

前編(第20回, 2011年1月号, pp.159-166)に引き続き, Xilinx社のSpartan-6 LXTシリーズ内蔵のPCI Expressエンドポイント・ブロックの使い方について解説する。今回はユーザ・ロジック部分の設計事例について解説する。また最後にSpartan-6とVirtex-5に内蔵のPCI Expressエンドポイント・ブロックを比較してみる。(編集部)

Spartan-6の内蔵エンドポイントは, CoreGeneratorで作ったLogiCORE IPというラッパを被せて使います。内蔵エンドポイントとLogiCORE IPは, コンフィグレーション関係のパケットなら自動的に処理してくれますが, ユーザ・アプリケーションの通信には全く関与しません。そのため実際の設計では, FPGAのユーザ・リソースを使ってパケットを解釈したり組み立てたりする回路を作らなければなりません。今回はそのようなユーザ・ロジック回路を作ります。

## 1. PCI Expressの送受信機能作成

LogiCORE IPは, メモリやI/O, メッセージの各パケットを受信すると, ユーザ回路にその処理を委託します。本来はコンフィグレーション以外のすべてのパケットを処理しなければなりません, 話を簡単にするために, ここではメモリ関係のパケットだけ処理することにします。パソコン上のデバイス・ドライバでIO命令を使わなければ, I/O関係のパケットは送信されませんし, メッセージ・パ

ケットは基本的に無視しても問題ありません。

したがって, ユーザ回路で処理すべき内容は以下の二つです。

- メモリ・ライト・リクエストを受け取った場合は, ユーザ・ロジックに伝える
- メモリ・リード・リクエストを受け取った場合は, ユーザ・ロジックからの応答を待って, コンプリーションを返す

この機能を持った回路を作るには, 受信TLP(トランザクション・パケット)解釈ステート・マシンと送信TLP組み立てステート・マシン, それから, コンプリーション用のデータを作るしくみが必要です。

### ● TLP受信の方法

TLPを受信する際に使う信号は, `trn_rsof_n`と`trn_reof_n`, `trn_rd [31:0]`, `trn_rsrc_rdy_n`, `trn_rdst_rdy_n`, `trn_rbar_hit_n [6:0]`です。これらの信号の基本的な使用方法を図1に示します。

リンクを通じてパケットが送られてくると, コアは`trn_rd [31:0]`に32ビット単位(以下, DWと略す)でデータ

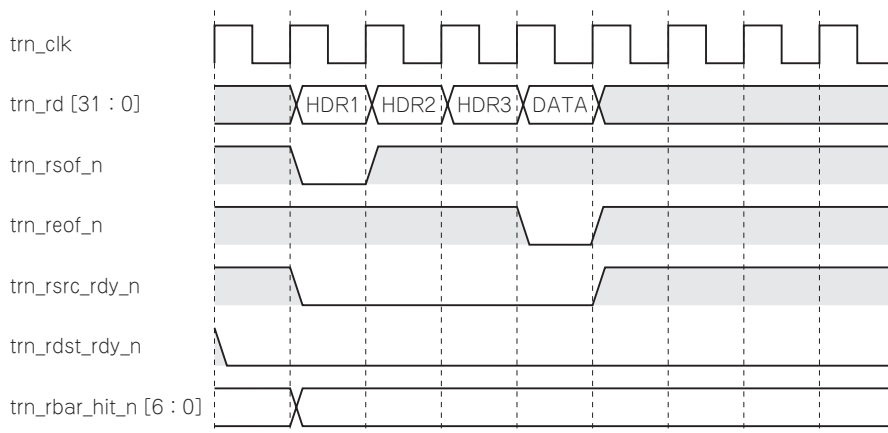


図1  
TLP受信時の信号のタイミング  
(3DWヘッダ+1DWのデータの場合)

ユーザ・ロジックが`trn_rdst_rdy_n`をアサートすると, コアはデータを送ってくる。 `trn_rd [31:0]`にはTLPがヘッダの先頭からそのまま順に現れる。先頭のDWのときには`trn_rsof_n`がアサートされ, 最後のDWのときには`trn_reof_n`がアサートされる。 `trn_rd [31:0]`に有効なデータが出続けている限り, `trn_rsrc_rdy_n`はアサートされ続ける。