

Kintex-7 FPGA ベース ターゲット リファレンス デザイン

スタートアップ ガイド

UG883 (v1.0) 2012 年 1 月 13 日



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.

© Copyright 2012 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. PCI, PCIe, and PCI Express are trademarks of PCI-SIG and used under license. All other trademarks are the property of their respective owners.

Fedora Information

Xilinx obtained the Fedora Linux software from Fedora (<http://fedoraproject.org/>), and you may too. Xilinx made no changes to the software obtained from Fedora. If you desire to use Fedora Linux software in your product, Xilinx encourages you to obtain Fedora Linux software directly from Fedora (<http://fedoraproject.org/>), even though we are providing to you a copy of the corresponding source code as provided to us by Fedora. Portions of the Fedora software may be covered by the GNU General Public license as well as many other applicable open source licenses. Please review the source code in detail for further information. To the maximum extent permitted by applicable law and if not prohibited by any such third-party licenses, (1) XILINX DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE; AND (2) IN NO EVENT SHALL XILINX BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Fedora software and technical information is subject to the U.S. Export Administration Regulations and other U.S. and foreign law, and may not be exported or re-exported to certain countries (currently Cuba, Iran, Iraq, North Korea, Sudan, and Syria) or to persons or entities prohibited from receiving U.S. exports (including those (a) on the Bureau of Industry and Security Denied Parties List or Entity List, (b) on the Office of Foreign Assets Control list of Specially Designated Nationals and Blocked Persons, and (c) involved with missile technology or nuclear, chemical or biological weapons). You may not download Fedora software or technical information if you are located in one of these countries, or otherwise affected by these restrictions. You may not provide Fedora software or technical information to individuals or entities located in one of these countries or otherwise affected by these restrictions. You are also responsible for compliance with foreign law requirements applicable to the import and use of Fedora software and technical information.

本資料は英語版 (v1.0) 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。

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 までお知らせください。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。

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	改訂内容
2012 年 1 月 13 日	1.0	初版リリース

目次

改訂履歴.....	3
Kintex-7 FPGA ベース ターゲット リファレンス デザイン スタートアップ ガイド	
はじめに.....	7
ハードウェア テスト セットアップ要件.....	9
TRD デモのセットアップ.....	10
Kintex-7 FPGA ベース TRD の評価.....	20
Kintex-7 FPGA ベース TRD の変更.....	24
次のステップ.....	30
デザイン サマリ.....	35
付録 A：その他のリソース	
ザイリンクスのリソース.....	37
参考資料.....	37

Kintex-7 FPGA ベース ターゲット リファレンス デザイン スタートアップ ガイド

はじめに

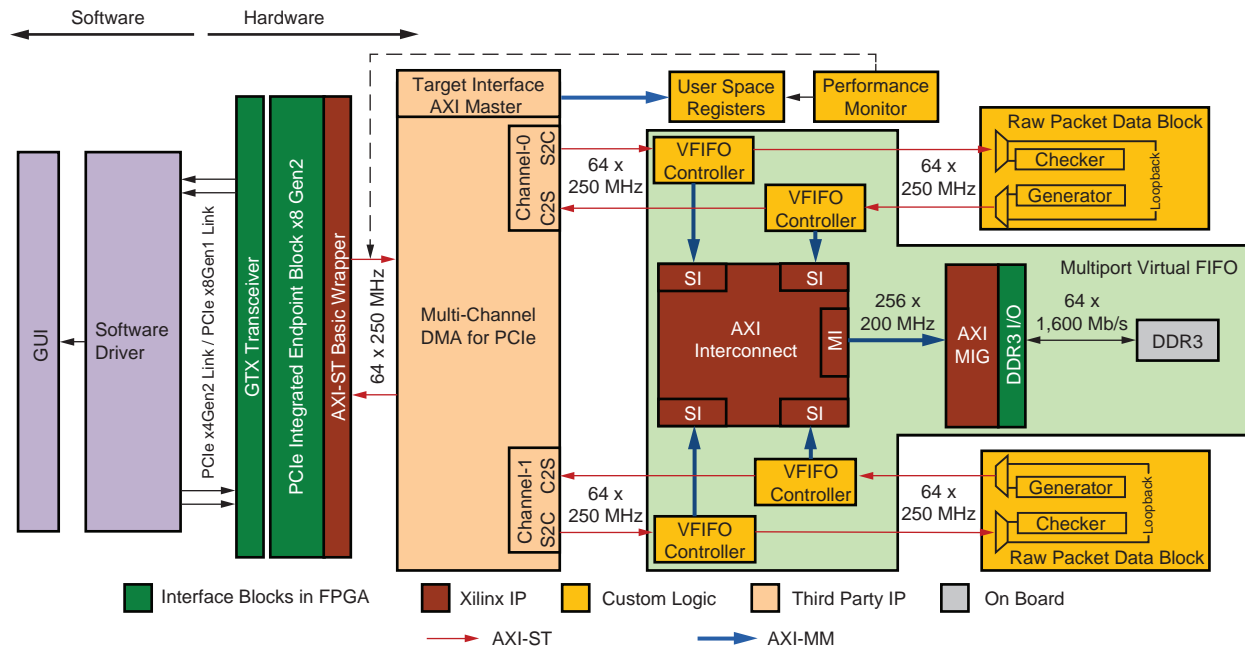
Kintex™-7 FPGA KC705 評価キットは、Kintex-7 FPGA ファミリを使用する、包括的で高性能な開発およびデモ用プラットフォームで、さまざまな市場分野の広帯域幅で高性能なアプリケーションに対応します。このキットを使用することで、DDR3 を使用する設計や、FMC や一般的なシリアル規格 (PCI Express®、XAUI など) による I/O の拡張、および SMA インターフェイスを介した独自のシリアル規格に対応する設計が可能になります。[参照 1] [参照 12]

Kintex-7 FPGA のベース ターゲット リファレンス デザイン (TRD) は、このキットで開発されています。TRD の主要なコンポーネントは、次のとおりです。

- PCI Express® (PCIe®) 用インテグレイテッド エンドポイント ブロック [参照 2] [参照 3] [参照 4]
- Northwest Logic 社製 Packet DMA [参照 14]
- マルチポート仮想 FIFO

TRD システムは、エンド ツー エンドで最大 10Gb/s のスループットに対応しています。

図 1 に TRD の概略図を示します。



UG883_01_010912

図 1 : Kintex-7 FPGA ベース TRD のブロック図

注記：図 1 の矢印は、AXI インターフェイスの方向 (マスターからスレーブへ) を示しています。データ フローの方向ではありません。

コンポーネント

Kintex-7 FPGA ベース TRD には、次のコンポーネントが含まれます。

Kintex-7 FPGA PCI Express 用内蔵ブロック

- PCI Express v2.0 に準拠した 5Gb/s リンク レートの 4 レーン (Gen2)、または 2.5Gb/s リンク レートの 8 レーン (Gen1) のコンフィギュレーション
- AXI4-Stream インターフェイス プロトコルに準拠したユーザー インターフェイス
- パフォーマンス モニターが、内蔵ブロックの AXI4-Stream インターフェイスの PCIe トランザクションをトラッキング

Northwest Logic 社製のバス マスタリング スキャッター / ギャザー Packet DMA、マルチチャネル DMA

- 独立した送受信パスによる持つ全二重処理をサポート
- バック エンドに AXI4-Stream インターフェイス
- 送受信両方向のデータ転送をモニター
- ユーザー定義のレジスタにアクセスする AXI4 メモリマップ方式のターゲットインターフェイス

注記：ベース TRD に同梱されている Northwest Logic 社製 Packet DMA は評価版であり、12 時間しか実行できません。フルライセンス版を入手するには、Northwest Logic 社にお問い合わせください [参照 16]。

マルチポート仮想 FIFO

- パケットのバッファには、DDR3 SDRAM SODIMM (1600Mb/s で 64 ビット、800MHz) を使用。メモリ インターフェイス ジェネレーター (MIG) ツールによって提供されるメモリ コントローラーは、DDR3 メモリに接続。
- AXI インターコネクト IP およびメモリ コントローラーは、メモリの複数のポートをサポート。
- パケット化された仮想 FIFO コントローラーは、各ポートの DDR3 メモリのアドレス割り当てを制御し、DDR3 を仮想パケット FIFO として使用できるようにします。

Linux プラットフォーム向けソフトウェア ドライバー

- ハードウェア デザインのパラメーターを設定
- トラフィックを生成および使用
- ステータスおよびパフォーマンス統計をレポートするグラフィカル ユーザー インターフェイス (GUI) を提供

Kintex-7 FPGA の PCI Express 用内蔵エンドポイント ブロックおよび Packet DMA は、ホスト システムからエンドポイント カード (S2C)、およびエンドポイント カードからホスト システム (C2S) にデータを転送します。ホストが送受信するデータは、DDR3 メモリを中心に構築された仮想 FIFO に保存されます。DDR3 メモリを中心としたマルチポート仮想 FIFO の抽象化層を使用することで、メモリ インターフェイスでのアドレスの割り当てやアービトレーションを管理する必要なく、トラフィックを効率的に移動できます。また、ブロック RAM でインプリメントしたストレージよりもが深さあります。

PCI Express 用内蔵ブロック、Packet DMA、およびマルチポート仮想 FIFO が、ベース システムを構成します。ベース システムは、もう一端で実行されている任意のユーザー アプリケーションにホストをブリッジングできます。Raw Data Packet モジュールは、パケットを生成および使用するダミー アプリケーションです。Aurora または XAUI などのユーザー固有のプロトコルに簡単に置き換えることができます。

ソフトウェア ドライバーは、ホスト システムで動作します。S2C 方向の送信処理のためにロー データ トラフィックを生成します。また、C2S 方向のアプリケーション側でループバックまたは生成されたデータを使用します。

ハードウェア テストのセットアップ要件

ハードウェアでデザインをテストするには、次が必要です。

- Kintex xc7k325t-2-ffg900 FPGA デバイスを搭載した KC705 評価ボード
- 次を含むデザインの ZIP ファイル (USB メモリで提供)
 - デザイン ソース ファイル
 - デバイス ドライバー ファイル
 - ボード デザイン ファイル
 - 資料
- ISE® Design Suite Logic Edition ツール、v13.4 以降
- マイクロ USB ケーブル
- 4 ピン ~ 6 ピン PCIe アダプター ケーブル
- Fedora 16 Live DVD [参照 15]

- PCIe v2.0 スロットを搭載した PC。推奨される PCI Express Gen2 PC システム マザーボードは、ASUS P5E (Intel X38)、ASUS Rampage II Gene (Intel X58)、および Intel DX58SO (Intel X58)。中でも、Intel X58 チップセットは、高性能となる傾向があります。この PC にも、Fedora Core 16 Linux OS をインストール可能です。

TRD デモ用のセットアップ

このセクションでは、ハードウェアですぐにデザインを使用するためのハードウェアのセットアップおよびアプリケーション GUI の使用方法を説明します。ハードウェアの立ち上げの具体的な手順とアプリケーション GUI の使用方法が記載されています。

注記：Kintex-7 FPGA ベース TRD のデモで問題が生じた場合は、ウェブサイトのサポート ページからデザイン アドバイザリや既知の問題を参照してください [参照 11]。

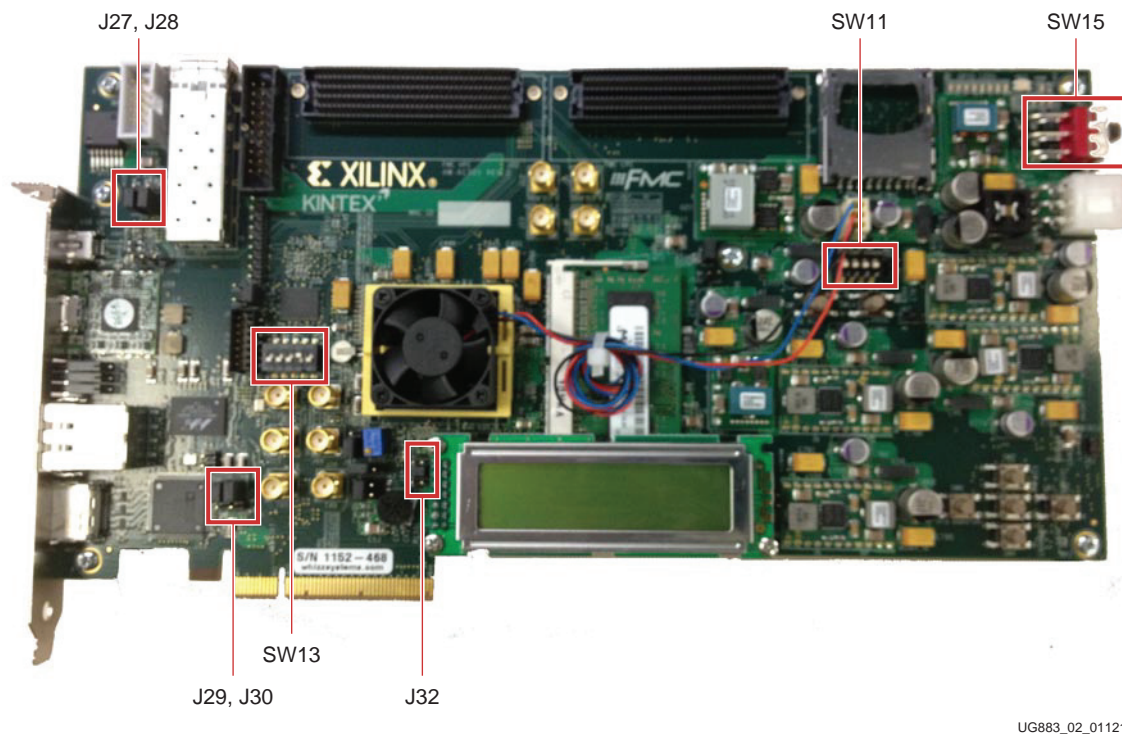
ボードのセットアップ

このセクションでは、TRD のデモに必要な KC705 評価ボードのセットアップ方法を説明します。

1. KC705 のジャンパーとスイッチを設定します。KC705 評価ボードのジャンパーとスイッチが、表 1 および図 2 のように設定されていることを確認します。

表 1：スイッチとジャンパーの設定

ジャンパー	機能		設定
J32	PCIe コンフィギュレーション幅 - 4 レーン デザイン		3-4 をジャンパー接続
スイッチ	機能またはタイプ		設定
SW15	ボード電源スライド スイッチ		オフ
SW11	ユーザー GPIO DIP スイッチ		
	4		オフ
	3		オフ
	2		オフ
	1		オフ
S13	DIP スイッチ SW13 のポジション 1 および 2 は、フラッシュのアドレス ビットの設定を制御		
	DIP スイッチ SW13 のポジション 3、4、5 は、コンフィギュレーション モードを制御		
	5 (M0)	M2 =0 M1=1 M0=0 - マスター BPI	オフ
	4 (M1)	M2 =0 M1=0 M0=1 - マスター SPI	オン
	3 (M2)	M2 =1 M1=0 M0=1 - JTAG	オフ
	2		オフ
	1		オフ



UG883_02_011212

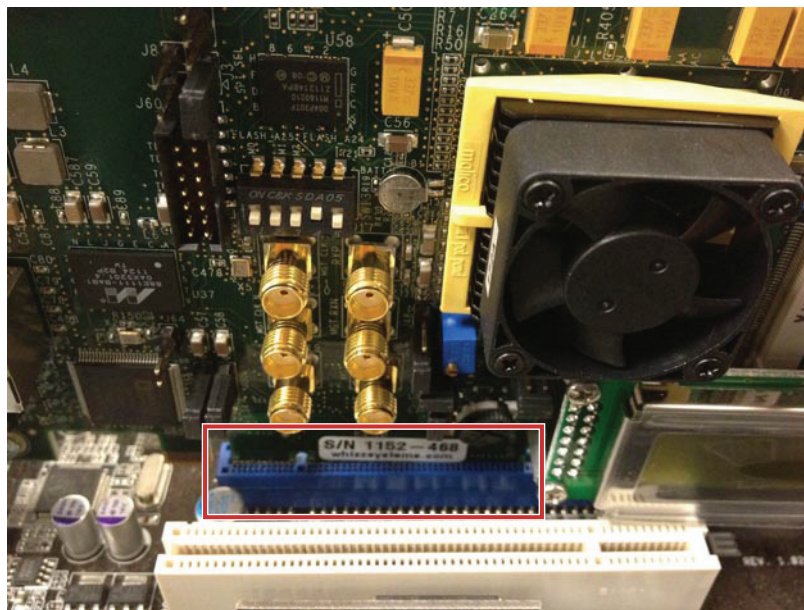
図 2：スイッチとジャンパーの設定

ハードウェアの立ち上げ

このセクションでは、ハードウェア立ち上げの手順を説明します。

1. ホスト システムがオフの状態、PCI Express x8 または x16 エッジ コネクタを介して KC705 ボードを PCIe スロットに挿入します (図 3)。

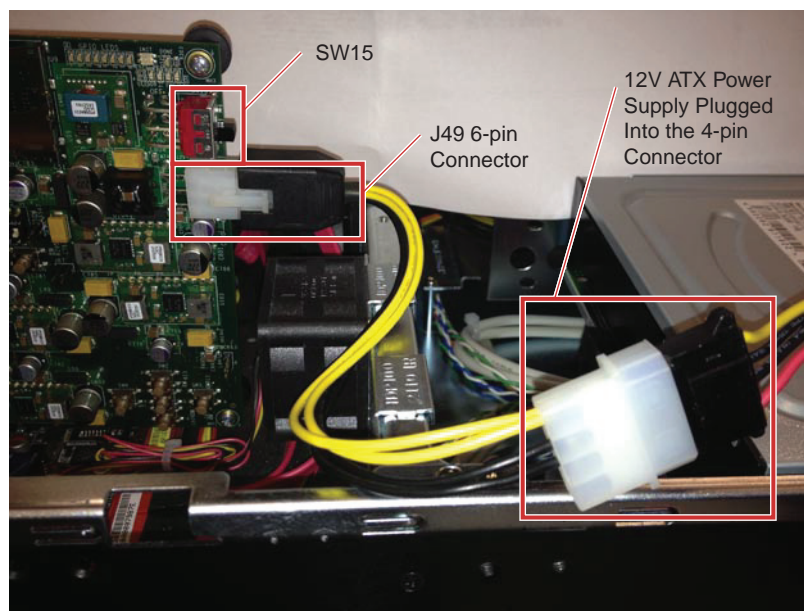
KC705 ボードでプログラムする TRD は、4 レーン PCIe v2.0 が 1 レーン当たり 5Gb/s のリンクレートで動作するよう構成されています。PCI Express の仕様では、レーン幅の狭いエンドポイントを、よりレーン幅の広い PCIe コネクタにインストールできます。



UG883_03_011212

図 3 : PCIe x16 スロットに挿入された KC705 ボード

2. 図 4 は、12V の電源接続を示しています。12V ATX 電源の未使用 4 ピン コネクタを、4 ピン ~ 6 ピンの PCIe アダプター ケーブルを介してボード (J49) に接続します。電源スイッチ SW15 を ON の位置にします。



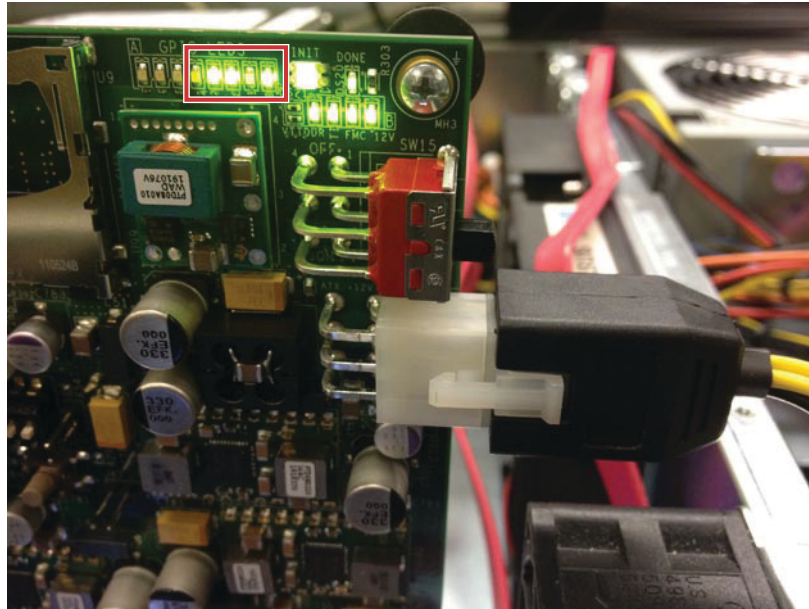
UG883_04_011212

図 4 : 電源の接続

3. しっかりと接続されていることを確認してから、PC システムの電源を入れます。

注記 : Linux を Fedora 16 Live DVD から起動する場合、PC システムの起動後すぐに DVD を PC の CD-ROM ドライブに挿入します。

4. KC705 の LED でデザインのステータスを確認します。デザインのステータスは、KC705 ボードの右上にある GPIO LED でわかります (図 5)。PC システムが起動し、TRD が正常にコンフィギュレーションされると、各ステータス LED は、右から順に次の情報を示します。
 - LED 0 - PCIe リンクが確立されると点灯
 - LED 1 - PCIe ユーザー クロックが存在すると点滅
 - LED 2 - レーン幅が期待されたものであると点灯、そうでない場合は点滅
 - LED 3 - メモリ キャリブレーションが完了すると点灯
 - LED 4 ~ LED 7 - 未接続



UG883_05_011212

図 5 : GPIO ステータス LED の位置 (TRD ステータスを表示)

Linux ドライバーのインストール

1. Fedora 16 が PC システムのハード ディスクにインストールされている場合、ルート権限を持つユーザーとして起動し、[14 ページの手順 3](#)に進みます。
2. キットに同梱されている Fedora 16 Live DVD から起動する場合は、DVD を PC の CD-ROM ドライブに挿入します。Fedora 16 Live Media は、Intel 互換 PC 用です。DVD には、TRD デモ環境に適したパッケージがインストール済みの完全な 32 ビット Fedora 16 環境が含まれており、これが起動可能です。詳細は、[2 ページの「Fedora Information」](#)を参照してください。PC が CD-ROM ドライブから起動したら、liveuser アカウントにログインします。このアカウントには、デバイス ドライバー モジュールのインストールおよび削除に必要なカーネル開発ルート権限が与えられています。

注記：CD-ROM ドライブが最初の起動ドライブになるように BIOS の起動順の設定を変更しなければならない場合があります。起動順を変更するために BIOS メニューを開くには、システムの起動中に Del または F2 キーを押します。起動順を設定し、変更を保存します。(Del または F2 キーは、ほとんどの PC システムで BIOS セットアップの開始するために使用されます。一部の PC では、この方法が異なる場合があります。)

CD-ROM ドライブから PC が起動します。[図 6](#) の画像が、起動中にモニターに表示されます。

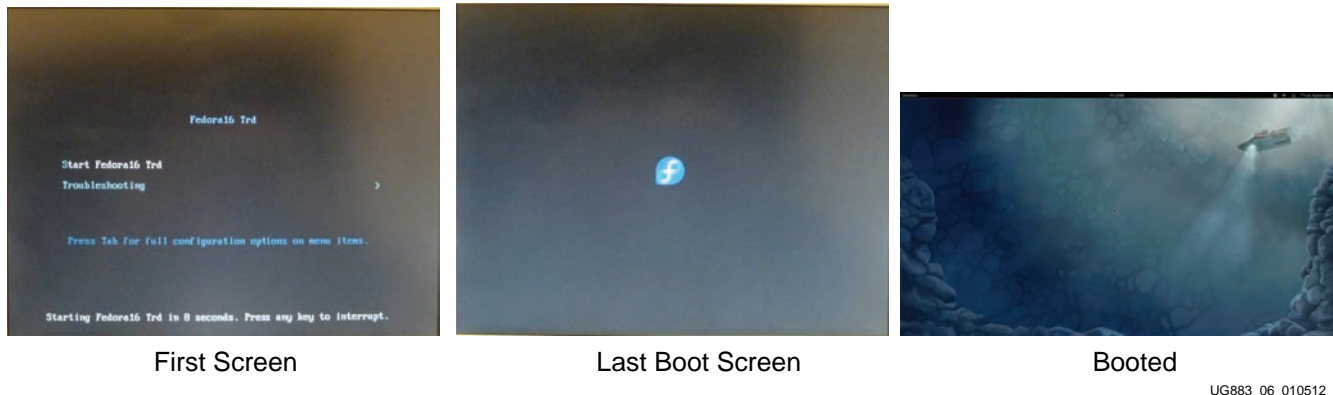


図 6 : Fedora 16 Live DVD の起動画面

3. Fedora 16 Core が起動した後、ターミナル ウィンドウを開きます ([Activities] → [Application] をクリックしてから下にスクロールし、[Terminal] アイコンをクリック)。PCIe エンドポイント ブロックが検出されているかどうかを確認するため、ターミナルのコマンド ラインに次のコマンドを入力します。

```
$ lspci
```

`lspci` コマンドは、PC の PCI バスおよび PCI Express バスのデバイスを表示します。KC705 カード スロットのバスについて次のメッセージが表示されます。

```
Communication controller: Xilinx Corporation Device 7042
```

これは、KC705 にプログラムされているデザインが BIOS と Fedora 16 OS によって検出されたことを表しています。バス番号は、使用する PC マザーボードとスロットによって異なります。図 7 に、サンプル システムでの `lspci` 出力を示します。ザイリックス デバイス 7042 が、BIOS によってバス番号 2 (02:00.0 - bus:dev.function) で検出されています。

```
liveuser@localhost:~
File Edit View Search Terminal Help
00:14.2 PIC: Intel Corporation 5520/5500/X58 I/O Hub Control Status and RAS Registers (rev 12)
00:14.3 PIC: Intel Corporation 5520/5500/X58 I/O Hub Throttle Registers (rev 12)
00:19.0 Ethernet controller: Intel Corporation 82567LM-2 Gigabit Network Connection
00:1a.0 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB UHCI Controller #4
00:1a.1 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB UHCI Controller #5
00:1a.2 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB UHCI Controller #6
00:1a.7 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB2 EHCI Controller #2
00:1b.0 Audio device: Intel Corporation 82801JI (ICH10 Family) HD Audio Controller
00:1c.0 PCI bridge: Intel Corporation 82801JI (ICH10 Family) PCI Express Root Port 1
00:1c.1 PCI bridge: Intel Corporation 82801JI (ICH10 Family) PCI Express Port 2
00:1c.4 PCI bridge: Intel Corporation 82801JI (ICH10 Family) PCI Express Root Port 5
00:1d.0 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB UHCI Controller #1
00:1d.1 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB UHCI Controller #2
00:1d.2 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB UHCI Controller #3
00:1d.7 USB Controller: Intel Corporation 82801JI (ICH10 Family) USB2 EHCI Controller #1
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev 90)
00:1f.0 ISA bridge: Intel Corporation 82801JIR (ICH10R) LPC Interface Controller
00:1f.2 IDE interface: Intel Corporation 82801JI (ICH10 Family) 4 port SATA IDE Controller #1
00:1f.3 SMBus: Intel Corporation 82801JI (ICH10 Family) SMBus Controller
00:1f.5 IDE interface: Intel Corporation 82801JI (ICH10 Family) 2 port SATA IDE Controller #2
02:00.0 Communication controller: Xilinx Corporation Device 7042
03:00.0 VGA compatible controller: nVidia Corporation GT218 [GeForce 210] (rev a2)
03:00.1 Audio device: nVidia Corporation High Definition Audio Controller (rev a1)
06:00.0 IDE interface: Marvell Technology Group Ltd. 88SE6121 SATA II Controller (rev b2)
07:03.0 FireWire (IEEE 1394): Texas Instruments TSB43AB22A IEEE-1394a-2000 Controller (PHY/Link)
) [iOHCI-Lynx]
```

UG883_07_010512

図 7: PCI および PCI Express のバス デバイス

4. USB メモリを PC システムの USB コネクタに挿入します。Fedora 16 が USB デバイスをマウントすると、デスクトップにアイコンが表示されます。

注意: システムの電源切断前または USB メモリを取り外す前には、必ず USB メモリのマウントが解除されていることを確認してください。USB メモリのマウントを解除するには、USB メモリのアイコンを右クリックし、[Safely Remove Drive] をクリックします。このようにしないと、ファイルが破損したり、カーネルがクラッシュしたりする可能性があります。

USB メモリのアイコンをダブルクリックして、k7_pcie_dma_ddr3_base フォルダを任意のディレクトリにコピーします。

5. TRD デモをセットアップおよび実行するには、ソフトウェア ドライバーが PC システムにインストールされている必要があります。ソフトウェア ドライバーのインストールには、次の作業が必要です。

- a. カーネル オブジェクトと GUI の構築
- b. カーネルへのドライバー モジュールの挿入

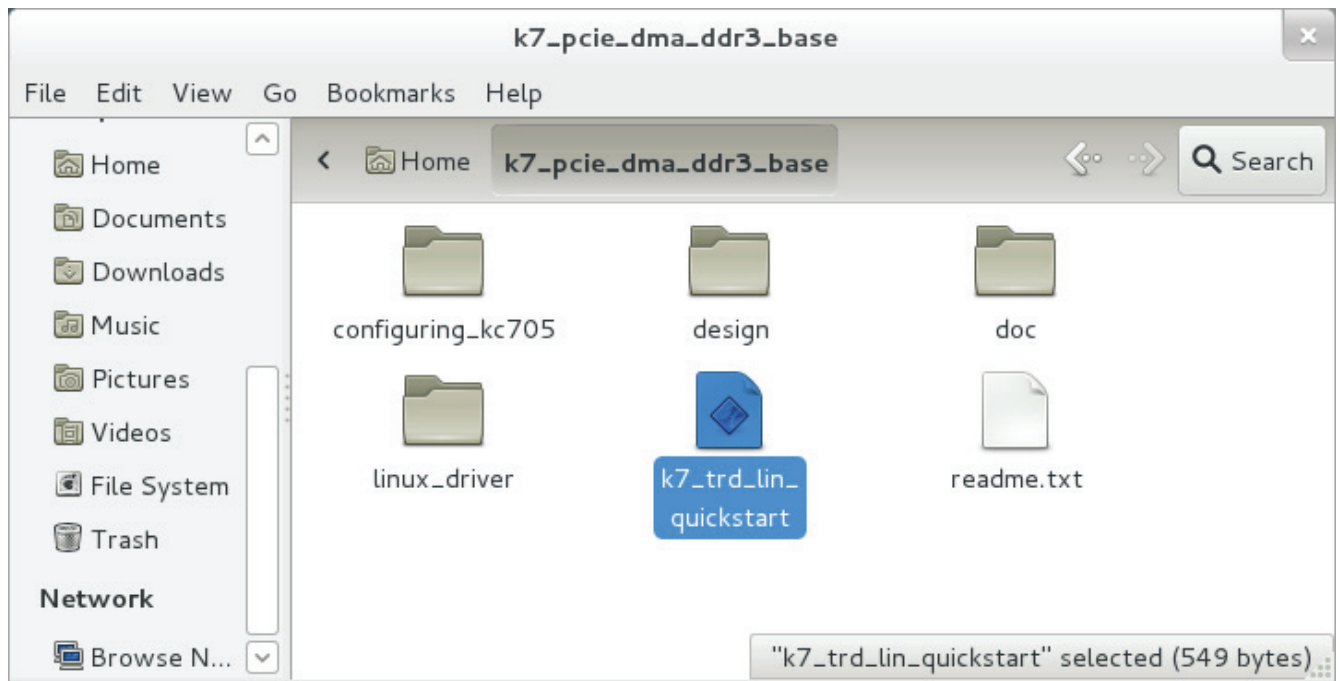
ドライバー モジュールがロードされると、アプリケーション GUI を起動させることができます。ユーザーは、パラメーターを GUI で設定し、TRD を実行できます。

TRD の実行が完了したら、アプリケーション GUI を閉じてドライバーを削除できます。

以上をすべて実行するスクリプトが提供されているため、ユーザーはすぐに TRD を開始できます。

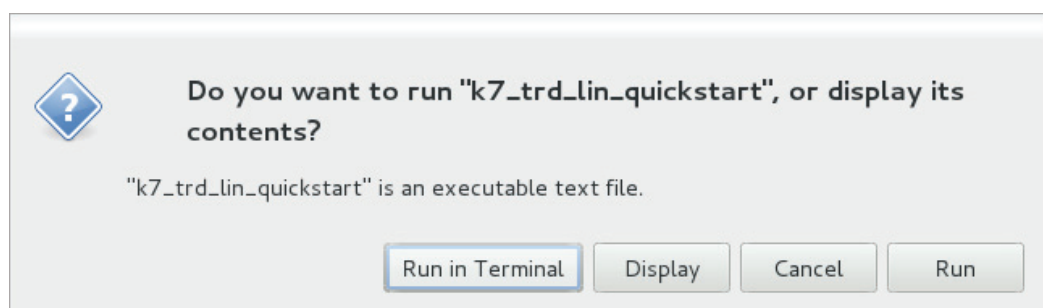
k7_trd_lin_quickstart スクリプトは、k7_pcie_dma_ddr3_base フォルダー内にあります。スクリプトを右クリックして [Properties] をクリックし、[Permissions] タブの [Allow executing file as program] をオンにします。これで、スクリプトが実行可能になります。ウィンドウを閉じます。

スクリプトを実行するには、k7_pcie_dma_ddr3_base フォルダーの k7_trd_lin_quickstart をダブルクリックします (図 8)。図 9 に示すプロンプトが表示されます。



UG883_08_010412

図 8：ドライバーのロードとアプリケーション GUI の起動



UG883_09_010412

図 9：ターミナルで実行

[Run in Terminal] をクリックして次に進みます。アプリケーション GUI が起動されます。

次の「アプリケーション GUI の使用方法」に進み、デザイン パラメーターを設定して TRD を実行します。

アプリケーション GUI の使用方法

ドライバーがロードされ、Performance Monitor の GUI が起動されると、データの送受信を設定できます。GUI では、収集された統計およびその他のステータス情報を確認できます。

1. [System Status] タブをクリックし、KC705 ボードと PCIe リンク (図 10 および表 2) のステータスを検証します。

Xilinx Performance & Status Monitor - Kintex-7 Base TRD

Raw Data Path0: Enable TX->RX Loopback ☒ Packet Size 32768 Start Test

Enable TX Checker ☐

Enable RX Generator ☐

Raw Data Path1: Enable TX->RX Loopback ☒ Packet Size 32768 Start Test

Enable TX Checker ☐

Enable RX Generator ☐

Payload Statistics **System Status** PCIe Statistics

DMA & Software Status

Raw Data Path0:

	Transmit	Receive
Throughput (Gbps)	0.000	0.000
DMA Active Time (ns)	1000000000	1000000000
DMA Wait Time (ns)	1000000000	4
BD Errors	0	0
BD Short Errors	0	n/a
# SW BDs	2999	2999
# SW Buffers	3000	3000

Interrupts Enabled ☐

Raw Data Path1:

	Transmit	Receive
Throughput (Gbps)	0.000	0.000
DMA Active Time (ns)	1000000000	1000000000
DMA Wait Time (ns)	1000000000	4
BD Errors	0	0
BD Short Errors	0	n/a
# SW BDs	2999	2999
# SW Buffers	3000	3000

Interrupts Enabled ☐

PCIe Transmit (writes) (Gbps) 0.000

PCIe Receive (reads) (Gbps) 0.000

PCIe Endpoint Status

Link Status	Up	Vendor ID	0x10ee
Link Speed	5 Gbps	Device ID	0x7042
Link Width	x4	MPS (bytes)	128
Interrupts	Legacy	MRRS (bytes)	512

Host System's Initial Flow Control Credits

Posted Header	96	Posted Data	432
Non-Posted Header	96	Non-Posted Data	16
Completion Header	0	Completion Data	0



[INFO] Kintex-7 Base TRD v1.0

UG883_10_010512

図 10 : Performance Monitor でのボード ステータスの検証

表 2 : KC705 ボード ステータスのフィールドの説明

フィールド	ステータス	説明
Link Status	Up	PCIe リンクが立ち上がり、Kintex-7 FPGA Endpoint for PCI Express と PC マザーボード チップセット間の PCIe 接続が確立されていることを表します。
Link Speed	5.0Gb/s	PCIe リンクが PCI Express v2.0 に準拠したライン レート速度で動作していることを表します。
Link Width	x4	PCIe リンクが x4 リンクとして接続されていることを表します。

2. 次のように、2 つのデータ パス上でデータ トラフィックを開始します。
 - a.  11 に示す [Raw Data Path0] の [Start Test] をクリックします。これで、ドライバーが Raw Data Path0 用のデータの生成を開始できます。
 - b.  11 に示す [Raw Data Path1] の [Start Test] をクリックします。これで、ドライバーが Raw Data Path1 用のデータの生成を開始できます。

Xilinx Performance & Status Monitor – Kintex-7 Base TRD

Raw Data Path0: Enable TX->RX Loopback ☒ Packet Size 32768 **Start Test**

Enable TX Checker ☐

Enable RX Generator ☐

Raw Data Path1: Enable TX->RX Loopback ☒ Packet Size 32768 **Start Test**

Enable TX Checker ☐

Enable RX Generator ☐

Payload Statistics **System Status** PCIe Statistics

DMA & Software Status

Raw Data Path0:

	Transmit	Receive
Throughput (Gbps)	0.000	0.000
DMA Active Time (ns)	1000000000	1000000000
DMA Wait Time (ns)	1000000000	4
BD Errors	0	0
BD Short Errors	0	n/a
# SW BDs	2999	2999
# SW Buffers	3000	3000
Interrupts Enabled	<input type="checkbox"/>	

PCle Transmit (writes) (Gbps) 0.000

PCle Receive (reads) (Gbps) 0.000

PCle Endpoint Status

Link Status	Up	Vendor ID	0x10ee
Link Speed	5 Gbps	Device ID	0x7042
Link Width	x4	MPS (bytes)	128
Interrupts	Legacy	MRRS (bytes)	512

Raw Data Path1:

	Transmit	Receive
Throughput (Gbps)	0.000	0.000
DMA Active Time (ns)	1000000000	1000000000
DMA Wait Time (ns)	1000000000	4
BD Errors	0	0
BD Short Errors	0	n/a
# SW BDs	2999	2999
# SW Buffers	3000	3000
Interrupts Enabled	<input type="checkbox"/>	

Host System's Initial Flow Control Credits

Posted Header	96	Posted Data	432
Non-Posted Header	96	Non-Posted Data	16
Completion Header	0	Completion Data	0

[INFO] Kintex-7 Base TRD v1.0

UG883_11_010512

図 11 : Performance Monitor からのデータ トラフィックの開始

3. GUI で表示されるステータス情報によって TRD 動作を検証します (図 12)。
 - a. PCIe スループットを検証します。
 - b. Raw Data Path0 用の DMA チャンネル スループットを検証します。
 - c. Raw Data Path1 用の DMA チャンネル スループットを検証します。
 - d. バッファ ディスクリプターにエラーがなく、エラー フリー動作であることを確認します。

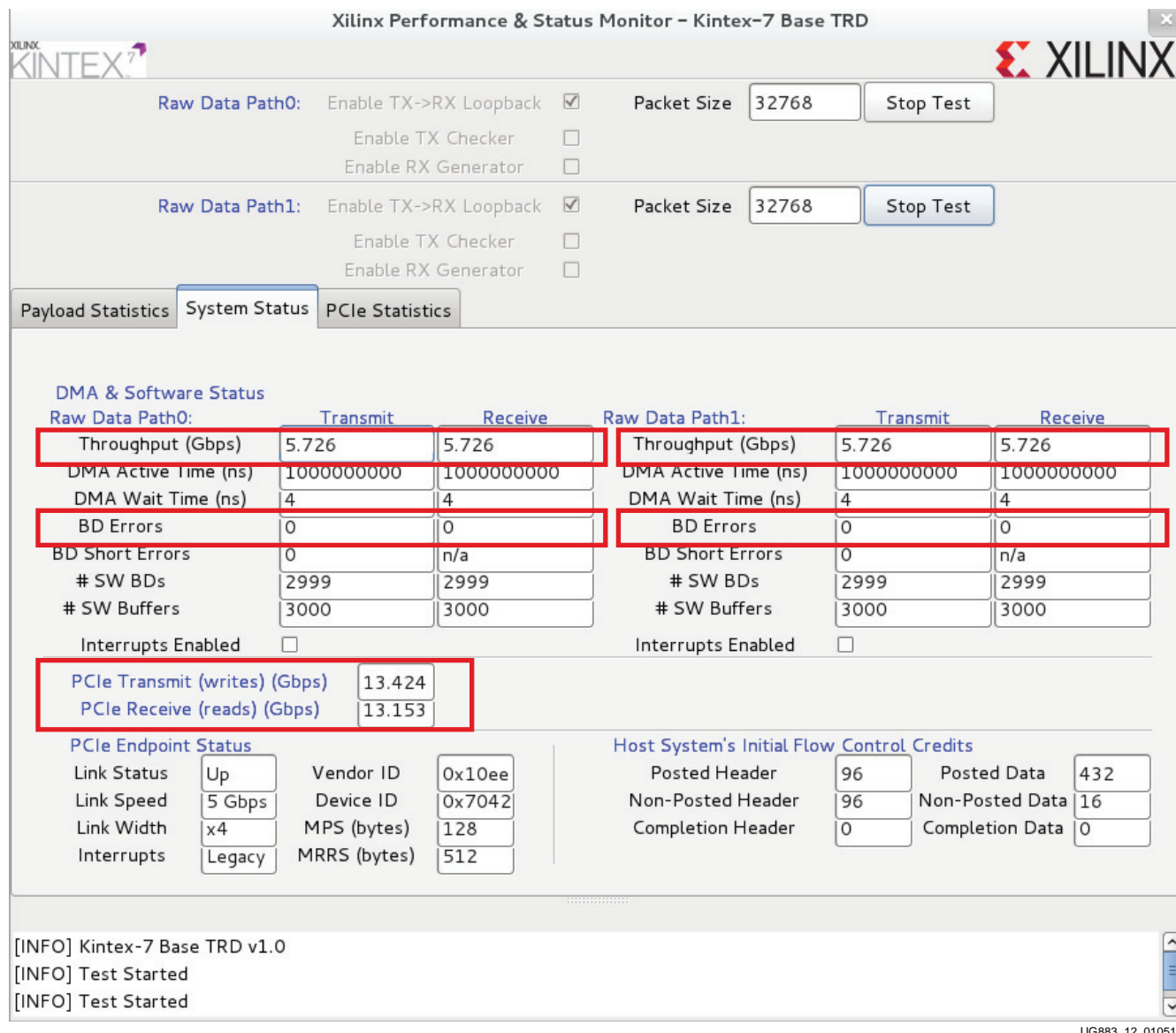


図 12 : Performance Monitor でのエラー フリー動作の検証

これで、Kintex-7 FPGA PCIe-DMA TRD のセットアップが完了し、動作しています。アプリケーション GUI を閉じ、ソフトウェア ドライバーをアンロードしてトラフィック フローを停止します。

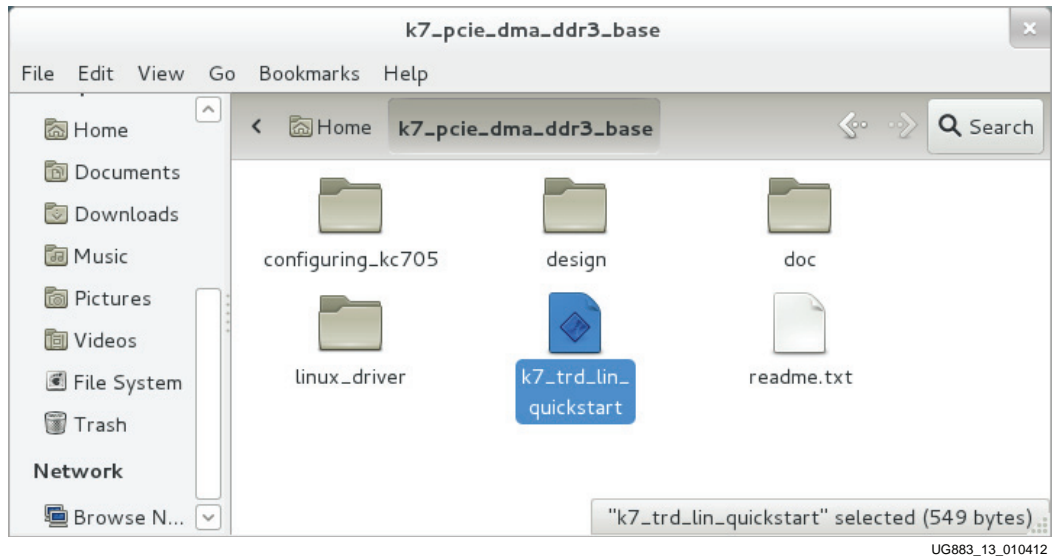
Kintex-7 FPGA ベース TRD の評価

Kintex-7 FPGA のベース TRD は、Performance & Status Monitor アプリケーションと GUI を提供します。このアプリケーションを使用して、さまざまなシステム パラメーターを検証できます。このセクションでは、パラメーター セットに基づく PCI Express および DMA インターフェイスのパフォーマンスの変化を検証します。

Kintex-7 FPGA のベース TRD を検証する手順は次のとおりです。

1. Kintex-7 FPGA ベース TRD の Performance Monitor を起動します。

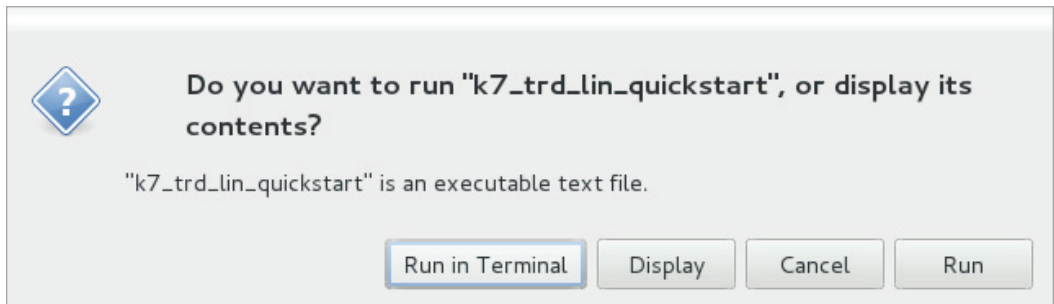
- a. k7_pcie_dma_ddr3_base フォルダに移動します。
- b. k7_lin_trd_quickstart をダブルクリックし (スクリプトの実行権限があることを確認)、Performance & Status Monitor GUI を起動します (図 13)。



UG883_13_010412

図 13 : Performance & Status Monitor GUI の起動

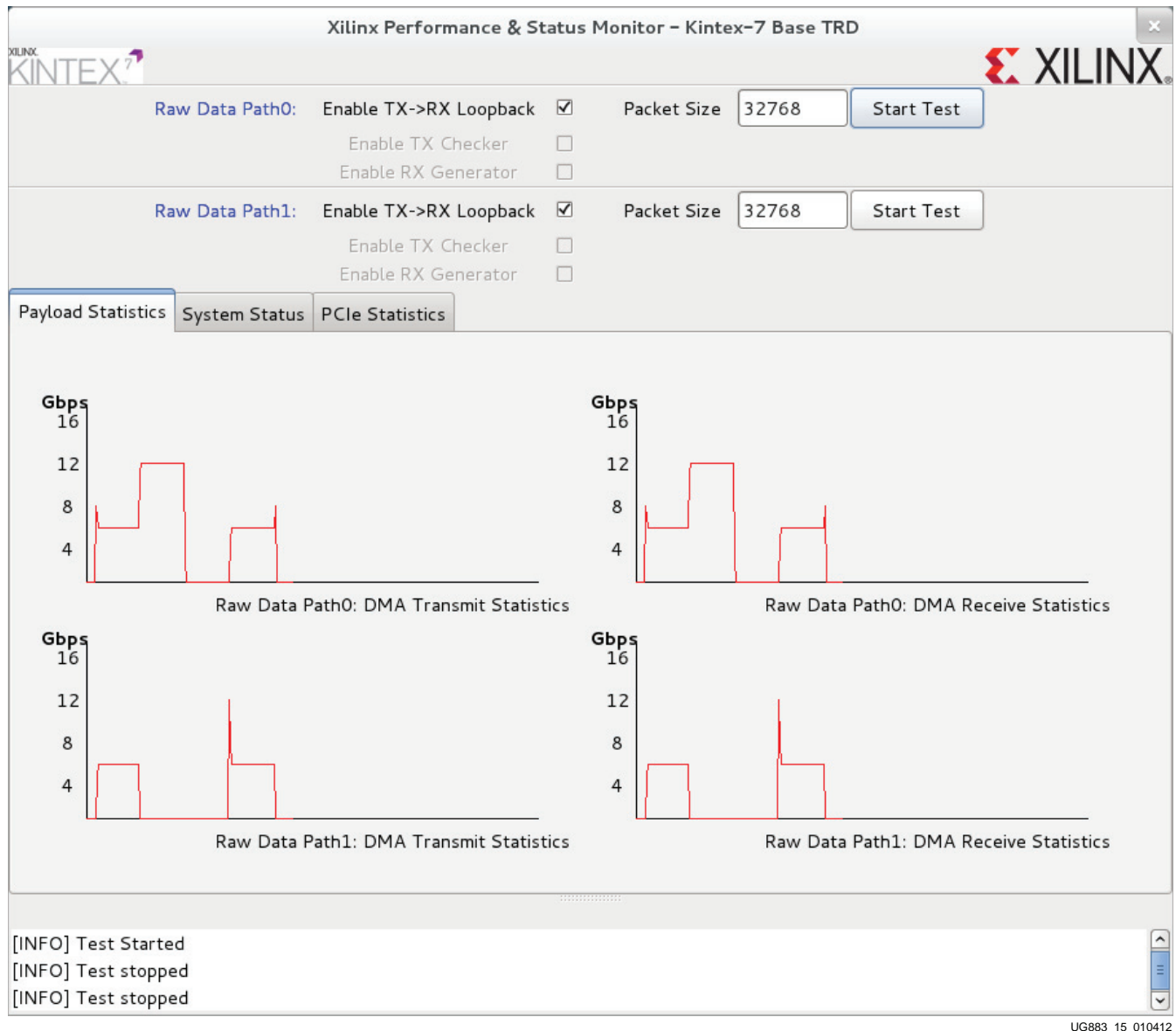
- c. 図 14 に示すプロンプトが表示されます。[Run in Terminal] をクリックして次に進みます。



UG883_14_010412

図 14 : k7_lin_trd_quickstart の実行

2. Performance Monitor でテスト パラメーターを設定します。
 - a. Raw Data Path0 と Raw Data Path1 の 2 つのデータパスが使用できます。各パスの [Packet Size] に、64 ~ 32,768 バイトの間で値を設定します。
3. テストを実行し、Performance & Status Monitor でペイロードの統計情報 ([Payload Statistics]) を確認します (図 15)。
 - a. [Start Test] をクリックして、パフォーマンス テストを開始します。
 - b. [Payload Statistics] タブをクリックし、DMA チャネル上のデータ転送を確認します。
 - c. [Stop Test] をクリックし、データ転送を停止します。



UG883_15_010412

図 15 : ペイロード統計

- d. 両方の Raw Data Path の [Packet Size] パラメーターを変更し (図 16)、[Start Test] をクリックします。ペイロード統計を表示して、DMA チャンネル上のデータ転送速度を確認します。パケット サイズを小さくすると、パフォーマンスが低下します。

注記：パケット サイズを変更する前に [Stop Test] をクリックし、サイズを変更してから [Start Test] をクリックします。

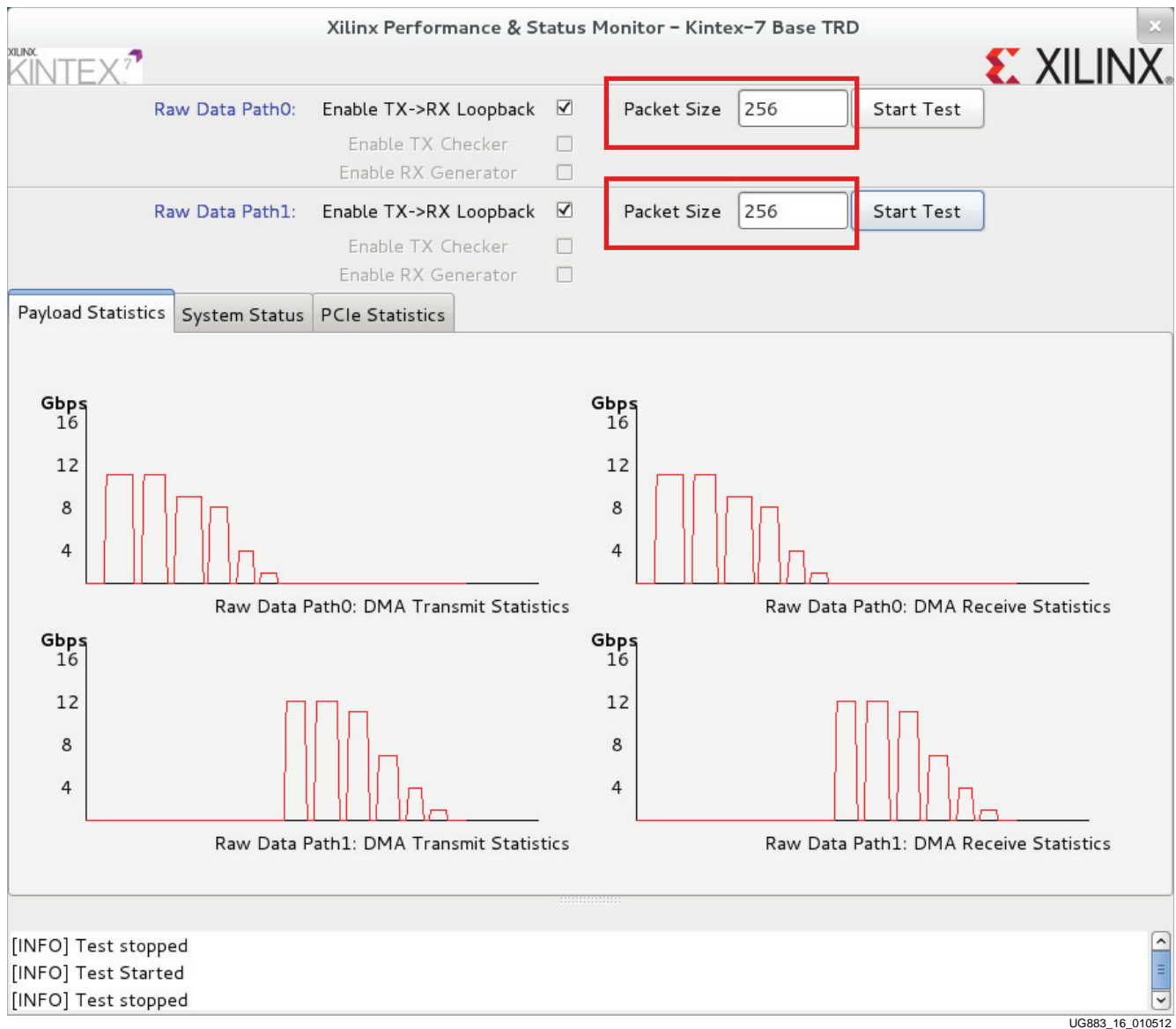


図 16：パケット サイズの変更によるパフォーマンスへの影響

注記：パケット サイズが 64 ～ 128 バイトの場合、スループットが低下して [Payload Statistics] タブに表示されないことがあります。正確な値は、[System Status] タブで確認できます。

4. [PCIe Statistics] タブをクリックし、PCIe インターフェイスでパケット サイズを変更したときのデータ転送数を確認します (図 17)。

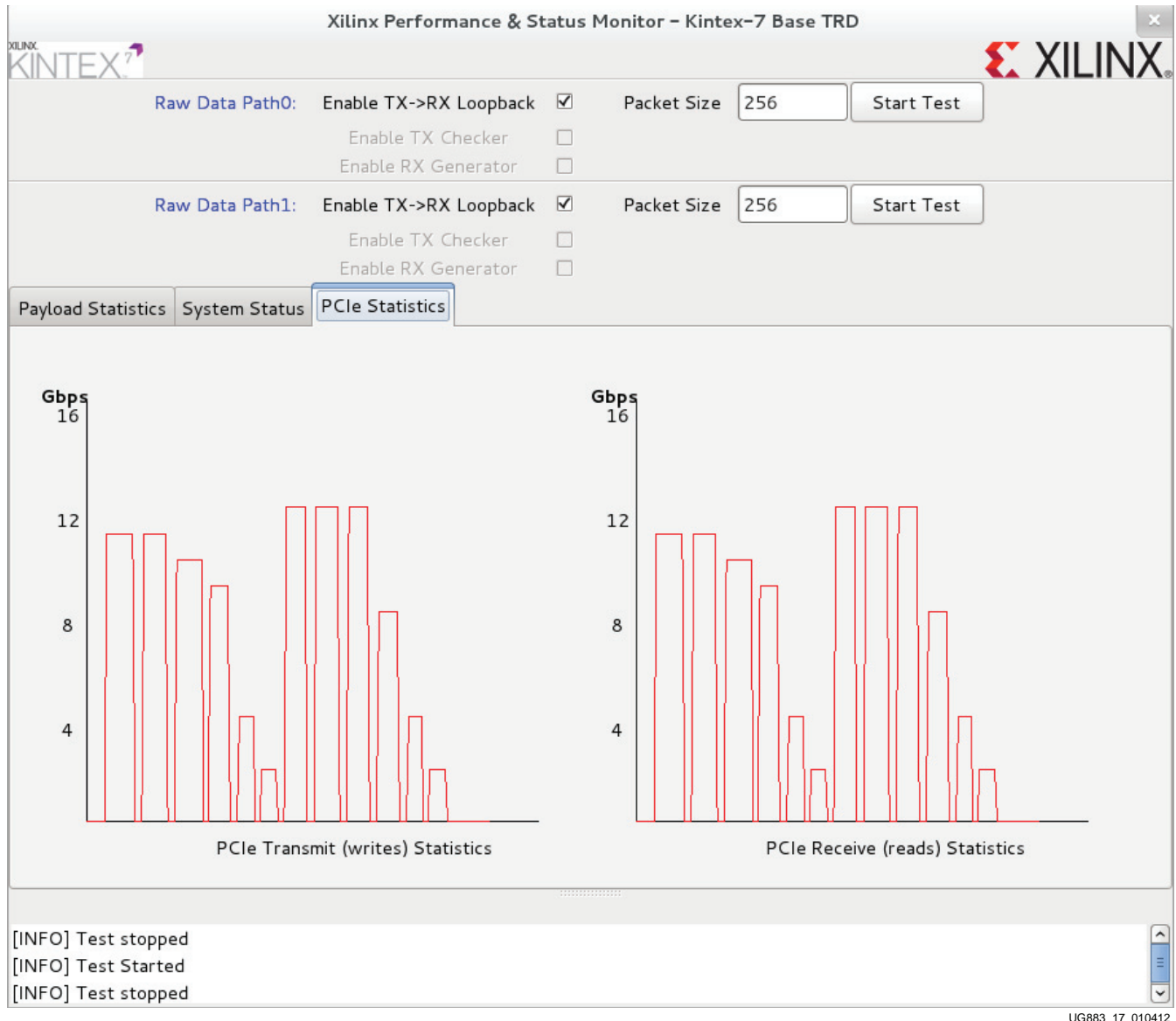


図 17 : Performance Monitor での PCIe 統計

ここまで、構築済みデモ デザインの BIT ファイルを使用して Kintex-7 FPGA のベース TRD のシステム パフォーマンスを検証しました。

Kintex-7 FPGA のベース TRD デモのセットアップと検証が完了したので、次にデザインを変更します。デザインを変更する前に、PC に ISE Design Logic Edition ツールをインストールします。KC705 評価ボードが PCIe エッジ コネクタを介して装着されている PC システムには、ツールのインストールは必要ありません。

Kintex-7 FPGA ベース TRD の変更

このセクションでは、デザインに簡単な変更を加える方法を説明します。

- ハードウェアの変更
- ソフトウェアの変更

重要: コマンド ライン スクリプトを実行する前に、『ISE Design Suite 13: インストールおよびライセンス ガイド』(UG798) の「OS 別インストール手順」を参照し、オペレーティング システムに適した環境変数を設定する方法を理解しておく必要があります [参照 2]。この文書に記載するすべてのスクリプトは、XILINX 環境変数が設定されていることを前提としています。

ハードウェアの変更

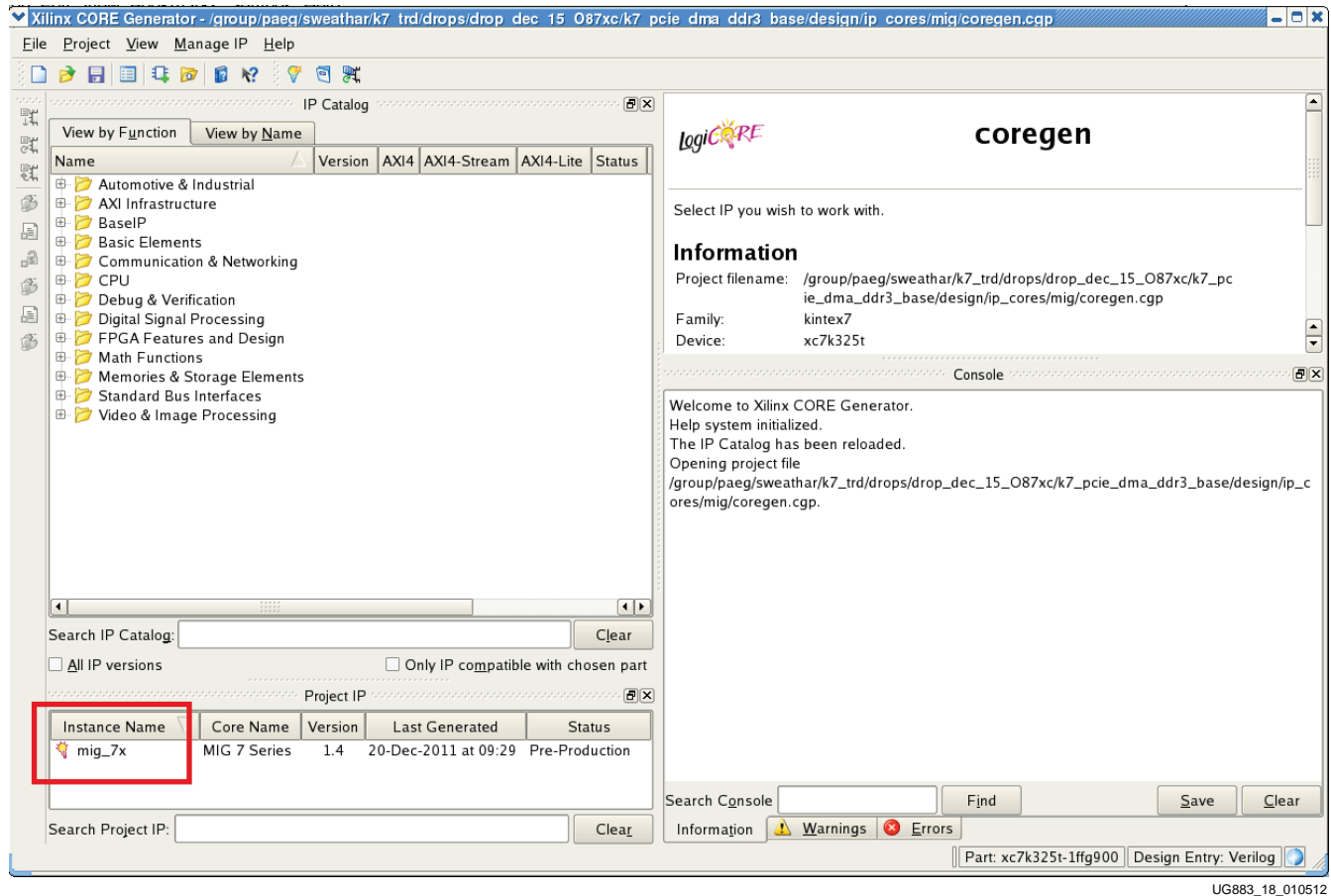
ここでは、ハードウェアの変更方法を説明します。PCI Express のベンダー ID を変更します。

RTL デザインを変更およびインプリメントするには、次の手順に従ってください。

1. ザイリンクスのデザイン ツールがインストールされている PC システムまたはラップトップを使用します。
2. 付属の USB メモリに保存された内容を、このマシンのローカル ディレクトリにコピーします。
3. デザインを変更します。
 - a. `k7_pcie_dma_ddr3_base/design/source/` ディレクトリに移動します。
 - b. `k7_pcie_dma_ddr3_base.v` ファイルを編集します。
 - c. `VENDOR_ID` という文字列を検索します。
 - d. その行の英数字 `10EE` を、PCI-SIG から割り当てられたベンダー ID に変更します (`10EE` はザイリンクスのベンダー ID)。ベンダー ID がまだ割り当てられていない場合は、`19AA` (Northwest Logic 社のベンダー ID) に変更します。
 - e. 変更を保存して終了します。
4. MIG IP コアを生成します。

MIG IP コアは、ユーザーが Micron 社のシミュレーション モデルの使用権許諾契約に同意する必要があるため、ベース TRD ソースの一部として提供されていません。これらのモデルは、ベース TRD のシミュレーションに使用されます。TRD をインプリメントまたはシミュレートする前に、次の手順を完了する必要があります。

- a. ターミナル ウィンドウまたは ISE Design Suite のコマンド ウィンドウを開きます。
- b. `k7_pcie_dma_ddr3_base/design/ip_cores/mig` に移動します (このディレクトリに `mig.xco`、`mig.prj`、および `coregen.cgp` ファイルがある)。
- c. CORE Generator ツールを起動します。次のコマンドをコマンド ラインに入力します。
`$ coregen`
- d. CORE Generator ツールで、[File] → [Open project] をクリックします。coregen.cgp を選択します。
- e. [Instance Name] で `mig_7x` をダブルクリックします (図 18)。mig.xco ファイルおよび mig.prj ファイルで定義された設定の Memory Interface Generator GUI が開きます。



UG883_18_010512

図 18 : MIG IP コアを生成するための CORE Generator GUI

- f. [Micron Tech Inc Simulation Model License Agreement] ページが表示されるまで、[Next] をクリックします。[Accept] をクリックし、[Next] をクリックします。これで、シミュレーションに必要なメモリ モデルが生成されます。
- g. 次のページで [Next] をクリックします。次に [Generate] をクリックして MIG IP コアを生成します。[Release Notes] ウィンドウが表示されます。
- h. [Release Notes] ウィンドウを閉じ、CORE Generator GUI を閉じます。mig フォルダにソース ファイルが生成されています。

注記 : MIG コアの最上位ファイル `k7_pcie_dma_ddr3_base/design/ip_cores/mig/mig_7series_v1_3/user_design/rtl/mig_7x.v` が、KC705 評価ボードをサポートするように変更されています。変更されたファイルは、`k7_pcie_dma_ddr3_base/design/source / modified_ip_files/mig/mig_7x.v` にあります。

5. デザインを構築し、インプリメントします。

- a. ターミナル ウィンドウまたは ISE Design Suite のコマンド ウィンドウを開きます。
- b. `k7_pcie_dma_ddr3_base/design/implement/` ディレクトリに移動します。
- c. オペレーティング システムに応じて、次の手順に従います。

Linux の場合 : 次のコマンドをコマンド ラインで実行します。

```
$ source implement.sh x4 gen2
```

Windows の場合：次のコマンドをコマンドラインで実行します。

```
$ implement.bat -lanemode x4gen2
```

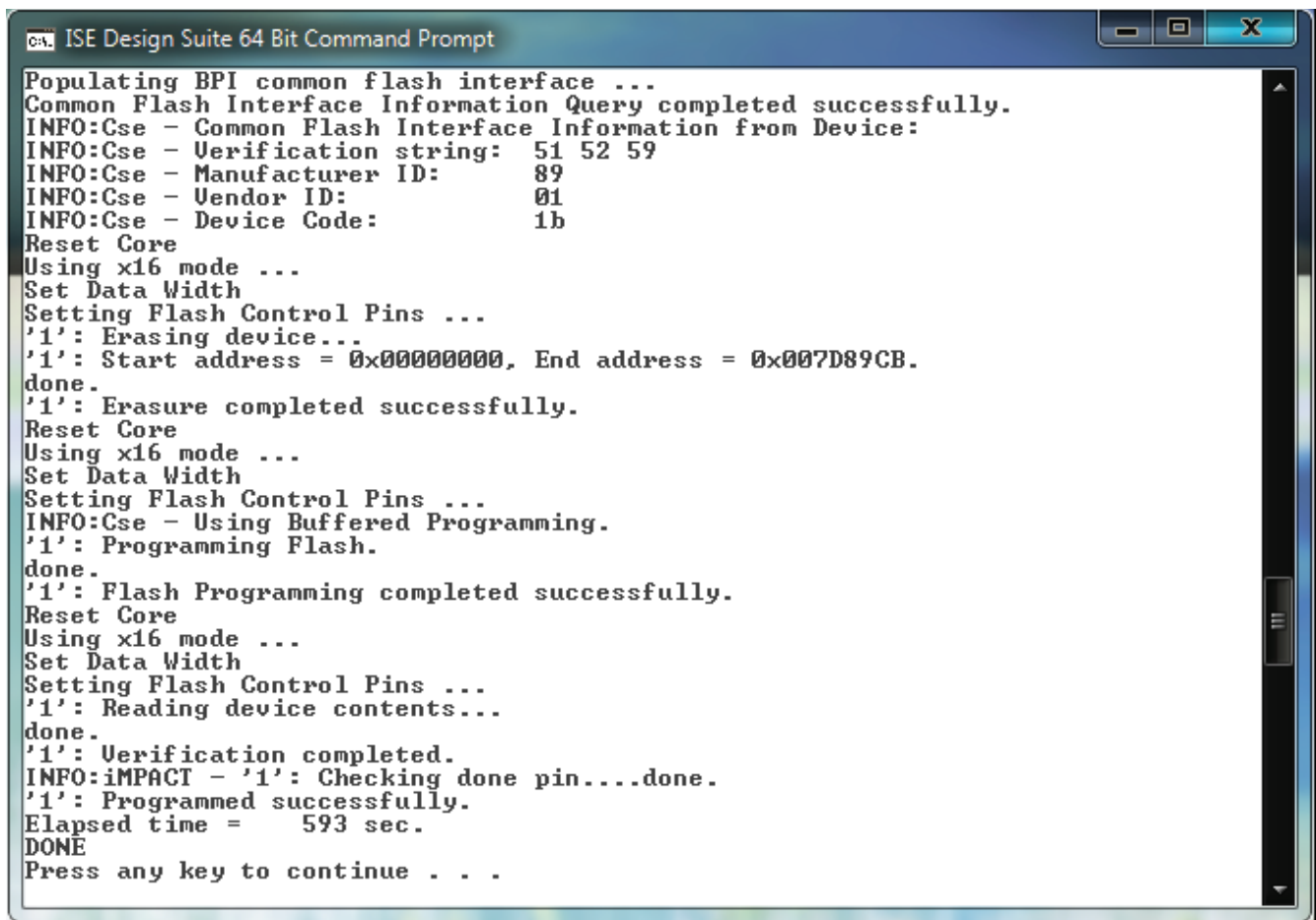
6. デザインのインプリメンテーションが正常に終了すると、次の FPGA プログラミング ファイルを含む results フォルダが生成されます。
 - FPGA プログラミングの BIT ファイル：<ファイル名>.bit
(この場合、k7_pcie_dma_ddr3_base_x4_gen2.bit)
 - BPI フラッシュのプログラミング MCS ファイル：<ファイル名>.mcs
(この場合、KC705.mcs)
 7. ボードを設定します。
 - a. KC705 ボードが PC システムに装着されたままの場合は、PC システムをシャットダウンして KC705 ボードを取り外します。システムをシャットダウンする方法は、次のとおりです。
 - Alt キーを押しながら [Live System User] → [Power off] オプションをクリックします (Alt キーを押さないと、[Suspend] オプションのみが表示される)。システムが徐々にプロセスをシャットダウンします。
- 注記：コピーされたファイルや作成されたアイコンは、次に Fedora 16 Live DVD を起動するときは表示されません。
- b. プラットフォーム フラッシュを使用して FPGA をプログラムする方法については、[10 ページの手順 1](#) で説明したジャンパーの設定を参照してください。
 - c. KC705 のほかのスイッチおよびジャンパーの設定は、『Kintex-7 FPGA KC705 評価ボード ユーザー ガイド』(UG810) [\[参照 1\]](#) に記載されている工場出荷時のデフォルト設定のままにしてください。
 - d. Micro USB ケーブルを、USB コネクタに接続します ([図 19](#))。USB ケーブルのもう一端は、ザイリンクスのデザイン ツールがインストールされている PC システムまたはラップトップに接続します。
 - e. KC705 ボードの電源をオフにします。付属の AC 電源アダプターを使用して 6 ピン コネクタに 12V 電源を供給します。
 - f. KC705 ボードの電源をオンにします。



UG883_19_011212

図 19：ボードの設定

8. オンボードのプラットフォーム フラッシュをプログラムします。
 - a. ターミナル ウィンドウ (Linux の場合) または ISE Design Suite のコマンド ウィンドウを開きます。
 - b. `k7_pcie_10Gdma_ddr3_base/design/implement/results_x4_gen2_325t` ディレクトリに移動します。
 - c. コマンド プロンプトで FPGA プログラミング スクリプトを実行します。完了するまで約 550 ~ 800 秒かかります。次のコマンドを使用します。
 - `$ impact -batch kc705program.cmd` (Linux ベースのマシンの場合)
 - `$ kc705program.bat` (Windows ベースのマシンの場合)
 - d. 正常に終了すると、「Programmed successfully」というメッセージが表示されます (図 20)。



```

C:\> ISE Design Suite 64 Bit Command Prompt

Populating BPI common flash interface ...
Common Flash Interface Information Query completed successfully.
INFO:Cse - Common Flash Interface Information from Device:
INFO:Cse - Verification string: 51 52 59
INFO:Cse - Manufacturer ID: 89
INFO:Cse - Vendor ID: 01
INFO:Cse - Device Code: 1b
Reset Core
Using x16 mode ...
Set Data Width
Setting Flash Control Pins ...
'1': Erasing device...
'1': Start address = 0x00000000, End address = 0x007D89CB.
done.
'1': Erasure completed successfully.
Reset Core
Using x16 mode ...
Set Data Width
Setting Flash Control Pins ...
INFO:Cse - Using Buffered Programming.
'1': Programming Flash.
done.
'1': Flash Programming completed successfully.
Reset Core
Using x16 mode ...
Set Data Width
Setting Flash Control Pins ...
'1': Reading device contents...
done.
'1': Verification completed.
INFO:iMPACT - '1': Checking done pin....done.
'1': Programmed successfully.
Elapsed time = 593 sec.
DONE
Press any key to continue . . .
  
```

UG883_20_122011

図 20 : 正常に完了したフラッシュ プログラミング

- e. 電源スイッチをオフにして、電源コネクタを取り外します。
- f. Micro USB ケーブルを慎重に取り外します。

これで Kintex-7 FPGA のベース TRD が変更され、プラットフォーム フラッシュにプログラムされました。今後は電源投入時に自動的にコンフィギュレーションされます。

9. テストをセットアップします。
 - a. 10 ページの「ボードのセットアップ」および 11 ページの「ハードウェアの立ち上げ」に従って PC システムにボードを挿入し、インプリメントされたデザイン変更を FPGA に適用します。

ソフトウェアの変更

ここでは、ソフトウェアの変更方法を説明します。PCI Express のベンダー ID を変更します。

ソフトウェア デザインを変更するには、次の手順に従ってください。

1. KC705 評価ボードが装着された PC システムを使用します。
2. ドライバー コードを変更します。
 - a. 付属の USB メモリに保存された内容を、このマシンのローカル ディレクトリにコピーします。
 - b. `k7_pcie_dma_ddr3_base/driver/xdma/` ディレクトリに移動します。
 - c. `xdma_base.c` ファイルを編集します。
 - d. `#define PCI_VENDOR_ID_DMA` という文字列を検索します。
 - e. その行の英数字 10EE を、PCI-SIG から割り当てられたベンダー ID に変更します。たとえば、ザイリンクスのベンダー ID は 10EE です。ベンダー ID がまだ割り当てられていない場合は、19AA (Northwest Logic 社のベンダー ID) に変更します。
 - f. 変更を保存して終了します。
3. ドライバーをロードして、Performance Monitor アプリケーションを起動します。
 - a. `k7_pcie_dma_ddr3_base` フォルダーに移動します。
 - b. `k7_lin_trd_quickstart` をダブルクリックします。これで、デバイス ドライバーがロードされ、Performance Monitor アプリケーションが起動されてカーネル オブジェクトが作成されます。
 - c. [Run in Terminal] をクリックして次に進みます。
4. 17 ページの「アプリケーション GUI の使用方法」の手順に従い、変更した設定を確認します。
5. 20 ページの「Kintex-7 FPGA ベース TRD の評価」の手順に従い、変更したデザインのパフォーマンスを検証します。
6. Performance Monitor GUI の [System Status] タブで、ステータス情報を確認します。この画面に表示されるベンダー ID が、先ほどの手順で変更したベンダー ID になっている必要があります。

これで、ベース TRD でプログラムされた Kintex-7 FPGA KC705 評価キットのセットアップをすべて完了し、システムのパフォーマンスを検証しました。また、ベース TRD を変更するザイリンクスのデザイン フローを確認しました。このデザインは、内蔵の PCI Express 用エンドポイント ブロック (PCI Express v2.0 準拠の 4 レーン、5Gb/s コンフィギュレーション)、オンボード DDR3 SODIMM デバイスと接続するように設計されたマルチポート仮想 FIFO メモリ コントローラー、PCI Express 用のサードパーティ製 DMA コントローラーなどから構成されています。

次のステップ

ベース TRD モジュール

このセクションでは、デザイン内のさまざまなブロックにおけるデザイン モジュールとソース ファイル間の相関関係概要を示します。Kintex-7 FPGA ベース TRD の詳細なブロック図は、[図 1](#) を参照してください。表 3 に、各モジュールのデザイン ファイルの構成を示します。

表 3 : Kintex-7 FPGA ベース TRD デザイン ファイルの構成

デザイン コンポーネント	ソース ファイル	LogiCORE™ IP	ベース TRD ソース
最上位モジュール	k7_pcie_dma_ddr3_base.v		✓
PCI Express (x4)	pcie	✓ (Endpoint for PCIe コア - CORE Generator 出力)	
Packet DMA	dma		ネットリスト (Northwest Logic 社 提供)
マルチポート VFIFO コントローラー	mig、virtual_packet_fifo、 axi_ic	✓ (MIG - CORE Generator 出力) (AXI インターコネクト - CORE Generator 出力)	✓
レジスタ インターフェイス	register_map.v		✓
PCIe パフォーマンス モニター	pcie_performance_monitor.v		✓
ソフトウェア ドライバー	linux_driver		✓
ソフトウェア GUI	xpmon		✓

これらモジュールの機能の詳細は、『Kintex-7 FPGA ベース ターゲット リファレンス デザイン ユーザー ガイド』(UG882) の「機能の説明」の章を参照してください [\[参照 5\]](#)。

PCI Express

図 21 に PCI Express のデザイン モジュール、図 22 にデザイン ファイルの構成を示します。

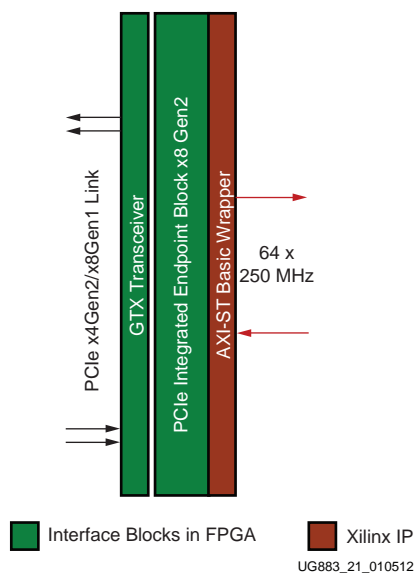


図 21 : PCI Express のデザイン モジュール

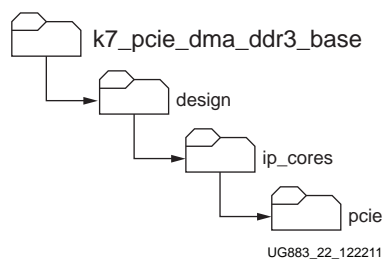


図 22 : PCI Express のデザイン ファイル

Packet DMA

図 23 に Packet DMA のデザイン モジュール、図 24 にデザイン ファイルの構成を示します。

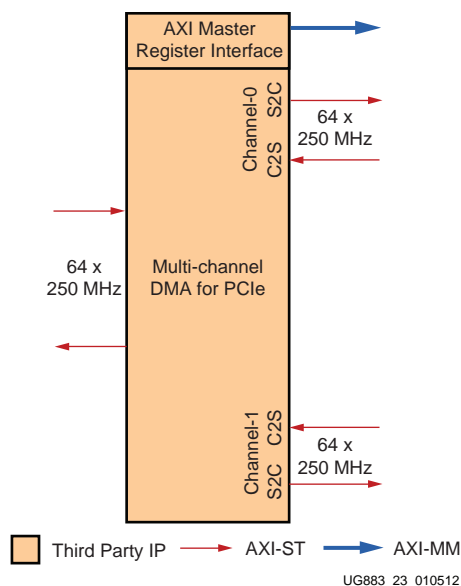


図 23 : Packet DMA のデザイン モジュール

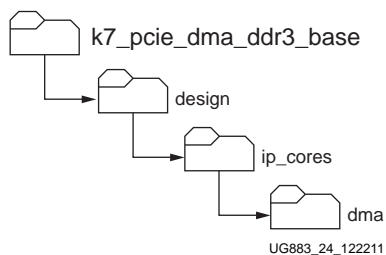


図 24 : Packet DMA のデザイン ファイル

マルチポート仮想 FIFO およびメモリ コントローラー ブロック

図 25 にマルチポート仮想 FIFO およびメモリ コントローラー ブロックのデザイン モジュール、図 26 にデザイン ファイルの構成を示します。

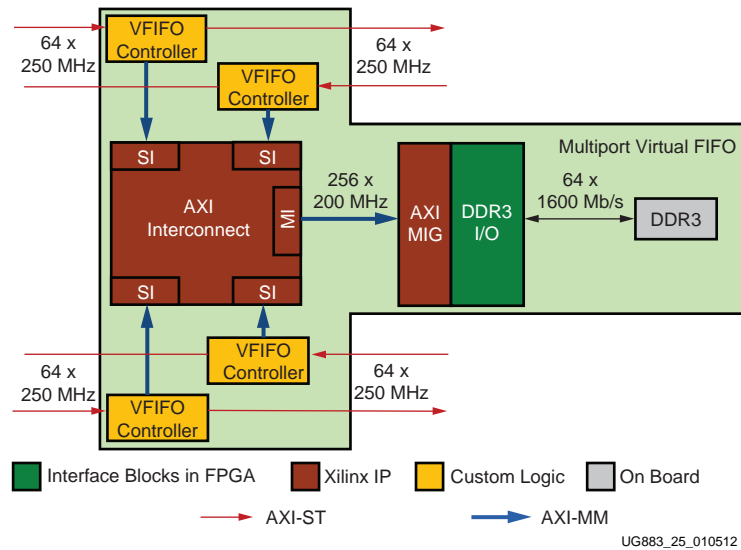


図 25 : マルチポート仮想 FIFO およびメモリ コントローラー ブロックのデザイン モジュール

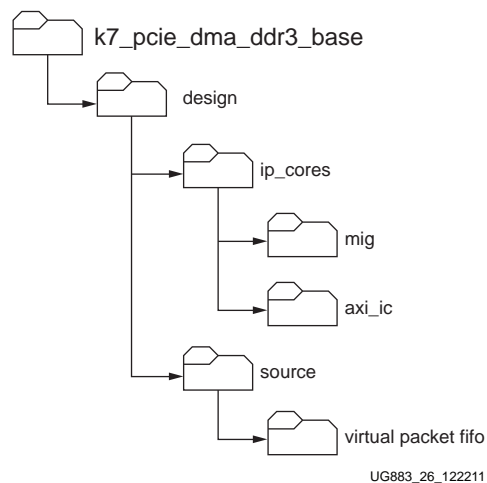
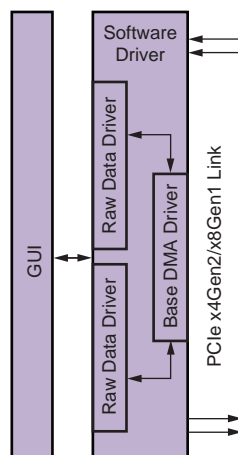


図 26 : マルチポート仮想 FIFO およびメモリ コントローラーのデザイン ファイル

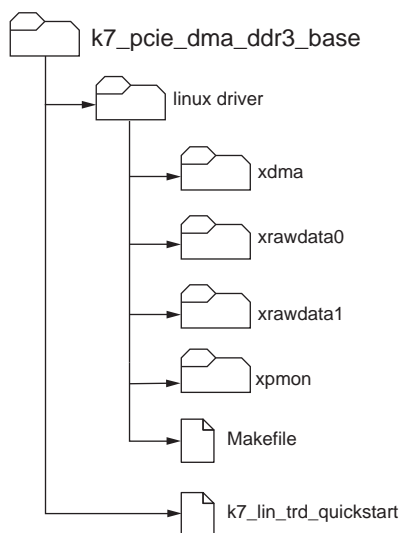
ソフトウェア デバイス ドライバーおよびソフトウェア アプリケーション/ GUI のファイルおよびスクリプト

図 27 に、ソフトウェア デバイス ドライバーおよびソフトウェア アプリケーション/GUI のファイルとスクリプトのデザイン モジュールを示します。図 28 には、ソフトウェア デバイス ドライバー、ソフトウェア アプリケーション、および GUI のデザイン ファイルの構成を示します。



UG883 27 010412

図 27: ソフトウェア デバイス ドライバーおよびソフトウェア アプリケーション/
GUI のデザイン モジュール



UG883 28 122211

図 28: ソフトウェア デバイス ドライバーおよびソフトウェア アプリケーション/GUI のファイル

ベース TRD のシミュレーション

Kintex-7 FPGA ベース TRD には、完全なシミュレーション環境が用意されています。シミュレーション環境および関連するシミュレーションファイルの詳細は、『Kintex-7 FPGA ベース ターゲットリファレンスデザイン ユーザーガイド』(UG882) の「はじめに」の章にある「シミュレーション」を参照してください。[参照 5]

Northwest Logic 社製 DMA IP の再利用

Kintex-7 FPGA ベース TRD に含まれる Packet DMA Controller は、Northwest Logic 社の PCIe Packet DMA IP コアの評価版です。DMA デザインには、次が含まれます。

- シミュレーション モデル
- ハードウェア評価ネットリスト (12 時間に制限)

Northwest Logic 社の PCIe Packet DMA IP コアの完全製品版の注文は、<http://www.nwlogic.com> から可能です。

ベース TRD の変更

Kintex-7 FPGA のベース TRD は、システム設計者がデザインを拡張したり変更する際の枠組みとなります。カスタム IP やデザイン ブロックを使用するデザインの強化、変更、再構築については、『Kintex-7 FPGA ベース ターゲット リファレンス デザイン ユーザー ガイド』(UG882) の「TRD プラットフォームによるデザイン」の章を参照してください。[参照 5]

デザイン サマリ

表 4 のデザイン サマリは、リファレンス デザインのシミュレーション、インプリメンテーション、ハードウェアの詳細についてまとめたものです。

表 4: デザイン サマリ

パラメーター	説明
全般	
開発元	ザイリンクス
ターゲット デバイス (ステッピング レベル、ES、製造、スピード グレード)	xc7k325t-2-ffg900
ソース コードの提供	あり (カスタム ロジックのみ)
ソース コードの形式	Verilog
既存のリファレンス デザインまたはアプリケーション ノート、サードパーティ、CORE Generator ソフトウェアからデザインへのコードまたは IP 流用の有無	サードパーティのコードおよび CORE Generator ソフトウェアによる LogiCORE IP を使用
シミュレーション	
機能シミュレーション	実施済み
タイミングシミュレーション	未実施
機能およびタイミング シミュレーションでのテストベンチ利用	あり (機能シミュレーション時)
テストベンチの形式	System Verilog (社内検証用)、Verilog (顧客用)
シミュレータ ソフトウェア/使用バージョン	ModelSim Questa 6.6d (社内検証) ModelSim 6.6d (セットアップ不要のシミュレーション サポート)
SPICE/IBIS シミュレーション	未実施

表 4：デザイン サマリ (続き)

パラメーター	説明
インプリメンテーション	
合成ソフトウェア ツール/使用バージョン	XST
インプリメンテーション ソフトウェア ツール/ 使用バージョン	ISE 13.4 ツール以降
スタティック タイミング解析	実施済み
ハードウェア検証	
ハードウェア検証	実施済み
検証に使用したハードウェア プラットフォーム	KC705 評価ボード

Kintex-7 FPGA のベース TRD に関する最新情報は、japan.xilinx.com/kc705 を参照してください。このページには、最新の資料、FAQ、リファレンス サンプル デザイン、製品アップデート、既知の問題点などが掲載されます。定期的に確認してください。

その他のリソース

ザイリンクスのリソース

シリコンやソフトウェア、IP に関するアンサー データベースを検索したり、テクニカル サポートのウェブケースを開く場合は、次のサポート ウェブサイトにアクセスしてください。

<http://japan.xilinx.com/support>

ザイリンクスの資料で使用されている技術用語については、次の用語集を参照してください。

http://japan.xilinx.com/support/documentation/sw_manuals/glossary.pdf

参考資料

次の文書は、このユーザー ガイドに役立つ補足資料です。

1. [UG810](#) : 『Kintex-7 FPGA 用 KC705 評価ボード ユーザー ガイド』
2. [UG798](#) : 『ISE Design Suite 13 : インストールおよびライセンス ガイド』
3. [UG477](#) : 『7 シリーズ FPGA PCI Express 用インテグレイテッド ブロック ユーザー ガイド』
4. [WP350](#) : 『PCI Express システム パフォーマンスの理解』
5. [UG882](#) : 『Kintex-7 FPGA ベース ターゲット リファレンス デザイン ユーザーガイド』
6. [UG476](#) : 『7 シリーズ FPGA GTX トランシーバー ユーザー ガイド』
7. [UG586](#) : 『7 シリーズ FPGA メモリ インターフェイス ソリューション ユーザー ガイド』
8. [UG883](#) : 『Kintex-7 FPGA ベース ターゲット リファレンス デザイン スタートアップ ガイド』
(このガイド)
9. AXI インターコネクト IP :
http://japan.xilinx.com/products/intellectual-property/axi_interconnect.htm
10. 合成/シミュレーション デザイン ガイド :
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx13_3/sim.pdf

ボードおよびキットに役立つその他のサイト

11. Kintex-7 FPGA KC705 評価キットのソフトウェア リリースによるデザイン アドバイザリ
<http://japan.xilinx.com/support/#nav=sd-nav-link-179661&tab=tab-bk>
12. Kintex-7 FPGA ベース TRD および Kintex-7 FPGA KC705 評価キットの最新情報
japan.xilinx.com/kc705
13. KC705 サポート ページ
<http://japan.xilinx.com/products/boards-and-kits/EK-K7-KC705-G.htm>

サードパーティのリソース

ベース TRD によって使用されるほかのソフトウェア、ツール、および IP に関連する文書は、各ベンダーのウェブサイトで見ることができます。

14. Northwest Logic 社製 DMA バック エンド コア
<http://www.nwlogic.com/packetdma>

15. Fedora プロジェクト
<http://fedoraproject.org>

Fedora は、この TRD の開発に使用される Linux ベースのオペレーティング システムです。

16. GTK+ プロジェクトの API 資料
<http://www.gtk.org/documentation.php>

GTK+ は、グラフィカル ユーザー インターフェイス (GUI) を作成するためのツールキットです。