



XAPP1212 (v1.0) 2015 年 1 月 9 日

# KC705 評価キットで Aurora 64B/66B コア (シンプルレックス) を使用するシステムを設計

著者 : Dinesh Kumar, Ramachandra Thupalli, K Krishna Deepak

## 概要

このアプリケーション ノートでは、Kintex®-7 FPGA KC705 評価キットでザイリンクスの LogiCORE™ Aurora 64B/66B IP コアを検証するために必要な手順を説明します。Aurora 64B/66B コアは、スケーラブル、軽量、そして高データ レートの高速シリアル通信向けのリンク レイヤー プロトコルです。Aurora は、直感的なウィザード インターフェイスを使用して、ザイリンクスのトランシーバーを簡単に実装することを目的としています。Aurora プロトコルの仕様は公開されており、リクエストに応じて提供されます。Aurora コアは Vivado® IP カタログから無償で利用可能で、ライセンスを取得してザイリンクスのシリコン デバイスで使用できます。

一般的に Aurora は、ほかの業界標準シリアル インターフェイスでは複雑すぎたり、リソースを消費しすぎるといったアプリケーションで使用されます。Aurora は、低コスト、高データ レート、スケーラブル、そして柔軟なシリアル データ チャンネルを構築できます。そのシンプルなフレーム構造は、既存プロトコルからのデータを容易にカプセル化でき、また電氣的要件も汎用システムと互換性があります。Aurora の使用によって、FPGA リソースの大量消費やソフトウェアの再開発、または物理的なインフラを新たに構築することなくパフォーマンスを向上させることができます。

リファレンス デザインは、Kintex-7 FPGA KC705 評価ボードをターゲットにしています。

## 含まれるシステム

リファレンス デザインは、Vivado Design Suite : System Edition 2014.1 を使用して作成および構築されています。Vivado Design Suite を利用することによって、IP ブロックをインスタンスエート、コンフィギュレーション、および接続して複雑な統合システムを構築する作業が簡略化されます。リファレンス デザインには、信号をプローブするための VIO および ILA コアも含まれています。

## はじめに

このアプリケーション ノートでは、Vivado Design Suite を使用して Aurora 64B/66B コアをコンフィギュレーションし、VIO および ILA コアでさまざまな信号をプローブしてシンプルレックス (単方向通信) モードの Aurora コアの動作を検証する手順を詳しく説明します。

ここで提供するサンプル デザインは、2 つのプラットフォームを使用するシングル レーン シンプルレックス コンフィギュレーションを示しています (図 1)。より複雑なシステムの構築ブロックを作成するために、完成したサンプル デザインを利用することが可能です。

サンプル テスト セットアップでは、2 つのクロック ソースを使用して 156.25MHz クロック信号を生成します。これらのリファレンス デザインの再現には、適切に調整された 156.25MHz クロック ソースであればどれでも利用可能です。

© Copyright 2015 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

本資料は表記のバージョンの英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。資料によっては英語版の更新に対応していないものがあります。日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。

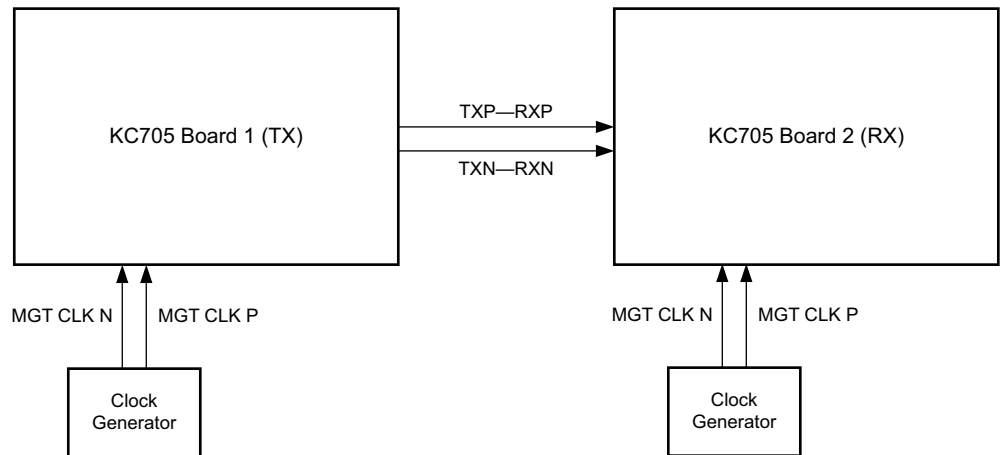


図 1: シンプレックス リファレンス デザイン

## ハードウェア要件

シングルレーン シンプレックス コンフィギュレーションには、次のハードウェア コンポーネントが必要です。

- Kintex-7 FPGA KC705 評価ボード (x 2)
- KC705 ユニバーサル 12v 電源アダプター (x 2)
- 156.25MHz の生成に適したクロック ジェネレーター (x 2)
- JTAG プラットフォーム USB ケーブル (x 2)
- 両端 SMA コネクタ付きケーブル (x 4) (基準クロック用)
- 両端 SMA コネクタ付きケーブル (x 2) (シリアル データ用)

## ソフトウェア要件

Aurora 64B/66B シンプレックス サンプル デザインのソフトウェア要件は次のとおりです。

- Vivado Design Suite 2014.1

## ハードウェアの構築

### シンプレックス サンプル デザイン

#### Aurora コアのカスタマイズ

次の手順に従って、シンプレックス サンプル デザイン用に Aurora 64B/66B コアをカスタマイズして生成します。

1. Vivado Design Suite を起動します。
2. [Create New Project] をクリックして [Next] をクリックします (図 2)。

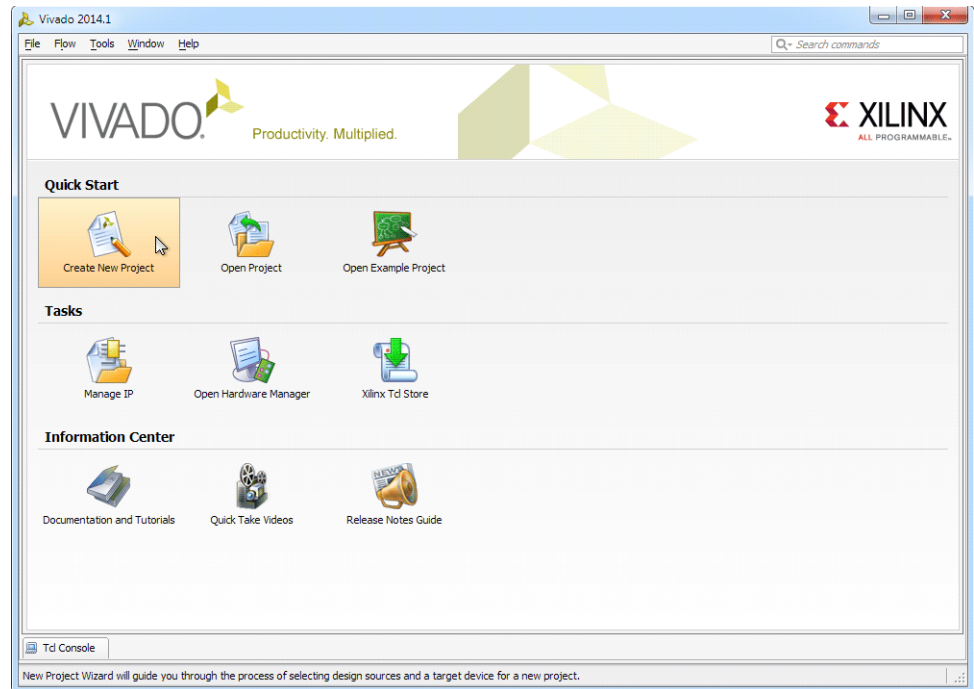


図 2 : Vivado ツールの新規プロジェクトの作成

3. プロジェクト名とパスを選択して [Next] をクリックします (図 3)。

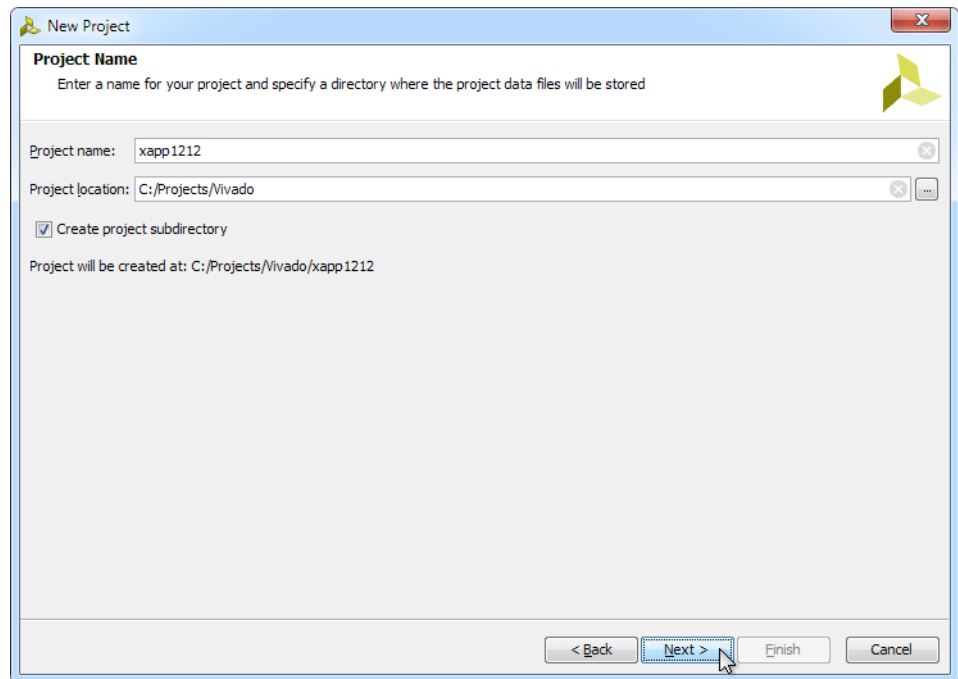


図 3 : 新しいプロジェクトの名前を設定

4. [RTL Project] をオンにしてサンプル デザインの実行を許可し、[Do not specify sources at this time] をオンにします (図 4)。[Next] をクリックします。

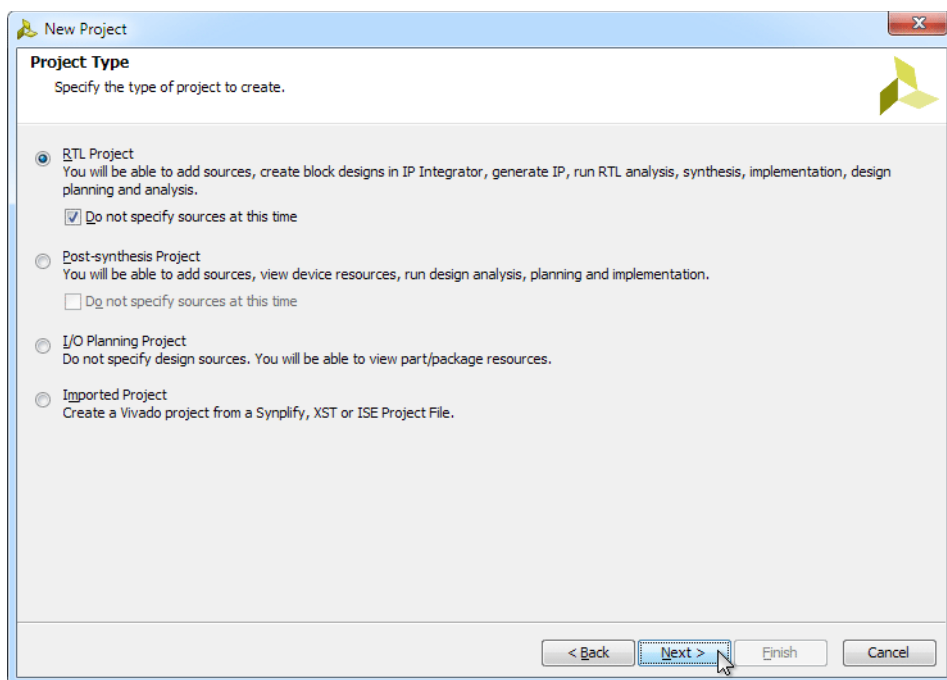


図 4 : 新しいプロジェクトのタイプを設定

5. [xc7k325tffg900-2] をクリックするか、または [Boards] をクリックして [Kintex-7 KC705 Evaluation Platform] をクリックします (図 5)。

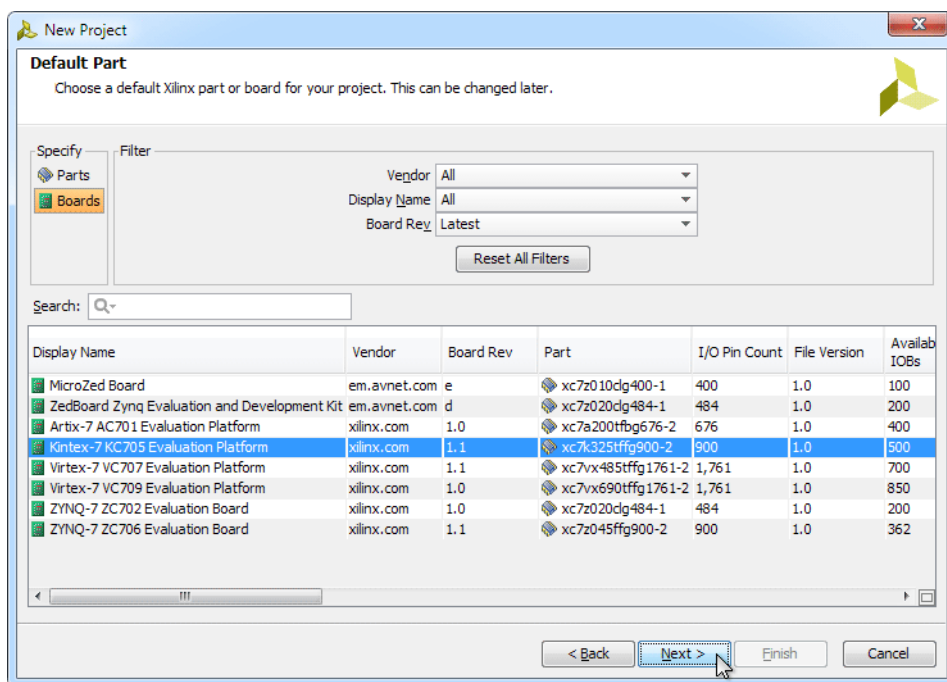


図 5 : [New Project] の [Default Part] ページ

6. [Next] をクリックして [Finish] をクリックします
7. Flow Navigator の [Project Manager] 下にある [IP catalog] をクリックし、「Aurora 64B66B」を検索します。Aurora コアは、[Communication & Networking] → [Serial Interfaces] の下にあります (図 6)。

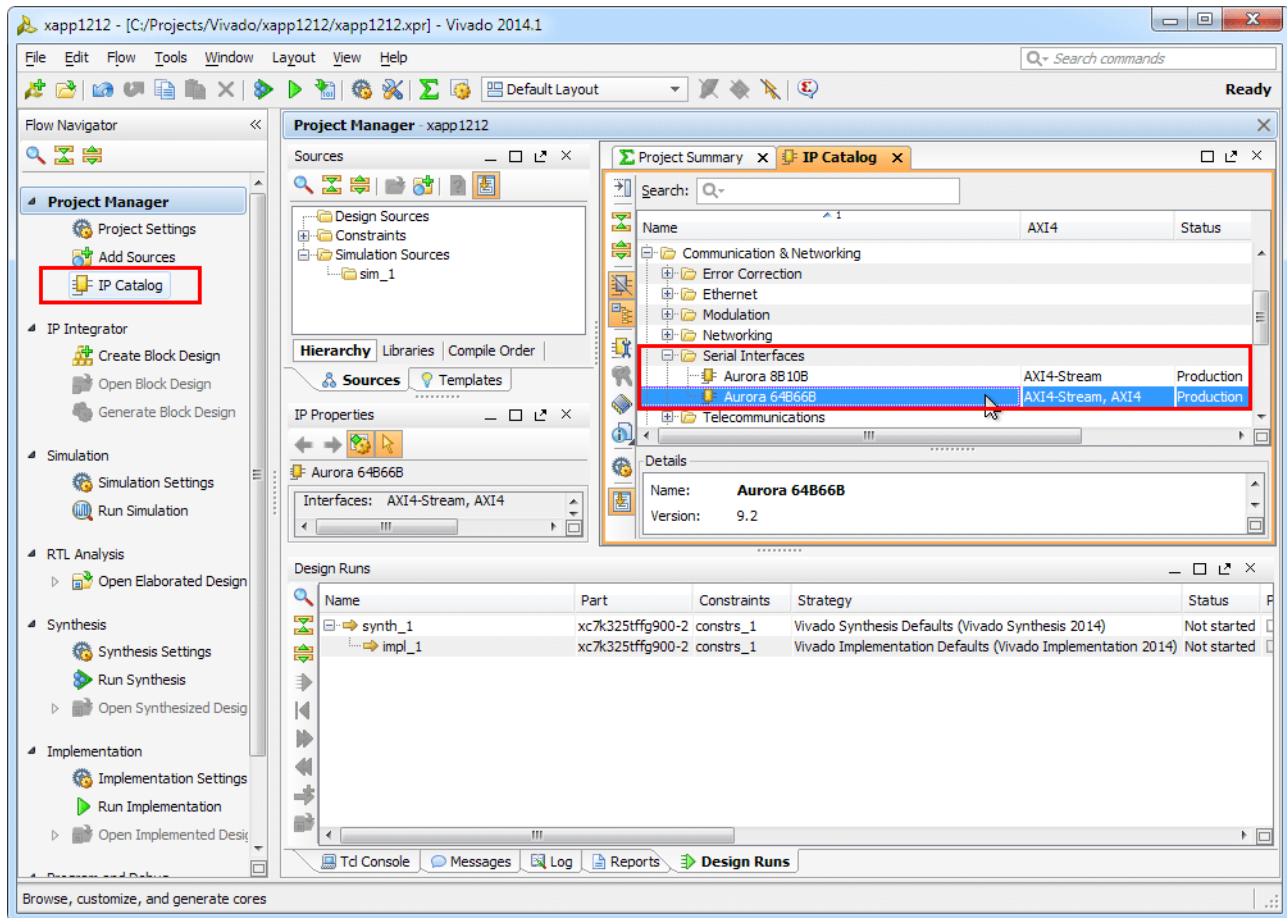


図 6 : Vivado IP カタログの Aurora 64B/66B コア

- [Aurora 64B66B] を右クリックして [Customize IP] をクリックします。

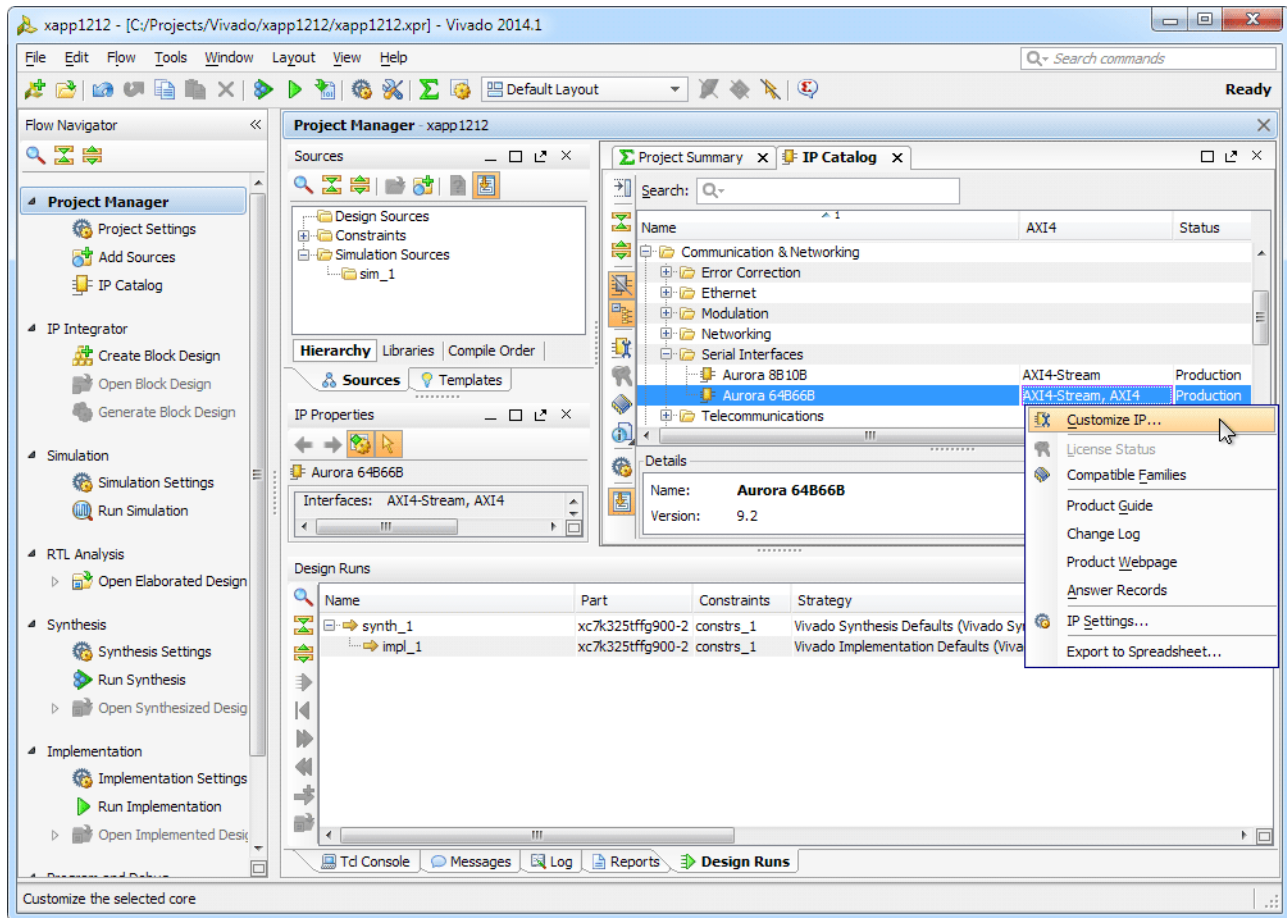


図 7 : [Customize IP] の選択

9. [Customize IP] ダイアログ ボックスの [Core Options] タブで、次のようにオプションを設定します (図 8 参照)。
  - [Line Rate (Gb/s)] に [3.125]、[GT Refclk (MHz)] に [156.250] を指定します。
  - コンフィギュレーションされるプラットフォームに応じて、[Dataflow Mode] に [TX-only Simplex] または [RX-only Simplex] を指定します。
  - [Interface] に [Framing]、[Flow Control] に [None] を指定します。
  - [Vivado Lab Tools] オプションをオンにします。

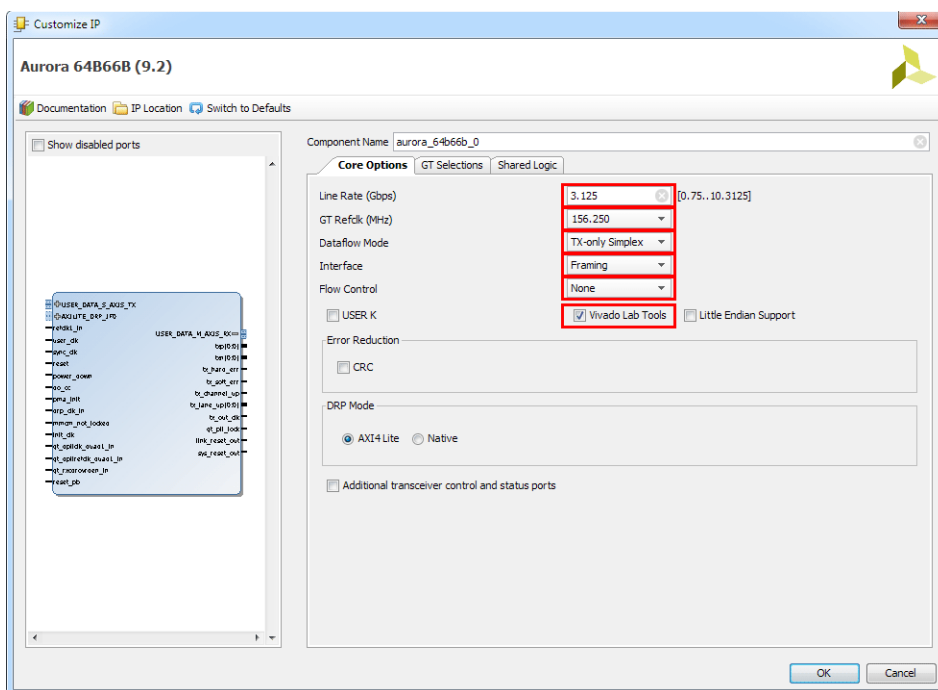


図 8 : Aurora 64B/66B シンプレックス コアのオプション設定

10. [GT Selections] タブをクリックします。
11. GTXQ0 のリスト ボックス左下のデフォルト設定「1」を「X」に変更します。
12. GTXQ2 のリスト ボックス左下の設定「X」を「1」に変更します (図 9)。

注記 : GTXQ2 トランシーバーは、KC705 ボードの SMA コネクタへ割り当てられる唯一のトランシーバーです。リスト ボックスの設定の上にカーソルを置くと、ツールチップが表示されて選択したトランシーバーの位置を確認できます。

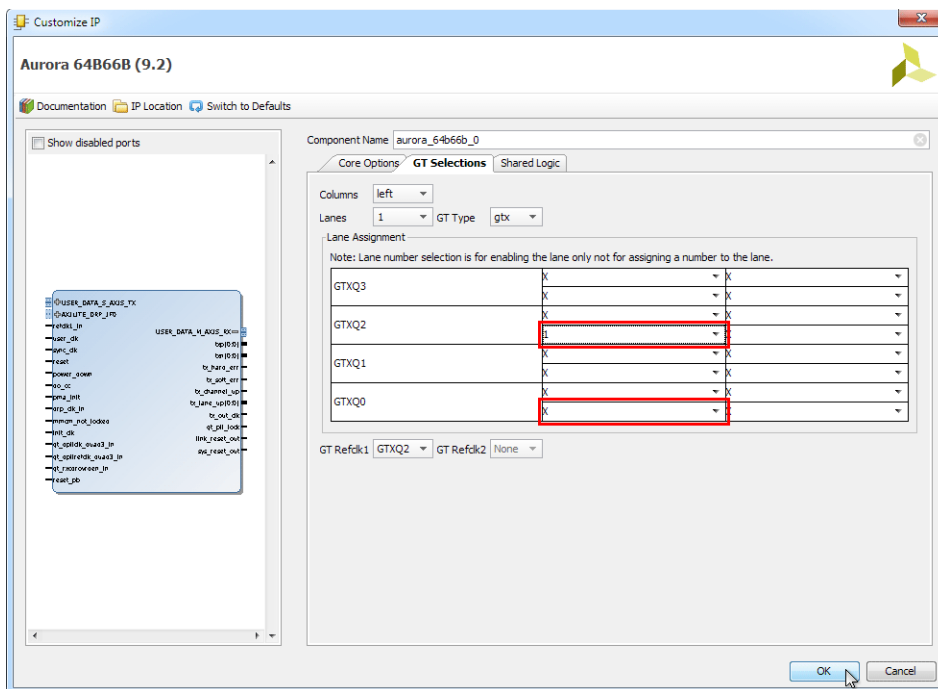


図 9 : Aurora 64B/66B シンプレックス GT の選択

13. [Shared Logic] タブのオプションはデフォルト値のまま変更しないでください。[OK] をクリックします。
14. [Generate Output Products] ダイアログ ボックスで、[Generate] をクリックします。

### サンプル デザインの合成

1. 出力ファイルの生成が完了したら、Vivado IDE の [Project Manager] でコア名を右クリックして [Open IP Example Design] をクリックします (図 10)。

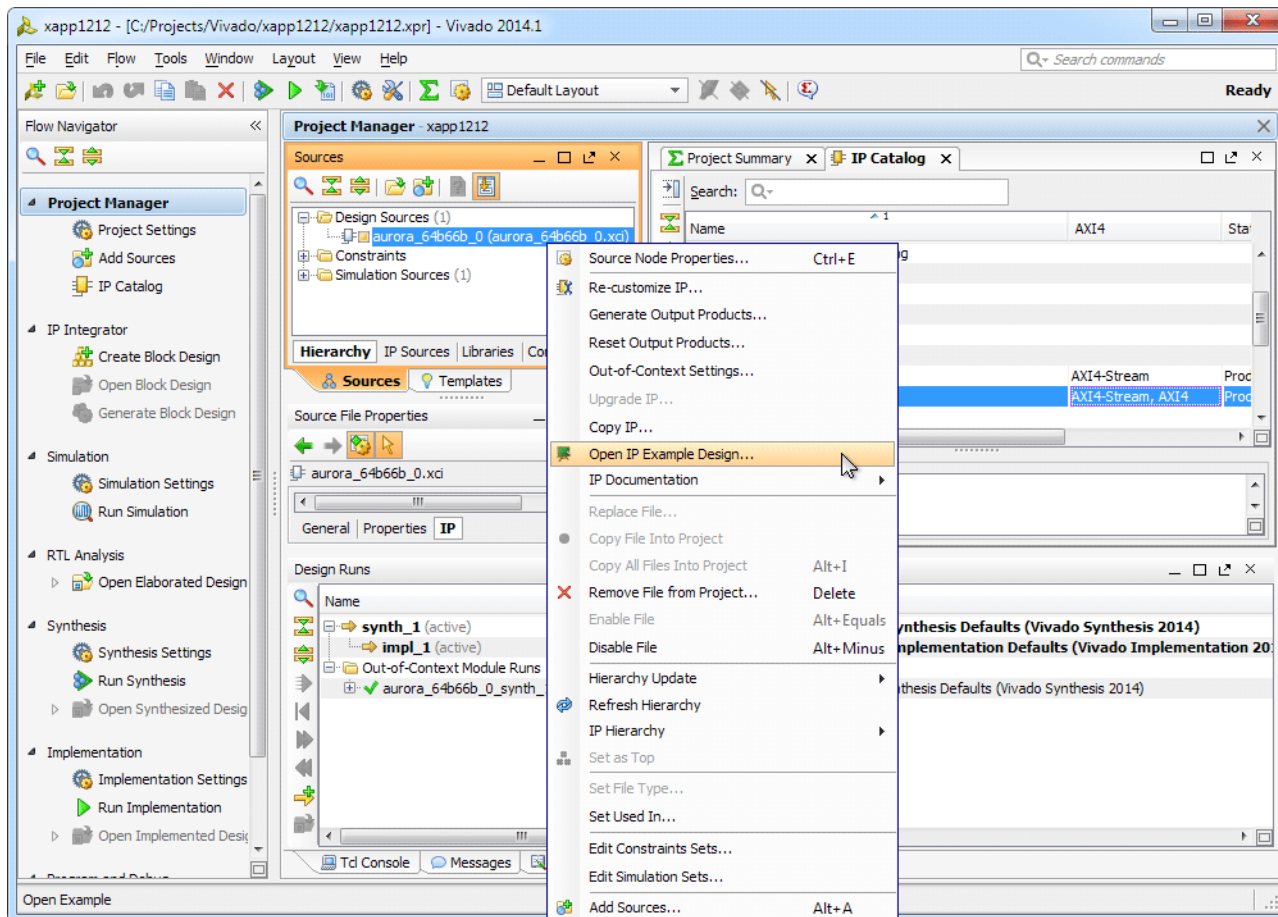


図 10 : IP サンプル デザインを開く

2. [OK] をクリックして既存のサンプル デザインを上書きします。
3. 新しく開いた Vivado IDE ウィンドウの [Project Manager] の [Sources] ビューで [Constraints] を展開します。制約ファイル (aurora\_64b66b\_0\_exdes.xdc) を右クリックして、[Open file] をクリックします (図 11)。

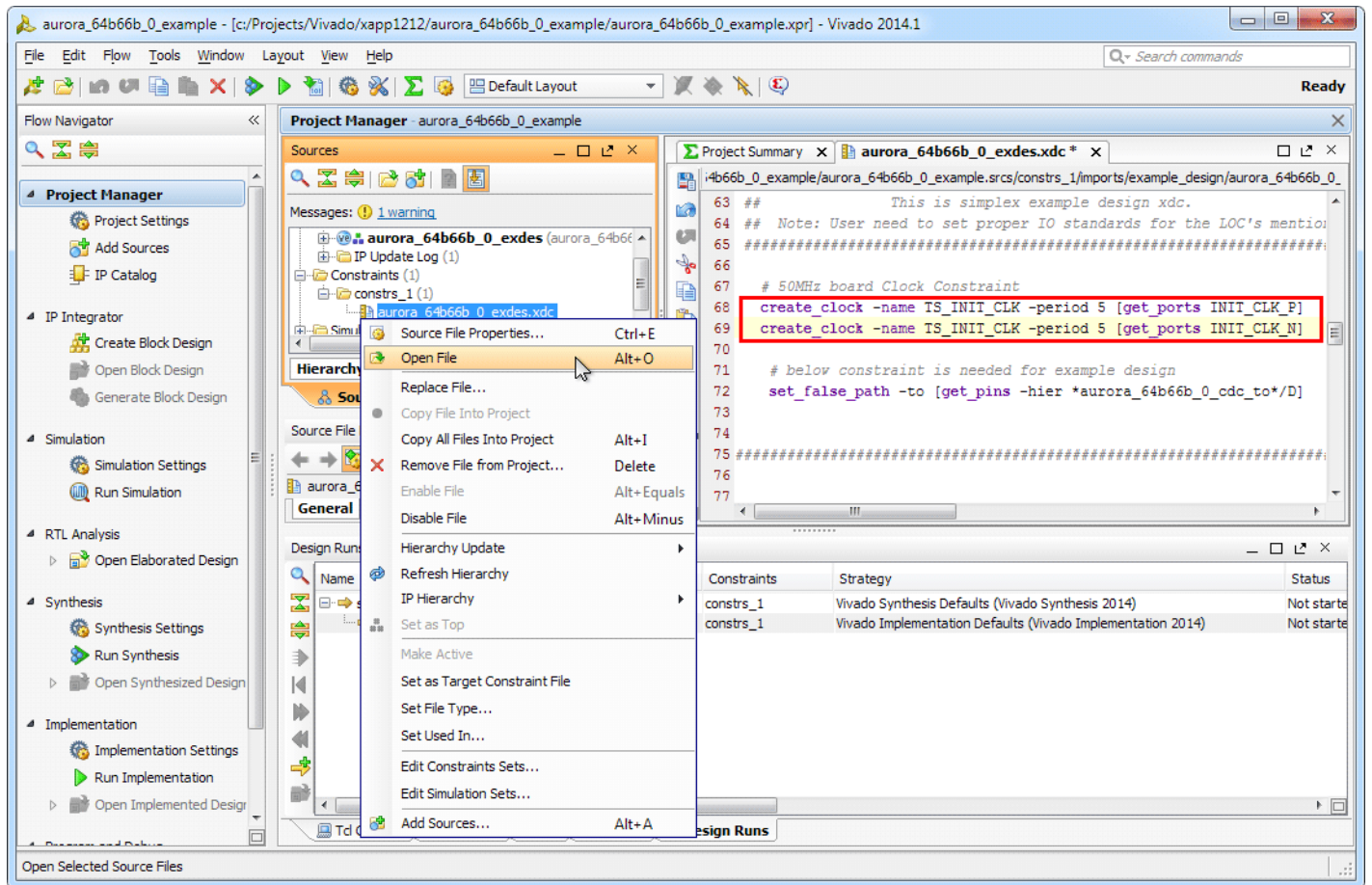


図 11：制約ファイルを開く

4. 50MHz のボード クロックに関する 2 つの制約を検索します (図 11 参照)。
5. オンボードの 200MHz クロックを供給するには、クロック周期を 20ns から 5ns に変更します。修正後の制約ステートメントは次のようになります。  

```
create_clock -name TS_INIT_CLK -period 5 [get_ports INIT_CLK_P]
create_clock -name TS_INIT_CLK -period 5 [get_ports INIT_CLK_N]
```
6. 表 1 に示すとおり Aurora コア ポートのピン位置を割り当てます (図 12 参照)。

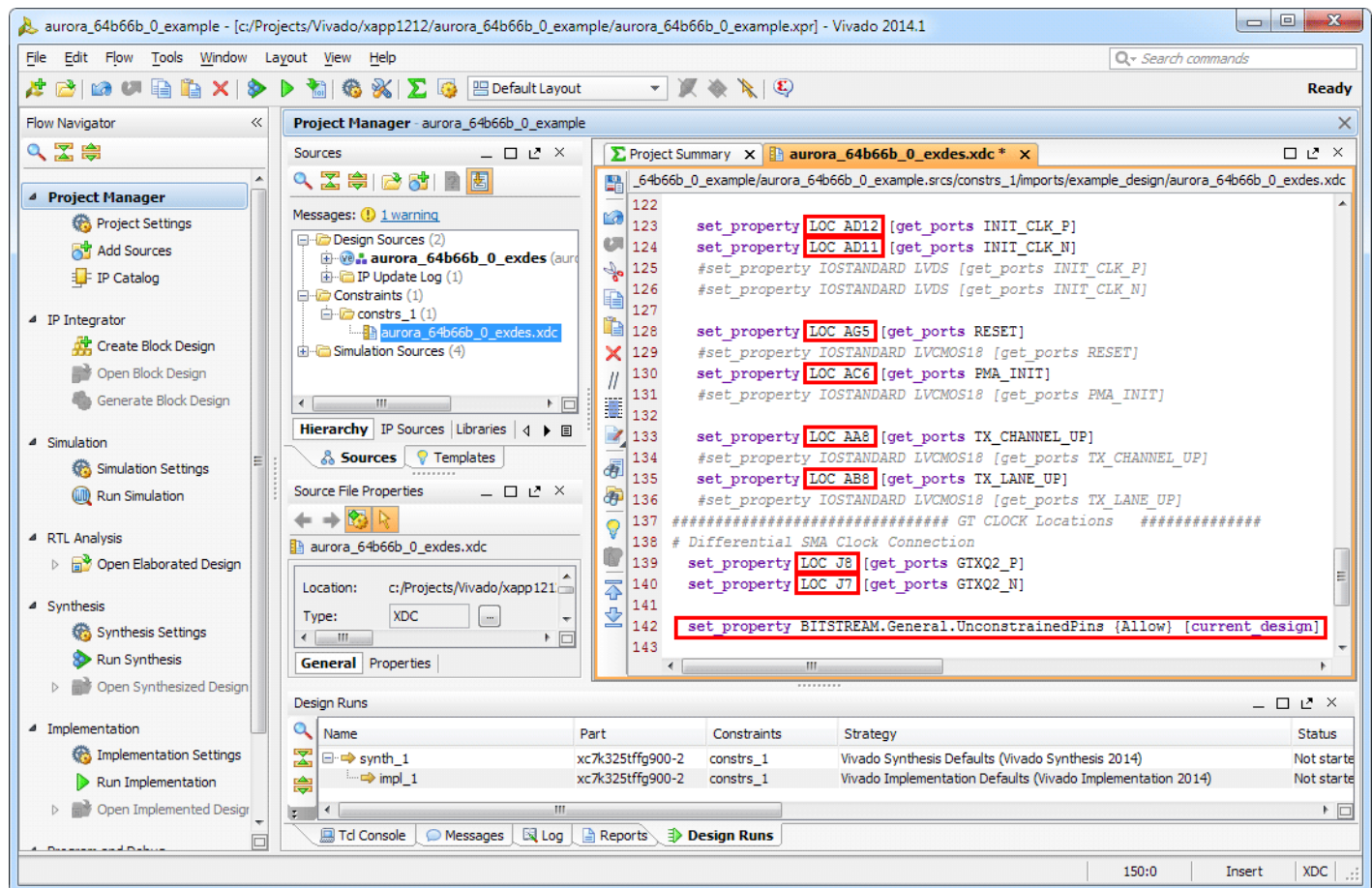


図 12 : Aurora 64B/66B シンプレックスの LOC 制約

表 1 : Aurora 64B/66B シンプレックスの制約

ピン名	LOC 値
INIT_CLK_N	AD11
INIT_CLK_P	AD12
RESET	AG5
PMA_INIT	AC6
TX_CHANNEL_UP/RX_CHANNEL_UP	AA8
TX_LANE_UP/RX_LANE_UP	AB8
GTXQ2_N	J7
GTXQ2_P	J8

- このサンプル デザインには、制約されていないピンが含まれています。ビットストリーム ファイルの生成を可能にするには、制約ファイルの最後に次の行を追加してください (図 12)。  
`set_property BITSTREAM.General.UnconstrainedPins {Allow} [current_design]`  
**注意：**スペリングに注意します。制約ファイルの変更箇所をダブルチェックしてから次の手順へ進んでください。
- 制約ファイルのエディター ウィンドウで右クリックし、[Save File] をクリックします。制約ファイルのエディター ウィンドウを閉じます。
- Flow Navigator で [Generate Bitstream] をクリックします。

10. [Yes] をクリックして合成とインプリメンテーションを実行し、ビットストリームの生成を開始します。
11. 「Aurora コアのカスタマイズ」および「サンプルデザインの合成」の手順を繰り返し、各プラットフォームのビットストリーム ファイルを生成します。
  - 送信プラットフォームの場合は、[Dataflow Mode] に [TX-only Simplex] を指定します。
  - 受信プラットフォームの場合は、[Dataflow Mode] に [RX-only Simplex] を指定します。

## ハードウェア上でのリファレンス デザインの実行

### シンプレックス サンプル デザインのセットアップ

このサンプル デザインは、2つのプラットフォームにおけるシングル レーン Aurora 64B/66B シンプレックスの接続を示しています (2 ページの図 1 参照)。プラットフォームは、2つの Kintex-7 FPGA KC705 評価キット ボードで構成されています (図 13)。

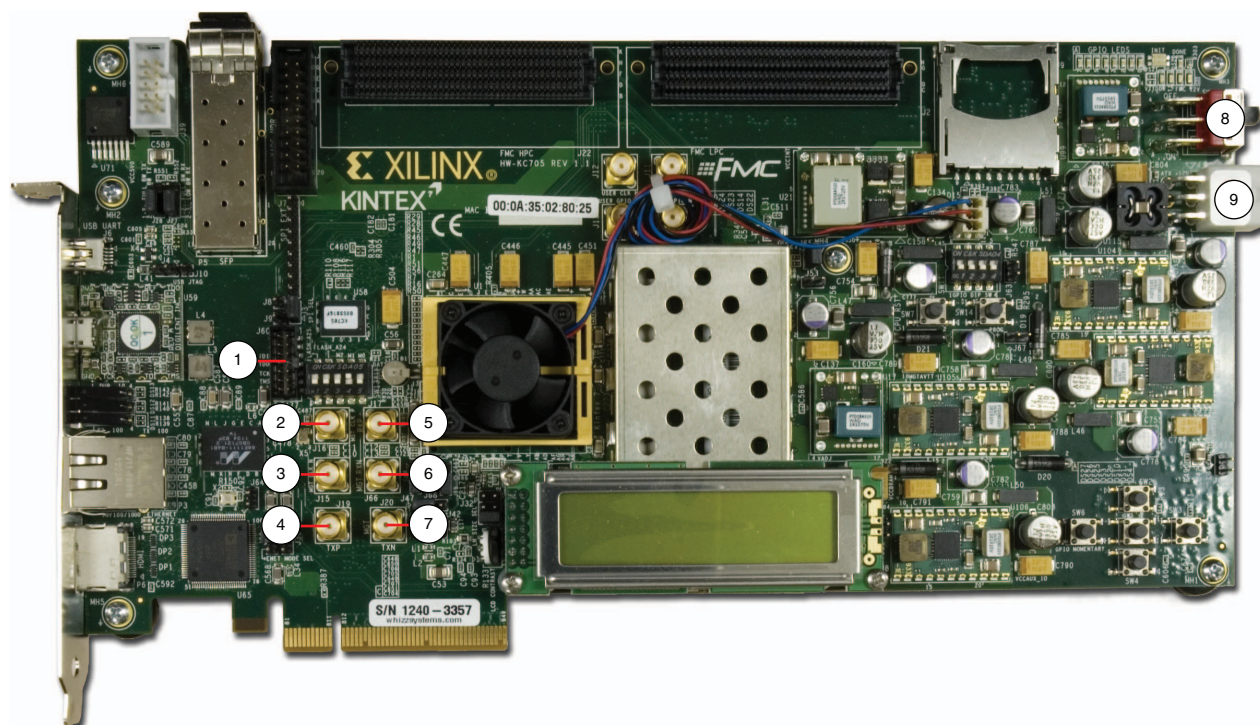


図 13 : KC705 ボードの画像

次の手順のカッコ内の番号は図 13 に示す番号に対応しています。両端 SMA コネクタ付きケーブルを使用して、これらの接続を行います。

- ボード 1 の TXP (4) をボード 2 の RXP (5) へ接続します。
- ボード 1 の TXN (7) をボード 2 の RXN (6) へ接続します。
- クロック ソース 1 の CLKP をボード 1 の MGT CLK P (2) へ接続します。
- クロック ソース 1 の CLKN をボード 1 の MGT CLK N (3) へ接続します。
- クロック ソース 2 の CLKP をボード 2 の MGT CLK P (2) へ接続します。
- クロック ソース 2 の CLKN をボード 2 の MGT CLK N (3) へ接続します。
- ホスト PC の JTAG プラットフォーム USB ケーブルをボード 1 のプラットフォーム ケーブル ヘッダー (1) へ接続します。
- ホスト PC の JTAG プラットフォーム USB ケーブルをボード 2 のプラットフォーム ケーブル ヘッダー (1) へ接続します。
- KC705 ユニバーサル 12v 電源アダプター ケーブルを両方のボードの電源コネクタ (9) へ接続します。
- 両方のボードの電源スイッチ (8) を ON の位置に設定します。

セットアップが完了すると、[図 14](#) のようになります。

注記：各ボードにはそれぞれ独立したクロック ソースを使用してください。

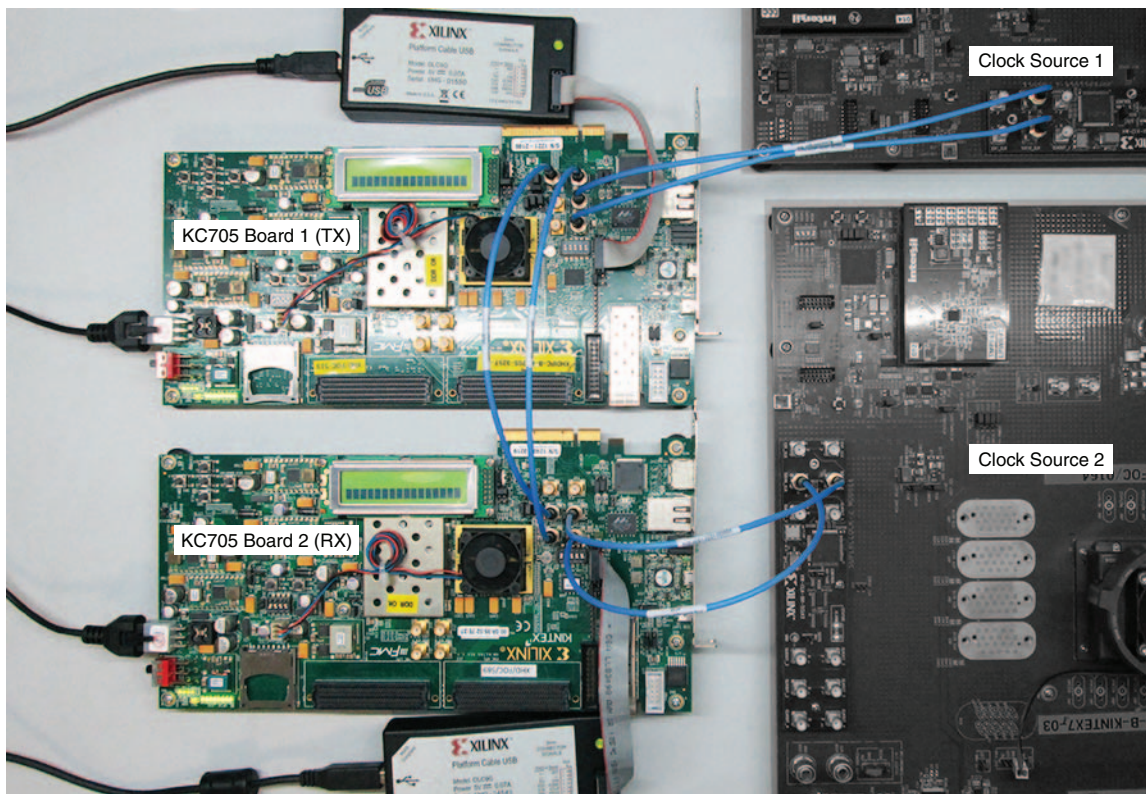


図 14 : Aurora 64B/66B シンプレックスのセットアップ

## シンプレックスのサンプル デザイン セッションのセットアップ

### デバイスのプログラム

1. ビットストリームの生成が完了したら、[Flow] → [Open Hardware Manager] をクリックします ([図 15](#))。

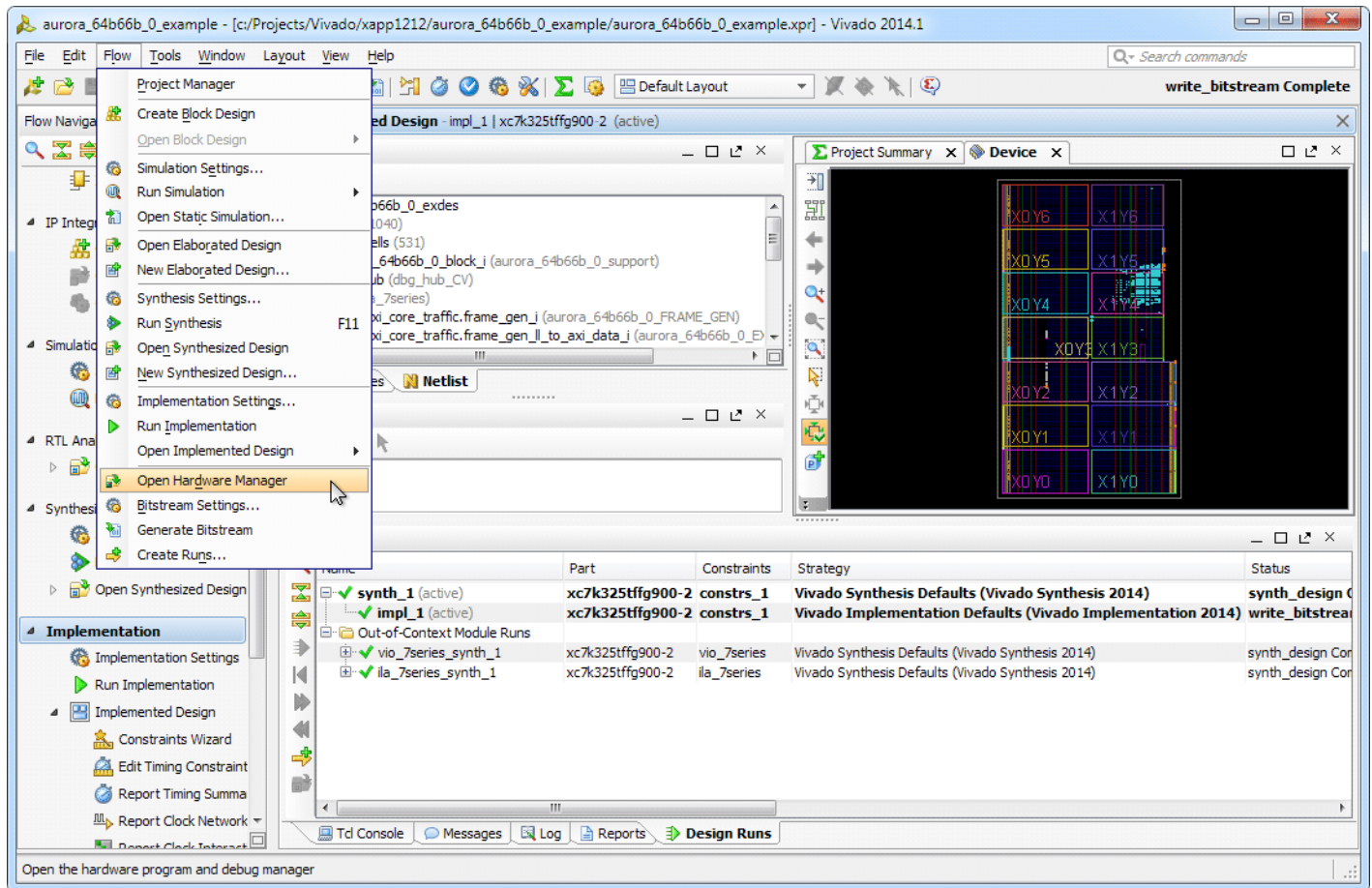


図 15 : [Open Hardware Manager] の選択

2. [Hardware Manager] ページの上部にある [Open a new hardware target] をクリックして [Next] をクリックします (図 16)。

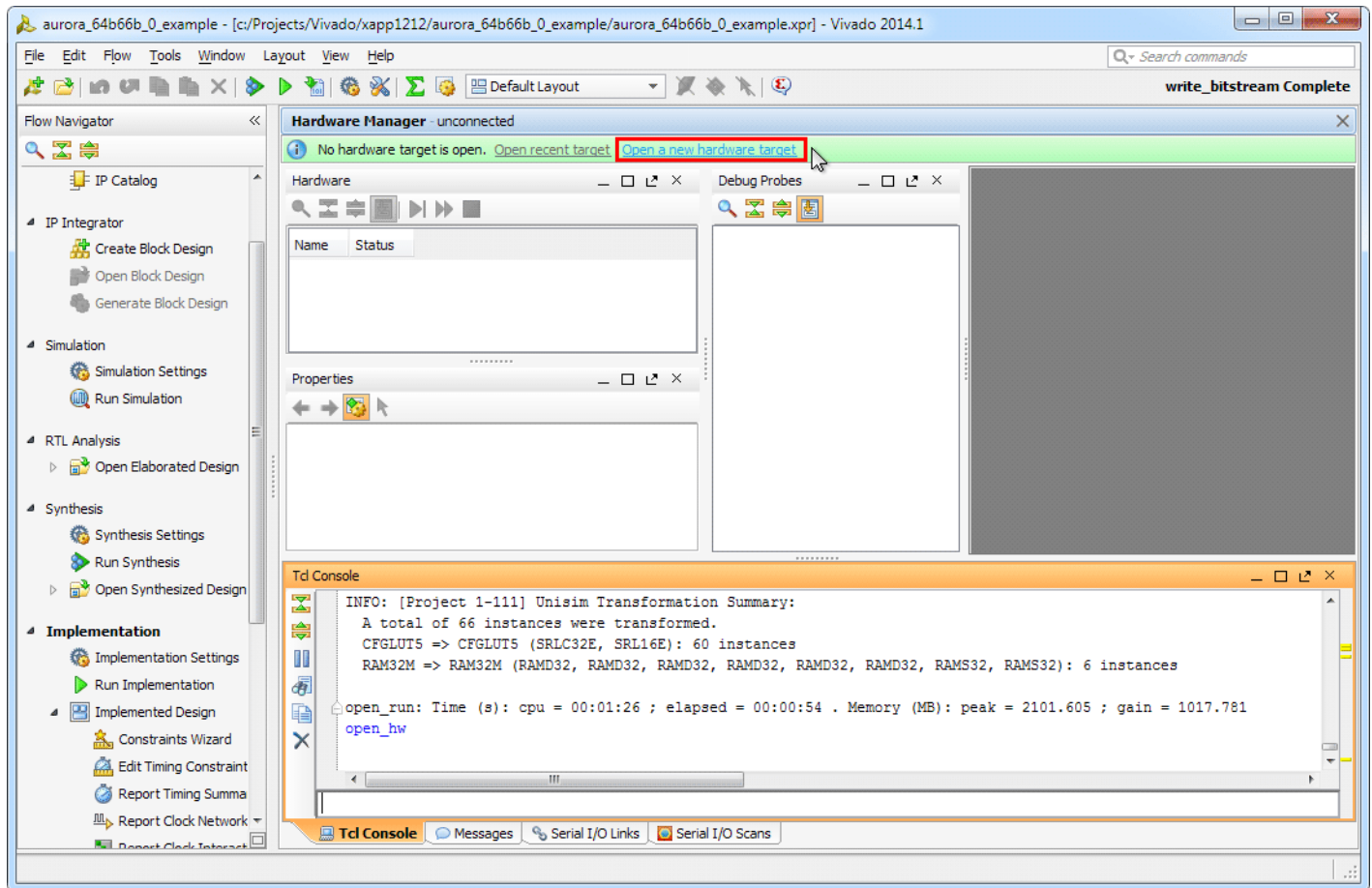


図 16 : [Open a New Hardware Target] の選択

3. [Local server] を選択して [Next] をクリックします (図 17)。

注記：この手順は、ハードウェア ターゲットが Vivado Design Suite を駆動するホスト PC へ接続されていることを前提とします。Vivado CSE Server アプリケーションを使用するネットワーク上の 2 番目のホスト PC へハードウェア ターゲットを接続することも可能です。詳細は、『Vivado Design Suite ユーザー ガイド：プログラムおよびデバッグ』(UG908) [参照 4] を参照してください。

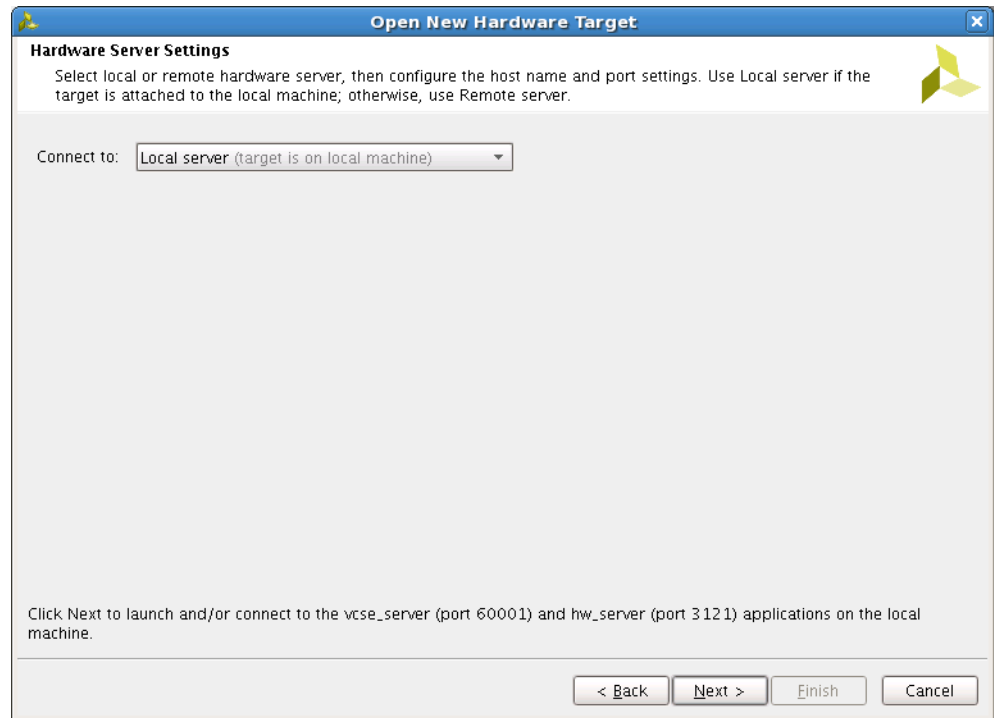


図 17 : [Hardware Server Settings]

4. [Select Hardware Target] ページで、両方のボードについて [JTAG Clock Frequency] に [750000Hz] を指定します (図 18)。

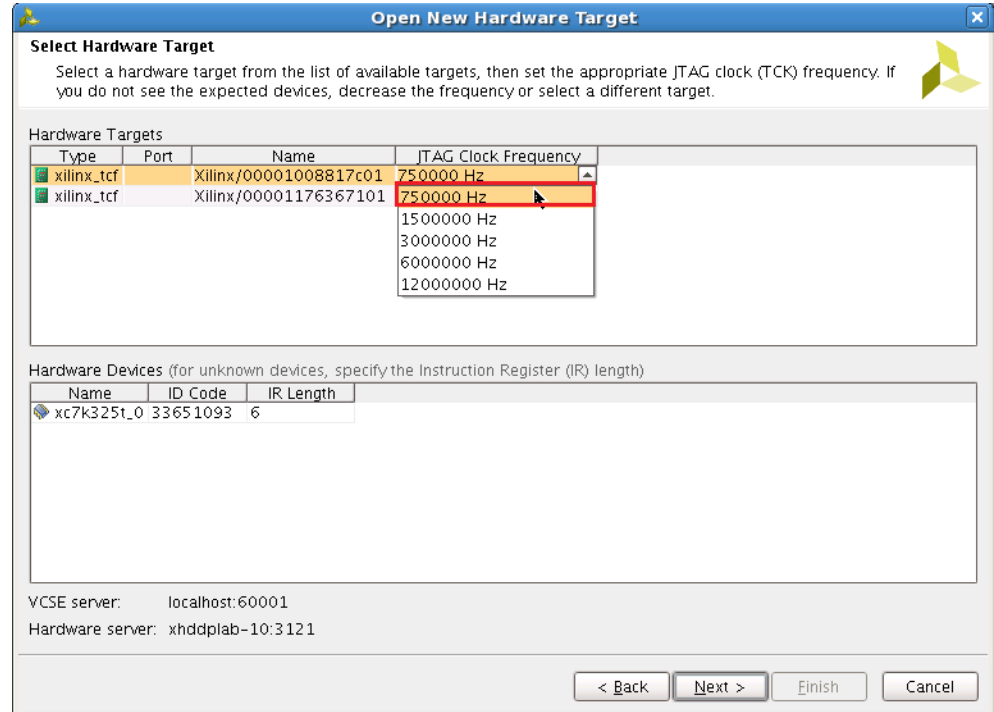


図 18 : [Select Hardware Target]

5. プログラムするターゲット ボードを選択し、[Next] をクリックして [Finish] をクリックします。
6. [Hardware] ビューでアクティブなデバイス [XC7K325T\_0 (0)] をクリックします。

7. [Hardware Device Properties] ビューで、[Programming file] に受信プラットフォームのビットストリーム名 (aurora\_64b66b\_0\_exdes.bit) を指定し、[Probes file] に対応する .ltx プローブ ファイル名 (debug\_nets.ltx) を指定します。図 19 を参照してください。

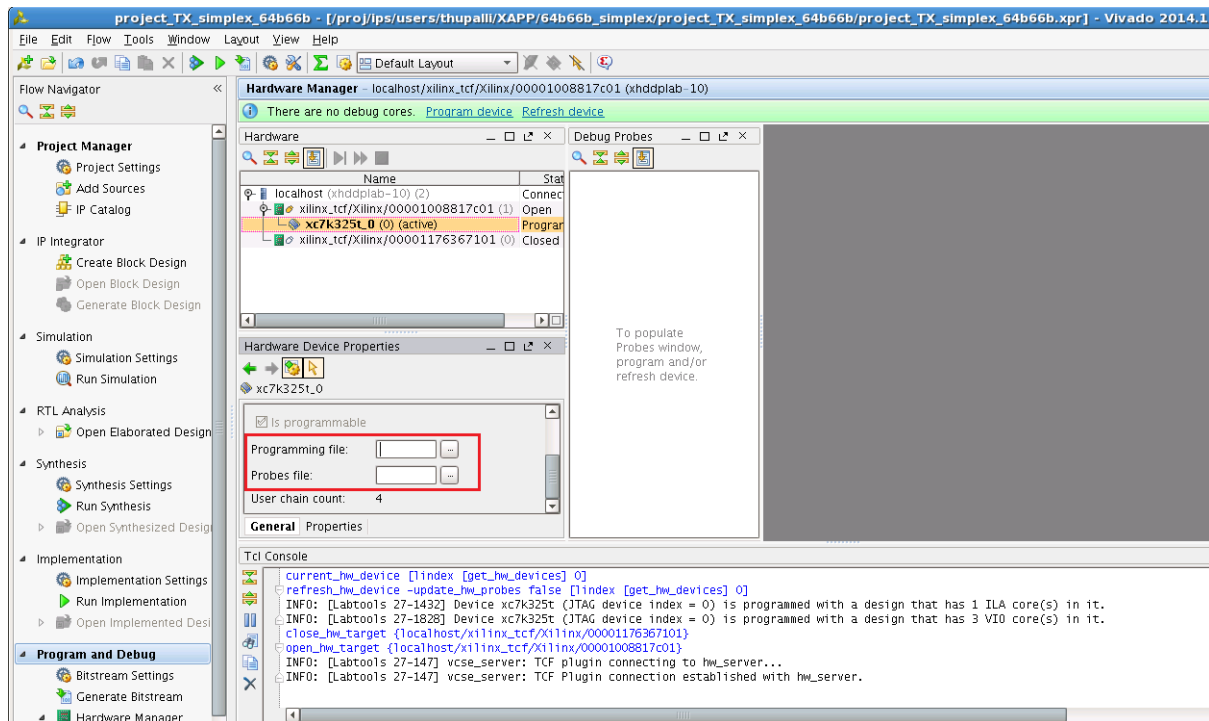


図 19 : [Hardware Device Properties] ビュー

8. [Hardware] ビューでデバイスを右クリックし、[Program Device] をクリックします (図 20)。ビットストリーム ファイルのパスと名前が正しいことを確認して [OK] をクリックします。

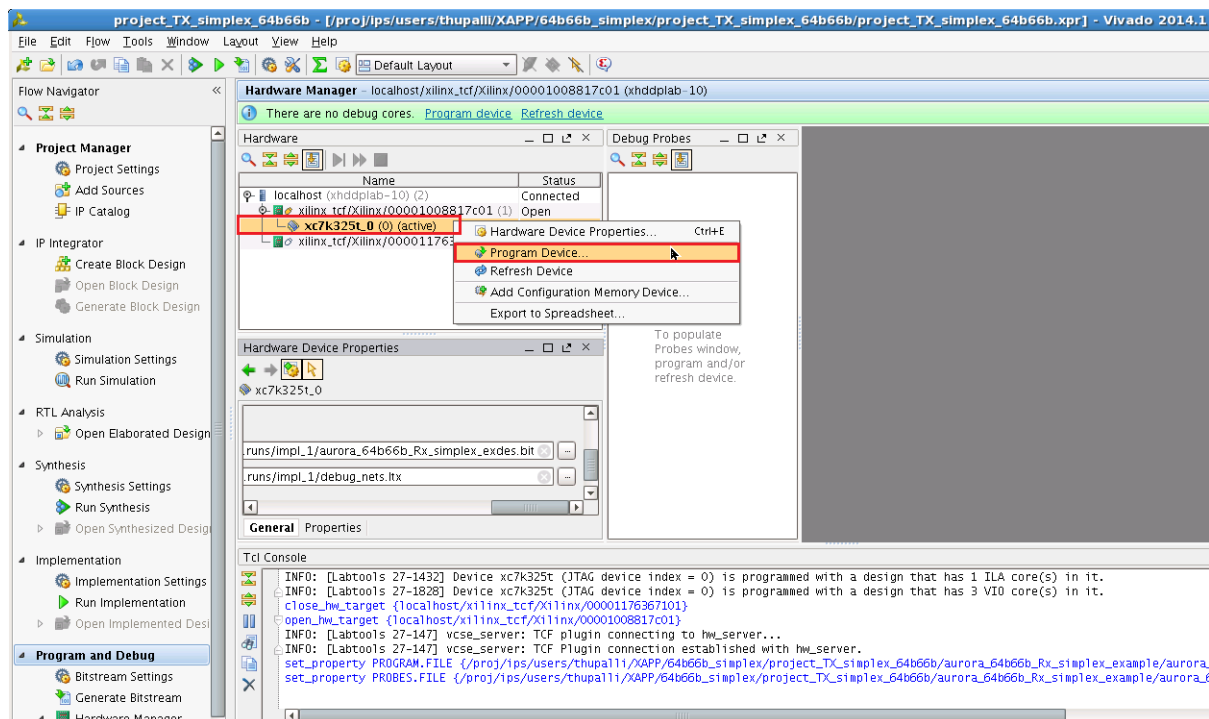


図 20 : [Program Device...] の選択

9. プログラムが完了したら [Hardware] ビューでプログラムされたターゲット デバイスを右クリックし、[Close Target] をクリックします (図 21)。

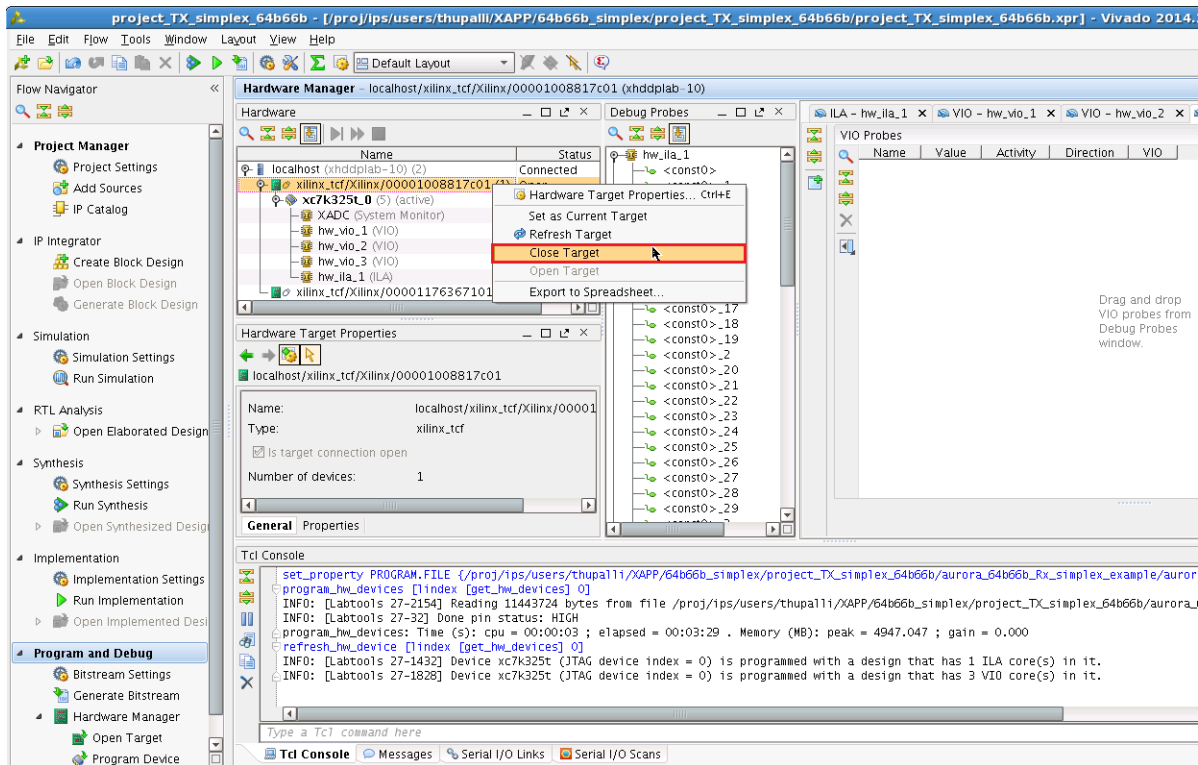


図 21 : [Close Target] の選択

10. [Hardware] ビューで 2 つ目のターゲット プラットフォームを右クリックし、[Close Target] をクリックします (図 22)。

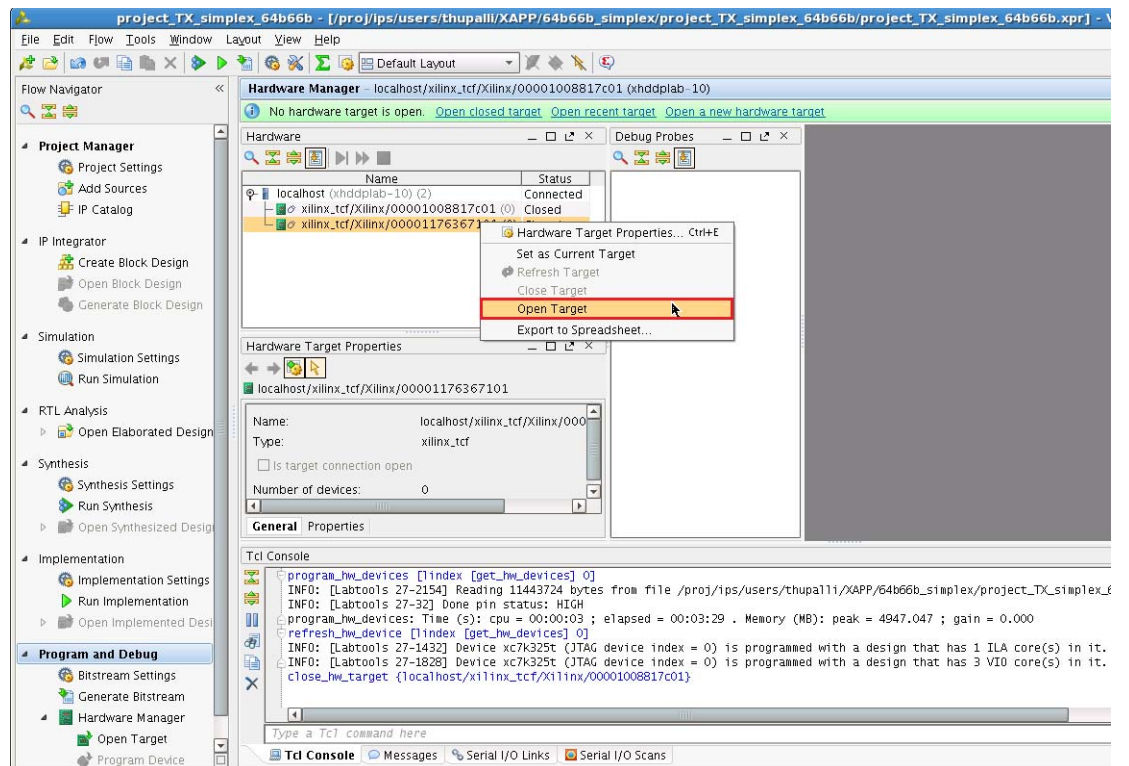


図 22 : 2 つ目のターゲット プラットフォームを開く

11. 送信プラットフォームのビットストリーム ファイル名および対応する .ltx プローブ ファイル名を使用して、手順 6 と手順 7 を繰り返します。
12. 手順 8 を繰り返してデバイスをプログラムします。
13. プログラムが完了したら [Hardware] ビューでプログラムされたターゲット デバイスを右クリックし、[Refresh Device] をクリックします (図 23)。

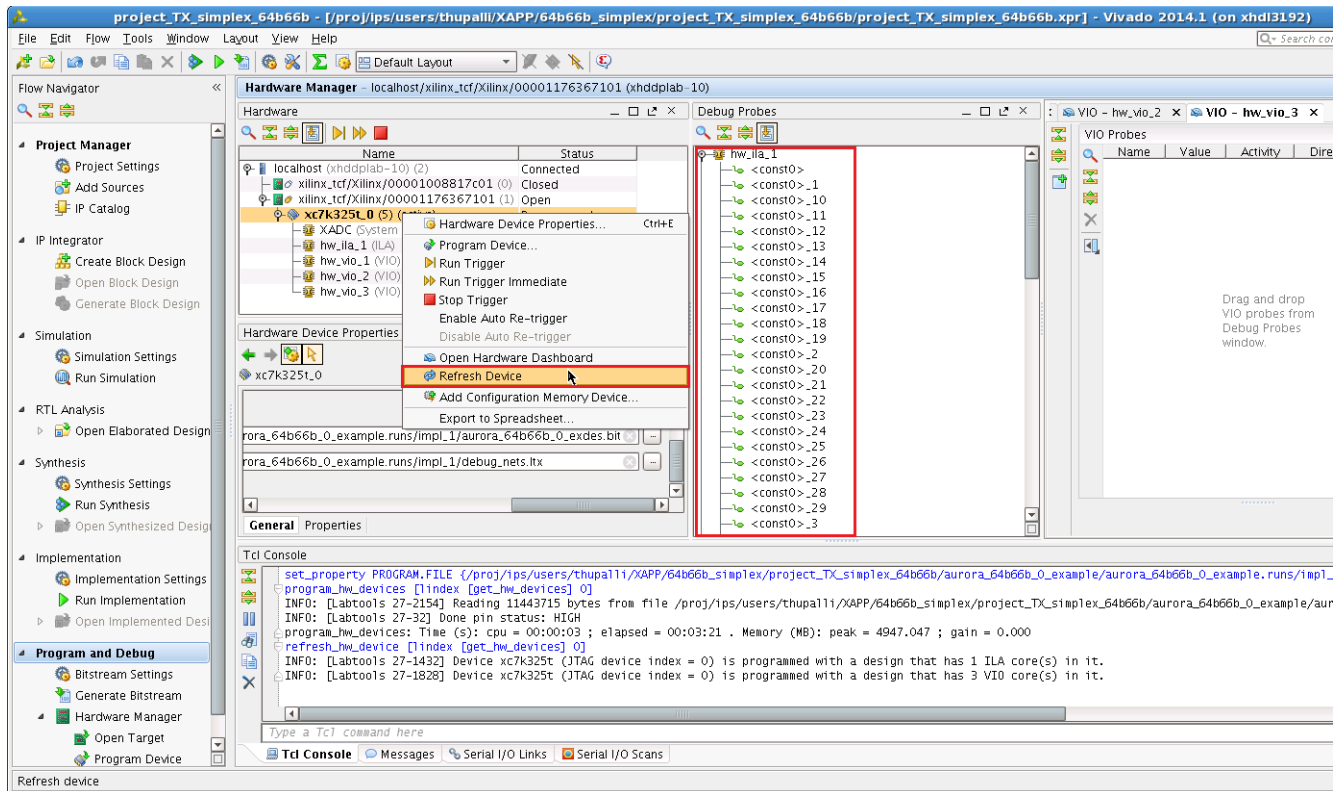


図 23 : [Refresh Device] の選択

## デザインの実行

1. [Hardware] ビューでデバイスを右クリックし、[Run Trigger] をクリックします (図 24)。

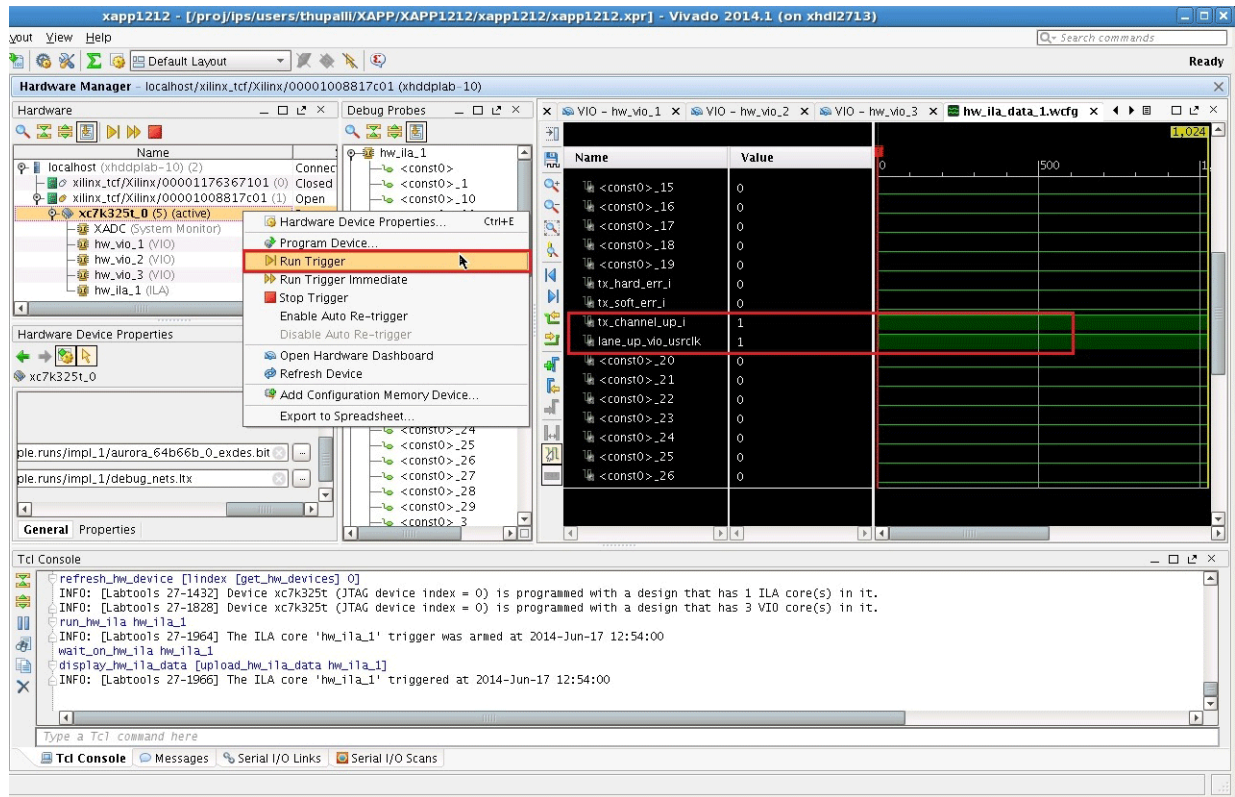


図 24 : [Run Trigger] の選択

2. 表示された波形画面で、lane\_up\_vio\_usrclk 信号と tx\_channel\_up\_i 信号が High であることを確認します。
3. [Debug Probes] で [hw\_vio\_1] の下にある次の信号を Ctrl キーを押しながらクリックしてすべて選択します。
  - channel\_up\_in\_initclk
  - lane\_up\_vio\_i
  - gtrreset\_from\_vio\_i
  - sysreset\_from\_vio\_i
4. 選択した信号上で右クリックして [Add Probes to VIO Window] をクリックします (図 25)。

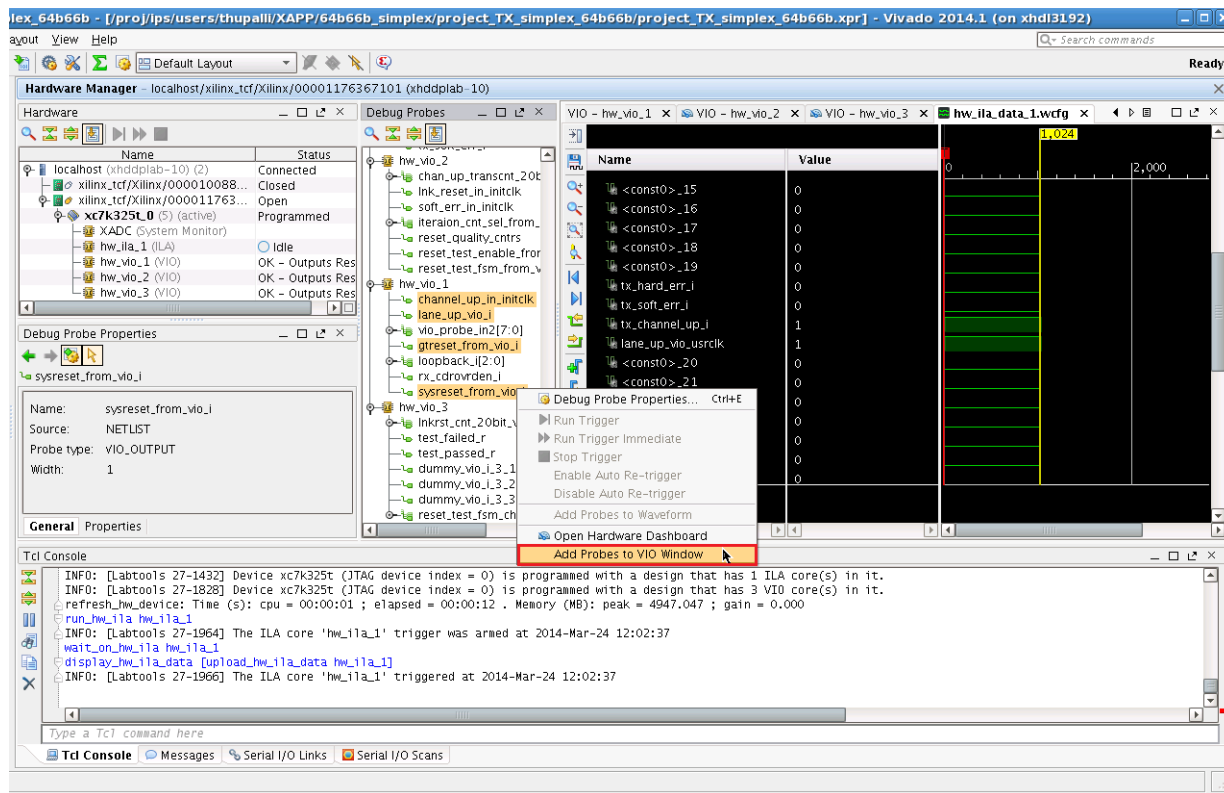


図 25 : [Add Probes to VIO Window] の選択

5. 各信号について [Value] 列の値をクリックしてリセット信号をトグルします (図 26 参照)。「1」または「0」を入力して [OK] をクリックします。

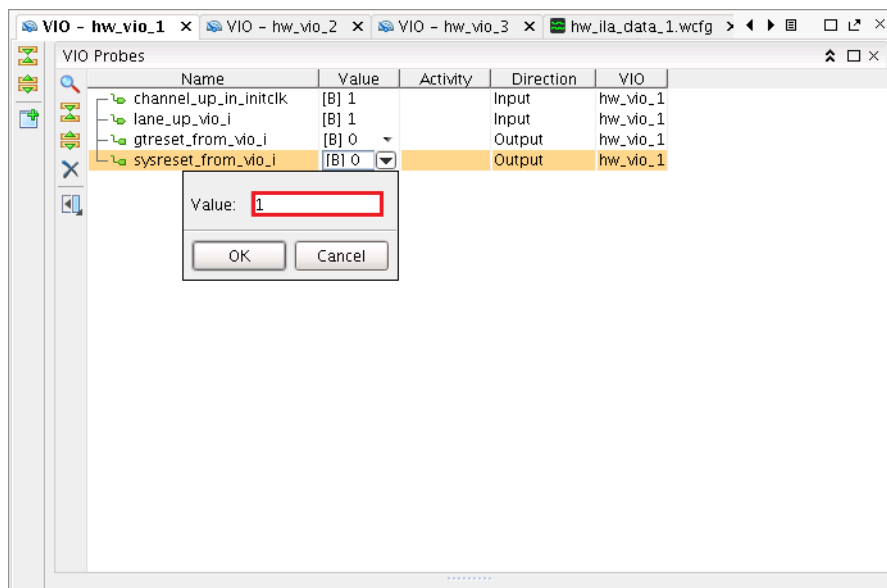


図 26 : リセット信号のトグル

6. channel\_up\_in\_initclk 信号と lane\_up\_vio\_i 信号は Low に遷移し、各リセット信号がトグルした後に High へ戻るはずですが。

次の手順に従って、波形画面でリセット信号の結果を確認します。

1. 1つのリセット信号を High に設定します。
2. [Hardware] ビューでデバイスを右クリックし、[Run Trigger] をクリックします。
3. 波形表示のタブをクリックして、リセット信号の結果を確認します (図 27)。

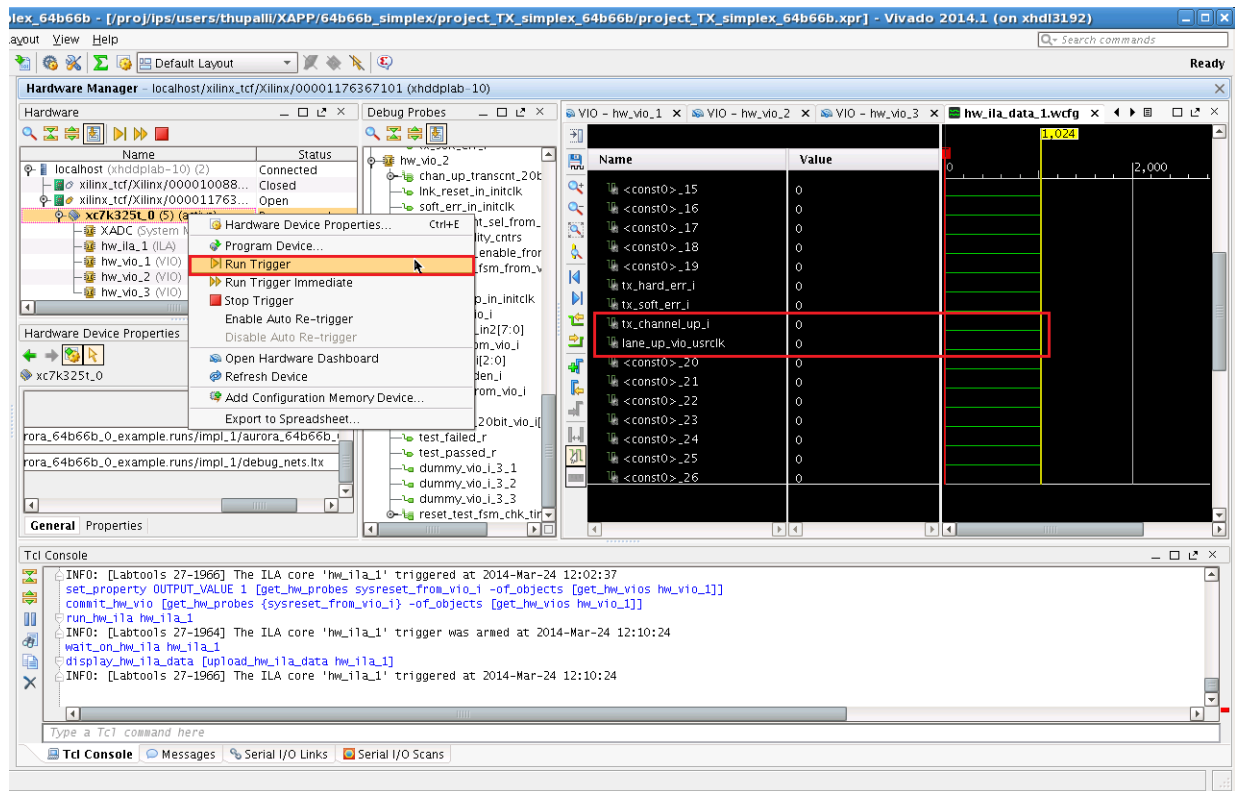


図 27：リセット信号の結果の波形表示

4. リセット信号をトグルするたびに、手順 2 と手順 3 を繰り返して結果を確認します。

前述の手順で、sysreset\_from\_vio\_i または gtrreset\_from\_vio\_i のいずれかがアサートされると、コア (またはトランシーバー) がリセット状態になるため、channel\_up\_in\_initclk と lane\_up\_vio\_i の両方が Low に遷移することを検証しました。ただし、sysreset\_from\_vio\_i と gtrreset\_from\_vio\_i の両方が Low の場合、コアはリセット状態から遷移し、channel\_up\_in\_initclk と lane\_up\_vio\_i は両方とも High になります。

## リファレンス デザイン

表 2 に、リファレンス デザインの詳細を示します。

表 2: リファレンス デザインの詳細

パラメーター	説明
<b>全般</b>	
ターゲット デバイス (ステッピング レベル、ES、プロダクション、スピード グレード)	Kintex-7 XC7K325T-2FFG900
ソース コードの提供	あり
ソース コードの形式	VHDL/Verilog (一部は暗号化済み)
既存のザイリンクス アプリケーション ノート/リファレンス デザイン、Vivado IP カタログ、サードパーティからデザインへのコード/IP の使用	Vivado IP カタログから生成された Aurora コアを使用
<b>シミュレーション</b>	
論理シミュレーションの実施	なし
タイミング シミュレーションの実施	なし
論理シミュレーションおよびタイミング シミュレーションでのテストベンチの利用	N/A
テストベンチの形式	N/A
使用したシミュレータ/バージョン	N/A
SPICE/IBIS シミュレーションの実施	なし
<b>インプリメンテーション</b>	
使用した合成ツール/バージョン	Vivado Design Suite 2014.1
使用したインプリメンテーション ツール/バージョン	Vivado Design Suite 2014.1
スタティック タイミング 解析の実施	あり
<b>ハードウェア検証</b>	
ハードウェア検証の実施	あり
使用したハードウェア プラットフォーム	Kintex-7 FPGA KC705 評価キット

## まとめ

Kintex-7 FPGA KC705 評価キットは、LogiCORE IP Aurora 64B/66B コアを実装およびテストするための最適なプラットフォームを提供します。このアプリケーション ノートで説明した手順に従うと、アプリケーションに応じて Aurora 64B/66B シンプレックス デザインを検証し、拡張することができます。KC705 ボード、クロック ソース、および Vivado Design Suite を使用するだけで、さまざまなコンフィギュレーションを素早く評価できます。

## 参考資料

このアプリケーション ノートの参考資料は次のとおりです。

- 『LogiCORE IP Aurora 64B/66B 製品ガイド』([PG074](#))
- 『Kintex-7 FPGA KC705 評価キット スタートアップ ガイド』([UG883](#))
- 『Vivado Design Suite ユーザー ガイド : IP を使用した設計』([UG896](#))
- 『Vivado Design Suite ユーザー ガイド : プログラムおよびデバッグ』([UG908](#))
- 『エンベデッド システム ツール リファレンス マニュアル』([UG111](#))

## 改訂履歴

次の表に、この文書の改訂履歴を示します。

日付	バージョン	内容
2015年1月9日	1.0	初版

## Notice of Disclaimer

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available “AS IS” and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at <http://www.xilinx.com/warranty.htm>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications: <http://www.xilinx.com/warranty.htm#critapps>.

## Automotive Applications Disclaimer

XILINX PRODUCTS ARE NOT DESIGNED OR INTENDED TO BE FAIL-SAFE, OR FOR USE IN ANY APPLICATION REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS APPLICATIONS RELATED TO: (I) THE DEPLOYMENT OF AIRBAGS, (II) CONTROL OF A VEHICLE, UNLESS THERE IS A FAIL-SAFE OR REDUNDANCY FEATURE (WHICH DOES NOT INCLUDE USE OF SOFTWARE IN THE XILINX DEVICE TO IMPLEMENT THE REDUNDANCY) AND A WARNING SIGNAL UPON FAILURE TO THE OPERATOR, OR (III) USES THAT COULD LEAD TO DEATH OR PERSONAL INJURY. CUSTOMER ASSUMES THE SOLE RISK AND LIABILITY OF ANY USE OF XILINX PRODUCTS IN SUCH APPLICATIONS.

この資料に関するフィードバックおよびリンクなどの問題につきましては、[jpn\\_trans\\_feedback@xilinx.com](mailto:jpn_trans_feedback@xilinx.com) まで、または各ページの右下にある [フィードバック送信] ボタンをクリックすると表示されるフォームからお知らせください。いただきましたご意見を参考に早急に対応させていただきます。なお、このメールアドレスへのお問い合わせは受け付けておりません。あらかじめご了承ください。