

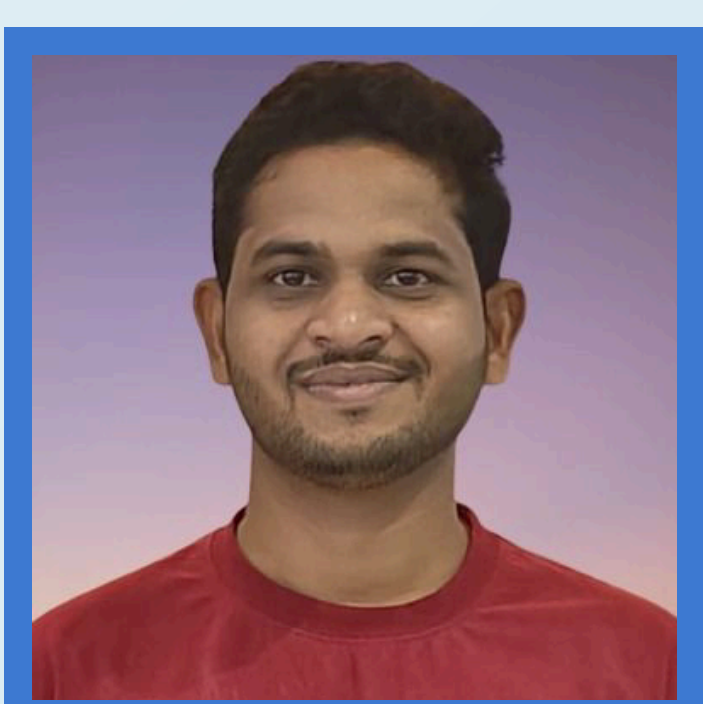


WHITEPAPER

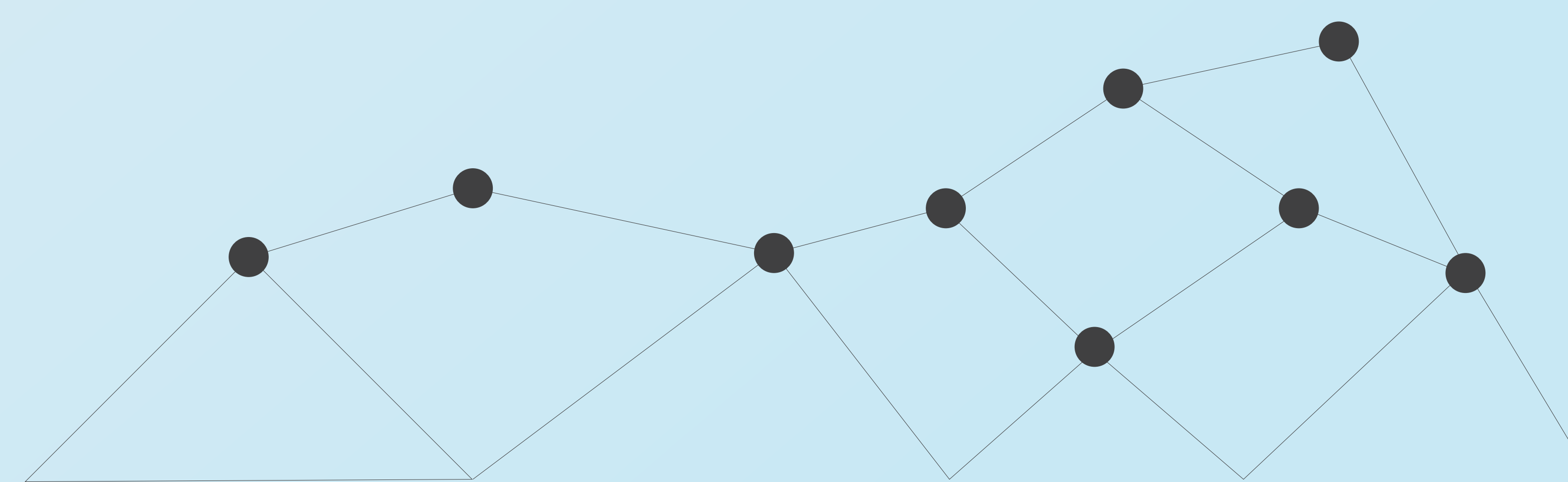
PCIe GEN-5

TRANSACTION LAYER - AN ULTIMATE GUIDE

PCIe has revolutionized the electronic industry with its High-speed data transfer, scalability, compatibility and continuous innovation. This white paper emphasizes the overview of the PCIe Transaction layer which plays a crucial role in interconnecting the application layer of the device with the other PCIe layers.



BY :
SUSRITH POLSANI,
R&D ENGINEER - FPGA



Contents

| | |
|---------------------------------------------|----|
| 1. Introduction | 03 |
| 2. The role of the Transaction layer | 03 |
| 3. Transaction Layer Packet (TLP) | 03 |
| 4. Structure of TLP | 03 |
| 5. Type of TLPs | 04 |
| i. Memory request TLP header | 05 |
| ii. Configuration request TLP header | 05 |
| iii. IO request TLP header | 06 |
| iv. Message request TLP header | 06 |
| v. Completion TLP header | 07 |
| 6. Data packet structure flow in PCIe | 07 |
| 7. Flow control in TL | 08 |
| 8. Definitions and acronyms | 09 |
| 9. About Us | 10 |
| 10. Logic Fruit Technologies and PCIe | 10 |

Introduction:

PCIe is a serial protocol that is accessible to transfer data between two devices. It is comprised of three layers namely Transaction Layer(TL), Data Link Layer(DLL), and Physical Layer(PL). The transaction layer is the topmost layer of the PCIe, which intermediates between the core logic of the external device and the DLL of the PCIe.

The Role of Transaction Layer

PCIe protocol follows packet-based data transmission. Transaction Layer plays a crucial role in data transmission at both transmitter and receiver devices.

- ✔ **Packetization:-** The transaction layer is responsible for the assembling and deassembling the TLP packet.
- ✔ **Data exchange:** The transaction layer receives the data from the application layer (transmitter core logic) and creates the TLPs. These TLP packets are forwarded to the Data link layer for further processing of data. At the receiver, the transaction layer verifies the TLP received from the DLL layer and extracts the data, which is passed to the application layer(receiver core logic).
- ✔ **Error detection:** The transaction layer reports the presence of ECRC(End-to-End CRC) in case any error is inserted in the data during transmission.
- ✔ **Flow control:-** Credit information addresses the buffer space available in the receiver device so that TLPs are forwarded accordingly.

Transaction Layer Packet (TLP)

The Transaction layer receives the data from the transmitter device and creates the packet known as Transaction Layer Packets(TLPs) for the transmission of data through the PCIe. TLPs contain the header information, data payload, and ECRC(optional) if supported by the core.

The transaction layer supports four address spaces, namely Memory requests, I/O requests, Configuration requests, and Message requests. Based on the type of request, TLP packet information and size will be varied.

Structure of TLP

The size of TLP, unlike DLLP size (8 bytes), is not fixed as it contains a TLP header, varied TLP data size (based on the request and type of TLP), and optionally ECRC is appended when enabled. So, the general structure of TLP is represented as shown in the below figure.

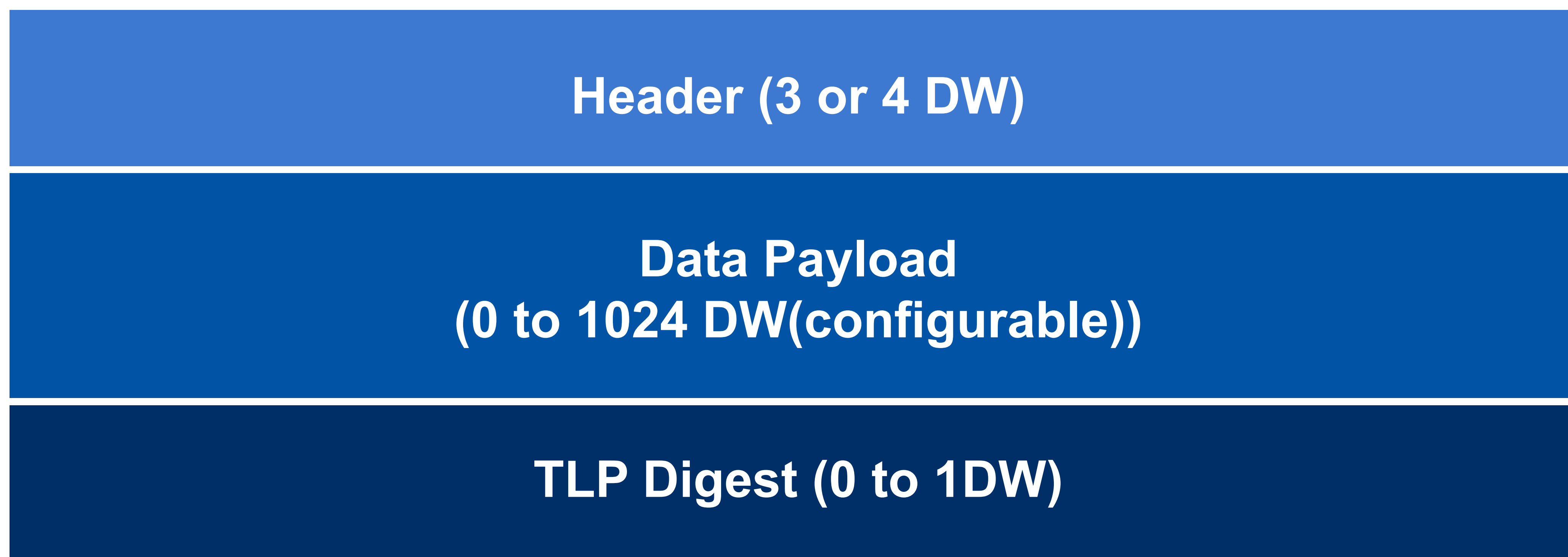


Figure 1 :- TLP packet structure

TLP header format size will be 3DW or 4DW based on the type of the packet and address width used. In a TLP header format, if “bit 0” of the format field is ‘0’ it’s a 3DW header, else it’s a 4DW header. If “bit 1” of the format field is ‘0’ then there is no data payload, else the payload is included. The Generic TLP header format fields are as shown in below figure.

| Byte 0 | | Byte 1 | | | | Byte 2 | | | | Byte 3 | | | |
|-----------------------------------------------------------|------|---------|----|---|------|---------|----|----|----|------------|-------------|--------|--|
| Format | Type | R | TC | R | Attr | R | TH | TD | EP | Attr | AT | length | |
| Byte 4 | | Byte 5 | | | | Byte 6 | | | | Byte 7 | | | |
| (Byte 4-6 depends on type of TLP) | | | | | | | | | | Last DW BE | First DW BE | | |
| Byte 8 | | Byte 9 | | | | Byte 10 | | | | Byte 11 | | | |
| (Byte 8-11 depends on type of TLP) | | | | | | | | | | | | | |
| Byte 12 | | Byte 13 | | | | Byte 14 | | | | Byte 15 | | | |
| (Byte 12-15 depends on type of TLP(not necessary always)) | | | | | | | | | | | | | |

Figure 2 :- General TLP header structure

Types of TLPs

Generally, TLPs are classified among three types as follows:-

- ✓ **Posted:-** Posted packets are ones where no response is expected. These include memory write and message TLPs.
- ✓ **Non-Posted:-** Non-posted packets are the ones where the response is required. These include Memory read, Configuration write & read, and I/O write & read TLPs.
- ✓ **Completion:-** A completion packet is a returned packet for an earlier packet in the opposite direction. These are in general the response packets to the Non - posted packets.

i. Memory request TLP header

Memory access TLPs are the fundamental means for reading and writing over the PCIe links. Based on the memory space, Memory access TLP can have either 3DWs (32-bit memory space) or 4DWs(64-bit memory space).

| Byte 0 | | Byte 1 | | | | Byte 2 | | | | Byte 3 | | | |
|----------------|-------|--------|----|---|------|---------|----|----|----|------------|-------------|--------|--|
| 0x0 | 0000x | 0 | TC | 0 | Attr | 0 | TH | TD | EP | Attr | AT | length | |
| Byte 4 | | Byte 5 | | | | Byte 6 | | | | Byte 7 | | | |
| Requester ID | | | | | | Tag | | | | Last DW BE | First DW BE | | |
| Byte 8 | | Byte 9 | | | | Byte 10 | | | | Byte 11 | | | |
| Address [31:2] | | | | | | | | | | | 00 | | |

Figure 3 :- 3DW Memory request header

| Byte 0 | | Byte 1 | | | | Byte 2 | | | | Byte 3 | | | |
|-----------------|-------|---------|----|---|------|---------|----|----|----|------------|-------------|--------|--|
| 0x0 | 0000x | 0 | TC | 0 | Attr | 0 | TH | TD | EP | Attr | AT | length | |
| Byte 4 | | Byte 5 | | | | Byte 6 | | | | Byte 7 | | | |
| Requester ID | | | | | | Tag | | | | Last DW BE | First DW BE | | |
| Byte 8 | | Byte 9 | | | | Byte 10 | | | | Byte 11 | | | |
| Address [63:32] | | | | | | | | | | | | | |
| Byte 12 | | Byte 13 | | | | Byte 14 | | | | Byte 15 | | | |
| Address [31:2] | | | | | | | | | | | 00 | | |

Figure 4 :- 4DW Memory request header

ii. Configuration request TLP header

The configuration access TLPs are used to access the configuration space of the PCIe. The configuration space is effectively the control and status registers of the PCIe interface.

| Byte 0 | | Byte 1 | | Byte 2 | | | Byte 3 | |
|--------------|-------|---------------|-----------------|---------|--------------------------|------|-----------------|-------------|
| 0x0 | 0010x | 00000000 | | TD | EP | 0000 | 00000001 | |
| Byte 4 | | Byte 5 | | Byte 6 | | | Byte 7 | |
| Requester ID | | | | Tag | | | 0000 | First DW BE |
| Byte 8 | | Byte 9 | | Byte 10 | | | Byte 11 | |
| Bus number | | Device number | Function number | 0000 | Extended Register number | | Register number | 00 |

Figure 5 :- Configuration request header

iii. I/O request TLP header

I/O transactions technically support the 32-bit IO range, but in reality, only the lower 16-bit(64KB) of its range is used by many systems(CPU). The size of IO TLP is always 4DW because it contains a 3DW header and only 1DW(32-bit) data which is being supported by IO transactions.

| Byte 0 | | Byte 1 | | Byte 2 | | | Byte 3 | |
|----------------|-------|----------|--|---------|----|------|----------|-------------|
| 0x0 | 00010 | 00000000 | | TD | EP | 0000 | 00000001 | |
| Byte 4 | | Byte 5 | | Byte 6 | | | Byte 7 | |
| Requester ID | | | | Tag | | | 0000 | First DW BE |
| Byte 8 | | Byte 9 | | Byte 10 | | | Byte 11 | |
| Address [31:2] | | | | | | | 00 | |

Figure 6 :- IO request header

iv. Message request TLP header

Message header always contains 4DW header but all the fields are not used by every message request ie; based on the type of message transmitted, the field related to it is used.

| Byte 0 | | Byte 1 | | | | Byte 2 | | | Byte 3 | |
|-----------------|-------|---------|----|---|------|---------|----|----|--------------|--------|
| 0x1 | 10xxx | 0 | TC | 0 | Attr | 00 | TD | EP | 0000 | length |
| Byte 4 | | Byte 5 | | | | Byte 6 | | | Byte 7 | |
| Requester ID | | | | | | Tag | | | Message code | |
| Byte 8 | | Byte 9 | | | | Byte 10 | | | Byte 11 | |
| Address [63:32] | | | | | | | | | | |
| Byte 12 | | Byte 13 | | | | Byte 14 | | | Byte 15 | |
| Address [31:2] | | | | | | | | | | 00 |

Figure 7 :- Message request header

v. Completion TLP header

Generally, Completion TLPs are the response to the non-posted packets. The completion header has the same values as the associated request, including Traffic Class, Attribute bits, and the original Requester ID (used to route the completion back to the Requester)

| Byte 0 | | Byte 1 | | | | Byte 2 | | | Byte 3 | |
|--------------|-------|--------|----|---|------|-------------------|----|-----|------------|---------------|
| 0x0 | 01010 | 0 | TC | 0 | Attr | 00 | TD | EP | 0000 | 00000001 |
| Byte 4 | | Byte 5 | | | | Byte 6 | | | Byte 7 | |
| Completer ID | | | | | | Completion status | | BCM | Byte Count | |
| Byte 8 | | Byte 9 | | | | Byte 10 | | | Byte 11 | |
| Requester ID | | | | | | Tag | | | 0 | Lower address |

Figure 8 :- Completion header

Data Packet structure flow in PCIe

The flow of data packets from the transmitter device to the receiver device will be done as follows:-

- Based on the transmitter device request, the Transaction layer forms a TLP packet containing a header, data payload, and ECRC/Digest(optional). This TLP is then passed on to the Data Link Layer.

- ☞ The Data Link Layer adds the sequence number and LCRC fields to form DLLP, then appends it to the TLP received and passes the DLLP to the physical layer.
- ☞ The physical layer appends the required control characters such as start and end to the DLLP received. Then does byte striping, scrambling, encoding, and serializing the bits to make the serial data transmission over the PCIe lanes.
- ☞ On the receiver end, the Physical layer receives the serial data and then de-serializes the data, decodes, and un-stripes the bytes. Then passes on the DLLP to the Data Link Layer.
- ☞ Data Link Layer checks the CRC and then sequence number is checked. After then Link Layer sends the TLP to the transaction layer.
- ☞ The transaction layer decodes the TLP and checks the ECRC if enabled and then passes the appropriate information to the receiver device.

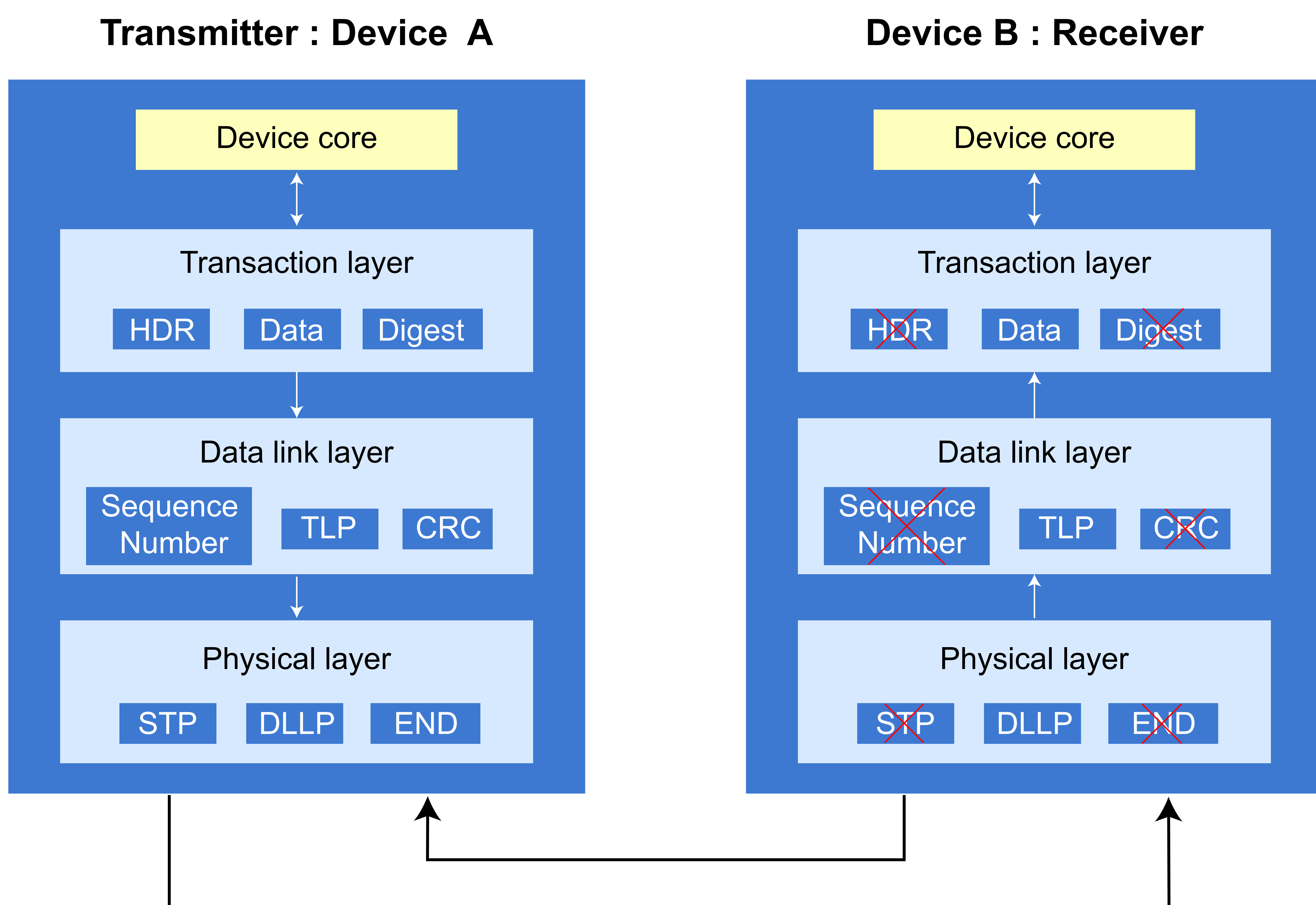


Figure 9 :- Data transmission in PCIe

Flow control in TL

During packet transmission in PCIe, the transmitter checks whether the receiver port has enough buffer space to send the packet. This process is required for the efficient transmission of the packets without any disturbances and is known as the Flow control mechanism. The Flow control can be regulated using a Virtual channel (VC).

The flow control mechanism is a credit-based mechanism to send buffer information. The receiver reports the size of the flow control buffer in units known as Credits.

Using multiple Virtual channels can increase the efficiency of the packet transmission. Each Virtual channel carries transactions that are independent of the traffic flow in other Virtual channels because the flow-control buffers are maintained separately. PCIe supports a maximum of 8 Virtual channels.

The flow control mechanism is a shared process between the transmit layer and the link layer. The receiver's transaction layer sends the availability of the credit information to the link layer of the transmitter. The link layer then creates a flow control DLLP to forward the credit information to the receiver's link layer. Upon receiving the flow control DLLP, the receiver's link layer transfers the credit value to the transmit layer of the transmitter. The process continues until total credit information is sent to the target device. Based on credit information, the transmitter sends the TLP for the smooth flow of the packet transmission.

Definitions and acronyms:-

Format (Fmt):- Format indicates whether the header is 3DW or 4DW, and the presence of payload in a TLP.

Type:- The type field indicates the type of TLP used for transmission.

Length:- Length field defines the size of the data payload that needs to be read/written when the TLP is being transmitted.

Requester ID:- Requester ID consists of the information about the bus number, device number, and function number.

Completion ID:- Completion ID consists of the information about the bus number, device number, and function number of the device issuing the completion.

Tag:- Tag is a unique identifier assigned to each TLP by the sender to associate it with a specific transaction.

It is used by the receiver to match incoming completion packets with the corresponding TLPs that initiated the transaction.

Transaction ID:- The combination of the Requester ID and the Tag field of the TLP is termed as Transaction ID of the TLP.

Attribute (Attr):- Attribute field includes the ID-based Ordering, Relaxed Ordering, and No Snoop bits information of the TLP.

- ✔ **ID-based Ordering:-** Generally packets from different requesters do not have dependencies on each other, this information is passed through the ID-based ordering concept.
- ✔ **Relaxed ordering:-** When a transaction doesn't have dependencies on other transactions of the same requester, then the TLP is said to follow relaxed ordering.
- ✔ **No Snoop:-** It means that no host cache coherency issues exist in a TLP.

Traffic class (TC):- The priorities of TLPs are passed through traffic class field within a TLP.

Poisoned data (EP):- This field indicates whether the data payload is poisoned.

TLP Processing Hints (TH):- This field suggest the system on how to handle the TLP.

Reserved(R) :- The reserved bits are generally set to zero(0).

TLP Digest (TD):- This field indicates the presence of a digest field at the end of the TLP.

About Us

At Logic Fruit, we specialize in the development and Validation of high-quality real-time high throughput FPGA/SoC embedded solutions, and Developing Proof-of-concept (PoC) designs/prototypes with real-time data generation, acquisition, and analysis.

Our engineers have expertise in many high-speed protocols and interfaces, including 1G/10G/40G/100G Ethernet, PCIe(Gen1-Gen6), USB3.0/4.0, CPRI/ORAN, DisplayPort, ARINC818.

The team also has deep expertise in Signal processing for wireless and Imaging-based solution development, software-defined radio (SDR), as well as encryption, protocol compliance, signal generation, data analysis, IoT technology, and multiple image processing techniques.

Logic Fruit Technologies and PCIe

We have worked on RTL IP and sub-system design for the Latest generation of PCIe and other high-speed serial protocols, and the development of device drives. Verification is done using the latest methodologies like UVM and RTL-SW co-simulation. We also perform FPGA prototyping, validation, and testing with real DUTs. The development of reference hardware is done with different kinds of PCIe devices and links.



Thank You!

Does anyone have any questions?

Contact Us



Gurugram (Headquarter)

806, 8th Floor
BPTP Park Centra Sector-30,
NH-8 Gurgaon - 122001
Haryana (India)

info@logic-fruit.com

+91-124 4643950



Bengaluru (R&D House)

Sy. No 118, 3rd Floor,
Gayathri Lakefront,
Outer Ring Road, Hebbal,
Bangalore - 560 024

sales@logic-fruit.com

+91 80-69019700/01



United States (Sales Office)

Logic Fruit Technologies
INC 691 S Milpitas Blvd
Ste 217 (Room 9) Milpitas
CA 95035

info@logic-fruit.com

+1-408 338 9743