

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



MICROCHIP

MPLAB® Data Visualizer ユーザガイド

開発ツールをご使用のお客様へ



重要:

どのような文書でも内容は時間が経つにつれ古くなります。本書も例外ではありません。Microchip 社のツールとマニュアルは、お客様のニーズを満たすために常に改良を重ねており、実際のダイアログやツールの内容が本書の説明とは異なる場合があります。最新の PDF 文書については、Microchip 社のウェブサイト (www.microchip.com/) をご覧ください。

文書は各ページのフッタに表記している「DS」番号で識別します。DS 番号のフォーマットは「DS<文書番号><リビジョン>」です。<文書番号>は 8 桁の番号、<リビジョン>はアルファベットの大文字です。

各ツールの最新文書は、onlinedocs.microchip.com/ で見つける事ができます。



目次

開発ツールをご使用のお客様へ.....	1
1. まえがき.....	4
1.1. 本書の表記規則.....	
1.2. 推奨参考資料.....	5
2. 概要.....	6
2.1. 特長.....	6
2.2. 基本動作.....	7
2.3. インストール.....	10
3. 外部接続.....	12
3.1. 接続オプション.....	12
3.2. DGI (Data Gateway Interface).....	12
3.3. シリアルポート.....	15
4. データの表示.....	17
4.1. データのキャプチャと表示.....	17
4.2. ターミナル([Terminal]ウィンドウ) にデータをテキストとして表示する.....	18
4.3. 時系列プロットにおけるデータ表示.....	22
4.4. Power データの表示.....	52
4.5. プログラム カウンタ(PC)のサンプリング.....	60
4.6. XY プロットに2セットのデータを表示する.....	62
4.7. ダッシュボード ウィジェットを使ったデータ表示.....	64
5. Variable Streamer.....	81
5.1. 変数データ型.....	81
5.2. ストリーム フォーマット.....	81
5.3. Variable Streamer のセットアップとプロット.....	82
5.4. Variable Streamer のインポート.....	84
5.5. Auto-Configure Variable Streamer.....	85
6. DVRT プロトコル.....	87
6.1. プロジェクトに DVRT サポートを追加する.....	87
6.2. DVRT プロジェクト ELF ファイルを Data Visualizer に読み込む.....	87
6.3. シンボルの選択.....	89
6.4. 変数値のストリーミング.....	91
6.5. 変数名を変更してストリーミングを再開する.....	93
6.6. ストリーミング データの表示.....	94
6.7. ファームウェア内の変数の読み書き.....	95
6.8. DVRT プロトコルのオプションとステータス情報.....	96
7. トラブルシューティング.....	98
7.1. データストリーミング.....	98
7.2. データストリーム デコーダ.....	98
7.3. 注意事項.....	99

8. リリースノートの閲覧方法	100
9. データプロット用サンプルコード	101
9.1. C ヘッダコード	101
9.2. C ソースコード	103
10. 改訂履歴	106
10.1. リビジョン E (2023 年 1 月)	106
10.2. リビジョン D (2022 年 1 月)	106
10.3. リビジョン C (2021 年 11 月)	106
10.4. リビジョン B (2021 年 5 月)	106
10.5. リビジョン A (2020 年 6 月)	106
Microchip 社ウェブサイト	107
製品変更通知サービス	107
カスタマサポート	107
Microchip 社のデバイスコード保護機能	107
法律上の注意点	107
商標	108
品質管理システム	108
各国の営業所とサービス	109

1. 序章

以下には、本書の表記規則と推奨参考資料を記載しています。

1.1 本書の表記規則

本書には以下の表記規則を適用しています。

表 1-1 本書の表記規則

表記	適用	例
Arial、MS ゴシックフォント		
二重かぎカッコ: 『』	参考資料	『MPLAB® Data Visualize ユーザガイド』
太字	テキストの強調	...は 唯一 のコンパイラです...
角カッコ: []	ウィンドウ名	[Output]ウィンドウ
	ダイアログ名	[Settings]ダイアログ
	メニューの選択肢	[File]、[Save]を順に選択します。
かぎカッコ: 「 」	ウィンドウまたはダイアログのフィールド名	「Save project before build」
右山カッコ(>)で区切った下線付き斜体テキスト	メニュー項目の選択	<i>File>Save</i>
角カッコ([])で囲んだ太字	ダイアログのボタン	[OK] をクリックする
	タブ	[Power] タブをクリックする
N'Rnnnn	Verilog 形式の数値(N は総桁数、R は基数、n は各桁の値)	4'b0010, 2'hF1
山カッコ(<>)で囲んだテキスト	キーボードのキー	<Enter>、<F1>を押す
Courier New フォント		
標準書体の Courier New	サンプル ソースコード	#define START
	ファイル名	autoexec.bat
	ファイルパス	C:\Users\User1\Projects
	キーワード	static, auto, extern
	コマンドライン オプション	-Opa+, -Opa-
	ビット値	0, 1
	定数	0xFF, 'A'
斜体 Courier New	変数の引数	<i>file.o</i> (<i>file</i> は有効な任意のファイル名)
角カッコ: []	オプションの引数	xc8 [options] files
中カッコとパイプ文字:{ }	引数のどれかを選択する場合 (OR 選択)	errorlevel {0 1}

.....続き		
表記	適用	例
省略記号: ...	繰り返されるテキスト	<code>var_name [, var_name...]</code>
	ユーザが提供するコード	<code>void main (void) { ... }</code>

1.2 推奨参考資料

本書には MPLAB Data Visualizer の使い方を記載しています。参考資料として、Microchip 社が提供する以下の文書を推奨します。これらの文書にも役に立つ情報が記載されています。

『Atmel Data Visualizer ユーザガイド』

以前のバージョンの Data Visualizer ユーザガイドです。Atmel Data Visualizer は、各種ソース(例: 組み込みデバッガの DGI (Data Gateway Interface) および COM ポート)からデータを受信できます。

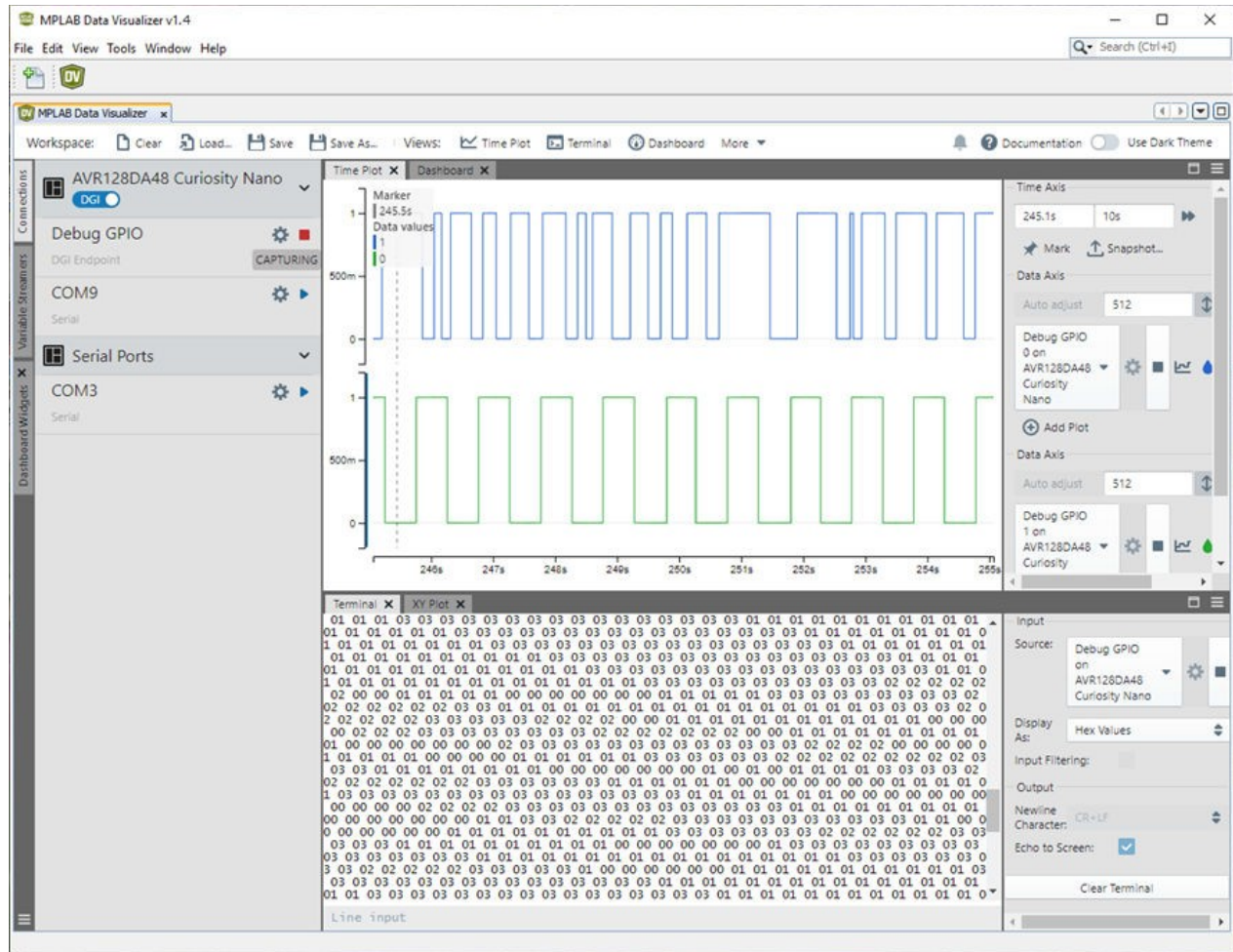
『Data Gateway Interface ユーザガイド』

Data Gateway Interface (DGI) は、ターゲット MCU との間で双方向に比較的低レートで小容量のデータ転送を処理するための USB インターフェイスです。DGI は各種ツールおよびオンボード デバッガ(Xplained Pro ボードが備える Power Debugger および EDBG 等)で利用できます。

2. 概要

MPLAB® Data Visualizer を使うと、実行中の組み込みターゲット デバイスからのデータをリアルタイムに処理して各種グラフやテーブルに表示できます。このプログラムは MPLAB X IDE 内から使える他、スタンドアロン プログラムとして使う事も可能です。

図 2-1. ストリーミング データの出力



2.1 特長

アプリケーションの実行中に組み込みターゲット上でデータの問題を解析する事は困難な場合があります。デバッガがアプリケーション コードのデバッグに役立つと同様に、MPLAB Data Visualizer はデータのデバッグに役立ちます。MPLAB Data Visualizer を使うと、アプリケーション内の重要データが実行中にどのように変化しているか観察できます(例: 開発ボード上のセンサが取り込んだ値を表示)。

MPLAB Data Visualizer により以下が可能です。

- 実行中の組み込みターゲット デバイスからシリアルポート(CDC)または Data Gateway Interface (DGI)を介してストリーミングしたデータをキャプチャする。
- 消費電力データを監視する。
- データストリーム プロトコル フォーマットを使って実行中にデータ フィールドをデコードする。
- Data Visualizer Run Time (DVRT)プロトコルを使ってデータのストリーミングとグローバル変数の読み書きを行う。
- 生データまたはデコード済みデータをグラフ(時系列または XY) にプロットすると共に、端末画面にテキストで値を表示する。
- ダッシュボードでウィジェットを使って受信データを監視すると共に、アプリケーションへ送信するデータを入力する。

- MPLAB® X IDE を使ってデータのストリーミングとターゲットコードのデバッグを同時に行う。
- データストリームおよびダッシュボード向け設定を自動的にロードする。
- カーソルを使ってプロットされたデータを解析する事により、信号の帯域幅やパルス幅等を計測する。
- プロットのスナップショットを CSV または JSON ファイルに保存する。
- プラグインの追加により機能を拡張する。



2.2 基本動作

Data Visualizer は、MPLAB X IDE プラグインまたはスタンドアロン プログラムとして動作します。プラグインとして使う場合、Data Visualizer 機能を使いながら同時にコードをデバッグできます。スタンドアロン プログラムとして使う場合、コードのデバッグはできません。しかし、スタンドアロンの Data Visualizer でデータをストリーミングしながら MPLAB X IDE 内で同じキットのコードをデバッグする事は可能です。

組み込みターゲットをホスト コンピュータに接続する方法は、3. 「外部接続」に記載しています。データを表示する各種の方法とそれらの例は 4. 「データの表示」に記載しています。変数値のプロット方法は 5. 「Variable Streamer」に記載しています。

2.2.1 ユーザ インターフェイス

MPLAB Data Visualizer のユーザ インターフェイスは 3 つの表示領域で構成されます。

1. **中央の領域**はデータ表示用のタブ付きウィンドウです。最もよく使うのが[Time Plot]タブです。タブをドラッグしてウィンドウを移動させる事ができます。タブをダブルクリックするとウィンドウは中央領域全体に拡大され、もう一度ダブルクリックするとウィンドウは元のサイズに戻ります。
2. **左側の領域**は、中央に表示するデータの選択と設定用に使います。[Connections]タブには接続されているデータソースが表示され、それらの表示方法を指定するためのオプションが利用できます。[Variable Streamers]タブでは、表示する変数を設定するための制御オプションが表示されます。左側の[Dashboard Widgets]タブを中央の[Dashboard]タブと一緒に使う事で、アプリケーション ファームウェアからのパラメータを制御および表示できます。タブをドラッグしてウィンドウ枠を移動させる事ができます。タブをクリックする事で、ウィンドウ枠の折り畳み/展開ができます。
3. **右側のウィンドウ枠**では、表示するデータのフォーマットを設定します。表示するデータの種類に応じて設定可能なオプションが異なります。最上段の[Window]アイコンをクリックすると、対応するウィンドウは中央領域全体に拡大され、再度クリックするとウィンドウは元のサイズに戻ります。[Toggle Properties]アイコンをクリックすると、ウィンドウ枠の表示/非表示を切り換える事ができます。

インターフェイスの制御項目については後述します。







ヒント: 大部分の制御オプションにマウスカーソルを重ねると、そのオプションの情報がポップアップに表示されます。

2.2.1.1 ツールバーの制御オプション

Data Visualizer ユーザ インターフェイスの最上段に配置されたツールバーには、表 2-1 に示す制御オプションが含まれます。

表 2-1. ツールバーの制御オプション

操作	追加の操作	概要
 Workspace: Clear		ワークスペース内のデータと設定をクリアします。データのストリーミングは全て停止します。ワークスペースは Data Visualizer の構成や設定の状態を保持したデータを表します。ワークスペースを保存し、後で読み込む事でデータと設定を復元できます。
 Workspace: Load		以前のセッションのデータと設定を Data Visualizer に読み込みます。
 Workspace: Save		現在のセッションのデータと設定をファイルに保存します。
 More Operations		ワークスペースに対するその他の操作を選択します。

.....続き		
操作	追加の操作	概要
	 Workspace: Save As	異なるパスまたは名前(もしくはその両方)を指定してワークスペースを保存します。
 Views: Time Plot		[Time Plot] ウィンドウを選択してアクティブにします。
 Views: Terminal		[Terminal] ウィンドウを選択してアクティブにします。
 More Operations		表示に関するその他の操作を選択します。
	 Views: Variable Streamers	[Variable Streamers] タブを選択してアクティブにします。
	 Views: DVRT Protocol	[DVRT Protocol] タブを選択してアクティブにします。
	 Views: DVRT Streaming Table	[DVRT Streaming Table] ウィンドウを選択してアクティブにします。
	 Views: DVRT Read/Write Data	[DVRT Read/Write Data Table] ウィンドウを選択してアクティブにします。
	 Views: Dashboard	「Dashboard」 ウィンドウを選択してアクティブにします。
	 Views: Dashboard	[Dashboard Widgets] タブを選択してアクティブにします。ドラッグによりウィジェットを[Dashboard] ウィンドウに対して追加/削除します。
	 Views: XY Plot	[XY Plot] ウィンドウを選択してアクティブにします。
 Protocols: DVRT Session		[DVRT Protocol] タブ、[Streaming Data Table] ウィンドウ、[Read/Write Data Table] ウィンドウを選択してアクティブにします。
 Messages		Data Visualizer が生成したメッセージを表示します。
 Help		Data Visualizer のオンラインヘルプを表示します。
 Dark Theme		有効時: ワークスペースの背景色を黒にして補色色相配色を適用します。 無効時: ワークスペースの背景色を白にして補色色相配色を適用します。

2.2.1.2 通知メッセージ


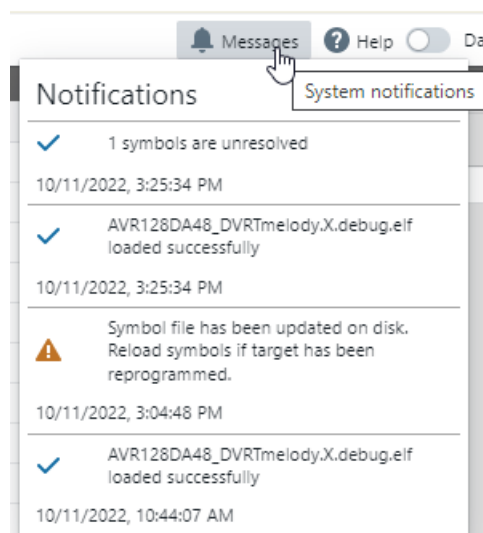
MPLAB Data Visualize は、特定の動作が完了した時に通知メッセージを表示します。これらのメッセージは短時間しか表示されません。見のがした場合は、 (Messages) をクリックすると現在のセッションに関する通知のリストをいつでも表示できます(図 2-2 参照)。

図 2-2. 通知リストの表示



2.2.1.3 Data Visualizer をスタンドアロンとして使う場合のメニュー

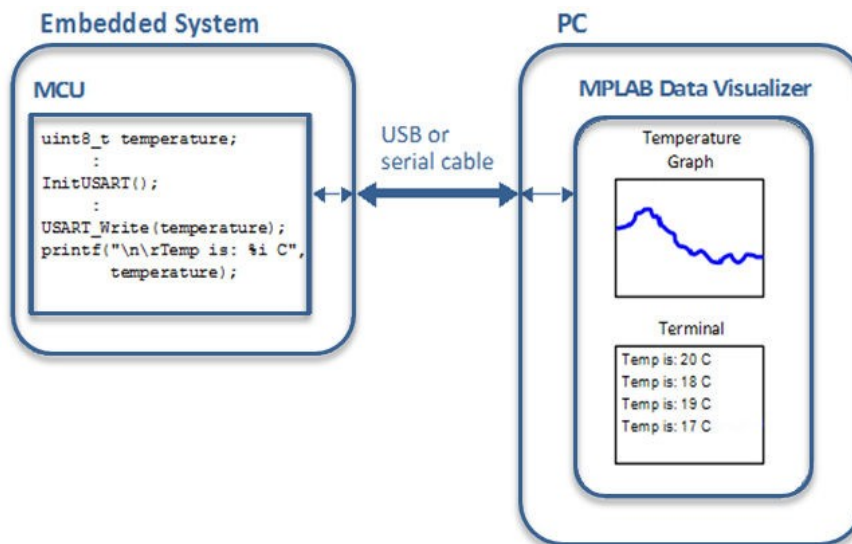
MPLAB Data Visualizer をスタンドアロン アプリケーションとして使う場合、メニューバーから MPLAB X IDE メニューと同様の機能が利用できます。基本的なテキストエディタ機能がメニューに含まれており、プロトコル定義ファイル(.ds)等の編集が可能です。

メニューの選択	概要
File>Items	基本的なファイル操作メニューです。Items として以下が選択できます。 New File、Open (Recent) File、Exit File
Edit>Items	基本的な編集メニューです。Items として以下が選択できます。 Undo/Redo、Cut/Copy/Paste、Delete、Find/ Replace
View>Items	基本的な表示メニューです。Items として以下が選択できます。 Editors、Split、IDE Log、Toolbars、Show Only Editor、Full Screen
Tools>Plugins	[Plugins]ダイアログを開きます。 このダイアログは、プラグインを追加/削除/管理するために使います。
Tools>Options	以下の Data Visualizer オプションを選択します。 <ul style="list-style-type: none"> • General - ウェブおよびプロキシ オプション • Keymap - キーマッピング オプション • Appearance - ユーザ インターフェイスの外観オプション • Miscellaneous - ファイルおよび出力のフォント/色オプション
Window> Items	基本的なウィンドウ メニューです。Items として以下が選択できます。 Favorites、Output、Editor、IDE Tools (Notifications または Processes)、Configure (size、float、dock、split 等)、Reset、Close、Close all Documents、Close Other Documents、Document Groups、Documents
Help>Items	基本的なヘルプ メニューです。Items として以下が選択できます。 Help Contents、Online Docs and Support、Keyboard Shortcuts Card、Check for Updates、About

2.2.2 接続

MPLAB Data Visualizer は、実行中の組み込みターゲットからのデータをキャプチャして表示します。例えば、Data Visualizer へ送信された温度値は、グラフにプロットする事もターミナルにテキストとして表示する事もできます。

図 2-3.接続の例

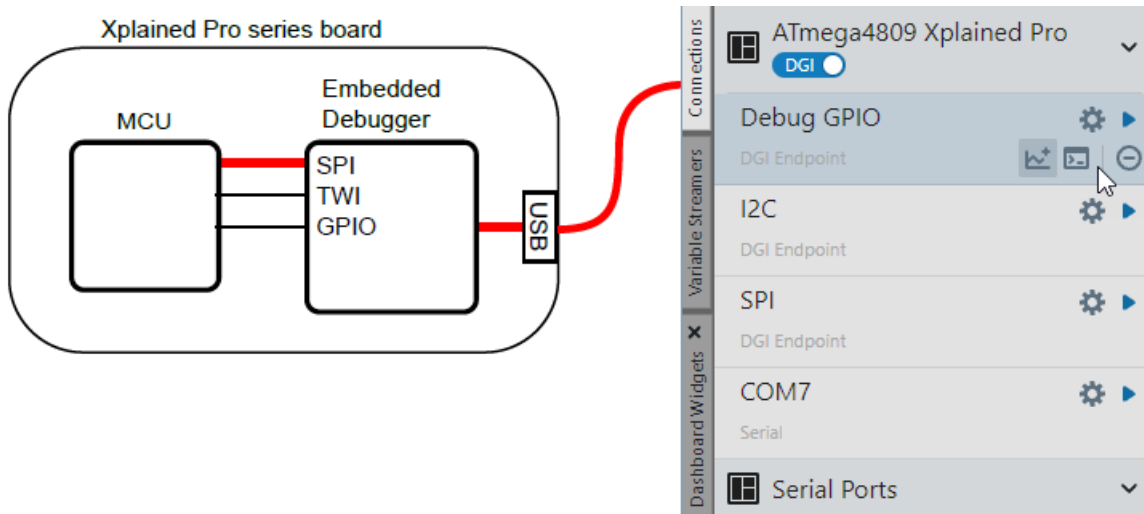


2.2.3 DGI (Data Gateway Interface)

MPLAB Data Visualizer は、DGI (Data Gateway Interface)を介してターゲットボードの組み込みデバッガと通信できます。ターゲットボードを USB ケーブルでホスト コンピュータに接続すると、ターゲットとの通信用に DGI が利用可能である事が Data Visualizer に表示され、その下に利用可能なインターフェイス(SPI、UART 等) のリストが示されます。

図 2-4 示す例では SPI インターフェイスが有効にされています。この場合、MCU は SPI ポート上で Data Visualizer と通信できます。

図 2-4. DGI (Data Gateway Interface)



2.3 インストール

Data Visualizer は MPLAB X IDE プラグインまたはスタンドアロン プログラムとして動作可能です。

Data Visualizer は v5.30 以上の MPLAB X IDE でサポートされます。


Note: Catalina 以上の macOS を使っている場合、インストール トラブル(Blocking) に関するリリースノートを参照してください(8. 「[リリースノートの閲覧方法](#)」参照)。

2.3.1 MPLAB X IDE プラグイン

MPLAB Data Visualizer プラグインは v5.50 以上の MPLAB X IDE に含まれています。Data Visualizer の各バージョンの MPLAB バージョンに対する依存性については、[リリースノート](#)を参照してください。

MPLAB X IDE 内で Data Visualizer にアクセスする方法を表 2-2 に示します。

表 2-2. MPLAB X IDE 内で Data Visualizer にアクセスする方法

方法	概要
Data Visualizer アイコン: 	このアイコンをクリックするか、ドロップダウン メニューから「MPLAB Data Visualizer」を選択すると、MPLAB X IDE 内で Data Visualizer が開きます。 ドロップダウン メニューから「DVRT for Project Name」を選択すると、DVRT セッションをプロジェクトの ELF ファイルに同期させた状態で Data Visualizer を開く事ができます。 6.2.2. 「ELF ファイルをプラグイン Data Visualizer に自動的に読み込む」 を参照してください。
MPLAB X IDE メニュー: <u>Window>Debugging>Data Visualizer</u>	Data Visualizer を開きます。
MPLAB X IDE メニュー: <u>Window>Debugging>Power Monitoring</u>	消費電力監視用に Data Visualizer を開きます。

[Tools>Plugins>Updates](#) とメニューを選択して[**Check for Updates**]をクリックすると、更新された最新のプラグインバージョンが入手可能かどうか確認できます。プラグインに関するその他の情報は、MPLAB X IDE ユーザガイド内の「[Add Plugin Tools](#)」を参照してください。

2.3.2 スタンドアロンアプリケーション

Data Visualizer をスタンドアロン プログラムとしてダウンロードおよびインストールするには、「[MPLAB® Data Visualizer](#)」ウェブページにアクセスし、インストール ウィザードに従ってください。Data Visualizer をインストールしたら、インストール位置からプログラムを起動します。

スタンドアロン アプリケーションとして使う場合、メニューバーから MPLAB X IDE メニューと同様の機能が利用できます。詳細は [2.2.1.3. 「Data Visualizer をスタンドアロンとして使う場合のメニュー」](#)を参照してください。

3. 外部接続

外部接続とは、ターゲット ハードウェアとホスト コンピュータの間のハードウェア接続の事です。これらの接続により、ターゲットと MPLAB Data Visualizer の間の通信タイプが決まります。利用可能な接続タイプはデバイス側のサポートによって決まります。









ターゲットをホスト コンピュータに接続する: デバイスまたはデモボードのマニュアルに従います。

Data Visualizer への接続を選択する: 左側の[Connections]タブで以下が選択できます。

- **DGI ツール** - DGI を備えたツールとの接続を指定します。SPI、I2C、USART、Debug GPIO を介する入カストリーミング通信が可能です。利用可能な機能セットはツールによって異なります。
- **シリアル/CDC 通信** - システム上のシリアルポートとの通信を指定します。これらのポートでは baud レート/パリティ/データビット数/ストップビット数を設定します。

3.1 接続オプション

[Connections]タブに接続が表示されたら、各データソースに対して以下のオプションが利用できます。


アイコンオプション	概要
 Source options	データソースのオプションを設定します。
 Start Streaming  Stop Streaming	データソースからの生データのストリーミングを開始/停止します。Debug DGIO の場合、全てのピンの値がストリーミングされます。グラフにプロットするピン値は、右側ウィンドウ枠で選択します。
 Add to new plot 	生データ用に新しいプロットを作成します。 生データを選択したプロットに追加します。
 Send to Terminal	ストリーミング データを[Terminal]ウィンドウに表示します。
 New variable streamer	変数値を設定してプロットします。
 Remove one or more selected visualizations	現在のプロット、[Terminal]ウィンドウの表示、Variable Streamer の全てまたは一部を削除します。

3.2 DGI (Data Gateway Interface)

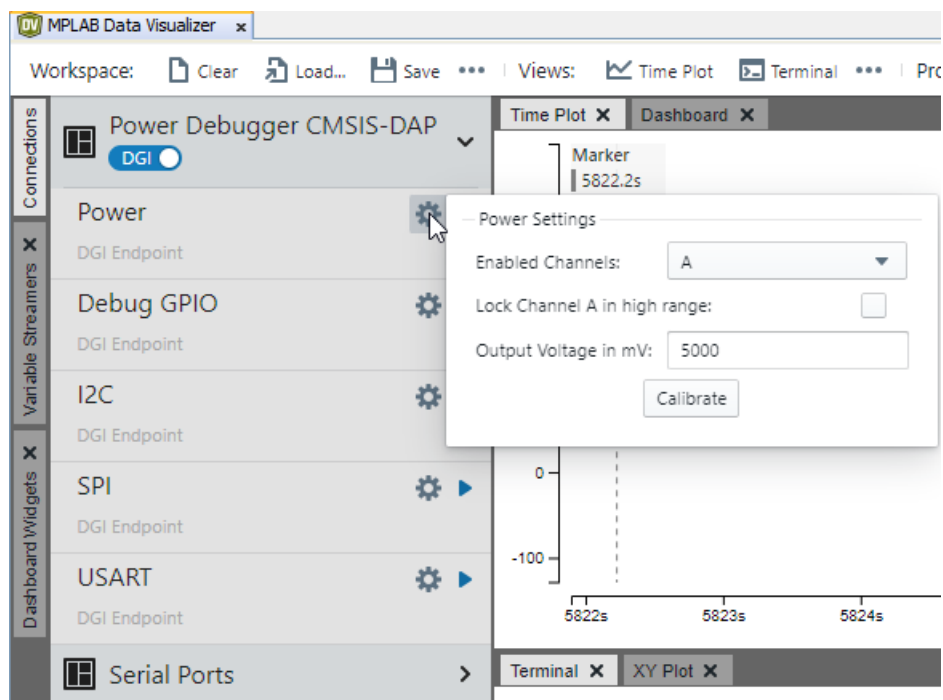
組み込みデバッガを備えた大部分のキットは DGI (Data Gateway Interface)を備えています。Data Visualizer DGI は DGI デバイスとの通信を制御可能です。左側の「Connections」タブ内で DGI を設定します。

検出された全ての DGI デバイスが **[Connections]**タブに表示されます。Data Visualizer は、DGI 対応ボードからのストリーミング入力を受け付けます。

インターフェイス オプションを設定するには、各インターフェイス(次ページの図では「Power」)のギアアイコン

 をクリックします。設定を入力した後に再度このアイコンをクリックすると、その設定が保存されて設定用ダイアログは閉じます。

インターフェイスを有効にする方法は、[4.1.3. 「データのキャプチャと表示の制御」](#)に記載しています。



3.2.1 Power インターフェイス

Power インターフェイスは、接続された回路の消費電力を計測します。Power インターフェイス ハードウェアの詳細は、消費電力の計測に使うデバッグツールのユーザガイドを参照してください。


Power インターフェイスは、デバッグツール DGI (上図の例では Power Debugger) の下で選択します。ギアアイコン  をクリックして「Power Settings」パネル内で制御パラメータを設定します(上図参照)。接続しているデバッグツールの機能に応じて設定可能なパラメータが異なります。

表 3-1. Power インターフェイスの設定

制御パラメータ	値	用途
Enabled Channels	A、A+B	有効にするチャンネル(Aのみ、または、AとBの両方)を選択します。チャンネルAは常に有効です。
Lock Channel A in high range	チェックマーク	チェックマークを付けると、Power Debugger のチャンネルAをHighレンジにロックできます(Lowレンジへの自動的な切り換えを禁止)。これにより、消費電流の短時間のスパイクを確実に検出できます。レンジをロックしないと、Lowレンジ中にスパイクが発生した場合にHighレンジへの切り換え中に重要なデータをサンプリングし損なう可能性があります。
Output Voltage in mV	1600 mV~5500 mV または 0	Power Debugger は電圧レベルの調整が可能なターゲット向け電源を備えています。このパラメータにより、電圧を設定して電源を有効にできます。0を選択すると、この電源は無効になります。
Calibrate	ボタン	このボタンをクリックすると、電流計測回路のキャリブレーションが実行されます。最大限の計測精度を達成するには、計測の前に電流計測ハードウェアをキャリブレートする必要があります。



ヒント: 全ての設定変更は、「Power Settings」パネル内の[Apply]をクリックするまで実際には適用されません。例えば、電源の出力電圧を有効にする場合、「Output Voltage」で電圧値を設定した後に[Apply]をクリックするまで、設定した電圧は出力されません。



ヒント: デバッガが既に Low レンジで動作中である時にチャンネル A を High レンジにロックしてもデバッガはすぐに High レンジに切り換わりません。デバッガを High レンジへ戻すには、電流レベルの上昇によって High レンジへ自動的に切り換わるまで待つか、デバッガを一度停止して再起動する必要があります。



ヒント: 全ての Power 信号(電流、電圧、電力) を時系列プロットに表示するとシステムリソースを消費します。表示性能を向上させるには、同時に表示するプロットの数減らす必要があります。

3.2.2 GPIO インターフェイス

GPIO インターフェイスは、有効にされている Debug GPIO ピンのビット値を転送します。ピンがトグルするたびに、符号なし 8 ビットデータを格納した 1 つのパケットが送信されます。GPIO インターフェイスの物理部に関する詳細は、GPIO データのサンプリングに使うデバッグツールのユーザガイドを参照してください。

左側の[Connections]タブ内で GPIO インターフェイスを選択すると、GPIO 設定が下方のセクションに表示されます。

表 3-2. GPIO 設定

フィールド名	値	用途
GPIO 0 Change Triggers Bus Read	ON、OFF	バス読み出しをトリガするために GPIO ピン 0 の遷移を監視します。
GPIO 1 Change Triggers Bus Read	ON、OFF	バス読み出しをトリガするために GPIO ピン 1 の遷移を監視します。
GPIO 2 Change Triggers Bus Read	ON、OFF	バス読み出しをトリガするために GPIO ピン 2 の遷移を監視します。
GPIO 3 Change Triggers Bus Read	ON、OFF	バス読み出しをトリガするために GPIO ピン 3 の遷移を監視します。



重要: Debug GPIO データソースをプロットする場合、全ての GPIO ピンがサンプリングされますが、遷移トリガを有効にした GPIO ピンが遷移した時にのみサンプリングがトリガされます。例えば、「GPIO n Change Triggers Bus Read」を GPIO 3 でのみ有効にした場合、GPIO 値は GPIO 3 が遷移した時にのみサンプリングされます。すなわち、GPIO 3 の遷移時にのみ 4 つの GPIO 値の全てが読み出されます。

3.2.3 USART インターフェイス

USART データソースは、USART インターフェイス上で受信したデータをストリーミングします。USART インターフェイスの物理部に関する詳細は、USART データのサンプリングに使うデバッグツールのユーザガイドを参照してください。

左側の[Connections]タブ内で USART ソースを選択すると、USART 設定が下方のセクションに表示されます。

Note: 非同期シリアル プロトコル(例: DGI USART および CDC Virtual COM ポート インターフェイスによって使われる UART プロトコル)は、3.3.「シリアルポート」に記載した **baud レート**を使います。

表 3-3. USART 設定

フィールド名	値	用途
Baud Rate	0-2000000	非同期モードの UART インターフェイス向け baud レート
Char Length	5、6、7、8 ビット	各転送データのビット数
Parity	None、Even、Odd、Mark、Space	通信に使うパリティのタイプ
Stop bits	1、1.5、2 ビット	ストップビットの数

3.2.4 I2C インターフェイス

I2C ソースは、I2C インターフェイス上で受信したデータをストリーミングします。I2C インターフェイスの物理部に関する詳細は、I2C データのサンプリングに使うデバッグツールのユーザガイドを参照してください。

I2C 設定オプションは、左側の[Connections]タブの **DGI** セクション内の **I2C** インターフェイスの下に表示されます。

I2C インターフェイスは、左側の[Connections]タブの **DGI** セクションの下に表示されます。I2C 接続を選択すると、I2C 設定はこのウィンドウ枠の下方のセクションに表示されます。

表 3-4. I2C 設定

フィールド名	値	用途
Speed	0	インターフェイスの予想速度を設定します。スレーブデバイスは、この設定に基づいてタイミングを調整します。最大 400 kHz までサポートされます。
Address	1	スレーブデバイスのアドレスです。
Kit-side Timestamping	チェックマーク	ターゲットのタイムスタンプを有効にします。

3.2.5 SPI インターフェイス

SPI ソースは、SPI インターフェイス上で受信したデータをストリーミングします。SPI インターフェイスの物理部に関する詳細は、SPI データのサンプリング用に使うデバッグツールのユーザガイドを参照してください。



重要: SPI ハードウェア モジュールは、アクティブ Low のチップセレクト(CS) 信号を使います。CS ピンが High 状態の時に送信されたデータは全て無視されます。

I2C インターフェイスは、左側の[Connections]タブの **DGI** セクションの下に表示されます。SPI 接続を選択すると、SPI 設定はこのウィンドウ枠の下方のセクションに表示されます。

表 3-5. SPI 設定

フィールド名	値	用途
Char Length	5、6、7、8 ビット	各転送データのビット数
Mode	<ul style="list-style-type: none"> Clock idle normally low, Sample data on rising edge Clock idle normally low, Sample data on falling edge Clock idle normally high, Sample data on falling edge Clock idle normally high, Sample data on rising edge 	SPI モードを選択します。これに基づいてクロック位相とサンプリングが制御されます。
Force CS Sync	チェックマーク	SPI インターフェイスはチップセレクト(CS)ラインが 2 回トグルした後にのみ有効になります。
Kit-side Timestamping	チェックマーク	ターゲットのタイムスタンプを有効にします。

3.3 シリアルポート

Data Visualizer は、ホスト コンピュータの標準シリアルポートを介してターゲットボードに接続できます。左側の[Connections] タブ内でシリアル通信を設定します。

baud レート、ストップビット数、パリティを通信先の要件に応じて設定する必要があります。シリアルポート データは、グラフと[Terminal]ウィンドウ内で符号なし 8 ビットデータとして扱われます。

表 3-6. シリアルポート設定

フィールド名	値	用途
Baud rate	9600-2000000	シリアル インターフェイスの baud レート
Char Length	5、6、7、8 ビット	各転送データのビット数
Parity	None、Even、Odd、Mark、Space	通信に使うパリティのタイプ
Stop bits	1/1.5/2 bit	ストップビットの数

Note: 非同期シリアル プロトコル(例: DGI USART および CDC 仮想 COM ポート インターフェイスによって使われる UART プロトコル) は、以下の **baud レート** を使います。

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

上記以外の baud レートは、非同期インターフェイス(DGI UART とシリアルポート/CDC 仮想 COM ポート)を介するプロトコル向けに使いません。

tty ポートと cu ポートの違い

Godfrey van der Linden によって Apple® Listserv に投稿された記事 (lists.apple.com/archives/darwin-dev/2009/Nov/msg00099.html) を参照してください。

4. データの表示

受信データは、以下で説明するモジュールを使って表示できます。

- **ターミナル(Terminal)** は、データをテキスト(生データまたは ASCII コード文字)として表示します。ターミナルからテキストベース データを送信する事もできます。
- **時系列プロット(Time Plot)** は、受信データの時系列変化をプロット、バンド、文字列フラグとして表示します。データの解析(例: しきい値を設定するために出力値を読み取る等)にはカーソルが役立ちます。
- **消費電力解析(Power Analysis)**モジュールは、消費電力の時間変化を解析するために専用設計されています。プログラム カウンタ(PC)サンプルを表示するコード プロファイリング機能と一緒に使う事で、プログラム実行と消費電力の関係を調べる事もできます。
- **XY プロット(XY Plot)** は、受信した x データと y データを XY グラフに表示します。
- **ダッシュボード(Dashboard)**ではウィジェットを使って受信データを表示できる他、アプリケーションへ送信するデータを入力する事ができます。

4.1 データのキャプチャと表示

ターゲットをホスト コンピュータに接続すると、Data Visualizer に利用可能なデータソースが表示されます。以下で説明する各種の方法により、これらのデータソースからのデータのストリーミングを開始できます。

4.1.1 最初にデータの表示方法を選択する場合




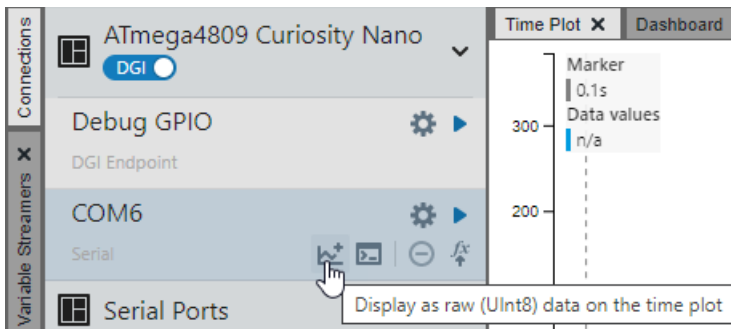

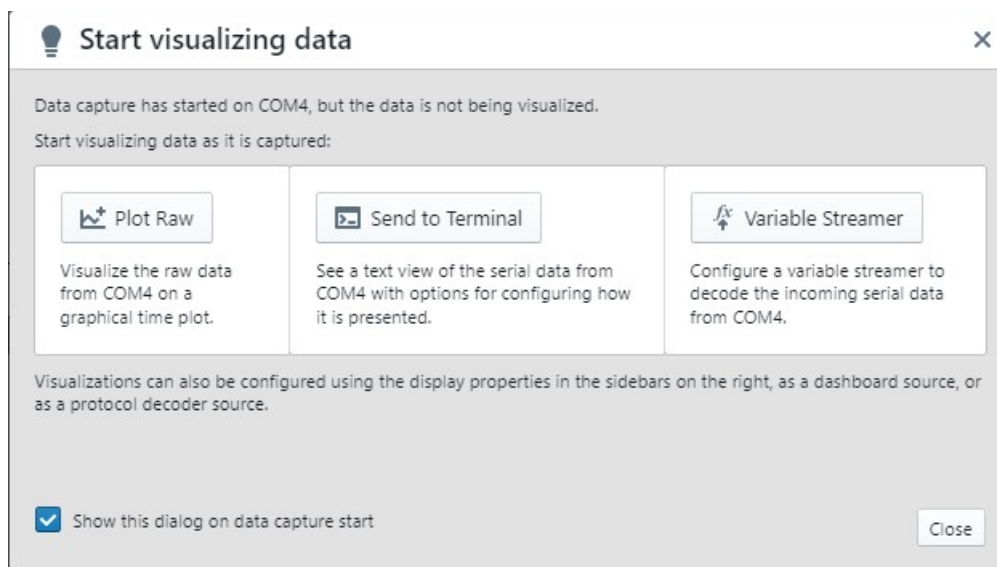
データソース名の 1 つにマウスカーソルを重ねると、利用可能な表示オプションのアイコン(例:  (Plot raw data)、 (Display text in the terminal)、) が表示されます(下図参照)。アイコンを選択すると、そのオプションを使ってデータのストリーミングが始まります。1 つまたは複数の表示方法を選択できます。

図 4-1. データ表示方法の選択



4.1.2 最初にデータのキャプチャを開始する場合

データ表示オプションを選択する前にキャプチャ開始ボタン  をクリックした場合、クイック アクション ダイアログが開いて利用可能な表示オプションが表示されます(次ページの図参照)。1 つまたは複数のクイック アクション ボタンをクリックできます(例: [Plot Raw] をクリックして時系列プロットで表示を開始)。

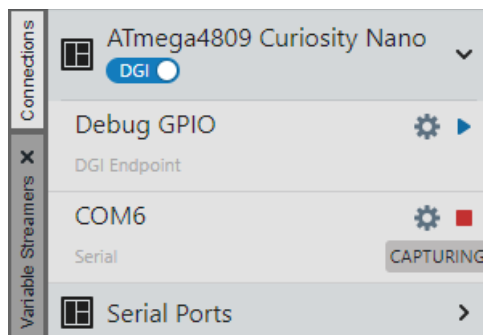




Note: [Show this dialog on data capture start] のチェックマークを外すと、データキャプチャ開始時にこのダイアログは表示されなくなります。このダイアログを再び表示させるには、メニューで [Tools>Options>Embedded>MPLAB Data Visualizer](#) を選択します。

4.1.3 データのキャプチャと表示の制御

データソース名からマウスカーソルを外すと、データ表示オプションのアイコンは消えて、メッセージ「CAPTURING」が表示されます(下図参照)。

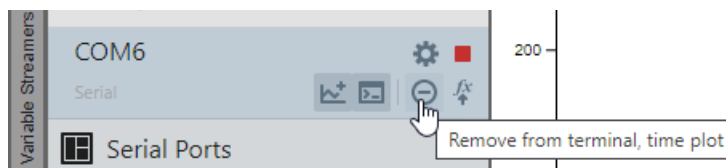
図 4-2. データキャプチャ実行中の状態



キャプチャ停止ボタン  をクリックするとデータフローは停止しますが、データ表示設定は保持されます。キャプチャ開始ボタン  を再度クリックすると、データフローは再開します。

全てのデータキャプチャおよび表示方法を削除するには、そのデータソースの上にマウスカーソルを重ねて削除アイコンをクリックします(図 4-3 参照)。

図 4-3. 表示およびデータキャプチャの削除



あるいは、以前に選択した表示オプションのアイコンを再度クリックする事により、データキャプチャおよび表示方法を個別に削除する事もできます。

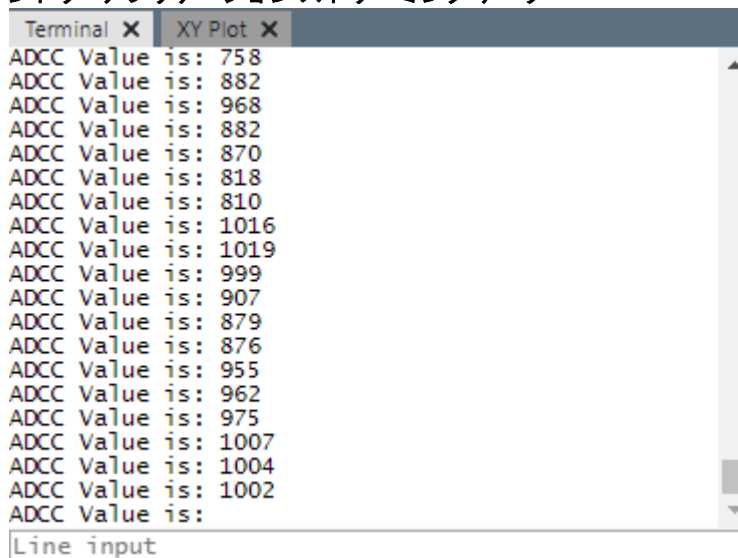
4.2 ターミナル([Terminal]ウィンドウ)にデータをテキストとして表示する

ターミナルはテキストまたは数値を表示または送信するためのデータ端末です。

4.2.1 [Terminal]ウィンドウ

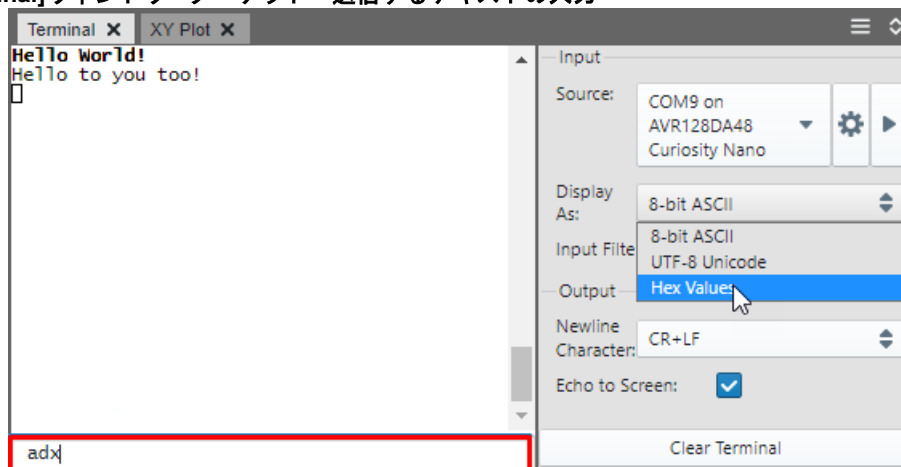
MPLAB Data Visualizer の[Terminal] ウィンドウは、ターゲットからのストリーミング データを各種書式で表示します。

図 4-4. [Terminal]ウィンドウ - アプリケーションストリーミング データ



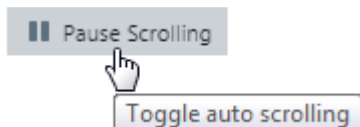
[Terminal] ウィンドウから COM ポートを通じてターゲットへ文字列またはテキスト行を送信することも可能です。この場合、ターミナル領域内またはテキストボックス(入力行) にテキストをタイプします(図 4-5 参照)。ターミナルのエコーは太字で表示されます。テキスト ボックスに不正な文字を入力すると、ボックスの枠が赤で示されます。

図 4-5. [Terminal]ウィンドウ - ターゲットへ送信するテキストの入力



左側の[Connections]タブを使ってデータソースを選択します。baud レートが正しいことを確認します。[Terminal] ウィンドウの右側のウィンドウ枠で、ドロップダウン リストから 1 つまたは複数のフォーマットを選択します。有効にするとターミナルは常時アクティブになり、ターゲットから受信したデータをスクロールしながら表示するか、ターゲットへ送信するテキストの入力を受け付けます。





[Terminal] ウィンドウの最上部にマウスカーソルを重ねて[Pause Scrolling / Scroll to End] ボタンをクリックすることで、表示のスクロールを停止または再開できます。スクロールを停止しても、データのストリーミングはバックグラウンドで続行されます。



スクロールを一時停止した状態で<Ctrl + C>および<Ctrl + V> (MAC OS では<Cmd + C>および<Cmd + V>)を使う事で、ウィンドウ内のテキストをコピー/ペーストできます。

4.2.2 [Connections]タブ

左側の[Connections]タブには利用可能な接続が表示されます。ターゲットをホスト コンピュータに接続すると、その接続とターゲット上のデータソースが表示されます。

ターミナルヘータを送信可能なデータソース([Enable send to terminal] アイコン  が表示されるデータソース)を見つけます。[Source options]アイコン  をクリックしてソース設定を表示し、必要に応じて設定を変更します。次に  をクリックすると、そのデータソースはターミナルに接続されます。接続した後に設定を変更する必要が生じた場合、 (接続後は[Disable send to terminal] と表示される)をクリックする事により接続を無効にします。設定を変更した後に、再度このボタンをクリックする事により再接続します。

接続の詳細は、3. 「外部接続」を参照してください。

4.2.3 ターミナルの表示設定

[Terminal] ウィンドウの右側のウィンドウ枠(図 4-5 参照)では、データソースを選択してストリーミング データのフォーマットを制御します。

表 4-1. 入力データの設定(図 4-5 参照)

制御パラメータ	概要
Source	Debug GPIO 以外の全てのデータソースがサポートされます。
Display As	データストリームをターミナル文字へ変換する方法を選択します。現在、以下が選択可能です。 <ul style="list-style-type: none"> • ASCII • UTF-8 • HEX 値 Note: このモードでは改行文字は出力されません。
Input Filtering	チェックマークを付けると、入力ストリームから ANSI/VT100 端末制御文字(1B、90、98、9B、9D、9E、9F) が排除されます(これらの制御文字は、組み込み端末コンポーネントに対して特殊な意味を持ちます)。

出力データの制御パラメータは、COM ポート接続向けにのみ有効です。

表 4-2. 出力データの設定(図 4-5 参照)

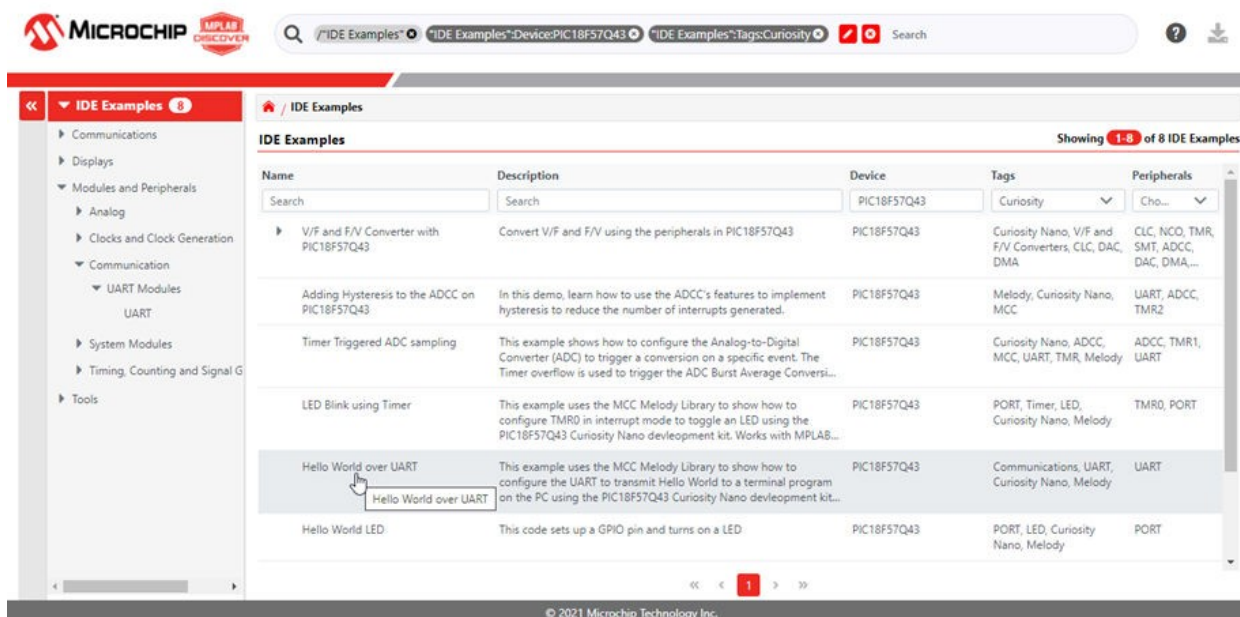
制御パラメータ	概要
Newline Character	出力ストリーム内で改行を表す文字を選択します。 <ul style="list-style-type: none"> • None • CR+LF: キャリッジ リターン + ラインフィード • LF: ラインフィード
Echo to Screen	チェックマークを付けると、入力した文字が画面にエコーバックされます。

[Terminal] ウィンドウ内の内容を削除するには[Clear Terminal] ボタンをクリックします。

4.2.4 ターミナル向けサンプルコード

MPLAB Discover 等を使ってサンプルコードを見つけます(図 4-6 参照)。メインページのリストで検索条件を指定します(「Device」列で PIC18F57Q43 を指定し、「Tags」列で Curiosity を指定)。リストから「Hello World over UART」という名前のサンプルコードを見つけます。

図 4-6. MPLAB Discover - サンプルコードの検索



このサンプルコードをクリックすると、使用ソフトウェア/ハードウェア、コード開発時の MCC Melody の使用方法、Data Visualizer での出力表示方法を記載したページが開きます(図 4-7 参照)。サンプルコードは、このページからダウンロードする事もできます。

図 4-7. MPLAB Discover - サンプルコード「Hello World over UART」のページ

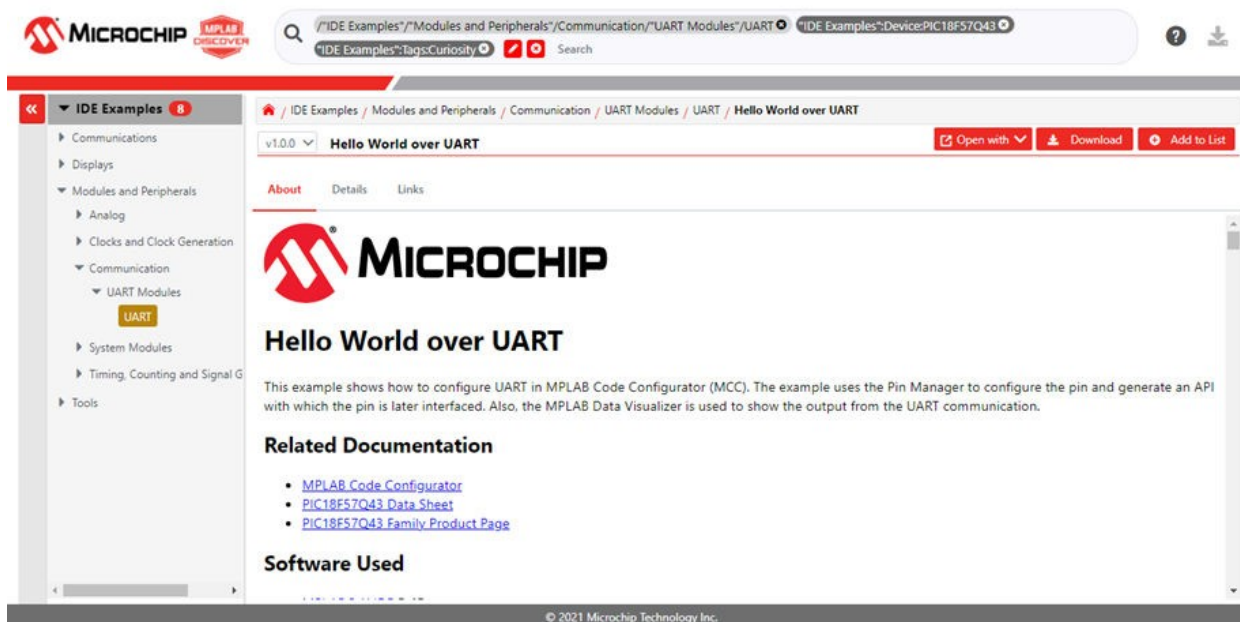
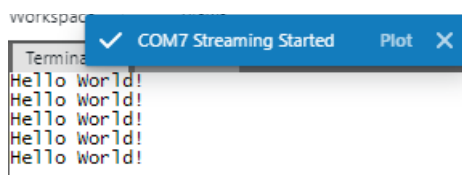


図 4-8. ターミナル出力



4.3 時系列プロットにおけるデータ表示

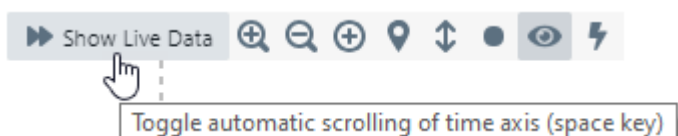
時系列プロット(Time Plot)は多目的に使えるグラフ表示ツールです。大きなプロット領域に1つの時間軸と、1つまたは複数のデータ軸(Y軸)を持つグラフを表示できます。1つのデータ軸を使って全てのデータを表示する事も、複数のデータ軸を使って複数のプロットに分けて表示する事もできます。

4.3.1 [Time Plot] ウィンドウ

[Time Plot] ウィンドウは、時間軸に対するデータの変化を示します。左側の[Connections] タブを使ってデータソースを選択します。[Variable Streamers] タブは、データストリーム プロトコル向けに使用します。右側のウィンドウ枠では、軸オプションを選択して複数のデータ軸を追加できます。

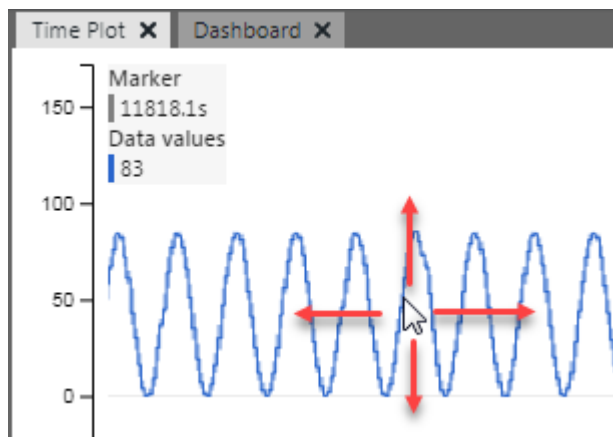
4.3.1.1 プロットのスクロール

プロットのスクロールを有効にしないと、ストリーミングが実行中であってもデータが見えない場合があります。グラフの最上部にマウスカーソルを置いて、[Show Live Data] ボタンをクリックすると、プロットのスクロールが始まります。スクロールが始まると、このボタンの名前は[Pause Scrolling] に変化し、これをクリックするとスクロールは止まります。



あるいは、ダブルクリックするかスペースキーを押す事によってデータのスクロールを停止させる事もできます(データのストリーミングはバックグラウンドで続行します)。再度ダブルクリックするかスペースキーを押すとスクロールは再開します。

手でプロットを移動させるには、そのプロットをクリックしてドラッグします。データ軸をドラッグおよび停止するには、プロットの左側をクリックします。時間軸をドラッグおよび停止するには、プロット上またはウィンドウの中央部をクリックします。



4.3.1.2 プロットの拡大/縮小

いずれか1つの軸をクリックしてから以下の操作を行います。





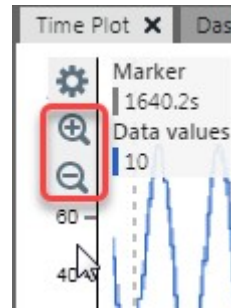

- その軸上でマウスホイールを使って拡大/縮小する。
- グラフ領域内でマウスホイールを使って拡大/縮小する。
- グラフの最上部にマウスカーソルを置き、  をクリックする。
- いずれかの軸の上にマウスカーソルを置き、  をクリックする(図 4-9 参照)。

図 4-9. 軸の拡大/縮小



4.3.1.3 データ軸を追加する

データ軸を追加するには、グラフの最上部にマウスカーソルを置いてアイコン  をクリックします。

右側のウィンドウ枠内で軸のプロパティを設定するか(図 4-10 参照)、軸上にマウスカーソルを置いてギアアイコン (Axis Options) をクリックします(図 4-11 参照)。右側ウィンドウ枠内でデータ軸を追加することもできます(図 4-10 参照)。どちらの方法でも、軸にプロットを追加するための制御オプションが利用できます。

図 4-10. 軸オプション(右側ウィンドウ枠)

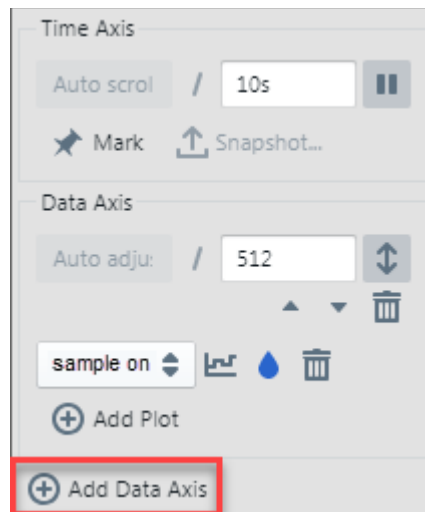
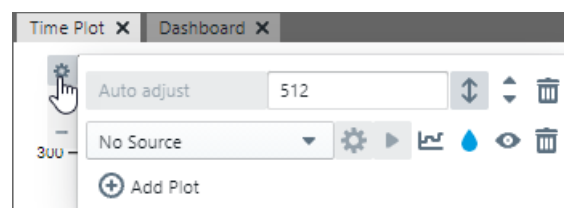


図 4-11. 軸オプション




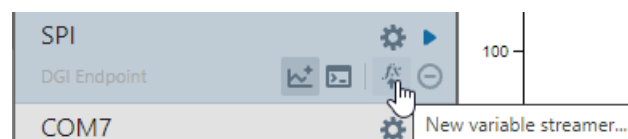
加えて、軸を設定した後は、左側の[Connections]タブ内で、選択したデータソースの  (Plot raw data) ボタンをクリックする事により、データをプロットできます(図 4-12 参照)。

図 4-12. データソースの[Plot]ボタン



4.3.1.4 プロット値の表示

MPLAB Data Visualizer は、プロットデータを表示および解析するために以下の機能を内蔵しています。

4.3.1.4.1 プロットマーカー


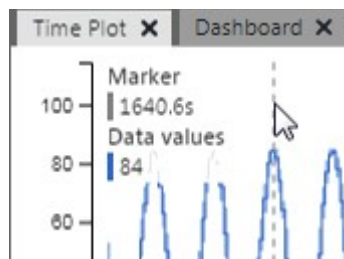
時系列プロットのマーカーは縦の破線で示され、マウスの時間軸方向の動きに追従して移動します。マーカーの現在位置における時間とデータ値がグラフの左上隅に表示されます(図 4-13 参照)。これらの値の表示/非表示を切り換えるには、グラフの最上部にマウスカーソルを置いて、[Toggle Inspect Values] アイコン  をクリックします。

図 4-13. マーカーとマーカー値の表示



4.3.1.4.2 プロットカーソル


プロットカーソルはプロットマーカーと同様の方法で使えますが、マウスの動きに追従しないため、ドラッグして移動させる必要があります。1 本または複数のカーソルを追加するには、グラフの最上部にマウスカーソルを置いて、[Add vertical cursor...]アイコン  をクリックします。グラフの左上隅には、マーカー値の代わりにカーソル値が表示されます(図 4-14 参照)。

図 4-14. 縦カーソル A

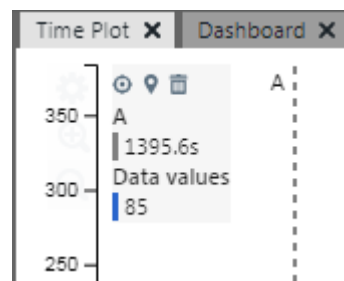





表 4-3. マーカー値の表示内容

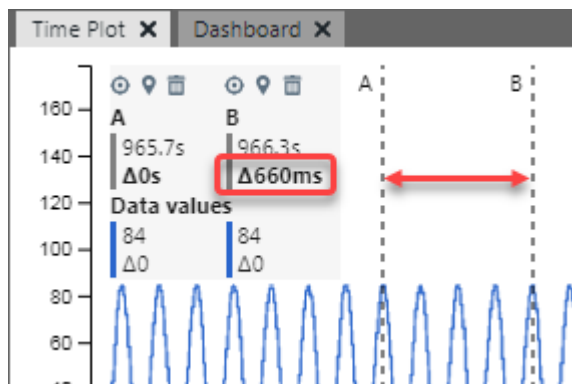
	このアイコンをクリックするとプロットマーカー位置がグラフの中央に表示されスクロールは停止します。スクロールによってマーカーが表示域外へ移動してしまった場合にマーカー位置へジャンプできます。
	グラフの中央にマーカーを再配置します。シフトキーを押しながらこのアイコンをクリックする事により、全てのマーカーを同じ時間幅を維持したまま移動させる事ができます。
	マーカーを削除します。
A、B、C、....	マーカーの識別文字です。
時間値	各マーカーの時間軸上の位置(タイムスタンプ)を示します。複数のマーカーが存在する場合、左端のマーカー位置からの時間差(Δ 値)が表示されます。
周波数値	複数のマーカーが存在する場合、時間軸目盛りの上にマウスカーソルを置くと、左端のマーカー位置を基準とした周波数が表示されます。
データ値	各マーカー位置におけるデータの値です。データ値の左にプロットと同色の縦バーが表示されます。複数のプロットが存在する場合、この色で見分ける事ができます。複数のマーカーが存在する場合、左端マーカーのデータ値との差(Δ 値)が表示されます。

2本のカーソルを使った帯域幅の計算

2本の縦カーソルを使って帯域幅を計算できます。図 4-15 の例では、 Δ 時間値からカーソル位置 A と B の間の時間差が 660 ms である事が分かります。従って、帯域幅は $660 \text{ ms} / 4 \text{ cycle} \cong 165 \text{ ms/cycle}$ です。

Note: 帯域幅はサイクルごとに変動する可能性があるため、通常は複数サイクルを含む時間を計測して平均値を求める事を推奨します。

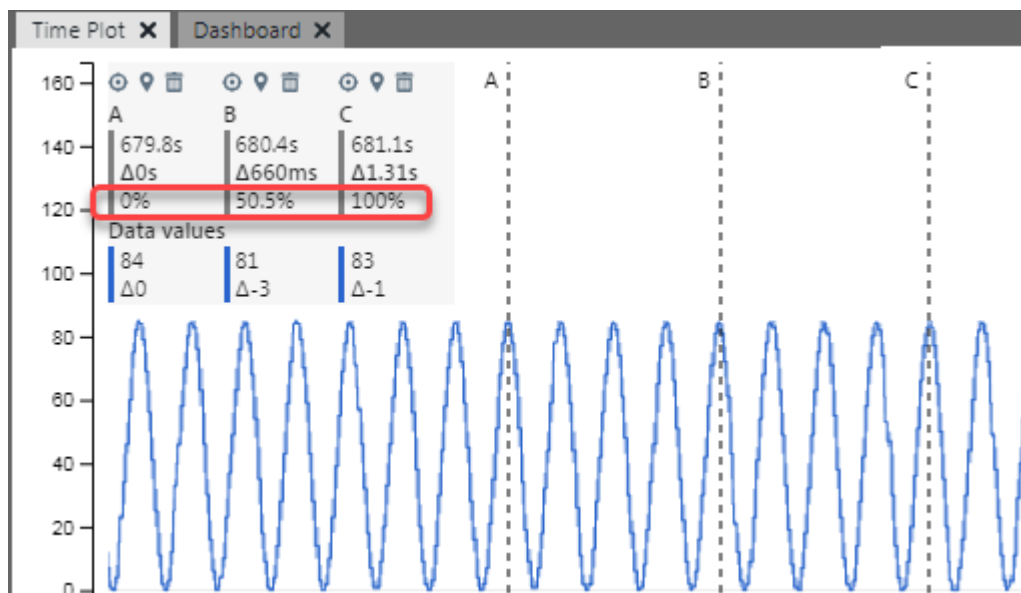
図 4-15. 2本のカーソルを使った帯域幅の計算



3本のカーソルを使ったデューティサイクルの計算

3本のカーソルを使うとデューティサイクルを計算できます。図 4-16 の例の場合、カーソル A-C 期間を周期とすると、カーソル A-B 期間の周期に対する百分率は 50.5%です。

図 4-16. 3本のカーソルを使ったデューティサイクルの計算



カーソルはさらに追加できます。

4.3.1.5 データ軸の自動調整


グラフの最上部にマウスカーソルを置いて[Toggle Auto Adjust]アイコン  をクリックする事により、全てのデータ軸の自動調整の有効/無効を切り換える事ができます。データ軸上でダブルクリックする事により、軸ごとに自動調整の有効/無効を切り換える事もできます。

図 4-17. 自動調整適用前

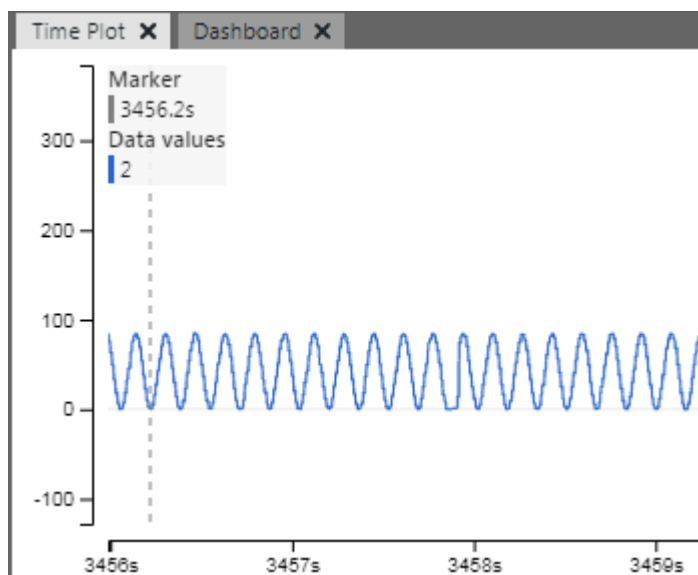
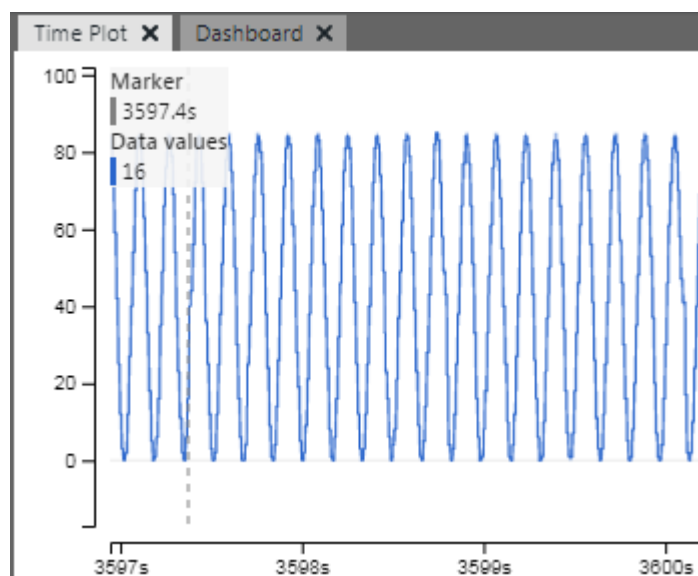
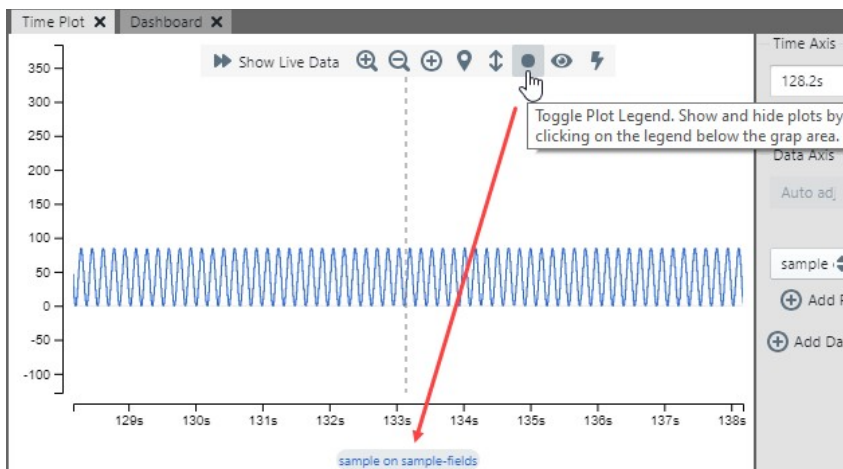


図 4-18. 自動調整適用後






4.3.1.6 凡例の表示


時系列プロットの下に、選択されたデータソースを示す凡例を表示できます。次ページの図に示すアイコンを使うと、この凡例の表示/非表示を切り換える事ができます。



4.3.2 [Connections]タブ

左側の[Connections]タブには利用可能な接続が表示されます。ターゲットをホスト コンピュータに接続すると、その接続とターゲット上のデータソースが表示されます。

必要なデータソースの[Source options]ボタン  をクリックしてソース設定を表示し、必要に応じて設定を変更します。次に  をクリックすると生データがグラフにプロットされます。接続後に設定を変更する必要がある場合、再度  をクリックする事により接続を切断します。設定を変更した後に、再びこのボタンをクリックして再接続します。

Note:  をクリックしてデータソースを時系列プロットから切断しても、**ダッシュボード**上のウィジェット プロットからは切断されません。接続の詳細は、3. 「外部接続」を参照してください。

4.3.3 [Variable Streamers]タブ

画面の左側で[Variable Streamers]タブをクリックして開くと、Variable Streamer を作成、編集、削除できます。**[Import Variable Streamer]**ボタンを使うと、Atmel Studio Data Visualizer で使われた .ds または .txt ファイルをインポートできます(図 4-19 参照)。フィル書式の詳細は、下の文書内の 5.1.1 「Configuration Format」に記載されています。

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

[View decoding statistics]アイコンをクリックすると、リアルタイムでデコーディング統計値(ストリーミング データ情報)を見る事ができます(図 4-20 参照)。

データストリーム内に変数が存在する場合、変数を表示する前にストリームをデコードする必要があります。詳細は 5. 「Variable Streamer」を参照してください。

図 4-19. [Variable Streamers]タブ

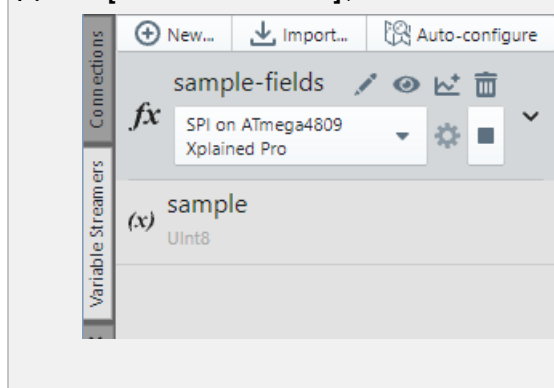
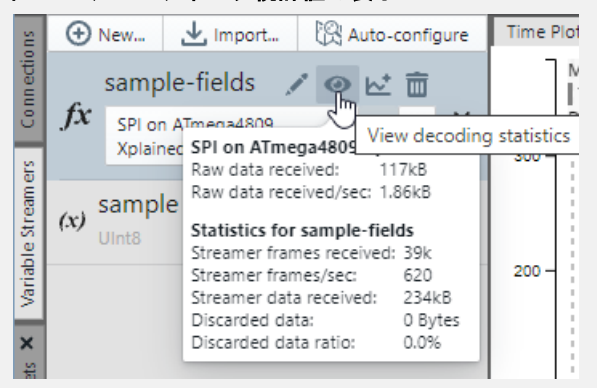


図 4-20. デコーディング統計値の表示



4.3.4 時系列プロットの制御パラメータ(右側ウィンドウ枠)

時系列プロットの右側のウィンドウ枠では、ストリーミング データのグラフ表示パラメータを設定します(図 4-21 参照)。

図 4-21. 時系列プロットの制御パラメータ

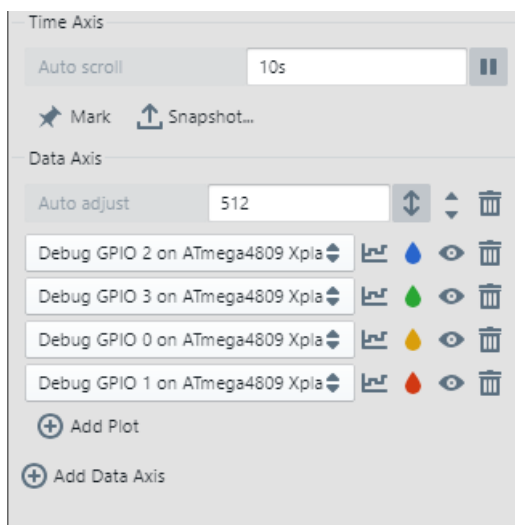


表 4-4. 時間軸





制御項目	概要
Offset	スクロールが有効である場合、「Auto scroll」と表示されます。スクロールが無効である場合、オフセット (グラフの左端の時間)が表示されます。
Scale	時間軸の分解能(単位: sec)を指定します。
 	スクロールを停止します。 スクロールを開始/続行します。
	グラフの表示を現在位置で固定させ、時間軸上の左端と右端にマーカーを表示します。これらのマーカーによりスナップショットの範囲が決まります。マーカー位置は必要に応じて調整できます。
	データソース プロットのスナップショットをテキストファイル(CSV または JSON)に保存します。スナップショットを取り込む前に上記の[Mark]を実行する必要があります。

表 4-5. データ軸









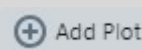
制御項目	概要
Offset	データ軸の自動調整が有効である場合、「Auto adjust」と表示されます。自動調整が無効である場合、プロットの現在のオフセットが表示されます。
Scale	データ軸の分解能を指定します。
	自動調整の有効/無効を切り換えます。 有効である場合、軸のレンジは自動的に調整されます。 無効である場合、軸のレンジは手動で調整します。
	複数のデータ軸が存在する場合、この軸の配置順序を 1 つ上または 1 つ下に移動させます。
	グラフからこの軸を削除します。
	この軸の下に別のデータ軸を追加します。

表 4-6. データ軸 - プロットのソースと書式

制御項目	概要
Data Source	プロットするデータソースをドロップダウン リストから選択します。データソースの選択と設定は左側の[Connections]タブを参照してください。
	データ点のプロット方法を選択します。 <ul style="list-style-type: none"> データ点をステップ状の線で結びます(既定値)。 データ点を直線で結びます。 データ点のみを表示します(線なし)。
	プロットの色を選択します。
	プロットの表示/非表示を切り換えます。
	このプロットを削除します。
	この軸にプロットするデータソースを追加します。

4.3.5 時系列プロットの例

以下では、時系列プロットの使用例を示します。

4.3.5.1 データプロットの例

ATmega4809 Xplained Pro ボードを使って生成したデータを DGI/SPI 経由で Data Visualizer へ出力し、Variable Streamer を使ってプロットします。

4.3.5.1.1 セットアップ

以下の手順でサンプル ソフトウェアとハードウェアをセットアップします。

MPLAB X IDE

以下のリンクから MPLAB X IDE 6.05 以上を無償でダウンロードしてインストールします。

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

以下のパックを選択します。

ATmega_DFP v3.0.158 以上、EDBG_TP v1.5.495 以上

MPLAB XC8 C コンパイラ

以下のリンクから MPLAB XC8 C コンパイラ v2.35 以上を無償でダウンロードしてインストールします。追加の最適化機能とその他の機能を備えた PRO バージョンのコンパイラは有償で入手可能です。

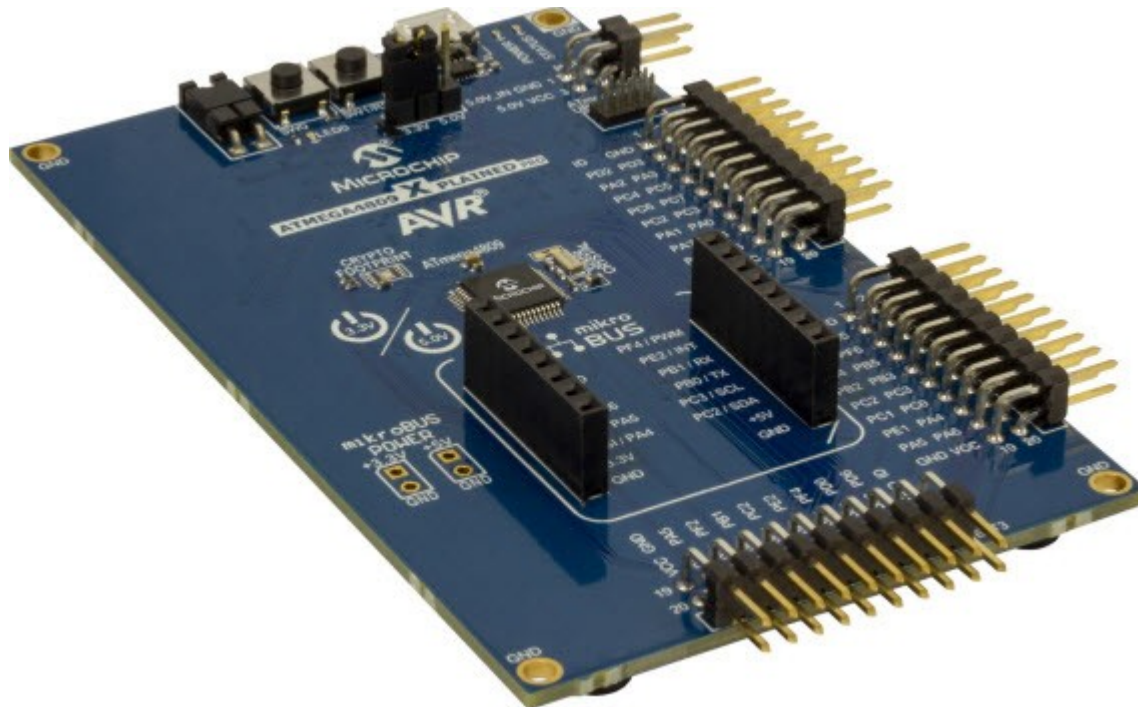
www.microchip.com/mplab/compilers

ATmega4809 Xplained Pro ボード - ATMEGA4809-XPRO

この評価用ボード(次ページの写真参照) は、Microchip 社のオンライン販売サイトまたは販売代理店から購入できます。同梱の USB ケーブルを使って本ボードをホスト コンピュータに接続し、ドライバをインストールします。

本ボードの詳細は以下を参照してください。

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



4.3.5.1.2 サンプルプロジェクトを作成する

MPLAB X IDE を使ってアプリケーションコードを開発するためにプロジェクトを作成する必要があります。

準備

プロジェクトを作成する前に以下の準備が必要です。

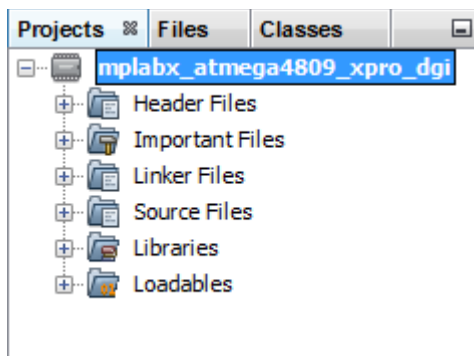
- コンパイラがインストール済みであり、MPLAB X IDE によって検出される必要があります。コンパイラが検出されない場合、メニューで [Tools>Options>Embedded>BuildTools](#) を選択してツールチェーンリストを表示します。その中にコンパイラが存在しない場合、**[Add]**をクリックしてインストール済みのコンパイラを指定します。
- USB ケーブルを介して Xplained Pro ボードがホストコンピュータに接続済みである必要があります。

プロジェクトを作成する

メニューから [File>New Project](#) を選択するか、**[New Project]**アイコン  をクリックして、プロジェクトウィザードを開きます。以下の手順によりプロジェクトを作成します。**[Next]**をクリックして次の手順に進みます。

1. **プロジェクトを選択する:** [Microchip Embedded] カテゴリをクリックしてから[Standalone Project] プロジェクトを選択します。
2. **デバイス(およびツール)を選択する:** デバイス名「ATmega4809」を入力します。次にツール名「ATmega4809 Xplained Pro- SN:ATML#」を入力します。ツールのシリアル番号(SN)は、接頭辞「ATML」に続く複数桁の数字です。
3. **コンパイラを選択する:** メニューで [Compiler Toolchains>XC8](#) と選択し、最新のコンパイラバージョンを選択します。
4. **プロジェクト名とフォルダを選択する:** プロジェクトの名前を指定します(例: mplabx_atmega4809_xpro_dgi)。Windows OS の場合、既定値プロジェクトフォルダは C:\Users\\MPLABXProjects です。

[Finish]をクリックすると、[Project]ウィンドウにプロジェクトツリーが表示されます。



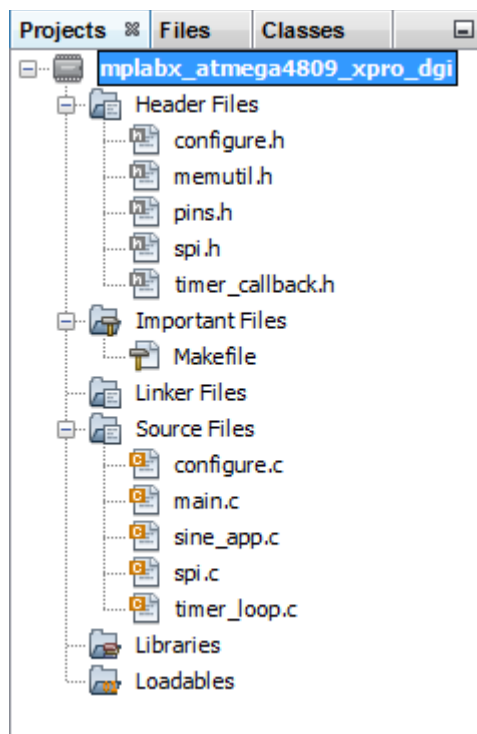
4.3.5.1.3 プロジェクトにファイルを追加する

C 言語ヘッダファイルを追加するため、「Header Files」フォルダを右クリックし、メニュー New>C Header File を選択します。

C 言語ソースコードを追加するため、「Source Files」フォルダを右クリックし、メニュー New>C Main File を選択して `main.c` を追加します。その他のソースファイルは全てメニュー New>C Source File を選択して追加します。

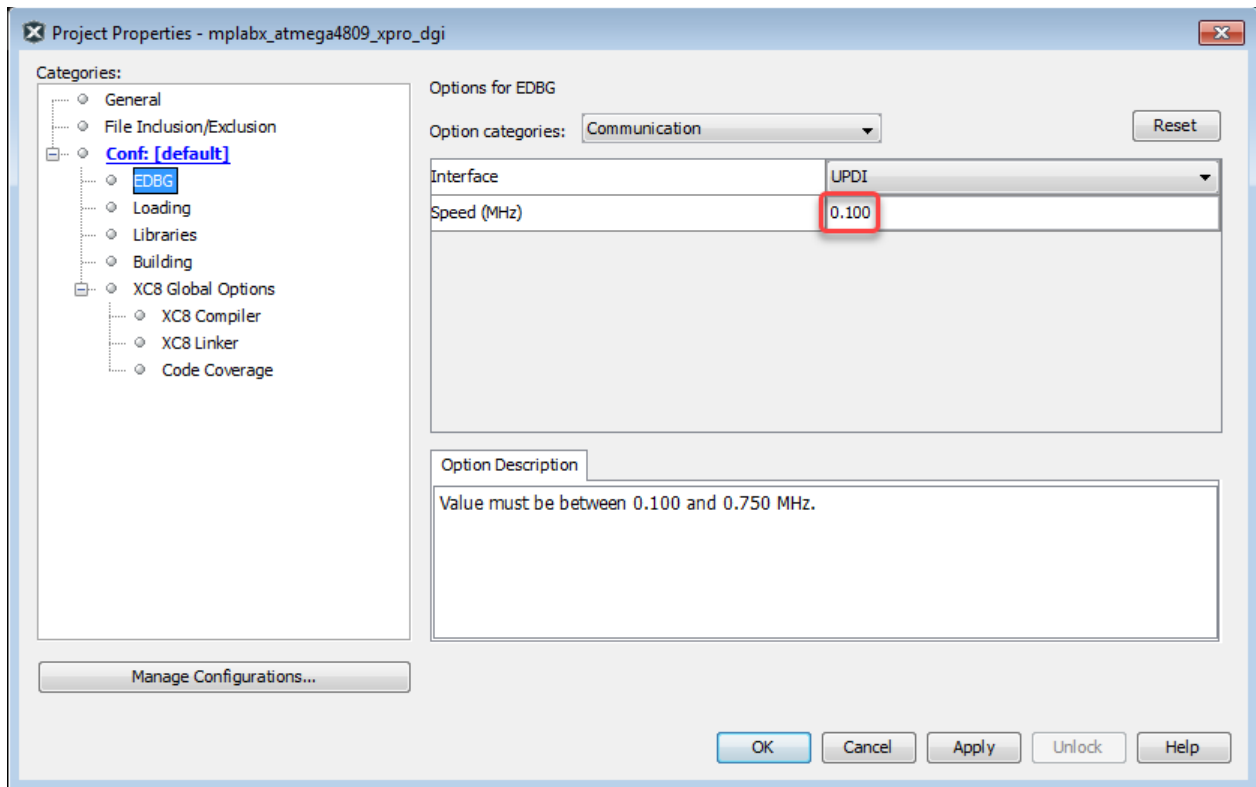
このプロジェクト向けのサンプルコードは、9. 「データプロット用サンプルコード」に記載しています。完成したプロジェクトは、図 4-22 のように表示されます。

図 4-22. プロジェクトのファイル構成




4.3.5.1.4 プロジェクトの通信オプションを設定する

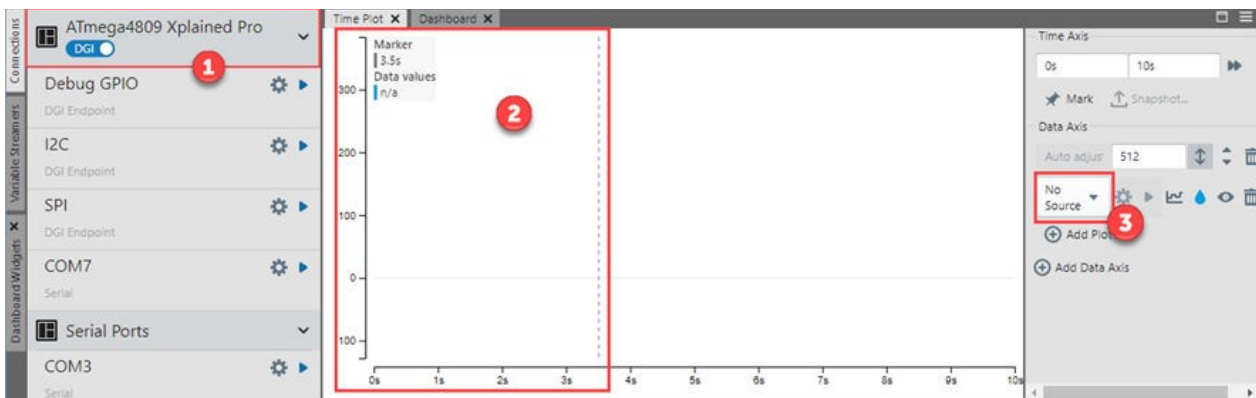
[Projects] ウィンドウ内でプロジェクト名を右クリックして [Project Properties] を開きます。「Options for EDBG」の下で、コード内の設定に適合する速度(この例では「0.100」)を設定します(次ページの図参照)。



4.3.5.1.5 MPLAB Data Visualizer を開く

ツールバー アイコン  をクリックするか、メニュー Windows>Debugger>Data Visualizer を選択する事により、MPLAB Data Visualizer プラグインを開きます。

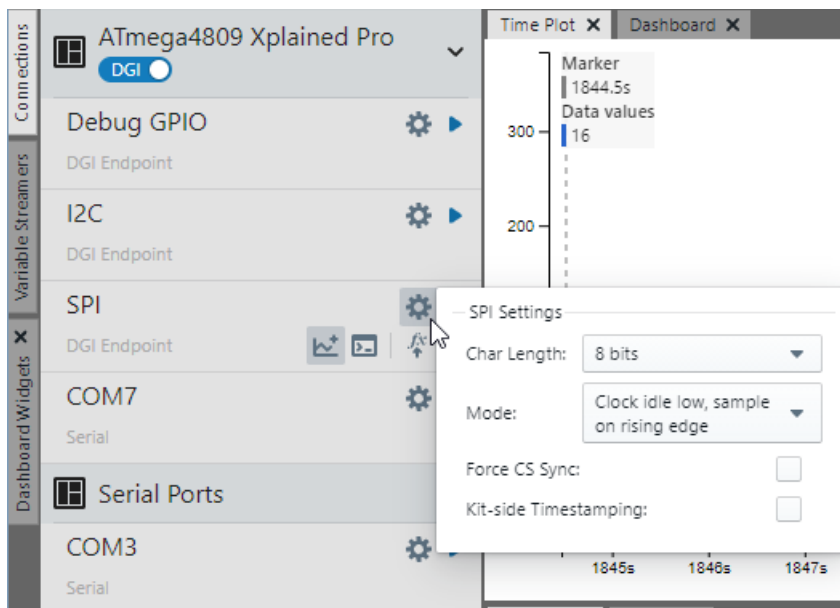
1. 左側の[Connections]タブ内に「ATmega4809 XPlained Pro」が表示され、「DGI」が有効である事がが必要です。
2. [Time Plot]ウィンドウでは、スライディング マーカー(灰色の縦破線)が 0 s 位置に表示され、データ値は何も表示されません(データ値が存在する場合、ソースのプロット色と同じ青色の縦バーの右横に値が表示されます)。
3. 右側ウィンドウ枠内のデータ軸向け[Source]ボックスには「No Source」と表示されます。



4.3.5.1.6 プロジェクトのデバッグと出力の表示

プロジェクトのデバッグを始めるには、[Debug Project] アイコン  をクリックします。

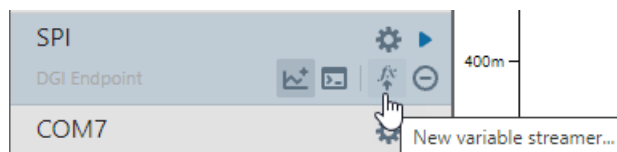
MPLAB Data Visualizer の[Connections]タブには、「ATmega4809 Xplained Pro」(DGI 有効)の下に利用可能な接続が表示されます。SPI 接続のギア アイコンをクリックして SPI 設定を表示します。これらの設定の詳細は、3.2.5. 「SPI インターフェイス」を参照してください。



通常は、▶ をクリックする事によりデータのストリーミングと生データのプロットを開始します。しかし、この例では変数データをプロットします。



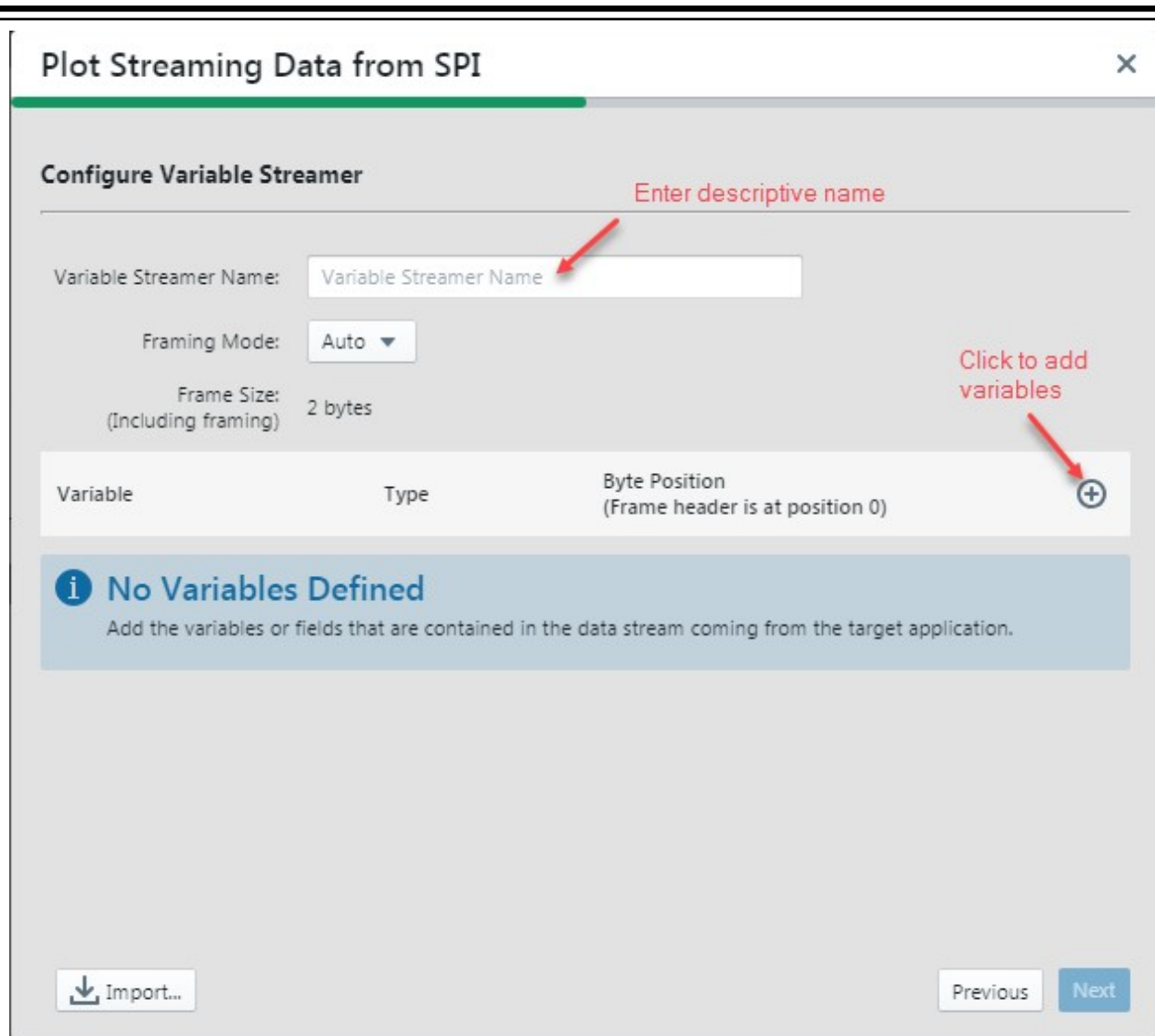
変数データをプロットするには、[New Variable Streamer] アイコンをクリックする事により Variable Streamer を設定します。



SPI を介するストリーミング データをプロットする Variable Streamer の設定 - 初期画面

この例では、変数の値をストリーミングしてプロットします。プロット ウィザードの初期ダイアログを次ページに示します。最初に Variable Streamer の名前を指定します。この名前は、この後の設定用に使います。次に ⊕ アイコンをクリックして変数を追加します(入力用のテキストボックスが開きます)。

Note: 以前の設定を保存してある場合、ダイアログの最下段にある[Import]ボタンをクリックする事により読み込む事ができます。<?>キーを使ってキーボード ショートカット ダイアログを開く事ができます。このダイアログは<ESC>キーで閉じます。



Variable Streamer の設定 - データの入力

次ページに、変数を追加した後のダイアログを示します。前のダイアログで指定した変数の情報が表示されます。

データストリームをデコードするために、変数(またはデータストリーム内のフィールド)が定義されている必要があります。データは、5.2. 「ストリーム フォーマット」に記載した書式でストリーミングされます。

[Next]をクリックして次に進みます。

Plot Streaming Data from SPI

Configure Variable Streamer

Variable Streamer Name:

Framing Mode:

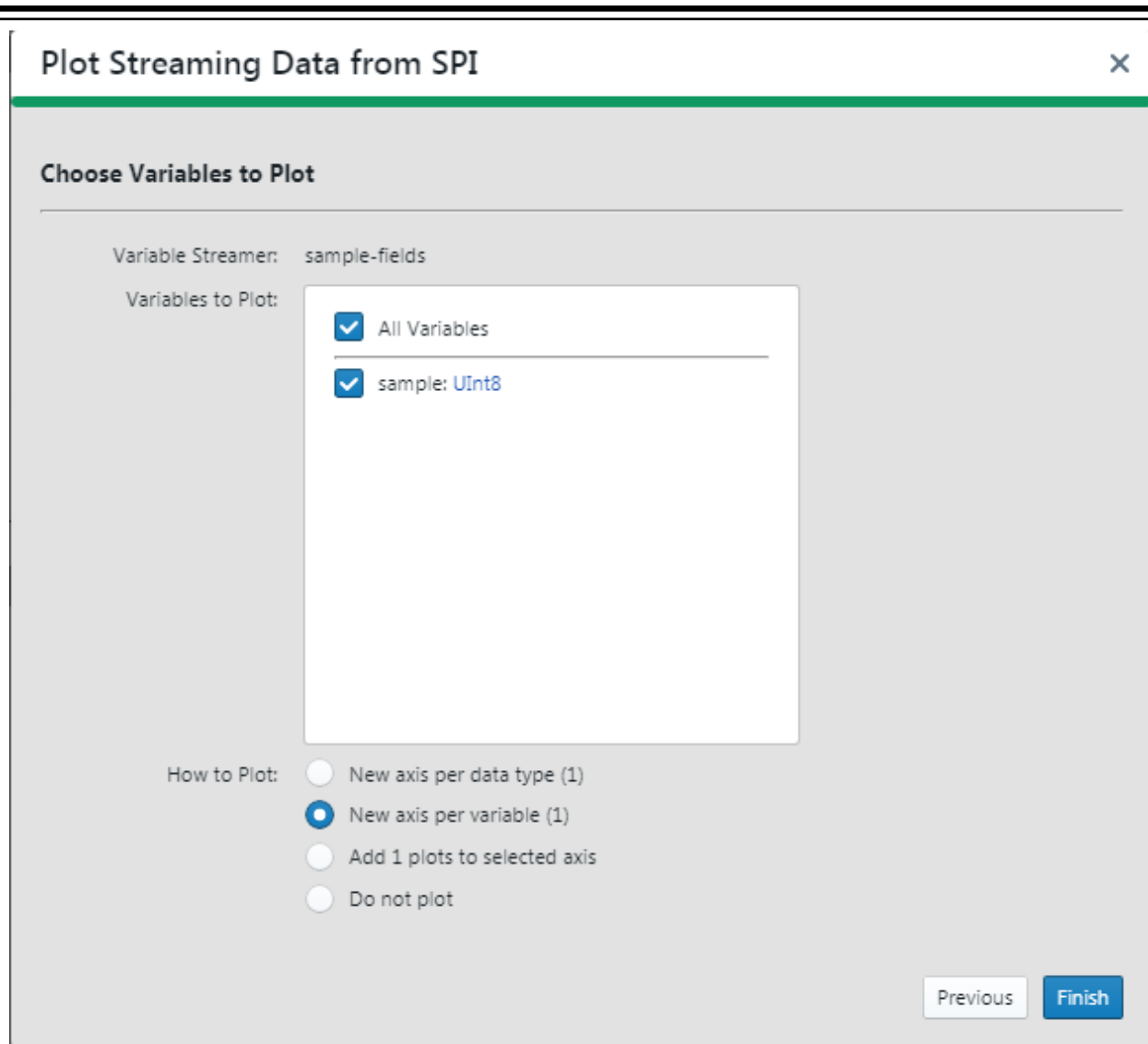
Frame Size:
(Including framing)

Variable	Type	Byte Position (Frame header is at position 0)	
<input type="text" value="sample"/>	<input type="button" value="UInt8"/>	1	<input type="button" value="✎"/> <input type="button" value="🗑"/>

Variable Streamer の設定 - プロット方法

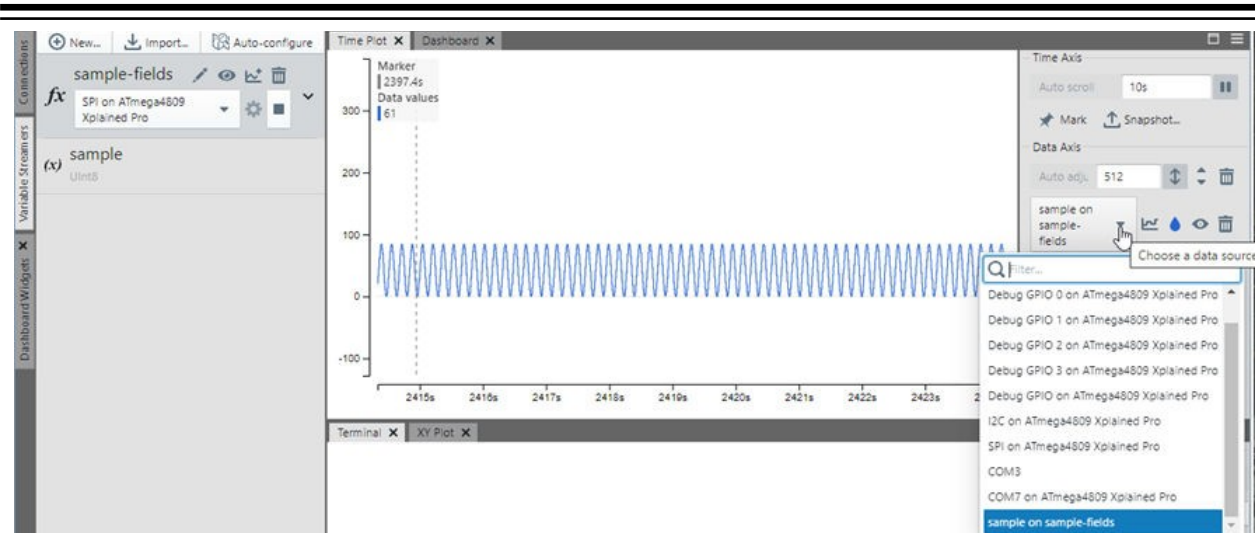
次ページに示すダイアログに、前のダイアログの要約と、データのプロット方法を選択するためのリストが表示されます。この例では、「New axis per variable (1)」を選択しています。

[Finish]をクリックしてプロット画面に進みます。

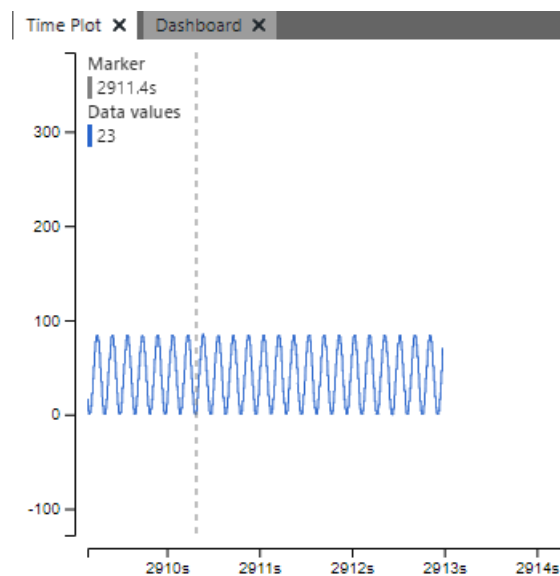


プロット設定に基づいてデータを表示する

次ページの図に示す通り、グラフの右側でデータ軸ソースが「sample on sample-fields」に設定されており、そのデータがグラフにプロットされています。データのスクロールは、グラフの最上部で[Pause Scrolling/Show Live Data]ボタンを押すか、スペースキーを使う事により有効/無効を切り換える事ができます。

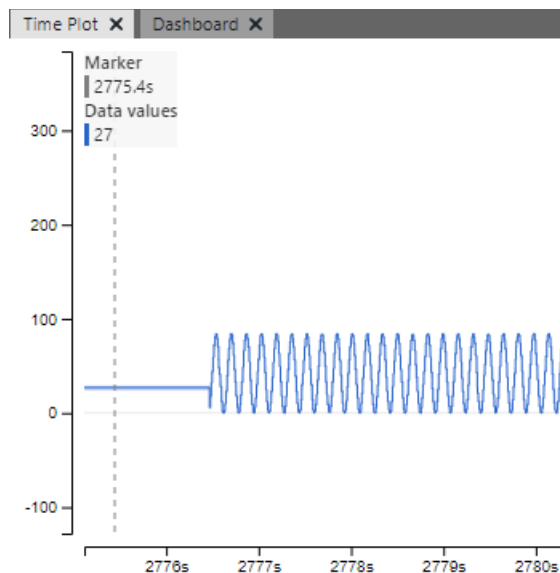


を使ってデバッグを一時停止すると、データ出力は停止します。





を使ってデバッグを再開すると、データは再びグラフに表示されます。

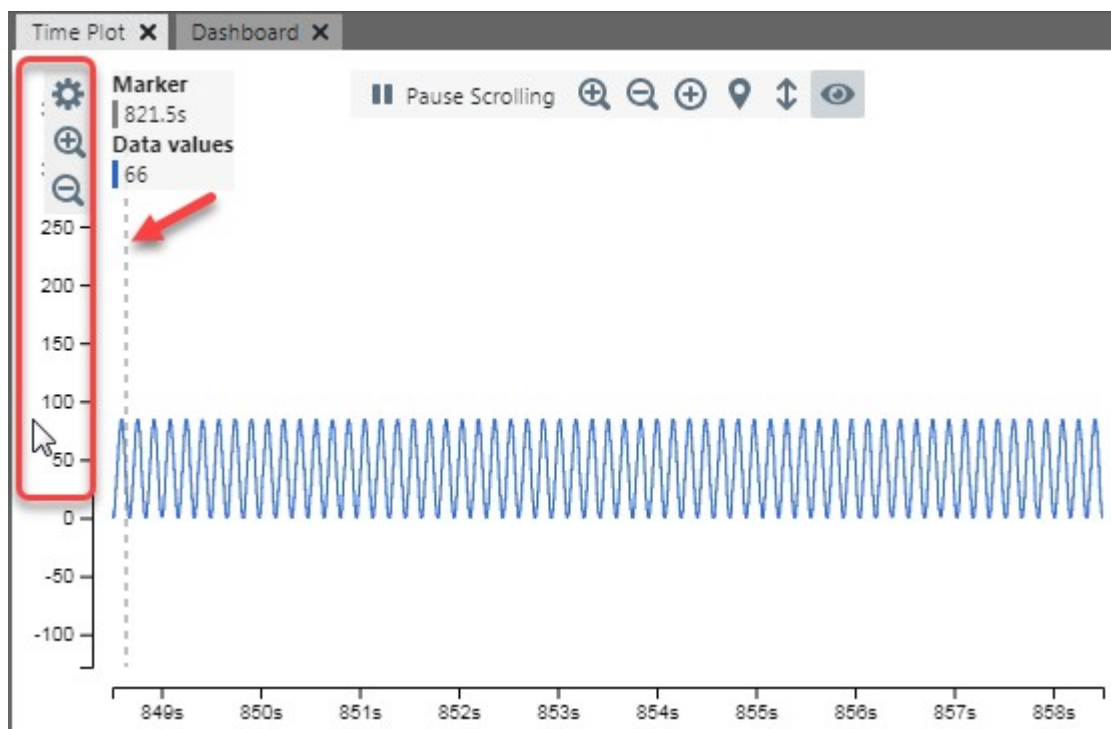


ツールを使ってプロットデータを解析する

グラフツールを使ってプロットデータの表示設定の変更と解析が行えます。

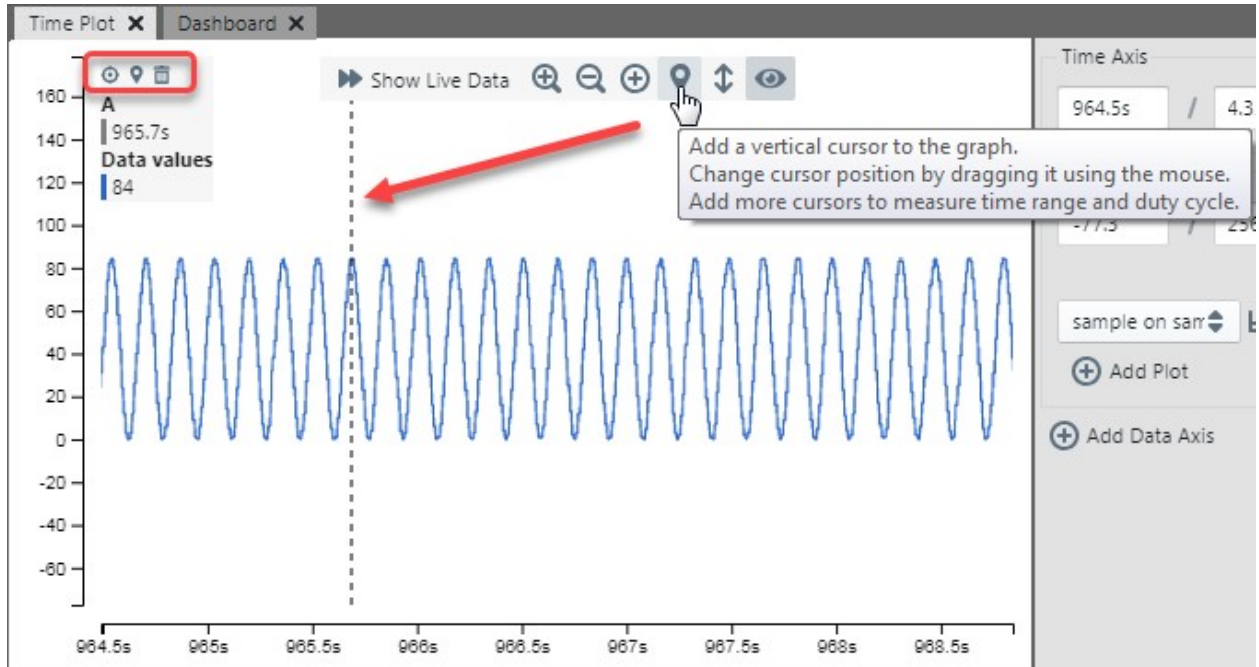
軸レンジとプロット位置を調整する事により、データを見やすくできます。下図では、説明のためにデータ軸(縦軸)を赤の四角で囲んでいます。軸上または軸の近くをクリックしてからマウスホイールを使って軸のレンジを拡大/縮小できます。軸上で軸方向にドラッグする事で、プロットを移動させる事ができます。軸の上端に表示されるアイコンをクリックすると、軸レンジの拡大/縮小ができる他に、右側の設定ウィンドウによってプロット特性も制御できます。

マウスの動きに追従する縦破線のマーカーがグラフに表示され、グラフの左上隅にマーカー位置の時間(横軸値)とデータ値(縦軸値)が表示されます。

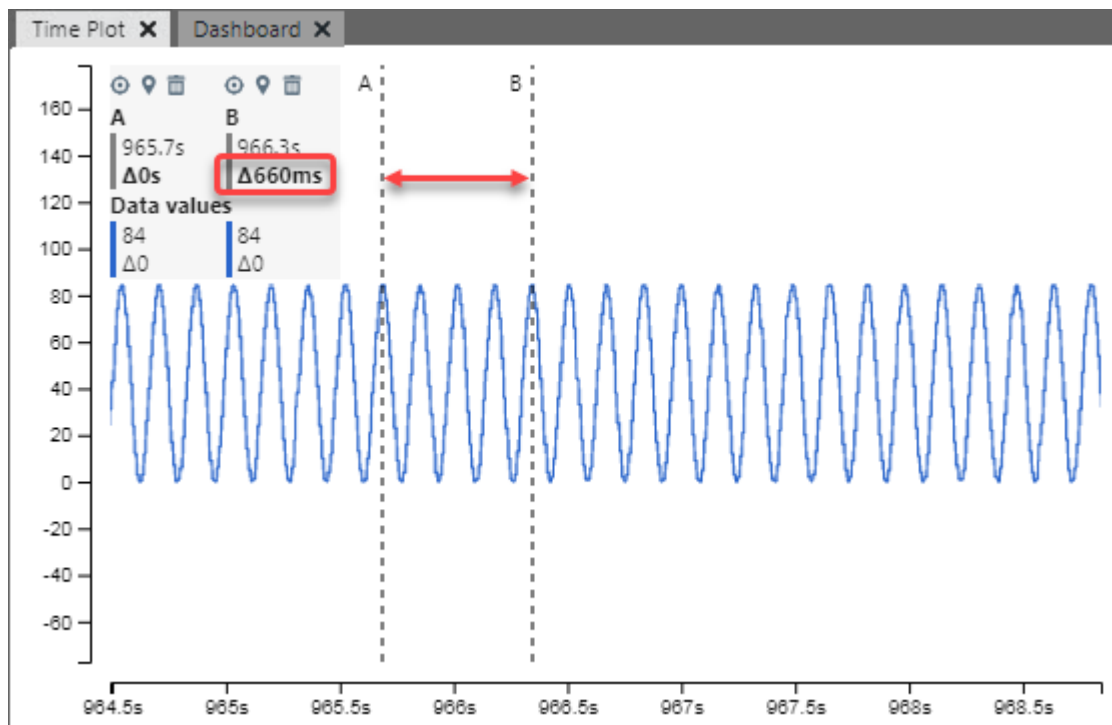


ドラッグ&ドロップで位置を変更可能な縦カーソルは、次ページの図に示すアイコンをクリックする事で有効にできます。縦カーソルを有効にすると、マーカーは無効になります。

グラフの左上隅に表示されていたマーカー情報の代わりに、マーカー用の制御アイコンとマーカー値(位置とデータ値)が表示されます。



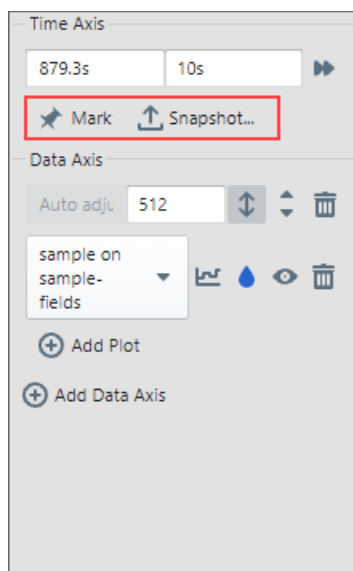
もう1本の縦カーソルを有効にする事で、2つのプロット点の間の時間差(Δ 時間)を調べる事ができます。



4.3.5.1.7 プロットデータを出力する

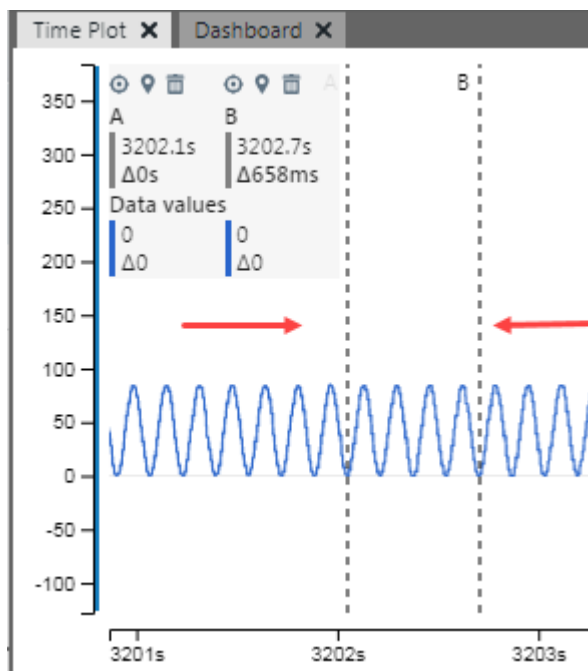
プロットデータのスナップショットをテキストファイルとして出力できます。プロットデータをキャプチャするには、右側ウィンドウ枠内で[Mark]ボタンと[Snapshot]ボタンを使います(図 4-23 参照)。ソースが何もプロットされていない場合、[Snapshot]ボタンは無効になります。

図 4-23. [Mark]ボタンと[Snapshot]ボタン



プロットデータのスナップショットをキャプチャするには、最初に[Mark]ボタンを押してスクロールを止める必要があります。スナップショットの範囲を指定するマーカー(A/B)が時間軸の左端と右端に表示されるので、必要に応じてこれらのマーカー位置を調整します (図 4-24 参照)。

図 4-24. スナップショット範囲の指定



次に[Snapshot]をクリックしてスナップショット ウィザードを開き、データと出力ファイルのフォーマットを選択します。

図 4-25. スナップショット - データ キャプチャ

Save Data Snapshot [X]

Data Capture

Snapshot: Time offset:

Duration

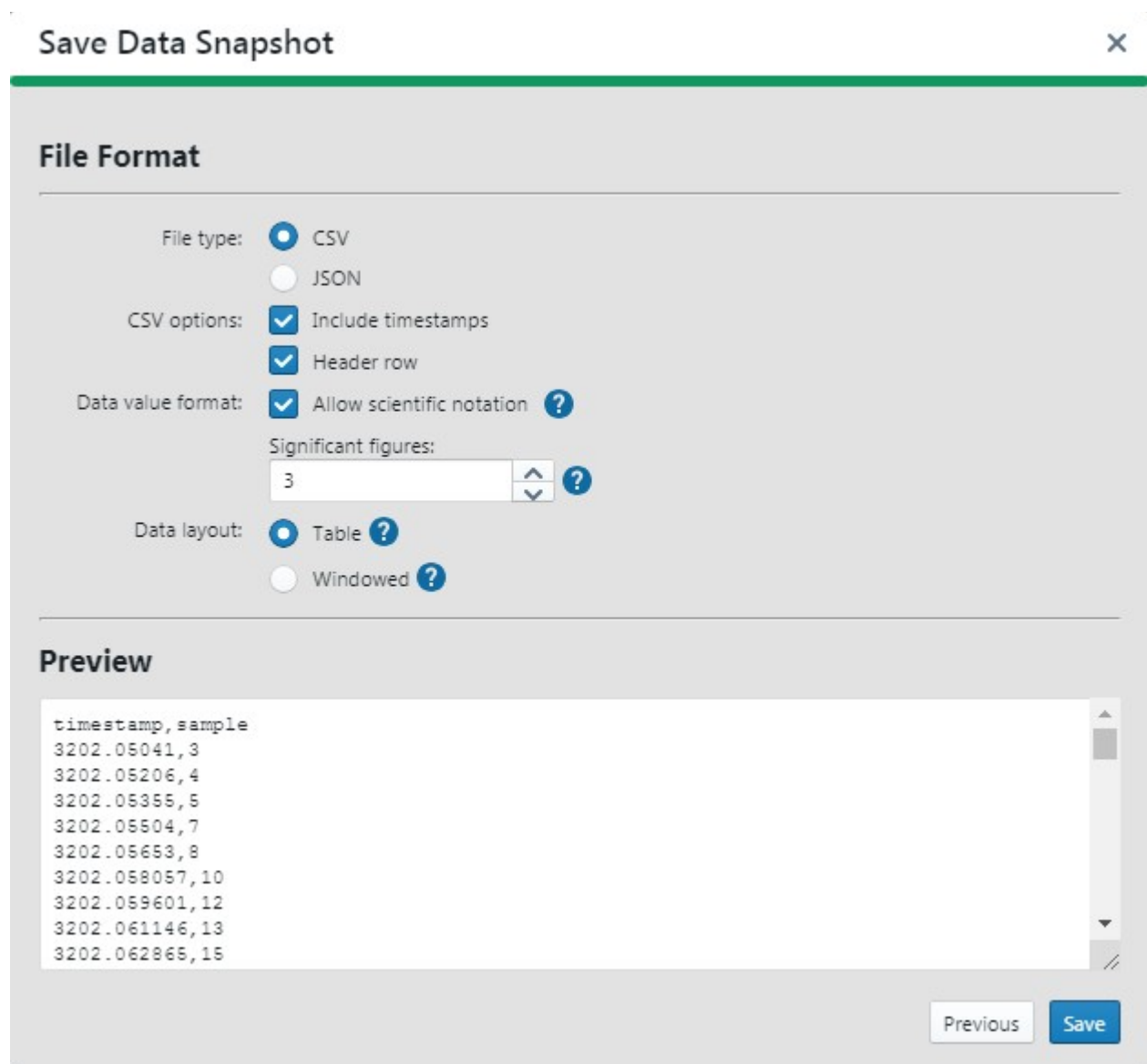
Sources to include:

- All sources
- sample on sample-field
- Debug GPIO 0 on ATmega4809 Xplained Pro
- Debug GPIO 1 on ATmega4809 Xplained Pro
- Debug GPIO 2 on ATmega4809 Xplained Pro
- Debug GPIO 3 on ATmega4809 Xplained Pro
- Debug GPIO on ATmega4809 Xplained Pro
- I2C on ATmega4809 Xplained Pro
- SPI on ATmega4809 Xplained Pro

Previous Next

スナップショットの時間オフセット(開始位置)と期間はカーソルの位置に応じて設定されます。スナップショットに含めるデータソースはリストから選択できます。

図 4-26. スナップショット - ファイルフォーマット



出力ファイルのタイプを選択します。その他のオプションは、選択したファイルタイプに応じて異なります。各オプションの右に表示される「？」アイコンにマウスカーソルを重ねると、そのオプションに関するヘルプが表示されます。

「Preview」の下に、オプションの選択に基づく出力ファイルのプレビューが表示されます。設定を終えたら[Save]をクリックします。既定値により、ファイルはホストコンピュータ上の「User」ディレクトリに保存されます。

4.3.5.2 複数データプロットの例

以下では AVR128DA48 Curiosity Nano ボードを使って GPIO ピンから複数のデータプロットを生成する方法を説明します。これらのプロットは、同じ 1 つの軸を使ってプロットする事も、個別の軸を使ってプロットする事もできます。

4.3.5.2.1 セットアップ

以下の手順に従って、サンプルソフトウェアおよびハードウェアをセットアップします。

MPLAB X IDE

以下のリンクから MPLAB X IDE 6.05 以上を無償でダウンロードしてインストールします。

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

以下のパックが選択されている必要があります。

AVR-Dx_DFP v2.2.157 以上

MPLAB XC8 C コンパイラ

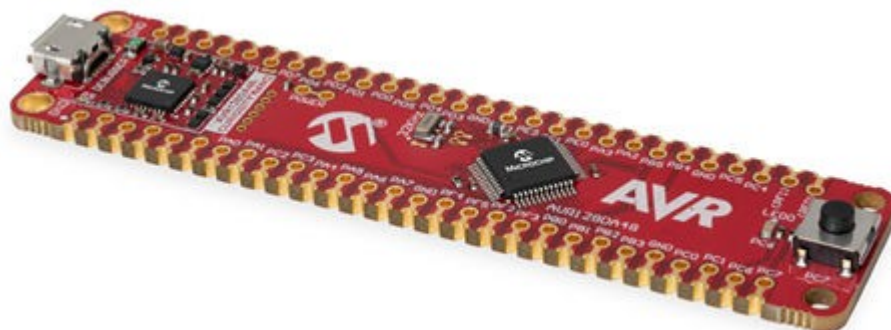
以下のリンクから MPLAB XC8 C コンパイラ v2.35 以上を無償でダウンロードしてインストールします。追加の最適化機能とその他の機能を備えた PRO バージョンのコンパイラは有償で入手可能です。

www.microchip.com/mplab/compilers

AVR128DA48 Curiosity Nano 評価用キット(DM164151)

この評価用ボード(下の写真)は、Microchip 社のオンライン販売サイトまたは販売代理店から購入できます。同梱の USB ケーブルを使って本ボードをホスト コンピュータに接続し、ドライバをインストールします。本ボードの詳細は以下を参照してください。

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



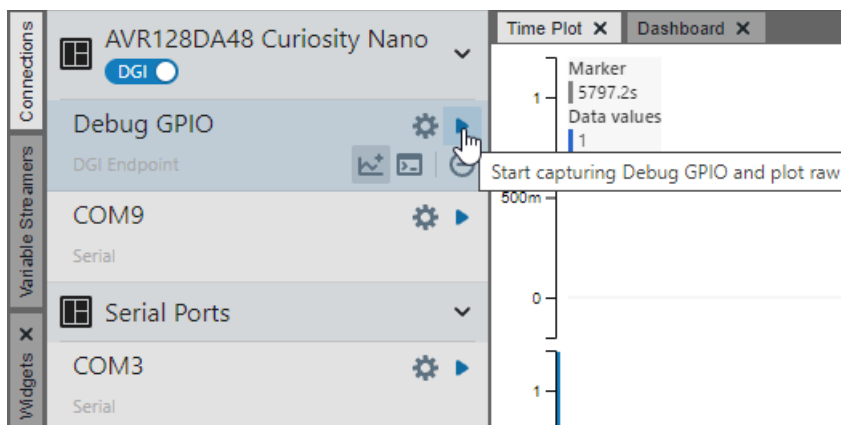
4.3.5.2.2 プラグ&プレイで Curiosity Nano を接続する

AVR128DA48 Curiosity Nano 評価用キットはプラグ&プレイ対応です。このため、USB ケーブルを使って本ボードをホスト コンピュータに接続すると MPLAB X IDE が起動します。IDE が開くと、Curiosity Nano に関する情報が [Kit Window] タブに表示されます。ボードの回路図へのリンクをクリックして開きます。回路図の最初のページには MPLAB Data Visualizer で使用可能な GPIO ピンの一覧が記載されています。PC7 はボード上の押しボタンに接続され、PC6 は LED0 に接続されています。

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

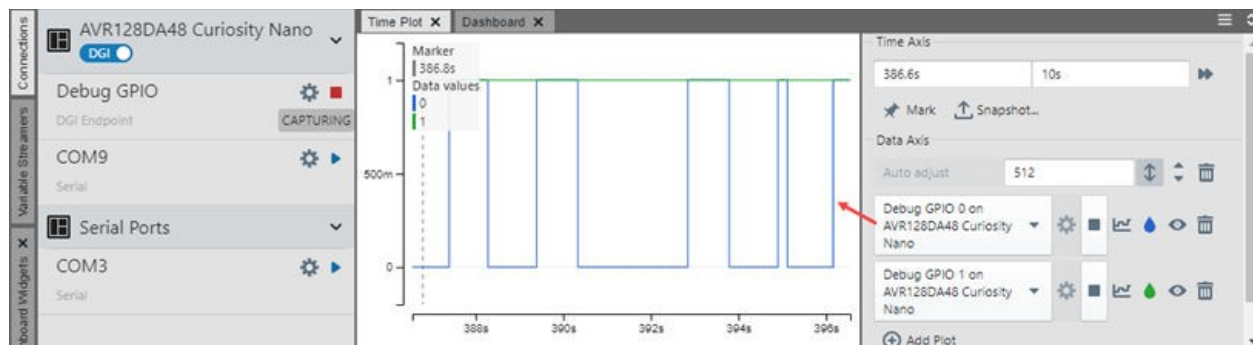


MPLAB Data Visualizer プラグインを開きます。Data Visualizer には利用可能なデータソース(GPIO ピン向けの Debug GPIO を含む) が表示されます。Debug GPIO の  をクリックすると、全てのピンからのデータがプロットされます(下図参照)。




両方の GPIO ピンからの出力が 1 つの縦軸を使ってプロットされます。グラフの右側には各プロットの情報(プロットの描画色を含む)が表示されます。

次に Curiosity Nano ボード側で操作を行います。ボード上のスイッチを押すと、GPIO 0 のプロットにパルスが表示されます。ボタンを押してからプロットにパルスが表示されるまでに遅れがあるので注意してください。回路図を見ると、ボタンにはプルアップがありません。しかし、プルアップはソフトウェアを使って有効にできます。GPIO 1 のプロットは一直線ですが、タイマからパルスが生成されると興味深いプロットを見る事ができます。このためにプロジェクトを作成し、一部のコードを追加します。

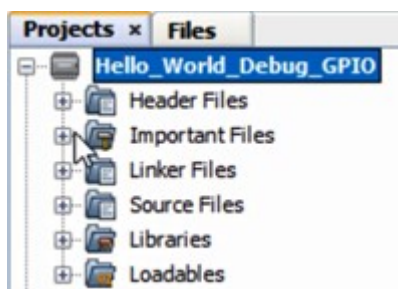


4.3.5.2.3 サンプル プロジェクトを作成する

メニュー **File > New Project** を選択するか、**[New Project]** アイコン  をクリックして、プロジェクト ウィザードを開きます。以下の手順によりプロジェクトを作成します。**[Next]** をクリックして次の手順に進みます。

1. **プロジェクトを選択する**: [Microchip Embedded] カテゴリをクリックしてから [Standalone Project] プロジェクトを選択します。
2. **デバイス(およびツール)を選択する**: デバイス名「AVR128DA48」を入力します。次いでツール名「AVR128DA48 Curiosity Nano-SN:MCHPL#」を入力します。ツールのシリアル番号(SN) は、接頭辞「MCHP」に続く複数桁の数字です。
3. **コンパイラを選択する**: メニュー **Compiler Toolchains > XC8** を選択し、この例で使うバージョンまたは最新バージョンを選択します。
4. **プロジェクト名とフォルダを選択する**: プロジェクトの名前を指定します(例: Hello World Debug GPIO)。Windows OS の場合、既定値のプロジェクト フォルダは C:\Users\\MPLABXProjects です。

[Finish] をクリックすると、[Project] ウィンドウにプロジェクト ツリーが表示されます。



4.3.5.2.4 アプリケーションを作成する

プロジェクトにソースコードを追加する事によりアプリケーションを作成します。

新しい main ソースファイルを作成する

以下の手順により、プロジェクトに新しいソースファイルを追加します。

1. プロジェクトの「Source Files」フォルダを右クリックし、メニュー **New > avr-main.c** を選択します。
2. 「Name and Location」の下でファイル名を main.c に変更し、**[Finish]** をクリックします。この新しいファイルが [Editor] ウィンドウに表示されます。既定値によるコードは以下の通りです。

```
/*
 * 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) {
    }
}
```

PC7 のプルアップを有効にするコードを追加する

ポート C (ピン 7) のプルアップを有効にするコードをソースコードに追加します。レジスタを正しく選択するために [View IO] ウィンドウが役立ちます。

図 4-27. [View IO] ウィンドウ

	PIN7CTRL	0x0457	0x00	0	7					3	2	1	0	
	INVEN		0	0	7									
	PULLUPEN		0	0						3				
	ISC		0	0								2	1	0

```
#include <avr/io.h>

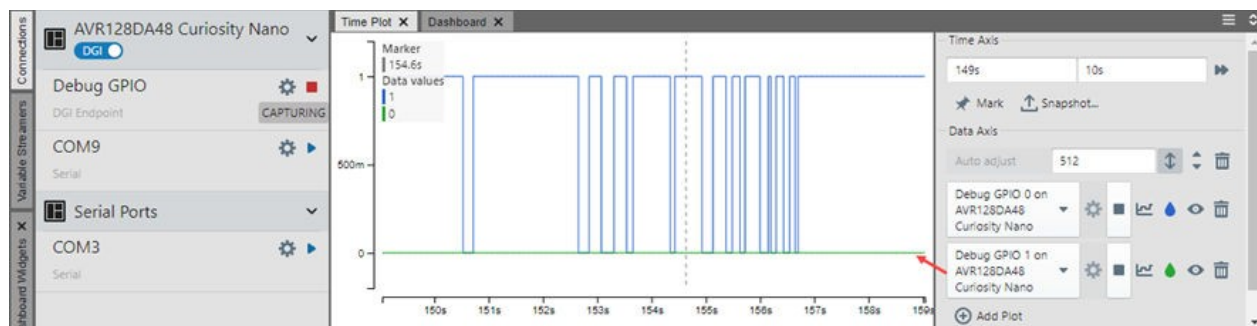
int main(void) {

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

    while (1) {
    }
}
```



をクリックしてデバイスをプログラムする事で、PC7 スイッチ (GPIO 0) の応答性が向上し、PC6 LED (GPIO 1) が点灯します。



遅延付きで LED を点滅させるコードを追加する

追加した main ソースコードにより、PC7 のプルアップを有効にして応答性が向上し、ユーザ LED が点灯しました。さらにコードを追加する事で、LED の点灯/消灯を可能にします。デバイスが高速であるため、遅延時間を追加して動作を遅くしない限り、LED の点滅を視認する事はできません。この遅延に関する情報は以下に記載しています。

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

CPU 周波数に関する #define と遅延をサポートするための #include を追加する必要があります。

Curiosity Nano 向けの `F_CPU` 値を特定するため、[View IO] ウィンドウを再度開いてコードのデバッグランを実行し、ウィンドウ内でライブ値を取得します。コードが `while(1)` ループに入ったら実行を一時停止し、(CLKCTRL) の下でシステムクロック周波数を確認します(次ページの図参照)。

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

これに基づいて更新したコードを以下に示します。

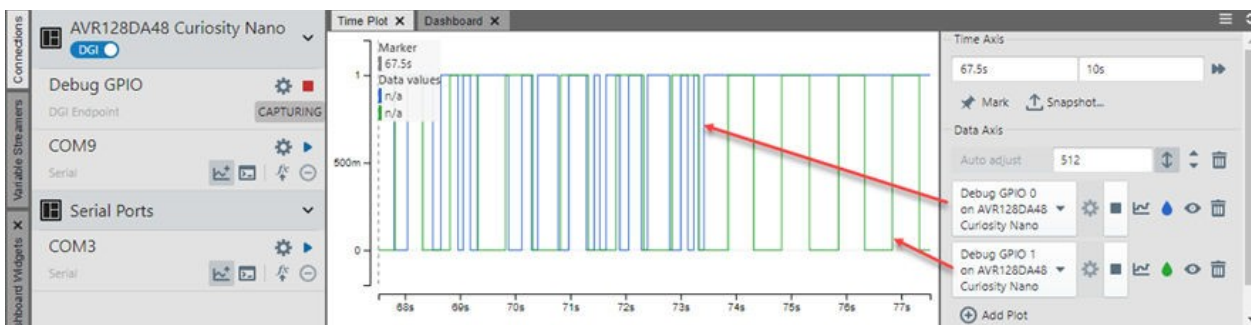
```
#include <avr/io.h>
#define F_CPU 4000000UL
#include <util/delay.h>

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 */
    }
}
```

GPIO ピンからのデータを再度プロットすると、PC7 (GPIO 0)はユーザのボタンの操作に応じて異なる幅のパルスを示し(表示色は青)、PC6 (GPIO 1) はLEDのトグルに応じて一定幅のパルスを示します(表示色は緑)。



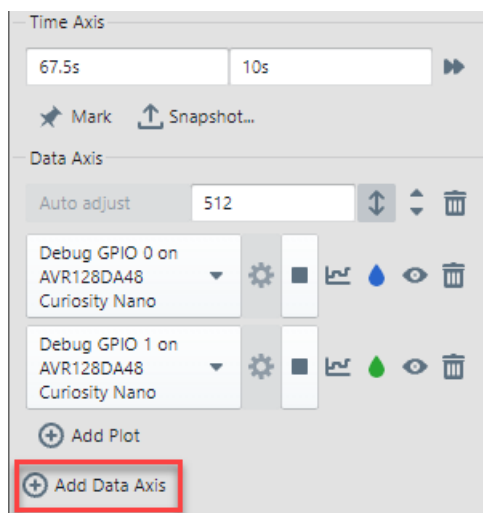
4.3.5.25 プロット設定

このプロジェクトコードは2つのGPIOピンで出力を生成します。これらの出力を1つのデータ軸に表示すると、たとえプロットを色分けしても見やすくありません。各プロットに別々のデータ軸を割り当てる事でグラフは見やすくなり、GPIOオプションが各プロットに及ぼす影響をより容易に観察する事ができます。

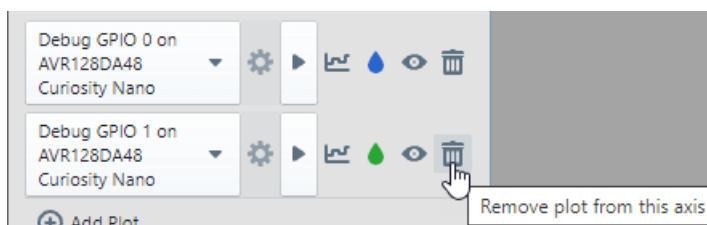
各プロットに異なるデータ軸を割り当てる

この例ではプロットが2つしかありませんが、プロットの数が増えても手順は同じです。

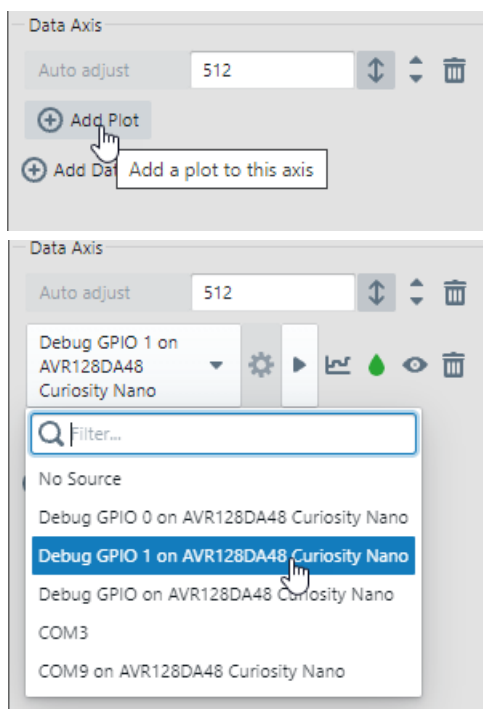
1. グラフに1本のデータ軸を追加します。何もプロットされない軸が、既存の軸の下に表示されます。



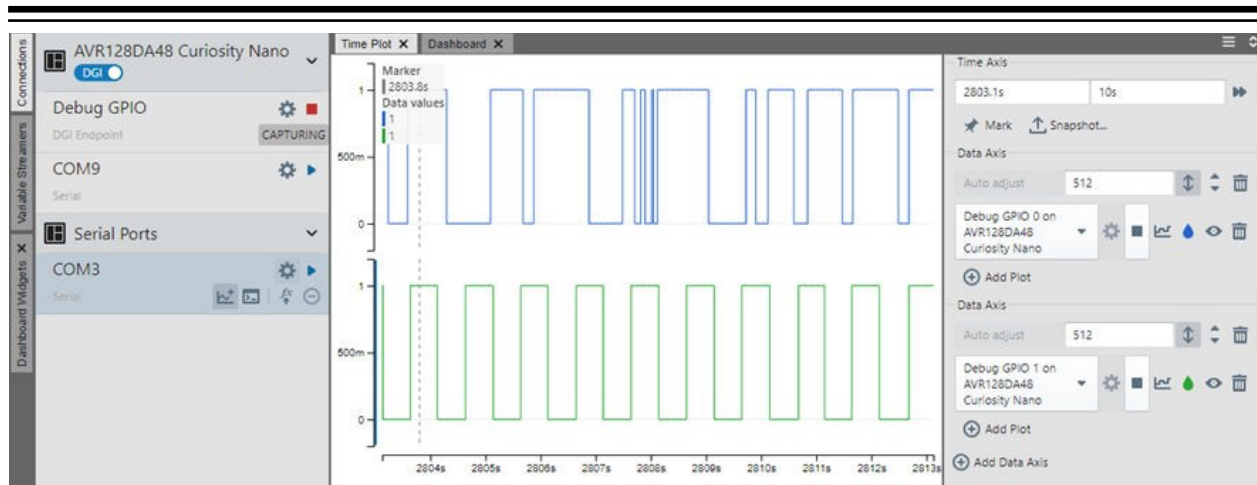
2. 既存の軸から1つのプロットを削除します。



3. 削除したプロットを新しい軸に追加します。[Add Plot] をクリックし、ドロップダウンメニューから先ほど削除したプロットを選択します。



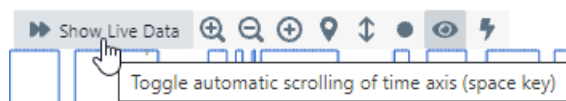
4. これにより、各プロットは別々の軸に表示されて見やすくなります。



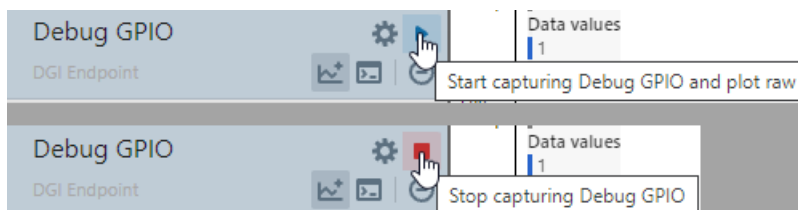
プロットの開始と停止

プロットのストリーミングは各種の方法で開始/停止できます。

1つ目の方法は、ツールバー上の[Show Live Data/Pause Data] ボタンをクリックするかスペースキーを押す事で、プロットのスクロールを開始または停止します。この方法では、データのキャプチャは開始/停止しません。



2つ目の方法は、データのキャプチャを開始または停止します。この場合、左側の[Connection]タブ内で、データのストリーミング用に使われている接続のキャプチャ開始/停止アイコンを使います。



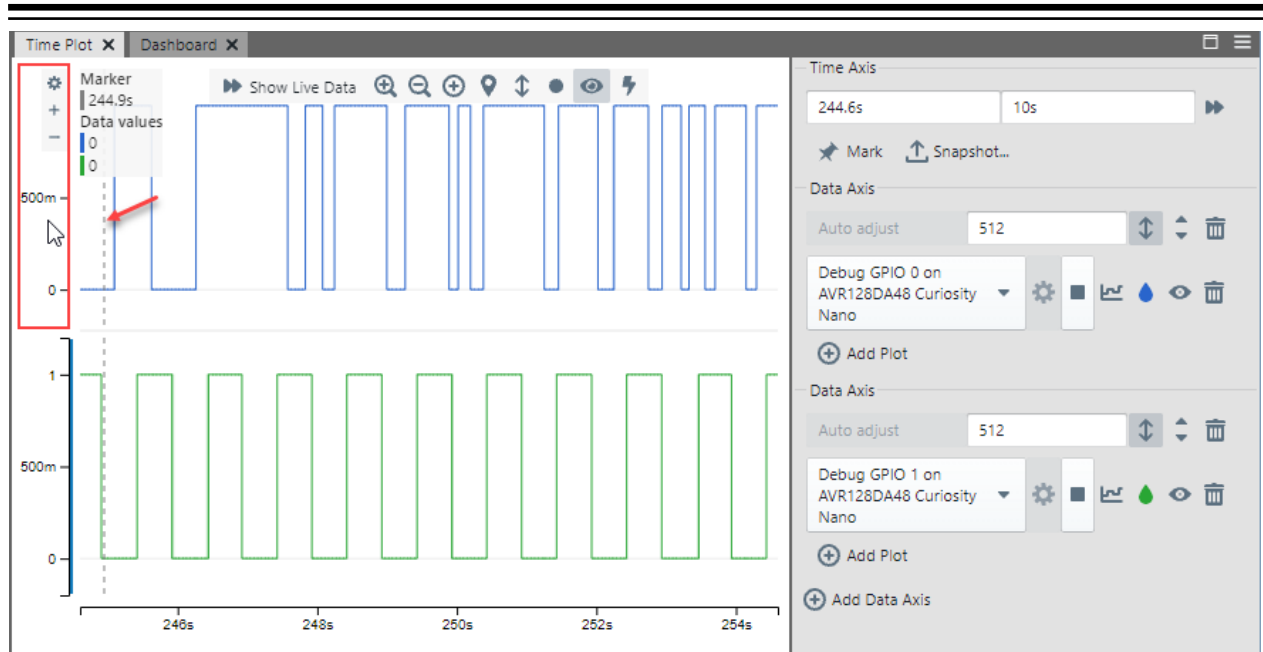
3つ目の方法は、 (Pause)および (Continue)ボタンを使って、MPLAB X IDE 内のアプリケーションの実行を停止および再開します。この方法は、MPLAB Data Visualizer プラグインに対してのみ使えます。

ツールを使ってプロットデータを解析する

グラフツールを使ってプロットデータの表示設定の変更と解析が行えます。

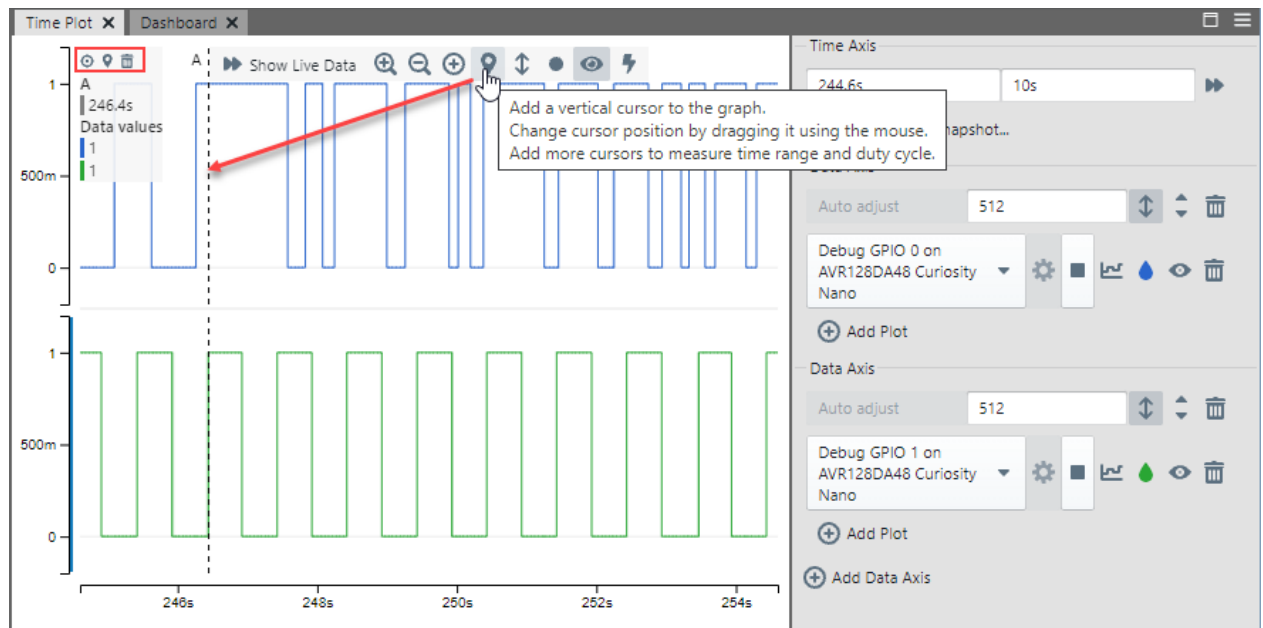
軸レンジとプロット位置を調整する事により、データを見やすくできます。次ページの図では、説明のためにデータ軸(縦軸)を赤の四角で囲んでいます。軸上または軸の近くをクリックしてからマウスホイールを使って軸レンジを拡大/縮小します。軸上で軸方向にドラッグする事で、プロットを移動させる事ができます。軸の上端に表示されるアイコンを使うと、軸レンジの拡大/縮小ができる他に、右側ウィンドウ枠で設定するのと同じプロット特性も設定できます。

マウスの動きに追従する縦破線のマーカーがグラフに表示され、グラフの左上隅にマーカー位置の時間(横軸値)とデータ値(縦軸値)が表示されます。

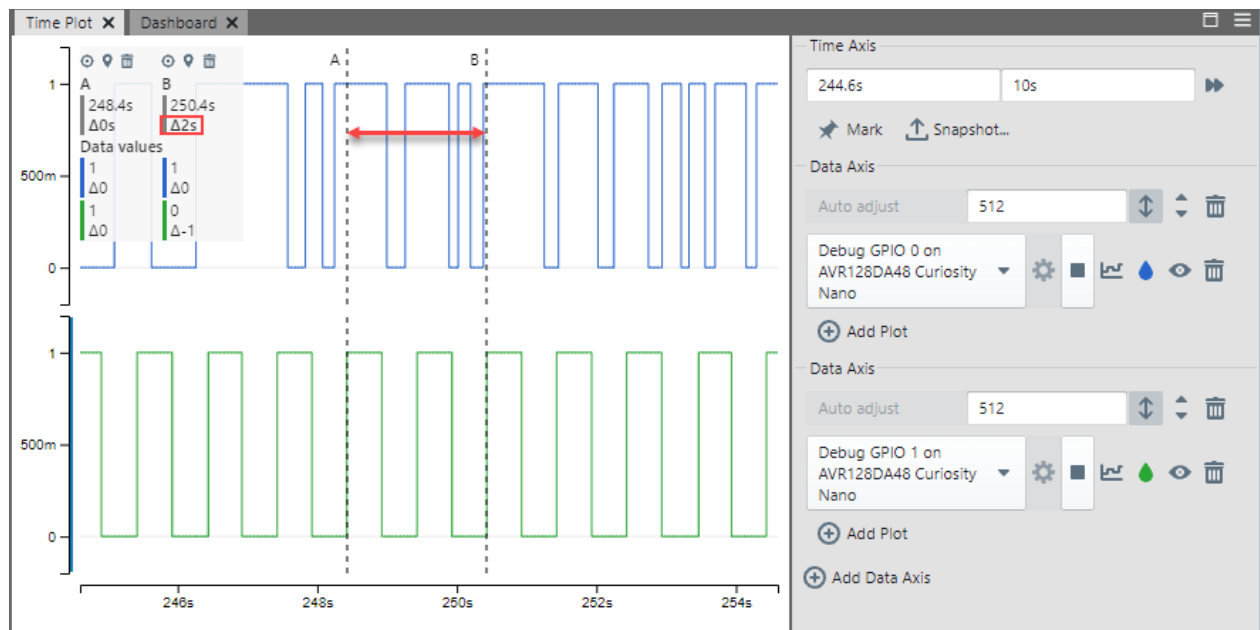


ドラッグ&ドロップで位置を変更可能な縦カーソルは、下図に示すアイコンをクリックする事で有効にできます。縦カーソルを有効にすると、マーカーは無効になります。

グラフの左上隅に表示されていたマーカー情報の代わりに、カーソル用の制御アイコンとカーソル値が表示されます。



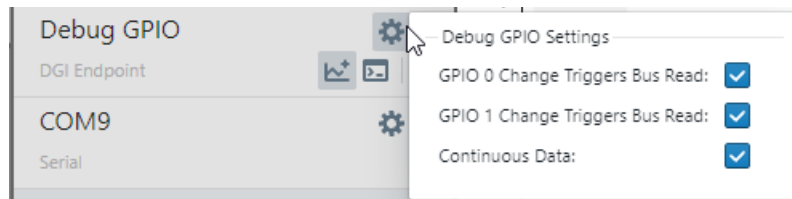
もう1本の縦カーソルを有効にする事で、2つのプロット点の間の時間差(Δ 時間)を調べる事ができます。



Debug GPIO オプション

Debug GPIO データソースが選択されている場合、ギアアイコンをクリックするとオプションが表示されます。既定値により全てのオプションが有効になります。

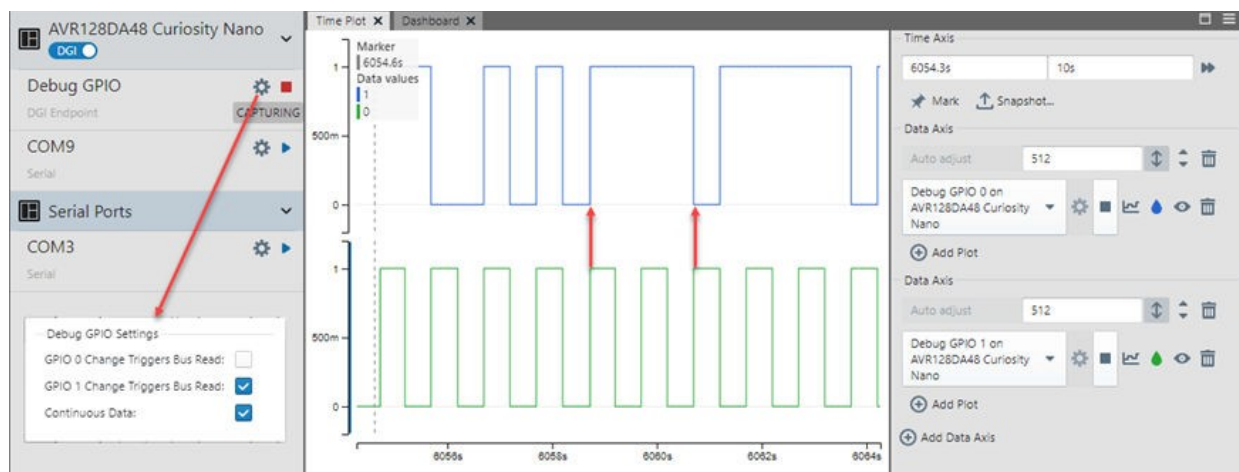
図 4-28. Debug GPIO オプション



「GPIO x Change Triggers Bus Read」を有効にすると、GPIO x の状態が変化するたびに GPIO バスが読み出されてデータがプロット上に表示されます。

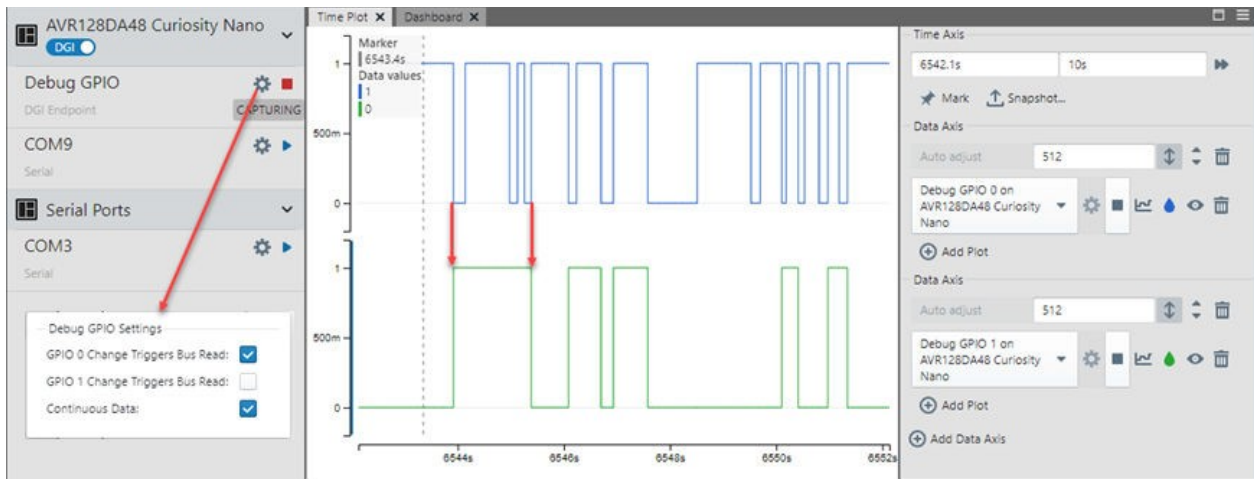
「GPIO 0 Change Triggers Bus Read」を無効にし、「GPIO 1 Change Triggers Bus Read」を有効にした場合、GPIO 1 が変化した時のみバスが読み出されます。従ってボタンの早押しにより GPIO 1 トグル間の変化が元の状態に戻った場合、変化は表示されません。

図 4-29. GPIO 0 オプションを無効にし GPIO 1 オプションを有効にした場合のプロット



「GPIO 0 Change Triggers Bus Read」を有効にし、「GPIO 1 Change Triggers Bus Read」を無効にした場合、ボタンが押されて GPIO 0 が変化した時のみバスが読み出されます。従って、たとえ GPIO 1 が一定周期でトグルしても、GPIO 0 が変化した時の GPIO 1 の値が表示されます。

図 4-30. GPIO 0 オプションを有効にし GPIO 1 オプションを無効にした場合のプロット



4.3.5.3 機械学習の例

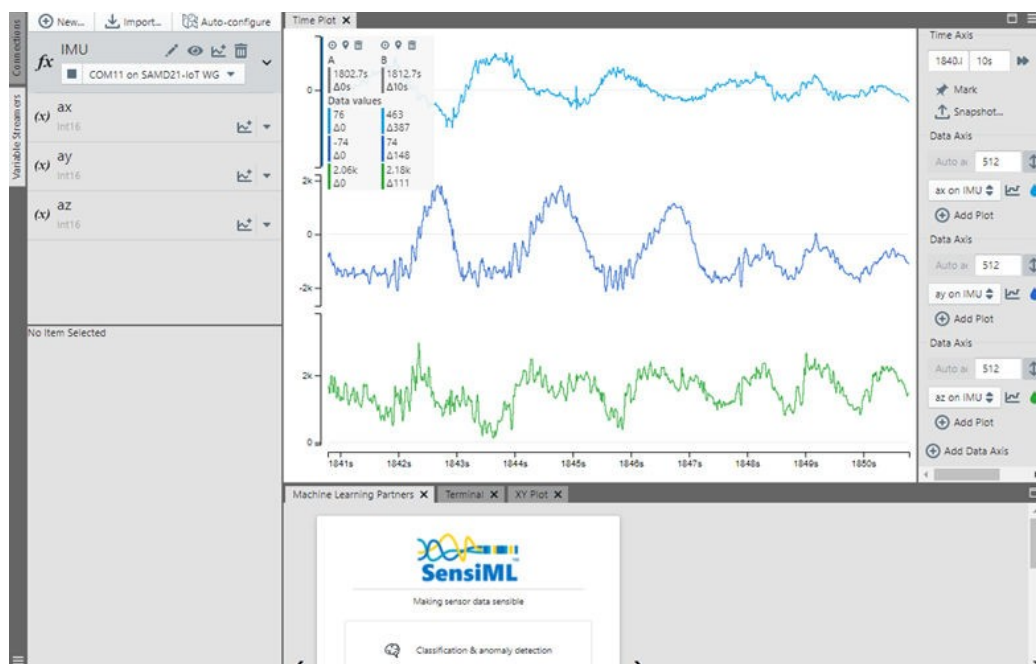
機械学習(ML)プラグインは MPLAB Data Visualizer 内のみでの利用なので、ターゲット デバイスからライブデータストリームをキャプチャし、必要なデータ領域を選択して、データを外部の ML プラットフォームにアップロードする事も可能です。これによりパートナーの ML プラットフォームを使用して、組み込みアプリケーション向けに展開可能な機械学習ソリューションを開発できます。

MPLAB Data Visualizer 向けにサポートされるプラグインを見つけるには、MPLAB X IDE 内でメニュー Tools>Plugins を選択します。プラグインを選択し、[Install] をクリックして画面の指示に従います。プラグインをインストールした後は、再起動が必要です。[Machine Learning Partners] ウィンドウが表示され、機械学習パートナーウェブサイトへのアクセスが提供されます。表示されない場合、[Views] の下で開きます。

詳細は [ML プラグイン ユーザガイド](#) を参照してください。

Note: サンプルコードは更新される場合があります。そのコードの GitHub ページで提供される全ての指示を読んでください。

図 4-31. 機械学習の例



4.4 Power データの表示

時系列プロットに Power データを表示してターゲット アプリケーションの消費電力を調べる事ができます。

4.4.1 基本的な電流計測

消費電力を計測するには、これをサポートするツールが必要です。以下では、デモ用に Power Debugger を使います。このツールの詳細は以下を参照してください。

www.microchip.com/DevelopmentTools/ProductDetails/ATPOWERDEBUGGER

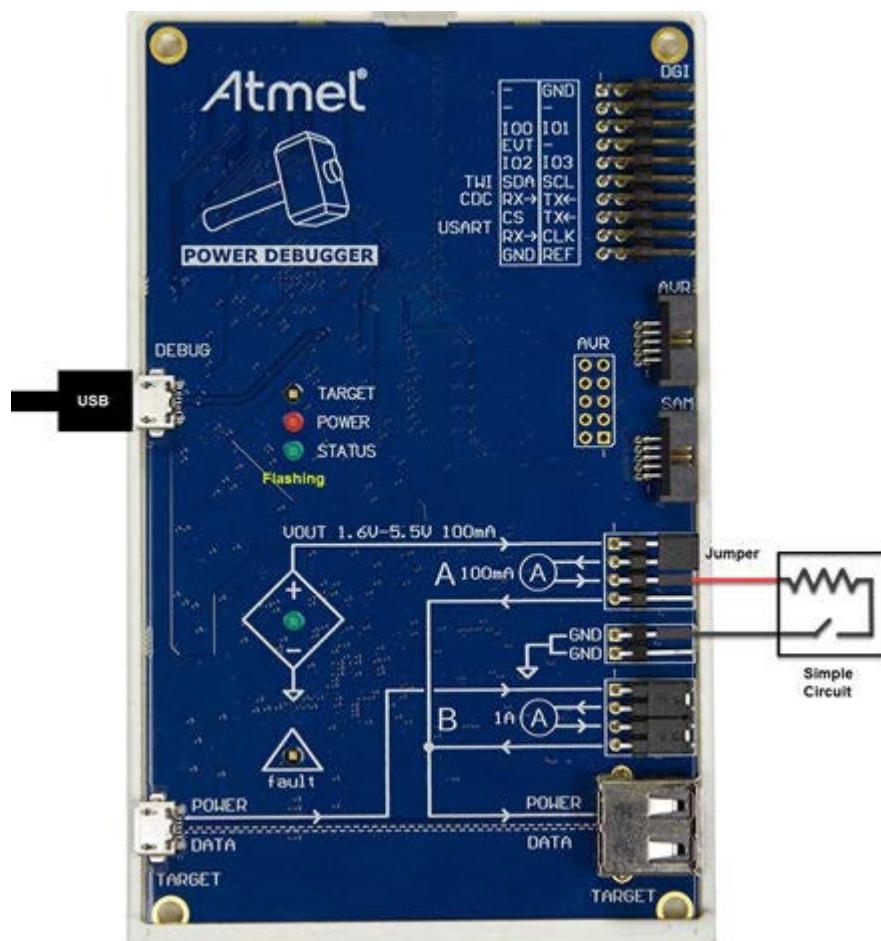
4.4.1.1 ハードウェアのセットアップ

USB ケーブルを使って Power Debugger をホスト コンピュータに接続します。POWER LED (赤)が点灯します。

次にターゲットに接続します。下図に示すシンプルな回路の消費電流をチャンネル A で計測します。この回路はデバッガの VOUT から給電されます(VOUT はジャンププラグで A+ 端子に接続しています)。A- 端子をこの回路に接続します。回路の他端はボードのグラウンドに接続します。

MPLAB Data Visualizer がデバッガを検出して通信を確立すると、STATUS LED (緑)が点滅します。VOUT の電圧レベルは Data Visualizer 内で設定できます(VOUT LED (緑)が点灯)。

図 4-32. シンプルな回路の接続



4.4.1.2 MPLAB Data Visualizer のセットアップ

MPLAB Data Visualizer を起動します。Power Debugger が検出されると、このツールの DGI 接続の下に「Power」ソースが表示されます(図 4-33 参照)。


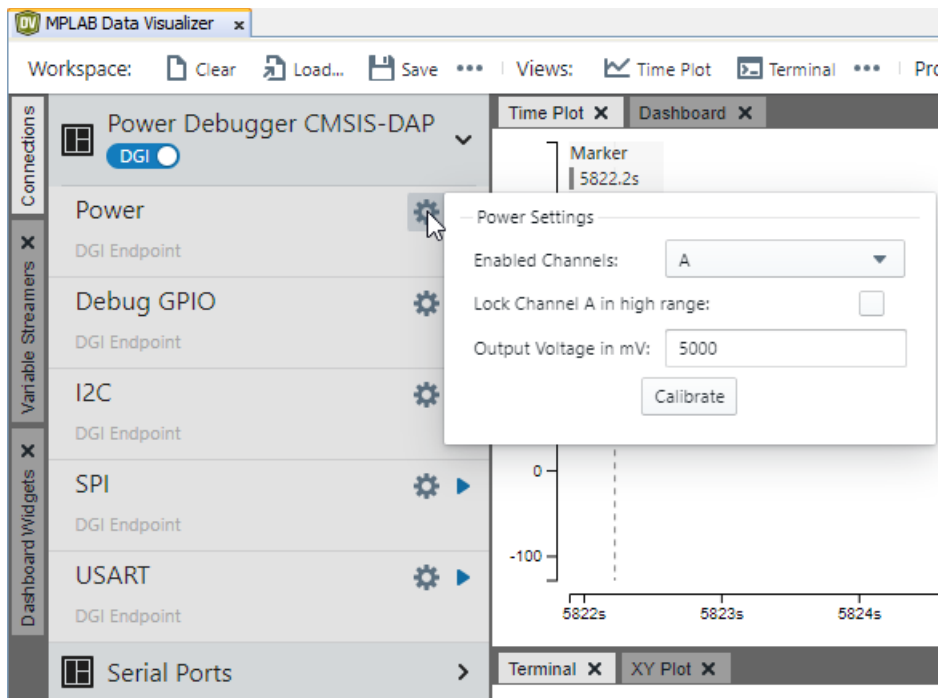
「Power」ソースのオプション アイコン  をクリックし、開いたダイアログに設定を入力します。設定を入力した後に再度このアイコンをクリックすると、その設定が保存されてダイアログは閉じます。

図 4-33 では、電流検出用にチャンネル A が選択され、VOUT は 5000 mV (5 V) に設定されています。

図 4-33. 「Power」ソースのセットアップ




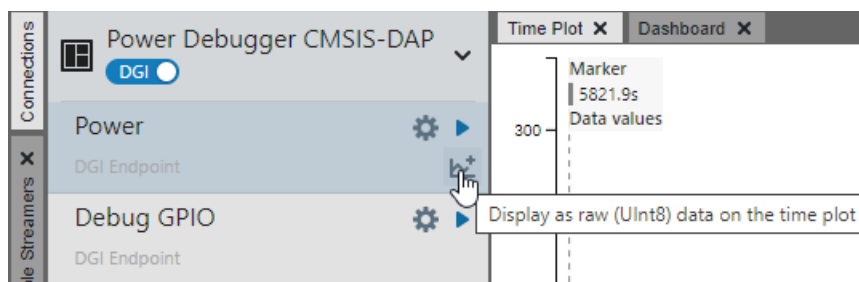
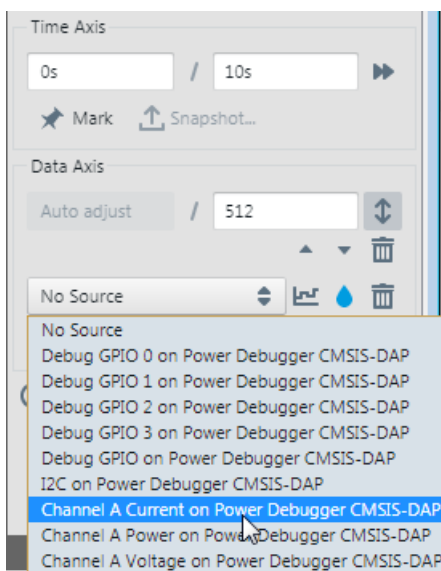
「Power」データプロットを表示するには、「Power」インターフェイスの上にマウスカーソルを置き、アイコン  が表示されたらこれをクリックします。これにより全てのソース(電力、電流、電圧)が時系列プロットに表示されます。

図 4-34. 「Power」ソースをプロットする



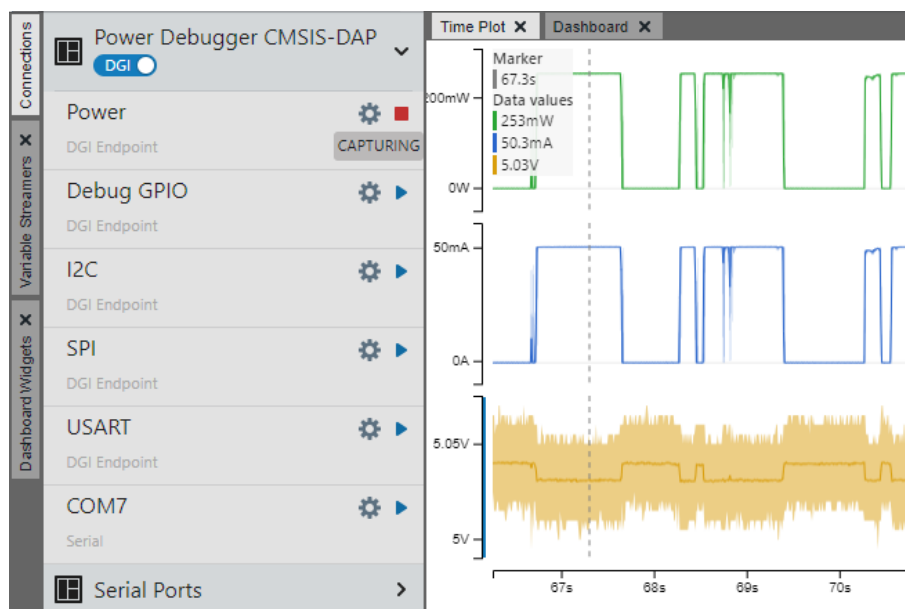
1つのプロットを選択するには、右側ウィンドウ枠でプロット オプションを選択します(図 4-35 参照)。

図 4-35. 1つのソースのみプロットする



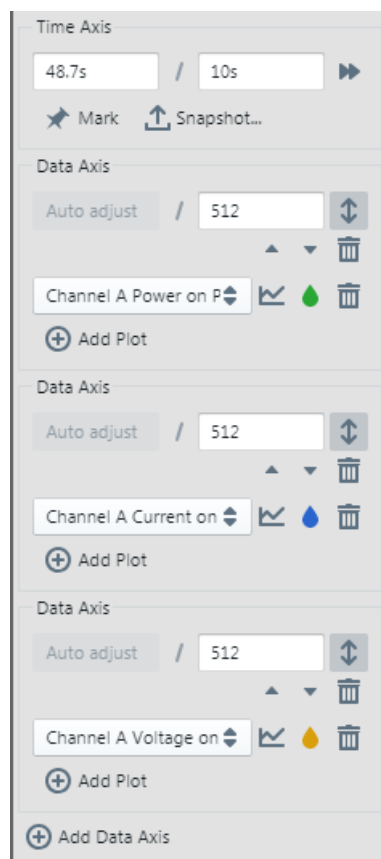
全てのソースをプロットする場合、既定値により上から順番に電力、電流、電圧プロットが表示されます。3番目の電圧プロットのノイズについては、4.4.1.4. 「Power 計測におけるノイズ」を参照してください。

図 4-36. 「Power」ソースのプロット



プロットの書式を変更するには、右側ウィンドウ枠内で制御パラメータを設定します(図 4-37 参照)。

図 4-37. プロットの制御パラメータ



4.4.1.3 カーソルを使った消費電力解析


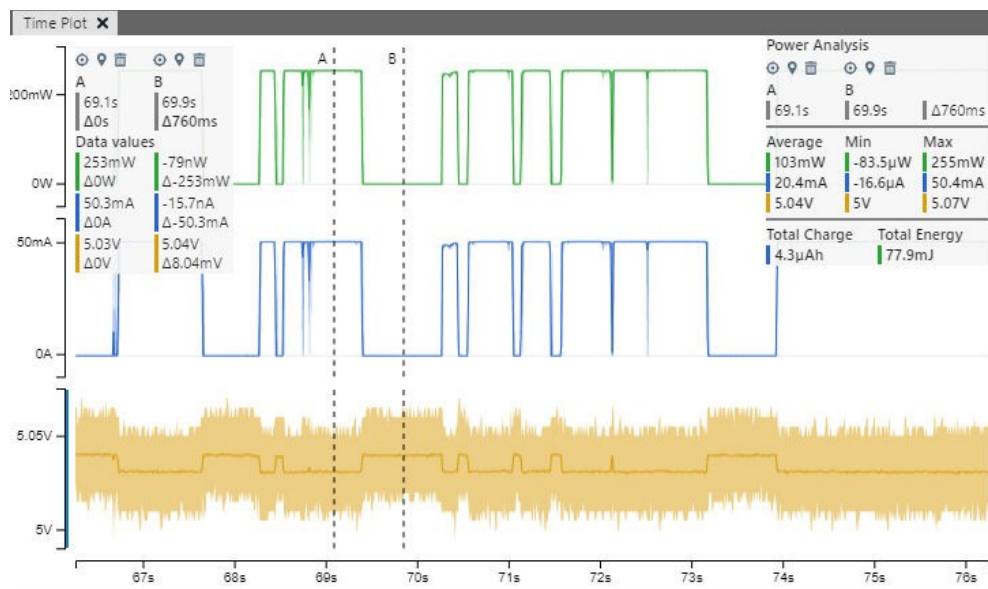
電流およびその他の Power 計測値をより詳細に解析するために、ウィンドウ ツールバーからアイコン  (Power Analysis) をクリックします。2 本のカーソル(A と B)が表示され、それらのカーソル値がグラフの左上隅に表示されます。さらに、グラフの右上隅に Power Analysis データが表示されます。カーソルはドラッグにより移動できます。

図 4-38. Power Analysis を使った表示



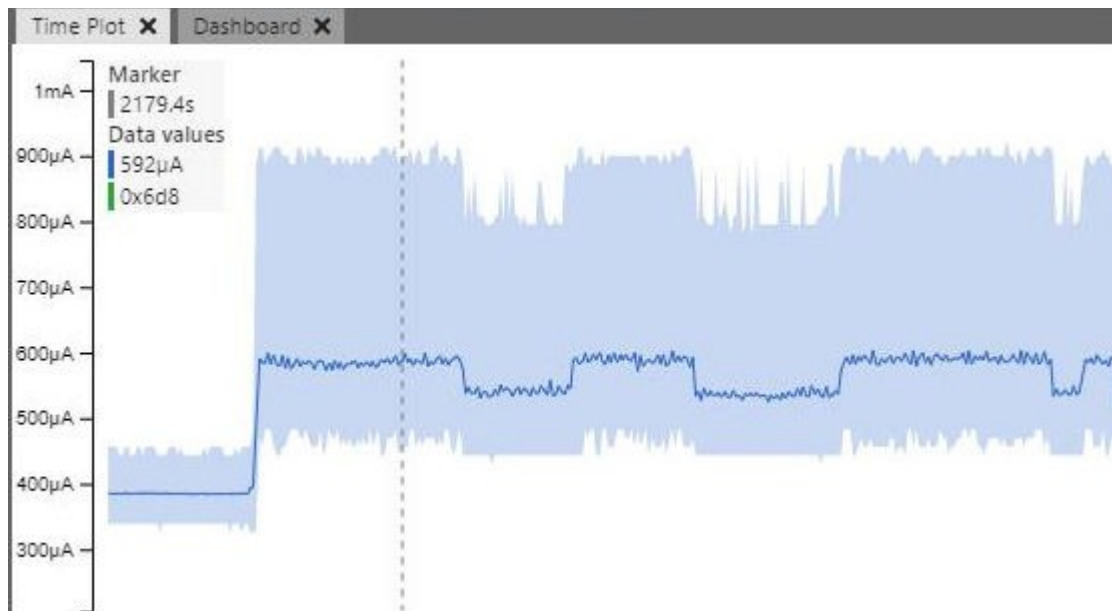
関連リンク

[4.3.1.4.2. プロット カーソル](#)

4.4.1.4 Power 計測におけるノイズ

ターゲット デバイス自体の消費電力が非常にわずかである場合、計測される電力に対してノイズが相対的に大きくなるため、プロットに顕著なノイズが現れる可能性があります。このノイズは、デバッグ セッション中のターゲット デバイスとデバッガの相互作用または電源あるいは計測回路内のノイズ等の影響により生じます。

例えば、下図の低消費電力回路の電流プロットでは、ノイズが顕著に現れています。



しかし、下図の高消費電流回路の電流プロットではノイズは非常に小さく見えます。

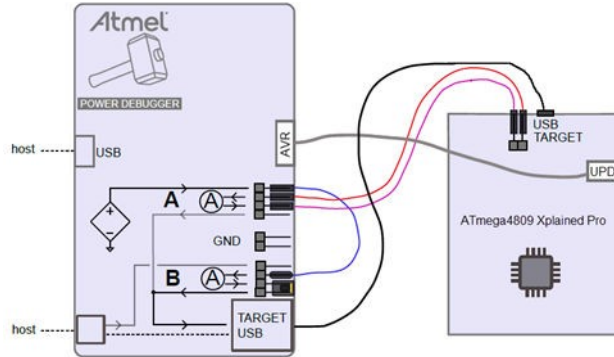


4.4.2 複数回路の消費電流計測

各軸に複数のプロットを表示できます。この例では、Power Debugger のチャンネル A と B の両方を使って、ATmega4809 Xplained Pro ターゲットボードからデバイス単体とターゲットボード全体の Power データを収集します。

4.4.2.1 ハードウェアのセットアップ

Power Debugger を次ページの接続図に従ってターゲットボードに接続します。



Power Debugger の可変電圧電源を使ってターゲットボードに給電します。これにより、USB 仕様の全電圧レンジで USB デバイスを試験できます。

手順

- ストラップ ケーブルを使って、可変電圧出力をチャンネル B の入力に接続します。これにより、ターゲットボード全体が消費する電流を計測できます。
- チャンネル B の出力を USB type-A ジャックの POWER 入力に接続します(ジャンパプラグを使用)。
- USB ケーブルを使って type-A ジャックからターゲットボードの「TARGET」USB micro ジャックに接続します。

ターゲット MCU が消費する電流を計測するため、Xplained PRO ボードは TARGET USB コネクタの隣に 2 ピン電源ヘッダを備えています。通常このヘッダにはジャンパプラグが装着されており、ヘッダの VCC_TARGET ピンと VCC_MCU ピンは互いに接続されています。このジャンパプラグを抜き取り、この回路を電流計測チャンネルへ経由させる事により、MCU のみによって消費される電流を計測できます。

手順

- ターゲットボードの VCC_TARGET ピンをチャンネル A の入力に接続します。
- チャンネル A の出力をターゲットボードの VCC_MCU ピンに接続します。

ターゲット デバイスをプログラミングおよびデバッグします。Xplained PRO はデバッグ機能を備えていますが、この例では Power Debugger を使います。

手順

- 10 ピン デバッグケーブルを使って、Power Debugger の AVR ヘッダから Xplained PRO の UPDI DEBUG ヘッダへ接続します。

最後に Power Debugger をホスト コンピュータに接続します。

手順

- USB ケーブルを使って、Power Debugger の DEBUG コネクタからホスト コンピュータへ接続します。
- USB ケーブルを使って、Power Debugger の TARGET コネクタからホスト コンピュータへ接続します。



ヒント: プログラミング ヘッダを接続する際に GND 接続は不要です。

4.4.2.2 デバッグと表示

ターゲットボード向けに使用するアプリケーションは、以下のサンプルコード内の Example 2 に基づきます。

[MPLAB XC8 User's Guide For Embedded Engineers AVR Code Examples](#)


本書の例では Atmega4809 Curiosity の代わりに ATmega4809 Xplained Pro を使うため、コード内の「PORTD」を「PORTB」に変更します。

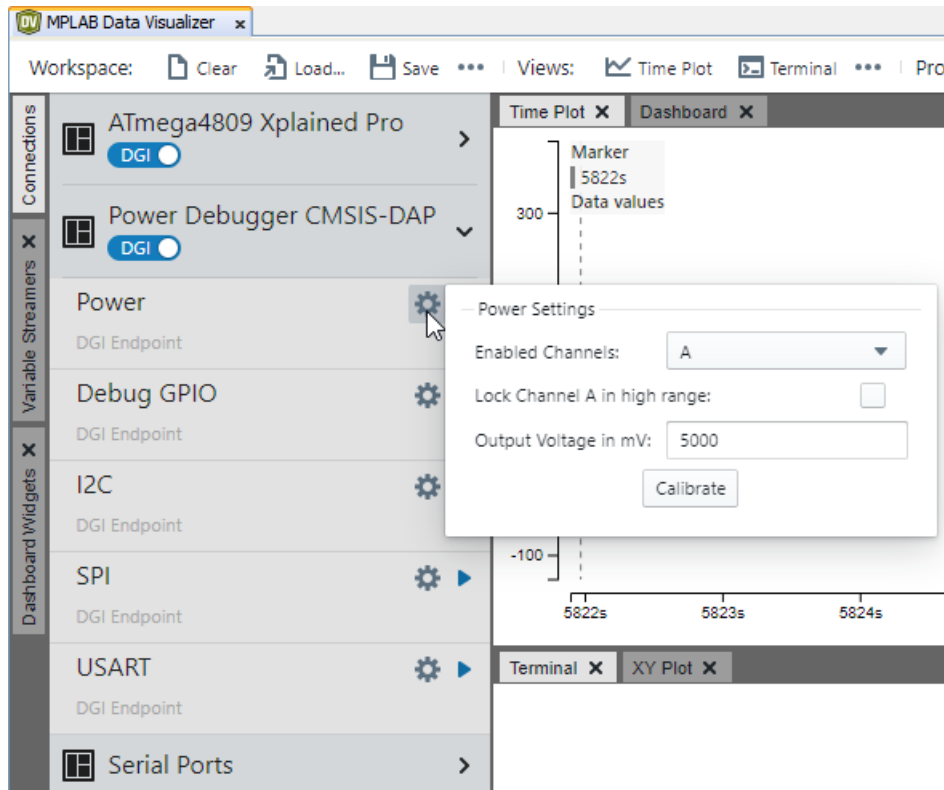
手順

1. MPLAB X IDE v5.50 以上で Example 2 を開きます。
2. コードを前述の通りに変更します。

MPLAB X IDE 内で MPLAB Data Visualizer を開きます。

手順

1. メニュー **Window > Debugging > Data Visualizer** を選択します。これにより MPLAB Data Visualizer が開きます。
2. Data Visualizer によって Power Debugger DGI インターフェイスが検出される必要があります。「Power」ソースのオプション アイコン  をクリックし、ダイアログに設定を入力します。設定を入力した後に再度このアイコンをクリックすると、その設定が保存されてダイアログが閉じます。

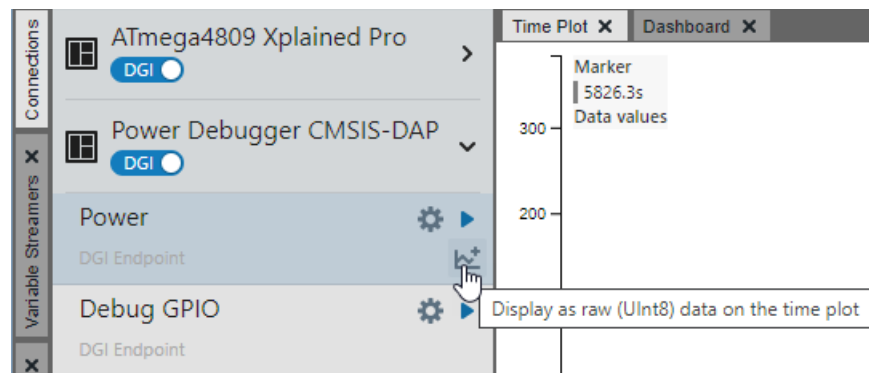


3. ATmega4809 Xplained Pro に電源が投入され、このターゲットボードが Data Visualizer によって検出される必要があります。ただし、本例では Power Debugger 接続のみ使用します。

この時点で、Data Visualizer 内でのデバッグと出力のプロットが可能となります。

手順

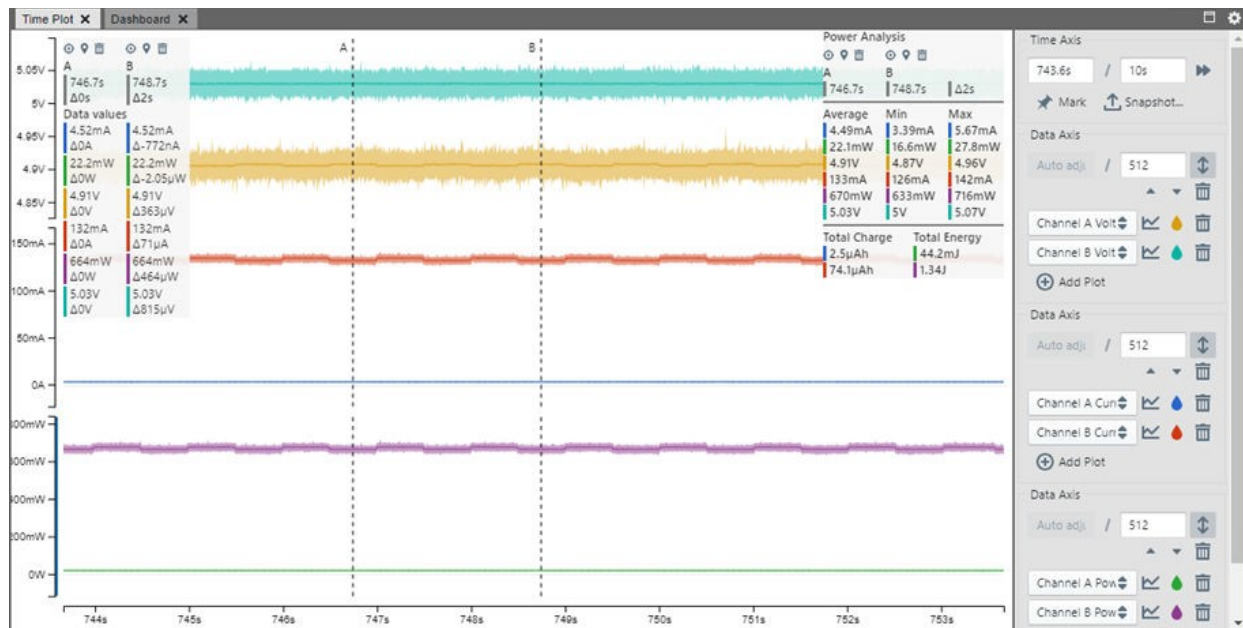
1. サンプルコードをデバッグします。実行中に Xplained Pro ボード上のユーザ LED が点滅します。
2. 「Power」インターフェイスで下図のアイコンをクリックする事により、時系列プロットに全てのソースを表示させます。



4.4.2.3 プロットの表示を拡大する

MPLAB Data Visualizer は電圧、電流、電力の各グラフに 2 つのプロット(ATmega4809 デバイスからのデータと ATmega4809 Xplained Pro ターゲットボードからのデータ)を表示します。Power Analysis オプションを使ってデータの解析は可能ですが、プロットは見やすくありません。

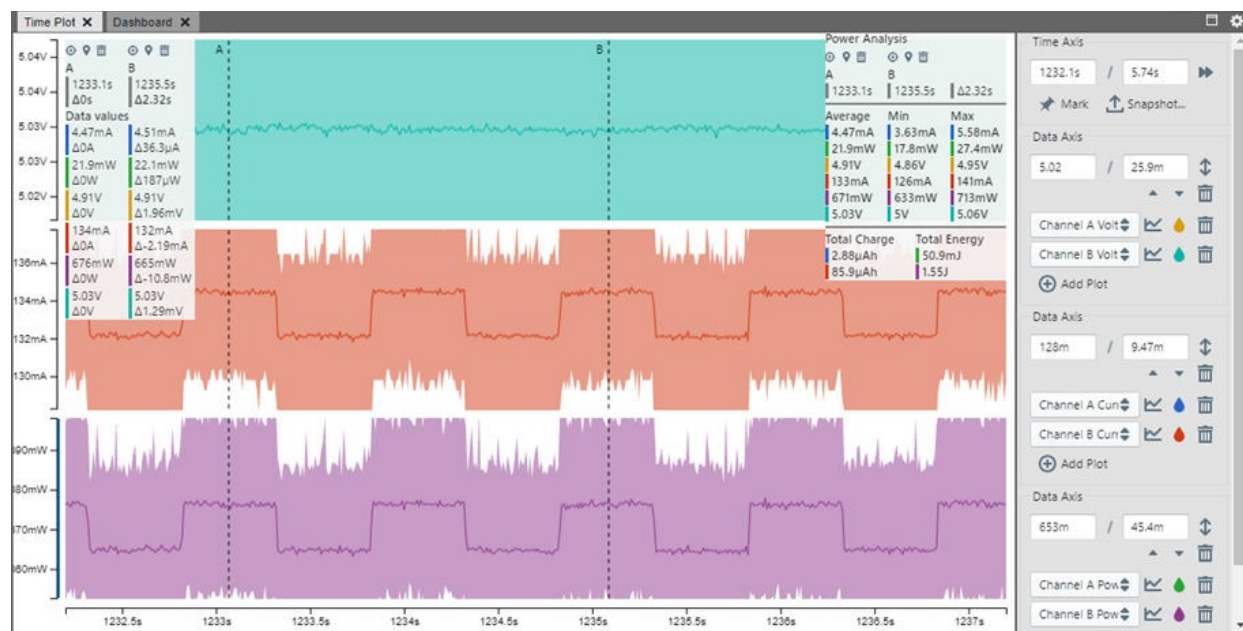
図 4-39. 各グラフにチャンネル A とチャンネル B のデータが一緒にプロットされる



マウスカーソルをデータ軸上に置き、ホイールを使って軸レンジを調整する事で、どちらか 1 つのチャンネルのプロットを見やすくします。

プロットのノイズについては、4.4.1.4. 「Power 計測におけるノイズ」を参照してください。

図 4-40. チャンネル B のプロットを見やすく拡大する



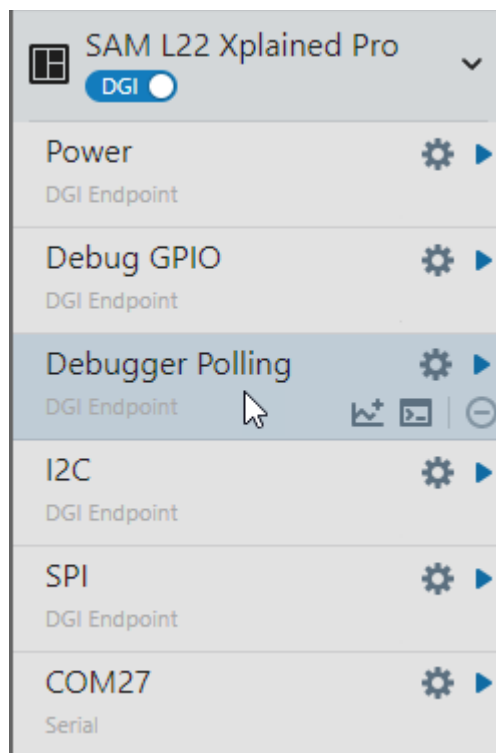
関連リンク

[4.3.1.2. プロットの拡大/縮小](#)

4.5 プログラム カウンタ(PC)のサンプリング

左側の[Connections]タブでは、接続されたデバッガまたはキットの下の DGI セクションに「Debugger Polling」インターフェイスが表示されます(図 4-41 参照)。Data Visualizer でこのインターフェイスを使う事により、プログラム カウンタ(PC)値をサンプリングして時系列プロットに表示する事ができます。

図 4-41. Debugger Polling インターフェイス



Note: プログラム カウンタ(PC)サンプリング機能は、デバッガ実装内のリソースに依存するため、プラグイン バージョンの MPLAB Data Visualizer でのみ利用可能です。

Note: MPLAB ICE 4 や Power Debugger 等のデバッガは、ターゲット デバイスが PC サンプリングをサポートするかどうかに関係なく、常に Debugger Polling インターフェイスが利用可能であると示します。しかし、デバイスアーキテクチャが PC サンプリングをサポートしない場合があります。一般的に、PC サンプリング機能は全ての SAM デバイスと比較的新しい AVR デバイス向けに利用可能です。

Data Visualizer では、PC 値を例えば電流計測値と同じグラフにプロットして、消費電力と実行コードの関係を観察する事ができます。

ほとんどの場合、ターゲットの命令実行速度と同じ速度で PC 値を読み出す事はできないため、サンプリングされる PC 値は実際のコード実行と完全には同期しません。しかし、どの時点でどのコード セグメントが実行中であるのかを示すには十分に役立ちます。ターゲットから PC 値が読み出されるたびに、現在実行中のコード位置の正確なアドレスを取得する事ができます。

4.5.1 PC サンプリングの例

以下では、同じ時系列プロットに Power データと PC サンプリング値を一緒に表示する方法について説明します。

この例は、Atmel Start からの [USB MSC サンプル プロジェクト](#)に基づきます。この例では SAM L22 Xplained Pro キットを使いますが、PC サンプリングをサポートする他のデバッガおよびキットでも手順は同じです。

デバッグを開始し、Data Visualizer ウィンドウを開いておく必要があります。


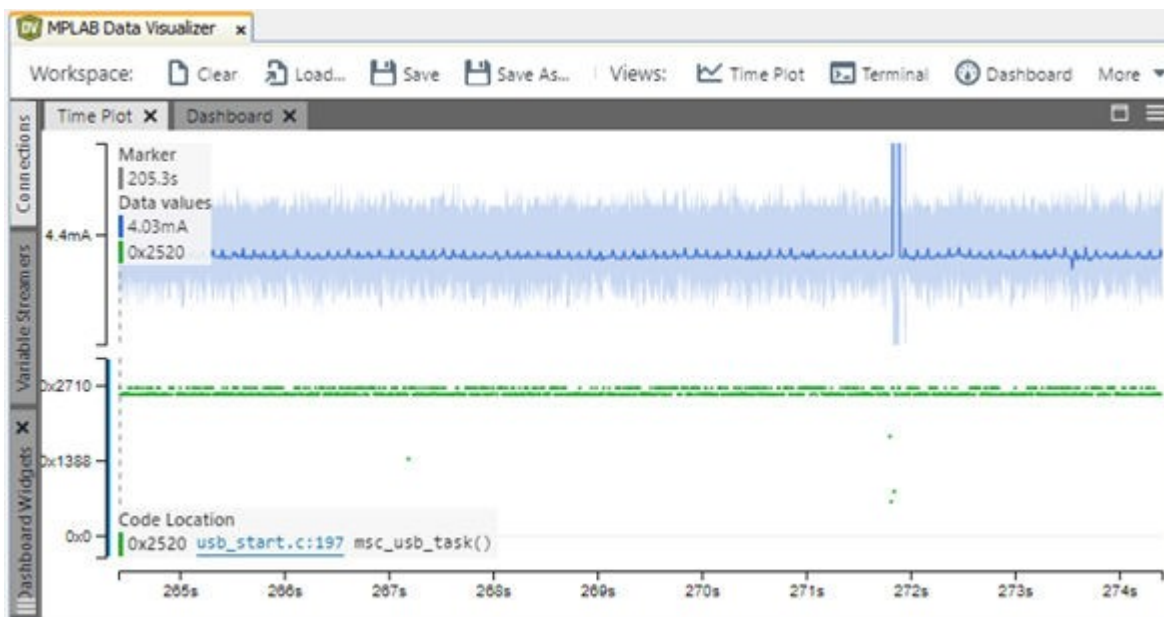
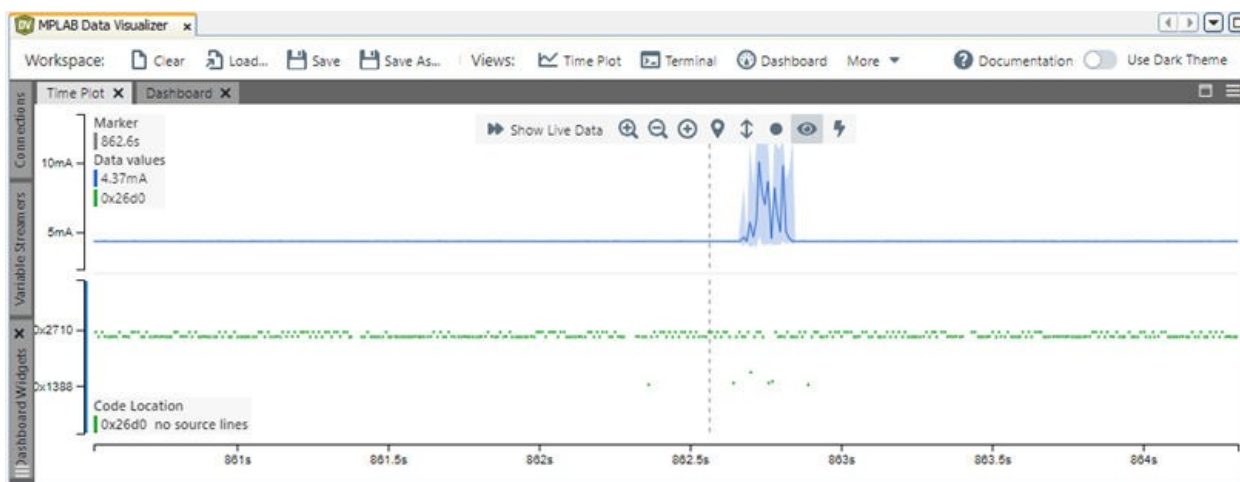
接続したキットの DGI インターフェイスのリストから「Debugger Polling」インターフェイスを見つけます。このインターフェイスの上にマウスカーソルを置き、 をクリックする事により全てのソースをプロットします。これにより、PC サンプリング信号が現在の時系列プロットに追加されます(図 4-42 内の緑プロット)。この図には、既に追加されていた消費電流のプロットが青で示されています。

図 4-42. PC サンプリグ値を追加したプロット



この例では、Windows Explorer 内にマストレージ デバイスがドライブとして表示されます。このドライブが Data Visualizer 内でプロファイリングされる際に、ドライブに対する「フォーマット」動作が発生します。図 4-43 では、このフォーマット動作中に消費電力のスパイクが発生し、それと同時に PC 値にも明確な変化が見られます。

図 4-43. PC サンプリグ値と消費電力値の変化




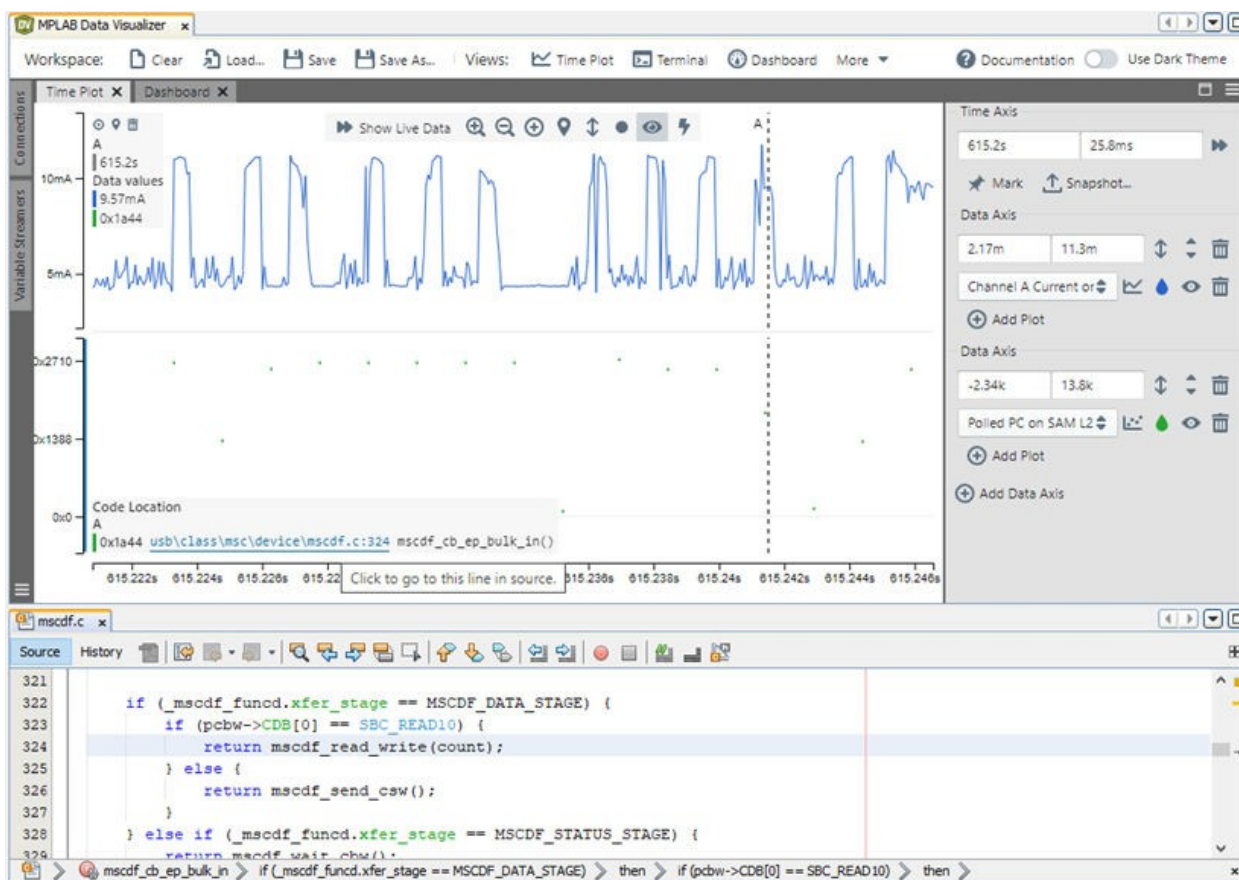
時系列プロット内で消費電力が増加している区間を拡大します(図 4-44 参照)。解析のために「Inspect」 機能を有効にしておく必要があります。これにより、カーソルに最も近い位置の PC サンプリグ値が、プロット領域の左上隅に表示されます。加えて、ソースコード位置がデコードされて(可能な場合のみ)、プロット領域の左下隅に表示されます。図 4-44 に示す例では、カーソル位置の PC 値は 0x1A44 であり、これは USB MSC 処理に関連する関数へとデコードされています。表示されたソースコード リンクをクリックするとソースファイルが開き、対応するソース行にテキストカーソルが置かれます。

図 4-44. PC 値とソースコード位置の表示



4.6 XY プロットに 2 セットのデータを表示する

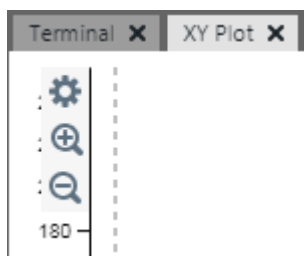
時系列プロットの他に XY プロットが使えます。このプロットでは、2 つのデータセットの間の関係をグラフに表示することができます。これは、前述の機械学習向けの入力データ用に役立ちます。

4.6.1 [XP Plot]ウィンドウ

[XP Plot]ウィンドウでは、X 軸と Y 軸に異なるデータソースを割り当てます。このウィンドウ内のデータ点は、[Time Plot] ウィンドウ内の時間軸と連動しています。[Time Plot] ウィンドウでライブデータを表示している時、または時間軸を左右にドラッグした時に、[XY Plot] ウィンドウに表示されるデータも連動して変化します。

軸オプションは、右側ウィンドウ枠で選択します(図 4-46 参照)。あるいは、データ軸の上端に表示されるアイコン(⚙️)を使って軸オプションを編集する事もできます。XY プロットでも軸の拡大/縮小とマーカーが利用できます。

図 4-45. XY プロットの制御アイコン



サポートされるプラグインツールから追加の機能が利用できる場合があります。

関連リンク

4.3.1.4.1. プロットマーカー

4.3.1.2. プロットの拡大/縮小

4.6.2 XY プロットのパラメータ設定

右側ウィンドウ枠では、X軸とY軸の制御パラメータを設定できます。

図 4-46. XY プロットの制御パラメータ

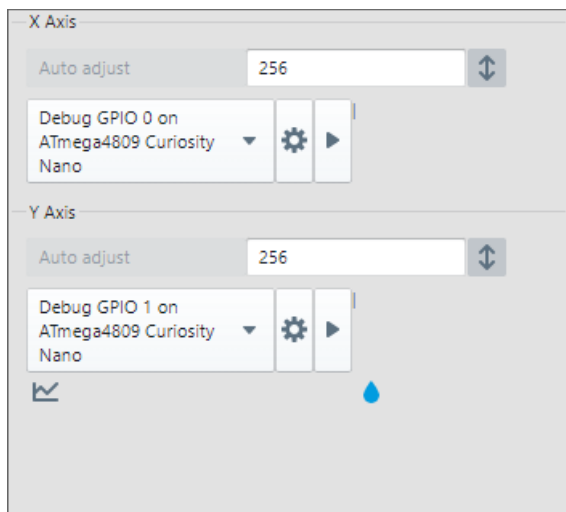


表 4-7. データ軸

制御項目	概要
Offset	データ軸の自動調整が有効である場合、「Auto adjust」と表示されます。自動調整が無効である場合、プロットの現在のオフセットが表示されます。
Scale	データ軸の分解能を指定します。
	自動調整の有効/無効を切り換えます。 有効である場合、軸のレンジは自動的に調整されます。無効である場合、軸のレンジは手動で調整します。

表 4-8. データ軸 - プロットのソースと書式

制御項目	概要
Data Source	プロットするデータソースをドロップダウン リストから選択します。データソースの選択と設定は左側の[Connections]タブを参照してください。
	ソース設定を選択します。
	データのストリーミングを開始/停止します。X軸またはY軸のどちらかのデータソースでこれらをクリックすると、両方のデータソースでストリーミングが開始/停止します。
	データ点のプロット方法を選択します。 <ul style="list-style-type: none"> データ点をステップ状の線で結びます(既定値)。 データ点を直線で結びます。 データ点のみを表示します(線なし)。
	プロットの色を選択します。

4.6.3 XY プロットの例

以下では、XY プロットと時系列プロットの使用例を示します。

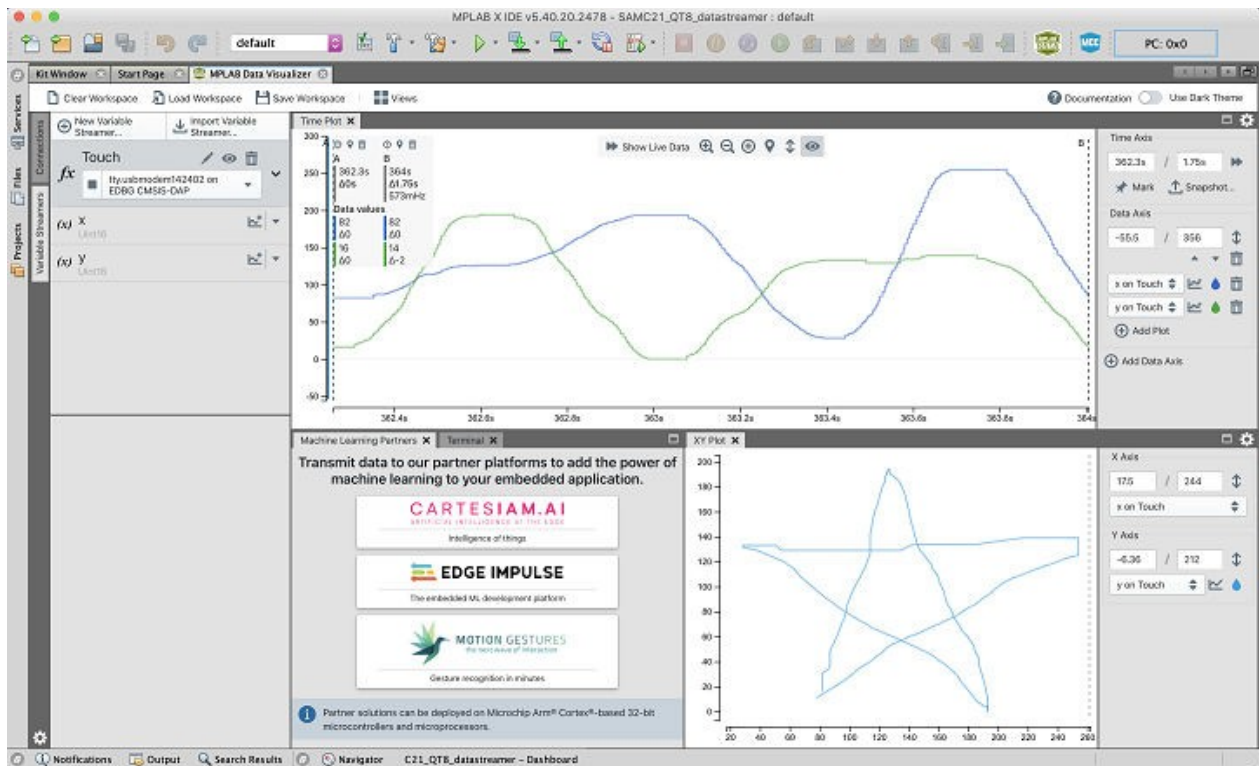
4.6.3.1 MPLAB Touch プラグインの例

MPLAB Touch プラグインを使うと、MPLAB Data Visualizer にタッチ調整データを表示できます。このプラグインは双方向通信をサポートし、プロジェクトをリビルドする事なくタッチ設定を変更できるため、調整に要する時間を節約できます。

MPLAB Data Visualizer でサポートされるプラグインを見つけるには、MPLAB X IDE 内でメニュー **Tools > Plugins** を選択します。プラグインを選択し、[Install] をクリックして画面の指示に従います。プラグインをインストールした後は、再起動が必要です。左側に「Touch」接続タイプが表示され、その下に適用可能な接続が示されます。

詳細は、弊社ウェブページ「[Introduction To Touch Plugin](#)」を参照してください。タッチ入力は機械学習と一緒に使います。

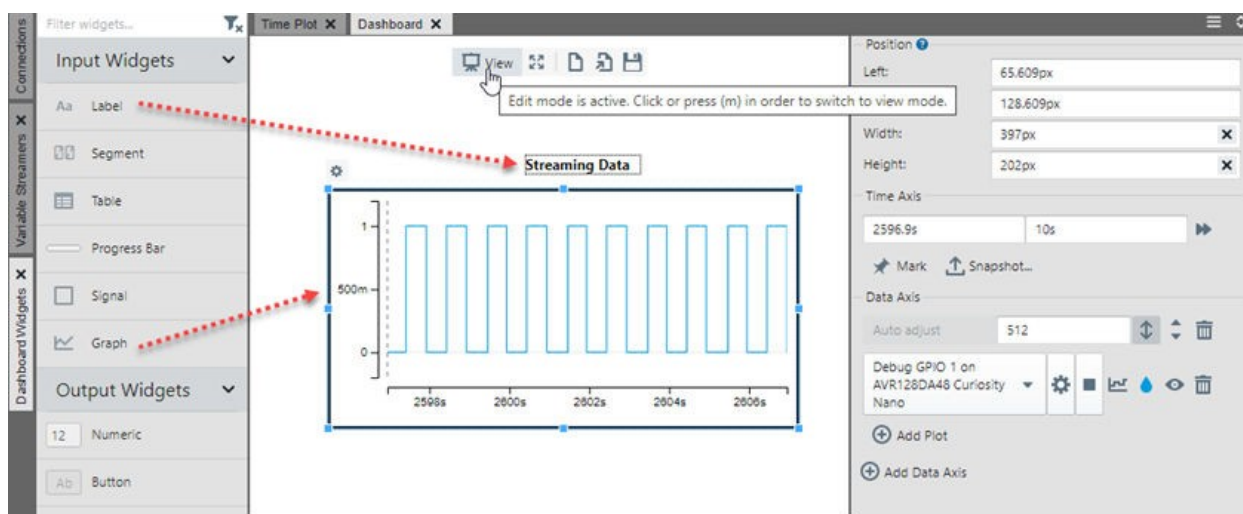
図 4-47. MPLAB Touch プラグイン



4.7 ダッシュボード ウィジェットを使ったデータ表示

ダッシュボードはカスタマイズ可能なグラフィカル ユーザ インターフェイス (GUI) であり、アプリケーション ファームウェアからのパラメータを制御および表示するために使えます。ウィジェット (ボタン、ラベル、スライダ等) を左側の [Dashboard Widgets] タブから中央の [Dashboard] ウィンドウへドラッグする事により GUI を構成します。各ウィジェットには、値を受信または送信するためのデータフィールドを割り当てる事ができます。これを行うには、ソースをウィジェットへドラッグするか、ウィジェットのデータフィールド選択アイコンを使います。このアイコンは、ウィジェットを選択すると表示されます。あるいは、右側ウィンドウ枠で各種パラメータを設定する事もできます。

図 4-48. ダッシュボード



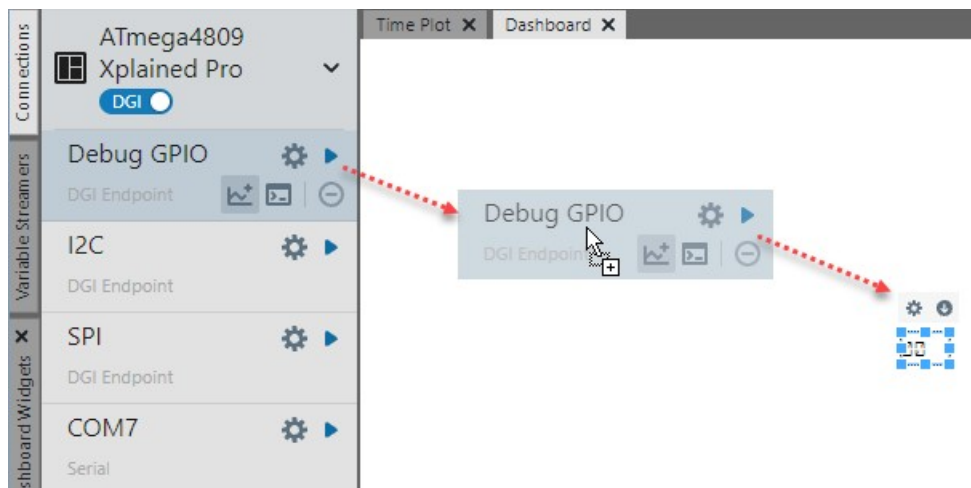
各ウィジェットの設定例を以下に示します。

4.7.1 [Dashboard]ウィンドウ

[Dashboard] ウィンドウ内へウィジェットまたはデータソースをドラッグし、それらを使ってアプリケーションパラメータを制御または表示できます。利用可能なウィジェットは左側の[Dashboard Widgets] タブ内に表示されます。[Connections]タブ内のデータソースまたは[Variable Streamers]タブ内の Variable Streamer フィールドを[Dashboard] ウィンドウにドラッグして Segment ウィジェットとして表示することもできます(図 4-49 参照)。ストリーマグループは Table ウィジェットとして表示されます。

この図のように[Connections]タブから「Debug GPIO」データソースを[Dashboard] ウィンドウにドラッグすると、ウィジェットは GPIO バス値を表示します。個々の GPIO ラインをソース選択ドロップダウンリストから選択できます。同様に「Power」データソースを[Dashboard]ウィンドウにドラッグすると、ウィジェットはチャンネル A の電力値を表示します。その他の Power データフィールドはソース選択ドロップダウンリストから選択できます。

図 4-49. [Dashboard]ウィンドウ

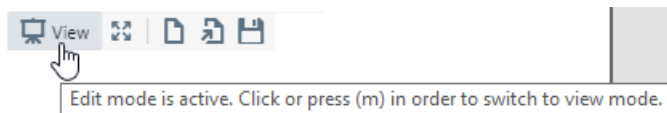


関連リンク

[4.7.2.\[Dashboard Widgets\]タブ](#)

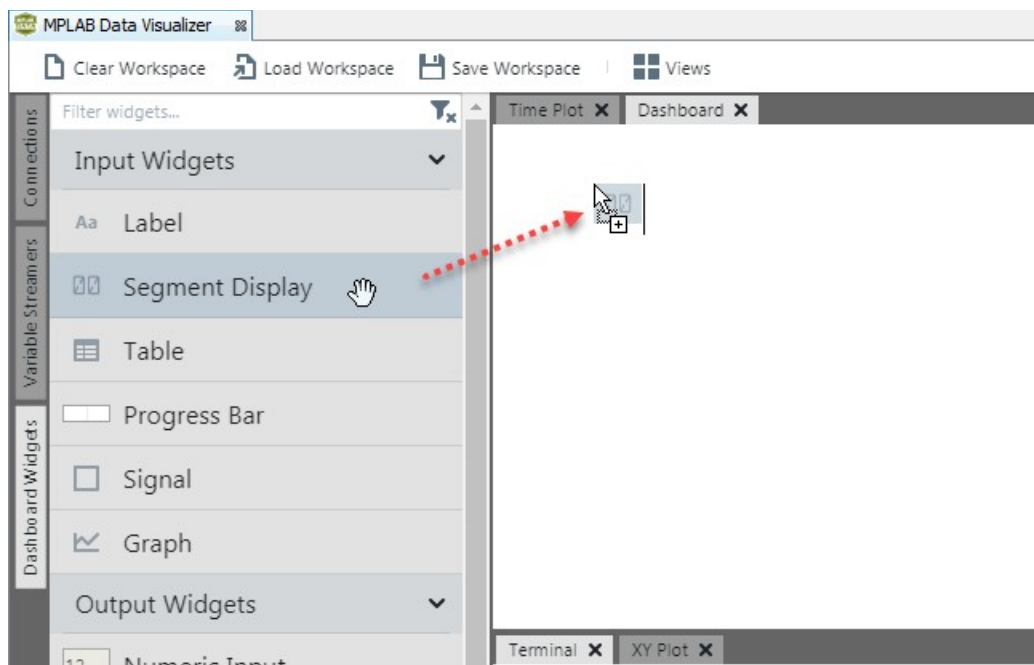
4.7.1.1 ダッシュボードのモード

[Dashboard] ウィンドウには 2 つのモード(Edit モードと View モード)があります。メニューバー上のアイコンをクリックする事により、モードを切り換える事ができます(次ページの図参照)。どちらのモードでもウィジェットを追加できますが、ウィジェットプロパティの編集は Edit モードでのみ行えます。



4.7.1.1.1 ウィジェットを[Dashboard]ウィンドウ内に配置する

左側の[Dashboard Widgets]タブからドラッグ&ドロップにより中央の[Dashboard] ウィンドウにウィジェットを追加します。これは View モードでも Edit モードでも可能です。各ウィジェットの詳細は 4.7.2. 「[Dashboard Widgets] タブ」を参照してください。



4.7.1.1.2 ウィジェットの設定

全てのダッシュボード ウィジェットは、Edit モード時に設定できます。パラメータはウィジェットのタイプに応じて異なりますが、変更方法は同じです。図 4-50 に進捗バーの例を示します。


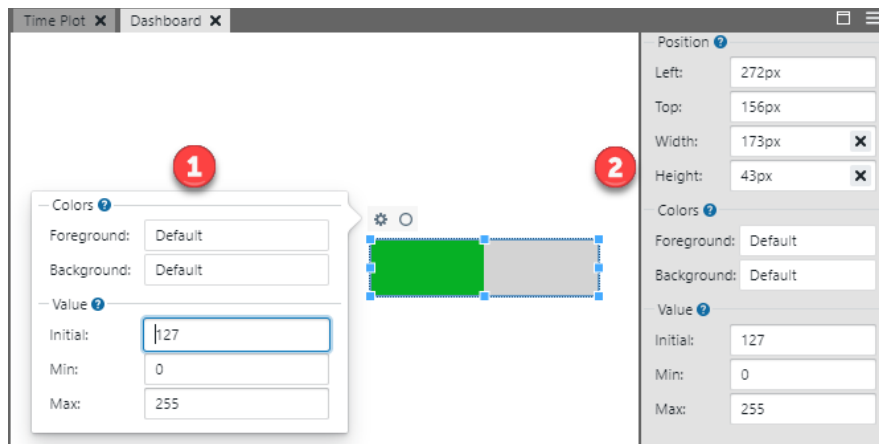
1. **オプション アイコンを使う方法:** ウィジェットをクリックして選択します。次にギアアイコン  をクリックし、表示されたウィジェット設定オプションを変更します。
2. **右側ウィンドウ枠を使う方法:** ウィジェットをクリックして選択します。次に右側ウィンドウ枠内でウィジェットの位置および設定オプションを変更します。

図 4-50. ウィジェットの設定



4.7.1.1.3 ウィジェットの移動とサイズ変更

ウィジェットの位置とサイズに関連する全てのパラメータは、右側ウィンドウ枠内の「Position」の下で設定できます。ウィジェットの位置とサイズは、絶対値または相対値で指定できます。絶対値で指定する場合、値の後に「px」を付けます。相対値で指定する場合、値の後に「%」を付けます。

図 4-51. 絶対値によるウィジェットの位置およびサイズ指定



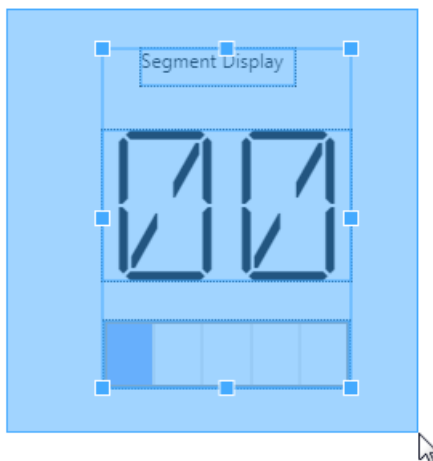
図 4-52. 相対値によるウィジェットの位置およびサイズ指定



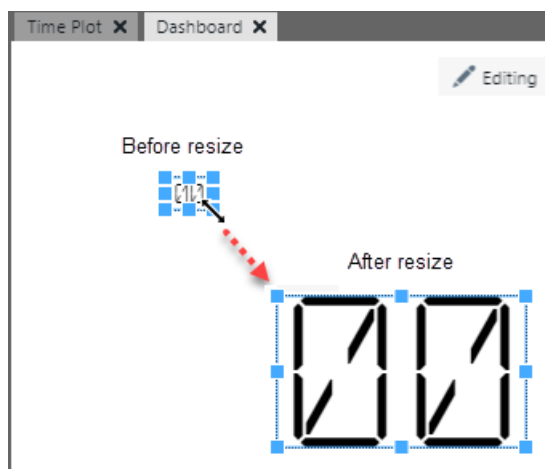
[Dashboard] ウィンドウ内でドラッグ&ドロップによってウィジェットの位置とサイズを変更する事もできます。

Note: [Dashboard]ウィンドウ内でウィジェットの位置/サイズを変更した場合、右側ウィンドウ枠内の[Position] パラメータの値は絶対値(px)となります。

- [Dashboard] ウィンドウ内で 1 つのウィジェットまたは複数ウィジェットのグループをドラッグ&ドロップによって移動させる事ができます。複数ウィジェットを 1 つのグループとして選択するには、グループの外側をクリックし、四角の領域でそれらの全てを囲むようにドラッグします。



- 1 つのウィジェットまたはウィジェットのグループのサイズを変更する場合、それらを選択した後に、ウィジェットの四角の頂点の 1 つをクリックしてドラッグします。



4.7.1.1.4 ウィジェットの削除

Edit モードでは、ウィジェットまたはウィジェットのグループを選択した後に<Delete>キーを押す事により、それらのウィジェットを削除できます。複数ウィジェットを 1 つのグループとして選択するには、グループの外側をクリックし、四角の領域でそれらの全てを囲むようにドラッグします。

Note: 削除すると全ての設定は失われます。削除を取り消す事ができません。

4.7.1.1.5 ダッシュボードのクリア、インポート、エクスポート

これらの操作は、Edit モード中にのみ行えます。

表 4-9. ダッシュボードのオプション

ツールバー アイコン	概要
Clear Dashboard	ダッシュボードの全ての内容をクリアします。取り消しはできません。
Import Dashboard	以前に保存した Dashboard ファイル(.json または .db)をインポートします。インポート後に、ウィジェットのソース設定*を選択する必要があります。
Export Dashboard	現在のダッシュボードの内容をファイル(.json)へエクスポートします。ウィジェットのみが保存され、ソース設定*は保存されません。
* ウィジェットのソース設定の読み込みまたは保存には、Workspace Load または Save を使います。	

4.7.1.1.6 ダッシュボードの全画面表示

これは Edit モード中にのみ行えます。

ダッシュボードを全画面表示に拡大するには、ダッシュボード メニュー上の[Toggle Fullscreen]アイコン をクリックするか、キーボードで<f> キーを押します。同じ操作を繰り返す事で、表示を元に戻す事ができます。

4.7.1.1.7 ヒントとコツ

ダッシュボードを使う際は、以下の情報が役立ちます。

- 配色が既定値のままである場合(あるいはクリアした場合)、ほとんどのウィジェットにはライトモードまたはダークモードのテーマカラーが適用されます(で選択)。色を設定した場合、これらのテーマカラーの影響を受けません。色設定はいつでも変更またはクリアできます。
- テキストを表示するウィジェットのサイズを変更すると、テキストがウィジェットの境界線からはみ出してしまう場合があります。「テキスト折り返し」(Text Wrapping) 設定により、ウィジェット内のテキストの表示方法を変更できます。しかし、ウィジェットが小さすぎると、やはりテキストははみ出してしまうます。テキストのはみ出しを防ぐには、右側ウィンドウ枠内でウィジェットの「Width」および「Height」パラメータをクリアします(右の[X]をクリック)。すると、ウィジェットのサイズはテキストが収まるように自動的に調整されます。

4.7.2 [Dashboard Widgets]タブ

[Dashboard Widgets] タブをクリックして開くと、[Dashboard] ウィンドウ内で表示、配置、使用可能なウィジェットが表示されます。ウィジェットを[Dashboard] ウィンドウ内にドラッグ&ドロップした後に、データフィールド(シンクまたはソース)を選択する事により、そのウィジェットの用途(ストリーミング データの入力用またはアプリケーションへの出力用)を指定します。

図 4-53. [Dashboard Widgets]タブ

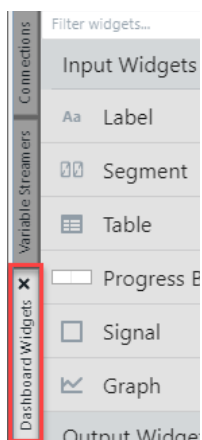


表 4-10 に全てのウィジェットに共通する「Position」パラメータを示します。その後で、ウィジェットのタイプごとに専用の設定パラメータについて説明します。

表 4-10. 全ウィジェットに共通のパラメータ - Position

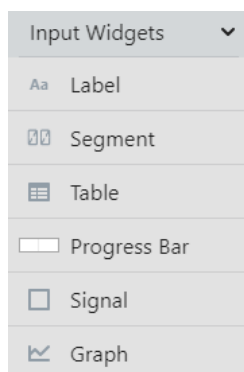
パラメータ	タイプ	用途
Left	数値	ウィジェットの左端の横方向位置(単位: px または%)
Top	数値	ウィジェットの上端の縦方向位置(単位: px または%)
Width	数値	ウィジェットの幅(単位: px または%)
Height	数値	ウィジェットの高さ(単位: px または%)

関連リンク

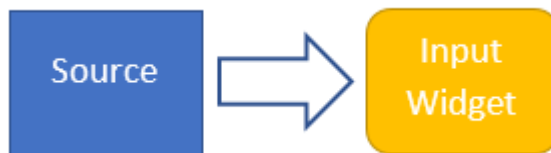
[4.7.1.\[Dashboard\]ウィンドウ](#)

4.7.2.1 入力ウィジェット

図 4-54. 入力ウィジェットのリスト



入力ウィジェットは、シンク データフィールドとしてデータの入力を受け付けます。



4.7.2.1.1 Label

Label ウィジェットはテキスト文字列を表示します。



データフィールド

Label ウィジェットは、全てのタイプのソースを受け付けるシンク データフィールドを持ちます。Label ウィジェットへ送信された全てのデータは、文字列に変換されてテキストとして表示されます。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-11. Label グループ

パラメータ	タイプ	用途
Value	テキスト	ラベルのテキストを入力します。

表 4-12. Colors グループ

パラメータ	タイプ	用途
Foreground	色選択ダイアログ	前景色を指定します。
Background	色選択ダイアログ	背景色を指定します。

表 4-13. Font グループ

パラメータ	タイプ	用途
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	フォントの太さ(bold 等)を選択します。
Font Style	ドロップダウン ボックス	フォントのスタイル(italics 等)を選択します。
Font Family	ドロップダウン ボックス	フォントファミリー(segment 等)を選択します。

表 4-14. Alignment グループ

パラメータ	タイプ	用途
Horizontal	ドロップダウン ボックス	表示内容の横方向の配置を選択します。
Vertical	ドロップダウン ボックス	表示内容の縦方向の配置を選択します。
Line Wrap	ドロップダウン ボックス	表示内容の折り返し条件を選択します。

4.7.2.1.2 Segment ウィジェット

Segment ウィジェット エレメントは、16 セグメント LED ディスプレイをシミュレートします。



データフィールド

Segment ウィジェット エレメントは、全ての数値データタイプを受け付けるシンク データフィールドを持ちます。このウィジェットは、受信した値を表します。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-15. Segment グループ

パラメータ	タイプ	用途
Min. Digits	数値	表示する桁数を入力します。 既定値 = 2
Num.Base	数値	基数を入力します。 既定値 = 10 (10 進値)

表 4-16. Colors グループ

パラメータ	タイプ	用途
Foreground	色選択ダイアログ	前景色を指定します。
Background	色選択ダイアログ	背景色を指定します。

表 4-17. Font グループ

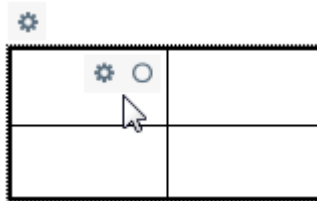
パラメータ	タイプ	用途
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	フォントの太さ(bold 等)を選択します。
Font Style	ドロップダウン ボックス	フォントのスタイル(italics 等)を選択します。
Font Family	ドロップダウン ボックス	フォントファミリ(segment 等)を選択します。

表 4-18. Alignment グループ

パラメータ	タイプ	用途
Horizontal	ドロップダウン ボックス	表示内容の横方向の配置を選択します。
Vertical	ドロップダウン ボックス	表示内容の縦方向の配置を選択します。
Line Wrap	ドロップダウン ボックス	表示内容の折り返し条件を選択します。

4.7.2.1.3 Table ウィジェット

Table ウィジェットは、1 つまたは複数のデータソースを表形式で表示します。このウィジェットは 2 つのモード(手動ラベルモードと自動ラベルモード)をサポートします。手動ラベルモードでは、表の各セルを手動でラベルセルまたはデータセルに設定できます。自動ラベルモードでは、表の各セルが 2 つのフィールドに分割されます。左のフィールドはデータストリームの名前を持つラベルであり、右のフィールドはストリームの実際のデータです。このモードは、チェックボックス[Auto Labels]により選択します。



手動ラベル

手動ラベルモードの場合、表の各セルをラベルセルまたはデータセルのどちらかに設定します。既定値では、全てのセルがデータセルです。ラベル値を指定すると、そのセルはラベルセルになります(下図参照)。



データセルのみがデータ フィールドを持ちます。

自動ラベル

各セルに 1 つのデータソースが割り当てられ、データソースの名前がセルの左側フィールドに表示され、実際のデータ値がセルの右側フィールドに表示されます。ソース名は、シンク データフィールドに接続されたソースから自動的に取得されます。

Delta1	0	Delta6	0
Delta2	0	Delta7	0
Delta3	0	Delta8	0
Delta4	0	Delta9	0
Delta5	0	Delta10	0

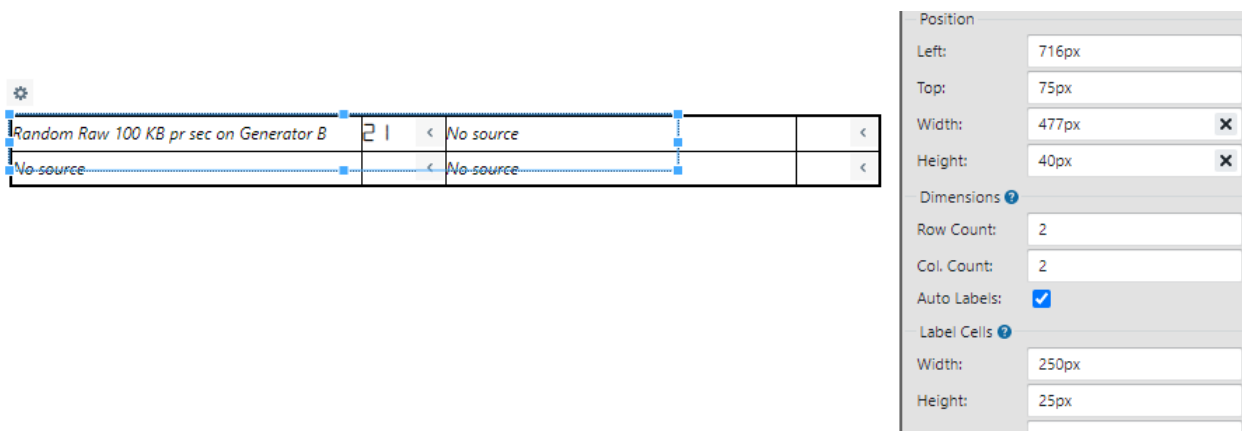
Table ウィジェットの各セルは 1 つのデータ フィールドを持ち、全てのデータソースを受け付けます。データは文字列に変換されてテキストとして表示されます。

テーブルのサイズ変更

セルのサイズは最小サイズとして指定されます。テーブルのサイズが指定されたセルサイズの合計よりも大きくなるよう変更された場合、セルはテーブルサイズに比例して拡大されます。テーブルのサイズが指定されたセルサイズの合計よりも小さくなるよう変更された場合、テーブルは最小セルサイズを維持し、ウィジェットは枠からはみ出しません(図 4-45 参照)。

右側ウィンドウ枠内の幅(Width)および高さ(Hight)の値をクリア(対応する「X」ボタンをクリック)すると、Table ウィジェットのサイズは自動的に調整されます。

図 4-55. サイズ変更後のテーブル - 枠からはみ出し



設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-19. Dimensions グループ

パラメータ	タイプ	用途
Row Count	数値	テーブルの行数を指定します。
Col. Count	数値	テーブルの列数を指定します。
Auto Labels	チェックボックス	チェックマークを付ける事で自動ラベルモードに設定します。

表 4-20. Colors グループ

パラメータ	タイプ	用途
Foreground	色選択ダイアログ	前景色を指定します。
Background	色選択ダイアログ	背景色を指定します。

表 4-21. Label Cells、Data Cells グループ

パラメータ	タイプ	用途
Min. Width	数値	各セルのラベル部またはデータ部の最小幅を指定します。この値を変更すると、テーブル全体の幅が変化します。
Min. Height	数値	各セルのラベル部またはデータ部の最小高さを指定します。この値を変更すると、テーブル全体の高さが変化します。
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	文字の太さ(Normal または Bold)を選択します。
Font Style	ドロップダウン ボックス	文字のスタイル(Normal または Italics)を選択します。
Font Family	ドロップダウン ボックス	Segment を選択します。
Horizontal	ドロップダウン ボックス	セル内容の横方向の配置を選択します。

4.7.2.1.4 Progress Bar ウィジェット

Progress Bar ウィジェットは、入力された値を指定された最小値と最大値の間の位置で示すバーグラフです。



データフィールド

Progress Bar ウィジェットは、全ての数値データタイプを受け付けるシンク データフィールドを持ちます。このウィジェットの色付き領域の長さは、入力された値に応じて最小値と最大値の間で変化します。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-22. Value グループ

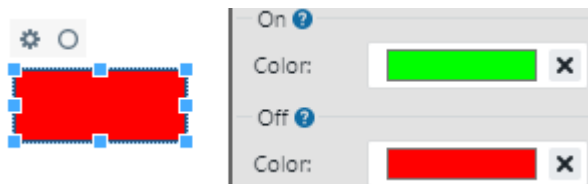
パラメータ	タイプ	用途
Initial	数値	初期値を指定します。
Min	数値	入力の最小値を指定します。
Max.	数値	入力の最大値を指定します。

表 4-23. Colors グループ

パラメータ	タイプ	用途
Foreground	色選択ダイアログ	前景色を指定します。
Background	色選択ダイアログ	背景色を指定します。

4.7.2.1.5 Signal ウィジェット

Signal ウィジェットは、On 状態と Off 状態を色で示すシンプルなインジケータです。



データフィールド

Signal ウィジェットは、全ての数値データタイプを受け付けるシンク データフィールドを持ちます。このウィジェットの表示色は論理評価(真/偽)に基づきます。入力値が0よりも大きな数値であれば論理値は「真」であり、ウィジェットは On 状態を示す色で表示されます。

設定

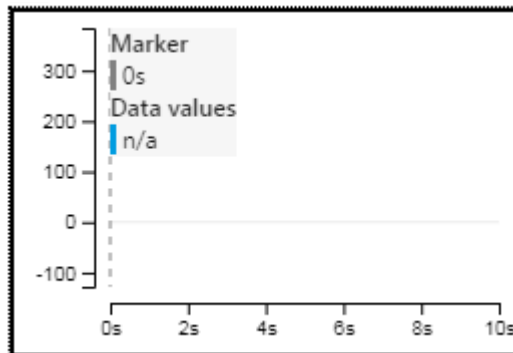
Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-24. On/Off グループ

パラメータ	タイプ	用途
On Color	色選択ダイアログ	ウィジェットの「On」状態の色を指定します。
Off Color	色選択ダイアログ	ウィジェットの「Off」状態の色を指定します。

4.7.2.1.6 Graph ウィジェット

Graph ウィジェットは、入力データストリームを 2 次元グラフにプロットします。マウス操作によるグラフの拡大とスクロールを受け付けるか全てのマウス操作を無視するか選択できます。



データフィールド

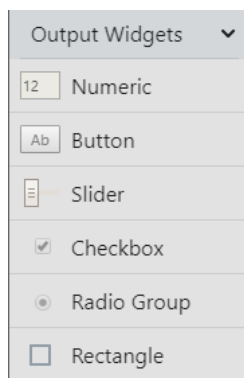
時系列プロットと同様に、データソースをプロットに追加します。

設定

4.3. 「時系列プロットにおけるデータ表示」を参照してください。

4.7.2.2 出力ウィジェット

図 4-56. 出力ウィジェットのリスト



出力ウィジェットは、接続されているデータシンクへデータ値を送信します。



4.7.2.2.1 Numeric ウィジェット

Numeric ウィジェットを使うと、ダッシュボードへ数値を入力できます。



データフィールド

Numeric ウィジェットは、1つのソースデータフィールドを持ちます。入力値が変更されるたびにパケットが値を送信します。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-25. Value グループ

パラメータ	タイプ	用途
Initial	数値	初期値を指定します。
Min	数値	入力の最小値を指定します。
Max.	数値	入力の最大値を指定します。

表 4-26. Colors グループ

パラメータ	タイプ	用途
Foreground	色選択ダイアログ	前景色を指定します。
Background	色選択ダイアログ	背景色を指定します。

表 4-27. Font グループ

パラメータ	タイプ	用途
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	フォントの太さ(bold 等)を選択します。
Font Style	ドロップダウン ボックス	フォントのスタイル(italics 等)を選択します。
Font Family	ドロップダウン ボックス	フォントファミリ(segment 等)を選択します。

表 4-28. Alignment グループ

パラメータ	タイプ	用途
Horizontal	ドロップダウン ボックス	表示内容の横方向の配置を選択します。

4.7.2.2.2 Button ウィジェット

Button ウィジェットは、ボタンが押されるたびに 1 つの値を送信します。ボタンは通常の押しボタンまたはトグルボタンとして設定できます。通常タイプのボタンには、その機能を表すテキストを表示できます。

トグルボタンとして設定した場合、テキストの代わりにボタンの状態に応じて「ON」または「OFF」が表示されます。「ON/OFF」テキストを他のテキストに置き換える事ができます。その場合、テキストパラメータは「On 状態のテキスト/Off 状態のテキスト」(2 つのテキストを「/」で区切る)として指定します。例えば「CLICKED/UNCLICKED」とした場合、On 時に「CLICKED」と表示され、Off 時に「UNCLICKED」と表示されません。



データフィールド

Button ウィジェットは、1 つのソースデータ フィールドを持ちます。ボタンが押されるたびにパケットが送信されます。通常タイプのボタンの場合、パケットの値は常に「0」です。トグルタイプのボタンの場合、状態が Off の時に値は「0」であり、On の時に値は「1」です。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-29. Button グループ

パラメータ	タイプ	用途
Label	テキスト	ラベルテキストを指定します。
Mode	ドロップダウン ボックス	ボタンのタイプ(Fixed または Toggle)を選択します。

.....続き		
パラメータ	タイプ	用途
Value	数値	モードが「Fixed」の時に送信される値を指定します。

表 4-30. Colors グループ

パラメータ	タイプ	用途
Foreground	色選択ダイアログ	前景色を指定します。
Background	色選択ダイアログ	背景色を指定します。

表 4-31. Font グループ

パラメータ	タイプ	用途
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	フォントの太さ(bold 等)を選択します。
Font Style	ドロップダウン ボックス	フォントのスタイル(italics 等)を選択します。
Font Family	ドロップダウン ボックス	フォントファミリ(segment 等)を選択します。

表 4-32. Alignment グループ

パラメータ	タイプ	用途
Horizontal	ドロップダウン ボックス	表示内容の横方向の配置を選択します。
Vertical	ドロップダウン ボックス	表示内容の縦方向の配置を選択します。
Line Wrap	ドロップダウン ボックス	表示内容の折り返し条件を選択します。

4.7.2.2.3 Slider ウィジェット

Slider ウィジェットは、マウスでマーカーを移動させる事ができるスライダです。マーカーの移動によってスライダの値を変更します。



データフィールド

Slider ウィジェットは、1つのソースデータ フィールドを持ちます。スライダの値が変更された時にパケットが値を送信します。

設定

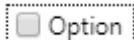
Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-33. Value グループ

パラメータ	タイプ	用途
Initial	数値	初期値を指定します。
Min	数値	入力の最小値を指定します。
Max.	数値	入力の最大値を指定します。

4.7.2.2.4 Checkbox ウィジェット

Checkbox ウィジェットは、状態(チェックマークの有無)が変更されるたびに1つの値を送信します。



データフィールド

Checkbox ウィジェットは、1 つのソースデータ フィールドを持ちます。ウィジェットの状態が変更されるたびに値が送信されます。チェックボックスにチェックマークが付けられた時に「1」が送信され、チェックマークが外された時に「0」が送信されます。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-34. Checkbox グループ

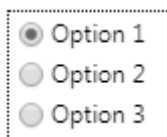
パラメータ	タイプ	用途
Label	テキスト	チェックボックスのオプション名を示すテキストを指定します。

表 4-35. Font グループ

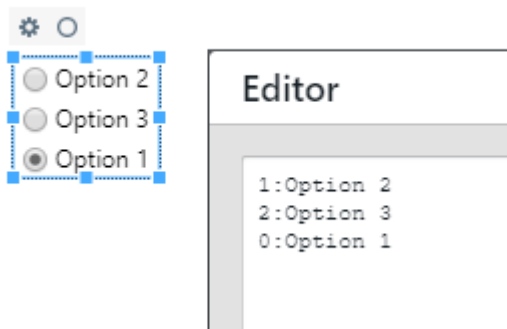
パラメータ	タイプ	用途
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	フォントの太さ(bold 等)を選択します。
Font Style	ドロップダウン ボックス	フォントのスタイル(italics 等)を選択します。
Font Family	ドロップダウン ボックス	フォントファミリ(segment 等)を選択します。

4.7.2.2.5 Radio Group ウィジェット

Radio Group ウィジェットは複数のラジオボタンで構成され、常に 1 つのオプションだけが選択可能です。初期状態では、最初のオプションが選択されます。



オプションのリストを変更するには、エディタを使ってオプション「Values」(表 4-36 参照)を変更します。



オプションを横方向に並べる事もできます。

データフィールド

Radio Group ウィジェットは、1 つのソースデータ フィールドを持ちます。ウィジェットの状態が変更されるたびに値が送信されます。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-36. Radio グループ

パラメータ	タイプ	用途
Values	エディタ	エディタを使ってオプションのリストを編集する事で、オプション名の指定とオプションの追加/削除/並び換えができます。
Horizontal	チェックボックス	オプションを横方向に並べる場合はチェックマークを付けます。

表 4-37. Font グループ

パラメータ	タイプ	用途
Font Size	数値	フォントサイズを指定します。
Font Weight	ドロップダウン ボックス	フォントの太さ(bold 等)を選択します。
Font Style	ドロップダウン ボックス	フォントのスタイル(italics 等)を選択します。
Font Family	ドロップダウン ボックス	フォントファミリ(segment 等)を選択します。

4.7.2.2.6 Rectangle ウィジェット

このウィジェットは、マウスでクリックされるたびに1つの値を送信します。



データフィールド

Rectangle ウィジェットは1つのソースデータ フィールドを持ちます。エレメントがマウスポインタによってクリックされるたびに1つの値が送信されます。

設定

Data Visualizer 内でパラメータの上にマウスカーソルを置くと説明が表示されます。

表 4-38. Rectangle グループ

パラメータ	タイプ	用途
Color	色選択ダイアログ	ウィジェットの色を指定します。

4.7.3 ダッシュボード ウィジェットの制御オプション

[Dashboard]ウィンドウ内でウィジェットを選択すると、右側のウィンドウ枠にそのウィジェットを設定するためのオプション(位置、サイズ、その他)が表示されます。選択されているウィジェットに応じて、表示されるオプションは異なります。

図 4-57. 右側ウィンドウ枠内の制御オプション - Progress Bar ウィジェットの例

Position ?	
Left:	<input type="text" value="127px"/>
Top:	<input type="text" value="157px"/>
Width:	<input type="text" value="Default"/>
Height:	<input type="text" value="Default"/>
Colors ?	
Foreground:	<input type="text" value="Default"/>
Background:	<input type="text" value="Default"/>
Value ?	
Initial:	<input type="text" value="0"/>
Min:	<input type="text" value="0"/>
Max:	<input type="text" value="255"/>

5. Variable Streamer

ほとんどの通信インターフェイスは、データを転送するためにバイトのストリームを使います。これは 8 ビット精度の単一データ値の転送には十分ですが、同じインターフェイスを介して複数の値を送信する必要がある場合、データを一定のプロトコルでパックする必要があります。MPLAB Data Visualizer は**データストリーム** プロトコルをサポートします。

データストリーム プロトコルは、軽負荷のフレーミング フォーマットを使って、複数の数値を単一インターフェイス向けにパックします。このプロトコルは受信データを処理する事しかできず、同期ストリームのみをサポートします(すなわち、全てのデータパケットは各データストリームから 1 つずつのサンプルを格納する必要があります)。**データストリーム デコーダ**情報は Data Visualizer ワークスペース内に保存されます。

Data Visualizer のデータストリーム モジュールは、入力される 1 つの生データストリームを複数のデータストリームに分割します。データストリーム フォーマットは、ユーザが提供する **Variable Streamer** によって指定されます。

5.1 変数データ型

Variable Streamer のセットアップとプロットにはウィザードを使います。このセットアップはワークスペース内に保存されます。Variable Streamer 内で定義された変数は、表 5-1 に記載したデータ型のいずれかである必要があります。

表 5-1.使用可能なデータ型

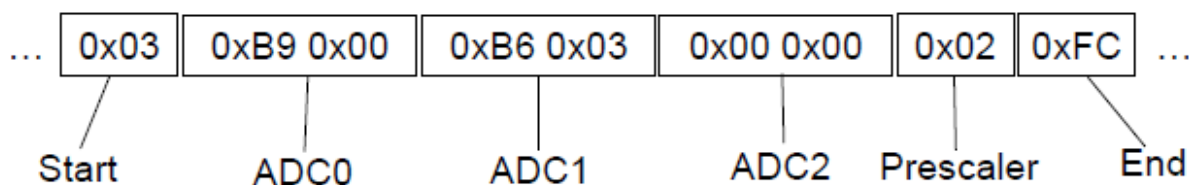
型	サイズ (バイト)
int8	1
int16	2
int32	4
uint8	1
uint16	2
uint32	4
float32	4
float64	8

5.2 ストリーム フォーマット

データ ストリーム フォーマットは、Variable Streamer 内で指定された変数と同じ順番で処理されます。全てのデータはリトルエンディアン形式の値として与えられる必要があります(最下位バイトから順番に送信)。加えて、変数データストリームの直前と直後に 1 バイトのラッパーを追加する必要があります。このラッパーは、データストリームに同期するためにインタープリタによって使われます。開始バイトのラッパーには任意の値が使えますが、終了バイトのラッパーは開始バイトの反転値である必要があります。開始バイトと終了バイトの値は設定内で定義されません。

図 5-1 に、生データ送信の例を示します。この例では ADC0 が 185、ADC1 が 950、ADC2 が 0、プリスケアラが 2 です。

図 5-1. データストリーマ

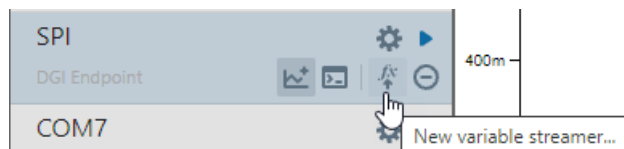


5.3 Variable Streamer のセットアップとプロット

Variable Streamer は、データストリームに格納された変数をユーザ アプリケーションからの出力として定義します。Variable Streamer は、データストリーム デコーダ インスタンスの設定内で使われます。このデコーダ インスタンスは、グラフ上のプロットとして表示可能なデータストリーム出力となります。

5.3.1 データソースの選択

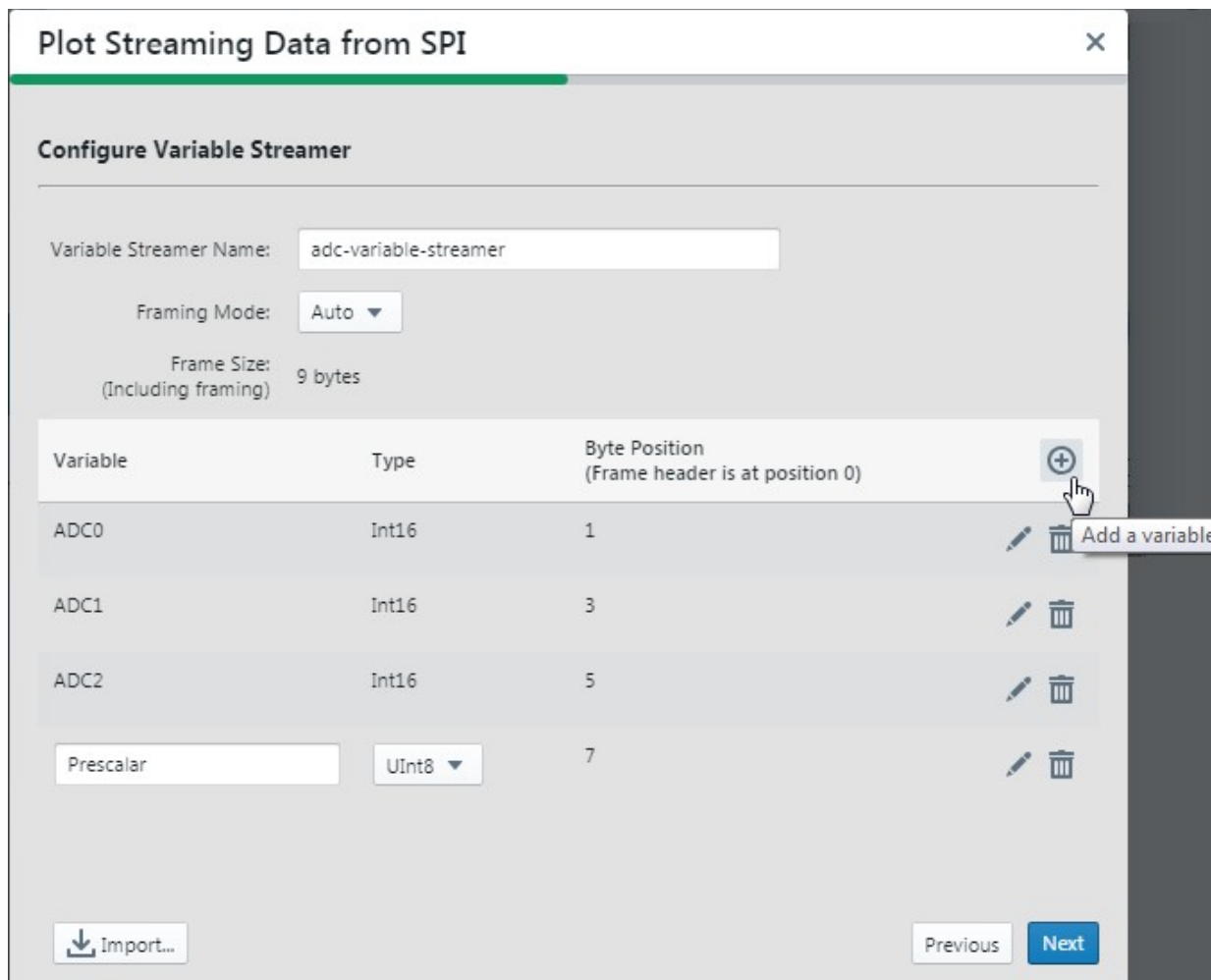
Variable Streamer を新規作成するには、左側の[Connections]タブ内で、利用可能なデータソースの[New Variable Streamer]アイコンをクリックします(下図参照)。



すると「Plot Streaming Data」ウィザードの[Configure Variable Streamer]ダイアログ(下図)が開きます。

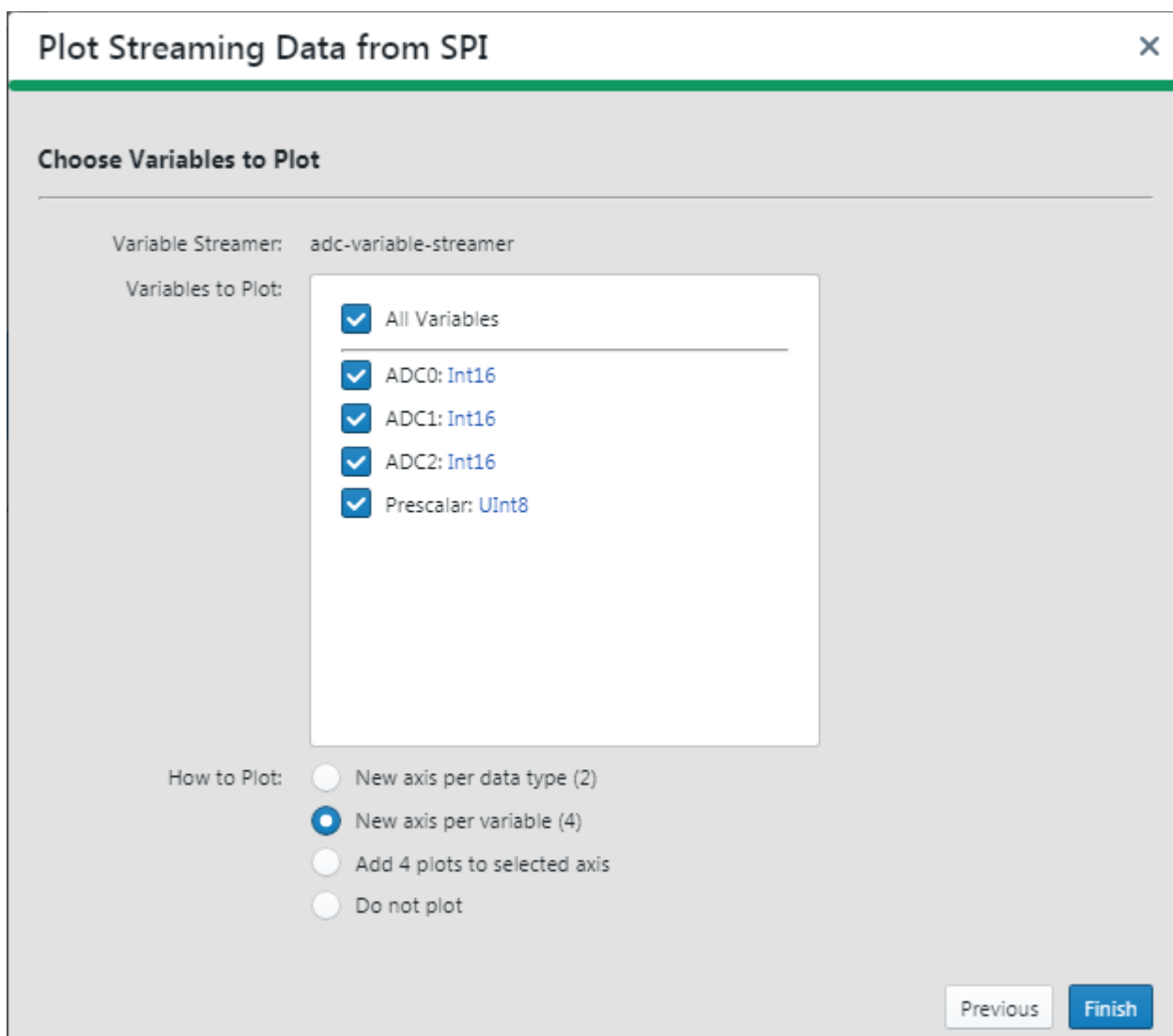
5.3.2 Variable Streamer を設定してプロットする

[Configure Variable Streamer] ダイアログを使って変数を設定します。各制御オプションの説明は次ページの表に記載しています。



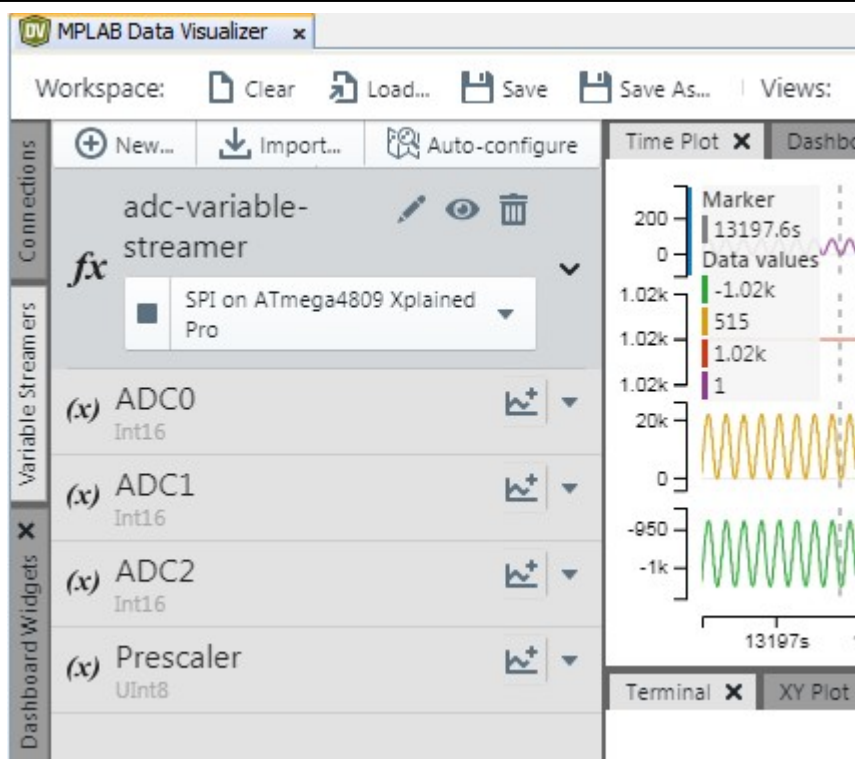
オプション	概要
Variable Streamer Name	その Variable Streamer に分かりやすい名前を付けます。
Framing Mode	「Auto」を選択すると、データストリーマ プロトコルの開始バイトは 0x5F に設定され、終了バイトは 0xA0 に設定されます。他の開始/終了バイトを使う場合、「one's complement」を選択できます。
Framing Size	5.2. 「ストリーム フォーマット」を参照してください。
Variable	アプリケーション コードからの変数の名前を入力します。
Type	5.1. 「変数データ型」を参照してください。
Byte Position	5.2. 「ストリーム フォーマット」を参照してください。

[Next]をクリックして[Choose Variables to Plot] ダイアログへ進みます。プロット方法(How to Plot)を選択してから[Finish]をクリックします。



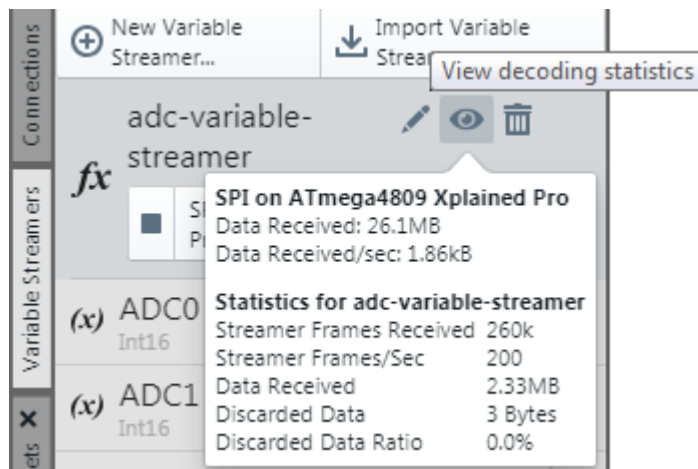
5.3.3 Variable Streamer 出力を表示する

左側の[Variable Streamers]タブに新しい Variable Streamer が表示されます(次ページの図参照)。これらの設定は、セッションの最後でワークスペースを保存する事により保存できます。保存した設定は、次のセッションでワークスペースに読み込む事ができます。



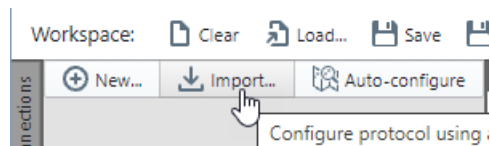
5.3.4 統計値を表示する

目アイコン[View decoding statistics]をクリックすると、ストリーミング データに関する統計値が表示されます(下図参照)。



5.4 Variable Streamer のインポート

Variable Streamer を新規作成する代わりに、既存のデータストリーム プロトコル定義ファイル(.ds または.txt)をインポートする事もできます。

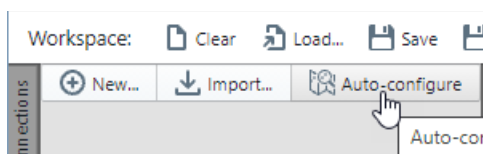


インポート後は、新規ストリーマと同様に設定して使うか、5.5. 「Auto-Configure Variable Streamer」を使う事ができます。後でシリアルソースが Variable Streamer 上で入力として選択された場合、Variable Streamer は auto-configure ID に一致する .db および .sc ファイルを探します。一致するファイルが見つかった場合、auto-configure 処理が完了してダッシュボードが表示されます。Variable Streamer の名前は適切に変更されます。

5.5 Auto-Configure Variable Streamer

Auto-Configure ストリーミング フォーマットをサポートする MPLAB X IDE プロジェクトを開きます(5.5.1. 「Auto-Configure の例とフォーマット」参照)。ターゲットにアプリケーションをプログラミングします。

MPLAB Data Visualizer を開いて[Variable Streamer]タブをクリックします。次に[Auto-Configure]をクリックして Auto-Configure Variable Streamer を作成します(下図参照)。ソースを選択してから、シリアルポートの baud レートをプロジェクトの設定に一致するよう設定します(baud レートの自動検出は未サポートです)。



次に、Auto-configure Variable Streamer 向けにシリアルソースを選択し、ターゲット上のシリアルポートからストリーミングを開始します。

Data Visualizer は、入力として選択されたシリアルデータ内で Auto-configure フレームを探します。Auto-configure フレームは、ディスク上のファイルの検索用に使う config ID を格納しています。システムは、シリアル ストリーム内の config ID に一致する 3 つのファイル(<config id>.ds、<config id>.sc、<config id>.db)をディスク上で検索して読み込みます。

検索パス

MPLAB X プラグインは、hot プロジェクトまたは main プロジェクト(プロジェクト コンテキスト)に従います。プロジェクト コンテキストが存在しない(例: プロジェクトが何も開かれていない)場合、「MPLABXProjects」ディレクトリが既定値の検索パスとなります。この設定は上書き可能ですが、現在の Aauto-configure エントリ向けにのみ保持されます。新しい Auto-configure エントリは、プロジェクト コンテキストに従う既定値設定に戻ります。

スタンドアロンの Data Visualizer の場合、既定値検索パスはユーザのホーム ディレクトリとなります。異なるディレクトリを選択した場合、Data Visualizer はそのディレクトリを新しい Auto-configure エントリ向けにも使います。この設定は、再起動しても保持されます。

ID に一致したファイルを読み込む

- Auto-configure エントリは、.ds ファイルからのフィールドを使って Variable Streamer へと変換されます。
- ユーザは Auto-configure の設定情報を見る事ができ、この情報を使ってダッシュボードの読み込みに使われるファイルシステム位置を知る事ができます。
- プロジェクト コンテキストが存在する場合、新しいストリーマにはプロジェクトと同じ名前が付けられます。存在しない場合は親ディレクトリと同じ名前が付けられます。
- ダッシュボードの内容が置き換えられ、[Dashboard] タブが開きます。
- Data Visualizer 内のダッシュボード以外のウィンドウとダッシュボード プロパティ サイドバーは読み込まれ*、ダッシュボードは Edit モードに設定されます。

* Auto-configure オプション内の「Maximize dashboard」のチェックマークを外していない場合

ファイル読み込み時のエラー

ファイルの読み込みに失敗した場合またはファイルが存在しなかった場合、エラーが通知されます。この場合、異なるプロジェクトを選択するか異なるパスを指定する等により、検索パスの設定を調整します。

```
Could not find 3 files named 03EB000000000000AA5501[.db, .sc, and .ds] in /Users/
User/example-fw-projects/ATtiny817-Xpro-touch-project.X/build.
Try changing the search path.
```

5.5.1 Auto-Configure の例とフォーマット

MPLAB Data Visualizer の Auto Configure は、Atmel Data Visualizer の Auto-Configuration と似ています。

Auto-configure を使うサンプル アプリケーションと Auto-Configure ファイルおよびフォーマットについては、[『Atmel Data Visualizer User's Guide』](#) (DS40001903) 内の「Auto-Configuration Example」、「Auto-Configuration Format」、「Signal Connections File Format」を参照してください。

6. DVRT プロトコル

DVRT (Data Visualizer Run Time)は、小型マイクロコントローラに適したシンプルな双方向プロトコルです。DVRT はターゲット アプリケーションのグローバル変数(メモリ位置)のストリーミングと読み書きをサポートします。

DVRT プロトコルにより、ストリーミング変数またはメモリ位置の選択とストリーミング速度を、実行時に動的に変更できます。これとは異なり、Data Streamer プロトコルでは、ストリーミングする変数をコンパイル時に定義します。

DVRT プロトコル オプションの詳細は、6.8. 「DVRT プロトコルのオプションとステータス情報」を参照してください。最新の DVRT デバイス サポートについては、8. 「リリースノートの閲覧方法」を参照してください。

6.1 プロジェクトに DVRT サポートを追加する

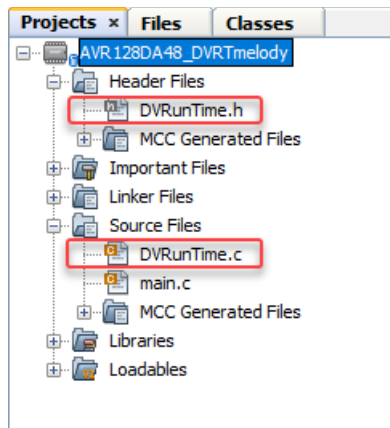
ターゲット内の DVRT 実装はホスト コンピュータとの通信用に UART を使い、コマンドとストリームデータを正しいタイミングで処理するためにタイマを利用したコールバックを使います。従って、プロジェクト デバイスはこれらのリソースを備えている必要があります。

DVRT プロジェクトの例(zip ファイル) は、MPLAB® Data Visualizer ウェブページ内の「Downloads」の下で見つかります。以下の手順により、プロジェクトを DVRT 向けにセットアップします。

1. DVRT 機能を追加するプロジェクトを新規作成するか、既存プロジェクトを開きます。
Note: 現時点では、任意の非ライブラリ プロジェクトが DVRT 向けに使えます。
2. プロジェクトにファイル(DVRunTime.c* と DVRunTime.h*)を追加します(図 6-1 参照)。
3. デバッグ用にプロジェクトをビルドして実行します(後で必要になる ELF ファイルを生成するため)。
4. DVRT ドライバを初期化するため、アプリケーションの開始時に DVRT_Initialize()*を呼び出します。
5. DVRT コマンドを処理するため、main ループ内で DVRT_Process()*を周期的に(例: 1 ms 周期で)呼び出します。
6. DVRunTime.h の最後で、ターゲット上の UART 実装に従って UART の define を更新します。

* 内容は例を参照してください。

図 6-1. DVRT サンプル プロジェクト



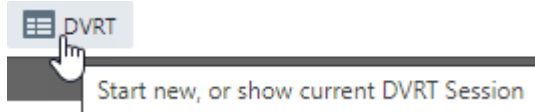
6.2 DVRT プロジェクト ELF ファイルを Data Visualizer に読み込む

DVRT プロトコルを使うには、DVRT プロジェクト ELF ファイルを MPLAB Data Visualizer に読み込む必要があります。ELF ファイルを読み込んだ後は、グローバル変数(または構造体および配列エレメント)を選択してそれらをストリーミング データテーブルに追加し、データテーブルを読み書きする事が可能になります。Data Visualizer は、ストリーミングする変数の位置とサイズをターゲットへ送信します。

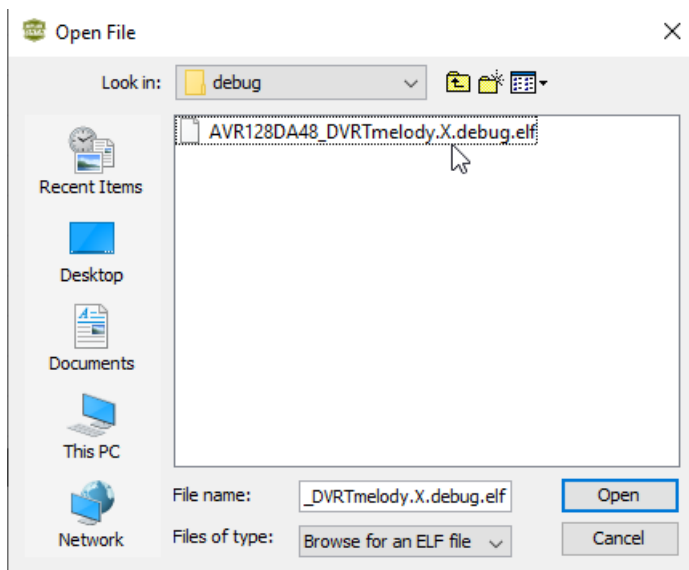
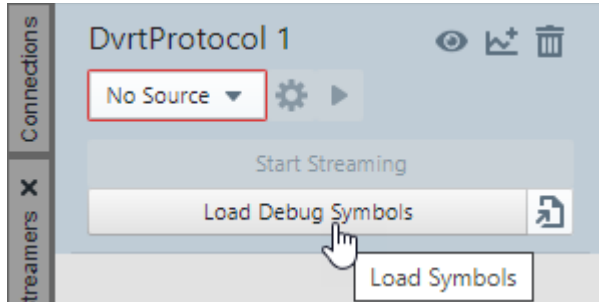
6.2.1 ELF ファイルをスタンドアロン Data Visualizer に手動で読み込む


スタンドアロン Data Visualizer を使っている場合の手順は以下の通りです。

1. MPLAB X IDE 内で、DVRT カーネルコード(DVRunTime.c および DVRunTime.h ファイル)を含むプロジェクトを開きます。
2. DVRT 用の表示を開きます。



3. [Load Debug Symbols]をクリックし、プロジェクト ELF ファイルを指定して読み込みます。



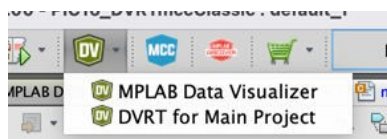
4. 異なる ELF ファイルを読み込む場合、 をクリックします。

この方法で ELF ファイルを読み込んだ場合、変更検出用に ELF ファイルのパスが監視されます。リビルド時に、Data Visualizer はこの ELF ファイルをリロードするためのオプションを提供します。ELF ファイルをターゲットにプログラミングする際にシンボルがリロードされ、完了時にメッセージが表示されます。

6.2.2 ELF ファイルをプラグイン Data Visualizer に自動的に読み込む

MPLAB X IDE Data Visualizer プラグインを使っている場合の手順は以下の通りです。

1. MPLAB X IDE 内で、DVRT カーネルコード(DVRunTime.c および DVRunTime.h ファイル)を含むプロジェクトを開きます。
2. ツールバー内の[MPLAB Data Visualizer]アイコンをクリックし、ドロップダウンメニューから「DVRT for Project Name」を選択します。



3. この方法でシンボルを読み込む事ができない場合、6.2.1. 「ELF ファイルをスタンドアロン Data Visualizer に手動で読み込む」を参照してください。

加えて、以下の状況が発生する可能性もあります。

エラー要因	結果
ELF ファイルが存在しない	プロジェクトは ELF ファイルを生成するためにビルドを実行します。
量産用 ELF を使って DVRT セッションを作成してデバッグセッションを開始した	デバッグ ELF が DVRT セッションの ELF ファイルを置き換えます。デバッグセッション開始時に Data Visualizer 内の [Symbol Loaded] ボタンにマウスカーソルを重ねる事でパスを確認できます。
デバッグ ELF が現在の DVRT セッション向けに使われている時に、量産用「run」またはデバッグ ELF ファイルを置き換えるプログラミング動作をユーザが開始した	プログラミング/実行動作が完了すると、量産用 ELF が DVRT セッションの ELF ファイルを置き換えます。
DVRT セッションが Data Visualizer 内で既に定義されていた	既存セッションを本当に置き換えるかどうか確認するためのダイアログが表示されます。キャンセルすると Data Visualizer は変更されません。置き換えを許可すると、既存セッションは新しいプロジェクトの ELF パスを使って置き換えられます。

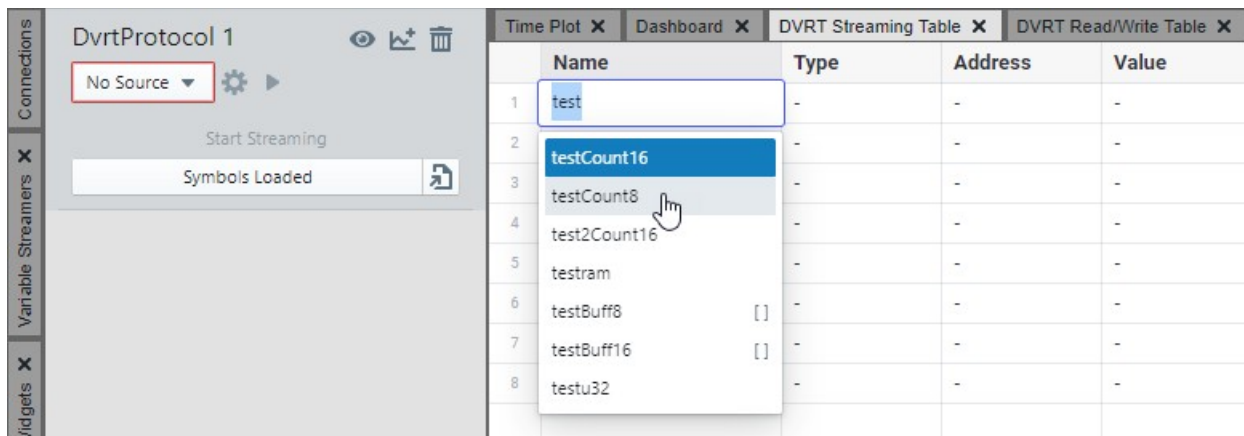
6.3 シンボルの選択

シンボルを読み込んだ後は、それらを選択して [DVRT Streaming Table] タブに追加できます。[Name] 列内の 1 つのセルをクリックしてからタイプ入力を始めると、シンボルの候補がドロップダウン リストに表示されます。

6.3.1 変数の選択

必要な変数をクリックして選択します。

図 6-2. 変数の選択



6.3.2 構造体の選択

構造体エレメントを選択する場合、最初に構造体の名前をクリックし、リストから必要なメンバーを選択します。

図 6-3. 構造体の選択

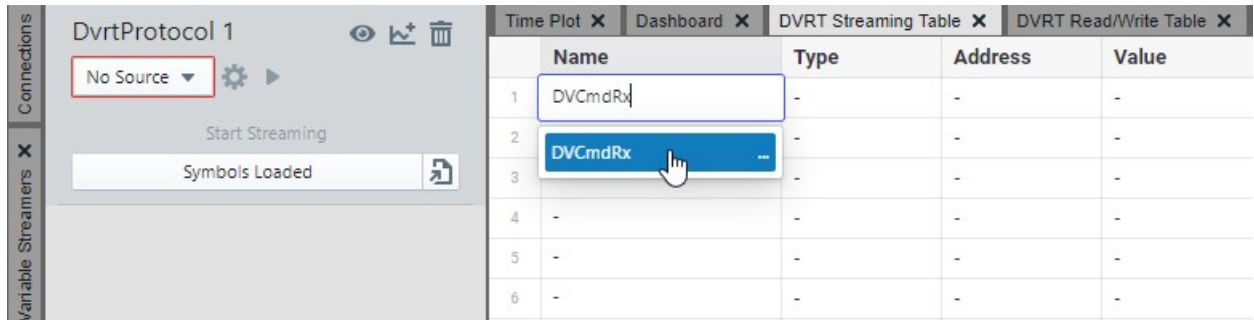
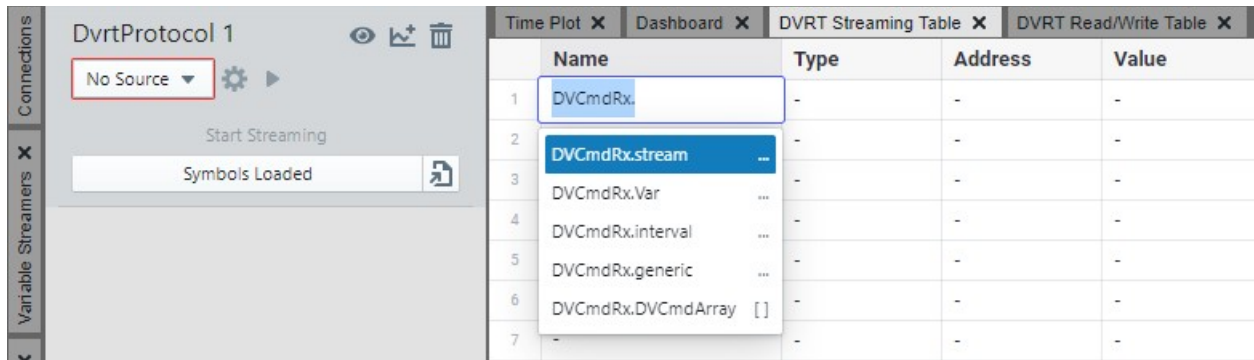


図 6-4. 構造体エレメントの選択



6.3.3 配列エレメントの選択

配列エレメントを選択する場合、最初に配列の名前をクリックし、配列エレメントのインデックスを手動で入力します(既定値は 0)。

図 6-5. 配列の選択

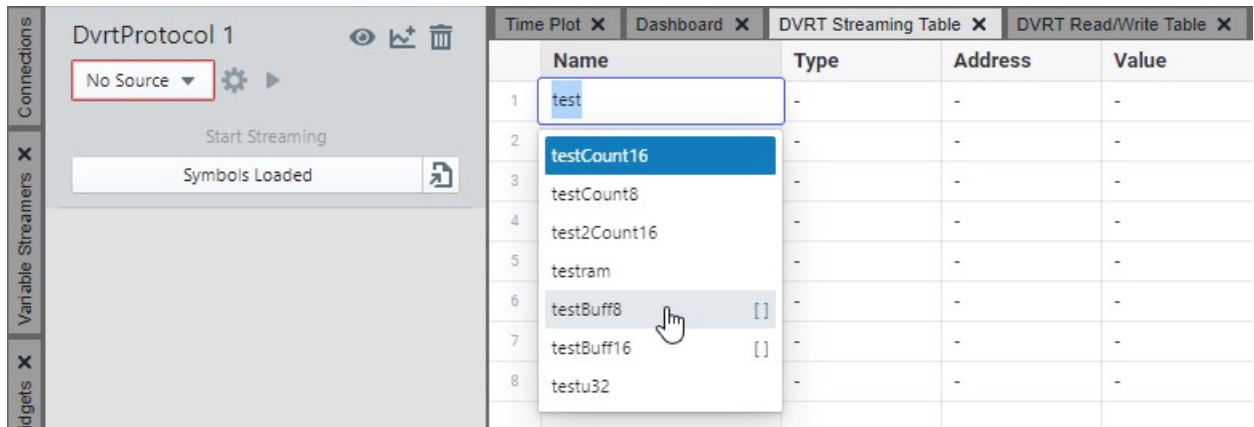
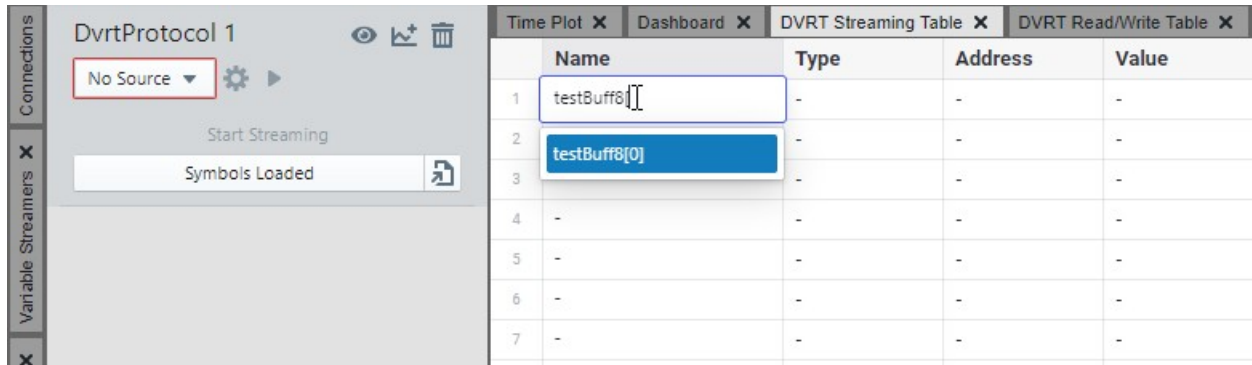


図 6-6. 配列要素の選択

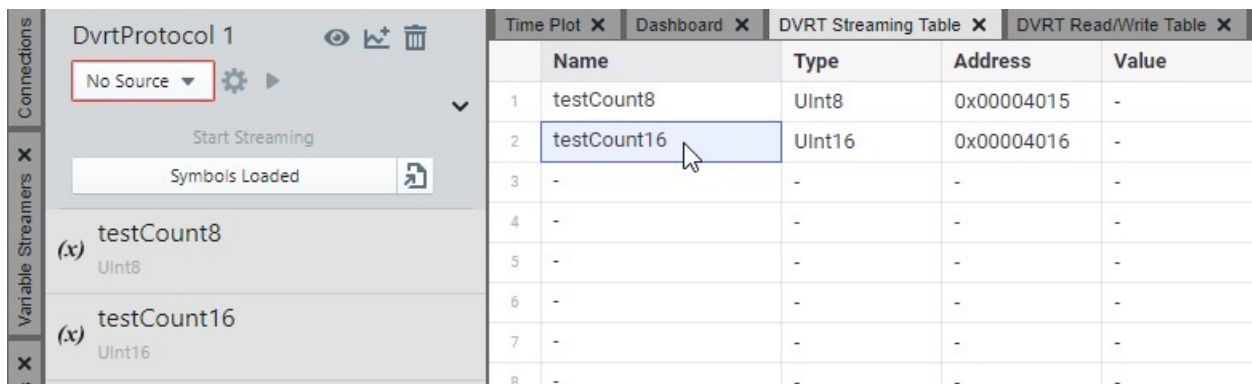


6.3.4 選択したシンボルを表示する

シンボルを選択すると、テーブルの他の列(Type、Address 等)に関連する情報が表示されます。選択操作を繰り返して必要なシンボルを追加します。

Note: 空白行の下の方にシンボルを入力した場合、入力された内容は上の行へ移動します(空白行は詰められます)。行を削除するには、その行内のセルをクリックしてから<Delete>キーを押します。

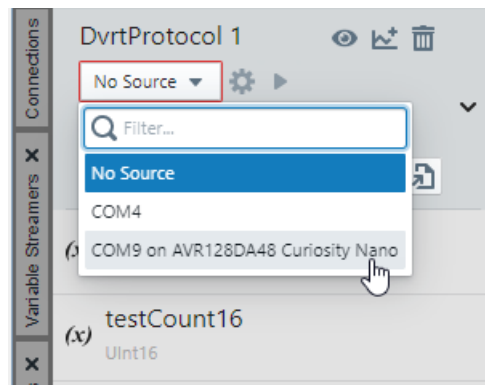
図 6-7. 追加されたシンボル



6.4 変数値のストリーミング

ストリーミングするソースを選択します。

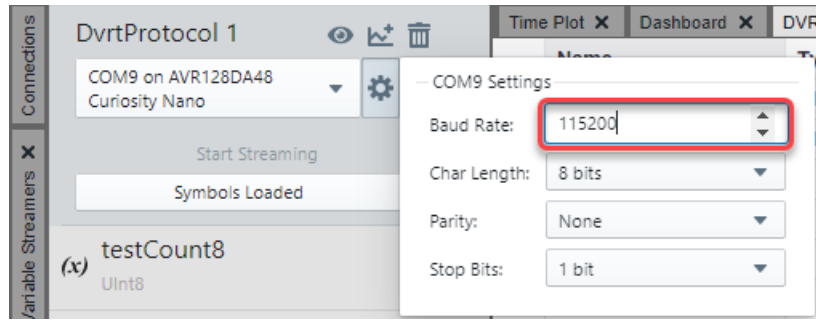
図 6-8. 接続ソースの選択



ギアアイコンをクリックする事により接続を設定します。DVRT の場合、図 6-9 に示す値を使います。

Note: 全ての DVR 実装に対して 115200 の baud レートを使います。

図 6-9. DVRT 向けの接続設定




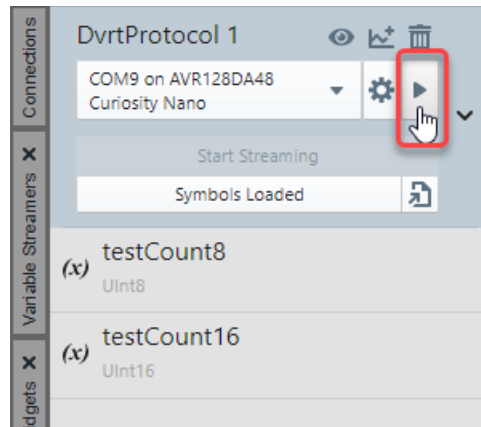
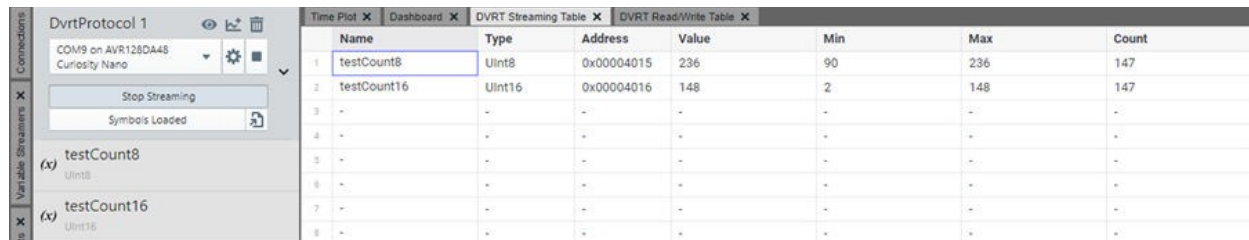
[Play]ボタン  をクリックしてストリーミングを開始します。ストリーミングが始まるとこのボタンは[Stop]ボタンに変化し、これをクリックする事でストリーミングを停止できます。

図 6-10. ストリーミングの開始



実行中にテーブル内の変数値の変化を観察します。

図 6-11. Int 値のストリーミング



右側ウィンドウ枠内でオプションを選択する事により、テーブル内の Int 値を 16 進数で表す事ができます(図 6-12、図 6-13 参照)

図 6-12. 16 進フォーマット

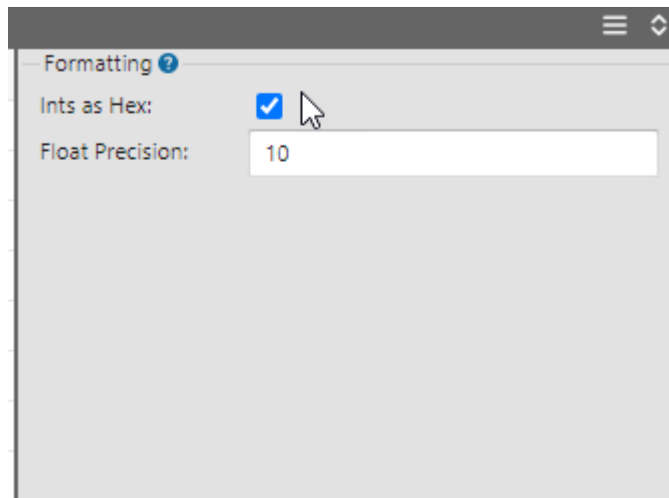


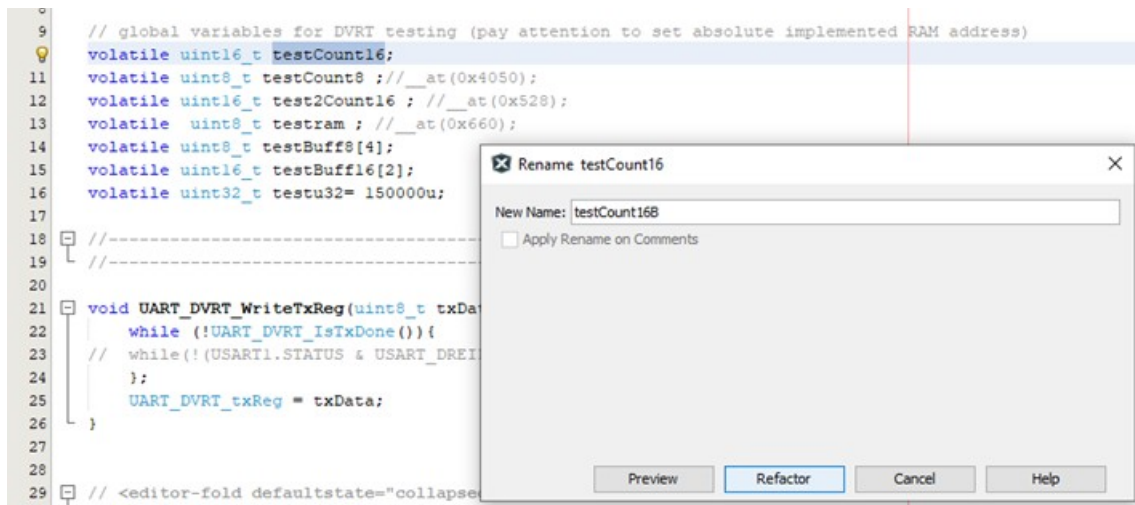
図 6-13. 16 進 Int 値のストリーミング

Name	Type	Address	Value	Min	Max	Count
testCount8	UInt8	0x00004015	0xda	0x00	0xff	0x00000381
testCount16	UInt16	0x00004016	0x0382	0x0002	0x0382	0x00000381
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-

6.5 変数名を変更してストリーミングを再開する

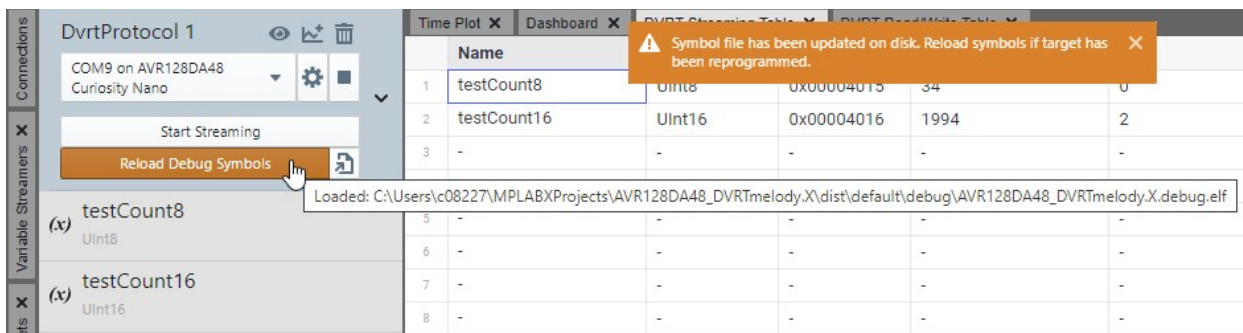
DVRunTime.c ファイルを開いて変数名 testCount16 を testCount16B に変更し(メニュー *Refactor>Rename* を使用)、プロジェクトをリビルドして新しい ELF ファイルを生成します。

図 6-14. 変数名を変更する



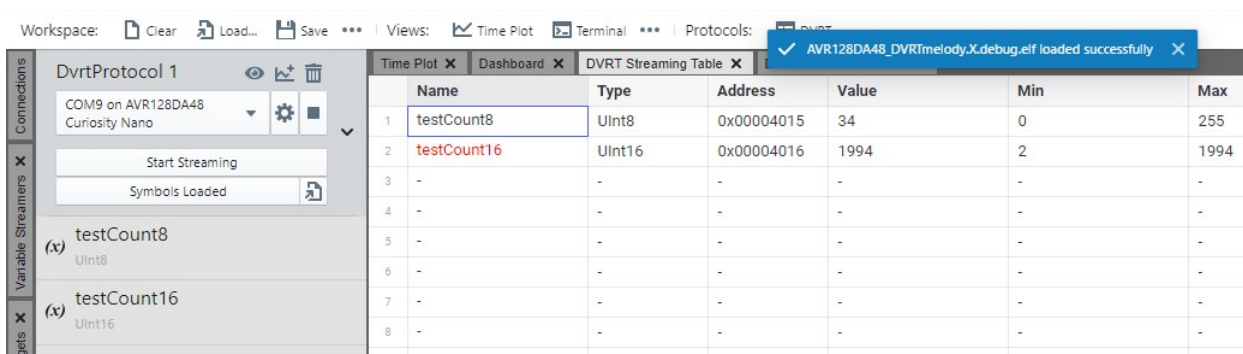
Data Visualizerに戻ります。DVRTセッションはELFファイルと同期しているため、ELFファイルが更新された事とシンボルをリロードする必要がある事を示すメッセージが表示されます。

図 6-15. デバッグシンボルをリロードする



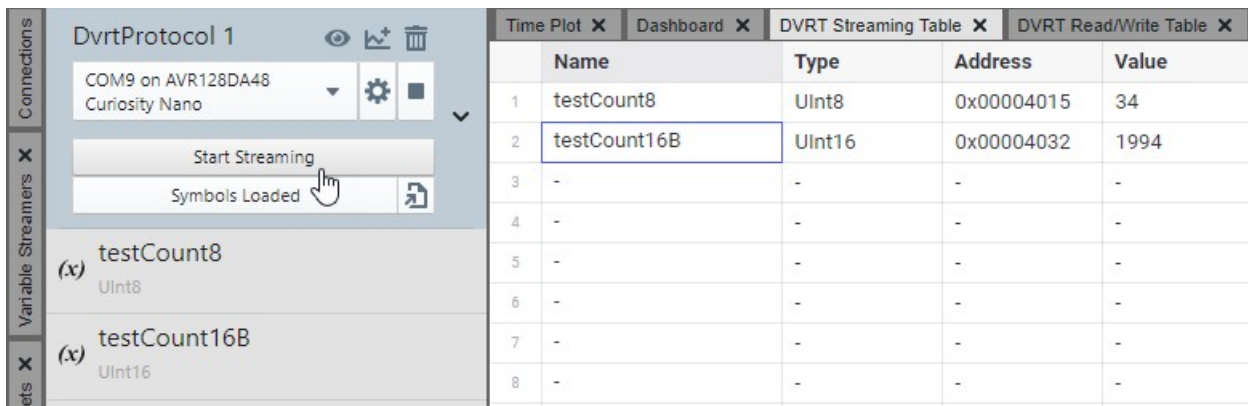
[Reload Debug Symbols] をクリックします。新しい ELF ファイルが読み込まれると、以前の変数名は赤字で表示されます(その名前の変数はもう見つからないため)。

図 6-16. 見つからなくなった変数名が赤字で示される



変数名を新しい名前に変更し、その変数の値がストリーミングされて表示される事を確認します。

図 6-17. 名前を変更した変数を追加する

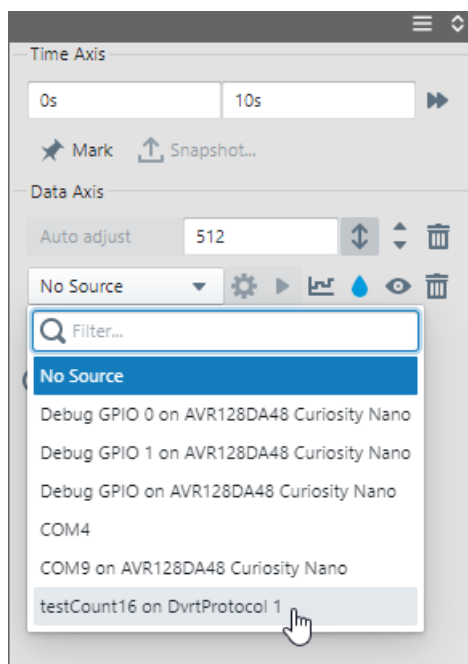


6.6 ストリーミング データの表示

[DVRT Streaming Table] タブに表示される変数データは、他のストリーミング データと同様に時系列プロット、ターミナル、ダッシュボード、XY プロット上で表示できます。例えば、時系列プロットにデータを表示する場合、変数をソースとして選択します(図 6-18 参照)。

詳細は 4. 「データの表示」に記載しています。

図 6-18. DVRT ストリーミング変数を時系列プロットに表示する



6.7 ファームウェア内の変数の読み書き

[DVRT Read/Write Table] タブを使うと、ストリーミングせずに個々の変数を読み書きできます。

- フィールドの編集方法は[DVRT Streaming Table] タブと同じです。
- 「Actions」列のセルをクリックすると、ターゲットから変数値が読み出されます。読み出された値は、右隣の「Value」列に表示されます。
- 「Value」列の値をダブルクリックして値を変更した後に<Return>キーを押すか、他の場所をクリックすると、その値がターゲットに書き込まれます。

図 6-19. ターゲットから値を読み出す

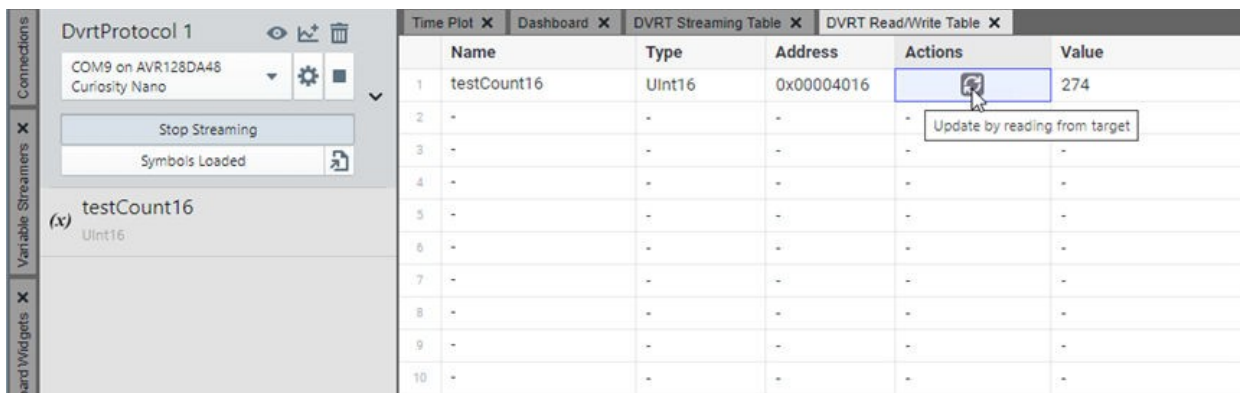
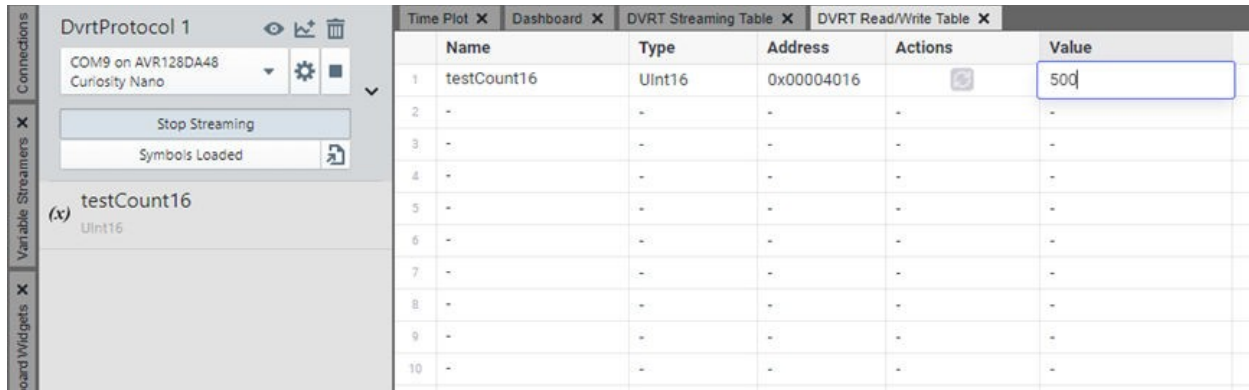


図 6-20. ターゲットに値を書き込む



6.8 DVRT プロトコルのオプションとステータス情報

左側の[DVRT Protocol]タブ内の何も無い領域(例: 下図の赤丸位置)をクリックすると、その下に DVRT プロトコルのオプションとステータスが表示され、それらを変更する事ができます(図 6-21 参照)。入力フィールドにマウスマウスカーソルを重ねると、それらの定義が表示されます(図 6-22 参照)。

図 6-21. オプションおよびステータス情報

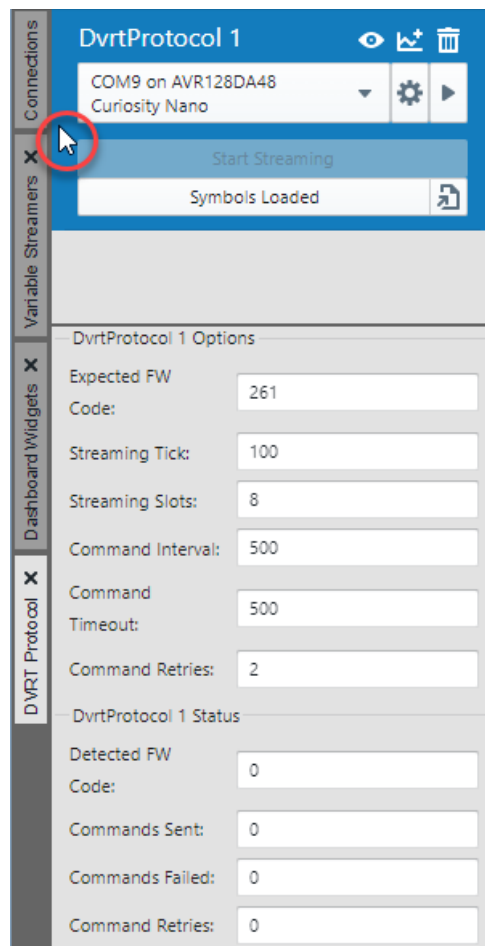
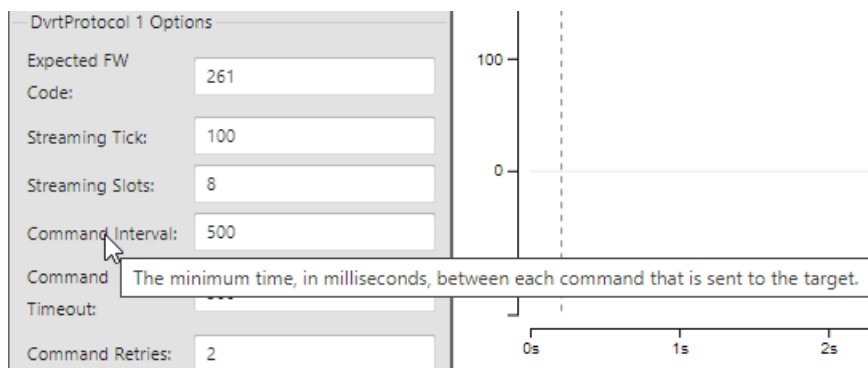


図 6-22. フィールドにマウスカーソルを重ねる



7. トラブルシューティング

以下では、MPLAB Data Visualizer のコンポーネント別にトラブルシューティングに関するヒントを記載しています。最後に、全般的な「注意事項」も記載しています。

7.1 データストリーミング

DGI を介する SPI

正しいデータ波形が表示されない場合、またはデータストリーム デコーダによってデコードできない場合、「Force synchronization on CS」を有効にして再試行します。Xplained Pro 開発キット上でデバッグ セッションを開始し、同キットから SPI 経由でストリーミングを行う場合、特に同期の問題がよく発生します。

プロット用のリソース

複数のデータストリーム(例: 電圧、電流、電力)をプロットする際にホスト コンピュータで問題が生じる場合は以下の対策を試みます。

- 解析を行う時はデータのストリーミングを停止する
- ライブデータのスクロール中は時系列プロットの表示を縮小(Zoom out)する
- 右側ウィンドウ枠内で選択するプロットソースの数を減らす
- 1つまたは複数のプロットを削除する

上記の対策はシステムリソースへの要求を削減します。

7.2 データストリーム デコーダ

データが入力されない

データソースが接続されてデータを送信している事を確認します。これは、生データストリームをプロットするか、ターミナルにデータを表示する事で容易に確認できます。プロットまたはターミナルに値が表示されない場合、データは受信されていないと考えられます。

デコーダ不一致

Variable Streamer の不一致エラーが発生する場合、Variable Streamer 内の変数と受信データが完全に一致しているか確認します。デコーダは、フレーム開始バイト(SoF)の後に Variable Streamer 内で定義されたフィールドに一致するバイトシーケンスが続き、最後に SoF の 1 の補数を格納したフレーム終了バイト(EoF)が送信される事を期待します。データパケットのデコードを妨げる原因として以下が考えられます。

- SoF バイトまたは EoF バイトが存在しない
- 送信されるデータパケット内で必要なフィールドが欠落している
- 変数のサイズが Variable Streamer 内の定義と異なる

各変数はデータ型に適合するバイト数を持つ必要があります。例えば、Int8 は 1 バイトであり、Int16 は 2 バイトです。

ターゲットから送信されたデータフレームと Data Visualizer 内の Variable Streamer の間で不一致があっても、不一致エラーがクリアされてしまう場合があります。例えば全体のバイト数が一致していても各フィールド長が定義と一致していない場合に発生する可能性があります。各フィールドのサイズが正確に一致している必要があります。

フレームの構造:

[Start of Frame character][Data Field 1 * data type size in bytes][Data Field 2 * data type size in bytes]...[Data Field n * data type size in bytes][End of Frame character]

例:

2 個の Int16 フィールド(x と y)が存在する場合のプロトコル バイトシーケンス: [SoF][x1][x0][y1][y0][EoF]
このプロトコルに一致するデータストリームの例: [0x3][0x34][0x12][0x78][0x56][0xFC]

この場合、デコードされた x の値は 0x1234、y の値は 0x5678 です。

7.3 注意事項

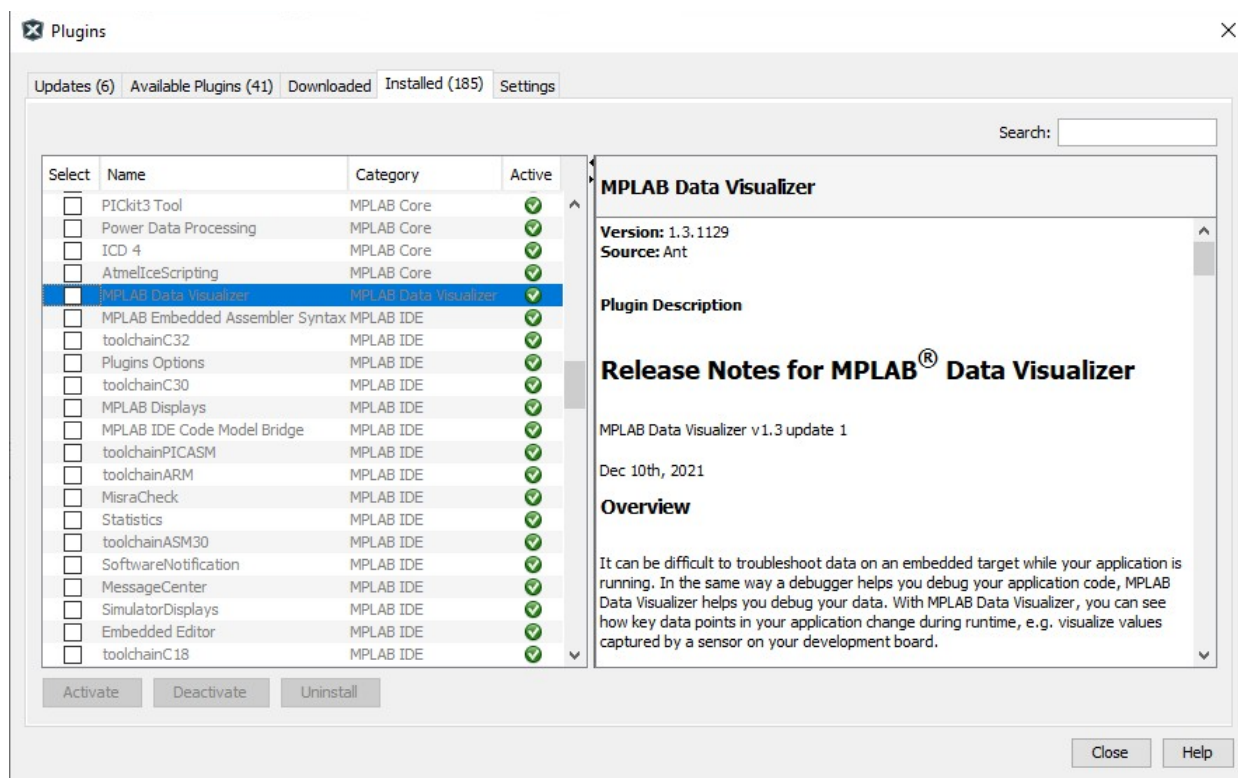
- Xplained Pro または Curiosity Nano キットから DGI を介してタイムスタンプ付きデータをストリーミングする場合、データレートが高過ぎるとバッファ オーバーフローが生じて動作安定性が悪化する恐れがあります。これは Debug GPIO と Power Data にも当てはまります。Data Visualizer のユーザ インターフェイスは、キットまたはツールがバッファ オーバーフロー状態を示した時に警告を表示します。バッファ オーバーフローが発生する恐れのあるサンプリング レートは、ホスト コンピュータとシステム負荷に応じて異なります。GPIO の場合、許容サンプリングレートはシステムに負荷がかかった状態で数 kHz レベルまで低下します。その他のインターフェイスの場合、許容サンプリングレートは GPIO の場合よりも一般的に高くなります。
- データのストリーミングにおいて期待通りの結果を得るには、ターゲット システムを適正に設定する事が重要です。ターゲット開発ボードまたはツールのユーザガイドを注意深く読む事が大切です。それらには重要な情報が記載されています。
- スタンドアロンの MPLAB Data Visualizer を起動した時に、開発キット上で実行中であったデバッグセッションが中断する可能性があります。MPLAB X 内で DGI 対応キットに対してデバッグ セッションが進行中である場合、スタンドアロン MPLAB Data Visualizer の起動時にデバッグ セッションが中断し、そのキットの接続が Data Visualizer に表示されなくなる可能性があります。この場合、MPLAB X を終了し、キットを一度切り離してから再度接続する事により対応します。デバッグとデータ表示を同時に行う場合、MPLAB X プラグイン バージョンの Data Visualizer (Window>Debugging>Data Visualizer) を使ってください。

8. リリースノートの閲覧方法

MPLAB Data Visualizer に関するリリースノートは、以下の方法で閲覧できます。

- プラグインの場合: [Tools>Plugins>Installed>MPLAB Data Visualizer](#)
- スタンドアロン プログラムの場合: Help メニューから

図 8-1. プラグイン Data Visualizer のリリースノート



9. データプロット用サンプルコード

ATmega4809 Xplained Pro プロジェクト向けのヘッダおよびソース C 言語コードのサンプルを以下に記載します。

Note: PDF ページからコードをコピーすると、ページのフッタがコードに含まれてしまう可能性があります。本書の WebHelp バージョンからコピーする事を推奨します。

9.1 C ヘッダコード

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
```

```
#ifndef cplusplus
extern "C" {
#endif

#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 */
```

9.2 C ソースコード

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,0x3
d,
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();
    }
}
```

10. 改訂履歴

本書の各リビジョンでの改訂内容を以下に記載します。

10.1 リビジョン E (2023 年 1 月)

- 2.2.1.1 ツールバーボタン: 制御構造を更新しました。
- 2.2.1.2 通知とメッセージ: 新しい項目を追加しました。
- 4. データ表示: 「Start visualizing data」ダイアログに関する情報を追加しました。
- 4.2.1 [Terminal]ウィンドウ: ターゲットへの入力用に使う場合にこのウィンドウに適用される更新に関する情報を追加しました。
- 4.3.2 [Connections]タブ: 時系列プロットのストリーミングを停止してもウィジェット プロット向けのストリーミングは停止しない事を記載した Note を追加しました。
- 4.3.5.3 機械学習の例: オンライン Developer Help 内の機械学習(ML)プラグインへの参照を追加しました。
- 4.6.3.1 MPLAB Touch プラグインの例: XY プロットに関連する内容を充実させました。オンライン Developer Help 内のタッチプラグインへの参照を追加しました。
- 6. DVRT Protocol: 8/16 ビット ターゲット向けの実行時設定可能 Data Visualizer Run Time (DVRT)プロトコルに関する新しい章を追加しました。

10.2 リビジョン D (2022 年 1 月)

- MPLAB Data Visualizer プラグインを使って SAM および AVR デバイスで PC サンプリングによる DGI デバッグ プロファイリングが利用可能となりました。
- Power プロットにおけるノイズに関する説明を追加しました。

10.3 リビジョン C (2021 年 11 月)

- ターゲットとの間で双方向にデータ ストリーミングが可能なウィジェットを備えたダッシュボードを追加しました。
- Atmel Data Visualizer のダッシュボード ファイルのインポートをサポートしました(Touch サンプル プロジェクトに関連)。
- Atmel Data Visualizer の Auto configuration をサポートしました。

10.4 リビジョン B (2021 年 5 月)

- Power Monitor のサポート: DGI 有効時の Power 接続と、これに関連する 3 つのソース(電流、電圧、電力)のプロットが利用可能になりました。
- 時系列プロットにおいて、マウスホバー メニューから Power Analysis 機能が利用可能になりました。
- プロットのスナップショットを CSV または JSON ファイルに保存する事が可能になりました。
- プラグイン サポートが利用可能になりました(MPLAB X IDE またはスタンドアロン Data Visualizer アプリケーションからメニュー *Tools>Plugins* を選択)。
- [Graph]タブを[Time Plot]タブに改名しました。
- 機械学習をサポートするために[XY Plot]タブを追加しました。
- 時系列プロット、XY プロット、ターミナル、プラグインのいずれかをアクティブ ウィンドウとして選択するための[Views]メニューアイコンを追加しました。

10.5 リビジョン A (2020 年 6 月)

本書の初版です(PDF 版と WebHelp 版.)

初期リリース(2019 年 12 月)

WebHelp として初期リリースしました。

Microchip 社ウェブサイト

Microchip 社はウェブサイト(www.microchip.com)を通してオンライン サポートを提供しています。当ウェブサイトでは、お客様に役立つ情報やファイルを簡単に見つけ出せます。以下を含む各種の情報をご覧になれます。

- **製品サポート** - データシートとエラッタ、アプリケーション ノートとサンプル プログラム、設計リソース、ユーザガイドとハードウェア サポート文書、最新のソフトウェアと過去のソフトウェア
- **技術サポート** - FAQ(よく寄せられる質問)、技術サポートのご依頼、オンライン ディスカッション グループ、Microchip 社のデザイン パートナー プログラムおよびメンバーリスト
- **ご注文とお問い合わせ** - 製品セレクトと注文ガイド、最新プレスリリース、セミナー/イベントの一覧、お問い合わせ先(営業所/正規代理店)の一覧

製品変更通知サービス

Microchip 社の製品変更通知サービスは、お客様に Microchip 社製品の最新情報をお届けする配信サービスです。ご興味のある製品ファミリまたは開発ツールに関する変更、更新、リビジョン、エラッタ情報をいち早くメールにてお知らせします。

<http://www.microchip.com/pcn> にアクセスし、登録手続きをしてください。

カスタマサポート

Microchip 社製品をお使いのお客様は、以下のチャンネルからサポートをご利用になれます。

- 正規代理店
- 技術サポート

サポートは販売代理店にお問い合わせください。各地の営業所もご利用になれます。本書の最後のページに各国の営業所の一覧を記載しています。

技術サポートは以下のウェブページからもご利用になれます。 www.microchip.com/support

Microchip 社のデバイスコード保護機能

Microchip 社製品のコード保護機能について以下の点にご注意ください。

- Microchip 社製品は、該当する Microchip 社データシートに記載の仕様を満たしています。
- Microchip 社では、通常の条件ならびに仕様に従って使った場合、Microchip 社製品のセキュリティ レベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- Microchip 社はその知的財産権を重視し、積極的に保護しています。Microchip 社製品のコード保護機能の侵害は固く禁じられており、デジタル ミレニアム著作権法に違反します。
- Microchip 社を含む全ての半導体メーカーで、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、Microchip 社が製品を「解読不能」として保証するものではありません。コード保護機能は常に進歩しています。Microchip 社では、常に製品のコード保護機能の改善に取り組んでいます。

法律上の注意点

本書および本書に記載されている情報は、Microchip 社製品を設計、テスト、お客様のアプリケーションと統合する目的を含め、Microchip 社製品に対してのみ使用する事ができます。それ以外の方法でこの情報を使用する事はこれらの条項に違反します。デバイス アプリケーションの情報は、ユーザの便宜のためにのみ提供されるものであり、更新によって変更となる事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。
#TOIN#

その他のサポートは Microchip 社正規代理店にお問い合わせ頂くか、www.microchip.com/en-us/support/design-help/client-support-services をご覧ください。

Microchip 社は本書の情報を「現状のまま」で提供しています。Microchip 社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、商品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。

#TOIN#

いかなる場合も Microchip 社は、本情報またはその使用に関連する間接的、特殊的、懲罰的、偶発、的または必然的損失、損害、費用、経費のいかにかわらず、また Microchip 社がそのような損害が生じる可能性について報告を受けていた場合あるいは損害が予測可能であった場合でも、一切の責任を負いません。法律で認められる最大限の範囲を適用しようとも、本情報またはその使用に関連する一切の申し立てに対する Microchip 社の責任限度額は、使用者が当該情報に関連して Microchip 社に直接支払った額を超えません。

Microchip 社の明示的な書面による承認なしに、生命維持装置あるいは生命安全用途に Microchip 社の製品を使用する事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、Microchip 社は擁護され、免責され、損害をうけない事に同意するものとします。特に明記しない場合、暗黙的あるいは明示的を問わず、Microchip 社が知的財産権を保有しているライセンスは一切譲渡されません。

商標

Microchip 社の名称とロゴ、Microchip ロゴ、Adaptec、AVR、AVR ロゴ、AVR Freaks、BesTime、BitCloud、CryptoMemory、CryptoRF、dsPIC、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maXStylus、maXTouch、MediaLB、megaAVR、Microsemi、Microsemi ロゴ、MOST、MOST ロゴ、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32 ロゴ、PolarFire、Prochip Designer、QTouch、SAM-BA、SenGenuity、SpyNIC、SST、SST ロゴ、SuperFlash、Symmetricom、SyncServer、Tachyon、TimeSource、tinyAVR、UNI/O、Vectron、XMEGA は米国およびその他の国における Microchip Technology Incorporated の登録商標です。

AgileSwitch、APT、ClockWorks、The Embedded Control Solutions Company、EtherSynch、Flashtec、Hyper Speed Control、HyperLight Load、Libero、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus ロゴ、Quiet-Wire、SmartFusion、SyncWorld、Temux、TimeCesium、TimeHub、TimePictra、TimeProvider、TrueTime、ZL は米国における Microchip Technology Incorporated の登録商標です。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、Augmented Switching、BlueSky、BodyCom、Clockstudio、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、Espresso T1S、EtherGREEN、GridTime、IdealBridge、In-Circuit Serial Programming、ICSP、INICnet、Intelligent Paralleling、IntelliMOS、Inter-Chip Connectivity、JitterBlocker、Knob-on-Display、KoD、maxCrypto、maxView、memBrain、Mindi、MiWi、MPASM、MPF、MPLAB Certified logo、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、SmartHLS、SMART-I.S.、storClad、SQI、SuperSwitcher、SuperSwitcher II、Switchtec、SynchroPHY、Total Endurance、Trusted Time、TSHARC、USBCheck、VariSense、VectorBlox、VeriPHY、ViewSpan、WiperLock、XpressConnect、ZENA 米国およびその他の国における Microchip Technology Incorporated の商標です。

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

Adaptec ロゴ、Frequency on Demand、Silicon Storage Technology、Symmcom はその他の国における Microchip Technology Incorporated の登録商標です。

GestIC は、その他の国における Microchip Technology Germany II GmbH & Co. KG (Microchip Technology Inc.の子会社)の登録商標です。

その他の商標は各社に帰属します。

© 2024, Microchip Technology Incorporated and its subsidiaries.All Rights Reserved.

ISBN: 978-1-6683-3315-0

品質管理システム

Microchip 社の品質管理システムについては www.microchip.com/quality をご覧ください。

各国の営業所とサービス

北米

本社

2355 West Chandler Blvd.

Chandler, AZ 85224-6199

Tel:480-792-7200

Fax:480-792-7277

技術サポート:

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

URL:

www.microchip.com

アトランタ

Duluth, GA

Tel:678-957-9614

Fax:678-957-1455

オースティン、TX

Tel:512-257-3370

ボストン

Westborough, MA

Tel:774-760-0087

Fax:774-760-0088

シカゴ

Itasca, IL

Tel:630-285-0071

Fax:630-285-0075

ダラス

Addison, TX

Tel:972-818-7423

Fax:972-818-2924

デトロイト

Novi, MI

Tel:248-848-4000

ヒューストン、TX

Tel:281-894-5983

インディアナポリス

Noblesville, IN

Tel:317-773-8323

Fax:317-773-5453

Tel:317-536-2380

ロサンゼルス

Mission Viejo, CA

Tel:949-462-9523

Fax:949-462-9608

Tel:951-273-7800

ローリー、NC

Tel:919-844-7510

ニューヨーク、NY

Tel:631-435-6000

サンノゼ、CA

Tel:408-735-9110

Tel:408-436-4270

カナダ - トロント

Tel:905-695-1980

Fax:905-695-2078

アジア/太平洋

オーストラリア - シドニー

Tel:61-2-9868-6733

中国 - 北京

Tel:86-10-8569-7000

中国 - 成都

Tel:86-28-8665-5511

中国 - 重慶

Tel:86-23-8980-9588

中国 - 東莞

Tel:86-769-8702-9880

中国 - 広州

Tel:86-20-8755-8029

中国 - 杭州

Tel:86-571-8792-8115

中国 - 香港 SAR

Tel:852-2943-5100

中国 - 南京

Tel:86-25-8473-2460

中国 - 青島

Tel:86-532-8502-7355

中国 - 上海

Tel:86-21-3326-8000

中国 - 瀋陽

Tel:86-24-2334-2829

中国 - 深圳

Tel:86-755-8864-2200

中国 - 蘇州

Tel:86-186-6233-1526

中国 - 武漢

Tel:86-27-5980-5300

中国 - 西安

Tel:86-29-8833-7252

中国 - 厦門

Tel:86-592-2388138

中国 - 珠海

Tel:86-756-3210040

アジア/太平洋

インド - バンガロール

Tel:91-80-3090-4444

インド - ニューデリー

Tel:91-11-4160-8631

インド - プネ

Tel:91-20-4121-0141

日本 - 大阪

Tel:81-6-6152-7160

日本 - 東京

Tel:81-3-6880-3770

韓国 - 大邱

Tel:82-53-744-4301

韓国 - ソウル

Tel:82-2-554-7200

マレーシア - クアラルンプール

Tel:60-3-7651-7906

マレーシア - ペナン

Tel:60-4-227-8870

フィリピン - マニラ

Tel:63-2-634-9065

シンガポール

Tel:65-6334-8870

台湾 - 新竹

Tel:886-3-577-8366

台湾 - 高雄

Tel:886-7-213-7830

台湾 - 台北

Tel:886-2-2508-8600

タイ - バンコク

Tel:66-2-694-1351

ベトナム - ホーチミン

Tel:84-28-5448-2100

ヨーロッパ

オーストリア - ヴェルス

Tel:43-7242-2244-39

Fax:43-7242-2244-393

デンマーク - コペンハーゲン

Tel:45-4485-5910

Fax:45-4485-2829

フィンランド - エスポー

Tel:358-9-4520-820

フランス - パリ

Tel:33-1-69-53-63-20

Fax:33-1-69-30-90-79

ドイツ - ガーヒング

Tel:49-8931-9700

ドイツ - ハーン

Tel:49-2129-3766400

ドイツ - ハイムブロン

Tel:49-7131-72400

ドイツ - カールスルーエ

Tel:49-721-625370

ドイツ - ミュンヘン

Tel:49-89-627-144-0

Fax:49-(89-627)-144/-44

ドイツ - ローゼンハイム

Tel:49-8031-354-560

イスラエル - ラーナナ

Tel:972-9-744-7705

イタリア - ミラノ

Tel:39-0331-742611

Fax:39-0331-466781

イタリア - パドヴァ

Tel:39-049-7625286

オランダ - ドリュエネン

Tel:31-416-690399

Fax:31-416-690340

ノルウェー - トロンハイム

Tel:47-7288-4388

ポーランド - ワルシャワ

Tel:48-22-3325737

ルーマニア - ブカレスト

Tel:40-21-407-87-50

スペイン - マドリッド

Tel:34-91-708-08-90

Fax:34-91-708-08-91

スウェーデン - ヨーテボリ

Tel:46-31-704-60-40

スウェーデン - ストックホルム

Tel:46-8-5090-4654

イギリス - ウォーキングム

Tel:44-118-921-5800

Fax:44-118-921-5820