



MPLAB®数据可视化器用户指南

MPLAB®数据可视化器用户指南

开发工具客户须知



重要:

开发工具手册如同所有其他文档一样具有时效性。Microchip 将不断改进工具和文档以满足客户的需求，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) 获取文档的最新版本。

文档每页的底部均标有“DS”编号。DS 编号的格式为 DS <文档编号> <版本>，其中<文档编号>为 8 位数字，<版本>为大写字母。

有关最新信息，请参见 onlinedocs.microchip.com/上的 MPLAB®数据可视化器帮助信息。



目录

开发工具客户须知.....	1
1. 前言.....	4
1.1. 本指南使用的约定.....	4
1.2. 推荐读物.....	5
2. 概述.....	6
2.1. 特性.....	6
2.2. 基本操作.....	6
2.3. MPLAB 数据可视化器安装.....	9
3. 外部连接.....	11
3.1. 数据网关接口 (DGI)	11
3.2. 串行端口.....	15
4. 可视化.....	16
4.1. 在 Terminal 中以文本形式查看数据.....	16
4.2. 在 Time Plot 中查看数据.....	25
4.3. 查看功率数据.....	57
4.4. 在 XY Plot 中查看两组数据.....	64
5. 变量设置器.....	67
5.1. 变量数据类型.....	67
5.2. 流格式.....	67
5.3. 变量流设置器的设置和绘制.....	67
5.4. 查看统计信息.....	71
6. 故障排除.....	72
6.1. 数据流传输.....	72
6.2. 数据流解码器.....	72
7. 查找发行说明.....	73
8. 绘制数据示例——代码清单.....	74
8.1. C 头文件代码.....	74
8.2. C 源代码.....	76
9. 版本历史.....	79
Microchip 网站.....	80
产品变更通知服务.....	80
客户支持.....	80
Microchip 器件代码保护功能.....	80
法律声明.....	80

商标.....	81
质量管理体系.....	81
全球销售及服务网点.....	82

1. 前言

本小节介绍 MPLAB 数据可视化器的文档和支持信息。

1.1 本指南使用的约定

本指南采用以下文档约定：

表 1-1. 文档约定

说明	表示	示例
Arial 字体:		
斜体字	参考书目	<i>MPLAB® IDE User's Guide</i>
	需强调的文字	<i>...仅有的编译器...</i>
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”
带右尖括号有下划线的斜体文字	菜单路径	<i><u>File>Save</u></i>
粗体字	对话框按钮	单击 OK (确定)
	选项卡	单击 Power 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>
Courier New 字体:		
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> 可以是任何有效的文件名
方括号[]	可选参数	mcc18 [options] file [options]
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) { ... }

1.2 推荐读物

本文档介绍如何使用 MPLAB 数据可视化器。下面列出了其他有用的文档。以下 Microchip 文档均已提供，并建议读者作为补充参考材料。

Atmel Data Visualizer User's Guide

以前的数据可视化器，此程序用于处理和可视化数据。Atmel 数据可视化器可从各个源接收数据，例如嵌入式调试器数据网关接口（Data Gateway Interface, DGI）和 COM 端口。

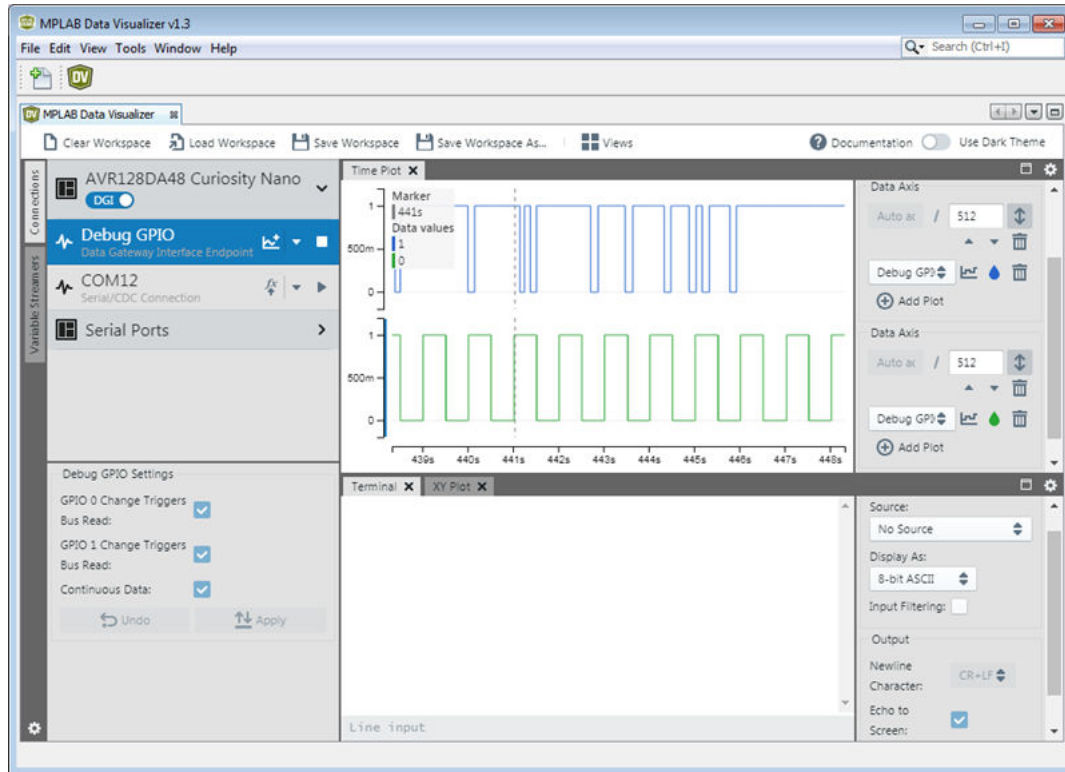
Data Gateway Interface User's Guide

数据网关接口（DGI）是一个 USB 接口，用于处理与目标 MCU 之间的低级数据传输。有一些工具和板上调试器支持 DGI，例如 Xplained Pro 板上的功率调试器和 EDBG。

2. 概述

MPLAB 数据可视化器是一款用于处理运行中的嵌入式目标板的数据并使数据可视化的程序。该程序可以从 MPLAB X IDE 中访问，也可将其作为独立程序使用。

图 2-1. 数据流输出



2.1 特性

应用正在运行时，很难排除解决嵌入式目标板上的数据问题。就像调试器可以帮助您调试应用程序代码一样，MPLAB 数据可视化器可以帮助您调试数据。借助 MPLAB 数据可视化器，您可以查看应用程序中的关键数据点在运行时如何变化，例如，使开发板上的传感器捕捉到的值可视化。

MPLAB 数据可视化器具有以下特性：

- 通过串行端口（CDC）或数据网关接口（DGI）从正在运行的嵌入式目标板捕捉数据流
- 监控功率数据
- 使用数据流协议格式在运行时解码数据字段
- 将原始数据或解码数据可视化为时间序列图或 XY 图，或在 Terminal 中显示该数据。
- 使用 MPLAB® X IDE 同时进行数据流传输和目标板代码调试
- 使用光标分析绘制数据，以测量带宽和脉冲宽度等
- 将图形快照保存为 CSV 或 JSON 文件
- 添加插件，获得更多的功能

2.2 基本操作

数据可视化器可作为 MPLAB X IDE 插件或独立程序运行。作为**插件**时，可以在使用数据可视化器功能的同时调试代码。作为**独立程序**时，则不能调试代码。但是，您可以利用同一工具包，在 MPLAB X IDE 中调试，同时通过独立程序进行数据流传输。

了解如何将嵌入式目标板连接到您的 PC，请参见 [3. 外部连接](#)。

了解进行数据可视化的不同方式及其示例，请参见 [4. 可视化](#)。

了解变量值的绘制方法，请参见 [5. 变量设置器](#)。

2.2.1 界面

MPLAB 数据可视化器界面 由三个区域组成（见下图）：

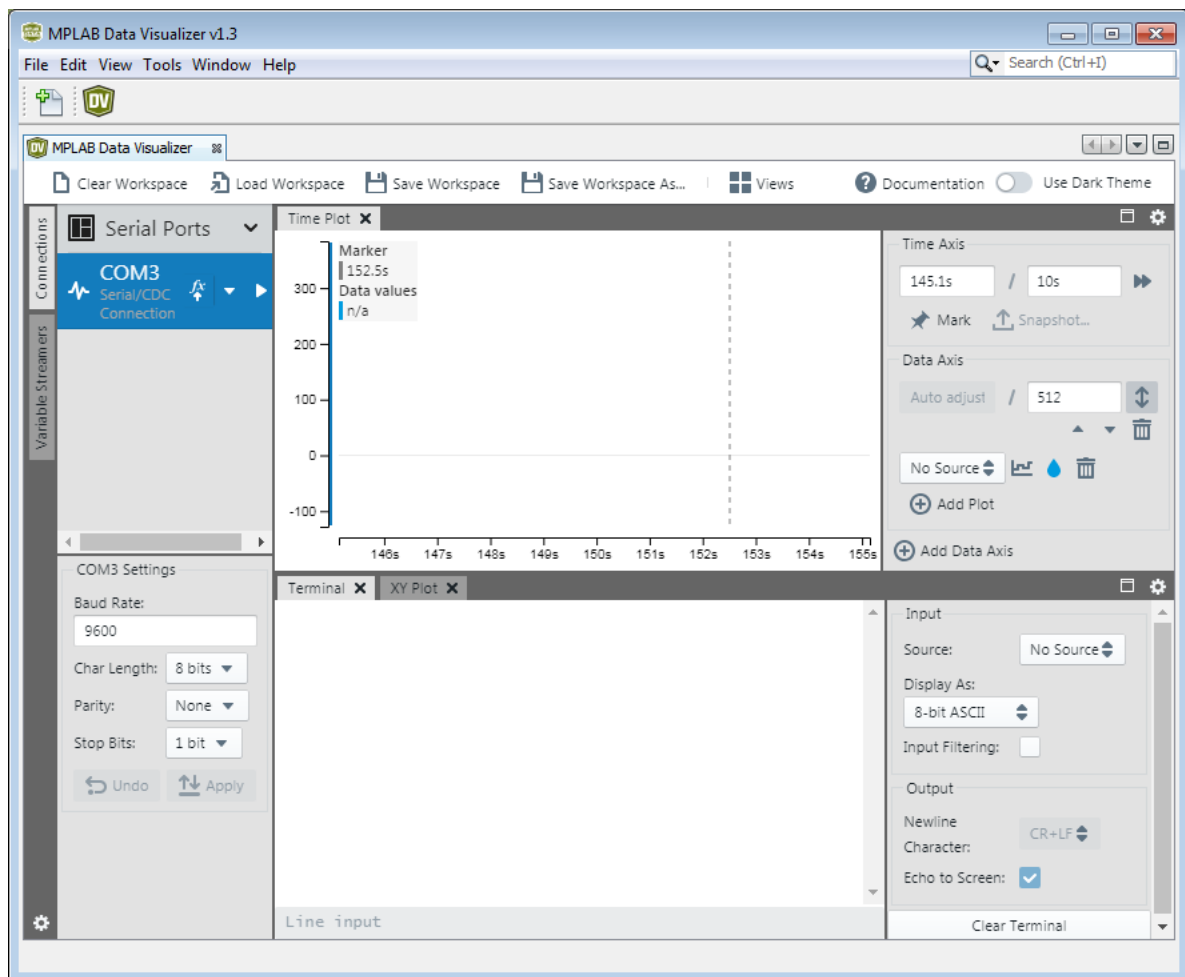
1. 中间是数据显示窗口。**Time Plot**（时间图）是最常用的窗口。
2. 左侧窗格用于标识和设置连接的数据源。单击 **Variable Streamers**（变量流设置器）将显示控件，以设置要显示的变量值。
3. 右侧窗格用于设置数据的显示格式。

所用的中央窗口不同，两侧窗格中的信息会随之改变。我们将在以下小节中讨论界面中的控件。



提示：多数控件的信息位于弹出窗口中。只需将鼠标悬停在控件上即可看到。

图 2-2. 界面布局



2.2.1.1 工具栏控件

可视化器工具栏位于界面顶部，包含下表中所列的控件。

表 2-1. 工具栏控件

控件图	控件	说明
	Clear Workspace (清除工作区)	清除工作区中的数据 and 设置。将停止所有数据流传输。
	Load Workspace (加载工作区)	将上一个会话中的数据 and 设置加载到可视化器中。
	Save Workspace (保存工作区) Save Workspace As (将工作区保存为)	将当前会话中的数据 and 设置保存到文件中。
	Views (视图)	从可用视图中选择一个窗口视图。例如, Time Plot、XY Plot (XY 图)、Terminal (终端) 或插件窗口。
	Documentation (文档)	显示可视化器的在线帮助。
	Use Dark Theme (使用深色主题)	使能: 工作区背景为黑色 禁止: 工作区背景为白色

2.2.1.2 独立菜单

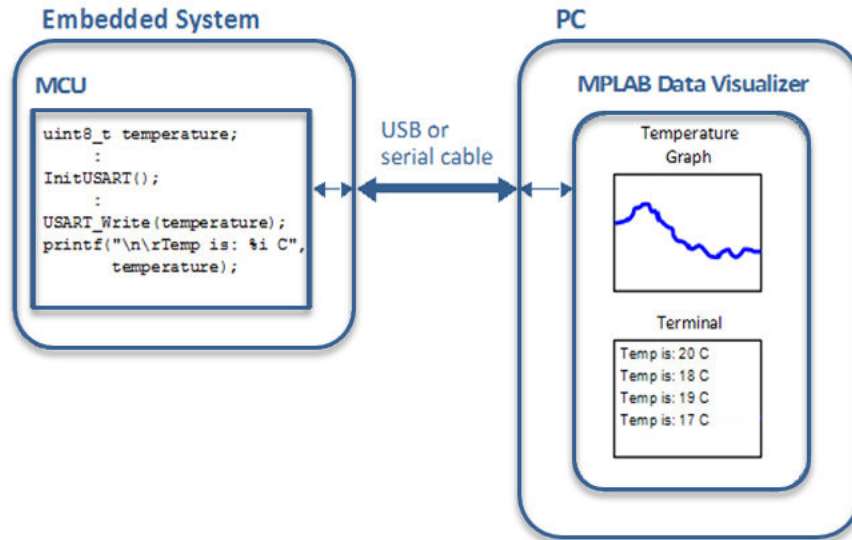
如果将 MPLAB 数据可视化器用作独立的应用程序, 它将带有一个菜单栏, 其中包含与 MPLAB X IDE 菜单项功能相似的菜单和项目。包括基本的文本编辑器功能, 允许编辑协议定义 (.ds) 文件等。

菜单>项目	项目和说明
File>Items (文件>项目)	基本的 File (文件) 菜单项。选项包括: New File (新建文件)、Open (Recent) File (打开 (最近的) 文件) 和 Exit File (退出文件)。
Edit>Items (编辑>项目)	基本的 Edit (编辑) 菜单项。选项包括: Undo/Redo (撤消/恢复)、Cut/Copy/Paste (剪切/复制/粘贴)、Delete (删除) 和 Find/Replace (查找/替换)。
View>Items (查看>项目)	基本的 View (查看) 菜单项。选项包括: Editors (编辑器)、Split (拆分)、IDE Log (IDE 日志)、Toolbars (工具栏)、Show Only Editor (仅显示编辑器) 和 Full Screen (全屏)。
Tools>Embedded (工具>已安装工具)	选择已安装的嵌入式工具。
Tools>Plugins (工具>插件)	打开 “Plugins” (插件) 对话框, 以添加、删除或管理插件。
Tools>Options (工具>选项)	选择数据可视化器选项: <ul style="list-style-type: none"> • General (通用) —— 网站和代理选项 • Keymap (键盘映射) —— 键盘映射选项 • Appearance (外观) —— 界面外观选项 • Miscellaneous (其他) —— 文件和输出字体/颜色选项
Window> Items (窗口>项目)	基本的 Window (窗口) 菜单项。选项包括: Favorites (收藏夹)、Output (输出)、Editor (编辑器)、IDE Tools (Notifications/Processes) (IDE 工具 (通知/进程))、Configure (配置, 包括大小、浮动、停靠和拆分等) Reset (复位)、Close (关闭)、Close all Documents (关闭所有文档)、Close Other Documents (关闭其他文档)、Document Groups (文档组) 和 Documents (文档)。
Help>Items (帮助>项目)	基本的 Help (帮助) 菜单项。选项包括: Help Contents (帮助内容)、Online Docs and Support (在线文档和支持)、Keyboard Shortcuts Card (键盘快捷方式卡)、Check for Updates (检查更新) 和 About (关于)。

2.2.2 连接

MPLAB 数据可视化器从正在运行的嵌入式目标板中捕捉数据并显示数据。例如，将一个温度值发送到可视化器，将其绘制到图形中或在 Terminal 上显示。

图 2-3. 连接示例

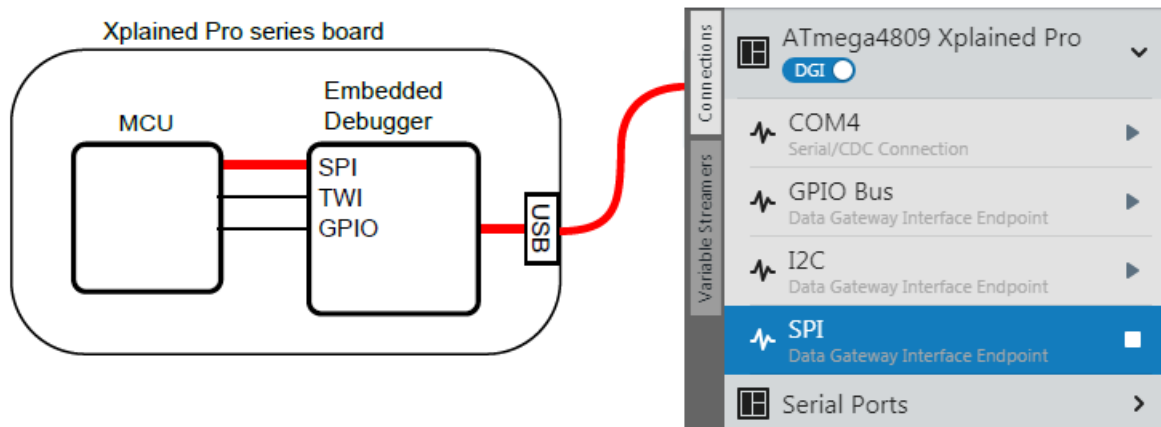


2.2.3 数据网关接口 (DGI)

MPLAB 数据可视化器可以通过数据网关接口 (DGI) 与目标板的嵌入式调试器进行通信。当通过 USB 线缆将目标板连接至计算机时，可视化器将显示目标板上具有 DGI，并会在下方显示可用接口 (SPI 和 UART 等) 的列表。

在下图中，已使能 SPI 接口。MCU 现在可通过其 SPI 端口与可视化器通信。

图 2-4. 数据网关接口



2.3 MPLAB 数据可视化器安装

该可视化器有两种运行方式：作为 MPLAB X IDE 插件或作为独立程序。

从 MPLAB X IDE v5.30 开始提供对数据可视化器的支持。

2.3.1 MPLAB X IDE 插件

从 MPLAB X IDE v5.5 开始，MPLAB 数据可视化器插件成为 MPLAB X IDE 的一部分。

要在 MPLAB X IDE 中访问数据可视化器，选择 *Window>Debugging>Data Visualizer*（窗口>调试>数据可视化器）或 *Window>Debugging>Power Monitoring*（窗口>调试>功率监控），以打开数据可视化器进行功率监控。

要查看是否有插件版本更新，转到 *Tools>Plugins>Updates*（工具>插件>更新）并单击 **Check for Updates**（检查更新）。有关插件的更多信息，请参见 MPLAB X IDE 文档中的“**Add Plugin Tools**”（添加插件工具）。

2.3.2 独立应用程序

要将该可视化器作为独立程序安装，请访问以下“Microchip Gallery”链接：

[MPLAB-Data-Visualizer-Standalone](#)

Launch-and-install 或 download-and-install。按照安装向导屏幕提示进行操作。安装完成后，从安装位置启动该可视化器。

对于独立应用程序，它将带有一个菜单栏，其中包含与 MPLAB X IDE 菜单项功能相似的菜单和项目。请参见 [2.2.1.2 独立菜单](#)。

3. 外部连接

外部连接是指目标板硬件和 PC 之间使用的硬件连接。这些连接定义了目标板和 MPLAB 数据可视化器之间的通信类型。连接类型取决于器件支持。

要将目标板连接到 PC: 请按照有关器件或演示板的说明操作。

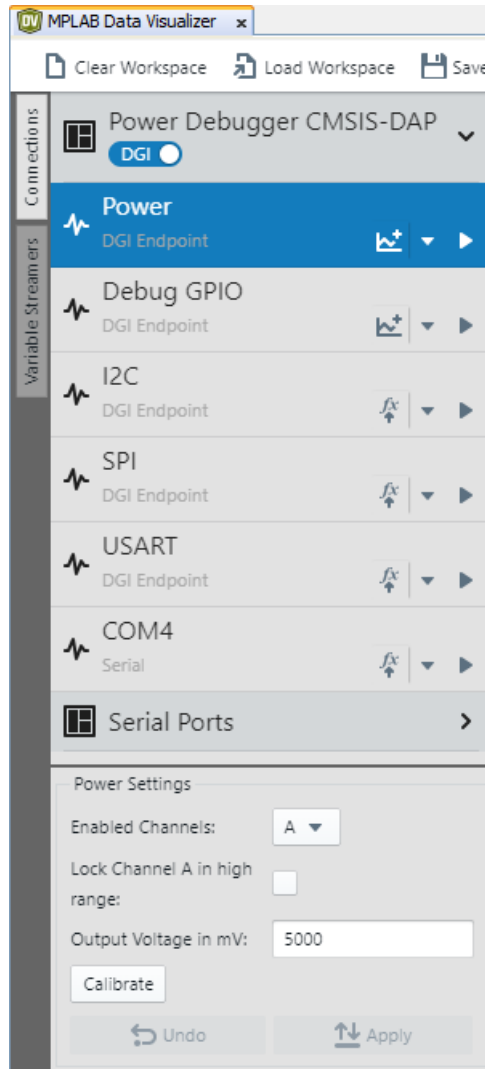
要选择可视化器的连接, 请执行以下操作: 在数据源 (左侧) 窗格上:

- 串行/ CDC 连接——该选项将指定与系统上任何可以使用波特率、奇偶校验、数据位和停止位进行设置的串行端口的通信。
- DGI 工具——该选项将指定与任何具有数据网关接口的工具的通信。支持通过 SPI、I2C、USART 和 Debug GPIO (调试 GPIO) 进行输入数据流通信。功能集因工具而异。

3.1 数据网关接口 (DGI)

数据网关接口 在大多数带有嵌入式调试器的工具包中提供。可视化器 DGI 控件可以与 DGI 设备通信。在数据源 (左侧) 窗格中设置 DGI 控件。

所有检测到的 DGI 设备都列在左侧窗格的 **Connections** (连接) 下。要启用某个接口, 请单击其名称 (如 **Power** (功率))。在下方的窗格中设置接口 (如 **Power Settings** (功率设置))。可视化器接受来自支持 DGI 的评估板的数据流输入。



3.1.1 Power 接口

Power 接口测量连接电路的功耗。有关 Power 接口硬件部分的更多信息，请参见用于功率测量的调试工具的用户指南。

选择调试工具 DGI 下的 Power 接口。使用 Power Settings 下面的控件设置接口。可用控件将根据所连接调试工具的功能而有所不同。

表 3-1. Power Settings 控件

控件	值	用途
Enabled Channels (使能通道)	A, A+B	仅使能通道 A，或使能通道 A 和通道 B。通道 A 始终使能。
Lock Channel A in high range (将通道 A 锁定为高电流范围)	取消勾选, 勾选	在功率调试器中，可以将通道 A 锁定在高电流范围，以避免自动切换到低电流范围。这样就可以检测电流消耗中的短尖峰，并且在不同范围间切换时不会丢失关键采样。
Output Voltage in mV (输出电压，单位为 mV)	介于 1600 mV 和 5500 mV 之间，或者为 0	功率调试器提供可调节的目标电源，可用于为目标应用供电。此设置用于使能和控制该电源的输出电压。选择 0 将禁止电源。

..... (续)

控件	值	用途
Calibrate (校准)	单击以激活	触发电流测量电路的校准过程。为了能通过当前的测量硬件获得完全测量精度，应该在执行任何测量之前对其进行校准。



提示： 只有在 Power Settings 窗格中单击 **Apply** (应用) 之后，所有设置更改才会生效。例如，要使用 Voltage Output (电压输出)，必须先选择 Output Voltage (输出电压) 值并单击 **Apply**，然后才能实际使用和设置电压输出。



提示： 如果调试器已经在低电流范围内运行，则通道 A 范围锁定不会强制调试器返回高电流范围。要么等待电流足够大以强制其改变，要么直接停止后再启动调试器。



提示： 每个功率信号时间图都会使用系统资源。减少同时绘制的图形数量将会获得更好的性能。

3.1.2 GPIO 接口

GPIO 接口包含使能的 Debug GPIO 引脚的位值。每次引脚翻转状态时，都会发送一个无符号的 8 位数据包。有关 GPIO 接口物理部分的更多详细信息，请参见用于对 GPIO 数据进行采样的调试工具的用户指南。

在数据源 (左侧) 窗格中，选择了 GPIO 接口后，该窗格的下部将显示 GPIO 设置。

表 3-2. 配置

字段名称	值	用途
GPIO 0 Change Triggers Bus Read (GPIO 0 变化触发总线读取)	开, 关	监控 GPIO 引脚 0 的变化以触发总线读取
GPIO 1 Change Triggers Bus Read (GPIO 1 变化触发总线读取)	开, 关	监控 GPIO 引脚 1 的变化以触发总线读取
GPIO 2 Change Triggers Bus Read (GPIO 2 变化触发总线读取)	开, 关	监控 GPIO 引脚 2 的变化以触发总线读取
GPIO 3 Change Triggers Bus Read (GPIO 3 变化触发总线读取)	开, 关	监控 GPIO 引脚 3 的变化以触发总线读取



重要： 当绘制 Debug GPIO 数据源时，将对所有 GPIO 进行采样，但只有那些使能了变化触发功能的 GPIO 发生状态变化时，才会触发采样。例如，如果 GPIO n (n = 0, 1, 2) 都禁止了“GPIO n Change Triggers Bus Read” (GPIO n 变化触发总线读取)，但 GPIO 3 使能了该功能，那么只有当 GPIO 3 发生变化时才会对 GPIO 值进行采样。也就是说，所有四个 GPIO 值只有在 GPIO 3 变化时才会被读取。

3.1.3 USART 接口

USART 源对 USART 接口上接收到的原始值进行流传输。有关 USART 接口物理部分的更多详细信息，请参见用于对 USART 数据进行采样的调试工具的用户指南。

在数据源 (左侧) 窗格上，选择了 USART 源后，该窗格的下部将显示 USART 设置。

注： 异步串行协议（例如 DGI USART 和 CDC 虚拟 COM 端口等接口使用的 UART 协议）使用 3.2 串行端口中所列的波特率：

表 3-3. USART 设置

字段名称	值	用途
Baud Rate（波特率）	0-2000000	异步模式下 UART 接口的波特率
Char Length（字符长度）	5、6、7 或 8 位	每次传输的位数
Parity（奇偶校验）	None（无校验）、Even（偶校验）、Odd（奇校验）、1 校验（Mark）或 0 校验（Space）	用于通信的奇偶校验类型
Stop bits（停止位）	1、1.5 或 2 位	停止位数

3.1.4 I2C 接口

I2C 源对 I2C 接口上接收到的原始值进行流传输。有关 I2C 接口物理部分的更多详细信息，请参见用于对 I2C 数据进行采样的调试工具的用户指南。

I2C Configuration（I2C 配置）选项显示在左侧窗格 **DGI** 部分的 **I2C** 接口下。

I2C 接口位于数据源（左侧）窗格的 **DGI** 部分下。选择 I2C 连接时，该窗格的下部将显示 I2C 设置。

表 3-4. I2C 设置

字段名称	值	用途
Speed（速度）	0	接口的预期运行速度（单位为赫兹）有助于从设备调整时序。最高支持 400 kHz。
Address（地址）	1	从设备的地址。
Kit-side Timestamping（工具包侧时间戳）	选中使能。	目标板时间戳

3.1.5 SPI 接口

SPI 源对 SPI 接口上接收到的原始值进行流传输。有关 SPI 接口物理部分的更多详细信息，请参见用于对 SPI 数据进行采样的调试工具的用户指南。



重要： SPI 硬件模块使用低电平有效的片选（Chip Select, CS）信号。CS 引脚为高电平时发送的任何数据都将被忽略。

SPI 接口位于数据源（左侧）窗格 **DGI** 的部分下。选择 SPI 连接时，该窗格的下部将显示 SPI 设置。

表 3-5. SPI 设置

字段名称	值	用途
Char Length	5、6、7 或 8 位	每次传输的位数
Mode（模式）	<ul style="list-style-type: none"> 时钟空闲状态通常为低电平，在上升沿采样数据 时钟空闲状态通常为低电平，在下降沿采样数据 时钟空闲状态通常为高电平，在下降沿采样数据 时钟空闲状态通常为高电平，在上升沿采样数据 	SPI 模式，控制时钟相位和进行采样。
Force CS Sync（强制 CS 同步）	选中使能。	片选线翻转两次后，才使能 SPI 接口。

..... (续)		
字段名称	值	用途
Kit-side Timestamping (工具包侧时间戳)	选中使能。	目标板时间戳

3.2 串行端口

数据可视化器可以通过标准 PC 串行端口连接到目标板。在数据源（左侧）窗格中设置串行控件。

波特率、停止位和奇偶校验必须设置为与通信对端所需的设置相匹配。在图形和 Terminal 中，将串行端口数据视为无符号的 8 位数据。

表 3-6. 配置

字段名称	值	用途
Baud rate	600-2000000	串行接口波特率
Char Length	5、6、7 或 8 位	每次传输的位数
Parity	None（无校验）、Even（偶校验）、Odd（奇校验）、1 校验（Mark）或 0 校验（Space）	用于通信的奇偶校验类型
Stop bits	1、1.5 或 2 位	停止位数

注：异步串行协议（例如 DGI USART 和 CDC 虚拟 COM 端口等接口使用的 UART 协议）使用以下波特率：

- 9600
- 19200
- 38400
- 57600
- 115200
- 230400
- 500000
- 1000000
- 2000000

任何其他波特率都不适用于异步接口（DGI UART 和串行端口/CDC 虚拟 COM 端口）协议。

tty 和 cu 端口之间的区别

请参见 Godfrey van der Linden 发表在 Apple® Listserv 上的列表文章，网址为 lists.apple.com/archives/darwin-dev/2009/Nov/msg00099.html。

4. 可视化

可以使用本部分中包含的模块来可视化传入的数据。

Terminal 将数据显示为文本，可以是原始值，也可以是 ASCII 编码字符。它还能够发送基于文本的数据。

Time Plot 将随时间变化的传入数据可视化图形、带和字符串标志。可通过光标帮助分析数据，并可以提供用于设置阈值的输出值。

XY Plot 将输入的 x 与 y 数据可视化图形。

Power Analysis（功率分析）模块专用于分析随时间变化的功耗。它也可以与代码性能分析配合使用，以可视化计数器采样，从而允许大概了解程序执行与功耗之间的关系。

4.1 在 Terminal 中以文本形式查看数据

Terminal 是用于显示和发送简单文本或数值的原始终端。

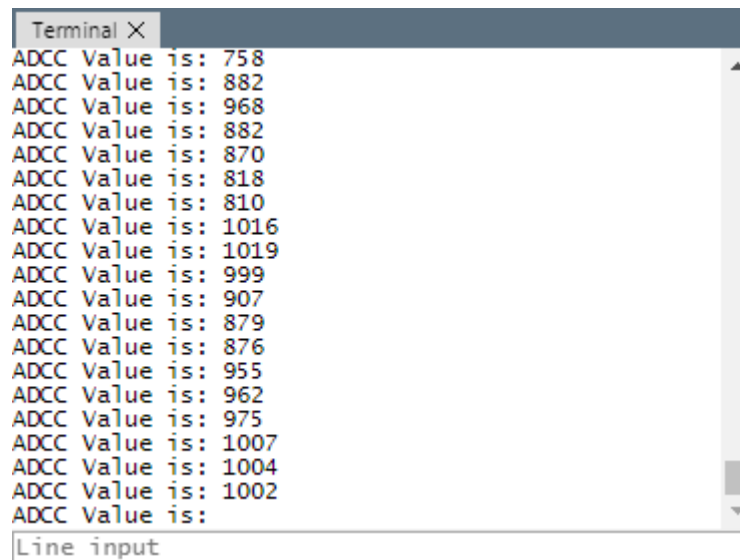
4.1.1 Terminal 窗口

MPLAB 数据可视化器的 Terminal 窗口能够以不同的格式显示数据流。使用左侧窗格中的 Connections 选项卡选择数据源。使用终端可视化控件（右侧）窗格，可从下拉列表中选择一种或多种格式。

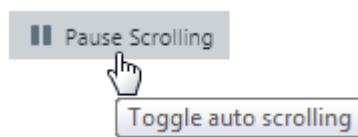
使能后，数据将连续滚动显示。

当通过 COM 端口将 MPLAB 数据可视化器连接到目标板时，还可以通过在终端区域或下方文本框中输入字符和文本行，将字符和文本行流传输至目标板。

图 4-1. 包含数据的 Terminal 窗口



您可以将鼠标悬停在 Terminal 窗口的顶部，单击“Pause Scrolling/Scroll to End”（暂停滚动/滚动到结尾）来切换数据流视图。虽然窗口视图已暂停，但仍继续在后台进行数据流传输。



4.1.2 Connections 选项卡

单击 Connections 选项卡，查看可用的连接。将目标板连接到 PC 后，将显示该连接并列板上数据源。

要选择一个数据源，例如“SPI”，单击该数据源，即可对其设置进行编辑。如果已连接并希望更改设置，则可以单击 **Stop Streaming**（停止流传输）按钮以断开连接。再次单击该按钮则重新连接。

有关连接的详细信息，请参见 [3. 外部连接](#)。





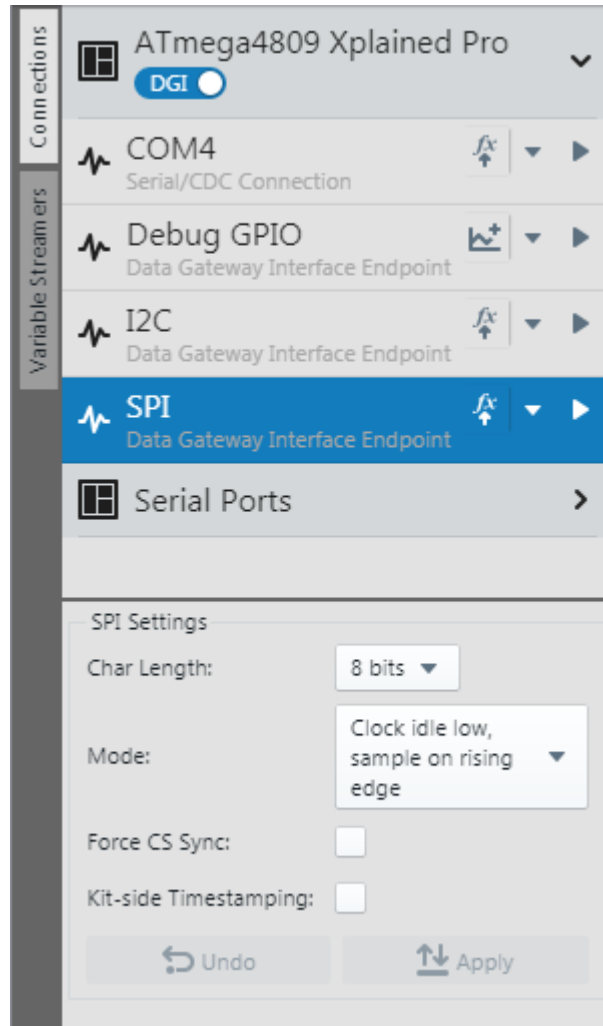
按钮	操作
	Start/stop streaming ——切换按钮：开始/结束来自数据源的数据流传输。图形和 Terminal 窗口将继续滚动。
	Plot ——根据数据源选择绘制类型：绘制原始数据、绘制所有引脚、绘制变量流设置器或将数据发送到 Terminal。此外，还可以移除所有图形中的数据源。
	New Variable Streamer ——打开向导以设置新的变量流设置器。
	Plot All Pins ——在图形上绘制所有 GPIO 引脚。

图 4-2. Connections 选项卡



4.1.3 终端可视化控件（右侧）窗格

在 Terminal 区域中，终端可视化控件（右侧）窗格用于选择源并控制流数据的格式。

图 4-3. 终端可视化控件

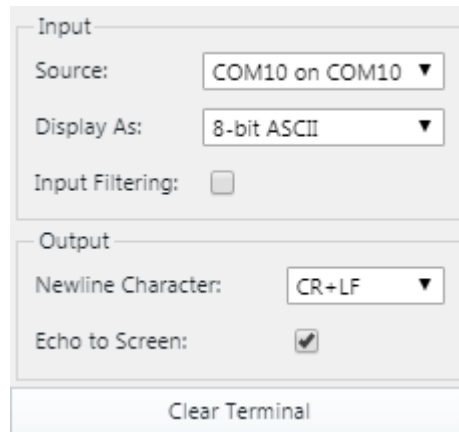


表 4-1. 输入数据

控件	说明
Source（源）	支持除 Debug GPIO 外的所有数据源。
Display As（显示为）	选择如何将数据流转换为终端字符。当前选项为： <ul style="list-style-type: none"> • ASCII • UTF-8 • Hex values（十六进制值）
Input Filtering（输入筛选）	单击复选框，将从输入流中滤除 ANSI/VT100 终端控制字符 1B、90、98、9B、9D、9E 和 9F，因为它们对于嵌入式终端组件具有特殊含义。

输出部分中的控件仅对 COM 端口连接使能。

表 4-2. 输出数据

控件	说明
Newline Character（换行符）	选择哪个（些）字符将在输出流中表示换行符。 <ul style="list-style-type: none"> • None（无） • CR+LF：回车+换行 • LF：换行
Echo to Screen（屏幕回显）	选中时，将输入的字符回显到屏幕上。

要删除 Terminal 窗口的内容，单击 **Clear Terminal**（清除终端）。

4.1.4 终端示例

以下是终端的使用示例。

4.1.4.1 原始数据流传输示例

MPLAB Xpress 评估板 用于根据电位器的值生成数据。该评估板支持串行/CDC 连接，这是适用于可视化器的有效连接。

一个 MPLAB Xpress 代码示例用于快速开发用于生成数据的 MPLAB X IDE 应用程序。另一个代码示例用于演示如何设置 USART，从而通过 USB CDC 连接将数据发送到可视化器。

本示例将使用的工具包括：

- MPLAB X IDE v5.40（或更高版本）
- MPLAB XC8 C 编译器 v2.20（或更高版本）
- MPLAB Xpress 评估板（PIC16F18855）——DM164140

4.1.4.1.1 获取示例软件和硬件

请按照以下说明完成操作，以设置示例软件和硬件。

MPLAB X IDE

访问以下链接，免费下载并安装 MPLAB X IDE 5.40（或更高版本）。

www.microchip.com/mplab/mplab-x-ide

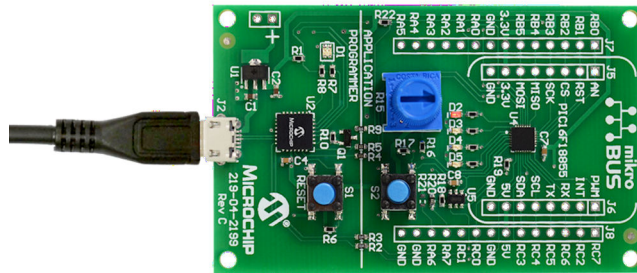
MPLAB XC8 C 编译器

访问以下链接，下载并安装 MPLAB XC8 C 编译器 v2.20（或更高版本）。本示例使用了免费编译器。还可以购买包含更多优化和功能的 PRO 编译器。

www.microchip.com/mplab/compilers

MPLAB Xpress 评估板——DM164140

此评估板可通过 Microchip 直销网站（<https://www.microchipdirect.com/>）或代理商购买。



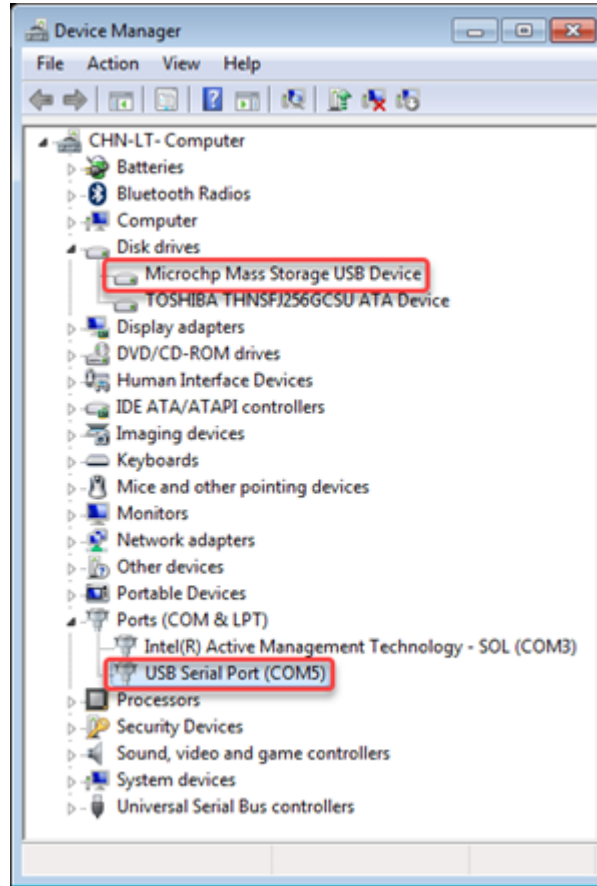
4.1.4.1.2 设置 MPLAB Xpress 评估板

首先，使用随附的 USB 线缆，将 MPLAB Xpress 评估板连接到计算机，并安装驱动程序。接下来，您需要决定：

- 是否需要使用支持串行通信的 USB CDC 驱动程序。该驱动程序可在 MPLAB Xpress 网页的“Downloads”（下载）选项卡下获取：
www.microchip.com/mplab/mplab-xpress
- USB 端口的 COM 号。稍后的数据显示将需要此信息。

请按照以下步骤进行操作。

1. 打开 Device Manager（设备管理器，Windows）或 System Information/Profiler（系统信息/配置器，Mac）或 lshw（Linux）等。
2. 找到评估板并确保已连接。
注：MPLAB Xpress 评估板的作用就如同一个 USB 大容量存储设备。
3. 查看端口。记下 COM 号，例如 USB 串行端口（COMx）。
4. 拔出评估板，再重新插入。如果串行端口消失又重新出现，则表明串行连接已经可用。如果没有，则需要安装 USB CDC 驱动程序。




4.1.4.1.3 获取示例代码

MPLAB Xpress 评估板的代码示例可在下面的链接中找到：

mplabxpress.microchip.com/mplabcloud/example

对于此示例，需要使用电位计（pot）和器件 ADC 的代码。要找到此类项目，请在“Tags”（标签）下选中“ADC”，然后在“Boards”（评估板）下选中“Xpress Board”（Xpress 板）。此示例中使用的项目是“**analogReadSerialWrite using ADCC in Basic Mode**”。

MPLAB Xpress Code Examples



Title	Author	Like	Watch	Import	Tags	Board
<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Search	<input type="checkbox"/>	From	From	From	<input type="checkbox"/> #Getting Started	<input type="checkbox"/> Curiosity Board
	<input type="checkbox"/>	To	To	To	<input checked="" type="checkbox"/> ADC	<input type="checkbox"/> Curiosity HPC Board
					<input type="checkbox"/> ADCC	<input type="checkbox"/> Custom
LED brightness control using potentiom...		1	6	3066	ADC, Lighting, PWM	Xpress Board
Analog Input and Output		0	1	879	ADC, PWM	Xpress Board
analogReadSerialWrite using ADCC in...		0	0	608	ADC, ADCC, UART	Xpress Board

单击“Title”（标题）下的项目名称，转到具体网页。在此页面上，有一个关于该项目的开发人员帮助链接，以及器件数据手册和评估板文档的链接。查看此页面后，单击 **Download**（下载）按钮即可下载该项目和代码。

下载项目后，将其解压缩到计算机上的 MPLABXProjects 文件夹中。MPLABXProjects 可以在您的用户目录下找到，其路径如 [Help>About](#)（帮助>关于）窗口中所示。

4.1.4.1.4 设置项目属性和版本号

在 MPLAB X IDE 中打开项目。右键单击项目名称，然后选择“Properties”（属性）。在 Project Properties（项目属性）窗口中（见下图），选择您的配置（免费或 pro 编译器）。

1. 在“Connected Hardware Tool”（连接的硬件工具）下，选择“Simulator”（模拟器）。MPLAB Xpress 开发板不会显示在此窗口中，因为它不是一款调试工具。其编程方法如 [4.1.4.1.5 编程示例代码](#) 中所述。
2. 在“Packs”（包）下，选择最高编号的软件包。这将对应于随 MPLAB X IDE 安装的包版本。
3. 选择最高编号的 XC8 编译器版本。在 www.microchip.com/mplab/compilers 下查找 MPLAB XC8 编译器。
4. 单击 **OK**，关闭窗口。然后调试项目，以确保其编译和运行正常。

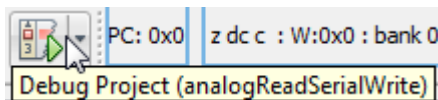
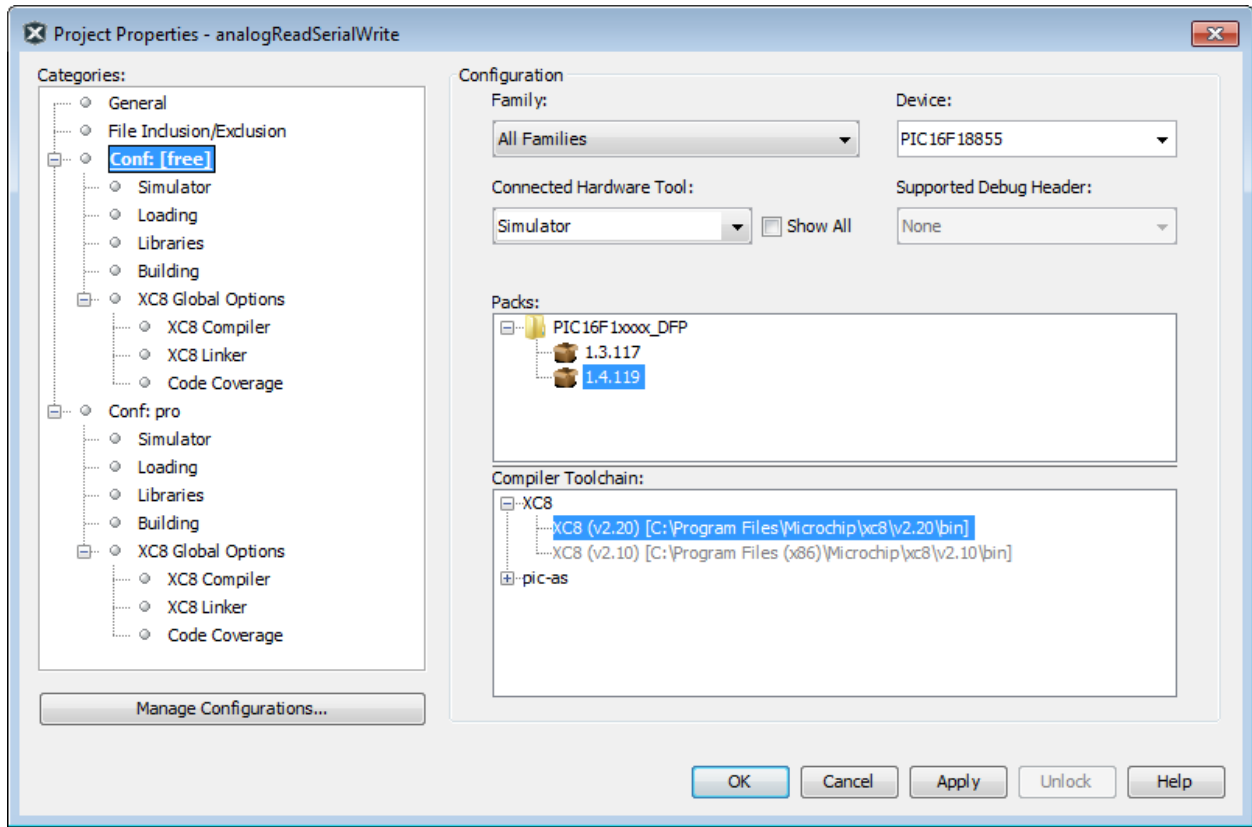


图 4-4. 项目属性窗口



4.1.4.1.5 编程示例代码

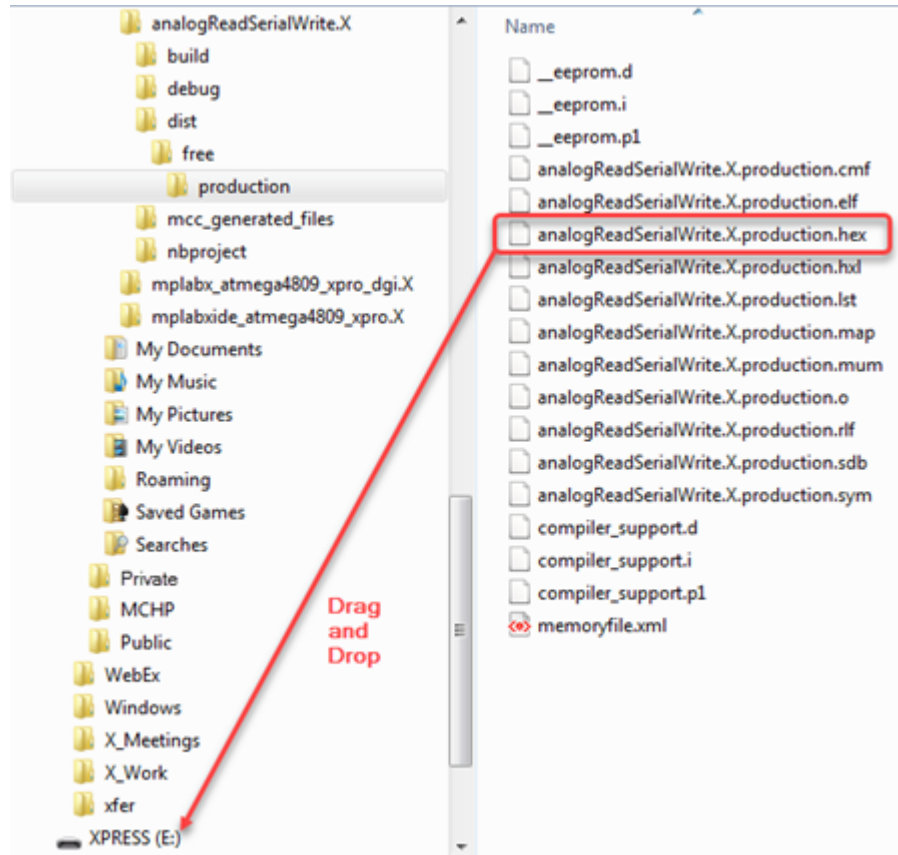
将 MPLAB Xpress 开发板连接到 PC 时，它类似于一个 USB 驱动器。因此，要对板上的 PIC16F18855 器件编程，只需将项目可执行（Hex）文件拖放到板上即可。

在项目文件夹中查找 Hex 文件

在您的 PC 上找到“analogReadSerialWrite.X”项目。MPLABXProjects 可以在您的用户目录下找到，其路径如 [Help>About](#) 窗口中所示。然后，在 dist/free/production/analogReadSerialWrite.X.production.hex 下找到 Hex 文件。

对器件编程

将“XPress”作为 USB 连接设备进行查找。将 Hex 文件拖放到“XPress”开发板上，对其进行编程。



4.1.4.1.6 打开 MPLAB 数据可视化器

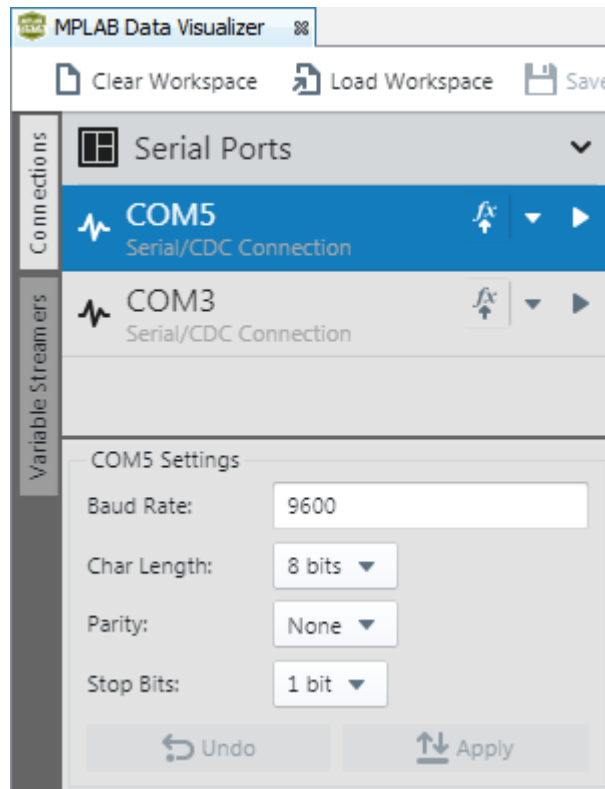
双击桌面上的图标，打开 MPLAB 数据可视化器。



在数据源窗格的 **Connections** 选项卡中，您会看到“Serial Ports”（串行端口）。在其下方显示了可用的 COM 端口。从 [4.1.4.1.2 设置 MPLAB Xpress 评估板](#)中，我们已知 MPLAB Xpress 评估板的 COM 端口。单击选中该 COM 连接。

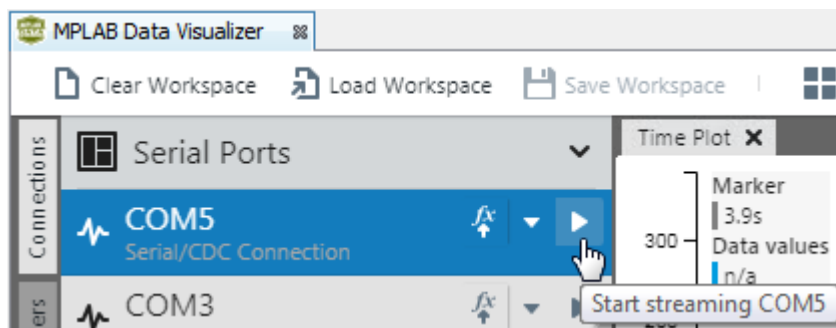
有关 COM 设置的更多信息，请参见 [3.2 串行端口](#)。

图 4-5. 单击以高亮选中 COM 连接



单击右侧的箭头以启动数据流传输。在图形或 Terminal 中尚未显示任何输出。

图 4-6. 单击以启动流传输



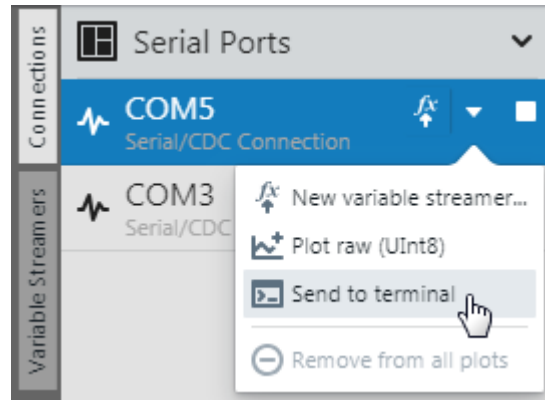
4.1.4.1.7 在 Terminal 中查看示例输出

在 main.c 的 while 循环中，有这样一行代码：

```
printf("\n\rADCC Value is: %i   ", ADCC_GetSingleConversion(POT));
```

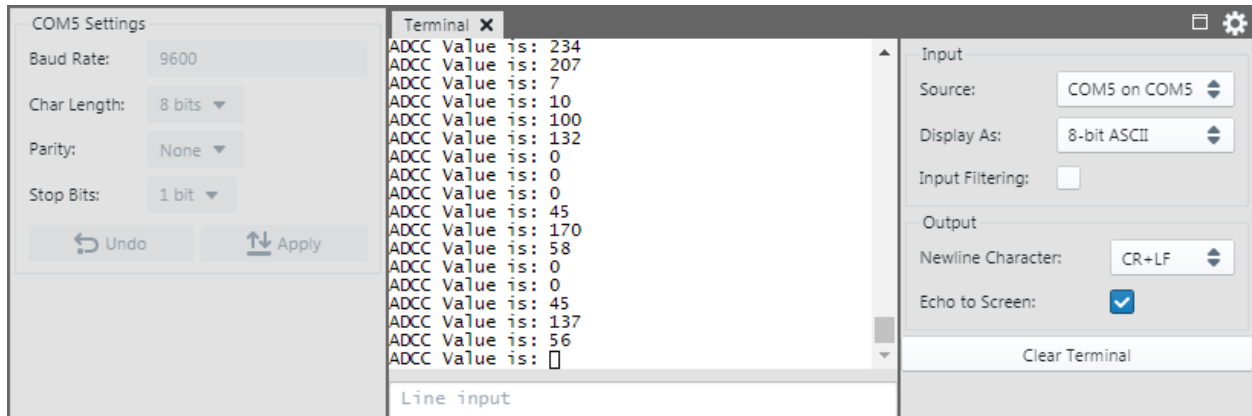
该语句将输出可在 Terminal 窗口中查看的数据。要查看此数据，单击向下箭头并选择“Send to Terminal”（发送到终端）。

图 4-7. 选择 Send to Terminal



您会看到 printf 语句的输出。改变评估板上的电位器，查看值的变化。

图 4-8. Terminal 窗口中的输出



4.2 在 Time Plot 中查看数据

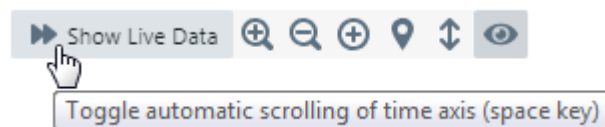
Time Plot 是一个多功能图形绘制工具。大绘图区具有一个时间轴和一个或多个数值轴（Y 轴）。数值轴可以彼此重叠，也可以分成多个图形。

4.2.1 Time Plot 窗口

Time Plot 窗口 显示数据与时间的关系图，并提供用于数据分析的工具。使用左侧窗格中的 **Connections** 选项卡选择数据源。使用 **Variable Streamers** 选项卡以使用数据流协议。使用图形可视化控件（右侧）窗格以选择轴选项并添加多个源轴。

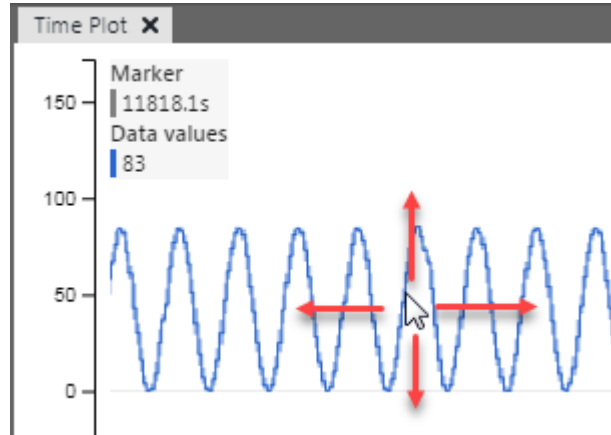
4.2.1.1 图形滚动

如果图形滚动 未激活，即使已使能流传输，仍有可能看不到数据。将鼠标悬停在图形顶部，单击“Show Live Data”（显示实时数据）可启动图形滚动。一旦选中，该文本将变为“Pause Scrolling”（暂停滚动），可停止图形滚动。



或者，双击或单击空格键，可停止数据滚动（尽管数据流传输继续在后台运行）。再次双击或单击空格键即可恢复时间轴滚动。

要手动移动图形，单击并拖动该图形。单击图形的左侧，可拖动并暂停数据轴。单击图形或窗格中央，可拖动并暂停时间轴。



4.2.1.2 放大和缩小

单击轴 以使用以下控件：

- 使用鼠标滚轮放大或缩小轴范围。图形大小会相应调整。
- 使用鼠标滚轮放大或缩小图形区域。
- 将鼠标悬停在轴上，即可看到放大和缩小选项。
- 将鼠标悬停在图形顶部，可以放大或缩小图形。

图 4-9. 轴缩放

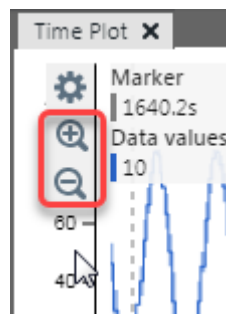


图 4-10. 图形缩放



4.2.1.3 添加数据轴

要添加另一个数据轴，将鼠标悬停在图形顶部，然后选择添加轴图标。



在可视化控件（右侧）窗格中设置轴属性，或者将鼠标悬停在该轴上，然后单击“Axis Options”（轴选项）。可视化控件是添加数据轴的另一种可选方法。两种方法都可提供向轴添加图形的控件。

图 4-11. 可视化控件——Time Axis（时间轴）

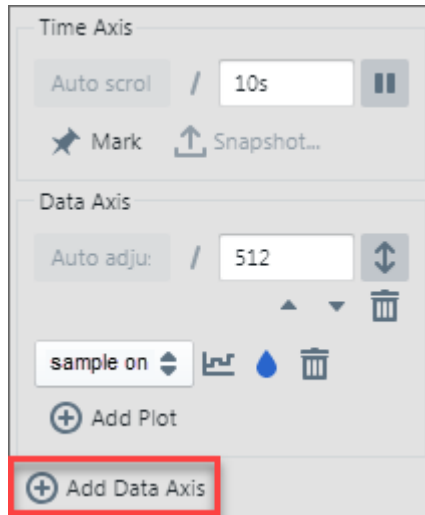
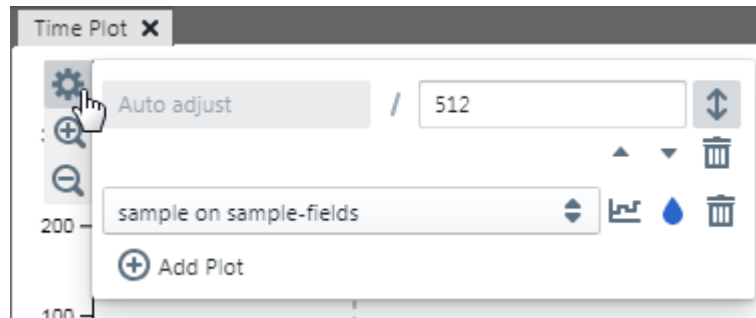
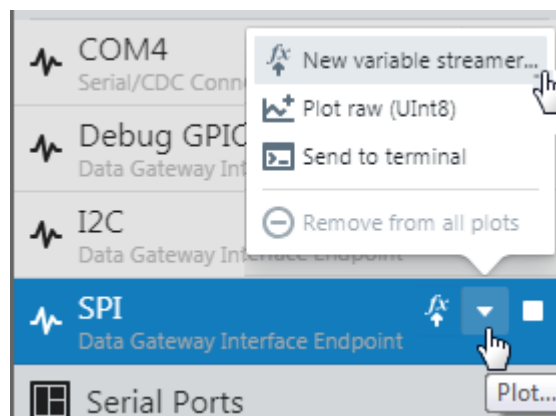


图 4-12. 轴选项



此外，设置轴后，就可以通过在数据源窗格中单击所选数据源上的“Plot”（绘制）按钮绘制图形。

图 4-13. 数据源——Plot



4.2.1.4 查看图形数值

MPLAB 数据可视化器提供以下内置工具，用于查看和分析图形数据。

4.2.1.4.1 图形标记

时间图标记 是一条垂直虚线，它跟随鼠标沿时间轴移动。标记的当前位置的时间和数据值显示在图形的左上角。要切换该显示，将鼠标悬停在图形顶部，单击“Inspect Values”（检查值）图标。

图 4-14. 标记和检查值显示

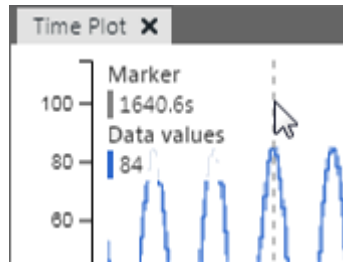


图 4-15. 切换检查值显示



4.2.1.4.2 图形光标

垂直光标 的使用方式类似于图形标记，只是光标不跟随鼠标移动；必须将其拖动到某个位置，在下次拖动前，它不会移动。“Inspect Values”（检查值）显示光标值而不是标记值。

图 4-16. 添加垂直光标



图 4-17. 垂直光标 A

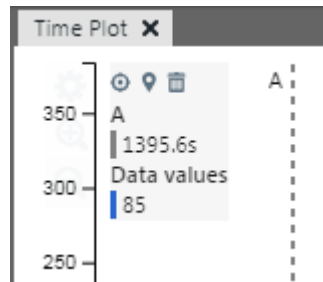


表 4-3. 检查值显示

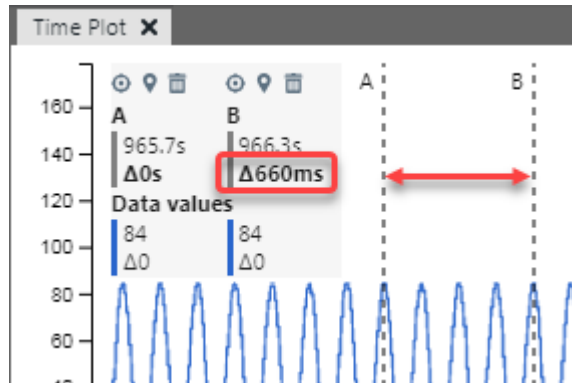
	跳转到光标。光标将在图形上居中，并且暂停滚动。如果光标已随图形滚动偏离，则跳转到其位置。
	将光标重新定位到图形的中心。重新定位时，按住 Shift 键即可使所有光标移动相同的量。
	删除光标。
A、B 和 C 等。	光标的字母编号。
Time Values（时间值）	对于单个光标，将显示光标与时间轴相交的值（时间戳）。 对于两个或多个光标，将显示时间差（增量），以最左边的光标为参考。
Frequency Value（频率值）	对于两个或多个光标，将鼠标悬停在时间值上会看到一个频率值，以最左边的光标为参考。
Data Values（数据值）	对于单个光标，将显示光标与数据图形相交的值。如果有多个图形，则与图形颜色相对应的色条将表示关联的数据值。 对于两个或多个光标，将显示数据差（增量），以最左边的光标为参考。

对带宽使用两个光标

可使用两个垂直光标来确定带宽。使用时间增量，以下图为例，位置 A 和位置 B 之间的时间差为 660 ms。因此，带宽为 $660 \text{ ms} / 4 \text{ 个周期} \cong 165 \text{ ms/周期}$ 。

注：由于各周期之间可能存在差异，通常最好测量几个周期的时间以提供平均值。

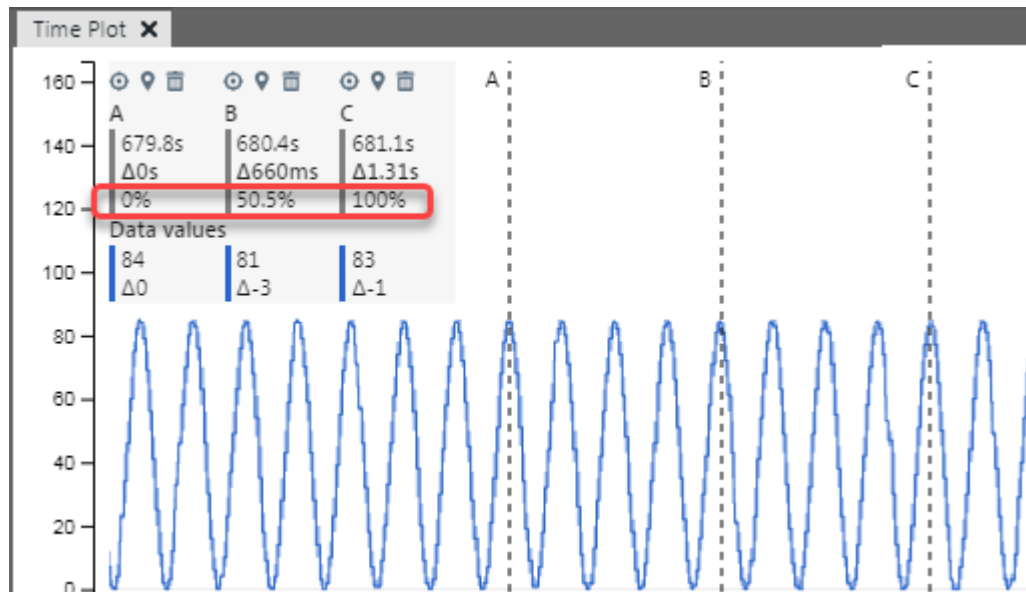
图 4-18. 添加两个垂直光标——带宽



对占空比使用三个光标

添加第三个光标将能够计算占空比。如果 A-C 为周期，则 A-B 将显示为该周期的百分比（50.5%）。

图 4-19. 添加三个垂直光标——占空比



可以在图形中添加更多光标。

4.2.1.5 自动调整数据轴

要切换所有数据轴的自动调整，将鼠标悬停在图形顶部，然后选择自动调整图标。此外，双击某个轴也会切换该轴的自动调整。



图 4-20. 自动调整前

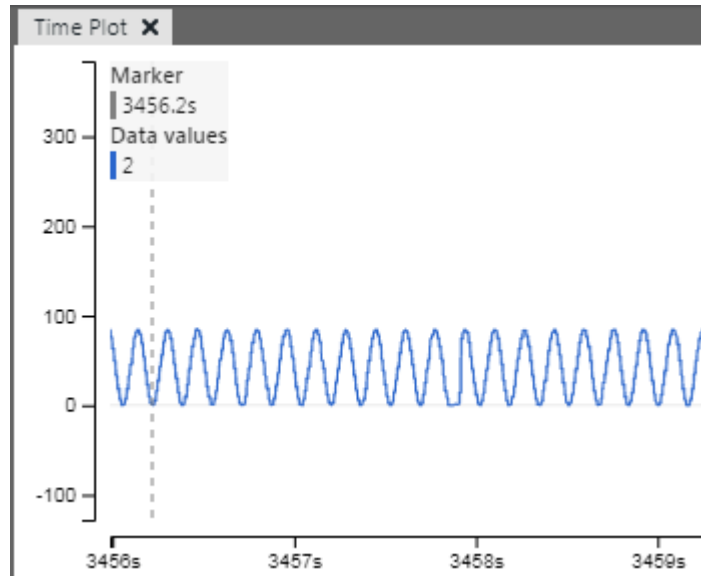
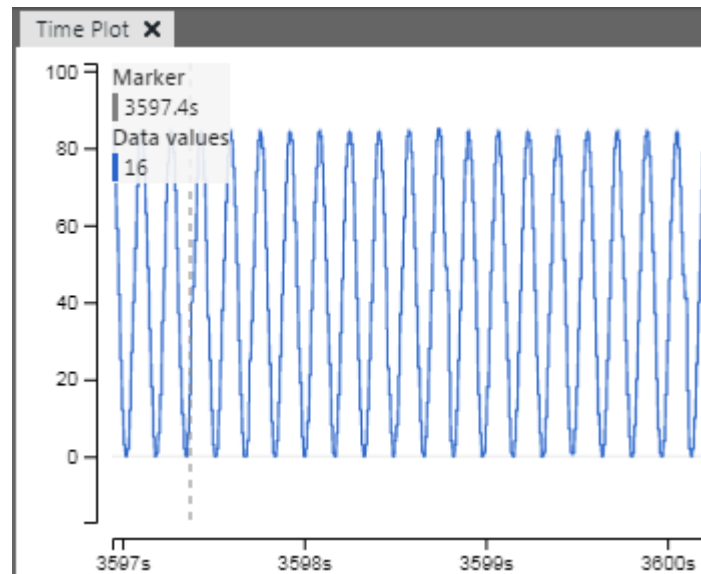


图 4-21. 自动调整后





4.2.2 Connections 选项卡

单击 **Connections** 选项卡，查看可用的连接。将目标板连接到 PC 后，将显示该连接并列出板上数据源。

要选择一个数据源，例如“SPI”，单击该数据源，即可对其设置进行编辑。如果已连接并希望更改设置，则可以单击 **Stop Streaming**（停止流传输）按钮以断开连接。再次单击该按钮则重新连接。

有关连接的详细信息，请参见 3. 外部连接。

按钮	操作
	Start/stop streaming ——切换按钮：开始/结束来自数据源的数据流传输。图形和 Terminal 窗口将继续滚动。
	




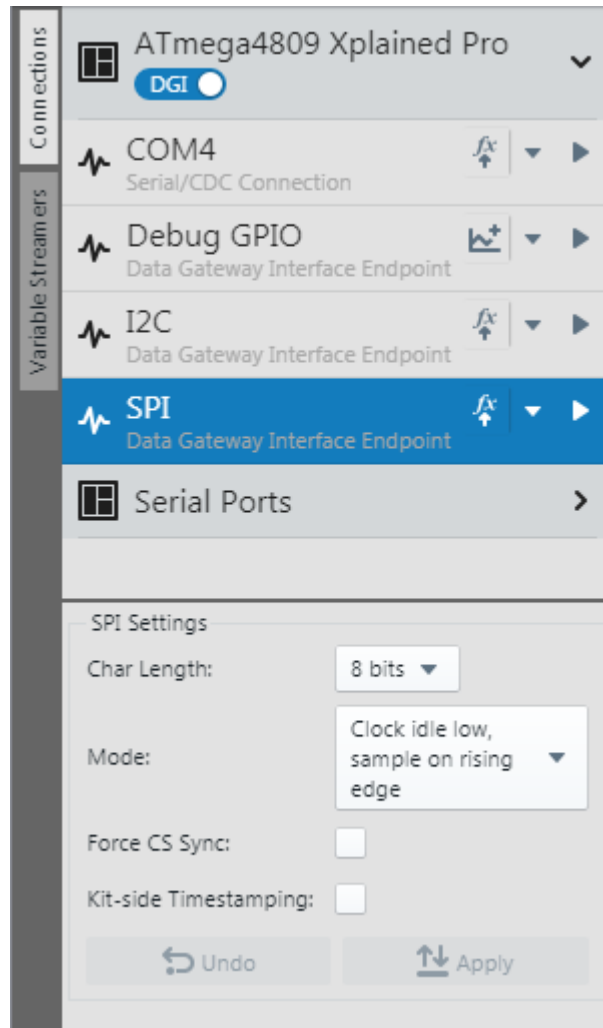
..... (续)	
按钮	操作
	Plot——根据数据源选择绘制类型：绘制原始数据、绘制所有引脚、绘制变量流设置器或将数据发送到 Terminal。此外，还可以移除所有图形中的数据源。
	New Variable Streamer——打开向导以设置新的变量流设置器。
	Plot All Pins——在图形上绘制所有 GPIO 引脚。

图 4-22. Connections 选项卡



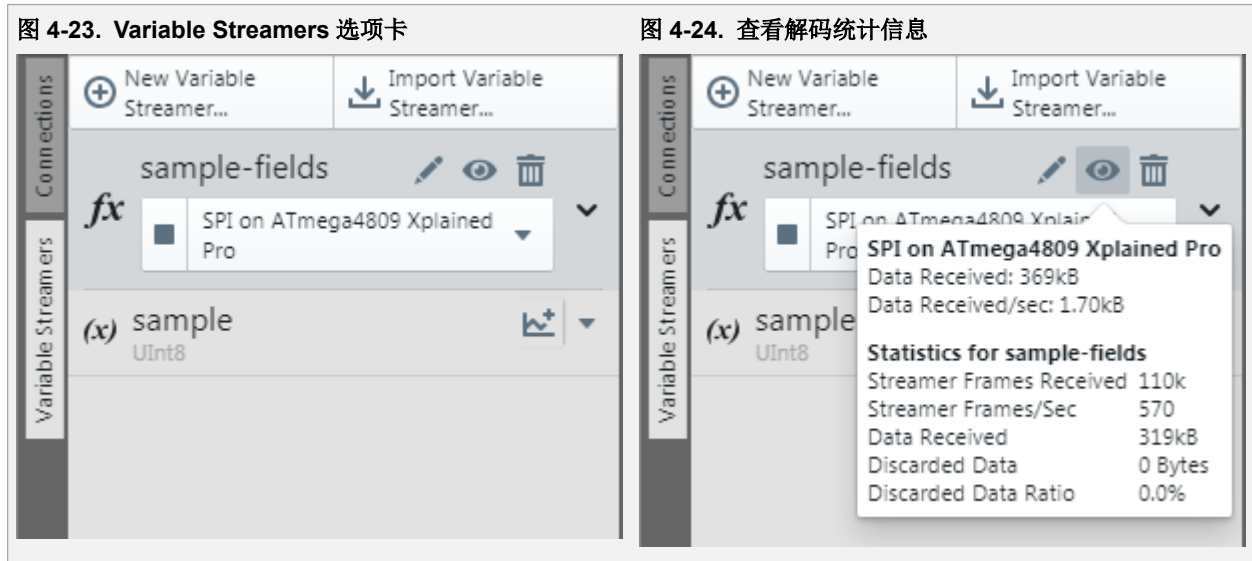
4.2.3 Variable Streamers 选项卡

单击 **Variable Streamers** 选项卡以创建、编辑或删除变量流设置器。**Import Variable Streamer**（导入变量流设置器）是一种导入 Atmel Studio 数据可视化器所用的 .ds 或 .txt 文件的方法。有关文件格式的更多详细信息，请参见下文的第 5.1.1 节“配置格式”。

ww1.microchip.com/downloads/Secure/en/DeviceDoc/40001903B.pdf

要实时查看解码统计信息和流传输数据信息，单击眼睛图标。

当数据流中有多个变量时，必须先对数据流进行解码，然后才能可视化这些变量。有关详细信息，请参见 5. 变量设置器。



4.2.4 时间图可视化控件（右侧）窗格

时间图可视化控件（右侧）窗格 用于控制流传输数据的可视化（图形化）。

图 4-25. 图形可视化控件

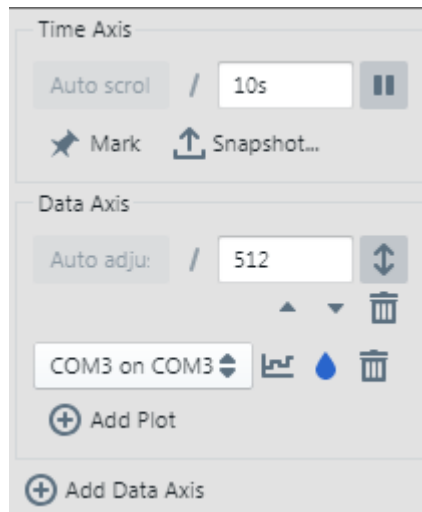


表 4-4. Time Axis

控件	说明
Offset（偏移量）	滚动时，显示“Auto scroll”（自动滚动）。 暂停时，在图形右侧显示时间。
Scale（缩放）	指定时间轴的分辨率，单位为秒。
	暂停滚动或开始/继续滚动。
	冻结当前的 Time Plot 窗口，并在边界处使用光标标记。根据需要调整光标。

..... (续)

控件	说明
 Snapshot...	将数据源图形的快照保存为文本文件（CSV 或 JSON）。拍摄图形快照前使用“Mark”（标记）。

表 4-5. Data Axis




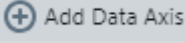



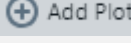
控件	说明
Offset	使能自动调整时，将显示“Auto adjust”。 禁止自动调整时，将显示当前的图形偏移量。
Scale	指定数据轴的分辨率，单位为秒。
	自动调整使能/禁止。 使能时，自动调整轴的范围。 禁止时，手动调整轴的范围。
	如果有多个数据轴，可相对于其他轴向上或向下移动此轴。
	单击以从图形中删除此轴。
 Add Data Axis	在当前轴下方为图形添加另一个数据轴。

表 4-6. Data Axis——绘制数据源和格式

控件	说明
Data Source（数据源）	从下拉菜单中选择要绘制的数据源。请参见数据源窗格的选择和设置。
	单击以选择数据点在图形上的显示方式。 <ul style="list-style-type: none"> • 使用阶梯线连接点（默认） • 使用直线连接点 • 仅画点，不连线
	单击以选择图形颜色。
	单击以删除此图形。
 Add Plot	在当前轴上添加要绘制的另一个数据源。

4.2.5 Time Plot 示例

以下是 Time Plot 使用示例。

4.2.5.1 原始数据流传输示例，第 2 部分

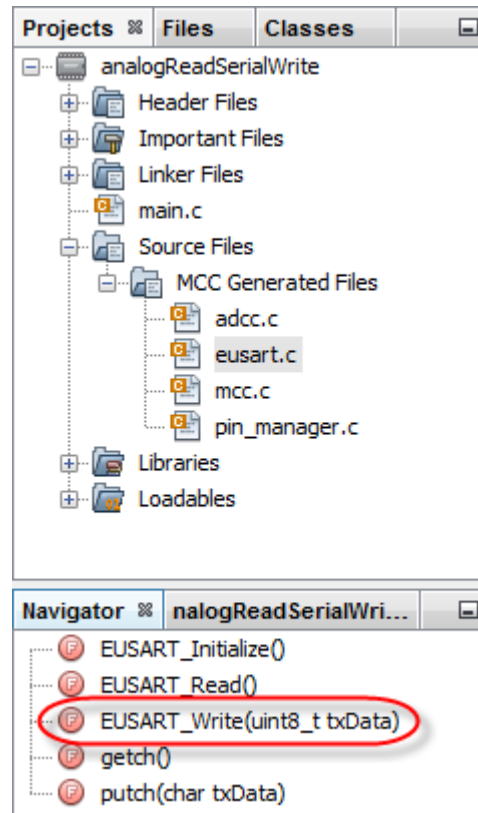
可以修改此终端示例，以将数据输出到时间图。有关设置的详细信息，请参见 [4.1.4.1 原始数据流传输示例](#)。有关修改信息，请参见以下主题。

4.2.5.1.1 修改示例代码

为了绘制下载的 MPLAB Xpress 代码示例中的电位器 ADC 数据的图形，将使用 PIC16F18855 上的 EUSART 按字节发送数据，因为 USART 是 MPLAB 数据可视化器支持的格式。

使用 EUSART 函数

在示例项目中，提供了用于一个 EUSART 的代码以实现此目的。在 **Projects**（项目）树中，单击 `eusart.c`。在此文件中，将使用 `EUSART_Write()` 函数。



在 `main.c` 的 `while` 循环中，添加一行 `EUSART_Write()` 代码：

```
EUSART_Write(ADCC_GetSingleConversion(POT));
```

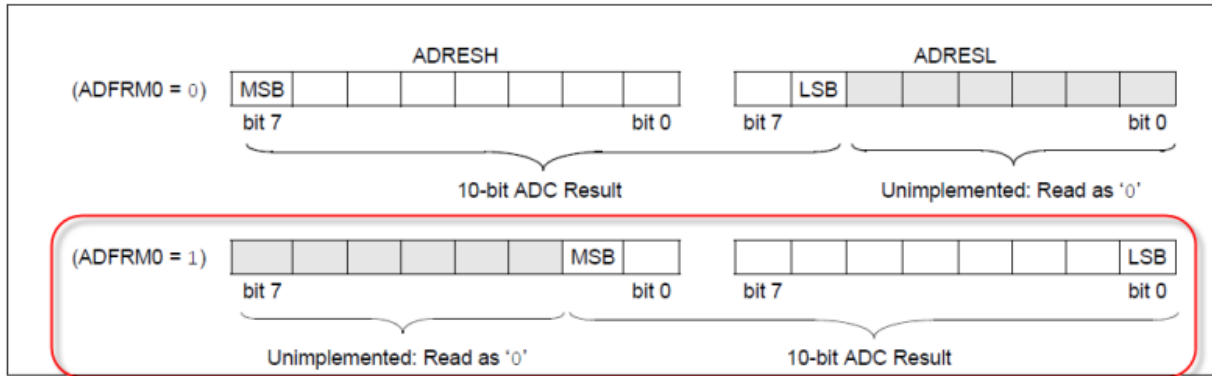
不过，这将无法正常运行，因为 ADC 产生的是 10 位结果，而 EUSART 写函数需要的是 8 位值。因此，需要对该值进行调整。

根据 PIC16F18855 数据手册，ADC 结果可采用两种格式（见下图）。具体采用的格式取决于 `ADCON0` 寄存器的 `ADFRM0` 位的值，其中“1”表示数据右对齐，“0”表示数据左对齐，并用零进行填充。查看 `adcc.c` 文件，以查看如下关于 `ADCON0` 寄存器的信息，其中 `ADFM = ADFRM0`：

```
// ADGO stop; ADFM right; ADON enabled; ADCONT disabled; ADCS FOSC/ADCLK;  
ADCON0 = 0x84;
```

因此，本示例中采用下图中所示的第二种 ADC 格式。

FIGURE 23-3: 10-BIT ADC CONVERSION RESULT FORMAT



所以，应修改 EUSART 写函数的输入，以使用最高 8 位：

```
EUSART_Write((ADCC_GetSingleConversion(POT) & 0x03FC)>>2);
```

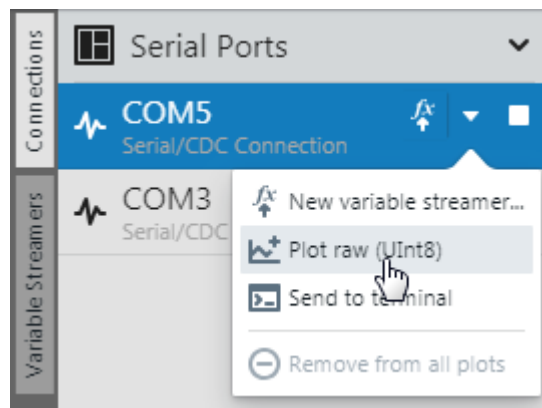
调试修改后的代码

调试运行新代码，确保其能够执行。如果出现错误，请查看上述说明。

4.2.5.1.2 可视化示例

重新编程 MPLAB Xpress 评估板以通过 EUSART 获得数据。要查看此数据，单击向下箭头，然后选择“Plot raw (UInt8)”（绘制原始数据 (UInt8)）。

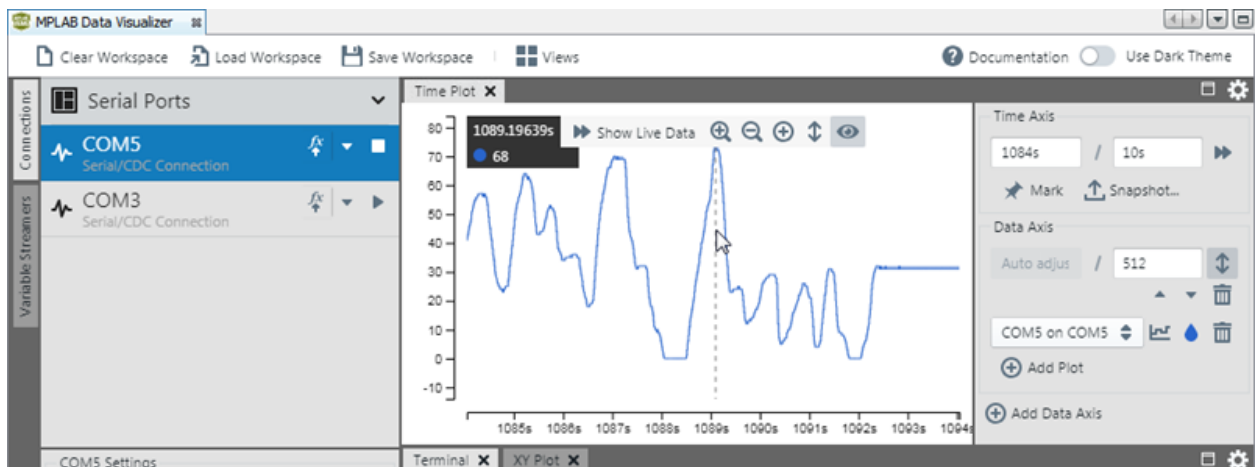
图 4-26. 选择绘制原始数据



您现在会看到传入的数据显示在中心窗格的图形上。在 MPLAB Xpress 评估板上 来回旋转电位器，查看数据图形的变化。

Time Axis 中还有其他选项可用于设置数据范围、偏移量或其他图形格式。有关更多信息，请参见 4. 可视化。

图 4-27. 原始数据的图形



相关信息

[4.2.4 时间图可视化控件（右侧）窗格](#)

[4.2.1 Time Plot 窗口](#)

4.2.5.2 绘制数据示例

ATmega4809 Xplained Pro 板 用于生成数据，该数据通过 DGI/SPI 输出到可视化器，并使用变量流设置器进行绘制。

本示例将使用的工具包括：

- MPLAB X IDE v5.40（或更高版本）
- MPLAB XC8 C 编译器 v2.20（或更高版本）
- ATmega4809 Xplained Pro 板——ATMEGA4809-XPRO

4.2.5.2.1 示例设置

请按照以下说明完成操作，以设置示例软件和硬件。

MPLAB X IDE

访问以下链接，免费下载并安装 MPLAB X IDE 5.40（或更高版本）。

www.microchip.com/mplab/mplab-x-ide

MPLAB XC8 C 编译器

访问以下链接，免费下载并安装 MPLAB XC8 C 编译器 v2.20（或更高版本）。还可以购买包含更多优化和功能的 PRO 版编译器。

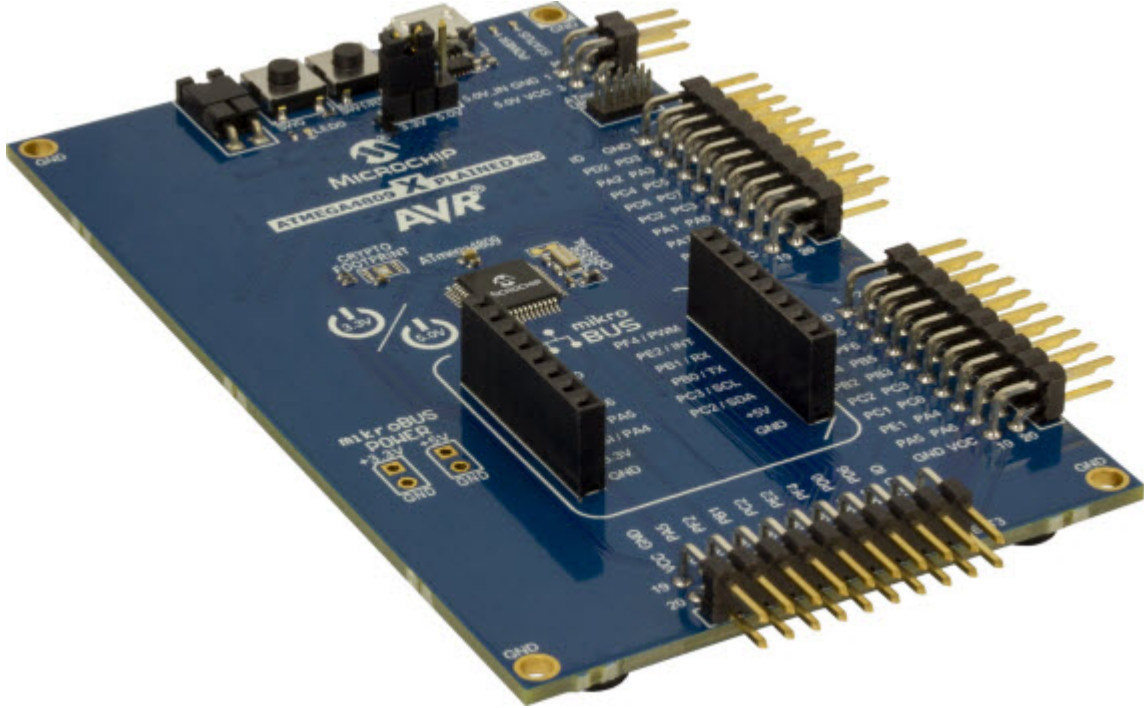
www.microchip.com/mplab/compilers

ATmega4809 Xplained Pro 板——ATMEGA4809-XPRO

此评估板（见下图）可通过 Microchip 直销网站（<https://www.microchipdirect.com/>）或代理商购买。然后使用随附的 USB 线缆，将评估板连接到计算机，并安装驱动程序。

有关此评估板的更多信息，请访问：

www.microchip.com/developmenttools/ProductDetails/PartNo/ATMEGA4809-XPRO



4.2.5.2.2 创建示例项目

MPLAB X IDE 需要使用项目来开发应用程序代码。

前期准备

在创建项目之前，请确保：

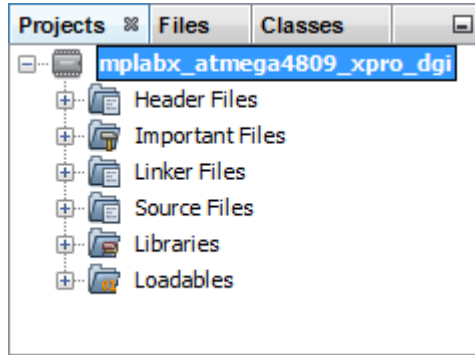
- 您已经安装了编译器，并且 MPLAB X IDE 可以检测到它。如果尚未安装，请转到 [Tools>Options>Embedded>Build Tools](#) (工具>选项>已安装工具>编译工具)，查看工具链列表。如果编译器不在列表中，单击 **Add** (添加) 以浏览并添加该编译器。
- 您已使用 USB 线缆将 Xplained Pro 板插入计算机。

创建项目

选择 **File>New Project** (文件>新建项目) 或 **New Project** (新建项目) 图标 ，打开项目向导。按照以下步骤创建项目。单击 **Next** (下一步) 转到下一步。

1. **选择项目：**单击“Microchip Embedded”（Microchip 嵌入式）类别，然后选择“Standalone Project”（独立项目）项目。
2. **选择器件（和工具）：**输入器件“ATmega4809”。然后输入工具“ATmega4809 Xplained Pro-SN: ATML#”，其中工具序列号（serial number, SN）包含前缀“ATML”，后跟一个多位数字。
3. **选择编译器：**在 [Compiler Toolchains>XC8](#)（编译器工具链>XC8）下，选择编译器的最新版本。
4. **选择项目名称和文件夹：**为您的项目命名。例如，“mplabx_atmega4809_xpro_dgi”。对于 Windows 操作系统，默认项目文件夹为 C:\Users\<\UserName>\MPLABXProjects。

单击 **Finish** (完成) 后，项目树会出现在 **Projects** (项目) 窗口中。



4.2.5.2.3 将文件添加到项目中

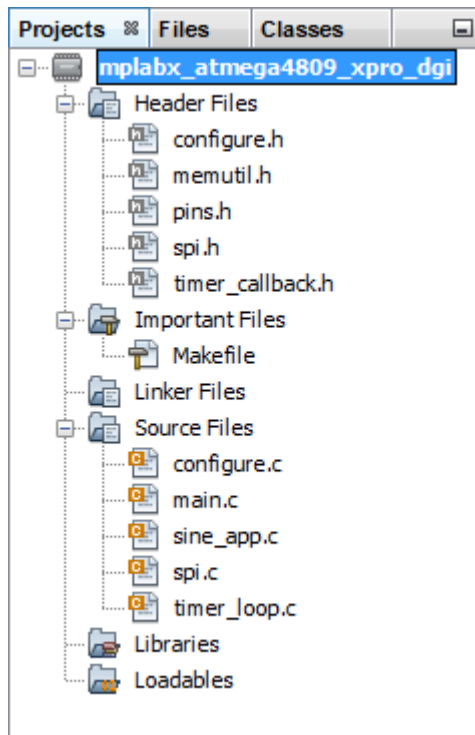
要将 C 头文件添加到项目中，右键单击“Header Files”（头文件）文件夹，然后选择 **New>C Header File**（新建> C 头文件）。

要将 C 源代码添加到项目中，右键单击“Source Files”（源文件）文件夹，然后选择 **New>C Main File**（新建>C 主文件，仅一次，适用于 main.c）或 **New>C Source File**（新建> C 源文件，适用于所有其他文件）。

该项目的示例代码参见 [8. 绘制数据示例——代码清单](#)。

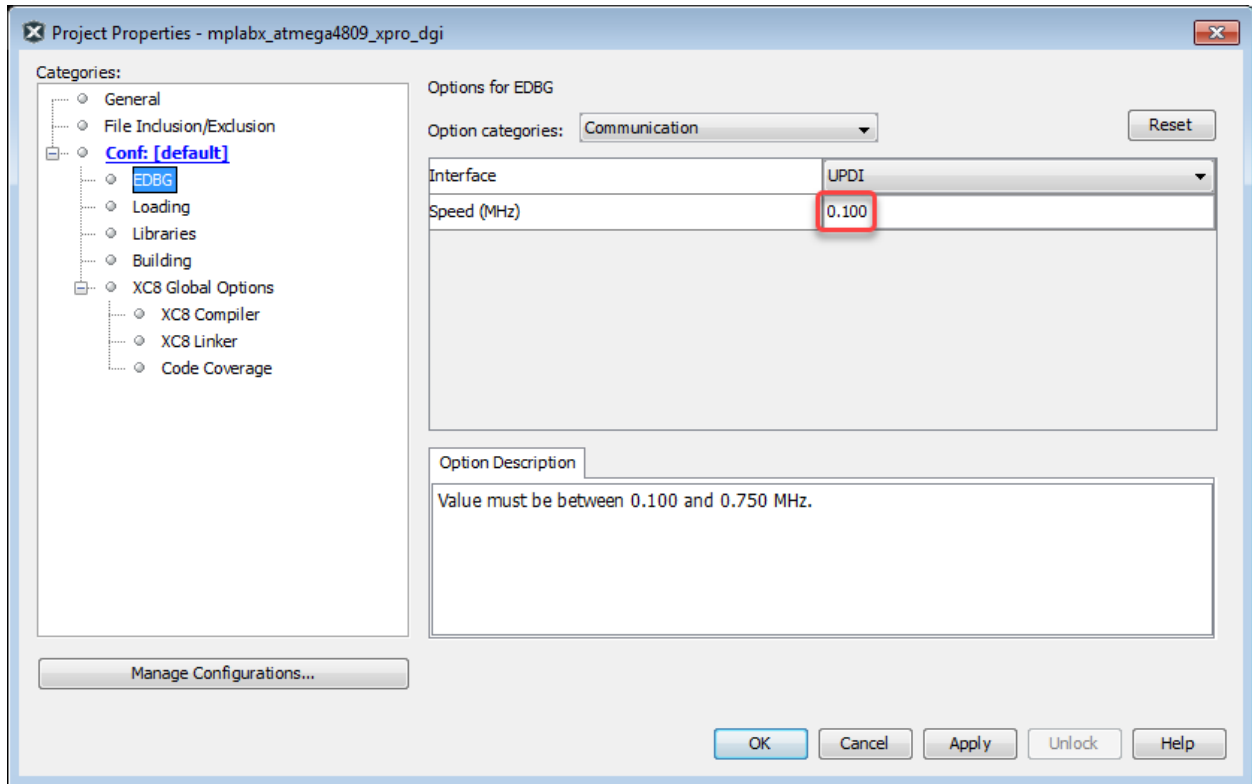
完成的项目应如下图所示。

图 4-28. 包含文件的项目




4.2.5.2.4 设置项目通信选项

右键单击 Projects 窗口中的项目名称，以打开 Project Properties（项目属性）。在 EDBG 的 Communication（通信）选项下，将 Speed（速度）设为“0.100”，以与代码中的设置匹配。




4.2.5.2.5 打开 MPLAB 数据可视化器

单击工具栏图标  或选择 **Tools>Embedded>Data Visualizer** (工具>已安装工具>数据可视化器)，以打开 MPLAB 数据可视化器插件。

1. 在数据源窗格的 **Connections** 选项卡中，您会看到“ATmega4809 XPlained Pro”，且已使能“DGI”。
2. 在可视化窗格中，Time Axis 的“Source”（源）框中显示为“No Source”（没有源）。
3. 在图形选项卡中，滑动标记（灰色虚线）大概位于零秒处，且没有数据值（如果有，则应为与源的颜色相匹配的蓝色）。

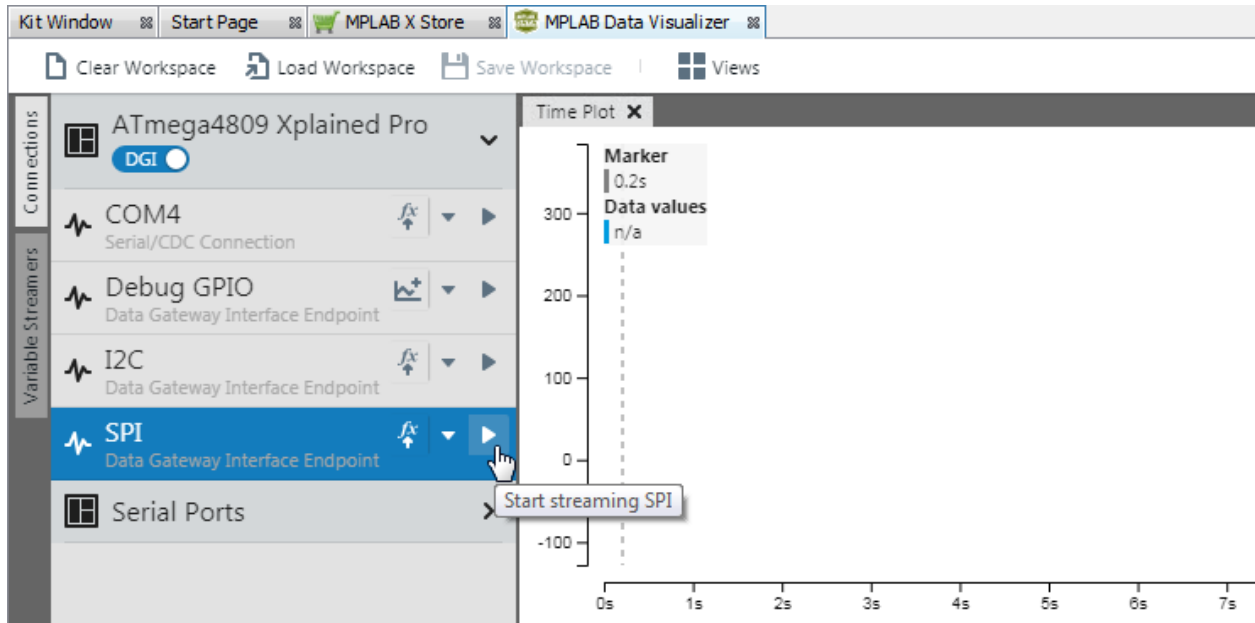


4.2.5.2.6 调试项目并可视化输出

要开始调试项目，单击“Debug Project”（调试项目）图标 。

在 MPLAB 数据可视化器中的“ATmega4809 Xplained Pro”（使能 DGI）下，单击 SPI 按钮的向右箭头控件，以启用 SPI 流传输。

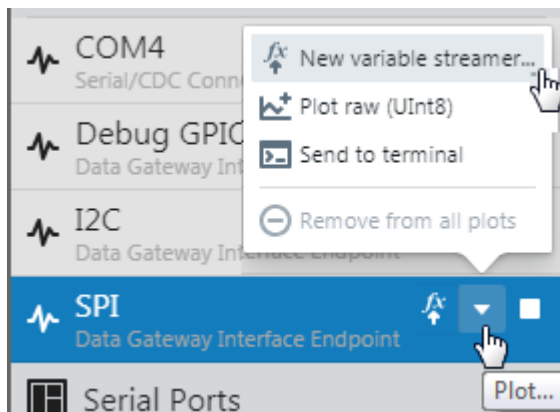
注：单击 SPI 连接时，您会看到下面的“SPI Settings”（SPI 设置）。有关这些设置的具体含义，请参见 [3.1.5 SPI 接口](#)。



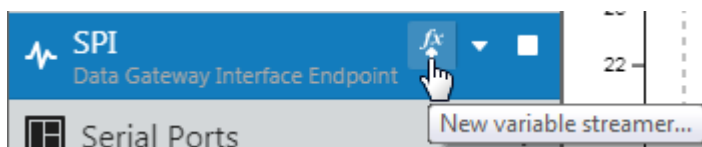
一旦开始流传输，图形顶部会出现一个横幅。您可以单击“Plot”以开始对绘制进行设置。但是，该横幅仅显示几秒钟。



您可以单击 SPI 按钮上的向下箭头（Plot）控件。然后选择要设置的绘制类型。



如果要设置变量流设置器，则可以单击 fx（新建变量流设置器）控件。



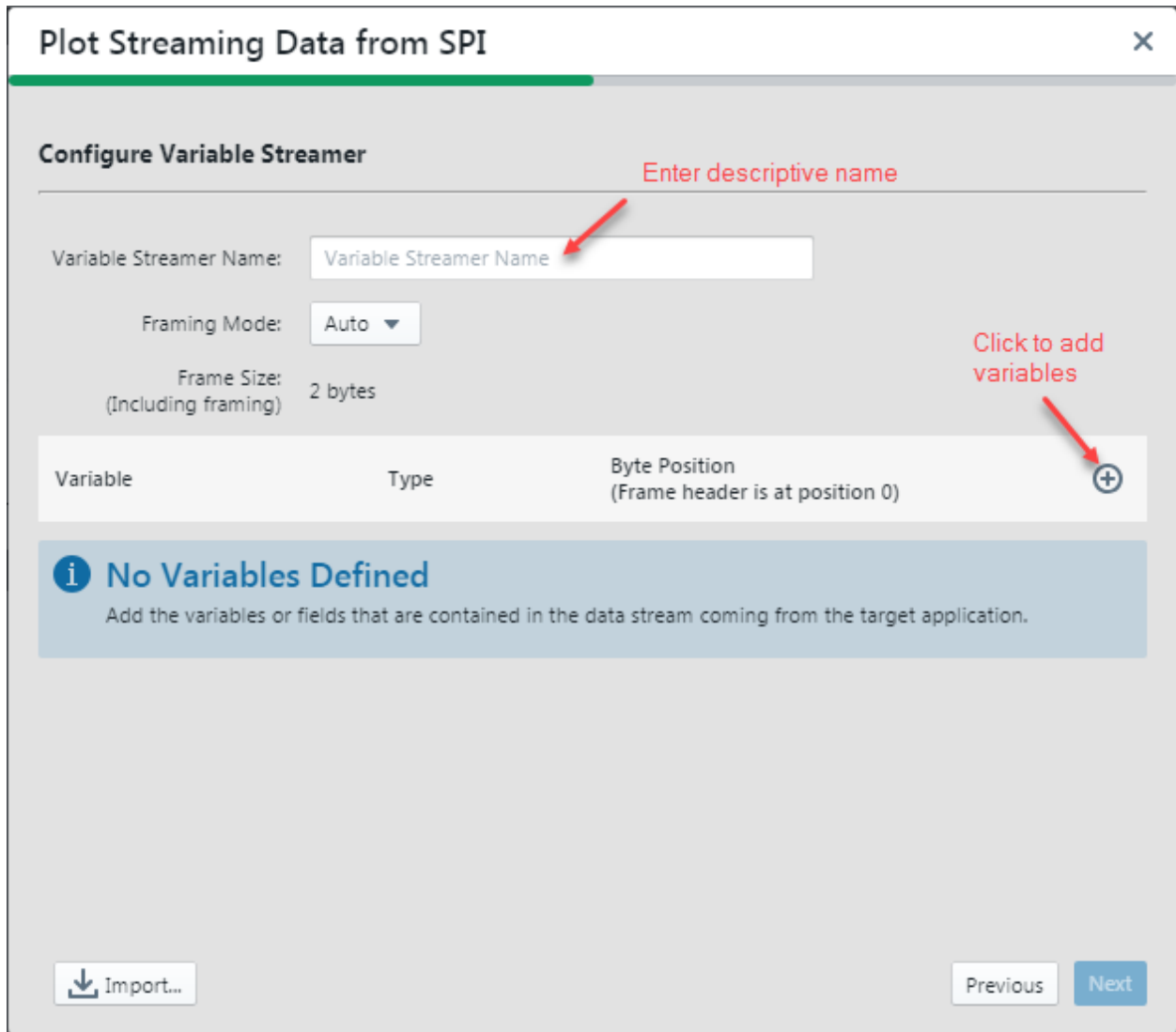
绘制 SPI 的流传输数据

配置变量流设置器——初始视图

在此示例中，将对一个变量值进行流传输并绘制。绘制向导的第一个对话框如下所示。为变量流设置器命名，以便以后识别设置。然后单击“+”以添加变量（如图所示），将显示用于输入的文本框。

注：如果保存了以前的设置，则可以通过单击对话框底部的 **Import**（导入）按钮进行加载。

此外，使用“?”键可打开键盘快捷方式对话框，使用“Esc”键可关闭对话框。

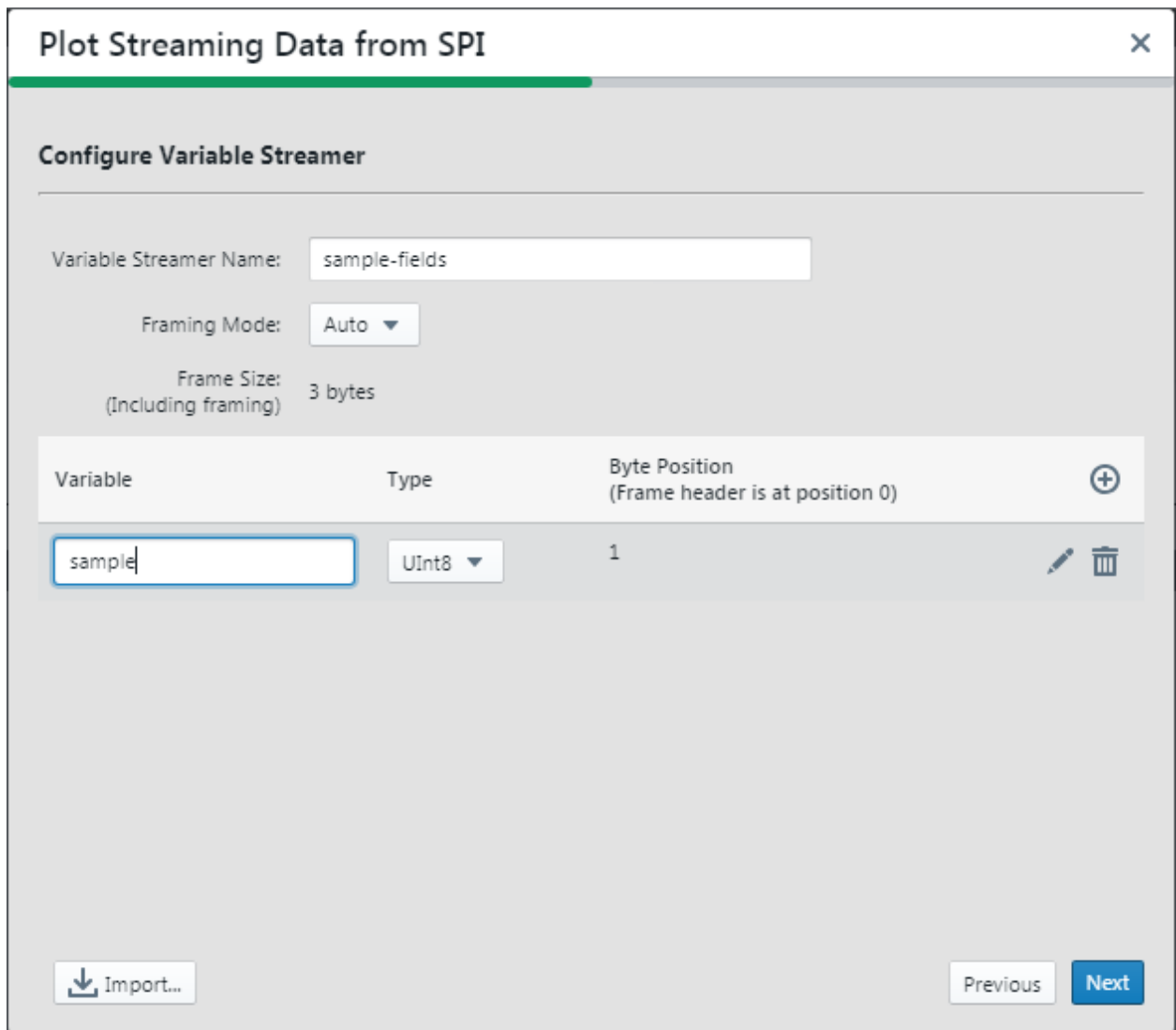


配置变量流设置器——输入数据

上一个对话框中输入数据后如下所示。其中已提供有关变量的指定信息。

为了解码数据流，必须对变量（即数据流中的字段）进行定义。流传输数据的格式如 [5.2 流格式](#) 所示。

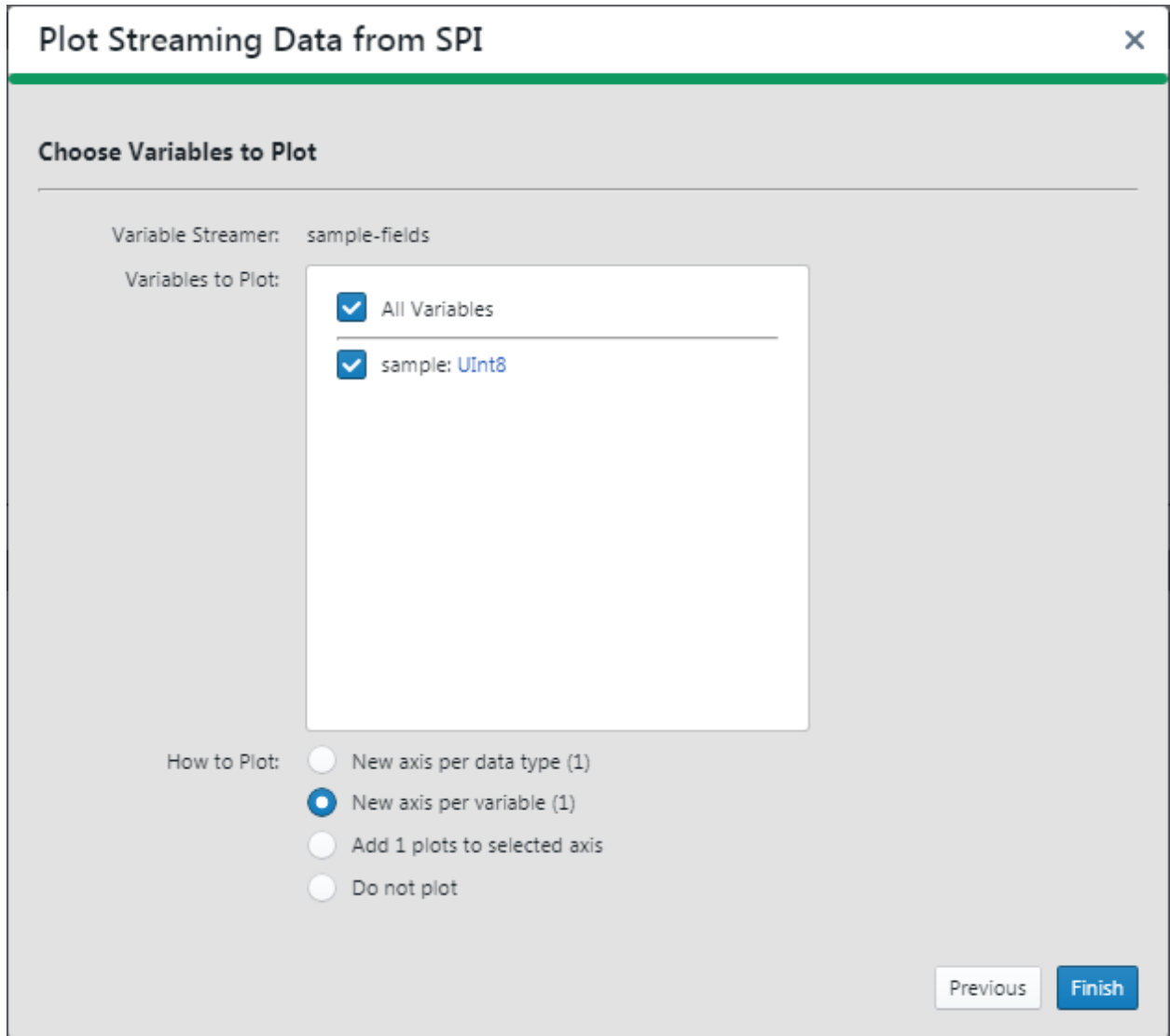
单击 **Next** 继续设置。



配置变量流设置器——如何绘制

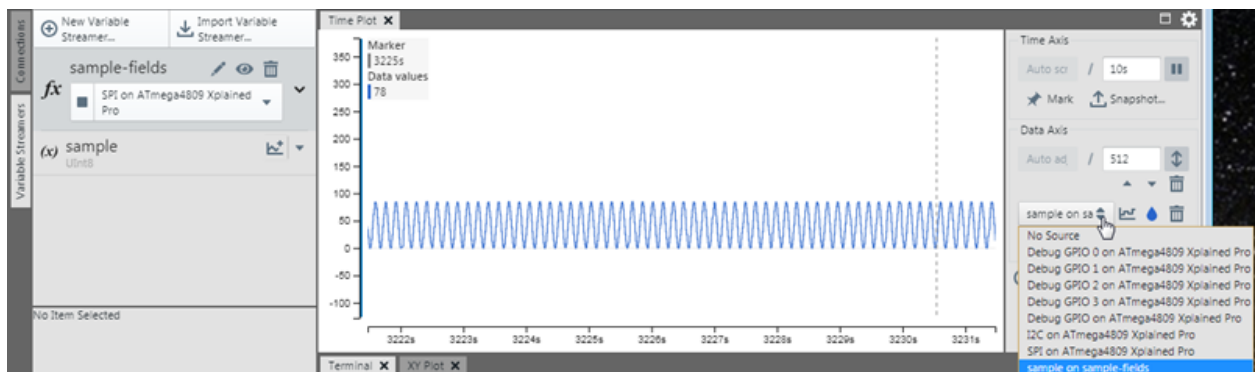
该对话框显示了前一个对话框的摘要以及如何绘制数据的选项列表。在此示例中，选择了“New axis per variable (1)”（每个变量（1）新建一个轴）

单击 **Finish** 即可开始绘制。

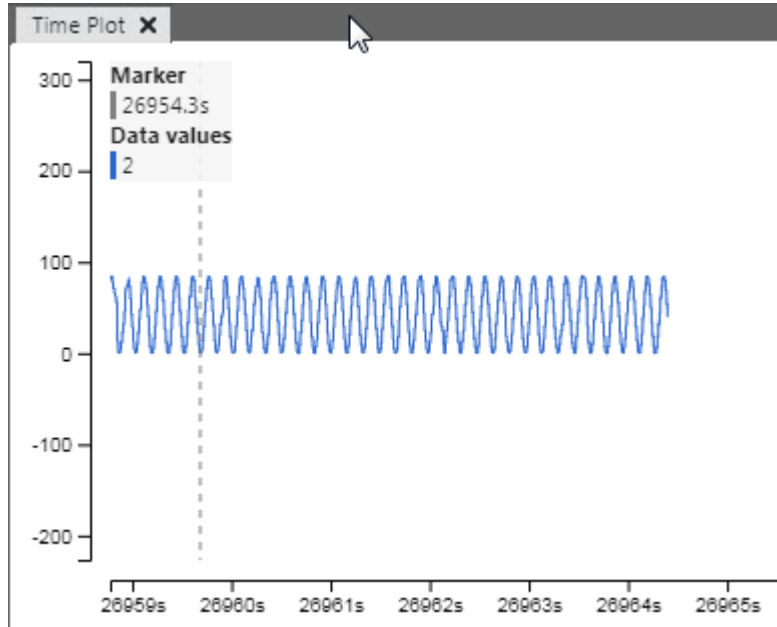



根据绘制设置查看数据

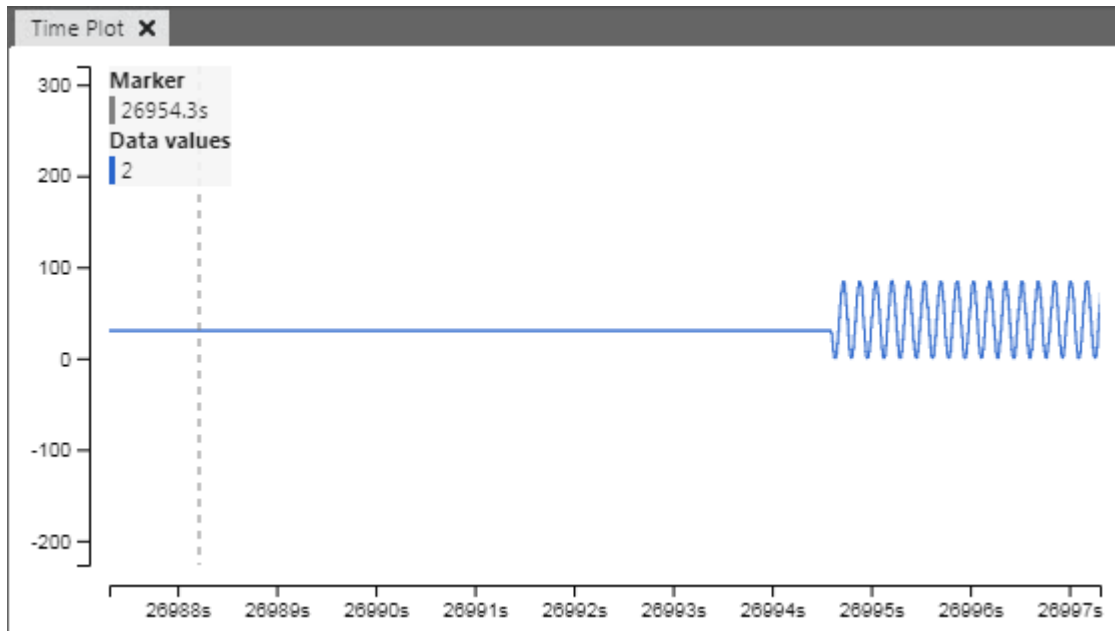
现在应将 Time Axis 中的 Source 设置为“sample on sample-fields”（在采样字段上采样），数据应绘制到图形。要切换图形中的数据滚动，请按图形横幅上的“Pause Scrolling/Show Live Data”（停止滚动/显示实时数据），或者使用空格键。



当暂停调试  时，停止输出数据。



当继续调试  时，数据会再次显示在图形上。

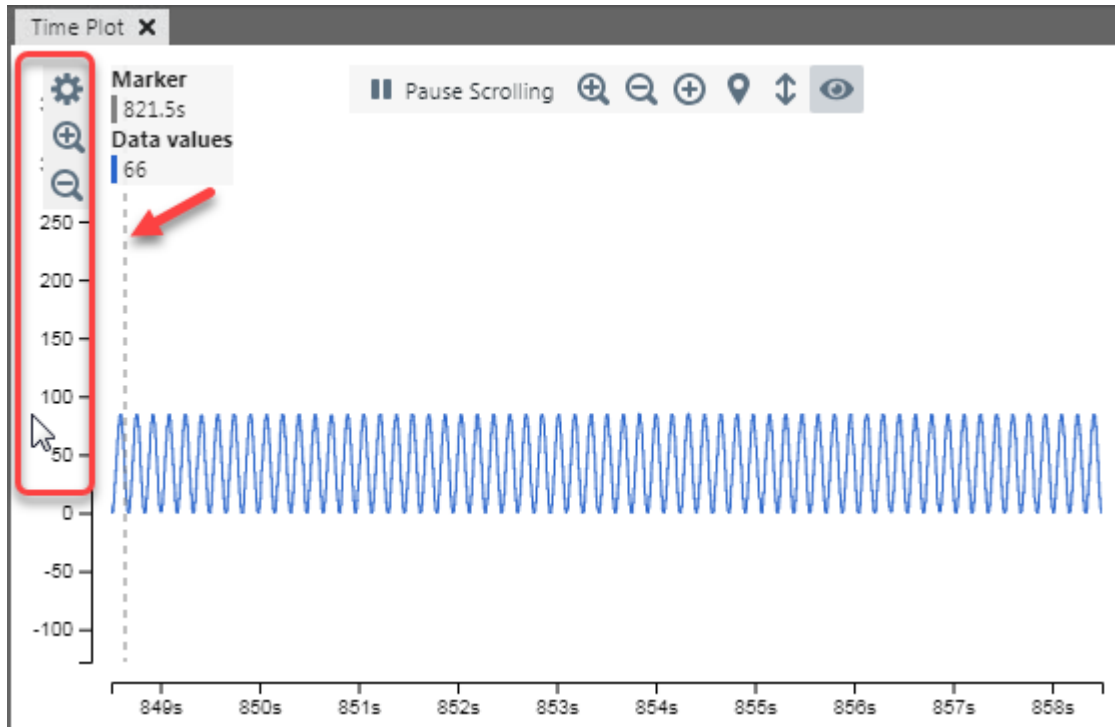


使用工具分析绘制数据

可使用图形工具更改绘制数据的视图并对其进行分析。

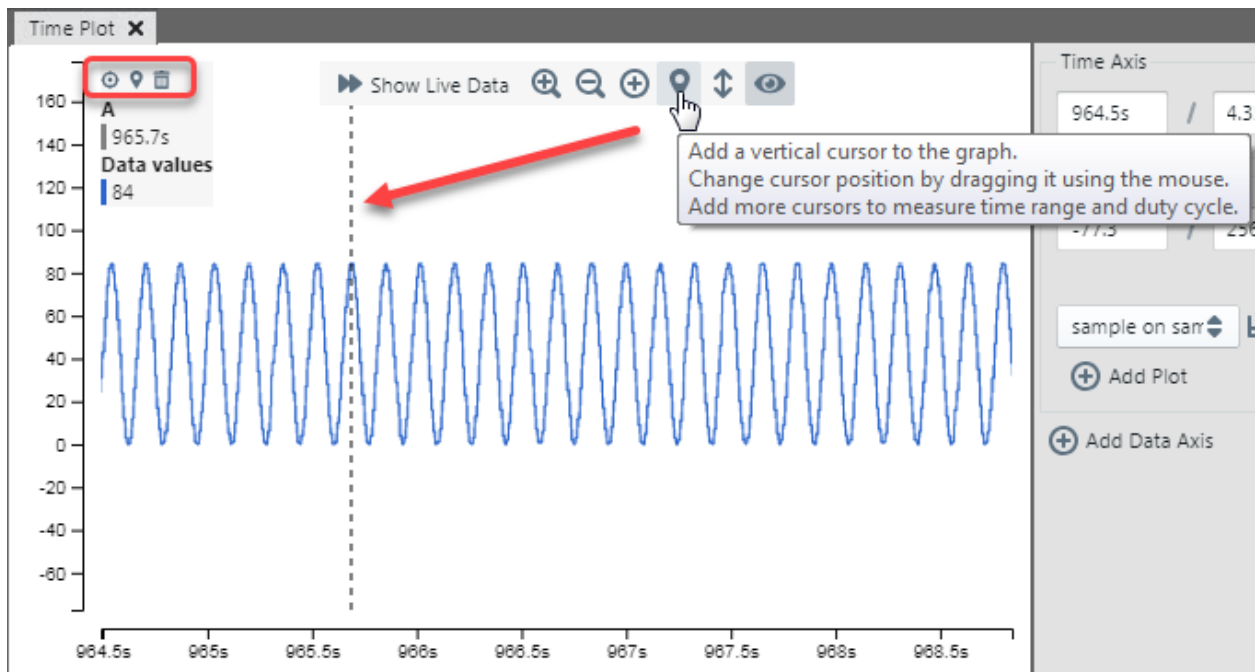
通过调整轴范围和图形位置可以更方便地查看数据。在下图中，我们以圈起的 **Data Value**（数据值，垂直）轴为例进行说明。单击要调整的轴或其附近区域。然后使用鼠标滚轮放大或缩小轴范围。您也可以单击并按住鼠标向一个方向或另一个方向拖动轴，从而相应地移动图形。此外，位于轴一端的几个控件可用于放大或缩小以及设置图形特性，如可视化窗格中所示。

在图形上有一个垂直滚动标记，它将跟随鼠标移动并显示相应的 **Time**（时间，水平轴）和 **Data Value**（数据值，垂直轴）。

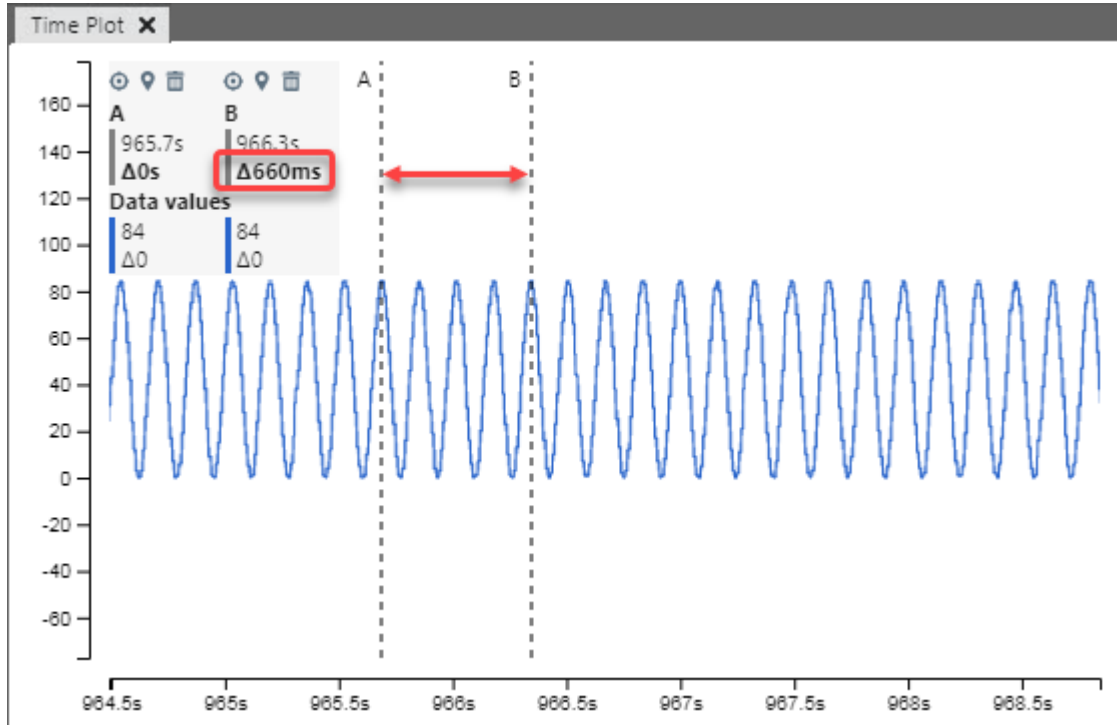


要启用可在绘图位置拖放的垂直光标，按如下所示单击以启用光标 A。这将禁止滚动标记。

光标控件将出现在滚动标记信息所在的位置。



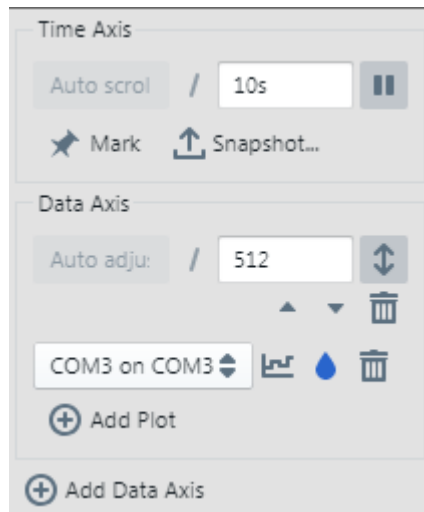
要确定绘图点之间的时间，请启用另一个垂直光标，设置其位置，然后查看时间差。



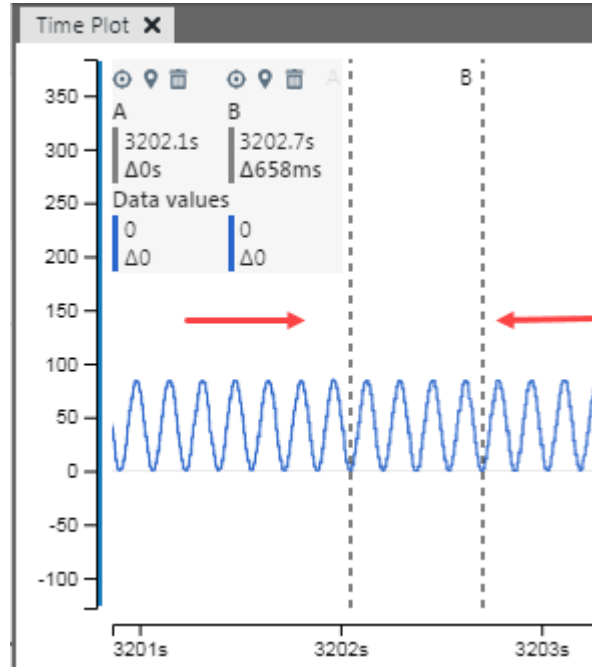
4.2.5.2.7 输出绘制数据

绘制数据的快照可以输出到文本文件。要捕捉绘制数据，使用时间图可视化控件窗格中的 **Mark** 和 **Snapshot**（快照）按钮。如果没有绘制数据源，将禁止 **Snapshot** 按钮。

图 4-29. Mark 和 Snapshot 按钮

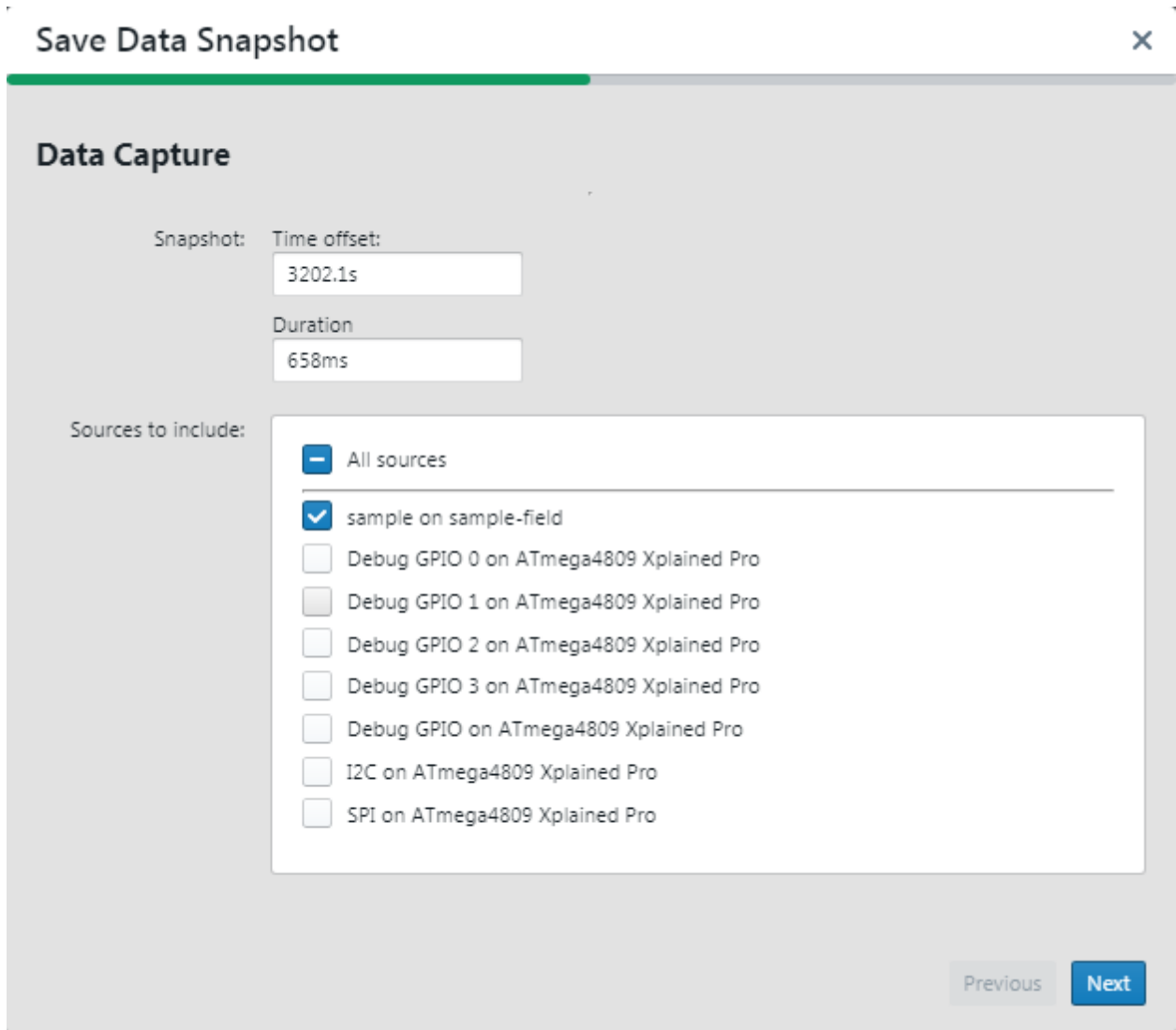


要捕捉绘制数据的快照，首先点击 **Mark** 以暂停图形滚动，并将光标设置于可见图形窗口的边界处。根据需要调整光标，如下图所示。



然后单击 **Snapshot**，打开 Snapshot 向导，选择数据和输出格式。

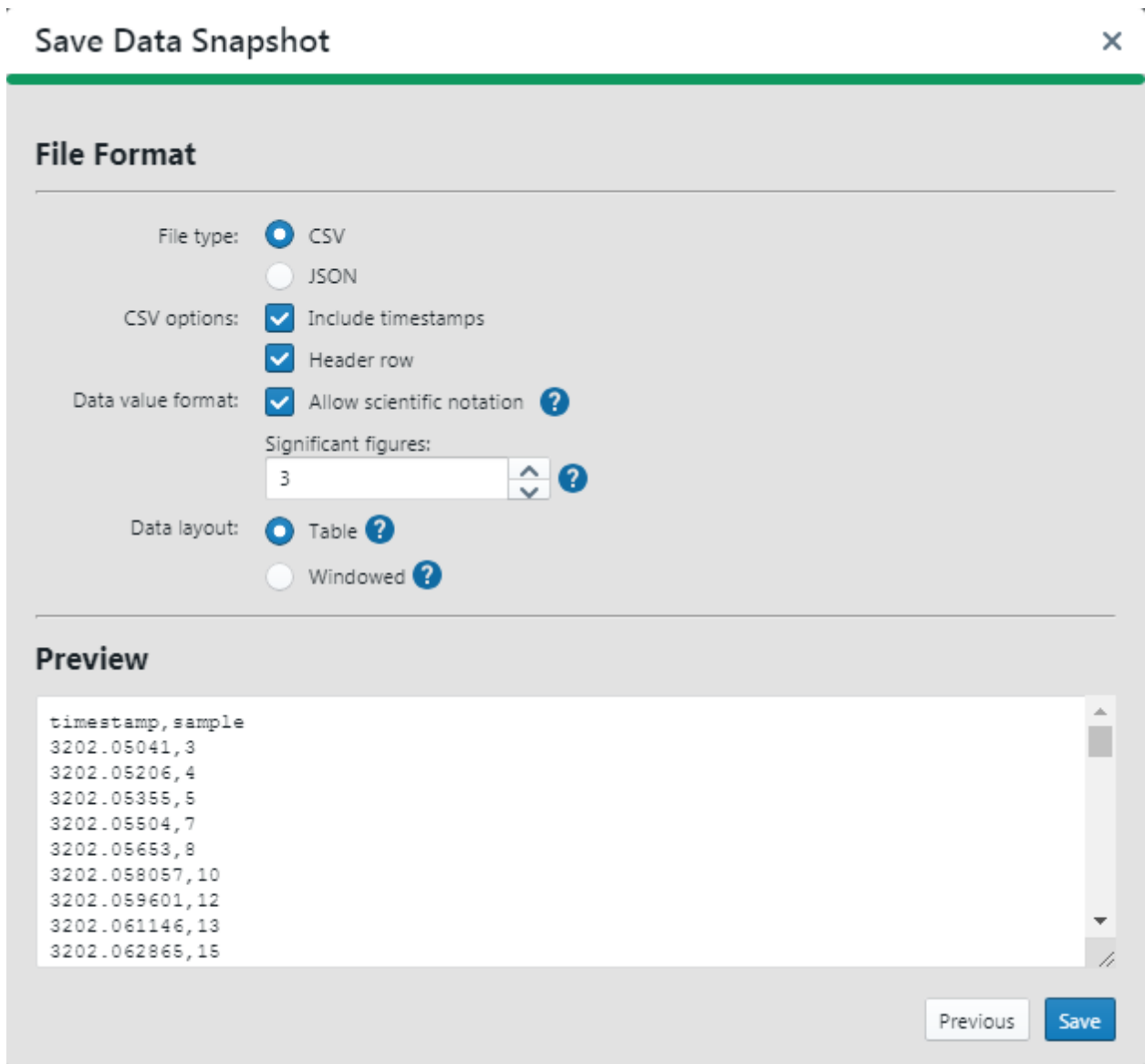
图 4-30. Snapshot——Data Capture (数据捕捉)



Snapshot 的 Time offset (时间偏移量) 和 Duration (持续时长) 通过光标位置进行设定。

Snapshot 数据源可以从所提供列表中选择。

图 4-31. Snapshot——File Format（文件格式）



选择一种可用的输出文件类型。其他选项取决于该文件类型选择。要了解有关每个选项的更多信息，将鼠标悬停在选项旁边的问号图标上会显示工具提示信息。

Preview（预览）窗口将显示所选结果。完成后，单击 **Save**（保存）。默认情况下，文件将保存到您的用户目录下。

4.2.5.3 多个数据图形的示例

AVR128DA48 Curiosity Nano 评估板用于演示如何使用 GPIO 引脚在同一轴或不同轴上生成多个数据图形。

本示例将使用的工具包括：

- MPLAB X IDE v5.40（或更高版本）
- MPLAB XC8 C 编译器 v2.20（或更高版本）
- AVR128DA48 Curiosity Nano 评估工具包——DM164151

4.2.5.3.1 示例设置

请按照下述小节中的说明设置示例软件和硬件。

MPLAB X IDE

访问以下链接，免费下载并安装 MPLAB X IDE 5.40 或更高版本。

www.microchip.com/mplab/mplab-x-ide

MPLAB XC8 C 编译器

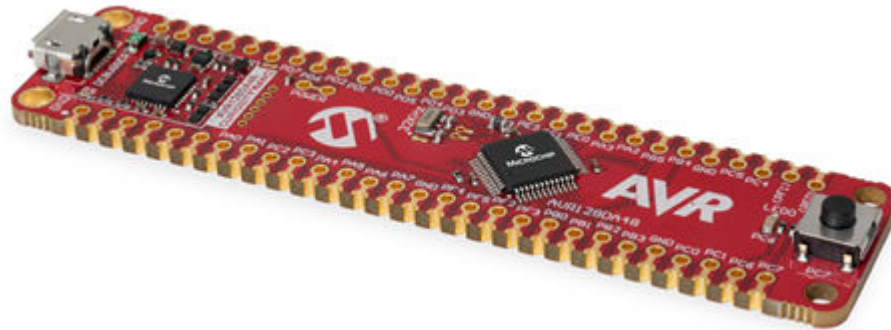
访问以下链接，免费下载并安装 MPLAB XC8 C 编译器 v2.20 或更高版本。还可以购买包含更多优化和功能的 PRO 版编译器。

www.microchip.com/mplab/compilers

AVR128DA48 Curiosity Nano 评估工具包——DM164151

此评估板（见下图）可通过 Microchip 直销网站（<https://www.microchipdirect.com/>）或代理商购买。然后使用随附的 USB 线缆，将评估板连接到计算机，并安装驱动程序。有关此评估板的更多信息，请访问：

www.microchip.com/DevelopmentTools/ProductDetails/PartNO/DM164151



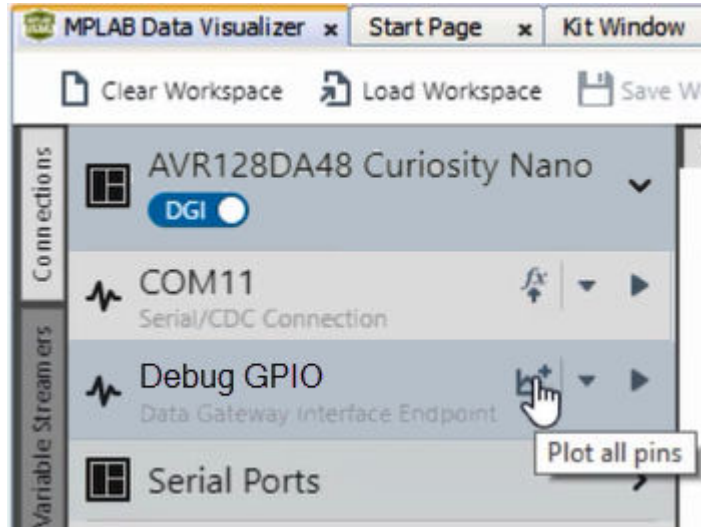
4.2.5.3.2 即插即用 Curiosity Nano

AVR128DA48 Curiosity Nano 评估板设计为即插即用型。因此，使用 USB 线缆将评估板插入 PC，然后启动 MPLAB X IDE。当 IDE 打开时，您会看到一个 **Kit Window**（工具包窗口）选项卡，其中包含有关此 Curiosity Nano 评估板的信息。单击评估板原理图链接，在原理图的第一页找到可用于 MPLAB 数据可视化器的 GPIO 引脚参考。将 PC7 连接至评估板的按钮开关，将 PC6 连接至 LED0。

DGI (Data Gateway Interface)	
Debug GPIO 0	PC7 (SW0)
Debug GPIO 1	PC6 (LED0)

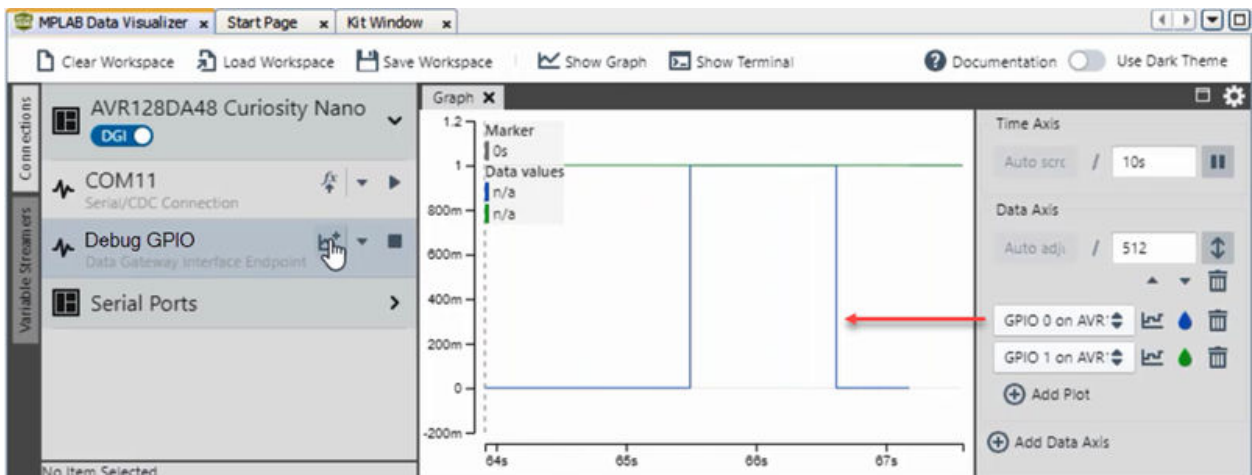


打开 MPLAB 数据可视化器插件。可视化器将显示可用的数据源，包括对应于 GPIO 引脚的 Debug GPIO。单击 Plot 图标以绘制所有引脚。



在图形中，将两个 GPIO 引脚的输出均绘制在同一个轴上。在图形的右侧，您将看到有关每个图形的信息，包括其颜色代码。

现在就可以使用 Curiosity Nano 了。按下评估板上的开关，以查看 GPIO 0 上的脉冲。注意，在按下按钮和出现脉冲之间存在延迟。再次参考原理图，可以看到按钮上没有上拉。然而，可以使用软件使能引脚上拉功能。另外，GPIO 1 仅显示为单根直线图，但是由定时器产生的脉冲可以提供一个更有趣的图形。因此，现在就可以创建项目并添加代码了。

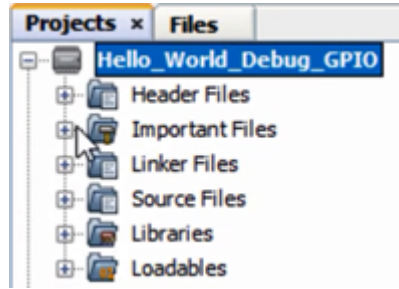


4.2.5.3.3 创建示例项目

选择 **File>New Project** (文件>新建项目) 或 **New Project** 图标 ，打开项目向导。按照以下步骤创建项目。单击 **Next** 转到下一步。

1. **选择项目**：单击“Microchip Embedded”类别，然后选择“Standalone Project”项目。
2. **选择器件 (和工具)**：输入器件“AVR128DA48”。然后输入工具“AVR128DA48 Curiosity Nano-SN: MCHPL#”，其中工具序列号 (serial number, SN) 包含前缀“MCHP”，后跟一个多位数字。
3. **选择编译器**：在 **Compiler Toolchains>XC8** 下，选择编译器的最新版本。
4. **选择项目名称和文件夹**：为您的项目命名。例如“Hello World Debug GPIO”。对于 Windows 操作系统，默认项目文件夹为 C:\Users\\MPLABXProjects。

单击 **Finish** 后，项目树会出现在 **Projects** 窗口中。



4.2.5.3.4 创建应用程序

通过将源代码添加到项目中来创建应用程序。

创建一个新的主源文件

将新的源文件添加到项目中：

1. 右键单击项目源文件夹，然后选择 **New>Other**（新建>其他）。
2. 在“New File”向导的“Choose File Type”（选择文件类型）中，选择类别 **Microchip Embedded>XC8**（Microchip 嵌入式>XC8），然后选择文件类型 **avr-main.c**。
3. 在“Name and Location”（名称和位置）下，将文件名更改为 **main.c**。然后单击 **Finish**。

将在 **Editor**（编辑器）窗口中打开新文件。默认情况下，代码将如下所示。

```

/*
 * File:    main.c
 * Author: Microchip Technology Inc.
 *
 * Created on May 19, 2020, 1:37 PM
 */

#include <avr/io.h>

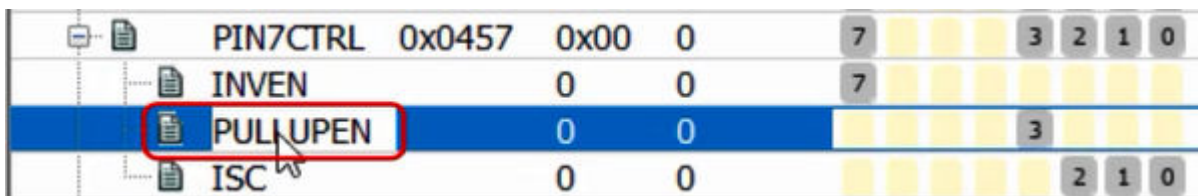
int main(void) {
    /* Replace with your application code */
    while (1) {
    }
}

```

添加用于使能 PC7 上上拉的代码

将用于使能 Port C 引脚 7 的上拉的代码添加到源代码中。**View IO**（查看 IO）窗口有助于查看寄存器位设置是否正确。

图 4-32. View IO 窗口



```

#include <avr/io.h>

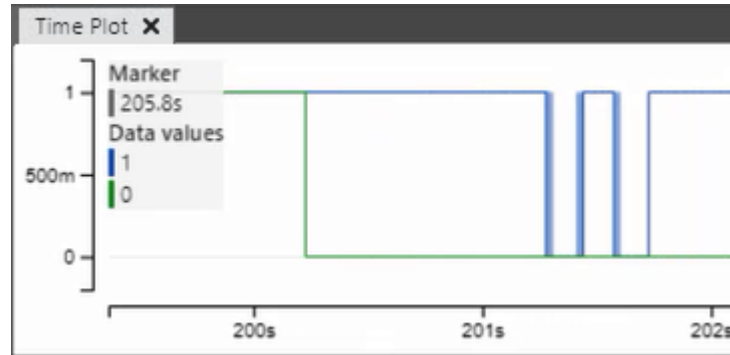
int main(void) {

    PORTC.PIN7CTRL = PORT_PULLUPEN_bm; /* Enable PC7 Pullup */
    PORTC.DIR = PIN6_bm; /* Turn on LED */

    while (1) {
    }
}

```

对器件编程  以获得 PC7 开关的更好响应并 点亮 PC6 LED。



添加代码以延迟翻转 LED

将前述代码添加到主源文件中，以使用 PC7 上的上拉功能来获得更好的开关响应，并点亮用户 LED。现在我们添加代码来翻转 LED 的开关状态。由于器件速度较快，这可能很难察觉到，除非添加一个延迟来减慢翻转速度。有关延迟的信息，请参见：

www.nongnu.org/avr-libc/user-manual/group__util__delay.html

需要为设定 CPU 频率添加一个 #define 语句，以及为支持延迟添加 #include 语句。

为了确定 Curiosity Nano 的 F_CPU，需再次打开 View IO 窗口并执行代码调试运行，在窗口中获取实时值。一旦代码进入 while(1) 循环，暂停并查看 CLKCTRL 以找到该值。

Icon	Peripheral	Option
	(CLKCTRL)	
	clock select (MCLKCTRLA)	0x0 - Internal high-frequency oscillator
	Prescaler division (MCLKCTRLB)	0x0 - 2X
	Frequency select (OSCHFCTRLA)	0xC - 4 MHz system clock (default)
	Multiplication factor (PLLCTRLA)	0x1 - 2 x multiplication factor
	Crystal startup time (XOSC32KCTRLA)	0x0 - 1k cycles

更新后的代码如下所示。

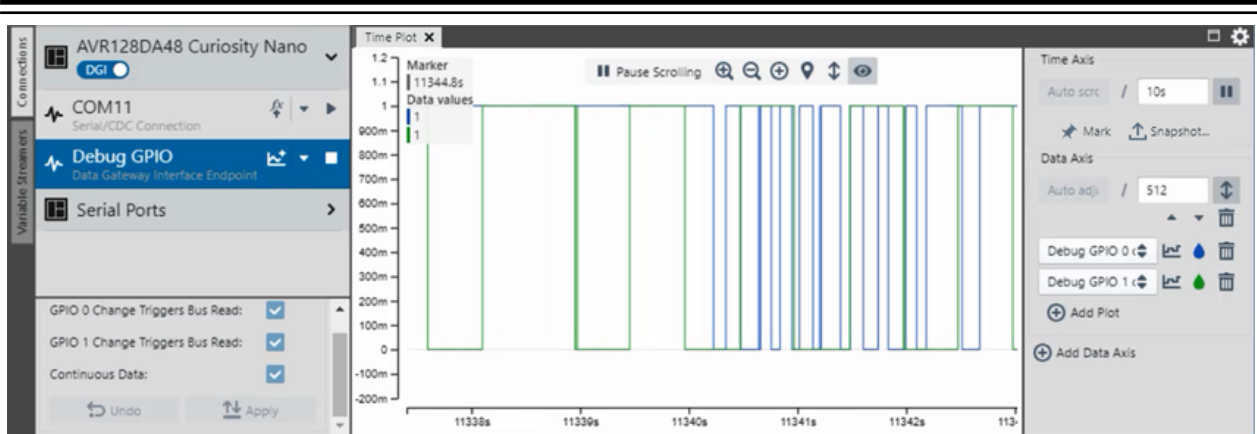
```
#include <avr/io.h>
#define F_CPU 4000000UL
#include <util/delay.h>

int main(void) {

    PORTC.PIN7CTRL = PORT_PULLUPEN_bm; /* Enable PC7 Pullup */
    PORTC.DIR = PIN6_bm; /* Turn on LED */

    while (1) {
        PORTC.OUTTGL = PIN6_bm; /* Toggle LED on/off */
        _delay_ms(500); /* wait between toggles */
    }
}
```

再次绘制 GPIO 引脚图，您可以看到通过翻转 LED 产生的绿色脉冲 PC6 (GPIO 1) 以及通过按下按钮产生的更窄的蓝色脉冲 PC7 (GPIO 0)。



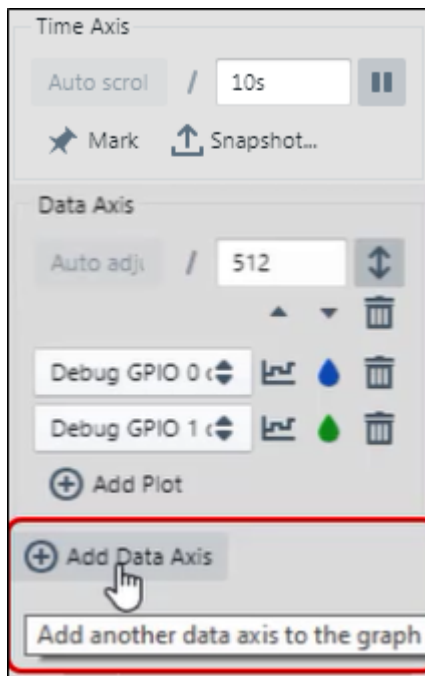
4.2.5.3.5 图形配置

该项目代码在两个 GPIO 引脚上都产生了输出，这使得在单个轴上显示两个图形较为拥挤，即使采用彩色编码绘制图形亦是如此。最好是每个图形分别置于自己的轴上。将这些图形都置于各自的轴上，GPIO 选项对图形的影响将一目了然。

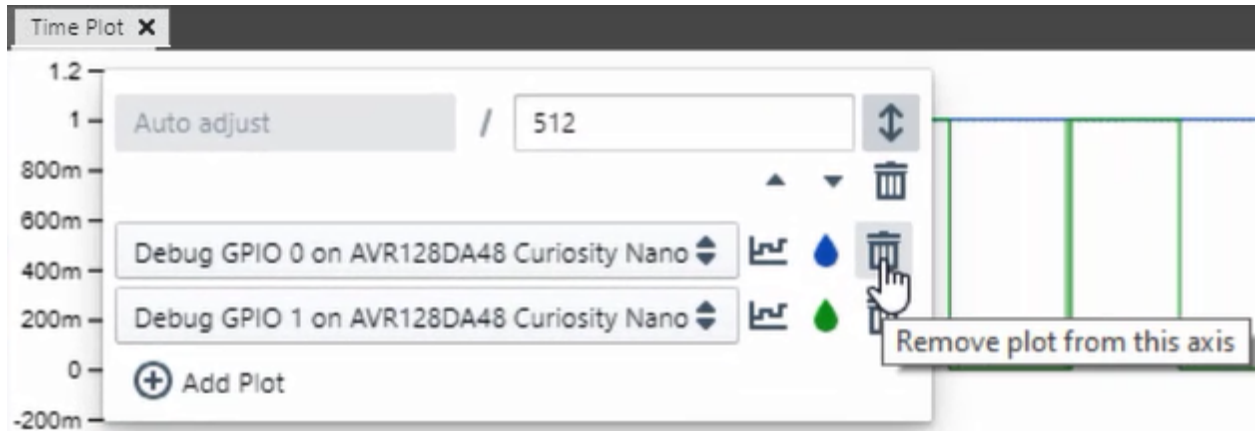
将每个图形移到一个轴上

在本示例中，一个轴上只有两个图形，但同样的步骤也适用于更多图形。

首先在图形中添加另一个轴。在原有轴下方将出现一个空轴。



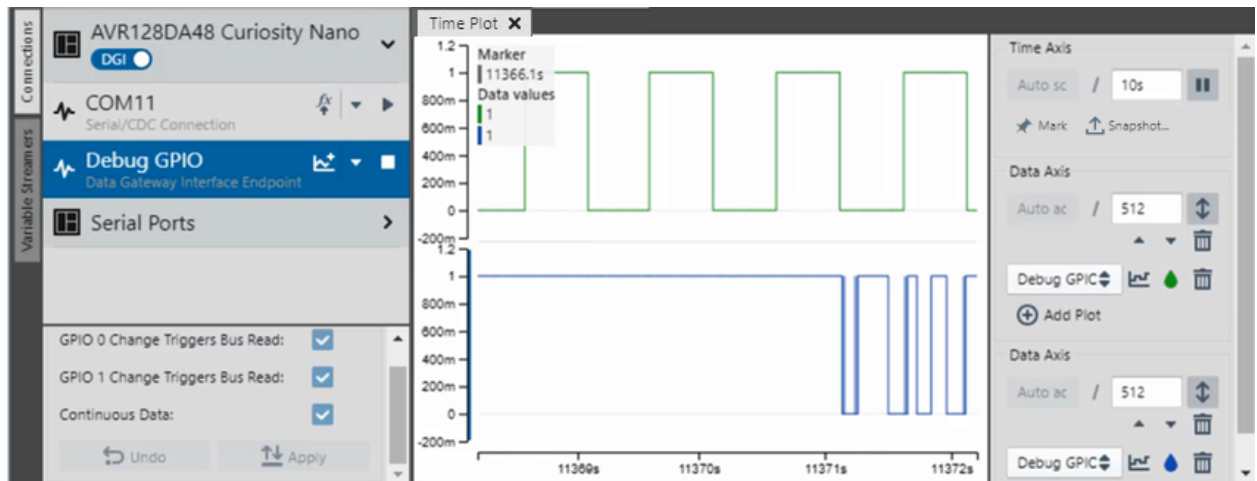
然后，从原有轴上删除一个图形。



之后，将删除的图形添加到新轴上。



最后，每个图形将位于自己的轴上，从而更加清晰可见。



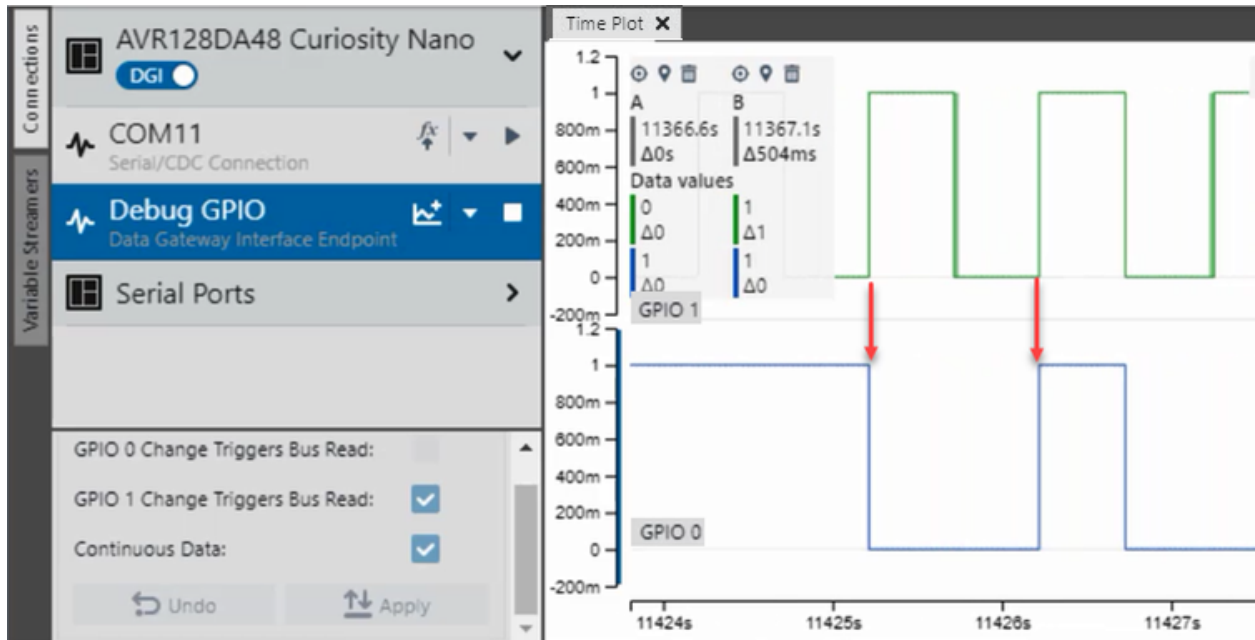
Debug GPIO 选项

选择了 Debug GPIO 数据源后，其下面的以下选项将变为可见选项。默认情况下，使能所有选项。

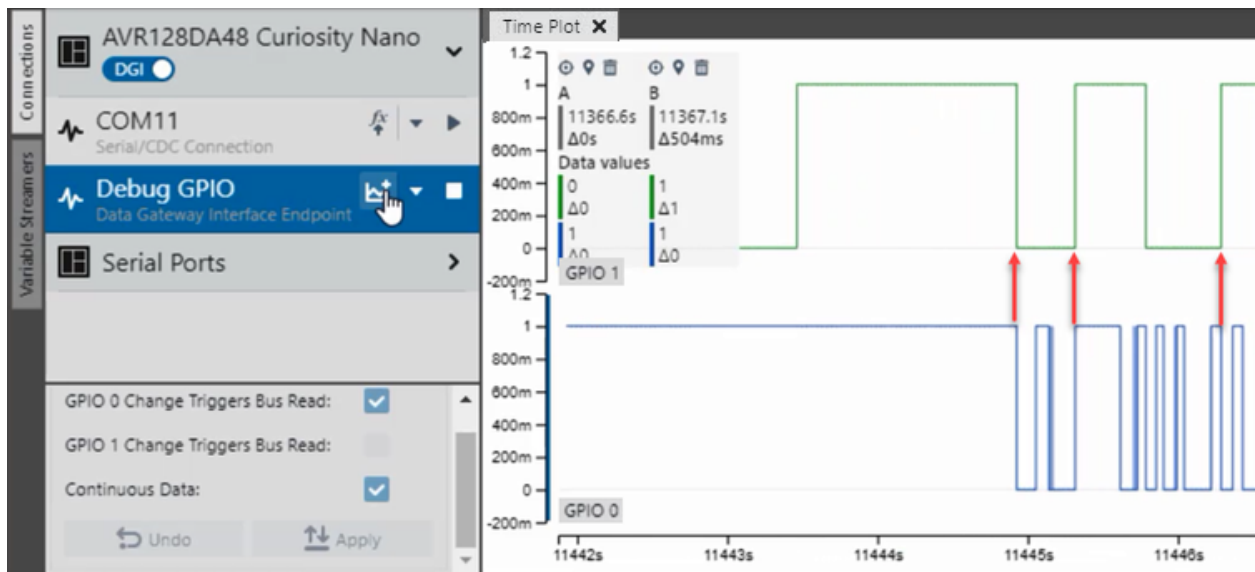
- GPIO 0 Change Triggers Bus Read
- GPIO 1 Change Triggers Bus Read
- Continuous Data (连续数据)

“GPIO x Change Triggers Bus Read” (GPIO x 变化触发总线读取) 意味着每当 GPIO x 发生变化时, 就会读取 GPIO 总线并在图形上显示数据。

如果禁止 “GPIO 0 Change Triggers Bus Read” 并使能 “GPIO 1 Change Triggers Bus Read”, 那么只有当 GPIO 1 翻转状态时才会读取总线, 这意味着即使快速按下按钮, 也只会显示 GPIO 1 翻转之间发生的变化。



如果使能 “GPIO 0 Change Triggers Bus Read” 并禁止 “GPIO 1 Change Triggers Bus Read”, 那么只有当按下按钮使 GPIO 0 状态改变时才会读取总线, 这意味着即使 GPIO 1 以一致的速率翻转状态, 也只会显示 GPIO 0 状态改变时发生的变化。



4.3 查看功率数据

使用 Time Plot 查看功率数据（如电流和电压测量值），并确定目标应用的功耗。

4.3.1 基本电流测量

为了进行功率测量，您必须使用支持此功能的工具。出于演示目的，我们将使用功率调试器。有关此工具的更多信息，请参见：

www.microchip.com/DevelopmentTools/ProductDetails/ATPOWERDEBUGGER

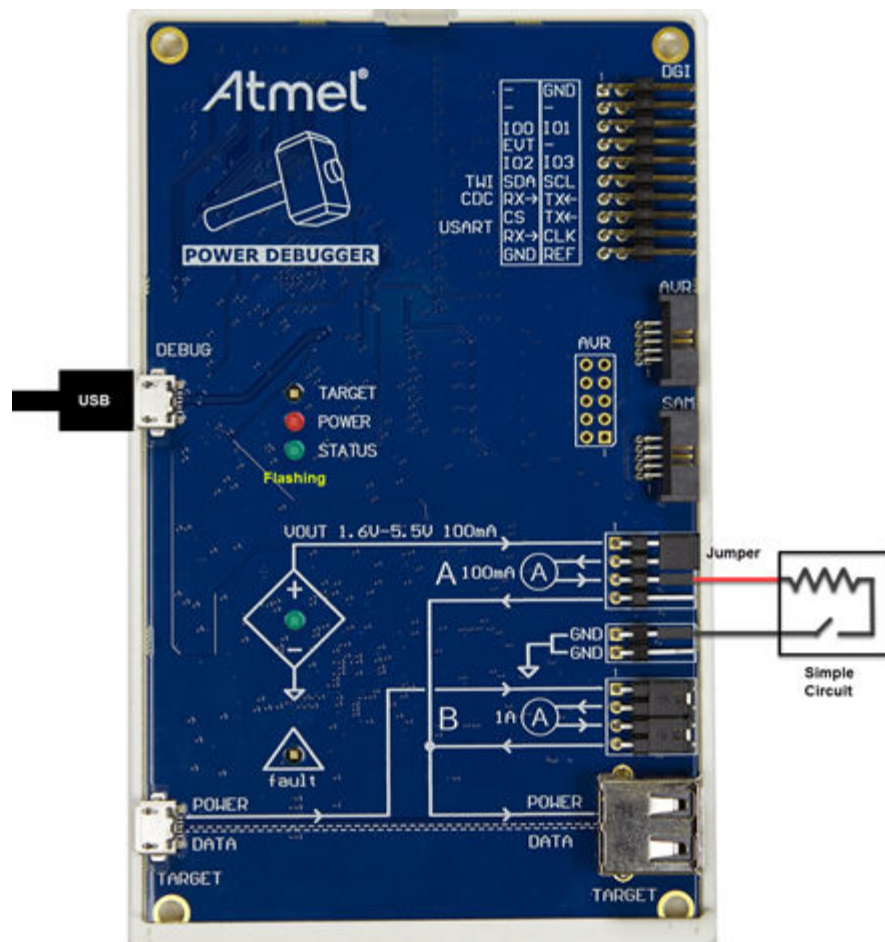
4.3.1.1 硬件设置

使用 USB 线缆将功率调试器连接至 PC。POWER（电源）LED 应变为红色。

然后将功率调试器连接到目标板。采用简单的连接（如下图所示）通过通道 A 来测量电路电流。电路将由调试器的 VOUT 供电，VOUT 已使用跳线连至 A+。A-已经连接至电路。另外，将电路的地连接到开发板的地。

当 MPLAB 数据可视化器检测到调试器并建立通信时，STATUS（状态）LED 将以绿色闪烁。可在数据可视化器中设置 VOUT 电压，VOUT 的 LED 会点亮为绿色。

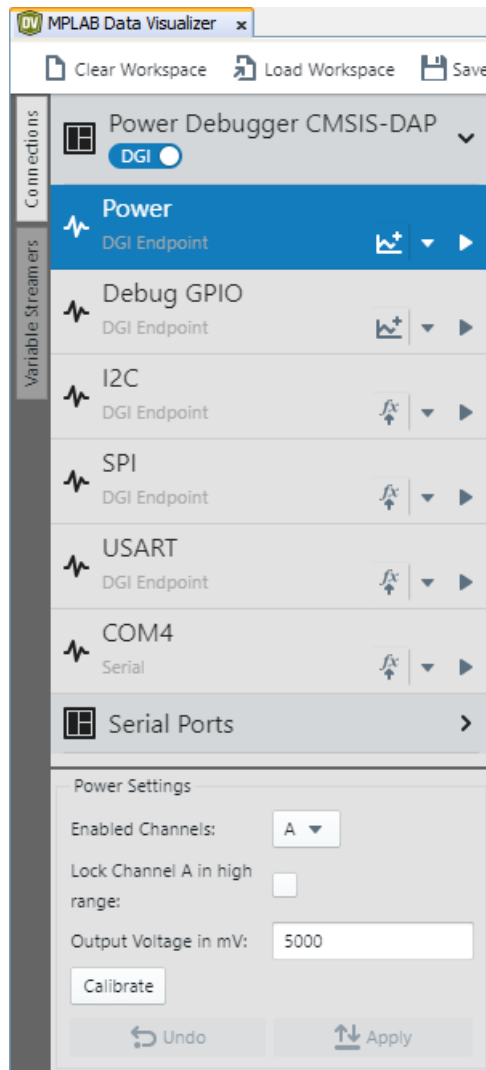
图 4-33. 简单连接



4.3.1.2 MPLAB 数据可视化器设置

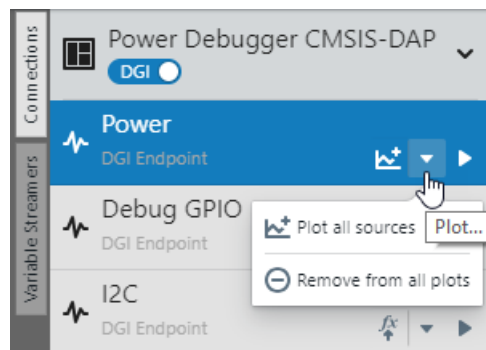
启动 MPLAB 数据可视化器。当检测到工具时，在工具 DGI 连接下会提供一个电源选择。单击 Power，选择 Power Settings。在下图中，选择通道 A 用于电流检测，VOUT 为 5000 mV（5V）。

图 4-34. 电源选择设置



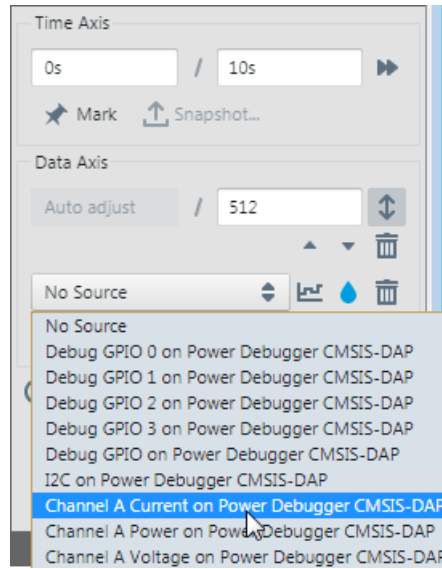
要查看功率数据图，选择迷你图形或向下箭头。

图 4-35. 绘制电源



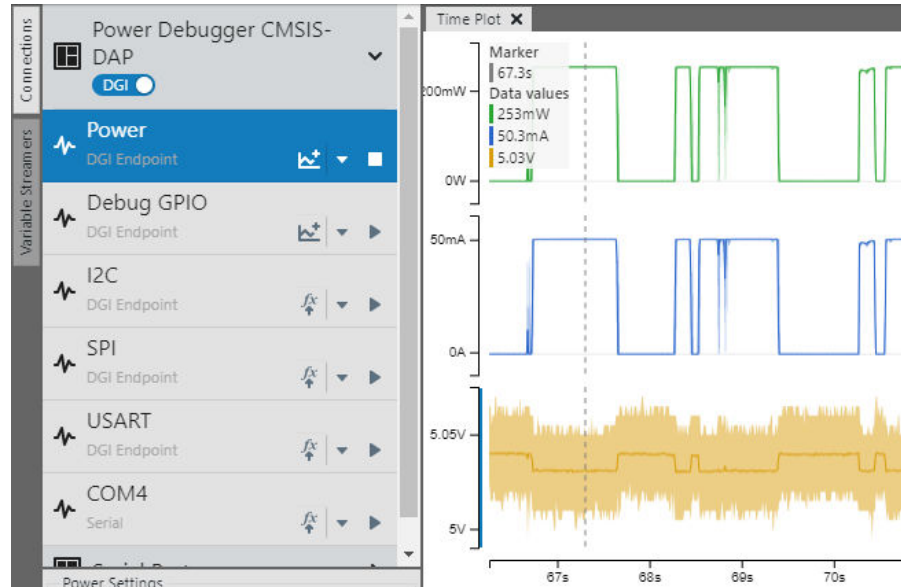
要选择单个图形，从绘制选项中选择该选项。

图 4-36. 绘制单个源



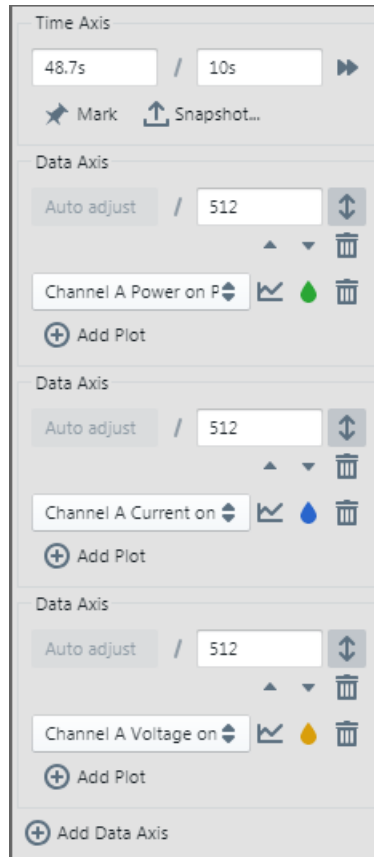
如果绘制所有源，在默认情况下，第一个是功率图形，第二个是电流图形，第三个是电压图形。

图 4-37. 功率图形



要改变图形格式，使用右侧窗格中的控件。

图 4-38. 图形控件

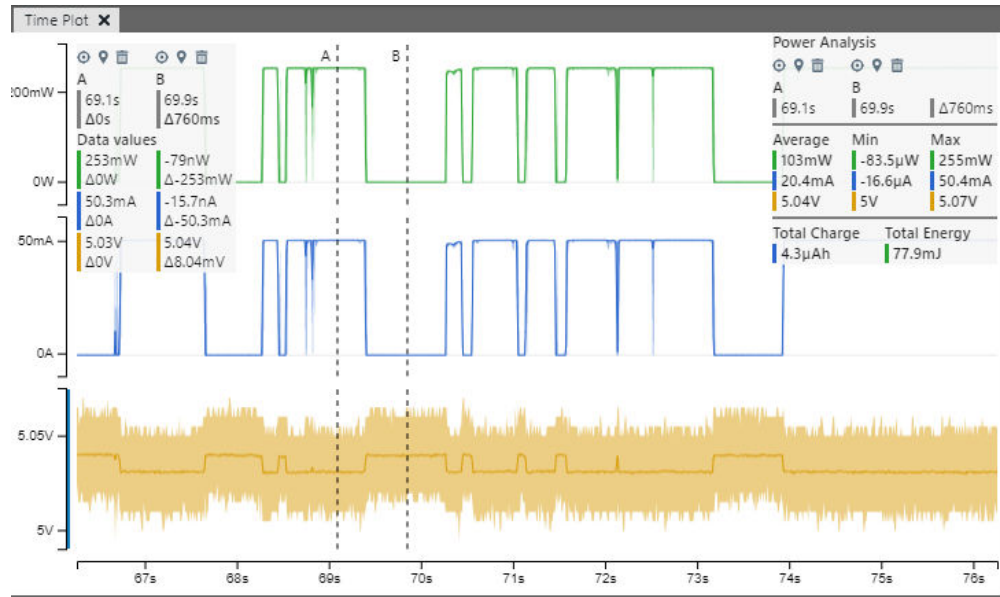


4.3.1.3 使用光标进行功率分析

为了进一步分析电流和其他功率测量值，从窗口工具栏上选择 **Power Analysis**。将出现两个光标（A 和 B），左上方是值检查器，右上方是功率分析数据。拖动光标可以看到不同的值。



图 4-39. 使用功率分析的值



相关信息

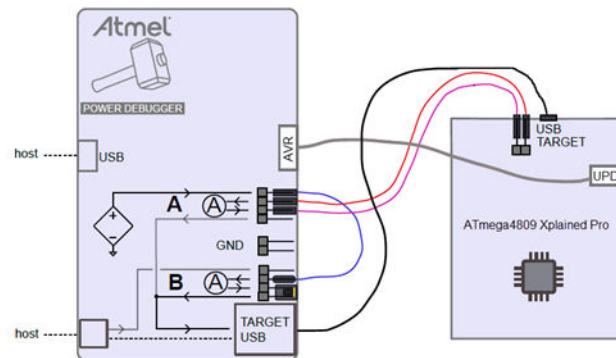
4.2.1.4.2 图形光标

4.3.2 多个电流测量

每个轴上可以显示多个图形。为此，将使用功率调试器的 A、B 两个通道从 ATmega4809 Xplained Pro 目标板上收集器件和目标板的功率数据。

4.3.2.1 硬件设置

按照下面的框图将功率调试器连接到目标板。连接方式如下所述。



为目标板供电：将使用功率调试器的可变电压电源。这样就可以在 USB 规范的整个电压范围内测试 USB 设备。

执行以下操作：

- 使用带状线缆将可变电压输出连接到 B 通道的输入。这样就可以测量目标板消耗的总电流。
- 将 B 通道的输出连接到 USB Type-A 插孔的 POWER 输入。只需使用一个跳线即可完成。
- 将 USB 线缆从 Type-A 插孔连接到 TARGET USB 微型插孔。

测量目标 MCU 消耗的电流：Xplained PRO 板在 TARGET USB 连接器旁边配有一个 2 引脚电源插针。通常会在此插针上放置一个跳线将 VCC_TARGET 连接到 VCC_MCU。通过移除该跳线并通过电流测量通道连接此电路，可以测量仅 MCU（而非整个目标板）消耗的电流。

执行以下操作：

- 将目标板上的 VCC_TARGET 连接到 A 通道的输入。

- 将 A 通道的输出连接至目标板的 VCC_MCU 引脚。

编程和调试目标器件：尽管 Xplained PRO 具有调试功能，但我们将使用功率调试器实现此目的。

执行以下操作：

- 将 10 引脚调试线缆从功率调试器的 AVR®插针连接到 Xplained PRO 的 UPDI DEBUG 插针。

将所有这些连接到主机计算机：

执行以下操作：

- 将 USB 线缆从功率调试器的 DEBUG 连接器连接到主机计算机。
- 将 USB 线缆从功率调试器的 TARGET 连接器连接到主机计算机。



提示： 连接编程连接器时，无需 GND 连接。

4.3.2.2 调试和可视化

用于目标板的应用程序基于以下代码示例中的示例 2：

[MPLAB XC8 嵌入式工程师用户指南之 AVR 代码示例](#)

要将代码用于 ATmega4809 Xplained Pro 板而非 Atmega4809 Curiosity 开发板，将代码中的“PORTD”改为“PORTB”。

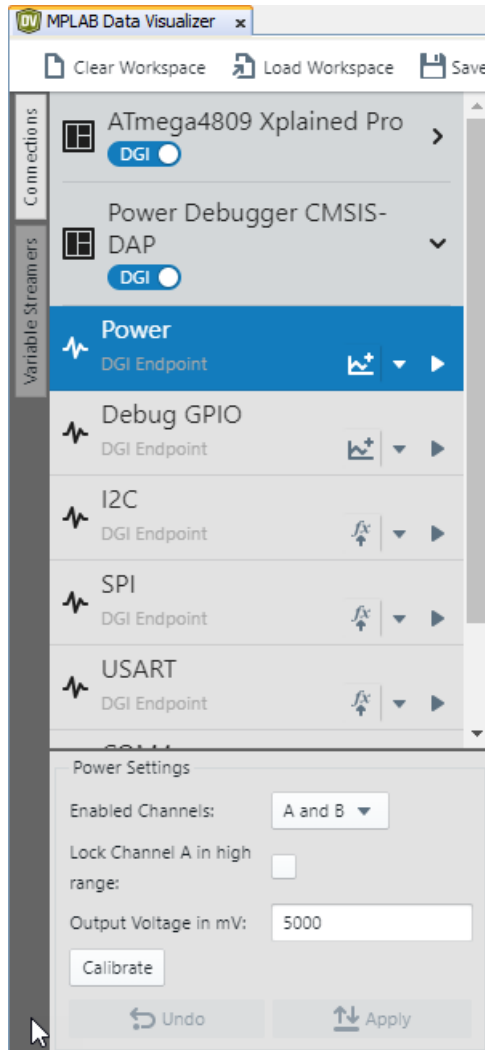
执行以下操作：

1. 在 MPLAB X IDE v5.50 或更高版本中打开 Example 2（示例 2）。
2. 按上述说明修改代码。

在 MPLAB X IDE 中打开 MPLAB 数据可视化器。

执行以下操作：

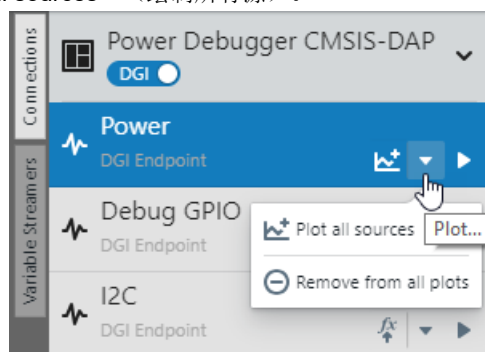
1. 选择 *Window>Debugging>Data Visualizer*（窗口>调试>数据可视化器）。这将打开 MPLAB 数据可视化器。
2. 数据可视化器应找到功率调试器 DGI 接口。从列表中选择 Power，然后选择 Power Settings。完成后单击 Apply。



3. 数据可视化器应找到 ATmega4809 Xplained Pro，因为它现在已经上电。不过，将只会使用功率调试器连接。现在可以对这个示例进行调试，并在可视化器中绘制输出。

执行以下操作：

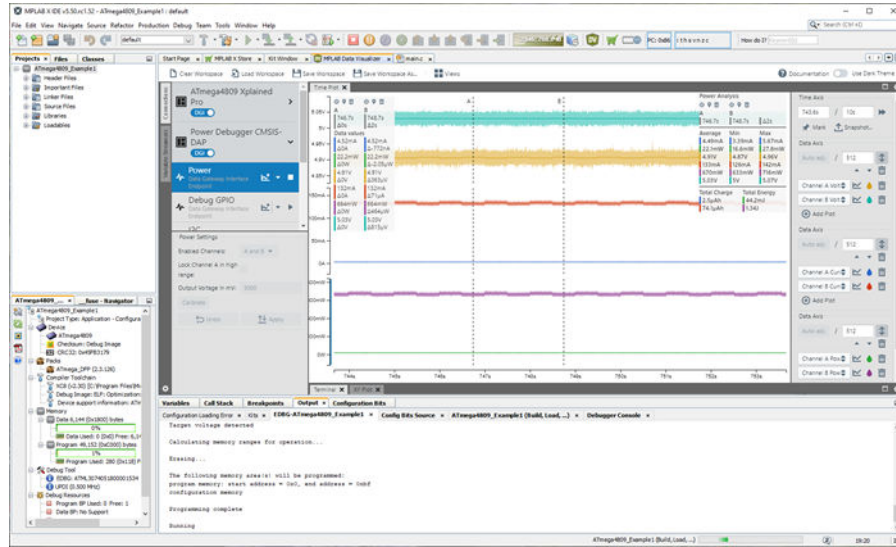
1. 调试示例。Xplained Pro 板运行时，板上的用户 LED 将闪烁。
2. 在 Power 界面上，选择“Plot all sources”（绘制所有源）。



4.3.2.3 放大

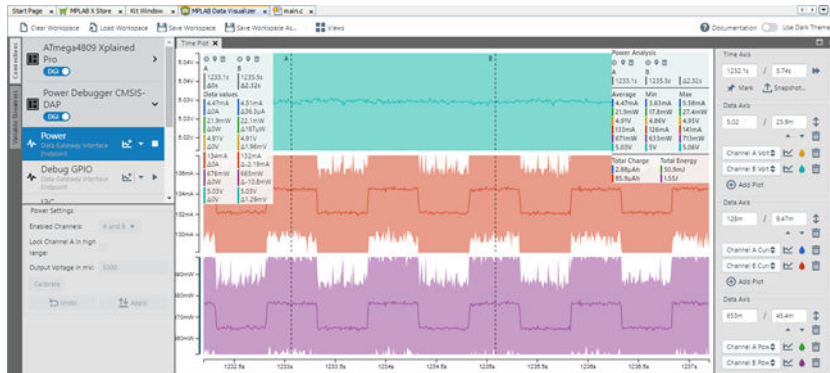
MPLAB 数据可视化器将在每个图中显示两个功率测量图形。一个是 ATmega4809 器件的，另一个是目标板的。尽管可以使用 Power Analysis 选项来帮助理解数据，但是很难看清楚图形。

图 4-40. 通道 A 和通道 B 的功率数据图形



为了更好地查看这些图形，放大每个通道的图形会很有用。将鼠标光标移到图形数值轴上，使用滚轮放大和缩小。

图 4-41. 通道 B 图形的缩放



相关信息

4.2.1.2 放大和缩小

4.4 在 XY Plot 中查看两组数据

使用 XY Plot 绘制一个数据集与另一个数据集的关系，而非绘制数据与时间的关系。这对于为机器学习输入数据很有用，下面将通过一个示例详细说明。

4.4.1 XY Plot 窗口

XY Plot 窗口 显示由连接的数据源提供的 x 与 y 数据关系图。该窗口中的数据点与 Time Plot 窗口中的时间轴同步。在 Time Plot 窗口上查看实时数据，或者向左或向右拖动时间轴将更改此处显示的数据。


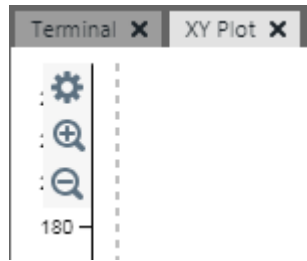
使用 XY 图可视化控件（右侧）窗格选择 Axis（轴）选项。或者，可以通过图标（）编辑轴信息。也可为此图使用轴放大/缩小和图形标记。

图 4-42. XY Plot 控件



可通过支持的插件工具提供更多功能。

相关信息

[4.2.1.4.1 图形标记](#)

[4.2.1.2 放大和缩小](#)

4.4.2 XY 图可视化控件窗格

XY 图可视化控件（右侧）窗格 用于控制 xy 数据的可视化（图形化）。

图 4-43. 图形可视化控件

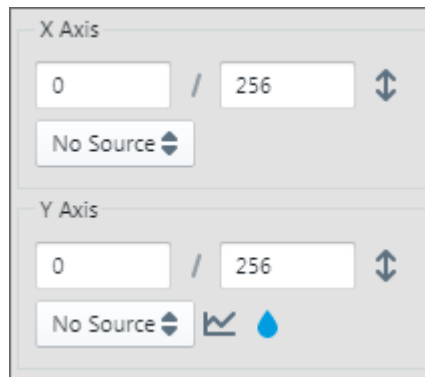


表 4-7. 数据轴




控件	说明
Offset	使能自动调整时，将显示“Auto adjust”。 禁止自动调整时，将显示当前的图形偏移量。
Scale	指定数据轴的分辨率，单位为秒。
	自动调整使能/禁止。 使能时，自动调整轴的范围。 禁止时，手动调整轴的范围。

表 4-8. 数据轴——绘制数据源和格式

控件	说明
Data Source	从下拉菜单中选择要绘制的数据源。请参见数据源窗格的选择和设置。
	单击以选择数据点在图形上的显示方式。 <ul style="list-style-type: none"> 使用阶梯线连接点（默认） 使用直线连接点 仅画点，不连线

..... (续)

控件	说明
	单击以选择图形颜色。

4.4.3 XY Plot 示例

以下是 XY Plot 和 Time Plot 的使用示例

4.4.3.1 机器学习示例

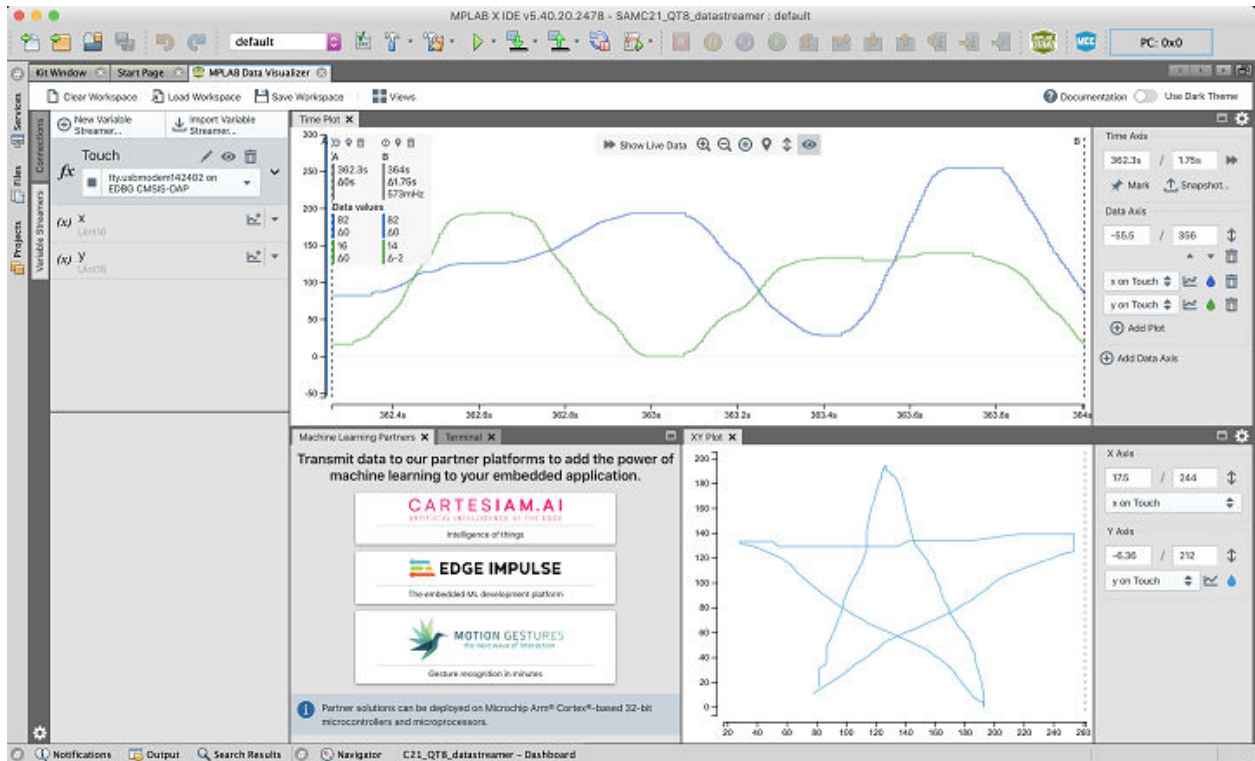
要查找 MPLAB 数据可视化器支持的插件，从 MPLAB X IDE 中选择 [Tools>Plugins](#)。找到插件，单击“Install”（安装），然后按照说明进行操作。

ML（机器学习）插件如下所示。安装插件后，重新启动应用程序，从工具栏的 **Views**（视图）列表中打开机器学习窗格。

ML 插件可在 MPLAB 数据可视化器中使用，因此您可以从目标器件捕捉实时数据流，选择所需的数据区域，然后将其上传到所选平台。借助合作伙伴的平台，您可以开发能够部署到您的嵌入式应用中的机器学习解决方案。

有关更多信息，请参见 [ML 插件用户指南](#)。

图 4-44. 机器学习插件



5. 变量设置器

大多数通信接口使用字节流来传输数据。这对于 8 位精度的单个数据值已足够，但是当需要通过同一接口发送多个值时，必须按照协议对数据进行打包。MPLAB 数据可视化器支持**数据流**协议。

数据流协议使用轻量级帧格式打包通过一个接口的多个数值。它只能处理传入数据，并且只支持同步流（即，每个数据包必须包含来自每个数据流的一个采样）。**数据流解码器**信息位于可视化器工作区。

可视化器的数据流模块获取传入的原始数据流，并将其拆分为多个数据流。数据流格式由您提供的 **Variable Streamer** 指定。

5.1 变量数据类型

使用向导设置和绘制变量设置器。该设置保存在工作区中。

在变量流设置器中定义的变量必须是下表中列出的类型。

表 5-1. 允许的数据类型

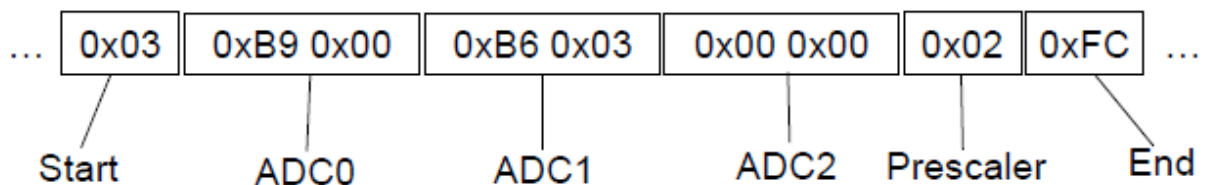
类型	大小（字节）
int8	1
int16	2
int32	4
uint8	1
uint16	2
uint32	4
float32	4
float64	8

5.2 流格式

数据流 Stream Format（流格式）处理顺序与变量流设置器中指定的顺序相同。所有数据的值必须以小尾数法形式给出，也就是说，必须先发送最低字节。此外，必须添加一个由数据流变量前的一个字节和数据流变量后的一个字节组成的封装结构。解释器使用该封装结构以与数据流进行同步。起始字节可以为任意值，但结束字节必须为起始字节的反码。在配置中未定义起始字节和结束字节。

下图给出了一个原始数据传输示例，其中 ADC0 为 185，ADC1 为 950，ADC2 为 0，Prescaler 为 2。

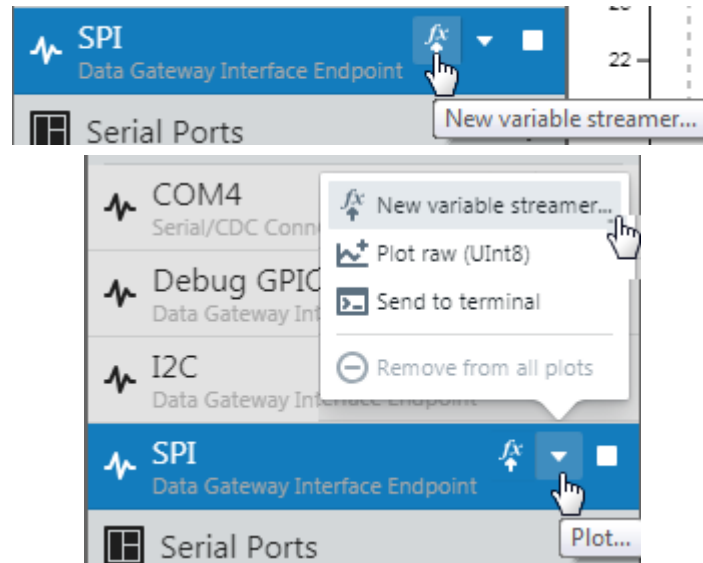
图 5-1. 数据流格式设置



5.3 变量流设置器的设置和绘制

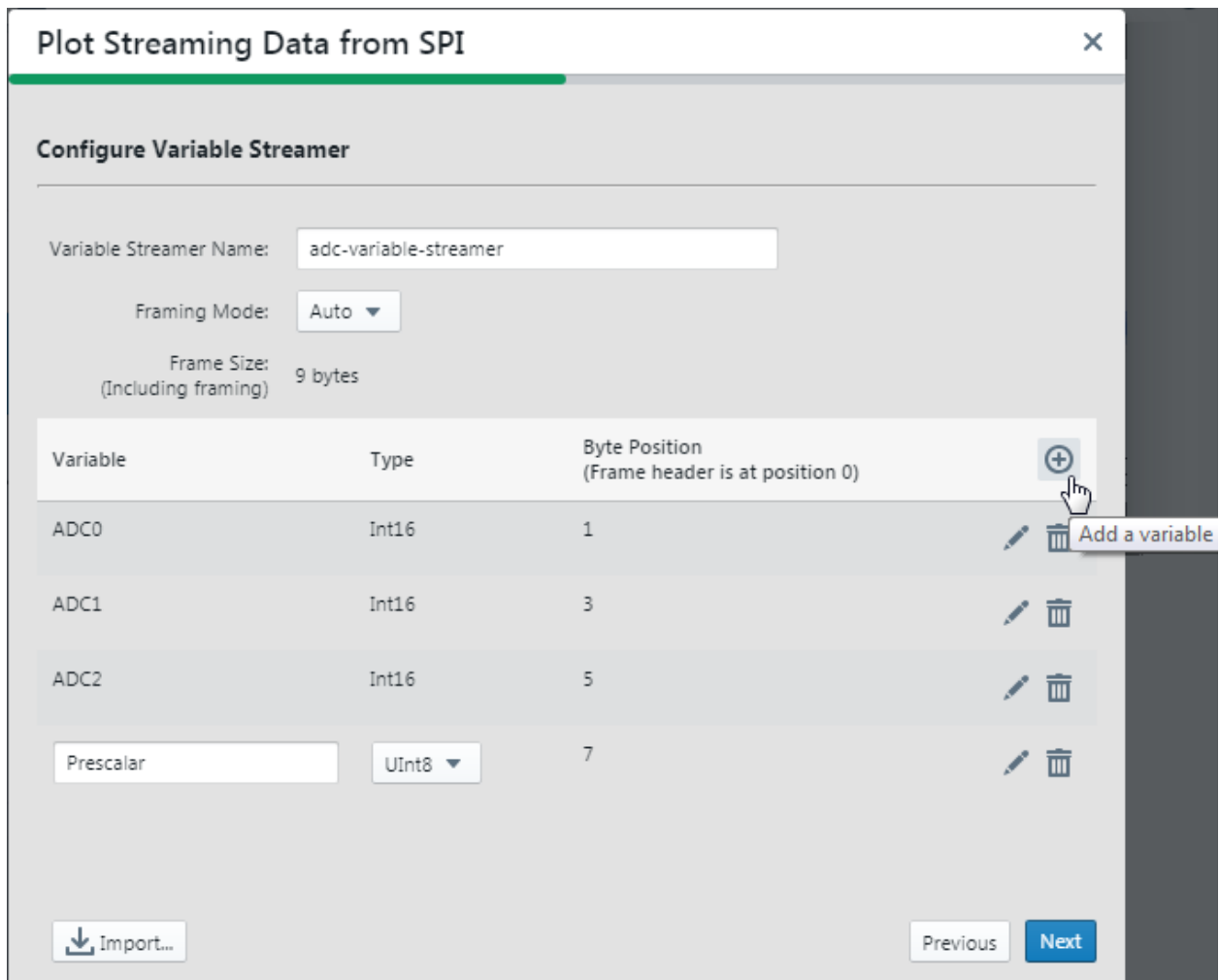
变量流设置器 定义作为应用的输出的数据流中嵌入的变量。变量流设置器用于数据流解码器实例的配置。解码器实例的输出是可以使用图形可视化的新数据流。

要创建一个新的变量流设置器，转到数据源窗格的 **Connections** 选项卡。从可用数据源中，选择“New Variable Steamer”（新建变量流设置器）或“Plot>New Variable Steamer”（绘制>新建变量流设置器）。

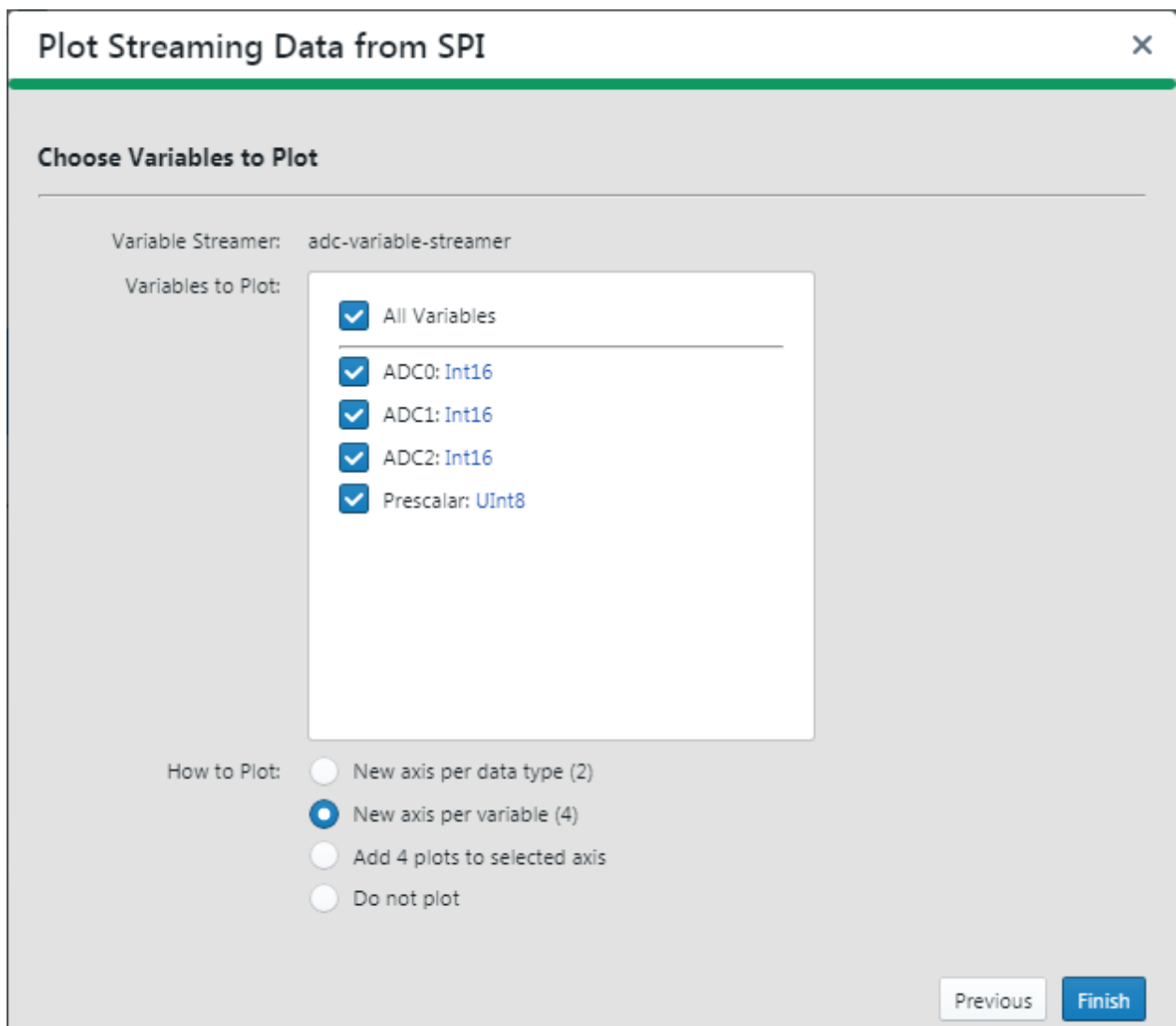


Plot Streaming Data（绘制流传输数据）向导将打开 **Configure Variable Steamer**（配置变量流设置器）窗口。

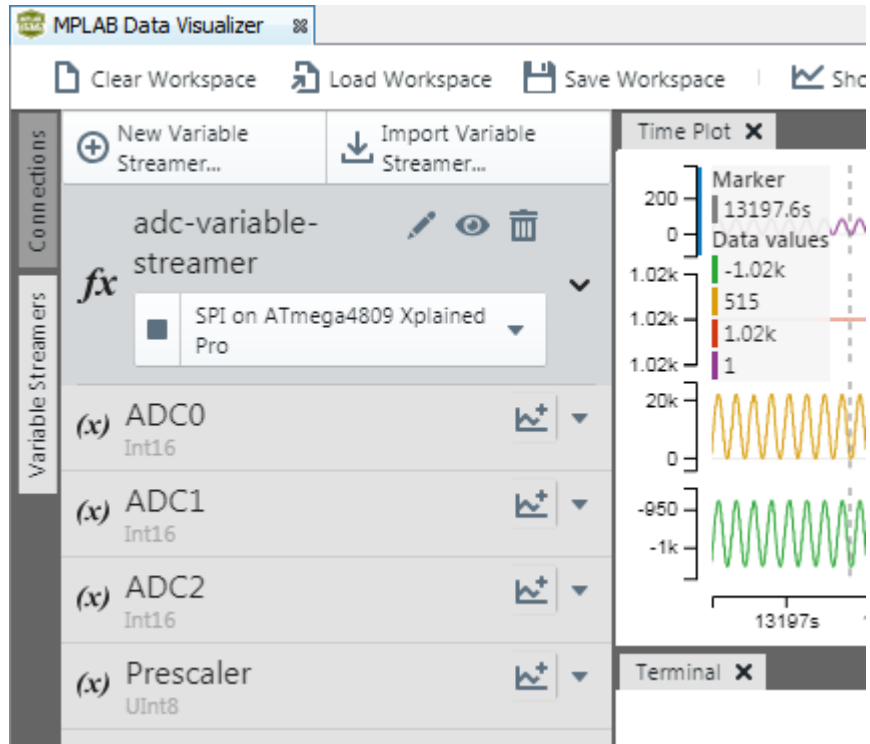
选项	说明
Variable Steamer Name（变量流设置器名称）	为变量流设置器选择一个描述性名称
Framing Mode（帧模式）	如果数据流格式设置协议的起始字节是 0x5F，结束字节是 0xA0，则可以使用“Auto”（自动）。 对于任何其他起始字节和结束字节模式，可以使用“二进制反码”。
Framing Size（帧大小）	请参见 5.2 流格式 。
Variable（变量）	输入应用程序代码中的变量名称。
Type（类型）	请参见 5.1 变量数据类型 。
Byte Position（字节位置）	请参见 5.2 流格式 。



单击 **Next** 以 **Choose Variables to Plot**（选择要绘制的变量）。选择一个绘制方法，然后单击 **Finish**。

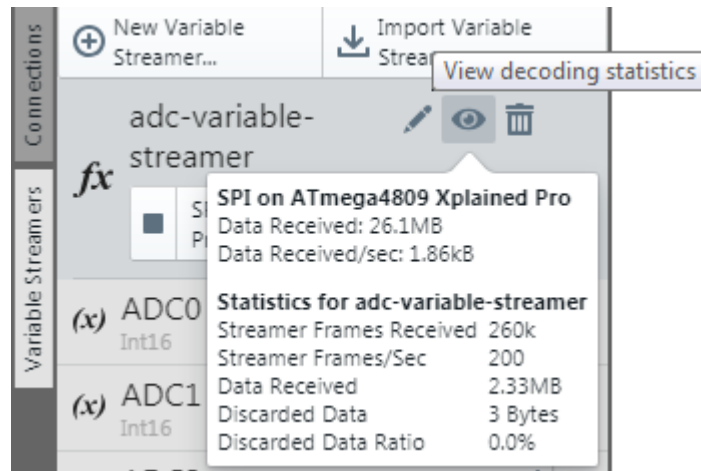


新的变量设置器将显示在 **Variable Streamers** 选项卡中。要保存您的设置，在会话结束时保存工作区，然后为下一个会话加载工作区。



5.4 查看统计信息

要查看有关流传输数据的统计信息，单击眼睛图标，以查看变量流设置器数据。



6. 故障排除

请参见下面的 MPLAB 数据可视化器组件，以掌握解决相关问题的技巧。

- [流传输](#)
- [数据流协议](#)

6.1 数据流传输

DGI SPI 接口

如果数据波形与预期不符，或无法通过数据流解码器解码，请尝试使能“Force synchronization on CS”（强制基于 CS 同步）。当在 Xplained Pro 开发工具包上启动调试会话，然后从此工具包通过 SPI 进行流传输时，同步问题尤为常见。

绘制资源

如果您的计算机在绘制多个数据流（如电压、电流和功率）时发生问题，请按照以下建议操作：

- 执行分析时停止数据流
- 滚动实时数据时缩小时间图
- 使用右侧窗格上的控件选择较少的绘制数据源
- 删除一个或多个图形

这些建议都能够减少所需的系统资源。

6.2 数据流解码器

无数据输入

确保连接了数据源，并且正在发送数据。一种简单的验证方法是绘制原始数据流或在 Terminal 中显示数据。如果图形中或 Terminal 中没有任何值，则可能没有接收到数据。

解码器不匹配

如果收到变量流设置器不匹配错误，请确保变量流设置器中的变量与输入数据完全匹配。解码器期望的数据为帧起始（start of frame, SoF）字节，后跟与变量流设置器中定义的字段匹配的字节序列，以及帧结束（end of frame, EoF）字节，该字节为 SoF 的二进制反码。以下情况会阻止数据包解码：

- 没有 SoF 或 EoF 字节
- 数据包传输中缺少任何一个字段
- 变量的大小与变量流设置器中定义的大小不匹配

每个变量的字节数必须与数据类型匹配。例如，Int8 应为一个字节，而 Int16 则为 2 个字节。

有时，虽然目标板发送的数据帧与 DV 中的变量流设置器之间不匹配，但不匹配错误会自行清除。这种情况可能发生在数据帧之间的字节长度仅有微小差异时。检查每个字段的大小，以确保其匹配。

[帧起始字符][数据字段 1 * 数据类型大小（字节）][数据字段 2 * 数据类型大小（字节）]……[数据字段 n * 数据类型大小（字节）][帧结束字符]

示例：

对于两个 Int16 字段 x 和 y，协议字节序列将如下所示：

[SoF][x1][x0][y1][y0][EoF]

与此协议匹配的数据流可能如下所示：

[0x3][0x34][0x12][0x78][0x56][0xFC]

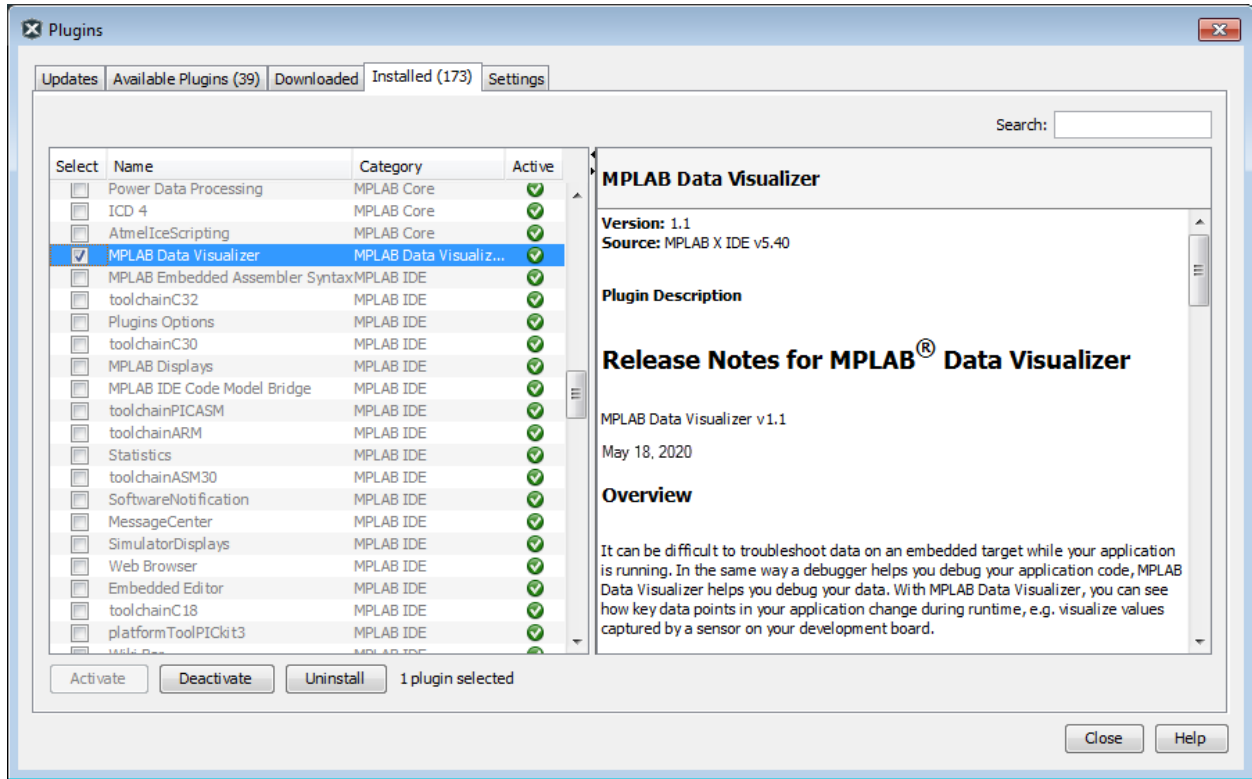
解码后的 x 和 y 值分别为 0x1234 和 0x5678。

7. 查找发行说明

可在以下位置找到 MPLAB 数据可视化器的发行说明：

- 对于插件，在 **Tools>Plugins>Installed>MPLAB Data Visualizer**（工具>插件>已安装插件>MPLAB 数据可视化器）下。
- 对于独立程序，在 **Help**（帮助）菜单下。

图 7-1. 插件发行说明



8. 绘制数据示例——代码清单

以下小节给出了 ATmega4809 Xplained Pro 项目的头文件和 C 代码示例。

注： 在跨页复制时应该多加注意，因为页脚可能出现在代码清单中。

8.1 C 头文件代码

configure.h

```
/*
 * File:    configure.h
 * Author:  Microchip Technology Inc.
 *
 * Created on September 20, 2018, 11:00 AM
 */

#ifndef CONFIGURE_H
#define CONFIGURE_H

#ifdef __cplusplus
extern "C" {
#endif

void initializePeripherals();

#ifdef __cplusplus
}
#endif

#endif /* CONFIGURE_H */
```

memutil.h

```
/*
 * File:    memutil.h
 * Author:  Microchip Technology Inc.
 *
 * Created on September 19, 2018, 1:03 PM
 */

#ifndef MEMUTIL_H
#define MEMUTIL_H

#ifdef __cplusplus
extern "C" {
#endif

#define LEN(a) (sizeof(a) / sizeof(*a))

#ifdef __cplusplus
}
#endif

#endif /* MEMUTIL_H */
```

pins.h

```
/*
 * File:    pins.h
 * Author:  Microchip Technology Inc.
 *
 * Created on September 19, 2018, 11:22 AM
 */

#ifndef PINS_H
#define PINS_H

#ifdef __cplusplus
```

```
extern "C" {
#ifdef

#define MISO_PIN 5
#define MOSI_PIN 4
#define CS_PIN 3
#define SCK_PIN 6

#ifdef __cplusplus
}
#endif

#endif /* PINS_H */
```

spi.h

```
/*
 * File: spi.h
 * Author: Microchip Technology Inc.
 *
 * Created on September 19, 2018, 11:21 AM
 */

#ifndef SPI_H
#define SPI_H

#ifdef __cplusplus
extern "C" {
#endif

void init_spi0(void);

void select_dgi_spi(void);
void deselect_dgi_spi(void);
void tx_spi0(uint8_t tx_usart1);
void tx_string_spi0(char* tx_string);
void tx_data_spi0(uint8_t tx_byte[], int length);

#ifdef __cplusplus
}
#endif

#endif /* SPI_H */
```

timer_callback.h

```
/*
 * File: timer_callback.h
 * Author: Microchip Technology Inc.
 *
 * Created on September 19, 2018, 11:15 AM
 */

#ifndef TIMER_CALLBACK_H
#define TIMER_CALLBACK_H

#ifdef __cplusplus
extern "C" {
#endif

void timer_callback();

#ifdef __cplusplus
}
#endif

#endif /* TIMER_CALLBACK_H */
```

8.2 C 源代码

configure.c

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/cpufunc.h>
#include "timer_callback.h"
#include "spi.h"
#include "pins.h"

void init_sysclock(void);
void init_tcb0(void);

#define PORTB_PIN2CTRL _SFR_MEM8(0x0432)

void initializePeripherals() {
    init_sysclock();
    init_tcb0();
    init_spi0();

    PORTB_PIN2CTRL |= 0x8;
}

void init_sysclock(void)
{
    CPU_CCP = 0xD8;
    CLKCTRL_MCLKCTRLB = 0x00;
}

void init_tcb0(void)
{
    TCB0.CTRLA = TCB_CLKSEL_CLKDIV2_gc; // base clock 16Mhz / 2 = 8 MHz
    TCB0.CTRLB = 0x00; // all the defaults
    TCB0.CCMP = 7999; // 8Mhz / 8000 = 1 kHz
    TCB0.INTCTRL = 0x01; // enable interrupt
    TCB0.CTRLA |= 0x01; // enable timer
}
}
```

main.c

```
/*
 * File:   main.c
 */

#include <stdio.h>
#include <stdlib.h>
#include "configure.h"
#include <avr/interrupt.h>

int main(int argc, char** argv)
{
    initializePeripherals();
    sei(); //set global interrupt flag

    while(1) ;
}
}
```

sine_app.c

```
/*
 * There are two waveforms in this application:
 * 1. sine wave
 * 2. triangle wave
 *
 * There are two global variables for control in this application:
 * - amp_factor - this defines the amplitude of the waveform
 * - wave_select - this defines the waveform selection.
 *   It can either be 0 for sine or 1 for triangle.
 */
#include <stdio.h>
```

```
#include "timer_callback.h"
#include "spi.h"
#include "memutil.h"

int amp_factor = 1;
int wave_select = 0;
int counter = 0;

int sine[] = {
    0x2b,0x2d,0x30,0x32,0x35,0x38,0x3a,0x3d,
    0x3f,0x41,0x43,0x46,0x48,0x49,0x4b,0x4d,
    0x4e,0x50,0x51,0x52,0x53,0x54,0x54,0x55,
    0x55,0x55,0x55,0x55,0x54,0x54,0x53,0x52,
    0x51,0x50,0x4e,0x4d,0x4b,0x49,0x48,0x46,
    0x43,0x41,0x3f,0x3d,0x3a,0x38,0x35,0x32,
    0x30,0x2d,0x2b,0x28,0x25,0x23,0x20,0x1d,
    0x1b,0x18,0x16,0x14,0x12,0xf,0xd,0xc,
    0xa,0x8,0x7,0x5,0x4,0x3,0x2,0x1,
    0x1,0x0,0x0,0x0,0x0,0x0,0x1,0x1,
    0x2,0x3,0x4,0x5,0x7,0x8,0xa,0xc,
    0xd,0xf,0x12,0x14,0x16,0x18,0x1b,0x1d,
    0x20,0x23,0x25,0x28,0x2b
};

int tri_1k[] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1};

struct
{
    int cnt;
    int *amp;
} waveform[] =
{
    {LEN(sine), sine},
    {LEN(tri_1k), tri_1k}
};

void timer_callback()
{
    uint8_t sample = (amp_factor * waveform[wave_select].amp[counter]) & 0x7F;
    if (++counter >= waveform[wave_select].cnt) {
        counter = 0;
    }

    select_dgi_spi();
    tx_spi0(0x03);
    tx_spi0(sample);
    tx_spi0(0xFC);
    deselect_dgi_spi();
}
```

spi.c

```
#include <avr/io.h>
#include "spi.h"
#include "pins.h"

void init_spi0(void)
{
    VPORTA.DIR |= (1 << MOSI_PIN) | (1 << SCK_PIN);
    VPORTF.DIR |= (1 << CS_PIN);

    SPI0.CTRLA = 0 << SPI_CLK2X_bp | 0 << SPI_DORD_bp | 1 << SPI_MASTER_bp |
    SPI_PRESC_DIV64_gc;
    SPI0.CTRLB = (1 << SPI_SSD_bp); // disable SS#
    SPI0.CTRLA |= 1 << SPI_ENABLE_bp;
}

void select_dgi_spi(void)
{
    VPORTF.OUT &= ~(1 << CS_PIN);
}

void deselect_dgi_spi(void)
{
}
```

```
    VPORTF.OUT |= (1 << CS_PIN);
}

void tx_spi0(uint8_t tx_byte)
{
    uint8_t tx_rdy = 0;

    SPI0.DATA = tx_byte;
    while(!tx_rdy)
        tx_rdy = (SPI0.INTFLAGS & SPI_IF_bm );
}

void tx_string_spi0(char* tx_string)
{
    while (*tx_string)
        tx_spi0(*(tx_string++));
}

void tx_data_spi0(uint8_t tx_byte[], int length)
{
    while (length--)
        tx_spi0(*(tx_byte++));
}
```

timer_loop.c

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/cpufunc.h>
#include "timer_callback.h"

unsigned char period = 1;
unsigned char tick = 0;

ISR(TCB0_INT_vect)
{
    TCB0.INTFLAGS = 0x01;

    if (++tick > period) {
        tick = 0;
        timer_callback();
    }
}
```

9. 版本历史

以下是本文档的每个版本中的更改列表。

初始版本（2019 年 12 月）

本文档的初始版本为在线帮助。

版本 A（2020 年 6 月）

本文档的第一版（PDF 和在线帮助）。

版本 B（2021 年 5 月）

- 功率监控器支持：DGI 下的 Power 连接和激活后的三个相关功率图；Time Plot 悬停菜单下的 Power Analysis 功能。
- 将图形快照保存为 CSV 或 JSON 文件。
- 提供插件支持。从 MPLAB X IDE 或独立的可视化应用程序中选择 [Tools>Plugins](#)。
- 将 Graph 选项卡重命名为 Time Plot 选项卡。
- 添加了 XY Plot 选项卡窗口，以支持机器学习。
- 添加了 Views 菜单图标，从中可以选择 Time Plot、XY Plot、Terminal 或插件窗口作为活动窗口。

Microchip 网站

Microchip 网站 (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。我们的网站提供以下内容：

- **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 设计伙伴计划成员名单
- **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

产品变更通知服务

Microchip 的产品变更通知服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个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。

欲注册，请访问 www.microchip.com/pcn，然后按照注册说明进行操作。

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 (ESE)
- 技术支持

客户应联系其代理商、代表或 ESE 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。

也可通过 www.microchip.com/support 获得网上技术支持。

Microchip 器件代码保护功能

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规范。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品非常安全。
- 目前，仍存在着用恶意、甚至是非法的方法来试图破坏代码保护功能的行为。我们确信，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。这种试图破坏代码保护功能的行为极可能侵犯 Microchip 的知识产权。
- 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中。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。

法律声明

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。

本出版物中提供的信息仅仅是为方便您使用 Microchip 产品或使用这些产品来进行设计。本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，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 “按原样”提供这些信息。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，包括但不限于针对非侵权性、适销性和特定用途的适用性的暗示担保，或针对其使用情况、质量或性能的担保。

在任何情况下，对于因这些信息或使用这些信息而产生的任何间接的、特殊的、惩罚性的、偶然的或间接的损失、损害或任何类型的开销，Microchip 概不承担任何责任，即使 Microchip 已被告知可能发生损害或损害可以预见。在法律允许的最大范围内，对于因这些信息或使用这些信息而产生的所有索赔，Microchip 在任何情况下所承担的全部责任均不超出您为获得这些信息向 Microchip 直接支付的金额（如有）。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损害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任。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。

商标

Microchip 的名称和徽标组合、Microchip 徽标、Adaptec、AnyRate、AVR、AVR 徽标、AVR Freaks、BesTime、BitCloud、chipKIT、chipKIT 徽标、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maXStylus、maXTouch、MediaLB、megaAVR、Microsemi、Microsemi 徽标、MOST、MOST 徽标、MPLAB、OptoLyzer、PackeTime、PIC、picoPower、PICSTART、PIC32 徽标、PolarFire、Prochip Designer、QTouch、SAM-BA、SenGenuity、SpyNIC、SST、SST 徽标、SuperFlash、Symmetricom、SyncServer、Tachyon、TimeSource、tinyAVR、UNI/O、Vectron 及 XMEGA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的注册商标。

AgileSwitch、APT、ClockWorks、The Embedded Control Solutions Company、EtherSynch、FlashTec、Hyper Speed Control、HyperLight Load、IntelliMOS、Libero、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus 徽标、Quiet-Wire、SmartFusion、SyncWorld、Temux、TimeCesium、TimeHub、TimePictra、TimeProvider、WinPath 和 ZL 均为 Microchip Technology Incorporated 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、Augmented Switching、BlueSky、BodyCom、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、Espresso T1S、EtherGREEN、IdealBridge、In-Circuit Serial Programming、ICSP、INICnet、Intelligent Paralleling、Inter-Chip Connectivity、JitterBlocker、maxCrypto、maxView、memBrain、Mindi、MiWi、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICKit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、RTAX、RTG4、SAM-ICE、Serial Quad I/O、simpleMAP、SimpliPHY、SmartBuffer、SMART-I.S.、storClad、SQI、SuperSwitcher、SuperSwitcher II、Switchtec、SynchroPHY、Total Endurance、TSHARC、USBCheck、VariSense、VectorBlox、VeriPHY、ViewSpan、WiperLock、XpressConnect 和 ZENA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的商标。

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

Adaptec 徽标、Frequency on Demand、Silicon Storage Technology 和 Symmcom 均为 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。

© 2021, Microchip Technology Incorporated 版权所有。

ISBN: 978-1-5224-8289-5

质量管理体系

有关 Microchip 的质量管理体系的信息，请访问 www.microchip.com/quality。

全球销售及服务中心

美洲	亚太地区	亚太地区	欧洲
公司总部 2355 West Chandler Blvd. Chandler, AZ 85224-6199 电话: 480-792-7200 传真: 480-792-7277 技术支持: www.microchip.com/support 网址: www.microchip.com	澳大利亚 - 悉尼 电话: 61-2-9868-6733 中国 - 北京 电话: 86-10-8569-7000 中国 - 成都 电话: 86-28-8665-5511 中国 - 重庆 电话: 86-23-8980-9588 中国 - 东莞 电话: 86-769-8702-9880 中国 - 广州 电话: 86-20-8755-8029 中国 - 杭州 电话: 86-571-8792-8115 中国 - 香港特别行政区 电话: 852-2943-5100 中国 - 南京 电话: 86-25-8473-2460 中国 - 青岛 电话: 86-532-8502-7355 中国 - 上海 电话: 86-21-3326-8000 中国 - 沈阳 电话: 86-24-2334-2829 中国 - 深圳 电话: 86-755-8864-2200 中国 - 苏州 电话: 86-186-6233-1526 中国 - 武汉 电话: 86-27-5980-5300 中国 - 西安 电话: 86-29-8833-7252 中国 - 厦门 电话: 86-592-2388138 中国 - 珠海 电话: 86-756-3210040	印度 - 班加罗尔 电话: 91-80-3090-4444 印度 - 新德里 电话: 91-11-4160-8631 印度 - 浦那 电话: 91-20-4121-0141 日本 - 大阪 电话: 81-6-6152-7160 日本 - 东京 电话: 81-3-6880-3770 韩国 - 大邱 电话: 82-53-744-4301 韩国 - 首尔 电话: 82-2-554-7200 马来西亚 - 吉隆坡 电话: 60-3-7651-7906 马来西亚 - 槟榔屿 电话: 60-4-227-8870 菲律宾 - 马尼拉 电话: 63-2-634-9065 新加坡 电话: 65-6334-8870 台湾地区 - 新竹 电话: 886-3-577-8366 台湾地区 - 高雄 电话: 886-7-213-7830 台湾地区 - 台北 电话: 886-2-2508-8600 泰国 - 曼谷 电话: 66-2-694-1351 越南 - 胡志明市 电话: 84-28-5448-2100	奥地利 - 韦尔斯 电话: 43-7242-2244-39 传真: 43-7242-2244-393 丹麦 - 哥本哈根 电话: 45-4485-5910 传真: 45-4485-2829 芬兰 - 埃斯波 电话: 358-9-4520-820 法国 - 巴黎 电话: 33-1-69-53-63-20 传真: 33-1-69-30-90-79 德国 - 加兴 电话: 49-8931-9700 德国 - 哈恩 电话: 49-2129-3766400 德国 - 海尔布隆 电话: 49-7131-72400 德国 - 卡尔斯鲁厄 电话: 49-721-625370 德国 - 慕尼黑 电话: 49-89-627-144-0 传真: 49-89-627-144-44 德国 - 罗森海姆 电话: 49-8031-354-560 以色列 - 若那那市 电话: 972-9-744-7705 意大利 - 米兰 电话: 39-0331-742611 传真: 39-0331-466781 意大利 - 帕多瓦 电话: 39-049-7625286 荷兰 - 德卢内市 电话: 31-416-690399 传真: 31-416-690340 挪威 - 特隆赫姆 电话: 47-72884388 波兰 - 华沙 电话: 48-22-3325737 罗马尼亚 - 布加勒斯特 电话: 40-21-407-87-50 西班牙 - 马德里 电话: 34-91-708-08-90 传真: 34-91-708-08-91 瑞典 - 哥德堡 电话: 46-31-704-60-40 瑞典 - 斯德哥尔摩 电话: 46-8-5090-4654 英国 - 沃金厄姆 电话: 44-118-921-5800 传真: 44-118-921-5820
亚特兰大 德卢斯, 佐治亚州 电话: 678-957-9614 传真: 678-957-1455 奥斯汀, 德克萨斯州 电话: 512-257-3370 波士顿 韦斯特伯鲁, 马萨诸塞州 电话: 774-760-0087 传真: 774-760-0088 芝加哥 艾塔斯卡, 伊利诺伊州 电话: 630-285-0071 传真: 630-285-0075 达拉斯 阿迪森, 德克萨斯州 电话: 972-818-7423 传真: 972-818-2924 底特律 诺维, 密歇根州 电话: 248-848-4000 休斯顿, 德克萨斯州 电话: 281-894-5983 印第安纳波利斯 诺布尔斯特维尔, 印第安纳州 电话: 317-773-8323 传真: 317-773-5453 电话: 317-536-2380 洛杉矶 米慎维荷, 加利福尼亚州 电话: 949-462-9523 传真: 949-462-9608 电话: 951-273-7800 罗利, 北卡罗来纳州 电话: 919-844-7510 纽约, 纽约州 电话: 631-435-6000 圣何塞, 加利福尼亚州 电话: 408-735-9110 电话: 408-436-4270 加拿大 - 多伦多 电话: 905-695-1980 传真: 905-695-2078			