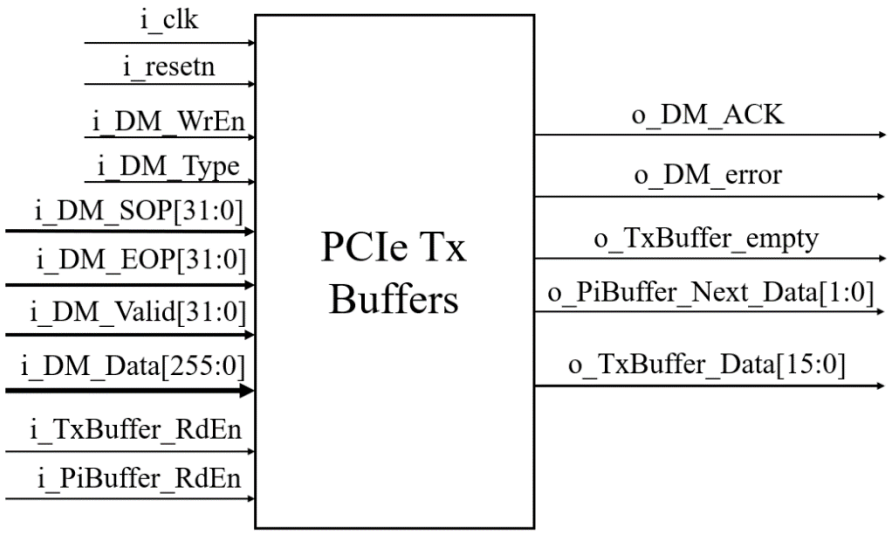
Tx Buffers' IP Specification

1. Introduction

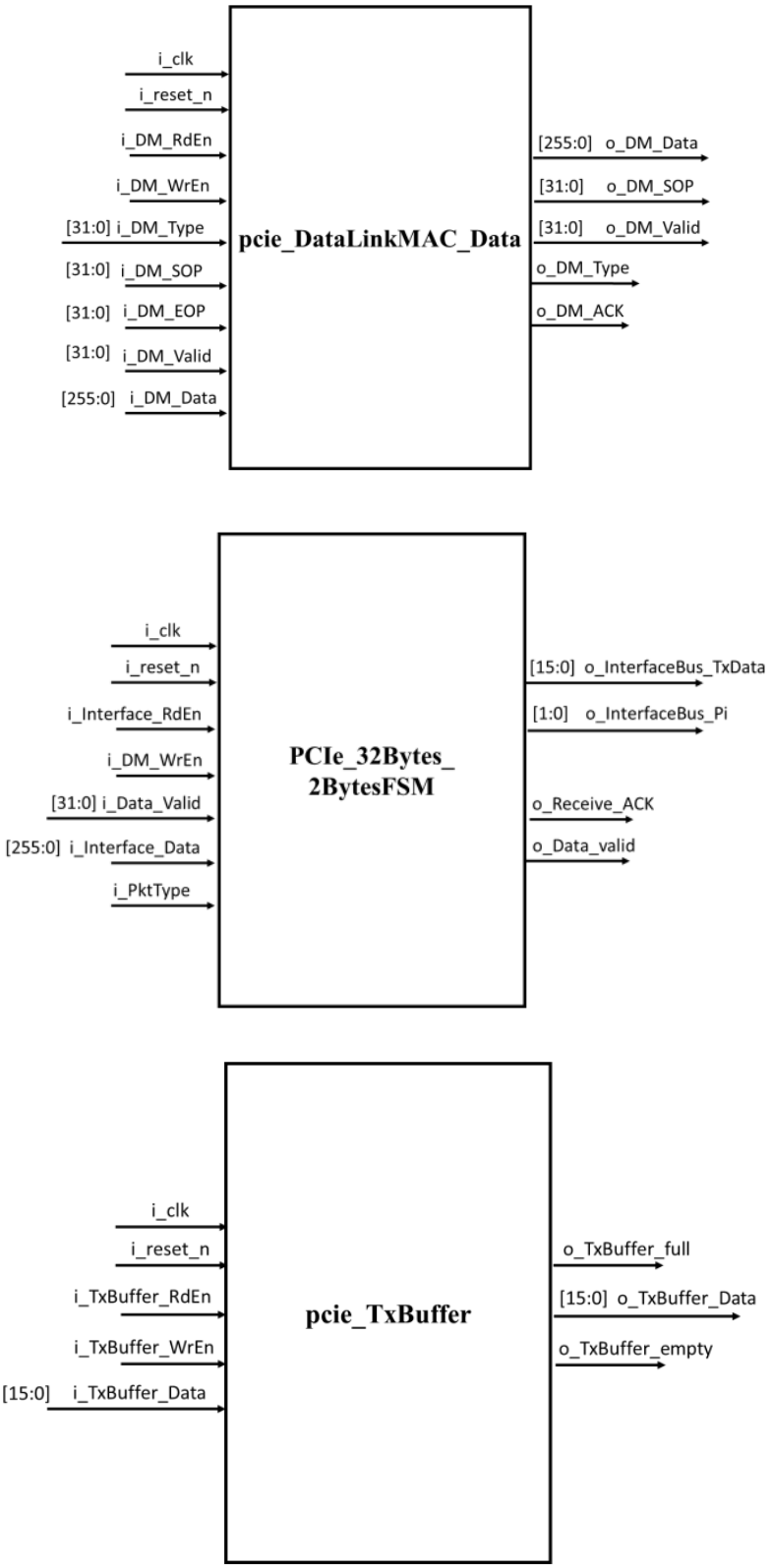
1.1. Basic Operation

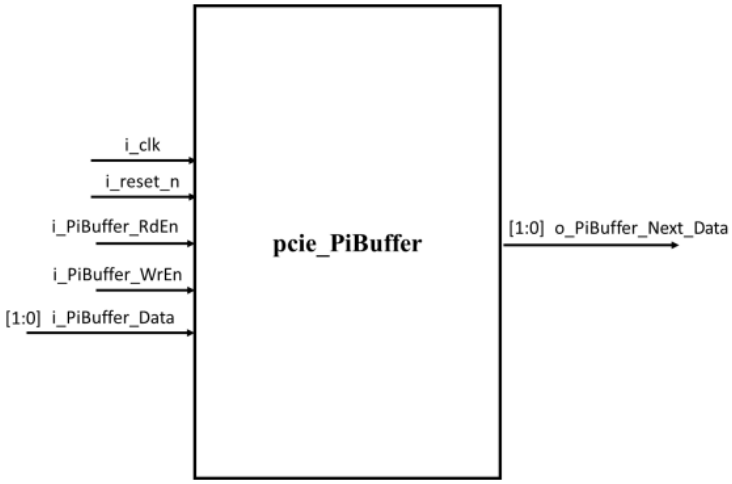
The group of modules (Interface Buffer, Interface Bus, Tx Buffer, and Packet Indicator Buffer) are supposed to receive a complete single packet at a time from the data link layer and then start transmitting it two bytes by two bytes to the multiplexer. In parallel, it outputs signals that indicate the start and the type (TLP or DLLP) of any transmitted packet to help Start – End Framing module to capsulate each packet correctly.

1.2. Block I/O Signals



Sub-blocks


i_clk
i_reset_n
i_DM_RdEn
i_DM_WrEn
[31:0] i_DM_Type
[31:0] i_DM_SOP
[31:0] i_DM_EOP
[31:0] i_DM_Valid
[255:0] i_DM_Data
pcie_DataLinkMAC_Data
[255:0] o_DM_Data
[31:0] o_DM_SOP
[31:0] o_DM_Valid
o_DM_Type
o_DM_ACK
i_clk
i_reset_n
i_Interface_RdEn
i_DM_WrEn
[31:0] i_Data_Valid
[255:0] i_Interface_Data
i_PktType
PCIe_32Bytes_
2BytesFSM
[15:0] o_InterfaceBus_TxData
[1:0] o_InterfaceBus_Pi
o_Receive_ACK
o_Data_valid
i_clk
i_reset_n
i_TxBuffer_RdEn
i_TxBuffer_WrEn
[15:0] i_TxBuffer_Data
pcie_TxBuffer
o_TxBuffer_full
[15:0] o_TxBuffer_Data
o_TxBuffer_empty



1.3. Interaction with Other Modules

1- Data Link Layer: When enabled, Data Link Layer can write data packets (TLP or DLLP) in the buffers along with the flags (SOP, EOP, Valid). Also, Buffers output an acknowledgement signal when they receive the end of a packet.

2- Tx Controller: Provides the buffers with control signals such as read enable. Also, buffers send flags that indicate emptiness, and the start and type of the stored packet.

3- Tx Multiplexer: When buffers are enabled to be read, they output the stored packet two bytes by two bytes to the multiplexer.

2. I/O Signals Description

2.1. Inputs

Name	Width	Source	Description
i_clk	1 bit	Global	Clock signal for positive edge registers.
i_reset_n	1 bit	Global	Synchronous low-level reset.
i_DM_WrEn	1 bit	Data Link Layer	Write enable, the data link layer asserts it to insert a packet if no other packet is already stored (checked by "o_DM_ACK = 0").

i_DM_Type	1 bit	Data Link Layer	Type of packet: 0 for TLP and 1 for DLLP.
i_DM_SOP	32 bits	Data Link Layer	The start of packet indicator. Each bit refers to the corresponding byte in "i_DM_Data". The bit which equals 1 indicates the start byte of the packet.
i_DM_EOP	32 bits	Data Link Layer	The end of packet indicator. Each bit refers to the corresponding byte in "i_DM_Data". The bit which equals 1 indicates the end byte of the packet.
i_DM_Valid	32 bits	Data Link Layer	The valid indicator. Each bit refers to the corresponding byte in "i_DM_Data".
i_DM_Data	32 bytes (256 bits)	Data Link Layer	The data which comes from the data link layer. If the packet is more than 32 Bytes, each 32 bytes of it comes in a single clock cycle.
i_TxBuffer_RdEn	1 bit	Controller	Read enable of the Tx buffer. When it is 1, the buffer outs two bytes to the MUX.
i_PiBuffer_RdEn	1 bit	Controller	Read enable of the Packet Indicator buffer. When it is 1, the controller reads the data in the buffer.

2.2. Outputs

Name	Width	Destination	Description
o_DM_ACK	1 bit	Data Link Layer	When the end of a packet is received, this flag is asserted to prevent the data link layer from sending any other packet till the stored packet is completely transmitted. On the other hand, when this signal is low, the data link layer can send packets to the buffer and the interface bus can't read the data from the buffer.
o_DM_error	1 bit	Data Link Layer	No info.
o_TxBuffer_Data	2 bytes (16 bits)	Multiplexer	Two bytes of the transmitted packet.
o_TxBuffer_empty	1 bit	Controller	Empty flag to indicate that the buffer is empty.
o_PiBuffer_Next_Data	2 bits	Controller	Two bits to indicate the start of packet and its type.

3. Detailed Operation

• Stores “i_DM_Data” in Interface Buffer as long as “i_DM_WrEn” is asserted till the completion of the packet is detected using status signals (SOP, EOP, Valid, Type).

• When the packet is completely stored, o_DM_ACK is driven to 1, and then the packet is transmitted two bytes by two bytes.

• In case “i_TxBuffer_RdEn” and “i_PiBuffer_RdEn” are asserted, the two bytes are transmitted to the Multiplexer as “o_TxBuffer_Data”. In parallel, the two bits of “o_PiBuffer_Next_Data” are also transmitted to the controller to indicate the start of packet and its type.

• While in case “i_TxBuffer_RdEn” and “i_PiBuffer_RdEn” are de-asserted, bytes are stored in Tx buffer till it’s filled. Only then, the whole transmission of data is stopped till “i_TxBuffer_RdEn” is asserted by the controller. This can be observed in the status signals of Tx Buffer: “o_TxBuffer_empty” and “o_TxBuffer_full”.

4. Configuration

4.1. Latency

• Interface Buffer either reads or transmits 32 bytes at a clock cycle.

• Interface Bus transmits 2 bytes at a clock cycle.

• Tx Buffer transmits 2 bytes at a clock cycle and may work in parallel with Interface Bus depending on “i_TxBuffer_RdEn” input signal from the Controller.

• Packet Indicator Buffer transmits 2 bits. It always works in parallel with Tx Buffer and may work in parallel with Interface Bus depending on “i_PiBuffer_RdEn” input signal from the controller.

Example: Transmitting a 320-byte TLP:

1. Storing the complete packet in Interface Buffer: 10 clock cycles.

2. Reading the stored packet two-byte by two-byte in Interface Bus: 160 clock cycles.

3. Assumed delay in which Controller is disabling Tx Buffer and Packet Indicator Buffer: 16 clock cycles.

4. Total latency = 10 + 160 + 16 = 186 clock cycles

4.2. Input Format and Constraints

Input Packets x1 Packet Format General Rules

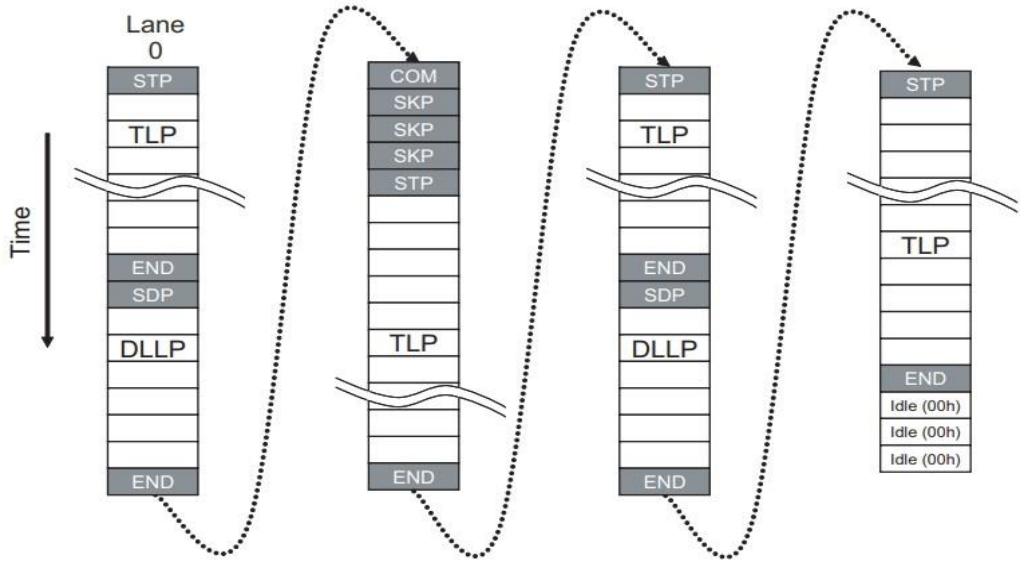
• The total packet length (including Start and End characters) of each packet is always a multiple of four characters.

• TLPs start with the STP character and finish with either an END or EDB character.

• DLLPs start with SDP, terminate with the END character and are exactly 8 characters long (SDP + 6 characters + END).

• Note: Start and End characters are not stored in the buffers, rather they are inserted using another IP (SE framing).

Example



Constraints

- The maximum allowable size of a packet is 544 Bytes, so it would take 17 clock cycles (17*32 bytes = 544 bytes) to store it and then start transmitting it.
- The start of a packet should be the first byte in the 32 input bytes of “i_DM_Data” signal.
- It is assumed that all input data from Data Link Layer is valid, meaning that all bytes between the start byte and the end byte of any packet must have a corresponding bit in “i_DM_Valid” that is equal to 1.
- Tx Buffer and Packet Indicator Buffer always work or stop in parallel, meaning that “i_PiBuffer_RdEn” and “i_TxBuffer_RdEn” input signals always have the same value.
- The capacity of Tx Buffer is 2048 rows, each has 2 bytes.
- The capacity of Packet Indicator Buffer is 2048 rows, each has 2 bits.

4.3. Expected Outputs

- o_DM_Ack:
  - ❖ 1 when the packet is completely received (when END character is received).
  - ❖ 0 when the packet is being received (after STP or SDP is received).
- o_TxBuffer_Data:
  - ❖ zeros during reset and subsequent cycles during which packet is being stored in Interface Buffer.
  - ❖ 2 bytes of the packet in case of transmission (“i_TxBuffer_RdEn” = 1).
- o_TxBuffer_full:
  - ❖ in case the 2048 rows of Tx Buffer are filled, each with 2 bytes, as “i_TxBuffer_RdEn” = 0 throughout this time.
  - ❖ 0 otherwise.
- o_TxBuffer_empty:
  - ❖ 1 during reset and subsequent cycles during which packet is being stored in Interface Buffer.
  - ❖ 0 otherwise.
- o_PiBuffer_Next_Data:
  - ❖ no_op = 2'b00
  - ❖ write = 2'b10
  - ❖ read = 2'b01
  - ❖ read_write = 2'b11

5. Sequences and Waveforms

5.1 Reset sequence:

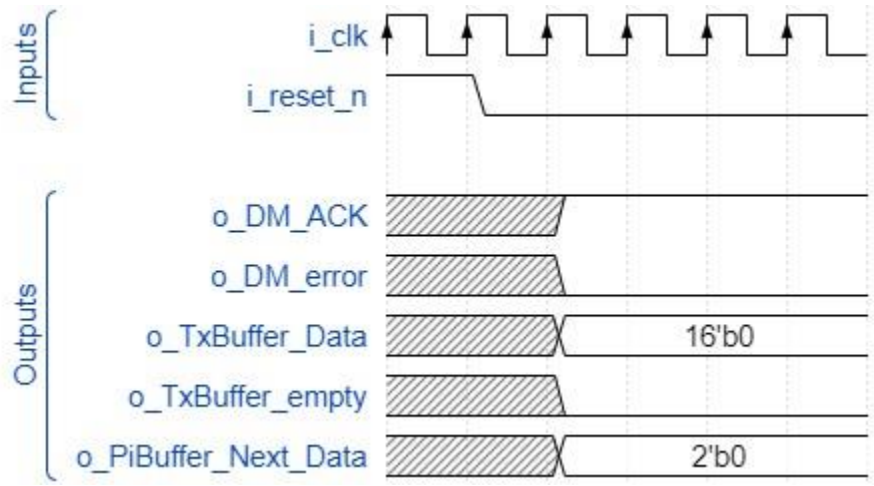
Inputs:

i_reset_n
0

Expected Outputs:

o_DM_ACK	o_DM_error	o_TxBuffer_Data	o_TxBuffer_empty	o_PiBuffer_Next_Data
1'b1	1'b0	16'b0	1'b1	2'b00

Waveform:



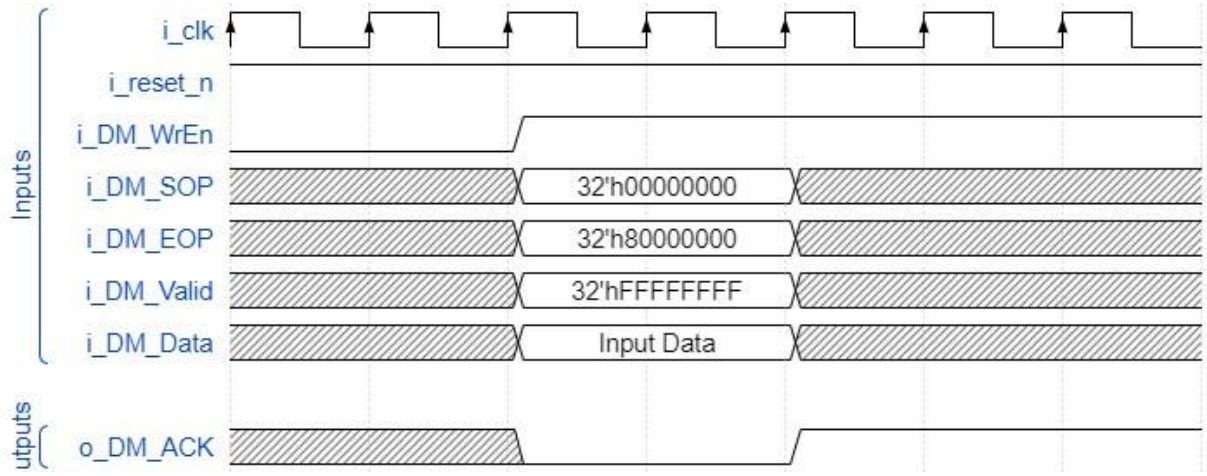
5.2 Write into Buffer Sequence:

32-byte write operation.

Inputs:

i_reset_n	i_DM_WrEn	i_DM_SOP	i_DM_EOP	i_DM_Valid	i_DM_Data
1	1'b1	32'h00000000	32'h80000000	32'hFFFFFFFF	256'b(Input Data)

Waveform:



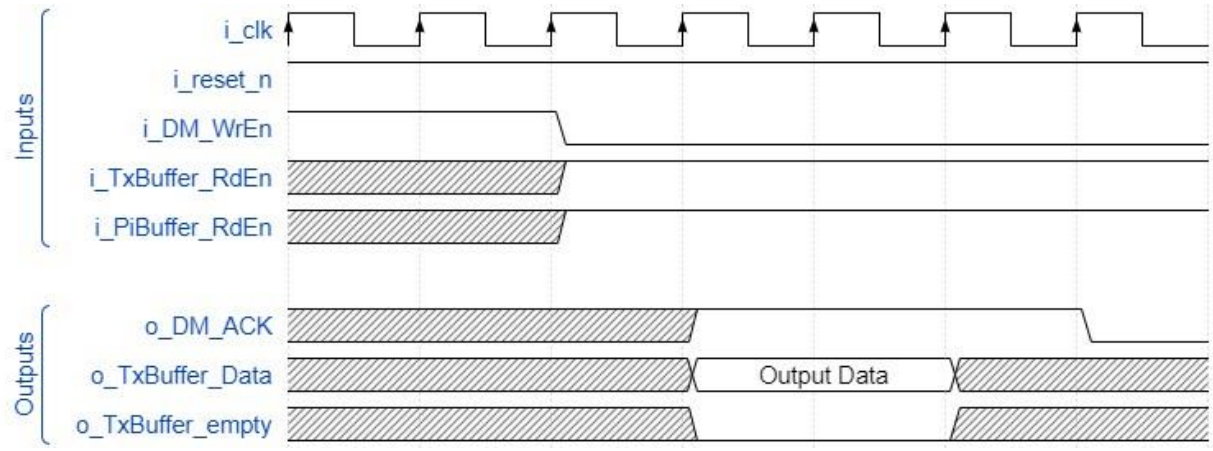
5.3 Read from Buffer Sequence:

2-byte read operation.

Inputs:

i_reset_n	i_DM_WrEn	i_TxBuffer_RdEn	i_PiBuffer_RdEn
1	1'b0	1'b1	1'b1

Waveforms:



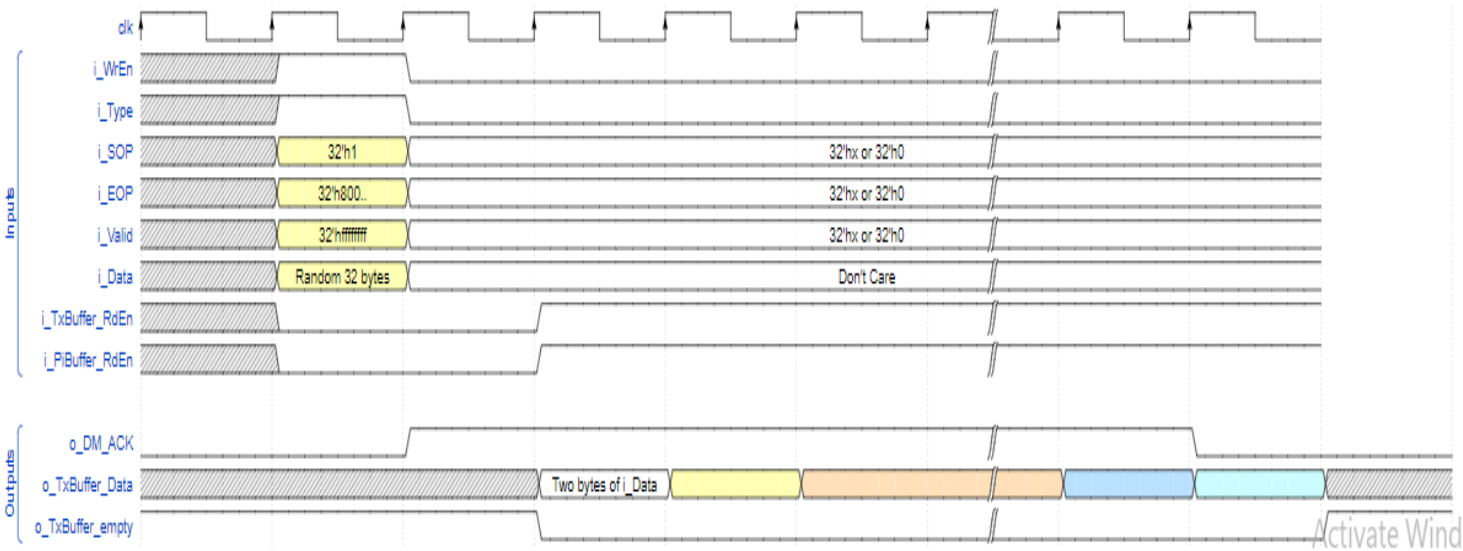
5.4 Flow of 32-byte TLP:

- Storing the complete packet in Interface Buffer: 1 clock cycle.
- Reading the stored packet two-byte by two-byte in Interface Bus: 16 clock cycles.
- Assumed delay in which Controller is disabling Tx Buffer and Packet Indicator Buffer by disabling both i_TxBuffer_RdEn & i_PiBuffer_RdEn signals: 1 clock cycle. Total latency = 1 + 16 + 1 = **18** clock cycles

Inputs:

Clk Cycle	i_WrEn	i_Type	i_SOP	i_EOP	i_Valid	i_Data	i_TxBuffer_RdEn	i_PiBuffer_RdEn
1	1 'b1	1'b0	32'h0000_0001	32'h8000_0000	32'hffff_ffff	Random 256 byte	1'b0	1'b0
2 (delay)	1'b0 (or x)	1'b0 (or x)	32'hx (or zeros)	32'hx (or zeros)	32'hx (or zeros)	Don't care	1'b0	1'b0
3	1'b0 (or x)	1'b0 (or x)	32'hx (or zeros)	32'hx (or zeros)	32'hx (or zeros)	Don't care	**1'b1**	**1'b1**
4 → 18	1'b0 (or x)	1'b0 (or x)	32'hx (or zeros)	32'hx (or zeros)	32'hx (or zeros)	Don't care	1'b1	1'b1

Waveforms:


clk
Inputs
i_WrEn
i_Type
i_SOP
32'h1
32'hx or 32'h0
i_EOP
32'h800..
32'hx or 32'h0
i_Valid
32'hffffffff
32'hx or 32'h0
i_Data
Random 32 bytes
Don't Care
i_TxBuffer_RdEn
i_PlBuffer_RdEn
Outputs
o_DM_ACK
o_TxBuffer_Data
Two bytes of i_Data
o_TxBuffer_empty