

ATtiny3227 Curiosity Nano

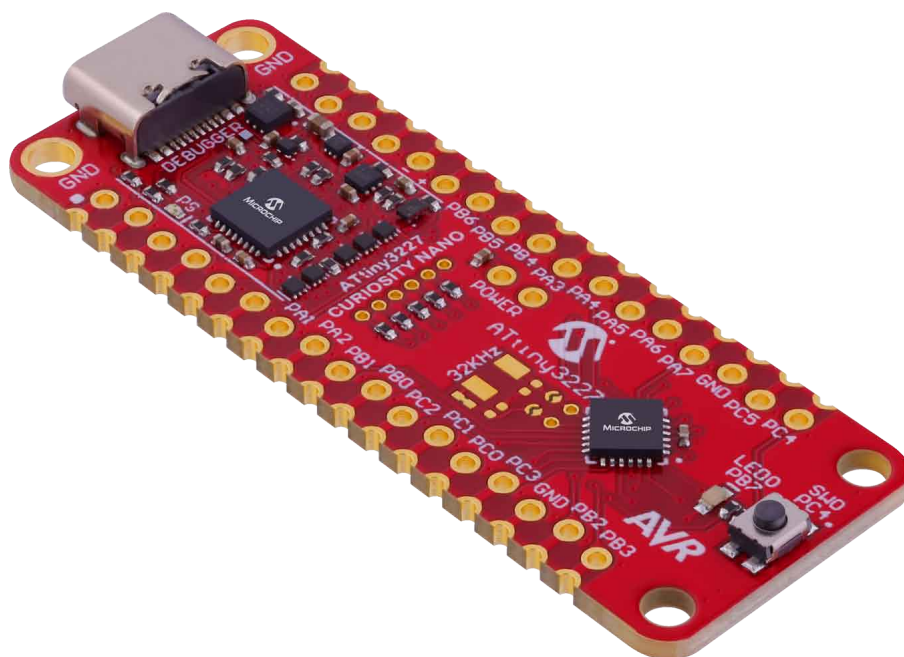


www.microchip.com製品頁: ATtiny3227

序文

ATtiny3227 Curiosity Nano評価キット(EV58A83A)はtinyAVR® 2系のマイクロ コントローラを評価するためのハードウェア基盤です。この基板は実装されたATtiny3227マイクロ コントローラ(MCU)を持ちます。

Curiosity Nano系列の評価キットは基板上デバッグを含みます。ATtiny3227を書いてデバッグするのに外部ツールが必要ありません。



- [ATtiny3227 Curiosity Nano ウェブサイト](#) – キット情報、最新の使用者の手引き、設計資料
- [ATtiny3227 ウェブサイト](#) – 資料、データシート、試供品を見つけて、マイクロコントローラを購入
- [MPLAB® Discoverでのコード例](#) – コード例で開始
- [Microchip直販でのATtiny3227 Curiosity Nano](#) – Microchip直販でこのキットを購入
- [ATtiny3227 Curiosity Nano 回路図](#) – 基板回路図と履歴
- [ATtiny3227 Curiosity Nano Altiumプロジェクト](#) – 最新製品改訂
- [ATtiny3227 Curiosity Nano 設計資料](#) – 全ての版に対する製作ファイル

本書は一般の方々の便宜のため有志により作成されたもので、Microchip社とは無関係であることを御承知ください。しおりの[はじめに]での内容にご注意ください。

目次

序文	1
1. 特徴とピン配置	3
1.1. ATtiny3227主な特徴	3
1.2. 基板特徴	3
1.3. 基板概要	4
1.4. 構成図	4
1.5. ピン配置	4
2. 開始に際して	5
2.1. AVR®で今すぐ開始	5
2.2. MPLABツール エコシステムへのCuriosity Nano適合方法	5
2.3. Curiosity NanoのMPLABデータ可視器支援	7
2.4. ピンヘッダ使用法	7
3. 基板上デバッグ	9
3.1. 基板上デバッグ概要	9
3.2. 基板上デバッグ接続	9
3.3. デバッグUSB列挙(接続認識)	10
3.4. 仮想シリアルポート(CDC)	10
3.5. 大容量記憶装置	12
3.6. データ中継器インターフェース(DGI)	14
3.7. デバイス構成設定保護	14
4. ハードウェア実装	16
4.1. クリスタル	16
4.2. LED	16
4.3. 機械的な切替器	17
4.4. 電源	17
5. 改訂履歴	21
5.1. ハードウェア改訂履歴と既知の問題	21
6. 文書改訂履歴	22
7. 追補	23
7.1. 回路図	23
7.2. 組立図	25
7.3. Click boards™用Curiosity Nano Base	26
7.4. 外部マイクロコントローラ書き込み	27
7.5. 外部デバッグ接続	28
7.6. 基板上デバッグ切断	29
7.7. IAR™でとの開始に際して	30
Microchip情報	31
商標	31
法的通知	31
Microchipデバイスコード保護機能	31

1. 特徴とピン配置

MCUとCuriosity Nanoの特徴、基板配置図、基板構成図、ピン配置構成図

1.1. ATtiny3227主な特徴

tinyAVR® 2系のマイクロ コントローラは32Kバイトのフラッシュ メモリ、3KバイトのSRAM、256バイトのEEPROMと共に最大20MHzで走行してハードウェア乗算器を持つAVR® CPUを使います。この系統は事象システム、高度なデジタル周辺機能、設定可能な利得増幅器(PGA)を持つ12ビット差動ADCのような正確なアナログ機能を含む柔軟で低電力な基本設計を持つMicrochipの最新技術を使います。

ハードウェア乗算器を持つAVR® RISC CPU

- ・ 最大20MHzで走行
- ・ 単一周期入出力(I/O)アクセス
- ・ 2段位の割り込み制御器
- ・ 休止動作: アイドル、スタンバイ、パワーダウン
- ・ 6チャンネルの事象システム
- ・ 構成設定可能な注文論理回路 (CCL)
- ・ 単一ピンのプログラム/デバッグ インターフェース (UPDI)

アナログ機能

- ・ 設定可能な利得増幅器(PGA)を持つ12ビット差動ADC
- ・ 内部DAC参照基準を持つアナログ比較器

タイマ/カウンタ

- ・ 1つの16ビット タイマ/カウンタA型 (TCA)
- ・ 2つの16ビット タイマ/カウンタB型 (TCB)
- ・ 周期的割り込み制御器(PIT)を持つ1つの16ビット実時間計数器 (RTC)

組み込み安全機能

- ・ 電源ONリセット (POR)
- ・ 低電圧検出 (BOD)
- ・ 電圧水準監視器 (VLM)
- ・ 巡回冗長検査(CRC)走査
- ・ 窓動作を持つウォッチドッグ タイマ (WDT)

通信

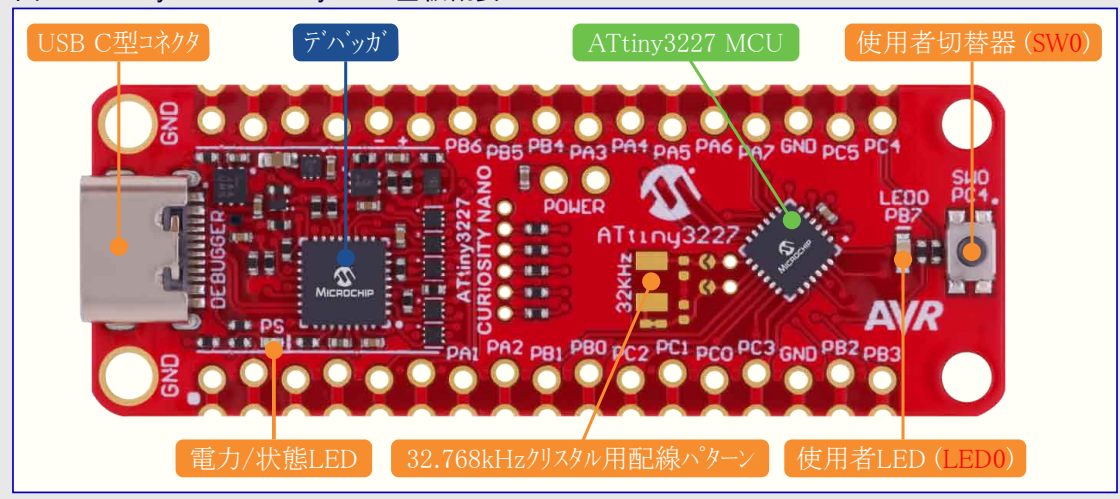
- ・ 直列通信インターフェース: SPI、I²C、2つのUSART

1.2. 基板特徴

- ・ ATtiny3227マイクロ コントローラ
- ・ USB C型コネクタ
- ・ 1つの黄色の使用済みLED
- ・ 1つの機械的な使用者切替器
- ・ 1つの緑色の電源と状態のLED
- ・ 32.768kHzクリスタル用配線パターン
- ・ Microchip MPLAB® X IDEでの基板上デバッグ支援
 - 基板識別
 - 仮想シリアル ポート (CDC)
 - 書き込みとデバッグ
 - 2つのデバッグGPIOチャンネル (DGI GPIO)
- ・ USB給電
- ・ 調整可能な目的対象電圧
 - 基板上デバッグによって制御されるMIC5353 LDO調整器
 - (USB入力電圧によって制限される)1.8~5.0Vの出力電圧
 - (周囲温度と出力電圧によって制限される)500mA最大出力電流

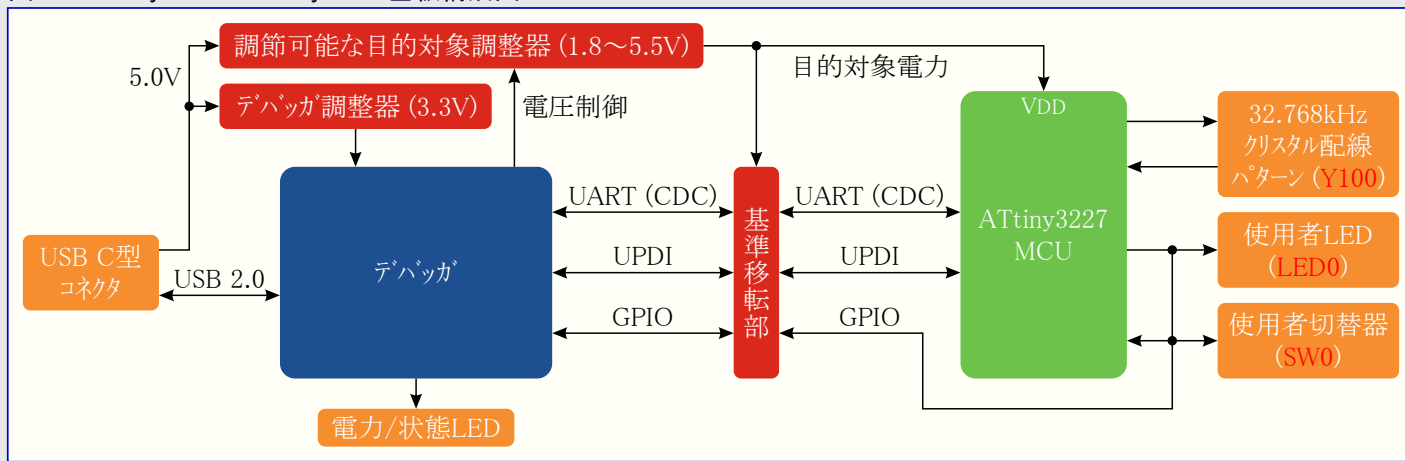
1.3. 基板概要

図1-1. ATtiny3227 Curiosity Nano基板概要



1.4. 構成図

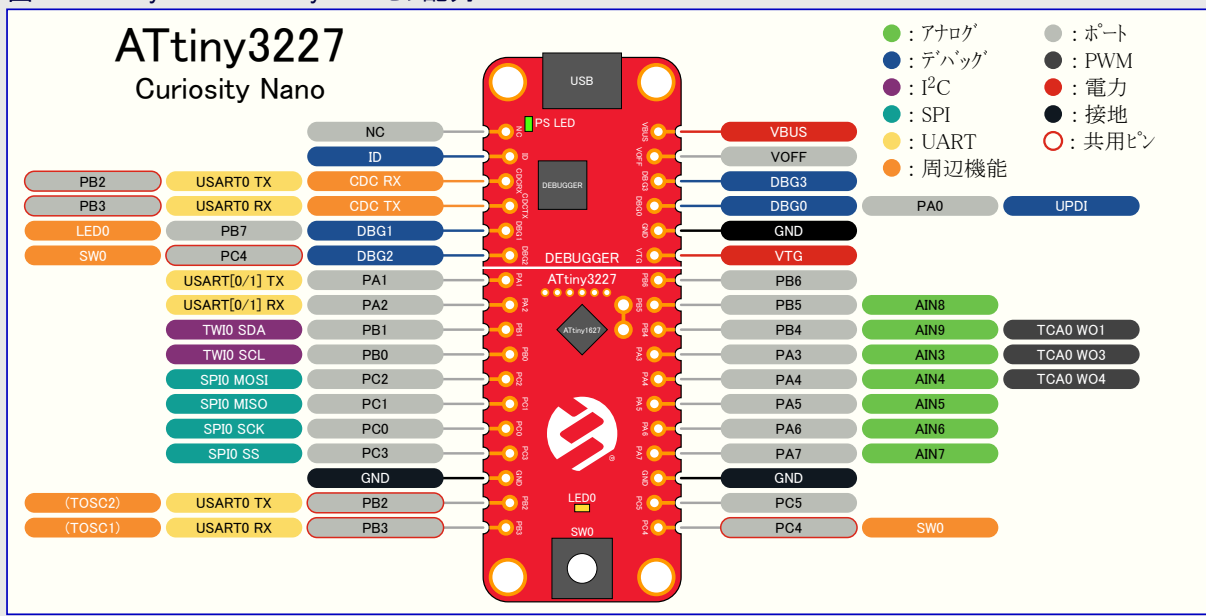
図1-2. ATtiny3227 Curiosity Nano基板構成図



1.5. ピン配置

ATtiny3227の全ての入出力ピンは基板上の端コネクタでアクセス可能です。下の画像は基板のピン配置を示します。各ピンで利用可能な機能についてはATtiny3227データシートの「入出力多重化と考察」章を参照してください。

図1-3. ATtiny3227 Curiosity Nanoピン配列



2. 開始に際して

MPLABツール エコシステムでのATtiny3227 Curiosity Nano基板用資源の開始に際して

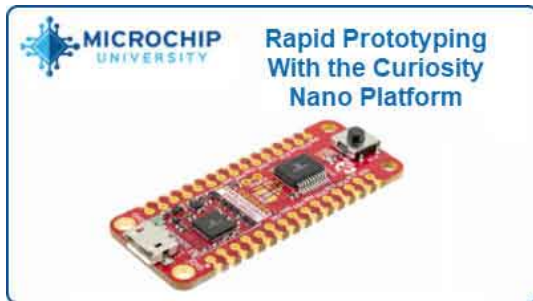
2.1. AVR®で今すぐ開始

AVRマイクロ コントローラ(MCU)を使うのは始めてですか?。我々の包括的なMPLAB®開発エコシステムはMicroshpのMCUを支援し、試作の稼働を容易にさせ、製品水準に対応するコード生成ツールを含みます。必要な資源を見つけるための対話型案内([Get Started Now with AVR MCUs](#))を試してください。

1. 例: [MPLAB Discover](#)はMicrochip例プロジェクトを見つけるのを手助けするツールです。これは開発の開始点として使うために応用の要件に可能な限り近いMicrochip検査済みの例を効率的に見つけるため、プロジェクトを選別する様々な方法を提供します。
2. 構成設定: [MCC Melody](#)はMPLAB® [Code Configurator\(MCC\)](#)を使って構成設定され、[支援されるMicrochip MCU](#)の範囲に対する組み込みソフトウェアを開発するためのライブラリ、ドライバ、周辺機能ライブラリ(PLIB)を提供します。
3. 開発: [MPLAB X IDE](#)(Integrated Development Environment)、これはWindows、Linux、macOS用が利用可能で、[MPLAB® XC コンパイラ](#)とAVR®用GCCコンパイラもです。
4. デバッグ: MPLAB X IDEと[MPLAB Data Visualizer](#)でATtiny3227 Curiosity Nano基板上のATtiny3227デバイスが基板上デバッグによって書かれてデバッグされます。従って、外部書き込み器やデバッグ ツールは必要とされません。
5. 基板: ATtiny3227 Curiosity Nano基板は[Curiosity Nano開発基盤](#)の支援対象部分で、これは[Click Boards™](#)用Curiosity Nano Baseを含みます。



助言: この無料のMicrochip University課程をご覧ください。



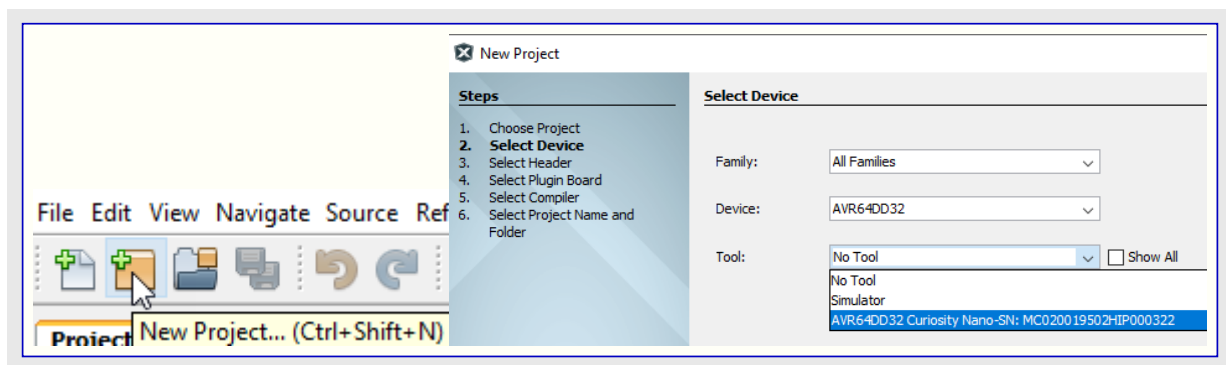
Curiosity Nano開発基盤は高速試作に良く適合します。この課程はCuriosity Nano基盤の明確な特徴を紹介し、新しいマイクロ コントローラの使用や新しい試作の開発でそれがどう手助けするかを示します。この課程はこの基盤に対する中心である基板上デバッグの詳細な考察、より早く目標に達するのを手助けするための様々な使用者インターフェースの使い方を含みます。

2.2. MPLABツール エコシステムへのCuriosity Nano適合方法

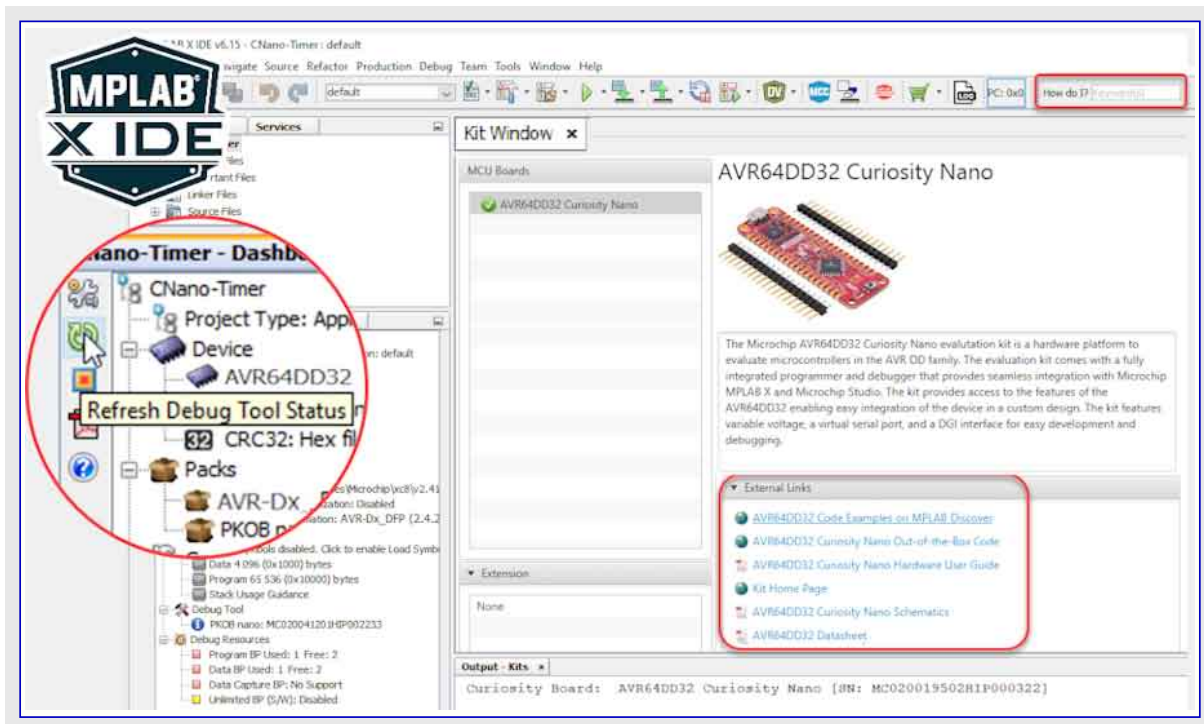
2.2.1. Curiosity NanoのMPLAB X IDE支援

この基板が始めてコンピュータに接続すると、オペレーティング システムはドライバ ソフトウェアをインストールします。この基板用のドライバはMPLAB® X IDEと共に含まれます。これが一旦行われると、USB経由でCuriosity NanoをホストPCに接続する時にMPLAB X IDEが開いていれば、そのCuriosity Nano用の様々な重要なリンクと共にキット ウィンドウが開かれます。

新しいプロジェクト作成時、Curiosity Nano上の部品番号が検出され、デバッグ ツールも同様になります。



- 助言: ・ ATtiny3227 Curiosity Nano基板にはMPLAB® X IDE 6.20版、デバイス系統一括”ATtiny_DFP” 2.6.122版、ツール一括”nEDBG_TP” 1.13.715版またはより新しい版を使ってください。
- ・ 最新デバイス系統一括はMPLAB® X IDEでTools(ツール)⇒Packs(一括)を通して、またはMPLAB® X一括貯蔵庫でのオンラインで入手可能です。一括でのより多くの情報とそれらの更新方法については[MPLAB® X IDE User's guide – Work with Device Packs](#)(MPLAB® X IDE使用者の手引き – デバイス一括との作業)を参照してください。

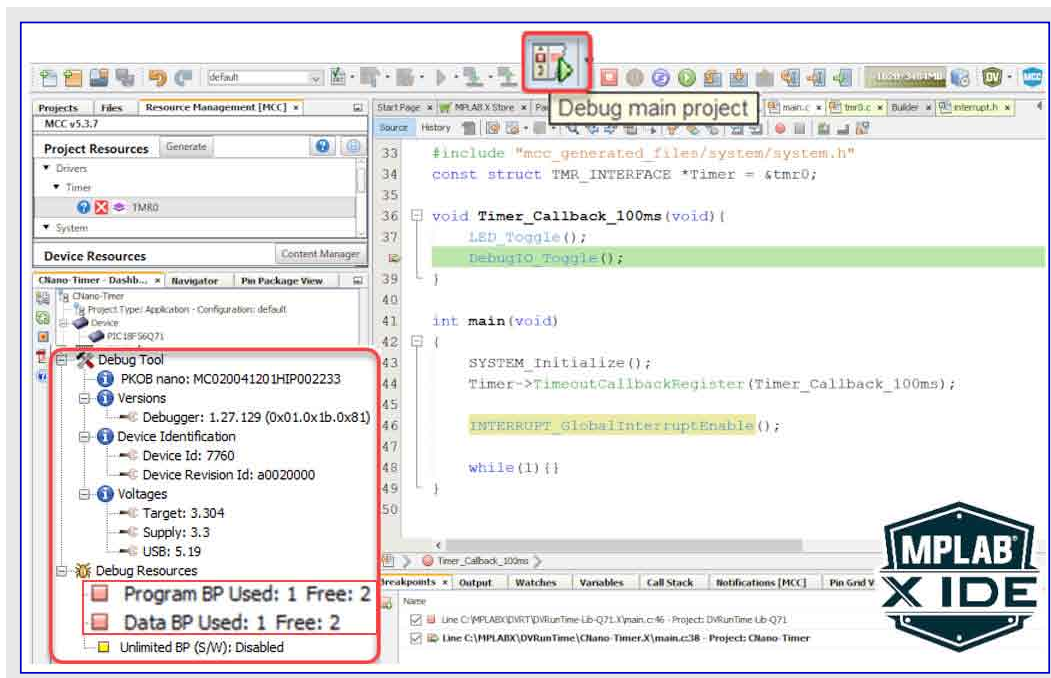


下の画像で示されるように、一旦”Refresh Debug Tool Status(デバッグ ツール状態更新)”をクリックすると、”Debug Tool(デバッグ ツール)”ウィンドウでCuriosity Nanoについての追加情報を見ることができます。



助言: ・ 閉じている場合、メニュー バーのWindow(ウィンドウ)⇒Kit Window(キット ウィンドウ)を通してMPLAB® X IDEでキット ウィンドウを再び開いてください。

- ・ MPLAB X IDEが始めての場合、’How do I?(どうすれば?)’ 検索バーが度々素晴らしい結果を与えます。
- ・ ’Debug main project(主プロジェクトをデバッグ)’ がデバッグ作業を開始します。

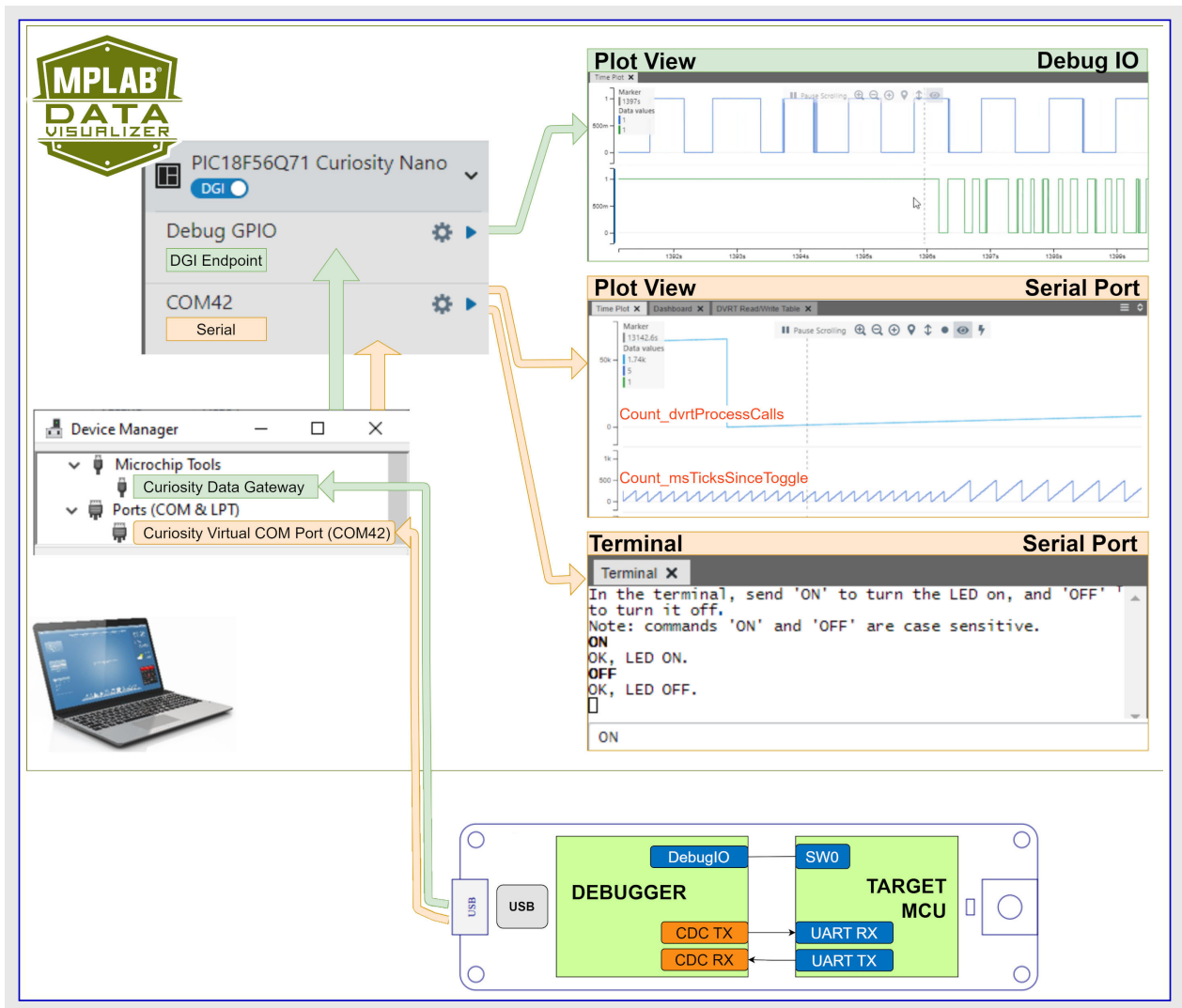


助言: ・ ’Refresh Debug Tool Status(デバッグ ツール状態更新)’をクリック後、MCU目的対象電圧のような情報を見ることができます。

- ・ プログラム中断点(Program-Break Point)とデータ中断点(Data-Break Point)とは何か?。 [MPLAB X IDE Advanced Debugging – Breakpoints Demo](#)(MPLAB X IDE高度なデバッグ – 中断点実演)
- ・ 前の例に対する参照 – [MCC Melody Timer0 Driver: 100 ms Timer, API Ref Code](#)

2.3. Curisity NanoのMPLABデータ可視器支援

USB/シリアル交換器経由のCuriosity Nanoは目的対象MCU上のUARTとコンピュータのCOMポート間の接続を容易にします。例えば、[MPLAB Data Visualizer](#)(MPLABデータ可視器)や他の端末プログラムへ接続するのにこれを使うかもしれません。



助言: 上のPlot View(作図視野)とTerminal(端末視野)の例に対する参照は次のとおりです。

1. Plot View – DebugIO: [DebugIO Heool World \(Microchip University\)](#)
2. Plot View – Serial Port: [MCC Melody Use Case Data Visualizer Run Time Use Case 1](#)
3. Terminal – Serial Port: [MCC Melody UART Driver: LED Control Commands](#)

2.4. ピン ヘッド 使用法

ATtiny3227 Curiosity Nano上の端コネク配線パターンは各々の穴が中心から8mil(約0.2mm)移動された千鳥足設計です。この穴ずれは通常の100mil(2.54mm)ピン ヘッドに基板上での半田付けなしで使うことを許します。ピン ヘッドは一旦それらがしっかりと実施すれば応用で問題なくピン ソケットと試作基板のように使うことができます。

図2-1. Curiosity Nano基板へのピン ヘッド取り付け

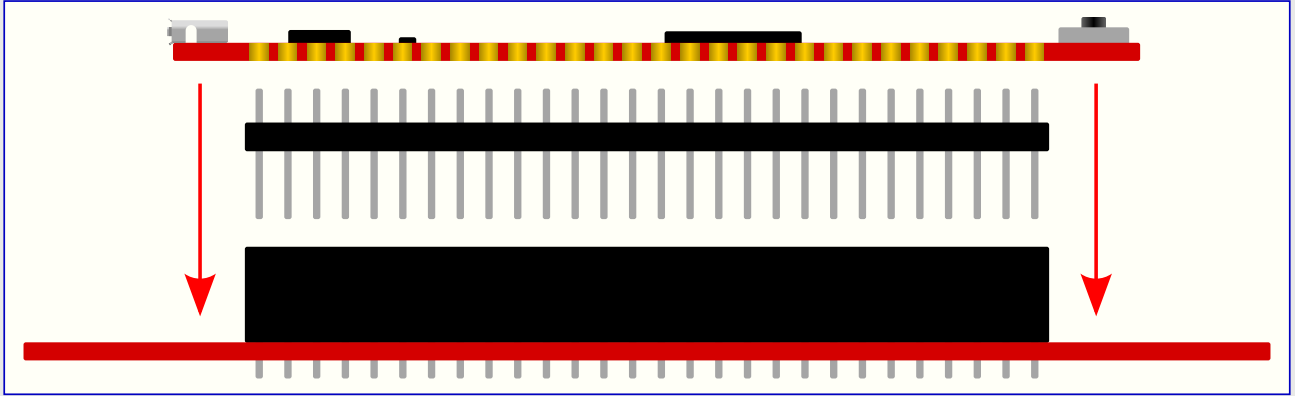
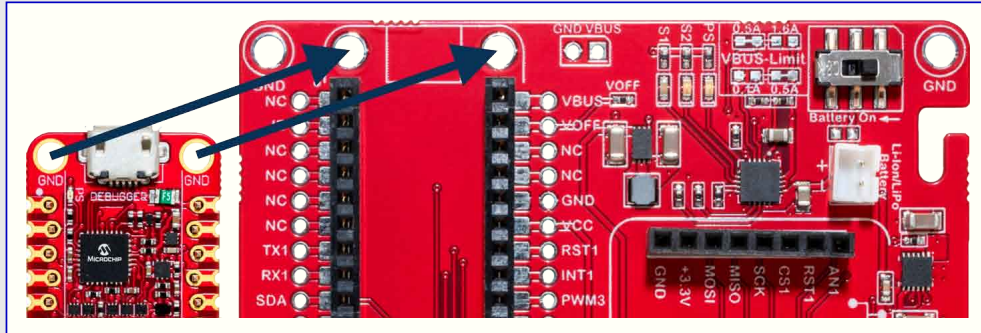


図2-2. Click boards™(Click基板)用Curiosity Nano Baseへの接続



- 助言:**
- ピン ヘッドの一端から開始し、基板の長手に沿ってヘッドを徐々に挿入してください。一旦全てのピンが実施されたなら、それらを幅広く押すのに平面を使ってください。
 - ピン ヘッドを定常的に使う応用については未だそれらを所定位置に半田付けすることが推奨されます。
 - 一旦ピン ヘッドが所定位置に置かれると、それらは手で取り外すのが難しくなります。ピン ヘッドとPCBへの損傷を避けるため、1組のペンチを使ってピン ヘッドを注意深く取り外してください。

3. 基板上デバッガ

書き込みとデバッガ用の基板上デバッガの特徴とインターフェース

3.1. 基板上デバッガ概要

ATtiny3227 Curiosity Nanoは書き込みとデバッガ用の基板上デバッガを含みます。基板上デバッガは以下のいくつかのインターフェースから成る複合USB装置です。

- MPLAB® X IDEでATtiny3227の書き込みとデバッガをすることができるデバッガ
- ATtiny3227の万能非同期送受信器(UART)に接続され、端末ソフトウェアを通して目的対象応用と通信する容易な方法を提供する仮想シリアルポート(CDC)
- ATtiny3227のトラッグ&ドロップ書き込みを許す大容量記憶装置
- プログラムの流れを可視化するための論理分析部チャネル(デバッガGPIO)でのコード計装用データ中継器インターフェース(DGI)

基板上デバッガはATtiny3227 Curiosity Nano基板上の(PSと記された)電力と状態のLEDを制御します。下表は各種動作形態がLEDをどう制御するかを示します。

表3-1. 基板上デバッガLED制御

動作形態	電力と状態のLED
ブートローダ動作	電源投入の間、LEDが低速点滅
電源投入	LEDがON
標準動作	LEDがON
書き込み	活動表示部: 書き込み/デバッガの間、LEDが低速点滅
トラッグ&ドロップ書き込み	成功: LEDが2秒間低速点滅 失敗: LEDが2秒間高速点滅
障害	電力障害が検出された場合にLEDが高速点滅
OFF	基板上デバッガは電力断で、LEDはOFF

情報: 低速点滅は概ね1Hzで、高速点滅は約5Hzです。

3.2. 基板上デバッガ接続

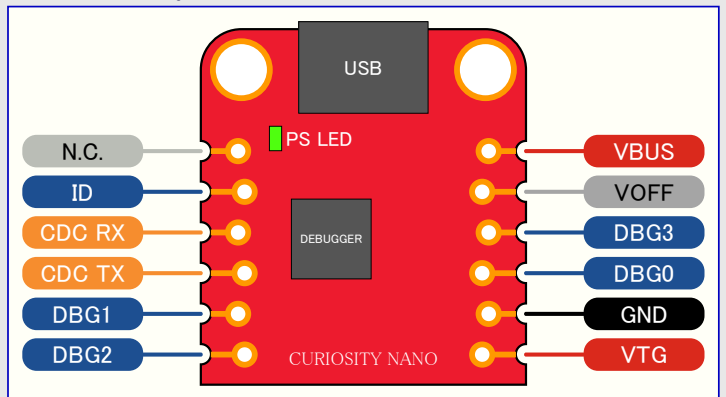
下表は目的対象とデバッガ部分間の接続を示します。目的対象とデバッガ間の全ての接続はデバッガがインターフェースを使わない時にHi-Zにされます。従って、僅かな信号の混入しかなく、例えば、ピンは使用者が望むどの構成設定にもすることができます。

情報: Curiosity Nano基板上のUSBコネクタに最も近い12端子の端コネクタは標準化されたピン配列を持ちます。書き込み/デバッガピンは目的対象インターフェースに依存して異なる機能を持ちます。

表3-2. 基板上デバッガ接続

デバッガピン	ATtiny3227ピン	説明
CDC TX	PB3	UART RX
CDC RX	PB2	UART TX
DBG0	PA0	UPDI
DBG1	PB7	LED0/GPIO1
DBG2	PC4	SW0/GPIO0
DBG3	-	-
ID	-	-
N.C.	-	-
VBUS	-	-
VOFF	-	-
VTG	-	-
GND	-	-

図3-1. Curiosity Nanoデバッガピン配列



助言: 完全なATtiny3227 Curiosity Nanoピン配置については「ATtiny3227 Curiosity Nanoピン配置」をご覧ください。

3.3. デバッグUSB列挙(接続認識)

ATTiny3227 Curiosity Nanoの基板上デバッグはホストコンピュータのUSBサブシステムで人インターフェース装置(HID:Human Interface Device)として現れます。このデバッグはMPLAB® X IDEといくつかの第三者IDEを使うことによってATTiny3227の完全な機能の書き込みとデバッグを支援します。



留意: デバッグのファームウェアを最新に保ってください。ファームウェアはMPLAB® X IDE使用時に自動的に更新します。

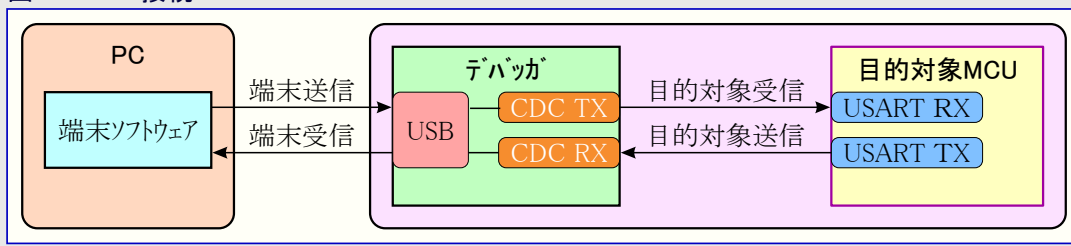
3.4. 仮想シリアルポート(CDC)

仮想シリアルポート(CDC)はホストPCと目的対象デバイス間の汎用シリアル橋渡しです。

3.4.1. 概要

基板上デバッグはホストで仮想シリアルポートとして現れる標準通信装置クラス(CDC:Communications Device Class)を持つ複合USB装置を実装します。ホストコンピュータと目的対象間の両方向で任意データを流すのにCDCを使ってください。ホストコンピュータで仮想シリアルポートを通して送られた全ての文字はデバッグのCDC TXピンでUARTとして送られます。デバッグのCDC RXピンで捕獲されたUART文字は仮想シリアルポートを通してホストコンピュータに返されます。

図3-2. CDC接続



情報: 上図で示されるように、ホストコンピュータから受け取る文字に対してデバッグのCDC TXピンは目的対象のUSART RXピンに接続されます。同様に、ホストコンピュータへ送られる文字に対してデバッグのCDC RXピンは目的対象のUSART TXピンに接続されます。

3.4.2. オペレーティングシステム支援

Windows®機ではCDCがCuriosity Virtual COM Port(Curiosity仮想COMポート)として列挙(接続認識)され、Windowsデバイス マネージャのポート部分に現れます。COMポート番号はそこで見つけることもできます。



情報: 古いWindowsシステムではCDCがUSBドライバを必要とします。MPLAB® X IDEのインストールはこのドライバを含みます。

Linux®機ではCDCが/dev/ttyACM#として列挙(接続認識)されて現れます。



情報: Linuxでtty*装置は”dialout”群に属し、故にCDCアクセスする許可を持つ群の一員になることが必要かもしれません。

MAC®機ではCDCが/dev/tty.usbmodem#として列挙(接続認識)されて現れます。使う端末プログラムに依存し、usbmodem#として利用可能なモデムの一覧で現れます。



情報: 全てのオペレーティングシステムに対して: DTR信号を支援する端末模倣部を使ってください。「合図」をご覧ください。

3.4.3. 制限

基板上デバッグのCDCで全てのUART機能が実装される訳ではありません。制限は以下のようにここで概説されます。

- ・ **ボーレート:** 1200bps～500kbpsの範囲でなければなりません。この範囲外のどのボーレートも警告なしに最も近い限度に設定されます。ボーレートは実行中に変えることができます。
- ・ **文字形式:** 8ビット文字だけが支援されます。
- ・ **パリティ:** 奇数、偶数、なしにすることができます。
- ・ **ハードウェア流れ制御:** 支援なし
- ・ **停止ビット:** 1または2のビットが支援されます。

3.4.4. 合図

USB列挙(接続認識)の間、ホストOSはCDCインターフェースの通信とデータのパイプを開始します。この時点で、CDCのボーレートと他のUARTパラメータを設定して読み戻すことが可能ですが、データの送出と受け取りは許可されません。

ホストに接続する時に端末はDTR信号を活性にしなければなりません。これがUSBインターフェースで実装される仮想制御信号のため、基板には物理的に存在しません。ホストからのDTR活性化は基板上デバッグにCDC作業が活性であることを示します。デバッグは(利用可能ならば)その基準移転器(レベルシフタ)を許可してCDCデータの送受信機構を開始します。

デバッグ ファームウェア 1.20 またそれ以前版でのDTR信号の不活性化は以下の動きを持ちます。

- ・デバッグUART受信部が禁止され、それ以上データはホスト コンピュータへ転送されません。
- ・デバッグUART送信部は転送用に準備し待ち行列にしたデータを送り続けますが、ホスト コンピュータからの新データを受け入れません。
- ・(利用可能なら)基準移転器は禁止されず、デバッグCDC TX線は駆動されたままに留まります。

デバッグ ファームウェア 1.21 またそれ以降版でのDTR信号の不活性化は以下の動きを持ちます。

- ・デバッグUART受信部が禁止され、更なるデータはホスト コンピュータへ転送されません。
- ・デバッグUART送信部は転送用に準備し待ち行列にしたデータを送り続けますが、ホスト コンピュータからの新データを受け入れません。
- ・一旦進行中の送信が完了すると、基準移転器が禁止され、故にデバッグCDC TX線は高インピーダンスになります。

 **留意:** 端末模倣部をDTR信号有効に設定してください。その信号なしでは基板上デバッグがそのUARTを通すどのデータの送信も受信もしません。

 **助言:** 基板上デバッグのCDC TXピンはホストによってCDCインターフェースが許可されるまで駆動されません。また、デバッグと目的対象に接続しているCDC線上に外部プルアップ抵抗がなく、電源投入中にその線が浮くことを意味します。フレーミング異常などのような予測不能な動きに帰着する不具合も避けるため、目的対象デバイスはデバッグのCDC TX線に接続されたピンで内部プルアップを許可することができます。

3.4.5. 高度な使い方

CDC置き換え動作

普通の動作では、基板上デバッグはホストとデバイス間のUART橋渡しです。けれども、或る使用事例で、基板上デバッグは基本動作形態を置き換えて他の目的のためにCDC TXとRXのピンを使うことができます。

基板上デバッグの大容量記憶ドライブへの文書ファイル引き摺りで、デバッグのCDC TXピンの出力に文字を送ることができます。ファイル名と拡張子は普通ですが、文書ファイルは次のような文字で始まります。

CMD:SEND_UART=

デバッグ ファームウェア 1.20 またそれ以前版は以下の制限を持ちます。

- ・最大メッセージ長は50文字で、フレーム内の全ての残りデータは無視されます。
- ・この動作で使われる既定ボーレートは9600bpsですが、CDCが既に活性、または構成設定されていた場合、以前に使われたボーレートが未だ適用されます。

デバッグ ファームウェア 1.21 またそれ以降版は以下の制限/機能を持ちます。

- ・最大メッセージ長はホスト コンピュータとオペレーティング システムでのMSC/SCSI層制限時間に依存して変わるでしょう。512バイトの単一SCSI フレーム(498文字の本体)が保証され、4Kバイトまでのファイルが殆どのシステムで動くでしょう。転送はファイルで出会った最初のNULL文字で完了されます。
- ・使うボーレートは既定命令に対して常に9600bpsです。

CMD:SEND_UART=

CDC置き換え動作はCDC/端末上でのデータ転送と同時に使わないでください。ファイルがCDC置き換え動作経由で受信している時にCDC端末作業が活動の場合、その操作の間一時休止され、一旦完了すると再開されます。

- ・明示的なボーレートで以下のような追加命令が支援されます。

CMD:SEND_9600=

CMD:SEND_115200=

CMD:SEND_460800=

USB段階のフレームの考慮

ホストからCDCへ送るデータはバイト単位、または64バイトUSBフレーム内に切り分けられる塊で行うことができます。このような各々のフレームはデバッグのCDC TXピンへ送るため、待ち行列にされます。フレーム毎に少量のデータを送ると、基板上デバッグがバイトではなくフレームを緩衝するため、特に低ボーレートで非効率になり得ます。最大4つの64バイト フレームを何時でも活性にすることができます。基板上デバッグはそれによってやって来るフレームを調整します。データを含む完全な64バイト フレームの送信が最も効率的な方法です。

デバッグのCDC RXピンでデータを受け取る時に、基板上デバッグはやって来るバイトを64バイト フレームへ一列に並べ、それらが満たされた時にホストへ送るためにUSB待ち行列に送られます。不完全なフレームも概ね100ms間隔でUSB待ち行列へ押し込まれ、USBフレーム開始通票によって起動されます。何時でも最大8つの64バイト フレームを活性にすることができます。

ホスト(または走行しているソフトウェア)が充分速くデータを受け取ることに失敗した場合、オーバーランが発生します。これが起きると、USB待ち行列に送られつつあるものに代わって最後に満たされた緩衝部フレームが再使用され、完全なフレームデータが失われます。これを防ぐため、使用者はCDCデータパイプが継続的に読まれるか、またはやって来るデータ速度が減らされるかを保証してください。

中断(Break)文字送出

ホストはCDCを使う装置にUART中断文字を送ることができ、これは受信部の状態機構をリセットしたり、装置で走行する応用にホストから例外条件を合図するのに使えます。

中断文字はホストから装置へ送られる最低11ビットの'0'の連続です。

全てのUART受信部が中断を検出するための支援を持つ訳ではありませんが、正しい形式の中断文字は通常、受信部でフレーミング異常を起動します。

デバッグのCDCを使う中断文字送信は以下の制限を持ちます。

- ・ 中断送出はCDC置き換え動作(ドラッグ&ドロップ)と同時に行ってはなりません。これら両機能は(限られた時間続く)一時的な状態で、独立して使われなければなりません。
- ・ 中断送出は送られつつあるどのデータも損失させます。中断を送る前に送信緩衝部内の全ての文字が送られるのを許すために充分量の時間待つのを確実にしてください。これは予期した中断文字の使い方に沿っていても、例えば、制限時間後の受信部状態機構リセットはホストに返されるデータを待たせます。
- ・ CDC仕様は最大65534msの持続時間のデバッグ計時の中断の要求を許します。簡単にするため、デバッグは中断持続時間を支援する最低ボーレートで最大11ビット持続に制限します。
- ・ CDC仕様は不定のホスト計時中断を許します。この場合に中断状態を開放するのは端末応用/使用者の責任です。

注: 中断文字送出はデバッグ ファームウェア1.24とそれ以降版で利用可能です。

3.5. 大容量記憶装置

基板上デバッグはそれが接続されるホスト オペレーティング システム経由で読み書き操作に対してアクセスができる簡単な大容量記憶装置実装を含みます、

これは以下を提供します。

- ・ キットの情報と支援を詳述するための基本的な文書とHTMLのファイルに対する読み込みアクセス
- ・ Intel® HEXとUF2の形式のファイルを目的対象デバイスのメモリに書くための書き込みアクセス
- ・ 有用な目的用の簡単な文書ファイルのための書き込みアクセス

注: UF2形式の支援はデバッグ ファームウェア1.31またはそれ以降版で利用可能です。

3.5.1. 大容量記憶装置実装

基板上デバッグは部分的にFAT12それ自身の特質とそれの組み込み応用に対する目的を満たすための最適化のため、いくつかの制限を持ち高く最適化されたFAT12ファイル システムの変種を実装します。

Curiosity Nano USB装置は大容量記憶装置としてUSB第9節適合ですが、汎用大容量記憶装置で期待するものを多少なりとも満たしません。この動きは意図的です。

Windowsオペレーティング システム使用時、基板上デバッグはデバイス マネージャのディスク部分で見つかるCuriosity Nano USB装置として列挙(認識)されます。CURIOSITYドライブはファイル マネージャに現れ、システムで次に利用可能なドライブ文字を獲得します。

CURIOSITYドライブは概ね1Mバイトの空き空間を含み、目的対象デバイスのフラッシュ メモリの大きさを反映しません。Intel HEXまたはUF2のファイル書き込み時、膨大な付随負荷を与える付加データを持つASCIIで符号化され、故に1Mバイトはディスクの大きさ用に適当に選ばれた値です。

CURIOSITYドライブをフォーマットすることは不能です。目的対象へのファイル書き込み時、ファイル名がディスク ディレクトリ一覧に現れるかもしれませんが、これは単にオペレーティング システムのディレクトリ表示にすぎず、現実には更新されません。そのファイル内容を読み出すことは不可能です。基板を取り外して再接続すると、ファイル システムをその元の状態に戻しますが、目的対象は未だ直前に書かれた応用を含みます。

目的対象デバイスを消去するにはディスクに”CMD:ERASE”で始まる文字ファイルを複写してください。

既定でCURIOSITYドライブはアイコン生成だけでなく、状態と更なる情報へのリンクを報告するための以下のようないくつかの読み込み専用ファイルを含みます。

- ・ **AUTORUN.ICO** – Microchipロゴ用アイコン ファイル
- ・ **AUTORUN.INF** – アイコン ファイルを表示するためにWindowsのエクスプローラに対して必要とされるシステム ファイル
- ・ **KIT-INFO.HTM** – 開発基板ウェブサイトへの向け直し
- ・ **KIT-INFO.TXT** – 基板のデバッグ ファームウェア版、基板名、USB通番、デバイス、ドラッグ&ドロップ支援の詳細を含む文字ファイル
- ・ **STATUS.TXT** – 基板の書き込み状態を含む文字ファイル



情報: 基板上デバッグは**STATUS.TXT**を動的に更新します。その内容はOSがキャッシュするため、正しい状態を反映しないかもしれません。

3.5.2. UF2形式を使うドラッグ&ドロップ書き込み

ドラッグ&ドロップ用UF2形式

ドラッグ&ドロップ書き込みはCuriosity Nanoキットで基板上のマイクロコントローラの不揮発性メモリを書くための簡単な仕組みを提供します。これは代表的に書式の一部として必要なアドレスとセグメントの情報を含むIntel® HEX形式を使って行われます。Intel HEXファイルはASCII文字として符号化したメモリ内容を含み、これは厳密な順番で解析されなければならない、ホストオペレーティングシステムが正しい順序で内容を送らなければならないことを意味します。これは全てのオペレーティングシステムの全ての変種の場合に常に該当する訳ではありません。UF2形式は順序どおり以外で転送されるメモリを許す手段としてMicrosoftによって開発されました。これは部分的な書き込みが防がれるようにデータ転送経路全体を通して使われる固定の塊の大きさを強制することによって行われます。

UF2形式のより多くの情報は[GitHub](#)で利用可能です。

UF2ファイルの生成

プロジェクトコンパイル手順の結果は代表的にIntel HEXファイルで、これは構築後段階を使ってUF2ファイルに変換することができます。pypi.orgで配給されるpymcuprog一括は3.17またはそれ以降版でIntel HEXファイルをUF2ファイルに変換する関数を含みます。

注: この手順はその環境にPyrhon 3とpymcuprog一括の最新公開版のインストールと、Pythonスクリプトフォルダがシステムまたはユーザーのパスに含まれることが必要です。

Intel HEXファイルは次のようなpymcuprogコマンド行インターフェースを使ってUF2ファイルに変換することができます。

```
pymcuprog makeuf2 -f app.hex --uf2file app.uf2
```

この手順はMPLAB X IDEで構築後段階を追加することによって合理化することができます。Project Properties(プロジェクト特性)構成設定ダイアログでBuilding(構築)タブを選び、Execute this line after build(構築後この行を実行)をチェックしてください。編集枠に次のような指令を追加してください。

```
pymcuprog makeuf2 -f ${ImagePath} --uf2file ${ImageDir}%${ProjectName}.X.${IMAGE_TYPE}.uf2
```

応用が構築される時毎に生成したIntel HEXファイルは同じファイル名で.uf2ファイル拡張子を持つUF2ファイルへ自動的に変換されます。

3.5.3. ドラッグ&ドロップ書き込みの制限

施錠ビット

Hexファイルに含まれる施錠ビットはドラッグ&ドロップ書き込みを使う時に無視されます。施錠ビットを書くにはMPLAB® X IDEを使ってください。

ヒューズでのCRC検査許可

ドラッグ&ドロップ書き込みを使う時にデバイスのヒューズでCRC検査を許可することはお勧めできません。これは(ヒューズビットに影響を及ぼさない)後続するチップ消去がCRC不整合を起こし、応用が起動に失敗するからです。MPLAB® X IDEを使ってチップ消去が行われなければならない、この状態から目的対象を回復ため、消去後、自動的にCRCヒューズを解消します。

3.5.4. 特殊命令

大容量記憶装置への文字ファイル複写によっていくつかの有用な命令が支援されます。ファイル名と拡張子は無関係で、命令処理部は内容だけに反応します。

表3-3. 特殊ファイル命令

命令内容	説明
CMD:ERASE	目的対象チップ消去を実行
CMD:SEND_UART=	CDC UARTに文字列を送信。「CDC置き換え動作」をご覧ください。
CMD:SEND_9600= CMD:SEND_115200 CMD:SEND_460800	指定したボーレートで文字列をCDC UARTへ送ります。ここで明示的に指定したボーレートだけが支援されることに注意してください。「CDC置き換え動作」をご覧ください。(デバッグファームウェア1.25.6版またはより新しい版)
CMD:RESET	書き込み動作へ入って直ちに書き込み動作を抜け出すことによって目的対象をリセット。正確なタイミングは目的対象デバイスの書き込みインターフェースに従って変わり得ます。(デバッグファームウェア1.25.6版またはより新しい版)
CMD:POWERTOGGLE	目的対象を電源断して100ms遅延後に電源を回復します。外部電力が提供されている場合、これは無効です。(デバッグファームウェア1.25.6版またはより新しい版)
CMD:0V	目的対象供給調整器を禁止することによって目的対象デバイスを電源断。外部電力が提供されている場合、これは無効です。(デバッグファームウェア1.25.6版またはより新しい版)
CMD:1V8	目的対象電圧を1.8Vに設定。外部電力を使う場合、これは無効です。(デバッグファームウェア1.25.6版またはより新しい版)
CMD:3V3	目的対象電圧を3.3Vに設定。外部電力を使う場合、これは無効です。(デバッグファームウェア1.25.6版またはより新しい版)

情報: 大容量記憶模倣ディスクに送られた内容は前の表で一覧にされる命令を起動し、成功と失敗のどちらの場合も反応は提供されません。

3.6. データ中継器インターフェース (DGI)

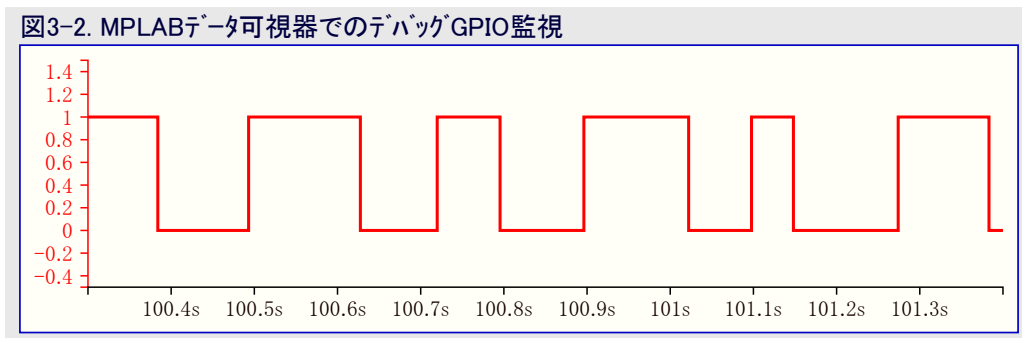
データ中継器インターフェース(DGI:Data Gateway Interface)は基板上デバッグとホスト コンピュータに基づく可視化ツール間で生と時刻印されたデータを転送するUSBインターフェースです。ホスト コンピュータで何れのデバッグGPIOデータを表示するのにMPLAB Data Visualizer(データ可視器)が使われます。これはMPLAB® X IDE用プラグインとしてまたはMPLAB® X IDEと並行して使うことができる独立型応用として利用可能です。

ATtiny3227 Curiosity NanoはDGI GPIO0とDGI GPIO1の利用可能な2つのデバッグGPIOチャンネルを持ちます。

3.6.1. デバッグGPIO

デバッグGPIOチャンネルは目的対象応用をホスト コンピュータ可視化応用に接続する時刻印されたデジタル信号線です。これらは例えば、与えられた応用状態遷移が起きた時のような、代表的に時間軸での低周波数事象の発生を作図するのに使われます。

下図はMPLABデータ可視器(Data Visualizer)でデバッグGPIOに接続された機械的な切替器のデジタル状態の監視を示します。



デバッグGPIOチャンネルは時刻印され、故にDGI GPIO事象の分解能はDGI時刻印単位部分分解能によって決められます。

重要: より高い周波数の信号集中の捕獲が可能でも、GPIOが使える信号の周波数範囲は最大約2kHzです。この周波数を超える信号を捕獲する試みはデータの飽和と溢れに帰着し、DGI作業の中断を引き起こすかもしれません。

3.6.2. 時刻印

デバッグによって捕獲される時にDGI供給元は時刻印されます。Curiosity Nanoデバッグで実装される時刻印計数器は0.5μsの時刻印分解能を提供する2MHzの周波数で増されます。

3.7. デバイス構成設定保護

ATtiny3227 Curiosity Nanoの基板上デバッグは、特にドラッグ&ドロップ書き込み使用時にMCUが回復不能にされるのを防止することを意図される保護機構を持ちます。プログラミングとデバッグのインターフェースの禁止やメモリの恒久的な施錠はいくつかのMCUの特徴で、それは開発キット基盤での評価の助けになりません。

この保護機構は関連するヒューズへの書き込み操作を阻止して書かれるアドレスと値を条件付きで遮蔽することによって働きます。

情報: 保護機構は予想外で元に戻せない変更を防ぐことを意図されます。使用者の意図的な元に戻せない変更は、自己責任で行ってください。

情報: 利用可能な機能一式と対応する保護機構はデバイスに依存します。更なる情報についてはデータシートを調べてください。

保護機構は元に戻せない変更を必要とする与えられた機能の完全な経験を望む使用者に対して禁止することができます。これを行うと、恒久的な変更になります。

pypi.orgで配給されるpydebuggerconfig一括は構成設定保護を含め、デバッグの多くの面を細かく調整するのに使うことができます。この手続きはPython 3の最新公開版とその環境のpydebuggerconfig一括のインストールを必要とします。

手順1: 現在の状態

次を実行することによって保護が現在許可されているかどうかを確定してください。

```
pydebuggerconfig read
```

その後に以下の部分を調べてください。

```
Register TARGET_DEBUG_FEATURES: 0x0F (15) # Program/debug features
                                bit 0, SINGLE_DEVICE: 1 # Single-device
                                bit 1, PROG_ENABLED: 1 # Programming
                                bit 2, DEBUG_ENABLED: 1 # Debug
                                bit 3, FUSE_PROTECTION: 1 # Fuse protection
```

‘1’のFUSE_PROTECTION値は保護が実施されていることを示します。

手順2: 保護設定の変更

FUSE_PROTECTIONを‘0’に設定することによってTARGET_DEBUG_FEATURESに対して望む設定を計算してください。例えば、次を実行することによってTARGET_DEBUG_FEATURESレジスタで現在の値を置き換えてください。

```
pydebuggerconfig replace -r TARGET_DEBUG_FEATURES=0x07
```

新しい値を確認するために手順1からの手続きを繰り返し、その後にキットの電源をOFF/ONしてください。全ての保護機能が今や禁止されます。

手順3: 保護の復元

保護は初期値で手順2を繰り返すか、または次を実行することによって工場復元機能を使うかのどちらかによって再許可することができます。

```
pydebuggerconfig restore
```

 **重要:** 復元機能は製造以降に変更されたキット構成設定の何れかの変更だけを復元し、デバuggに内部的に保存されたキット構成設定の複製を復元することによって行われます。この操作はMCUに影響を及ぼさず、また、その構成設定に行われた元に戻せない変更は元に戻しません。

4. ハードウェア実装

LED、機械的な切替器、電源、クリスタル

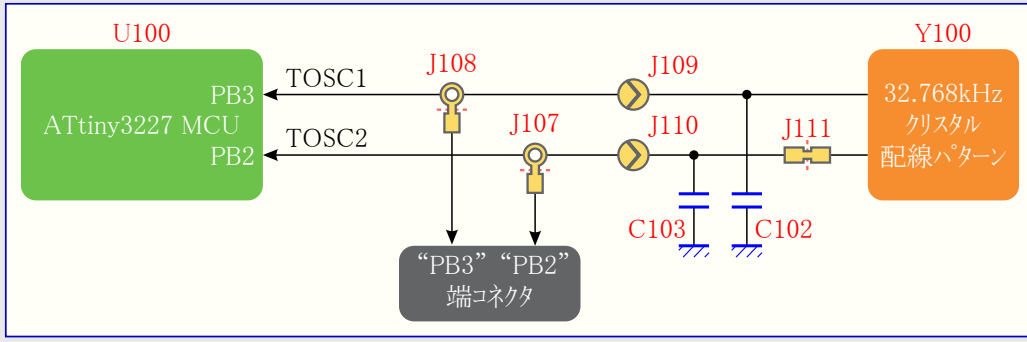
4.1. クリスタル

ATtiny3227 Curiosity Nano基板は2つの端子で標準3.2×1.5mm面実装クリスタル用32.768kHzクリスタル配線パターンを持ちます。GPIOが端コネクタに配線されているため、この配線パターンは既定でATtiny3227に接続されていません。

表4-1. クリスタル接続

ATtiny3227ピン	機能	共用機能
PB3	TOSC1	端コネクタ
PB2	TOSC2	端コネクタ

図4-1. 32.768kHzクリスタル構成図

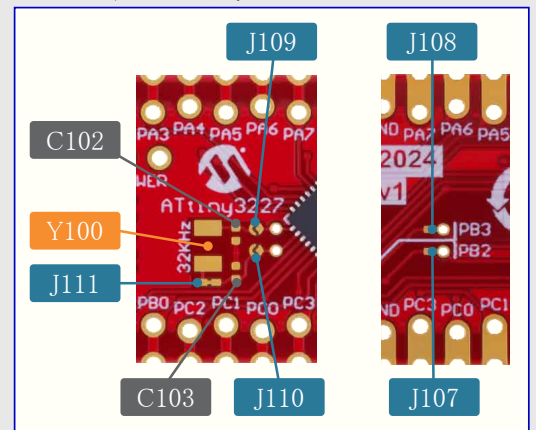


ATtiny3227でクリスタルを使うにはいくつかのハードウェア変更が必要とされます。

- クリスタルへの競合の機会を減らしてこの線での過度な容量を取り去るため、端コネクタに配線された2つの入出力線は切断されなければなりません。
- 基板の裏側でJ107とJ108の2つの切断帯を切ることによって端コネクタを切断してください。シルクスクリーンでPB2とPB3と記されます。
- 基板の表側でクリスタル傍のJ109とJ110の各円形半田点に於いて小さな半田塊で半田付けすることによってクリスタルを接続してください。
- C102とC103に適切なコンデンサを追加してください。

クリスタルは発振器の安全係数を測定するのに使うことができる切断帯 (J111) を持ち、この帯を切断して帯を渡して0402 SMD抵抗器を追加することによって行われます。MicrochipからのAN2648応用記述は発振器許容量と安全係数についてより多くの情報を含みます。

図4-2. クリスタル切断帯



4.2. LED

ATtiny3227 Curiosity Nano基板で1つの使用者用黄LEDが利用可能です。GPIOまたはPWMのどちらかでこれを制御することができます。接続した入出力線をGNDに駆動することもこのLEDを活性(点灯)にすることができます。

図4-3. ATtiny3227 Curiosity Nano LED0構成図

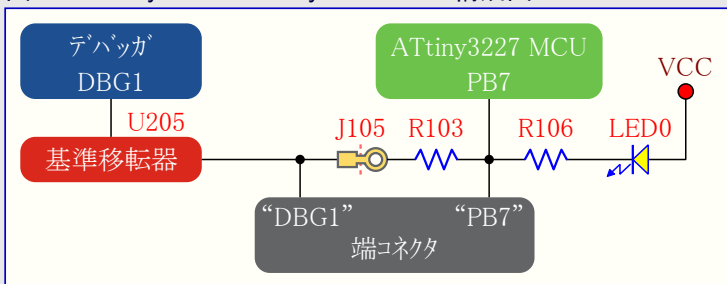


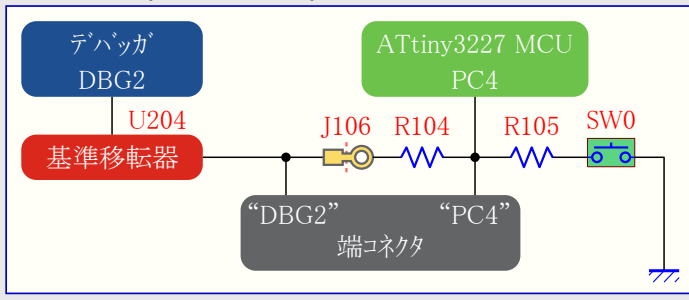
表4-2. LED接続

ATtiny3227ピン	機能	共用機能
PB7	黄色LED0	端コネクタ、基板上デバッグ

4.3. 機械的な切替器

ATtiny3227 Curiosity Nano基板は一般的な使用者構成設定可能な切替器、1つの機械的な切替器を持ちます。その押下はその入出力ピンを接地(GND)に接続します。

図4-4. ATtiny3227 Curiosity Nano SW0構成図



助言: 切替器に外部的に接続されたプルアップ抵抗器はありません。これを使うにはPC4ピンの内部プルアップ抵抗を許可してください。

表4-3. 機械的な切替器接続

ATtiny3227ピン	機能	共用機能
PC4	使用者切替器 (SW0)	端コネクタ、 基板上デバッグ

4.4. 電源

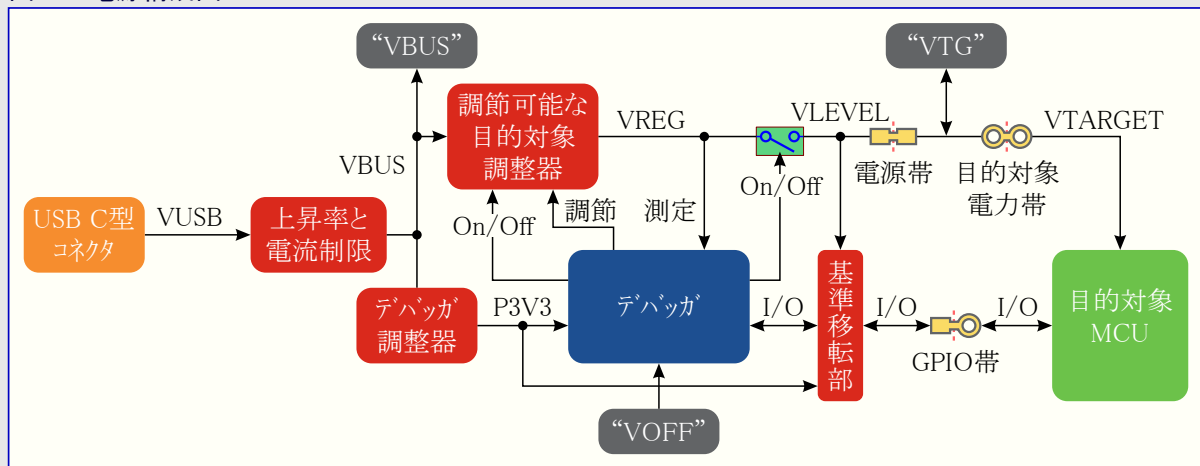
USBポートは基板に給電します。VBUS網はU202(MIC2008)によって2V/msの上昇率に制限され、500mAに電流制限されます。



助言: C206とR211の値を変更すると、U202によって設定される上昇率と電流制限を変えることができます。

電源は基板上デバッグ用3.3Vを生成するための1つと、目的対象ATtiny3227とその周辺機能用の調節可能なLDO調整器の2つのLDO調整器から成ります。USBコネクタからの電圧は(USB仕様に応じて)4.4~5.25V間で変わり得て、目的対象へ供給される最大電圧を制限します。下図はATtiny3227 Curiosity Nanoの電源系全体を示します。

図4-5. 電源構成図

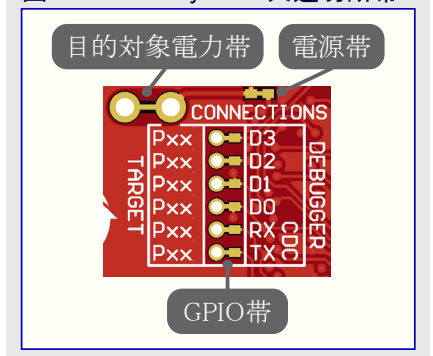


4.4.1. 切断帯

全ての電力とデバッグ信号は既定で目的対象に接続されます。測定するため、または目的対象からデバッグを分離するために以下の切断帯が利用可能です。

- 目的対象電力帯 (J201)
- 電源帯 (J201)
- デバッグピン (J101, J102, J103, J104, J105, J106)

図4-6. Curiosity Nano共通切断帯



4.4.2. 目的対象調整器

目的対象電圧調整器はMIC5353可変出力LDOです。基板上デバッグはMIC5353の帰還電圧を操作することによって基板目的対象部分へ供給する電圧出力を調節することができます。このハードウェア実装は概ね1.7～5.1Vの電圧範囲に制限されます。出力電圧がATTiny3227マイクロコントローラのハードウェア限度を決して超えないことを保証するためにデバッグファームウェアで付加的な出力電圧制限が構成設定されます。ATTiny3227 Curiosity Nanoの基板上デバッグで構成設定される電圧制限は1.8～5.5Vです。

情報: 工場既定目的対象電圧は3.3Vです。MPLAB® X IDEのプロジェクト特性はそれを変更することができます。例えば電力OFF/ON後も、目的対象電圧へのどの変更も持続します。分解能は5mVですが、調整プログラムによって10mVに制限されるかもしれません。

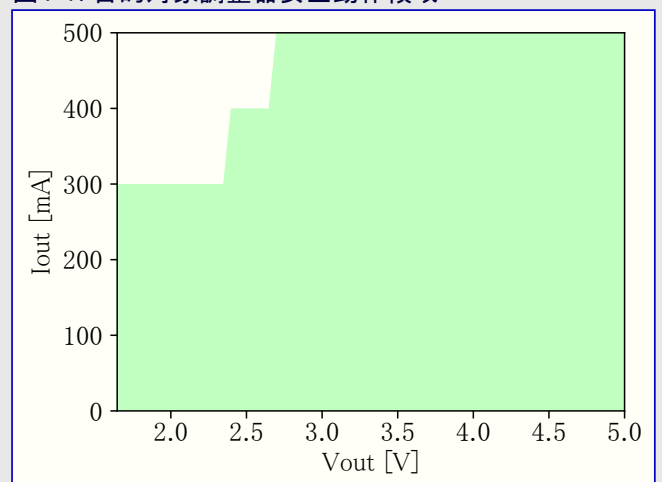
情報: MPLAB® X IDEでの電圧設定は直ちに基板へ提供されません。新しい電圧設定はプロジェクト操作盤タブで**Refresh Debug Tool Status**(デバッグツール状態更新)釦のクリックまたはプログラムメモリ読み書きのようなデバッグをアクセスする時に基板へ適用されます。

情報: 命令文字ファイルの基板へのドラッグ&ドロップで目的対象電圧を調整する簡単な任意選択があり、これは一般的な目的対象電圧一式を支援します。更なる詳細については「**特殊命令**」項をご覧ください。

MIC5353は500mAの最大電流負荷を支援します。これは小さな印刷回路基板(PCB)上に配置された小さな外周器のLDO調整器で、温度遮断条件は500mAより低い負荷で達し得ます。最大電流負荷は入力電圧、選んだ出力電圧、周囲温度に依存します。右図は5.1Vの入力電圧と23℃の周囲温度での調整器に対する安全動作領域を示します。

目的対象調整器の電圧出力は基板上デバッグによって継続的に監視(測定)されます。これが電圧設定値の上下100mVよりも大きい場合、異常状態が合図され、目的対象電圧調整器はOFFにされ、どの短絡回路状態も検出して処理します。VOFFピンのLow設定なしにVCC_TARGETを電圧設定監視窓の±100mVの外に移動させる外部電圧がVTGピンに急に印加された場合も検出して処理します。

図4-7. 目的対象調整器安全動作領域



情報: 基板上デバッグは目的対象電圧設定±100mVの監視窓を持ち、外部電圧がこの限度よりも低い場合、状態LEDが高速点滅します。外部電圧がこの限度を超える場合、基板上デバッグ状態LEDは点灯を続けます。外部電圧が取り去られると、状態LEDは基板上デバッグが新しい状況を検出して目的対象電圧調整器をONに戻すまで高速点滅を始めます。

4.4.3. 外部供給

基板上の目的対象調整器の代わりに外部電圧がATTiny3227 Curiosity Nanoに給電することができます。電圧OFF(VOFF)ピンを接地(GND)に短絡すると、基板上デバッグファームウェアは目的対象調整器を禁止し、VTGピンに外部電圧を印加しても安全です。

USBケーブルが基板上の**DEBUG**コネクタに接続されていない時にVTGピンに外部電圧を印加しても安全です。

VOFFピンは何時でもLowになる/することができ、これは基板上デバッグに対するピン変化割り込みによって検出され、それによって目的対象電圧調整器を制御します。

警告: GNDへのVOFF短絡なしでのVTGピンへの外部電圧印加は基板を恒久的な損傷にさせるかもしれません。

警告: VOFFピンへどんな電圧も印加しないでください。電源を許可するにはこのピンを浮かせてください。

警告: 基板上の基準移転器に対する絶対最大外部電圧は5.5Vで、ATTiny3227の標準動作条件は1.8～5.5Vです。より高い電圧の印加は基板を恒久的な損傷にさせるかもしれません。

情報: 基板上デバッグは基板に供給される電圧を監視します。VOFFピンがLowに引かれずに、外部供給が目的対象電圧設定±100mVを超えて異なる電圧を供給した場合、基板上デバッグは目的対象調整器を停止し、状態LEDを高速点滅を開始して異常状態を示します。一旦入力電圧が目的対象調整器設定±100mV内に戻ると、基板上デバッグは目的対象調整器をONにして状態LEDの点滅を停止します。

書き込み、デバッグ、データ流しは外部電源で未だ可能です。USBケーブルはデバッグと信号基準移転器に給電します。両調整器、デバッグ、基準移転器はUSBケーブルが取り去られると電力断にされます。



情報: ATtiny3227とそれの周辺機能によって消費される電力に加えて、USBケーブルを基板上の**DEBUG**コネクタに接続する時に基板上の基準移転器と電圧監視回路に給電するためにどの外部電源からも概ね100 μ Aが引き出されます。USBケーブルが接続されないと、基準移転器電圧ピンに供給するために多少の電流が使われ、これは概ね5 μ Aの最悪消費電流を持ちます。代表的な値は100nA位の低さでしょう。

4.4.4. 電源特例

本項は電源で起き得る殆どの問題を要約します。

目的対象電圧切断

目的対象部分が与えられた電圧で多すぎる電流を引き出す場合に設定した目的対象電圧に達しないことが起き得て、MIC5353調整器の過熱遮断安全機能を起動させます。これを避けるには目的対象部分の電流負荷を減らしてください。

目的対象電圧設定不到達

(4.4~5.25Vで指定される)USB入力電圧は与えられた電圧設定と消費電流に於いてMIC5353調整器の最大出力電圧を制限します。より高い出力電圧が必要とされる場合、より高い入力電圧のUSB電源を使うか、またはVTGピンで外部電源を使ってください。

目的対象電圧が設定と違う

これはVOFFピンをLowに設定することなく、VTGピンへ外部的に印加された電圧がこれを引き起こし得ます。目的対象電圧が電圧設定の上下100mVを超えて変動する場合、基板上デバッグがそれを検出し、内部電圧調整器が停止します。この問題を修正するにはVTGピンから印加した電圧を取り去ってください。基板上デバッグは新しい状況が検出されると、基板上調整器を許可します。目的対象電圧が設定の100mV未満の場合に**PS** LEDが高速点滅しますが、設定の100mV越えよりも大きい時に通常点灯することに注意してください。

非常に低い目的対象電圧またはなしで**PS** LEDが高速点滅

完全または部分的な短絡回路がこれを引き起こし得て、上の問題の特別な場合です。短絡回路を取り除いてください。基板上デバッグが基板上目的対象電圧調整器を再許可するでしょう。

目的対象電圧なしで**PS** LEDが点灯1

この状況は目的対象電圧が0.0Vに設定される場合に起きます。これを修正するには目的対象電圧を目的対象デバイスに対して指定された電圧範囲内の値に設定してください。

目的対象電圧なしで**PS** LEDが点灯2

この状況は**J200**や**J201**の電力ジャンパを切断し、目的対象電圧調整器を目的対象デバイスに対して指定された電圧範囲内の値に設定している場合に問題になり得ます。これを修正するには**J200**/**J201**用パッド間を線/橋渡しで半田付けするか、またはピンヘッダが実装されているなら**J201**にジャンパを追加してください。

VBUS出力電圧が低いまたは存在しない

VBUS電圧が低いまたは失っている場合、その理由はおそらくVBUSでの高電流引き出しで、U200(MIC2008)によって設定した電流制限が作動されてVBUSを完全に遮断します。この問題を修正するにはVBUSピンでの消費電流を減らしてください。

4.4.5. 低電力測定

ATtiny3227への電力はシルクスクリーンで**POWER**と記された100milピンヘッダ(**J201**)を通して基板上電源とVTGピンから来ます。ATtiny3227と基板に接続された他の周辺機能の消費電力を測定するには**目的対象電力切断帯(J201)**を切断し、それに渡して電流計を接続してください。



助言: 電流計の簡単な接続のために**目的対象電力切断帯(J201)**配置パターンに100milピンヘッダを半田付けすることができます。一旦電流計がもはや必要とされなくなれば、このピンヘッダに短絡ジャンパを置いてください。

可能な最低消費電力を測定するにはこれらの手順に従ってください。

1. 鋭利な工具で**POWER**切断帯を切断してください。
2. 配置パターンに1×2 100milピンヘッダを半田付けしてください。
3. ピンヘッダに電流計を繋げてください。
4. 次のようなファームウェアを作ってください。
 - a. 基板上デバッグに接続されるどの入出力もHi-Z(トライステート)にしてください。
 - b. マイクロコントローラを最低電力の休止状態に設定してください。
5. このファームウェアをATtiny3227に書き込んでください。

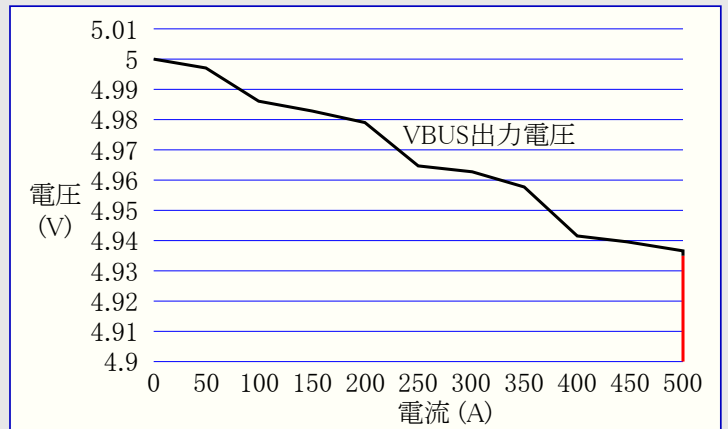


情報: 基板上の基準移転器は例え使われない時でも少量の電流を引き込みます。各基準移転器は最大2 μ Aの漏れ電流を持ちます。従って、基板上の5つの基準移転器に対する最悪最大引き込み電流は10 μ Aです。入出力ピンをHi-Z(トライステート)に保つことによって基準移転器に接続される入出力ピンを通る漏れ電流を防いでください。基板上デバッグに接続される全ての入出力は「[基板上デバッグ接続](#)」で一覧にされます。基板上の基準移転器は完全に切断することができ、「[基板上デバッグ切断](#)」で記述されるように漏れを防ぎます。

4.4.6. VBUS出力ピン

ATtiny3227 Curiosity Nanoは5V供給が必要な外部部品に給電するのに使うことができるVBUS出力ピンを持ちます。VBUS出力ピンは残りの電源と同じ上昇率と電流制限での始動遅延によって保護されます。副作用はより高い負荷電流でのVBUS出力での電圧降下です。右図は電圧対VBUS出力電圧対VBUS出力の負荷電流を示します。

図4-8. VBUS出力電圧対電流



5. 改訂履歴

ハードウェアと文書の改訂履歴

5.1. ハードウェア改訂履歴と既知の問題

この使用者の手引きは基板の入手可能な最新版についての情報を提供します。以下の項は既知の問題、旧版の改訂履歴、旧版が最新版とどう違うのかについての情報を含みます。

5.1.1. 製品IDと改訂の識別

ATtiny3227 Curiosity Nano基板の改訂と製品識別子を見つけるには2つの方法、MPLAB® X IDEのキット ウィンドウを利用して、またはPCBの裏側の張り紙を見ることによってのどちらかがあります。

キット ウィンドウはMPLAB® X IDEが走行しているコンピュータにATtiny3227 Curiosity Nanoを接続する時に現れます。

キット詳細下で一覧にされる通番の最初の6桁が製品識別子と改訂を含みます。



助言: 閉じられている場合、キット ウィンドウはMPLAB X IDEでメニュー バーのWindow(ウィンドウ)⇒Kit Window(キット ウィンドウ)を通して開くことができます。

同じ情報はPCBの裏側の張り紙で見つけられます。張り紙のデータ マトリックス符号は製品識別子の02-01280、改訂、通番を持つ文字列を含みます。

データ マトリックス符号の文字列は以下の形式を持ちます。

```
“nnnnnnnnrTssssssss”
n = 製品識別子
r = 改訂
s = 通番
```

5.1.2. 改訂1

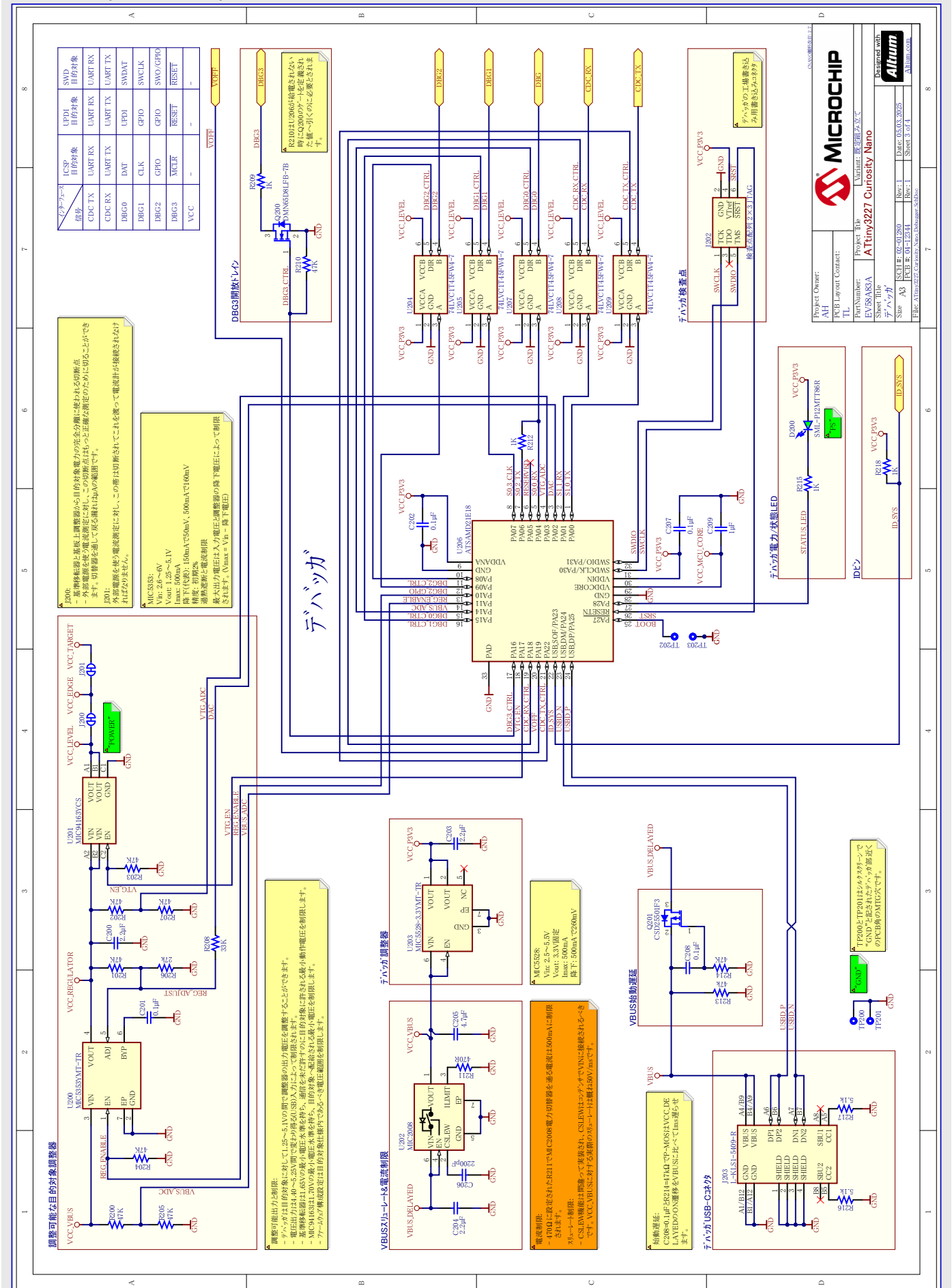
改訂1は初公開基板版です。これは実装されたATtiny3227改訂Aを持ちます。

既知の問題: **U208** – MIC2008に対するスルーレート制限が間違えて実装されています。CSLEWピンのコンデンサが現在GNDに接続されていて、これは**VCC_P5V0**に対するスルーレートを概ね50V/msにさせます。正しい実装は意図するスルーレート制限を達成するため、代わりにコンデンサがVINとCSLEWのピン間に接続されるかもしれません。

6. 文書改訂履歴

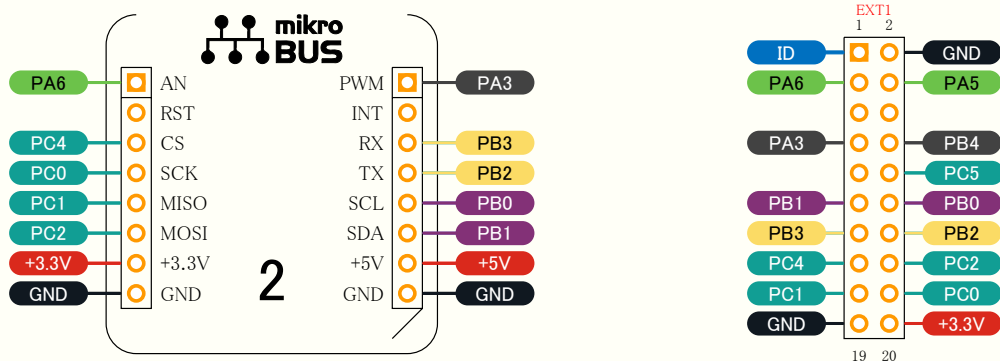
文書改訂	日付	注釈
A	2025年3月	初版文書公開

図7-1. ATtiny3227 Curiosity Nano回路図 (2/2)



ATtiny3227

Curiosity Nano



7.4. 外部マイクロ コントローラ書き込み

外部ハードウェア上のマイクロ コントローラ の書き込みとデバッグを行うのにATtiny3227 Curiosity Nanoの基板上デバッグを使ってください。

7.4.1. 支援デバイス

Microchip MPLAB X IDEと基板上デバッグでUPDIインターフェースを持つ全ての外部AVRマイクロ コントローラに書き込んでデバッグすることができます。

Microchip MPLAB X IDEと基板上デバッグでCuriosity Nano基板を持つ外部SAM、PIC16、PIC18のマイクロ コントローラに書き込んでデバッグすることができます。

ATtiny3227 Curiosity NanoはMPLAB X IDEで外部ATtiny3227マイクロ コントローラに書き込んでデバッグすることができます。

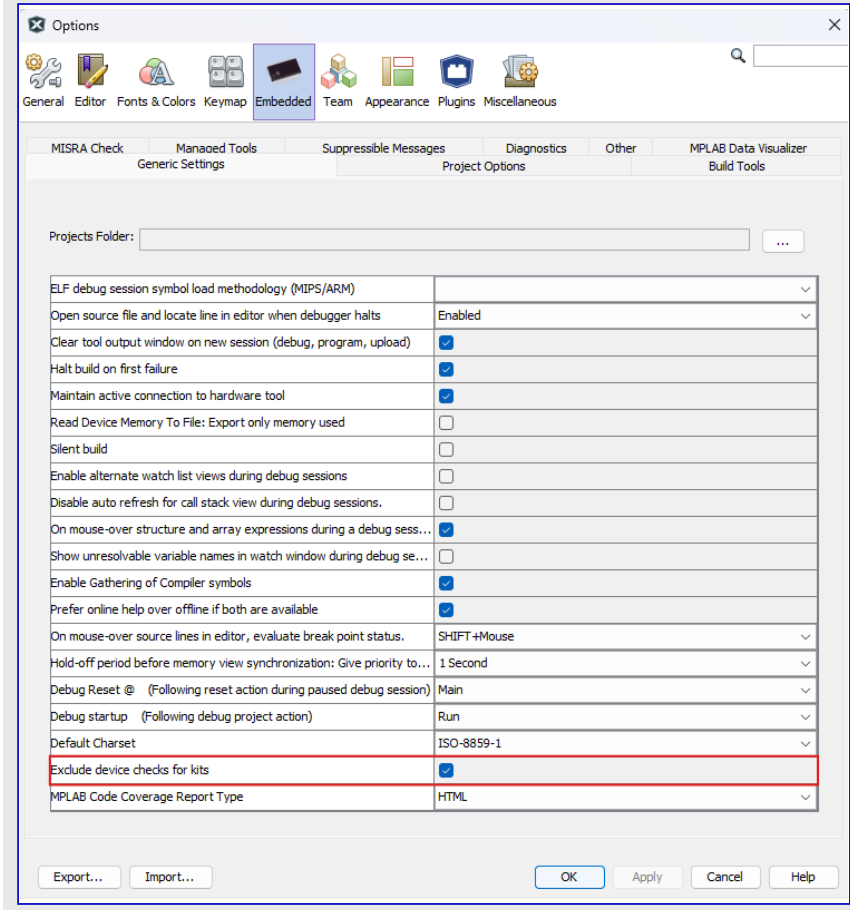
7.4.2. ソフトウェア構成設定

基板上に実装された同じデバイスに書き込んでデバッグするのにソフトウェア構成設定は必要とされません。

基板上に実装されたのとは違うマイクロ コントローラに書き込んでデバッグするには、デバイスとプログラム インターフェースの独立した選択を許すようにMicrochip MPLAB X IDEを構成設定してください。

1. 応用上部のメニュー システムを通して**Tools(ツール)⇒Options(任意選択)**へ誘導してください。
2. **Options(任意選択)**ウィンドウで**Embedded(組み込み)⇒Generic Settings(一般的な設定)**区分を選んでください。
3. **Exclude device checks for kits(キットに対するデバイス検査を除外)**任意選択をチェックしてください。

図7-5. キットに対するデバイス検査を除外



情報: Microchip MPLAB X IDEは**Exclude device checks for kits**設定がチェックされると、基板上デバッグによって支援されないマイクロ コントローラとインターフェースも含め、どのマイクロ コントローラとインターフェースも選ばれることを許します。

7.4.3. 外部マイクロ コントローラへの接続

以降の図と表は外部マイクロ コントローラに書き込んでデバッグするために書き込みとデバッグの信号を接続する場所を示します。基板上デバッグは外部ハードウェアに電力を供給するか、またはその基準移転器用の参照基準として外部電圧を使うことができます。図4-5で電源についてもっとお読みください。

基板上デバッグと基準移転器は書き込みとデバッグに使われるデータとクロックの信号(DBG0、DBG1、DBG2)を積極的に駆動します。PIC®マイクロ コントローラをデバッグするのにICSP™のデータとクロックの信号にプルダウン抵抗が必要とされます。他の全てのインターフェースはプルアップやプルダウンの抵抗があってもなくても機能します。

DBG3は開放ドレイン接続で、機能するのにプルアップ抵抗を必要とします。



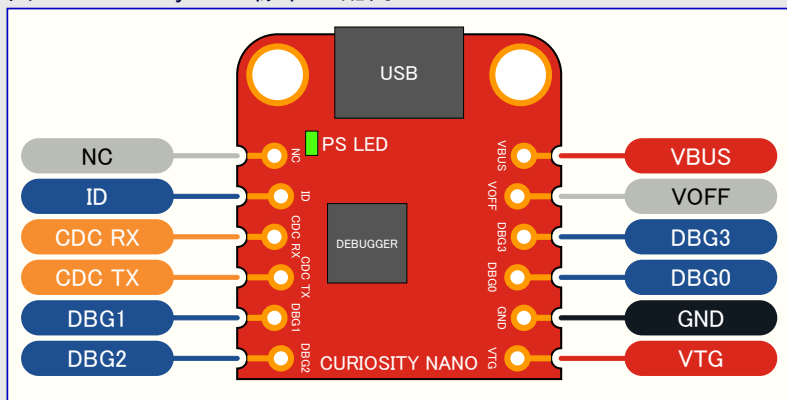
留意: ・ GNDとVTGを外部マイクロ コントローラに接続してください。

- ・ 外部ハードウェアが電源を持つ場合、VOFFピンをGNDに繋げてください。
- ・ PICマイクロ コントローラのデバッグを支援するにはICSPのデータとクロックの信号(DBG0とDBG1)にプルダウン抵抗があることを確実にしてください。

表7-1. プログラム/デバッグ インターフェース

Curiosity Nanoピン	UPDI	ICSP™	SWD
DBG0	UPDI	DATA	SWDIO
DBG1	—	CLK	SWCLK
DBG2	—	—	—
DBG3	—	MCLR	RESET

図7-6. Curiosity Nano標準ピン配列



7.5. 外部デバッグ接続

基板上デバッグがあるとは言え、ATtiny3227の書き込みとデバッグのためにATtiny3227 Curiosity Nanoに外部デバッグを直接的に接続することができます。基板上デバッグは積極的に使われない時にATtiny3227と基板端に接続された全てのピンをHi-Zに保ちます。従って、基板上デバッグはどの外部デバッグ ツールにも干渉しません。

図7-7. ATtiny3227 Curiosity NanoへのMPLAB® PICKit™5実装デバッグ/書き込み器接続

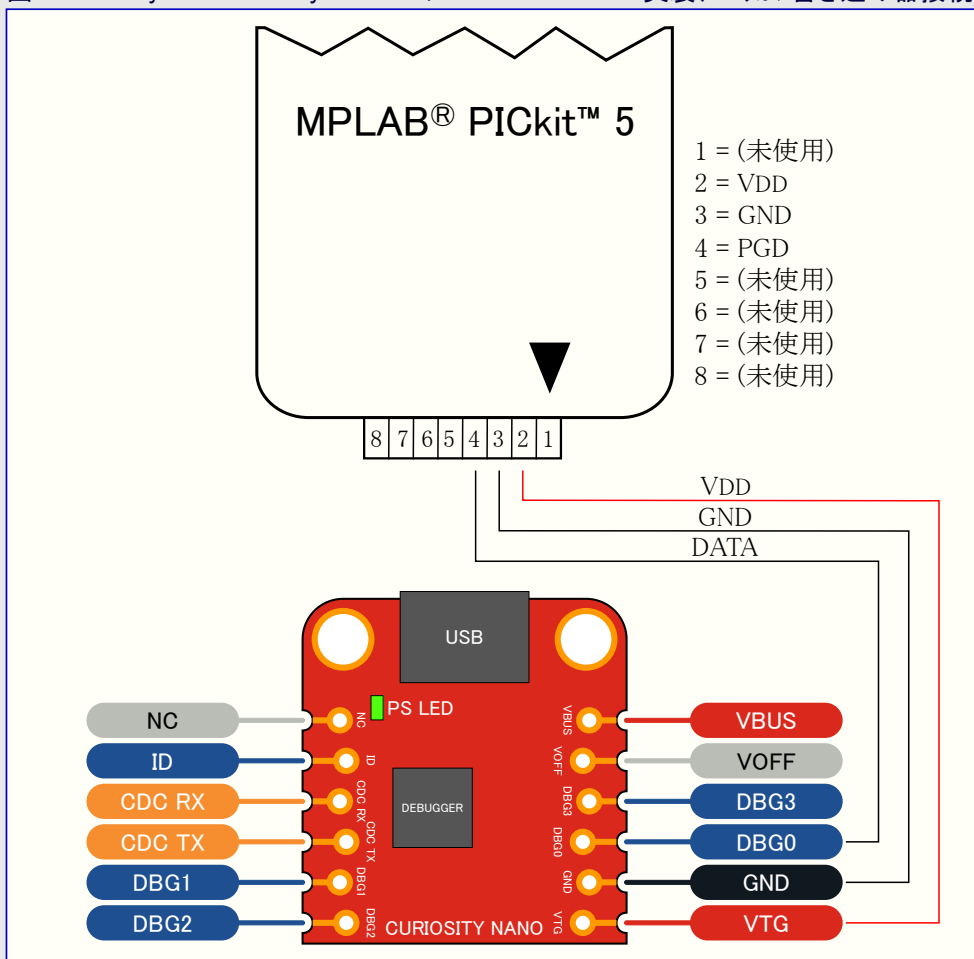
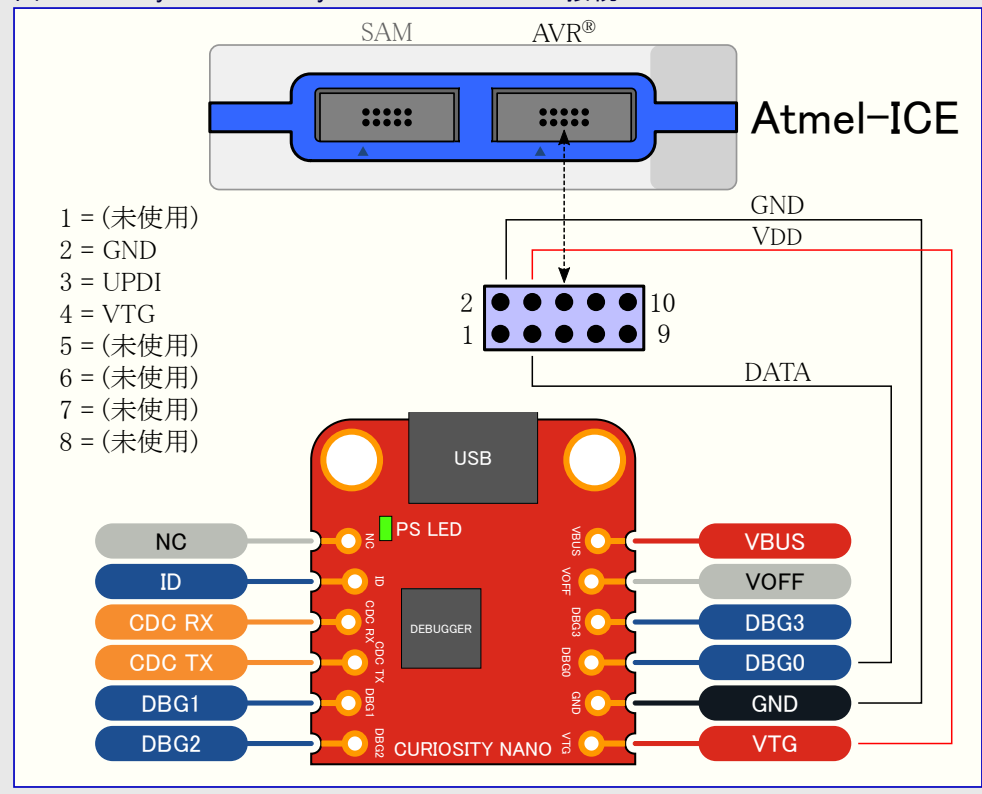


図7-8. ATtiny3227 Curiosity NanoへのAtmel-ICE接続



警告 外部デバッガと基板上デバッガ間の競合を避けるため、外部ツールが活性な間にMPLAB® X IDEを通す基板上デバッガでのどの書き込み/デバッグの操作開始や大容量記憶の書き込みも行わないでください。

7.6. 基板上デバッガ切断

基板上デバッガと基準移転器はATtiny3227から切断することができます。

電源構成図(図4-5)はデバッガとATtiny3227マイクロコントローラ間の全ての接続を示します。信号名は基板の裏側でシルクスクリーンで印刷されます。

デバッガを切断するには図4-6の切断帯を切ってください。

注意: 基板上デバッガに対するGPIO切断帯の切断は機能から仮想シリアルポート、書き込み、デバッグ、データ流しを阻止します。電力切断帯の切断は基板上電源を切断します。

助言: 切ったどの接続も半田を使うことによって再接続してください。代わりに0402の0Ω抵抗を実装してください。

助言: デバッガが切断された時に外部デバッガを穴に接続することができます。「外部デバッガ接続」は外部デバッガ接続方法を記述します。

7.7. IAR™でとの開始に際して

AVR®用IAR Embedded Workbench®はGCCに基づかない専売の高効率コンパイラです。ATtiny3227 Curiosity Nanoの書き込みとデバッグはAtmel-ICEインターフェースを使ってAVR用IAR™ Embedded Workbenchで支援されます。書き込みとデバッグの作業を始めるにはプロジェクトでいくつかの初期設定が構成設定されなければなりません。

以下の手順は書き込みとデバッグのためにプロジェクトの準備を整える方法を説明します。

1. 構成設定されるべきプロジェクトが開かれているのを確実にしてください。プロジェクトに対して**OPTIONS**ダイアログを開いてください。
2. **General Options**区分で、**Target**タブを選んでください。プロジェクト用のデバイス、または一覧になれば、**図7-9.**で示されるようにデバイスのコアを選んでください。
3. **Debugger**区部で、**Setup**タブを選んでください。**図7-10.**で示されるようにドライバとして**Atmel-ICE**を選んでください。
4. **Debugger**⇒**Atmel-ICE**区部で、**Atmel-ICE 1**タブを選んでください。**図7-11.**で示されるようにインターフェースとして**UPDI**を選んでください。任意選択で**UPDI**周波数を選んでください。

情報: (段階4. で言及した)デバッグ ポートの選択が灰色で操作不可の場合、インターフェースが予め選ばれており、この構成設定段階を飛ばすことができます。

図7-9. 目的対象デバイス選択

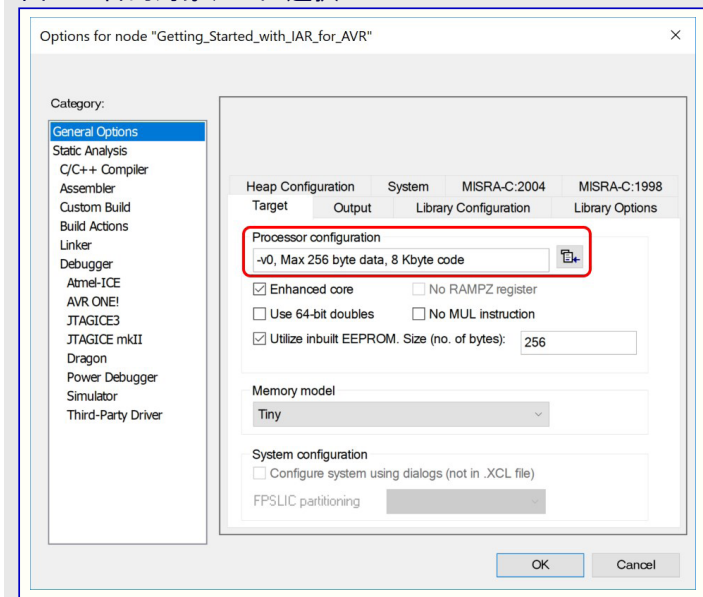


図7-10. デバッグ選択

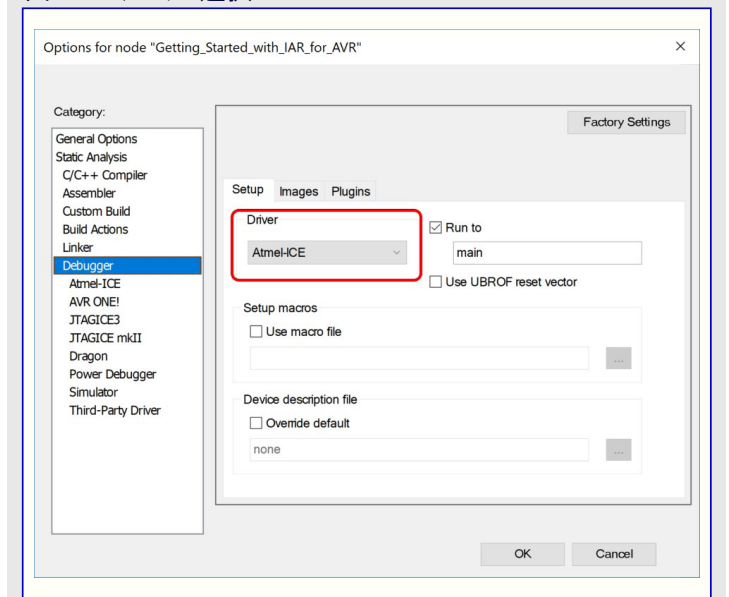
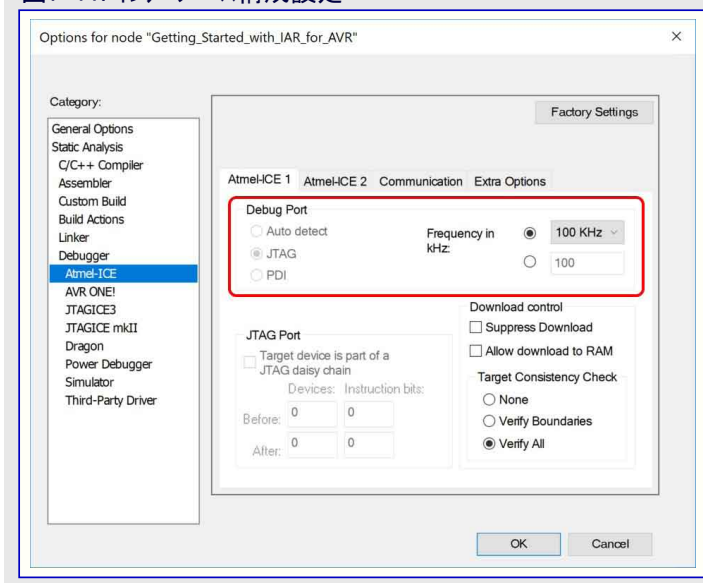


図7-11. インターフェース構成設定



Microchip情報

商標

“Microchip”の名称とロゴ、“M”のロゴ、それと他の名称、ロゴ、商標は米国や他の国に於けるMicrochip Technology Incorporatedまたはその系列会社や子会社の登録または未登録の商標です(“Microchip商標”)。Microchip商標に関する情報は<https://www.microchip.com/en-us/about/legalinformation/microchip-trademarks>で見つけることができます。

法的通知

この刊行物と契約での情報は設計、試験、応用とのMicrochip製品の統合を含め、Microchip製品でだけ使えます。他の何れの方法でのこの情報の使用はこれらの条件に違反します。デバイス応用などに関する情報は皆さまの便宜のためにだけ提供され、更新によって取り換えられるかもしれません。皆さまの応用が皆さまの仕様に合致するのを保証するのは皆さまの責任です。追加支援については最寄りのMicrochip営業所にお問い合わせ頂くか、www.microchip.com/en-us/support/design-help/client-support-servicesで追加支援を得てください。

この情報はMicrochipによって「現状そのまま」で提供されます。Microchipは非侵害、商品性、特定目的に対する適合性の何れの黙示的保証やその条件、品質、性能に関する保証を含め、明示的にも黙示的にもその情報に関連して書面または表記された書面または黙示の如何なる表明や保証もしません。

如何なる場合においても、Microchipは情報またはその使用に関連するあらゆる種類の間接的、特別的、懲罰的、偶発的または結果的な損失、損害、費用または経費に対して責任を負わないものとします。法律で認められている最大限の範囲で、情報またはその使用に関連する全ての請求に対するMicrochipの全責任は、もしあれば、情報のためにMicrochipへ直接支払った料金を超えないものとします。

生命維持や安全応用でのMicrochipデバイスの使用は完全に購入者の危険性で、購入者はそのような使用に起因する全ての損害、請求、訴訟、費用からMicrochipを擁護し、補償し、免責にすることに同意します。他に言及されない限り、Microchipのどの知的財産権下でも暗黙的または違う方法で許認可は譲渡されません。

Microchipデバイス コード保護機能

Microchip製品での以下のコード保護機能の詳細に注意してください。

- Microchip製品はそれら特定のMicrochipデータシートに含まれる仕様に合致します。
- Microchipは動作仕様内で意図した方法と通常条件下で使われる時に、その製品系統が安全であると考えます。
- Microchipはその知的所有権を尊重し、積極的に保護します。Microchip製品のコード保護機能を侵害する試みは固く禁じられ、デジタルミレニアム著作権法に違反するかもしれません。
- Microchipや他のどの半導体製造業者もそのコードの安全を保証することはできません。コード保護は製品が”破ることができない”ことを当社が保証するということを意味しません。コード保護は常に進化しています。Microchipは当社製品のコード保護機能を継続的に改善することを約束します。

日本語© HERO 2025.

本使用者の手引きはMicrochipのATtiny3227 Curiosity Nano使用者の手引き(DS50003854A-2025年3月)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。