

Streamline PCIe 6.0 Switch Design with effective Verification strategies

Deep Mehta, Cadence Design Systems Pvt Ltd, Ahmedabad, India (deep@cadence.com)

Meghvendra Rathod, Cadence Design Systems Pvt Ltd, Ahmedabad, India (rathodm@cadence.com)

Nicolas Dai, Cadence Design Systems Pvt Ltd, Velizy, France (ndai@cadence.com)

Abstract— The purpose of this paper is to address PCIe 6.0 Switch design verification challenges by providing different verification strategies and highlighting the performance metric criteria that aid in the closure of switch design verification. PCIe 6.0 Switch must translate between Flit Mode (FM) and Non-Flit Mode (NFM) Transaction Layer Packet (TLP) formats when the Ingress Port and Egress Port are in different speed or modes. Applying newly introduced translation rules, routing logic in switch design may add performance penalty that conventional data integration testing cannot detect. Verifying PCIe 6.0 switch design demands interoperability verification with FM mode and NFM mode traffic, Reserved TLP verification in FM mode traffic and Performance testing.

Keywords—*PCIe Switch; Flit Mode (FM) Verification; Non-Flit Mode (NFM) Verification*

I. INTRODUCTION

The demand for PCIe 6.0 switches has surged due to the exponential growth in global data traffic. PCIe 6.0 switches play a crucial role in enabling high-performance computing (HPC) systems, particularly in data centers, for applications that demand massive bandwidth and ultra-low latency. Yet, ensuring these switches meet strict criteria for performance, power efficiency, and cost presents a formidable challenge. The complexity of designing these switches can be mitigated through thorough testing and verification processes.

Traditional verification methods, like data integrity and virtual channel arbitration testing used for PCIe 5.0 switches, remain valuable. However, PCIe 6.0 demands a more comprehensive approach. This includes generating backpressure traffic to identify potential performance bottlenecks and ensure the switch operates optimally in real-world scenarios. By proactively addressing these challenges, we can guarantee low latency and high bandwidth for demanding high performance computing applications.

II. BACKGROUND

To ensure robust functionality of PCIe 6.0 switches, this section explores three critical verification areas which includes FM/NFM mode interoperability, reserved TLP verification in FM mode, and performance testing.

A. Flit Mode Vs Non-Flit Mode Interoperability Verification

The introduction of Flit Mode in PCIe 6.0 presents new verification challenges for Switches. As shown in Figure 1, we can breakdown the key areas to consider for interoperability testing with dedicated traffic generation. For a detailed explanation of the specific translation rule, please refer to the Transaction Layer chapter of the PCI Express 6.0 Base Specification [1].

- Switches must seamlessly handle a mix of Flit Mode and Non-Flit Mode traffic across different ports. This verification ensures no performance degradation occurs due to traffic mode heterogeneity. The relevant performance metrics criteria are discussed in this paper.
- Non-Flit Mode and Flit Mode utilize distinct header formats. As per PCIe specification [1] the switch design needs to translate between these formats when ingress and egress ports operate in different modes. Verification focuses on the correctness of this translation process.
- Flit Mode and Non-Flit Mode employ different replay mechanisms. Verification must account for these differences to ensure proper error handling and data integrity during retries.

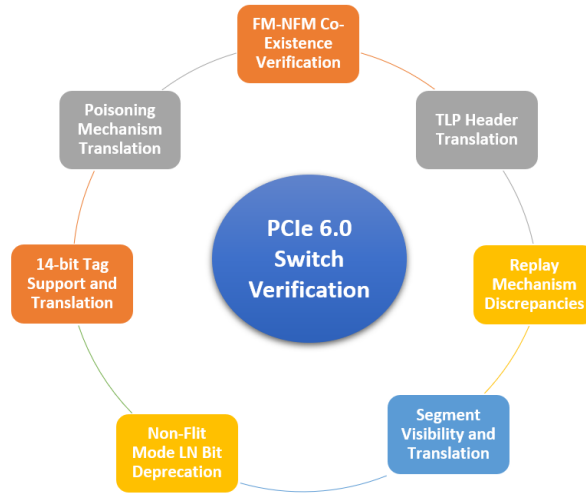


Figure 1. Flit Mode & Non-Flit Mode Interoperability

- The “Segment” information in packet header is only readily accessible in Flit-Mode links. This adds complexity when routing between Flit-Mode and Non-Flit Mode TLPs, especially when ingress and egress ports operate in different modes. The configuration space “*Device 3 capability*” register has a “Segment Captured” bit; there are specific translation rules for switch while translating a TLP from NFM to FM w.r.t segment captured bit. A Switch for which “Segment Captured” is set must handle as a TLP Translation Egress Blocked error a Non-Posted Memory request received at the upstream port destined for a downstream port in NFM that includes a “Requester Segment” that does not match the Switch’s captured Segment value.
- The Lightweight Notification (LN) bit used in Non-Flit Mode, is deprecated in PCIe 6.0 specification [1]. Verification should encompass dedicated translation rules for handling this deprecated bit.
- The 14-bit tagging scheme is exclusively for Flit-Mode [1]. Any Non-zero Tag[13:10] transaction is blocked by switch design when targeted for Non Flit-Mode. There are specific error reporting rules if violation observed at Switch port when the transaction is considered for posted vs non-posted requests.
- If either poisoning mechanism is applied to FM TLP during translation to NFM, the NFM TLP’s Error Poisoned bit (EP bit) must be set. Conversely, while translating from NFM to FM, if the EP bit is set in the NFM TLP, it must be preserved in the resulting FM TLP. Verification ensures the switch adheres to these translation rules for poisoning mechanisms.

B. Verification of Reserved TLPs in Flit-Mode

The PCIe 6.0 specification [1] introduces a concept called “Reserved” TLP types. These are essentially placeholders for future functionality. Each reserved TLP type has a defined header size, flow control mechanism, data payload presence, and routing behavior. This provision ensures that the PCIe 6.0 switch designs will be compatible with future versions of the specification, even if those reserved TLP types are redefined with specific functionalities. In simpler terms, PCIe 6.0 leaves room for growth by anticipating future expansion without requiring hardware upgrades for existing switches. It is important to cover those “Reserved” TLPs in switch verification by generating relevant traffic from Root Complex/Endpoints Verification IP (VIP).

C. Performance penalties to consider

While applying these translation rules and different shared/dedicated credit management, switch design may add some performance penalties by adding extra No Payload (NOP) Flits or NOP TLPs in Flit Mode and may add extra Logical Idle (IDL) DWORDs in Non-Flit Mode. This is why performance testing is now more crucial as part of switch design verification signoff criteria.

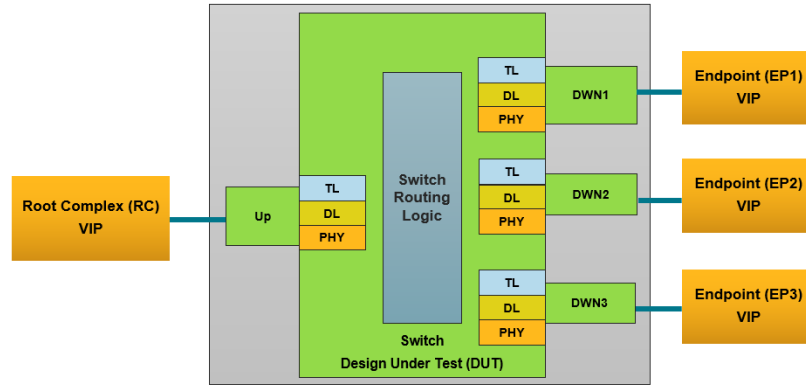


Figure 2. Conventional Switch with 1 Upstream port and 3 Downstream Ports

As shown in Figure 2, conventional switch port consists of Transaction layer (TL), Data Link layer (DL) and Physical Layer (PHY). Communication between different ports handled by Switch routing logic which is generally implementation specific.

Let's take an example of a DUT bug (corner case) where it has observed performance issue in Non-Flit Mode.

- In Figure 2 switch topology Upstream port is connected to Root complex (RC) VIP at Gen6 speed Flit Mode and Downstream port no. 1 (DWN1 in Figure 2) is connected to Endpoint (EP) VIP at Gen3 speed Non-Flit Mode. The configured lane count at each switch port is x16.
- RC VIP is sending backpressure traffic (back-to-back packets) with 1000 packets of Memory Write (MWr) TLPs to EP VIP via Switch upstream port in Flit Mode at Gen6 speed. The Switch routing logic is sending this traffic to EP VIP via Switch Downstream port in Non-Flit Mode at Gen3 speed.
- The length of each MWr packet is fixed to 16 Bytes of header with 16 Bytes of payload. So, actual MWr packet arrived at PHY layer become 40 Bytes (16B Header + 16B Payload + 4B STP + 4B LCRC) including all framing tokens. So, in this scenario the goal is to measure switch downstream port performance to transfer 40 KBytes of data to EP VIP.
- In case of x16 lane connection at Gen3 Speed in Non-Flit Mode with 128b/130b encryption, these bytes are sent across the lanes in sets of 16 symbols in each Data Block (Stream). So, in each data block 256 Bytes of data can be transferred.
- As per PCIe specification [1] SKIP or SKP Ordered Sets are used to compensate differences in frequencies between bit rates at two ends of a Link, so transmitters are required to send SKP Ordered Sets periodically.
- At one of the timestamps, Switch Downstream port got condition for transmitting SKP Ordered Set scheduled in the next Ordered Set Block. This condition happened when PHY layer has already put the last 16 Bytes of MWr TLP, which was spread across Lane0 to Lane15 at symbol 0 as shown in Figure 3. In this Data Block, still 236 Bytes of data can be accommodated before EDS framing token.
- In this scenario, Switch Downstream port should accommodate 4 more MWr TLPs (each of 40 Bytes of size including framing tokens) in this Data Block before sending SKP Ordered Set in the next Ordered Set Block. But, as shown in Figure 3 the DUT was having bug that it was transmitting unwanted extra Logical Idle (IDL) DWORDs in place of 5 MWr TLPs data.



Figure 3. Switch DUT Downstream Port Buggy behavior (left) Vs Expected behavior (right)

- As per PCIe specification [1], the rationale behind the DUT issue was condition mentioned in section “Transmitter Framing Requirements in Non-Flit Mode”. This logic was not developed efficiently to accommodate leftover Data Bytes from PHY layer buffer (i.e., 5 MWr TLPs) and considered SKP Order Set scheduling event with “not transmitting TLP” condition which cause unwanted IDL tokens to be added across the lanes in the relevant Data Block.

Transmitter Framing Requirements in Non-Flit Mode [1]:

- To Transmit a SKP Ordered Set within a Data Stream:
 - Transmit an EDS Token in the last DW of the current Data Block. For example, the Token is transmitted on Lane 0 in Symbol Times 12-15 of the Block for a x1 Link, and on Lanes 12-15 of Symbol Time 15 of the Block for a x16 Link.
 - Transmit the SKP Ordered Set following the current Data Block.
- The IDL Token must be transmitted on all Lanes when not transmitting a TLP, DLLP, or other Framing Token.

III. IMPLEMENTATION AND RELATED WORK

A. Switch DUT Verification Topology

In this topology, RC VIP is connected to Switch Upstream Port with serial interface (or PIPE) and Switch Downstream Ports are connected to different EP VIPs with serial interface (or PIPE). Figure 4 illustrates one sample scenario with mixed mode traffic (FM and NFM) on Switch ports containing one upstream and three downstream ports having different speed configurations.

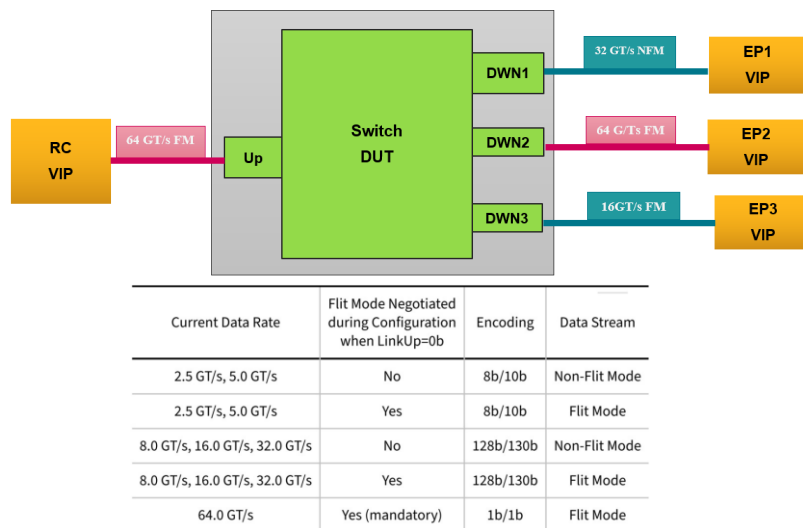


Figure 4. Switch DUT verification topology with 1 upstream and 3 downstream ports with different encoding scheme and different speed configuration on each link

B. Traffic modeling for Data Integrity and Interoperability Verification

The Introduction section of this paper highlights the scenarios that a verification engineer should target for simulate the traffic pattern and validate those translation rules and data integration rules either by Verification IP’s in-built assertions or by writing dedicated checkers in the relevant testcase. Testing strategies need to be evaluated against random factors such as different speed, different link width, along with different TLP type traffic generation (including “Reserved” TLPs) to make sure all the switch ports are verified with Flit Mode and Non-Flit Mode.

C. Performance Metrics Evaluation

Performance evaluation of Switch DUT seems very complex at first place considering the different possible Link width, Link speed and Mode of operations (FM/NFM). This paper proposes a simplified solution leveraging few performance measurement utilities (Performance Banner and Flit TLP Tracker) that a verification engineer can design with help of Verification IPs.

The main idea behind performance metrics measurement starts with Traffic modeling. Usually, the processor supports fixed size cache lines (e.g., 128 Bytes, 64 Bytes, 32 Bytes, 16 Bytes). The verification strategy should adopt the same fixed size TLP traffic generation for performance testing. When these fixed size TLP traffic with back pressure (multiple back-to-back TLPs) applied on the Switch DUT for different Link Widths, Link Speeds, different TLP types and in different operation modes (FM/NFM), it gives valuable insights into the switch's capability to handle various traffic patterns and stress conditions. This allows us to evaluate metrics like latency and throughput across these different scenarios, ensuring the switch performs as expected under real-world workloads.

Table I shows bandwidth and data rates comparison. It also demonstrates data latency for 256 Bytes in Flit mode across different link width combinations.

Table I. Data Rate Vs Bandwidth Vs 1 Flit Latency (with respect to possible link width)

Data Rates	Encoding	Bandwidth in GB/s (Unidirectional: TX or RX)							1 Flit Latency in Flit Mode (nsec)				
		x1	x2	x4	x8	x12 (NFM)	x16	x32 (NFM)	x1	x2	x4	x8	x16
2.5 GT/s	8b/10b	0.25	0.5	1	2	3	4	8	1024	512	256	128	64
5.0 GT/s	8b/10b	0.5	1	2	4	6	8	16	512	256	128	64	32
8.0 GT/s	128b/130b	1	2	4	8	12	16	32	256	128	64	32	16
16.0 GT/s	128b/130b	2	4	8	16	24	32	64	128	64	32	16	8
32.0 GT/s	128b/130b	4	8	16	32	48	64	128	64	32	16	8	4
64.0 GT/s	1b/1b	8	16	32	64	N/A	128	N/A	32	16	8	4	2

1) Performance Banner for Flit Mode:

```

1 *Denali* *****
2 *Denali* * Performance Measurement Report
3 *Denali* * Instance : uvm_test_top.pcie0v00.swmnv0.activeSw[1]
4 *Denali* * Start time: 19363.83103 ns (User)
5 *Denali* * End time: 19475.79519 ns (User)
6 *Denali* * Total time: 111.964 ns (User)
7 *Denali* *****
8 *Denali* *****
9 *Denali* *****
10 *Denali* *****
11 *Denali* [ 0] Rx_L0 3236.32148 ns 16.52%
12 *Denali* [ 1] Tx_L0 3236.32148 ns 16.52%
13 *Denali* [ 2] Rx_L0p 0 ns --
14 *Denali* [ 3] Tx_L0p 0 ns --
15 *Denali* [ 4] Link_L1 0 ns --
16 *Denali* [ 5] Link_L2 0 ns --
17 *Denali* [ 6] In Training 6334 ns 32.34%
18 *Denali* [ 7] In Recovery 10012.18955 ns 51.12%
19 *Denali* [ 8] In DL Up 13248.51103 ns 67.65%
20 *Denali* [ 9] Power_D0 19982.51103 ns 100.0%
21 *Denali* [10] Power_D1 0 ns --
22 *Denali* [11] Power_D2 0 ns --
23 *Denali* [12] Power_D3 0 ns --
24 *Denali* *****
25 *Denali* *****
26 *Denali* *****
27 *Denali* *****
28 *Denali* [ 0] Port_0 # of Configured Lanes 8 --
29 *Denali* [ 1] Port_0 # of TLP (TX) 200 --
30 *Denali* [ 2] Port_0 # of TLP (RX) 0 --
31 *Denali* [ 3] Port_0 # Outbound data 6400 Byte 57.14 GByte/sec
32 *Denali* [ 4] Port_0 # Inbound data 0 Byte --
33 *Denali* [ 5] Port_0 # of Payload Flits (TX) 28 --
34 *Denali* (User configured Payload Flits)
35 *Denali* [ 6] Port_0 # of Payload Flits (RX) 0 --
36 *Denali* (User configured Payload Flits)
37 *Denali* [ 7] Port_0 # of NOP Flits (TX) 0 --
38 *Denali* (Between user configured Payload Flits)
39 *Denali* [ 8] Port_0 # of NOP Flits (RX) 0 --
40 *Denali* (Between user configured Payload Flits)
41 *Denali* [ 9] Port_0 # of NOP TLPs (TX) 0 --
42 *Denali* (Between user configured Payload Flits)
43 *Denali* [10] Port_0 # of NOP TLPs (RX) 0 --
44 *Denali* (Between user configured Payload Flits)
45 *Denali* [11] Port_0 # Speed (on last Transaction) gen6 --
46 *Denali* *****
  
```

Figure 5. Performance Banner for Gen6 Flit Mode

Flit TLP Tracker at Gen6:

1279 E H Debug: [port_0] PL_TX_end_packet PIFlit(Nop)	1319	1359
1280 -19347.83613 ns -	1320 Packet: 33(96 to 111)	1360 Packet: 41(224 to 235)
1281 E H Debug: [port_0] PL_TX_end_packet PIFlit(Nop)	1321 ----	1361 ----
1282 -19351.83487 ns -	1322 20 00 00 01 00 00 00 06	1362 20 00 00 01 00 00 00 0E
1283 E H Debug: [port_0] PL_TX_end_packet PIFlit(Nop)	1323 00 00 02 00 00 10 00 1C	1363 00 00 02 00
1284 -19355.83359 ns -	1324	1364 -19367.82975 ns -
1285 E H Debug: [port_0] PL_TX_end_packet PIFlit(Nop)	1325 Packet: 34(112 to 127)	1365 E H Debug: [port_0] PL_TX_end_packet PIFlit(payload)
1286 -19359.83231 ns -	1326 ----	1366 Packet: 41(0 to 3)
1287 E H Debug: [port_0] PL_TX_end_packet PIFlit(Nop)	1327 20 00 00 01 00 00 00 07	1367 ----
1288 -19363.83103 ns -	1328 00 00 02 00 00 10 00 20	1368 00 10 00 3C
1289 E H Debug: [port_0] PL_TX_end_packet PIFlit(payload)	1329	1369 Packet: 42(4 to 19)
1290 Packet: 27(0 to 15)	1330 Packet: 35(128 to 143)	1370 ----
1291 ----	1331 ----	1371 20 00 00 01 00 00 00 0F
1292 20 00 00 01 00 00 00 00	1332 20 00 00 01 00 00 00 08	1372 00 00 02 00 00 10 00 40
1293 00 00 02 00 00 10 00 04	1333 00 00 02 00 00 10 00 24	1373
1294	1334	1374 Packet: 43(20 to 35)
1295 Packet: 28(16 to 31)	1335 Packet: 36(144 to 159)	1375 ----
1296 ----	1336 ----	1376 20 00 00 01 00 00 00 10
1297 20 00 00 01 00 00 00 01	1337 20 00 00 01 00 00 00 09	1377 00 00 02 00 00 10 00 44
1298 00 00 02 00 00 10 00 08	1338 00 00 02 00 00 10 00 28	1378
1299	1339	1379 Packet: 44(36 to 51)
1300 Packet: 29(32 to 47)	1340 Packet: 37(160 to 175)	1380 ----
1301 ----	1341 ----	1381 20 00 00 01 00 00 00 11
1302 20 00 00 01 00 00 00 02	1342 20 00 00 01 00 00 00 0A	1382 00 00 02 00 00 10 00 48
1303 00 00 02 00 00 10 00 0C	1343 00 00 02 00 00 10 00 2C	1383
1304	1344	1384 Packet: 45(52 to 67)
1305 Packet: 30(48 to 63)	1345 Packet: 38(176 to 191)	1385 ----
1306 ----	1346 ----	1386 20 00 00 01 00 00 00 12
1307 20 00 00 01 00 00 00 03	1347 20 00 00 01 00 00 00 0B	1387 00 00 02 00 00 10 00 4C
1308 00 00 02 00 00 10 00 10	1348 00 00 02 00 00 10 00 30	1388
1309	1349	1389 Packet: 46(68 to 83)
1310 Packet: 31(64 to 79)	1350 Packet: 39(192 to 207)	1390 ----
1311 ----	1351 ----	1391 20 00 00 01 00 00 00 13
1312 20 00 00 01 00 00 00 04	1352 20 00 00 01 00 00 00 0C	1392 00 00 02 00 00 10 00 50
1313 00 00 02 00 00 10 00 14	1353 00 00 02 00 00 10 00 34	1393
1314	1354	1394 Packet: 47(84 to 99)
1315 Packet: 32(80 to 95)	1355 Packet: 40(208 to 223)	1395 ----
1316 ----	1356 ----	1396 20 00 00 01 00 00 00 14
1317 20 00 00 01 00 00 00 05	1357 20 00 00 01 00 00 00 0D	1397 00 00 02 00 00 10 00 54
1318 00 00 02 00 00 10 00 18	1358 00 00 02 00 00 10 00 38	1398

Figure 8. VIP TX Path Flit TLP Tracker for FM Gen6 Speed

D. Testcase Development For Performance Verification

- Use Different TLP Types having fixed TLP lengths on specific link width (preferably as 128 Bytes, 64 Bytes, 32 Bytes and 16 Bytes).
 - Proposed TLP Types as Memory Read (MRd), Memory Write (MWr), Configuration Write (CfgWr), Message with and without Data (Msg/MsgD), Completions with Data (CplD), Fetch and Add AtomicOp Request (FetchAdd), Unconditional Swap AtomicOp Request (Swap), Compare and Swap AtomicOp Request (CAS), etc.
- Let's take an example for MWr 64-bit TLP request test scenarios with FM and NFM (launch traffic from RC/EP VIPs and Monitor traffic at respective switch egress port). As shown in below Table II, the verification engineer can repeat the same scenarios on different modes and on different Speeds.

Table II. Testcase list for MWr 64-bit TLP request on different speed and different operating modes

TLP Type	TLP length	Testcase Scenario names for MWr TLP at Gen6 Flit Mode (FM)				
		x1	x2	x4	x8	x16
MWr_64	32B	MWr_32B_1L_G6_FM	MWr_32B_2L_G6_FM	MWr_32B_4L_G6_FM	MWr_32B_8L_G6_FM	MWr_32B_16L_G6_FM
MWr_64	64B	MWr_64B_1L_G6_FM	MWr_64B_2L_G6_FM	MWr_64B_4L_G6_FM	MWr_64B_8L_G6_FM	MWr_64B_16L_G6_FM
MWr_64	128B	MWr_128B_1L_G6_FM	MWr_128B_2L_G6_FM	MWr_128B_4L_G6_FM	MWr_128B_8L_G6_FM	MWr_128B_16L_G6_FM
Testcase Scenario names for MWr TLP at Gen5 Flit Mode (FM)						
MWr_64	32B	MWr_32B_1L_G5_FM	MWr_32B_2L_G5_FM	MWr_32B_4L_G5_FM	MWr_32B_8L_G5_FM	MWr_32B_16L_G5_FM
MWr_64	64B	MWr_64B_1L_G5_FM	MWr_64B_2L_G5_FM	MWr_64B_4L_G5_FM	MWr_64B_8L_G5_FM	MWr_64B_16L_G5_FM
MWr_64	128B	MWr_128B_1L_G5_FM	MWr_128B_2L_G5_FM	MWr_128B_4L_G5_FM	MWr_128B_8L_G5_FM	MWr_128B_16L_G5_FM
Testcase Scenario names for MWr TLP at Gen5 Non-Flit Mode (NFM)						
MWr_64	32B	MWr_32B_1L_G5_NFM	MWr_32B_2L_G5_NFM	MWr_32B_4L_G5_NFM	MWr_32B_8L_G5_NFM	MWr_32B_16L_G5_NFM
MWr_64	64B	MWr_64B_1L_G5_NFM	MWr_64B_2L_G5_NFM	MWr_64B_4L_G5_NFM	MWr_64B_8L_G5_NFM	MWr_64B_16L_G5_NFM
MWr_64	128B	MWr_128B_1L_G5_NFM	MWr_128B_2L_G5_NFM	MWr_128B_4L_G5_NFM	MWr_128B_8L_G5_NFM	MWr_128B_16L_G5_NFM
.... Similar testcases for Gen4 FM, Gen4 NFM, Gen3 FM, Gen3 NFM, Gen2 FM, Gen2 NFM, Gen1 FM and Gen1 NFM						

E. Performance Banner Development

- Critical items to identify in Flit Mode Performance Banner is to have number of TX/RX Payload Flits and number of TX/RX NOP Flits. This can be natively supported in any verification IP with some register configuration control mechanism to enable/disable performance capture timing or this can be simply developed at testbench level through callback mechanism as shown in below sample Figure 9. The callback analysis port function is developed to capture TX NOP Flits and TX Payload Flits.

```

84 virtual function void write_rc_active_PL_TX_end_packet (denaliPciePacket trans);
85   denaliPciePlFlitPacket pl_flit;
86
87   if($cast(pl_flit,trans))
88   begin
89     if((pl_flit.PlFlitType inside {PCIE_PL_FLIT_Payload,PCIE_PL_FLIT_Nop}) &&
90       pcieSve0.SwUpsEnv0.activeRc.nop_flit_recording_start == 1)
91     begin
92       if(pl_flit.PlFlitType == PCIE_PL_FLIT_Payload)
93         payload_flit_count ++;
94       if(pl_flit.PlFlitType == PCIE_PL_FLIT_Nop && payload_flit_count > 0)
95         nop_flit_count ++;
96     end
97   end
98 end
99
100 endfunction : write_rc_active_PL_TX_end_packet
  
```

Figure 9. Cadence VIP Analysis Port Callback Function logic for capturing TX NOP Flits and TX Payload Flits

- Similar callback approach can be made available for capturing NOP TLPs for Flit Mode or this count can be calculated using below equation (1),

$$NOP\ TLPs = ((Number\ of\ Payload\ Flits * 236\ Bytes) - (Outbound/Inbound\ Data\ in\ Bytes)) / 4\ Bytes \quad (1)$$

- The verification engineer can enhance the Performance Banner to capture other Other performance metrics including Initial credits (TX/RX), Average round trip delay for Non-Posted request, Max Round Trip delay for Non-Posted request, Max TLP to Ack delay, Max TLP to NAK delay, IDLE DWORD Count as Inter packet gap for NFM etc. Those can be implemented using similar callback logic in the VIP.

IV. CONCLUSION

This paper outlines a comprehensive verification strategy for PCIe 6.0 Switch DUT which includes FM/NFM interoperability checks and reserved TLP verification in FM. Additionally, we emphasize the importance of performance testing in the PCIe 6.0 Switch DUT verification process. A case study demonstrates how our proposed traffic modeling approach can uncover corner-case bugs. Furthermore, we discuss latency monitoring techniques using performance measurement utilities like Performance Banner, along with implementation guidelines. These findings and methodologies, while demonstrated with Cadence Verification IP, extend beyond the realm of IP verification, and hold value for pre-silicon (emulation) and post-silicon validation as well.

V. ABBREVIATIONS USED

A. Abbreviations and Acronyms

Flit: Flow Control Unit

NOP: No Operation

TX: Transmission path

RX: Reception path

SKP: Skip Ordered Set

DWORD: Double Word represents a 32-bit unsigned integer

ACKNOWLEDGMENT

We would like to thank our employer, Cadence Design systems India Pvt. Ltd. for supporting this paper. We would also like to express our sincere appreciation to the organizers of the Design and Verification Conference (DVCON) for their valuable feedback and the opportunity to present our research at this esteemed forum.

REFERENCES

- [1] PCI Express® Base Specification Revision 6.2