2025
DESIGN AND VERIFICATION™
DVCON
CONFERENCE AND EXHIBITION
EUROPE
MUNICH, GERMANY
OCTOBER 14-15, 2025

ACT with Confidence: Formal Verification of Packet Based Designs using Array Centric Tracking

Anka Babu Appikatla, Cadence Design Systems, Bengaluru, India (ankababu@cadence.com)

Sakthivel Ramaiah, Cadence Design Systems, Bengaluru, India (sramaiah@cadence.com)

***Abstract*— Accurate data flow is the backbone of any reliable hardware design. Ensuring data integrity by preventing corruption, duplication, dropping, or reordering is key to validate system correctness. However, verifying data integrity using Formal Verification (FV) becomes highly challenging in packet-based designs where complex packing rules, shifting, and alignment introduce possibilities of subtle and hard-to-detect issues. The challenge intensifies in PCIe 6.0, where TLP Bytes are packed into Flits under strict packet packing rules involving frequent data shifting and alignment. Traditional FV data integrity techniques ineffective when designs modify or partially shift input data before sending it to output. In this paper, we present our array based novel data integrity approach called Array Centric Tracking (ACT), a scalable technique that tracks input TLP Bytes across Flits to validate data integrity and packing rules. We demonstrate various applications, benefits of ACT and caught 38 bugs using our approach including many subtle corner cases that traditional verification methods failed to catch.**

Keywords*— *Data Integrity, Flit, PCIe, Packet Based Designs, Formal Verification, Array Centric Tracking

I. Introduction

Flow Control Unit (Flit) Mode introduced in PCIe 6.0 specification as a new data stream mode. Flit has a fixed 256-byte length of size which consists of 236-bytes for Transaction Layer Packets (TLP), 6-bytes for Data Link Layer Payload (DLP), 8-bytes for Cyclic Redundancy Check (CRC) and 6-bytes for Forward Error Correction (FEC) as shown in Figure-1.

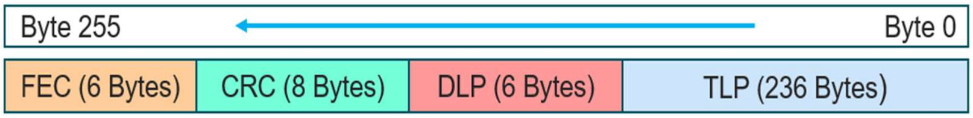


Figure 1: Flit Structure – 256 Bytes

The Flit Encoder (FE) is a complex and timing critical module and responsible for efficiently packing the 236 TLP Bytes in the Flit. It processes TLP data bus and control bus, which provides metadata like start of packet (SOP) and end of packet (EOP) data word (DW) positions, and Error TLP (Nullify and Poison) information.

TLP length varies from 3 DWs to 1032 DWs and a single TLP may span multiple Flits depending on its length and placement. Flit Encoder performs shifting and alignment of TLP DWs if Saved TLP DWs presents in design pipeline stages due to below

1. Allocating space for DLP, CRC and FEC Bytes.
2. Once a NOP (No operation) TLP is scheduled, it must continue until the next 4DW aligned boundary.
3. No more than 8 non - NOP TLPs, including partial TLPs in
    a. First half of the Flit i.e., the first 32DW (Bytes 0 through 127)
    b. Last 27 DWs of the Flit (Bytes 128 through 235)
4. Error TLP (Nullified or Poisoned) must be succeeded by only NOP TLPs through the end of the Flit.

Example: TLP Packing in Flits

Figure-2 shows an example of TLP packing in a Flit and how TLPs are shifted and aligned based on the above packing rules.

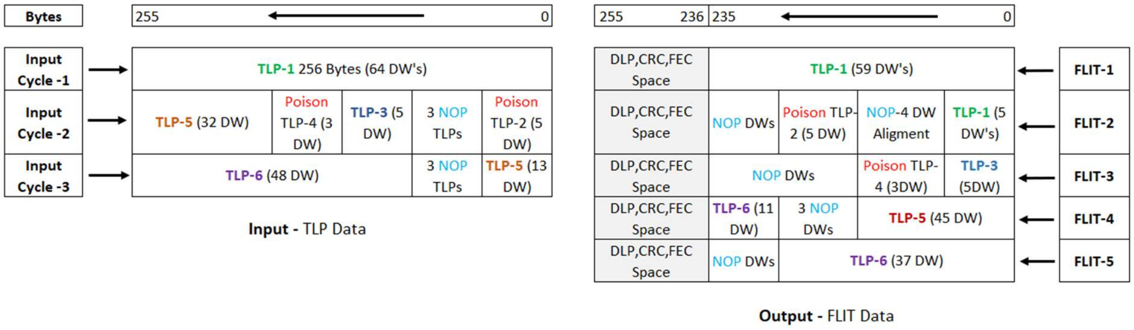


Figure 2: Output Flit's formation from multiple cycles of input TLP data

Input Cycle-1 contains TLP-1 with a length of 64 DWs. The encoder can only fill 59 DWs of TLP-1 into FLIT-1, so the remaining 5 DWs are placed in FLIT-2 (DW positions 0-4). Since TLP-1 ends at DW 4, the encoder inserts a NOP TLP at DW 5 to maintain NOP 4DW alignment, filling additional NOP DWs at positions 6 and 7. A Poisoned TLP (TLP-2) is placed in DW positions 8-12 of FLIT-2. The encoder fills the remainder of FLIT-2 with NOP TLPs to ensure that the Poisoned TLP is followed only by NOPs, as per the specification. The process continues for subsequent FLITs based on the next incoming TLPs.

II. Related work

The below Table-1 highlights various data integrity techniques used in Formal Verification (FV). Those work well when output data exactly matches the input in same order and content. Initially, the Non-Deterministic Constant [3] formal verification technique was applied to verify data integrity, but the assertions failed (false negatives) due to TLP data shifting and alignment caused by Flit packing rules.

We also explored other data-integrity techniques described in Table-1. Those work well when output data exactly matches the input in same order and content. But such techniques are not suitable for designs where part of the input data is shifted, and the output does not match the input directly.

Technique	Description
FIFO Method [1]	The FIFO method stores each input data element in a queue to track and sequentially compare it with output data to verify correctness and ordering.
Wolper Coloring Technique [1]	The input is constrained to follow the Wolper coloring sequence, verify if the same sequence is received at the output.
Formal Scoreboard [2]	FPV apps provide Formal Scoreboards. Scoreboard acts as a monitor of the DUT and embedded with all the required assertions for verification of end-to-end data integrity.
Non-Deterministic Constant (NDC) [3]	The NDC technique assertions use a variable whose value remains constant during a formal run. The assertions are triggered when the NDC variable value is seen at the input and verifies its appearance at the output after a defined delay.

Table 1: Various Formal Data-Integrity Techniques

III. Problem statement

The Flit packing rules introduce significant complexity which require extensive shifting and alignment of TLP data. This increases the risk of data integrity issues such as TLP DW corruption, duplication, dropping, and reordering. The Flit interface does not provide required TLP level details such as SOP DW, EOP DW, Nullify or Poison TLP end DW information. Without this information, it is impossible to verify the complex Flit packing rules.

Traditional FV Data-Integrity techniques are ineffective for Flit Encoder data integrity verification, where TLPs are packed into Flits following the specified rules require shifting and alignment of TLP data in Flit.

To address these challenges, a new FV approach is required to verify data integrity and confirm that packet packing rules are being followed as expected.

IV. Methodology

Array Centric Tracking (ACT) is a circular array-based approach designed to verify data integrity and the Flit packing rules of the Flit Encoder described earlier. ACT verifies data integrity at a DW (Data Word) level and tracks the start and end of each TLP to validate complex Flit rules. Overview of ACT method is shown in Fig 3.

The array size (N) is determined by the design configuration—for example, the data path width, number of pipeline stages, and other architectural factors.

The ACT method is divided into two parts: ACT Ingress Flow and ACT Egress Flow.

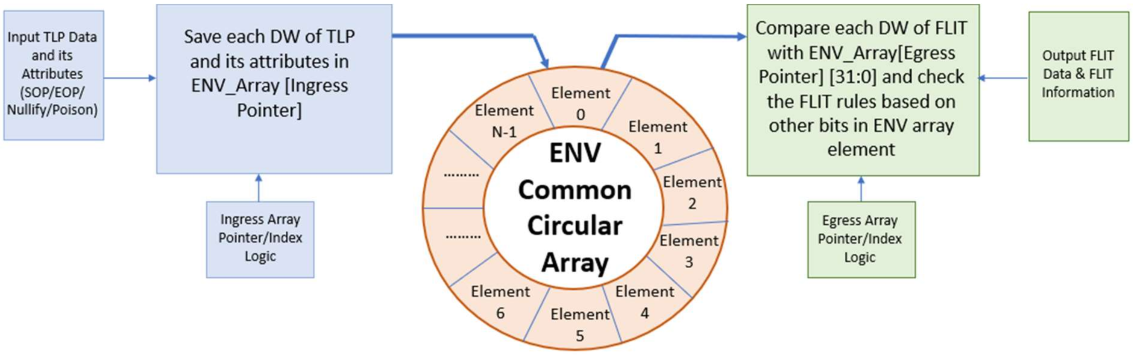
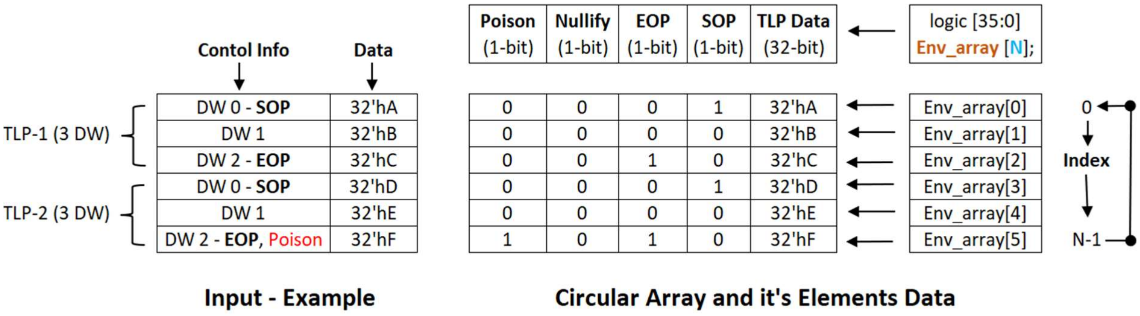

Input TLP Data and its Attributes (SOP/EOP/ Nullify/Poison)
Save each DW of TLP and its attributes in ENV_Array [Ingress Pointer]
Ingress Array Pointer/Index Logic
ENV Common Circular Array
Element 0
Element 1
Element 2
Element 3
Element 4
Element 5
Element 6
....
....
Element N-1
Compare each DW of FLIT with ENV_Array[Egress Pointer] [31:0] and check the FLIT rules based on other bits in ENV array element
Output FLIT Data & FLIT Information
Egress Array Pointer/Index Logic

Figure 3: Circular Array Based TLP Data and Attributes Tracking Process

In the ACT Ingress Flow, each DW of a non-NOP TLP is stored in the array, along with its attributes: SOP, EOP, Nullify, and Poison as illustrated in Figure-4.



Input - Example

	Contol Info	Data
TLP-1 (3 DW)	DW 0 - **SOP**	32'hA
	DW 1	32'hB
	DW 2 - **EOP**	32'hC
TLP-2 (3 DW)	DW 0 - **SOP**	32'hD
	DW 1	32'hE
	DW 2 - **EOP**, Poison	32'hF

Circular Array and it's Elements Data

logic [35:0]
Env_array [N];

Poison (1-bit)	Nullify (1-bit)	EOP (1-bit)	SOP (1-bit)	TLP Data (32-bit)		Index
0	0	0	1	32'hA	Env_array[0]	0
0	0	0	0	32'hB	Env_array[1]	
0	0	1	0	32'hC	Env_array[2]	
0	0	0	1	32'hD	Env_array[3]	
0	0	0	0	32'hE	Env_array[4]	
1	0	1	0	32'hF	Env_array[5]	N-1

Figure 4: Circular Array Data Formed from Inputs

The ACT Egress Flow in Figure-5 illustrates the verification logic implemented to detect data integrity issues and Flit packing rule violations on the output side of the design. The process begins by validating whether each Flit data word (DW) position falls within the range 0 to 58, ensuring it corresponds to the TLP DWs and not to DLP, CRC, or FEC. If the Flit DW data matches the corresponding entry in the ENV array, the flow proceeds to check whether it is a Start-of-Packet (SOP) or End-of-Packet (EOP). This check is essential for tracking TLP boundaries, updating packet counters, and validating the maximum TLPs per half-Flit and adherence to Nullify/Poison packing rules. In case of a mismatch, the logic first checks whether the Flit DW data is zero. A zero value may either represent a valid NOP or indicate a potential data integrity issue. To distinguish between these cases, the flow verifies whether the TLP has ended. If it has, the logic validates the NOP 4DW alignment rule. If not, the zero is interpreted as an unexpected gap between SOP and EOP, leading to a data integrity error. Any mismatched non-zero value is immediately treated as data corruption. Upon detecting a data integrity issue or Flit packing rule violation, the Egress Flow asserts the corresponding flag signals to indicate an error.

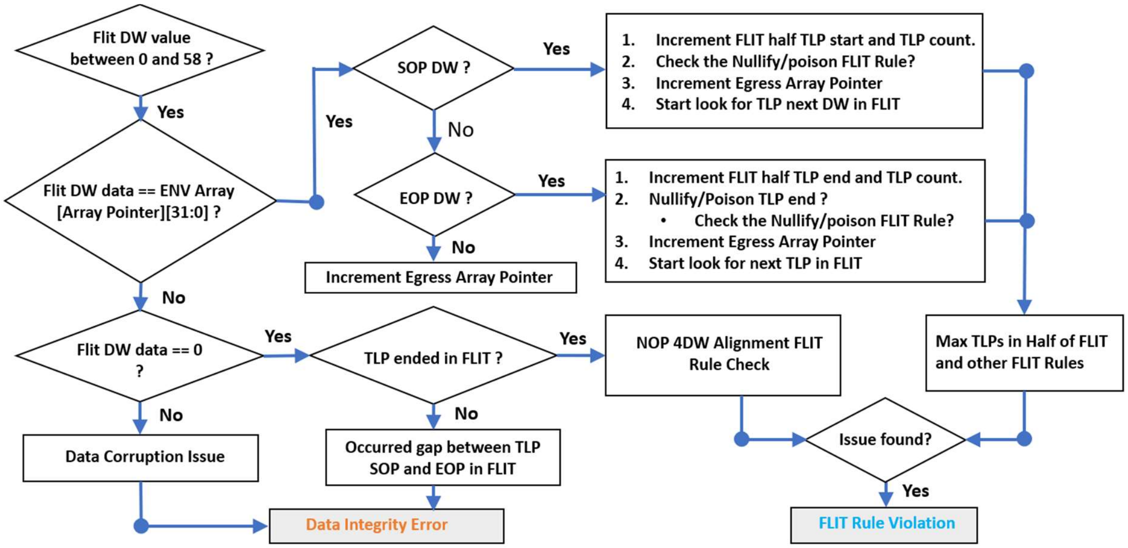

Flit DW value between 0 and 58 ?
Yes
Flit DW data == ENV Array [Array Pointer][31:0] ?
Yes
SOP DW ?
Yes
1. Increment FLIT half TLP start and TLP count.
2. Check the Nullify/poison FLIT Rule?
3. Increment Egress Array Pointer
4. Start look for TLP next DW in FLIT
No
EOP DW ?
Yes
1. Increment FLIT half TLP end and TLP count.
2. Nullify/Poison TLP end ?
• Check the Nullify/poison FLIT Rule?
3. Increment Egress Array Pointer
4. Start look for next TLP in FLIT
No
Increment Egress Array Pointer
No
Flit DW data == 0 ?
Yes
TLP ended in FLIT ?
Yes
NOP 4DW Alignment FLIT Rule Check
Max TLPs in Half of FLIT and other FLIT Rules
No
Data Corruption Issue
No
Occurred gap between TLP SOP and EOP in FLIT
Issue found?
Yes
Data Integrity Error
FLIT Rule Violation

Figure 5: ACT Egress Flow

V. Application

"**Array Centric Tracking**" is a versatile data integrity approach designed to handle complex Flit packing rules. It addresses the verification challenges and limitations of traditional FV techniques. ACT has been successfully applied to verify the PCIe PCS Elastic Buffer module, which involves shifting input data and checking output features based on input presence. As newer protocols like CXL, UCIe and UALink adopt Flit mode, the ACT method is well-suited for their verification too.

Advantages of ACT:

1. Applicable to any packet-based design requiring data integrity verification.
2. Easily adaptable to the spec updates. PCIe Gen7 specification reduces the allowed TLPs per half Flit from 8 to 4 – ACT handles this with only change the value in assertion, without altering the core approach.
3. Enables accurate checking of features that depend on part of input data presence in output data.
4. Effective even in verifying designs where traditional data integrity methods are already used.

VI. Results

Despite late deployment of FPV, we identified a total number of 38 bugs in the design, highlighting the efficiency of our Array Centric Tracking (ACT) approach in uncovering a variety of elusive design issues. The bugs found represents a spectrum of challenging corner scenarios, demonstrating the depth and thoroughness of our analysis. Figure-6 describes high-level categories of the bugs found by using ACT approach.

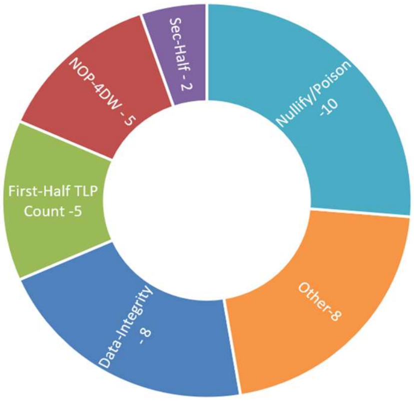


Figure 6: Categories & Number of Bugs Found

Flit Packing Rule Violation Bug: The CEX Figure-7 shows a violation of the maximum allowed TLPs in the first half of a Flit (0 to 127 Bytes) under a 512-bit data path, where 1 partial TLP and 8 fresh TLPs ended. For debugging, TLP SOP and EOP DWs are constrained to *32'haaaa_aaaa* and *32'hffff_ffff* respectively.

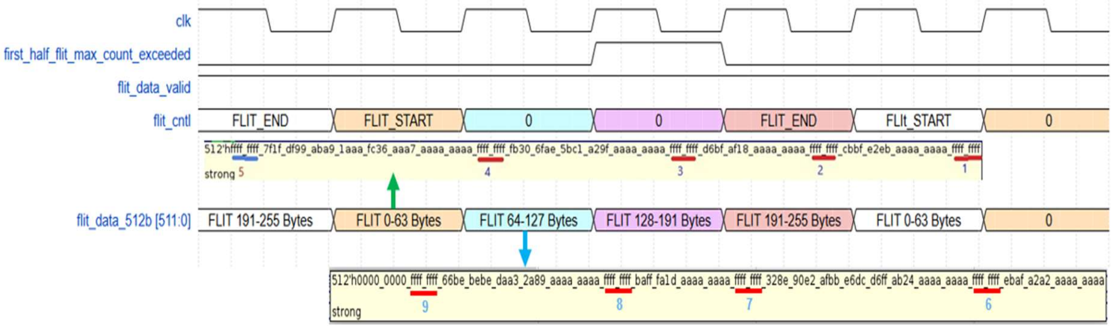

clk
first_half_flit_max_count_exceeded
flit_data_valid
flit_cntl
FLIT_END
FLIT_START
0
0
FLIT_END
FLIt_START
0
512'hffff_ffff_7f1f_df99_aba9_1aaa_fc36_aaa7_aaaa_aaaa_ffff_ffff_fb30_6fae_5bc1_a29f_aaaa_aaaa_ffff_ffff_d6bf_af18_aaaa_aaaa_ffff_ffff_cbbf_e2eb_aaaa_aaaa_ffff_ffff
strong
5
4
3
2
1
flit_data_512b [511:0]
FLIT 191-255 Bytes
FLIT 0-63 Bytes
FLIT 64-127 Bytes
FLIT 128-191 Bytes
FLIT 191-255 Bytes
FLIT 0-63 Bytes
0
512'h0000_0000_ffff_ffff_66be_bebe_daa3_2a89_aaaa_aaaa_ffff_ffff_baff_fa1d_aaaa_aaaa_ffff_ffff_328e_90e2_afbb_e6dc_d6ff_ab24_aaaa_aaaa_ffff_ffff_ebaf_a2a2_aaaa_aaaa
strong
9
8
7
6

Figure 7: CEX - Maximum TLP allowed in Flit First Half

Data Integrity Bug: The CEX Figure-8 highlights a data integrity issue where the DUT incorrectly inserted two NOP DWs before the TLP ended within the Flit (bytes 64 to 95) in a 256-bit data path. A partial TLP ended at the first DW of the cycle, TLP-5 began at the next DW, but two unexpected NOP DWs were inserted at DW positions 5 and 6, as highlighted.

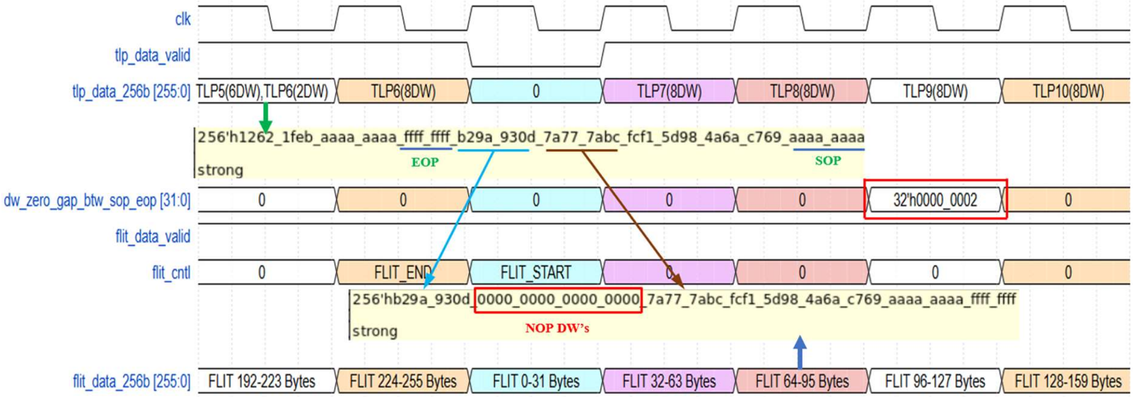


Figure 8: CEX – Data Integrity Issue

VII. Conclusion

Coverage-driven simulation has been the primary verification methodology for PCIe controllers from Gen 1 to Gen 7. Additionally, formal techniques are applied to complex modules across different layers of the PCIe, with block-level formal sign-off. The Flit Encoder module was introduced in Gen 6 and was initially verified using top-level simulation, which exposed several bugs. Formal verification began a year later. We successfully deployed circular array-based Array Centric Tracking (ACT) technique and benefited greatly from the results.

ACT successfully addresses the verification challenges in Flit Encoder design where traditional FV Data-Integrity techniques fall short. Its successful application led to the detection of 38 critical bugs, proving its strength in handling complex packet rules and corner cases. It ensures both data integrity and correct rule implementation, making it scalable for future protocol use.

References

[1] Ipshita Tripathi, Ankit Saxena, Anant Verma, Prashant Aggarwal, "The Process and Proof for Formal Sign-off A Live Case Study" DVCon US 2016.

[2] Vedprakash Mishra, Carlston Lim, Zhi Feng Lee, Jian Zhong Wang, Anshul Jain and Achutha KiranKumar V M, "OIL check of PCIe with Formal Verification" DVCon India 2022.

[3] Dr. Shahid Ikram, Mark Eslinger, "Demystifying Formal Testbenches: Tips, Tricks, and Recommendations" DVCon US 2023.