

TOE10G-IP マルチセッション・リファレンス・デザイン説明書 (Xilinx 版)

Rev1.0J 2017/12/04

1. はじめに

本ドキュメントに先立って、TOE10G-IP コアの標準デザインを説明した以下の[TOE10G-IP 標準リファレンス・デザイン説明書]を参照してください。

URL: http://www.dgway.com/products/IP/TOE10G-IP/dg_toe10gip_refdesign_xilinx_jp.pdf

この標準デザイン説明書で TOE10G-IP の基本的な動作を理解してください。本マルチセッション・デモは 8 個の TOE10G-IP コアを FPGA 内にインスタンスすることで最大 8 セッションの同時データ転送をサポートします。UserCtrl モジュール自体は標準デザインと同一です、ただし全 8 個の TOE10G-IP コアと単一の EMAC-IP を接続するため、マルチプレクサが挿入されています。より詳細については本ドキュメント内で説明します。

2. リファレンス・デザインの動作環境

本マルチセッション・デモの動作環境を図 1 に示します。

- ZC706/VC707/KCU105 プラットフォーム
- Bit ファイル・コンフィグレーション用の Vivado/ISE ツール
- 10 ギガビット SFP+ DAC ケーブルまたは 10Gbit 光モジュールおよび光ケーブル
- 10 ギガビット・イーサネットのポートを持つかあるいは 10 ギガビットのイーサネット・カードを装着した PC
- FPGA コンフィグレーションを行うためのマイクロ USB ケーブル
- DOS テスト・アプリケーション“tcpdatatest.exe” (DesignGatway 社の Web サイトよりダウンロード可)

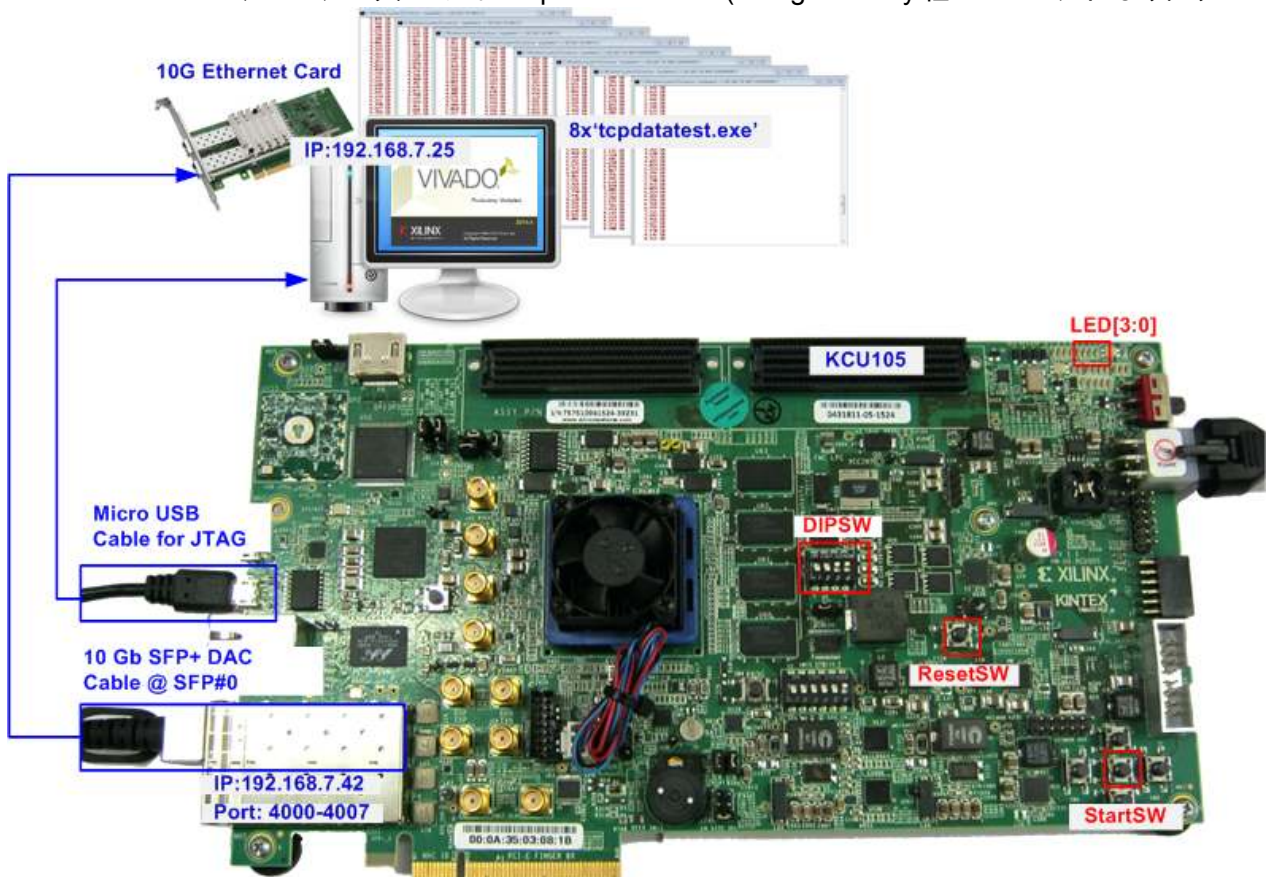


図 1: TOE10G-IP マルチセッション・リファレンス・デザインの実機評価環境

3. ハードウェアの説明

TOE10G-IP 標準リファレンス・デザインのバッファ・サイズが最高のパフォーマンスが得られるよう最大値としているのに対し、本マルチスレッドのデザインは同時8セッションの接続をサポートするために最小のバッファ容量としています。従って TOE10G-IP コアのバッファ・サイズは 4K バイトです。このデモでは DIPSW 設定により 2 セッションごとに転送方向を独立して切り替えられるようデザインされています。各 TOE10G-IP にて異なるポート番号をセットすることで、PC と FPGA 間の全セッションのデータを同時に転送できます。

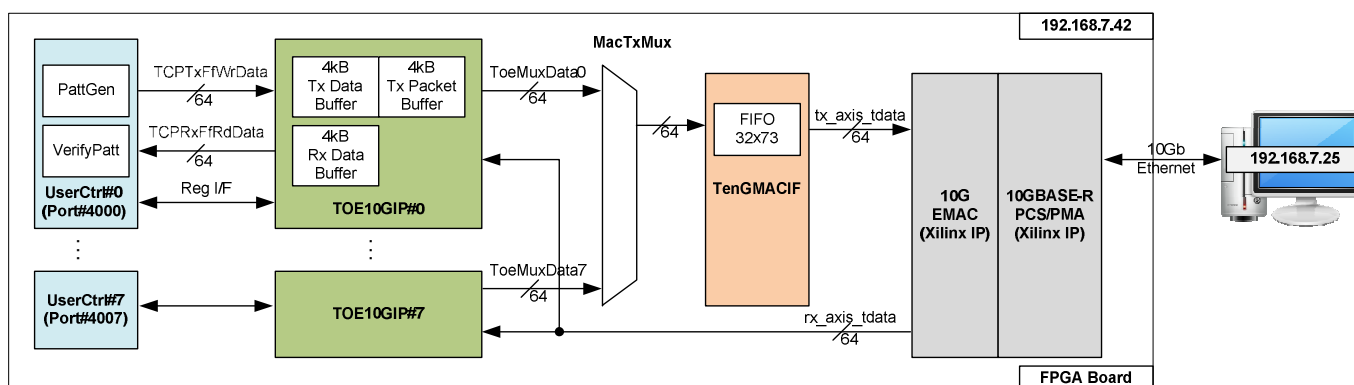


図 2: マルチセッション・リファレンス・デザインのハードウェア構造

- 10GBASE-R IP

このモジュールは Xilinx 社より無償で提供される 10 ギガビット・イーサネット PCS/PMA IP コアです。詳細については以下を参照してください。

<https://www.xilinx.com/products/intellectual-property/10gbase-r.html>

- 10GEMAC

このモジュールは Xilinx 社提供の 10Gb イーサネット MAC IP コアです。評価用ライセンスのリクエストや MAC IP コア詳細については以下を参照してください。

<https://www.xilinx.com/products/intellectual-property/do-di-10gemac.html>

- TenGMACIF

Vivado2015 版の Xilinx 10GEMAC において送信インターフェイスは Vivado2014 版から更新されており、"tx_axis_tready"信号はパケット転送中に'0'ネゲートされることがあります。TOE10G-IP コアはパケット転送中の"tx_axis_tready"ネゲートを許容していません。TOE10G-IP コアはデータ転送前のみこのフラグをチェックします。

少量のバッファ(FIFO32x73)を内蔵する TenGMACIF モジュールは 10GEMAC が送信データを受領できない場合 IP コアからのデータを一時的に保持します。TenGMACIF モジュールは内部 FIFO の空き容量が十分ではない場合次のパケット送信前にレディ信号をネゲートし送信を抑制します。

このモジュールは Vivado2014.1 またはそれより以前の 10GEMAC-IP コアを使う場合は不要です。

- MacTxMux

このモジュールは 8 個の TOE10G-IP コアからの送信信号を選択し 10GEMAC へ出力するアービターです。現在送信中のパケットが完了し他チャンネルが別のパケット送信を要求している場合に選択チャンネルが変更されます。

EMAC の受信側インターフェイスにはデマルチプレクサ等はありません。EMAC からの受信信号は全ての TOE10G-IP コアへ接続され、各コアにて受信パケットをフィルタリングします。このため受信パケットのヘッダ情報に合致するコアのみがパケットを処理します。

- TOE10G-IP

本リファレンス・デザインの RAM サイズは 4K バイトです。DesignGateway 社内で実機確認したところ、4K バイトのデータ・バッファとした場合 1 チャンネルの転送速度は約 200~250MB/sec です。このパフォーマンスは RAM サイズを最大の 64K バイトとした標準デザインで場合のパフォーマンスと比較すると約 1/4 となります。各コアの RAM サイズを 4K バイトとして全 8 セッションでのデータ転送を同時に実行した場合、合計のパフォーマンスは標準デザインのパフォーマンスとほぼ等しくなります。

- UserCtrl

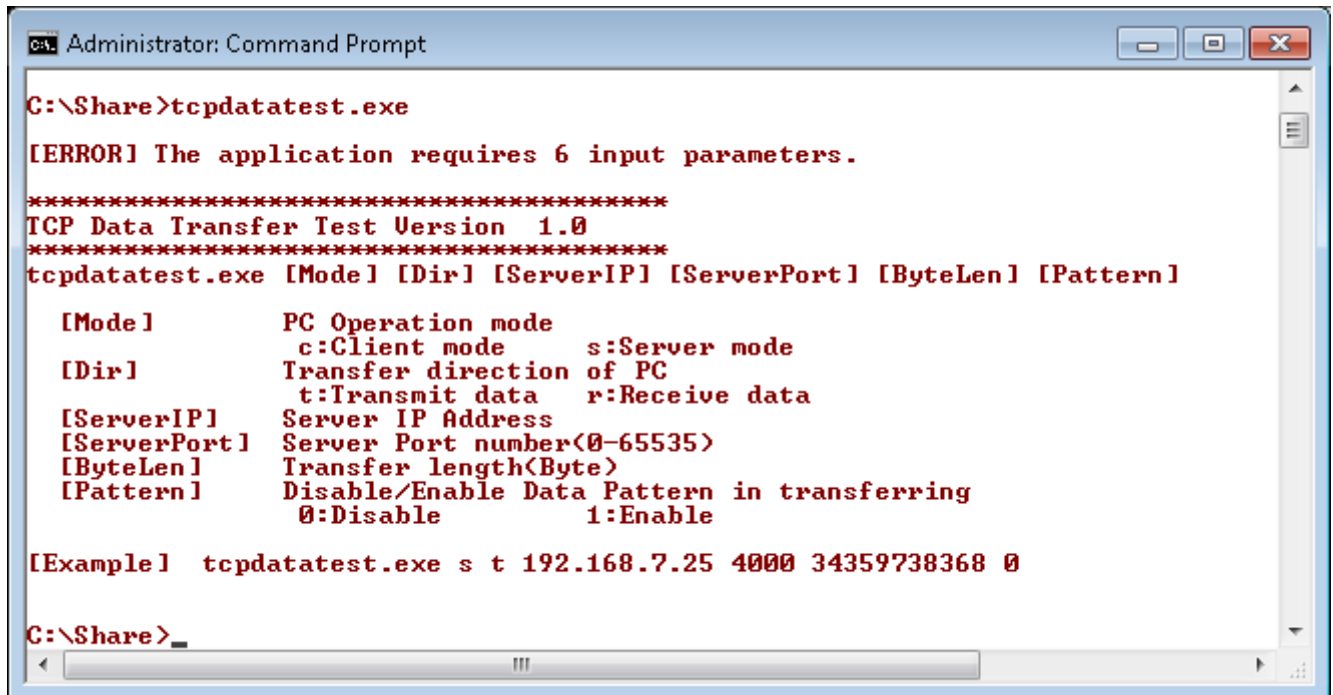
本モジュールの HDL コードは標準リファレンス・デザインのユーザ回路と同一です。詳細は以下を参照してください。
http://www.dgway.com/products/IP/TOE10G-IP/dg_toe10gip_refdesign_xilinx_jp.pdf.

ポート番号を除いた全てのパラメータは標準デザインと同じです。マルチセッションをサポートするため、各インスタンスした IP コアでのポート番号は異なります。最初のポート番号は 4000 で、次のポートはそれよりインクリメントします (IP#0 が Port#4000、IP#1 が Port#4001 で、IP#7 が Port#4007 となります)。

各 IP コアの RAM 容量が 4K バイトのため、TOE10G-IP の WIN レジスタは 1K バイト・サイズにセットされ、ユーザ回路が 1K バイト・データをベリファイするごとに IP コアがウインドウ・アップデート・パケットを PC へ送ります。このウインドウ・アップデート・パケットは IP コア内の受信データ・バッファの空き容量情報を通知するために使われます。よって PC 側のテスト・アプリケーションでは十分な空き容量が回復すると次のパケットを送ることができます。

UserCtrl の転送方向は DIPSW で設定されます。DIPSW のうち 4 ビットが 8 チャンネルの方向を選択するため、1 ビット当たり 2 セッションの転送方向が設定されることとなります。

4. PC 側テスト・アプリケーション



```

Administrator: Command Prompt

C:\Share>tcpdatatest.exe

[ERROR] The application requires 6 input parameters.

*****
TCP Data Transfer Test Version 1.0
*****
tcpdatatest.exe [Mode] [Dir] [ServerIP] [ServerPort] [ByteLen] [Pattern]

[Mode]          PC Operation mode
                 c:Client mode      s:Server mode
[Dir]           Transfer direction of PC
                 t:Transmit data   r:Receive data
[ServerIP]     Server IP Address
[ServerPort]   Server Port number(0-65535)
[ByteLen]      Transfer length(Byte)
[Pattern]      Disable/Enable Data Pattern in transferring
                 0:Disable         1:Enable

[Example] tcpdatatest.exe s t 192.168.7.25 4000 34359738368 0

C:\Share>_

```

図 3: テスト・アプリケーション(ヘルプ画面)

”tcpdatatest”はサーバー/クライアントの両モードにてTCPデータを送受信するPC側テスト・アプリケーションで、DOSプロンプト上で動作します。アプリケーション起動時の引数で転送方向やモードを指定します。以下の6引数を必要とします。

- 1) Mode: c – PC側がクライアントでFPGA側がサーバーで起動する
s – PC側がサーバーでFPGA側がクライアントで起動する
- 2) Dir: t – PCからFPGAへとデータを送信する
r – PCがFPGAからデータを受信する
- 3) ServerIP: PCがクライアントで起動する場合のFPGA側のIPアドレス
本リファレンス・デザインで有効な値は192.168.7.42である
- 4) ServerPort: PCがクライアントで起動する場合のFPGA側のポート番号
本リファレンス・デザインで有効な値は4000~4007である
- 5) ByteLen: 総転送サイズをバイト単位で指定。このパラメータは送信モードのみ有効で受信モードでは無視される。受信モードで送信相手から接続がクローズされる場合、アプリケーション側では受信データ・サイズや受信動作完了のタイミングは判らない。
- 6) Pattern: 0 – 送信モードではダミーのデータ・パターンが生成される/受信モードではベリファイは行わない
1 – 送信モードではインクリメンタル・パターンが生成される/受信モードではベリファイを行う

4.1 データ送信モード

テスト・アプリケーションをデータ送信モードで動作させた場合は以下のシーケンスとなります。

- 1) 送信バッファに 1M バイトのメモリ空間をアロケートします。
- 2) ソケットを生成し送信バッファのプロパティを設定します。
- 3) 新規コネクションを生成します。
 - a) クライアント・モードの場合、ユーザが指示した IP アドレスとポート番号でサーバーと新規コネクションを生成します。
- 4) テスト・パターン生成が指示された場合はインクリメンタル・テスト・パターンを送信バッファ内に作成します。ダミー・パターンの場合このステップはパスします。
- 5) データを送信し残り転送サイズを減算します。
- 6) 毎秒ごとに総送信サイズを表示します。
- 7) 残り転送サイズがゼロになるまでステップ 4)~6)を繰り返します。
- 8) ソケットをクローズし総転送サイズとパフォーマンス結果を表示します。

4.2 データ受信モード

テスト・アプリケーションをデータ受信モードで動作させた場合は以下のシーケンスとなります。

- 1) 受信バッファに 1M バイトのメモリ空間をアロケートします。
- 2) ソケットを生成し受信バッファのプロパティを設定します。
- 3) 送信モードのステップ 3)と同じ処理を行います。
- 4) 受信バッファからデータを読み出し総受信データ・サイズを加算します。
- 5) ベリファイがイネーブルとなっていた場合、データはインクリメンタル・パターンと比較され合致していない場合エラー・メッセージを表示します。ベリファイが指定されていない場合このステップはパスします。
- 6) 毎秒ごとに総受信サイズを表示します。
- 7) コネクションがクローズされるまでステップ 4)~6)を繰り返します。
- 8) ソケットをクローズし総転送サイズとパフォーマンス結果を表示します。

5. 更新履歴

リビジョン	日付	説明
1.0	4-Nov-16	Initial Release
1.0J	2017/12/04	日本語版の初版翻訳

Copyright: 2016 Design Gateway Co,Ltd.