ	全て [1]
PIC24	全て [1,2]
dsPIC33	全て [1,2]
dsPIC30	全て [1]
dsPIC30 SMPS	全て [1]
PIC32	全て

Note 1: MPLAB X IDE がサポートする全てのファミリのデバイスをサポートします。アプリケーションがサポートするデバイス一覧を見るには、MPLAB X IDE の [Start page] で [Release Notes and Support Documentation]>[Device Support.htm] をクリックします。メモリ容量の大きいデバイスはサポートされています。

2: PICkit 3 の Programmer-To-Go 機能では、これらのデバイスの拡張 ICSP (Programming Executive) は使えません。これらのデバイスで PICkit 3 の Programmer-To-Go 機能を利用する場合、通常の ICSP でプログラミングが行われます。

© 2014 Microchip Technology Inc.

5.4 PICkit 3 で Programmer-To-Go 機能を使うための設定

メモリイメージを PICkit 3 にダウンロードして Programmer-To-Go 機能を利用する前に、PICkit 3 プログラマのソフトウェア オプションとバッファを Programmer-To-Go 用に設定する必要があります。必要なオプションを全て設定してまずソフトウェアからターゲット デバイスにプログラミングを実行し、デバイスに正しくプログラミングが行える事を確認してから Programmer-To-Go 用にイメージをダウンロードする事を強く推奨します。MPLAB X IDE を使ってデバイスをプログラムする方法は、MPLAB X IDE のオンラインヘルプを参照してください。

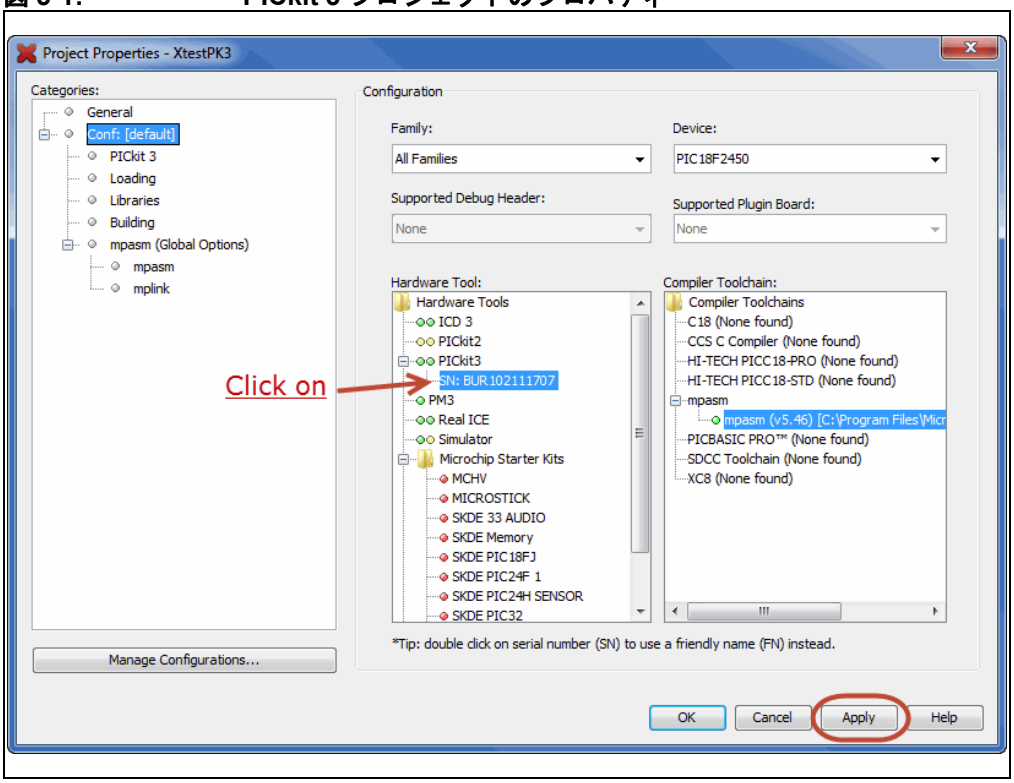
> **Note:** PICkit 3 は、量産プログラマではありません。開発専用です。Programmer-To-Go 機能を使うとフィールドでプログラムが可能となりますが、量産用でない事に注意してください。

5.4.1 Programmer-To-Go の設定

MPLAB X IDE で、以下の手順に従って Programmer-To-Go 機能を設定します。

1. プロジェクトをクリックして、[File]>[Project Properties] を選択する。ハードウェア ツールの領域で、プロジェクトで使う PICkit 3 のシリアル番号をクリックする。**[Apply]** をクリックする。

図 5-1: PICkit 3 プロジェクトのプロパティ

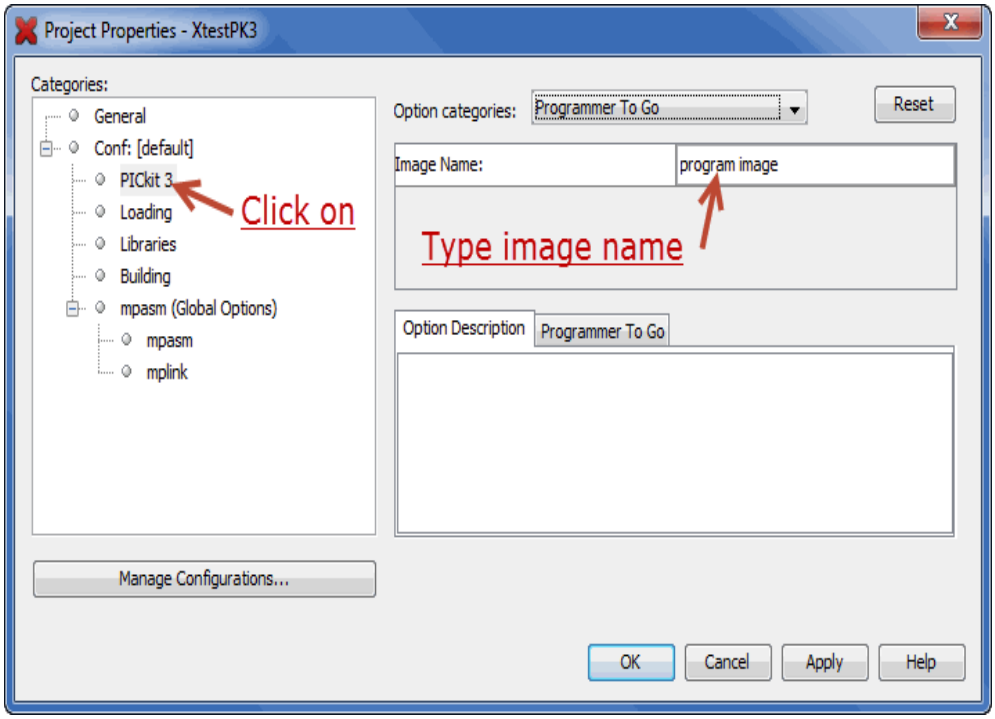


2. ダイアログの左側の [Categories] の下で、[PICkit 3] をクリックする。これで、[Options categories] ドロップダウン メニューから [Programmer To Go] を選択できる (図 5-2 参照)。

> **Note:** Programmer-To-Go モードではメモリ保持オプションをサポートしていません。

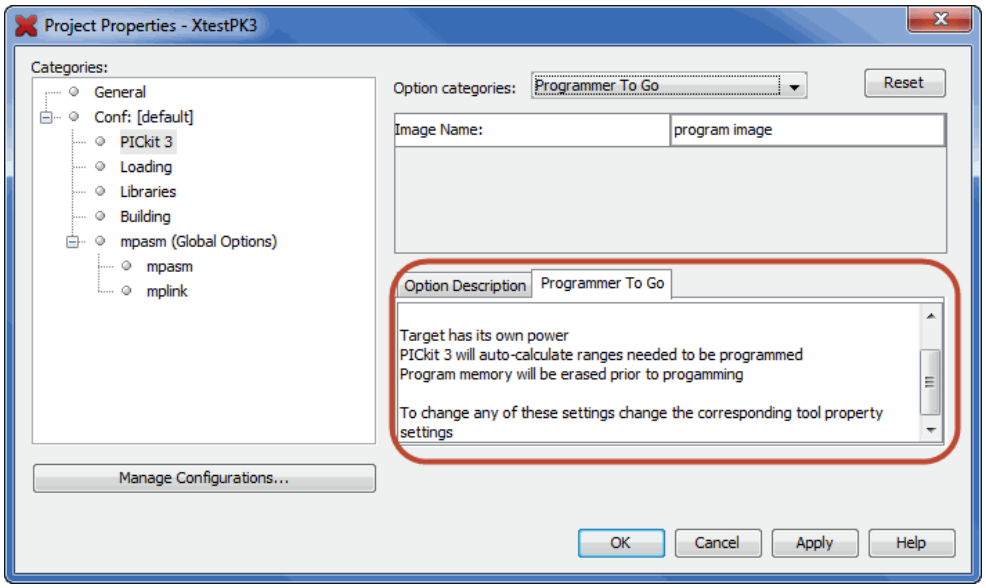
3. [Image Name] フィールドには、プログラミング イメージに使う名前を入力する。

図 5-2: イメージに名前を付ける



4. 図 5-3 に示す **[Programmer To Go]** タブをクリックすると、デバイスのプログラミングに関して選択した設定が表示される。これらの設定を変更する場合、[Project Properties] ダイアログを使う。

図 5-3: [Programmer-To-Go] タブ



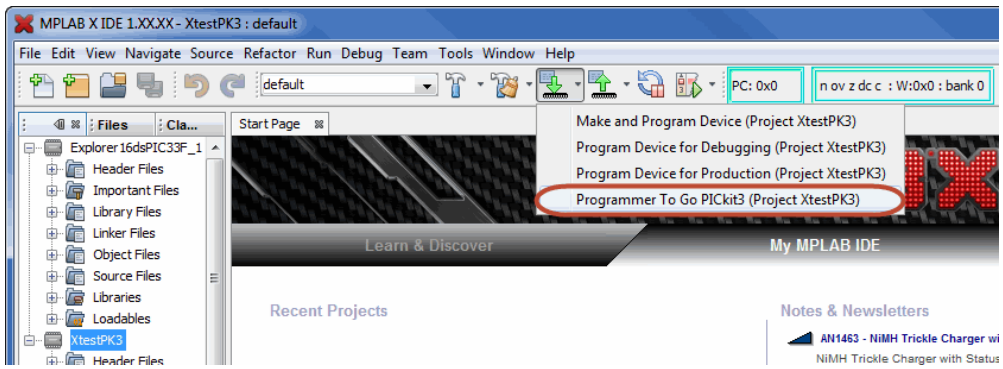
5. **[OK]** をクリックする。

© 2014 Microchip Technology Inc.

5.4.2 PICkit 3 へのダウンロードの完了

[Make and Program Device] アイコンの横の下矢印を使って [Programmer To Go PICkit3] を選択し (図 5-4 参照)、PICkit 3 へイメージ転送を実行します。イメージが PICkit 3 に格納されたら、MPLAB X IDE または PC がなくてもデバイスをプログラムできます。PICkit 3 のボタンを押すだけです。

図 5-4: PICkit 3 へのダウンロード



イメージのダウンロードが完了すると、[Output] ウィンドウにメッセージが表示されます。

> 「PICkit 3 is now in Programmer to go mode. The next time you connect to this unit, you will have the choice to take it out of Programmer to go mode.」
> (これで、PICkit 3 が Programmer-To-Go モードになりました。次にこのユニットに接続した時に、Programmer-To-Go モードを終了する事を選択できます。)

PICkit 3 本体の「ACTIVE」LED が点滅していれば、Programmer-To-Go モードへの移行が完了して書き込み準備が整った事を示します。

PC の USB ポートから PICkit 3 を取り外します。次に USB ケーブルから PICkit 3 に電源を供給すると「ACTIVE」LED が点滅し、PICkit 3 が Programmer-To-Go で起動した事を示します。

5.5 PICkit 3 Programmer-To-Go の使用

PICkit 3 の Programmer-To-Go の設定完了後、これを使ってターゲット デバイスをプログラムするには、以下の手順に従います。

1. **セクション 5.2「PICkit 3 Programmer-To-Go の USB 電源」**で説明した方法で、PICkit 3 に USB 電源を接続する。
2. PICkit 3 本体の「POWER」LED が点灯している事を確認する。PICkit 3 が Programmer-To-Go モードになり、プログラムする準備が完了すると、「ACTIVE」LED が 1 回点滅する。
3. PICkit 3 の ICSP コネクタをターゲットに接続する。PICkit 3 から電源を供給しない場合、ターゲットに適切に電源が供給されている事を確認する。
4. PICkit 3 のプッシュボタンを押すと、プログラム動作が開始する。

プログラム動作中は、PICkit 3 の「STATUS」LED がオレンジ色に変わり、動作が終了するまで点灯します。

プログラム動作が終了すると、PICkit 3 の LED がエラーの有無を示します。成功した場合は「STATUS」LED が緑に点灯します。プログラムエラーがあった場合は赤に点灯します。表 5-2 に、LED 表示と意味を示します。

表 5-2: Programmer-To-Go 動作後の LED 表示と意味

LED のステータス		意味	
「ACTIVE」LED	「STATUS」LED	コード	説明
青 1 回点滅	緑	成功 / 準備完了	プログラム動作中にエラーは発生しませんでした。PICkit 3 の Programmer-To-Go 機能で再度書き込みが可能です。
消灯	赤 間隔をあけず繰り返し点滅 : ••••••••••••••	VDD/VPP エラー	PICkit 3 が VDD または VPP の電圧を所定の値に設定できませんでした。 PICkit 3 が VDD を供給していない場合、VPP エラーです。 VDD と VPP の詳細は、**セクション 2.5「通信の接続」**を参照してください。
消灯	赤 2 回ずつ繰り返し点滅 : •• •• •• •• ••	デバイス ID エラー	PICkit 3 がターゲットから予測しないデバイス ID を受信しました。 PICkit 3 の Programmer-To-Go 設定時に選択したデバイスとターゲット デバイスが一致する事を確認します。このエラーは、ICSP 接続に問題があり PICkit 3 がターゲットと通信できていない事を示している場合があります。ベースライン デバイスではこのエラーは発生しません。
消灯	赤 3 回ずつ繰り返し点滅 : ••• ••• ••• •••	ベリファイエラー	プログラム後にターゲットのベリファイに失敗しました。 ターゲットの VDD が要件を満たしている事を確かめます。ベースライン デバイスの場合、このエラーは ICSP 通信の問題を示す可能性があります。
消灯	赤 4 回ずつ繰り返し点滅 : •••• •••• ••••	内部エラー	Programmer-To-Go で予期しない内部エラーが発生しました。 このエラーがもう一度発生したら、PICkit 3 へのダウンロードをやり直してみてください。

Note: PICkit 3 のプッシュボタンを押すとエラーコードがクリアされ、新しいプログラム動作を開始できます。

© 2014 Microchip Technology Inc.

5.6 Programmer-To-Go モードの終了

Programmer-To-Go モードを終了するには、PICkit 3 を PC の USB ポートに接続し、MPLAB X IDE と通信します。[Run]>[Run Project] の順に選択すると、以下のメッセージが表示されます。

「PICkit 3 is in Programmer-to-go mode. The name of the stored image is: ____. Do you want to stay in Programmer-to-go mode? (Saying no will erase the image in the PICkit 3.)」
(これで、PICkit 3 が Programmer-To-Go モードになりました。格納されたイメージの名前は ____ です。Programmer-To-Go モードにとどまりますか？
([No] を選択すると PICkit 3 内のイメージが削除されます))

[No] を選択するとイメージが削除され、Programmer-To-Go モードを終了します。

NOTES:

© 2014 Microchip Technology Inc.

第 2 部 – トラブルシュート

Chapter 6. トラブルシュートの最初のステップ 47
Chapter 7. よく寄せられる質問 (FAQ)..... 49
Chapter 8. エラーメッセージ..... 53
Chapter 9. エンジニアリング技術ノート (ETN) 57

NOTES:

© 2014 Microchip Technology Inc.

Chapter 6. トラブルシュートの最初のステップ

6.1 はじめに

PICkit 3 インサーキット デバッガ / プログラマの動作に問題が生じた場合、まずこちらの内容を確認してください。

- 最初に確認する 5 つの項目
- デバッグに失敗する主な理由
- その他の確認事項

6.2 最初に確認する 5 つの項目

1. 使っているデバイスを確認してください。
   新しいデバイスを使う場合、より新しいバージョンの MPLAB X IDE へのアップグレードが必要な場合があります。つまり、黄色信号 = 未テストのサポート (ベータサポートレベル) の場合があるという事です。
2. Microchip 社のデモボードを使っていますか。それともお客様が設計したボードを使っていますか。
   通信の接続に必要な抵抗とコンデンサに関するガイドラインが守られていますか。
   Chapter 2.「動作」を参照してください。
3. ターゲットに電源が供給されていますか。
   30 mA を超える場合、デバッガからターゲットに電源を供給できません。
4. USB ハブを使っていますか。ハブに電源は供給されていますか。
   それでも問題が解決しない場合、ハブを使わずにデバッガを PC に直接接続してください。
5. デバッガに付属の標準通信ケーブル (RJ-11) を使っていますか。
   これよりも長いケーブルを使った場合、通信エラーが起こる可能性があります。

6.3 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グレーション ビットの設定を確認してください。
   外部オシレータを使っている場合、内部オシレータを試してください。内部 PLL を使っている場合、PLL の設定が正しい事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. V_{DD} 電圧がデバイス仕様外である。
   詳細は、デバイスのプログラミング仕様を参照してください。
4. デバッガと PC またはターゲットボードが物理的に接続されていない。
   通信ケーブルの接続を確認してください。
5. デバイスがコード保護されている。
   コード保護に関するコンフィグレーション ビットの設定を確認してください。
6. デバッガと PC の通信が中断された。
   MPLAB X IDE でデバッガに再接続します。

7. デバッグしようとしている量産デバイスにデバッグ機能がない。
   デバッグヘッダを使ってください (「推奨参考資料」に記載されている『Processor Extension Pak and Header Specification』参照)。
8. ターゲット アプリケーションが何らかの理由で破損したか、エラーがある。
   例えば、デバッガ版のリンカスクリプトではなく、正規のリンカスクリプトを使った場合です (例 : 18F8722i.lkr の代わりに 18F8722.lkr を使用)。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、ターゲットのパワーオン リセットを実行します。
9. コンフィグレーション ビットで正しい PGC/PGD ピンペアをプログラムしていない (複数の PGC/PGD ピンペアを備えたデバイスの場合)。
10. 他の設定が干渉してデバッグできない。
    ターゲットがコードを実行できないようなコンフィグレーション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事ができません。
11. ブラウンアウト検出電圧が、動作電圧 VDD よりも高い。
    この場合、デバイスはリセット状態に移行し、デバッグできません。
12. **Chapter 2.「動作」**の通信接続のガイドラインに従っていない。
13. 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。

6.4 その他の確認事項

1. 一過性のエラーの場合があります。
   同じ操作を繰り返してみてください。
2. プログラミングに関する一般的な問題の可能性もあります。
   実行モードに切り換えて、なるべくシンプルなアプリケーション (例 : LED 点滅プログラム)でターゲットへの書き込みテストを行ってください。このアプリケーションが動作しない場合、ターゲットの設定に何らかの問題がある事が分かります。
3. ターゲット デバイスが何らかの損傷 (例 : 過電流) 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事がよくあります。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. Microchip 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用意しています。
   PICkit 3 インサーキット デバッガ / プログラマが正しく動作するかどうかを検証するには、正常動作が確認済みのこれらのボードを使うのも 1 つの方法です。
5. デバッガの動作を確認して、アプリケーションを正しくセットアップしてください。
   詳細は、**Chapter 2.「動作」**を参照してください。
6. それでも解決しない場合は Microchip 社までお問い合わせください。

© 2014 Microchip Technology Inc.

Chapter 7. よく寄せられる質問 (FAQ)

7.1 はじめに

ここでは、PICkit 3 インサーキット デバッガ / プログラマ 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。

- 動作に関する FAQ
- 不具合に関する FAQ

7.2 動作に関する FAQ

- **PICkit 3 インサーキット デバッガ / プログラマは通信にデバイス内のどの機能を利用しているのですか。**
  PICkit 3 インサーキット デバッガ / プログラマは、ICSP インターフェイスを介してフラッシュ デバイスと通信します。そして、プログラムメモリまたはテストメモリにダウンロードしたデバッガ実行プログラムを使います。

- **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にどのような影響がありますか。**
  デバッガ実行プログラムは実行モードでは動作しないため、コード実行時にスループットの低下はありません。つまり、デバッガがターゲット デバイスの実行サイクルを「奪う」事はありません。

- **PICkit 3 インサーキット デバッガ / プログラマ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。**
  セクション「ツールの比較」を参照してください。

- **PICkit 3 インサーキット デバッガ / プログラマには、他のインサーキット エミュレータ / デバッガのような複雑なブレークポイントがありますか。**
  いいえ。しかし、特定のデータメモリ アドレスに格納された値またはプログラム アドレスに基づいてブレークを実行する事はできます。

- **PICkit 3 は光学的または電気的に絶縁されていますか。**
  いいえ。現在のシステムにフローティング電圧または高電圧 (120 V) を印加する事はできません。

- **標準ケーブルには何か制約はありますか。**
  標準の ICSP RJ-11 ケーブルは、15 Mbps を超えるクロック速度には対応していません。

- **PICkit 3 によってプログラムの実行速度は低下しますか。**
  いいえ。デバイスはデータシートに記載されている速度で動作します。

- **dsPIC DSC を任意の速度で動作させてデバッグを行う事はできますか。**
  PICkit 3 は、デバイスのデータシートに記載されている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。

- **6 ピン (LVP ピン) はどのような働きをしますか。**
  6 ピンは、LVP (Low-Voltage Programming) 接続用に予約済みです。

7.3 不具合に関する FAQ

- **デバイスのプログラミング後にベリファイを実行すると不合格になります。これはプログラミングの問題ですか。**

  実行を選択した場合 ([Run]>[Run Program])、デバイスはプログラミング直後に自動的に実行を開始します。従って、実行されたコードがフラッシュメモリを書き換えた場合、ベリファイが失敗する可能性があります。プログラミング後のコード実行を抑止するには、[Hold in Reset] を選択してください。

- **PC が省電力 / 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) 長時間使う場合、お使いの PC のオペレーティング システムの電源オプション設定画面で休止モードを無効にしておいてください。Windows XP の場合、[電源オプションのプロパティ] ダイアログ ボックスの [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、全ての USB サブシステム コンポーネントで全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 と PIC24F/H の場合、デバッガは周辺制御レジスタの予約済みビット (通常は bit 14 または 5) を Freeze ビットとして使います。レジスタ全体に書き込みを実行した時に、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`) を使います。

- **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。どのようにすれば原因を特定できますか。**

  以下の点を確認してください。

  - 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、以下のような `trap.c` 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);
}
        :
```

  - アサートを使う。

© 2014 Microchip Technology Inc.

- **コードのデバッグが完了しました。デバイスをプログラムしましたが、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事があります。デバッガをプログラマとして使って書き込めるのは、「通常」のデバイスのみです。通常のデバイスにも 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。
  - PICkit 3 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。PICkit 3 をデバッガとして選択してプログラミングを行うと、デバッグ実行プログラムがプログラムメモリに書き込まれ、デバッグ用のその他のデバイス機能がセットアップされます (**セクション「デバッグ開始までの操作手順」**参照)。完成した (リリース用) コードをプログラムする場合、PICkit 3 をプログラマに設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。完成した (リリース用) コードを書き込むには、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。

- **どのようにすればファームウェアを手動でダウンロードできますか。**

  以下のようにして、手動でダウンロードできます。[File]>[Project Properties] を選択します。[Categories] の下で [PICkit 3] をクリックし、[Option Categories] ドロップダウンから [Firmware] を選択します。[Use Latest Firmware] のチェックを外し、[Press to browse for a specific firmware version] をクリックします。[MPLABX\mplab_ide\mplablibs\modules\ext\PICKIT3.jar] の下の [Directories] ウィンドウ枠にある [Firmware File] を参照します。[Firmware Files] ウィンドウ枠で `.jam` ファイルを選択して、**[OK]** をクリックします。[Project Properties] ダイアログで **[Reset]** をクリックします。

- **ファームウェアのダウンロード中に誤って PICkit 3 を切断してしまいました。対処の方法を教えてください。**

  PICkit 3 を再接続します。エミュレータは、再起動できるように書き込み済みの内容を消去し始めます。この消去には、約 7 秒かかります。完了するまでお待ちください。処理中は全ての LED が ON になります。処理が完了すると MPLAB X IDE はデバイスを認識し、復帰プロセスすなわちファームウェアのダウンロードを開始します。

- **メモリウィンドウに変更が反映されません。**

  ウィンドウに変更を表示させるには、メモリを読み出す必要があります。

- **この「よく寄せられる質問」に記載されていない問題が発生しました。どうすれば良いでしょうか。**

  以下を参照してください。

  - **セクション「デバッガが使うリソース」**
  - **セクション「一般的な対処方法」**
  - **セクション「特定の状況で表示されるエラーメッセージ」**
  - **Chapter 9.「エンジニアリング技術ノート (ETN)」**

NOTES:

© 2014 Microchip Technology Inc.

Chapter 8. エラーメッセージ

8.1 はじめに

PICkit 3インサーキット デバッガ/プログラマは各種のエラーメッセージを生成します。特定の状況で表示されるエラーメッセージもありますが、一般的な対処法で解決できるものもあります。通常、エラーメッセージの下に説明がある場合はそれらをお読みください。説明を読んでも問題を解決できない場合、または説明がない場合、以下のセクションを参照してください。

- 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

8.2 特定の状況で表示されるエラーメッセージ

8.2.1 デバッガとターゲット間の通信エラー

Failed to send database (データベースの送信に失敗しました)

このエラーが表示された場合、以下の解決方法を試してください。

1. もう一度ダウンロードする (一過性のエラーの場合があるため)。
2. 最も番号の大きい `.jam` ファイルの手動ダウンロードを試みる。

これらの操作で問題が解決しない場合、セクション「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。

8.2.2 インストール ファイルの破損 / 期限切れエラー

Failed to download firmware (ファームウェアのダウンロードに失敗しました)

Hex ファイルが存在する場合、以下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、ファイルが破損している場合があります。MPLAB X 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、以下の解決方法を試してください。

MPLAB X IDE を再インストールする。

Unable to download debug executive (デバッグ実行プログラムをダウンロードできません)

デバッグ時にこのエラーが表示された場合、以下の操作を実行してください。

1. デバッガのデバッグツールとしての選択を解除する。
2. プロジェクトを閉じて MPLAB X IDE を終了する。
3. MPLAB X IDE を再起動して再度プロジェクトを開く。
4. デバッグツールとしてデバッガを選択し、再度ターゲット デバイスをデバッグしてみる。

Unable to download program executive (プログラム実行ファイルをダウンロードできません)

プログラムしようとしている時にこのエラーが表示された場合、以下の操作を実行してください。

1. デバッガのプログラマとしての選択を解除する。
2. プロジェクトを閉じて MPLAB X IDE を終了する。
3. MPLAB X IDE を再起動して再度プロジェクトを開く。
4. プログラマとしてデバッガを選択し、再度ターゲット デバイスをプログラムしてみる。

これらの操作で問題が解決しない場合、セクション「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。

8.2.3 デバッグ障害のエラー

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 デバイス側でデバッグの準備ができていません。コンフィグレーション ビットの設定を確認して、デバイスをプログラムしてから作業を続けてください)

デバイスをプログラムする前に実行しようとすると、このメッセージが表示されます。実行を試みた後にこのメッセージが表示された場合、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、セクション「デバッグ障害の対処方法」を参照してください。

The device is code protected (デバイスはコード保護されています)

操作 (読み出し、書き込み、ブランクチェック、ベリファイ) を行おうとしたデバイスのコード保護が有効になっており、コードの読み出しまたは変更が行えません。コード保護に関するコンフィグレーション 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グレーション ビットをコード内または [Configuration Bits] ウィンドウでセットまたはクリアします。その後で、デバイス**全体**を消去してから再プログラムします。

8.2.4 その他のエラー

PICkit 3 is busy. Please wait for the current operation to finish (PICkit 3 がビジーです。現在の動作が終了するまで待機してください)

PICkit 3 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、以下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解除を試みる。
2. [Halt] を選択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解除を試みる。
3. デバッガと PC の接続ケーブルを抜く。その後、もう一度デバッガの選択解除を試みる。
4. MPLAB X IDE を終了する。

8.3 一般的な対処方法

以下の一般的な対処方法で問題が解決する場合があります。

- 読み書きエラーの対処方法
- 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
- デバッグ障害の対処方法
- 内部エラーの対処方法

8.3.1 読み書きエラーの対処方法

読み書きエラーが発生した場合、以下の事項を確認してください。

1. [Abort] を押しましたか。この操作で読み書きエラーが発生する事があります。
2. 同じ操作を繰り返してみる (一過性のエラーの場合があるため)。
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事を確認する。デバイスに必要な電圧レベルは、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事 (PGC と PGD が接続されている事) を確認する。
5. 書き込みエラーの場合、[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事を確認する。
6. 使っているケーブルの長さが適切である事を確認する。

© 2014 Microchip Technology Inc.

8.3.2 デバッガとターゲットの通信エラーの対処方法

PICkit 3 インサーキット デバッガ / プログラマとターゲット デバイスの同期がとれていません。

1. **[Reset]** を選択してから同じ操作を再試行する。
2. 使っているケーブルの長さが適切である事を確認する。

8.3.3 デバッガと PC の通信エラーの対処方法

PICkit 3 インサーキット デバッガ / プログラマと MPLAB X IDE の同期がとれていません。

1. デバッガと PC の接続ケーブルを一度抜いて挿し直す。
2. デバッガに再接続する。
3. 同じ操作を繰り返してみる (一過性のエラーの場合があるため)。
4. インストールされている MPLAB X IDE のバージョンが、PICkit 3 インサーキット デバッガ / プログラマに読み込ま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る。**セクション「インストール ファイルが破損している場合の対処方法」**の手順に従う。

8.3.4 インストール ファイルが破損している場合の対処方法

この問題は、ほとんどの場合、MPLAB X IDE のインストールが不完全であるか、インストールしたファイルが破損した事が原因で発生します。

1. 全てのバージョンの MPLAB X IDE を PC からアンインストールする。
2. 使っているバージョンの MPLAB X IDE を再インストールする。
3. それでも解決しない場合は Microchip 社までお問い合わせください。

8.3.5 USB ポート通信エラーの対処方法

この問題は、ほとんどの場合、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事が原因で発生します。

1. PICkit 3 インサーキット デバッガ / プログラマを再接続する。
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事を確認する。
3. デバッガの [Settings] で適切な USB ポートが選択されている事を確認する。
4. 指定した USB ポートを他のデバイスが使っていない事を確認する。
5. USB ハブを使う場合、電源が供給されている事を確認する。
6. USB ドライバが読み込まれている事を確認する。

8.3.6 デバッグ障害の対処方法

PICkit 3 インサーキット デバッガ / プログラマでデバッグを実行できない場合、これには多くの原因があります。**セクション「デバッグに失敗する主な理由」**と**セクション「その他の確認事項」**を参照してください。

8.3.7 内部エラーの対処方法

内部エラーは想定外のエラーであり、通常は発生しません。これらは主に Microchip 社内の開発で使います。

主な原因はインストール ファイルの破損です (**セクション「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)。

システムリソースの枯渇が原因である可能性もあります。

1. システムを再起動してメモリを解放する。
2. HDD に十分な空き容量がある事、過度なフラグメンテーションが発生していない事を確認する。

それでも解決しない場合は Microchip 社までお問い合わせください。

NOTES:

© 2014 Microchip Technology Inc.

Chapter 9. エンジニアリング技術ノート (ETN)

PICkit 3 に関連する ETN は、以下の通りです。詳細は、www.microchip.com のサイトで PICkit 3 インサーキット デバッガのページを開き、ダウンロード セクションで ETN をクリックして参照してください。

- **ETN-32 PICkit 3 の低電圧動作に関する改修 :** アセンブリ番号 10-00424-R4 またはそれ以下に適用されます。

NOTES:

© 2014 Microchip Technology Inc.

第 3 部 – リファレンス

補遺 A. ハードウェア仕様 61
補遺 B. PICkit 3 の回路図 67
補遺 C. 改訂履歴 69

NOTES:

© 2014 Microchip Technology Inc.

補遺 A. ハードウェア仕様

A.1 はじめに

ここでは、PICkit 3 インサーキット デバッガ / プログラマ システムのハードウェア仕様と電気的仕様を詳しく説明します。

A.2 ハイライト

この章では以下について説明します。

- 適合宣言書
- USB ポート / 電源
- PICkit 3 インサーキット デバッガ / プログラマ
- 標準通信ハードウェア
- ターゲットボードに関する注意事項

A.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

PICkit 3 インサーキット デバッガ / プログラマ

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事を宣言します。

規格 : EN61010-1 試験所用電気機器
Microchip Technology, Inc.
日付 : 2009 年 1 月

PICkit 3 インサーキット デバッガ / プログラマの使用に関する重要な注意事項

PICkit 3 インサーキット デバッガ / プログラマは、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等あらゆる種類の機器の動作に影響を与える事があるため注意してください。

従って、上記欧州規格に適合するためには、以下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) のみに限定する事。
2. 無線受信機やテレビ等、電磁波の影響を受ける可能性のある機器から 20 メートル以内の場所で本システムを使わない事。

A.4 USB ポート / 電源

PICkit 3 インサーキット デバッガ / プログラマは、mini USB 2.0 ポートを介してホスト PC に接続します。USB コネクタはポッドの上部にあります。

システムはUSBインターフェイスを利用してファームウェアを再読み込みできます。

システムの電源は USB インターフェイスから供給します。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どちらの場合も 100 mA をやや上回る電力を USB から供給する必要があります。

Note: PICkit 3 インサーキット デバッガ / プログラマには、USB 接続を介して電源を供給します。ターゲットボードには他の専用電源から給電します。ただし、ターゲットの消費電流が 30 mA 未満の場合に限り、PICkit 3 から電源を供給する事もできます。

ケーブル長 — PC とデバッガを接続するケーブルは、正常動作に適した長さのものをキットに同梱しています。

セルフパワードハブ — USB ハブを使う場合、バスパワードではなくセルフパワードタイプを使います。PC キーボード内蔵の 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 の休止 / 省電力モード — USB 接続による PC とデバッガの通信を確保するため、休止モード等 PC の省電力モードを無効にします。

A.5 PICkit 3 インサーキット デバッガ / プログラマ

デバッガの筐体はメインボードを格納しており、USB コネクタとシングル インライン コネクタが付いています。デバッガの筐体にはインジケータ ライト (LED) もあります。

A.5.1 メインボード

メインボードには、インターフェイス プロセッサ、USB 2.0 インターフェイス、内蔵フラッシュ エミュレーション デバイスへのプログラミング用の SPI シリアル EE、インジケータ LED が実装されています。

A.5.2 インジケータ ライト (LED)

インジケータ ライトの意味は以下の通りです。

LED	カラー	説明
POWER	緑	電源を投入またはターゲットを接続すると点灯します。
ACTIVE	青	PICkit™ 3 と PC の接続が確立されている時、またはコマンド送受信時に点灯します。
STATUS	緑	デバッガが正常に動作しており、スタンバイ状態の時に点灯します。
	橙	動作中 (ビジー) の時に点灯します。
	赤	デバッガにエラーが発生すると点灯します。

© 2014 Microchip Technology Inc.

A.6 標準通信ハードウェア

デバッガとターゲットの標準通信（ **セクション「デバッガとターゲットの通信」**の「標準 ICSP による通信」) には、RJ-11 コネクタ付きのアダプタを使います。

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) が必要になる事があります。

> **Note:** 旧式のヘッダボードは 6 ピンの SIL コネクタではなく 6 ピンの RJ-11 コネクタを使っているため、デバッガと直接接続できます。

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

A.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 インターフェイスとなるものです。これは、ターゲット デバイスへの接続と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、データ接続を含みます。

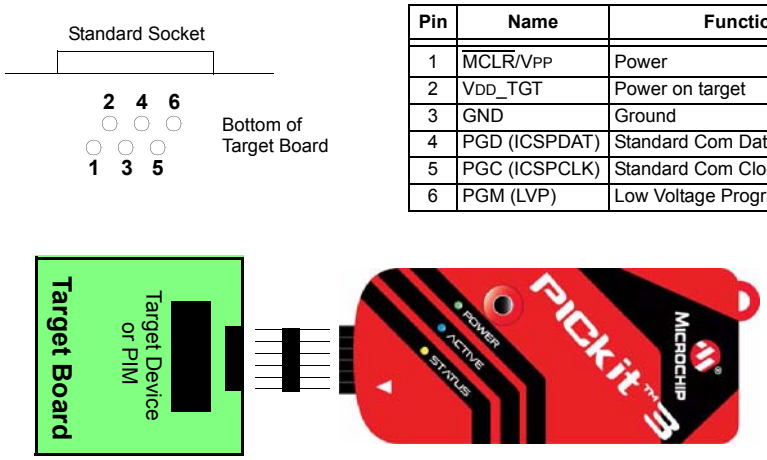
VPP の高電圧ラインは、エミュレーション プロセッサの電圧要件に応じて 1.8 ～ 14 V の電圧を生成できます。

VDD 検出ラインには、ターゲット プロセッサからの電流が流れます。

クロックとデータの接続は、以下の特性を備えたインターフェイスです。

- 高インピーダンス モードのクロックおよびデータ信号 (PICkit 3 インサーキット デバッガ / プログラマ 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
- プロトタイプまたはターゲット　システムの電気的短絡による過電流からクロックとデータ信号を保護

図 A-1: 6 ピン標準ピン配置



Pin	Name	Function
1	$\overline{\text{MCLR}}$/VPP	Power
2	VDD_TGT	Power on target
3	GND	Ground
4	PGD (ICSPDAT)	Standard Com Data
5	PGC (ICSPCLK)	Standard Com Clock
6	PGM (LVP)	Low Voltage Programming

A.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。

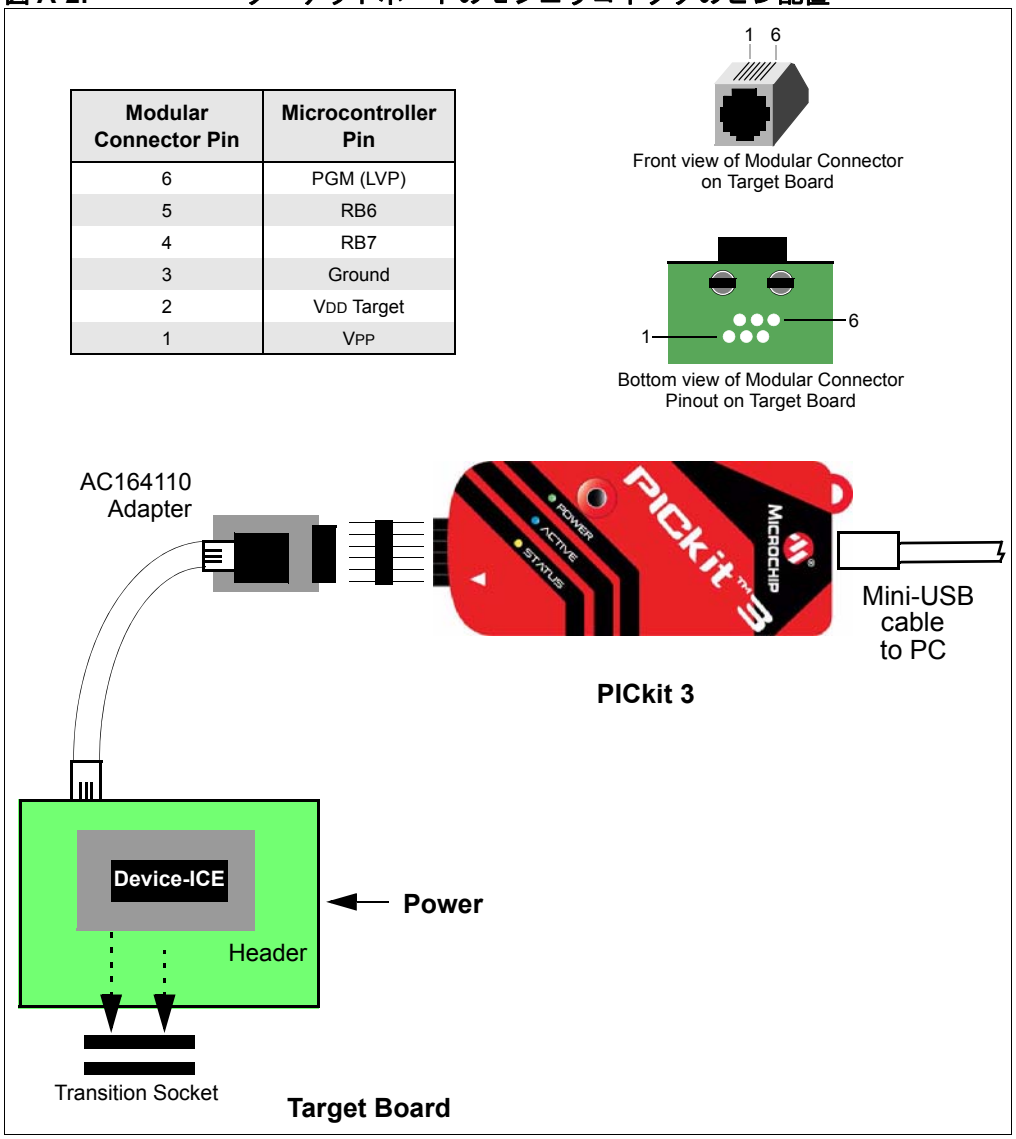
A.6.2.1 モジュラコネクタの仕様

- 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

図 A-2 の表に、アプリケーション側のモジュラコネクタの各ピンとマイクロコントローラのピンの対応を示します。この構成では、ICD の機能を全て利用できます。

図 A-2: ターゲットボードのモジュラコネクタのピン配置

Modular Connector Pin	Microcontroller Pin
6	PGM (LVP)
5	RB6
4	RB7
3	Ground
2	VDD Target
1	VPP



© 2014 Microchip Technology Inc.

A.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

A.6.2.3 モジュラケーブルの仕様

- メーカー名、製品番号 - Microchip Technology 社、07-00024

A.7 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 (1.8 ～ 5.0 V) と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何点かの注意事項があります。詳しくは以下のセクションを参照してください。

- **セクション「ターゲット回路との接続」**
- **セクション「デバッガの正常動作を妨げる回路」**

NOTES:

© 2014 Microchip Technology Inc.

補遺 B. PICkit 3 の回路図

ここでは、PICkit 3 インサーキット デバッガ / プログラマの回路図を示します。デモボードの回路図は、各ボードのユーザガイドを参照してください。

図 B-1: PICkit™ 3 の回路図 (1/2 ページ)

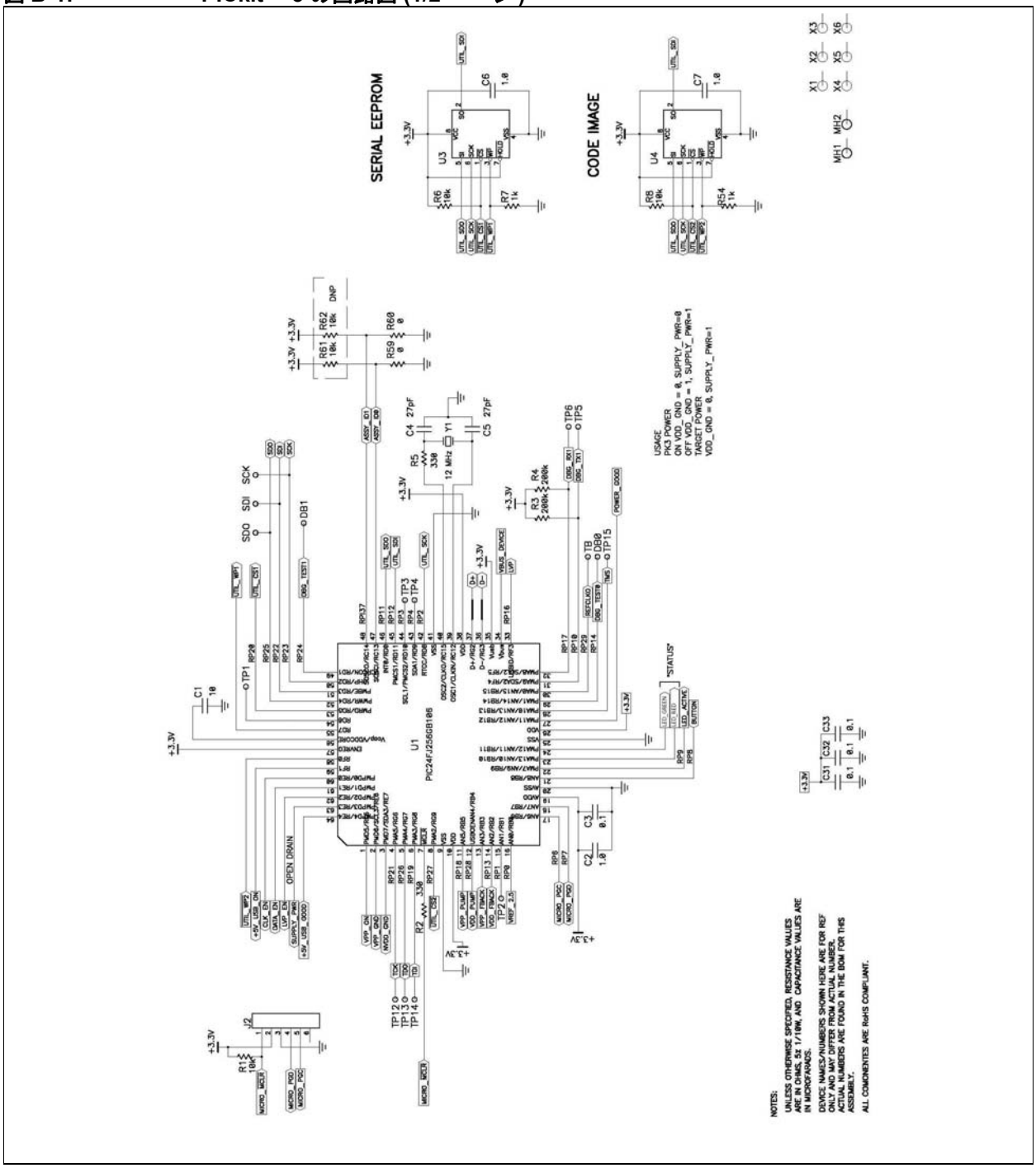
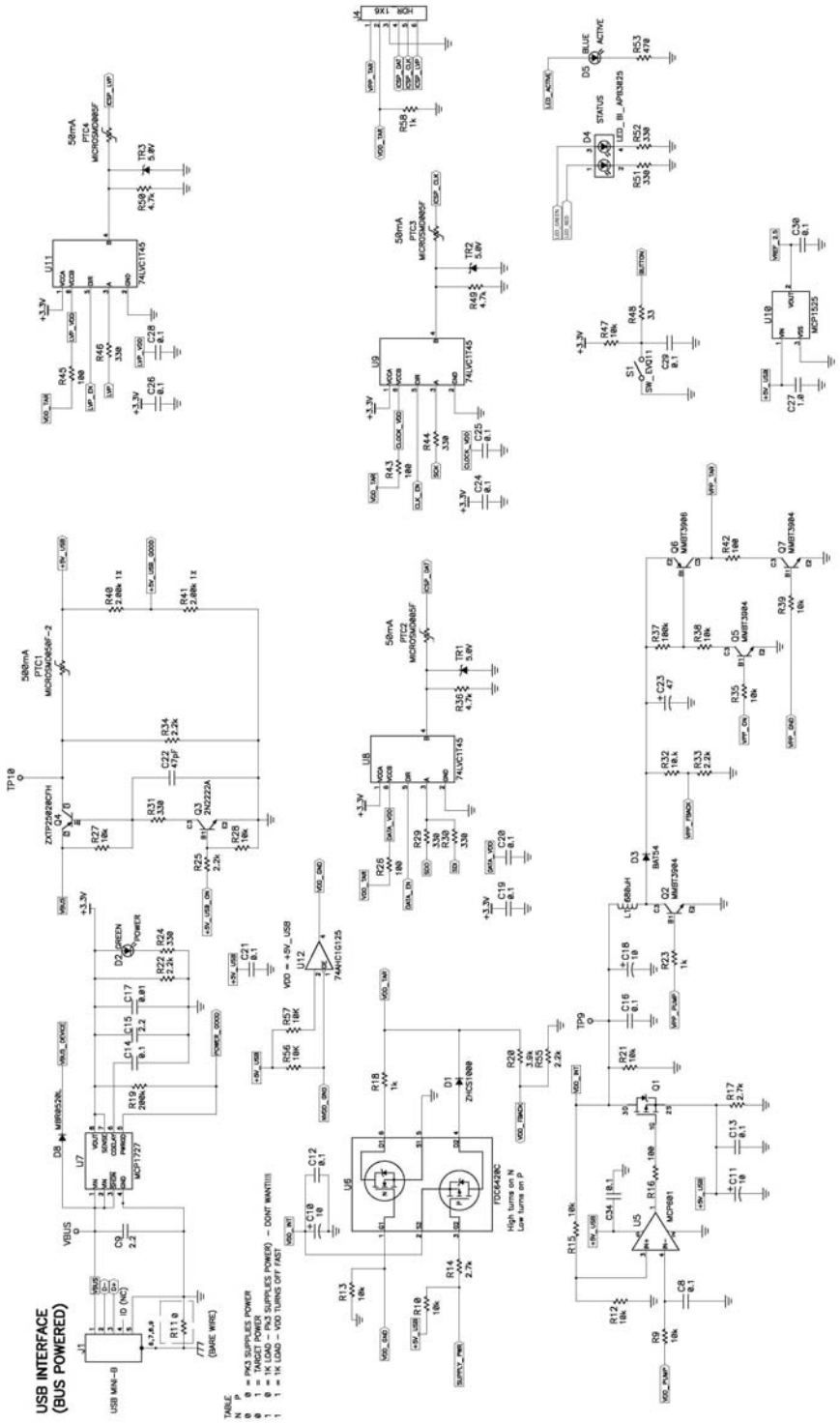


図 B-2: PICkit™ 3 の回路図 (2/2 ページ)



© 2014 Microchip Technology Inc.

補遺 C. 改訂履歴

リビジョン A (2013 年 4 月)

本書の初版

NOTES:

© 2014 Microchip Technology Inc.

用語集

記号、数字

[Watches] ウィンドウ (Watch Window)

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。

16 進数 (Hexadecimal)

0 ～ 9 の数字と A ～ F (または a ～ f) のアルファベットを使う、16 を底とした記数法。A ～ F で 10 進数の 10 ～ 15 を表現する。一番右の桁が 1 の位、次の桁が 16 の位、その次の桁が 16^2 = 256 の位を表す。

2 進数 (Binary)

0 と 1 の数字を使う、2 を底とした記数法。一番右の桁が 1 の位、次の桁が 2 の位、その次の桁が 2^2 = 4 の位を表す。

8 進数 (Octal)

0 ～ 7 の数字のみを使う、8 を底とした記数法。一番右の桁が 1 の位、次の桁が 8 の位、その次の桁が 8^2 = 64 の位を表す。

A

AND 条件ブレークポイント (ANDed Breakpoints)

プログラム実行を停止するために設定する 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)。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。

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、制御文字等を含む。

C

C\C++

C 言語は、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等を特長とする汎用プログラミング言語。C++ とは、C 言語のオブジェクト指向バージョン。

COFF

Common Object File Format の略。このフォーマットのオブジェクト ファイルは、マシンコードの他、デバッグ等に関する情報を含む。

CPU

「中央演算処理装置」参照。

D

DWARF

Debug With Arbitrary Record Format の略。ELF ファイルのデバッグ情報フォーマット。

E

EEPROM

Electrically Erasable Programmable Read Only Memory の略。電気的に消去可能なタイプのPROM。データの書き込みと消去をバイト単位で行う。EEPROMは電源をOFFにしても内容を保持する。

ELF

Executable and Linking Format の略。この形式のオブジェクト ファイルはマシンコードを含む。デバッグその他の情報は DWARF で指定する。ELF/DWARF の方が COFF よりも最適化したコードのデバッグに適している。

EPROM

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

F

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空間の次の命令をプリフェッチする。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。

G

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、汎用目的に使える部分。

H

Halt

プログラム実行を停止する事。Halt を実行する事は、ブレークポイントで停止する事と同じ。

HEX コード /HEX ファイル (Hex Code/Hex File)

HEX コードは、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 ファイルは、HEX コードを格納したファイル。

I

ICE/ICD

インサーキット エミュレータ / インサーキット デバッガの略。ターゲットデバイスのデバッグとプログラミングを行うためのハードウェア ツール。エミュレータは、デバッガよりも多くの機能 (トレース等) を備える。

インサーキット エミュレーション/インサーキット デバッグとは、インサーキット エミュレータまたはデバッガを使った作業の事を指す。

-ICE/-ICD: インサーキット エミュレーション / デバッグ用の回路を内蔵したデバイス (MCU または DSC)。このデバイスは必ずヘッダボードにマウントし、インサーキット エミュレータまたはデバッガによるデバッグ用に使う。

© 2014 Microchip Technology Inc.

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。Microchip 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 の略。MPLAB IDE/MPLAB X IDE の IDE と同じ意味。

IEEE

Institute of Electrical and Electronics Engineers の略。

L

LVDS

Low Voltage Differential Signaling の略。銅線を使って、低ノイズ、低消費電力、低振幅でデータを高速伝送 (Gbps) する方法。

標準の I/O シグナリングでは、データストレージは実際の電圧レベルに依存する。電圧レベルは信号線の長さによって影響を受ける (信号線が長いと抵抗が増大し、電圧が低下する)。これに対し LVDS では、データストレージは実際の電圧レベルでなく正と負の電圧値によってのみ区別する。従って、長い信号線でもクリアで安定したデータストリームを維持した伝送が可能。

出典 : http://www.webopedia.com/TERM/L/LVDS.html

M

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。

make ファイル (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/MPLAB X 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う。

MCU

Microcontroller Unit の略。マイクロコントローラの事。「µC」と表記する事もある。

MPASM™ アセンブラ (MPASM Assembler)

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

MPLAB (言語ツール名) **for** (デバイス名) **(MPLAB Language Tool for Device)**

特定のデバイスに対応した Microchip 社の C コンパイラ、アセンブラ、リンカ。言語ツールは、アプリケーションで使うデバイスに対応したものを選択する必要がある。例えば PIC18 MCU 用の C コードを作成する場合は「MPLAB C Compiler for PIC18 MCU」を使う。

MPLAB ICD

MPLAB IDE/MPLAB X IDE と連携する Microchip 社のインサーキット デバッガ。「ICE/ICD」参照。

MPLAB IDE/MPLAB X IDE

Microchip 社の統合開発環境。エディタ、プロジェクト マネージャ、シミュレータが付属する。

MPLAB PM3

Microchip 社のデバイス プログラマ。PIC18 マイクロコントローラと 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/MPLAB X IDE との併用も、単体での使用も可能。PRO MATE II の後継製品。

MPLAB REAL ICE インサーキット エミュレータ (MPLAB REAL ICE In-Circuit Emulator)

MPLAB IDE/MPLAB X IDE と組み合わせて使う Microchip 社の次世代インサーキット エミュレータ。「ICE/ICD」参照。

MPLAB SIM

MPLAB IDE/MPLAB X IDE と組み合わせて使う Microchip 社のシミュレータで、PIC MCU と dsPIC DSC に対応する。

MPLAB Starter Kit for (デバイス名) **(MPLAB Starter Kit for Device)**

特定のデバイスでの作業を開始する上で必要となるものを全てセットにした Microchip 社のスタータキット。既製のアプリケーションの動作を確認した後で、一部を変更してカスタム アプリケーションとしてデバッグとプログラムを行う。

MPLIB™ オブジェクト ライブラリアン (MPLIB Object Librarian)

MPLAB IDE/MPLAB X IDEと組み合わせて使うMicrochip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 C18 C コンパイラで作成した COFF オブジェクト モジュールに使うオブジェクト ライブラリアン。

MPLINK™ オブジェクト リンカ (MPLINK Object Linker)

Microchip 社の MPASM アセンブラと C18 C コンパイラに対応したオブジェクト リンカ。Microchip 社の MPLIB ライブラリアンとの併用も可能。MPLAB IDE/MPLAB X IDE との併用を前提に設計されているが、必須ではない。

MRU

Most Recently Used の略。最近使ったファイルとウィンドウの事。MPLAB IDE/MPLAB X IDE のメインメニューで選択できる。

N

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。

O

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。

P

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 (PIC MCUs)

Microchip 社の全てのマイクロコントローラ ファミリの総称。

PICkit 2/3

Microchip 社の開発用デバイス プログラマで、Debug Express によるデバッグ機能を備える。サポートしているデバイスの種類は、各ツールの Readme ファイル参照。

© 2014 Microchip Technology Inc.

Psect

GCC のセクションに相当する OCG の用語。プログラム セクション (program section) の略語。リンカが 1 つのまとまりとして処理するコードまたはデータのブロック。

PWM 信号 (PWM Signals)

パルス幅変調 (Pulse Width Modulation) 信号。一部の PIC MCU は周辺モジュールとして PWM を内蔵している。

R

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

エミュレータを Halt から解放するコマンド。エミュレータ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。

S

Serialized Quick Turn Programming (SQTP)

デバイス プログラマでマイクロコントローラをプログラムする際に、各デバイスに異なるシリアル番号を書き込めるようにする機能。エントリコード、パスワード、ID 番号等を書き込む目的で使う。

SQTP

「Serialized Quick Turn Programming」参照。

Step Into

Single Step と同じコマンド。Step Over とは異なり、Step Into では CALL 命令が呼び出すサブルーチン内もステップ実行する。

Step Over

Step Over を実行すると、サブルーチン内をステップ実行せずにコードをデバッグできる。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等、正しくリターンしない場合、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。

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を全て実行し、サブルーチンのリターンアドレスで実行が停止する。

U

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イス規格。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 (ハイスピード USB) は最大 480 Mbps のデータレートをサポートしている。

V

Volatile

メモリ内の変数へのアクセス方法に影響を与えるコンパイラの最適化を抑制する変数修飾子。

あ

アーカイブ / アーカイバ (Archive/Archiver)

アーカイブ / ライブラリは、再配置可能なオブジェクト モジュールの集まり。複数のソースファイルをオブジェクト ファイルにアセンブルした後、アーカイバ / ライブラリアンを使ってこれらオブジェクト ファイルを 1 つのアーカイブ / ライブラリ ファイルにまとめると生成される。アーカイブ / ライブラリをオブジェクト モジュールや他のアーカイブ / 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 エントリポイント (Access Entry Points)

リンク時に定義されていない可能性のある関数に、セグメントの境界を越えて制御を渡すための手段。ブートセグメントとセキュア アプリケーション セグメントを別々にリンクする方法を提供する。

アクセスメモリ (Access Memory)

PIC18 のみ - PIC18 でバンクセレクト レジスタ (BSR) の設定に関わらずアクセスできる特殊なレジスタ。

アセンブリ / アセンブラ (Assembly/Assembler)

アセンブリとは、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。アセンブラとは、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。

アップロード (Upload)

エミュレータやプログラマ等のツールからホスト PC へ、またはターゲットボードからエミュレータへデータを転送する事。

アドレス (Address)

メモリ内の位置を一意に特定する値。

アプリケーション (Application)

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 (Alphabetic Character)

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 (Event)

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

入れ子の深さ (Nesting Depth)

マクロに他のマクロを含める事のできる階層の数。

インポート (Import)

Hex ファイル等の外部ソースから MPLAB IDE/MPLAB X IDE にデータを取り込む事。

ウォッチドッグ タイマ (WDT)

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、設定はコンフィグレーション ビットで行う。

ウォッチ変数 (Watch Variable)

デバッグ セッション中に [Watches] ウィンドウで観察できる変数。

© 2014 Microchip Technology Inc.

英数字 (Alphanumeric)

アルファベット文字と 0 ～ 9 の 10 進数の数字の総称 (0,1,•••,9)。

永続データ (Persistent Data)

クリアも初期化もされないデータ。デバイスをリセットしてもアプリケーションがデータを保持できるようにするために使う。

エクスポート (Export)

MPLAB IDE/MPLAB X IDE のデータを標準フォーマットで外部に出力する事。

エピローグ (Epilogue)

コンパイラで生成したコードのうち、スタック領域の割り当て解除、レジスタの復帰、ランタイムモデルで指定したその他のマシン固有の要件を実行するコード部分。関数のユーザコードの後、関数リターンの直前にエピローグを実行する。

エミュレーション / エミュレータ (Emulation/Emulator)

「ICE/ICD」参照。

エラー / エラーファイル (Error/Error File)

プログラムの処理を継続できない問題が発生するとエラーとして報告される。可能な場合、エラーは問題が発生したソースファイル名と行番号を特定する。エラーファイルは、言語ツールから出力されたエラーメッセージと診断結果を格納する。

演算子 (Operators)

定義可能な式を構成する際に使う「+」や「-」等の記号。各演算子に割り当てられた優先順位に基づいて式を評価する。

エンディアン (Endianness)

マルチバイト オブジェクトにおけるバイトの並び順。

オブジェクト コード / オブジェクト ファイル (Object Code/Object File)

オブジェクト コードとは、アセンブラまたはコンパイラが生成するマシンコードの事。オブジェクト ファイルとは、マシンコードを格納したファイル。デバッグ情報を含む事もある。そのまま実行できるものと、他のオブジェクト ファイル (例 : ライブラリ) 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 (Object File Directives)

オブジェクト ファイル作成時にのみ使うディレクティブ。

オフチップメモリ (Off-Chip Memory)

PIC18 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うか、または全てのプログラムメモリをエミュレータから供給する。[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。

オペコード (Opcodes)

Operational Code の略。「ニーモニック」参照。

か

外部 RAM (External RAM)

オフチップの読み書き可能なメモリ。

外部シンボル (External Symbol)

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 (External Symbol Resolution)

リンカが全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、全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、リンカエラーとなる。

外部入力ライン (External Input Line)

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 (External Label)

外部リンケージを持つラベル。

外部リンケージ (External Linkage)

関数または変数が、それ自身を定義し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。

拡張マイクロコントローラ モード (Extended Microcontroller Mode)

拡張マイクロコントローラ モードでは、内蔵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 PIC18 の内部メモリ空間より大きい場合、自動的に外部メモリの実行に切り換わる。

拡張モード (Extended Mode) (PIC18 MCU)

コンパイラの動作モードの 1 つ。拡張命令 (`ADDFSR`、`ADDULNK`、`CALLW`、`MOVSF`、`MOVSS`、`PUSHL`、`SUBFSR`、`SUBULNK`) とリテラル オフセットによるインデックス アドレス指定を利用できる。

環境 (Environment)

MPLAB PM3 - 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。

クリーン (Clean)

クリーンする事により、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、デバッグファイル等、全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。

緩和 (Relaxation)

ある命令を、機能が同じでよりサイズの小さい命令に変換する事。コードサイズを抑えるために便利である。最新の MPLAB ASM30 には、CALL 命令を RCALL 命令に緩和する機能がある。現在の命令から +/- 32k 命令ワード以内にあるシンボルを呼び出す場合にこの変換を行う。

記憶域クラス (Storage Class)

指定されたオブジェクトを格納する記憶場所の持続期間を決定する。

記憶域修飾子 (Storage Qualifier)

宣言されるオブジェクトの特別な属性を示す (例 : `const`)。

基数 (Radix)

アドレスを指定する際の記数法 (16 進法、10 進法) の底。

警告 (Warning)

MPLAB IDE/MPLAB X IDE - 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、ユーザに注意を促すために表示されるメッセージ。

16 ビットアセンブラ / コンパイラ - 問題となる可能性のある状態を警告として報告するが、処理は停止されない。MPLAB C30 の警告メッセージはソースファイル名と行番号を報告するが、エラーメッセージと区別するために「`warning:`」の文字列も付加する。

高級言語 (High Level Language)

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 (Calibration Memory)

PIC マイクロコントローラの内蔵 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またはレジスタ。

© 2014 Microchip Technology Inc.

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、多くのテクノロジとビジネス関連の標準規格の策定を行っている団体。一般的に ISO と呼ぶ。

コマンドライン インターフェイス (Command Line Interface)

プログラムとユーザ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 (Compiler)

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。

コンパイルド スタック (Compiled Stack)

コンパイラが管理するメモリの領域で、この領域内で変数に静的に空間を割り当てる。ターゲット デバイス上にソフトウェア スタックまたはハードウェア スタックのメカニズムを効率的に実装できない場合、コンパイルド スタックがソフトウェア スタックまたはハードウェア スタックに置き換わる。

コンフィグレーション ビット (Configuration Bits)

PIC MCU と dsPIC DSC の動作モードを設定するために書き込む専用ビット。コンフィグレーション 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。

さ

再帰 (Recursion)

定義した関数またはマクロがそれ自身を呼び出す事。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ため注意が必要。

再帰呼び出し (Recursive Calls)

直接または間接的に自分自身を呼び出す関数。

再入可能 (Reentrant)

1 つの関数を複数呼び出して同時に実行できる事。直接または間接再帰、あるいは割り込み処理中の実行によって起こる事がある。

再配置 (Relocation)

リンカが絶対アドレスを再配置可能セクションに割り当てる事。再配置可能セクション内の全てのシンボルを新しいアドレスに更新する。

再配置可能 (Relocatable)

アドレスがメモリの固定番地に割り当てられていないオブジェクト。

再配置可能セクション (Relocatable Section)

16 ビットアセンブラ - アドレスが固定されていない (絶対アドレスでない) セクション。再配置可能セクションには、再配置と呼ばれるプロセスによって、リンカがアドレスを割り当てる。

左辺値 (L-value)

検査または変更が可能なオブジェクトを指し示す式。左辺値は代入演算子の左側で使う。

シーケンス ブレークポイント (Sequenced Breakpoints)

シーケンスで発生するブレークポイント。ブレークポイントのシーケンスはボトムアップ方式で実行される。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する。

シェル (Shell)

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 インターフェイス。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。

式 (Expressions)

算術演算子または論理演算子で区切った定数または記号の組み合わせ。

識別子 (Identifier)

関数または変数の名前。

システム ウィンドウ コントロール (System Window Control)

ウィンドウと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] 等のメニュー項目がポップアップ表示される。

実行可能コード (Executable Code)

読み込んで実行できる形式のソフトウェア。

シナリオ (Scenario)

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 (Simulator)

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 (Qualifier)

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

条件付きアセンブリ (Conditional Assembly)

アセンブリ言語で、ある特定の式のアセンブル時の値に基づいて含まれたり除外されたりするコード。

条件付きコンパイル (Conditional Compilation)

プログラムの一部を、プリプロセッサ ディレクティブで指定した特定の定数式が真の場合のみコンパイルする事。

初期化済みデータ (Initialized Data)

初期値を指定して定義されたデータ。C では、

```
int myVar=5;
```

として定義した変数は初期化済みデータセクションに格納する。

シングルステップ (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/MPLAB X 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ため、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 行から生成される全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。

シンボル (Symbol)

プログラムを構成する各種の要素を記述する汎用のメカニズム。関数名、変数名、セクション名、ファイル名、struct/enum/union タグ名等がある。MPLAB IDE/MPLAB X 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。

推奨しない機能 (Deprecated Features)

後方互換性確保のためにサポートしているだけで現在は使っておらず、いずれ廃止になる事が決まっている機能。

スキッド (Skid)

ハードウェア ブレークポイントを使ってプロセッサを停止する場合、ブレークポイントからさらに 1 つ以上の命令を実行してプロセッサが停止する事がある。ブレークポイントの後に実行する命令の数をスキッドと呼ぶ。

© 2014 Microchip Technology Inc.

スキュー (Skew)

命令実行に対応する情報は、異なる複数の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、並びに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時にバスに現れる。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。トレースバッファは、1 インスタンスでバス上に存在する情報をキャプチャする。従って、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、コンパイルド (Stack, Compiled)

コンパイラが管理し割り当てるメモリの領域で、この領域内で変数に静的に空間を割り当てる。ターゲット デバイス上にソフトウェア スタックのメカニズムを効率的に実装できない場合、ソフトウェア スタックがコンパイルド スタックに置き換わる。このメカニズムでは、関数は再入可能ではなくなる。

スタック、ソフトウェア (Stack, Software)

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うメモリ。このメモリはプログラムでの命令の実行時に動的に割り当てられる。これによって、再入可能な関数の呼び出しが可能になる。

スタック、ハードウェア (Stack, Hardware)

PIC マイクロコントローラで関数を呼び出す時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 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。

スティミュラス (Stimulus)

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、テキストファイルにアクションのリストとしてこのデータを記述する。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。

ステータスバー (Status Bar)

MPLAB IDE/MPLAB X IDE ウィンドウの一番下にあるバーで、カーソル位置、開発モードとデバイス、アクティブなツールバー等に関する情報が表示される。

ストップウォッチ (Stopwatch)

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 (Control Directives)

アセンブリ言語コード内で使うディレクティブで、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。

セクション (Section)

OCG の psect に相当する GCC の用語。リンカが 1 つのまとまりとして処理するコードまたはデータのブロック。

セクション属性 (Section Attribute)

GCC のセクションの特徴を表す情報 (例 : `access` セクション)。

絶対シンボル (Symbol, Absolute)

アセンブリの `.equ` ディレクティブによる定義等、即値を表す。

絶対セクション (Absolute Section)

リンカで変更されない固定 (絶対) アドレスを持つ GCC コンパイラのセクション。

絶対変数 / 関数 (Absolute Variable/Function)

OCG コンパイラの `@ address` 構文を使って絶対アドレスに配置される変数または関数。

相互参照ファイル (Cross Reference File)

シンボルテーブルと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、リストの最初のファイルがシンボル定義の位置となる。残りのファイルはシンボルへの参照を含む。

ソースコード (Source Code)

プログラマ (人間) 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、またはインタプリタで実行される。

ソースファイル (Source File)

ソースコードを記述した ASCII テキストファイル。

属性 (Attribute)

GCC の C プログラムの変数または関数の特徴を表す情報で、マシン固有の特性を記述する目的で使う。

属性 (セクション属性) (Attribute、Section)

「executable」、「readonly」、「data」等、GCC のセクションの特徴を表す情報。アセンブラの `.section` ディレクティブでフラグとして指定できる。

た

ターゲット アプリケーション (Target Application)

ターゲットボードに読み込んだソフトウェア。

ターゲット プロセッサ (Target Processor)

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

ターゲット (Target)

ユーザ ハードウェアの事。

ターゲットボード (Target Board)

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 (Download)

ホストから別のデバイス (エミュレータ、プログラマ、ターゲットボード等) にデータを送信する事。

致命的エラー (Fatal Error)

コンパイルがただちに停止するようなエラー。エラー以降はメッセージも出力されない。

中央演算処理装置 (Central Processing Unit)

デバイス内で、実行する正しい命令をフェッチし、デコードして実行する装置。必要に応じて、算術論理演算装置 (ALU) と組み合わせて命令実行を完了する。プログラムメモリのアドレスバス、データメモリのアドレスバス、スタックへのアクセスを制御する。

ツールバー (Tool Bar)

MPLAB IDE/MPLAB X 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。

ディレクティブ (Directives)

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。

© 2014 Microchip Technology Inc.

データ ディレクティブ (Data Directives)

アセンブラが行う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) で参照する手段として使う。

データメモリ (Data Memory)

Microchip 社の MCU と DSC 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) から成る。EEPROM データメモリを内蔵したデバイスもある。

データ監視および制御インターフェイス (DMCI)

MPLAB X IDE 内のツール。このインターフェイスは、プロジェクト内のアプリケーション変数の動的な入力制御を提供する。4 つの動的に割り当て可能なグラフィック ウィンドウを使って、アプリケーションが生成するデータをグラフィカルに表示できる。

デジタルシグナル コントローラ (Digital Signal Controller)

デジタル信号処理機能をサポートしたマイクロコントローラ。Microchip 社の dsPIC DSC 等。

デジタル信号処理 / デジタルシグナル プロセッサ (Digital Signal Processing/Digital Signal Processor)

デジタル信号処理 (DSP) とは、デジタル信号をコンピュータで処理する事。通常は、アナログ信号 (音声または画像) をデジタル形式に変換 (サンプリング) して処理する事をいう。デジタルシグナル プロセッサとは、信号処理用に設計されたマイクロプロセッサの事。

デバイス プログラマ (Device Programmer)

マイクロコントローラ等、電気的に書き込み可能な半導体デバイスをプログラミングするためのツール。

デバッグ / デバッガ (Debug/Debugger)

「ICE/ICD」参照。

デバッグ情報 (Debugging Information)

コンパイラと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える各種レベルの情報を出力できる。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参照。

テンプレート (Template)

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

特殊機能レジスタ (Special Function Registers: SFR)

I/O プロセッサ機能、I/O ステータス、タイマ等の各種モードや周辺モジュールを制御するレジスタ専用に使うデータメモリ (RAM) 領域。

匿名構造体 (Anonymous Structure)

16 ビット C コンパイラ - 無名の構造体

PIC18 C コンパイラ - C 共用体のメンバーである名前のない構造体。匿名構造体のメンバーは、その構造体を包含している共用体のメンバーと同じようにアクセスできる。例えば以下のサンプルコードでは、`hi` と `lo` は共用体 `caster` に含まれる匿名構造体のメンバーである。

```
union castaway {
 int intval;
 struct {
  char lo; //accessible as caster.lo
  char hi; //accessible as caster.hi
 };
} caster;
```

トライグラフ (Trigraphs)

「??」で始まる3文字のシーケンス。ISO Cで定義されており、1つの文字に置換される。

トリガ出力 (Trigger Output)

任意のアドレスまたはアドレス範囲で生成でき、トレースとブレークポイントの設定から独立したエミュレータ出力信号の事。トリガ出力ポイントはいくつでも設定できる。

トレース (Trace)

プログラム実行のログを記録するエミュレータまたはシミュレータの機能。エミュレータ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/MPLAB X IDE のトレース ウィンドウにアップロードする。

トレースマクロ (Trace Macro)

エミュレータ データからのトレース情報を提供するマクロ。これはソフトウェア トレースのため、トレースを利用するには、マクロをコードに追加し、コードを再コンパイルまたは再アセンブルし、ターゲット デバイスにこのコードをプログラムする必要がある。

レースメモリ (Trace Memory)

エミュレータ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。

な

内部リンケージ (Internal Linkage)

関数または変数が、それ自身を定義し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。

生データ (Raw Data)

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。

ニーモニック (Mnemonics)

マシンコードと 1 対 1 で対応したテキスト命令。オペコードとも呼ぶ。

ネイティブ データサイズ (Native Data Size)

ネイティブ トレースの場合、[Watches] ウィンドウで使う変数のサイズは選択したデバイスのデータメモリと同じサイズ (PIC18 の場合は同じバイトサイズ、16 ビットデバイスの場合は同じワードサイズ) である必要がある。

ノード (Node)

MPLAB IDE/MPLAB X IDE のプロジェクトを構成するコンポーネント。

は

パスカウンタ (Pass Counter)

イベント (特定のアドレスの命令を実行する等) が発生するたびに値をデクリメントするカウンタ。パスカウンタの値がゼロになると、イベントの条件を満たす。パスカウンタはブレークロジック、トレースロジック、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 (Power-on-Reset Emulation)

データ RAM 空間に無作為な値を書き込んで、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無作為化処理。

ヒープ (Heap)

動的メモリ割り当てに使うメモリ空間。メモリブロックの割り当てと解放は実行時に任意の順序で行う。

非拡張モード (Non-Extended Mode) (PIC18 MCU)

コンパイラの動作モードの 1 つ。拡張命令もリテラル オフセットによるインデックス アドレス指定も利用しない。

© 2014 Microchip Technology Inc.

非初期化データ (Uninitialized Data)

初期値なしで定義されたデータ。C では、

```
int myVar;
```

は、非初期化済みデータセクションに格納される変数を定義する。

非同期 (Asynchronously)

複数のイベントが同時には発生しない事。一般に、プロセッサ実行中の任意の時点で発生する割り込みに言及する際に使う。

非同期スティミュラス (Asynchronous Stimulus)

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 (Non Real Time)

ブレー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/MPLAB X IDE を指す。

ビルド (Build)

全てのソースファイルのコンパイルとリンクを行ってアプリケーションを作成する事。

ファイルレジスタ (File Registers)

汎用レジスタ (GPR) と特殊機能レジスタ (SFR) から成る内蔵のデータメモリ。

ファントムバイト (Phantom Byte)

dsPIC アーキテクチャで、24 ビット命令ワードを 32 ビット命令ワードと見なして扱う場合に使う未実装バイト。dsPIC の hex ファイルに見られる。

フィックスアップ (Fixup)

リンカによる再配置後にオブジェクト ファイルのシンボル参照を絶対アドレスに置き換える処理。

フィルタ (Filter)

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。

ブックマーク (Bookmark)

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。

[Editor] ツールバーの [Toggle Bookmarks] を選択してブックマークを追加または削除する。このツールバーの他のアイコンをクリックすると、次または前のブックマークに移動する。

プラグイン (Plug-ins)

MPLAB IDE/MPLAB X 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事で、各種ソフトウェア / 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。

プラグマ (Pragma)

特定のコンパイラにとって意味を持つディレクティブ。一般に、実装で定義した情報をコンパイラに伝達するために使う。MPLAB C30 は属性を利用してこの情報を伝達する。

フラッシュ (Flash)

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。

不揮発性ストレージ (Non-Volatile Storage)

電源を OFF にしても内容が失われないストレージ デバイス。

フリースタンディング (Free-Standing)

複素数型を使っておらず、ライブラリ (ANSI C89 規格第 7 節) で規定する機能の使用が標準ヘッダ (<float.h>、<iso646.h>、<limits.h>、<stdarg.h>、<stdbool.h>、<stddef.h>、<stdint.h>) の内容のみに限定されている厳密な規格合致プログラムを受理する処理系。

ブレークポイント (Breakpoint)

ハードウェア ブレークポイント : 実行するとファームウェアの実行が停止するイベント。

ソフトウェア ブレークポイント : 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 命令で実行が停止される。

フレームポインタ (Frame Pointer)

スタックベースの引数とスタックベースのローカル変数の境界となるスタック番地を指し示すポインタ。ここを基準にすると、現在の関数のローカル変数やその他の値に容易にアクセスできる。

プログラム カウンタ ユニット (Program Counter Unit)

16 ビットアセンブラ - プログラムメモリのレイアウトを概念的に表現したもの。プログラム カウンタは 1 命令ワードで 2 つインクリメントする。実行可能セクションでは、2 プログラム カウンタ ユニットは 3 バイトに相当する。読み出し専用セクションでは、2 プログラム カウンタ ユニットは 2 バイトに相当する。

プログラム カウンタ (Program Counter)

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 (Program Memory)

MPLAB IDE/MPLAB X IDE - デバイス内で命令を保存するメモリ空間。また、エミュレータ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空間もプログラムメモリと呼ぶ。

16 ビットアセンブラ / コンパイラ - デバイス内で命令を保存するメモリ空間。

プロジェクト (Project)

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 ファイル等) 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 (Prototype System)

ユーザのターゲット アプリケーションまたはターゲットボードの事。

プロファイル (Profile)

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

プロローグ (Prologue)

コンパイラで生成したコードのうち、スタック領域の割り当て、レジスタの退避、ランタイムモデルで指定したその他のマシン固有の要件を実行するコード部分。プロローグは、関数のユーザコードの前に実行する。

ベクタ (Vector)

リセットまたは割り込みが発生した時にアプリケーションのジャンプ先となるメモリ番地。

ホスト PC (PC Host)

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 (Pod)

インサーキット エミュレータまたはデバッガの筐体。丸型の場合「パック」(Puck) と呼ぶ事もある。あるいは「プローブ」(Probe) とも呼ぶが、「論理プローブ」と混同せぬよう注意が必要。

© 2014 Microchip Technology Inc.

ま

マイクロコントローラ モード (Microcontroller Mode)

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、内部実行のみを許可する。つまり、マイクロコントローラモードでは内蔵プログラムメモリしか利用できない。

マイクロコントローラ (Microcontroller)

CPU、RAM、プログラムメモリ、I/O ポート、タイマ等、多くの機能を統合したチップ。

マイクロプロセッサ モード (Microprocessor Mode)

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、内蔵プログラムメモリは使わない。プログラムメモリ全体を外部にマッピングする。

マクロ ディレクティブ (Macro Directives)

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。

マクロ (Macro)

マクロ命令。一連の命令シーケンスを短い名前で表現した命令。

マシンコード (Machine Code)

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事もある) から成る。ある特定のプロセッサで使える全ての命令の集合を「命令セット」という。

マシン語 (Machine Language)

ある CPU が翻訳を必要とせず実行できる命令の集合。

未割り当てセクション (Unassigned Section)

リンカのコマンドファイルで特定のターゲット メモリブロックに割り当てられていないセクション。リンカは、未割り当てセクションを割り当てるターゲット メモリブロックを検出する必要がある。

命令 (Instructions)

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事もできる。

命令セット (Instruction Set)

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 (Message)

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

メモリモデル (Memory Model)

C コンパイラの場合、アプリケーションで利用可能なメモリを表現したもの。PIC18 C コンパイラの場合、プログラムメモリを指し示すポインタのサイズに関する規定を記述したもの。

モジュール (Module)

プリプロセッサ ディレクティブ実行後の前処理済みのソースファイル出力。翻訳単位とも呼ぶ。

や

優先順位 (Precedence)

式の評価順を定義した規則。

読み出し専用メモリ (Read Only Memory)

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、データの追加または変更は不可。

ら

ライブラリ / ライブラリアン (Library/Librarian)

「アーカイブ / アーカイバ」参照。

ランタイム ウォッチ (Runtime Watch)

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

ランタイムモデル (Run-time Model)

ターゲット アーキテクチャのリソースの使用を記述したもの。

リアルタイム (Real Time)

インサーキット エミュレータまたはデバッガが 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、通常のチップと同じ挙動をする。リアルタイム モードでは、エミュレータのリアルタイム トレースバッファが有効になり、選択した全てのサイクルを常時キャプチャする。また、全てのブレークロジックが有効になる。インサーキット エミュレータまたはデバッガでは、有効なブレークポイントで停止するか、またはユーザが実行を停止す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事をリアルタイムと呼ぶ。

リスティング ディレクティブ (Listing Directives)

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等、リスティング ファイルに関する各種の設定を行う。

リスティング (Listing File)

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。

リトル エンディアン (Little Endian)

マルチバイト データで最下位バイト (LSB) を最下位アドレスに格納するデータ並び順方式。

量産プログラマ (Production Programmer)

デバイスを高速にプログラミングできるようにリソースを強化したプログラマ。各種電圧レベルでのプログラミングに対応し、プログラミング仕様に完全に準拠している。量産環境ではアプリケーション回路が組立ラインにとどまる時間をなるべく短くする必要があるため、デバイスへの書き込み時間が特に重要である。

リンカ (Linker)

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可能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 (Linker Script Files)

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。

ループバック テストボード (Loop-Back Test Board)

MPLAB REAL ICE インサーキット エミュレータの動作をテストするために用いる。

レイテンシ (Latency)

イベントが発生してからその応答までの時間の長さ。

© 2014 Microchip Technology Inc.

ローカルラベル (Local Label)

マクロ内で LOCAL ディレクティブを使って定義されたラベル。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とラベルには、ENDM マクロ以降はアクセスできない。

ロジックプローブ (Logic Probes)

Microchip 社製エミュレータには、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、+5 V、共通グランドを提供する。

わ

ワークブック (Workbook)

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。

割り当てセクション (Assigned Section)

リンカのコマンドファイルで特定のターゲット メモリブロックに割り当てられた GCC コンパイラのセクション。

割り込み (Interrupt)

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を再開する。

割り込みサービス要求 (Interrupt Service Request (IRQ))

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、優先度の異なる割り込みを処理できるものもある。

割り込みサービスルーチン (ISR)

言語ツールの場合、割り込みを処理する関数。

MPLAB IDE/MPLAB X IDE の場合、割り込みが発生すると実行されるユーザ作成コード。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を実行する。

割り込みハンドラ (Interrupt Handler)

割り込み発生時に専用のコードを実行するルーチン。

割り込みベクタ (Interrupt Vector)

割り込みサービスルーチンまたは割り込みハンドラのアドレス。

NOTES:

© 2014 Microchip Technology Inc.

索引

数字

44 ピンデモボード 35

D

Debug Express 35

E

ETN 57

F

Freeze on Halt 50

I

ICD ヘッダ 16
ICSP 24, 25, 27, 63
ICSPCLK 63
ICSPDAT 63

L

LED 15, 62

P

PC、省電力 50, 62
PGC 21, 22, 23, 24, 25, 26
PGD 21, 22, 23, 24, 25, 26
PICkit 3 について 13
PICkit 3 の構成要素 16
PIM 19

R

Readme 10

U

USB 62, 75
　ケーブル 16
　ハブ 62
USB ポート 14

V

Vcap 22
Vdd 21, 22, 24, 25
Vpp 21, 22, 23, 25
Vss 21, 22, 24, 25

い

一般的な対処方法 54
インジケータ ライト 62

う

ウォッチドッグ タイマ 25
ウォッチドッグ タイマ (Watchdog Time) 76

え

エンジニアリング技術ノート 57

き

休止モード 50, 62

け

ケーブル
　長さ 62, 65

こ

コード内で設定する
　コンフィグレーション ビット 29
コード保護 25
コネクタ、6 ピン 15
コンデンサ 22, 23
コンフィグレーション ビット 25

し

省電力モード 50, 62

す

推奨参考資料 10

た

ターゲット デバイス 25
ターゲットとの接続
　回路 22
　標準 21
　不適切な回路 23
耐久性、カードガイド 62

て

抵抗 23
テーブル読み出し保護 25
デバイスによる予約済みリソース 27
デバッガの正常動作を妨げる回路 23
デバッグ
　　実行プログラム 26
デバッグに失敗する主な理由 47
デバッグモード
　　操作手順 25

は

ハードウェア ツールの常時接続 29
ハブ、USB 62

ひ

標準 ICSP による通信 19
標準通信
　　接続 21

ふ

ファームウェア
　　ダウンロード中の切断 51
プルアップ 23
プロセッサ拡張キット 16
プロセッサ拡張パックとヘッダ
　　仕様 10

へ

変換ソケット 16
　　仕様 10, 31

ほ

本書
　　構成 8
　　表記 9

め

メモリウィンドウに変更が反映されない 51

も

モジュラ インターフェイス ケーブル 25

NOTES:

各国の営業所とサービス

北米

本社
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
技術サポート：
http://www.microchip.com/support
URL:
www.microchip.com

アトランタ
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

オースティン、TX
Tel: 512-257-3370

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
Novi, MI
Tel: 248-848-4000

ヒューストン、TX
Tel: 281-894-5983

インディアナポリス
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

ロサンゼルス
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

ニューヨーク、NY
Tel: 631-435-6000

サンノゼ、CA
Tel: 408-735-9110

カナダ - トロント
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: 86-10-8569-7000
Fax: 86-10-8528-2104

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

中国 - 重慶
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

中国 - 杭州
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

中国 - 香港 SAR
Tel: 852-2943-5100
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 - 深圳
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

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

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

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

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: 91-80-3090-4123

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: 91-20-3019-1500

日本 - 大阪
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

日本 - 東京
Tel: 81-3-6880-3770
Fax: 81-3-6880-3771

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: 886-3-5778-366
Fax: 886-3-5770-955

台湾 - 高雄
Tel: 886-7-213-7830

台湾 - 台北
Tel: 886-2-2508-8600
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 - デュッセルドルフ
Tel: 49-2129-3766400

ドイツ - ミュンヘン
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

ドイツ - プフォルツハイム
Tel: 49-7231-424750

イタリア - ミラノ
Tel: 39-0331-742611
Fax: 39-0331-466781

イタリア - ヴェニス
Tel: 39-049-7625286

オランダ - ドリューネン
Tel: 31-416-690399
Fax: 31-416-690340

ポーランド - ワルシャワ
Tel: 48-22-3325737

スペイン - マドリッド
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

スウェーデン - ストックホルム
Tel: 46-8-5090-4654

イギリス - ウォーキンガム
Tel: 44-118-921-5800
Fax: 44-118-921-5820

10/28/13

© 2014 Microchip Technology Inc.