■謹賀新年

あけましておめでとうございます.
皆様，今年もよろしくお願いいたします.

■ PCIe のサンプル

7 シリーズ PCIe Gen3 Integrated Block 用の PIO なサンプルコードを読む.
これまでのコアと違って，AXI4-Stream が 4 本あるのが特徴
PG023 の p.14 によると，

Completer reQuest (CQ) Interface

The interface through which all received requests from the link are delivered to the user application.

Completer Completion (CC) Interface

The interface through which completions generated by the user application responses to the completer requests are transmitted. You can process all Non-Posted transactions as split transactions. That is, it can continue to accept new requests on the Requester Completion interface while sending a completion for a request.

Requester reQuest (RQ) Interface

The interface through which the user application generates requests to remote PCIe devices.

Requester Completion (RC) Interface

The interface through which the completions received from the link in response to your requests are presented to the user application.
とのこと.

パラメタで構成がいろいろ変わるので,

```
C_DATA_WIDTH = 256
AXISTEN_IF_RQ_ALIGNMENT_MODE == "FALSE"
AXISTEN_IF_CC_ALIGNMENT_MODE == "FALSE"
AXISTEN_IF_CQ_ALIGNMENT_MODE == "FALSE"
AXISTEN_IF_RC_ALIGNMENT_MODE == "FALSE"
```

に限ってみてみる.
256bit 幅なので，4DWORD，3DWORD ヘッダの別がなくて，実は状態遷移すっきり ... か？

■ PCIe のサンプル・受信側

実装コードは PIO_RX_ENGINE.v.

PIO_RX_RST_STATE

待機ステート.

```
m_axis_cq_tready <= #TCQ 1'b1;
m_axis_rc_tready <= #TCQ 1'b1;
```

sop がアサートされた (新しいパケットが到着した) ら，到着パケットのフォーマット (m_axis_cq_tdata[78:75]) に応じて次の状態に遷移.
ただし，256bit 幅 AXI4-Stream なので，これが最後のパケットの場合には遷移しないことも.
遷移先は↓な感じ.

```
PIO_RX_MEM_RD_FMT_TYPE → PIO_RX_WAIT_STATE へ
PIO_RX_MEM_WR_FMT_TYPE → (m_axis_cq_tdata[74:64] == 11'h001) || (m_axis_cq_tdata[74:64] == 11'h002)
の場合 (= ペイロードが 2 以下の場合 ) に PIO_RX_WAIT_STATE へ．でなければ，PIO_RX_RST_STATE へ ( 強制的に後
続のパケットを読み捨てる，と思われる ).
PIO_RX_IO_RD_FMT_TYPE → PIO_RX_WAIT_STATE
PIO_RX_IO_WR_FMT_TYPE → (m_axis_cq_tdata[74:64] == 11'h001) || (m_axis_cq_tdata[74:64] == 11'h002)
の場合 (= ペイロードが 2 以下の場合 ) に，PIO_RX_WAIT_STATE へ．でなければ，PIO_RX_RST_STATE へ．
PIO_RX_ATOP_FAA_FMT_TYPE，PIO_RX_ATOP_UCS_FMT_TYPE，PIO_RX_ATOP_CAS_FMT_TYPE → PIO_RX_WAIT_STATE
PIO_RX_MEM_LK_RD_FMT_TYPE → PIO_RX_WAIT_STATE
PIO_RX_MSG_FMT_TYPE → PIO_RX_RST_STATE
PIO_RX_MSG_VD_FMT_TYPE → PIO_RX_RST_STATE
PIO_RX_MSG_ATS_FMT_TYPE → PIO_RX_RST_STATE
```

PIO_RX_WAIT_STATE

PIO_RX_RST から遷移するステート.
フォーマットに応じて "wr_busy でない " か compl_done なら PIO_RX_RST_STATE に.
PIO なサンプルだから，1 発受け取ったらおしまいってことでいいのかな.

PIO_RX_DATA

ALIGNMENT_MODE が FALSE なので，実はここには遷移しない.
ALIGNMENT_MODE が TRUE のときに, (RST_STATE で書けないので) データを内部に書き出す.

■ PCIe のサンプル・送信側

実装コードは PIO_TX_ENGINE.v.
FPGA 側から PC にアクセスすることはなく PC からのリクエストに応じてデータを送信する.
もし, FPGA から主体的にデータをリクエストを出すのなら, こちらの RST_STATE を参考にしつつ仕掛けをすればよい，と思われる.
送信処理は, AXI4-Stream でデータを適切にセット, s_axis_cc_tready のアサートで終了, というのが大筋.
こちらもペイロード長の最大が

PIO_TX_RST_STATE

待機状態. ↓な感じで次に遷移.

```
if(req_compl) begin
```

```
      state <= #TCQ PIO_TX_COMPL_C1;
   else if (req_compl_wd) begin
      state <= #TCQ PIO_TX_COMPL_WD_C1;
   end else if (req_compl_ur) begin
      state <= #TCQ PIO_TX_CPL_UR_C1;
   end else if (gen_transaction) begin
      state <= #TCQ PIO_TX_MRD_C1;
   end
```

req_compl，req_compl_wd，req_compl_ur は PIO_RX_ENGINE でセットされる．
受信パケットが

```
PIO_RX_MEM_RD_FMT_TYPE
 req_compl_wd あるいは req_compl_ur がセットされる
PIO_RX_IO_RD_FMT_TYPE
 req_compl_wd あるいは req_compl_ur がセットされる
PIO_RX_IO_WR_FMT_TYPE
 req_compl がセットされる
PIO_RX_ATOP_FAA_FMT_TYPE，PIO_RX_ATOP_UCS_FMT_TYPE，PIO_RX_ATOP_CAS_FMT_TYPE
 req_compl あるいは req_compl_ur がセットされる
PIO_RX_MEM_LK_RD_FMT_TYPE
 req_compl と，req_compl_wd あるいは req_compl_ur がセットされる
```

gen_transaction は，EP_MEM::PIO_EP_MEM_ACCESS(ソースは PIO_EP_MEM_ACCESS.v) でセットされる．

```
   if(wr_data[31:0] == 32'hAAAA_BBBB && !trn_sent)
      gen_transaction <= #TCQ 1'b1;
   else
      gen_transaction <= #TCQ 1'b0;
```

PIO_TX_COMPL_C1

コメントは，"Completion Without Payload - Alignment doesnt matter Sent in a Single Beat When Interface Width is 256 bit."
s_axis_cc_tready がアサートされたら，PIO_TX_RST_STATE に遷移.

PIO_TX_COMPL_WD_C1

コメントは，"Completion With Payload. Possible Scenario's Payload can be 1 DW or 2 DW. Alignment can be either of Dword aligned or address aligned."
ペイロードが 1 のときは, s_axis_cc_tready がアサートされたら, PIO_TX_RST_STATE に遷移．でなければ，PIO_TX_COMPL_WD_2DW へ.

PIO_TX_COMPL_PYLD

コメントは，"Completion with 1DW Payload in Address Aligned mode"
s_axis_cc_tready がアサートされたら，PIO_TX_RST_STATE に遷移.

PIO_TX_CPL_UR_C1

コメントは，"Completions with UR - Alignement mode matters here"

s_axis_cc_tready がアサートされたら，PIO_TX_RST_STATE に遷移.

PIO_TX_MRD_C1

コメントは，"Memory Read Transaction - Alignment Doesnt Matter"
s_axis_cc_tready がアサートされたら，PIO_TX_RST_STATE に遷移.

PIO_TX_COMPL_WD_2DW

コメントは，"Completion with 2DW Payload in DWord Aligned mode. Requires 2 states to get the 2DW Payload."

PIO_TX_COMPL_WD_2DW_ADDR_ALGN_C1

ALIGNMENT_MODE が FALSE なので，実はここには遷移しない.