



适用于MPLAB<sup>®</sup> X IDE 的

MPLAB REAL ICE<sup>™</sup>

在线仿真器

用户指南

---

请注意以下有关 Microchip 器件代码保护功能的要点:

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信: 在正常使用的情况下, Microchip 系列产品是当今市场上同类产品中更安全的产品之一。
- 目前, 仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知, 所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下, 能访问您的软件或其他受版权保护的成果, 您有权依据该法案提起诉讼, 从而制止这种行为。

---

提供本文档的中文版本仅为为了便于理解。请勿忽视文档中包含的英文部分, 因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为为您提供便利, 它们可能由更新之信息所替代。确保应用符合技术规范, 是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保, 包括但不限于针对其使用情况、质量、性能、适用性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用, 一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时, 会维护和保障 Microchip 免于承担法律责任, 并加以赔偿。除非另外声明, 在 Microchip 知识产权保护下, 不得暗或以其他方式转让任何许可证。

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2009 认证。Microchip 的 PIC<sup>®</sup> MCU 与 dsPIC<sup>®</sup> DSC、KEELOQ<sup>®</sup> 跳码器件、串行 EEPROM、单片机外设、非易失性存储器 and 模拟产品严格遵守公司的质量体系流程。此外, Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

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

#### 商标

Microchip 的名称和徽标组合、Microchip 徽标、AnyRate、AVR、AVR 徽标、AVR Freaks、BeaconThings、BitCloud、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KEELOQ、KEELOQ 徽标、Kleer、LANCheck、LINK MD、maXStylus、maXTouch、MediaLB、megaAVR、MOST、MOST 徽标、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32 徽标、Prochip Designer、QTouch、RightTouch、SAM-BA、SpyNIC、SST、SST 徽标、SuperFlash、tinyAVR、UNI/O 及 XMEGA 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTouch、Precision Edge 和 Quiet-Wire 均为 Microchip Technology Inc. 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、chipKIT、chipKIT 徽标、CodeGuard、CryptoAuthentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet 徽标、Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PureSilicon、QMatrix、RightTouch 徽标、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 为 Microchip Technology Inc. 在美国的服务标记。

Silicon Storage Technology 为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. & KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2013-2017, Microchip Technology Inc. 版权所有。

ISBN: 978-1-5224-1792-7

声明对象: MPLAB® REAL ICE™ 在线仿真器

EU Declaration of Conformity

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.

This development/evaluation tool, when incorporating wireless and radio-telecom functionality, is in compliance with the essential requirement and other relevant provisions of the R&TTE Directive 1999/5/EC and the FCC rules as stated in the declaration of conformity provided in the module datasheet and the module product page available at [www.microchip.com](http://www.microchip.com).

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](http://www.microchip.com).

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

  
Derek Carlson

VP Development Tools

  
Date

# 适用于 MPLAB X IDE 的仿真器用户指南

---

注:

---

---

## 目录

---

---

前言 .....	9
<b>第1部分——概述与操作</b>	
<b>第1章 关于仿真器</b>	
1.1 简介 .....	17
1.2 仿真器特性 .....	17
1.3 仿真器提供的功能 .....	17
1.4 仿真器工具包组件 .....	18
1.5 仿真器系统 .....	19
<b>第2章 器件与功能支持</b>	
2.1 开发工具选择器（DTS） .....	21
2.2 器件与功能支持HTML表 .....	23
<b>第3章 工作原理</b>	
3.1 简介 .....	25
3.2 工具比较 .....	26
3.3 操作概述 .....	27
3.4 仿真器与PC的通信 .....	28
3.5 仿真器与目标板的通信 .....	28
3.6 跟踪连接 .....	33
3.7 使用仿真器进行调试 .....	36
3.8 调试的要求 .....	37
3.9 使用仿真器编程 .....	39
3.10 仿真器使用的资源 .....	39
<b>第2部分——特性</b>	
<b>第4章 使用仿真器</b>	
4.1 简介 .....	43
4.2 安装和设置 .....	43
4.3 固件升级 .....	44
4.4 仿真器功能 .....	45
4.5 快速调试/编程参考 .....	45
4.6 调试器/编程器限制 .....	45
<b>第5章 基本调试功能</b>	
5.1 简介 .....	47
5.2 启动和停止仿真 .....	47

# 适用于MPLAB X IDE的仿真器用户指南

---

5.3 查看处理器存储器和文件 .....	48
5.4 断点和跑表 .....	48
5.5 器件调试功能 .....	49
<b>第6章 特定调试功能</b>	
6.1 简介 .....	51
6.2 数据捕捉和运行时观察 .....	51
6.3 插装跟踪 .....	54
6.4 PIC32指令跟踪——仅限PIC32 MCU .....	61
6.5 跳转跟踪——仅限EP器件 .....	66
6.6 PC采样——仅限8位和16位MCU .....	67
6.7 PC性能分析——仅限32位MCU .....	70
6.8 函数级别性能分析 .....	72
6.9 应用程序输入/输出 .....	75
6.10 外部触发信号 .....	79
6.11 额外调试功能 .....	80
<b>第3部分——故障诊断</b>	
<b>第7章 故障诊断首要步骤</b>	
7.1 简介 .....	83
7.2 要首先回答的5个问题 .....	83
7.3 无法调试的首要原因 .....	83
7.4 要考虑的其他事项 .....	84
<b>第8章 常见问题解答 (FAQ)</b>	
8.1 简介 .....	85
8.2 仿真器运行 .....	85
8.3 插装跟踪运行 .....	88
8.4 一般问题 .....	90
<b>第9章 消息</b>	
9.1 简介 .....	95
9.2 错误、警告和信息性消息 .....	95
9.3 常规纠正措施 .....	97
<b>第10章 工程技术说明 (ETN)</b>	
<b>第4部分——软件和硬件参考</b>	
<b>第11章 仿真器功能汇总</b>	
11.1 简介 .....	103
11.2 仿真器选择与切换 .....	103
11.3 仿真器选项选择 .....	103
11.4 仿真器窗口和对话框 .....	111
<b>第12章 硬件规范</b>	
12.1 简介 .....	113
12.2 重点 .....	113

12.3 USB 端口/电源 .....	113
12.4 仿真器主机 .....	114
12.5 标准通信硬件 .....	116
12.6 高速/LVDS通信硬件 .....	118
12.7 环回测试板 .....	119
12.8 目标板注意事项 .....	119
<b>附录A 版本历史 .....</b>	<b>121</b>
<b>支持 .....</b>	<b>123</b>
<b>术语表 .....</b>	<b>127</b>
<b>索引 .....</b>	<b>147</b>
<b>全球销售及服务网点 .....</b>	<b>150</b>

注:



## 前言

### 客户须知

所有文档均会过时，本文档也不例外。Microchip 的工具和文档将不断演变以满足客户的需求，因此实际使用中有些对话框和 / 或工具说明可能与本文档所述之内容有所不同。请访问我们的网站 ([www.microchip.com](http://www.microchip.com)) 获取最新文档。

文档均标记有“DS”编号。该编号出现在每页底部的页码之前。DS 编号的命名约定为“DSXXXXXXXXA\_CN”，其中“XXXXXXXX”为文档编号，“A”为文档版本。

欲了解开发工具的最新信息，请参考 MPLAB® IDE 在线帮助。从 Help（帮助）菜单选择 Topics（主题），打开现有在线帮助文件列表。

### 简介

本章包含使用 MPLAB® REAL ICE™ 在线仿真器前需要了解的一般信息。

内容包括：

- [文档编排](#)
- [本指南使用的约定](#)
- [推荐读物——仿真器](#)
- [推荐读物——仿真器附件](#)

## 文档编排

本文档介绍了如何使用MPLAB REAL ICE在线仿真器作为开发工具在目标板上仿真和调试固件，以及如何烧写器件。文档内容编排如下：

### 第1部分——概述与操作

- **第1章：关于仿真器**——介绍了MPLAB REAL ICE在线仿真器的概念及其如何帮助开发应用。
- **第2章：器件和功能支持**——器件系列支持的功能表。
- **第3章：工作原理**——讲述了MPLAB REAL ICE在线仿真器的工作原理，并给出了配置选项的说明。

### 第2部分——功能

- **第4章：常规设置**——讲述了如何设置MPLAB X IDE以使用仿真器。
- **第5章：常用调试功能**——讲述了选择MPLAB REAL ICE在线仿真器作为调试工具时，可在MPLAB X IDE中使用的基本仿真器功能。其中包括断点、跑表和外部触发等调试功能。
- **第6章：特定调试功能：8位和16位器件**——说明了数据捕捉、运行时观察、8位和16位（数据存储器）器件跟踪、PC采样、应用程序输入/输出以及其他调试功能。
- **第7章：特定调试功能：32位**——说明了数据捕捉、运行时观察、32位（数据存储器）器件跟踪、PC采样、应用程序输入/输出以及其他调试功能。

### 第3部分——故障诊断

- **第8章：故障诊断的第一步**——仿真器运行出现问题时首先应该尝试的操作。
- **第9章：常见问题（FAQ）**——关于仿真器运行的常见问题列表。
- **第10章：消息**——错误消息和建议解决方案的列表。
- **第11章：工程技术说明（ETN）**——列出了所有可以解决的硬件问题。

### 第4部分——软件和硬件参考

- **第12章：仿真器功能汇总**——汇总了选择MPLAB REAL ICE仿真器作为调试或编程工具时，可在MPLAB IDE中使用的仿真器功能。
- **第13章：硬件规范**——讲述了仿真器系统的硬件规范和电气规范。包括如何使用环回测试板的说明。

## 本指南使用的约定

本指南采用以下文档约定：

## 文档约定

说明	表示	示例
<b>Arial 字体:</b>		
斜体字	参考书目	<i>MPLAB<sup>®</sup> IDE User's Guide</i>
	需强调的文字	…… 为仅有的编译器 ……
首字母大写	窗口	Output 窗口
	对话框	Settings 对话框
	菜单选择	选择 Enable Programmer
引用	窗口或对话框中的字段名	“Save project before build”
带右尖括号且有下划线的斜体文字	菜单路径	<i>File&gt;Save</i>
粗体字	对话框按钮	单击 <b>OK</b>
	选项卡	单击 <b>Power</b> 选项卡
N'Rnnnn	verilog 格式的数字，其中 N 为总位数，R 为基数，n 为其中一位。	4'b0010, 2'hF1
尖括号 < > 括起的文字	键盘上的按键	按下 <Enter>, <F1>
<b>Courier New 字体:</b>		
常规 Courier New	源代码示例	#define START
	文件名	autoexec.bat
	文件路径	c:\mcc18\h
	关键字	_asm, _endasm, static
	命令行选项	-Opa+, -Opa-
	二进制位值	0, 1
	常量	0xFF, 'A'
斜体 Courier New	可变参数	<i>file.o</i> , 其中 <i>file</i> 可以是任一有效文件名
方括号 []	可选参数	mcc18 [选项] <i>file</i> [选项]
花括号和竖线: {}	选择互斥参数: “或”选择	errorlevel {0 1}
省略号 ...	代替重复文字	var_name [, var_name...]
	表示由用户提供的代码	void main (void) { ... }

## 推荐读物——仿真器

以下Microchip文档均已提供，并建议读者作为补充参考资料。

### **多工具设计忠告 (DS51764C\_CN)**

提供旨在确保与各种开发工具正确接口的有关准则和实现注意事项的小文档。

### **MPLAB REAL ICE在线仿真器的版本说明**

关于使用MPLAB REAL ICE在线仿真器的最新信息，请单击Start Page（起始页）上的“Release Notes, User's Guide and Support Docs”（版本说明、用户指南和支持文档）来阅读“Readme for MPLAB REAL ICE Emulator.htm”文件（HTML文件）。版本说明（自述文件）包含了本文档中可能未提供的更新信息和已知问题。

### **Using the MPLAB REAL ICE In-Circuit Emulator (DS51997)**

该手册介绍了MPLAB REAL ICE在线仿真器的硬件连接和软件安装方式。

### **MPLAB REAL ICE在线仿真器帮助**

MPLAB X IDE中综合仿真器用户指南的在线版本。其中涵盖仿真器的使用方法、故障排除以及硬件规范。

### **处理器扩展包 (PEP) 和调试头规范 (DS50001292W\_CN)， 仿真扩展包 (EEP) 和仿真头用户指南 (DS50002243B\_CN)**

这些小册子介绍了如何安装并使用调试头和仿真头。通过特殊-ME2/-ICE/-ICD器件版本使用调试头来更好地调试所选器件，而无需占用额外的引脚和资源。处理器扩展包包含调试头。另请参见相关帮助文件。

### **Transition Socket Specification (DS51194)**

可参考本文档获取关于适用于调试头的转接插座的信息。

## 推荐读物——仿真器附件

以下 Microchip 文档介绍了仿真器系统的可用附件。

### **MPLAB® REAL ICE™ 在线仿真器高性能工具包用户指南 (DS50002528A\_CN)**

高性能工具包 (AC244002) 包含用于高速 /LVDS 通信的连接器和电缆。该文档介绍了如何为仿真器与目标板进行通信而连接硬件, 说明了应使用和不应使用的目标电路, 并提供了关于硬件组件的详细信息。

### **MPLAB® REAL ICE™ 跟踪接口工具包规范 (DS50002531A\_CN)**

跟踪接口工具包 (AC244006) 提供了用于通过支持的 PIC32 MCU/PIM 或仿真头来进行指令跟踪的电缆和适配器板。该文档列出了所支持的 PIC32 PIM, 给出了 PIC32 PIM 配置的示例, 并提供了工具包组件的硬件详细信息。

### **MPLAB® REAL ICE™ 在线仿真器隔离器单元使用说明书 (DS50001858B\_CN)**

### **MPLAB® REAL ICE™ 在线仿真器隔离器单元规范 (DS50002529A\_CN)**

光电隔离单元 (AC244005) 用于在开发大功率应用时保护仿真器。说明书说明了如何连接单元硬件。规范提供了关于单元硬件的更多信息。

### **MPLAB® REAL ICE™ 在线仿真器 JTAG 适配器使用说明书 (DS50002094C\_CN)**

### **MPLAB® REAL ICE™ 在线仿真器 JTAG 适配器规范 (DS50002530A\_CN)**

JTAG 适配器板 (AC244007) 为仿真器提供 JTAG 功能。(注意, 使用此板时将无法使用全套调试功能。) 说明书说明了如何连接 JTAG 适配器板。规范提供了关于适配器板硬件的更多信息。

### **MPLAB® REAL ICE™ 在线仿真器功率监视器使用说明书 (DS50002156A\_CN)**

### **MPLAB® REAL ICE™ 在线仿真器功率监视器用户指南 (DS50002532A\_CN)**

功率监视器板 (AC244008) 可用于查看关于目标板的功率使用情况。说明书说明了如何连接电源监视器。用户指南提供了关于软件设置、所有硬件配置以及使用功率监视器的更多信息。

注:



---

---

## 第1部分——概述与操作

---

---

第1章 关于仿真器 .....	17
第2章 器件与功能支持 .....	21
第3章 工作原理 .....	25

注:



---

---

## 第1章 关于仿真器

---

---

### 1.1 简介

MPLAB REAL ICE在线仿真器是一款支持Microchip绝大部分PIC®单片机（MCU）和dsPIC®数字信号控制器（Digital Signal Controller, DSC）硬件开发的先进仿真器。

本章提供了仿真器的概述：

- [仿真器特性](#)
- [仿真器提供的功能](#)
- [仿真器工具包组件](#)
- [仿真器系统](#)

### 1.2 仿真器特性

MPLAB REAL ICE仿真功能具有以下特性：

- 处理器以最高速度运行
- 可对器件进行在线调试
- 处理器总线上没有任何仿真负荷
- 互连简单
- 具有I/O数据合并功能
- 插装（instrumented）跟踪（由MPLAB X IDE和编译器辅助支持）
- 指令跟踪（硬件跟踪）

除仿真器功能外，MPLAB REAL ICE在线仿真器系统还可以用作生产编程器。

### 1.3 仿真器提供的功能

MPLAB REAL ICE在线仿真器是开发工程师工具套件的组成部分。从软件开发到硬件集成、生产测试和现场服务等多种应用领域均可使用此仿真器。

MPLAB REAL ICE在线仿真器系统能够：

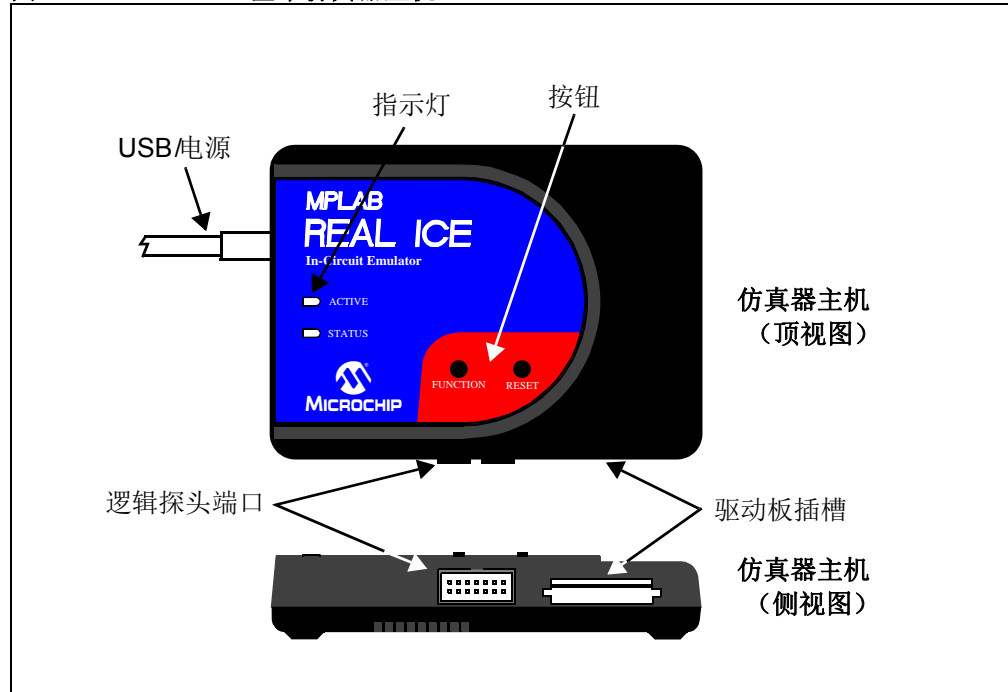
- 在硬件上实时调试应用程序
- 使用硬件断点调试
- 使用软件断点调试（取决于器件）
- 暂停（基于内部事件和/或外部信号）
- 监视内部文件寄存器
- 全速仿真
- 作为生产编程器来编程器件
- 跟踪代码行或记录变量/表达式值

## 1.4 仿真器工具包组件

下面列出了 MPLAB REAL ICE 在线仿真器系统工具包（DV244005）的组件：

- 仿真器主机 —— 见第 12.4 节“仿真器主机”。
- USB 电缆 —— 在仿真器和 PC 之间提供通信，并向仿真器供电。请参见第 12.3 节“USB 端口 / 电源”。
- 标准驱动板和电缆 —— 将仿真器主机连接到调试头模块或目标板。请参见第 12.5 节“标准通信硬件”。
- 环回测试板 —— 验证仿真器是否正常工作。请参见第 12.7 节“环回测试板”。

图 1-1: 基本仿真器主机



其他硬件可单独订购：

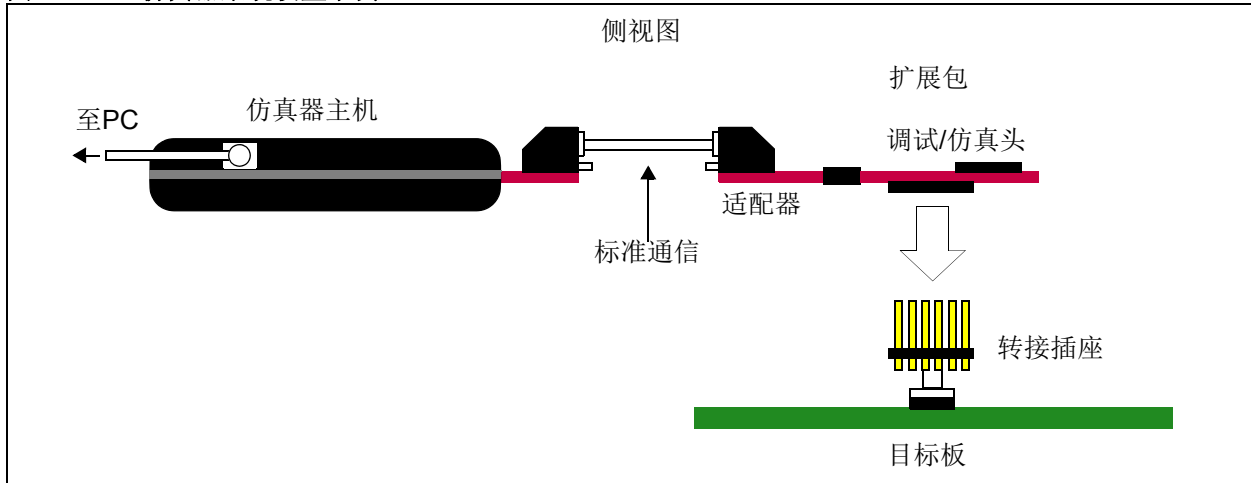
- 高性能工具包（AC244002） —— 用于高速 /LVDS 通信和 SPI 跟踪。
- 处理器 / 仿真扩展工具包 —— 提供特定于器件的调试 / 仿真头，以及其他硬件。
- 转接插座 —— 将转接头连接到目标板。
- 逻辑探头（ACICE0104） —— 用于外部触发信号和 I/O 端口跟踪。
- MPLAB REAL ICE 跟踪接口工具包（AC244006） —— 与支持指令跟踪的特定于器件接插模块（Plug-In Module, PIM）配合使用。
- MPLAB REAL ICE 隔离器单元（AC244005） —— 将仿真器与目标板光电隔离，对于大功率应用非常有用。
- MPLAB REAL ICE JTAG 适配器（AC244007） —— 在仿真器和目标 PIC32 器件之间提供 JTAG 通信。
- MPLAB REAL ICE 功率监视器（AC244008） —— 使仿真器可以监视目标板或器件的电流和电压。

## 1.5 仿真器系统

MPLAB REAL ICE 在线仿真器是一款由运行MPLAB X IDE 集成开发环境的PC控制的在线仿真器。仿真器可与调试/仿真头或具有片上调试/仿真电路的器件通信。器件通常与目标板直接相连，而调试/仿真头既可以与目标板直接相连也可以通过转接插座连接到目标板。

图1-2显示了一个仿真器系统配置的示例。

图1-2: 仿真器系统设置示例



注:

---

---

## 第2章 器件与功能支持

---

---

可以通过以下途径找到关于器件和工具的功能支持的信息：

- [开发工具选择器（DTS）](#) —— 按单个器件
- [器件与功能支持 HTML 表](#) —— 对于所有器件

### 2.1 开发工具选择器（DTS）

在 Microchip 网站上的 **Development Tools**（开发工具）部分中找到 **Development Tool Selector**（DTS，开发工具选择器）（复制并粘贴到浏览器中）：

<http://www.microchip.com/dts>

在 DTS 网页中，开始输入器件系列名称并从搜索框下面显示的列表中进行选择。选择器件后，单击“**Emulators and Debuggers**”（仿真器和调试器）选项卡显示工具（和仿真 / 调试头）支持的功能列表。

#### 注意事项

对于支持数据捕捉、运行时观察和本机跟踪的器件和工具：在高于 15 MIPS 的速度下，可能需要高性能工具包。

图 2-1: 开发工具选择器 (DTS) 网页


## Development Tools Selector - DTS

Search by Device Family Name:

### PIC16F1939 Development Tools

- Demo & Eval Boards 14
- Emulation & Debuggers 3**
- Programmers 4

### PICKit 3 In-Circuit Debugger ( PG164130 )



Microchip's PICKit™ 3 In-Circuit Debugger/Programmer uses in-circuit debugging logic incorporated into each chip with Flash memory to provide a low-cost hardware debugger and programmer. In-circuit debugging offers these benefits:

- Low cost
- Minimum of additional hardware needed for debug
- Expensive sockets or adapters are not required...

**Debug Features:**

- Stop watch:True
- Break on stack overflow:True
- Pgm-memory HW breakpoints:3
- Data-memory breakpoints:3
- WDT overflow:True
- Pass counter:True

**Header/Pak: AC244035 (required)**

**Alternate Header/Pak: AC244055 (required)**

## 2.2 器件与功能支持 HTML表

要查看 HTML 表中的功能支持，对于所有器件和工具：

1. 在 MPLAB X IDE 中，转到“Start Page”（起始页），“Learn & Discover”（学习与发现）选项卡，然后单击链接“Users Guide & Release Notes”（用户指南与发行说明）。您的默认浏览器将打开，其中会包含文档 *User's Guides, Release Notes and Support Documentation in the MPLAB® X IDE Installation*。
2. 在浏览器页面中，转到“Device & Feature Support”（器件与功能支持）部分。选择：
  - a) **Hardware Tool Debug Features by Device**（按器件列出的硬件工具调试功能）——按器件查看 Microchip 硬件工具（如 PICKIT™ 3）的调试功能（断点和跟踪等）。
  - b) **Simulator Debug Features by Device**（按器件列出的软件模拟器调试功能）——按器件查看 MPLAB X IDE 软件模拟器的调试功能（断点和跟踪等）。
  - c) **Simulator Peripheral Support by Device**（按器件列出的软件模拟器外设支持）——按器件查看 MPLAB 软件模拟器中所支持的外设。

**注：** 这些表很大，可能需要一些时间才能载入您的浏览器窗口。

3. 每个 HTML 表提供了 CSV（逗号分隔值）文档版本，以支持将数据导入电子表格程序并对数据排序。单击每个 HTML 表顶部的链接即可转到 CSV 文件。

注:



---

---

## 第3章 工作原理

---

---

### 3.1 简介

本章将简要介绍MPLAB REAL ICE在线仿真器系统的工作原理，旨在提供足够的信息，从而使用户能设计与仿真器兼容的目标板以进行仿真和编程操作。介绍在线仿真和编程的基本理论是为了使用户能够快速解决可能遇到的问题。

- [工具比较](#)
- [操作概述](#)
- [仿真器与PC的通信](#)
- [仿真器与目标板的通信](#)
- [跟踪连接](#)
- [使用仿真器进行调试](#)
- [调试的要求](#)
- [使用仿真器编程](#)
- [仿真器使用的资源](#)

# 适用于MPLAB X IDE的仿真器用户指南

## 3.2 工具比较

MPLAB REAL ICE 在线仿真器系统与 Microchip 其他调试工具在物理和操作上的差异如下。具体功能可能随器件的不同而有所不同（请参见在线帮助文件中的“Device and Feature Support”（器件与功能支持））。

表3-1: 调试工具比较

功能	MPLAB® REAL ICE™ 在线仿真器	MPLAB ICD 3 在线 调试器	PICKit™ 3 在线调试器
USB 速度	高速和全速	高速和全速	仅全速
USB 驱动程序	Microchip	Microchip	HID
USB 供电	✓	✓	✓
向目标板供电	✗ <sup>(1)</sup>	✓	✓
可编程的 VPP 和 VDD	✓	✓	✓
从目标板灌入 VDD 的电流	< 50 µA	< 50 µA	20 mA
流入目标板的最大电流	无 <sup>(1)</sup>	100 mA	30 mA
过压/过流保护	✓ (硬件)	✓ (硬件)	✓ (软件)
硬件断点	复杂	复杂	简单
跑表	✓	✓	✓
软件断点	✓	✓	✗
非易失性程序映像	✗	✗	✓ (512 KB)
串行 USB	✓	✓	✓
跟踪	✓	✗	✗
数据捕捉	✓	✗	✗
逻辑探头触发信号	✓	✗	✗
高速/LVDS 连接	✓	✗	✗
生产编程器	✓	✓	✗

注 1: 如果使用功率监视器附件, 则可以通过仿真器为目标板供电。

## 3.3 操作概述

仿真器通过 USB 端口连接到 PC，USB 端口用于通信和为仿真器供电（但不为目标板供电）。仿真器连接到目标应用来进行通信和数据采集，如跟踪。以下总结了可能的连接配置。

图3-1: 仿真器连接

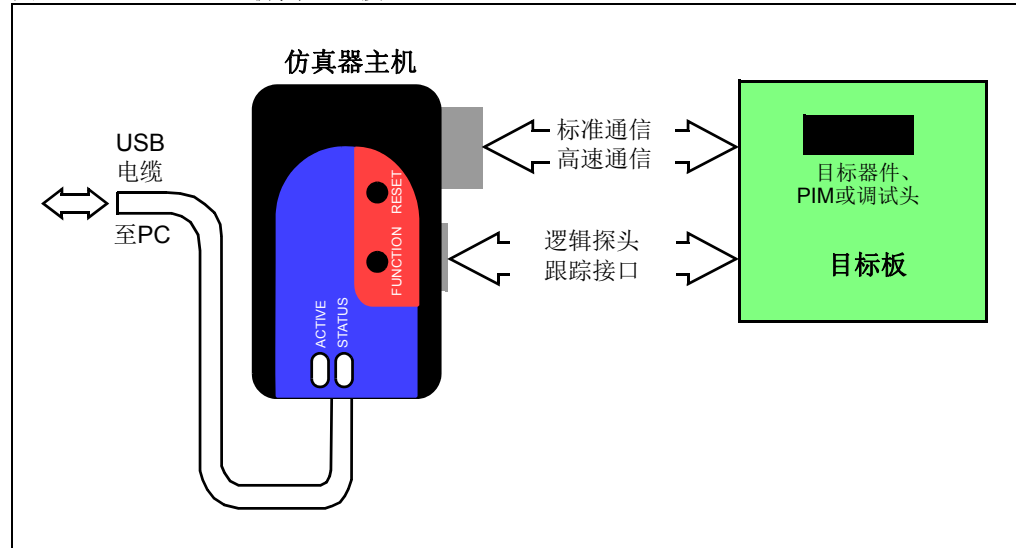


表 3-2: 仿真器件的连接

连接	调试支持 (1,2)	跟踪支持 (1,3)	支持速度
标准通信	数据捕捉，运行时观察	本机跟踪	15 MIPS 或更低 PIC32: 取决于器件
高速通信	数据捕捉，运行时观察	本机跟踪，SPI 跟踪	高于 15 MIPS PIC32: 取决于器件
逻辑端口探头 (4)	N/A	I/O 端口跟踪	取决于器件
MPLAB® REAL ICE™ 跟踪接口工具包 (逻辑端口)	N/A	指令跟踪	取决于器件

注 1: 支持取决于器件。请参见在线帮助。

2: 关于详细信息，请参见第 6.2 节“数据捕捉和运行时观察”。

3: 关于详细信息，请参见第 6.3.3 节“跟踪类型”或第 6.4 节“PIC32 指令跟踪——仅限 PIC32 MCU”。

4: 关于详细信息，请参见第 3.6.3 节“I/O 端口跟踪连接（逻辑端口）”。

## 3.4 仿真器与PC的通信


仿真器通过 USB 电缆供电和与 PC 进行通信。不能由仿真器通过 USB 电源为目标板供电。要通过仿真器为目标板供电，请使用 MPLAB REAL ICE 功率监视器 (AC244008)。


	<b>小心</b>
	通信故障。 在安装软件和 USB 驱动程序之前，不要连接硬件。

## 3.5 仿真器与目标板的通信

仿真器通过以下几节介绍的配置与目标板进行通信：

- 标准通信
- 高速 /LVDS 通信（高性能工具包）

	<b>小心</b>
	仿真器或目标板损坏。 不要在主机或目标板加电时更改硬件连接。

	<b>⚠ 危险</b>
	潜在电气危险。 如果您的应用使用不接地作为参考的交流线或高压电源，则应使用隔离电路和 MPLAB REAL ICE 隔离器单元 (AC244005)。

### 3.5.1 标准通信

仿真器系统可以配置为使用标准连接来向目标板传送调试和编程指令。这种 6 引脚连接与其他 Microchip 在线调试器使用的连接相同。

## 3.5.1.1 标准连接

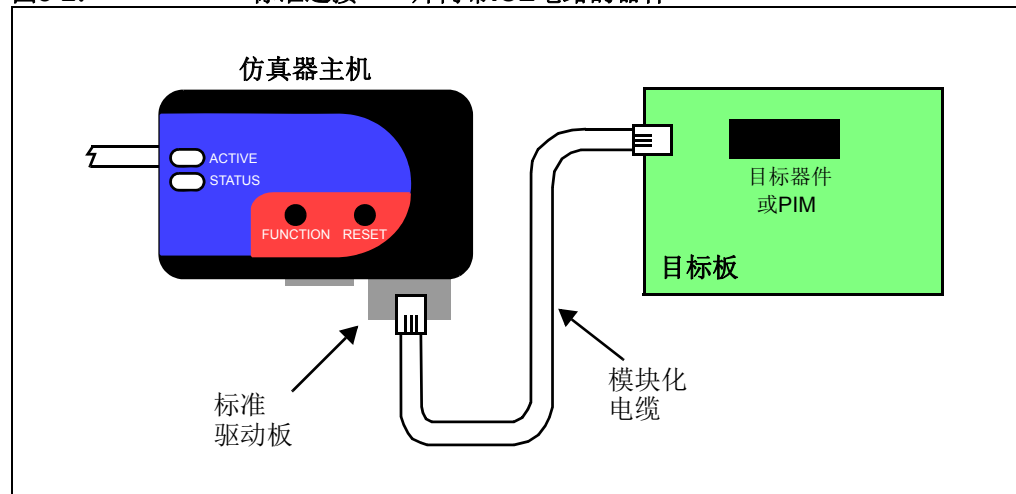
标准驱动板插入仿真器主机，将系统配置为与目标板进行通信。将模块化电缆的一端插入标准驱动板，另一端插入以下位置之一：

1. 目标板上的匹配插座，其中目标器件（或特定于器件的接插模块）位于目标板上，如图 3-2 所示
2. 然后，将标准适配器 / 调试头组合（可用作扩展工具包）插入到目标板，如图 3-3 所示。

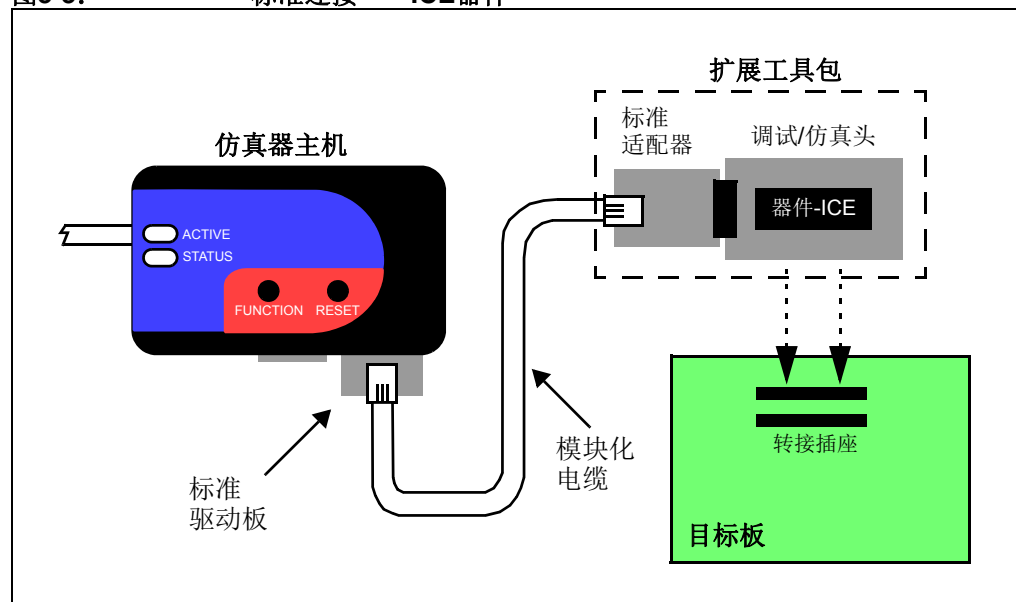
**注：** 早期的调试头使用 6 引脚（RJ-11）连接器，而不是 8 引脚连接器。在使用这些早期调试头的情况下，可以将电缆直接连接到仿真器。

要了解器件与调试头的调试功能，请参见位于 <http://www.microchip.com/dts> 的 DTS。关于硬件的更多信息，请参见第 12.5 节“标准通信硬件”。

**图3-2:** 标准连接——片内带ICE电路的器件



**图3-3:** 标准连接——ICE器件

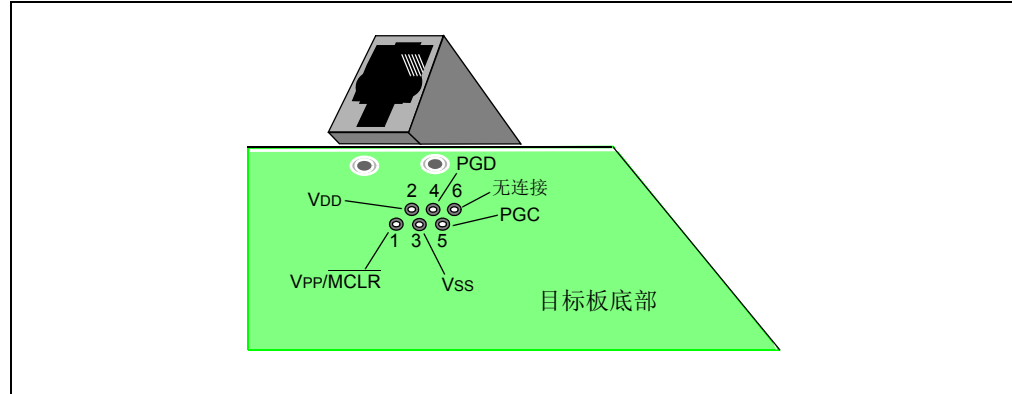


## 3.5.1.2 目标板上的标准连接器

要使用标准驱动板，可以使用模块化接口（6 芯）电缆将 MPLAB REAL ICE 在线仿真器连接到目标器件。在图 3-4 中，从目标 PCB 底部视角显示了连接器的引脚编号。

**注：** 仿真器和目标板上的电缆连接彼此构成镜像；即电缆一端的引脚 1 连接到电缆另一端上的引脚 6。请参见第 12.5.2.1 节“模块化电缆规范”。

图3-4: 目标板上的标准连接



## 3.5.2 高速 /LVDS 通信（高性能工具包）

仿真器系统可以配置为使用高速 /LVDS 连接来向目标板传送调试和编程指令。与标准通信相比，这种形式的通信具有以下特性。

- 通过低压差分信号（Low-Voltage Differential Signal, LVDS）技术消除噪声，从而可以支持：
  - 大于 15 MIPS 的通信速度，用于进行数据捕捉、运行时观察和本机跟踪
  - 仿真器和目标板相隔更长的距离
  - 在噪声环境下工作
- 两个用于 SPI 跟踪的额外引脚

要进行高速 /LVDS 通信，需要高性能工具包（AC244002）。关于连接信息，请参见《MPLAB<sup>®</sup> REAL ICE<sup>™</sup> 在线仿真器高性能工具包用户指南》（DS50002528A\_CN）。

### 3.5.3 目标板连接电路

图 3-5 显示了 MPLAB REAL ICE 在线仿真器通过目标板连接器连接到目标板上的器件。这种互连很简单。如果遇到任何问题，通常是由于其他连接或这些关键线路上的元件干扰仿真器操作而引起的，如第 3.5.4 节“目标电路设计注意事项”所述。

图3-5: 与目标电路的标准连接

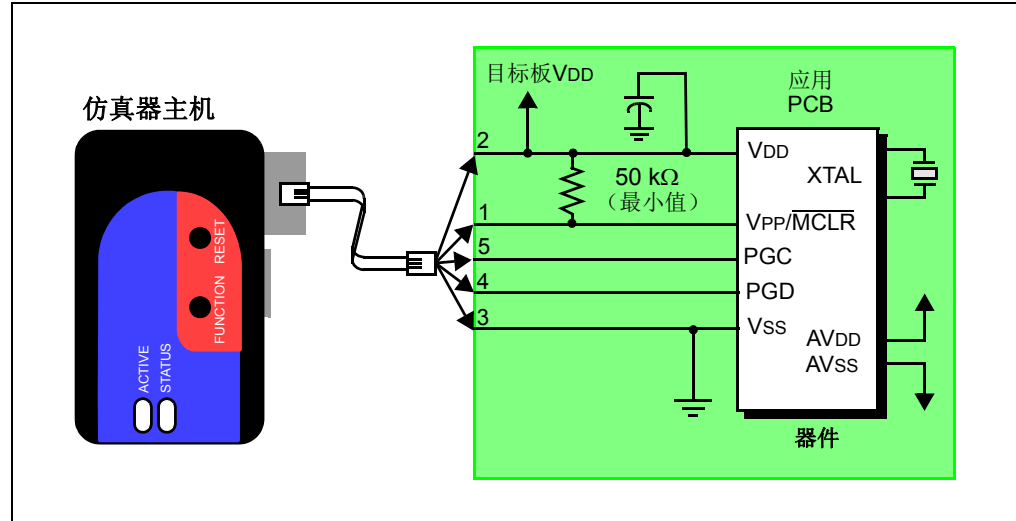


表 3-3: 目标板连接器与相关器件 I/O

引脚编号	器件 I/O	说明
1	VPP/MCLR	为了对器件进行编程和调试，仿真器需要接上 VPP。
2	VDD	仿真器会检测目标板 VDD，以便进行用于目标板低电压操作的电压转换，以及由此检测器件。如果仿真器未在其 VDD 线上检测到电压，则它不会与器件连接。 <b>注：</b> 仿真器不为目标板供电。
3	VSS	仿真器会检测目标板 VSS。 <b>注：</b> 仿真器不提供目标板 VSS 或地。
4	PGD	为了对器件进行编程和调试，仿真器需要接上 PGD 和 PGC。
5	PGC	

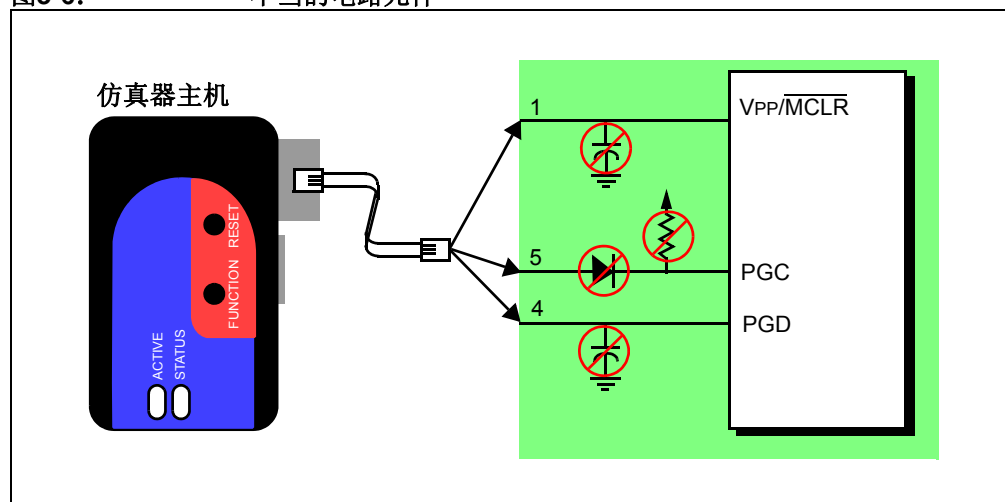
表 3-4: 器件 I/O 上的电路

器件 I/O	说明
VPP/MCLR	应在 VPP/MCLR 线到 VDD 之间连接一个上拉电阻（最小 50 kΩ），以便可以通过将该线选通为低电平来复位器件。
XTAL	要将仿真器用作调试器，目标器件必须使用振荡器运行。
AVDD, AVSS	并非所有器件都具有 AVDD 和 AVSS 线，但如果目标器件上存在这两个引脚，则必须将它们连接到适当的电压，以便仿真器可以正常工作。这也适用于稳压器引脚（例如，PIC24FJ MCU 上的 ENVREG/DISVREG）。
VDD, VSS, AVDD, AVSS	通常，建议根据器件数据手册将 VDD/AVDD 和 VSS/AVSS 线连接到适当的电压。对于具有 VCAP 引脚的器件（如 PIC18FXXJ 器件），应尽可能靠近 VCAP 引脚放置一个容值适当的电容。

## 3.5.4 目标电路设计注意事项

图 3-6 显示了在仿真器的三根有效线上一些示例元件会影响 MPLAB REAL ICE 在线仿真器系统的正常工作。

图3-6: 不当的电路元件



- 请勿在 MCLR 上使用电容 —— 它们会使 VPP 上电压无法快速变化。
- 请勿在 PGC/PGD 上使用上拉电阻 —— 因为在仿真器中这些线具有 4.7 kΩ 的下拉电阻，所以它们会产生分压。
- 请勿在 PGC/PGD 上进行复用 —— 它们专用于与仿真器的通信。
- 请勿在 PGC/PGD 上使用电容 —— 在编程和调试通信期间，这些电容会使数据和时钟线上电平无法快速变化。
- 请勿在 PGC/PGD 上使用二极管 —— 它们会使仿真器和目标器件之间无法进行双向通信。
- 请勿超过推荐的电缆长度 —— 关于可接受的电缆长度，请参见第 12.5.2 节“模块化电缆和连接器”。

《多工具设计忠告》(DS51764C\_CN) 中提供了更多的设计信息。

关于其他操作问题，请参见本文档中的以下章节：

- 第 8 章“常见问题解答 (FAQ)”
- 第 9 章“消息”
- 第 9.3.6 节“调试失败纠正措施”(无法调试的首要原因)
- 第 12.7 节“环回测试板”



### 3.6 跟踪连接

根据您选择的器件，在选择仿真器作为调试工具时，可能会有一种或多种跟踪功能可供使用。

#### 3.6.1 本机跟踪连接

使用本机跟踪不需要额外的连接。通信连接将使用 PGD/PGC/EMUC/EMUD 引脚来传送跟踪信息。但是，所选择的器件必须具有该功能。如果它不具有该功能，可以使用其他跟踪方法之一。

关于这种跟踪类型的更多信息，请参见第 6.3.3.1 节“本机跟踪”。

#### 3.6.2 SPI 跟踪连接（高速 /LVDS 连接）

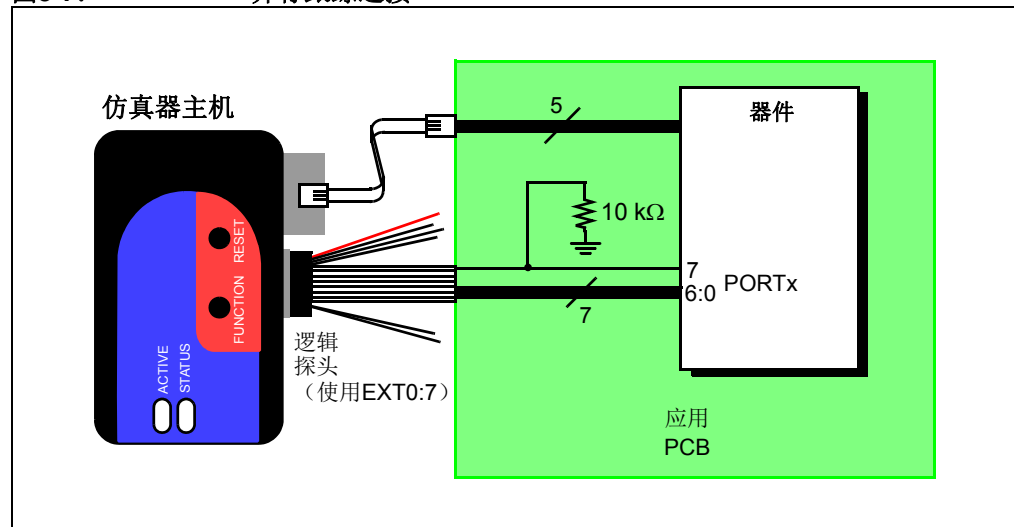
串行跟踪是一种可选的跟踪功能，它仅在使用高性能工具包（AC244002）时可用。关于详细信息，请参见《MPLAB® REAL ICE™ 在线仿真器高性能工具包用户指南》（DS50002528A\_CN）。

关于这种跟踪类型的更多信息，请参见第 6.3.3.2 节“SPI 跟踪”。

#### 3.6.3 I/O 端口跟踪连接（逻辑端口）

并行跟踪可使用器件 8 引脚 I/O 端口和仿真器逻辑探头来实现。它可以提供更高的跟踪速度和更大数据量，但仿真器到目标板的距离受到逻辑探头长度的限制。图 3-7 显示了这些额外的连接。

图3-7: 并行跟踪连接



对于该跟踪配置，有 7 条数据传输线和 1 条时钟传输线。PORTx 必须是具有 8 个引脚（都可用于跟踪）的端口。该端口不必是物理端口，而是可以由来自多个端口的引脚组成。端口引脚不得与当前使用的 PGC 和 PGD 引脚复用。

进行允许的 PORTx 配置：

1. 在 Projects（项目）窗口中右键单击项目，然后选择“Properties”（属性）。
2. 在 Project Properties（项目属性）窗口中，单击“REAL ICE”类别。
3. 从下拉列表中选择“Trace and Profiling”（跟踪与性能分析）选项类别。
4. 在“Data Collection Selection”（数据采集选择）下，选择您的器件支持的跟踪，例如“User Instrumented Trace”（用户插装跟踪）。
5. 在“Communications Medium”（通信介质）下，选择“I/O Port”（I/O 端口）。
6. 在“I/O Port Selection”（I/O 端口选择）下，从列表中选择您的端口配置。

下表给出了一个基本的配置。

**表 3-5: I/O 端口跟踪连接示例**

PORTx 引脚	逻辑探头引脚 <sup>(1)</sup>	内容
0	EXT0	数据
1	EXT1	数据
2	EXT2	数据
3	EXT3	数据
4	EXT4	数据
5	EXT5	数据
6	EXT6	数据
7	EXT7 <sup>(2)</sup>	时钟

注 1: 关于引脚说明，请参见第 12.4.4 节“逻辑探头 / 外部触发接口”。

2: 使用 10 kΩ 下拉电阻来降低噪声。

如第 3.5.4 节“目标电路设计注意事项”所述，除非另有说明，否则不要在端口引脚上使用上拉或下拉电阻、电容或二极管。

关于这种跟踪类型的更多信息，请参见第 6.3.3.3 节“I/O 端口跟踪”。

## 3.6.4 指令跟踪连接

指令跟踪是用于捕捉由器件执行的每条指令的非侵入性硬件跟踪功能。存在两种类型的指令跟踪：

- [PIC32 指令跟踪](#)
- [实时硬件指令跟踪](#)

### 3.6.4.1 PIC32指令跟踪

PIC32 指令跟踪仅对于 PIC32 MCU 器件可用。此外，只有部分 PIC32 MCU 器件具有跟踪功能。关于详细信息，请查询器件数据手册。

要使用该跟踪功能，需要以下硬件：

- 包含支持跟踪的器件并具有跟踪端口的 PIC32 接插模块（PIM）
- MPLAB® REAL ICE™ 跟踪接口工具包（AC244006），其中包含一条 12 英寸跟踪电缆和一个跟踪适配器板

如果没有跟踪电缆，可以使用逻辑探头。按下面所示连接它们。

**表 3-6: 逻辑探头连接**

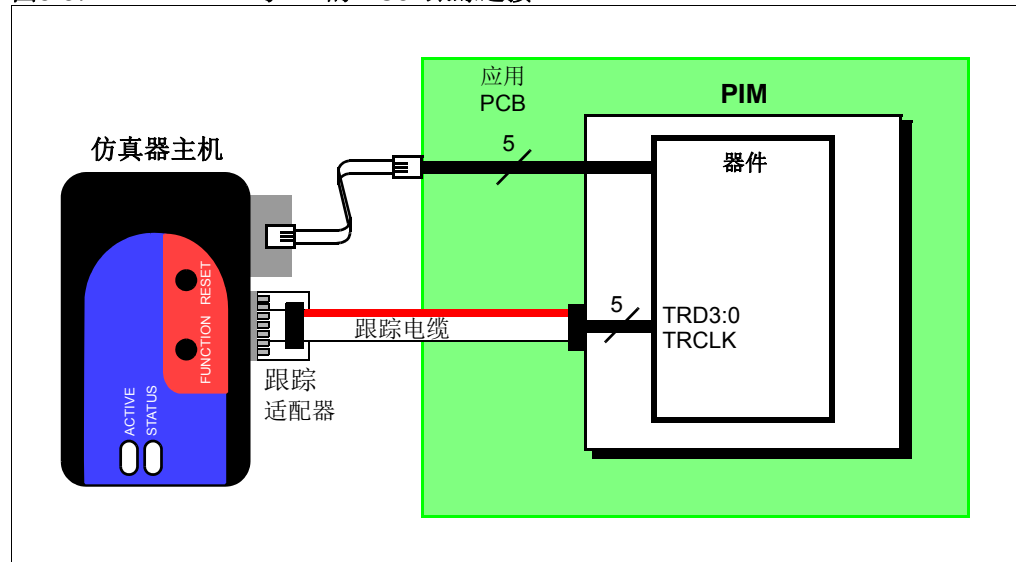
逻辑探头端口引脚 <sup>(1)</sup>		PIM 跟踪引脚 <sup>(2)</sup>	
编号	名称	编号	名称
4	TCLK	1	TRCLK
12	EXT0 (TRIG1)	3	TRD0
11	EXT1 (TRIG2)	5	TRD1
10	EXT2 (TRIG3)	7	TRD2
9	EXT3 (TRIG4)	9	TRD3

注 1: 更多信息，请参见第 12.4.4 节“逻辑探头 / 外部触发接口”。

注 2: 更多信息，请参见第 6.4.4 节“跟踪硬件规范”。

要使用 PIC32 指令跟踪功能，请参见第 6.4 节“PIC32 指令跟踪——仅限 PIC32 MCU”。

**图3-8: 与PIM的PIC32跟踪连接**



## 3.6.4.2 实时硬件指令跟踪

实时硬件指令跟踪是可以捕捉和分析的程序执行流的实时转储。这种跟踪在某些仿真头（-ME2 器件）上可用。

关于详细信息，请参见《仿真扩展包（EEP）和仿真头用户指南》（DS50002243B\_CN）。

## 3.7 使用仿真器进行调试

使用 MPLAB REAL ICE 在线仿真器系统作为调试器需要两个步骤。

1. 第一个步骤要求将应用程序编程到目标器件中。
2. 第二个步骤需要使用目标闪存器件的内部在线调试硬件来运行和测试应用程序。

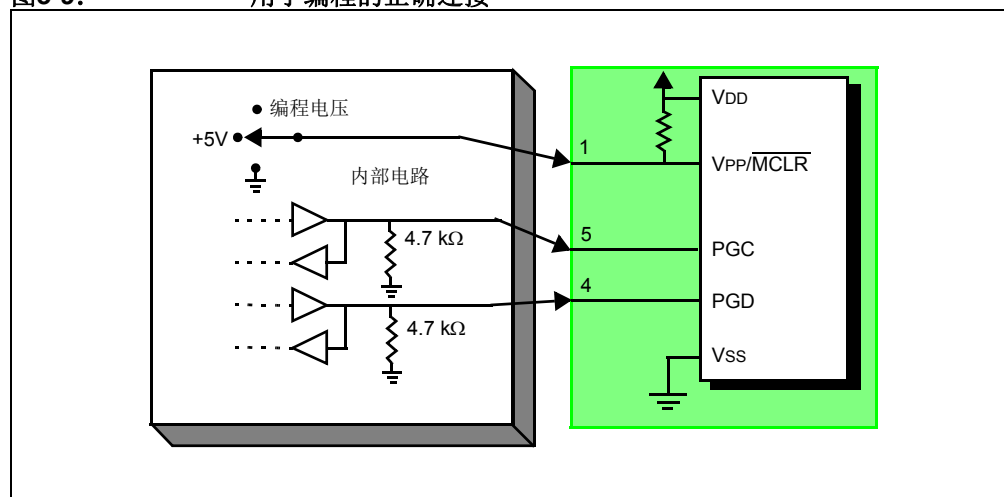
这两个步骤与以下 MPLAB IDE 操作直接相关：

- 将代码编程到目标器件中并激活特殊调试功能（详细信息请参见下一节）。
- 使用断点等功能调试代码。

如果无法对目标器件正确编程，MPLAB REAL ICE 在线仿真器将无法进行调试。

图 3-9 显示了进行编程和调试所需的基本连接。请注意，它与图 3-5 相同，但为清晰起见，其中未显示来自仿真器的 VDD 和 VSS 线。

图3-9: 用于编程的正确连接



这是 MPLAB REAL ICE 在线仿真器主机的一些内部接口电路的简化图。对于编程，目标器件上不需要有时钟，但必须为其供电。在编程时，仿真器在 VPP 上施加编程电压，并在 PGC 上发送时钟脉冲，通过 PGD 发送串行数据。为了验证器件是否已正确编程，可以向 PGC 发送时钟，并从 PGD 读回数据。这符合开发中所使用器件的在线串行编程（In-Circuit Serial Programming™, ICSP™）协议。更多详细信息，请参见器件编程规范。

### 3.8 调试的要求

要使用MPLAB REAL ICE在线仿真器系统进行调试（设置断点和查看寄存器等），以下几个关键因素必须正确：

- 仿真器必须连接到一台PC。仿真器必须通过USB电缆由PC为其供电，并且仿真器必须通过USB电缆与MPLAB IDE软件通信。关于详细信息，请参见第5章“基本调试功能”。
- 如图所示，必须通过模块化接口电缆（或类似电缆）将仿真器连接到目标器件的VPP、PGC和PGD引脚。仿真器和目标器件之间的VSS和VDD引脚也必须连接起来。
- 目标器件必须具有电源和一个正常运行的内部或外部振荡器。无论是什么原因使目标器件不工作，MPLAB REAL ICE在线仿真器都无法进行调试。
- 必须对目标器件的配置字正确编程：
  - 振荡器配置位应与RC和XT等相对应，具体取决于目标板的设计。
  - 看门狗定时器在某些器件上默认使能，需要禁止。
  - 不要使能目标器件的代码保护功能。
  - 不要使能目标器件的表读保护功能。
  - 对于一些具有多对PGC/PGD的器件，需要配置正确的PGC/PGD对。这一点仅适用于调试，因为编程时可以使用任意PGC/PGD对。

#### 3.8.1 进入调试的操作顺序

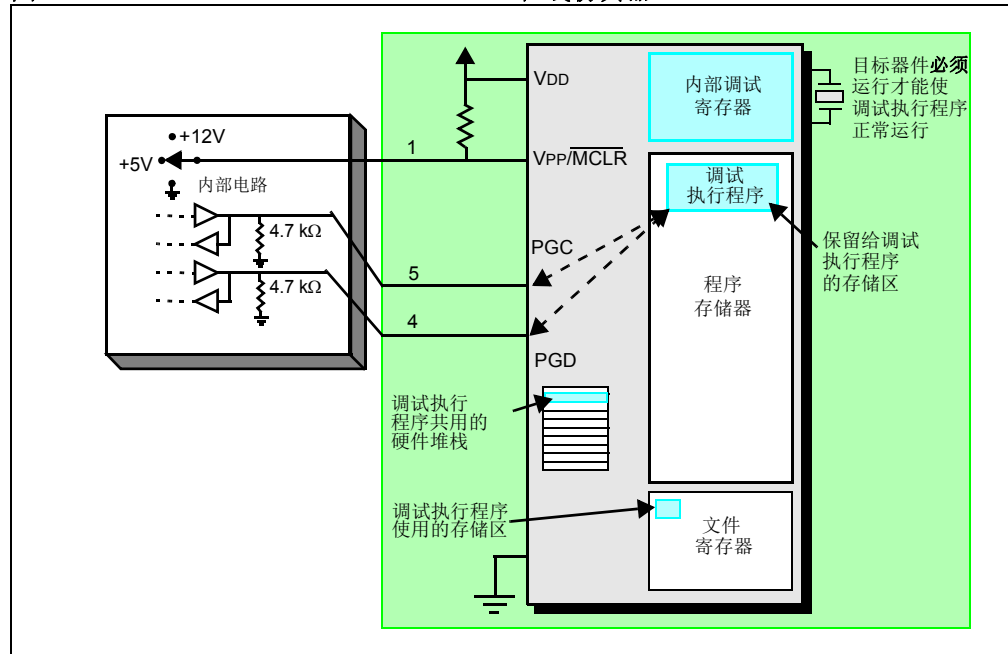
如果满足了调试要求（见上一节），则当MPLAB REAL ICE在线仿真器设置为当前工具（File>Project Properties（文件>项目属性），“Hardware Tool”（硬件工具）类别）时，可以执行下列操作：

- 选择Debug>Debug Project（调试>调试项目）时，应用程序代码就会通过前面所述的ICSP协议烧写到器件的存储器中。
- 一个小的“调试执行”程序被载入到目标器件的程序存储器的高地址段。由于调试执行程序必须驻留在程序存储器中，因此应用程序不能使用这段保留的空间。某些器件具有专用于调试执行程序的特殊存储区。关于详细信息，请查看器件数据手册。
- 通过MPLAB IDE使能目标器件中的特殊“在线调试”寄存器。这使得调试执行程序能被仿真器激活。关于器件保留资源的更多信息，请参见在线帮助文件。
- 目标器件运行在调试模式下。

## 3.8.2 调试细节

图3-10显示了准备好调试的MPLAB REAL ICE在线仿真器系统。

图3-10: MPLAB® REAL ICE™在线仿真器



一般来说，为判断应用程序能否正确运行，可在程序代码中设置断点。当通过MPLAB IDE的用户界面设置断点时，断点的地址保存在目标器件的特殊内部调试寄存器中。PGC和PGD引脚上的命令直接传送到这些寄存器来设置断点地址。

接下来，通常在MPLAB IDE中选择 *Debug>Debug Project* 功能。仿真器随后告知调试执行程序运行。目标器件将从复位向量处开始执行程序，直到程序计数器到达先前存储在内部调试寄存器中的断点地址为止。

在执行了断点地址处的指令之后，目标器件的在线调试机制就会启动，并向调试执行程序传送控制信号（很像中断），同时用户的应用程序也暂停执行。仿真器通过PGC和PGD与调试执行程序通信，获取断点状态信息并将这些信息传送回MPLAB IDE。MPLAB IDE接着向仿真器发送一系列查询以获取关于目标器件的信息，如文件寄存器的内容和CPU的状态。这些查询最终由调试执行程序来执行。

调试执行程序像程序存储器中的应用程序一样运行。它使用堆栈的某些地址单元存放临时变量。如果器件由于任何原因（例如，没有振荡器、电源连接错误或目标板短路等）无法运行，则调试执行程序将无法与MPLAB REAL ICE在线仿真器通信，并且MPLAB IDE会发出一条错误消息。

停止执行的另一种方式是选择 *Debug>Pause* (调试> 暂停)。这将触发PGC和PGD线，从而使目标器件的在线调试机制从执行程序存储器中的用户代码切换为执行调试执行程序。目标应用程序暂停，MPLAB IDE通过仿真器与调试执行程序通信以获得目标器件的状态。

## 3.9 使用仿真器编程

使用MPLAB REAL ICE在线仿真器作为编程器来编程生产器件（即不在调试头上的器件）。将MPLAB REAL ICE在线仿真器设置为当前工具（*File>Project Properties, Hardware Tool*）以执行这些操作：

- 选择 *Run>Run Project* (运行> 运行项目) 时，应用程序代码就会通过上节所述的ICSP协议烧写到器件的存储器中。编程时不需要目标振荡器时钟，并且处理器的所有模式都能被编程，包括代码保护、使能看门狗定时器以及表读保护。
- 可将一个小的“编程执行”程序载入到某个目标器件的程序存储器的高地址段。对于存储容量较大的器件，这可以提高编程速度。
- 通过MPLAB IDE禁止目标器件中的特殊“在线调试”寄存器以及所有调试功能。这意味着不能设置断点，也不能查看或改变寄存器的内容。
- 目标器件运行在发布模式下。作为编程器时，仿真器只能通过翻转MCLR线电平来复位和启动目标器件。

## 3.10 仿真器使用的资源

关于您的器件的仿真器所使用资源的完整列表，请参见MPLAB IDE中针对MPLAB REAL ICE在线仿真器的在线帮助文件。

注:





---

---

## 第2部分——特性

---

---

第4章 使用仿真器 .....	43
第5章 基本调试功能.....	47
第6章 特定调试功能.....	51

注:

## 第4章 使用仿真器

### 4.1 简介

本章讨论如何开始使用MPLAB REAL ICE在线仿真器。

- [安装和设置](#)
- [固件升级](#)
- [仿真器功能](#)
- [快速调试/编程参考](#)
- [调试器/编程器限制](#)

### 4.2 安装和设置

关于安装MPLAB X IDE以及设置仿真器以与IDE配合使用的详细信息，请参见帮助文件“Getting Started with MPLAB X IDE”（MPLAB X IDE入门）。

**概述：**

1. 安装MPLAB X IDE。
2. 按照说明安装USB驱动程序。
3. 连接至PC。关于目标连接和跟踪连接的信息，请参见[第3章“工作原理”](#)。

**注：** 不能由仿真器为目标板供电。  
但是，可以将功率监视器附件与仿真器搭配使用来为目标板供电。请参见《MPLAB® REAL ICE™ 在线仿真器功率监视器用户指南》（DS50002532A\_CN）。

4. 安装开发时要使用的语言工具套件/编译器。
5. 启动MPLAB X IDE。
6. 使用New Project（新建项目）向导（[File>New Project](#)（文件>新建项目））将“Real ICE”仿真器添加到项目中。
7. 使用Project Properties对话框（[File>Project Properties](#)）设置仿真器选项。
8. 通过[Run>Run Project](#)运行项目（编译和运行）。

**注意事项：**

1. 特定说明适用于在Windows®系统上安装USB驱动程序。关于详细信息，请参见MPLAB X IDE文档。
2. 每个仿真器均包含一个唯一标识符，首次安装时，无论使用哪个计算机USB端口都会由操作系统（OS）识别。
3. 运行时（运行或调试运行），MPLAB X IDE会连接到硬件工具。要始终与硬件工具相连（像MPLAB IDE v8那样），请单击[Tools>Options](#)（工具>选项）中的**Embedded**（已安装工具）按钮，然后在**Generic Settings**（通用设置）选项卡中，选中**Keep hardware tool connected**（硬件工具保持连接状态）复选框。
4. 现在必须在代码中设置配置位。可以在Configuration（配置）窗口中设置配置位（[Window>PIC Memory Views>Configuration Bits](#)（窗口>PIC存储器视图>配置位）），然后单击“Generate Code”（生成代码）。

## 4.3 固件升级

通常，您不需要执行手动固件升级。

安装某个版本MPLAB X IDE时，相关版本的MPLAB REAL ICE在线仿真器固件（操作系统）也将安装到您的系统中。下次使用仿真器时，MPLAB X IDE将自动升级仿真器上的固件。可以在Output（输出）窗口中查看进度。

如果想要使用的仿真器固件版本不是您所使用MPLAB X IDE版本默认使用的固件版本，则必须按照下面几节所述手动安装固件。

### 4.3.1 默认固件文件位置

默认固件文件（.jam）存储在调试工具的 Java 归档（JAR）文件（即，REALICE.jar）中。

固件的默认位置为：

<安装路径>MPLABX/vx.xx/mplab\_ide/mlablibs/modules/ext/REALICE.jar

### 4.3.2 手动固件安装

对于使用仿真器的活动项目：

1. 选择 *File>Project Properties*，打开Project Properties窗口。
2. 单击Real ICE类别并从Option Categories下拉菜单中选择Firmware（固件）。
3. 取消选中“Use Latest Firmware”（使用最新固件）。
4. 单击“Firmware File”（固件文件）字段，将显示“Press to browse for a specific firmware file”（按下以浏览特定固件文件）。将出现一个长按钮。单击该按钮。
5. 在“Directories”（目录）下的对话框中，浏览至包含您所需固件文件（REALICE.jar）的文件位置。选择该文件。
6. 在“Firmware Files”下的对话框中，选择固件文件（.jam）并单击**OK**（确定）。
7. 在Project Properties窗口中，单击**Reset**（复位）。

### 4.3.3 安装固件时通信中断

如果在安装固件时仿真器与PC断开，请重新连接仿真器。它将开始擦除已写入的内容，以便重新启动。该擦除操作将持续约75秒（1分15秒）。请耐心等待。

您将看到：

- 橙色忙碌灯点亮约25秒
- 灯变为红色并点亮约25秒
- 灯再次变成橙色并点亮约25秒

灯再次变成红色时，MPLAB IDE将识别器件并启动恢复过程，即开始下载固件。

## 4.4 仿真器功能

第5章“基本调试功能”中提供了使用所有器件通用的仿真器功能的相关信息。

第6章“特定调试功能”中提供了使用专用于特定器件系列的仿真器功能的相关信息。

## 4.5 快速调试/编程参考

将MPLAB REAL ICE在线仿真器用作调试或编程工具时，可使用下表作为快速参考。关于正确的仿真器设置和配置的信息，请参见前面的章节。

表4-1: 调试和编程操作

项目	调试	编程
所需硬件	PC和目标应用（Microchip演示板或您自己的设计）。	
	仿真器主机、USB电缆、通信驱动板和电缆。	
	带片内调试电路的器件或带特殊的-ME2/-ICE/-ICD器件的调试头。	器件（带或不带片内调试电路）。
MPLAB X IDE选择	Project Properties, REAL ICE as Hardware Tool（REAL ICE作为硬件工具）	
	<u>Debug&gt;Debug Run</u> （调试>调试运行）	Program Target Project（编程目标项目）工具栏按钮
编程操作	将应用程序代码烧写到器件中。根据Project Properties对话框上的选择，这可以是程序存储器的任何范围。此外，其他调试资源将保留。在MPLAB X IDE的Start Page上，单击“Release Notes and Support Documentation”查找为仿真器保留的资源。	将应用程序代码烧写到器件中。根据Project Properties对话框上的选择，这可以是程序存储器的任何范围。
可用的调试功能	目标器件可用的所有调试功能——断点和跟踪等。	N/A
命令行操作	N/A	使用REALICECMD，默认位于： <安装路径>\MPLABX\vx.xx\mplab_ipe
序列号快速批量编程（Serial Quick-Time Programming, SQTP）	N/A	右键单击调试的项目并选择“Export Hex”（导出Hex）。将hex文件导入MPLAB IPE中并使用MPLAB PM3来对器件执行SQTP。

## 4.6 调试器/编程器限制

关于器件仿真器限制的完整列表，请参见MPLAB IDE中针对MPLAB REAL ICE在线仿真器的在线帮助文件。

注:

---

---

## 第5章 基本调试功能

---

---

### 5.1 简介

下面将介绍基本调试功能。

- [启动和停止仿真](#)
- [查看处理器存储器和文件](#)
- [断点和跑表](#)
- [器件调试功能](#)

稍后将在[第6章“特定调试功能”](#)中讨论仅专门适用于仿真器的调试功能。

### 5.2 启动和停止仿真

要在MPLAB X IDE中调试应用程序，必须创建一个包含源代码的项目，以便按照如下所述编译代码、将代码烧写到器件中并执行：

- 要运行代码，请选择 [Debug>Debug Project](#) 或 Run（运行）工具栏中的 **Debug Project**。
- 要暂停代码，请选择 [Debug>Pause](#) 或 Debug 工具栏中的 **Pause**。
- 要再次运行代码，请选择 [Debug>Continue](#)（调试 > 继续）或 Debug 工具栏中的 **Continue**。
- 要单步执行代码，请选择 [Debug>Step Into](#)（调试>单步进入）或 Debug 工具栏中的 **Step Into**。注意不要单步进入 Sleep 指令，否则必须执行处理器复位来恢复仿真。
- 要单步跳过某一代码行，请选择 [Debug>Step Over](#)（调试>单步跳过）或调试工具栏中的 **Step Over**。
- 要结束代码执行，请选择 [Debug>Finish Debugger Session](#)（调试>完成调试器会话）或 Debug 工具栏中的 **Finish Debugger Session**。
- 要对代码执行处理器复位，请选择 [Debug>Reset](#)（调试>复位）或 Debug 工具栏中的 **Reset**。根据器件的不同，可能有其他复位（如POR/BOR、MCLR和系统复位）可用。

## 5.3 查看处理器存储器和文件

MPLAB X IDE 提供了几个窗口来查看调试信息和各处理器存储器信息。可在 **Window**（窗口）菜单中进行选择。关于使用这些窗口的帮助信息，请查看 **MPLAB X IDE** 在线帮助。

- **Window>PIC Memory Views**（窗口>PIC存储器视图）——查看不同类型的器件存储器

根据所选器件，存储器类型包括程序存储器、SFR和配置存储器等。

- **Window>Debugging**（窗口>调试）——查看调试信息

从变量、观察、调用堆栈、断点、跑表和跟踪中选择。（接下来的章节中将更全面地讨论跟踪。）

要查看源代码，在 **Projects** 窗口中找到要查看的源代码文件并双击以在 **Files**（文件）窗口中打开。该窗口中的代码根据所选处理器和编译工具采用不同颜色标识。要更改颜色标识的样式，在 **Tools>Options** 的 **Fonts & Colors**（字体和颜色）中，选择 **Syntax**（语法）选项卡。

关于编辑器的更多信息，请参见 **MPLAB X IDE** 在线帮助的“**Editor**”（编辑器）部分。

## 5.4 断点和跑表

使用断点在代码中的指定行暂停代码执行。使用跑表和断点为代码执行计时。

- [断点资源](#)
- [硬件断点或软件断点选择](#)
- [断点和跑表的使用](#)

### 5.4.1 断点资源

对于 16 位器件，断点、数据捕捉和运行时观察均使用相同的资源。因此，可用的断点数实际上就是可用的组合断点/触发数。

对于 32 位器件，断点使用的资源与数据捕捉和运行时观察不同。因此，可用的断点数与可用的触发数无关。

可用和/或已用的硬件断点和软件断点数在 **Dashboard**（仪表板）窗口（**Window>Dashboard**（窗口>仪表板））中显示。关于该功能的更多信息，请参见 **MPLAB IDE** 文档。并非所有器件都有软件断点。

关于断点操作的限制（包括每个器件硬件断点的常规数目和硬件断点 skidding 量），请参见在线帮助文件中的“**Limitations**”（限制）部分。



## 5.4.2 硬件断点或软件断点选择

要选择硬件断点或软件断点：

1. 在 **Projects** 窗口中选择项目。然后，选择 **File>Project Properties**；或右键单击，选择 **“Properties”**。
2. 在 **Project Properties** 对话框的 **Categories**（类别）下，选择 **“REAL ICE”**。
3. 在 **“Option Categories”**（选项类别）下，选择 **“Debug Options”**（调试选项）。
4. 选中 **“Use software breakpoints”**（使用软件断点）以使用软件断点。取消选中以使用硬件断点。

**注：** 使用软件断点进行调试会影响器件的耐用性。因此，建议不要将以这种方式使用过的器件用作生产器件。

为帮助您确定使用哪类断点（硬件或软件），下表对每种断点的特性进行了比较。

**表5-1： 硬件断点与软件断点**

特性	硬件断点	软件断点
断点数	有限	无限
断点写入到*	内部调试寄存器	闪存程序存储器
断点应用于**	程序存储器/数据存储器	仅程序存储器
设置断点的时间	最短	取决于振荡器速度、编程闪存存储器的时间以及页大小
断点 skidding	大多数器件。关于详细信息，请参见在线帮助的“Limitations”部分。	否

\* 器件中写入断点信息的位置。

\*\* 适用于断点的器件功能类型。这是设置断点的位置。

## 5.4.3 断点和跑表的使用

断点用于暂停代码的执行。要确定两个断点之间的时间，请使用跑表。

关于如何设置及使用断点和跑表的信息，请参见 **MPLAB X IDE** 帮助。

## 5.5 器件调试功能

除前面提到的调试功能外，器件还具有其他调试功能，例如“暂停时冻结”功能允许在暂停时冻结所选外设。关于这些功能的更多信息，请参见 **MPLAB X IDE** 文档。

注:

---

---

## 第6章 特定调试功能

---

---

### 6.1 简介

下列调试功能或调试功能的实现特定于此仿真器。部分调试功能依赖于其他调试功能。关于基本调试功能的信息，请参见第5章“基本调试功能”。

- 数据捕捉和运行时观察
- 插装跟踪
- PIC32 指令跟踪——仅限 PIC32 MCU
- 跳转跟踪——仅限 EP 器件
- PC 采样——仅限 8 位和 16 位 MCU
- PC 性能分析——仅限 32 位 MCU
- 函数级别性能分析
- 应用程序输入/输出
- 外部触发信号
- 额外调试功能

### 6.2 数据捕捉和运行时观察

*数据捕捉*为器件的片内调试功能。当捕捉数据地址寄存器中的值与SFR寄存器地址匹配时，会出现触发信号并发生数据捕捉。用户无法直接访问数据捕捉功能；必须选择一个使用数据捕捉功能的应用程序，例如DMCI插件。

*运行时观察*是在Watches（观察）窗口中选择的符号，这些符号随程序的运行而变化。可以在Watches窗口中选择运行时符号。关于详细信息，请参见MPLAB X IDE文档。

#### 8位和16位器件

并非所有8位和16位器件都支持数据捕捉和/或运行时观察。在线帮助或开发工具选择器DTS）（<http://www.microchip.com/developmenttools/>）中列出了器件支持的功能。

观察符号时，8位PIC器件只能观察8位变量，16位PIC器件只能观察16位变量。

对于8位和16位器件，数据捕捉、运行时观察和硬件断点使用相同的寄存器/资源。例如，如果一个符号使用数据捕捉资源，则将无法对该符号使用硬件断点或运行时观察资源。

当数据捕捉和运行时观察的速度高于15 MIPS时，可能需要高性能工具包。实际速度可能随布线、噪声和类似因素的不同而有所不同。

## 32 位器件

并非所有 32 位器件都支持数据捕捉和/或运行时观察。在线帮助中列出了器件支持的功能。

对于 PIC32 器件，硬件断点不使用数据捕捉和运行时观察资源。但数据捕捉和运行时观察使用相同的资源。因此，如果对符号使用数据捕捉资源，则将无法对该符号使用运行时观察资源，反之亦然。

**注：** 由于捕捉时钟速度与目标系统时钟无关，因此以最大速度调试时无需使用高性能工具包。

- [数据捕捉和 DMCI](#)
- [运行时观察和 DMCI——仅限 PIC32 MCU](#)
- [运行时观察和 Watches 窗口](#)

### 6.2.1 数据捕捉和 DMCI

数据捕捉可将数据流从器件传输到以下接口：

#### 数据监视和控制界面（DMCI）——插件

要安装 DMCI 插件：

1. 选择 [Tools>Plugins](#)（工具>插件）。将打开 Plugins 窗口。
2. 单击 **Available Plugins**（可用插件）选项卡。
3. 找到 DMCI 并选中其旁边的复选框。
4. 单击 **Install**（安装），按照屏幕提示进行操作。

另请参见 MPLAB X IDE 帮助文件“Additional Tasks”（其他任务）部分中的“Add Plug-in Tools”（添加插件工具）。

要设置数据捕捉：

1. 编译项目（在 **Projects** 窗口中右键单击项目名称，然后选择“Build”（编译））。必须编译项目才能查看可用符号。
2. 选择 [Window>Debugging>Watches](#)（窗口>调试>观察），打开 Watches 窗口。
3. 在窗口中右键单击，选择“New Watch”（新建观察）。在 New Watch 窗口中，选择想要观察的符号或 SFR。单击 **OK**。
4. 选择 [Tools>Embedded>DMCI>DMCI Window](#)（工具>已安装工具>DMCI>DMCI 窗口），打开 DMCI 对话框。
5. 为该数据捕捉设置 DMCI。关于详细信息，请参见 DMCI 帮助文件（[Help>Help Contents](#)（帮助>帮助内容）中的“Plug-In Tools”（插件工具）部分）。
6. 开始调试会话（[Debug>Debug Project](#)）。使用 DMCI 控件输入数据，或在 DMCI 图形窗口中查看数据。

### 6.2.2 运行时观察和 DMCI——仅限 PIC32 MCU

对于没有数据捕捉功能的器件，运行时观察可为以下界面提供数据查询：

#### 数据监视和控制界面（DMCI）——插件

要安装 DMCI 插件：

1. 选择 [Tools>Plugins](#)。将打开 Plugins 窗口。
2. 单击 **Available Plugins** 选项卡。
3. 找到 DMCI 并选中其旁边的复选框。
4. 单击 **Install**，按照屏幕提示进行操作。

另请参见 MPLAB X IDE 帮助文件“Additional Tasks”部分中的“Add Plug-in Tools”。

## 要设置运行时观察：

1. 编译项目（在 **Projects** 窗口中右键单击项目名称，然后选择“**Build**”）。必须编译项目才能查看可用符号。
2. 选择 **Window>Debugging>Watches**，打开 **Watches** 窗口。
3. 在窗口中右键单击，选择“**New Run Time Watch**”（新建运行时观察）。在 **New Run Time Watch** 窗口中，选择想要观察的符号或 SFR。单击 **OK**。
4. 选择 **Tools>Embedded>DMCI>DMCI Window**，打开 DMCI 对话框。
5. 为该运行时观察设置 DMCI。关于详细信息，请参见 DMCI 帮助文件（**Help>Help Contents** 中的“**Plug-In Tools**”部分）。
6. 开始调试会话（**Debug>Debug Project**）。使用 DMCI 控件输入数据，或在 DMCI 图形窗口中查看数据。

## 6.2.3 运行时观察和 Watches 窗口

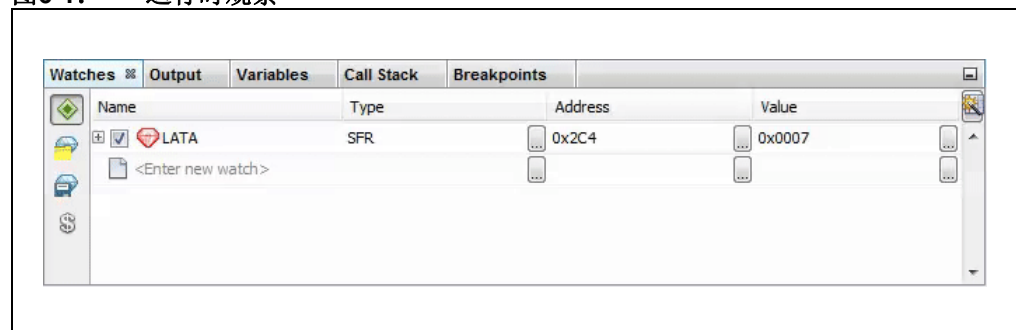
运行时观察可在程序执行（而非暂停）期间在以下窗口中提供变量更新：

- **Watches** —— **Window > Debugging** 菜单
- **Memory** —— **Window > PIC Memory Views** 菜单

## 要设置运行时观察：

1. 编译项目（在 **Projects** 窗口中右键单击项目名称，然后选择“**Build**”）。必须编译项目以查看可用的符号。
2. 选择 **Window>Debugging>Watches**，打开 **Watches** 窗口。
3. 在窗口中右键单击，选择“**New Run time Watch**”。在 **New Run Time Watch** 窗口中，选择想要观察的符号或 SFR。单击 **OK**。
4. 开始调试会话（**Debug>Debug Project**）。在 **Watches** 窗口中观察变量值的变化。
5. 要在 **Memory**（存储器）窗口中查看所观察变量的变化，可暂停并打开含有所观察变量的 **Memory** 窗口。继续运行程序并在该窗口中观察值的变化。

**图6-1：** 运行时观察



## 6.3 插装跟踪

本节将讨论可用的插装跟踪类型及其使用方法。

### 6.3.1 跟踪要求

以下是使用跟踪的要求：

支持跟踪的器件——请参见Microchip网站“开发工具”部分的“开发工具选择器（DTS）”。

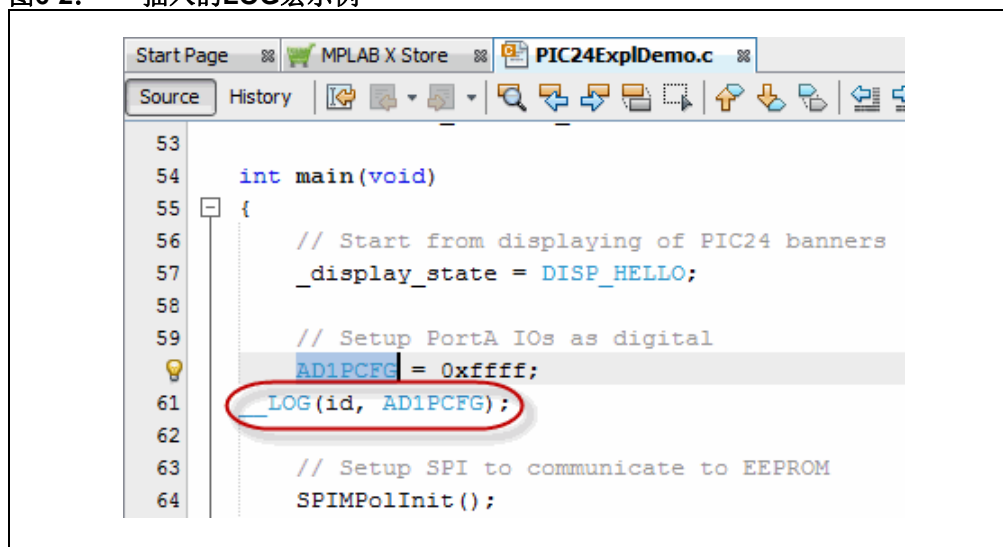
以下是跟踪的限制：

- 对于8位和16位器件，不能跟踪行内汇编代码（C代码中的汇编代码）。
- 关于其他限制，请参见MPLAB X IDE中MPLAB REAL ICE在线仿真器帮助的Limitations部分。

### 6.3.2 跟踪的工作方式

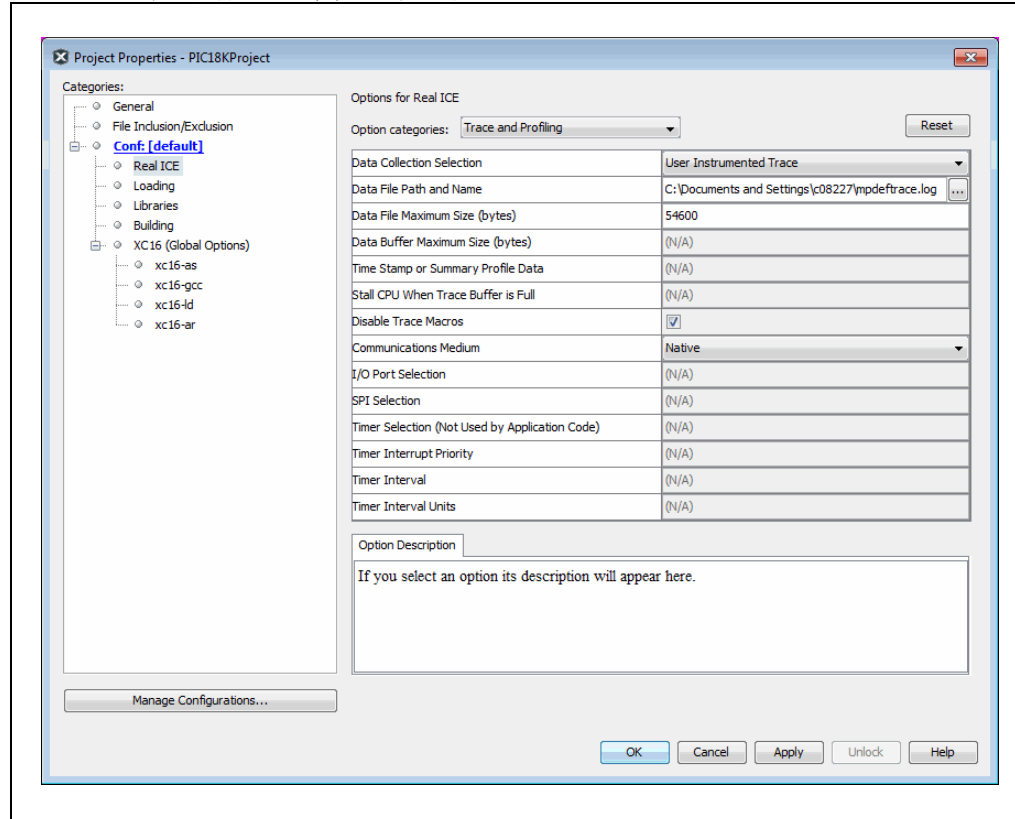
MPLAB REAL ICE在线仿真器的跟踪（插装跟踪）是一种能够提供基本跟踪信息的解决方案。利用TRACE()和LOG()宏，可在应用程序运行期间向MPLAB IDE报告程序存储单元或变量值，并在应用程序暂停后通过Trace（跟踪）窗口对其进行检查。可以手动键入这些宏名称，也可以在编辑器中右键单击并从上下文菜单中选择要插入的宏。要记录变量值，应先选中该变量，然后再从上下文菜单中选择。

图6-2: 插入的LOG宏示例



当前有三种可用的跟踪方法（见第6.3.3节“跟踪类型”）。可在Project Properties对话框中“REAL ICE”类别下的“Trace and Profiling”选项类别页面中找到这些方法。“Communications Medium”下的选项包括“Native Trace”（本机跟踪）（利用PGC/PGD通信线）、“SPI Trace”（SPI跟踪）和“I/O Port Trace”（I/O端口跟踪）。并非每种方法在每款器件上都可用，即这些选项是器件特定的。Instrumented Trace（插装跟踪）库在PIC18F MCU器件上支持C语言和汇编语言项目，但在16位器件上仅支持C语言项目。

图6-3: 项目属性——本机跟踪选择



无论使用哪种跟踪方法，传输的跟踪和记录信息都相同。对于 `TRACE()`，将发送 64-127 范围内的单个值。使用该编号生成的标签将自动插入到代码中，这样 MPLAB IDE 便可在跟踪缓冲区中识别发送该值的单元。对于 `LOG()`，先发送一个双字节标头，然后紧接正在记录的变量值。标头的第一个字节表示变量类型，是 0 至 63 之间的某个值。第二个字节则表示发送该变量的单元。此处的单元由 0 至 127 之间的某个值表示。（请参见第 6.3.9 节“跟踪/记录 ID 编号的更多信息”。）

每次 `TRACE()` 和 `LOG()` 调用期间，都会禁止中断。这是为了确保位于中断层的跟踪或记录语句不会干扰已在应用程序层执行的跟踪或记录语句。一个类似的参数可用于保护中断优先级较低的语句免受中断优先级较高的语句破坏。

## 6.3.3 跟踪类型

目前有三种跟踪类型。所有类型均取决于语言工具版本，并且会将数据流实时发送至 MPLAB IDE。

下面汇总了使用每种跟踪类型的利弊以及可用的通信类型（标准和 / 或高速）。

跟踪类型	速度	代码长度影响	实时运行	引脚使用情况	所需器件功能	通信	
						标准	高速
本机跟踪	快 <sup>(1)</sup>	大	近	无	内置调试	15 MIPS 或更低	高于 15 MIPS
SPI 跟踪	较快 <sup>(1)</sup>	中	较近	SPI 引脚	SPI	否	是
I/O 端口跟踪	最快	小	最近	8 引脚端口	无	是 <sup>(2)</sup>	是 <sup>(2)</sup>

**注 1:** 对于运行速度高于 15 MIPS 的本机跟踪以及 SPI 跟踪功能，可能需要高性能工具包。实际截止速度可能随布线、噪声和类似因素的不同而有所不同。

**2:** 还需将器件端口连接至仿真器逻辑探头端口。

### 6.3.3.1 本机跟踪

本机跟踪可用于标准或高速通信，无需额外连接——信息通过 PGD/PGC/EMUC/EMUD 引脚传递。该双线接口使用跟踪宏格式（请参见第 6.3.4 节“在 MPLAB X IDE 中设置跟踪”）。

对于 8 位和 16 位器件，本机跟踪需要输入时钟速度（Project Properties 对话框中 REAL ICE 类别下的 Clock（时钟）选项类别）。

如果使用本机跟踪，则无法使用数据捕捉，因为这两种功能使用相同的器件资源。断点仍然可以使用，但要注意的是，本机跟踪将使用一个断点进行跟踪。这不会反映在 Device Debug Resource（器件调试资源）工具栏上。

要使用数据捕捉触发，必须禁止本机跟踪（请参见第 6.3.7 节“禁止跟踪”）。

### 6.3.3.2 SPI 跟踪

SPI 跟踪只能与作为高性能工具包（AC244002）购买的高速 /LVDS 通信硬件配合使用。跟踪时钟和数据通过 LVDS 连接的接收板上的引脚 7（DAT）和 8（CLK）提供。要使用该功能，器件不一定要以高速运行。

当您将这些引脚专门用于跟踪时，应用将无法使用这些引脚上的任何复用功能。

对于具有可重映射外设引脚的器件，请注意 SPI 跟踪宏不会访问任何 PPS 寄存器，也无需知道外设如何映射到某个特定引脚——它将写入在 MPLAB X IDE 中选择的 SPI1 或 SPI2。

SPI 跟踪会要求您输入时钟速度（在 Properties 对话框中，REAL ICE 类别下的 Clock 选项类别）

关于硬件连接，请参见《MPLAB® REAL ICE™ 在线仿真器高性能工具包用户指南》（DS50002528A\_CN）。

SPI 接口使用跟踪宏格式（见第 6.3.4 节“在 MPLAB X IDE 中设置跟踪”）。



### 6.3.3.3 I/O端口跟踪

I/O端口跟踪可用于标准或高速通信。器件的8引脚I/O端口通过MPLAB REAL ICE在线仿真器逻辑探头连接器提供跟踪时钟和数据。

I/O端口的8个引脚必须均用于跟踪。该端口不可与当前使用的PGC和PGD引脚复用。因此，需查看所选器件的数据手册以确定未初始化/默认端口引脚状态并根据需要进行更改。

关于硬件连接的信息，请参见第3.6.3节“[I/O端口跟踪连接（逻辑端口）](#)”。

该端口接口使用跟踪宏格式（请参见第6.3.4节“[在MPLAB X IDE中设置跟踪](#)”）。

### 6.3.4 在MPLAB X IDE中设置跟踪

要设置MPLAB X IDE以使用MPLAB REAL ICE在线仿真器的跟踪功能：

1. 右键单击项目名称并选择“Properties”。在Project Properties对话框中，单击“Categories”下的“REAL ICE”。
2. 在“Option Categories”下，选择“Clock”。对于数据捕捉和跟踪，仿真器需要了解指令周期速度。
3. 在“Option Categories”下，选择“Trace and Profiling”。
4. 在“Data Collection Selection”下，选择“User Instrumented Trace”。
5. 在“Communications Medium”下，选择“Native”、“I/O PORT”或“SPI”跟踪之一。
6. 设置其他跟踪相关选项。（请参见第11.3.5节“[跟踪和性能分析](#)”。）
7. 单击OK。

接下来，在应用程序代码中输入跟踪宏。

- 为了记录PC的位置，请单击或选中某一代码行，然后右键单击，从弹出菜单中选择“Insert Language Line Trace”（插入语言行跟踪），其中Language（语言）可以是C或ASM。这将使以下含有宏的行插入到所选行之上：

```
__TRACE(id);
```

其中id是在编译过程中自动生成的行跟踪编号。更多信息，请参见第6.3.9节“[跟踪/记录ID编号的更多信息](#)”。

**注：** 向代码插入宏可能会修改程序的逻辑流程。检查必要的位置上是否已加大括号。

- 执行变量值记录的方法大致相同。首先选中变量名或表达式。然后右键单击并从弹出菜单中选择“Log Selected Language Value”（记录所选语言值），其中Language可以是C或ASM。这将使以下含有宏的行插入到包含该变量的行之上：

```
__LOG(id,selected variable);
```

其中id是在编译过程中自动生成的记录编号，selected variable是被选中的变量。更多信息，请参见第6.3.9节“[跟踪/记录ID编号的更多信息](#)”。

- 要删除跟踪点，只需选中并删除跟踪/记录宏。

## 6.3.5 运行跟踪

1. 调试运行 (*Debug>Debug Project*) 应用程序。
2. 暂停应用程序。
3. 在 Trace 窗口 (*Window>Debugging>Trace* (窗口>调试>跟踪)) 中查看跟踪数据。对于每个 `__TRACE` 宏，每次传递该宏时，宏后面的代码行都会出现在跟踪窗口中。对于每个 `__LOG` 宏，每次传递该宏时，宏后面代码行中的所选变量都会出现在跟踪窗口中。

**注：** 要跟踪多个代码行或变量，必须在想要跟踪的每一行/每个变量前面都放置一个宏。

每更改一个跟踪点就重复一次上述步骤。

## 6.3.6 跟踪技巧

在代码中使用 `__TRACE` 和 `__LOG` 宏时，需考虑以下因素：

- 关注应用程序的某个区域并放置 `__TRACE` 和 `__LOG` 宏，以使其在 Trace 窗口中形成一个“流程”。这样，便可按照执行流程进行操作并根据丢失/错误的跟踪点或跟踪流程的意外结束来调试应用程序。
- 在代码中将条件语句与 `__TRACE` 和 `__LOG` 宏一起使用将有助于调试。示例：当变量达到某一值时，开始对其进行记录。

```
If(var > 5)
{
    __LOG(ID, var)
}
```

- 如果允许，在代码中保留 `__TRACE` 和 `__LOG` 宏，以供将来调试使用。（在 **Project Properties** 对话框、**REAL ICE** 类别下的 **Trace Options Category** 页面中，选择 **Disable Trace Macros** (禁止跟踪宏)。）

## 6.3.7 禁止跟踪

要暂时关闭跟踪数据收集：

1. 选择 *File>Project Properties* 对话框，Categories: REAL ICE, Options categories: Trace and Profiling。
2. 选中 **Disable Trace Macros**。
3. 单击 **OK**。

要禁止全部跟踪功能：

1. 删除代码的所有跟踪和记录宏。
2. 选择 *File>Project Properties* 对话框，Categories: REAL ICE, Options categories: Trace and Profiling。
3. 在 **Data Collection Selection** 下，选择 **Off** (关)。
4. 单击 **OK**。

## 6.3.8 资源使用示例

以下示例仅供参考。其结果会根据编译器/汇编器版本、命令行选项、MPLAB IDE 版本、记录的数据变量的大小、中断状态以及使用的器件的不同而有所不同。所有示例均包含参数设置、函数调用以及用周期数表示的返回时间。

PIC18FXXJ MCU 示例是在使用无优先级处理的中断的情况下编译/汇编的（30 条指令）。使用有优先级处理的中断时，该值为 57；不使用中断时，该值为 15。

dsPIC33F DSC 示例显示了以 16 位库大小为 memcpy() 指定的 9 条指令。

### 例 6-1: 以 4 MHz (1 MIPS) 运行的 PIC18FXXJ 器件, 汇编语言项目

	本机	SPI	I/O 端口
库大小 (指令数)	23 + 30	37 + 30	25 + 30
使用的 GPR (字节数)	8	6	6
__TRACE(id) 指令周期数	80	54	42
__LOG(id, BYTE) 指令周期数	168	90	57

### 例 6-2: 以 40 MHz (10 MIPS) 运行的 PIC18FXXJ 器件, C 语言项目

	本机	SPI	I/O 端口
库大小 (指令数)	75 + 30	87 + 30	112 + 30
使用的 GPR (字节数)	10	8	8
__TRACE(id) 指令周期数	79	71	55
__LOG(id, INT) 指令周期数	225	169	162

### 例 6-3: 以 10 MIPS 运行的 dsPIC33F 器件, C 语言项目

	本机	SPI	I/O 端口
库大小 (指令数)	87 + 9	92 + 9	93 + 9
使用的 GPR (字节数)	18	14	0
__TRACE(id) 指令周期数	80	53	32
__LOG(id, INT) 指令周期数	212	124	106

### 例 6-4: 以 16 MIPS 运行的 dsPIC33F 器件, C 语言项目

	本机	SPI	I/O 端口
__TRACE(id) 指令周期数	88	53	32
__LOG(id, INT) 指令周期数	227	138	106

### 例 6-5: 以 34 MIPS 运行的 dsPIC33F 器件, C 语言项目

	本机	SPI	I/O 端口
__TRACE(id) 指令周期数	100	53	32
__LOG(id, INT) 指令周期数	251	152	106

## 6.3.9 跟踪/记录ID编号的更多信息

MPLAB IDE 将自动生成跟踪或记录宏所需的ID编号。但是，要想了解编号方法，请继续往下看。

您可以使用64个跟踪点和64个记录点。这些限制由端口跟踪（8位）决定。Bit 7用作时钟，Bit 6用作标志以指示跟踪条目（1）或记录条目（0）。

对于跟踪条目，低6位表示跟踪编号（nnnnnn）。这时，0-63即为合法跟踪编号，并且需要将跟踪标志位置1；但更简单的方法是将标志位与编号相结合，这时有效编号为64-127。

时钟	1	n	n	n	n	n	n
bit 7	bit 6						bit 0

对于记录条目，低6位表示记录编号（nnnnnnn）。

时钟	0	n	n	n	n	n	n
bit 7	bit 6						bit 0

## 6.3.10 快速跟踪参考

如果您刚开始使用MPLAB REAL ICE在线仿真器跟踪功能，建议您通读整个跟踪部分，以全面了解这一功能。

使用本节作为跟踪的快速参考。

1. 选择File>Project Properties对话框，Categories: REAL ICE，Options categories: Trace。使能跟踪、选择跟踪类型并设置其他跟踪选项。
2. 选择 Window>Debugging>Trace 打开跟踪窗口，然后在跟踪窗口中查看跟踪数据。
3. 在代码中右键单击，然后根据需要输入跟踪宏（\_\_TRACE和\_\_LOG）。
4. 调试运行项目。

## 6.4 PIC32指令跟踪——仅限PIC32 MCU

本节将讨论32位器件的跟踪功能及其使用方法。并非所有PIC32器件都具有指令跟踪功能，请参见相关器件的数据手册。

- [跟踪要求](#)
- [指令跟踪的工作方式](#)
- [设置并使用跟踪](#)
- [跟踪硬件规范](#)

### 6.4.1 跟踪要求

下面是使用32位（PIC32）器件的跟踪的要求：

- MPLAB X IDE 和兼容的32位C编译器（如MPLAB XC32 C编译器）
- PIC32接插模块（PIM），其中包含支持跟踪的器件和跟踪端口
- MPLAB® REAL ICE™ 跟踪接口工具包（AC244006），其中包含12英寸的跟踪电缆和跟踪适配器板

### 6.4.2 指令跟踪的工作方式

PIC32指令跟踪采用MIPS32 iFlowtrace™ 机制，即非侵入式硬件指令跟踪。可以使用该跟踪来捕捉器件执行的每条指令。跟踪数据通过引脚TRCLK和TRD3:0从器件发送至仿真器。仿真器将该数据传送至PC中的跟踪缓冲区，该缓冲区的行为方式类似于循环FIFO。

跟踪数据量仅受跟踪缓冲区大小的限制。即使设置为最大大小，该缓冲区也会被迅速填满，因此明智的做法是精确确定需要捕捉的内容。

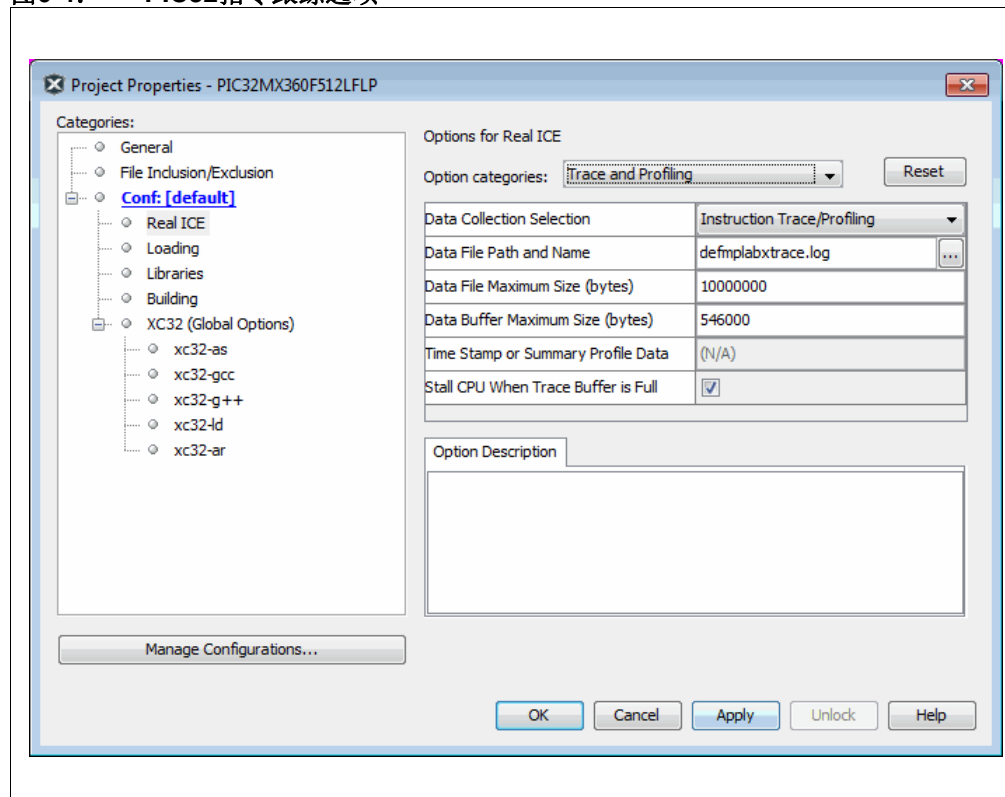
在Project Properties对话框中使能并设置跟踪选项，Categories: REAL ICE, Options categories: Trace and Profiling。可在此处设置：

- Data selection（数据选择）——使能/禁止跟踪并选择跟踪类型。
- Data file path and name（数据文件的路径和名称）——跟踪文件的位置
- Data file maximum size（数据文件的最大大小）——跟踪文件的大小
- Data Buffer maximum size（数据缓冲区的最大大小）——跟踪缓冲区的大小

片外（PC）跟踪缓冲区的最大大小为22 MB。该外部缓冲区填满时，执行过程不会暂停。

另请参见第11.3.5节“跟踪和性能分析”。

图6-4: PIC32指令跟踪选项



## 6.4.3 设置并使用跟踪

使用跟踪前，首先要对PIC32指令跟踪进行硬件和软件设置。

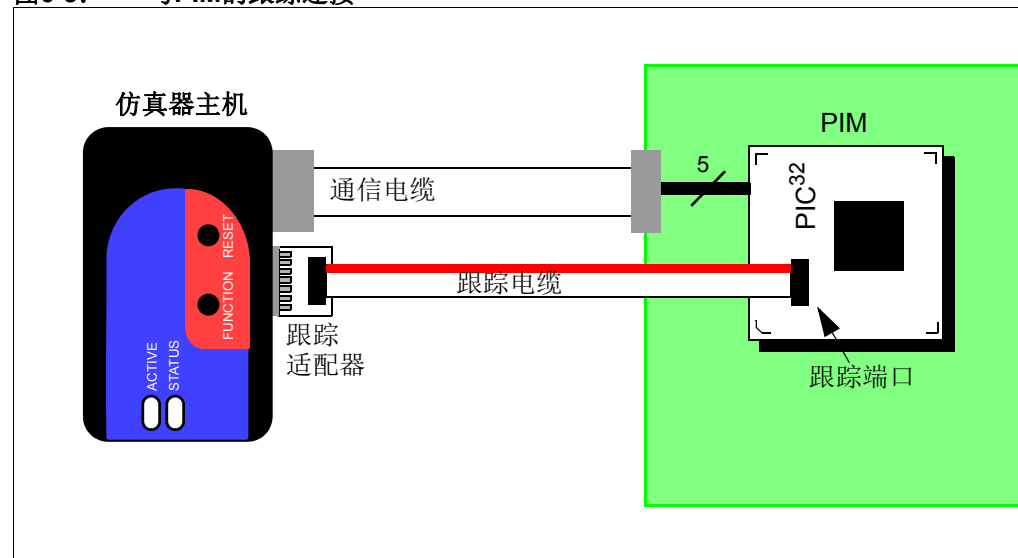
### 6.4.3.1 硬件设置——PIM上的PIC32 MCU

要将PIC32指令跟踪功能与PIM结合使用，请执行以下操作：

1. 将PIM插入未上电的目标板。
2. 在仿真器与目标板之间安装通信电缆。请参见第3.5节“仿真器与目标板的通信”或《MPLAB® REAL ICE™在线仿真器高性能工具包用户指南》(DS50002528A\_CN)。
3. 将跟踪电缆从PIM上的跟踪端口连接至跟踪适配器板。电缆连接方向如图5所示。关于硬件的详细信息，请参见《MPLAB® REAL ICE™跟踪接口工具包规范》(DS50002531A\_CN)。
4. 将跟踪适配器板插入MPLAB REAL ICE在线仿真器的逻辑探头端口。适配器板的顶部包含连接器，将该适配器板插入逻辑探头端口时应朝上(图6-5)。
5. 为目标板上电。

**注：** 使用跟踪时，会使用引脚TRCLK和TRD3:0。因此，不能在这些引脚上复用其他功能。对于PIC32MX360F512L，复用的功能是RG14:12和RA7:6。详细信息请参见器件数据手册。

图6-5: 与PIM的跟踪连接



## 6.4.3.2 硬件设置——目标板上的PIC32 MCU

在自己的目标板上设计指令跟踪功能时，需要做以下准备。

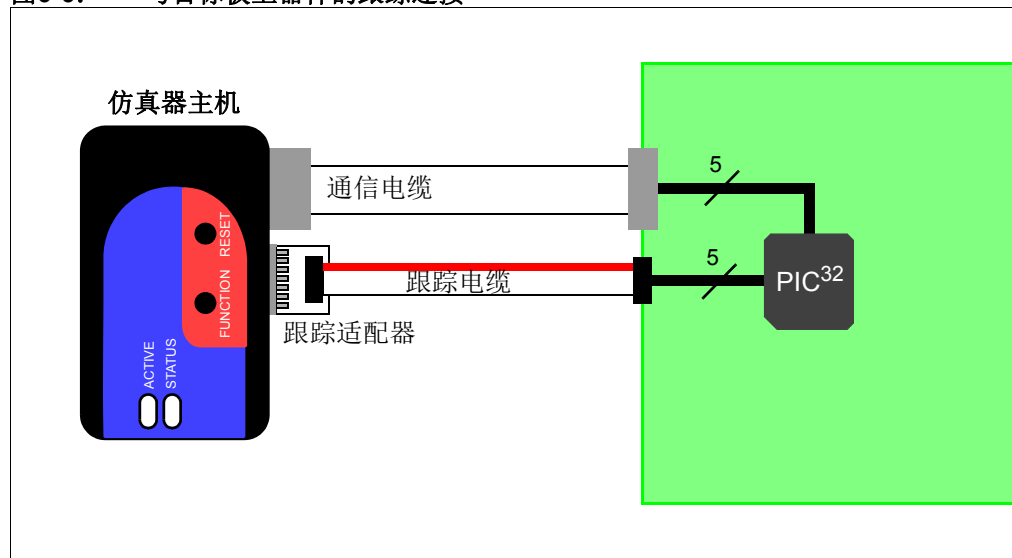
- 需要添加端接串联电阻。关于详细信息，请参见《MPLAB<sup>®</sup> REAL ICE™跟踪接口工具包规范》(DS50002531A\_CN)。
- 根据电路板布线以及用于跟踪的信号负载情况，最好放置0欧电阻，这样拆除这些电阻即可隔离跟踪信号TRCLK和TRD3:0。

要将PIC32指令跟踪功能与自己的目标板结合使用，请执行以下操作：

1. 目标板最初应处于未上电状态。
2. 在仿真器与目标板之间安装通信电缆。请参见第3.5节“仿真器与目标板的通信”或《MPLAB<sup>®</sup> REAL ICE™在线仿真器高性能工具包用户指南》(DS50002528A\_CN)。
3. 将跟踪电缆从目标板连接至跟踪适配器板。确保目标板上的PIC32 MCU的连接符合跟踪要求，如图6-6所示。关于硬件的详细信息，请参见《MPLAB<sup>®</sup> REAL ICE™跟踪接口工具包规范》(DS50002531A\_CN)。
4. 将跟踪适配器板插入MPLAB REAL ICE在线仿真器的逻辑探头端口。适配器板的顶部包含连接器，将该适配器板插入逻辑探头端口时应朝上(图6-6)。
5. 为目标板上电。

**注：** 使用跟踪时，会使用引脚TRCLK和TRD3:0。因此，不能在这些引脚上复用其他功能。对于PIC32MX360F512L，复用的功能是RG14:12和RA7:6。

图6-6: 与目标板上器件的跟踪连接





### 6.4.3.3 MPLAB X IDE 设置

要使用 MPLAB REAL ICE 在线仿真器的跟踪功能，请执行以下步骤来设置 MPLAB X IDE：

1. 右键单击项目名称，并选择“Properties”。在 Project Properties 对话框中，单击“REAL ICE”（在“Categories”（类别）下）。
2. 在“Option categories”（选项类别）下，选择“Trace and Profiling”。
3. 在“Data Collection Selection”下，选择“Instruction Trace/Profiling”（指令跟踪 / 性能分析）。
4. 设置与跟踪相关的所有其他选项。（见第 11.3.5 节“跟踪与性能分析”。）
5. 单击 **OK**（确定）。

在执行 Debug Run（调试运行）时，跟踪会不断用数据填充跟踪缓冲区，并在缓冲区满时返回缓冲区起始位置，直到程序暂停为止。

### 6.4.3.4 查看跟踪数据

当使能跟踪且代码在运行时，仿真器会收集跟踪数据。当器件暂停时，跟踪数据会被解码并显示在 Trace（跟踪）窗口中（Window（窗口） > Debugging（调试） > Trace）。

### 6.4.4 跟踪硬件规范

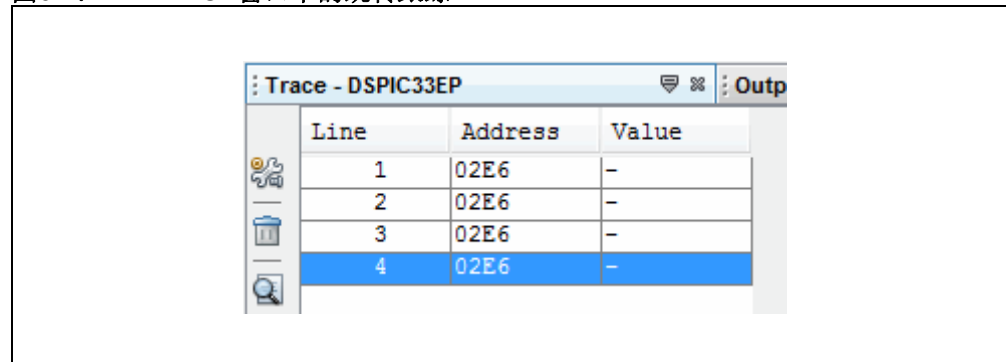
《MPLAB<sup>®</sup> REAL ICE<sup>™</sup> 跟踪接口工具包规范》（DS50002531A\_CN）和一些 PIC32 PIM 的网页上给出了支持 PIC32 指令跟踪的硬件的规范。

## 6.5 跳转跟踪——仅限EP器件

跳转跟踪是部分（EP）器件上提供的功能，用于记录代码中最多后四条指令流的变化，从而能够往回浏览调用链。跳转/转移地址显示在Trace窗口中（最多四个条目）。

在Project Properties窗口中Real ICE类别下的“Trace and Profiling”选项类别中，在“Data Collection Selection”下选择该功能。

图6-7: TRACE窗口中的跳转跟踪



## 6.6 PC采样——仅限8位和16位 MCU

PC采样是一种检查C代码的方法，用于确定每个函数所消耗时间的百分比。该信息可显示代码中消耗程序时间的位置，这样便可对代码进行优化。

对于PC采样，将设置一个器件定时器，以对程序执行进行采样，并将结果显示在PC性能分析窗口中。

PC性能分析与PC采样类似。关于详细信息，请参见第6.7节“PC性能分析——仅限32位MCU”。

### 6.6.1 要求

目前，要使用PC采样，项目必须进行如下设置：

- 支持的器件：
  - （注：不支持PIC16F1器件）
  - 具有数据捕捉功能的PIC18F——要确定器件是否具有数据捕捉功能，请查看在线开发工具选择器：<http://www.microchip.com/dts>。
  - PIC24F和PIC24EP
  - dsPIC33FJ和dsPIC33E
- MPLAB XC8或MPLAB XC16 C编译器版本1.10或更高版本

### 6.6.2 时钟设置

要设置时钟：

1. 打开Project properties窗口（*File>Project Properties*）。
2. 单击“Categories”下的“REAL ICE”，然后从“Options categories”下拉菜单中选择“Clock”。确保该值与实际目标速度匹配，即，根据代码中的配置位设置。
3. 单击**Apply**（应用）。

### 6.6.3 采样设置

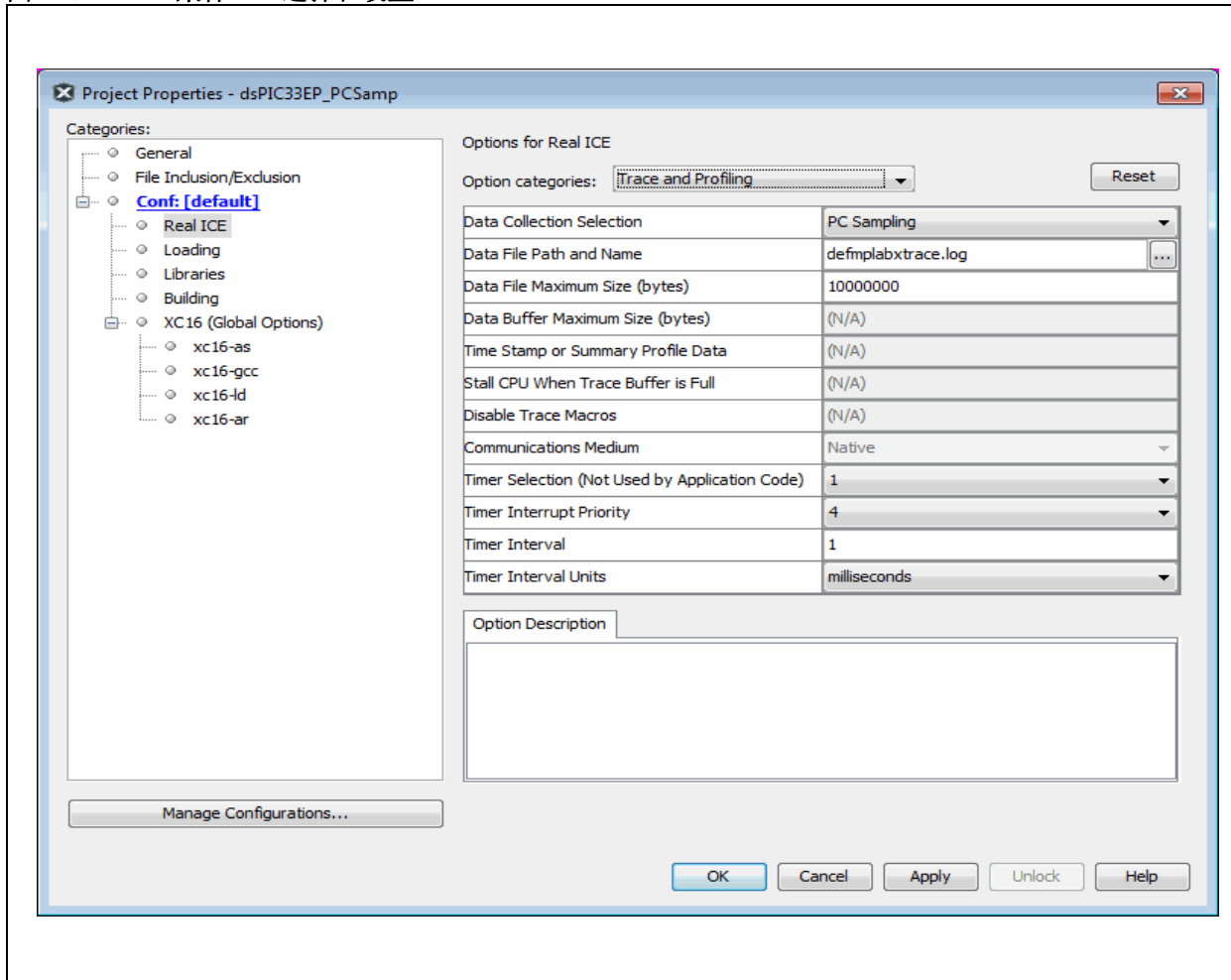
要设置采样：

1. 单击Categories下的REAL ICE，然后从Options categories下拉框中选择Trace and Profiling。
2. 在“Data Collection Selection”下，选择“PC Sampling”（PC采样）。
3. 在此窗口中设置数据文件和定时器。相关参考信息，请参见第11.3.5节“跟踪和性能分析”。

**注：** 确保您的程序中没有使用所选的定时器。

4. 单击**OK**。

图6-8: PC采样——选择和设置



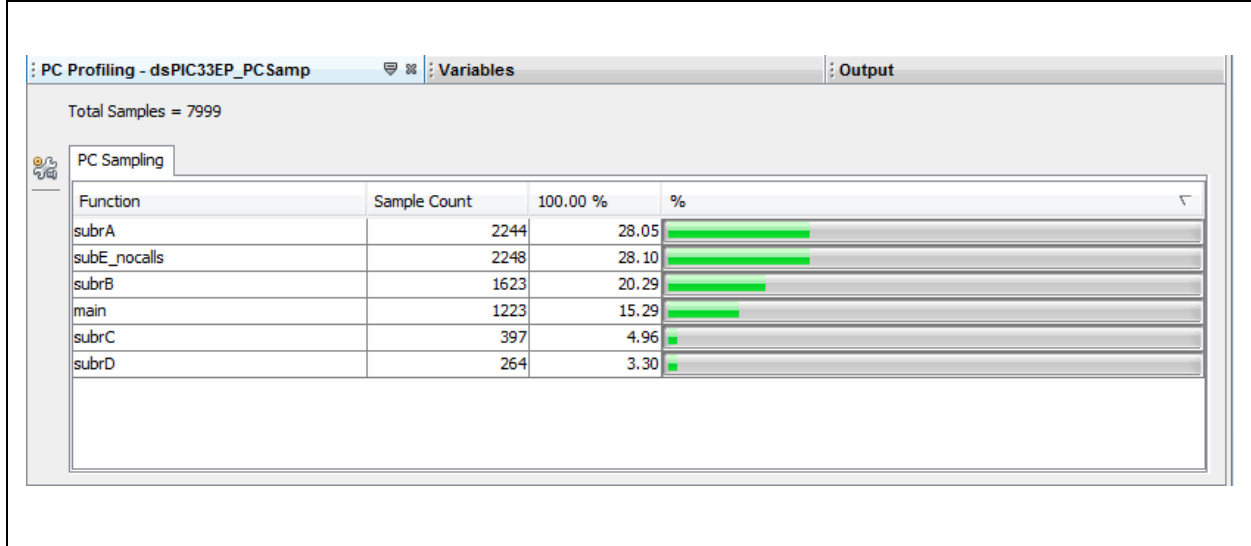
## 6.6.4 操作

要生成数据：

1. 选择 *Window>Debugging>PC Profiling*。将打开PC Profiling 窗口。
2. 运行代码然后暂停。
3. 在窗口中查看采样数据。数据只有在暂停时才会显示。
4. 在窗口中右键单击，从弹出菜单中选择清除数据或重载数据。

此外，也可以在代码性能分析插件中显示该数据，该插件可通过以下网址购买：  
<http://www.embeddedcodesource.com>

图6-9: PC采样——PC PROFILING窗口



## 6.7 PC性能分析——仅限32位MCU

PC性能分析是一种检查C代码的方法，用于确定每个函数所消耗时间的百分比。该信息可显示代码中消耗程序时间的位置，这样便可对代码进行优化。

对于PC性能分析，将从跟踪（数据）缓冲区中获取每个程序计数器（PC）的跟踪采样并对其进行性能分析。该数据显示在PC Profiling窗口中，如图6-11所示。在该示例中，跟踪缓冲区中包含21,000个PC采样数据点。其中，有7859个（或37.42%）与main()函数相关，6023个（或28.68%）与subrA()函数等相关。

PC性能分析与PC采样类似。关于详细信息，请参见第6.6节“PC采样——仅限8位和16位MCU”。

### 6.7.1 要求

目前，要使用PC性能分析，项目必须设置为：

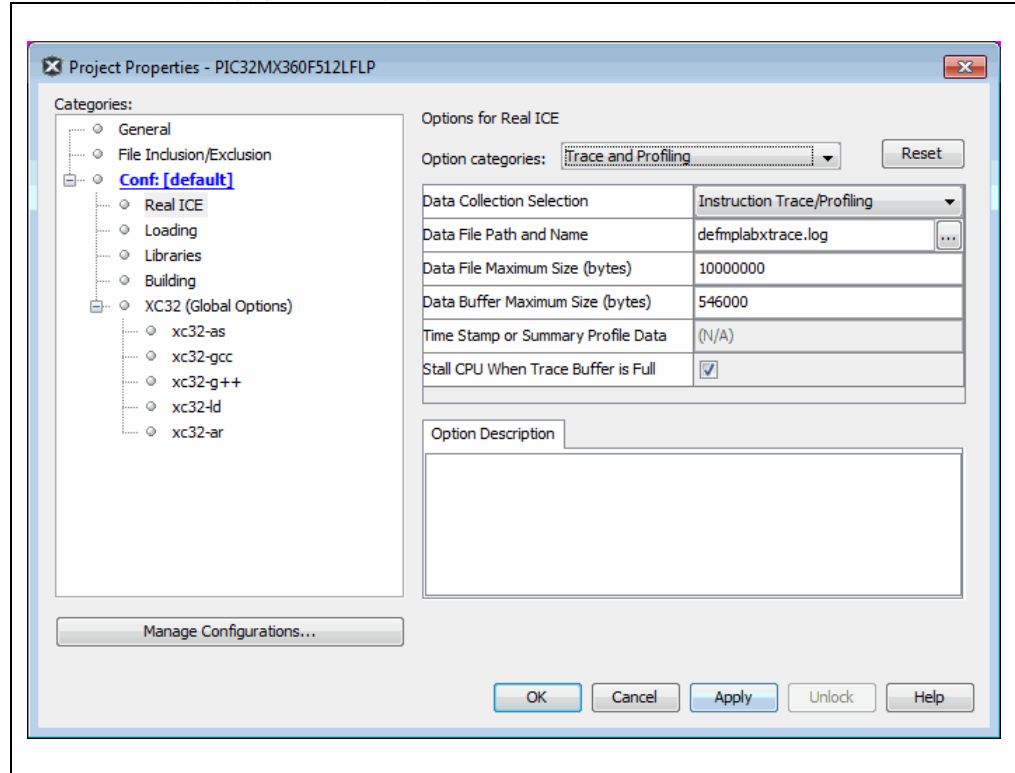
- 支持的器件：
  - 具有数据捕捉功能的PIC32MX——要确定器件是否具有数据捕捉功能，请查看在线开发工具选择器：  
<http://www.microchip.com/dts>。
  - 注：不支持PIC32MZ器件

### 6.7.2 性能分析设置

要设置性能分析：

1. 设置用于PIC32指令跟踪的硬件（请参见第6.4节“PIC32指令跟踪——仅限PIC32 MCU”）。
1. 打开Project properties窗口（*File>Project Properties*）。
2. 单击Categories下的REAL ICE，然后从Options categories下拉框中选择Trace and Profiling。
3. 在Data Collection Selection下，选择Instruction Trace/Profiling。
4. 在该窗口中设置数据文件。相关参考信息，请参见第11.3.5节“跟踪和性能分析”。  
然后单击OK。

图6-10: PC性能分析——选择和设置



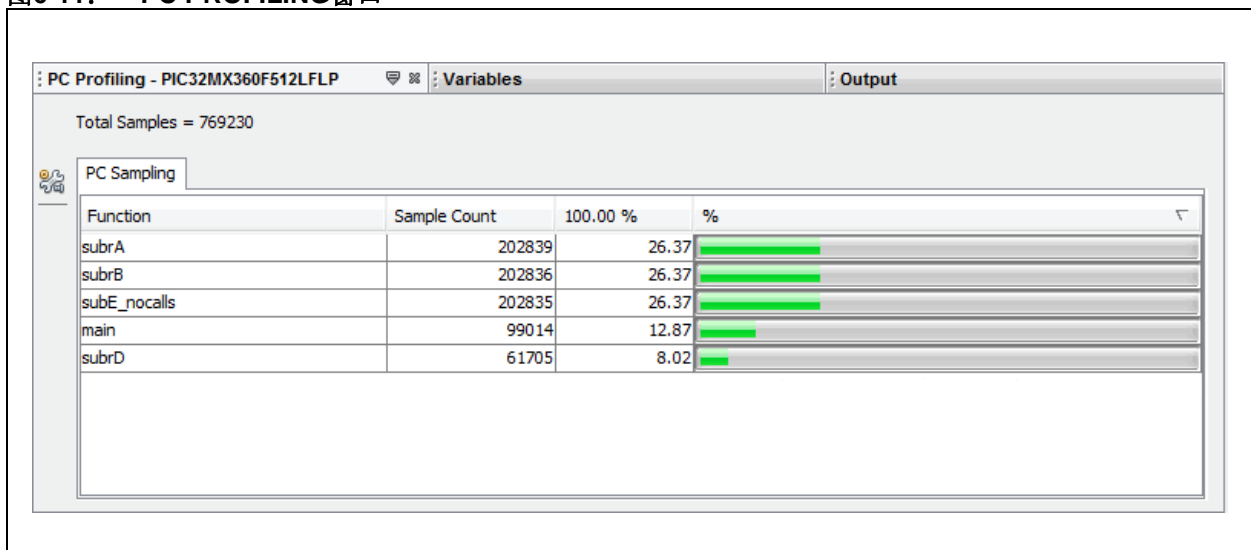
### 6.7.3 操作

要生成数据:

1. 选择 *Window>Debugging>PC Profiling*。将打开PC Profiling 窗口。
2. 运行代码然后暂停。
3. 在窗口中查看性能分析数据。数据只有在暂停时才会显示。
4. 在窗口中右键单击，从弹出菜单中选择清除数据或重载数据。

此外，也可以在代码性能分析插件中显示该数据，该插件可通过以下网址购买：  
<http://www.embeddedcodesource.com>

图6-11: PC PROFILING窗口



## 6.8 函数级别性能分析

功能等级性能分析（Function Level Profiling, FLP）是一种检查C代码的方法，用于确定每个函数所消耗时间（暂停流数据汇总）的百分比。该信息可显示代码中消耗程序时间的位置，这样便可对代码进行优化。

### 6.8.1 要求

目前，要使用FLP，项目必须设置为：

- **支持数据捕捉功能的器件：**
  - 具有数据捕捉功能的PIC18F——要确定器件是否具有数据捕捉功能，请查看在线开发工具选择器：<http://www.microchip.com/dts>。
  - PIC24F和PIC24EP
  - dsPIC33FJ和dsPIC33E
  - 具有数据捕捉功能的PIC32MX——要确定器件是否具有数据捕捉功能，请查看在线开发工具选择器：<http://www.microchip.com/dts>。
  - 注：不支持PIC16F1器件
  - 注：不支持PIC32MZ器件
- **MPLAB XC C编译器版本1.20或更高版本**
- **代码性能分析插件**（可选择购买）
  - 该插件可通过如下网址购买：<http://www.embeddedcodesource.com>
  - 按照MPLAB X IDE帮助“Additional Tasks”部分的“Add Plug-In Tools”（添加插件工具）中的说明安装插件（称为Code Profiling Viewer）。
  - 安装插件后，可以在 *Tools>Embedded>CodeProfiling*（工具>已安装工具>CodeProfiling）下访问“Code Profiling”窗口。

### 6.8.2 时钟设置（仅限8位和16位MCU）

要设置时钟：

1. 打开Project properties窗口（*File>Project Properties*）。
2. 单击“Categories”下的“REAL ICE”，然后从“Options categories”下拉菜单中选择“Clock”。确保该值与实际目标速度匹配，即，根据代码中的配置位设置。
3. 单击**Apply**。

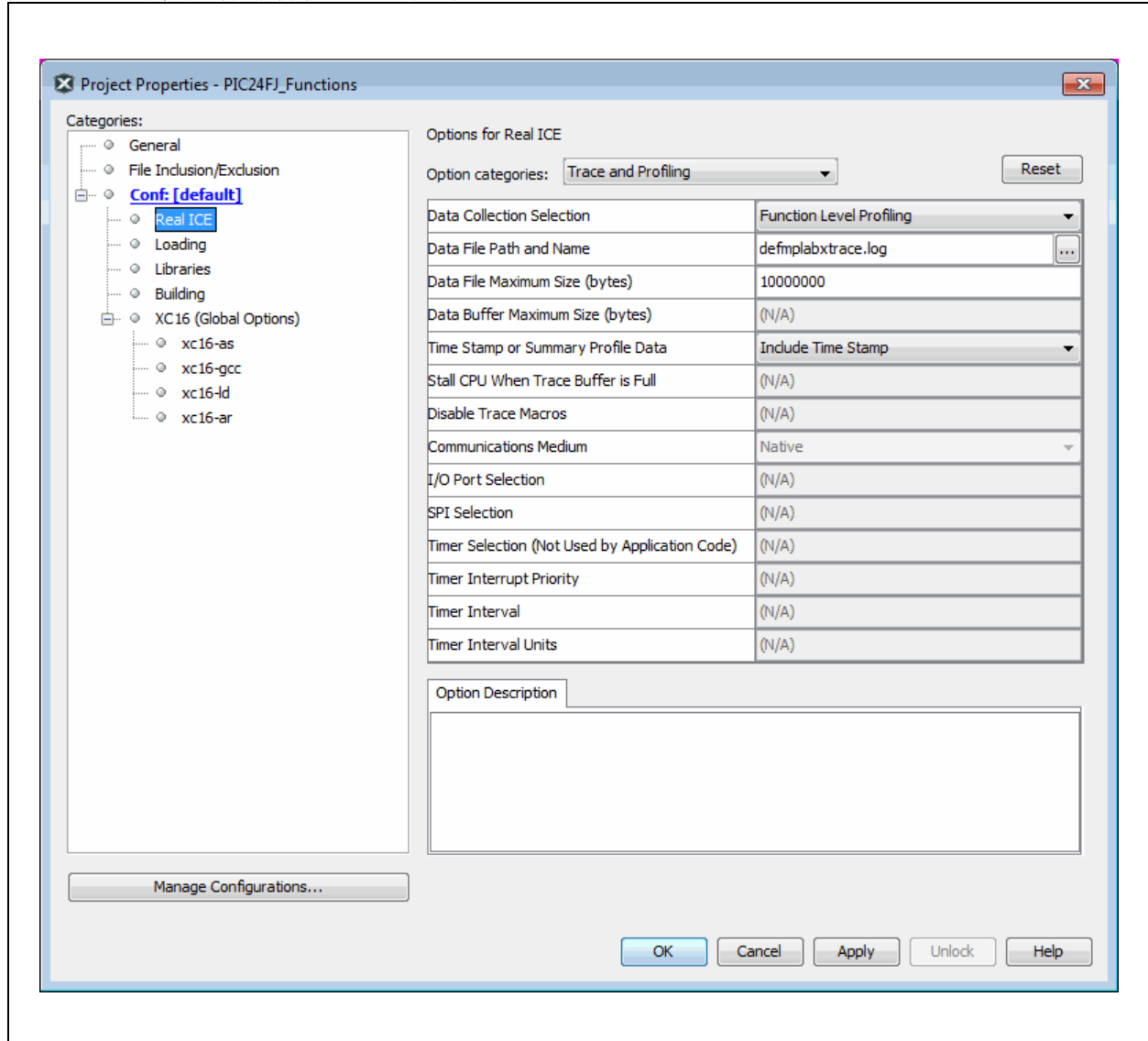
### 6.8.3 采样设置

要设置采样（图6-12）：

1. 单击Categories下的REAL ICE，然后从Options categories下拉框中选择Trace and Profiling。
2. 在“Data Collection Selection”下，选择“Function Level Profiling”（函数级别性能分析）。
3. 设置数据文件信息，即，Data File Path and Name和Data File Maximum Size。关于详细信息，请参见第11.3.5节“跟踪和性能分析”。
4. 选择函数级别性能分析的类型：
  - a) Include Time Stamp（包含时间戳）：使用的时间戳数据
  - b) Summary Profile Data Only（仅汇总性能分析数据）：使用的汇总数据
5. 单击**OK**。



图6-12: 函数级别性能分析——选择和设置



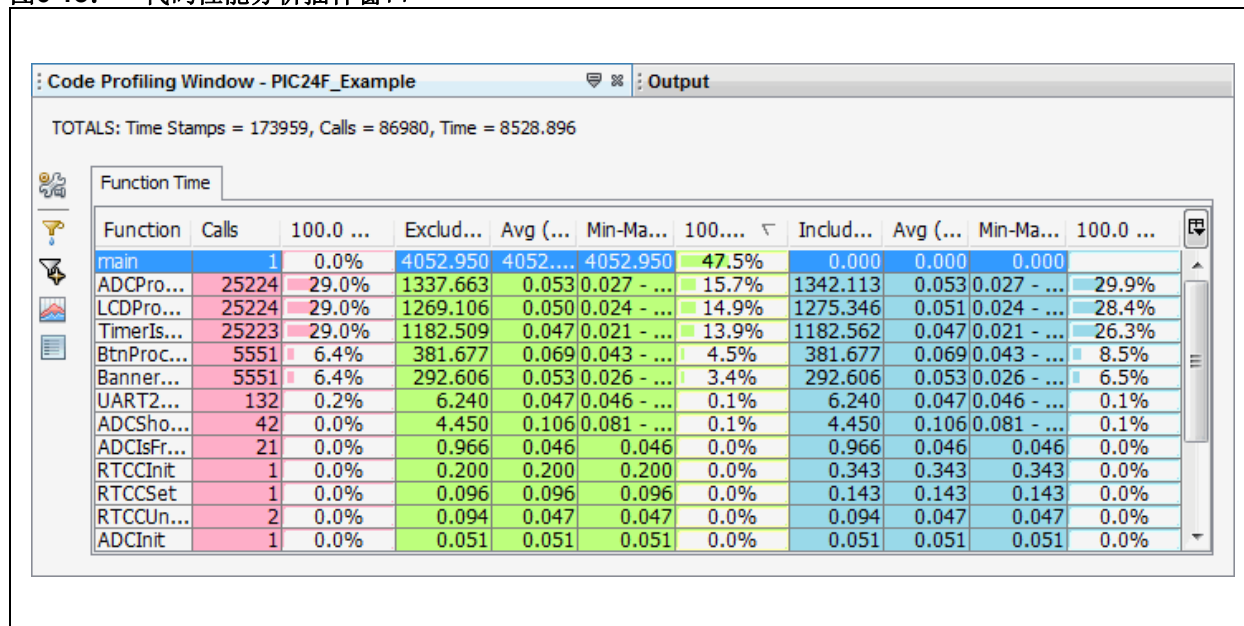
#### 6.8.4 操作

要生成数据:

1. 调试运行，然后暂停程序以将数据收集到文件中。
2. 根据是否安装了代码性能分析插件：
  - a) 无插件：不显示IDE数据。
  - b) 有插件：Plugin 窗口中将显示时间戳（图 6-13）或汇总数据。还提供有图表。关于详细信息，请参见插件帮助文件。

# 适用于 MPLAB X IDE 的仿真器用户指南

图6-13: 代码性能分析插件窗口



## 6.9 应用程序输入/输出

**注：** 仅在支持应用程序输入/输出功能的器件与MPLAB ICD 3或MPLAB REAL ICE在线仿真器结合使用时，本窗口才可用。关于详细信息，请参见在线帮助中的“Device and Feature Support”。  
应用程序输入/输出功能不能与PIC32指令跟踪功能结合使用。

Application In/Out（应用程序输入/输出）窗口允许您使用某些器件上支持的应用程序输入/输出（应用程序I/O）功能与运行中的应用程序进行交互。利用该功能，可在运行期间通过用于调试的相同PGC/PGD线以串行方式向/从目标应用程序发送数据。写入APPOUT寄存器的数据将在窗口中显示，同时用户输入将被发送到目标应用程序并保存在APPIN寄存器中。

### 6.9.1 应用程序I/O与8位和16位MCU结合使用

要使用Application In/Out窗口：

- 要直接对应用程序I/O进行写操作，构建应用程序时使用器件特定的头文件（按照例6-6）。
- 将来，16位器件库将包含printf()和scanf()函数，以便与应用程序输入/输出功能结合使用。

#### 例6-6： 应用程序输入/输出寄存器的使用

以下应用程序代码将读取应用程序输入寄存器，然后将输入的填充值写入应用程序输出寄存器。

```
// include file
#include <p33Exxxx.h>

// set up config bits
_FWDT( FWDTEN_OFF )

// initialize variables
unsigned int val;
int i;
unsigned int oval;

int main(void)
{
    while(1)
    {
        if(APPSbits.APIFUL) // APPI is full?
        {
            val = _APPIN; // Read User Input
            for(i=0; i<4; i++)
            {
                while(APPSbits.AROFUL); // APPO is full?
                oval = val&0xFF;
                if(oval < 0x20)
                    oval = 0x20;
                oval |= 0x20202000;
                _APPO = oval; // Send to MPLAB X IDE
                val >>= 8;
            }
        }
    }
}
```

## 6.9.2 应用程序I/O与32位MCU结合使用

要使用Application In/Out窗口：

- 在构建应用程序时使用器件特定的头文件，以使用头文件中分配的宏（根据例6-6）。
- 或者，PIC32 MCU库包含printf()和scanf()函数，以便与应用程序输入/输出功能结合使用。请参见*32-Bit Language Tools Libraries* (DS51685)中的“PIC32 Debug-Support Library”一章。

### 例6-7： 应用程序输入/输出宏的用法

以下应用程序代码将读取应用程序输入寄存器，然后将输入的填充值写入应用程序输出寄存器。

```
// include file
#include <p32xxxx.h>

// set up config bits
#pragma config FWDTEN = OFF

// initialize variables
unsigned int val;
int i;
unsigned int oval;

int main(void)
{
    while(1)
    {
        if(_DDPSTATbits.APIFUL) // APPI is full?
        {
            val = _APPI; // Read User Input
            for(i=0; i<4; i++)
            {
                while(_DDPSTATbits.APOFUL); // APPO is full?
                oval = val&0xFF;
                if(oval < 0x20)
                    oval = 0x20;
                oval |= 0x20202000;
                _APPO = oval; // Send to MPLAB X IDE
                val >>= 8;
            }
        }
    }
}
```

## 6.9.3 运行应用程序

要运行示例应用程序：

1. 使用Project Wizard（项目向导）创建项目：
  - a) 选择支持的器件。
  - b) 选择“REAL ICE”或“ICD 3”作为硬件工具。
  - c) 使用某款MPLAB XC C编译器作为语言工具套件。
2. 选择 **File>New File**（文件>新建文件）创建和命名一个新的C源文件。在打开的Editor窗口中，添加前面的代码并保存。
3. 在Projects窗口中，右键单击项目名称并选择“Properties”。在Project Properties窗口中，单击“Categories”下的“xcnn-gcc”。在“Option Categories”下，选择“General”（通用）并选中“Enable App IO”（使能应用程序IO）复选框。
4. 编译项目（右键单击项目名称，然后选择Build）。
5. 选择 **Window>Debugging>PIC App IO**（窗口>调试>PIC应用程序IO），打开Application In/Out窗口。
6. 单击“Properties”按钮（位于窗口左侧），启动Set App IO Properties（设置应用程序IO属性）对话框。在“Capture”（捕捉）下，单击“On”（开）。
7. 调试运行或运行项目。
8. 在Input（输入）文本框中输入文本值。按Enter键。
9. 在Output（输出）文本框中查看输出。

图6-14: SET APP I/O PROPERTIES对话框

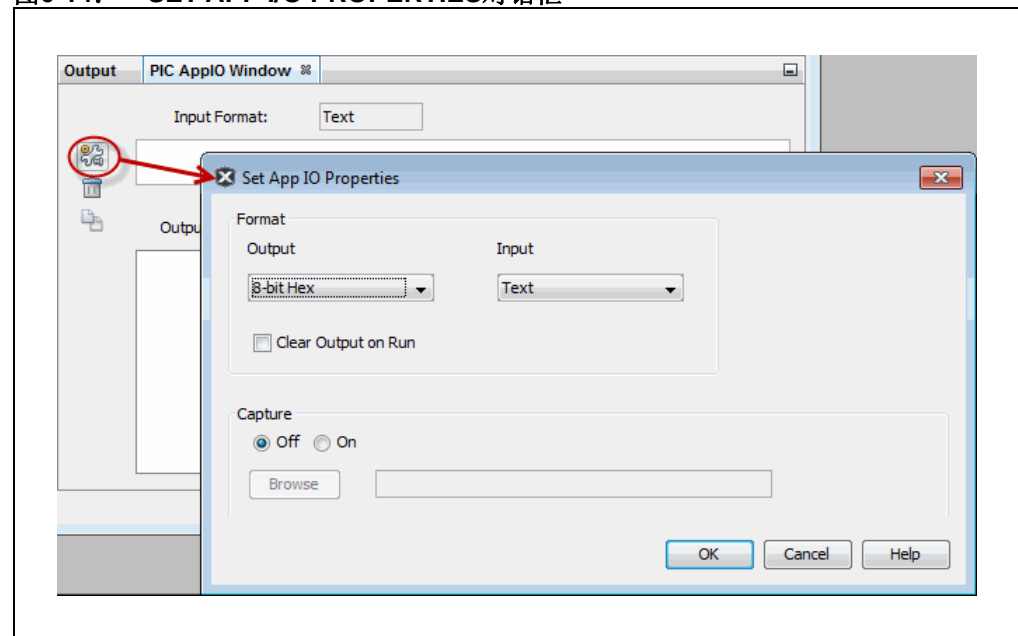
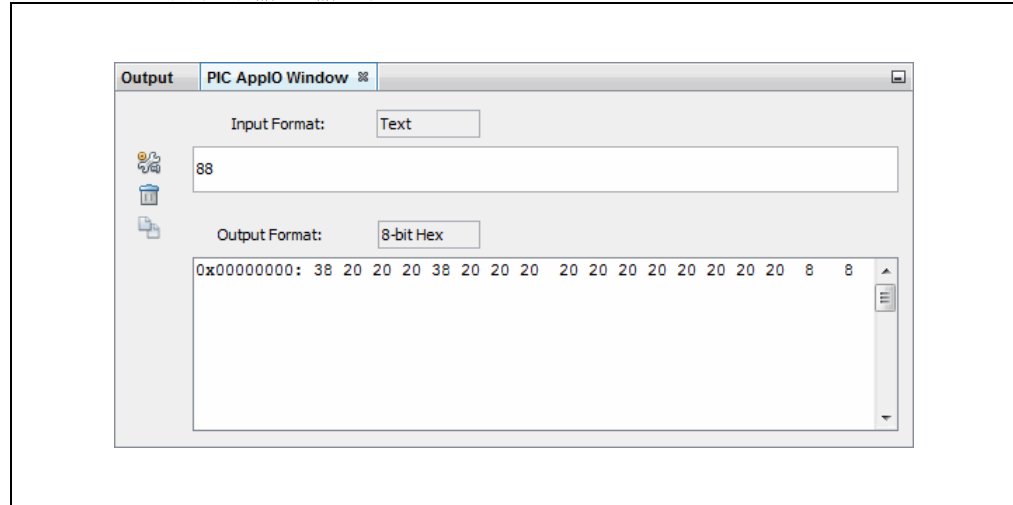


图6-15: 应用程序输入/输出窗口



打开 Set App IO Properties 对话框，更改 Input 和 Output 的格式，然后再次运行程序以查看不同输入对应的输出结果。

关于本窗口中可用选项的详细信息，请参见第 11.4.2 节“应用程序输入/输出窗口和相关对话框”。

## 6.10 外部触发信号

通过逻辑探头电缆使用外部触发信号设置硬件触发信号。可将外部触发信号用作输入以在发生外部事件时暂停程序执行。外部触发信号还可以用作输出来控制外部器件。

如果将仿真器逻辑探头端口用于下述其他调试功能，则无法使用外部触发信号：

- I/O 端口跟踪（见第 6.3.3.3 节“I/O 端口跟踪”）
- PIC32 指令跟踪（见第 6.4 节“PIC32 指令跟踪——仅限 PIC32 MCU”）

### 6.10.1 硬件连接

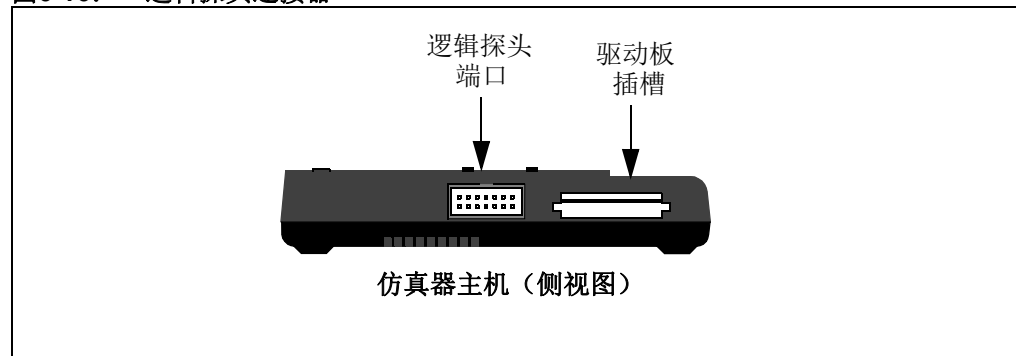
要连接逻辑探头电缆：

1. 将逻辑探头线束上的连接器插入仿真器上的逻辑探头端口（图 6-16）。如果仿真器没有随附逻辑探头，您可在 Microchip Direct 上订购（ACICE0104）。
2. 将探头连接到目标板。所有引脚（无论已使用或未使用）均应上拉或接地。保留引脚悬空可能产生错误的触发信号。

要将探头引脚用作输入，必须提供驱动电路（关于驱动电平的信息，请参见表 12-3：“逻辑探头电气规范”）。

关于外部触发硬件的更多信息，请参见第 12.4.4 节“逻辑探头/外部触发接口”。

图6-16： 逻辑探头连接器



### 6.10.2 软件设置

要选择逻辑探头端口引脚上的功能：

1. 右键单击 MPLAB X IDE 项目，然后选择“Properties”。
2. 单击“Categories”下的“Real ICE”，然后选择“External Triggers”（外部触发信号）选项类别。
3. 使用 Trigger（触发信号）旁的下拉菜单来设置该功能。另请参见第 11.3.8 节“外部触发信号”。

## 6.11 额外调试功能

关于调试功能、窗口和对话框的完整列表，请参见第 11 章 “仿真器功能汇总”。





---

---

## 第3部分——故障诊断

---

---

第7章 故障诊断首要步骤 .....	83
第8章 常见问题解答 (FAQ) .....	85
第9章 消息 .....	95
第10章 工程技术说明 (ETN) .....	99

注:

---

---

## 第7章 故障诊断首要步骤

---

---

### 7.1 简介

当MPLAB REAL ICE在线仿真器的运行发生问题时，请从这里开始。

- [要首先回答的5个问题](#)
- [无法调试的首要原因](#)
- [要考虑的其他事项](#)

### 7.2 要首先回答的5个问题

1. 您使用的是哪款器件？要支持新器件，通常要求将MPLAB X IDE升级到新版本。
2. 您使用的是 Microchip 演示板还是您自己的设计之一？您是否遵循了用于通信连接的电阻/电容的准则？请参见第3章“工作原理”。
3. 您是否已为目标板供电？不能由仿真器为目标板供电。
4. 您是否在设置中使用了USB集线器？是否已为其供电？如果仍然存在问题，请尝试不使用集线器而是将仿真器直接插入PC。
5. 您使用的电缆是否为仿真器随附的标准通信电缆（RJ-11）？如果您制作了较长的电缆，可能会出现通信错误。如果需要使用较长的电缆，应考虑使用高速通信。请参见第3.5.2节“高速/LVDS 通信（高性能工具包）”。

### 7.3 无法调试的首要原因

1. **振荡器不工作。**检查与振荡器有关的配置位设置。如果使用的是外部振荡器，请尝试使用内部振荡器。如果使用的是内部PLL，请确保PLL的设置正确。
2. **未对目标板供电。**检查电源线连接。
3. **错误的VDD电压。**VDD电压超出该器件的规范。更多详细信息，请参见器件编程规范。
4. **物理连接断开。**仿真器与PC和/或目标板物理连接断开。检查通信电缆的连接。
5. **通信丢失。**仿真器与PC之间的通信中断。在MPLAB IDE中重新连接仿真器。
6. **未固定器件。**没有将器件正确固定在目标板上。如果已正确连接仿真器并已成为目标板供电，但器件缺失或未完全插入，则可能得到如下消息：  
**Target Device ID (0x0) does not match expected Device ID (0x%x)**（目标器件ID（0x0）与预期器件ID（0x%x）不匹配）  
其中%x为预期器件ID。

7. **器件受到代码保护。** 检查与代码保护有关的配置位设置。
8. **没有器件调试电路。** 尝试调试没有调试功能的生产器件。请使用调试头代替。（请参见“[推荐读物——仿真器](#)”中的《处理器扩展包（PEP）和调试头规范》（DS50001292W\_CN）。）
9. **应用程序代码损坏。** 目标应用程序损坏或包含错误。尝试重新编译和重新编程目标应用程序，然后对目标器件启动上电复位。
10. **应用程序代码可运行但无法调试。** 如果您的应用程序可以调试编译并运行，但无法调试（代码左边的侧边栏中没有暂停箭头，断点失效等），说明可能无法找到源文件。请检查路径和文件名中是否存在非 ANSI 字符，或者长度是否太长（关于路径、文件和文件夹限制，请参见 MPLAB X IDE 文档）。
11. **错误的编程引脚。** 没有在配置位中编程正确的 PGC/PGD 引脚对（对于带多个 PGC/PGD 引脚对的器件）。
12. **需要其他设置。** 其他干扰调试的配置设置。任何阻止目标器件执行代码的配置设置同样会阻止仿真器将代码置于调试模式。
13. **错误的欠压电压。** 欠压检测电压大于工作电压 VDD。这意味着器件处于复位状态，不能调试。
14. **错误的连接。** 您未遵循[第 3 章“工作原理”](#)中的准则。

## 7.4 要考虑的其他事项

1. 使用环回测试板来验证仿真器本身是否正常工作（[第 12.7 节“环回测试板”](#)）。
2. 通常，编程可能会出现問題。作为测试，切换为运行模式（*Run (运行) > Run Project (运行项目)*），然后向目标器件烧写一个尽可能简单的应用程序（例如，使 LED 闪烁的程序）。如果程序没有运行，则说明目标设置存在错误。
3. 目标器件很可能因某种原因（如过流）遭到损坏。开发环境明显对元器件不利。请考虑尝试另一个目标器件。
4. 查看仿真器调试操作以确保应用的设置正确（见[第 3 章“工作原理”](#)）。
5. 如果问题仍然存在，请联系 Microchip。

---

---

## 第8章 常见问题解答 (FAQ)

---

---

### 8.1 简介

本章提供关于MPLAB REAL ICE在线仿真器系统的常见问题的解答。

- [仿真器运行](#)
- [插装跟踪运行](#)
- [一般问题](#)

### 8.2 仿真器运行

下面的问题与仿真器运行有关。

#### 8.2.1 器件中的什么组件允许其与MPLAB REAL ICE在线仿真器进行通信？

大多数芯片中都包含可与仿真器进行通信的调试模块。此外，还需要向存储器的高地址段（有时位于称为测试存储区的独立存储区中）中烧写一个小的调试执行程序。对于某些没有片上调试功能的小器件，可购买处理器扩展包或仿真扩展包来进行调试。

#### 8.2.2 运行调试执行程序对处理器的吞吐量有什么影响？

调试执行程序不会在代码运行时运行，因此不会降低吞吐量，即仿真器不会占用目标器件的任何周期。

#### 8.2.3 MPLAB REAL ICE在线仿真器与其他在线仿真器/调试器的对比情况如何？

请参见第3.2节“工具比较”。

#### 8.2.4 MPLAB X IDE如何与MPLAB REAL ICE在线仿真器进行交互，以实现比在线调试器更多的功能？

对于某些器件，MPLAB REAL ICE在线仿真器使用存储器特殊区域中的调试执行程序（不使用应用程序存储区）进行通信。此外，调试执行程序经过简化，可以实现更加高效的通信。仿真器中包含FPGA、大型SRAM缓冲区（1Mx8）和高速USB接口。程序存储器映像会被下载并包含在SRAM中，从而加快编程速度。仿真器中的FPGA可用作加速器，与器件的在线调试器模块进行交互。

## 8.2.5 MPLAB REAL ICE在线仿真器是否具有复杂断点？

是。您可基于次数计数、特定地址处程序存储器的执行或者特定地址处数据存储器的读取/写入操作来实现中断。您还可以设置事件断点（发生某个事件时中断）、顺序断点（多个事件按特定顺序发生时中断）或者逻辑与断点（多个事件同时发生时中断）。

## 8.2.6 其中一个探头引脚标记为5V。该探头的驱动能力有多大？

此引脚提供监视功能（允许您了解驱动器缓冲器上实际应用和使用的VDD有多大）。将任何电路连接到此引脚上都会影响仿真器运行。

MPLAB REAL ICE在线仿真器不能为目标供电，除非使用功率监视器（AC244008）。请参见《MPLAB® REAL ICE™在线仿真器功率监视器用户指南》（DS50002532A\_CN）。

## 8.2.7 是否有驱动板采用光电隔离或电气隔离？

它们均采用直流光电隔离。要与交流电压（120V）隔离，您将需要MPLAB® REAL ICE™隔离器单元（AC244005）。请参见《MPLAB® REAL ICE™在线仿真器隔离器单元规范》（DS50002529A\_CN）。

## 8.2.8 编程时钟（PGC）或编程数据（PGD）引脚存在哪些预防措施？

下面是编程预防措施：

- 标准模块化（ICSP）RJ-11电缆不允许超过约15 Mbps的通信时钟速度。对于这些高速应用，需要使用高性能工具包（高速/LVDS）电缆接口。
- 某些电路不可以使用这两个引脚。请参见第3.5.4节“目标电路设计注意事项”。另请参见《多工具设计忠告》（DS51764C\_CN）。

## 8.2.9 如何连接高性能工具包引脚CLK和DAT？

这些连接为可选，且用于SPI跟踪。更多信息，请参见第3.6.2节“SPI跟踪连接（高速/LVDS连接）”。

## 8.2.10 “使用标准板时，数据速率限制为15 MIPS”是什么意思？这是由内核处理器或者传输速率引起的吗？

标准板采用RJ-11电缆，使用跟踪、运行时观察和数据捕捉时，该电缆会对数据的传输速度加以限制以保证传输可靠。处理器以15 MIPS的速度运行时便会达到上限。跟踪时钟源自器件的主系统时钟。

15 MIPS是最坏情况下的限制。对于设计良好且信号完整性问题较少的电路板，该限值可能会更高。

## 8.2.11 15 MIPS限制是否适用于PIC32器件？

不适用。PIC32 MCU采用的时钟具有固定速率并且不源自目标系统时钟。此外，它采用的通信协议也更加可靠。

**8.2.12 调试运行速度为30 MIPS的dsPIC® DSC时，执行基本调试也需要采用高速通信吗？**

可以在任意器件频率下通过标准通信或高速（高性能工具包）通信完成基本调试。

**8.2.13 我的目标板连接器用于标准通信，但我想使用高速通信。是否可以结合使用高速/LVDS通信（高性能工具包）电缆、高速到标准转换器板和标准通信（ICSP）电缆来实现？**

不可以。您不能将标准/ICSP电缆用于高速通信，即高器件运行频率。使用RJ-11转换器板时，由于电缆的传输质量较低，因此会导致信号完整性问题。

**8.2.14 如果使用高速/LVDS通信，必须连接接收板的引脚7-8，还是可以将它们悬空？**

可以将它们悬空。高速接收板可对其实现弱下拉。

**8.2.15 高速接收板上的辅助引脚6具有什么功能？**

此引脚保留供仿真器使用。不要使用引脚6。

## 8.3 插装跟踪运行

插装跟踪有三种类型：本机跟踪、SPI跟踪和I/O端口跟踪。详细信息，请参见[第6.3.3节“跟踪类型”](#)。并非所有器件都支持每种类型的插装跟踪。

以下问题与这些跟踪类型有关。

### 8.3.1 使用插装跟踪时，连接是否无论如何都要采用电气隔离？

如果使用本机跟踪，则可使用光电隔离或MPLAB® REAL ICE™隔离器单元（AC244005）与高交流电压隔离。这不同于SPI跟踪和I/O端口跟踪。

### 8.3.2 仅使用5或6个ICSP引脚可以实现跟踪吗？

使用标准ICSP接口即可实现本机跟踪。SPI跟踪和I/O端口跟踪需要额外的硬件。请参见[第3.6节“跟踪连接”](#)。

### 8.3.3 在Project Properties对话框（File>Project Properties, REAL ICE类别, Clock选项类别）中输入执行速度实际上会设置通信时钟吗？

不会。该选项卡中的输入框仅向仿真器报告速度，以便其精确控制时序。只有本机跟踪、数据捕捉和运行时观察时需要报告时钟。如要正在使用时钟切换，这可能导致这些功能发生问题。

### 8.3.4 何时使用SPI跟踪？该跟踪具有哪些额外优势？

SPI跟踪可用于没有本机跟踪功能的器件。此外，SPI跟踪的速度快于本机跟踪，并且对代码长度的影响较小。关于比较的详细信息，请参见[第6.3.3节“跟踪类型”](#)。

### 8.3.5 要使用SPI跟踪，如何进行硬件连接？

对于串行SPI端口跟踪，需要使用器件的SPI SDO（串行数据输出）和SCK（串行时钟）引脚。这两个引脚必须分别连接到高性能工具包接收板上的DAT和CLK引脚接口。更多信息，请参见[第3.6.2节“SPI跟踪连接（高速/LVDS连接）”](#)。

### 8.3.6 SPI跟踪使用哪两个引脚？

使用的引脚为：

- SDO（串行数据输出）→ DAT（引脚7）
- SCK（串行时钟输出）→ CLK（引脚8）

更多信息，请参见[第3.6.2节“SPI跟踪连接（高速/LVDS连接）”](#)。

### 8.3.7 使用SPI跟踪时，如何进行正确的端口设置，即模式和同步/异步等？

该设置由MPLAB X IDE完成，因此您无需考虑该设置所需的代码。跟踪将支持64个跟踪点和64个记录点。详细信息，请参见[第6.3.9节“跟踪/记录ID编号的更多信息”](#)。

SPI——通信协议MODE1，时钟高电平，采样的下降沿。



## 8.3.8 使用 I/O 端口跟踪时，如何进行正确连接？

连接会根据所用端口的不同而有所不同。在 MPLAB X IDE 中，在属性页中选择端口后，将显示相应的端口分配。更多信息，请参见第 3.6.3 节“[I/O 端口跟踪连接（逻辑端口）](#)”。

## 8.3.9 可以使用任意端口吗？

器件上提供的端口都可以使用，但不可与当前使用的 PGC 和 PGD 引脚复用。

## 8.3.10 八（8）个端口引脚中，哪一个时钟引脚？

详细信息，请参见第 6.3.9 节“[跟踪/记录 ID 编号的更多信息](#)”。

## 8.3.11 调试期间，用于跟踪的 I/O 端口是否可用作通用 I/O？

对于 8 位器件，将端口定义为用于跟踪后，**不应**在代码中访问这些端口。

对于 16 位器件，如果使用字节写操作，则可以写入端口中没有用于跟踪功能的 8 位部分（高 8 位）。下面的示例只会写入端口的高位部分（高 8 位）。

```
#define high(num) (((BYTE *)&num)[1])
#define low(num)  (((BYTE *)&num)[0])
high(PORTA) = 0x12;
```

## 8.3.12 无法执行跟踪。这是怎么回事？

考虑以下情况：

- 使用跟踪需要某些特定的工具版本。请参见第 6 章“[特定调试功能](#)”。
- 只有 C 代码可与跟踪配合使用，汇编代码则不行。
- 不能跟踪行内汇编代码（C 代码中的汇编代码）。
- 添加跟踪宏后，必须重新编译并重新编程代码。
- 确保未禁止跟踪，即，Project Properties 窗口中 Real ICE 类别下的“Trace and Profiling”选项类别的“Data Collection Selection”不应为“Off”。
- 本机跟踪、数据捕捉和运行时观察无法同时使用。
- 必须在 Project Properties 窗口中 Real ICE 类别下的“Clock”选项类别中为本机和 SPI 跟踪报告目标时钟频率。
- 对于端口 I/O 跟踪：
  - 所有 8 个引脚必须专用于跟踪（即，不能与当前使用的 PGC 和 PGD 引脚复用）。
  - 确保所选的端口能够输出 0x00 和 0xFF。要进行测试，可首先将端口 TRIS 设置为 0（所有端口引脚为输出），然后在 Watches 窗口中将 LAT 设置为某个值。写入 LAT 的值应出现在端口引脚上。
- 确保引脚设置为数字 I/O 而非模拟。

## 8.4 一般问题

下列问题涵盖了一般的仿真器功能。

### 8.4.1 器件一直无法编程。这是怎么回事？

如果存在更改闪存的代码，编程后使用 **Run** (*Run>Run Project*) 将立即运行代码，并且校验操作会失败。为防止编程后运行代码：

1. 调试工具保持连接状态——*Tools>Options, Embedded* 按钮，**Generic Settings** (通用设置) 选项卡，选中“Maintain active connection to hardware tool” (与硬件工具保持活动连接)。
2. 选择“Hold in Reset” (保持复位)。

### 8.4.2 我已经设置了断点，但程序没有在断点处停止。为什么？

考虑以下情况：

1. 您是否在运行模式下执行代码 (即选择 *Run>Run Project*)？在此模式下设置的断点无效。
2. 这些断点是否为序列的一部分？请记住，顺序断点只有完全符合顺序才会暂停执行。
3. 您是否看到断点 `skid`？请参见仿真器 MPLAB X IDE 帮助文件的“Limitations”部分。
4. 是否使用编译器优化？如果使用，则程序可能没有执行您想要中断处的指令。尝试在代码中更早处设置断点，然后从该处单步执行，观察器件执行的实际代码流程。

### 8.4.3 我无法在特定的代码行上设置断点。为什么？

无法设置断点的原因有很多：

1. 试图中断的行是不可执行代码。MPLAB X IDE 只允许在可执行代码上设置断点。
2. 您尝试设置断点的代码行可能已被编译器优化掉。在这种情况下，请尝试关闭优化。
3. 您尝试设置断点的代码行可能是编译器伪指令，而不是实际代码。
4. 可能用尽了所有可用于断点的资源。
5. 您可能处于错误的文件中。如果打开了包含具有相似名称文件的多个项目，确保使用了需要的文件。
6. 如果使用 `ifdef` 代码，确保已将代码预处理为可执行代码。

您可以使用 **Breakpoints** (断点) 窗口帮助确定所有断点的状态。

### 8.4.4 我并没有设置断点，但代码中却有一个断点。这是怎么回事？

您所看到的断点是虚拟断点。断点有时会在其不该使能的时候变为使能状态。这时只需禁止或删除断点。为此，您可能需要跳转到反汇编窗口，或者只需右键单击并选择删除所有断点。如果这不起作用，请尝试关闭并重新打开反汇编窗口。

## 8.4.5 可以使用调试器来调试经过优化的代码吗？

强烈建议在调试期间关闭C编译器优化（见Project Properties窗口）。优化将极大地改变可执行代码与源文件的对应，因此调试器可能会在调试某些代码时发生异常。

## 8.4.6 数据捕捉未正常工作。这是怎么回事？

数据捕捉的速度对于USB通信或目标环境（噪声）来说过高。可采取以下措施：

1. 降低时钟速度。请参见第11.3.6节“时钟”。
2. 减少Watches窗口中运行时观察的数量。
3. 检查器件勘误表以确保不存在数据捕捉问题。

## 8.4.7 无法执行PIC32指令跟踪。这是怎么回事？

考虑以下情况：

- 使用跟踪需要某些特定的工具版本。请参见第6章“特定调试功能”。
- 确保没有禁止跟踪，即，Project Properties窗口中Real ICE类别下的“Trace and Profiling”选项类别的“Data Collection Selection”不应为“Off”。
- 确保引脚设置为数字I/O而非模拟。

## 8.4.8 我的程序在跟踪时发生暂停，并破坏了跟踪数据。这是怎么回事？

这可能是由看门狗定时器导致的。确保已通过正确设置配置位将其禁止。

## 8.4.9 我的PC进入掉电/休眠模式，并且现在仿真器无法工作。这是怎么回事？

长时间使用仿真器时（尤其是用作调试器时），请确保在PC操作系统的“电源选项”对话框窗口中禁止休眠模式。跳转到“休眠”选项卡，清除或取消选中“启用休眠”复选框。这可确保所有USB子系统组件之间保持所有通信。

## 8.4.10 我已将外设设置为暂停时不冻结，但它却突然发生冻结。这是怎么回事？

对于16位和32位器件，外设控制寄存器中的一个保留位（通常为bit 14或bit 5）由调试器用作冻结位。如果您对整个寄存器执行过写操作，那么您可能改写了此位。（用户可在调试模式下访问此位。）

为避免上述问题，请只对应用中想要更改的位（通过BTS和BTC）而不是整个寄存器（通过MOV）执行写操作。

## 8.4.11 发生意外复位。如何确定复位的原因？

考虑以下操作：

- 确保已使用配置位禁止了看门狗定时器。
- 要确定复位原因，请检查RCON寄存器。
- 在中断服务程序（Interrupt Service Routine, ISR）中处理陷阱/中断。请参见下文。

16位器件：应包括trap.c样式的代码：

```
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);
}
:
```

32位器件：应包括exception.c样式的代码：

```
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);
}
:
```

## 8.4.12 如何手动下载固件？

通常，您无需手动下载固件，MPLAB X IDE会自动完成此操作。详细信息，请参见[第4.3节“固件升级”](#)。

## 8.4.13 下载固件时意外断开仿真器的连接。该怎么办？

请参见[第4.3.3节“安装固件时通信中断”](#)。

## 8.4.14 我的代码更新了闪存；为什么没有在存储器窗口中看到变化？

为看到这些变化，必须对存储器执行读操作。

### 8.4.15 在这里没有找到我的问题。该怎么办？

请尝试下列资源：

- [第3.10节“仿真器使用的资源”](#)
- [第9.2节“错误、警告和信息性消息”](#)
- [第9.3节“常规纠正措施”](#)

注:

---

---

## 第9章 消息

---

---

### 9.1 简介

MPLAB REAL ICE 在线仿真器可产生多种不同的输出消息。许多错误消息都可以用常规纠正措施解决。

- [错误、警告和信息性消息](#)
- [常规纠正措施](#)

### 9.2 错误、警告和信息性消息

MPLAB REAL ICE 在线仿真器可产生以下类型的输出消息：

- 错误消息——在纠正指定错误前，将不能继续编译/执行应用程序。
- 警告消息——在解决指定警告前，可以继续编译/执行应用程序，但应用程序可能无法正常工作。
- 信息性消息——显示关于编译、执行或测试的信息。

许多错误或警告消息都含有关于如何解决该消息所提出问题的说明。如果遇到其他问题，请查阅[第9.3节“常规纠正措施”](#)。

一些错误或警告消息可能需要特定的纠正措施，如下文所述。

**The Debug Executive is found but can't be communicated with. Please ensure your oscillator settings are correct. If the device supports internal RC try to connect via that mode first.**（已找到调试执行程序，但无法与其通信。请确保振荡器设置正确。如果器件支持内部RC，请首先尝试通过该模式进行连接。）

另请参见[第9.3.2节“仿真器到目标器件的通信错误纠正措施”](#)。

#### **Failed to program device**（无法编程器件）

请参见[第8.4.1节“器件一直无法编程。这是怎么回事？”](#)。

#### **Failed to send database**（无法发送数据库）

如果收到此错误，请执行以下步骤：

1. 尝试重新下载。该错误可能只发生一次。
2. 尝试手动下载编号最高的 .jam 文件。

#### **Failed to download firmware**（无法下载固件）

如果存在 Hex 文件：

- 重新连接并重试。
- 如果无法解决问题，则表明该文件可能已损坏。重新安装 MPLAB X IDE。

如果不存在 Hex 文件，则重新安装 MPLAB X IDE。

**Invalid streaming data was been detected. Run time watch or trace data may no longer be valid. Is recommended that you restart your debug session.**（检测到无效的流数据。运行时观察或跟踪数据可能不再有效。建议重新启动调试会话。）

请参见FAQ中的[“数据捕捉未正常工作。这是怎么回事？”](#)。

**Loopback test completed successfully. Your REAL ICE is functioning properly. If you are still having problems with your target circuit please check the Target Board Considerations section of the online help.** (环回测试成功完成。REAL ICE 正在正常工作。如果您的目标电路仍存在问题，请查阅在线帮助的“目标板注意事项”部分。)

请参见第 12.8 节“目标板注意事项”。

**REAL ICE is busy. Please wait for the current operation to finish.** (REAL ICE 忙。请等待当前操作完成。)

如果在尝试取消将仿真器选作调试器或编程器时收到此错误：

1. 等待——使仿真器有时间完成任何应用程序任务。然后重新尝试取消选择仿真器。
2. 选择 Halt (暂停) 来停止所有运行中的应用程序。然后重新尝试取消选择仿真器。
3. 将仿真器从 PC 上拔下。然后重新尝试取消选择仿真器。
4. 关闭 MPLAB X IDE。

**Target Device ID (0x%x) does not match expected Device ID (0x%x).** (目标器件 ID (0x%x) 与预期器件 ID (0x%x) 不匹配。)

如果目标器件 ID 为 0x0，则器件可能未正确连接到目标板或未安装到目标板。

**The target device is not ready for debugging. Please check your Configuration bit settings and program the device before proceeding. The most common causes for this failure are oscillator and/or PGC/PGD settings.** (目标器件未准备好调试。请在检查配置位设置并编程器件后再继续。此错误的最常见原因是振荡器和/或 PGC/PGD 设置。)

当您第一次在尚未编程器件就尝试运行时将收到此消息。如果此后收到此消息，或者在编程器件后立即收到此消息，请参见第 9.3.6 节“调试失败纠正措施”。

**Target device was not found (could not detect target voltage VDD). You must connect to a target device to use MPLAB REAL ICE.** (未找到目标器件 (无法检测到目标电压 VDD)。必须连接到目标器件来使用 MPLAB REAL ICE。)

另请参见第 9.3.2 节“仿真器到目标器件的通信错误纠正措施”。

**Unable to download debug/program executive.** (无法下载调试/编程执行程序。)

如果尝试调试时收到此错误，请执行以下步骤：

1. 取消将仿真器选作调试工具。
2. 关闭项目，然后关闭 MPLAB X IDE。
3. 重新启动 MPLAB X IDE，然后重新打开项目。
4. 重新选择仿真器作为调试工具并再次尝试编程目标器件。



## 9.3 常规纠正措施

以下常规纠正措施可解决您遇到的问题：

- [读/写错误纠正措施](#)
- [仿真器到目标器件的通信错误纠正措施](#)
- [仿真器到PC的通信错误纠正措施](#)
- [安装损坏纠正措施](#)
- [USB端口通信错误纠正措施](#)
- [调试失败纠正措施](#)
- [内部错误纠正措施](#)

### 9.3.1 读/写错误纠正措施

如果收到读或写错误：

1. 是否按下了Abort（中止）？此操作可能产生读/写错误。
2. 重试操作。该错误可能只发生一次。
3. 确保目标板已上电并且器件的电压正确。关于所需器件电压，请参见器件数据手册。
4. 确保仿真器到目标板的连接正确（PGC和PGD已连接）。
5. 对于写错误，确保在Settings对话框的**Program Memory**（程序存储器）选项卡上选中“Erase all before Program”（编程之前全部擦除）。
6. 确保所使用的电缆的长度正确——标准通信最长6"，高速通信最长10"。

### 9.3.2 仿真器到目标器件的通信错误纠正措施

MPLAB REAL ICE在线仿真器与目标器件互不同步。

1. 选择**Reset**，然后重试操作。
2. 确保所使用的电缆的长度正确——标准通信最长6"，高速通信最长10"。

### 9.3.3 仿真器到PC的通信错误纠正措施

MPLAB REAL ICE在线仿真器与MPLAB X IDE互不同步。

1. 拔下仿真器，然后重新插入。
2. 重新连接到仿真器。
3. 重试操作。该错误可能只发生一次。
4. 所安装的MPLAB X IDE版本可能与MPLAB REAL ICE在线仿真器上加载的固件版本不一致。请按照第9.3.4节“[安装损坏纠正措施](#)”中所述的步骤进行操作。
5. PC USB端口可能存在问题。请参见第9.3.5节“[USB端口通信错误纠正措施](#)”。

### 9.3.4 安装损坏纠正措施

该问题很可能是由于MPLAB X IDE的安装不完整或者损坏引起的。

1. 卸载PC上所有版本的MPLAB X IDE。
2. 重新安装所需的MPLAB X IDE版本。
3. 如果问题仍然存在，请联系Microchip。

## 9.3.5 USB 端口通信错误纠正措施

该问题很可能是由于通信端口错误或者不存在引起的。

1. 重新连接到 MPLAB REAL ICE 在线仿真器。
2. 确保仿真器已与 PC 上适当的 USB 端口连接。
3. 确保在仿真器设置中选择了适当的 USB 端口。
4. 确保 USB 端口未被其他器件使用。
5. 如果要使用 USB 集线器，请确保集线器已上电。
6. 确保已加载 USB 驱动程序。

## 9.3.6 调试失败纠正措施

MPLAB REAL ICE 在线仿真器无法执行调试操作。引发此错误的原因很多。请参见 [第7章“故障诊断首要步骤”](#)。

## 9.3.7 内部错误纠正措施

内部错误通常难以预料并且不应该发生。它们主要用于 Microchip 内部开发。

最可能的原因是安装损坏（[第9.3.4节“安装损坏纠正措施”](#)）。

另一个可能的原因是系统资源不足。

1. 尝试重新启动系统以释放存储器。
2. 确保硬盘上有合理的可用空间（而且不是过多的磁盘碎片）。

如果问题仍然存在，请联系 Microchip。



---

---

## 第10章 工程技术说明 (ETN)

---

---

以下ETN与MPLAB REAL ICE在线仿真器相关。关于详细信息，请参见产品网页。

**ETN-30:** 适用于组装编号为10-00401-R1或更低的产品。

注:



---

---

## 第4部分——软件和硬件参考

---

---

第11章 仿真器功能汇总 .....	103
第12章 硬件规范 .....	113

注:

---

---

## 第11章 仿真器功能汇总

---

---

### 11.1 简介

下面列出了MPLAB REAL ICE在线仿真器功能的汇总。

- [仿真器选择与切换](#)
- [仿真器选项选择](#)
- [仿真器窗口和对话框](#)

### 11.2 仿真器选择与切换

使用Project Properties对话框可选择或切换项目的仿真器。要进行切换，必须有多个MPLAB REAL ICE在线仿真器与计算机相连。MPLAB X IDE将通过显示两个不同序列号来区分两个仿真器。

要选择或更改项目使用的仿真器：

1. 通过执行下列一项操作打开Project Properties对话框：
  - a) 单击Projects窗口中的项目名称并选择 [File>Project Properties](#)。
  - b) 右键单击Projects窗口中的项目名称并选择“Properties”。
2. 在“Categories”下，单击“Conf:[default]”（配置：[默认]）。
3. 在“Hardware Tools”（硬件工具）下，找到“REAL ICE”并单击序列号（Serial Number, SN）来选择项目中使用的仿真器。

### 11.3 仿真器选项选择

在Project Properties对话框的仿真器属性页面上设置仿真器选项。

1. 通过执行下列一项操作打开Project Properties对话框：
  - a) 单击Projects窗口中的项目名称并选择 [File>Project Properties](#)。
  - b) 右键单击Projects窗口中的项目名称并选择“Properties”。
2. 在“Categories”下，单击“REAL ICE”
3. 从“Options categories”中选择属性页面。单击一个选项并在其下面的文本框中查看其说明。单击选项的右侧可对其进行更改。

可用的选项类别有：

- [要编程的存储器](#)
- [调试选项](#)
- [编程选项](#)
- [冻结外设](#)
- [跟踪和性能分析](#)
- [时钟](#)
- [固件](#)
- [外部触发信号](#)

## 11.3.1 要编程的存储器

选择目标器件中要编程的存储器。

如果在第 11.3.3 节“编程选项”下选择了“Erase All Before Program”（编程前全部擦除），则将在编程之前擦除所有器件存储器。要选择在擦除后仅编程某些存储器，选中特定的存储器类型。要保护不同类型的存储器的值，选中时保护存储器类型并选中特定的存储器类型；选中“Preserve Memory”（保护存储器）在擦除前将当前内容写入缓冲区，选中“Memory”（存储器）在擦除后将内容写回该存储器，这里的 *Memory* 是存储器类型，例如 EEPROM。

表 11-1: 要编程的存储器选项类别

Auto select memories and ranges (自动选择存储器和范围)	<b>Allow REAL ICE to Select Memories (允许 REAL ICE 选择存储器)</b> —— 仿真器根据选择的器件和默认设置确定要编程的内容。 <b>Manually select memories and ranges (手动选择存储器和范围)</b> —— 选择要编程的存储器的类型和范围 (见下文)。
<i>Memory</i>	选中时可编程 <i>Memory</i> ，这里的 <i>Memory</i> 是存储器类型。类型包括：EEPROM、ID 存储器、引导闪存和辅助存储器。
Program Memory (程序存储器)	选中时可编程下面指定的目标程序存储器范围。
Program Memory Range(s) (hex) (程序存储器范围，十六进制) *	程序存储器中要编程、读取或校验的范围的起始地址和结束地址 (以十六进制表示)。 <b>注：</b> 地址范围不适用于擦除功能。执行擦除功能将擦除器件上的所有数据。
Preserve Program Memory (保护程序存储器)	选中时不会编程下面指定的目标程序存储器范围。确保代码不受代码保护。
Preserve Program Memory Range(s) (hex) (保护程序存储器范围，十六进制) *	编程、读取或校验时目标程序存储器中要保护的范围的起始地址和结束地址 (以十六进制表示)。此存储器从目标程序存储器中读取且与现有的 MPLAB X IDE 存储器重叠。
Preserve <i>Memory</i>	选中时可保护 <i>Memory</i> 以进行重新编程，这里的 <i>Memory</i> 是存储器类型。类型包括：EEPROM、ID 存储器、引导闪存和辅助存储器。确保代码不受代码保护。
Preserve <i>Memory</i> Range(s) (hex) (保护存储器范围，十六进制) *	编程、读取或校验时目标存储器中要保护的范围的起始地址和结束地址 (以十六进制表示)。 <i>Memory</i> 是存储器类型，包括 EEPROM、ID 存储器、引导闪存和辅助存储器。此存储器从目标程序存储器中读取且与现有的 MPLAB X IDE 存储器重叠。确保代码不受代码保护。

\* 如果接收到由于范围错误而产生的编程错误，请确保范围不超过可用 / 剩余的器件存储器。



## 11.3.2 调试选项

如果项目中所使用器件具有软件断点，则可使用软件断点。

**表11-2: 调试选项的选项类别**

Use Software Breakpoints (使用软件断点)	选中时可使用软件断点。 取消选中时则使用硬件断点。请参见以下讨论，了解哪种断点类型最适合您的应用程序。
--------------------------------------	--

**表11-3: 软件断点与硬件断点**

特性	软件断点	硬件断点
断点数	无限	有限
断点写入到	程序存储器	调试寄存器
设置断点的时间	取决于振荡器速度，可能需要数分钟	最短
Skidding	无	有

**注:** 使用软件断点进行调试会影响器件的耐用性。因此，建议不应将以这种方式使用过的器件用作生产器件。

## 11.3.3 编程选项

选择在编程之前擦除所有存储器或者合并代码。

**表11-4: 编程选项的选项类别**

Erase All Before Program	选中时可在开始编程之前擦除全部存储器。 除编程未用过的器件或已擦除的器件外，其余情况下都必须选中此复选框。如果未选中此复选框，则不对器件执行擦除操作并且程序代码将与器件中已存在的代码合并。
Enable Low Voltage Programming (使能低电压编程)	仅用于 <i>Programmer Settings</i> (编程器设置), <i>PIC12F/16F1xxx</i> 器件: <ul style="list-style-type: none"> <li>如果LVP配置位设置为“Low-voltage programming enabled” (使能低电压编程)，则可在高电压 (默认) 或低电压 (在此处使能) 下编程。</li> <li>如果LVP配置位设置为“High-voltage on MCLR/VPP must be used for programming” (必须使用MCLR/Vpp上的高电压进行编程)，则只能在高电压下编程。</li> </ul>
Do Not Erase Auxiliary Memory (不要擦除辅助存储器)	对于支持辅助存储器的器件: 选中以在编程时不擦除辅助存储器。 取消选中以在编程时擦除辅助存储器。

## 11.3.4 冻结外设

选择在程序暂停时冻结或不冻结的外设。可用选项取决于选择的器件或调试头。

表11-5: 冻结外设的选项类别

Freeze Peripherals (冻结外设)	选中以在暂停时冻结所有外设。 取消选中可取消冻结所有外设。 此选项适用于PIC12/16/18 MCU。
Peripheral Freeze Enable (外设冻结使能) <i>Peripheral List (外设列表)</i>	选中可选择要冻结的外设。 取消选中可取消冻结所有外设。 此选项适用于AC244066。
<i>Peripheral List</i>	选中以在暂停时冻结外设 <i>Peripheral</i> (外设)。 取消选中可取消冻结外设 <i>Peripheral</i> 。 此选项适用于16位和32位MCU。

### PIC12/16/18 MCU 器件

要在暂停时冻结/取消冻结器件所有外设，请选中/取消选中“Freeze Peripherals”（冻结外设）复选框。如果所需外设未暂停，请注意，某些外设没有暂停时冻结功能，因而无法受仿真器控制。

### AC244066 仿真扩展包 (PIC16F1619-ME2)

选中“Peripheral Freeze Enable”（外设冻结使能）复选框来选择要冻结的外设。默认情况下选择所有外设。取消选中此复选框会取消冻结所有外设。如果所需外设未暂停，请注意，某些外设没有暂停时冻结功能，因而无法由仿真器控制此功能。

### dsPIC、PIC24和PIC32 器件

要在暂停时冻结/取消冻结外设，请从列表中选中/取消选中该外设。如果未在列表中看到任何外设，则选中/取消选中“Freeze All Other Peripherals”（冻结所有其他外设）。如果所需外设未暂停，请注意，某些外设没有暂停时冻结功能，因而无法由仿真器控制此功能。

## 11.3.5 跟踪和性能分析

您可以根据为项目选择的器件，在调试时使用跟踪、PC采样/性能分析或其他数据收集功能。请按下面几节所述使能并设置这些功能。

### 8位和16位器件

此页面上的可用选项取决于项目器件的跟踪/性能分析功能。关于跟踪和性能分析的更多信息，请参见第6章“特定调试功能”。

**表11-6: 跟踪/性能分析选项类别**

Data Collection Selection	<p>使能/禁止数据收集。</p> <ul style="list-style-type: none"> <li>• Off——不收集目标数据。</li> <li>• User Instrumented Trace——请参见第6.3节“插装跟踪”。</li> <li>• PC Sampling——请参见第6.6节“PC采样——仅限8位和16位MCU”。</li> <li>• Function Level Profiling——请参见第6.8节“函数级别性能分析”。</li> <li>• Jump Trace（跳转跟踪）——请参见第6.5节“跳转跟踪——仅限EP器件”。</li> <li>• Power Monitor (Target Power Sampling)（功率监视器（目标功率采样））——请参见《MPLAB® REAL ICE™ 在线仿真器功率监视器用户指南》（DS50002532A_CN）</li> </ul>
Data File Path and Name	<p>输入或更改用于存储数据的文件的路径和/或名称。</p> <ul style="list-style-type: none"> <li>• 输入文件名称（路径将是相对于项目的相对路径）——推荐</li> <li>• 输入路径和文件名称（路径将是绝对路径）</li> <li>• Browse（浏览）（...）到文件，选择“Absolute”（绝对），然后选择该文件并单击Save（保存）（路径将是绝对路径）</li> </ul> <p><b>注：</b>浏览到文件时不要选择“Relative”（相对），否则MPLAB X IDE将无法找到该文件。运行时，将接收到路径不存在的警告消息。</p>
Data File Maximum Size (bytes)	<p>设置数据文件的最大大小。</p> <p>目标功率采样每次采样占用12字节或18字节（含PC数据）。</p> <p>可以根据选择的跟踪类型将文件大小调整为这些字节大小之一的倍数。其他跟踪数据类型可能使用与上述字节大小不同的记录字节大小。</p>
Data Buffer Maximum Size (bytes)	<p>设置数据缓冲区的大小，最大54600字节（仿真器内）。</p> <p>对于目标器件运行时在存储器中缓冲的跟踪/采样数据，各个跟踪或采样条目的大小根据跟踪/采样类型和正在使用的器件和工具的不同而有所不同。通常，最好将此缓冲区设置得尽量大。</p> <p>例如，带指令跟踪功能的增强型PIC16为每个存储器内条目使用1至3字节。每个条目还将生成一个13字节的REAL ICE指令跟踪条目。通常，每个这样的存储器内记录都将转换为跟踪数据文件条目行，数据文件大小说明中对此进行了详细介绍（参见跟踪/采样文件条目大小的数据文件大小说明）。</p>
Stall CPU When Trace Buffer is Full（跟踪缓冲区已满时暂停CPU）	<p>跟踪缓冲区已满时停止执行。在上述选项中设置缓冲区大小。</p>
<b>User Instrumented Trace选项</b>	
Disable Trace Macros	<p>选中时可暂时禁止跟踪宏，取消选中时可使能跟踪宏。</p> <p>要禁止跟踪，删除所有宏并在Data Collection Selection下选择Off。</p>
Communications Medium	<p>如果可用，从以下各项（取决于器件）中选择跟踪介质：Native、I/O Port和SPI。</p>
I/O Port Selection（I/O端口选择）	<p>指定用于I/O端口跟踪的器件端口。</p> <p>将列出所选器件的可用组合。</p>
SPI Selection（SPI选择）	<p>指定用于SPI跟踪的器件SPI引脚。将列出所选器件的可用引脚。</p>

# 适用于MPLAB X IDE的仿真器用户指南

表11-6: 跟踪/性能分析选项类别 (续)

PC Sampling (PC采样) 选项	
Timer Selection (定时器选择) (未被应用程序代码使用)	选择用于计数PC采样的器件定时器。 <b>注:</b> 应用程序中将无法再使用此定时器, 它将专用于PC采样。 <b>注:</b> 只能选择一个定时器; 不能通过组合两个定时器构成32位定时器。使用32位定时器对的其中一个定时器时, 无法将该定时器对作为32位定时器使用。
Timer Interrupt Priority (定时器中断优先级)	选择定时器的中断优先级。 <b>注:</b> 所选优先级应高于应用程序中设置的其他优先级。否则, 其他优先级将比采样优先级高, 您将无法捕捉这些采样。
Timer Interval (定时器时间间隔)	输入采样间隔。 该值必须是整数值 (1、2和3等)。 如果您未捕捉数据, 则可能按当前时间间隔已丢失采样。请尝试调整单位选择和间隔, 例如, 如果之前为1毫秒, 尝试改为990微秒。
Timer Interval Units (定时器时间间隔单位)	选择采样间隔单位: <ul style="list-style-type: none"><li>• microseconds (微秒)</li><li>• milliseconds (毫秒)</li><li>• seconds (秒)</li><li>• instruction cycles (指令周期)</li></ul>
<b>Function Level Profiling</b> ——使用代码性能分析插件	
Time Stamp or Summary Profile Data	选中时生成时间戳数据或汇总性能分析数据。默认选择汇总性能分析数据。 <b>注:</b> 如果Power Monitor为Data Collection Selection, 则收集带时间戳的功率数据时, 该选项默认为Include Time Stamp。

## 32位器件

此页面上的可用选项取决于项目所使用器件的跟踪/性能分析功能。关于跟踪和性能分析的更多信息，请参见第6章“特定调试功能”。

**表 11-7: 跟踪/性能分析选项类别**

Data Collection Selection	<p>使能/禁止数据收集。</p> <ul style="list-style-type: none"> <li>• Off——不收集目标数据。</li> <li>• Instruction Trace/Profiling——请参见第6.4节“PIC32指令跟踪——仅限PIC32 MCU”和第6.7节“PC性能分析——仅限32位MCU”。</li> <li>• User Instrumented Trace——请参见第6.3节“插装跟踪”。</li> <li>• Power Monitor (Target Power Sampling)——请参见《MPLAB® REAL ICE™ 在线仿真器功率监视器用户指南》(DS50002532A_CN)</li> </ul>
Data File Path and Name	<p>输入或更改用于存储数据的文件的路径和/或名称。</p> <ul style="list-style-type: none"> <li>• 输入文件名称（路径将是相对于项目的相对路径）——推荐</li> <li>• 输入路径和文件名称（路径将是绝对路径）</li> <li>• Browse (...) 到文件，选择“Absolute”，然后选择该文件并单击 <b>Save</b>（路径将是绝对路径）</li> </ul> <p><b>注：</b>浏览到文件时不要选择“Relative”，否则MPLAB X IDE将无法找到该文件。运行时，将接收到路径不存在的警告消息。</p>
Data File Maximum Size (bytes)	<p>设置数据文件的最大大小。</p> <p>使用仿真器时，跟踪数据文件中指令跟踪数据的每一行都需要13字节。目标功率采样每次采样占用12字节或18字节（含PC数据）。可以根据选择的跟踪类型将文件大小调整为这些字节大小之一的倍数。其他跟踪数据类型可能使用与上述字节大小不同的记录字节大小。</p>
Data Buffer Maximum Size (bytes)	<p>设置数据缓冲区的大小，最大54600字节（仿真器内）。</p> <p>对于目标器件运行时在存储器中缓冲的跟踪/采样数据，各个跟踪或采样条目的大小根据跟踪/采样类型和正在使用的器件和工具的不同而有所不同。通常，最好将此缓冲区设置得尽量大。</p> <p>例如，PIC32指令跟踪每“帧”占用8字节，这可在跟踪文件中产生超过50个13字节的REAL ICE指令跟踪条目。</p>
<b>User Instrumented Trace 选项</b>	
Disable Trace Macros	<p>选中时可暂时禁止跟踪宏，取消选中时可使能跟踪宏。</p> <p>要禁止跟踪，可移除所有宏并在Data Collection Selection下选择Off。</p>
Communications Medium	<p>如果可用，选择跟踪介质（取决于器件）：Native。</p>
<b>Function Level Profiling——使用代码性能分析插件</b>	
Time Stamp or Summary Profile Data	<p>选中时生成时间戳数据或汇总性能分析数据。默认选择汇总性能分析数据。</p> <p><b>注：</b>如果Power Monitor为Data Collection Selection，则收集带时间戳的功率数据时，该项目默认为Include Time Stamp。</p>

# 适用于MPLAB X IDE的仿真器用户指南

## 11.3.6 时钟

在此选项类别下输入运行时时钟（指令）速度。这并不设置速度，而是将其值告知仿真器以进行运行时观察、数据捕捉和跟踪。

**注：** 可对数据捕捉和跟踪执行时钟切换操作，但是必须正确设置时钟，否则可能出现的问题。请输入要使用的最快指令速度。

表 11-8: 时钟选项类别

Use FRC in Debug mode (在调试模式下使用FRC) (仅限 dsPIC33E/F 器件和 PIC24E/F/H 器件)	调试时，使用器件内部快速 RC (FRC) 振荡器提供时钟，而不使用为应用程序指定的振荡器。这在应用程序时钟比较慢时非常有用。 选中此复选框可让应用程序以低速运行，但以更快的 FRC 速度进行调试。 更改此设置后需重新编程。 <b>注：</b> 未被冻结的外设将在调试时以 FRC 速度运行。
Target run-time instruction speed (目标运行时指令速度)	为所选的“速度单位”输入一个值。 <b>示例 1：</b> 对于 PIC24 MCU 和 32 MHz (HS) 的目标时钟振荡器，指令速度 = 32 MHz/2 = 16 MIPS。 <b>示例 2：</b> 对于 PIC18F8722 MCU 和采用 PLL (x4 = 40 MHz) 的 10 MHz (HS) 目标时钟振荡器，指令速度 = 40 MHz/4 = 10 MIPS。
Instruction speed units (指令速度单位)	选择下列一个单位： KIPS——每秒千 (10 <sup>3</sup> ) 条指令 MIPS——每秒百万 (10 <sup>6</sup> ) 条指令

## 11.3.7 固件

选择并加载仿真器固件。MPLAB X IDE 会为项目自动下载正确的固件。只有出现问题时才需更改此设置。

表 11-9: 固件选项类别

Use Latest Firmware (使用最新固件)	选中时可使用最新固件。取消选中时则选择下面的固件版本。
Firmware File (固件文件)	单击右侧的文本框可搜索与仿真器相关的固件文件。详细信息，请参见第 4.3 节“固件升级”。

## 11.3.8 外部触发信号

选择外部触发信号功能。关于触发信号的更多信息，请参见第 6.10 节“外部触发信号”。

表 11-10: 外部触发信号选项类别

触发信号 0 : 触发信号 7	选择触发信号功能： Off - Input——上升沿触发——触发时暂停 - Input——上升沿触发——触发时复位 - Input——下降沿触发——触发时暂停 - Input——下降沿触发——触发时复位 - Output——高电平至低电平脉冲——暂停时置为有效 - Output——高电平至低电平脉冲——运行时置为有效 - Output——低电平至高电平脉冲——暂停时置为有效 - Output——低电平至高电平脉冲——运行时置为有效
-----------------------	--

## 11.4 仿真器窗口和对话框

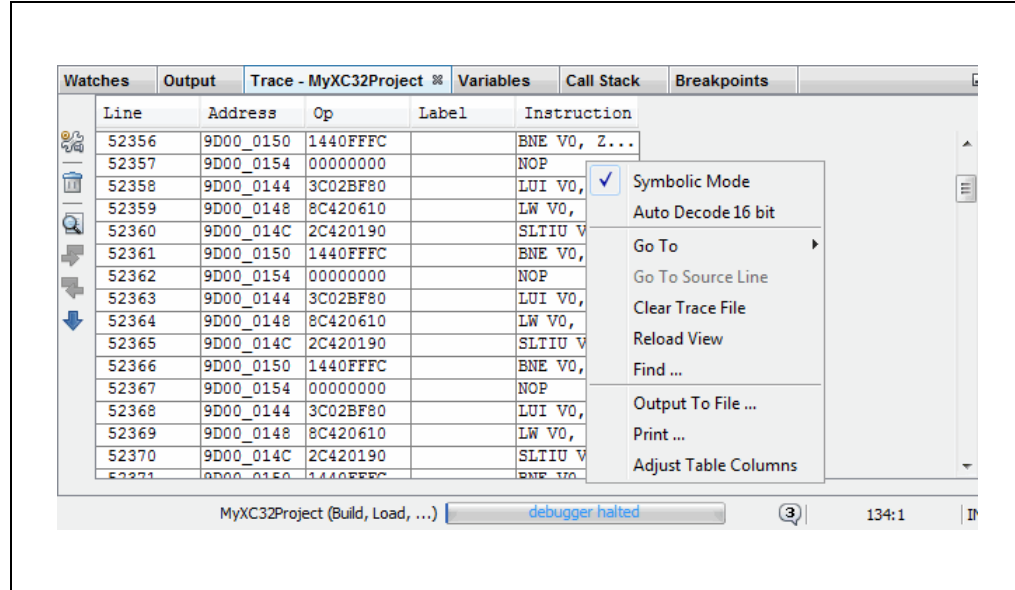
以下窗口和对话框专门用于仿真器和/或其他调试工具。

- [Trace 窗口和相关对话框](#)
- [应用程序输入/输出窗口和相关对话框](#)
- [PC Sampling 窗口和相关对话框](#)

### 11.4.1 Trace 窗口和相关对话框

跟踪窗口可显示跟踪的结果。此窗口可用于仿真器和模拟器。

图 11-1: TRACE 窗口



右键单击上面显示的窗口的某一列可弹出带有功能列表的菜单。关于这些功能的更多信息，请参见《MPLAB® X IDE 用户指南》（DS50002027C\_CN）中“MPLAB X IDE 窗口和对话框”下的“Trace 窗口”。

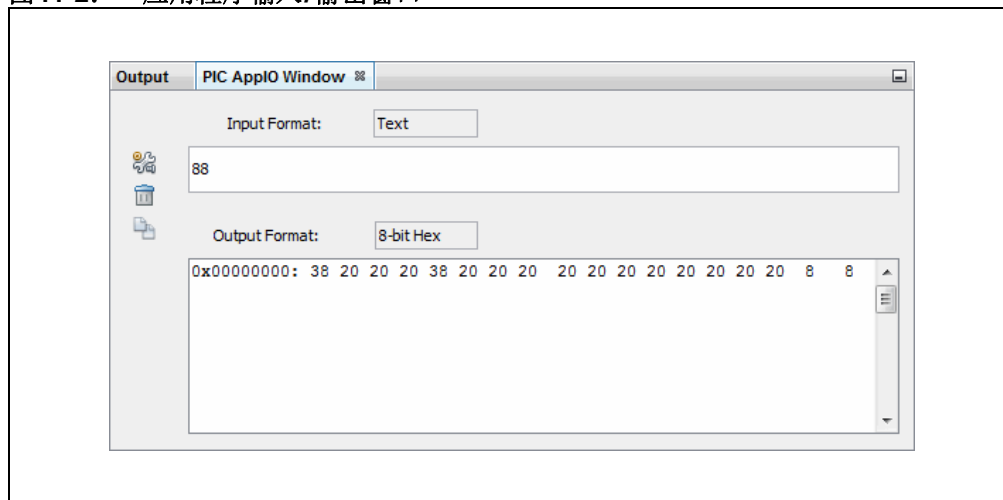
关于使用跟踪的更多信息，请参见：

- [第 6.3 节“插装跟踪”](#)
- [第 6.4 节“PIC32 指令跟踪——仅限 PIC32 MCU”](#)

## 11.4.2 应用程序输入/输出窗口和相关对话框

应用程序输入/输出窗口支持应用程序IO功能，其中运行时控制信息可通过MPLAB IDE发送到应用程序（APPIN），状态信息可由应用程序发送到MPLAB IDE（APPOUT）。

图 11-2: 应用程序输入/输出窗口



其他动作可通过按钮实现。右键单击Output文本框可弹出具有相同功能的菜单。

表 11-11: 应用程序输入/输出窗口按钮

按钮	说明
Properties	打开 Set App IO Properties 对话框。 设置输入和输出的格式：文本、8位十六进制、16位十六进制或32位十六进制。 使能/禁止数据捕捉。浏览至一个位置以将输出保存到文件中。
Clear App IO Output (清除 App IO 输出)	将Output文本框中的输出内容清除。
Copy App IO Output to Output View (将 App IO 输出复制到输出视图)	将Output文本框中的内容复制到Output窗口。

关于使用此窗口及其相关对话框的更多信息，请参见第6.9节“应用程序输入/输出”。

## 11.4.3 PC Sampling 窗口和相关对话框

程序计数器（PC）采样是确定各个应用函数所花费时间的方法，用于代码优化。PC Sampling窗口将显示函数名称、该函数的采样数量、采样数量相对于总采样数的百分比以及数量的条形图。

关于PC采样和性能分析的更多信息，请参见：

- 第6.6节“PC采样——仅限8位和16位MCU”
- 第6.7节“PC性能分析——仅限32位MCU”



---

---

## 第12章 硬件规范

---

---

### 12.1 简介

本章详细介绍了基本MPLAB REAL ICE在线仿真器系统的硬件和电气规范。

### 12.2 重点

本章讨论的主题包括：

- [USB 端口/电源](#)
- [仿真器主机](#)
- [标准通信硬件](#)
- [高速/LVDS通信硬件](#)
- [环回测试板](#)
- [目标板注意事项](#)

### 12.3 USB端口/电源

MPLAB REAL ICE在线仿真器通过与USB 2.0版兼容的USB端口与PC主机连接。USB连接器位于仿真器主机的背面。

系统可通过USB接口重载固件。

由USB接口为仿真器系统供电。根据USB规范，该仿真器被归类为高功耗系统，并且需要USB提供300 mA的电流以使仿真器在所有工作模式（仿真器/编程器）下正常工作。

**注：** MPLAB REAL ICE在线仿真器通过USB连接供电。目标板则由自身电源供电。仿真器不能为目标板供电。

**电缆长度**——已针对每块驱动板对计算机到仿真器的电缆长度进行了测试，以使系统正常工作。仿真器工具包中随附了相关电缆。

**集线器供电**——如果要使用USB集线器，请确保对集线器供电。此外，PC键盘的USB端口不能提供足够的电源来使仿真器工作。

## 12.4 仿真器主机

仿真器主机（DV244005）中包含封装在机壳中的主板，该板的端口可连接两种驱动板（用于与目标器件进行标准或高速通信）之一。仿真器机箱上有按钮、指示灯（LED）以及逻辑探头连接器接口。

### 12.4.1 主板

该组件带有接口处理器（dsPIC DSC）、传输速度为480 Mb/s的USB 2.0接口、用于常规系统控制和增加通信吞吐量的现场可编程门阵列（Field Programmable Gate Array, FPGA）、用于保存程序代码映像的SRAM（程序代码映像将被烧写到仿真器件片内闪存中）、外部触发逻辑、用户接口按钮以及LED指示灯。

MPLAB REAL ICE在线仿真器系统支持两种类型与目标处理器的接口。它们由标准驱动板和可选高速驱动板组成。这两种板通过板导槽插入仿真器主机。

板导槽的耐用性/可插入次数：10,000次。

### 12.4.2 按钮

按钮具有以下功能定义。

按钮	相关LED	说明
复位	状态	按下该按钮复位器件。
功能	状态	暂停——按下该按钮会使运行中的仿真器进入中断或停止状态。

### 12.4.3 指示灯（LED）

指示灯具有如下功能定义。

表12-1: LED指示灯

类型	颜色	状态	说明
工作	蓝色	点亮	已接通电源或者已连接目标器件。
状态	绿色	点亮	仿真器正常工作——待机。
		红色	操作失败。
	闪烁	USB出现通信错误或未安装驱动程序。	
橙色	点亮	仿真器忙。	

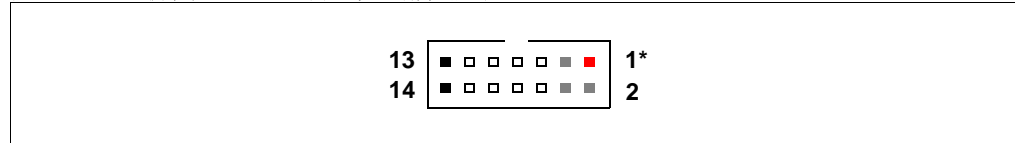
### 12.4.4 逻辑探头/外部触发接口

逻辑探头（ACICE0104）可以连接到监视器一侧的14引脚连接器，以处理用于触发外部设备的外部信号。此连接器包含8个可由用户选作输入或输出的输入/输出连接，其逻辑电平与目标工作电压成比例。

输出可以用于触发外部逻辑分析器或示波器，使开发人员可以根据在MPLAB X IDE内设置的触发标准捕捉关注的事件。外部触发信号是大约1.5 μs的脉冲。此值具有不确定性，应在脉冲边沿触发外部工具。

输入是触发总线的组成部分。输入可以有最长700 μs的延时，并且长度将根据所仿真架构（8位、16位或32位）的不同以及跟踪或流数据是否有效而有所变化。

**图 12-1: 仿真器上的逻辑探头引脚排列**



逻辑探头可与此连接器相连以提供表 12-2 中描述的功能。探头采用不同颜色标识便于识别。

**表 12-2: 逻辑探头引脚布局说明**

引脚	I/O	名称	功能	颜色
1	输出	VDD <sup>(1)</sup>	VDD 参考电压	红色
2	输出	NC	无连接	灰色
3	输出	NC	无连接	灰色
4	输入	TCLK	外部同步时钟	灰色
5	I/O	EXT7 <sup>(2)</sup>	外部输入/ 输出 bit 7	白色
6	I/O	EXT6	外部输入/ 输出 bit 6	白色
7	I/O	EXT5	外部输入/ 输出 bit 5	白色
8	I/O	EXT4	外部输入/ 输出 bit 4	白色
9	I/O	EXT3	外部输入/ 输出 bit 3	白色
10	I/O	EXT2	外部输入/ 输出 bit 2	白色
11	I/O	EXT1	外部输入/ 输出 bit 1	白色
12	I/O	EXT0 <sup>(2)</sup>	外部输入/ 输出 bit 0	白色
13	地	GND	系统地	黑色
14	地	GND	系统地	黑色

**注 1:** 不要将 VDD 与目标器件连接。

**2:** EXT0 和 EXT7 都临时用于环回测试。请确保二者没有连接在一起。

表 12-3 中列出了逻辑探头的电气规范。

**表 12-3: 逻辑探头电气规范**

逻辑输入	VIH = VDD x 0.7V (最小值)			
	VIL = VDD x 0.3V (最大值)			
逻辑输出	VDD = 5V	VDD = 3V	VDD = 2.3V	VDD = 1.65V
	VOH = 3.8V 最小值	VOH = 2.4V 最小值	VOH = 1.9V 最小值	VOH = 1.2V 最小值
	VOL = 0.55V 最大值	VOL = 0.55V 最大值	VOL = 0.3V 最大值	VOL = 0.45V 最大值

## 12.5 标准通信硬件

要使仿真器与目标板之间进行标准通信（第3.5.1节“标准通信”），应使用标准驱动板。

要将此类型的通信与调试头配合使用，可能需要器件特定的扩展包，其中包括包含所需-ICE/ICD器件的8引脚连接器调试头以及标准适配器板（8引脚到6引脚连接）。

**注：** 早期的调试头使用6引脚（RJ-11）连接器而不是8引脚连接器，所以这些调试头可以直接连接到仿真器。

关于调试头的更多信息，请参见“推荐读物”中的“*Processor Extension Pak and Header Specification*”。

### 12.5.1 标准驱动板

标准驱动板（AC244001）随仿真器一起提供，但也可单独购买，是与目标处理器连接的主要接口。它包含与高电压（VPP）和VDD检测线的连接，以及烧写和连接目标器件所需的时钟和数据连接。

VPP高压线可产生电压范围为0V至14V的可变电压，以满足特定仿真处理器的电压要求。

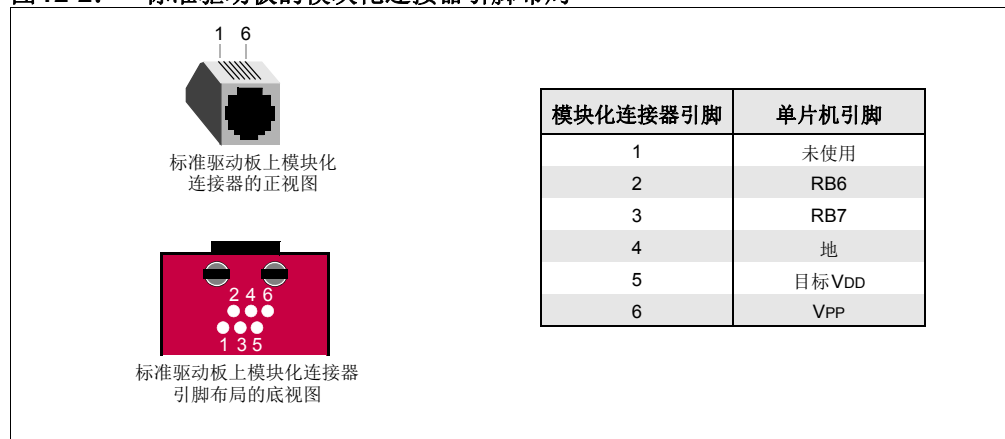
VDD检测连接对目标处理器的分流很小。实际的电源来自MPLAB REAL ICE在线仿真系统，而VDD检测线仅为跟踪目标电压提供基准。使用光电开关将VDD连接隔离。

时钟和数据连接接口具有如下特性：

- 时钟和数据信号处于高阻抗模式，即便没有给MPLAB REAL ICE在线仿真器系统加电也是如此
- 时钟和数据信号可免受错误目标系统或不正确的连接所导致的高电压的破坏
- 时钟和数据信号可免受错误目标系统中的电气短路所导致的大电流的破坏

**注：** 当使用标准驱动板时，实时数据流传输速率和跟踪速率被限制为15 MIPS。

图12-2： 标准驱动板的模块化连接器引脚布局



## 12.5.2 模块化电缆和连接器

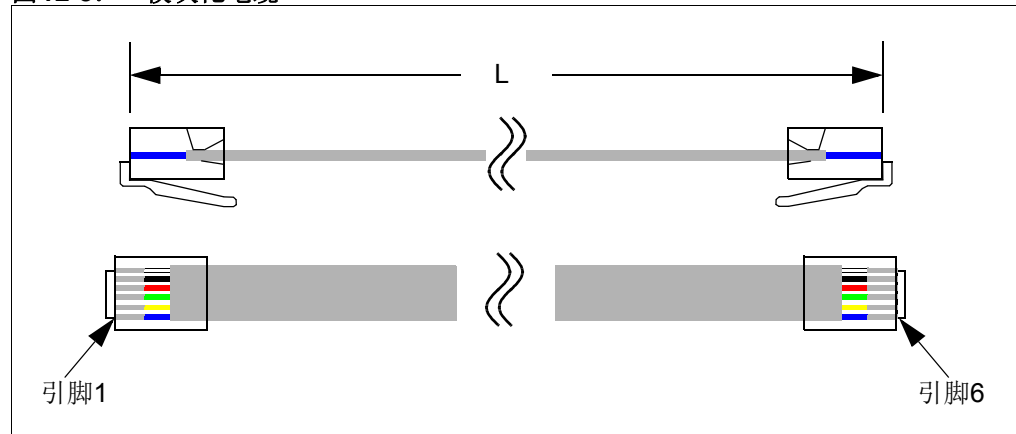
在标准通信中，模块化（ICSP）电缆用于连接仿真器和目标应用。此电缆及其连接器的规范如下文所述。

### 12.5.2.1 模块化电缆规范

制造商，部件编号——**Microchip Technology, 07-00024**

此电缆长度（L）为6英寸。不建议使用超过6英寸的模块化电缆，否则可能遇到通信问题。如果需要更长的电缆，可以考虑购买高性能工具包（AC244002）。

图12-3： 模块化电缆



## 12.5.2.2 模块化插头规范

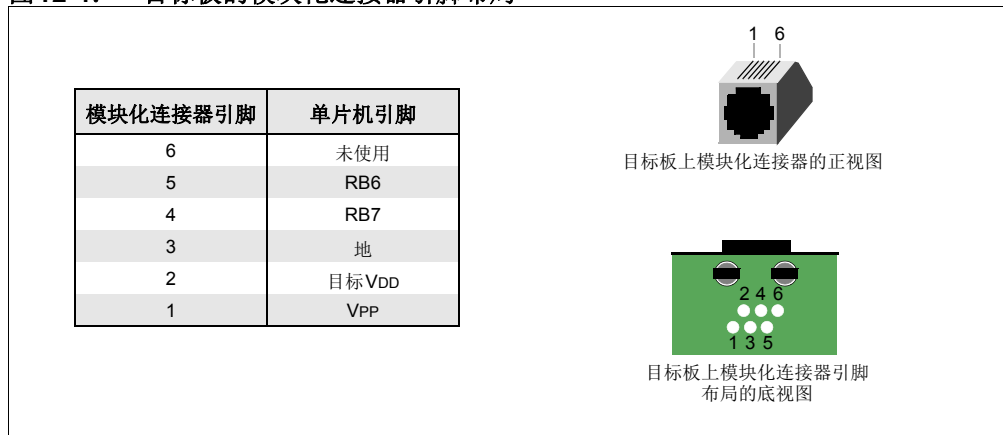
- 制造商，部件编号——AMP Incorporated, 5-554710-3
- 代理商，部件编号——Digi-Key, A9117ND

## 12.5.2.3 模块化连接器规范

- 制造商，部件编号——AMP Incorporated, 555165-1
- 代理商，部件编号——Digi-Key, A9031ND

下图介绍了应用中的模块化连接器引脚与单片机引脚的对应情况。这种配置可实现完整的ICD功能。

图12-4: 目标板的模块化连接器引脚布局



## 12.6 高速/LVDS通信硬件

关于高速/LVDS通信硬件的信息，请参见《MPLAB® REAL ICE™ 在线仿真器高性能工具包用户指南》（DS50002528A\_CN）。

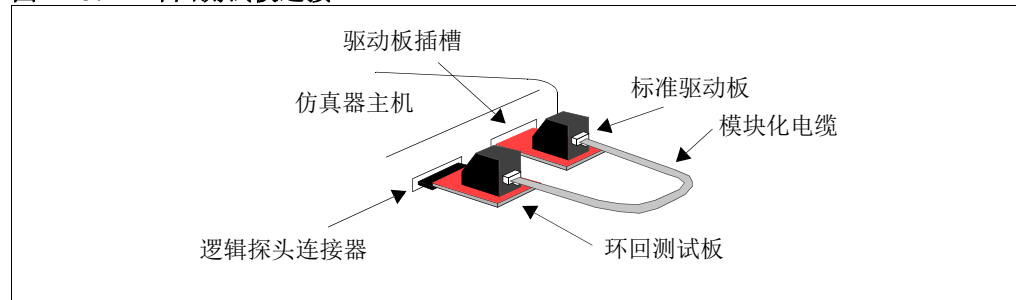
## 12.7 环回测试板

该板（AC244003）随仿真器一起提供，但也可单独购买，可用于验证仿真器是否正常工作。要使用该板：

1. 将仿真器从目标板和PC断开。
2. 若尚未安装标准驱动板，则插入标准驱动板。
3. 将环回测试板插入主机的逻辑探头连接器。
4. 使用模块化电缆将环回测试板连接到标准驱动板。
5. 将仿真器重新连接到计算机。
6. 启动MPLAB X IDE。确保现有项目均已关闭。
7. 选择 *Debug>Run Debugger/Programmer Self Test*（调试>运行调试器/编程器自检），然后选择想要测试的特定“REAL ICE”并单击 **OK**。
8. 确认环回测试板和电缆已连接，然后单击 **Yes**（是）继续。
9. 在仿真器的 **Output** 窗口中查看自检结果。
10. 仿真器通过自检后，将环回测试板与仿真器断开。

MPLAB IDE 检测并运行完整的环回测试，然后指示一个状态（通过/失败）。环回测试板的检测工作原理是：在逻辑探头连接器上对EXT0施加一个短脉冲，然后在EXT7上检测该脉冲（第12.4.4节“逻辑探头/外部触发接口”）。检测到该板之后，仿真器会将激励应用到时钟/数据线和VPP线，并从逻辑探头连接器接口中读回序列，从而确认信号电平以及连接器接口的连接是否正确。

**图12-5： 环回测试板连接**



## 12.8 目标板注意事项

应根据所选器件（1.6V-5.5V）和应用的要求为目标板供电。

**注：** 不能通过仿真器为目标板供电。

仿真器可检测目标板电源。VDD\_TGT上的负载为10 kΩ。

根据仿真器与目标板之间通信的类型，目标板电路有一些注意事项：

- 第3.5.3节“目标板连接电路”
- 第3.5.4节“目标电路设计注意事项”

注:



---

---

## 附录A 版本历史

---

---

### 版本A (2013年1月)

- 本文档的初始版本。

### 版本B (2014年5月)

- 修复或删除了所有MPLAB IDE v8链接或参考。
- 更新了整篇文档中的警告框以匹配更新的标准。
- **前言**“推荐读物”分为两个部分（“仿真器”和“附件”），并针对新文档进行了更新。
- **第2章**“器件与功能支持——仿真器和调试器”——表数据可以移植到DTS。
- **第3.2节**“工具比较”——更新了比较表。
- **第3.3节**“操作概述”——更新了表以阐明速度。
- **第3.5.3节**“目标板连接电路”——更改了上拉值。
- **第3.6.3节**“I/O端口跟踪连接（逻辑端口）”——增加了允许的PORTx配置。
- **第4.3节**“固件升级”——增加了小节。
- **第4.4节**“仿真器功能”——替换了之前的第4.3节和第4.4节。
- **第5.4.2节**“硬件断点或软件断点选择”——更新了表。
- **第5.5节**“插装跟踪”——许多器件的全部系列都支持插装跟踪。占位符仍保留在“特定调试功能”章节中。
- **第6.2节**“数据捕捉和运行时观察”和**第8.2节**“数据捕捉和运行时观察”——定义了数据捕捉和运行时观察。观察变量需为本机大小。
- 删除运行时观察的时钟要求：**第6.2.3节**“运行时观察和Watches窗口”和**第8.2节**“数据捕捉和运行时观察”。
- 增加了**第6.5节**“跳转跟踪——仅限EP器件”和**第6.8节**“函数级别性能分析”。
- **第8章**“特定调试功能：32位器件”——将PIC32MX更新为PIC32以包括所有32位器件。
- **第8.3.3节**“设置和使用跟踪”——增加了图示。
- 更新了**第7章**“故障诊断首要步骤”和**第8章**“常见问题解答（FAQ）”。
- **第4部分**——“软件和硬件参考”——“参考”一节名称更改为“软件和硬件参考”。
- **第5部分**——“仿真器附件”——从“参考”一节中删除了“仿真器附件”一章，以在此部分中创建章节。为可用的仿真器附件增加了附件章节。

# 适用于 MPLAB X IDE 的仿真器用户指南

---

## 版本 C（2015 年 5 月）

- 将对 PIC32MX 的引用替换为 PIC32（以包括 PIC32MZ 器件）。
- 根据需要更新了屏幕截图。
- **第 1.4 节“仿真器工具包组件”**——增加了关于额外硬件的信息。
- **第 3.2 节“工具比较”**——更新了“从目标板灌入 VDD 的电流”。
- **第 3.5.3 节“目标板连接电路”**——更新了上拉电阻值。
- **第 6 章“特定调试功能”**——将第 6 章“特定调试功能：8 位和 16 位器件”和第 7 章“特定调试功能：32 位器件”合并为一章。许多功能现在适用于所有器件。
- **第 6.6 节“PC 采样——仅限 8 位和 16 位 MCU”、第 6.7 节“PC 性能分析——仅限 32 位 MCU”和第 6.8 节“函数级别性能分析”**——针对代码分析插件的使用进行了更新。
- **第 11.3.1 节“要编程的存储器”**——增加了选择“Erase All Before Program”对存储器的影响。
- **第 11.3.5 节“跟踪和性能分析”**——针对代码分析插件的使用进行了更新。
- **第 11.3.8 节“外部触发信号”**增加了触发信号详细信息。
- **第 12.4.4 节“逻辑探头 / 外部触发接口”**——增加了关于触发信号输入的信息。
- **第 12.5.2.1 节“模块化电缆规范”**——固定电缆长度说明。
- **第 15 章“具有跟踪功能的 PIC32 PIM”**——更新了具有跟踪功能的 PIM 列表。
- **第 17 章“MPLAB REAL ICE 隔离器单元”**——重新排列了各节和文本以便更好地理解内容。
- **第 18 章“MPLAB REAL ICE JTAG 适配器（PIC32）”**——增加了建议的 JTAG 适配器板并将部分原理图指定为“可选”。

## 版本 D（2015 年 5 月）

- **第 3.3 节“操作概述”**——根据目前可在某些 PIC32 MCU 上使用插装跟踪，更新了文本和表。
- **第 4.3 节“固件升级”**——更改了固件升级过程；固件版本现在与 MPLAB X IDE 版本同步。
- **第 6.8 节“函数级别性能分析”**——更新了图 6-12，使之显示为选中“Time Stamp”（时间戳）。

## 版本 E（2016 年 12 月）

- 删除了“附件”章节，为每种附件创建了单独的文档。在整篇文档中根据需要引用这些新文档。
- 更新了 MPLAB X IDE v3.25 及更高版本的图形（盾牌图案）。
- **第 11.3.1 节“要编程的存储器”**——更新了关于要编程和 / 或保护的存储器范围的信息。
- **“支持”**——增加了关于 MySoftware 帐户的信息。

---

---

## 支持

---

---

### 简介

关于支持问题，请参考本章提供的内容。

- [保修登记](#)
- [myMICROCHIP 个性化通知客户服务](#)
- [MySoftware 帐户](#)
- [Microchip 网站](#)
- [Microchip 论坛](#)
- [客户支持](#)
- [关于 Microchip Technology](#)

### 保修登记

注册您的开发工具，这将使您可以接收到新产品更新。可在 [Microchip 网站](#) 上获得临时软件版本：

<http://www.microchipdirect.com>

### myMICROCHIP 个性化通知客户服务

Microchip 的个人通知客户服务有助于客户了解关于所关注 Microchip 产品的最新信息。注册客户可在指定产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

请访问 <http://www.microchip.com/pcn> 开始注册过程，并选择接收个性化通知的首选项。页面上提供了 FAQ 和注册详细信息，可以通过选择上面的链接打开。

在选择首选项时，选择“Development Systems”（开发系统）将会在列表中填入可用的开发工具。下面列出了主要的工具类别：

- **编译器**——Microchip C 编译器、汇编器、链接器及其他语言工具的最新信息，包括所有 MPLAB C 编译器、所有 MPLAB 汇编器（包括 MPASM™ 汇编器）、所有 MPLAB 链接器（包括 MPLINK™ 目标链接器）以及所有 MPLAB 库管理器（包括 MPLIB™ 目标库管理器）。
- **仿真器**——Microchip 在线仿真器的最新信息，包括 MPLAB REAL ICE™ 在线仿真器。
- **在线调试器**——Microchip 在线调试器的最新信息，包括 PICKit™ 2、PICKit 3 和 MPLAB ICD 3 在线调试器。
- **MPLAB® IDE**——关于开发系统工具的 Windows® 集成开发环境 Microchip MPLAB IDE 的最新信息，主要针对 MPLAB IDE、MPLAB IDE 项目管理器、MPLAB 编辑器和 MPLAB SIM 模拟器以及一般编辑和调试功能。

# 适用于 MPLAB X IDE 的仿真器用户指南

---

- **编程器** —— Microchip 编程器的最新信息。包括器件（生产）编程器 MPLAB REAL ICE 在线仿真器、MPLAB ICD 3 在线调试器和 MPLAB PM3，以及开发（非生产）编程器 PICKit 2 和 3。
- **入门 / 演示板** —— 它们包括 MPLAB 入门工具包电路板、PICDEM 演示板以及各种其他评估板。

## MySoftware 帐户

通过您的 MySoftware 帐户来管理 Microchip 软件下载、密钥和许可。通过以下方式访问您的帐户：

- MPLAB X IDE 桌面， My MPLAB X IDE（我的 MPLAB X IDE）选项卡，“Microchip Login”（Microchip 登录）。登录后，单击“View mySoftware Account”（查看 mySoftware 帐户）。
- microchipDirect，选择 My Account（我的帐户）>My Software Products（我的软件产品）。登录到您的帐户。
- MySoftware URL 为：<https://www.microchip.com/mySoftware>。

## Microchip 网站

Microchip 网站（<http://www.microchip.com>）为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的互联网浏览器即可访问。网站提供以下信息：

- **产品支持** —— 数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及归档软件
- **一般技术支持** —— 常见问题解答（FAQ）、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务** —— 产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

## Microchip 论坛

Microchip 网站（<http://www.microchip.com/forums>）为客户提供额外的在线支持。当前提供的论坛包含以下主题：

- 开发工具
- 8 位 PIC MCU
- 16 位 PIC MCU
- 32 位 PIC MCU

## 客户支持

Microchip产品的用户可通过以下渠道获得帮助:

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。请访问我们的网站获取完整、最新的销售办事处列表。

也可通过 <http://support.microchip.com> 获得网上技术支持。

文档错误或意见可通过电子邮件发送至 [CTRC@microchip.com](mailto:CTRC@microchip.com)。

## 关于 MICROCHIP TECHNOLOGY

Microchip Technology Inc. 是全球领先的单片机和模拟半导体供应商, 为全球数以千计的各种消费类应用提供低风险的产品开发、更低的系统总成本和更快的上市时间。公司总部位于亚利桑那州 Chandler 市, Microchip 提供出色的技术支持、可靠的产品和卓越的质量。

电话: (480) 792-7200

传真: (480) 792-7277

**myMicrochip:** <http://www.microchip.com/pcn>

**MySoftware:** <https://www.microchip.com/mySoftware>

网站: <http://www.microchip.com>

论坛: <http://www.microchip.com/forums>

支持: <http://support.microchip.com>

注:

---

---

## 术语表

---

---

### A

#### ANSI

美国国家标准学会，是美国负责制订和批准标准的组织。

#### ASCII

美国信息交换标准码是使用7个二进制位来表示每个字符的字符集编码。它包括大写和小写字母、数字、符号以及控制字符。

### B

#### 八进制 (Octal)

使用数字0-7，以8为基数的计数体制。最右边的位表示1的倍数，右边第二位表示8的倍数，右边第三位表示 $8^2 = 64$ 的倍数，以此类推。

#### 本机数据大小 (Native Data Size)

对于本机跟踪，Watches (观察) 窗口中使用的变量长度必须与所选器件的数据存储器长度相同：对于PIC18器件，为字节长度；对于16位器件，为字长度。

#### 编译 (Build)

编译并链接一个应用程序的所有源文件。

#### 编译堆栈 (Compiled Stack)

编译器管理的存储区域，静态分配变量的空间。当在目标器件上无法高效实现软件或硬件堆栈时，编译堆栈将替代它们。

#### 编译器 (Compiler)

将用高级语言编写的源文件翻译成机器码的程序。

#### 标识符 (Identifier)

函数名或变量名。

#### 表达式 (Expression)

被算术或逻辑操作符分隔开的常量和/或符号的组合。

### C

#### C/C++

C语言是具有表达式简练、现代控制流程和数据结构，以及操作符丰富等特点的通用编程语言。C++是C语言的面向对象版本。

#### COFF

公共目标文件格式。此格式的目标文件包含机器码、调试信息和其他信息。

## CPU

参见中央处理单元。

## 操作码 (Opcode)

操作码。请参见助记符。

## 插件 (Plug-in)

MPLAB IDE/MPLAB X IDE 具有内置组件和插件模块，用于针对各种软件和硬件工具来配置系统。可在 Tools (工具) 菜单下找到几个插件工具。

## 程序存储器 (Program Memory)

MPLAB IDE/MPLAB X IDE——器件中存储指令的存储区。也指仿真器或软件模拟器中包含已下载的目标应用固件的存储器。

MPLAB XC16 汇编器/编译器——器件中存储指令的存储区。

## 程序计数器 (Program Counter)

包含正在执行指令的地址的存储单元。

## 程序计数器单元 (Program Counter Unit)

MPLAB XC16 汇编器——程序存储器布局的概念化表示。对于每个指令字，程序计数器递增2。在可执行段中，2个程序计数器单元等同于3个字节。在只读段中，2个程序计数器单元等同于2个字节。

## 持久性数据 (Persistent Data)

永不清除或初始化的数据。其目的是使应用程序可以在器件复位时保存数据。

## 重定位 (Relocation)

链接器执行的过程，其中会将绝对地址分配给可重定位段，并且可重定位段中的所有符号都会更新为相应的最新地址。

## 触发输出 (Trigger Output)

指可在任意地址或地址范围产生的仿真器输出信号，与跟踪和断点的设置无关。可设置任意个触发输出点。

## 次数计数器 (Pass Counter)

每次发生一个事件（如执行特定地址处的一条指令）时都会递减1的计数器。当次数计数器的值为零时，事件满足。可将次数计数器分配给中断和跟踪逻辑，以及在 complex trigger（复杂触发）对话框中的任何连续事件。

## 存储类别 (Storage Class)

确定与已标识对象相关的存储器的使用寿命。

## 存储器模型 (Memory Model)

对于C编译器，为应用程序可使用的存储区的表示。对于PIC18 C编译器，为一种描述，指定指向程序存储器的指针的长度。

## 存储限定符 (Storage Qualifier)

指示被声明的对象的特殊属性（例如 const）。

## 错误/错误文件 (Error/Error File)

错误报告使程序不能继续处理的问题。而且，当问题比较明显时，错误还尽可能标识出源文件名和行号。错误文件包含由语言工具生成的错误消息和诊断信息。



**D****DWARF**

使用任意记录格式调试。DWARF是用于ELF文件的调试信息格式。

**带序列号的快速批量编程 (Serialized Quick Turn Programming)**

序列化允许您利用器件编程器将序列号编程到每个单片机器件中。该序列号可用作入口代码、密码或ID编号。

**单步进入 (Step Into)**

这一命令与Single Step相同。Step Into (与Step Over相对) 在CALL指令后, 单步执行子程序。

**单步跳出 (Step Out)**

Step Out允许跳出当前正在执行的子程序。此命令执行子程序中剩下的代码, 然后在子程序的返回地址处停止执行。

**单步跳过 (Step Over)**

Step Over允许单步调试代码而不单步执行子程序。当单步跳过一条CALL指令时, 下一个断点将设置在CALL指令后的下一条指令处。如果由于某种原因, 子程序陷入无限循环或不正确返回, 下一个断点将永远执行不到。除对CALL指令处理不同外, Step Over命令和Single Step相同。

**单步执行 (Single Step)**

这一命令单步执行代码, 一次执行一条指令。执行每条指令后, MPLAB IDE/MPLAB X IDE更新寄存器窗口、观察变量及状态显示, 使您可分析和调试指令。也可单步执行C编译器源代码, 但不是每次执行一条指令, MPLAB IDE/MPLAB X IDE将执行一行高级C语句生成的所有汇编指令。

**单片机 (Microcontroller)**

高度集成的芯片, 它包括CPU、RAM、程序存储器、I/O端口和定时器。

**单片机模式 (Microcontroller Mode)**

PIC18单片机的一种程序存储器配置。在单片机模式下, 仅允许内部执行。因此, 在单片机模式下仅可使用片内程序存储器。

**导出 (Export)**

以标准的格式将数据发送出MPLAB IDE/MPLAB X IDE。

**导入 (Import)**

从外面的源 (如hex文件) 将数据送入MPLAB IDE/MPLAB X IDE。

**递归 (Recursion)**

已定义的函数或宏可调用自身的概念。编写递归宏时要特别小心; 当递归没有出口时容易陷入无限循环。

**递归调用 (Recursive Call)**

直接或间接调用自身的函数。

**地址 (Address)**

标识存储器中位置的值。

## 段 (Section)

OCG psect 的 GCC 等效形式。被链接器视为整体的代码或数据块。

## 段属性 (Section Attribute)

属于段的 GCC 特性 (如, `access` 段)。

## 断点 (Breakpoint)

硬件断点: 一个事件, 执行这种事件会导致暂停。

软件断点: 一个地址, 固件会在这个地址处暂停执行。通常由特殊的 `break` 指令获得。

## 断点逻辑与 (ANDed Breakpoint)

为中断设置逻辑与条件, 即只有断点 1 和断点 2 同时出现时, 才会暂停程序。只有数据断点和程序存储器断点同时出现时, 才会完成此操作。

## 堆 (Heap)

用于动态存储器分配的存储区, 其中的存储器块按运行时确定的任意顺序进行分配和释放。

## 堆栈, 编译 (Stack, Compiled)

编译器管理和分配的存储区域, 静态分配变量的空间。当在目标器件上无法高效实现软件堆栈时, 编译堆栈将替代软件堆栈。它阻止可重入。

## 堆栈, 软件 (Stack, Software)

应用程序用来存储返回地址、函数参数和局部变量的存储区。该存储区由程序中的指令在运行时动态分配。它支持可重入函数调用。

## 堆栈, 硬件 (Stack, Hardware)

调用函数时, PIC 单片机中存储返回地址的存储单元。

## E

### EEPROM

电可擦除的可编程只读存储器。一种可电擦除的特殊类型 PROM。一次写或擦除一个字节。EEPROM 即使电源关闭时也能保留内容。

### ELF

可执行链接格式。这种格式的目标文件包含机器码。调试和其他信息使用 DWARF 指定。ELF/DWARF 可提供优于 COFF 的优化代码调试。

### Epilogue

编译器生成代码的一部分, 负责释放堆栈空间、恢复寄存器, 以及执行运行时模型中指定的任何其他特定于机器的要求。此代码在给定函数的任何用户代码之后、在函数返回之前执行。

### EPROM

可擦除的可编程只读存储器。通常通过紫外线照射来擦除的可编程只读存储器。

### 二进制 (Binary)

使用数字 0 和 1, 以 2 为基数的计数体制。最右边的位表示 1 的倍数, 右边第二位表示 2 的倍数, 右边第三位表示  $2^2 = 4$  的倍数, 以此类推。

**F****FNOP**

强制空操作。强制NOP周期是双周期指令的第二个周期。由于PIC单片机的架构是流水线型，在执行当前指令的同时预取物理地址空间中的下一条指令。但是，如果当前指令改变了程序计数器，那么这条预取的指令会被忽略，导致一个强制NOP周期。

**Free-Standing**

一种接受任何不使用复杂类型的严格符合程序的实现，而且在这种实现中，对库条款（ANSI 89标准条款第7条）中规定的特性的使用，仅限于标准头文件<float.h>、<iso646.h>、<limits.h>、<stdarg.h>、<stdbool.h>、<stddef.h> 和<stdint.h>的内容。

**访问入口点（Access Entry Point）**

访问入口点提供了一种方法，可跨段将控制转移到某个可能未在链接时定义的函数。它们支持独立链接引导应用程序段和安全应用程序段。

**仿真/仿真器（Emulation/Emulator）**

请参见ICE/ICD。

**非扩展模式（Non-Extended Mode）（PIC18 MCU）**

在非扩展模式下，编译器不使用扩展指令，也不使用立即数偏移量变址寻址。

**非实时（Non Real-Time）**

指处理器处于断点或正在单步执行指令或者MPLAB IDE/MPLAB X IDE正运行在软件模拟器模式下。

**非易失性存储器（Non-Volatile Storage）**

关断电源后，能保存自身内容的存储器件。

**符号（Symbol）**

符号是描述组成程序的各个部分的通用机制。这些部分包括函数名、变量名、段名、文件名、结构/枚举/联合标记名等。MPLAB IDE/MPLAB X IDE中的符号主要指变量名、函数名和汇编标号。链接后符号的值就是其在存储器中的值。

**符号，绝对（Symbol, Absolute）**

表示一个立即数的值，例如通过汇编.equ伪指令指定的定义。

**G****GPR**

通用寄存器。器件数据存储器（RAM）的一部分，作为一般用途。

**概要文件（Profile）**

对于MPLAB SIM软件模拟器，该文件是对寄存器执行的激励的汇总列表。

**高级语言（High Level Language）**

编写程序的语言，它比汇编语言更不依赖于具体的处理器。

## 跟踪 (Trace)

记录程序执行的仿真器或软件模拟器功能。仿真器将程序执行记录到其跟踪缓冲区内，并可上载到 MPLAB IDE/MPLAB X IDE 的跟踪窗口中。

## 跟踪存储区 (Trace Memory)

跟踪存储区包含在仿真器内部。跟踪存储区有时称为跟踪缓冲区。

## 跟踪宏 (Trace Macro)

一个通过仿真器数据来提供跟踪信息的宏。由于该宏属于软件跟踪，所以必须将它添加到代码中、必须重新编译或重新汇编代码，并且必须使用该代码对目标器件进行编程，之后跟踪才会工作。

## 工具栏 (Tool Bar)

一行或一系列图标，单击这些图标时将执行 MPLAB IDE/MPLAB X IDE 功能。

## 工作簿 (Workbook)

对于 MPLAB SIM 模拟器，是一种用于产生 SCL 激励的设置。

## 观察变量 (Watch Variable)

调试会话期间可在 Watches 窗口中监控的变量。

## 归档/归档器 (Archive/Archiver)

归档/库是可重定位目标模块的集合。由将多个源文件汇编/编译为目标文件，然后使用归档器/库管理器将目标文件组合为一个归档/库文件生成。可将归档/库与目标模块和其他归档/库链接，生成可执行代码。

## 国际标准化组织 (International Organization for Standardization)

制订多行业和技术（包括计算和通讯）方面的标准的一个组织。也称为 ISO。

## 过滤器 (Filter)

通过选择确定在跟踪显示或数据文件中包含/排除哪些数据。

## H

### Hex 代码/Hex 文件 (Hex Code/Hex File)

Hex 代码是以十六进制格式代码存储的可执行指令。Hex 代码包含在 hex 文件中。

### 宏 (Macro)

宏指令。宏指令是以缩写形式表示指令序列的指令。

### 宏伪指令 (Macro Directive)

控制宏定义体中执行和数据分配的伪指令。

### 环回测试板 (Loopback Test Board)

用于测试 MPLAB REAL ICE 在线仿真器的功能。

### 环境 (Environment)

MPLAB PM3 —— 包含关于如何编程器件的文件的文件夹。该文件夹可转移到 SD/MMC 卡。

### 汇编语言/汇编器 (Assembly/Assembler)

汇编语言是以符号形式描述二进制机器码的编程语言。汇编器是将汇编源代码翻译成机器码的语言工具。

## I

**ICE/ICD**

在线仿真器/在线调试器：用于对目标器件进行调试和编程的硬件工具。仿真器具有比调试器更多的功能，例如跟踪。

在线仿真/在线调试：使用在线仿真器或调试器进行仿真或调试的行为。

-ICE/-ICD：带有在线仿真或调试电路的器件（MCU或DSC）。该器件总是安装在调试头板上，并用于通过在线仿真器或调试器进行调试。

**ICSP**

在线串行编程。一种通过使用串行通信并使用最少数量的器件引脚对Microchip嵌入式器件编程的方法。

**IDE**

集成开发环境，如MPLAB IDE/MPLAB X IDE。

**IEEE**

电气与电子工程师协会。

## J

**机器码（Machine Code）**

处理器实际读和解释的计算机程序的表示。二进制机器码程序由一系列机器指令（可能还包含数据）组成。某个特定处理器的所有可用指令的集合称为它的“指令集”。

**机器语言（Machine Language）**

特定中央处理单元的指令集，不需翻译即可用于处理器。

**基数（Radix）**

数字基，十六进制或十进制，用于指定一个地址。

**激励（Stimulus）**

软件模拟器的输入（即为模拟对外部信号的响应而生成的数据）。通常数据采用文本文件中一系列动作的形式。激励可以是异步的，同步的（引脚），时钟激励或寄存器激励。

**交叉引用文件（Cross Reference File）**

引用符号表的一个文件及引用符号的文件的列表。如果定义了符号，则在列出的第一个文件中包含符号的定义。其他文件包含对符号的引用。

**校准存储区（Calibration Memory）**

用来保存PIC单片机内置RC振荡器或其他器件外设的校准值的特殊功能寄存器（或通用寄存器）。

**节点（Node）**

MPLAB IDE/MPLAB X IDE 项目组件。

**警告（Warning）**

MPLAB IDE/MPLAB X IDE——提供警告，警告您发生了可能导致器件、软件文件或设备物理损坏的情况。

MPLAB XC16汇编器/编译器——警告报告可能发生问题的情况，但并不暂停处理。

## 静态RAM或SRAM (Static RAM or SRAM)

静态随机访问存储器。目标板上可读/写且无需经常刷新的程序存储器。

## 局部标号 (Local Label)

用LOCAL伪指令在一个宏内部定义的局部标号。这些标号特定于宏实例化的一个给定实例。也就是说，声明为local的符号和标号在遇到ENDM宏后不再可访问。

## 绝对变量/函数 (Absolute Variable/Function)

使用OCG编译器的@ *address*语法放置在绝对地址的变量或函数。

## 绝对段 (Absolute Section)

具有链接器不能更改的固定（绝对）地址的GCC编译器段。

## K

## 看门狗定时器 (Watchdog Timer, WDT)

PIC单片机上在一段可选的时间长度后复位处理器的定时器。使用配置位来使能、禁止和设置WDT。

## 可重定位 (Relocatable)

地址未分配到存储器中固定位置的对象。

## 可重定位段 (Relocatable Section)

MPLAB XC16汇编器——地址不固定（绝对）的段。链接器通过一个称为重定位的过程来为可重定位段分配地址。

## 可重入函数 (Reentrant)

可以有多个同时运行的实例的函数。下面两种情况下可能发生函数重入：直接或间接递归调用函数；或者在中断处理期间执行此函数。

## 可执行代码 (Executable Code)

可装入来执行的软件。

## 控制伪指令 (Control Directive)

汇编语言代码中根据汇编时指定表达式的值包含或忽略代码的伪指令。

## 库/库管理器 (Library/Librarian)

参见归档/归档器。

## 快速存取存储区 (Access Memory)

仅限PIC18——PIC18器件中的一些特殊寄存器，对这些寄存器的访问与存储区选择寄存器（Bank Select Register, BSR）的设置无关。

## 扩展单片机模式 (Extended Microcontroller Mode)

在扩展单片机模式中，既可使用片内程序存储器，也可使用外部存储器。如果程序存储器地址大于PIC18器件的内部存储空间，执行自动切换到外部存储器。

## 扩展模式 (Extended Mode) (PIC18 MCU)

在扩展模式下，编译器将使用扩展指令（即ADDFSR、ADDULNK、CALLW、MOVSF、MOVSS、PUSHL、SUBFSR和SUBULNK）以及立即数变址寻址。

---

---

## L

### LVDS

低压差分信号传输。一种通过铜线进行高速（每秒千兆位）数据传输的低噪声、低功耗和低幅值方法。

对于标准 I/O 信号，数据存储取决于实际电压大小。电压大小可能受到传输线长度的影响（线路越长，电阻就越高，这会使电压下降）。但对于 LVDS，数据存储仅通过正负电压值区分，而不是实际电压大小。因此，数据可以传输更长的线路距离，同时保持干净、一致的数据流。

来源: <http://www.webopedia.com/TERM/L/LVDS.html>

### 链接描述文件 (Linker Script File)

链接描述文件是链接器的命令文件。它们定义链接器选项并描述目标平台上的可用存储器。

### 链接器 (Linker)

把目标文件和库文件组合起来生成可执行代码并解析一个模块对另外一个模块引用的语言工具。

### 列表伪指令 (Listing Directive)

列表伪指令是控制汇编器列表文件格式的伪指令。它们允许指定标题、分页及其他列表控制。

### 列表文件 (Listing File)

列表文件是列出为每条 C 源语句生成的机器码，源文件中遇到的汇编指令、汇编器伪指令或宏的 ASCII 文本文件。

### 逻辑探头 (Logic Probe)

Microchip 的某些仿真器最多可连接 14 个逻辑探头。逻辑探头提供外部跟踪输入、触发输出信号、+5V 和公共接地端。

## M

### Makefile

包含用于 Make 项目的指令的文件。使用该文件可以在 MPLAB IDE/MPLAB X IDE 之外使用 make Make 项目。

### Make 项目 (Make Project)

重新编译应用程序的命令，仅编译自上次编译完成后更改了的源文件。

### MCU

单片机。microcontroller 的缩写形式。也写作 uC。

### MPASM™ 汇编器 (MPASM™ Assembler)

Microchip Technology 用于 PIC 单片机、KeeLoq® 器件和 Microchip 存储器件的可重定位宏汇编器。

## **MPLAB ICD**

Microchip的在线调试器，与MPLAB IDE/MPLAB X IDE配合使用。请参见ICE/ICD。

## **MPLAB IDE/MPLAB X IDE**

Microchip的集成开发环境。MPLAB IDE/MPLAB X IDE附带有编辑器、项目管理器和软件模拟器。

## **MPLAB PM3**

Microchip的器件编程器。用于对PIC18单片机和dsPIC数字信号控制器进行编程。可与MPLAB IDE/MPLAB X IDE配合使用或独立使用。代替PRO MATE II。

## **MPLAB REAL ICE™在线仿真器（MPLAB REAL ICE™ In-Circuit Emulator）**

Microchip的新一代在线仿真器，与MPLAB IDE/MPLAB X IDE配合使用。请参见ICE/ICD。

## **MPLAB SIM**

Microchip的软件模拟器，可与MPLAB IDE/MPLAB X IDE配合使用，支持PIC MCU和dsPIC DSC器件。

## **MPLIB™目标库管理器（MPLIB™ Object Librarian）**

Microchip的库管理器，可与MPLAB IDE/MPLAB X IDE配合使用。MPLAB库管理器是用于将由MPASM汇编器（mpasm或mpasmwin v2.0）或MPLAB C18 C编译器生成的COFF目标模块组合成库文件的目标库管理器。

## **MPLINK™目标链接器（MPLINK™ Object Linker）**

MPLINK链接器是Microchip MPASM汇编器和Microchip C18 C编译器的目标链接器。MPLINK链接器还可以与Microchip MPLIB库管理器配合使用。MPLINK链接器设计为在MPLAB IDE/MPLAB X IDE中使用，但是并不一定要在MPLAB IDE/MPLAB X IDE中使用它。

## **MRU**

最近使用过的。指可从MPLAB IDE/MPLAB X IDE主下拉菜单中选择的文件和窗口。

## **命令行接口（Command Line Interface）**

程序及其用户之间的一种完全基于文本输入和输出的通信手段。

## **模块（Module）**

执行预处理器伪指令后，源文件的预处理输出。也称为编译单元。

## **模板（Template）**

为以后插入自己的文件中使用而创建的文本行。MPLAB编辑器将模板存储到模板文件中。

## **目标（Target）**

指用户硬件。

## **目标板（Target Board）**

构成目标应用的电路和可编程器件。



**目标处理器 (Target Processor)**

目标应用板上的单片机。

**目标代码/目标文件 (Object Code/Object File)**

目标代码是由汇编器或编译器生成的机器码。目标文件为包含机器码（可能还有调试信息）的文件。它可以直接执行，或为可重定位的，需要与其他目标文件（如库文件）链接，以生成完整的可执行程序。

**目标文件伪指令 (Object File Directives)**

仅当创建目标文件时使用的伪指令。

**目标应用程序 (Target Application)**

目标板上的软件。

**N****NOP**

空操作。除了让程序计数器加1外，没有其他作用的指令。

**内部链接 (Internal Linkage)**

如果不能从定义函数或变量的模块外部访问它们，则这样的函数或变量具有内部链接。

**匿名结构 (Anonymous Structure)**

MPLAB XC16 C编译器——未命名的结构。

MPLAB XC8 C编译器——属于C联合成员的未命名结构。匿名结构的成员可以像外围联合的成员一样进行访问。例如，在以下代码中，hi和lo是联合caster中的匿名结构的成员。

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

**O****OTP**

可一次编程。非窗口封装的EPROM器件。由于EPROM需要紫外线照射来擦除其存储内容，因此只有窗口片是可擦除的。

**P****PC**

个人计算机或程序计数器。

**PC主机 (PC Host)**

任何运行支持的Windows操作系统的PC。

**PIC MCU**

PIC单片机 (MCU) 指Microchip的所有单片机系列。

## **PICkit 2和3**

Microchip 的器件开发编程器，通过 Debug Express 实现调试功能。要了解支持哪些器件，请参见各工具的自述文件。

## **Pragma 伪指令 (Pragma)**

对特定编译器有意义的伪指令。通常一条 pragma 伪指令用于向编译器传达实现定义的信息。

## **Psect**

GCC 段的 OCG 等效形式，程序段的简称。被链接器视为整体的代码或数据块。

## **PWM 信号 (PWM Signal)**

脉冲宽度调制信号。某些 PIC MCU 器件具有 PWM 外设。

## **跑表 (Stopwatch)**

测量执行周期的计数器。

## **配置位 (Configuration Bit)**

可对其编程来设置 PIC MCU 或 dsPIC DSC 工作模式的专用位。配置位可或不可再编程。

## **片外存储器 (Off-Chip Memory)**

片外存储器指 PIC18 器件的一种存储器选择，这种情况下存储器可位于目标板上，或所有程序存储器都由仿真器提供。通过 *Options > Development Mode* (选项 > 开发模式) 访问 **Memory** (存储器) 选项卡可打开 Off-Chip Memory selection (片外存储器选择) 对话框。

## **Q**

### **弃用功能 (Deprecated Feature)**

由于历史原因仍然支持但最终将逐步淘汰且不再使用的功能。

### **器件编程器 (Device Programmer)**

用于对电可编程半导体器件 (如单片机) 进行编程的工具。

### **嵌套深度 (Nesting Depth)**

一个宏可以包含其他宏的最大层数。

### **清除 (Clean)**

清除操作删除活动项目的所有中间项目文件，例如目标文件、hex 文件和调试文件。项目编译时会通过其他文件重新创建这些文件。

### **情形 (Scenario)**

对于 MPLAB SIM 软件模拟器，是一种用于激励控制的特定设置。

**R****RAM**

随机访问存储器（数据存储器）。可以任意顺序访问其中信息的存储器。

**ROM**

只读存储器（程序存储器）。不能修改的存储器。

**软件模拟器（Simulator）**

模拟器件操作的软件程序。

**S****Shell**

MPASM 汇编器 shell 是宏汇编器的提示性输入接口。有两个 MPASM 汇编器 shell：一个针对 DOS 版本，一个针对 Windows 操作系统版本。

**Skew**

不同时间出现在处理器总线上与指令执行有关的信息。例如，执行前一条指令的过程中取指时，被执行的操作码出现在总线上；当实际执行该操作码时，源数据地址及其值以及目标数据地址出现在总线上，当执行下一条指令时，目标数据值出现在总线上。跟踪缓冲区一次捕捉总线上的这些信息。因此，跟踪缓冲区的一条记录将包含三条指令的执行信息。执行一条指令时，从一条信息到另一条信息的捕捉周期数称为 skew。

**Skid**

当使用硬件断点来暂停处理器时，在处理器暂停前可能再执行一条或多条额外的指令。在预期断点之后执行的额外指令数称为 skid。

**SQTP**

参见带序列号的快速批量编程。

**三字母词（Trigraph）**

由三个字符组成的序列，均以 ?? 开头，由 ISO C 定义用于替代单个字符。

**闪存（Flash）**

按块（而不是按字节）写或擦除数据的一种 EEPROM。

**上电复位仿真（Power-on-Reset Emulation）**

在应用刚上电时，将随机值写到数据 RAM 区中来模拟 RAM 中未初始化值的软件随机过程。

**上载（Upload）**

上载功能将数据从一个工具（如仿真器或编程器）传送到主机 PC，或将数据从目标板传送到仿真器。

## 生产编程器 (Production Programmer)

生产编程器是一种编程工具，其中设计了可对器件进行快速编程的资源。它具有在各种电压下进行编程的能力并完全符合编程规范。在生产环境中，应用电路需要在组上传送，时间是极其重要的，所以尽可能快地对器件编程至关重要。

## 十六进制 (Hexadecimal)

使用数字0-9以及字母A-F (或a-f)，以16为基数的计数体制。字母A-F表示 (十进制中) 值为10-15的十六进制数。最右边的位表示1的倍数，右侧第二位表示16的倍数，右侧第三位表示 $16^2 = 256$ 的倍数，以此类推。

## 实时 (Real Time)

当在线仿真器或调试器从暂停状态释放时，处理器将在实时模式下运行并且完全像正常芯片那样工作。在实时模式下，仿真器的实时跟踪缓冲区被使能并不断捕获所有选定的周期，同时所有中断逻辑也被使能。在在线仿真器或调试器中，处理器将实时执行，直到有效断点导致暂停，或者直到用户暂停仿真器。

对于软件模拟器，实时仅意味着单片机指令的执行速度与主机CPU模拟这些指令的速度相同。

## 事件 (Event)

对可能包含地址、数据、次数计数、外部输入、周期类型 (如取指和读/写) 及时间标记的总线周期的描述。事件用于描述触发、断点和中断。

## 书签 (Bookmark)

使用书签可以轻松定位到文件中的特定行。

在Editor (编辑器) 工具栏中，选择Toggle Bookmarks (切换书签) 添加/删除书签。单击该工具栏上的其他图标以移动到下一个书签或上一个书签。

## 舒缓 (Relaxation)

将某一指令转换为功能相同但大小较小的指令的过程。这对于节省代码长度非常有用。MPLAB XC16目前知道如何将CALL指令放宽 (relax) 为RCALL指令。当被调用的符号处于当前指令的 $\pm 32k$ 指令字范围内时，将会执行该操作。

## 属性 (Attribute)

C程序中变量或函数的GCC特性，用于描述特定于机器的性质。

## 属性, 段 (Attribute, Section)

段的GCC特性，如“可执行”、“只读”或“数据”，它们可在汇编器.section伪指令中指定为标志。

## 数据存储 (Data Memory)

在Microchip MCU和DSC器件中，数据存储 (RAM) 由通用寄存器 (General Purpose Register, GPR) 和特殊功能寄存器 (Special Function Register, SFR) 组成。某些器件还有EEPROM数据存储。

## 数据监视和控制界面 (Data Monitor and Control Interface, DMCI)

数据监视和控制界面或DMCI是MPLAB X IDE中的一个工具。此界面提供对项目应用程序变量的动态输入控制。可使用4个可动态分配的图形窗口中的任一个以图形形式查看应用程序生成的数据。

**数据伪指令 (Data Directive)**

数据伪指令是那些控制汇编器对程序存储器或数据存储器进行分配的指示性语句，它提供了用符号（即有意义的名称）引用数据项的方法。

**数字信号处理/数字信号处理器 (Digital Signal Processing/Digital Signal Processor)**

数字信号处理 (Digital signal processing, DSP) 是对数字信号（通常为已转换为数字形式（经过采样）的模拟信号（声音或图像））的计算机处理。数字信号处理器是设计为用于数字信号处理的微处理器。

**数字信号控制器 (Digital Signal Controller)**

数字信号控制器 (DSC) 是具有数字信号处理能力的单片机（即 Microchip 的 dsPIC DSC 器件）。

**顺序断点 (Sequenced Breakpoints)**

按顺序发生的断点。断点的执行顺序为从下到上；即序列中的最后一个断点最先发生。

**T****特殊功能寄存器 (Special Function Register, SFR)**

数据存储器 (RAM) 的一部分，专用于控制 I/O 处理器函数、I/O 状态、定时器或其他模式及外设的寄存器。

**条件编译 (Conditional Compilation)**

只有当预处理器伪指令指定的某个常量表达式为真时才编译程序段的操作。

**条件汇编 (Conditional Assembly)**

基于指定表达式在汇编时的值包含或忽略的汇编语言代码。

**调试/调试器 (Debug/Debugger)**

请参见 ICE/ICD。

**调试信息 (Debugging Information)**

编译器和汇编器选项，在选中时，它们将提供不同程度的信息来用于调试应用程序代码。关于选择调试选项的详细信息，请参见编译器或汇编器文档。

**U****USB**

通用串行总线。是计算机和外设之间通过电缆以双向串行传输方式通信的外部外设接口标准。USB 1.0/1.1 支持 12 Mbps 的数据传输速率。USB 2.0 支持最高 480 Mbps 的数据传输速率，通常称为高速 USB。

**V****Volatile**

阻止编译器应用会影响访问存储器中变量的方式的优化的变量限定符。

**W****Watches 窗口 (Watches Window)**

Watches 窗口包含在每次执行到断点时更新的观察变量的列表。

**外部 RAM (External RAM)**

芯片外的读/写存储器。

## 外部标号 (External Label)

有外部链接的标号。

## 外部符号 (External Symbol)

具有外部链接的标识符符号。这可能是一个引用或一个定义。

## 外部符号解析 (External Symbol Resolution)

链接器搜集所有输入模块的外部符号定义来解析所有外部符号引用的过程。没有相应定义的任何外部符号引用都会导致报告链接器错误。

## 外部链接 (External Linkage)

如果可以在定义函数或变量的模块外部对函数或变量进行引用，则函数或变量具有外部链接。

## 外部输入线 (External Input Line)

用于根据外部信号设置事件的外部输入信号逻辑探针线 (TRIGIN)。

## 微处理器模式 (Microprocessor Mode)

PIC18 单片机的一种程序存储器配置。在微处理器模式下，不使用片内的程序存储器。整个程序存储器映射到外部。

## 伪指令 (Directive)

源代码中所提供对语言工具的操作进行控制的语句。

## 尾数法 (Endianness)

多字节对象中的字节存储顺序。

## 未初始化数据 (Uninitialized Data)

定义时未指定初始值的数据。在 C 中，

```
int myVar;
```

定义了将驻留在未初始化数据段的一个变量。

## 未分配段 (Unassigned Section)

在链接器命令文件中尚未分配到特定目标存储器块的段。链接器必须找到用于分配未分配段的目标存储器块。

## 文件寄存器 (File Register)

片内数据存储器，包括通用寄存器 (GPR) 和特殊功能寄存器 (SFR)。

## X

## 系统窗口控件 (System Window Control)

系统窗口控件位于窗口和某些对话框的左上角。单击该控件时通常会弹出包含 “Minimize” (最小化)、 “Maximize” (最大化) 和 “Close” (关闭) 项的菜单。

## 下载 (Download)

下载是指将数据从主机发送到其他设备 (如仿真器、编程器或目标板) 的过程。

## 限定符 (Qualifier)

次数计数器使用的地址或地址范围，或用作复杂触发中另一个操作之前的事件。

**响应延时 (Latency)**

事件与其得到响应之间的延迟时间。

**向量 (Vector)**

复位或中断发生时应用程序跳转到的存储地址。

**项目 (Project)**

项目包含编译应用程序（源代码和链接描述文件等）所需的文件以及这些文件与各种编译工具和编译选项之间的关联。

**消息 (Message)**

显示出来的文本，警告在语言工具的操作中可能存在的问题。消息不会停止操作。

**小尾数法 (Little Endian)**

多字节数据的数据存储顺序机制，在这种机制中，低字节存储在较低的地址中。

**修正 (Fixup)**

链接器重新定位后将目标文件符号引用替换为绝对地址的过程。

**虚拟字节 (Phantom Byte)**

dsPIC架构中的未实现字节，在将24位指令字视为32位指令字时使用。虚拟字节出现在dsPIC hex文件中。

**序言 (Prologue)**

编译器生成代码的一部分，负责分配堆栈空间、保存寄存器，以及执行运行时模型中指定的任何其他特定于机器的要求。该代码在给定函数的任何用户代码之前执行。

**Y****样机系统 (Prototype System)**

指用户的目标应用或目标板的术语。

**已初始化数据 (Initialized Data)**

定义时已指定初始值的数据。在C中，

```
int myVar=5;
```

定义了将存放放到已初始化数据段中的一个变量。

**已分配段 (Assigned Section)**

在链接器命令文件中已分配到目标存储器块的GCC编译器段。

**异步 (Asynchronously)**

不同时发生的多个事件。通常用来指可能在处理器执行过程中的任意时刻发生的中断。

**异步激励 (Asynchronous Stimulus)**

为模拟被模拟器件的外部输入而生成的数据。

**应用 (Application)**

可以由PIC<sup>®</sup>单片机控制的一组软件和硬件。

**用于器件的MPLAB入门工具包 (MPLAB Starter Kit for Device)**

Microchip的入门工具包中包含着手研究指定器件所需的全部内容。查看工作中的应用程序，然后调试和编程您自己的更改。

**用于器件的MPLAB语言工具 (MPLAB Language Tool for Device)**

Microchip用于指定器件的C编译器、汇编器和链接器。根据应用要使用的器件选择语言工具的类型，例如，如果要为PIC18 MCU创建C代码，请选择适合PIC18 MCU的MPLAB C编译器。

## 优先顺序 (Precedence)

定义表达式中求值顺序的规则。

## 原始数据 (Raw Data)

与一个段有关的代码或数据的二进制表示。

## 源代码 (Source Code)

编程人员编写计算机程序所使用的形式。源代码以某种正规的编程语言编写，可翻译成机器码或被解释程序执行。

## 源文件 (Source File)

包含源代码的 ASCII 文本文件。

## 运算符 (Operator)

加号 “+” 和减号 “-” 之类的符号，它们在构成定义明确的表达式时使用。每个运算符都有用于确定求值顺序的指定优先级。

## 运行 (Run)

将仿真器从暂停状态释放，允许仿真器实时运行应用代码、实时改变 I/O 状态或实时响应 I/O 的命令。

## 运行时观察 (Run Time Watch)

Watches 窗口中的变量在应用程序运行时变化。要确定如何设置运行时观察，请参见相应的工具文档。并非所有工具都支持运行时观察。

## 运行时模型 (Run-time Model)

介绍目标架构资源的使用。

## Z

## 暂停 (Halt)

停止程序执行。执行 Halt 与在断点处停止相同。

## 帧指针 (Frame Pointer)

引用堆栈中地址，并将基于堆栈的参数和基于堆栈的局部变量分隔开的指针。为访问当前函数的局部变量和其他值提供了方便。

## 只读存储器 (Read Only Memory)

允许快速访问永久存储的数据，但不能增加或修改数据的存储器硬件。



**指令 (Instruction)**

告知中央处理单元执行特定操作，并可能包含操作中要使用数据的位序列。

**指令集 (Instruction Set)**

特定处理器理解的机器语言指令的集合。

**致命错误 (Fatal Error)**

引起编译立即停止的错误。不产生其他消息。

**中断 (Interrupt)**

传递到CPU的信号，它使CPU暂停执行正在运行的应用程序，把控制权转交给中断服务程序 (Interrupt Service Routine,ISR)，以处理事件。完成ISR时，将恢复应用程序的正常执行。

**中断处理程序 (Interrupt Handler)**

发生中断时处理特殊代码的子程序。

**中央处理单元 (Central Processing Unit)**

器件的一部分，负责取出要执行的正确指令，对指令进行译码，然后执行指令。需要时，它和算术逻辑单元 (Arithmetic Logic Unit, ALU) 配合工作，来完成指令的执行。中央处理单元控制程序存储器地址总线、数据存储器地址总线以及堆栈的访问。

**中断服务程序 (Interrupt Service Routine, ISR)**

语言工具——用于处理中断的函数。

MPLAB IDE/MPLAB X IDE——在产生中断时进入的用户生成的代码。代码在程序存储器中的位置通常取决于所产生中断的类型。

**中断服务请求 (Interrupt Service Request, IRQ)**

使处理器暂停正常的指令执行并开始执行中断处理程序的事件。某些处理器有几种中断请求事件，允许具有不同优先级的中断。

**中断向量 (Interrupt Vector)**

中断服务程序或中断处理程序的地址。

**主机 (Pod)**

在线仿真器或调试器的外壳。其他名称还有“Puck”（如果外壳是圆的）和“Probe”（不要与逻辑探头混淆）。

**助记符 (Mnemonic)**

可直接翻译为机器码的文本指令。也称为操作码。

**状态栏 (Status Bar)**

状态栏位于MPLAB IDE/MPLAB X IDE窗口的底部，指示诸如光标位置、开发模式、器件和有效工具栏之类的当前信息。

**字母数字字符 (Alphanumeric)**

字母数字字符由字母字符和十进制数字 (0,1, ...,9) 组成。

**字母字符 (Alphabetic Character)**

字母字符指属于拉丁字母表 (a, b, ..., z, A, B, ..., Z) 中字母的字符。

**左值 (L-value)**

引用可被检查和/或修改的对象的表达式。左值表达式用在赋值的左侧。

注:

## 索引

<b>符号</b>		代码保护 .....	37
__LOG() .....	57	代码性能分析 .....	72
__TRACE() .....	57	低电压编程, 使能 .....	105
<b>数字</b>		电缆	
32位器件		长度 .....	113, 117
PC性能分析 .....	70	电源监视器 .....	18
数据捕捉 .....	52	掉电模式 .....	91
指令跟踪 .....	61	调试	
8位和16位器件		操作序列 .....	37
PC采样 .....	67	无法调试的首要原因 .....	83
数据捕捉 .....	51	执行程序 .....	38
<b>A</b>		调试/编程快速参考 .....	45
AC244002 .....	18	调试头规范 .....	12
AC244005 .....	18	冻结外设设置 .....	106
AC244006 .....	18, 35	读物, 推荐 .....	12, 13
AC244007 .....	18	断点	
AC244008 .....	18	软件 .....	48
ACICE0114 .....	18	设置 .....	48
按钮 .....	114	硬件 .....	48
按器件列出的软件模拟器调试功能 .....	23	端口跟踪 .....	57
按器件列出的软件模拟器外设支持 .....	23	<b>E</b>	
按器件列出的硬件工具调试功能 .....	23	EEPROM存储器 .....	104
<b>B</b>		ETN .....	99
本机跟踪 .....	33, 56	EXTn .....	34, 115
编程		<b>F</b>	
命令行 .....	45	仿真扩展工具包 .....	18
生产 .....	45	仿真器固件, 手动下载 .....	92
表读保护 .....	37	仿真器选项选择 .....	103
标准通信 .....	28	辅助闪存 .....	104
连接 .....	30	<b>G</b>	
驱动板 .....	116	高速通信 .....	30
并行跟踪 .....	33	高性能工具包 .....	18
捕捉, 数据 .....	51	优点 .....	30
<b>C</b>		隔离器单元 .....	18
测试板 .....	18, 119	跟踪	
插装跟踪 .....	54	插装 .....	54
I/O 端口 .....	57	指令 .....	61
常规纠正措施 .....	97	跟踪接口工具包 .....	18
触发 .....	79	跟踪快速参考 .....	60
外部 .....	79	跟踪连接 .....	33
触发信号		跟踪设置 .....	107
外部 .....	114	工程技术说明 .....	99
触发信号, 设置 .....	110	工具包组件 .....	18
处理器扩展工具包 .....	18	功能支持 .....	21
<b>D</b>		固件	
DMCI .....	52	下载时断开了连接 .....	92
DTS .....	21	固件下载 .....	110
DV244005 .....	114	观察	
		运行时 .....	51
		关于 Microchip Technology .....	125

# 适用于MPLAB X IDE的仿真器用户指南

<b>H</b>			
函数级别性能分析.....	72		
互联网地址, Microchip.....	124		
环回测试板.....	18, 119		
<b>I</b>			
I/O 端口跟踪.....	57		
ICSP.....	36, 37, 39		
ID 存储器.....	104		
<b>J</b>			
JTAG 适配器.....	18		
集线器, USB.....	113		
<b>K</b>			
看门狗定时器.....	37, 134		
客户支持.....	125		
快速参考			
调试/编程.....	45		
跟踪.....	60		
<b>L</b>			
LED.....	114		
LVDS.....	30		
LVP 配置位.....	105		
逻辑探头连接器.....	114		
I/O 电气规范.....	115		
外部触发条件.....	79		
引脚排列.....	115		
逻辑探针.....	18		
逻辑探针连接器			
I/O 端口跟踪.....	33		
<b>M</b>			
MCLR.....	31		
MCLR 上的上拉电阻.....	31		
Memory Ranges.....	104		
MPLAB REAL ICE 定义.....	19		
myMICROCHIP 个性化通知客户服务.....	123		
命令行编程.....	45		
模块化接口电缆.....	37		
目标板连接			
标准.....	30		
电路.....	31		
I/O 端口.....	33		
SPI.....	33		
目标器件.....	37		
目标器件 ID (0x0).....	83		
<b>N</b>			
耐用性, 板导槽.....	114		
<b>P</b>			
PC, 掉电.....	91		
PC 采样.....	67		
PC 性能分析.....	70		
PGC.....	31		
PGC, PGD.....	30, 36, 37, 38		
PGD.....	31		
PIC32 指令跟踪.....	35, 63		
PIM.....	29, 63		
PORTx.....	33		
Preserve Memory.....	104		
跑表.....	48		
配置位.....	37		
<b>Q</b>			
器件和功能支持.....	21		
器件与功能支持 HTML 表.....	23		
驱动板			
标准.....	18, 116		
高速.....	18		
<b>R</b>			
REALICECMD.....	45		
软件断点选择.....	105		
<b>S</b>			
Skidding, 断点.....	105		
SPI 跟踪.....	33, 56		
SQTP.....	45		
闪存窗口不反映变化.....	92		
时钟切换.....	110		
时钟速度.....	56, 57, 110		
数据捕捉.....	51		
数据速率.....	116		
<b>T</b>			
TRCLK.....	63, 64		
TRDn.....	63, 64		
跳转跟踪.....	66		
通过器件保留资源.....	39		
头规范.....	12		
<b>U</b>			
USB.....	113, 141		
电缆.....	18		
集线器.....	113		
USB 驱动程序.....	43		
<b>V</b>			
Vdd, Vss.....	30, 31, 36, 37, 115		
Vpp.....	30, 31, 37		
Vss.....	31		
<b>W</b>			
外部触发信号.....	114		
外部触发信号设置.....	110		
网站, Microchip.....	124		
文档			
编排.....	10		
约定.....	11		
<b>X</b>			
休眠模式.....	91		
<b>Y</b>			
要编程的存储器.....	104		
仪器跟踪			
本机.....	33, 56		
I/O 端口.....	33		
SPI.....	33, 56		
引导闪存.....	104		
硬件工具保持连接状态.....	43		
应用程序 IO.....	75		

应用程序输入/输出.....	75
用户 ID 存储器.....	104
运行时观察.....	51

## Z

在编程之前擦除所有存储器.....	105
在代码中设置配置位.....	43
暂停时冻结.....	91
指令跟踪, PIC32.....	35, 63
指示灯.....	114
主机.....	18
转接插座.....	18
规范.....	12
自述文件.....	12

## 全球销售及服务中心

### 美洲

公司总部 **Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 1-480-792-7200  
Fax: 1-480-792-7277

技术支持:  
<http://www.microchip.com/support>

网址: [www.microchip.com](http://www.microchip.com)

**亚特兰大 Atlanta**  
Duluth, GA  
Tel: 1-678-957-9614  
Fax: 1-678-957-1455

**奥斯汀 Austin, TX**  
Tel: 1-512-257-3370

**波士顿 Boston**  
Westborough, MA  
Tel: 1-774-760-0087  
Fax: 1-774-760-0088

**芝加哥 Chicago**  
Itasca, IL  
Tel: 1-630-285-0071  
Fax: 1-630-285-0075

**达拉斯 Dallas**  
Addison, TX  
Tel: 1-972-818-7423  
Fax: 1-972-818-2924

**底特律 Detroit**  
Novi, MI  
Tel: 1-248-848-4000

**休斯敦 Houston, TX**  
Tel: 1-281-894-5983

**印第安纳波利斯 Indianapolis**  
Noblesville, IN  
Tel: 1-317-773-8323  
Fax: 1-317-773-5453  
Tel: 1-317-536-2380

**洛杉矶 Los Angeles**  
Mission Viejo, CA  
Tel: 1-949-462-9523  
Fax: 1-949-462-9608  
Tel: 1-951-273-7800

**罗利 Raleigh, NC**  
Tel: 1-919-844-7510

**纽约 New York, NY**  
Tel: 1-631-435-6000

**圣何塞 San Jose, CA**  
Tel: 1-408-735-9110  
Tel: 1-408-436-4270

**加拿大多伦多 Toronto**  
Tel: 1-905-695-1980  
Fax: 1-905-695-2078

### 亚太地区

亚太总部 **Asia Pacific Office**  
Suites 3707-14, 37th Floor  
Tower 6, The Gateway  
Harbour City, Kowloon  
Hong Kong  
Tel: 852-2943-5100  
Fax: 852-2401-3431

**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 - 重庆**  
Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

**中国 - 东莞**  
Tel: 86-769-8702-9880  
Tel: 86-20-8755-8029

**中国 - 广州**  
Tel: 86-20-8755-8029

**中国 - 杭州**  
Tel: 86-571-8792-8115  
Fax: 86-571-8792-8116

**中国 - 南京**  
Tel: 86-25-8473-2460  
Fax: 86-25-8473-2470

**中国 - 青岛**  
Tel: 86-532-8502-7355  
Fax: 86-532-8502-7205

**中国 - 上海**  
Tel: 86-21-3326-8000  
Fax: 86-21-3326-8021

**中国 - 沈阳**  
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 - 武汉**  
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 - 厦门**  
Tel: 86-592-238-8138  
Fax: 86-592-238-8130

**中国 - 香港特别行政区**  
Tel: 852-2943-5100  
Fax: 852-2401-3431

### 亚太地区

**中国 - 珠海**  
Tel: 86-756-321-0040  
Fax: 86-756-321-0049

**台湾地区 - 高雄**  
Tel: 886-7-213-7830

**台湾地区 - 台北**  
Tel: 886-2-2508-8600  
Fax: 886-2-2508-0102

**台湾地区 - 新竹**  
Tel: 886-3-5778-366  
Fax: 886-3-5770-955

**澳大利亚 Australia - Sydney**  
Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

**印度 India - Bangalore**  
Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

**印度 India - New Delhi**  
Tel: 91-11-4160-8631  
Fax: 91-11-4160-8632

**印度 India - Pune**  
Tel: 91-20-3019-1500

**日本 Japan - Osaka**  
Tel: 81-6-6152-7160  
Fax: 81-6-6152-9310

**日本 Japan - Tokyo**  
Tel: 81-3-6880-3770  
Fax: 81-3-6880-3771

**韩国 Korea - Daegu**  
Tel: 82-53-744-4301  
Fax: 82-53-744-4302

**韩国 Korea - Seoul**  
Tel: 82-2-554-7200  
Fax: 82-2-558-5932 或  
82-2-558-5934

**马来西亚 Malaysia - Kuala Lumpur**  
Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

**马来西亚 Malaysia - Penang**  
Tel: 60-4-227-8870  
Fax: 60-4-227-4068

**菲律宾 Philippines - Manila**  
Tel: 63-2-634-9065  
Fax: 63-2-634-9069

**新加坡 Singapore**  
Tel: 65-6334-8870  
Fax: 65-6334-8850

**泰国 Thailand - Bangkok**  
Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### 欧洲

**奥地利 Austria - Wels**  
Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

**丹麦 Denmark - Copenhagen**  
Tel: 45-4450-2828  
Fax: 45-4485-2829

**芬兰 Finland - Espoo**  
Tel: 358-9-4520-820

**法国 France - Paris**  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

**法国 France - Saint Cloud**  
Tel: 33-1-30-60-70-00

**德国 Germany - Garching**  
Tel: 49-8931-9700  
**德国 Germany - Haan**  
Tel: 49-2129-3766400

**德国 Germany - Heilbronn**  
Tel: 49-7131-67-3636

**德国 Germany - Karlsruhe**  
Tel: 49-721-625370

**德国 Germany - Munich**  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

**德国 Germany - Rosenheim**  
Tel: 49-8031-354-560

**以色列 Israel - Ra'anana**  
Tel: 972-9-744-7705

**意大利 Italy - Milan**  
Tel: 39-0331-742611  
Fax: 39-0331-466781

**意大利 Italy - Padova**  
Tel: 39-049-7625286

**荷兰 Netherlands - Drunen**  
Tel: 31-416-690399  
Fax: 31-416-690340

**挪威 Norway - Trondheim**  
Tel: 47-7289-7561

**波兰 Poland - Warsaw**  
Tel: 48-22-3325737

**罗马尼亚 Romania - Bucharest**  
Tel: 40-21-407-87-50

**西班牙 Spain - Madrid**  
Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

**瑞典 Sweden - Gothenberg**  
Tel: 46-31-704-60-40

**瑞典 Sweden - Stockholm**  
Tel: 46-8-5090-4654

**英国 UK - Wokingham**  
Tel: 44-118-921-5800  
Fax: 44-118-921-5820