

PCI-Express ハード IP を使用した DMA の実現
for Cyclone V GT FPGA 開発キット
(ソフトウェア編)

ver.1

PCI-Express ハード IP を使用した DMA の実現 for Cyclone V GT FPGA 開発キット (ソフトウェア編)

目次

1. はじめに	3
1-1. 概要	3
1-2. 対象デバイス・ボード	3
1-3. 環境	3
1-4. 対象デザイン	3
2. ソフトウェアのビルドから実行まで	4
2-1. ハードウェア設定	4
2-2. 簡易ソフトウェアの展開	4
2-3. Jungo 社 WinDriver によるドライバの実装	5
2-4. Microsoft Visual Studio 2010 でのビルド	9
2-5. ドライバのインストール	15
2-6. 生成したプログラムの実行	21
2-7. 実行結果	22
3. まとめ	24
改版履歴	25

1. はじめに

1-1. 概要

今回は、前回記事 “PCI-Express ハード IP を使用した DMA 転送の実現 for Cyclone V GT FPGA 開発キット” の基礎編およびハードウェア編の続編となるソフトウェア編となります。今回は、Qsys システム統合ツールでの PCI-Express ハード IP リファレンス・デザインのハードウェア部分の詳細について触れましたが、今回はソフトウェア編ということで、前回記事で紹介した Cyclone[®] V GT FPGA のハードウェア・デザインに対して Altera Wiki に掲載されている簡易ソフトウェア・デザインをベースとして Jungo 社 WinDriver[™] による簡単なデバイス・ドライバ作成の実例と Microsoft[®] Visual Studio 2010 によるビルドの方法について紹介します。この簡易ソフトウェア・デザインは、Altera Wiki から誰でもダウンロードでき、Windows[®] 環境から Qsys システムで作成された PCI-Express[®] ハードウェアにアクセスするための簡易ソフトウェアです。C++ 言語のソース・コードが公開されているため、ユーザはこのデザインを参考にして、ハードウェアの動作検証やオリジナルのソフトウェア作成の参考にすることができます。

今回のソフトウェア編を通じて、Qsys での PCI-Express ハード IP を使用したデザイン実装のノウハウを理解し、ユーザ独自のシステム構築の助けになることを目的とします。

1-2. 対象デバイス・ボード

今回対象としたアルテラ社 FPGA デバイスとボードの情報は以下となります。

Cyclone V GT FPGA 開発キット

PCI Express Gen1, Gen2 x1 / x2 / x4 に対応、最大ペイロード・サイズ 512Bytes

https://www.altera.co.jp/products/boards_and_kits/dev-kits/altera/kit-cyclone-v-gt.html

1-3. 環境

PC: ASUS P8Z77-M PRO

チップセット: インテル社製 Z77 Express

OS: Windows 7 64bit

ドライバ: Jungo 社 WinDriver (<http://www.jungo.com/>) v10.21

ソフトウェア: Microsoft Visual Studio 2010

1-4. 対象デザイン

Altera Wiki の以下の URL よりダウンロードしたアルテラ社 PCI-Express 開発キット簡易検証用のソフトウェア・デザイン

http://www.alterawiki.com/uploads/b/b4/Alt_pcie_qlsys_simple_sw.zip

2. ソフトウェアのビルドから実行まで

本章では、まず、簡易ソフトウェア・デザインを使用して、実機で動作させるまでの手順について説明します。なお、Jungo 社 WinDriver、Microsoft Visual Studio 2010 は既に正しくターゲット PC にインストールされていることを前提としています。Jungo 社 WinDriver は、Jungo 社の Web サイトからダウンロード&インストールして、30 日までは試用期間として使用できます。

2-1. ハードウェア設定

ハードウェア設定については、基礎編の3章のデモンストレーション(1)～(7)と同じですので、詳細についてはそちらを参照してください。2-3 章で説明する Jungo 社 WinDriver は、ターゲットとなるハードウェアを動的に検出して必要な情報を収集し、デバイス・ドライバを作成するツールです。したがって、WinDriver 起動の時点でハードウェアが正しく動作する必要があります。

2-2. 簡易ソフトウェアの展開

Altera Wiki よりダウンロードした簡易ソフトウェア・ファイル alt_pcie_qsys_simple_sw.zip を、適当なフォルダに解凍します。解凍したファイルを、今回は C:\work\alt_pcie_qsys_symplesw フォルダに格納しました。

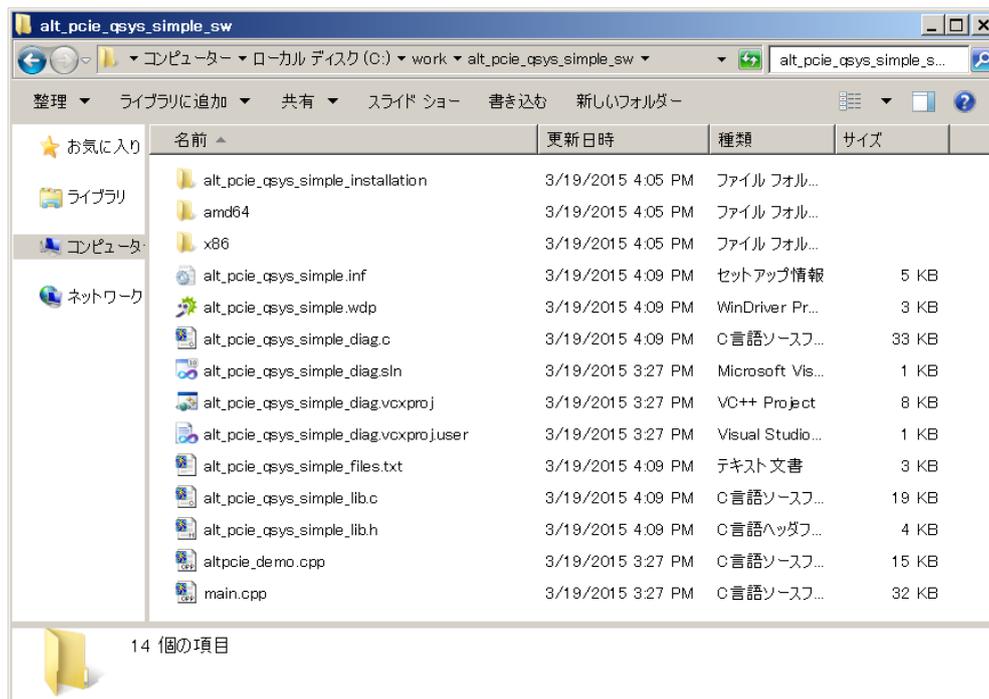


図 2-1. 簡易ソフトウェアの解凍

2-3. Jungo 社 WinDriver によるドライバの実装

- (1) スタート・メニュー ⇒ 全てのプログラム ⇒ WinDriver ⇒ Driver Wizard を実行し、WinDriver を起動します。



図 2-2. Jungo 社 WinDriver の起動

- (2) Open an existing project をクリックし、C:\work\alt_pcie_qsys_simple_sw フォルダ内の alt_pcie_qsys_simple_sw.wdp を選択し、“開く”をクリックします。

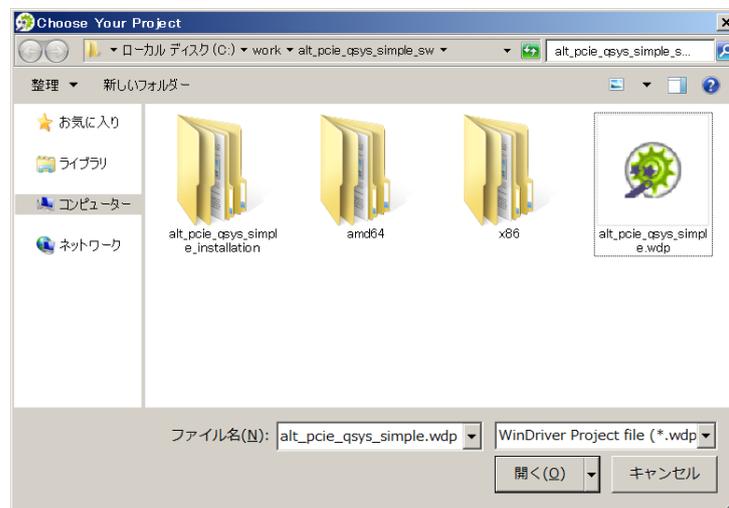


図 2-3. WinDriver プロジェクトの選択

(3) 以下のようなダイアログ・ボックスが表示された場合には、“Yes” をクリックしてください。



図 2-4. DriverWizard

(4) DriverWizard が起動します。ここではアルテラ社のベンダ ID (1172) やデバイス ID (E001)、その他、PCI コンフィギュレーション空間のレジスタ値が確認できます。

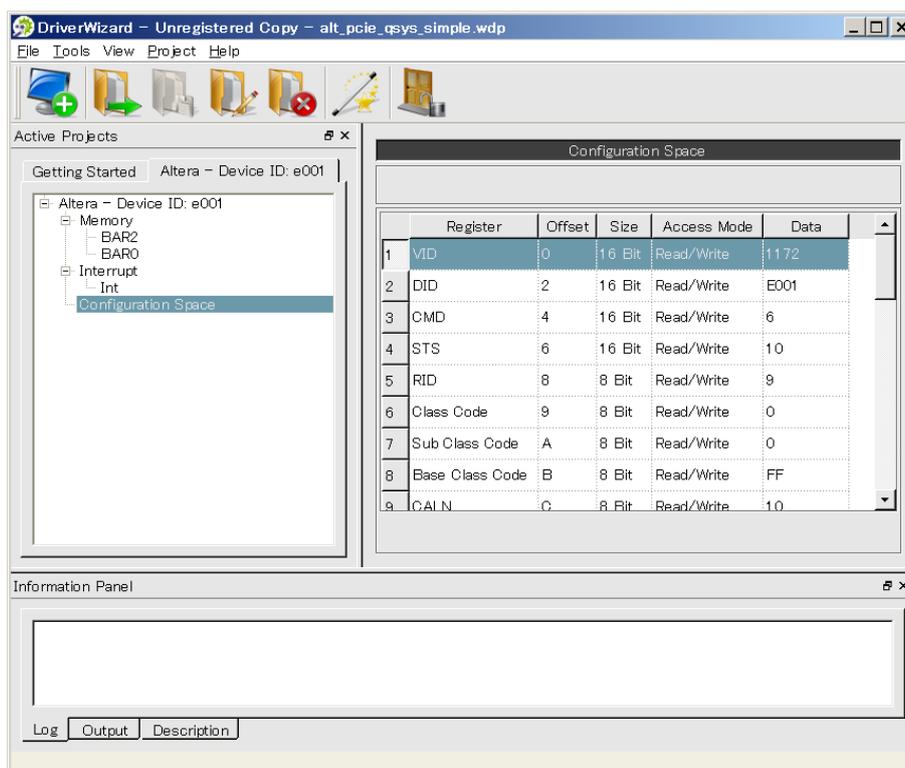


図 2-5. DriverWizard の起動

(5) Generate Code ボタンをクリックします。



図 2-6. Generate Code ボタン

(6) Select Code Generation Option を図のように設定します。IDE to Invoke で “MS Developer Studio .NET” を選択すると、自動的に Microsoft Visual Studio 用のプロジェクト・ファイル一式を作成してくれます。“OK” ボタンをクリックします。

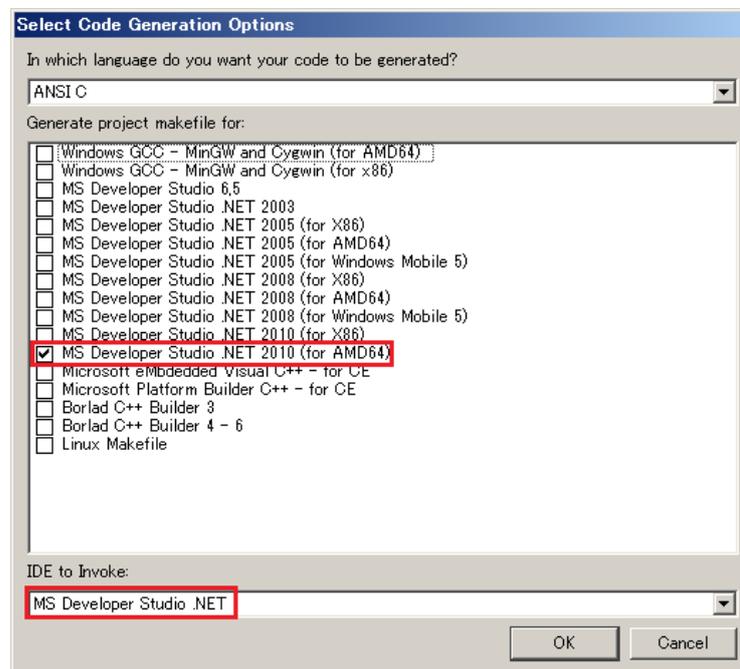


図 2-7. Select Code Generation Option

(7) Select Additional Option が表示されるので、そのまま “OK” をクリックします。

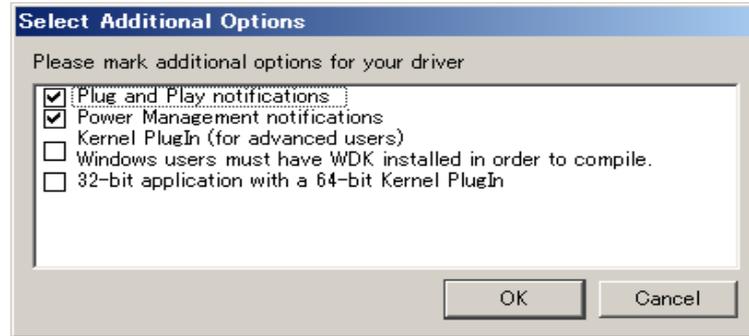


図 2-8. Select Additional Option

(8) Review Generated Project Files Information が表示されるので、そのまま “OK” をクリックします。すると、作成された Microsoft Visual Studio 用のプロジェクト・ファイルから自動的に Microsoft Visual Studio 2010 が起動します。

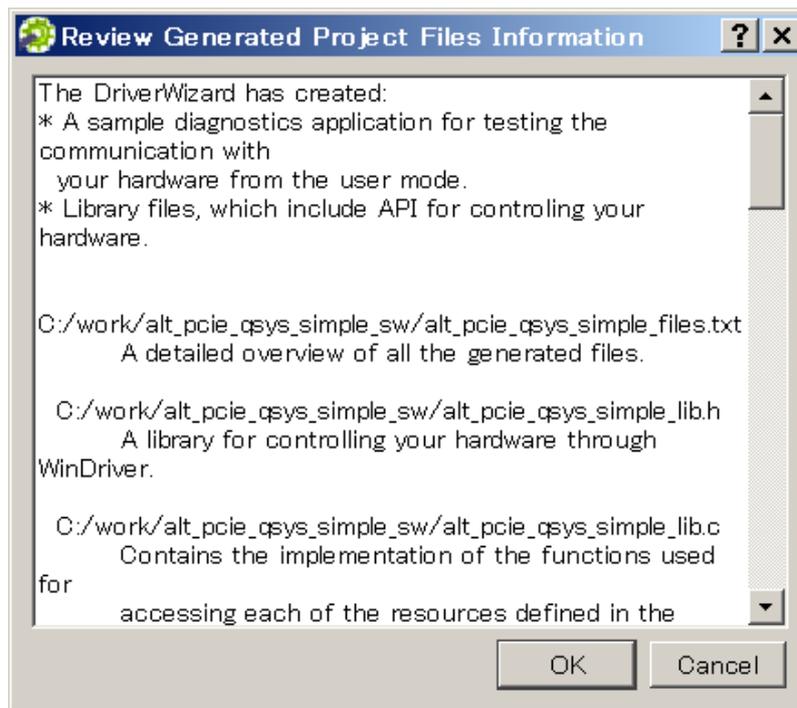


図 2-9. Review Generated Project Files Information

(9) Microsoft Visual Studio 2010 が起動したら、WinDriver (DriverWizard) は × ボタンをクリックしてシャットダウンしてください。起動したままだと、後に作成するプログラムを実行する際にリソースの競合が発生してプログラムが正しく動作しなくなります。

2-4. Microsoft Visual Studio 2010 でのビルド

WinDriver の手順が完了すると、自動的に Microsoft Visual Studio 2010 が起動します。初めの状態では、WinDriver が自動生成したファイルが登録されています。このうち、alt_pcie_qsys_simple_diag.c をプロジェクトから除外し、展開したフォルダにある main.cpp、altpcie_demo.cpp をプロジェクトに追加します。また、これらのファイルは、C++ 言語で書かれているため、オプション設定を変更して、C++ 言語のコンパイルを有効にする必要があります。

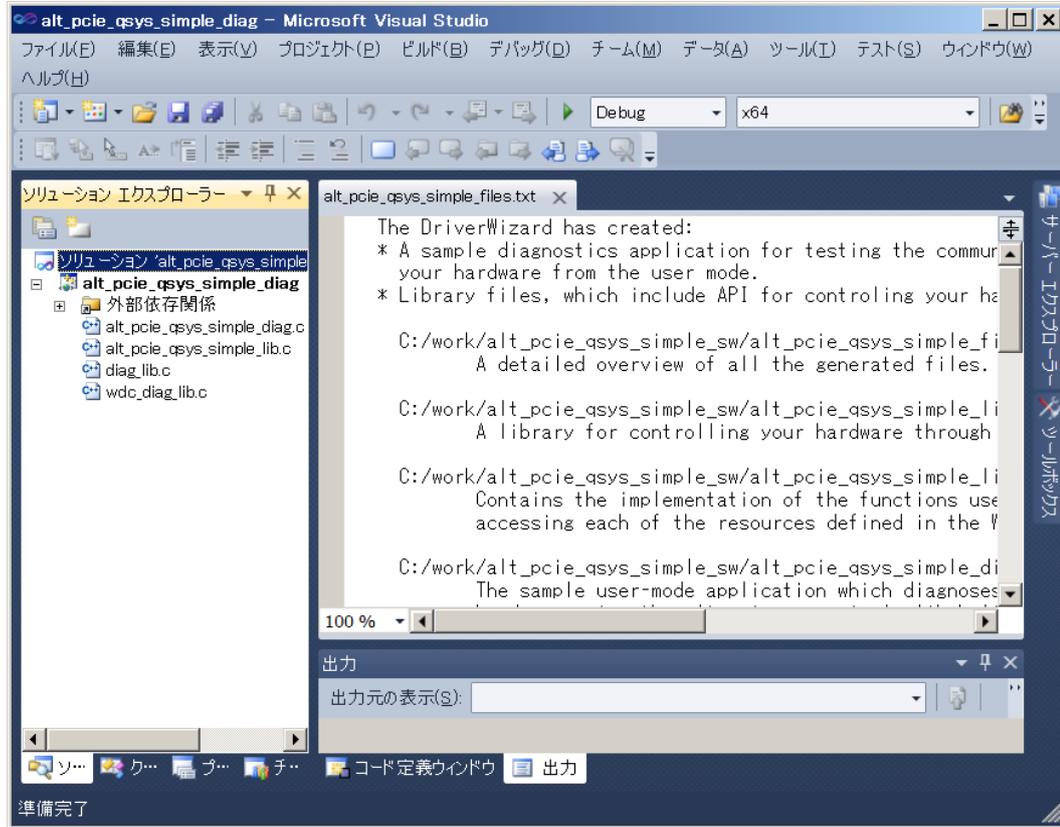


図 2-10. Microsoft Visual Studio 2010

- (1) alt_pcie_qsys_simple_diag.c を選択し、右クリックでメニューを表示させます。“プロジェクトからの除外” を左クリックし、alt_pcie_qsys_simple_diag.c をプロジェクトから除外します。

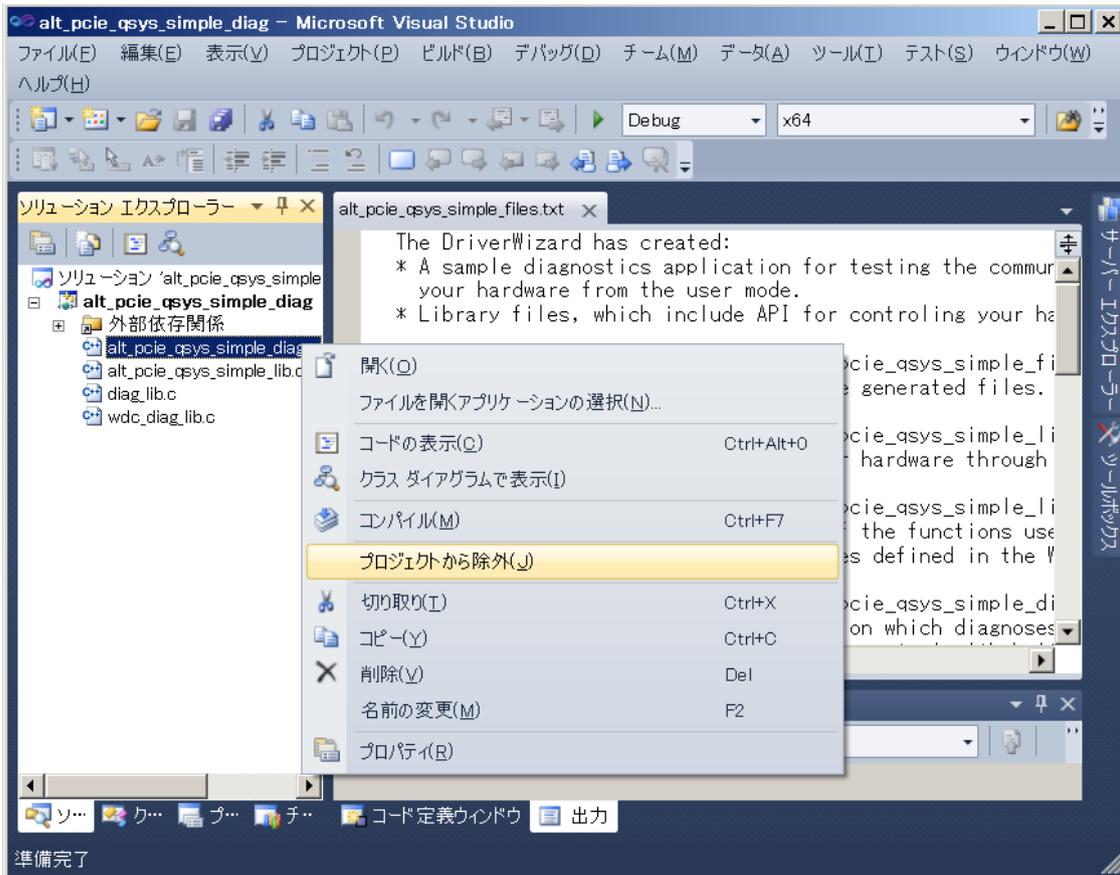


図 2-11. 不要ファイルの除外

- (2) ソリューション・エクスプローラ上の alt_pcie_qsys_simple.diag を右クリックし、“追加” ⇒ “既存の項目” を左クリックして、main.cpp と plt_pcie_demo.cpp を選択します。

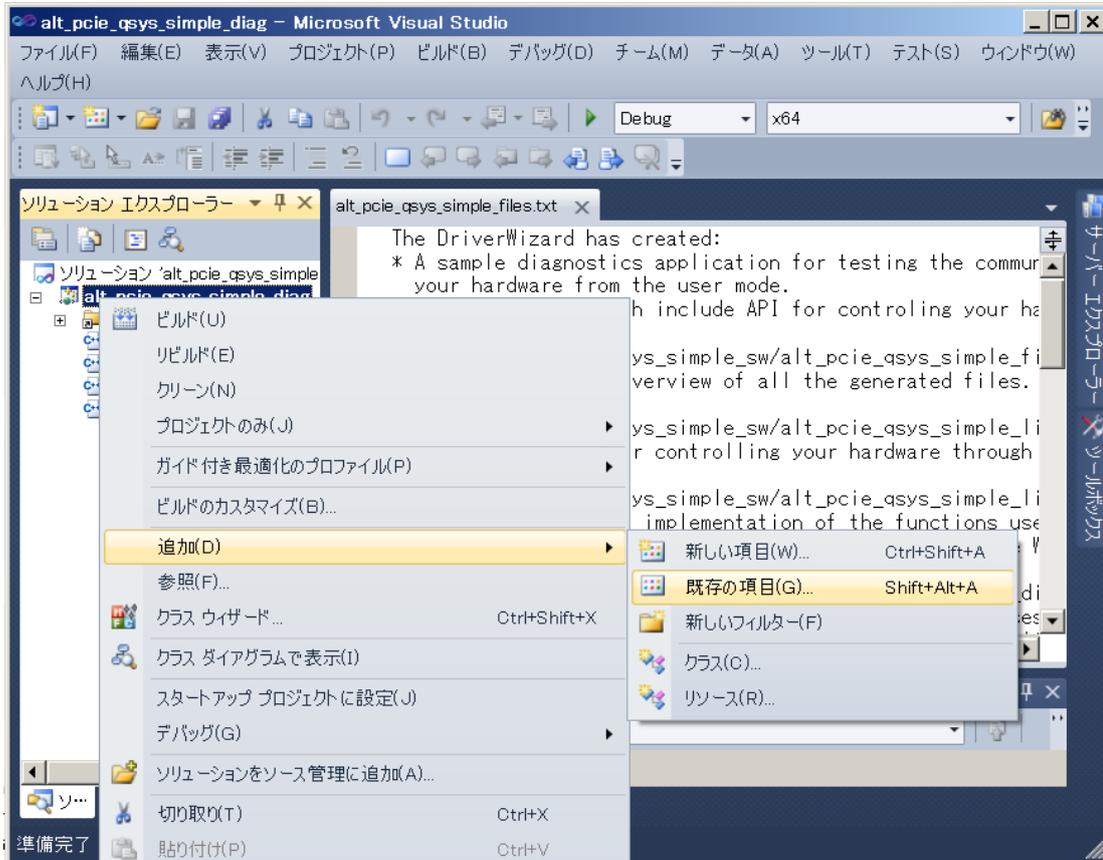


図 2-12. 既存ファイルの追加

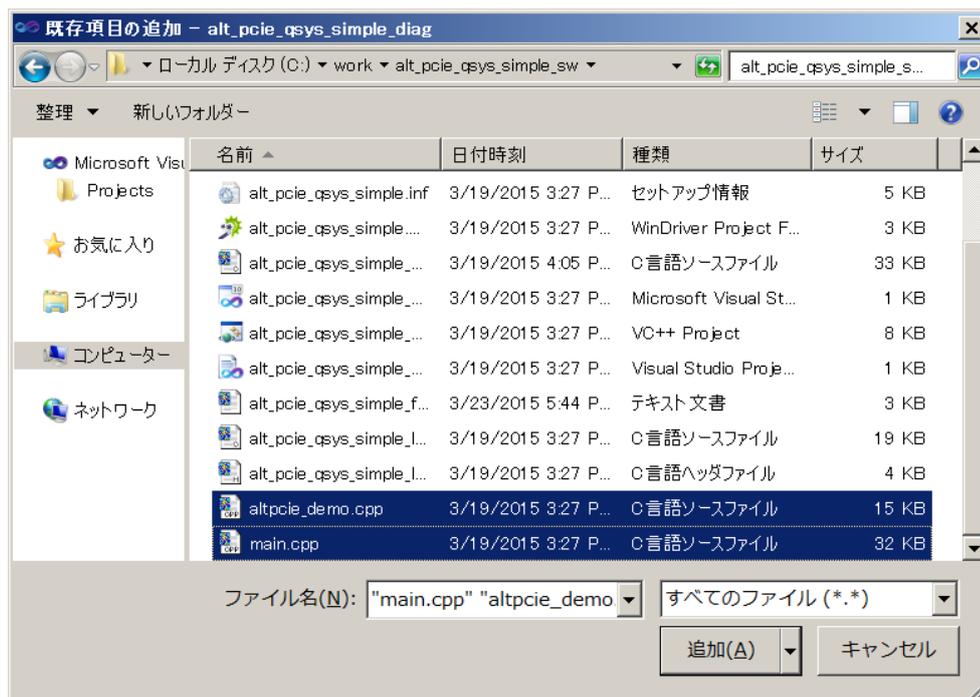


図 2-13. 既存ファイルの追加

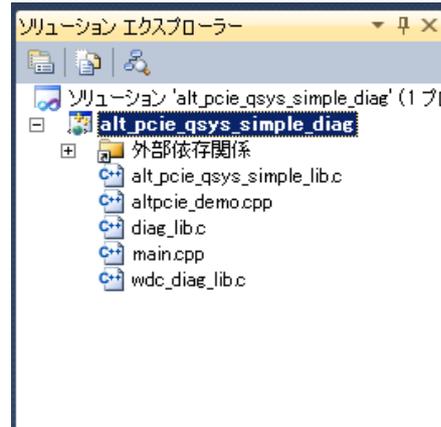


図 2-14. 除外、追加後の状態

- (3) プロジェクト・メニュー ⇒ プロパティをクリックし、alt_pcie_qsys_simple_diag プロパティ・ページを開きます。構成プロパティ ⇒ C/C++ ⇒ 詳細設定 ⇒ コンパイル言語の選択から “C++ コードとしてコンパイル (/TP)” を設定し、“OK” をクリックします。

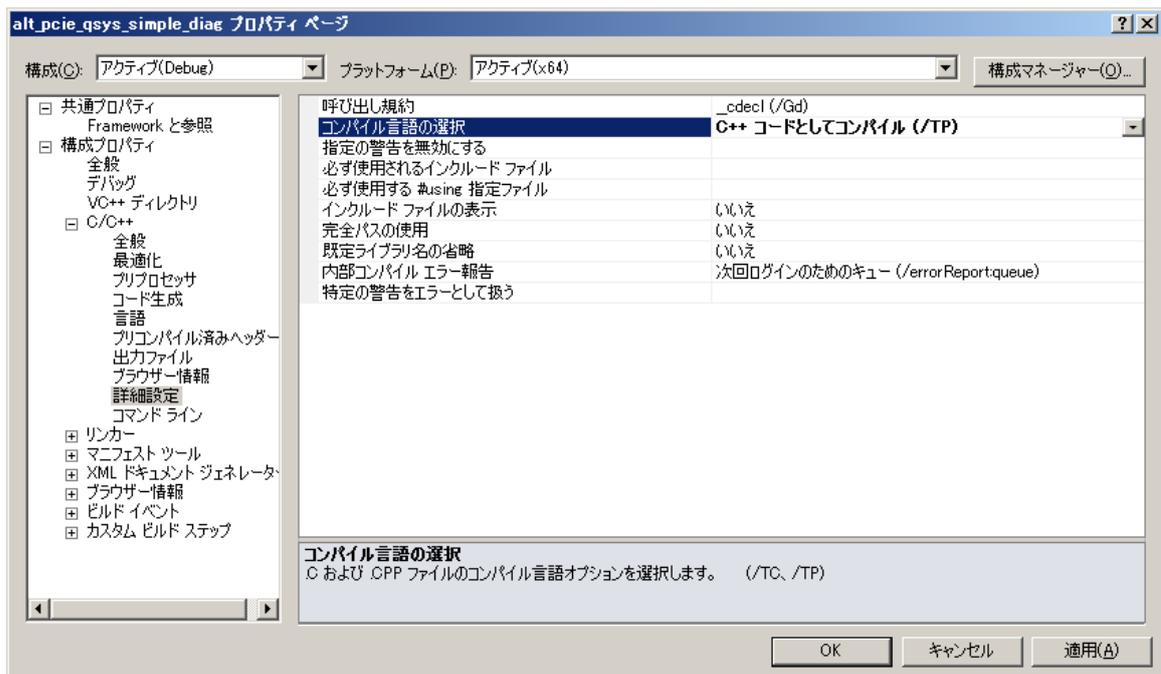


図 2-15. C++ 言語の有効化

(4)ビルド・メニュー ⇒ ソリューションのビルドを実行します。

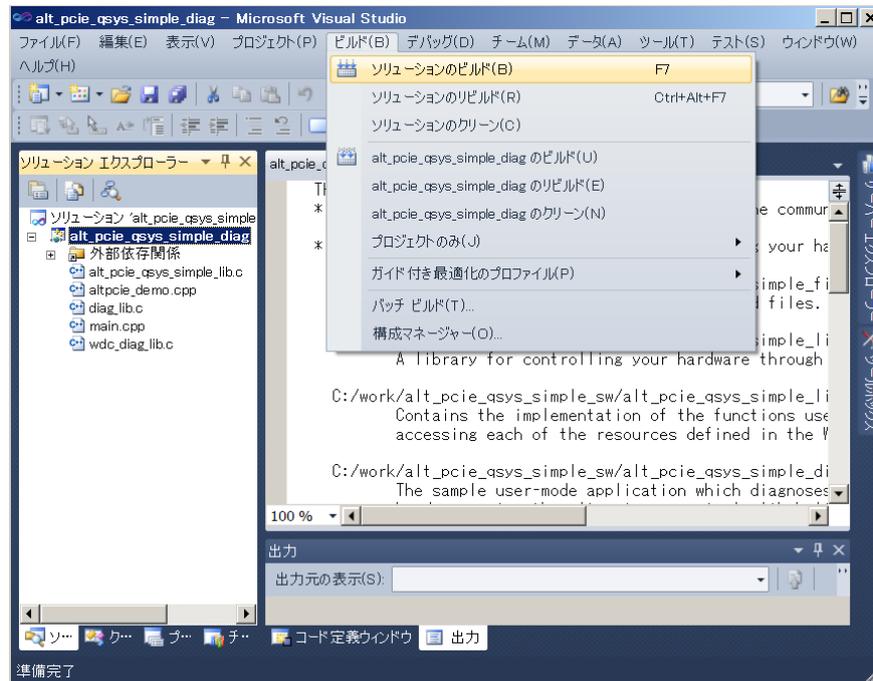


図 2-16. ビルドの実行

(5)ビルドが正常に終了すると、図のようなログが表示されます。

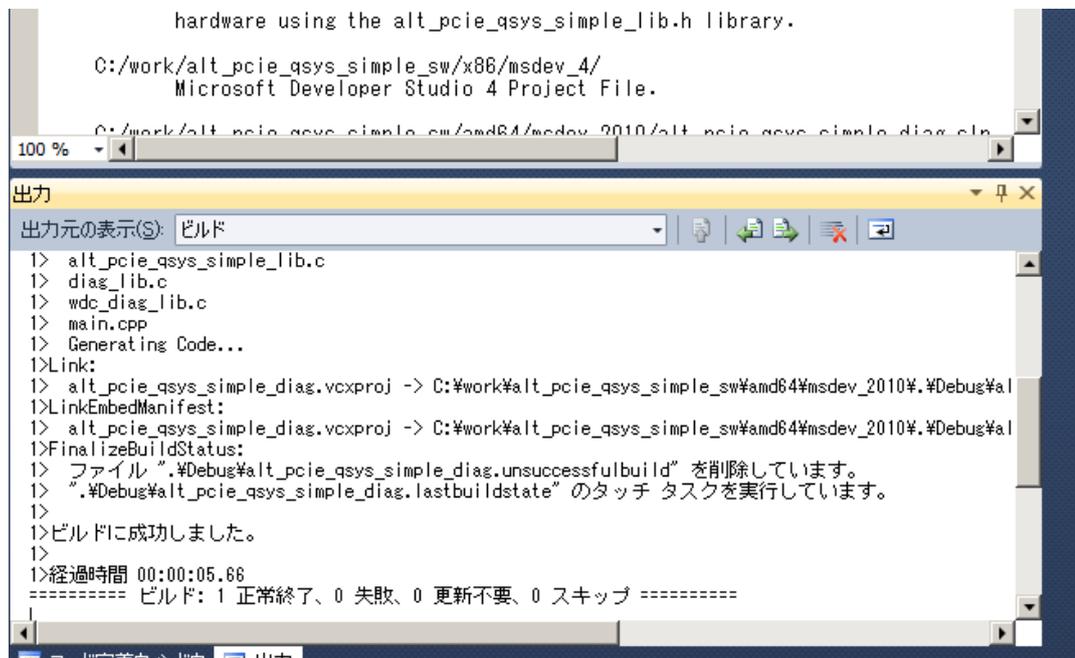


図 2-17. ビルド実行後のログ画面

(6) 実行形式ファイルが生成されていることを確認します。

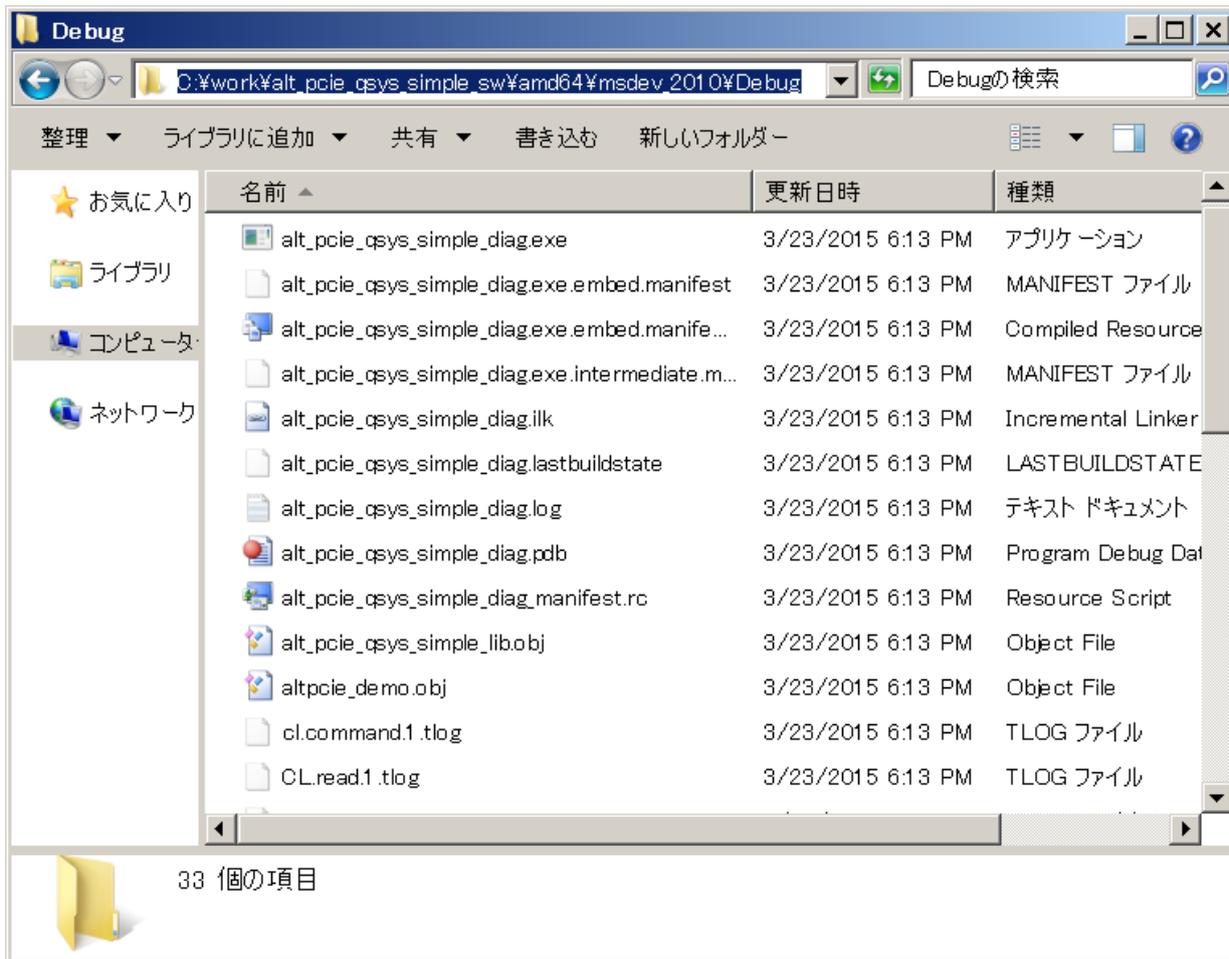


図 2-18. 実行形式ファイルの確認

2-5. ドライバのインストール

次に、ドライバのインストールを行います。WinDriver を実行した時点でプロジェクト・フォルダ内にハードウェア情報を認識した結果として、ドライバ・ファイル一式が生成されています。このドライバ・ファイルを使用して Cyclone V GT 開発キット上のハードウェアに対してドライバをインストールします。

(1) 以下のパスにドライバ設定用のファイルが生成されています。

C:\work\alt_pcie_qsys_simple_sw\alt_pcie_qsys_simple_installation\redist

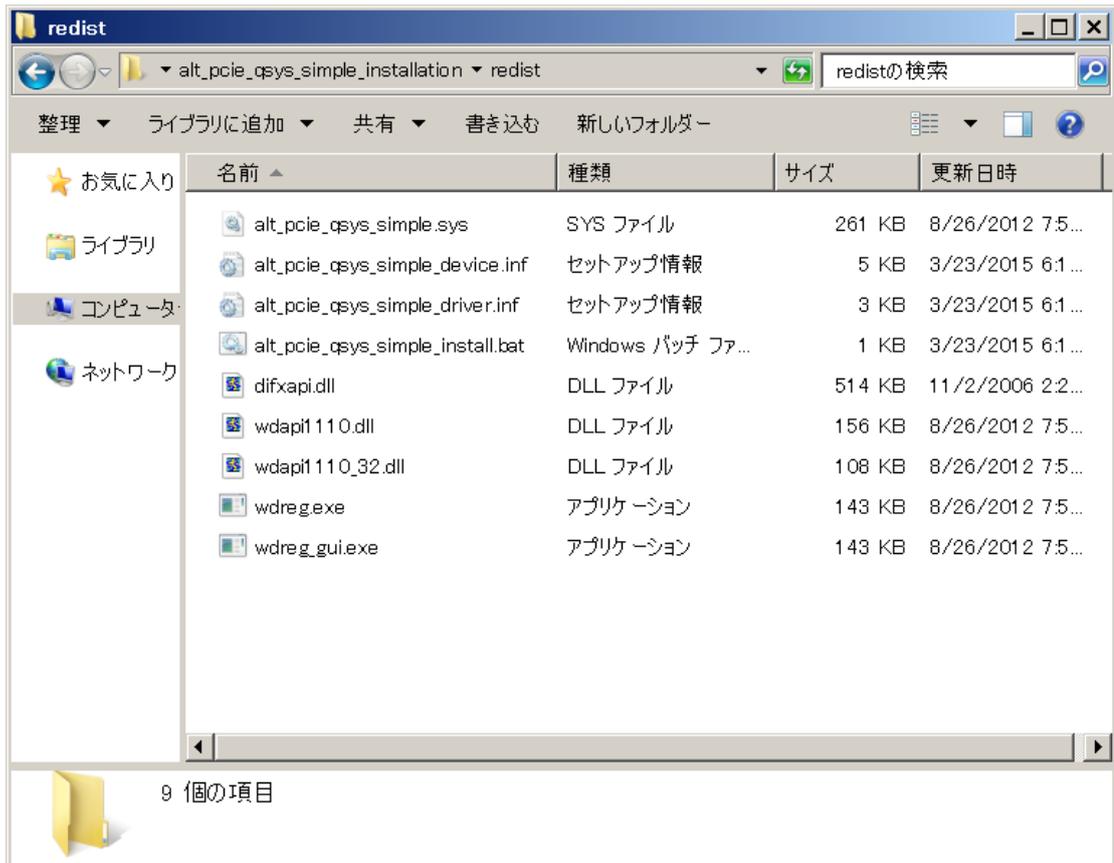
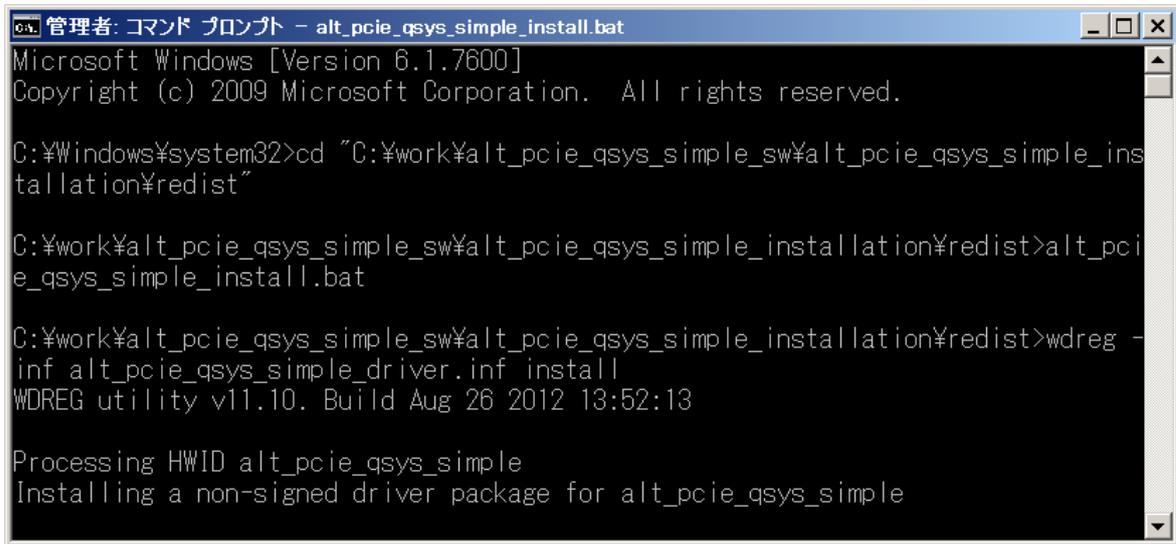


図 2-19. ドライバ・ファイル一式

- (2) コマンド・プロンプトを管理者権限で実行して、このフォルダ内の altpcie_qsys_simple_install.bat を実行します。



```

C:\Windows\system32>cd "C:\work\alt_pcie_qsys_simple_sw\alt_pcie_qsys_simple_installation\redist"

C:\work\alt_pcie_qsys_simple_sw\alt_pcie_qsys_simple_installation\redist>alt_pcie_qsys_simple_install.bat

C:\work\alt_pcie_qsys_simple_sw\alt_pcie_qsys_simple_installation\redist>wdreg -inf alt_pcie_qsys_simple_driver.inf install
WDREG utility v11.10. Build Aug 26 2012 13:52:13

Processing HWID alt_pcie_qsys_simple
Installing a non-signed driver package for alt_pcie_qsys_simple
    
```

図 2-20. ドライバのインストール

- (3) Windows のセキュリティに関する画面が表示されるので、“このドライバ・ソフトウェアをインストールします”を選択します。

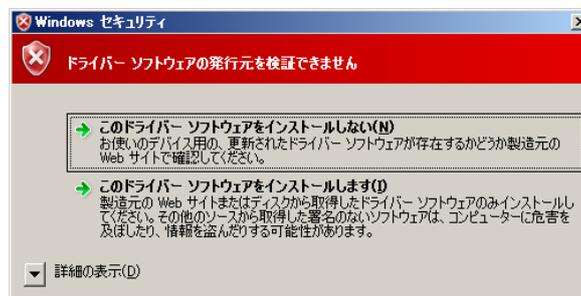


図 2-21. Windows セキュリティ画面

- (4) ドライバのインストールは数分かかる場合があります。インストールが完了すると“デバイス マネージャー”は以下のようになっています。Jungo の中に alt_pcie_qsys_simple が追加されたことを確認してください。WinDriver をインストールしたときに自動的に追加される汎用ドライバなので、ここでは使用しません。“ほかのデバイス”にドライバが割り当てられていない“PCI デバイス”があります。この時点では、まだドライバの割り当てが正しく行われていません。

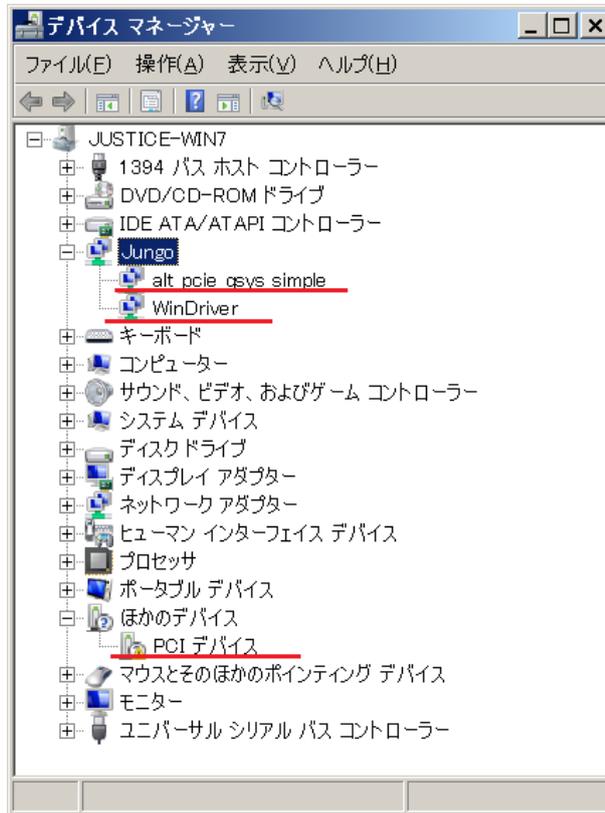


図 2-22. デバイス・マネージャー画面

- (5) デバイス・マネージャーの“ほかのデバイス”の“PCI デバイス”を右クリックで“ドライバ・ソフトウェアの更新”を実行します。

(6) “コンピュータを参照してドライバ・ソフトウェアを検索します” を選択します。

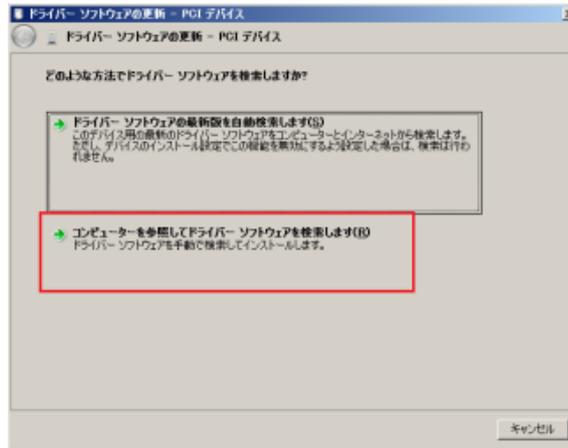


図 2-23. ドライバ・ソフトウェアの更新画面

(7) 次に、“コンピュータ上のデバイス・ドライバの一覧から選択します” を選択します。

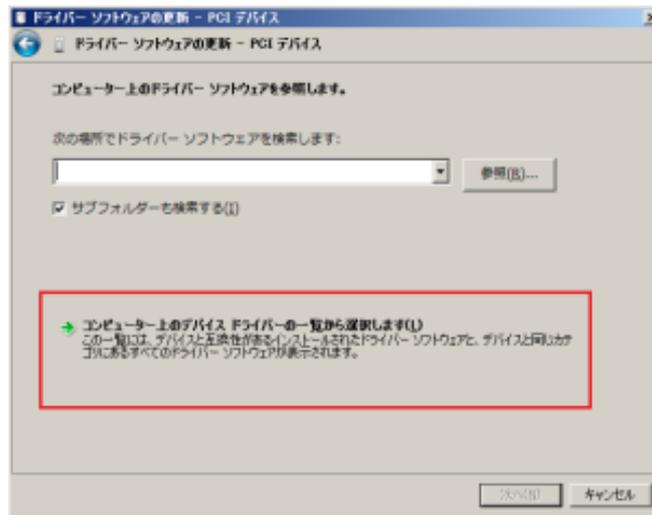


図 2-24. ドライバ・ソフトウェアの更新画面

(8) Jungo を選択します。

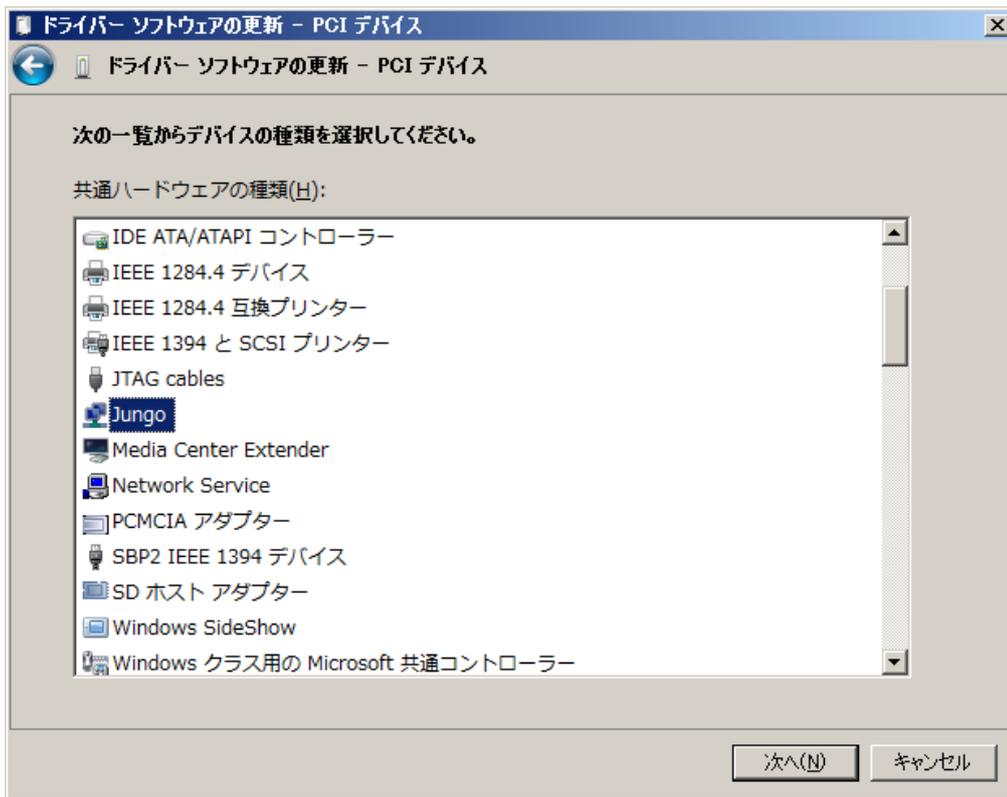


図 2-25. ドライバ・ソフトウェアの更新画面

- (9) alt_pcie_qsys_simple を選択します。“次へ” をクリックすると互換性について警告が表示されるので“はい” を選択しインストールを続行します。ドライバが正常にインストールをされたら“閉じる” をクリックします。alt_pcie_qsys_smiple が表示されない場合は“互換性のあるハードウェアを表示” のチェックを外してください。

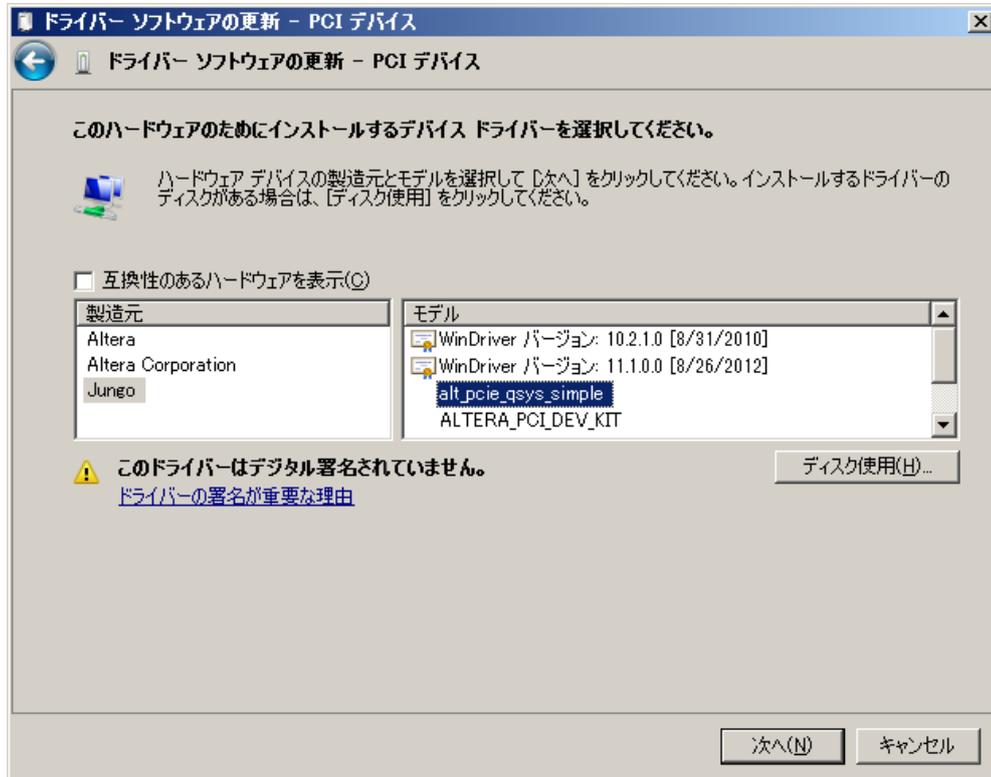


図 2-26. ドライバ・ソフトウェアの更新画面

- (10) ドライバが正常にインストールされると、以下の図のように“ほかのデバイス”にあった“PCI デバイス” がなくなり、alt_pcie_qsys_simple がもう一つ追加されます。

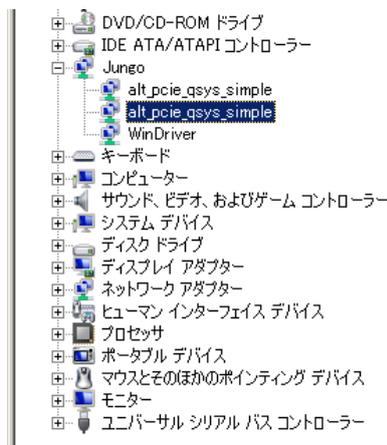


図 2-27. デバイス・マネージャー画面

2-6. 生成したプログラムの実行

生成したプログラムは、コマンド・プロンプトから実行できます。コマンド・プロンプトを起動して cd コマンドで実行形式のあるフォルダに移動します。

```
C:\> cd "C:\work\alt_pcie_qsys_simple_sw\amd64\msdev_2010\Debug"
```

生成した実行形式ファイルを実行します。

```
C:\work\alt_pcie_qsys_simple_sw\amd64\msdev_2010\Debug>alt_pcie_qsys_simple_diag.exe
```

```

ALT_PCIE_QSYS_SIMPLE diagnostic utility.
Application accesses hardware using WinDriver.

Found 1 matching device [ Vendor ID 0x1172, Device ID 0xE001 ]:

1. Vendor ID: 0x1172, Device ID: 0xE001
   Location: Bus 0x7, Slot 0x0, Function 0x0
   Interrupt: IRQ 16
   Interrupt Options (supported interrupts):
     Message-Signaled Interrupt (MSI)
     Level-Sensitive Interrupt
   Memory range [BAR 0]: base 0xB0000000, size 0x10000000
   Memory range [BAR 2]: base 0xF0000000, size 0x8000000

target read example
7000000 : dea172de
7000004 : 9f576bdb
7000008 : 7f66a9a5
700000c : c99a03f
7000010 : 82b1181c

target write example
7000000 : 0
7000004 : 1
7000008 : 2
700000c : 3
7000010 : 4

DMA read example
DMA read : 456.005632 MByte/sec
DMA read : 880.562600 MByte/sec
DMA read : 1154.183747 MByte/sec
DMA read : 1247.575714 MByte/sec

DMA write example
DMA write : 608.007509 MByte/sec
DMA write : 1064.013141 MByte/sec
DMA write : 1301.213523 MByte/sec
DMA write : 1389.731450 MByte/sec

C:\work\alt_pcie_qsys_simple_sw\amd64\msdev_2010\Debug>

```

図 2-28. 実行結果

2-7. 実行結果

(1) ハードウェア情報を読み出して表示しています。

```

ALT_PCIE_QSYS_SIMPLE diagnostic utility.
Application accesses hardware using WinDriver.

Found 1 matching device [ Vendor ID 0x1172, Device ID 0xE001 ]:

1. Vendor ID: 0x1172, Device ID: 0xE001
   Location: Bus 0x7, Slot 0x0, Function 0x0
   Interrupt: IRQ 16
   Interrupt Options (supported interrupts):
     Message-Signaled Interrupt (MSI)
     Level-Sensitive Interrupt
   Memory range [BAR 0]: base 0xB0000000, size 0x10000000
   Memory range [BAR 2]: base 0xF0000000, size 0x8000000
    
```

図 2-29. ハードウェア情報の表示

(2) ターゲットのリードとライトを実行しています。ここでは、まず 0x7000000 番地から 0x7000010 の値を読み出し、その後、同じアドレスに 0, 1, 2, 3, 4 をそれぞれ書き込んでいます。ちなみに、この 0x7000000 番地は、FPGA 内部のオンチップ・メモリに該当します。詳細はハードウェア編をご参照ください。

```

target read example
7000000 : dea172de
7000004 : 9f576bdb
7000008 : 7f66a9a5
700000c : c99a03f
7000010 : 82b1181c

target write example
7000000 : 0
7000004 : 1
7000008 : 2
700000c : 3
7000010 : 4
    
```

図 2-30. ターゲット・リード/ライト

(3) DMA リード/ライトをそれぞれ 4 回ずつ実行しています。結果は値でなく、実効転送レートとなっています。ソース・コードを見ると、ターゲット・アドレスが 0x7000000 となっているので、PC の共有メモリとオンチップ・メモリ間で DMA 転送を行っていることがわかります。

結果を見ると、それぞれ転送バイト数が異なります。ソフトウェアのソース・コードを見ると、

- 1 回目 : 4096Bytes
- 2 回目 : 16384Bytes
- 3 回目 : 65536Bytes
- 4 回目 : 262144Bytes

となっており、転送バイト数が多いほど実効転送レートが向上しているのがわかります。

実効転送レートは以下の式で計算しています。

$$\text{転送レート (MByte/sec)} = \text{転送バイト数} \div \text{実行時間 (sec)} \div 1000000$$

ハードウェアは Gen2 の 4 レーン なので、8b/10b のオーバー・ヘッド分を差し引いたデータ・レートは、5000Mbps × 8/10 × 4 レーン = 8000Mbps ⇒ 2000Mbyte/sec となります。

従って、262144Byte 転送時のデータ・レートに対する転送効率以下となります。

$$\text{リード時 : } 1247.575714 \text{ MByte/sec} \div 2000 \text{ MByte/sec} = \text{約 } 63\%$$

$$\text{ライト時 : } 1389.731450 \text{ MByte/sec} \div 2000 \text{ MByte/sec} = \text{約 } 69\%$$

```
DMA read example
DMA read : 456.005632 MByte/sec
DMA read : 880.562600 MByte/sec
DMA read : 1154.183747 MByte/sec
DMA read : 1247.575714 MByte/sec

DMA write example
DMA write : 608.007509 MByte/sec
DMA write : 1064.013141 MByte/sec
DMA write : 1301.213523 MByte/sec
DMA write : 1389.731450 MByte/sec
```

図 2-31. DMA リード・ライト

3. まとめ

今回は、“PCI-Express ハード IP を使用した DMA の実現 for Cyclone V GT FPGA 開発キット(ソフトウェア編)”ということで、Cyclone V GT 開発キット・ベースのリファレンス・デザインを用いて、Altera Wiki にある簡易ソフトウェアのコンパイル、WinDriver のインストール、生成した実行形式ファイルでの動作確認までを紹介しました。

このソフトウェアは、Altera Wiki のホームページから無償でダウンロードできるものです。これはソフトウェアに精通していないユーザ向けのリファレンスとして提供されています。

そのため、同様の手順で独自に作成した FPGA のハードウェア・デザインの簡単な動作確認を行うことができます。また、ソフトウェアのソース・コードをカスタマイズすることで、独自デザインの構成に応じた動作確認もできます。

改版履歴

Revision	年月	概要
1	2015 年 4 月	初版

免責およびご利用上の注意

弊社より資料を入手されましたお客様におかれましては、下記の使用上の注意を一読いただいた上でご使用ください。

1. 本資料は非売品です。許可無く転売することや無断複製することを禁じます。
2. 本資料は予告なく変更することがあります。
3. 本資料の作成には万全を期していますが、万一ご不明な点や誤り、記載漏れなどお気づきの点がありましたら、本資料を入手されました下記代理店までご一報いただければ幸いです。

株式会社アルティマ ホームページ: <http://www.altima.co.jp> 技術情報サイト EDISON: <https://www.altima.jp/members/index.cfm>

株式会社エルセナ ホームページ: <http://www.elsena.co.jp> 技術情報サイト ETS : <https://www.elsena.co.jp/elspear/members/index.cfm>

4. 本資料で取り扱っている回路、技術、プログラムに関して運用した結果の影響については、責任を負いかねますのであらかじめご了承ください。
5. 本資料は製品を利用する際の補助的な資料です。製品をご使用になる際は、各メーカー発行の英語版の資料もあわせてご利用ください。