Optimizing Multi-protocol IEEE 802.3 PCS Verification with Synopsys Ethernet VIP

Edward Gibson, Marvell
Aparna Bharat, Marvell
Ashutosh Agarwal, Synopsys
Pushpal Nautiyal, Synopsys

Presenters

- Edward Gibson, Marvell, Senior Principal Design Verification Engineer
- Aparna Bharat, Marvell, Principal Design Verification Engineer

- Representing the Teralynx Data Center Switching L1 and L2 Hardware Design Verification (DV) team

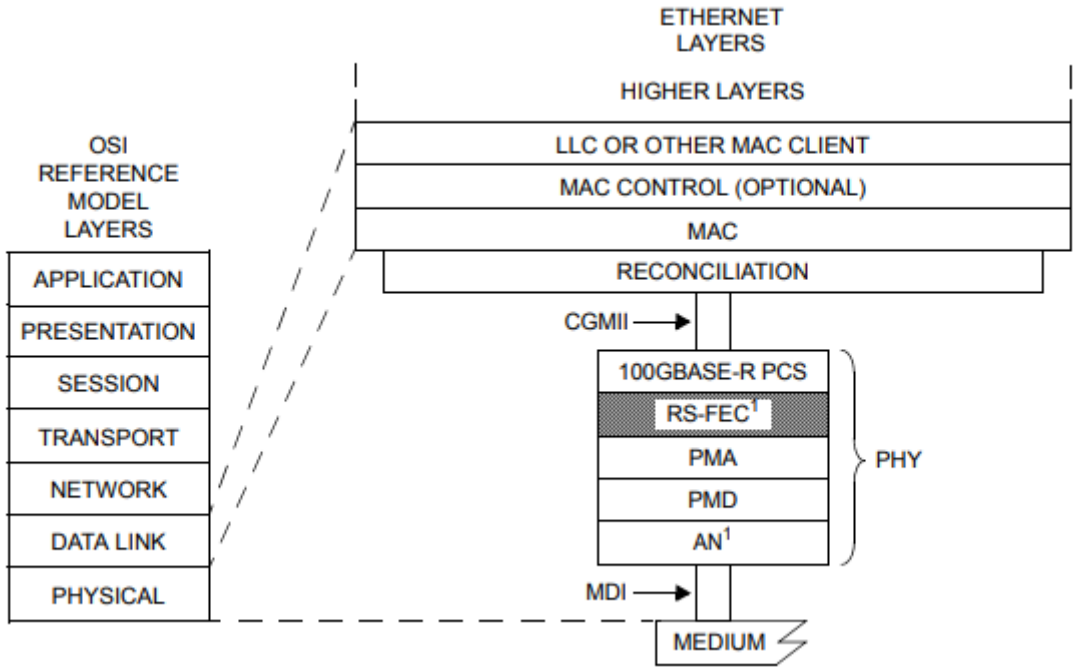
Agenda

- Ethernet Physical Coding Sublayer (PCS)
- Ethernet PCS Critical Features
- Marvell PCS Verification Strategy
- Marvell PCS Verification Examples
- Conclusions

Ethernet Physical Coding Sublayer (PCS)

Preparing data from the MAC for the Medium

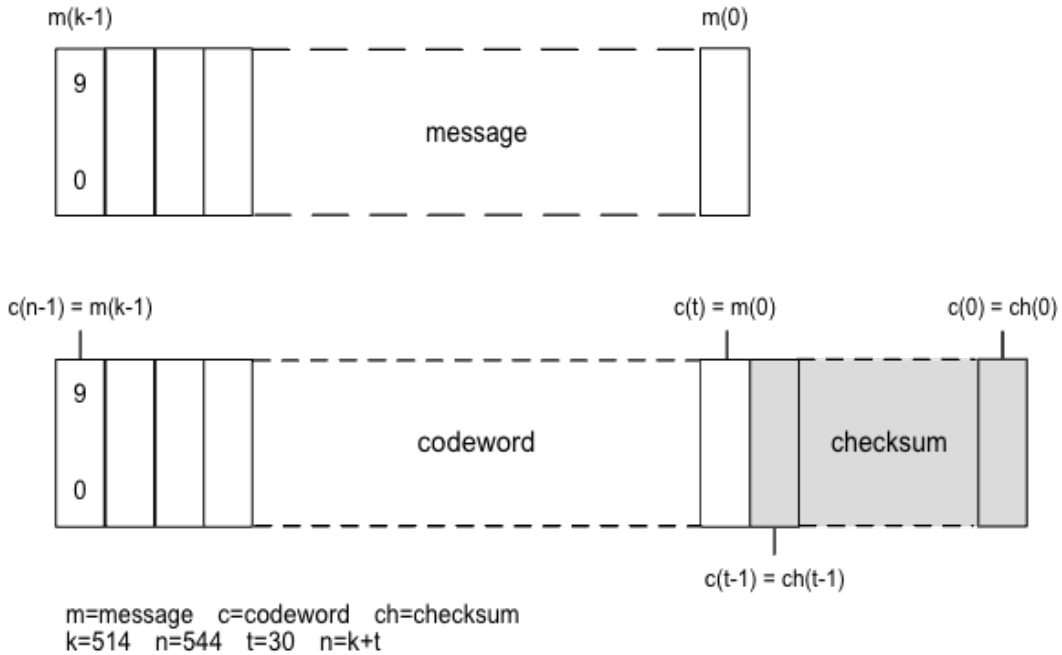
PCS within Ethernet Stack



RS-FEC relationship to the ISO/IEC Open Systems Interconnection (OSI) reference model and the IEEE 802.3 Ethernet model [1]

Evolution of High Speed IEEE PCS

- IEEE Clause 82 defined 40G to100G Ethernet Physical Coding Sublayer (PCS)
  - Required Bit Error Rate (BER) of 10^{-12}
- IEEE Clause 91 built on top of this introduced Reed Solomon Forward Error Correction (RSFEC) into the PCS
  - BER requirement drops to 10^{-4} allowing the physical medium to be far more error prone
  - RSFEC error correction is designed specifically for burst error correction



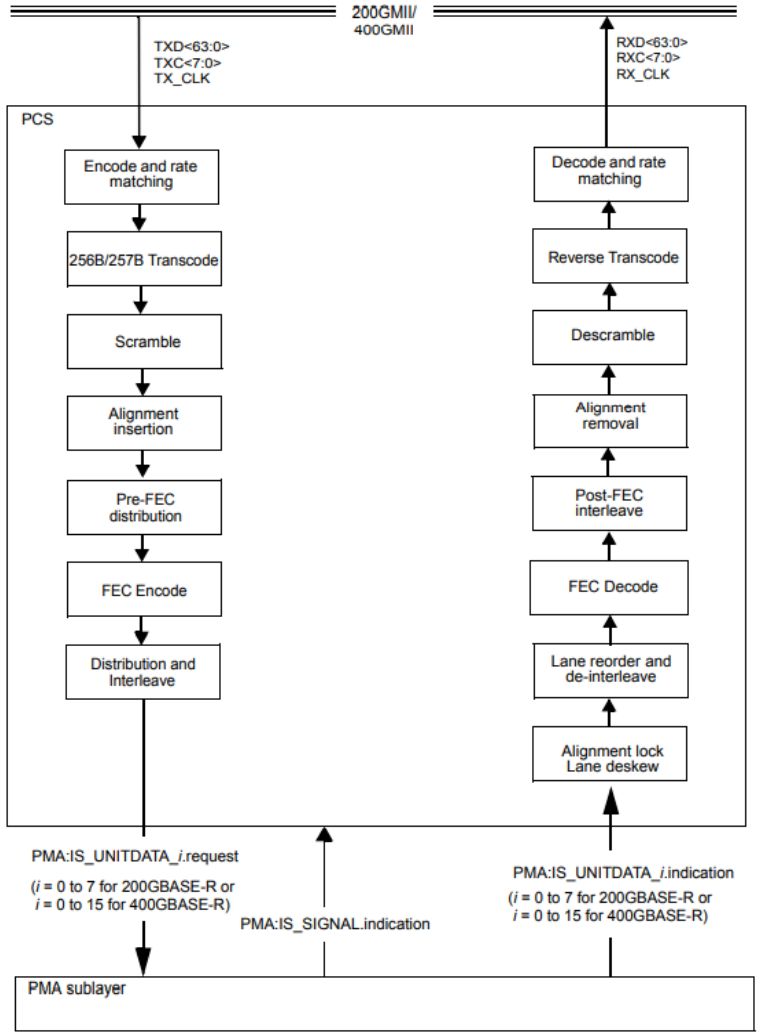
Evolution of High Speed IEEE PCS

- IEEE Clause 119 PCS defined 200G and 400G with interleaved RSFEC
  - 2 parallel RSFEC are present with the stream of symbols being round robin distributed
    - Adds further resilience to burst errors as now they are distributed across two correction engines
- IEEE Clause 172 PCS defined 800G
- 2 parallel 400G CL119 PCS
- IEEE Clause xxx PCS will define 1.6T
  - 4 parallel 400G CL119 PCS

Ethernet PCS Critical Features

Toughest to Verify Features

400G PCS operations



CL119 PCS functional block diagram [2]

Complex PCS Features for Design Verification

- Forward Error Correction (FEC) Rx Error correction (Example 1)
- FEC Rx Error counters (Example 1/2)
- FEC Rx Error indication (Example 2)
- Alignment marker error insertion (Example 3)
- Rx lane re-order and deskew (Example 4)
- High Symbol Error Rate (HiSER)
- IEEE CL82 PCS Encoder/Decoder

Marvell PCS Verification Strategy

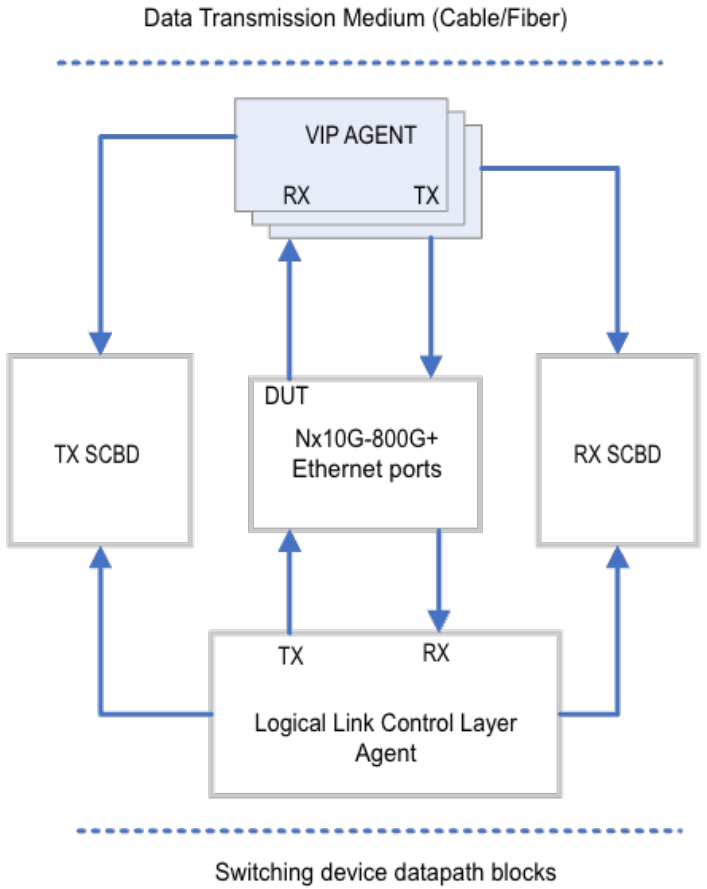
Testbench

Testbench Problem Statement

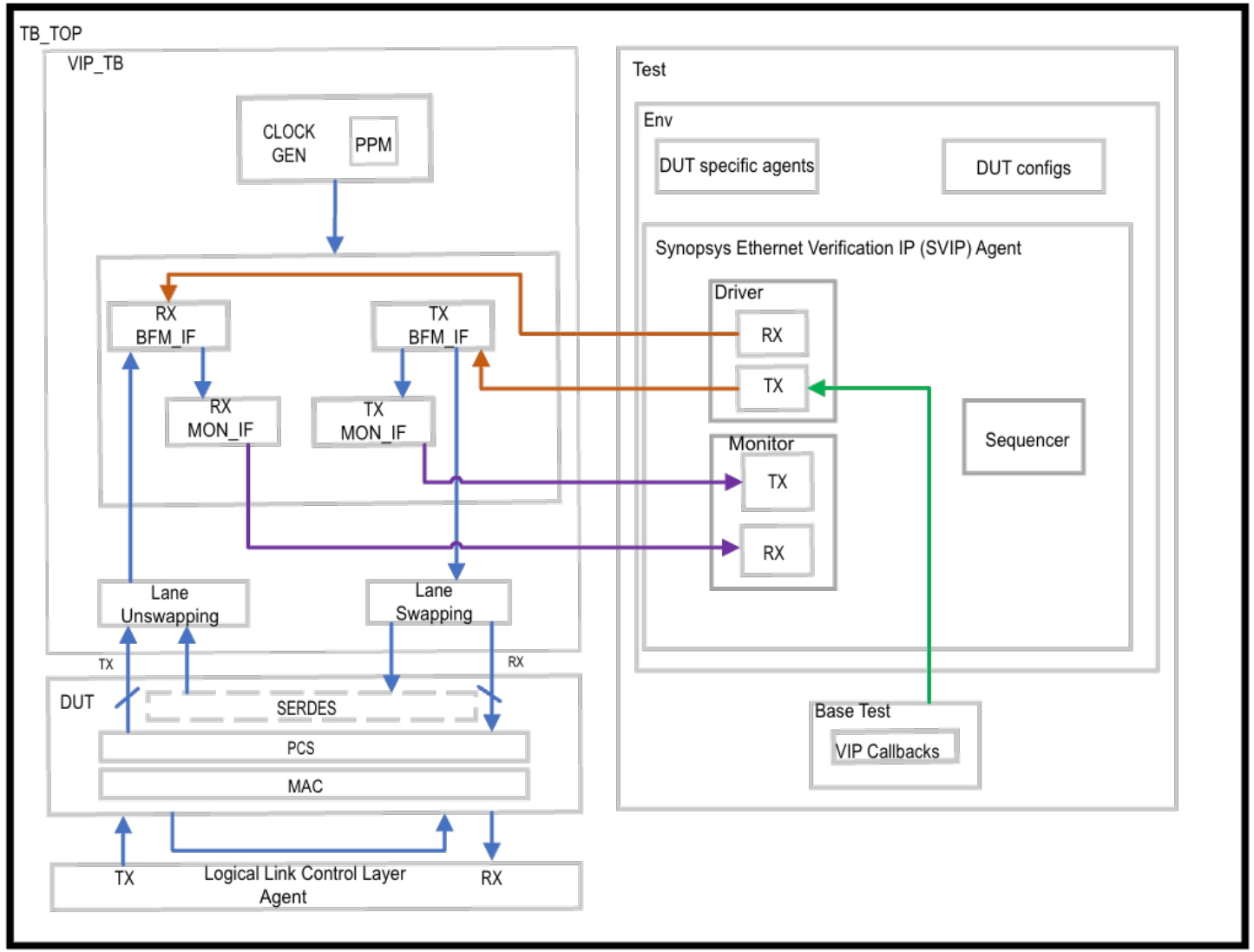
- Ethernet port is a duplex communication between local and link partner devices
- Data enters the design at a chosen speed and must exit in a lossless manner
- Verifying a local device requires a link partner RTL design or verification model
- A verification model for a link partner is a complex development task in articulating the IEEE features of Nx10G-800G+ ports
- This is especially true of a modern multi speed IEEE 802.3 PCS with RSFEC
- The PCS data is scrambled and RSFEC encoded to be non-human readable
- Development and maintenance of a model to support this design verification (DV) activity likely requires a dedicated team

TB Solution Statement

- The Marvell solution is to use the Synopsys Ethernet Verification IP (SVIP) as a link partner model to efficiently verify our IEEE 802.3 compliant IP



Testbench Structure



Key Verification Environment Features

- Design under test (DUT) can be MAC, MAC+PCS, or MAC+PCS+SERDES
- Test bench supports random per port Parts per Million (PPM) data clock
- PCS verification happens in a non-serdes test bench via the parallel interface between SVIP and DUT
- VIP monitor is used for protocol compliance checking on DUT TX traffic
- On the DUT RX path protocol violations are generated using SVIP callbacks with appropriate exceptions
- DUT counters+status are checked to validate the expected behavior for both compliance and non-compliance cases
- Scoreboard closes the loop on packet coherence

SVIP Rx Error Injection Callback Template

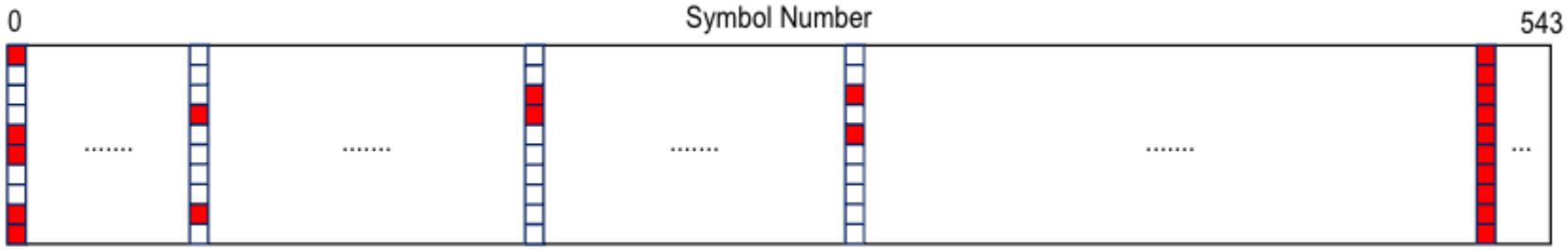
```
############################################################
class ethernet_user_inject_callbacks extends svt_ethernet_txrx_callback;

  virtual task svt_ethernet_txrx_rs_fec_encoder_start ( svt_ethernet_txrx driver , svt_ethernet_rs_fec_encoder_transaction xact ) ;

    static int packet_index = 0;
    svt_ethernet_rs_fec_encoder_transaction_exception_list  exception_list;
    svt_ethernet_rs_fec_encoder_transaction_exception  exception;
    <If_required_one_can_put_conditional_statements_for_selective_error_insertion>begin

    // Create the exception class
       exception = new("exception");

    // Create the exception list
      exception_list = new("exception_list",exception);

      exception.error_kind = <Program_based_on_desired_error_injection>;
      .....................................................................
     exception_list.add_exception(exception);
     xact.exception_list = exception_list;

  end
 endtask : svt_ethernet_txrx_rs_fec_encoder_start

endclass : ethernet_user_inject_callbacks
############################################################
```

Marvell PCS DV Examples

With SVIP Reference model

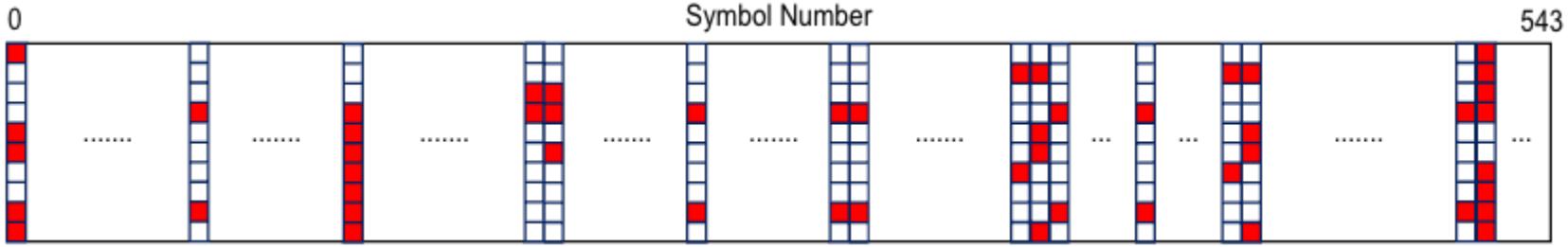
SVIP Callback Example1 - FEC Error Correction

- KPFEC CW is 5440 bits broken into 544 10bit symbols
- KPFEC corrects up to 15 10bit symbols per FEC CW
- Use callback to inject 0-15 symbol errors per codeword SVIP sends to DUT Rx
- Randomize 1-10bits inverted in each errored symbol
- Randomize location of errored symbols in full range 0-543
- Check 100% packet continuity at the switch side interface
- Verify corrected codeword and symbol histogram counters
- RSFEC correction enable/disable verification



SVIP Callback Example2 - FEC Uncorrected CWs

- KRPFEC uncorrectable CW has >15 symbol errors
- Using callback inject 16+ symbol errors to generate uncorrectable CWs
- Mix with Correctable CWs <16 symbol errors
- Avoid 3 uncorrectable CWs back to back on a single FEC engine
- Verify Correctable, uncorrectable CW counts and pcs errors (lossy scenario)
  - Counters must be atomic and clear-on-read
  - Clear counters continuously through test creating aggregate total for each
  - Check aggregate against callback injected counts in test check_phase
- Indication Enable Verification – 80 pcs errors injected per uncorrectable CW



SVIP Callback Example3 - Alignment Markers

- Alignment Markers (AMs) are used to align multi serdes lane ports
- For example, 400G R4 (4x100G serdes)
- 400G R4 internally consists of 16 25G PCS virtual lanes
- AMs are known patterns send periodically made up of:
  - Common bytes on all 16 lanes (6 Bytes or 12 4bit nibbles)
  - Unique bytes to each of the 16 lane (6 Bytes or 12 4bit nibbles)
- Link-up/link-down criteria of AMs is defined in the IEEE 802.3 Clause 119
  - Figure 119-12-Alignment marker lock state diagram

SVIP Callback Example3 - Alignment Markers

- AMs are not FEC protected because they are parsed by the PCS before correction
  - 9/12 Common nibbles must be correctly matching the standard AM for an AM match to be declared
  - 9/12 Unique nibbles must be correctly matching the standard AM for an AM match to be declared
- SVIP callback allows granular error insertion on AM nibbles

CM0	CM1	CM2	CM3	CM4	CM5			CM6	CM7	CM8	CM9	CM10	CM11			UM0	UM1	UM2	UM3	UM4	UM5			UM6	UM7	UM8	UM9	UM10	UM11

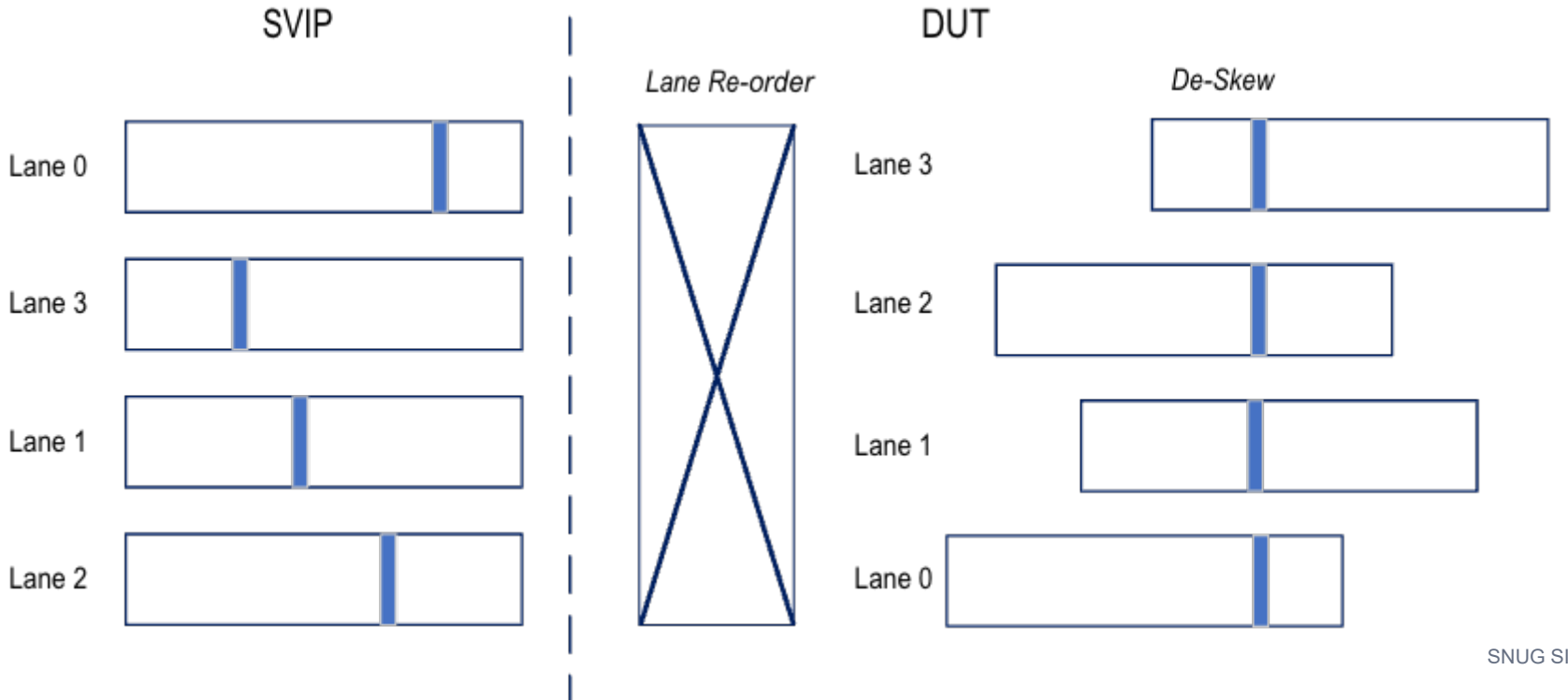
SVIP Callback Example3 - Alignment Markers

- 0-3 Common nibbles in error and 0-3 Unique nibbles in error

- 0-3 Common nibbles in error or 4-12 Unique nibbles in error

SVIP Skew and lane reorder stimulus - Example4

- Multi-lane PCS uses AMs to identify individual lanes and allow the bit stream to be re-merged
- Multi-lane PCS uses AMs to align the relative skew between the lanes which may have passed through different medium elements (cables/fibers)



Conclusions

Conclusions

- SVIP simple setup for any Ethernet mode with serial, parallel, or MII interface
  - Test MAC only, MAC+PCS or other combinations such as MAC+PCS+SERDES
- Interop with another interpretation of the standard
  - And one potentially other players in the market also interop with
- Maintenance of an in-house Ethernet VIP with the scope to cover many IEEE port configurations including the 'bleeding edge' speeds an expensive task
- SVIP Rx is a full PCS protocol checker
- SVIP error injection callbacks provide very powerful constrained random error injection for DUT Rx feature testing
- Approach has silicon proven success for 3 generations of Teralynx switches
  - Total of 0 PCS respins
  - Delivered to the industries most demanding data center customers

MARVELL snug

THANK YOU

YOUR
INNOVATION
YOUR
COMMUNITY

References

- IEEE 802.3 Clause 91 Figure 91-1, RS-FEC relationship to the ISO/IEC Open Systems Interconnection (OSI) reference model and the IEEE 802.3 Ethernet model [1]
- IEEE 802.3 Clause 119 Figure 119-2, Functional block diagram [2]