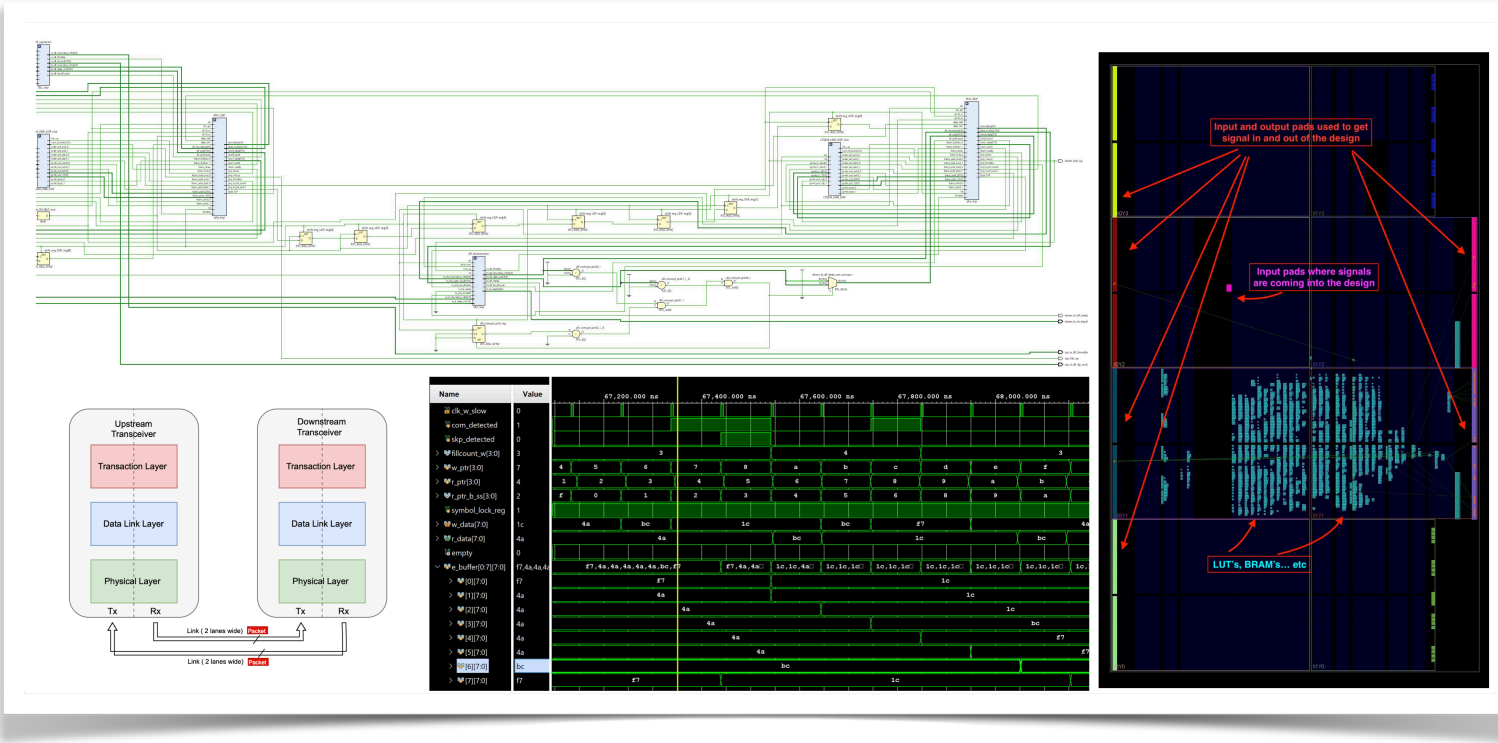
PCIe RTL Design
Simulation and FPGA PnR



Tools: Xilinx Vivado
FPGA board: Nexys A7 FPGA

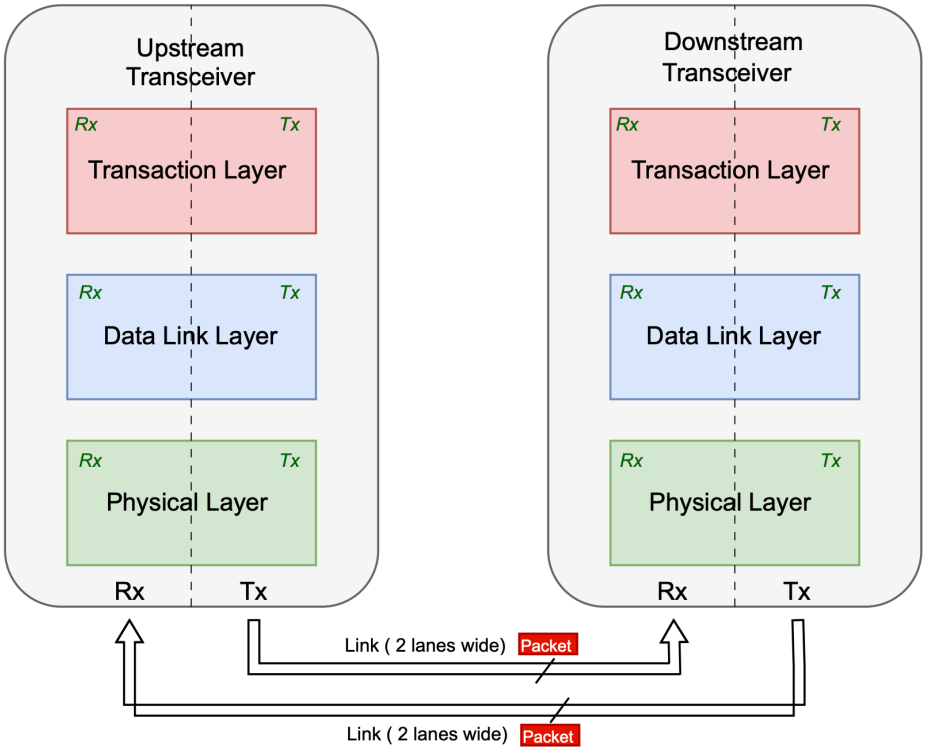
RTL Design of PCIe Protocol

Peripheral Component Interconnect express is a high speed serial bus standard that is

1. Point-to-point (Device A communicate with B over PCIe Data Link). This means interconnect capacitances are very limited, and so you can go to really high speed!
2. Bi-directional (Both A and B can transmit and receive data = transceivers using Dual Simplex lanes)
3. Scalable to accommodate varying bandwidth needs (One Data Link can have 1,2,4,8,16, 32 lanes)
4. Backwards compatible with previous PCIe versions. (You can run PCIe-2 GPU on a PCIe-4 motherboard)
5. Widely adopted across many markets.

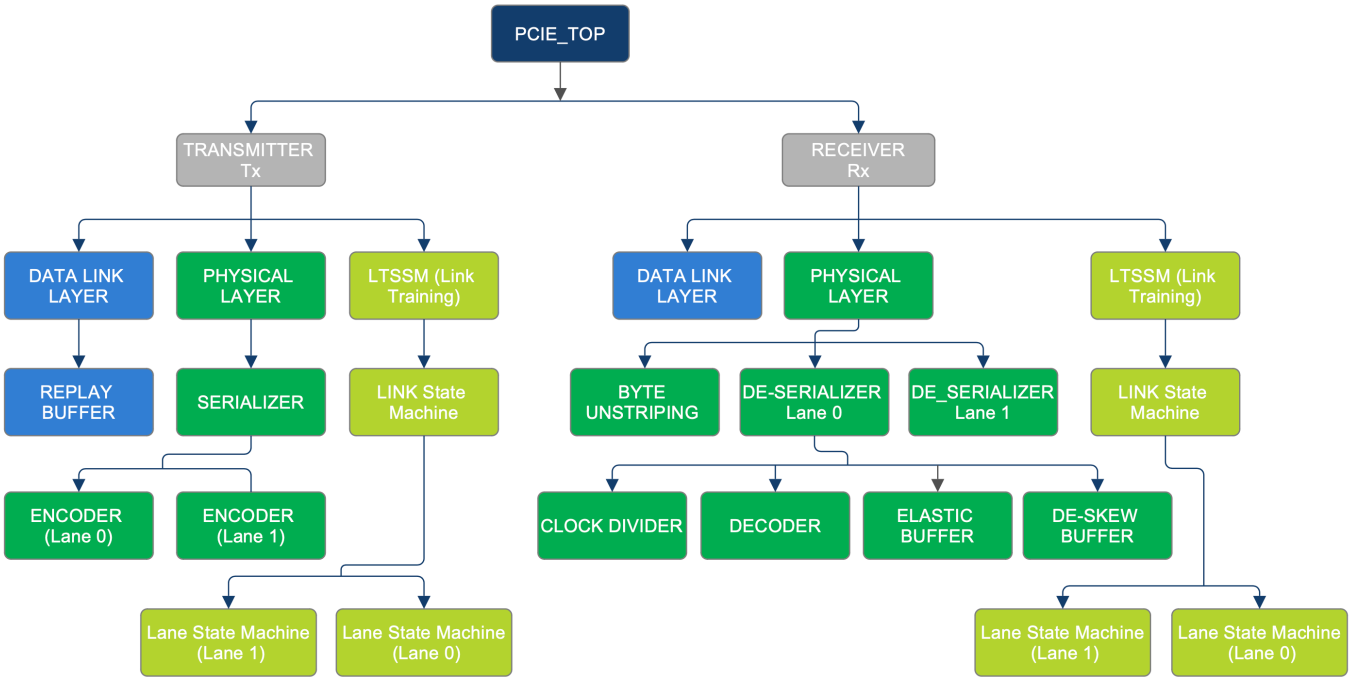
Some specs:

- Ideally for PCIe Gen1, the receiver and transmitter clock frequencies are around 2.5 GHz. Tx local clock and Rx local clock can differ by 600 ppm. (Max tolerance of +/- 300 ppm on data rate on each side Tx/Rx = 600 ppm difference in frequency in worse case scenario). This PCIe uses 10MHz and 9.8 MHz local clocks for Upstream and Downstream designs respectively.
- Supports 2 lane configuration.



PCIe Hierarchy and Sub-designs/ Sub-blocks:

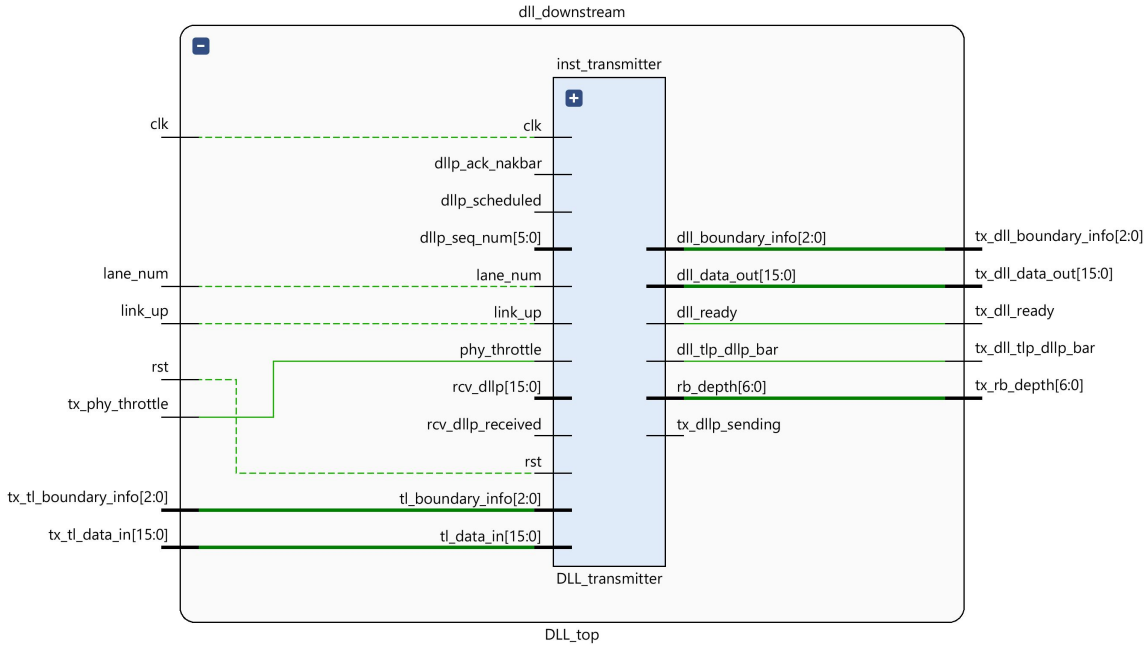
This PCIe design encompasses Data Link Layers (DLL's) and Physical Layers (PHY's) - Transaction Layers are excluded in the RTL design, with their functionality compensated for in the test-bench. This approach allows us to simulate and verify the desired transaction layer behaviors without impacting the RTL.



I. Data Link Layer (Tx_DLL, Rx_DLL):

The data link layer conveys information between Transaction Layer and Physical Layer (PHY). It maintains packet integrity and generates DLL packet when communicating. Before the TLP (Transaction Layer Packet) is transmitted to PHY Layer, DLL will attach a unique sequence number as well as an LCRC (Link Cyclic Redundancy Code) to the packet. The DLL on the receiver side will check the correctness of the sequence number and LCRC. If they the TLP being transmitted is good, and Ack DLLP with the sequence number will be sent back to the transmitter to acknowledge that it is good; otherwise, a Nak DLLP will be sent instead and the transmitter will re-send the TLP(s) again from the retry buffer. This PCIe will assume CRC errors never occur, but Tx_DLL will deliberately send a curropted sequence number so as to make Rx_DLL send a Nak back.

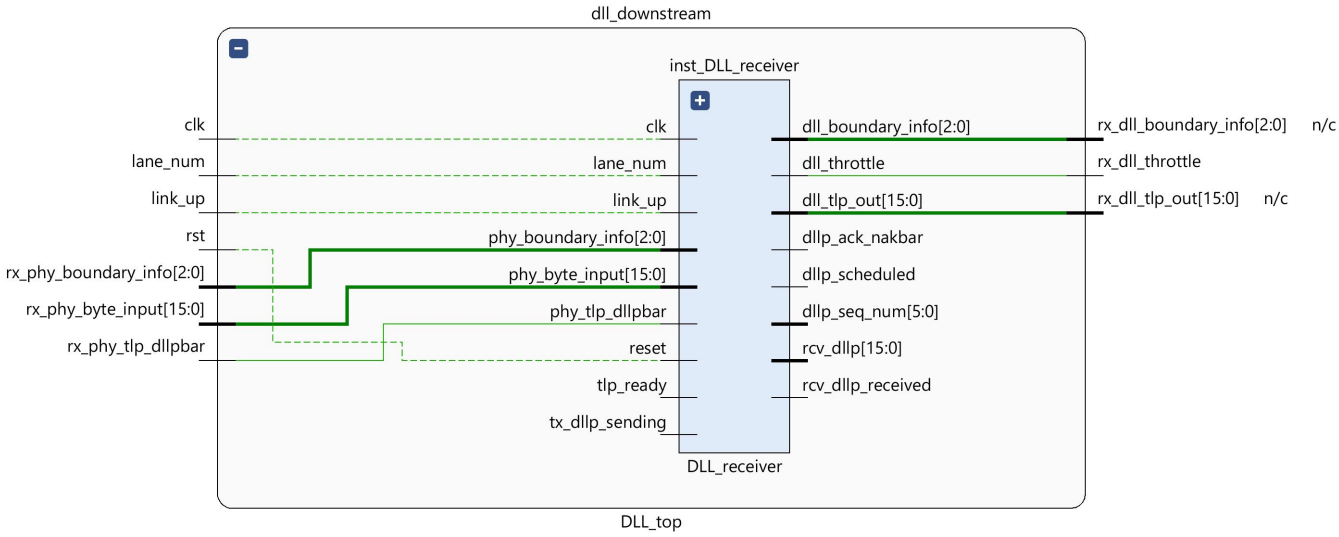
Tx_DLL interface (DLL_transmitter):



DLL_transmitter Interface Signals:

Name	Direction	Interface with	Description
clk	input		System clock
dllp_ack_nakbar	input	Rx_DLL	Used to determine sending either ack/nak. ack/nak is a 2 byte item. h'00 for ack .. h'10 for nak
dllp_scheduled	input	Rx_DLL	Scheduled flag for a dllp, for the transmitter to send the DLLP
dllp_seq_num[5:0]	input	Rx_DLL	6 bit sequence number that should be included in the DLLP
lane_num	input	Tx_PHY	0 = one lane link, 1 = two lane link
link_up	input	Tx_PHY	0 = The link is not ready, 1 = The link is ready for normal operation
phy_throttle	input	Tx_PHY	Handshake - To indicate if phy is not able to receive, DLL will not send the DLLP
rcv_dllp[15:0]	input	Rx_DLL	Storage for a DLLP (2 bytes)
rcv_dllp_received	input	Rx_DLL	To indicate if a DLLP is received by rx to extract its information.
rst	input		Reset signal
tl_boundary_info[2:0]	input	(Test_bench)	3 bit boundary info encoding for PKT_IDLE = not sending anything PKT_START = starting 16-bit word PKT_IN = 16-bit word somewhere in the middle of the TLP PKT_EN = ending 16-bit word Or else, starting and ending of a single 16-bit word TLP
tl_data_in[15:0]	input	(Test_bench)	Transaction Layer packet
dll_boundary_info [2:0]	output	Tx_PHY	3 bit boundary info encoding for PKT_IDLE = not sending anything PKT_START = starting 16-bit word PKT_IN = 16-bit word somewhere in the middle of the TLP PKT_EN = ending 16-bit word Or else, starting and ending of a single 16-bit word TLP
dll_data_out[15:0]	output	Tx_PHY	DLLP Packet
dll_ready	output	Tx_PHY	DLLP packet ready to be sent out
dll_tlp_dllp_bar	output	Tx_PHY	A flag indicating the byte is a tlp or a dllp: dllp:0 tlp:1
rb_depth[6:0]	output	Tx_PHY	Retry buffer address pointer
tx_dllp_sending	output	Rx_DLL	To indicate if tx_DLL will send the DLLP

Rx_DLL interface (DLL_receiver):



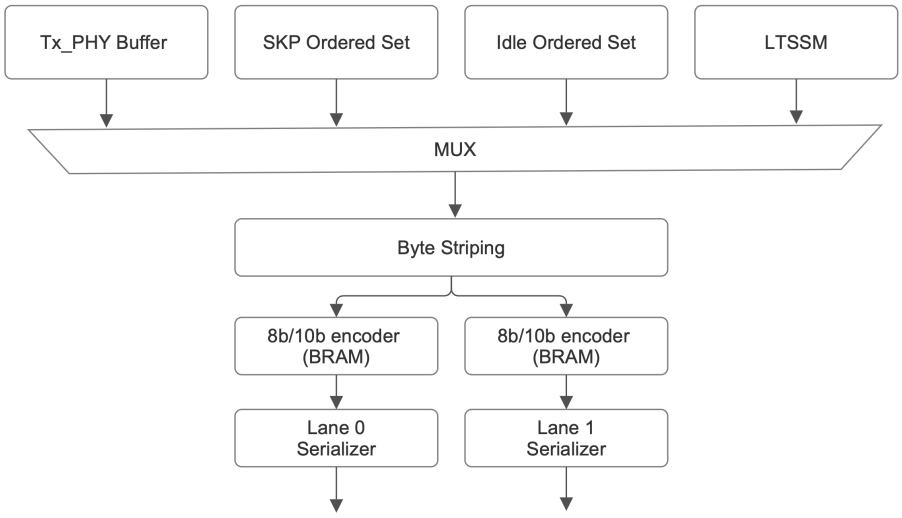
DLL_receiver Interface Signals:

Name	Direction	Interface with	Description
clk	input	—	Clock
lane_num	input	Tx_PHY	0 = one lane link, 1 = two lane link
link_up	input	Tx_PHY	0 = The link is not ready, 1 = The link is ready for normal operation
phy_boundary_info[2:0]	input	Rx_PHY	3 bit boundary info encoding for PKT_IDLE = not sending anything PKT_START = starting 16-bit word PKT_IN = 16-bit word somewhere in the middle of the packet PKT_EN = ending 16-bit word Or else, starting and ending of a single 16-bit word packet
phy_byte_input[15:0]	input	Rx_PHY	Physical Layer packet
phy_tlp_dllpbar	Input	Tx_PHY	A flag indicating the byte is a tlp or a dllp: dllp:0 tlp:1
reset	input	—	Reset signal
tlp_ready	input	Rx_PHY	Ready handshake signal from the transaction layer
tx_dllp_sending	input	Rx_DLL	To indicate if tx_DLL will send the DLLP
dll_boundary_info [2:0]	output	—	3 bit boundary info encoding for PKT_IDLE = not sending anything PKT_START = starting 16-bit word PKT_IN = 16-bit word somewhere in the middle of the packet PKT_EN = ending 16-bit word Or else, starting and ending of a single 16-bit word packet
dll_throttle	output	Rx_PHY	Handshake - To indicate that receiver is not able to receive, DLL will not send the DLLP
dll_tlp_out[15:0]	output		DLLP Packet to transaction layer
dllp_ack_nakbar	output	Rx_DLL	Used to determine sending either ack/nak. ack/nak is a 2 byte item. h'00 for ack .. h'10 for nak
dllp_scheduled	output	Rx_DLL	Scheduled flag for a dllp, for the transmitter to send the DLLP
dllp_seq_num[5:0]	output	Rx_DLL	6 bit sequence number that should be included in the DLLP
rcv_dllp[15:0]	output	Rx_DLL	Storage for a DLLP (2 bytes), will be forward to the transmitter to process
rcv_dllp_received	output	Rx_DLL	To indicate if a DLLP is received by rx to extract its information.

Physical Layer (Tx_PHY, Rx_PHY):

This layer connects to the physical PCIe link on one side and interfaces to the Data Link Layer (DLL) on the other side. The transmitter physical layer (Tx_PHY) processes outbound packets before they get transmitted to PCIe physical link, and the receiver physical layer (Rx_PHY) processes inbound packets received from the physical PCIe link.

Tx_PHY Design:



- **Tx_PHY_Buffer** is for data striping - DLL will send data to PHY in parallel and PHY will gather it in this buffer and will then further transmit it using byte striping (data is spread/striped across the available lanes.. in this case lane_0 & lane_1)

- **SKP Ordered Set (SOS)** is a 4 symbols set (COM, SKP, SKP, SKP): Once transmission begins on the link, SOS's are transmitted at regular intervals (Depending on ppm difference between PCIe upstream port and downstream local clock frequencies… in this case local clocks are 10MHz & 9.8 MHz). SOS's are also scheduled to be transmitted after the end of a DLLP or TLP, any time when idle data is being sent, or after a Training Sequence (TS1) or no data is being sent from LTSSM

- **Idle Ordered Set** is sent when Tx_PHY Buffer is empty or LTSSM has no data to send.

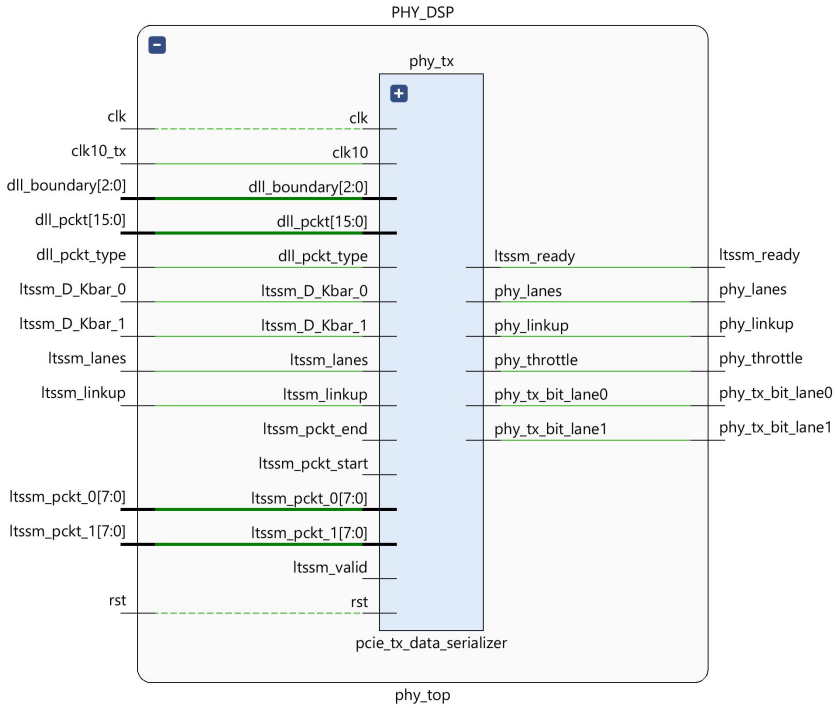
- **MUX** is select between transmitting DLLP/TLP from Tx_PHY Buffer, Idle Ordered Sets, Training Sequences TS1/TS2 Ordered Sets, or SKP Ordered Sets.

- **8b/10b encoder** is using BRAMs for 8b/10b mapping in this case.

- **Lane Serializer** is to transmit byte striped data on the 2 lanes.

- **LTSSM** (Link Training and Status Machine) is for the link up process between transmitter and receiver physical layers by achieving bit/ symbol lock using, ensuring synchronization and reliability. The link training process consists Detect, Polling, Configuration, and L0 states. The training sequences (TS1/ TS2) will be transmitted in Polling state to achieve bit/ symbol lock.

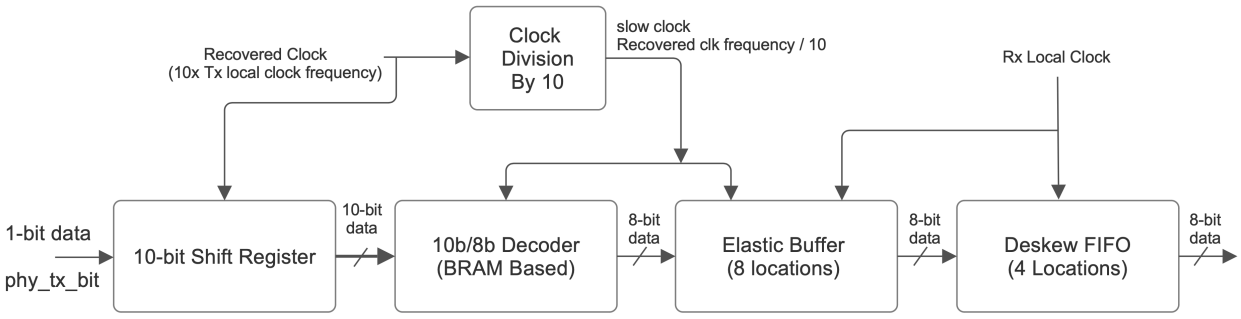
Tx_PHY Interface (phy_tx):



Phy_tx Interface Signals:

Name	Direction	Interface with	Description
clk	input	—	Local clock
clk10	input	—	10x frequency of local clock
dll_boundary[2:0]	input	Tx_DLL	3 bit boundary info encoding for PKT_IDLE = not sending anything PKT_START = starting 16-bit word PKT_IN = 16-bit word somewhere in the middle of the packet PKT_EN = ending 16-bit word Or else, starting and ending of a single 16-bit word packet
dll_pckt[15:0]	input	Tx_DLL	2 byte data from DLL; can be DLLP, TLP or IDLE
dll_pckt_type	input	Tx_DLL	Type of packet being sent by DLL to PHY - DLLP / TLP
ltssm_D_Kbar_0	input	LTSSM	1 bit signal to indicate data/control character
ltssm_D_Kbar_1	input	LTSSM	1 bit signal to indicate data/control character
ltssm_lanes	input	LTSSM	LTSSM indicates how many lanes are available
ltssm_linkup	input	LTSSM	LTSSM indicates link is up for transmission
ltssm_pckt_end	input	LTSSM	Flag for last 8 bit character of ordered set
ltssm_pckt_start	input	LTSSM	Flag for first 8 bit character of ordered set
ltssm_pckt_0[7:0]	input	LTSSM	1 byte data from LTSSM
ltssm_pckt_1[7:0]	input	LTSSM	1 byte data from LTSSM
ltssm_valid	input	LTSSM	1 bit signal to indicate if LTSSM is sending valid data
rst	input	—	Reset signal
ltssm_ready	output	LTSSM	Indicate to LTSSM physical layer that SKP ordered sets are being transmitted
phy_lanes	output	Tx_DLL, Rx_DLL	Indicate to DLL how many lanes are available for transmission
phy_linkup	output	Tx_DLL, Rx_DLL	Indicate to DLL that PHY layer is available for transmission
phy_throttle	output	Tx_DLL	Indicate to DLL if PHY TX buffer is full or empty
phy_tx_bit_lane0	output	Rx_PHY	Serial transmission on lane 0
phy_tx_bit_lane1	output	Rx_PHY	Serial transmission on lane 1

Rx_PHY Design & Interface:



- **10-bit Shift Register** is to shift in bits of data coming from Tx. Rx keeps checking the 10-bit symbol contained in this shift register - if it is COM (bc in hex), symbol lock is declared.
  (Note: Bit lock is needed however to set the frequency and phase of the recovered clock as well, which needs a PLL, but in this case Tx clock is carried over to Rx and so bit lock is always assumed to be achieved.)*

- **10b/8b Decoder** is using BRAMs for 10b/8b mapping in this case.

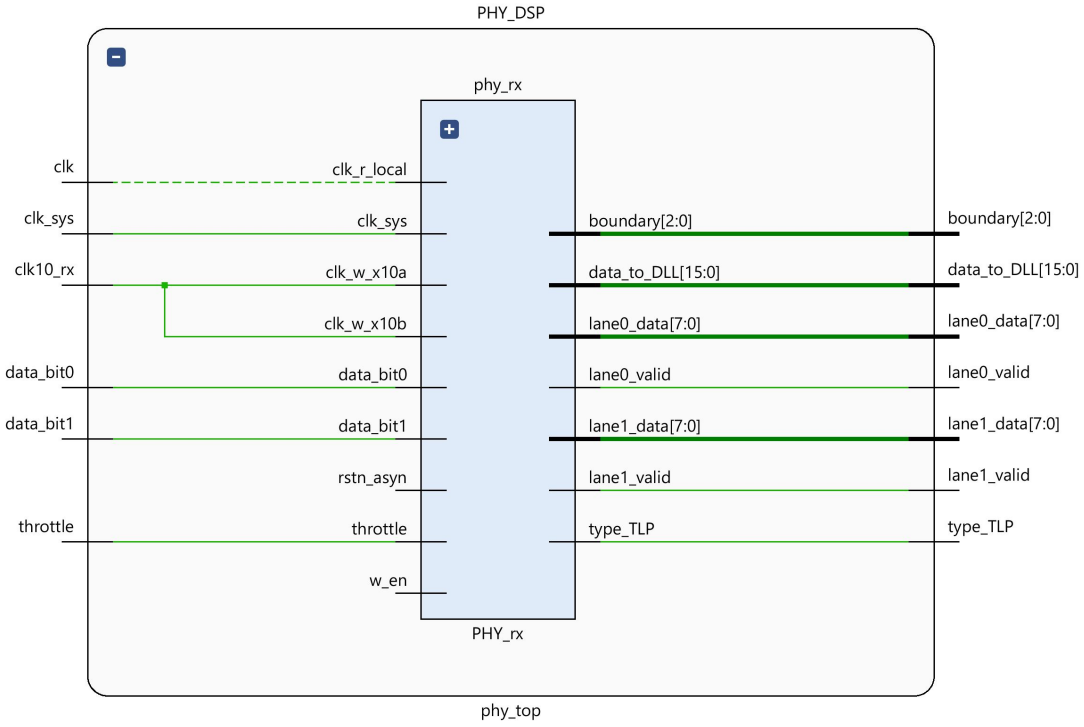
- **Elastic Buffer** is a two clock FIFO used to compensate the frequency and phase difference between Rx local clock and recovered clock from Tx. Tx will schedule Skip Ordered Sets (SOS's) regularly, which is inserted into the data stream and is sent to Rx. Elastic Buffer will decide to either use these SOS's to insert/remove dummy symbols based on whether it is about to run full/empty. This way, it compensates symbol shifts accumulated in it due to frequency differences between write clock (recovered clock f/10) and read clock (Rx local clk)

  Note 1: SOS's = 1 COM [bc in hex] + 3 SKP [1c in hex])*
  Note 2: Two clock FIFO pointers are gray coded and double flopped to avoid Clock Domain Crossing (CDC) issues!)*

- **De-skew FIFO** is used to align data streams coming from Lane_0 and Lane_1 to the same Rx local clock edge. This is because data streams on those lanes can arrive at Rx at different times due to lane-to-lane skew.

- **Clocking:** 10-bit shift register is working under high frequency recovered clock (10x frequency of Tx local clock). Decoder and write domain of Elastic Buffer are working under a divided-by-10 version of this clock generating a clock with frequency very close to Rx local clock. Every cycle of the 10x frequency clock, 1 bit of data gets shifted into the shift register. Every cycle of the slow clock, 10-bit data gets decoded into 8b data.

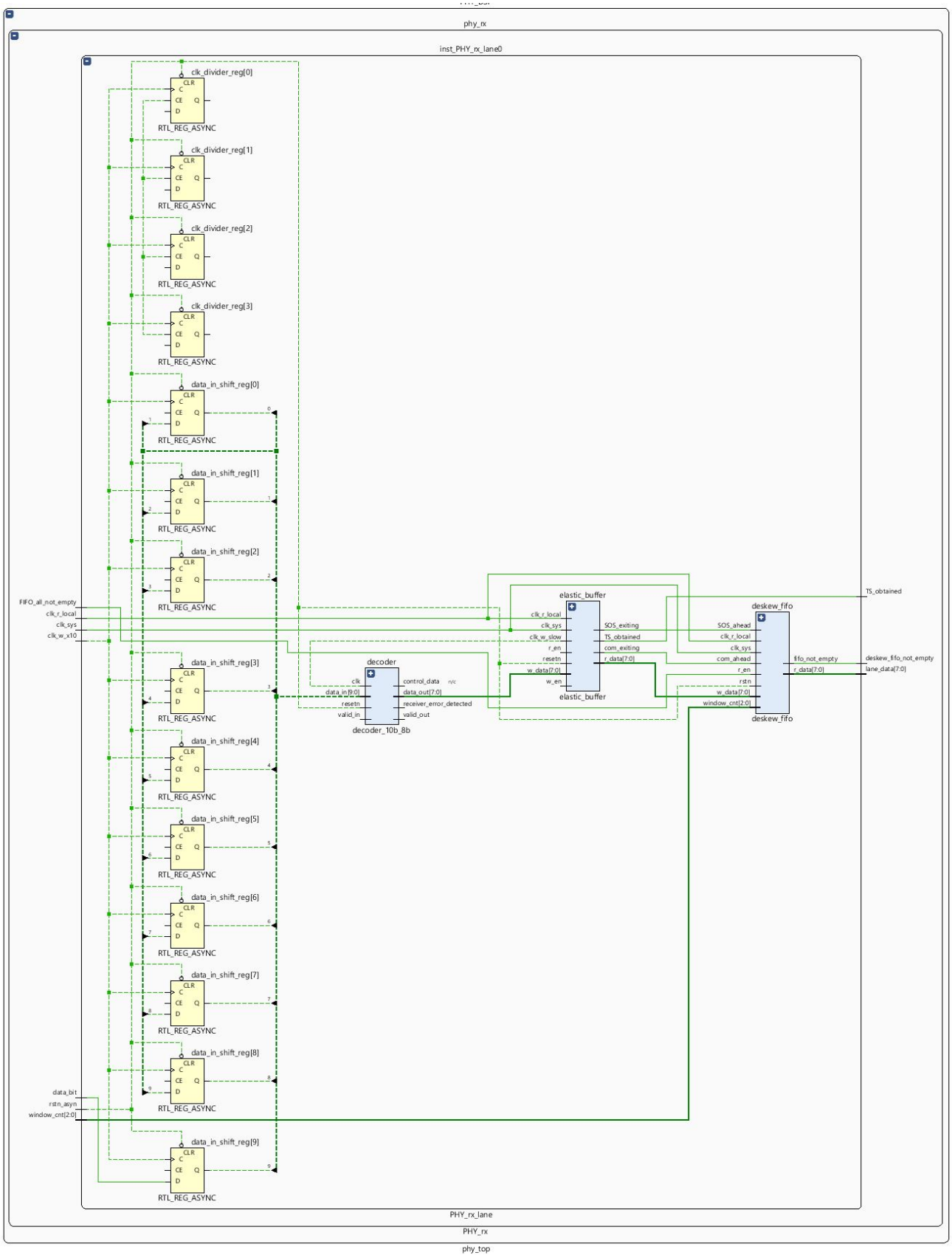
Rx_PHY Interface:



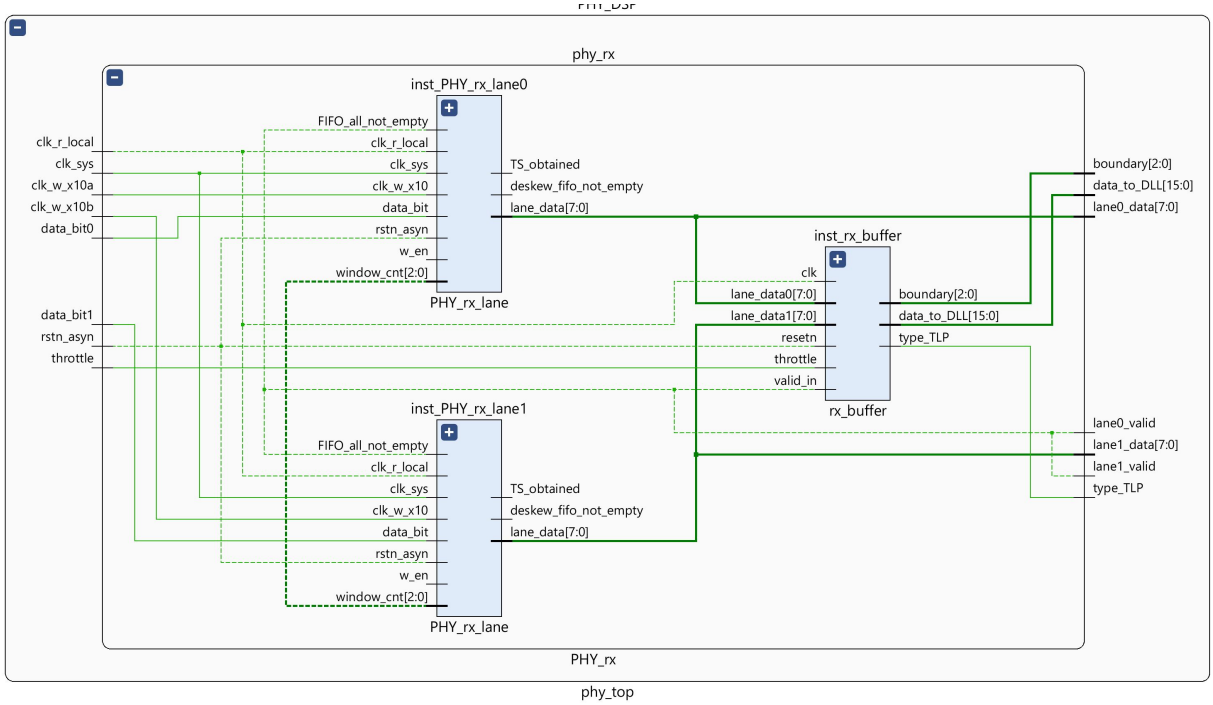
Phy_rx Interface Signals:

Name	Direction	Interface with	Description
clk_r_local	input	—	Local Rx clock
clk_sys	input	—	System clock
clk_w_x10a	input	—	Recovered clock from lane 0
clk w x10b	input	—	Recovered clock from lane 1
data_bit0	input	Tx_PHY	Data bit put on lane 0 transmission line
data bit1	input	Tx_PHY	Data bit put on lane 1 transmission line
rstn_asyn	input		Low asynchronous reset signal
throttle	input	Rx_DLL	Throttling signal given by DLL when Transaction Layer cannot receive.
w_en	input	—	Indidcates that there's valid transmission going on on physical lines
boundary[2:0]	output	Rx_DLL	Boundary info for corresponding 2 byte data
data_to_DLL[15:0]	output	Rx_DLL	Grouped 2 byte data sent to DLL
lane0_data [7:0]	output	LTSSM	1 byte data from deskew FIFO of lane 0, sent to LTSSM
lane0_valid	output	LTSSM	validity of lane 0 data
lane1_data[7:0]	output	LTSSM	1 byte data from deskew FIFO of lane 1, sent to LTSSM
lane1_valid	output	LTSSM	validity of lane 1 data
type_TLP	output	Rx_DLL	Indicate packet type of current packet to DLL

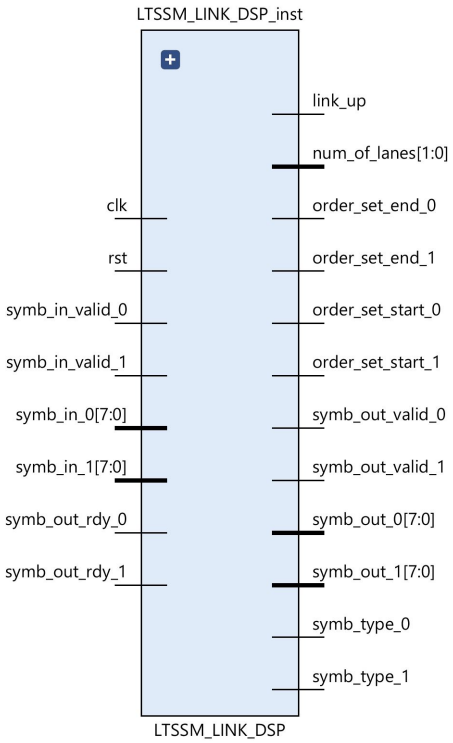
Rx_PHY design per lane:



Rx_PHY Data Deserializer: Un-striping buffer used to group 2-byte data (1 byte from each lane) to DLL.



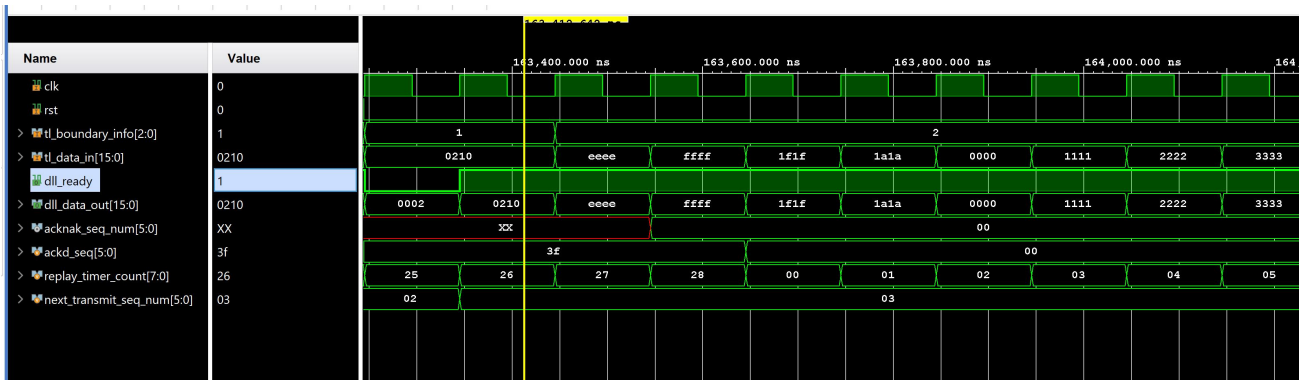
LTSSM Interface & Signals:



Name	Tupy	Interface	Description
clk	input		Clock Input
rst	input		Reset for the State Machine
symb_in_valid_0	input	Rx_PHY	Valid symbol in from physical Rx lane 0
symb_in_valid_1	input	Rx_PHY	Valid symbol in from physical Rx lane 1
symb_in_0[7:0]	input	Rx_PHY	Symbol in from physical Rx lane 0
symb_in_1[7:0]	input	Rx_PHY	Symbol in from physical Rx lane 1
symb_out_rdy_0	input	Tx_PHY	Physical Rx is ready to consume symbol out generated by LTSSM lane 0
symb_out_rdy_1	input	Tx_PHY	Physical Rx is ready to consume symbol out generated by LTSSM lane 1
link_up	output	Tx_PHY	Link up status
num_of_lanes[1:0]	output	Tx_PHY	Number lanes that can be used by Data link layer
order_set_end_0	output	Tx_PHY	Lane 0 order set end symbol
order_set_end_1	output	Tx_PHY	Lane 1 order set end symbol
order_set_start_0	output	Tx_PHY	Lane 0 order set start symbol
order_set_start_1	output	Tx_PHY	Lane 1 order set start symbol
symb_out_valid_0	output	Tx_PHY	Valid symbol in from physical Rx lane 0
symb_out_valid_1	output	Tx_PHY	Valid symbol in from physical Rx lane 1
symb_out_0[7:0]	output	Tx_PHY	Symbol out to physical Tx lane 0
symb_out_1[7:0]	output	Tx_PHY	Symbol out to physical Tx lane 1
symb_type_0	output	Tx_PHY	Data(1)/control(0) type of symbol
symb_type_1	output	Tx_PHY	Data(1)/control(0) type of symbol

Simulation waveforms:

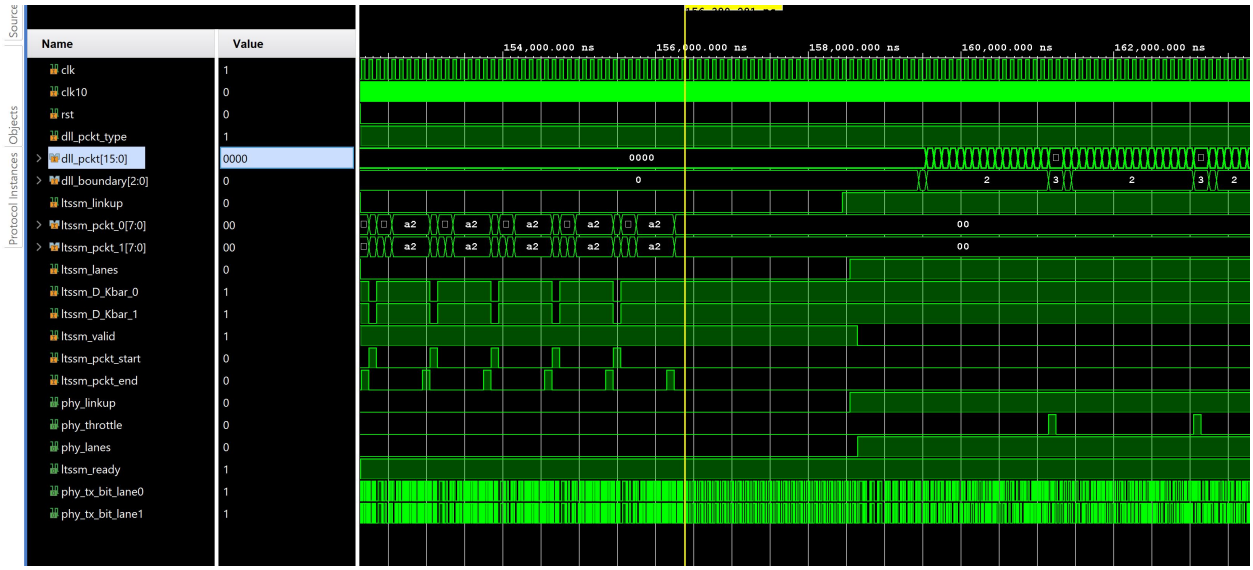
1. Tx_DLL Top:



- First clock DLL sees tl_boundary_info[2:0] == 3'b001 / 3'b1xx, it will reset dll_ready to 1'b0 and asks Transaction Layer to hold the data. In the meantime, Data Link Layer will transmit NEXT_TRANSMIT_SEQ first to Physical Layer, and indicates this is the first part of the TLP (dll_boundary_info == 3'b001). Also, NEXT_TRANSMIT_SEQ will be stored in Replay Buffer as a part of the TLP. At the end of this clock, NEXT_TRANSMIT_SEQ <= NEXT_TRANSMIT_SEQ + 1.

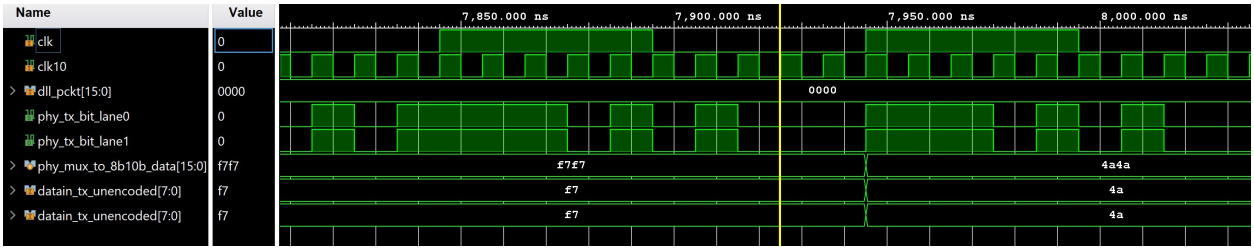
- In the next few clocks, DLL Transmitter sets dll_ready to 1'b1 and starts receiving the TLP from Transaction Layer (16 bits per clock). DLLPs will always be consumed in one clock. Transmitter forwards the AckNak_SEQ information in the DLLP to Replay Buffer, and update its own AckD_SEQ which stores the last acknowledged sequence number. Transmitter also gives the Ack/Nak information to Replay Buffer.

2. Tx_PHY Top:

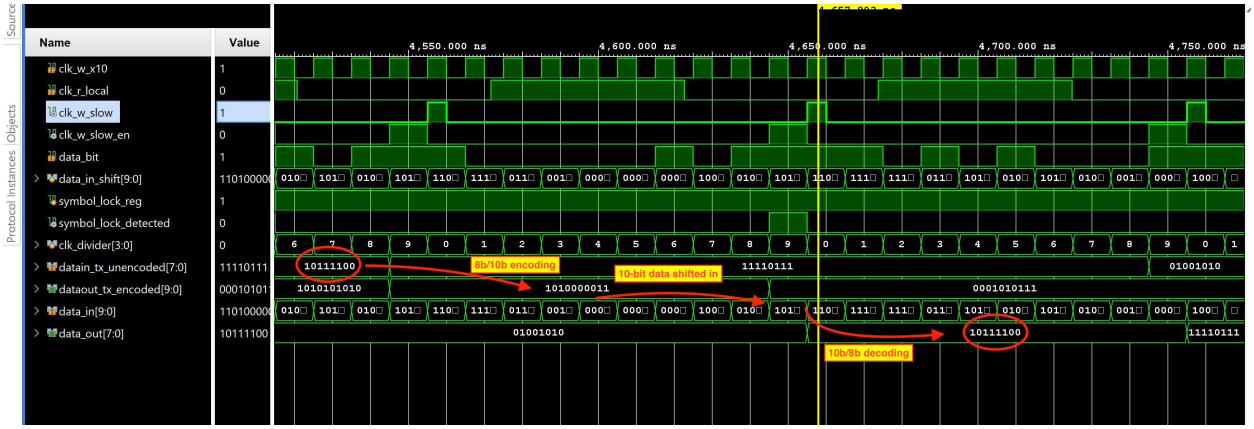


- LTSSM TS1/TS2 until link is up.

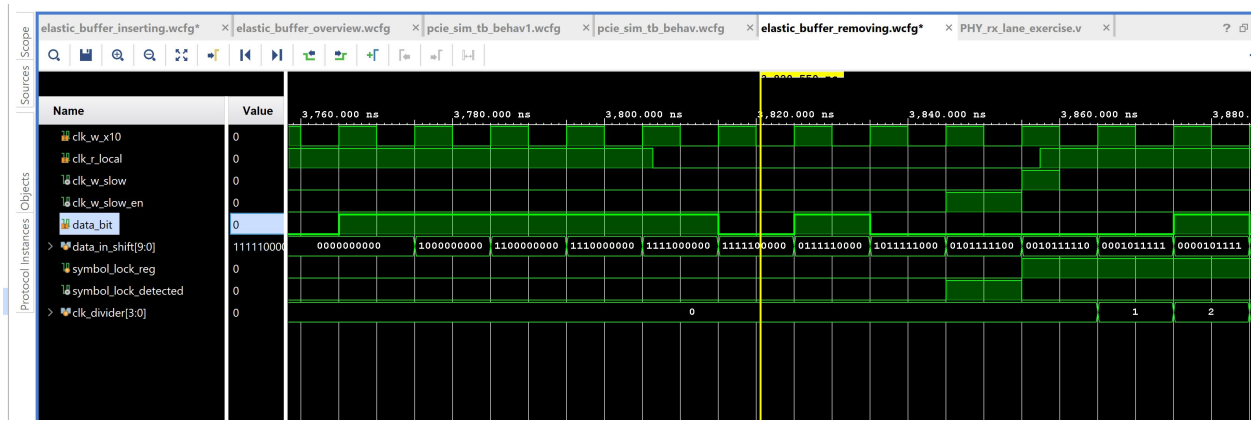
2. Tx_PHY Byte_Striping (lane 0, Lane 1):



3. Tx_PHY Encoding & Rx_PHY Decoding:

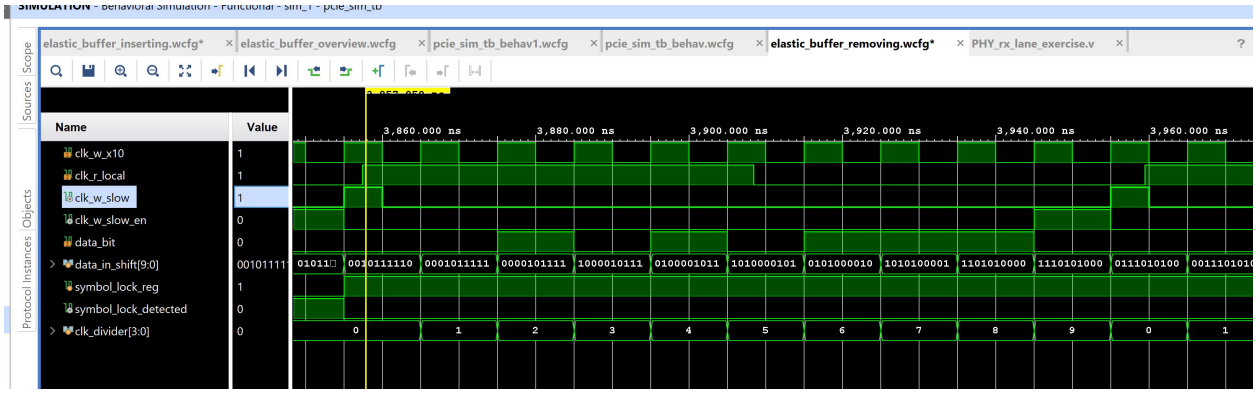


3. Rx_PHY 10-bit Shift Registering:



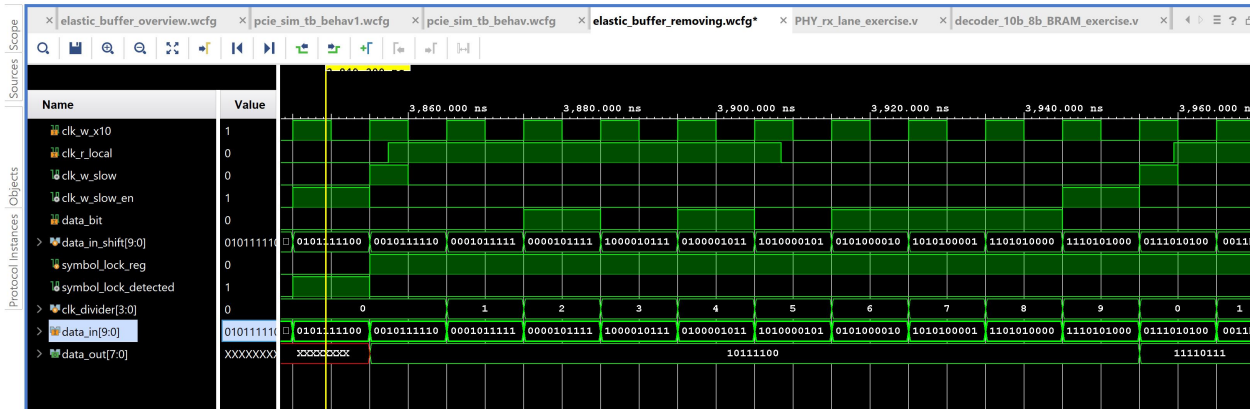
- 10-bit shift register is working under high frequency recovered clock (10x Tx-local clock frequency, **clk_w_x10**).
- Every **clk_w_x10** cycle, **data_in_shift** gets shifted by 1 bit.
- A COM symbol shows up in **data_in_shift.**
- symbol_lock_detected goes high.
- symbol_lock_reg latches symbol_lock_detected and stays high.
- clk_divider gets activated by symbol_lock_reg and keeps counting.
- clk_w_slow is generated as a divided by 10 version of clk_w_x10.

4. Rx_PHY Clocking:



- **10-bit shift register** is working under high frequency recovered clock (10x local clock frequency, **clk_w_x10**)
- **Decoder** and **write domain of EB** work under a divided-by-10 version of the high frequency recovered clock (**clk_w_slow**) (very close to local clock frequency of RX)
- Every **clk_w_x10** cycle, **data_in_shift** gets shifted by 1 bit. Every **clk_w_slow** cycle, input data to decoder (contents in **data_in_shift**) gets updated with a whole new 10-bit symbol.

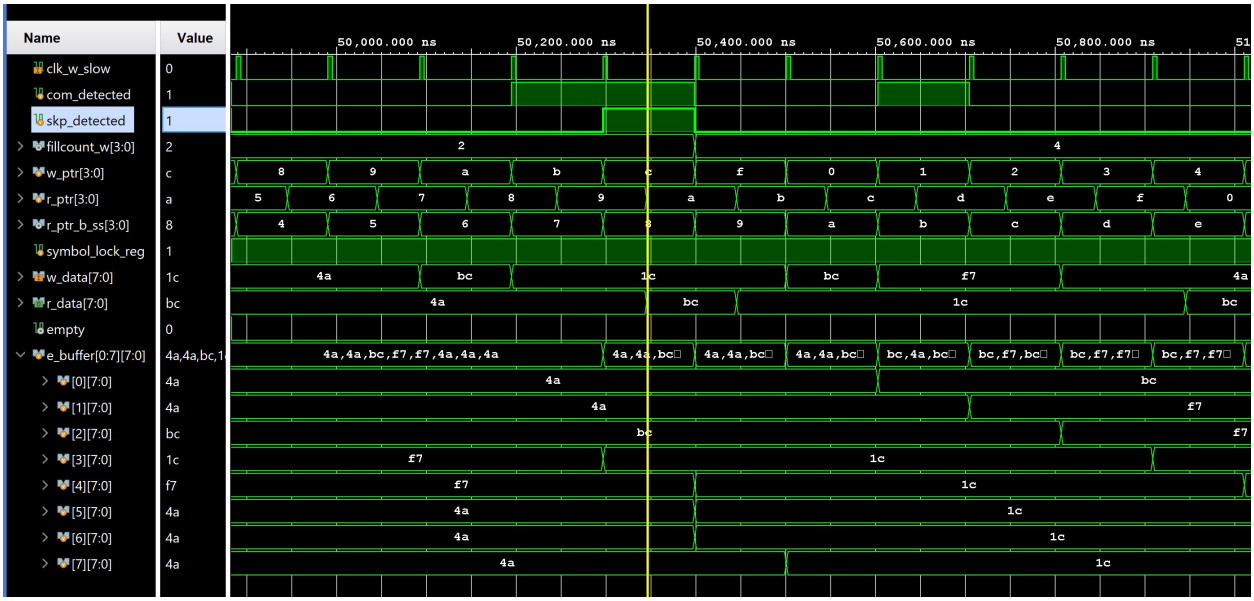
5. Rx_PHY COM Decoding:



6. Rx_PHY Elastic Buffer:

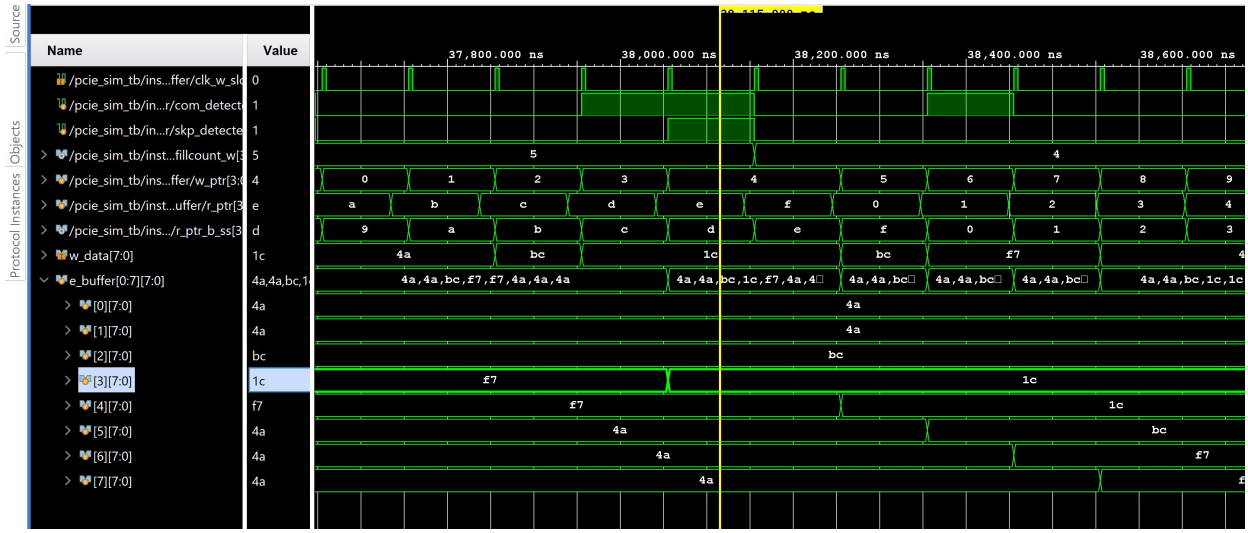
- As all Ordered Sets start with COM symbol, we can only tell whether it's an SOS after we know whether the 2nd symbol following COM is SKP symbol.
- The decision to insert/remove SKP(s) can only be made after the first SKP
- com_detected and skp_detected, along with the FIFO depth (fillcount_w) help to decide whether to insert or remove SKPs.
- Insertion is done by writing multiple entries in the same clock, an removal is done by stop incrementing write pointers and not writing any entry for current clock.
- Insertion/removal decision is made in write domain, so actually we

I. - Insert 1 SKP ordered set:

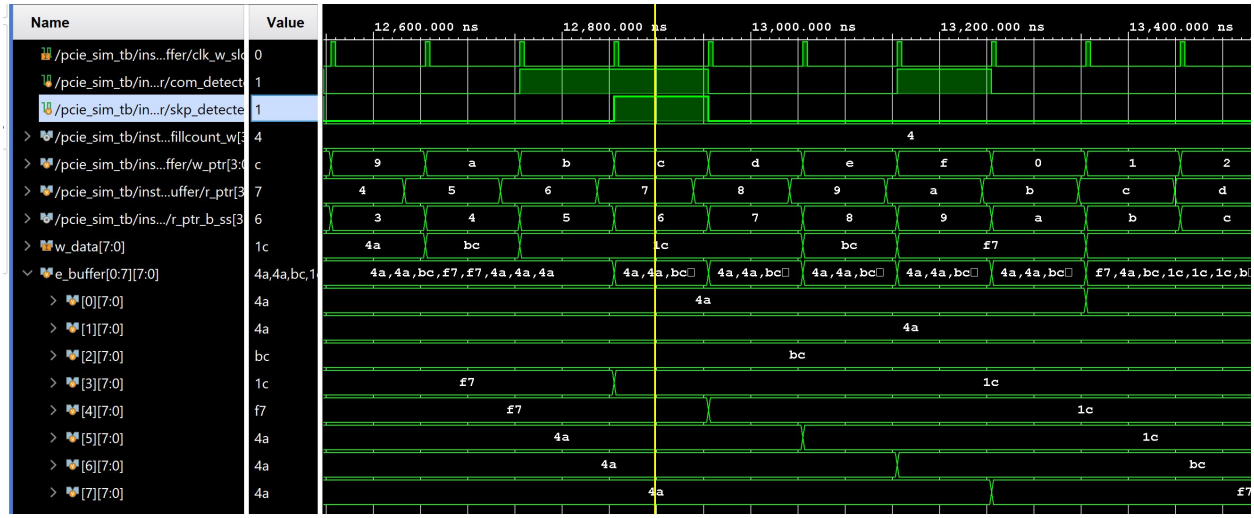


II. Insert 2 SKP ordered sets:

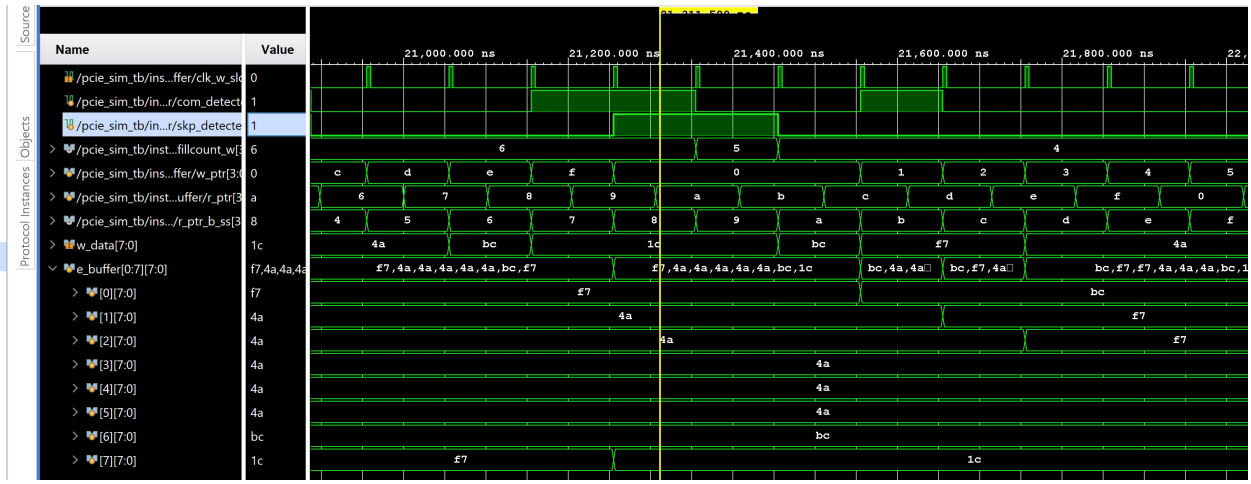
III. Normal Write:



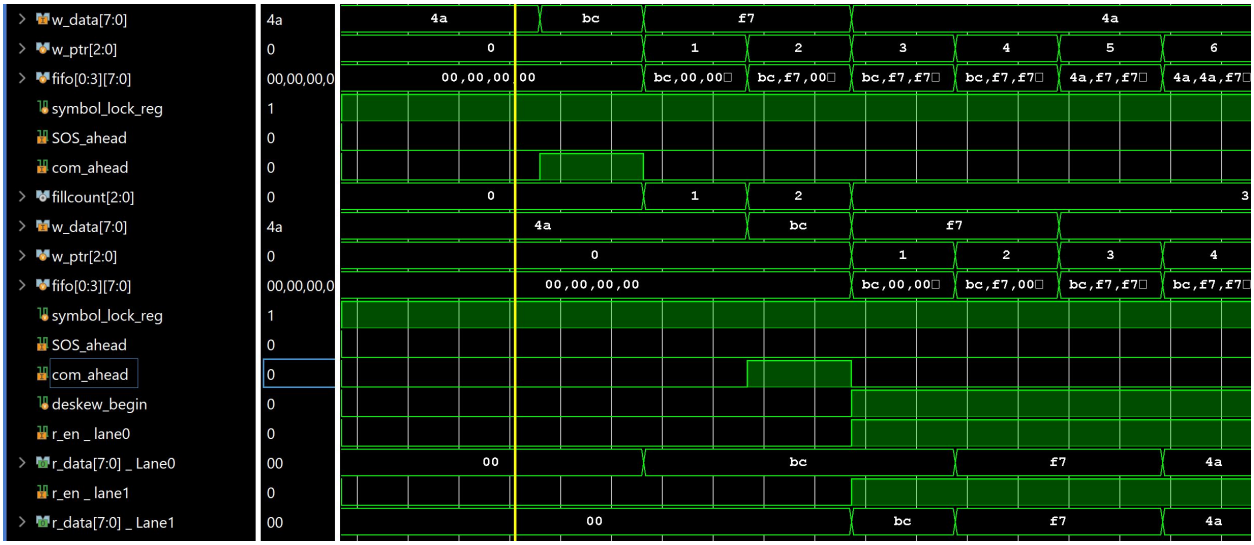
IV. Remove 1 SKP ordered set:



V. Remove 2 SKP ordered sets:

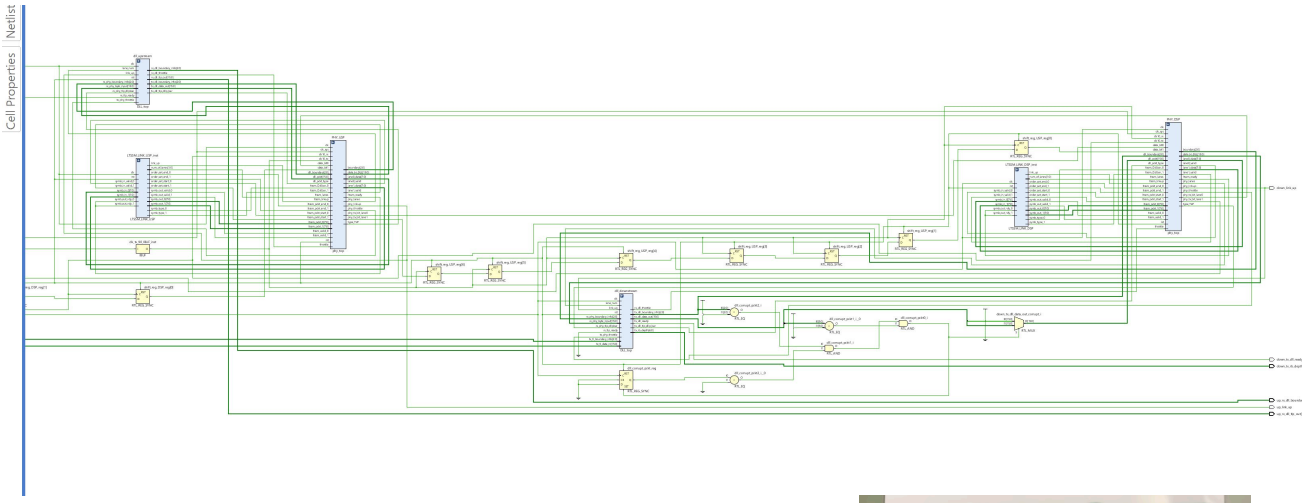


7. Rx_PHY De-skew FIFO:

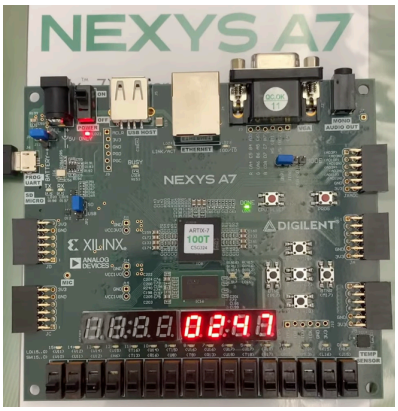


- In order for each lane deskew FIFO to safely begin to latch incoming symbols without worrying that they might belong to different groups, we **count for 4 strikes** after the **last symbol lock** is seen and then enable each lane deskew FIFO to accept the next **COM symbol (bc) onwards.**

PCEe Top Schematic:



Tools: Xilinx Vivado
FPGA Board: Nexys A7
https://digilent.com/shop/nexys-a7-fpga-trainer-board-recommended-for-ece-curriculum/



Synthesis (RTL to Circuit Netlist):

Synthesis Report and FPGA Cell Usage:

```
---------------------------------------------------------------------------------
Start Writing Synthesis Report
---------------------------------------------------------------------------------

Report BlackBoxes:
+-+--------------+----------+
| |BlackBox name |Instances |
+-+--------------+----------+
+-+--------------+----------+

Report Cell Usage:
+------+-----------+------+
|      |Cell       |Count |
+------+-----------+------+
|1     |BUFG       |    16|
|2     |BUFGCTRL   |     4|
|3     |CARRY4     |    96|
|4     |LUT1       |    86|
|5     |LUT2       |   461|
|6     |LUT3       |   593|
|7     |LUT4       |   572|
|8     |LUT5       |   767|
|9     |LUT6       |  2017|
|10    |MMCME2_ADV |     2|
|11    |MUXF7      |   271|
|12    |MUXF8      |    57|
|13    |RAM32M     |     8|
|14    |RAMB18E1   |     4|
|15    |RAMB18E1_1 |     3|
|16    |RAMB18E1_2 |     4|
|17    |RAMB18E1_3 |     4|
|18    |RAMB18E1_4 |     4|
|19    |RAMB18E1_5 |     4|
|20    |FDCE       |  1141|
|21    |FDPE       |   125|
|22    |FDRE       |  2144|
|23    |FDSE       |    31|
|24    |IBUF       |    65|
|25    |IBUFG      |     1|
|26    |OBUF       |   132|
+------+-----------+------+

Report Instance Areas:
+------+--------------------------+------------------------------+------+
|      |Instance                  |Module                        |Cells |
+------+--------------------------+------------------------------+------+
|1     |top                       |                              |  8612|
|2     |  inst_clk_wiz_0          |clk_wiz_0                     |     5|
|3     |    inst                  |clk_wiz_0_clk_wiz             |     5|
|4     |  inst_clk_wiz_1          |clk_wiz_1                     |     5|
|5     |    inst                  |clk_wiz_1_clk_wiz             |     5|
|6     |  bram0                   |bram                          |    56|
|7     |  bram1                   |bram_0                        |    12|
|8     |  bram2                   |bram_1                        |     4|
|9     |  bram3                   |bram_2                        |     2|
|10    |  debouncer_i0            |ee201_debouncer               |    69|
|11    |  inst_pcie               |pcie                          |  7732|
|12    |    LTSSM_LINK_DSP_inst   |LTSSM_LINK_DSP                |   850|
|13    |      LTSSM_LANE_0        |LTSSM_LANE_DSP                |   400|
|14    |      LTSSM_LANE_1        |LTSSM_LANE_DSP__parameterized0|   385|
|15    |    LTSSM_LINK_USP_inst   |LTSSM_LINK_USP                |   869|
|16    |      LTSSM_LANE_0        |LTSSM_LANE_USP                |   425|
|17    |      LTSSM_LANE_1        |LTSSM_LANE_USP__parameterized0|   381|
|18    |    PHY_DSP               |phy_top                       |  2439|
|19    |      phy_rx              |PHY_rx_14                     |  1240|
|20    |      phy_tx              |pcie_tx_data_serializer_15    |  1199|
|21    |        lane0_8b10b       |pcie_phy_8b10b_16             |    81|
|22    |        lane1_8b10b       |pcie_phy_8b10b_17             |   312|
|23    |    PHY_USP               |phy_top_7                     |  2701|
|24    |      phy_rx              |PHY_rx                        |  1509|
|25    |      phy_tx              |pcie_tx_data_serializer       |  1192|
|26    |        lane0_8b10b       |pcie_phy_8b10b                |    79|
|27    |        lane1_8b10b       |pcie_phy_8b10b_13             |   312|
|28    |    dll_downstream        |DLL_top                       |   598|
|29    |      inst_DLL_receiver   |DLL_receiver_9                |   103|
|30    |      inst_transmitter    |DLL_transmitter_10            |   495|
|31    |        inst_replay_buffer|replay_buffer_11              |   451|
```

```
|32     |          buffer_memory    |dp_bram_ft_12                 |   214|
|33     |    dll_upstream           |DLL_top_8                     |   249|
|34     |      inst_DLL_receiver    |DLL_receiver                  |   166|
|35     |        inst_dp_bram_ft    |dp_bram_2en_ft                |    44|
|36     |      inst_transmitter     |DLL_transmitter               |    83|
|37     |        inst_replay_buffer |replay_buffer                 |    59|
|38     |          buffer_memory    |dp_bram_ft                    |    34|
|39     |  receive_file_i0          |receive_file                  |   109|
|40     |    input_fifo             |data_fifo_oneclk_6            |    58|
|41     |  rst_gen_i0               |rst_gen                       |    10|
|42     |    reset_bridge_clk_i0    |reset_bridge                  |    10|
|43     |  send_file_i0             |send_file                     |    75|
|44     |    input_fifo             |data_fifo_oneclk_5            |    56|
|45     |  uart_rx_i0               |uart_rx                       |   106|
|46     |    data_fifo_i0           |data_fifo_oneclk_3            |    39|
|47     |    meta_harden_rxd_i0     |meta_harden                   |     2|
|48     |    uart_baud_gen_rx_i0    |uart_baud_gen_4               |    14|
|49     |    uart_rx_ctl_i0         |uart_rx_ctl                   |    50|
|50     |  uart_tx_i0               |uart_tx                       |    75|
|51     |    data_fifo_i0           |data_fifo_oneclk              |    25|
|52     |    uart_baud_gen_tx_i0    |uart_baud_gen                 |    14|
|53     |    uart_tx_ctl_i0         |uart_tx_ctl                   |    36|
+-------+---------------------------+------------------------------+------+
--------------------------------------------------------------------------------
Finished Writing Synthesis Report : Time (s): cpu = 00:01:24 ; elapsed = 00:01:55 . Memory (MB): peak = 974.047
; gain = 684.238
--------------------------------------------------
```

Place and Route (PnR):

Utilization report:

```
Utilization Design Information

Table of Contents
-----------------
1. Slice Logic
1.1 Summary of Registers by Type
2. Slice Logic Distribution
3. Memory
4. DSP
5. IO and GT Specific
6. Clocking
7. Specific Feature
8. Primitives
9. Black Boxes
10. Instantiated Netlists

1. Slice Logic
--------------

+----------------------------+------+-------+-----------+-------+
|          Site Type         | Used | Fixed | Available | Util% |
+----------------------------+------+-------+-----------+-------+
| Slice LUTs                 | 3883 |     0 |     63400 |  6.12 |
|   LUT as Logic             | 3851 |     0 |     63400 |  6.07 |
|   LUT as Memory            |   32 |     0 |     19000 |  0.17 |
|     LUT as Distributed RAM |   32 |     0 |           |       |
|     LUT as Shift Register  |    0 |     0 |           |       |
| Slice Registers            | 3441 |     0 |    126800 |  2.71 |
|   Register as Flip Flop    | 3441 |     0 |    126800 |  2.71 |
|   Register as Latch        |    0 |     0 |    126800 |  0.00 |
| F7 Muxes                   |  271 |     0 |     31700 |  0.85 |
| F8 Muxes                   |   57 |     0 |     15850 |  0.36 |
+----------------------------+------+-------+-----------+-------+

1.1 Summary of Registers by Type
--------------------------------

+-------+--------------+-------------+--------------+
| Total | Clock Enable | Synchronous | Asynchronous |
+-------+--------------+-------------+--------------+
| 0     |            _ |           - |            - |
| 0     |            _ |           - |          Set |
| 0     |            _ |           - |        Reset |
| 0     |            _ |         Set |            - |
| 0     |            _ |       Reset |            - |
| 0     |          Yes |           - |            - |
| 125   |          Yes |           - |          Set |
| 1141  |          Yes |           - |        Reset |
```

| 31 | Yes | Set | - |
| 2144 | Yes | Reset | - |
+-------+--------------+-------------+--------------+

2. Slice Logic Distribution

Site Type	Used	Fixed	Available	Util%
Slice	1728	0	15850	10.90
SLICEL	1170	0		
SLICEM	558	0		
LUT as Logic	3851	0	63400	6.07
using O5 output only	0			
using O6 output only	3213			
using O5 and O6	638			
LUT as Memory	32	0	19000	0.17
LUT as Distributed RAM	32	0		
using O5 output only	0			
using O6 output only	0			
using O5 and O6	32			
LUT as Shift Register	0	0		
LUT Flip Flop Pairs	1331	0	63400	2.10
fully used LUT-FF pairs	228			
LUT-FF pairs with one unused LUT output	1030			
LUT-FF pairs with one unused Flip Flop	1068			
Unique Control Sets	325			

* Note: Review the Control Sets Report for more information regarding control sets.

3. Memory

Site Type	Used	Fixed	Available	Util%
Block RAM Tile	11.5	0	135	8.52
RAMB36/FIFO*	0	0	135	0.00
RAMB18	23	0	270	8.52
RAMB18E1 only	23			

* Note: Each Block RAM Tile only has one FIFO logic available and therefore can accommodate only one FIFO36E1
or one FIFO18E1. However, if a FIFO18E1 occupies a Block RAM Tile, that tile can still accommodate a RAMB18E1

4. DSP

Site Type	Used	Fixed	Available	Util%
DSPs	0	0	240	0.00

5. IO and GT Specific

Site Type	Used	Fixed	Available	Util%
Bonded IOB	29	29	210	13.81
IOB Master Pads	13			
IOB Slave Pads	16			
Bonded IPADs	0	0	2	0.00
PHY_CONTROL	0	0	6	0.00
PHASER_REF	0	0	6	0.00
OUT_FIFO	0	0	24	0.00
IN_FIFO	0	0	24	0.00
IDELAYCTRL	0	0	6	0.00
IBUFDS	0	0	202	0.00
PHASER_OUT/PHASER_OUT_PHY	0	0	24	0.00
PHASER_IN/PHASER_IN_PHY	0	0	24	0.00
IDELAYE2/IDELAYE2_FINEDELAY	0	0	300	0.00
ILOGIC	0	0	210	0.00
OLOGIC	0	0	210	0.00

6. Clocking

Site Type	Used	Fixed	Available	Util%
BUFGCTRL	10	0	32	31.25
BUFIO	0	0	24	0.00
MMCME2_ADV	2	0	6	33.33
PLLE2_ADV	0	0	6	0.00
BUFMRCE	0	0	12	0.00
BUFHCE	0	0	96	0.00
BUFR	0	0	24	0.00

7. Specific Feature

Site Type	Used	Fixed	Available	Util%
BSCANE2	0	0	4	0.00
CAPTUREE2	0	0	1	0.00
DNA_PORT	0	0	1	0.00
EFUSE_USR	0	0	1	0.00
FRAME_ECCE2	0	0	1	0.00
ICAPE2	0	0	2	0.00
PCIE_2_1	0	0	1	0.00
STARTUPE2	0	0	1	0.00
XADC	0	0	1	0.00

8. Primitives

Ref Name	Used	Functional Category
FDRE	2144	Flop & Latch
LUT6	2005	LUT
FDCE	1141	Flop & Latch
LUT5	774	LUT
LUT3	585	LUT
LUT4	575	LUT
LUT2	468	LUT
MUXF7	271	MuxFx
FDPE	125	Flop & Latch
CARRY4	96	CarryLogic
LUT1	82	LUT
MUXF8	57	MuxFx
RAMD32	48	Distributed Memory
FDSE	31	Flop & Latch
OBUF	25	IO
RAMB18E1	23	Block Memory
RAMS32	16	Distributed Memory
BUFG	6	Clock
IBUF	4	IO
BUFGCTRL	4	Clock
MMCME2_ADV	2	Clock

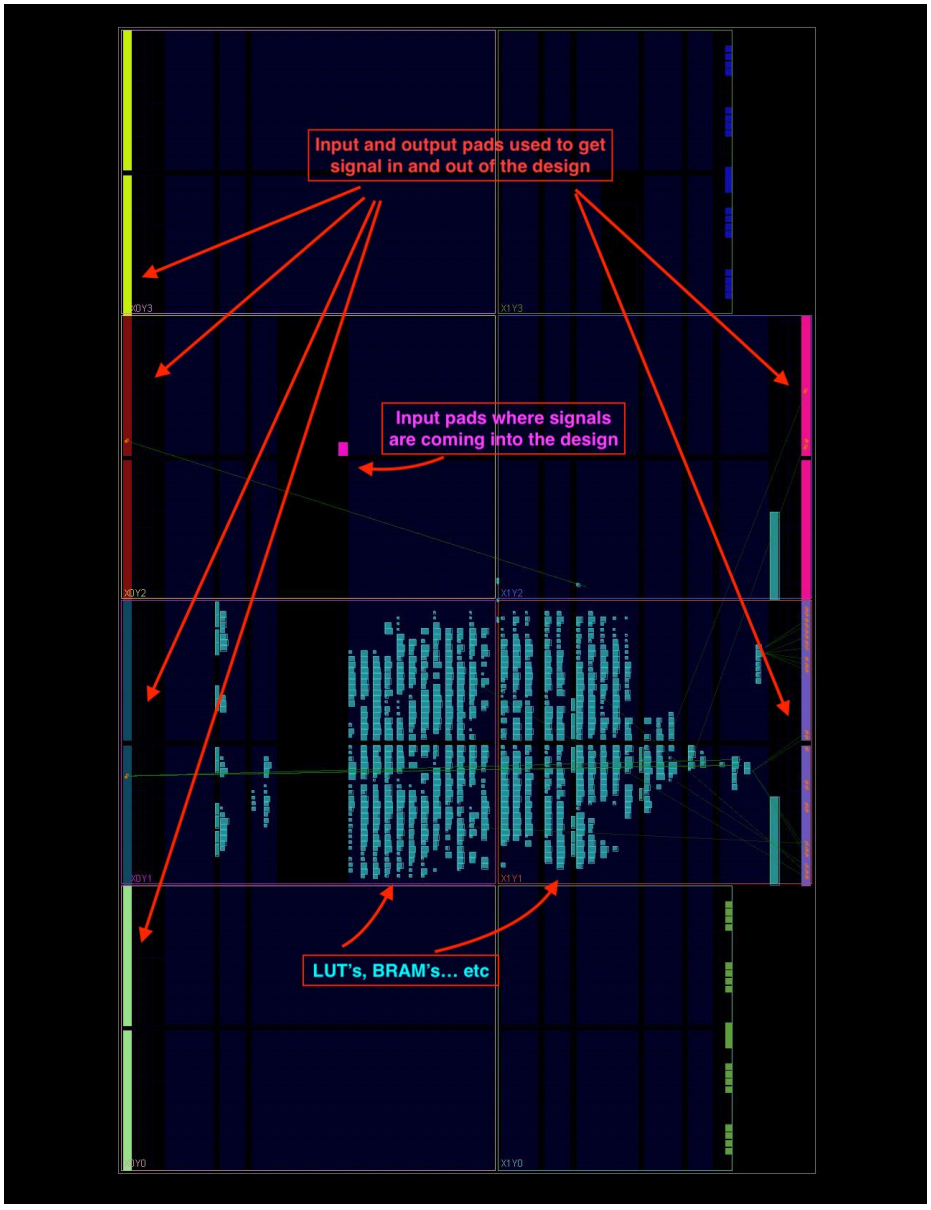
9. Black Boxes

Ref Name	Used

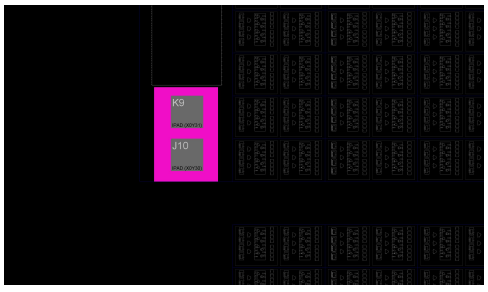
10. Instantiated Netlists

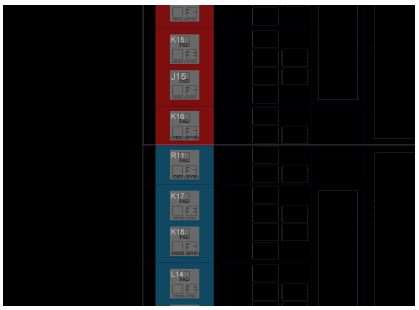
Ref Name	Used

Implementation Design:
This shows the physical layout of the FPGA and where the different parts of the design are mapped to.

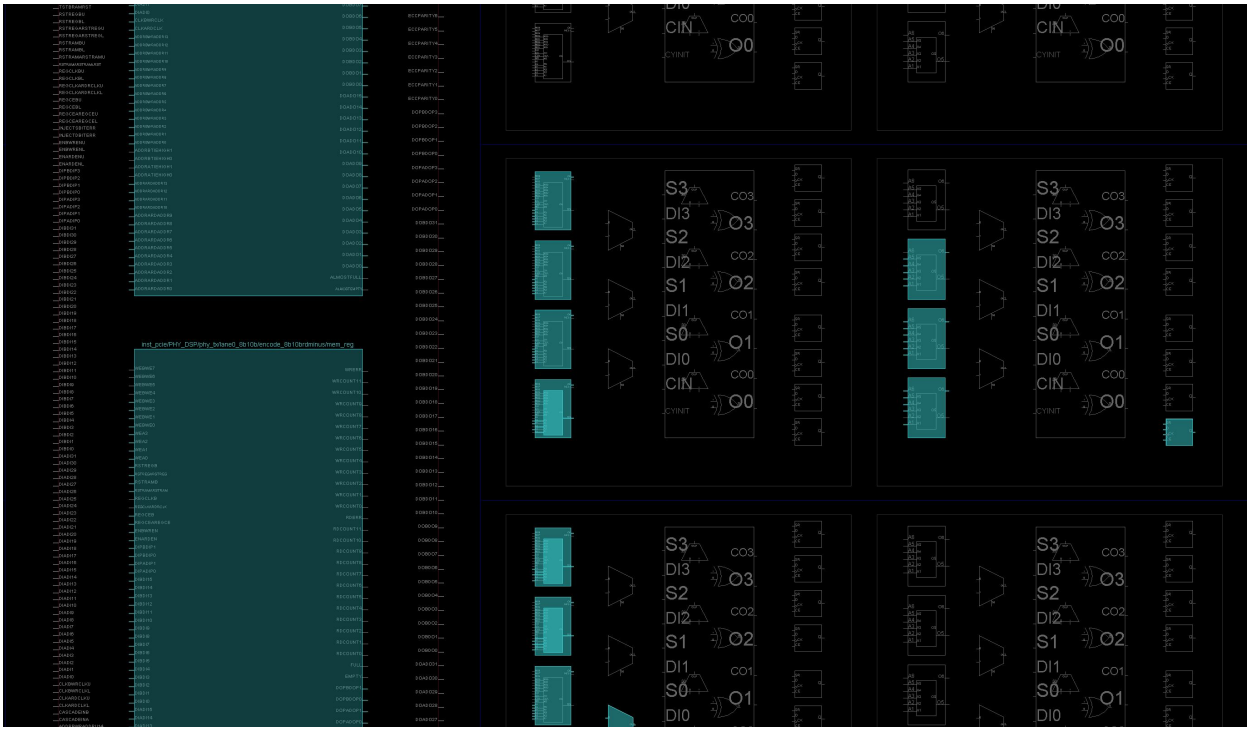


Zooming in for a closer view (input and output pads):





BRAM's and LUT's in the design:



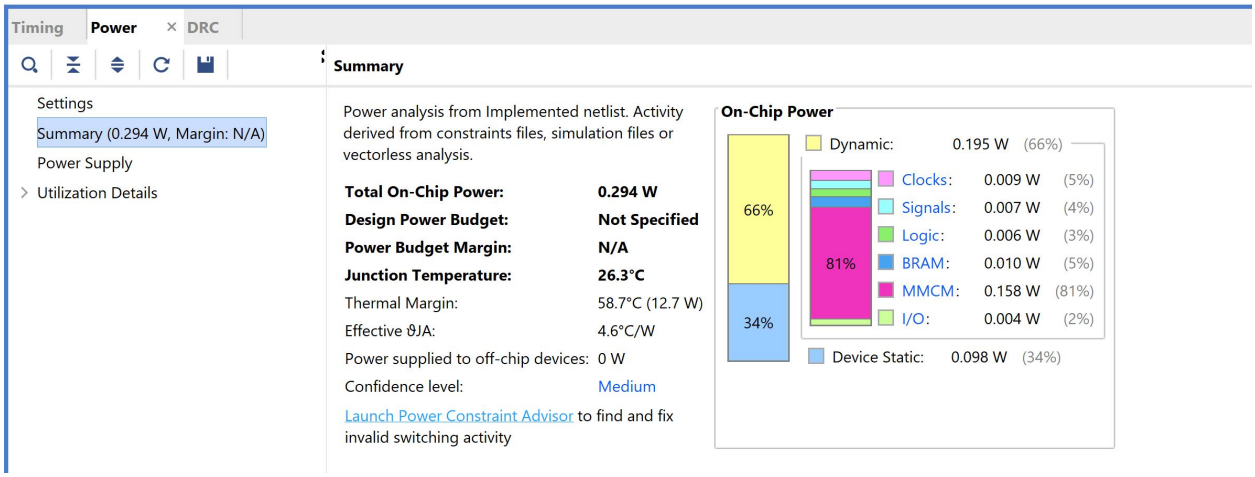
Timing Report:

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	2.353 ns	Worst Hold Slack (WHS):	0.072 ns	Worst Pulse Width Slack (WPWS):	3.000 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	7319	Total Number of Endpoints:	7319	Total Number of Endpoints:	3571

All user specified timing constraints are met.

Power Report:



Test-bench output:

```
launch_simulation: Time (s): cpu = 00:00:08 ; elapsed = 00:00:16 . Memory (MB): peak = 2428.391 ; gain = 0.000
run 200 us
At             159445000 (ns): TLP Packet sent Downstream (0210)
At             159645000 (ns): TLP Packet sent Downstream (0000)
At             159745000 (ns): TLP Packet sent Downstream (1111)
At             159845000 (ns): TLP Packet sent Downstream (2222)
At             159945000 (ns): TLP Packet sent Downstream (3333)
At             160045000 (ns): TLP Packet sent Downstream (4444)
At             160145000 (ns): TLP Packet sent Downstream (5555)
At             160245000 (ns): TLP Packet sent Downstream (6666)
At             160345000 (ns): TLP Packet sent Downstream (7777)
At             160445000 (ns): TLP Packet sent Downstream (8888)
At             160545000 (ns): TLP Packet sent Downstream (9999)
At             160645000 (ns): TLP Packet sent Downstream (aaaa)
At             160745000 (ns): TLP Packet sent Downstream (bbbb)
At             160845000 (ns): TLP Packet sent Downstream (cccc)
At             160945000 (ns): TLP Packet sent Downstream (dddd)
At             161045000 (ns): TLP Packet sent Downstream (eeee)
At             161102346 (ns): TLP Packet received Upstream (0210)
At             161145000 (ns): TLP Packet sent Downstream (ffff)
At             161204387 (ns): TLP Packet received Upstream (0000)
At             161306428 (ns): TLP Packet received Upstream (1111)
At             161345000 (ns): TLP Packet sent Downstream (0210)
At             161408469 (ns): TLP Packet received Upstream (2222)
At             161510510 (ns): TLP Packet received Upstream (3333)
At             161545000 (ns): TLP Packet sent Downstream (1f1f)
At             161612551 (ns): TLP Packet received Upstream (4444)
At             161645000 (ns): TLP Packet sent Downstream (1a1a)
At             161714591 (ns): TLP Packet received Upstream (5555)
At             161745000 (ns): TLP Packet sent Downstream (0000)
At             161816632 (ns): TLP Packet received Upstream (6666)
At             161845000 (ns): TLP Packet sent Downstream (1111)
At             161918673 (ns): TLP Packet received Upstream (7777)
At             161945000 (ns): TLP Packet sent Downstream (2222)
At             162020714 (ns): TLP Packet received Upstream (8888)
At             162045000 (ns): TLP Packet sent Downstream (3333)
At             162122755 (ns): TLP Packet received Upstream (9999)
At             162145000 (ns): TLP Packet sent Downstream (4444)
At             162224795 (ns): TLP Packet received Upstream (aaaa)
At             162245000 (ns): TLP Packet sent Downstream (5555)
At             162326836 (ns): TLP Packet received Upstream (bbbb)
At             162345000 (ns): TLP Packet sent Downstream (6666)
At             162428877 (ns): TLP Packet received Upstream (cccc)
At             162445000 (ns): TLP Packet sent Downstream (7777)
At             162530918 (ns): TLP Packet received Upstream (dddd)
At             162545000 (ns): TLP Packet sent Downstream (8888)
At             162632959 (ns): TLP Packet received Upstream (eeee)
At             162645000 (ns): TLP Packet sent Downstream (9999)
At             162734999 (ns): TLP Packet received Upstream (ffff)
At             162745000 (ns): TLP Packet sent Downstream (aaaa)
At             162845000 (ns): TLP Packet sent Downstream (bbbb)
At             162945000 (ns): TLP Packet sent Downstream (cccc)
At             163041122 (ns): TLP Packet received Upstream (0210)
At             163045000 (ns): TLP Packet sent Downstream (dddd)
At             163143163 (ns): TLP Packet received Upstream (1f1f)
At             163245000 (ns): TLP Packet sent Downstream (0210)
At             163245204 (ns): TLP Packet received Upstream (1a1a)
At             163347244 (ns): TLP Packet received Upstream (0000)
At             163445000 (ns): TLP Packet sent Downstream (eeee)
At             163449285 (ns): TLP Packet received Upstream (1111)
At             163545000 (ns): TLP Packet sent Downstream (ffff)
At             163551326 (ns): TLP Packet received Upstream (2222)
At             163645000 (ns): TLP Packet sent Downstream (1f1f)
At             163653367 (ns): TLP Packet received Upstream (3333)
At             163745000 (ns): TLP Packet sent Downstream (1a1a)
At             163755408 (ns): TLP Packet received Upstream (4444)
At             163845000 (ns): TLP Packet sent Downstream (0000)
At             163857448 (ns): TLP Packet received Upstream (5555)
At             163945000 (ns): TLP Packet sent Downstream (1111)
At             163959489 (ns): TLP Packet received Upstream (6666)
At             164045000 (ns): TLP Packet sent Downstream (2222)
At             164061530 (ns): TLP Packet received Upstream (7777)
At             164145000 (ns): TLP Packet sent Downstream (3333)
At             164163571 (ns): TLP Packet received Upstream (8888)
At             164245000 (ns): TLP Packet sent Downstream (4444)
At             164265612 (ns): TLP Packet received Upstream (9999)
At             164345000 (ns): TLP Packet sent Downstream (5555)
At             164367653 (ns): TLP Packet received Upstream (aaaa)
At             164445000 (ns): TLP Packet sent Downstream (6666)
```

```
At            164469693 (ns): TLP Packet received Upstream (bbbb)
At            164545000 (ns): TLP Packet sent Downstream (7777)
At            164571734 (ns): TLP Packet received Upstream (cccc)
At            164645000 (ns): TLP Packet sent Downstream (8888)
At            164673775 (ns): TLP Packet received Upstream (dddd)
At            164745000 (ns): TLP Packet sent Downstream (9999)
At            164845000 (ns): TLP Packet sent Downstream (aaaa)
At            164945000 (ns): TLP Packet sent Downstream (bbbb)
At            164979897 (ns): TLP Packet received Upstream (0210)
At            165081938 (ns): TLP Packet received Upstream (eeee)
At            165145000 (ns): TLP Packet sent Downstream (0210)
At            165183979 (ns): TLP Packet received Upstream (ffff)
At            165286020 (ns): TLP Packet received Upstream (1f1f)
At            165345000 (ns): TLP Packet sent Downstream (cccc)
At            165388061 (ns): TLP Packet received Upstream (1a1a)
At            165445000 (ns): TLP Packet sent Downstream (dddd)
At            165490102 (ns): TLP Packet received Upstream (0000)
At            165545000 (ns): TLP Packet sent Downstream (eeee)
At            165592142 (ns): TLP Packet received Upstream (1111)
At            165645000 (ns): TLP Packet sent Downstream (ffff)
At            165694183 (ns): TLP Packet received Upstream (2222)
At            165745000 (ns): TLP Packet sent Downstream (1f1f)
At            165796224 (ns): TLP Packet received Upstream (3333)
At            165845000 (ns): TLP Packet sent Downstream (1a1a)
At            165898265 (ns): TLP Packet received Upstream (4444)
At            165945000 (ns): TLP Packet sent Downstream (0000)
At            166000306 (ns): TLP Packet received Upstream (5555)
At            166045000 (ns): TLP Packet sent Downstream (1111)
At            166102346 (ns): TLP Packet received Upstream (6666)
At            166145000 (ns): TLP Packet sent Downstream (2222)
At            166204387 (ns): TLP Packet received Upstream (7777)
At            166245000 (ns): TLP Packet sent Downstream (3333)
At            166306428 (ns): TLP Packet received Upstream (8888)
At            166345000 (ns): TLP Packet sent Downstream (4444)
At            166408469 (ns): TLP Packet received Upstream (9999)
At            166445000 (ns): TLP Packet sent Downstream (5555)
At            166510510 (ns): TLP Packet received Upstream (aaaa)
At            166545000 (ns): TLP Packet sent Downstream (6666)
At            166612551 (ns): TLP Packet received Upstream (bbbb)
At            166645000 (ns): TLP Packet sent Downstream (7777)
At            166745000 (ns): TLP Packet sent Downstream (8888)
At            166845000 (ns): TLP Packet sent Downstream (9999)
At            166918673 (ns): TLP Packet received Upstream (0210)
At            167020714 (ns): TLP Packet received Upstream (cccc)
At            167045000 (ns): TLP Packet sent Downstream (0208)
At            167122755 (ns): TLP Packet received Upstream (dddd)
At            167224795 (ns): TLP Packet received Upstream (eeee)
At            167245000 (ns): TLP Packet sent Downstream (aaaa)
At            167326836 (ns): TLP Packet received Upstream (ffff)
At            167345000 (ns): TLP Packet sent Downstream (bbbb)
At            167428877 (ns): TLP Packet received Upstream (1f1f)
At            167445000 (ns): TLP Packet sent Downstream (cccc)
At            167530918 (ns): TLP Packet received Upstream (1a1a)
At            167545000 (ns): TLP Packet sent Downstream (dddd)
At            167632959 (ns): TLP Packet received Upstream (0000)
At            167645000 (ns): TLP Packet sent Downstream (eeee)
At            167734999 (ns): TLP Packet received Upstream (1111)
At            167745000 (ns): TLP Packet sent Downstream (ffff)
At            167837040 (ns): TLP Packet received Upstream (2222)
At            167845000 (ns): TLP Packet sent Downstream (1f1f)
At            167939081 (ns): TLP Packet received Upstream (3333)
At            167945000 (ns): TLP Packet sent Downstream (1a1a)
At            168041122 (ns): TLP Packet received Upstream (4444)
At            168143163 (ns): TLP Packet received Upstream (5555)
At            168145000 (ns): TLP Packet sent Downstream (0310)
At            168245204 (ns): TLP Packet received Upstream (6666)
At            168345000 (ns): TLP Packet sent Downstream (0000)
At            168347244 (ns): TLP Packet received Upstream (7777)
At            168445000 (ns): TLP Packet sent Downstream (0001)
At            168449285 (ns): TLP Packet received Upstream (8888)
At            168545000 (ns): TLP Packet sent Downstream (0002)
At            168551326 (ns): TLP Packet received Upstream (9999)
At            168645000 (ns): TLP Packet sent Downstream (0003)
At            168745000 (ns): TLP Packet sent Downstream (0004)
At            168845000 (ns): TLP Packet sent Downstream (0005)
At            168945000 (ns): TLP Packet sent Downstream (0006)
At            169045000 (ns): TLP Packet sent Downstream (0007)
At            169145000 (ns): TLP Packet sent Downstream (0008)
At            169163571 (ns): TLP Packet received Upstream (0208)
At            169245000 (ns): TLP Packet sent Downstream (0009)
At            169265612 (ns): TLP Packet received Upstream (aaaa)
```

```
At            169345000 (ns): TLP Packet sent Downstream (000a)
At            169367653 (ns): TLP Packet received Upstream (bbbb)
At            169445000 (ns): TLP Packet sent Downstream (000b)
At            169469693 (ns): TLP Packet received Upstream (cccc)
At            169545000 (ns): TLP Packet sent Downstream (000c)
At            169571734 (ns): TLP Packet received Upstream (dddd)
At            169645000 (ns): TLP Packet sent Downstream (000d)
At            169673775 (ns): TLP Packet received Upstream (eeee)
At            169745000 (ns): TLP Packet sent Downstream (000e)
At            169775816 (ns): TLP Packet received Upstream (ffff)
At            169845000 (ns): TLP Packet sent Downstream (000f)
At            169877857 (ns): TLP Packet received Upstream (1f1f)
At            169979897 (ns): TLP Packet received Upstream (1a1a)
At            170045000 (ns): TLP Packet sent Downstream (0310)
At            170245000 (ns): TLP Packet sent Downstream (001f)
At            170345000 (ns): TLP Packet sent Downstream (001b)
At            170445000 (ns): TLP Packet sent Downstream (0000)
At            170545000 (ns): TLP Packet sent Downstream (0001)
At            170645000 (ns): TLP Packet sent Downstream (0002)
At            170745000 (ns): TLP Packet sent Downstream (0003)
At            170845000 (ns): TLP Packet sent Downstream (0004)
At            170945000 (ns): TLP Packet sent Downstream (0005)
At            171045000 (ns): TLP Packet sent Downstream (0006)
At            171145000 (ns): TLP Packet sent Downstream (0007)
At            171245000 (ns): TLP Packet sent Downstream (0008)
At            171345000 (ns): TLP Packet sent Downstream (0009)
At            171445000 (ns): TLP Packet sent Downstream (000a)
At            171545000 (ns): TLP Packet sent Downstream (000b)
At            171645000 (ns): TLP Packet sent Downstream (000c)
At            171745000 (ns): TLP Packet sent Downstream (000d)
At            171945000 (ns): TLP Packet sent Downstream (0310)
At            174163571 (ns): TLP Packet received Upstream (0310)
At            174265612 (ns): TLP Packet received Upstream (0000)
At            174367653 (ns): TLP Packet received Upstream (0001)
At            174469693 (ns): TLP Packet received Upstream (0002)
At            174571734 (ns): TLP Packet received Upstream (0003)
At            174673775 (ns): TLP Packet received Upstream (0004)
At            174775816 (ns): TLP Packet received Upstream (0005)
At            174877857 (ns): TLP Packet received Upstream (0006)
At            174979897 (ns): TLP Packet received Upstream (0007)
At            175081938 (ns): TLP Packet received Upstream (0008)
At            175183979 (ns): TLP Packet received Upstream (0009)
At            175286020 (ns): TLP Packet received Upstream (000a)
At            175388061 (ns): TLP Packet received Upstream (000b)
At            175490102 (ns): TLP Packet received Upstream (000c)
At            175592142 (ns): TLP Packet received Upstream (000d)
At            175694183 (ns): TLP Packet received Upstream (000e)
At            175796224 (ns): TLP Packet received Upstream (000f)
At            175945000 (ns): TLP Packet sent Downstream (000e)
At            176045000 (ns): TLP Packet sent Downstream (000f)
At            176102346 (ns): TLP Packet received Upstream (0310)
At            176145000 (ns): TLP Packet sent Downstream (001f)
At            176204387 (ns): TLP Packet received Upstream (001f)
At            176245000 (ns): TLP Packet sent Downstream (001b)
At            176306428 (ns): TLP Packet received Upstream (001b)
At            176345000 (ns): TLP Packet sent Downstream (0000)
At            176408469 (ns): TLP Packet received Upstream (0000)
At            176445000 (ns): TLP Packet sent Downstream (0001)
At            176510510 (ns): TLP Packet received Upstream (0001)
At            176545000 (ns): TLP Packet sent Downstream (0002)
At            176612551 (ns): TLP Packet received Upstream (0002)
At            176645000 (ns): TLP Packet sent Downstream (0003)
At            176714591 (ns): TLP Packet received Upstream (0003)
At            176745000 (ns): TLP Packet sent Downstream (0004)
At            176816632 (ns): TLP Packet received Upstream (0004)
At            176845000 (ns): TLP Packet sent Downstream (0005)
At            176918673 (ns): TLP Packet received Upstream (0005)
At            176945000 (ns): TLP Packet sent Downstream (0006)
At            177020714 (ns): TLP Packet received Upstream (0006)
At            177045000 (ns): TLP Packet sent Downstream (0007)
At            177122755 (ns): TLP Packet received Upstream (0007)
At            177145000 (ns): TLP Packet sent Downstream (0008)
At            177224795 (ns): TLP Packet received Upstream (0008)
At            177245000 (ns): TLP Packet sent Downstream (0009)
At            177326836 (ns): TLP Packet received Upstream (0009)
At            177345000 (ns): TLP Packet sent Downstream (000a)
At            177428877 (ns): TLP Packet received Upstream (000a)
At            177445000 (ns): TLP Packet sent Downstream (000b)
At            177530918 (ns): TLP Packet received Upstream (000b)
At            177632959 (ns): TLP Packet received Upstream (000c)
At            177645000 (ns): TLP Packet sent Downstream (0310)
```

```
At            177734999 (ns): TLP Packet received Upstream (000d)
At            177845000 (ns): TLP Packet sent Downstream (000c)
At            177945000 (ns): TLP Packet sent Downstream (000d)
At            178045000 (ns): TLP Packet sent Downstream (000e)
At            178145000 (ns): TLP Packet sent Downstream (000f)
At            178245000 (ns): TLP Packet sent Downstream (001f)
At            178245204 (ns): TLP Packet received Upstream (0310)
At            178345000 (ns): TLP Packet sent Downstream (001b)
At            178347244 (ns): TLP Packet received Upstream (000e)
At            178445000 (ns): TLP Packet sent Downstream (0000)
At            178449285 (ns): TLP Packet received Upstream (000f)
At            178545000 (ns): TLP Packet sent Downstream (0001)
At            178551326 (ns): TLP Packet received Upstream (001f)
At            178645000 (ns): TLP Packet sent Downstream (0002)
At            178653367 (ns): TLP Packet received Upstream (001b)
At            178745000 (ns): TLP Packet sent Downstream (0003)
At            178755408 (ns): TLP Packet received Upstream (0000)
At            178845000 (ns): TLP Packet sent Downstream (0004)
At            178857448 (ns): TLP Packet received Upstream (0001)
At            178945000 (ns): TLP Packet sent Downstream (0005)
At            178959489 (ns): TLP Packet received Upstream (0002)
At            179045000 (ns): TLP Packet sent Downstream (0006)
At            179061530 (ns): TLP Packet received Upstream (0003)
At            179145000 (ns): TLP Packet sent Downstream (0007)
At            179163571 (ns): TLP Packet received Upstream (0004)
At            179245000 (ns): TLP Packet sent Downstream (0008)
At            179265612 (ns): TLP Packet received Upstream (0005)
At            179345000 (ns): TLP Packet sent Downstream (0009)
At            179367653 (ns): TLP Packet received Upstream (0006)
At            179469693 (ns): TLP Packet received Upstream (0007)
At            179545000 (ns): TLP Packet sent Downstream (0308)
At            179571734 (ns): TLP Packet received Upstream (0008)
At            179673775 (ns): TLP Packet received Upstream (0009)
At            179745000 (ns): TLP Packet sent Downstream (000a)
At            179775816 (ns): TLP Packet received Upstream (000a)
At            179845000 (ns): TLP Packet sent Downstream (000b)
At            179877857 (ns): TLP Packet received Upstream (000b)
At            179945000 (ns): TLP Packet sent Downstream (000c)
At            180045000 (ns): TLP Packet sent Downstream (000d)
At            180145000 (ns): TLP Packet sent Downstream (000e)
At            180183979 (ns): TLP Packet received Upstream (0310)
At            180245000 (ns): TLP Packet sent Downstream (000f)
At            180286020 (ns): TLP Packet received Upstream (000c)
At            180345000 (ns): TLP Packet sent Downstream (001f)
At            180388061 (ns): TLP Packet received Upstream (000d)
At            180445000 (ns): TLP Packet sent Downstream (001b)
At            180490102 (ns): TLP Packet received Upstream (000e)
At            180592142 (ns): TLP Packet received Upstream (000f)
At            180645000 (ns): TLP Packet sent Downstream (xxxx)
At            180694183 (ns): TLP Packet received Upstream (001f)
At            180745000 (ns): TLP Packet sent Downstream (0210)
At            180796224 (ns): TLP Packet received Upstream (001b)
At            180898265 (ns): TLP Packet received Upstream (0000)
At            180945000 (ns): TLP Packet sent Downstream (0000)
At            181000306 (ns): TLP Packet received Upstream (0001)
At            181045000 (ns): TLP Packet sent Downstream (1111)
At            181102346 (ns): TLP Packet received Upstream (0002)
At            181145000 (ns): TLP Packet sent Downstream (2222)
At            181204387 (ns): TLP Packet received Upstream (0003)
At            181245000 (ns): TLP Packet sent Downstream (3333)
At            181306428 (ns): TLP Packet received Upstream (0004)
At            181345000 (ns): TLP Packet sent Downstream (4444)
At            181408469 (ns): TLP Packet received Upstream (0005)
At            181445000 (ns): TLP Packet sent Downstream (5555)
At            181510510 (ns): TLP Packet received Upstream (0006)
At            181545000 (ns): TLP Packet sent Downstream (6666)
At            181612551 (ns): TLP Packet received Upstream (0007)
At            181645000 (ns): TLP Packet sent Downstream (7777)
At            181714591 (ns): TLP Packet received Upstream (0008)
At            181745000 (ns): TLP Packet sent Downstream (8888)
At            181816632 (ns): TLP Packet received Upstream (0009)
At            181845000 (ns): TLP Packet sent Downstream (9999)
At            181945000 (ns): TLP Packet sent Downstream (aaaa)
At            182045000 (ns): TLP Packet sent Downstream (bbbb)
At            182122755 (ns): TLP Packet received Upstream (0308)
At            182145000 (ns): TLP Packet sent Downstream (cccc)
At            182224795 (ns): TLP Packet received Upstream (000a)
At            182245000 (ns): TLP Packet sent Downstream (dddd)
At            182326836 (ns): TLP Packet received Upstream (000b)
At            182345000 (ns): TLP Packet sent Downstream (eeee)
At            182428877 (ns): TLP Packet received Upstream (000c)
```

```
At            182445000 (ns): TLP Packet sent Downstream (ffff)
At            182530918 (ns): TLP Packet received Upstream (000d)
At            182632959 (ns): TLP Packet received Upstream (000e)
At            182645000 (ns): TLP Packet sent Downstream (0210)
At            182734999 (ns): TLP Packet received Upstream (000f)
At            182837040 (ns): TLP Packet received Upstream (001f)
At            182845000 (ns): TLP Packet sent Downstream (1f1f)
At            182939081 (ns): TLP Packet received Upstream (001b)
At            182945000 (ns): TLP Packet sent Downstream (1a1a)
At            183045000 (ns): TLP Packet sent Downstream (0000)
At            183145000 (ns): TLP Packet sent Downstream (1111)
At            183245000 (ns): TLP Packet sent Downstream (2222)
At            183245204 (ns): TLP Packet received Upstream (0210)
At            183345000 (ns): TLP Packet sent Downstream (3333)
At            183347244 (ns): TLP Packet received Upstream (0000)
At            183445000 (ns): TLP Packet sent Downstream (4444)
At            183449285 (ns): TLP Packet received Upstream (1111)
At            183545000 (ns): TLP Packet sent Downstream (5555)
At            183551326 (ns): TLP Packet received Upstream (2222)
At            183645000 (ns): TLP Packet sent Downstream (6666)
At            183653367 (ns): TLP Packet received Upstream (3333)
At            183745000 (ns): TLP Packet sent Downstream (7777)
At            183755408 (ns): TLP Packet received Upstream (4444)
At            183845000 (ns): TLP Packet sent Downstream (8888)
At            183857448 (ns): TLP Packet received Upstream (5555)
At            183945000 (ns): TLP Packet sent Downstream (9999)
At            183959489 (ns): TLP Packet received Upstream (6666)
At            184045000 (ns): TLP Packet sent Downstream (aaaa)
At            184061530 (ns): TLP Packet received Upstream (7777)
At            184145000 (ns): TLP Packet sent Downstream (bbbb)
At            184163571 (ns): TLP Packet received Upstream (8888)
At            184245000 (ns): TLP Packet sent Downstream (cccc)
At            184265612 (ns): TLP Packet received Upstream (9999)
At            184345000 (ns): TLP Packet sent Downstream (dddd)
At            184367653 (ns): TLP Packet received Upstream (aaaa)
At            184469693 (ns): TLP Packet received Upstream (bbbb)
At            184545000 (ns): TLP Packet sent Downstream (0210)
At            184571734 (ns): TLP Packet received Upstream (cccc)
At            184673775 (ns): TLP Packet received Upstream (dddd)
At            184745000 (ns): TLP Packet sent Downstream (eeee)
At            184775816 (ns): TLP Packet received Upstream (eeee)
At            184845000 (ns): TLP Packet sent Downstream (ffff)
At            184877857 (ns): TLP Packet received Upstream (ffff)
At            184945000 (ns): TLP Packet sent Downstream (1f1f)
At            185045000 (ns): TLP Packet sent Downstream (1a1a)
$finish called at time : 185065 ns : File “C:/aabuhjar/Xilinx_projects/PCIe/Simulation/pcie_tb.v"
run: Time (s): cpu = 00:00:01 ; elapsed = 00:00:06 . Memory (MB): peak = 2428.391 ; gain = 0.000
```

***Reference**: Mindshare - https://www.mindshare.com/eLearning/Course/Core_PCIe_eLearning_Course#*