

Application Note

C2000™ MCU JTAG コネクティビティのデバッグ

Matthew Pate, Matt Kukucka, Benjamin Collier, and Nabil Saheb

概要

JTAG (ジョイントテストアクショングループ) プロトコルは、製品開発、エミュレーション、アプリケーションのデバッグ時にマイコン (MCU) と通信する主要な手段です。テキサス インスツルメンツ (TI) の C2000™ デバイスはいずれも JTAG エミュレーションをサポートしており、controlCARD や LaunchPad™ 開発キットなどの C2000 評価製品はオンボード JTAG エミュレーションを内蔵しています。このアプリケーション レポートでは、JTAG 実装の概要を簡単に説明し、Code Composer Studio™ ソフトウェアを使用する際に一般的に生じる JTAG 接続エラーの解決に使用する手順を説明します。

JTAG を使用して C2000™ デバイスと通信する際に問題が発生する場合、TI の E2E™ フォーラムなどのプラットフォームで支援を受ける前に、このデバッグ ガイドに従うことを強く推奨いたします。スクリーニング プロセスの一環として、TI のサポート担当者はこのガイドを使用して皆さんの進捗状況について質問します。

目次

1 JTAG とは？	2
2 一般的な JTAG デバッグ プロンプト	2
3 LaunchPad™ 開発キットと controlCARD のデバッグ手順	3
3.1 LaunchPad™ 開発キット	3
3.2 controlCARDs	3
4 一般的なエラー コード	5
4.1 一般的なエラー コード	5
5 JTAG チェーン内の複数のデバイス	9
6 非侵襲的デバッグ	10
7 JTAG TAP の無効化とリセット	12
8 JTAG コネクティビティのデバッグ フロー	14
8.1 全体的なデバッグ フロー	15
8.2 高電圧絶縁チェック フロー	15
8.3 メイン JTAG デバッグ フロー	16
9 フロー ステップの詳細情報	17
9.1 絶縁の事前チェック フロー	17
9.2 JTAG デバッグ フロー	17
10 参考資料	21
11 改訂履歴	22

商標

C2000™, LaunchPad™, Code Composer Studio™, E2E™, BoosterPack™, and Piccolo™ are trademarks of Texas Instruments.

Microsoft® and Windows® are registered trademarks of Microsoft Corporation.

すべての商標は、それぞれの所有者に帰属します。

1 JTAG とは？

JTAG は、プリント基板 (PCB) のバウンダリ スキャン テストに関する業界標準を形成したグループにちなんで名付けられました。JTAG の取り組みの結果は、後に IEEE (Institute of Electrical and Electronics Engineers: 米国電気電子学会) 規格: [IEEE Std 1149.1](#) に体系化されました。この規格は導入されると直ちに非常に広く使用されるようになり、オンチップ テスト アクセス ポート (TAP) の実装を含む、いくつかの追加規格が生まれました。これにより、JTAG は組込みシステム開発、デバッグ、テストに最も適応された手段となりました。JTAG はほとんどのシステムで以下の 5 ピンを実装しています:

- テスト データ入力 (TDI)
- テスト データ出力 (TDO)
- テスト クロック (TCK)
- テスト モード 選択 (TMS)
- テスト リセット (TRSTn)

カスタム ボード 設計でこれらのピンに必要なバイアスについては、デバイス固有のデータシートまたは TI のリファレンス デザインを参照してください。

2 一般的な JTAG デバッグ プローブ

表 2-1 は C2000 エコシステムの一般的なデバッグ プローブを示しています。大半の C2000 デバイスでは、より新しく低コストな XDS110 (www.ti.com/tool/TMDSEMU110-U) の使用が推奨されます。

表 2-1. C2000™ マイコンで使用する一般的な JTAG デバッグ プローブ

	XDS100v1 と XDS100v2	XDS110	XDS200	XDS560v2
価格/速度	+	+	++	+++
特長	<ul style="list-style-type: none"> • USB インターフェイス • フラッシュ プログラミング • 従来の多くの C2000 評価基板に内蔵されるデバッグ プローブ 	<ul style="list-style-type: none"> • XDS100 の後継 • USB 2.0 インターフェイス • フラッシュ プログラミング • 絶縁型デバッグ (XDS110ISO) • 新しい多くの TI C2000 評価基板に内蔵されるデバッグ プローブ 	<ul style="list-style-type: none"> • USB 2.0 ハイスピード インターフェイス (480Mbps) • フラッシュ プログラミング • コアおよびシステムのトレース 	<ul style="list-style-type: none"> • USB インターフェイス • フラッシュ プログラミング • コードトレース オプション (高帯域幅)
メーカー	<ul style="list-style-type: none"> • テキサス・インスツルメンツ • Blackhawk 	<ul style="list-style-type: none"> • テキサス・インスツルメンツ 	<ul style="list-style-type: none"> • Blackhawk 	<ul style="list-style-type: none"> • テキサス・インスツルメンツ • Blackhawk

『[C2000™ リアルタイム マイコンの設計 & 開発](#)』もご覧ください。

3 LaunchPad™ 開発キットと controlCARD のデバッグ手順

LaunchPad™ 開発キットと controlCARD はいずれも多少異なりますが、LaunchPad または controlCARD に接続できない場合に確認すべき一般的な点がいくつかあります。

3.1 LaunchPad™ 開発キット

LaunchPad に接続できない一般的な理由はいくつかあります。これらの手順のうちどれが関連するかを判断するには、「[CCS で JTAG 接続をテストする方法](#)」の説明に従い、LaunchPad に「テスト接続」します。以下の手順を実行する前に、BoosterPack™ プラグイン モジュール、または他の外部ハードウェアを LaunchPad から取り外してください。

- 「テスト接続」のログに、「テスト接続」がデバッグ プローブと通信できないという記載がある場合は、次の手順を実行します：
 - USB ケーブルを交換し、「ターゲットの構成」ファイルで正しいデバッグ プローブが選択されていることを確認します。
 - LaunchPad のハードウェア スイッチとジャンパを確認して、JTAG 接続を無効化する可能性のあるものがあるかどうかを確認します。この情報は、LaunchPad ユーザー ガイドに記載されており、通常、LaunchPad の TI.com ストア ページにリンクされています。
 - それでも「テスト接続」のログに、「テスト接続」がデバッグ プローブと通信できないという記載がある場合は、LaunchPad が損傷している可能性があります。デバッグ プローブが PC のデバイス マネージャーに表示されない場合、この可能性が特に高くなります。
- 「テスト接続」のログに、「テスト接続」がビットを循環できない、または IR か DR パスが破損しているという記載がある場合は、次の手順を実行してください：
 - LaunchPad のハードウェア スイッチを確認して、JTAG 接続を無効化しているものがあるかどうかを確認します。この情報は、LaunchPad ユーザー ガイドに記載されており、通常、LaunchPad の TI.com ストア ページにリンクされています。
 - 一部の LaunchPad は、2 ピンの cJTAG、または 4 ピンの JTAG でのみ動作する設計を採用しています。これはターゲット構成の **Advanced** (詳細) タブで設定します。この設定が正しいことを確認してください。
 - それでも「テスト接続」のログに、「テスト接続」がビットを循環できないという記載がある場合は、LaunchPad が損傷している可能性があります。
- 「テスト接続」のログに、テストに合格していても、デバイスにコードをロードできないという記載がある場合は、次の手順を実行してください：
 - LaunchPad のスイッチを使用してデバイスをブート待機モードにします。[手動起動](#)の手順で接続できる場合は、フラッシュ済みコードが接続を妨げていた可能性があります。
 - それでも接続が確立できない場合や、接続してコードをロードできない場合は、デバイスがロックされている可能性があります。

3.2 controlCARDs

ランチパッド開発キットと同様に、controlCARD と接続できない一般的な理由がいくつかあります。これらの手順のうち、どれが関連するかを判断するには、「テスト接続」の「[CCS で JTAG 接続をテストする方法](#)」を参照してください。次の手順を実行する前に、controlCARD から外部ハードウェアを取り外してください。

- ホスト PC がデバッグ プローブと通信できないことが「テスト接続」のログに記載されている場合は、オンボードのデバッグ プローブを使用する際に以下の手順を試してください：
 - USB ケーブルを交換して、「ターゲットの構成」ファイルで正しいデバッグ プローブが選択されていることを確認します。
 - JTAG 信号がオンボードのデバッグ プローブから供給されるかドッキング ステーションから供給されるかを選択する、controlCARD のハードウェア スイッチを確認します。この情報は、controlCARD ユーザー ガイドに記載されており、通常、controlCARD の TI.com ストア ページにリンクされています。
 - 「テスト接続」のログにデバッグ プローブとの通信が確立されていないという記載が残る場合は、controlCARD が損傷している可能性があります。

2. 「テスト接続」のログにデバッグプローブとの通信がまだ確立されていないという記載がある場合、スタンドアロン デバッグプローブを使用するには、次の手順を実行してください:
 - a. USB ケーブルをデバッグプローブに変更してみます。
 - b. 「テスト接続」のログにデバッグプローブとの通信が確立されていないという記載が残る場合は、デバッグプローブが損傷している可能性があります。
3. 「テスト接続」のログにデバッグプローブがビットを循環できない、または IR か DR パスが破損しているという記載がある場合は、次の手順を実行してください:
 - a. controlCARD のハードウェア スイッチを確認して、JTAG 接続を無効化しているものがあるかどうかを確認します。この情報は、controlCARD ユーザー ガイドに記載されており、通常、controlCARD の TI.com ストア ページにリンクされています。
 - b. 「テスト接続」のログにデバッグプローブがビットを循環できないという記載がある場合は、controlCARD が損傷している可能性があります。
4. 「テスト接続」のログに、テストに合格していてもデバイスにコードをロードできないという記載がある場合は、次の手順を実行してください:
 - a. controlCARD のスイッチを使用してデバイスをブート待機モードにします。手動起動の手順で接続が成功する場合は、フラッシュされたコードの何かが接続を妨げていたエラーである可能性が高いです。
 - b. それでも接続が確立できない場合や、接続が成功してもコードをロードできない場合は、デバイスがロックされている可能性があります。

4 一般的なエラーコード

このセクションでは、「テスト接続」の動作、接続の試行、またはデバイスにコードをロードしようとするときに Code Composer Studio が表示する、一般的なエラーコードをいくつか示します。対応するデバッグ手順に従っても問題が解決されない場合は、同じエラーコードの他のスレッドについて、TI の [E2E™ フォーラム](#)を検索してください。

「テスト接続」については [How to test your JTAG connection with CCS](#) のリンクを参照してください。また、デバイスを手動で接続してコードをロードする方法については「[手動起動](#)」の説明を参照してください。

エラーコードをデバッグする際は以下のフローを使用します：

1. 「テスト接続」により JTAG 信号が正しく接続されていることを確認
2. 手動起動のための前の手順を使用してデバイスに接続
3. デバイスにコードをロード

4.1 一般的なエラーコード

表 4-1 は一般的なエラーコードと関連デバッグ手順を示しています。

表 4-1. 一般的なエラーコード表

エラーメッセージ	デバッグ手順
<p>This error is generated by TI's USCIF driver or utilities. The value is '-233' (0xffffffff17). The title is 'SC_ERR_PATH_BROKEN'. The explanation is: The JTAG IR and DR scan-paths cannot circulate bits, they may be broken. An attempt to scan the JTAG scan-path has failed. The target's JTAG scan-path appears to be broken with a stuck-at-ones or stuck-at-zero fault.</p>	<p>このエラーは通常 JTAG 信号が正しく接続されていない場合に通知されますが、信号品質の低下によっても発生する可能性があります。ターゲットとプローブ間の JTAG 接続が、データシートの推奨事項に準拠していることを確認してください。</p> <p>これは、実行時に TDI と TDO を汎用入出力 (GPIO) として使用する際に、デバッグプローブが 4 ピン JTAG を使用している場合にも発生します。</p> <p>これはブルアップ抵抗またはブルダウン抵抗が強すぎる場合にも発生する可能性があるため、デバッグ時に削除を試みてください。データシートの推奨事項と比較してください。</p> <p>このエラーは、デバイスが正常に起動していない場合に発生することがあります。オシロスコープで電源レールと XRSn を観察して、デバイスが正常に起動し、XRSn が High になることを確認します。ウォッチドッグのせいで、プログラムされていないデバイスでは XRSn が周期的に再起動することが予想されます。</p> <p>ハードウェア設計ガイドの「バッファされたケース」セクションを参照してください。</p>

表 4-1. 一般的なエラー コード表 (続き)

エラー メッセージ	デバッグ手順
Error connecting to the target: (Error -1015 @ 0x0) Device is not responding to the request. Device may be locked, or the debug probe connection may be unreliable. Unlock the device if possible (e.g. use wait in reset mode, and power-cycle the board.) If error persists, confirm configuration and/or try more reliable JTAG settings (e.g. lower TCLK).	<ol style="list-style-type: none"> 1.ターゲット構成の「テスト接続」に合格することを確認します。接続に失敗する場合は、そのエラーコードの手順に従います。 2.デバイスをブート待機モードに設定します。 3.手動起動の指示に従ってデバイスに接続します。 4.メモリ ブラウザで PARTID を読み取りできることを確認します。 5.デバイスのプログラムをもう一度実行します。 6.上記の手順を実行してもエラーがクリアされない場合は、次の点を確認してください: デバイス上にパスワードの場所がありますか? オンチップ フラッシュ ツール の設定? RAM のみをプログラムすることは可能ですか? XRSn が適切にアサートされていますか?
Error connecting to the target:(Error -1135 @ 0x0)The emulator reported an error. Confirm emulator configuration and connections, reset the emulator, and retry the operation.	このエラー メッセージは通常、ハードウェア関連の問題が原因で発生します。VDDIO/VDD が適切に供給されていることを確認します。XRSn が High にアサートされて、デバイスがリセットから正常に復帰したことを示していることを確認します。長い JTAG ケーブル、不適切な終端、または過剰な容量も、JTAG のシグナルの整合性を低下させる可能性があります。
Error connecting to the target: (Error -151 @ 0x0) One of the FTDI driver functions used during the connect returned bad status or an error. The cause may be one or more of: no XDS100 is plugged in, invalid XDS100 serial number, blank XDS100 EEPROM, missing FTDI drivers, faulty USB cable. Use the xds100serial command-line utility in the 'common/uscif' folder to verify the XDS100 can be located. (Emulation package 8.0.903.2)	<ol style="list-style-type: none"> 1.ターゲット構成ファイルをチェックして、正しいデバッグ プロブが選択されていることを確認します。 2.PC のデバイス マネージャーにデバッグ プロブが表示されていることを確認します。 3.USB ケーブルを交換、または別のデバッグ プロブを試して、使用中のプロブが損傷していないことを確認してください。
Trouble Reading Register PC: (Error -1156 @ 0x0) Device may be operating in low-power mode. Do you want to bring it out of this mode? Choose 'Yes' to force the device to wake up and retry the operation. Choose 'No' to retry the operation without waking the device.	<ol style="list-style-type: none"> 1.ターゲット構成の「テスト接続」に合格することを確認します。接続に失敗する場合は、そのエラーコードの手順に従います。 2.デバイスをブート待機モードに設定します。 3.手動起動の指示に従ってデバイスに接続します。 4.メモリ ブラウザで PARTID を読み取りできることを確認します。 5.デバイスのプログラムをもう一度実行します。 6.上記の手順を実行してもエラーがクリアされない場合は、次の点を確認してください: デバイス上にパスワードの場所がありますか? オンチップ フラッシュ ツール の設定? RAM のみをプログラムすることは可能ですか? XRSn は適切にアサートされていますか?
C28xx_CPU1: Error: (Error -1044 @ 0x0) The debug probe reported an error. Confirm debug probe configuration and connections, reset the debug probe, and retry the operation.	ターゲット マイコンの電源を再投入せずに、デバッグ プロブの電源を再投入を試行します。低クロック周波数など、より信頼性の高い JTAG 設定を試してください。

表 4-1. 一般的なエラー コード表 (続き)

エラー メッセージ	デバッグ手順
<p>The value is '-230' (0xfffff1a). The title is 'SC_ERR_PATH_MEASURE'. The explanation is: The measured lengths of the JTAG IR and DR scan-paths are invalid. This indicates that an error exists in the link-delay or scan-path.</p>	<p>ターゲット構成ファイル (*.ccxml) で目的の JTAG/cJTAG モードが構成されていることを確認します。特定の TI 評価基板 (LaunchPad/controlCARD) がサポートできるのは、ボード設計に応じて 2 ピンまたは 4 ピン JTAG のみであることに注意してください。評価基板のユーザー ガイドを参照してご確認ください。 このエラーは通常、JTAG 信号の信号品質が低い場合にも通知されます。ターゲットとプローブ間の JTAG 接続をデータシートで確認してください。TCK 周波数を低くして、パターン長を確認してください。 ハードウェア設計ガイドの「バッファされたケース」セクションを参照してください。</p>
<p>C28xx_CPU1: Trouble Setting Breakpoint with the Action "Continue or Finish Stepping" at 0x83146: (Error -1066 @ 0x83146) Unable to set/clear requested breakpoint. Verify that the breakpoint address is in valid memory. (Emulation package 9.13.0.00201) C28xx_CPU1: Breakpoint Manager: Retrying with a AET breakpoint</p>	<p>一部のデバイスでは、ハードウェア ブレークポイントを使用する必要があるため、フラッシュからコードを実行する際に 1 つのブレークポイントのみが許可されます。使用可能なハードウェア ブレークポイント数については、デバイスのデータシートを確認してください。</p>
<p>Error Initializing Emulator: (Error -2083 @ 0x0) Unable to communicate with the debug probe. Confirm debug probe configuration and connections, reset the debug probe, and retry the operation.</p>	<ol style="list-style-type: none"> 1.ターゲット構成ファイルをチェックして、正しいデバッグプローブが選択されていることを確認します。 2.PC のデバイス マネージャーにデバッグプローブが表示されていることを確認します。 3.USB ケーブルを交換、または別のデバッグプローブを試して、使用中のプローブが損傷していないことを確認してください。
<p>Error connecting to the target: (Error -2131 @ 0x0) Unable to access the device register. Reset the device, and retry the operation. If error persists, confirm configuration, pwer-cycle the board, and/or try more reliable JTAG settings (e.g. lower TCLK).</p>	<ol style="list-style-type: none"> 1.ターゲット構成ファイル (*.ccxml) で目的の JTAG/cJTAG モードが構成されていることを確認します。特定の TI 評価基板 (LaunchPad/controlCARD) がサポートできるのは、ボード設計に応じて 2 ピンまたは 4 ピン JTAG のみであることに注意してください。評価基板のユーザー ガイドを参照してご確認ください。 2.ターゲット構成の「テスト接続」に合格することを確認します。接続に失敗する場合は、そのエラーコードの手順に従います。 3.デバイスをブート待機モードに設定します。 4.手動起動の指示に従ってデバイスに接続します。 5.メモリ ブラウザで PARTID を読み取りできることを確認します。 6.デバイスのプログラムをもう一度実行します。 7.上記の手順を実行してもエラーがクリアされない場合は、次の点を確認してください: デバイス上にパスワードの場所はありますか? オンチップ フラッシュ ツールの設定? RAM のみをプログラムすることは可能ですか? XRSn は適切にアサートされていますか?
<p>This error is generated by TI's USCIF driver or utilities. The value is '-242' (0xfffff0e). The title is 'SC_ERR_ROUTER_ACCESS_SUBPATH'. The explanation is: A router subpath could not be accessed. The board configuration file is probably incorrect.</p>	<ol style="list-style-type: none"> 1.ターゲット構成ファイルをチェックして、正しいデバッグプローブが選択されていることを確認します。 2.PC のデバイス マネージャーにデバッグプローブが表示されていることを確認します。 3.USB ケーブルを交換、または別のデバッグプローブを試して、使用中のプローブが損傷していないことを確認してください。

表 4-1. 一般的なエラーコード表 (続き)

エラー メッセージ	デバッグ手順
Error connecting to the target: (Error -267 @ 0x0) The controller could not detect valid target supply. JTAG connection and/or connection setting specifying voltage level.	このエラー メッセージはデバッグ プローブの VTREF ピンが 3.3V に接続されていない場合にのみ表示されます。ターゲット ボードに電源が供給され、XRSn が適切にアサートされていることを確認します。
C28xx: Failed CPU Reset: (Error -1137 @ 0x0) Device is held in reset. Take the device out of reset, and retry the operation.	<ol style="list-style-type: none"> 1. ターゲット構成の「テスト接続」に合格することを確認します。接続に失敗する場合は、そのエラー コードの手順に従います。 2. デバイスをブート待機モードに設定します。 3. 手動起動の指示に従ってデバイスに接続します。 4. メモリ ブラウザで PARTID を読み取りできることを確認します。 5. デバイスのプログラムをもう一度実行します。 6. 上記の手順を実行してもエラーがクリアされない場合は、次の点を確認してください: デバイス上にパスワードの場所がありますか? オンチップ フラッシュ ツール の設定? RAM のみをプログラムすることは可能ですか? XRSn は適切にアサートされていますか?
The JTAG IR Integrity scan-test has failed. The JTAG DR Integrity scan-test has failed.	このエラーは、JTAG 信号の信号品質が低い場合にも通知されます。TCK 周波数を低くして、パターン長を確認してください。 ハードウェア設計ガイドの「 バッファされたケース 」セクションを参照してください。

5 JTAG チェーン内の複数のデバイス

JTAG 規格では単一の JTAG デバッグ プロブで複数のデバイスとデジタイズ チェーン接続を行うことができます。実際には、さまざまな制約によって、チェーン内で接続できるターゲット デバイスの数が制限されます。XDS クラスのデバッガはすべて、返されたビットの検出を予期する前にデバッグ プロブが循環できる、命令レジスタ (IR) ビットの数が制限されています。多くの古い C2000 デバイスではデバイスごとに 38 の IR ビットを循環する必要がありますが、ICEPick JTAG ルート コントローラのおかげで、すべての新しいデバイスでは 6 つの IR ビットの循環で十分です。このアプリケーション ノート執筆時点では、以下のデバイスに ICEPick が搭載されています: F2807x、F28M3x、F2837xD、F2837xS、F28004x、F2838x、F28002x、F28003x、F280013x、F280015x、F28P65x、F28p55x。デバイスに ICEPick が存在するかどうかを確認するには、[図 5-1](#) に示すように、Code Composer Studio を使用してターゲット構成の *Advanced* (詳細) タブを確認します。

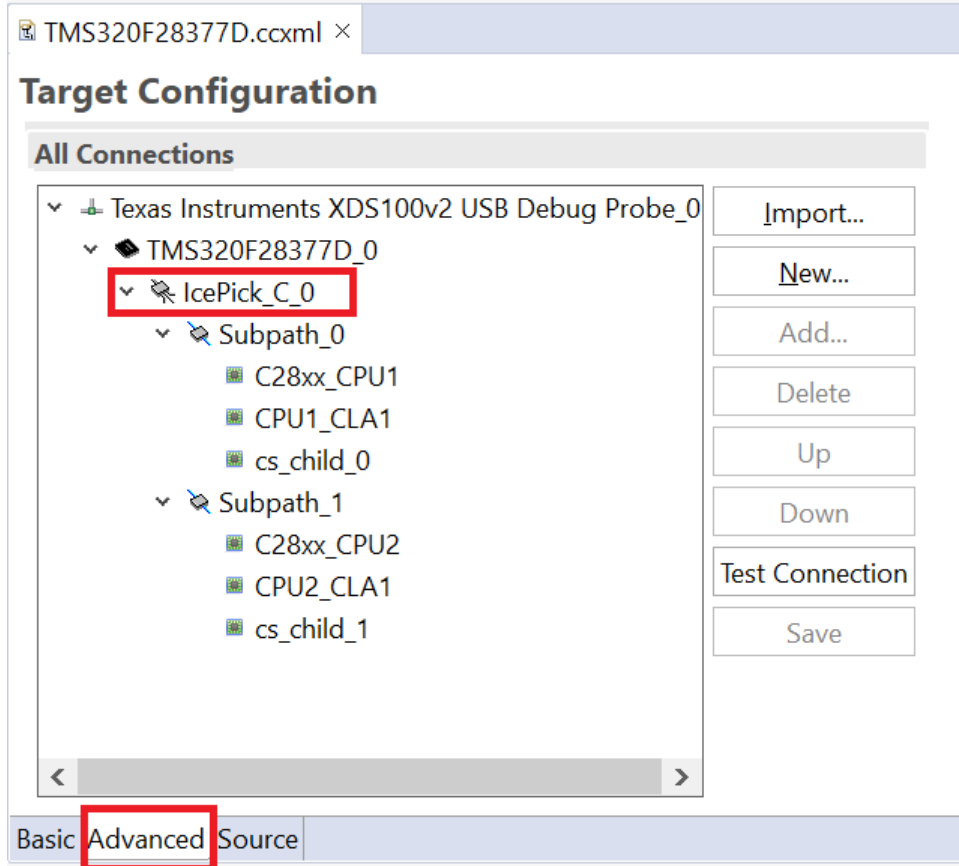


図 5-1. ターゲット構成の詳細ビュー

ICEPick を搭載していないデバイスでは、XDS100、XDS110、XDS200 のデバッグ プロブは、チェーン内で最大 2 つのデバイスのみを信頼性の高い方法で接続できます。ただし、JTAG チェーンの ICEPick では最大 12 個のデバイスを接続できます。XDS560 では、ICEPick なしでは 3 個のデバイス、ICEPick ありの場合は最大 18 個のデバイスに信頼性の高い方法で接続できます。『[TI ICEPick モジュールタイプ C リファレンス ガイド](#)』も参照してください。

6 非侵襲的デバッグ

ユーザーは非侵襲的に接続して、実行を中断することなくデバイスのリアルタイムの状態を確認できます。ターゲット接続時にリセットがトリガされるのを防ぐため、次の手順に従う必要があります：

注

次の手順は F28P55x デバイスに使用しましたが、どの F28x デバイスにも適用できます。

1. `ccs\ccs_base\emulation\gel` の CCS のインストールフォルダにある、デバイス固有の GEL ファイルを開きます。この例では、F28P55x デバイスは「f28p55x.gel」を使用します。
 - a. デバイスファミリによっては、GEL ファイル名も注文可能な型番に対応しています (例:「f2800157.gel」)。
2. `OnTargetConnect()` 関数で、`GEL_Reset()` と RAM の初期化行をコメントアウトします。ファイルを保存します。

```

OnTargetConnect()
{
//
// *(int *) (MEMCFG_BASE + MEMCFG_0_DXINIT) = 0x00FF; /* RAW INIT FOR M0/M1 Memory */
// while(!*(int *) (MEMCFG_BASE + MEMCFG_0_DXINITDONE) == 0xFF); /* Wait for InitDone Status */
//
// *(int *) (MEMCFG_BASE + MEMCFG_0_LSXINIT) = 0x00FF; /* RAW INIT FOR L51..L57 Memory */
// while(!*(int *) (MEMCFG_BASE + MEMCFG_0_LSXINITDONE) == 0xFF); /* Wait for InitDone Status */
//
// *(int *) (MEMCFG_BASE + MEMCFG_0_GSXINIT) = 0x00FF; /* RAW INIT FOR G50..G53 Memory */
// while(!*(int *) (MEMCFG_BASE + MEMCFG_0_GSXINITDONE) == 0xFF); /* Wait for InitDone Status */
//
// GEL_TextOut("\n\nRAM initialization done\n\n");

if (GEL_IsInRealtimeMode()) /* If in real-time-mode */
{
}
else /* Put device into C28x Mode */
{
C28x_Mode();
}

f28p55_Memory_Map(); /* Initialize the CCS memory map */

/* Check to see if CCS has been started-up with the DSP already */
/* running in real-time mode. The user can add whatever */
/* custom initialization stuff they want to each case. */

if (GEL_IsInRealtimeMode()) /* Do real-time mode target initialization */
{
}
else /* Do stop-mode target initialization */
{
// GEL_Reset(); /* Reset DSP */
}
}

```

図 6-1. コメントアウトする GEL `OnTargetConnect()` 行

3. CCS の、「Debug」(デバッグ) ウィンドウで *Target Configuration* (ターゲット構成) ドロップダウンに移動します。目的のターゲット構成ファイルを開きます。正しいデバッグ プローブ、デバイス、初期化スクリプト (GEL) が構成済みであることを確認します。

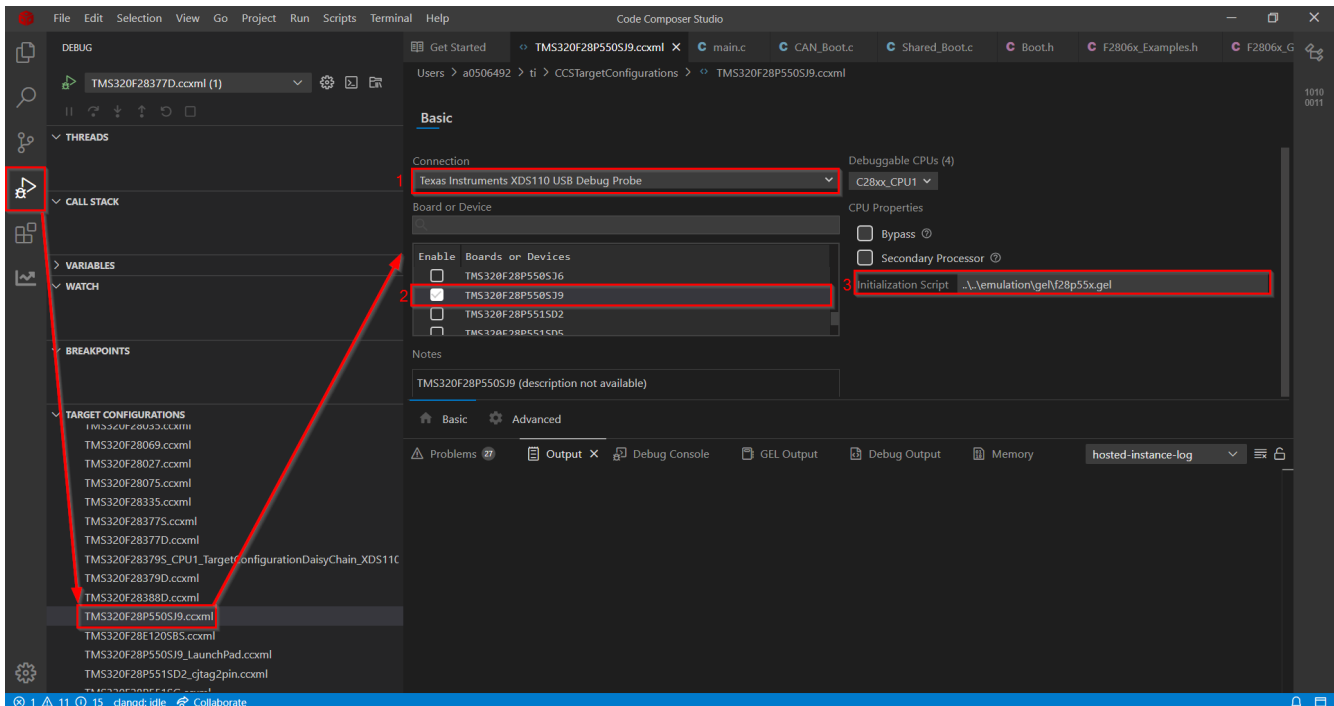


図 6-2. ターゲット構成ファイルの設定

4. **Target Configuration** (ターゲット構成) ドロップダウンで目的のターゲット構成 (ccxml) を右クリックして、**Start Project-less Debug** (プロジェクトレス デバッグを開始) を選択します。
5. **Threads** (スレッド) ドロップダウンで「C28xx_CPU1」(またはいずれかの CPU コア) を右クリックして、**Properties...** (プロパティ...) ウィンドウを開きます。
6. (目的のコアの) プログラム/メモリ ロード オプション メニューの **Connection Options** (接続オプション) の下で、「Halt the target on a connect」(接続中のターゲットを停止) と「Reset the target on a connect」(接続中のターゲットをリセット) の両方とも選択解除されていることを確認してください。構成が完了したら、プロパティを保存して閉じます。

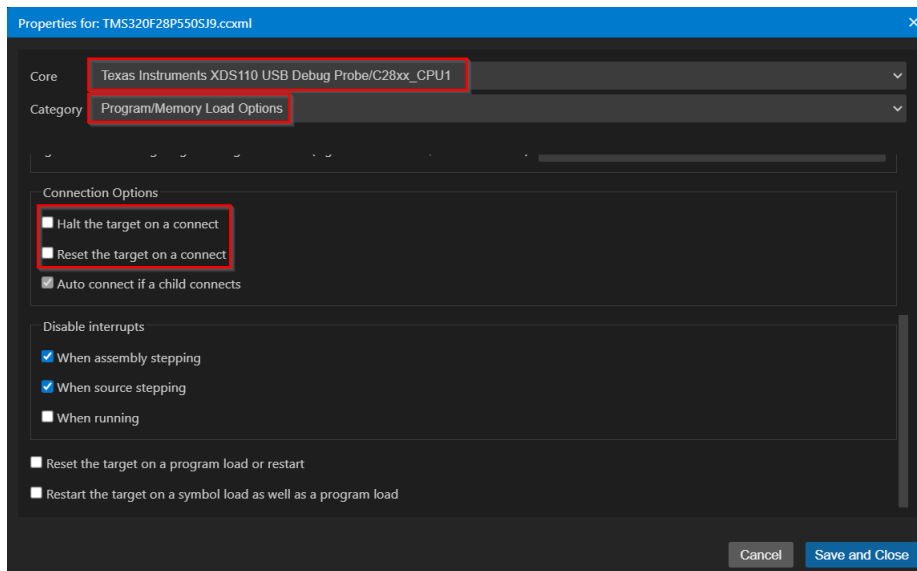


図 6-3. ターゲット構成のプロパティ

7. プロジェクトレス デバッグ セッションを再起動して、目的の CPU コアに接続します。これで接続を非侵襲的に確立できます。
 - a. プログラムの記号をロードするとデバイスの現在の状態を確認することもできます。

7 JTAG TAP の無効化とリセット

注

次のセクションは、以下のデバイス ファミリに関連しています：

- F28004x
- F28002x
- F28003x
- F280013x
- F280015x
- F28P55x
- F28P65x
- F28E12x

JTAG テスト用アクセス ポート (TAP) は、オンチップ ロジックのアクセス、テスト、デバッグに使用される標準化ハードウェア インターフェイス (IEEE 1149.1) です。TAP は FSM として機能し、TDI/TDO 経由でデータをシフト イン/シフト アウトして、バウンダリ スキャン レジスタを制御し、基板レベルのテスト、ハードウェア デバッグ、システム プログラミングを実行します。

TAP_STATUS レジスタは常に TAP のステータスを反映します。通常、デバイスに JTAG エミュレータが接続されていない場合、その状態は TLR (テスト ロジック リセット) 状態に保持されます。過剰な PCB ノイズが発生する場合は、望ましくない TMS および TCK トグルが存在して、JTAG が TLR 状態から外れる可能性があります。発生し続けると、最終的に JTAG バウンダリ スキャンやその他の JTAG モードが起動してしまい、目的のアプリケーションと直接干渉する可能性があります。バウンダリ スキャン (BSCAN) モードで通常のアプリケーション ロジックをバイパスすると、TAP は GPIO を制御できるようになり、最終的にシステムがハングアップする可能性があります。

図 7-1. TAP_STATUS レジスタ

31	30	29	28	27	26	25	24
DCON		予約済み					
R-0h		R-0-0h					
23	22	21	20	19	18	17	16
予約済み							
R-0-0h							
15	14	13	12	11	10	9	8
TAP_STATE							
R-0h							
7	6	5	4	3	2	1	0
TAP_STATE							
R-0h							

表 7-1. TAP_STATUS レジスタのフィールドの説明

ビット	フィールド	タイプ	リセット	説明
31	DCON	R	0h	IcePick からの DebugConnect 表示。 リセットタイプ:PORESETn
30-16	予約済み	R-0	0h	予約済み

表 7-1. TAP_STATUS レジスタのフィールドの説明 (続き)

ビット	フィールド	タイプ	リセット	説明
15-0	TAP_STATE	R	0h	TAP 状態ベクトル。ビット表現に基づき、対応する POTAP* 出力を以下に接続します。 0x0001:TLR、 0x0002:IDLE、 0x0004:SELECTDR、 0x0008:CAPDR、 0x0010:SHIFTDR、 0x0020:EXIT1DR、 0x0040:PAUSEDR 0x0080:EXIT2DR、 0x0100:UPDTR、 0x0200:SLECTIR、 0x0400:CAPIR、 0x0800:SHIFTIR、 0x1000:EXIT1IR、 0x2000:PAUSEIR、 0x4000:EXIT2IR、 0x8000:UPDIR リセットタイプ:PORESETn

表 7-2 は C2000 デバイスに実装された各種 BSCAN テストを示しています。これらのモードは、TI 製品ページで各デバイスファミリ向けに提供されている BSDL モデルで参照できます。すべての BSCAN モードが GPIO 機能をゲート制御するわけではないことに注意してください。スキャンシーケンス「xxxxxx」は TDI のバイナリ入力を表しており、TCK により最終ビットにクロックが供給されると、最終的にデバイスは対応する BSCAN モードになります。

表 7-2. バウンダリ スキャン モード

手順	TDI シリアル スキャン	GPIO の影響	システムへの影響
EXTEST	'b011000	GPIO をゲート制御 (構成可能)	対応可
SAMPLE	'b011011	なし	なし
バイパス	'b111111	なし	なし
HIGHZ	'b011110	GPIO をゲート制御	すべてのピンを接続解除
IDCODE	'b000100	なし	なし
PRELOAD	'b011011	なし	なし

意図せず BSCAN モードに移行することを避けるため、基板上には (特に TMS および TCK には) 十分強力な外付けブル抵抗を配置し、ノイズによって JTAG が有効になることを防止してください。最大限の信頼性を確保するため、システムで TDI を使用しない場合は、GPIO の方向を出力に変更して Low を駆動します。'b000000 は未使用の命令スキャンシーケンスであるため、スキャン時のデバイス動作は変更されません。

注

F28P55x および F28P65x デバイスでは、BSCAN_DIS ビットを設定して BSCAN モードを完全に無効にする、TAP_CONTROL レジスタが用意されています。BSCAN モードが無効化されていても、JTAG 経由のデバッグ アクセスは許可されたままです。

ソフトウェアの観点からは、アプリケーション コードで TAP_STATUS レジスタをポーリングしてデバイスの乱れを検出できます。SOFTPRES40[JTAG_nTRST] レジスタは、ソフトウェアを介して JTAG TAP をリセットするためにも使用できます。ユーザーは TLR から IDLE 状態に移移する (まだシステムには影響しない) TAP を検出して、デバイスが不明な命令を発行する前に、TAP を TLR 状態にリセットできます。レジスタが CCS やデバイス固有の技術リファレンス マニュアルに記載されていなくても、SOFTPRES40 を使用して JTAG TAP 状態をリセットする方法の詳細について、この E2E FAQ を参照してください。

注

SOFTPRES40 レジスタの使用にはご注意ください。コードが他の GPIO 状態でこのレジスタへの書き込みを承認されている、または他の手段によりノイズとデバッガ アクセスを区別する場合を除き、このレジスタによりデバッガに接続できなくなります。たとえば、TAP_STATUS レジスタの DCON ビットは、デバイスに接続されたデバッガが存在するかどうかを示すために使用できます。

8 JTAG コネクティビティのデバッグ フロー

以下のフローチャートは、JTAG 接続の問題を解決するための、一般的なトラブルシューティングの提案を特定して実行する手順を示しています。フローの最後にまだ問題がある場合は、「[TI Engineer to Engineer C2000 Support Forum](#)」に質問を投稿してサポートを受けてください。

8.1 全体的なデバッグ フロー

これらのフロー チャートの使用方法:

1. 図 8-1 の手順を見直し、まず高電圧絶縁フローについて調べます。絶縁が主な問題点でない場合でも、絶縁は PCB のエミュレーション機能に影響を及ぼす可能性があるため、これは重要です。
2. メイン JTAG デバッグ フロー チャートに従います。中間の手順が完了したら、メイン フローに戻り、問題が残っている場合は続行します。
3. すべてのフロー 要素を実行しても問題が解決されない場合は、TI E2E サポート フォーラムに質問を投稿してください。セクション 9 の末尾のリストには、TI から最も効率的に回答を得るために、質問に含めるべき情報について記載されています。

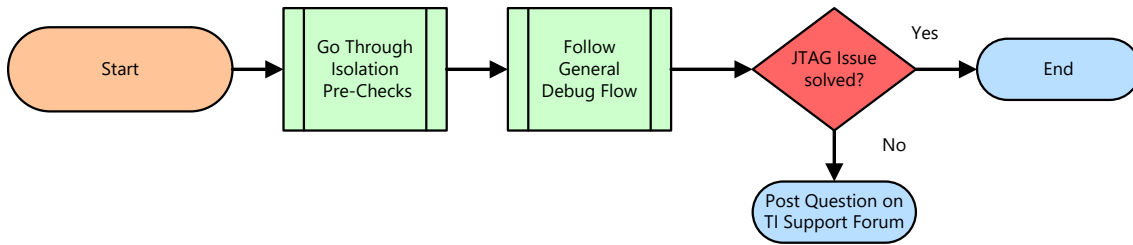


図 8-1. 全体的なデバッグ フロー

8.2 高電圧絶縁チェック フロー

実際、多くの C2000 アプリケーションは大電力です。このため、デバッグ中はターゲット ボードの電源プレーンをホスト コンピュータから分離します。TI 製の多くのボードは絶縁型エミュレーションを採用、またはオンボード オプションを使用して有効化できます。図 8-2 は、絶縁型 JTAG が存在するかどうかを識別し、存在する場合は、これらのシステムで一般的な問題のトラブルシューティングを実行するフローチャートを示しています。絶縁処理されたスタンドアロンのデバッグ プローブもあります。このフローは、これらのプローブ用ではありません。

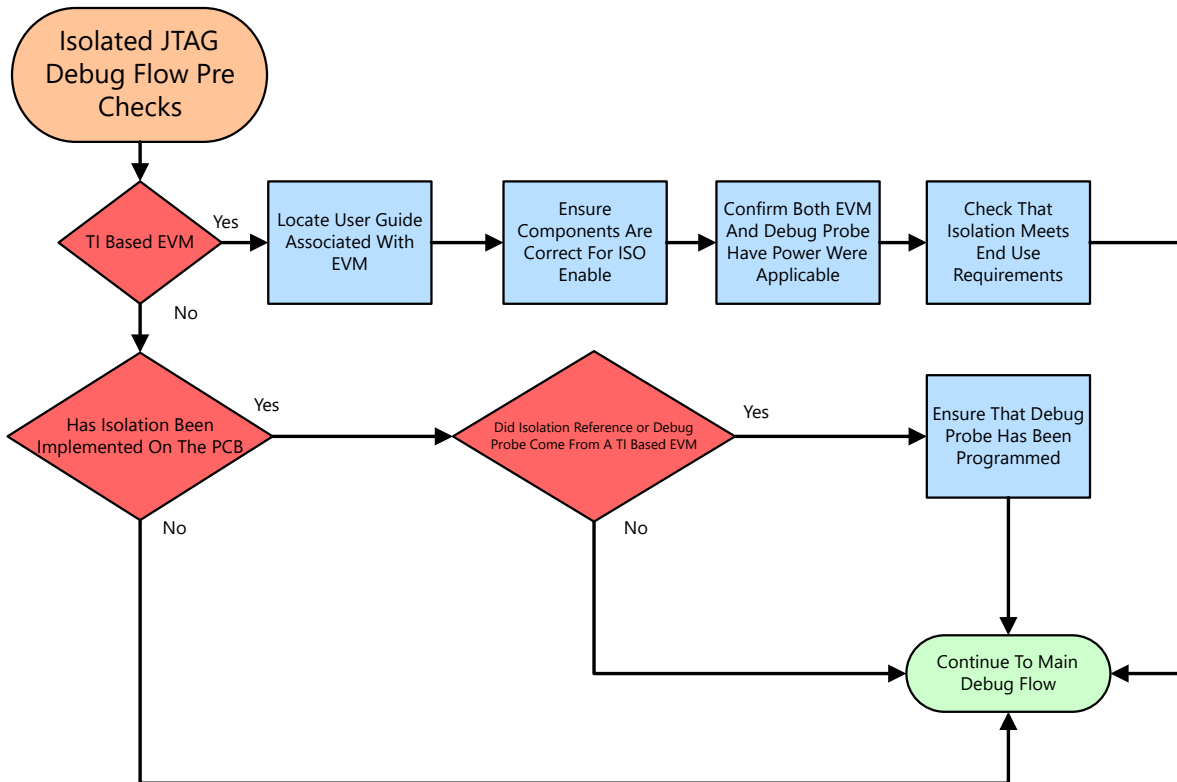


図 8-2. JTAG 絶縁の事前チェック

8.3 メイン JTAG デバッグ フロー

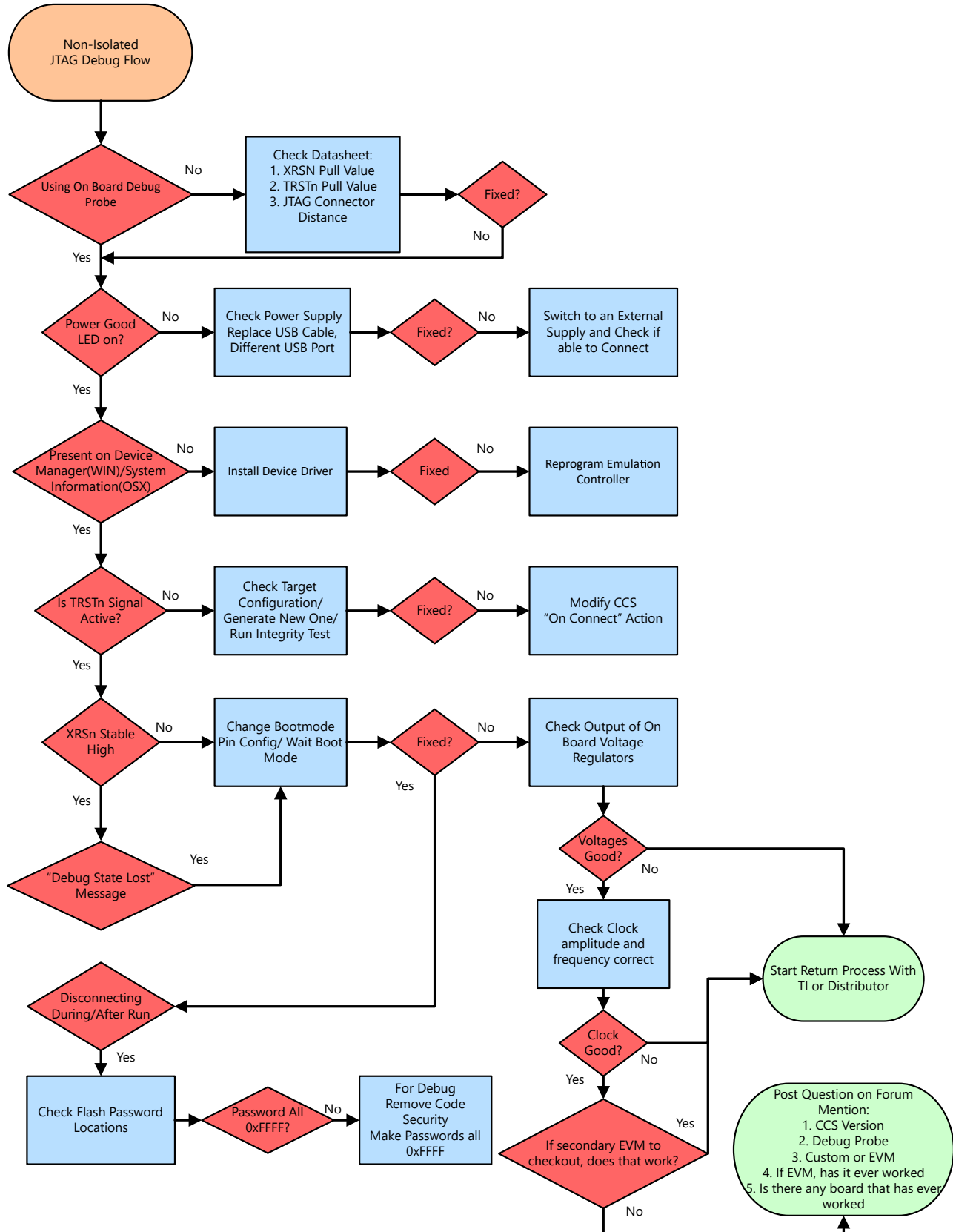


図 8-3. JTAG デバッグ フロー

9 フロー ステップの詳細情報

全体のデバッグ フロー (図 8-1) から始めて、高電圧絶縁チェック フロー (図 8-2) が必要かどうかを決定します。完了したら、メイン JTAG デバッグ フロー (図 8-3) を段階的に実行します。

これはフロー チャートの補足リストです。このリストは各手順のディレクティブの背景情報を説明しており、何を達成すべきかをよく理解するのに役立ちます。

9.1 絶縁の事前チェック フロー

1. **TI ベースの評価基板:** デバッグ中の評価基板が TI ベースである場合、フローはフロー チャートのこの分岐に従います。
2. **評価基板に関連するユーザー ガイドを探す:** すべての TI ベースの評価基板にユーザー ガイドまたはクイック スタート ガイドが用意されており、評価基板の機能と、評価基板を正しく動作させるために必須の部品に関する詳細が説明されています。これらを事前に確認するとデバッグ処理に役立ちます (このドキュメントで説明)。
3. **絶縁を実装した基板:** 前の手順のリファレンス ガイド、または他の資料を使用して、TI 製の評価基板でない場合に、デバッグ対象の基板に絶縁回路が実装されているかどうかを確認します。
4. **部品が ISO に適合していることを確認:** 評価基板のドキュメントを参照して、評価基板で目的の絶縁状態を実現するための、すべてのスイッチ、ジャンパ、またはシャントが正しく実装されていることを確認します。
5. **該当する場合は評価基板とエミュレーションの両方に電力が供給されていることを確認:** ローカルドメインと大電力ドメイン間の電源プレーンを適切に絶縁するため、アイソレータを使用して 2 つのプレーンを接続し、エミュレーション信号がマイコンに届くようにします。独立した電源プレーンがあるため、各プレーンに電力を供給する 2 つのパスが必要です。両方のプレーンに、エミュレーション経由でデバイスに接続する電源があることを確認します。
6. **絶縁が最終用途の要件を満たしていることを確認:** 最初のシステム デバッグに不可欠なチェックではありませんが、使用されるアイソレーション デバイスについて学んで、デバイスが最終システムの要件を満たしているかどうかを理解するために重要です。TI の評価基板は最終アプリケーションまでこの点を包含していますが、カスタム評価基板と TI の評価基板が混在している場合は確認する必要があります。
7. **TI ベースの評価基板が絶縁/エミュレーションのリファレンスを供給:** 多くの場合、TI の評価基板の絶縁回路とエミュレーション回路はカスタム設計に再使用されます。電気的には問題なくても、エミュレーション チップセットは依然としてプログラムが必要があることは見落とされがちです。TI ベースの評価基板の場合、評価基板の販売前にこれが考慮されます。ただし、カスタム基板では、製造フローにこのプロセスを含める必要があります。

9.2 JTAG デバッグ フロー

1. **オンボード デバッグ プロープの使用:**
 - a. **あり:** 多くの C2000 マイコン ボードでは、PCB 上に JTAG デバッグ プロープが実装されています。アプリケーション要件がある場合を除き、オンボード デバッグ プロープは開発用に使用することを TI では推奨しています。XDS100 と XDS110 は、TI の C2000 評価基板 (EVM) に搭載される 2 つのターゲット デバッグ プロープです。
 - b. **なし:** ボード設計がカスタムであるスタンドアロン デバッグ プロープを使用する場合は、デバッグ プロセスを続行する前に、JTAG ヘッドと受動素子の実装を検証する必要があります。デバイス固有のデータシートには、適切な動作を行うための、適切なプルアップ値またはプルダウン値のリファレンス回路図が含まれています。PCB が TI 製の場合、この手順はスキップできます。
2. **パワー グッド LED オン:** この手順は、電圧計などの外部機器を使用せずに、ターゲットに電源が正しく供給されていることを確認するためのものです。すべての TI C2000 開発ボードには、マイコンに電力が供給されていることを示す LED が搭載されています。他の LED は、標準搭載のコードが正常に動作していることを示すために使用できます。これらの LED の位置と機能については、デバッグ対象の評価基板のデバイス固有のユーザー ガイドを参照してください。

3. **ケーブルを交換:** パワー グッド LED が点灯しない場合、評価基板に供給される電源に問題がある可能性があります。多くの TI 評価基板では、ホストからターゲットへのデバッグ パスのためだけでなく、評価基板に電力供給する USB からの 5V を供給するためにも USB 接続を使用します。簡単なチェックとして、USB ケーブルを変更して問題がないことを確認します。ホストからの電力が不足している場合は、電力供給 USB ハブも役立ちます。
4. **外部電源に切り替え:** TI 製基板のオンボード電源では適切なレベルの電力を供給できず、USB ケーブルが良好であることがわかっている場合は、評価基板での外部電源の使用を試してください。これがサポートされているかどうかを知るためには、ご使用の評価基板のデバイス固有のユーザー ガイドを参照してください。この場合、電源に問題があるのか、または PCB 上の何かがマイコンへの電圧を抑制しているのかを判断するため、基板上の電圧を計測する必要があります。
5. **デバイス マネージャに存在:** JTAG デバッグ プローブと PC 間の通信には、ドライバ ファイルをインストールする必要があります。通常、Code Composer Studio (CCS) をインストールすると、このファイルもインストールされます。ドライバが正常にインストールされていることを確認するには、PC を JTAG デバッグ プローブに接続して電源を投入します。次に、コントロールパネル → デバイス マネージャ (図 9-1) に移動し、関連するデバッグ プローブを見つけます。

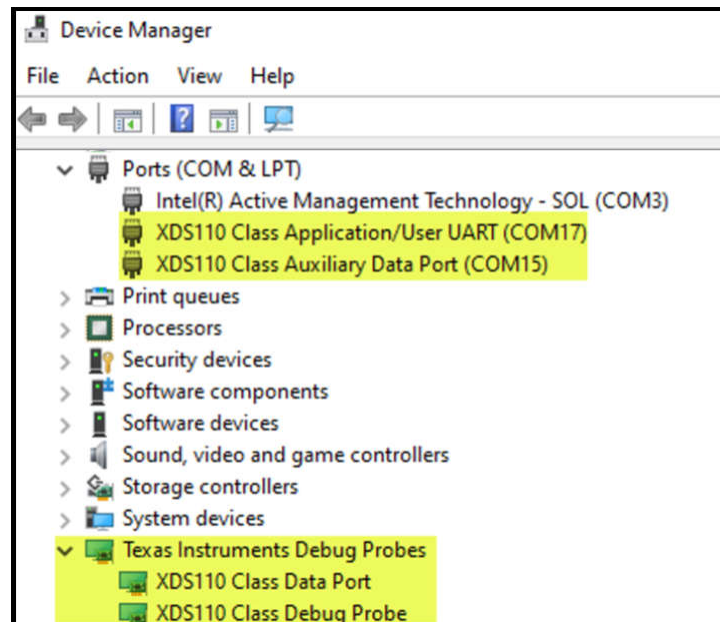


図 9-1. XDS110 デバッグ プローブが正常に検出されたことを示す Microsoft® Windows® 10 のデバイス マネージャ

6. **リプログラム エミュレーション コントローラ:** この手順では、エミュレーション コントローラとして機能するデバイスのファームウェアが正しいことを確認します。
 - a. XDS100v1: ホスト デバイスは、FTDI FT2232 です (再プログラミングの次のガイド)
 - b. XDS100v2: ホスト デバイスは FTDI FT2232 です (再プログラミングの次のガイド)
 - c. XDS110: ホスト デバイスは、TI マイコン TM4C1294NCPDRI3R です (再プログラミングの次のガイド)。
7. **デバイス ドライバのインストール:** ホスト PC/MAC システムにデバッグ プローブが表示されない別の可能性は、ドライバがインストールされていないことです。通常、このファイル CCS のインストール時にインストールされます。利用可能なドライバについては、デバッグ プローブの製品ページを参照してください。
8. **マイコンでの Is TRSTn 信号 High:** この手順では、CCS がターゲットに接続を試みる際の際の特定の動作を確認します。最初の動作の 1 つは、テストリセット (TRSTn) がインアクティブ High になり、外部デバッグ プローブへのコア デバッグ接続が有効になります。CCS 接続ターゲット動作中に TRSTn の状態が変化しない場合は、デバイス レベル、およびホストのオペレーティング システム内の両方で適切な構成があるかどうか、デバッグ プローブを確認する必要があります。

9. **ターゲット構成の確認:** ターゲット構成ファイル (.ccxml) には、ターゲット デバイスおよび使用する JTAG デバッグ プローブへの接続に必要な情報が含まれています。現在のターゲット構成を表示するには、CCS の「View」(表示) タブの下にある「Target Configurations」(ターゲット構成) (図 9-2) を選択します。デバッグ対象のターゲットに対応する .ccxml をダブル クリックします。デバッグ プローブ用のドライバが正しくインストールされ、正しいオプションが選択されている場合は、**Test Connections** (テスト接続) ボタン (図 9-3) を使用して実行します。このテストのデータログは接続の問題の原因を特定するのに役立つため、この手順は省略しないでください。多くのサンプル プロジェクトが C2000Ware の一部としてインストールされるか、controlSuite に *target configs* フォルダが作成されます。ここには、デフォルトの評価基板とデバッグを想定して、事前に ccxml ファイルが作成されます。デバッグ セッションの起動時に「デバッグ」アイコンを使用する際、このファイルが使用されます。「デバッグ」ボタンを使用してデバッグ セッションを起動する場合は、「ターゲット構成」の .ccxml を変更する必要があります。

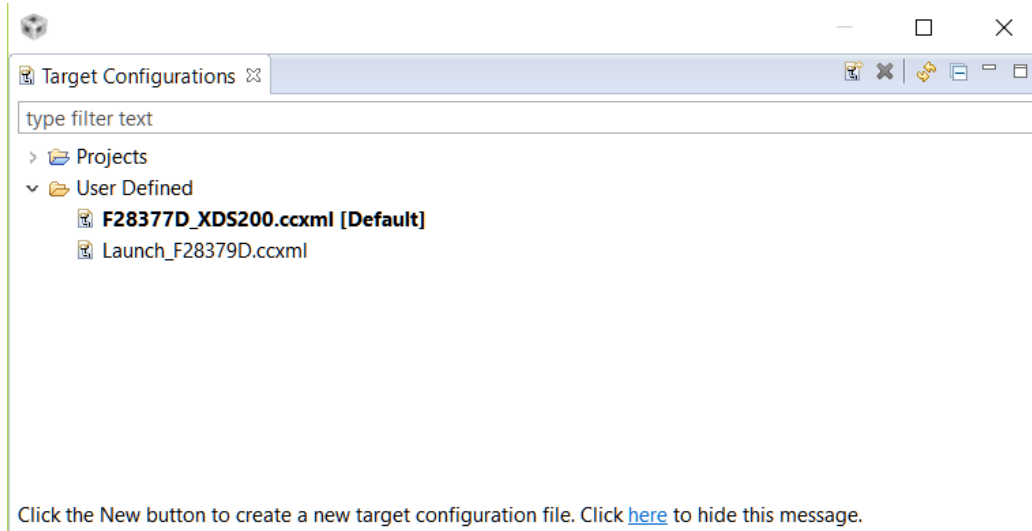


図 9-2. ターゲット構成ビュー

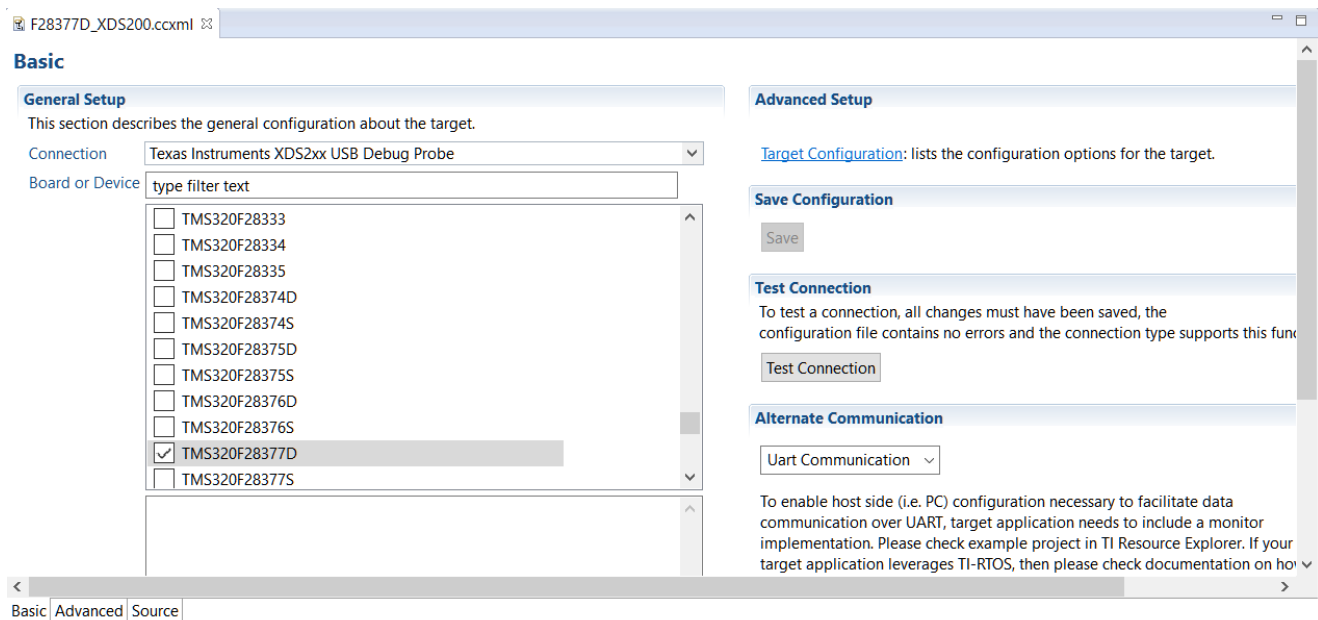



図 9-3. テスト接続

10. **CCS の On Connect アクションを変更:** CCS からデバッグ セッションを起動する方法は 2 つあります。1 つの方法は、前の手順から目的のターゲット構成を右クリックして、**Launch Selected Configuration** (選択したコンフィギュレーションを起動) を選択することです。この作業が完了すると、CPU コアを右クリックして「Connect Target」(ターゲットを接続) を選択してターゲット CPU を接続できます。もう 1 つの方法は、デバッグ ボタン (図 9-4) を使用することです。これを使用すると、構成の起動に加え、接続してターゲット プログラムファイルをメモリにロードし、「メイン」まで実行します。これらの設定は変更できますが、これがデフォルトの操作です。デフォルトのアクションは、使用中の .ccxml ファイルを右クリックするか、「デバッグ」ボタンの隣の矢印ドロップ ダウン メニューから「デバッグ オプション」を選択してターゲット サブメニューの自動実行オプションと起動オプションを変更することで修正できます。TI では、本書のトラブルシューティング フェーズにおいて従来の「ターゲット接続」方式を使用することを推奨しています。これにより、コードの実行またはその他のシステムのやり取りに起因する、JTAG には全く関連しない問題を特定できます。システムが安定してターゲットを起動して接続できることが確認できたら、デバッグ ボタンを使用してこれらの手順を処理します。

 workspace_v7 - CCS Debug - Code Composer Studio

File Edit View Project Run Tools Scripts Window Help



図 9-4. Code Composer のデバッグ ボタン

11. **XRSn の状態:** オシロスコープの XRSn を見ると、デバイス動作時には XRSn がインアクティブ High である必要があることがわかります。XRSn Low、または Low から High のパルス変化がある場合は、いずれかの問題が生じている可能性があります。パルスが周期的である場合は、マイコンのウォッチドッグ (WD) が動作していないか無効であるため、リセットされた可能性があります。この切り替え動作は、マイコンに電源が投入されてコードが実行されていることを示すため悪いことではありませんが、その動作によってデバッグ フローが不安定になる可能性があります。不明確なパルスが発生している、または XRSn が常に Low の場合は、電源電圧の問題または PCB の何らかの問題が原因で、内部ブラウン アウトリセット (BOR) がトリガされていることを示しています。これは前述のスタティック電源チェックとは異なります。コードの実行中にも、これら両方の潜在的な問題が発生することがあります。この問題により、デバッグ セッションが接続解除されるか、接続の信頼性が損なわれます。
12. **BootMode を変更:** ハードウェア ファイルをチェックして、BootMode ピンが予測されるモードに対応した正しい状態にあることを確認します。XRSn ピンが上述の動作を示している場合、またはフラッシュ メモリの状態が不明な場合、ブート待機モードに移行するとデバイスは安全な状態に移行し、メモリおよびレジスタの読み取り可能になります。ブートピンおよびブート待機モードに必要な選択の詳細については、デバイス固有のデータシートの「ブート」セクションを参照してください。
13. 「**デバッグ状態喪失**」 **CCS メッセージ:** XRSn が目的のインアクティブ High 状態であっても、デバッグ接続を妨げる、または終了する問題が存在する可能性があります。多くの場合、この動作はデバイスで実行中のコードに関連しています。このため、デバイスはブート待機モードに移行します。
14. **VREG 設定を確認:** 推奨動作条件の範囲外の電圧がデバイスに供給されると、ブラウン アウトリセット (BOR) イベントが発生する可能性があります。このような状況の場合は、デバイスの電圧レールを測定してください。C2000Ware または controlSuite にある回路図ファイルを使用すると、デバイスのレールのプローブ ポイントを検証できます。コード実行中にこの問題が発生する場合は、ソースがデバイスに供給できる電流量に問題がある可能性があります。デバッグ中の評価基板が TI 製デバイスである場合は、外部電源から生成されたレールはすべて問題がありませんが、設計上、この時点でボードの整合性が良好であることを検証するチェックが実行されます。
15. **クロック チェック (JTAG クロック/システム クロック):** データシートで定義されたレベルに従って、JTAG クロックと水晶または外部クロック ソースを測定して確認します。デバッグ プローブについては、メーカーのデータシートを参照してください。これは、デバイスが正常に機能するために必要な入力にデバイスに供給されていることを確認するための最後の手順です。多くの Piccolo™ クラスのデバイスには、ゼロピン発振器が内蔵されています。外部クロックが不確実な場合、これを機能クロックとして使用すると便利です。使用可能なクロック ソースと許容誤差については、デバイス固有のデータシートを参照してください。JTAG クロックは通常、初期設定ファイルのデフォルト速度を維持しますが、クロック レートを遅くすると、これにより初期接続または接続安定性が向上するかどうかを確認するのに役立ちます。これはカスタム設計の PCB で特に便利です。
16. **2 番目のデバイス チェック:** これまでの手順で問題が解決しない場合は、2 番目の PCB または評価基板を使用して、問題が 1 つの評価基板にあるかどうかを判断できます。2 番目のデバイスが同じ方法で機能しない場合は、設定の問題、または使用中の評価基板の外部の問題である可能性があります。

17. **実行中または実行後の接続解除:** デバイスがパスワードでロックされている場合、コードセキュリティモジュール (CSM) のエミュレーションコードセキュリティロジック (ECSL) によりデバイスへの JTAG エミュレーションが無効化され、JTAG 接続の問題が発生します。これは、上記に従って接続する前に発生する可能性があります。デバッガが接続されている間にメモリのセキュア領域にアクセスする場合は、デバッグ中にも発生することがあります。ブート待機モードにより接続が可能な間、このモードではデバッグ中のセキュアメモリへのアクセスの問題は訂正されません。これを修正するには、既知のパスワードを使用して CSM のロックを解除する必要があります。デバイスのロックおよびロック解除の詳細については、CSM モジュールのデバイス固有のデータシートおよび関連手順を参照してください。パスワードが不明な場合は、デバイスのロックは解除できません。デバッグは保護されていない領域に制限されます。
18. **E2E.ti.com に質問を投稿:** このフローの最後に JTAG を介したデバイスとの接続または接続維持に関する問題が依然としてある場合は、[TI C2000 Engineer to Engineer Forum](#) に質問または問題を投稿できます。投稿する際には、問題を文書化して、以下の情報を提供してください:
- 投稿の件名またはタイトル: JTAG 接続の問題 - (ここに型番を挿入)
 - CCS バージョン
 - 使用したデバッグプローブ
 - ターゲットのタイプ: TI 製の評価基板またはカスタム
 - このガイドのどの手順を実行しかの確認
 - TI の評価基板以外の場合、JTAG 接続のカスタムボードの回路図 (可能な場合)

10 参考資料

これらは、JTAG の問題のデバッグ、またはエンドシステムに適した JTAG エミュレーション デバイスの選択に役立つ、詳細なサポート ページです

- 『一般的な JTAG の問題をデバッグするための TI ガイド』
- テキサス インストルメンツ、『[C2000™ リアルタイム制御マイコン \(MCU\) 活用の基礎](#)』
- [C2000 リアルタイム マイコン](#)

11 改訂履歴

Changes from Revision C (August 2024) to Revision D (April 2026) Page

- XDS110 に絶縁デバッグ機能を追加..... 2
- 「一般的なエラー コード」セクションの形式を若干修正..... 5
- セクション 6 セクションを追加..... 10
- セクション 7 セクションを追加..... 12

Changes from Revision B (January 2023) to Revision C (August 2024) Page

- 「LaunchPad™ 開発キットと controlCARDs のデバッグ手順」セクションを追加..... 3
- 「一般的なエラー コード」セクションを追加..... 5

Changes from Revision A (January 2022) to Revision B (January 2023) Page

- ドキュメント全体にわたって表、図、相互参照の採番方法を更新。..... 2
- JTAG デイジー チェーンをサポートする特定の JTAG デバッグ プローブの機能に関する文書化..... 9

重要なお知らせと免責事項

TI は、技術データと信頼性データ (データシートを含みます)、設計リソース (リファレンス デザインを含みます)、アプリケーションや設計に関する各種アドバイス、Web ツール、安全性情報、その他のリソースを、欠陥が存在する可能性のある「現状のまま」提供しており、商品性および特定目的に対する適合性の黙示保証、第三者の知的財産権の非侵害保証を含むいかなる保証も、明示的または黙示的にかかわらず拒否します。

これらのリソースは、TI 製品を使用する設計の経験を積んだ開発者への提供を意図したものです。(1) お客様のアプリケーションに適した TI 製品の選定、(2) お客様のアプリケーションの設計、検証、試験、(3) お客様のアプリケーションに該当する各種規格や、その他のあらゆる安全性、セキュリティ、規制、または他の要件への確実な適合に関する責任を、お客様のみが単独で負うものとし、

上記の各種リソースは、予告なく変更される可能性があります。これらのリソースは、リソースで説明されている TI 製品を使用するアプリケーションの開発の目的でのみ、TI はその使用をお客様に許諾します。これらのリソースに関して、他の目的で複製することや掲載することは禁止されています。TI や第三者の知的財産権のライセンスが付与されている訳ではありません。お客様は、これらのリソースを自身で使用した結果発生するあらゆる申し立て、損害、費用、損失、責任について、TI およびその代理人を完全に補償するものとし、TI は一切の責任を拒否します。

TI の製品は、[TI の販売条件](#)、[TI の総合的な品質ガイドライン](#)、[ti.com](#) または TI 製品などに関連して提供される他の適用条件に従い提供されます。TI がこれらのリソースを提供することは、適用される TI の保証または他の保証の放棄の拡大や変更を意味するものではありません。TI がカスタム、またはカスタマー仕様として明示的に指定していない限り、TI の製品は標準的なカタログに掲載される汎用機器です。

お客様がいかなる追加条項または代替条項を提案する場合も、TI はそれらに異議を唱え、拒否します。

Copyright © 2026, Texas Instruments Incorporated

最終更新日 : 2025 年 10 月