

このチュートリアルでは XJTAG に標準の XJDemo ボード (Ver3.x) を使って、JTAG バウンダリスキャンテストツールによるリアルタイムデバッグ XJAnalyser の使用法を紹介します。チュートリアルは以下の3つのセクションから構成されます。

構成

XJAnalyser の構成はウィザードベースです。このセクションでは、XJDemo ボードを動作させるための XJAnalyser のウィザードを使った構成方法を説明します。

ピンのセットとモニタ

XJAnalyser はピンレベルのデバッグツールです。このセクションでは、XJAnalyser を使ってピンの値をセット・モニタするための、様々な方法を紹介합니다。

デバイスのプログラミング

XJAnalyser には、デバイスをプログラムするためのインターフェースを持っています。このセクションでは、プリコンパイルされた SVF プログラムをロードし、実行する方法を確認します。

チュートリアルには、XJAnalyser の動作とその操作方法についての手順が記載されています。

チュートリアルの全ての操作は、このようにハイライト表示されます。

XJAnalyser プロジェクトウィザード

XJAnalyser を使って回路のデバッグを行うには、幾つかの情報が必要です。このセクションでは、XJDemo ボードを使えるように XJAnalyser を構成する手順を説明します。

ステップ 1 - 新規或いは既存のプロジェクト

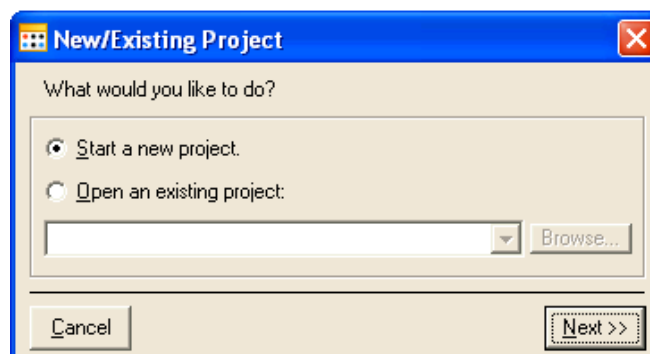
ステップ 2 - 電源、ピンマッピングとテストリセットシーケンスのオプション

ステップ 3 - BSDL ファイルの関連付け

ステップ 1 - 新規或いは既存のプロジェクト

XJAnalyser を実行すると、プロジェクトウィザードが自動的に開きます。

ウィザードの最初のステップは、“既存のプロジェクトを開く” か、“新規に作成する” かの選択です。



XJAnalyser プロジェクトには、個々の回路で XJAnalyser を動作させる為に必要な全ての構成情報が含まれます。新しい回路で XJAnalyser を使用する場合は、ウィザードで素早く簡単にプロジェクトを作成できます。その後プロジェクトは保存できるため、一度構成を行うだけで済みます。

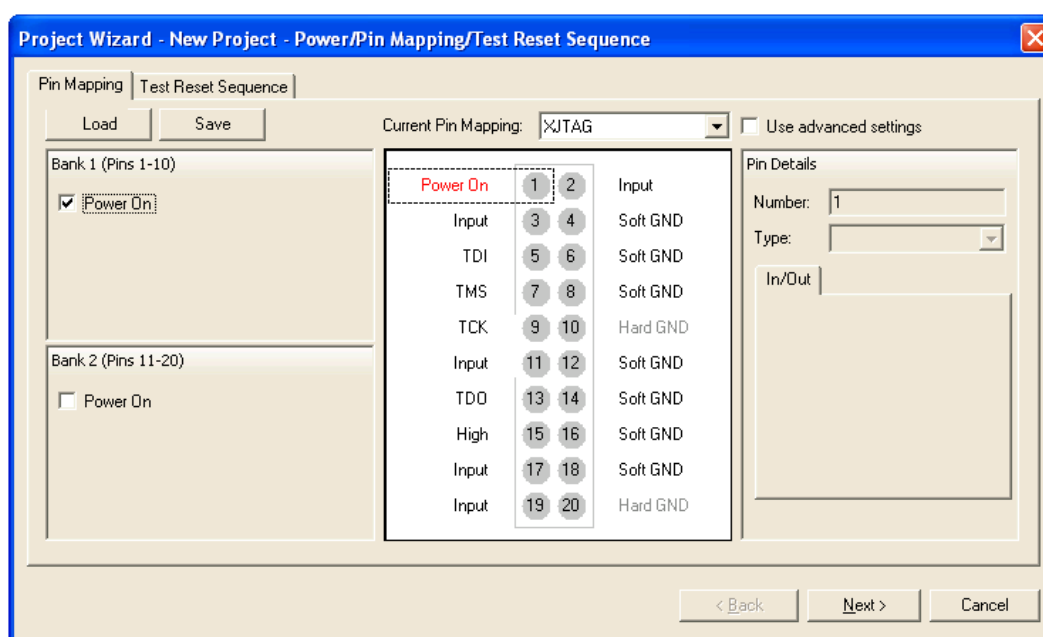
- Start a new project オプションを選択して Next をクリックします。

ステップ 2 - 電源、ピンマッピングとテストリセットシーケンスのオプション

XJAnalyser のウィザードでの 2 番目のステップは、XJLink 又は XJLink2 の JTAG インターフェースの構成を行います。

Pin mapping タブを使って、XJLink の使用するピンの指定と USB 経由での回路への電源供給の可否を指定できます。XJDemo ボードでは、標準の XJTAG ピンマッピングを使用し、USB からの電源供給が必要です。以下の手順で設定します。

- Current Pin Mapping ドロップダウンリストから XJTAG を選択します。
- ピンマッピングの表示上で、ピン 1 (Power と表示) をダブルクリックします。



- Next をクリックし、構成の最後のページに移動します。

回路への電源が USB に損傷を与える可能性がある旨のワーニングが表示されます。XJDemo ボードは USB の電源で動作するように設計されているので、Yes をクリックします。

ステップ 3 – BSDL ファイルの関連付け

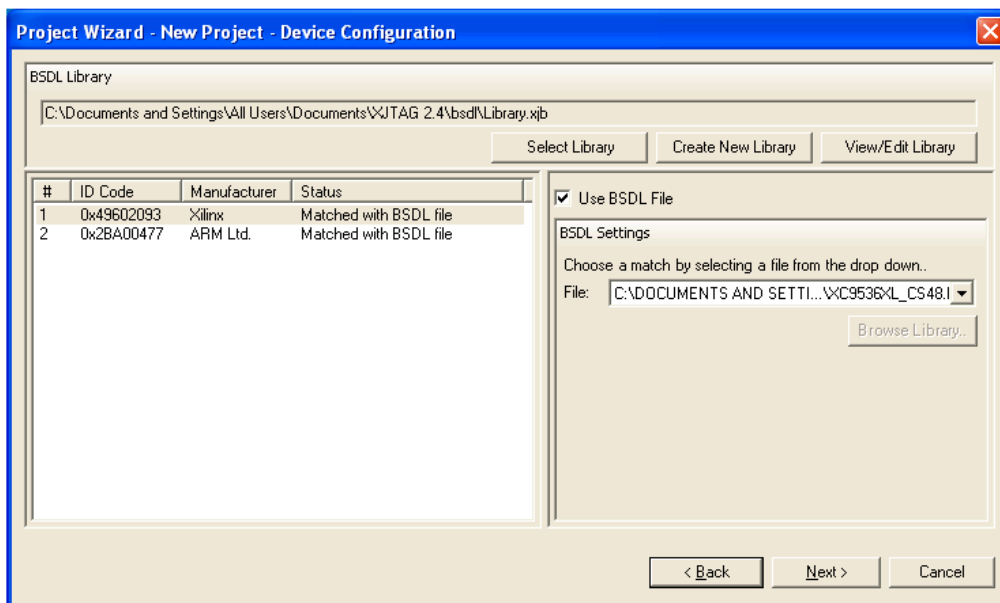
XJAnalyser で XJDemo ボードを動作させる構成の最後のステップは、JTAG チェイン内のデバイスへの BSDL ファイルの関連付けです。

電源とピンマッピングが設定できれば、XJAnalyser は即座に XJDemo ボードのスキャンを行い、JTAG チェイン内のデバイスの ID コードを特定します。

これら ID コードと BSDL ファイルライブラリを使って、そのデバイスの JTAG 実装方法を記述した BSDL ファイルが自動的に規定されます。デフォルトのライブラリは Library.xjb の名前で提供されています。

XJAnalyser は、見つかった各デバイスを（ライブラリとの）一致状況と共に表示します。ライブラリ内に一致するものが見つからない場合は、必要なファイルを追加する必要があります。View/Edit Library ボタンをクリックしてライブラリを編集できます。XJDemo ボード用の BSDL ファイルは、XJTAG\BSDL\Demo ディレクトリにインストールされています。

デバイスと BSDL ファイルの一致やバイパス設定に関するより詳しい情報は、XJAnalyser Help→Reference→The Dialogs→The Project Wizard→Device Configuration を参照してください。



- Next をクリックしてウィザードを終了し、XJAnalyser の使用を開始します。

XJDemo ボード上の JTAG デバイスが表示されます。

XJAnalyser – ピン操作

XJDemo ボードが動作するように XJAnalyser を構成するか、このボードを基にした既存のプロジェクトを開くと、JTAG チェイン内のデバイスに対して、ピンの値をモニタ及び制御できます。このセクションでは、ピンの値をモニタ・制御する 3 つの方法とピンの衝突を処理する仕組みを紹介します。

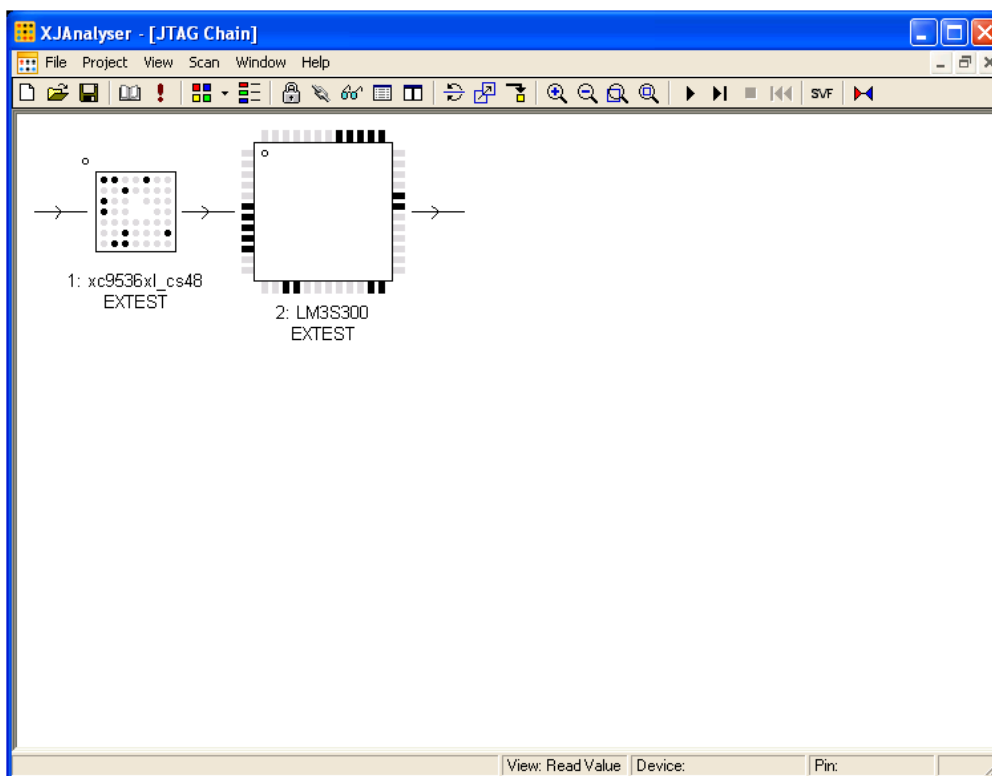
- Chain Window – JTAG チェインのグラフィカル表示
- Pin List – デバイス上の全てのピンに対して属性を表示
- Pin Watch – 選択したピンのモニタと制御
- Conflicting pins – XJAnalyser がピンの衝突を処理する方法

Chain Window – JTAG チェインのグラフィカル表示


XJAnalyser の Chain Window には、JTAG チェイン内のデバイスがグラフィカルに表示されます。XJDemo ボードのチェインは2つのデバイス（BGA パッケージの Xilinx CPLD と標準パッケージの Luminary Micro LM3S300 マイクロコントローラ）で構成されています。

インターフェース

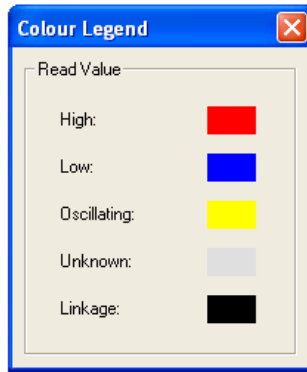
JTAG チェインの初期状態は、**unstarted**（スキャン未実行）です。この状態では、全てのピンは、Linkage ピン、JTAG 制御ピン、未読み取りピンの何れかになっています。チェインのスキャンを実行することで、これら未読み取りピンの値と色が変わります。



ピンの色

ピンの色の設定状況を確認するには、**pin colour legend** を表示します。表示には “Project メニューから **colour Legend...** を選択する” 又は “ツールバーの **colour legend** ボタン  をクリックする” の2つの方法があります。Project メニューから **pin colours** を選択することで、各設定色を変更できる **Pin Colours** ダイアログを表示することもできます。





- ツールバーの  ボタンをクリックし、**Colour Legend** を表示します。




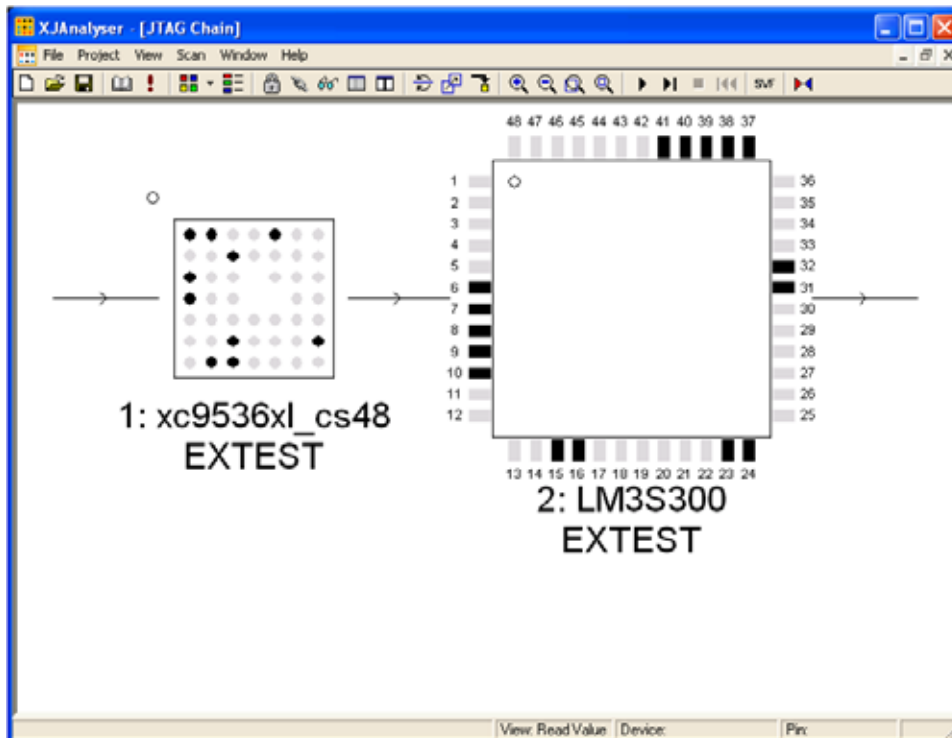
デバイス

XJAnalyser は、通常、実際の物理的外観とレイアウトを元にチェーン内のデバイスを表示します。しかしながら、左右反転や回転などの方法で、デバイスの表示方法を変更できます。また、デバイスを右クリックし、ポップアップメニューから適切な項目を選択することで、ジオメトリを変更することも可能です。


ズーム表示

XJAnalyser のズーム機能を使って画面上のデバイスの大きさを変更できます。View メニューの **Zoom In** 又は **Zoom Out**、若しくはツールバーの  や  ボタンを選択することで、拡大や縮小が可能です。View メニューの **Zoom Defined** 若しくはツールバーの  ボタンで、チェーンの特定の場所をフォーカスすることも可能です。View メニューの **Zoom Full** 若しくはツールバーの  ボタンで、チェーン全体を表示可能です。

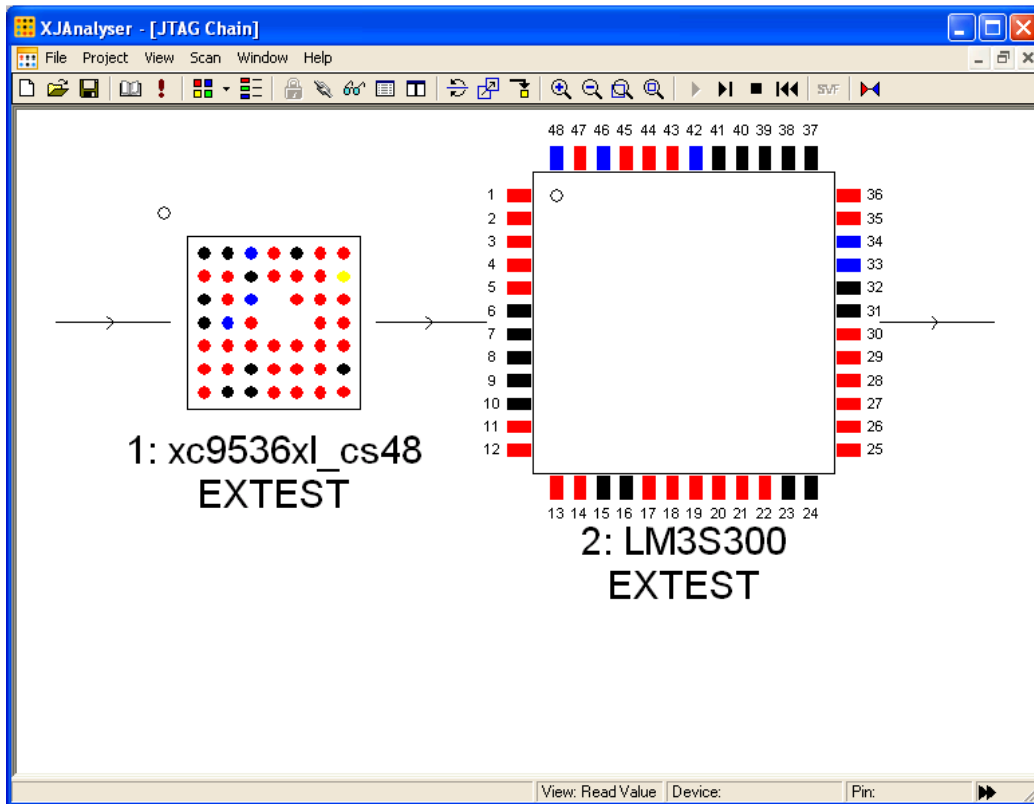
- ツールバーの  ボタンをクリックし、XJDemo ボードの両方のデバイスを含むように四角をドラッグします。
- ESC キーでズームモードから抜け出します。



チェーンの制御

JTAG チェインのスキャンを開始するには、Scan メニューの **Start JTAG Scan** を選択するか、ツールバーの Run ボタン  をクリック、若しくは、**F5** キーを押します。

- ツールバーの  をクリックし、チェーンのスキャンを開始します。



ピンのモニタ

Chain Window のピンの色は、その値をモニタする最も簡単な手段です。ピンの値が変われば、その色も新しい状態に反応して変わります。ピン B7 の色は、その値が切り替わる形で変化します。

ピンの制御

チェーンのスキャンを開始すると、全てのピンがデフォルトの状態になり、可能な限り入力として設定されます。しかしながら、XJAnalysers を使って、ピンの値を駆動して、テストを実行することができます。以下の 2 種類の手順でピンの値を設定できます。

- ピンを右クリックしピンオプションメニューを表示。メニューの中から、設定したい値を選択。
- ピンをダブルクリックすることで値が切り替わる。ピンが 1 ならダブルクリックで 0 に、逆も同様。

- View メニューから **Goto** を選択します。
- デバイスのドロップダウンリストから **Device 1** を選択し、ピンのリストからピン **D2** を選択します。ピン **D2** の場所が表示されます。
- ピン **D2** を右クリックし、オプションメニューを表示させます。
- メニューから **Set 1** を選択します。

ピンの色が青から赤に変わること、値が変わったことが確認できます。ピンに×マークが入ること、ピンが駆動されていることも分かります。ピンを駆動した結果は、XJDemo ボードで LED の1つが光ることでも分かります。


- ピン D2 をダブルクリックします。


ピンをダブルクリックすると駆動されているピンの値が切り替わります (0 に駆動されます)。XJDemo ボードの LED が光らなくなります。

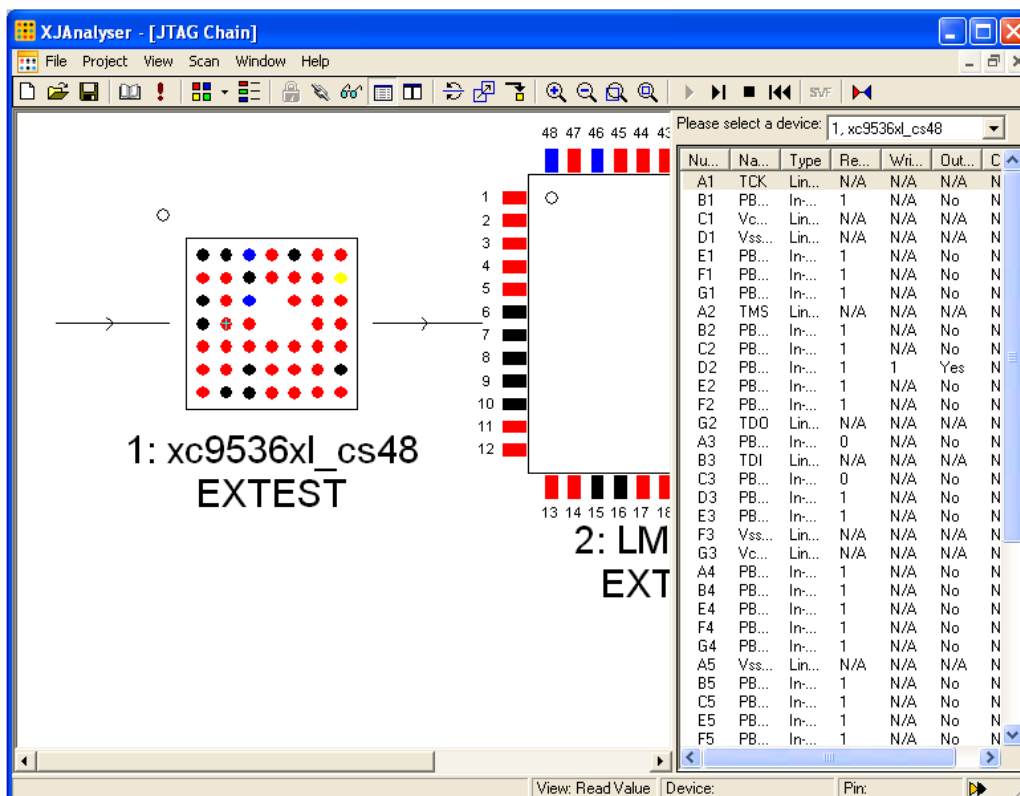
ピンリスト – デバイス上の全てのピンに対して属性を表示

XJAnalyser を使えば、JTAG チェイン上のデバイスの内1つに対して、全てのピンの属性を表示します。

ピンリストの表示

View メニューの Pin List... 又は、ツールバーの  ボタンでピンのリストを表示できます。

- ツールバーの  ボタンをクリックします。



コラム

ピンリスト内のコラムには、各ピンの様々な属性が表示されます。リスト枠内で右クリックし、ポップアップメニューから Columns を選択することで、表示するコラムの種類を選択できます。

- ピンリストを右クリックし、ポップアップメニューから Columns を選択します。
- Type をクリックしてチェックマークを外します。ピンリストが再描画され Type コラムがなくなります。

ピンに移動

ピンリスト内のピンで右クリックして **Goto** を選択することで、特定のピンを探すことができます。

- ピンリストのデバイス選択ドロップダウンリストから **Device 2** を選択します。
- ピン **17** で右クリックし、**Goto** を選択します。選択されたピンを表示する為に必要であれば、表示されているデバイスが切り替わります。ピンの周りを○で囲んでハイライト表示します。

ピンの駆動


メインの **Chain Window** と同様に、ピンリストからも同じ方法でピンを駆動できます。


- ピンを右クリックしピンオプションメニューを表示。メニューの中から、設定したい値を選択。
 - ピンをダブルクリックすることで値が切り替わる。ピンが1ならダブルクリックで0に、逆も同様。
- ピンリストを使って複数のピンの値を変更することも出来ます。**CTRL** キーを押しながらクリックするか、**SHIFT** キーを押しながら選択すると複数のピンを選択でき、選択された全てのピンを駆動できます。ピンオプションメニューを使って駆動する値を設定できます。

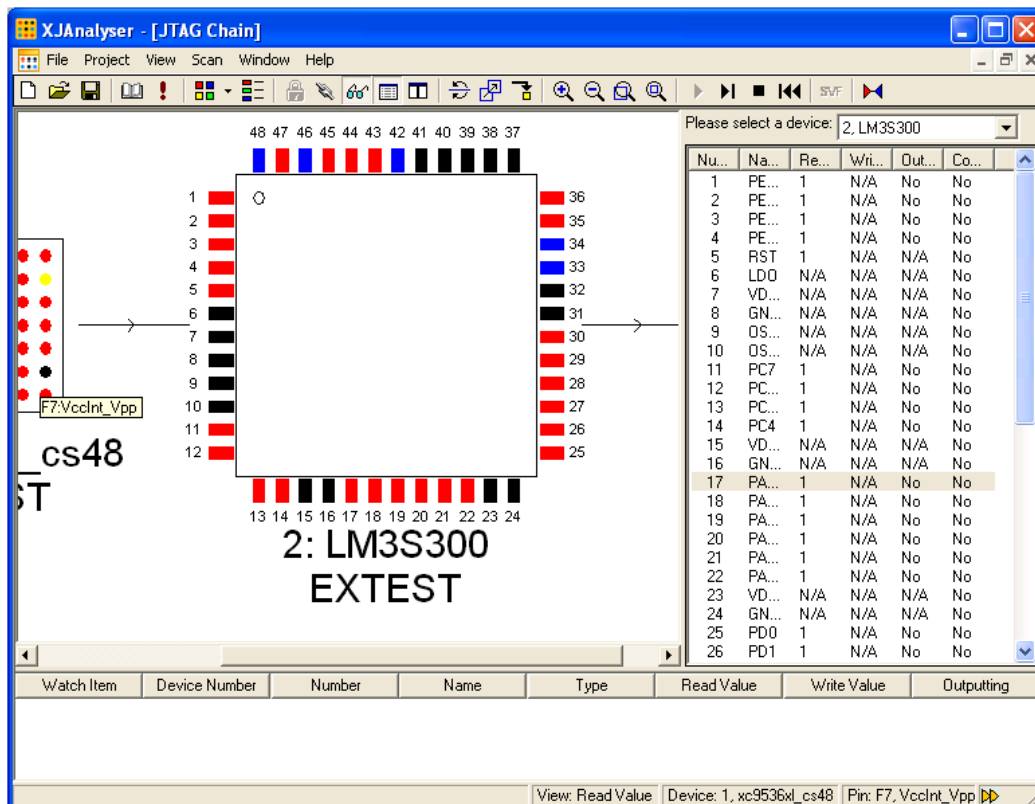
ピンウォッチ - 選択したピンのモニタと制御

XJAnalyser のピンウォッチでは、ピンリストと違って、**JTAG** チェインの1つ以上のデバイスのピンを、同時にモニタ・制御できます。

ピンウォッチの表示

View メニューから **Pin Watch...** を選択するか、ツールバーから  ボタンを選択することで、ピンウォッチを表示できます。

- ツールバーの  ボタンをクリックします。



ピンウォッチへのピンの追加

以下の3つの方法でピンウォッチにピンを追加できます。

- デバイスのイメージから単独のピンをドラッグする
- ピンリストから1つ以上のピンをドラッグする
- ピンウォッチにピンを自動で追加するための1つ以上のモードを選択する

- **Device2** のピン 12 (デバイスの左下端) をピンウォッチ枠にドラッグします。
- **CTRL** キーを押しながら、ピン 3 と 4 をクリックします。
- 選択したピンをピンウォッチ枠にドラッグします。

ピンウォッチ枠で右クリックしピンウォッチメニューを表示することで、ピンウォッチに自動的にピンを追加する以下の2つのモードを設定できます。

- **Monitor Changing Pins** – 値が変化した全てのピンを自動的に追加。
- **Monitor Conflicts** – 衝突が発生した全てのピンを自動的に追加。

※衝突が発生したピンに関しては、チュートリアルの後方で説明します。

- ピンウォッチ枠で右クリックし、**Monitor Changing Pins** を選択します。値が変化している全てのピンがピンウォッチに追加されます。
- **XJDemo** ボードのプッシュスイッチ **SW1** を押します。
- ピン **C3** (**SW1** に接続) がピンウォッチに自動的に追加されます。

ピンの駆動

メインの **Chain Window** と同様に、ピンウォッチからも同じ方法でピンを駆動できます。

- ピンを右クリックしピンオプションメニューを表示。メニューの中から、設定したい値を選択。
- ピンをダブルクリックすることで値が切り替わる。ピンが1ならダブルクリックで0に、逆も同様。

バスの作成

ピンウォッチ枠では、複数のピンをバスの形態にまとめることができます。これによって、ピンに個別に値を設定するのではなく、バスに値を設定できるようになります。

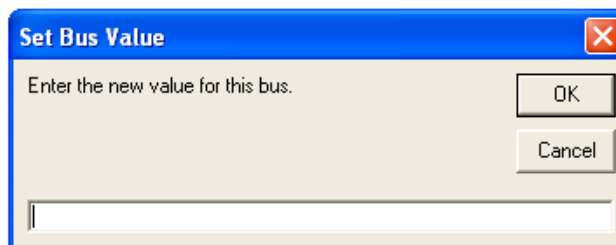
ピンウォッチ内で、1つのピンを別のピンに右ドラッグすることで、バスを作成できます。ドラッグ時にピンを放すことで、“**Create bus**”、“**Drop**”、“**Cancel**”の3つのオプションが表示されます。

バスに他のピンを右ドラッグすることで、さらにピンを追加することができます。

- ピンウォッチでピン3をピン4に右ドラッグします。
- “**Create Bus**”を選択します。
- バスを左クリックし、名前を変更します。バス名は“**LED Control**”とします。

バスに値を設定する

今作成したバスで、**XJDemo** ボード上の **LED** の1つを制御できます。バス上で右クリックし、メニューから“**Set Bus Value**”を選択することで、バス内の全てのピンに値を設定できます。



10進または16進（数字の前に **0x** を付ける）でバスの値を設定できます。値は2進に変換され、バス上のピンを駆動します。

バスから読み出される値は、16進、10進、2進で表示されます。バスを右クリックしてコンテキストメニューから **Display As** を選択することで、表示される値の基数を変更できます。

- バス上で右クリックし **Set Bus Value** を選択します。
- **3**を入力して **OK** をクリックします。バスの **Write Value** カラムが **3** になります。LED **D1** と **D2** が光ることも確認できます。
- バスの左の+マークをクリックして、バス内の全てのピンを表示します。両方のピンの **Write Value** の値が1になっていることが確認できます。
- バスの値に **2** を設定します。バスの **Write Value** が2に変わり、ピンの値が0と1に変わり、**D1** だけが点灯します。

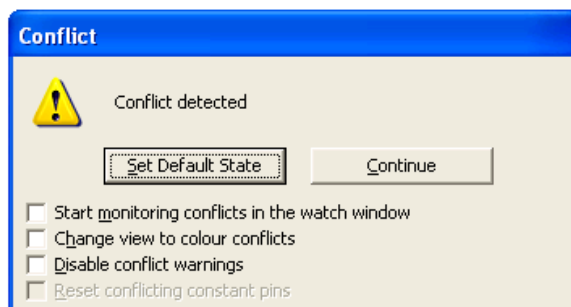
ピンに移動

ピンリスト内のピンで右クリックして **Goto** を選択することで、ピンを持つデバイスを表示して、特定のピンを探すことができます。

Conflicting pins – XJAnalyser がピンの衝突を処理する方法

ピンに書き込んだ値と読み取った値が異なった場合に衝突が発生します。

- デバイス 1 のピン C3 を 1 から 0 に変更します。
- XJDemo ボードの SW1 を押し、衝突が発生させます。



Conflict ダイアログでは、以下の 2 つのオプションが選択できます。



- **Continue** – ピンに対して衝突する値（ネット上で衝突の原因となった値）を書き込み続けます。
- **Set Default State** – 全てのピンをデフォルトの状態に戻し、可能な限りピンを入力にします。このオプションを選択すると、回路内の衝突が解消されます。

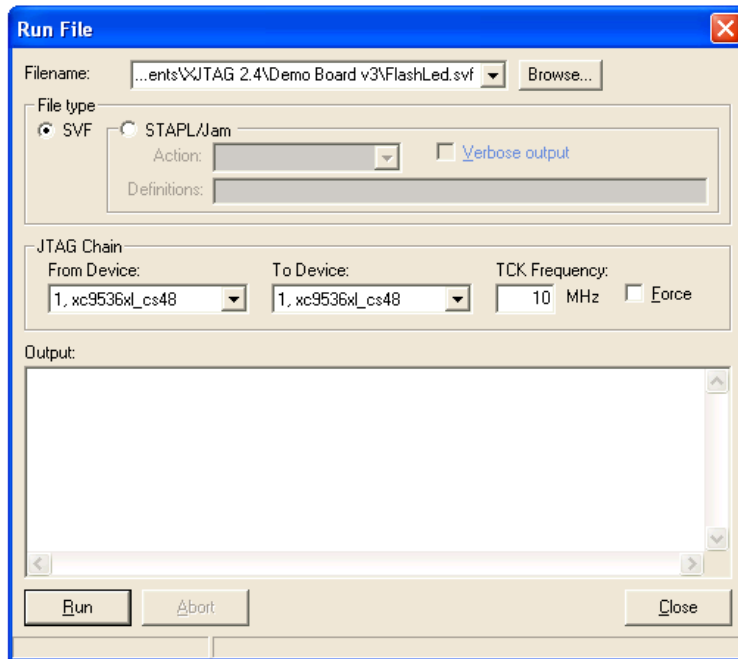
XJAnalyser の構成に影響を与える幾つかのオプションが選択できます。詳細は、**XJAnalyser Help**→**User Guide**→**Using XJAnalyser**→**Further information** の **Conflicting Pins** を参照してください。

- **Continue** をクリックして **XJAnalyser** を続行します。

XJAnalyser – デバイスプログラミング

XJAnalyser には、事前に作成された STAPL や SVF ファイルを使って、デバイスのプログラミングを行うためのグラフィカルなインターフェースがあります。

- ツールバーの **Stop** ボタン  をクリックし、**XJAnalyser** のチェーンのスキャンを停止します。
- ツールバーの **SVF** ボタン  をクリックし、**Run File** ダイアログを開きます。（**Run File** ダイアログの詳細は、**XJAnalyser Help**→**Reference**→**The Dialogs**→**The Properties Dialogs**→**Run File Dialog** を参照）



このダイアログを使って、実行したいファイルとプログラムしたいデバイスの範囲を規定できます。

- **Browse...** ボタンをクリックし、**SVF** ファイルを指定します。
- **FlashLed.svf** を選択します。このファイルは **Demo Board v3** ディレクトリにあります。
- **From Device** と **To Device** を **Device1** にセットします。
- **TCK Frequency** を **10** にセットします。
- **Run** をクリックします。ダイアログ下部にプログラミングプロセスの進行状況が表示されます。
- プログラミングが完了すれば、**SW1** を押し続けることで、**D4** が点滅します。

これでチュートリアルは終了です。XJAnalyser についての更なる情報はヘルプを参照ください。また、XJAnalyser を使って、独自のハードウェアをテストすることもできるでしょう。