

High-Speed Interface IP Validation based on Virtual Emulation Platform

Jaehun Lee, Daeseo Cha, Sungwook Moon

Design Technology Team
Foundry Business
Samsung Electronics Co., Ltd.

Agenda

- Introduction
- The concept of Virtual Emulation
- Existing HSI IP validation flow
- Proposed HSI IP validation flow
- Method to accelerate pre-silicon validation
- How to debug in Virtual Emulation Platform
- Experiment results
- Conclusion

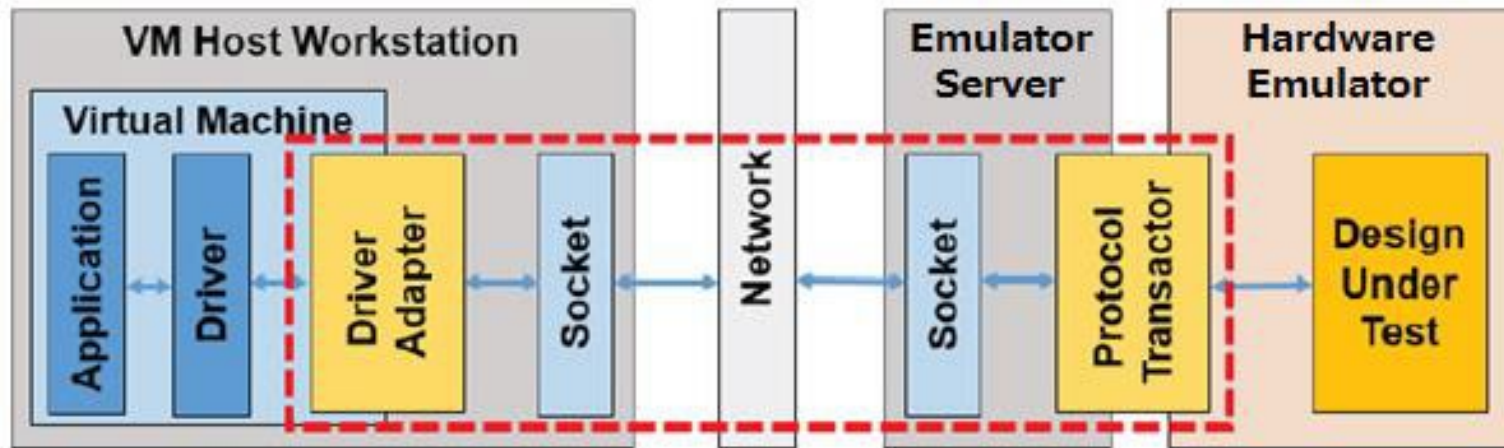
Introduction

- Modern SoC trend
 - Current electronic devices require large data storage to store numerous data and perform necessary calculations quickly
 - This requirement of high-speeds leads to wide usage of High-Speed Interface (HSI) IPs in various system-on-chip (SoC) products



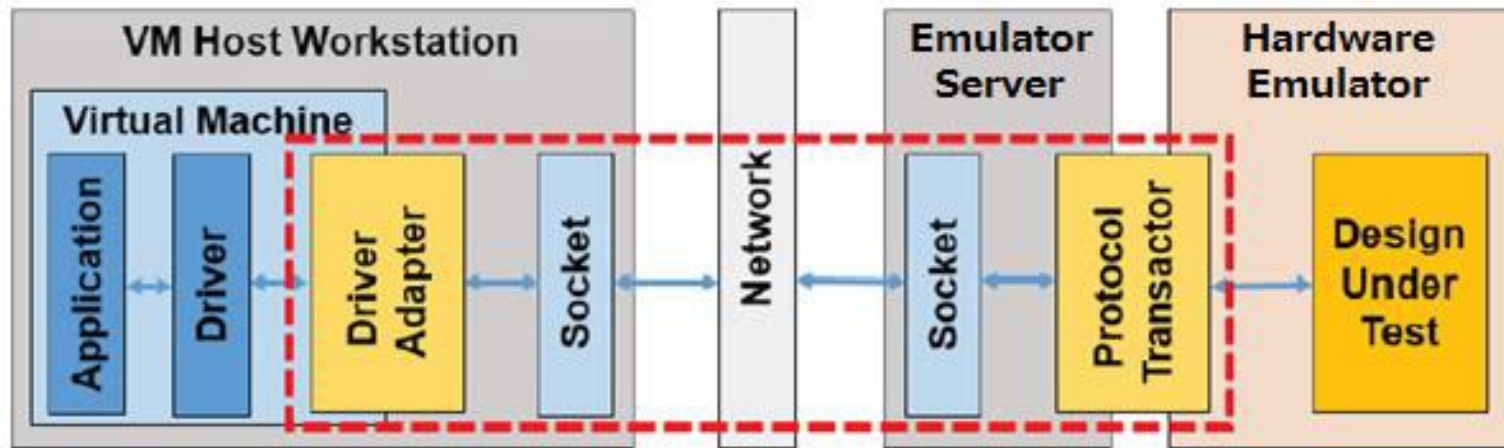
The concept of Virtual Emulation (1)

- What's the Virtual Emulation?
 - Virtual emulation technology needs a “**software adapter**” that enables user applications and OS drivers to establish a virtual protocol connection to the hardware emulator



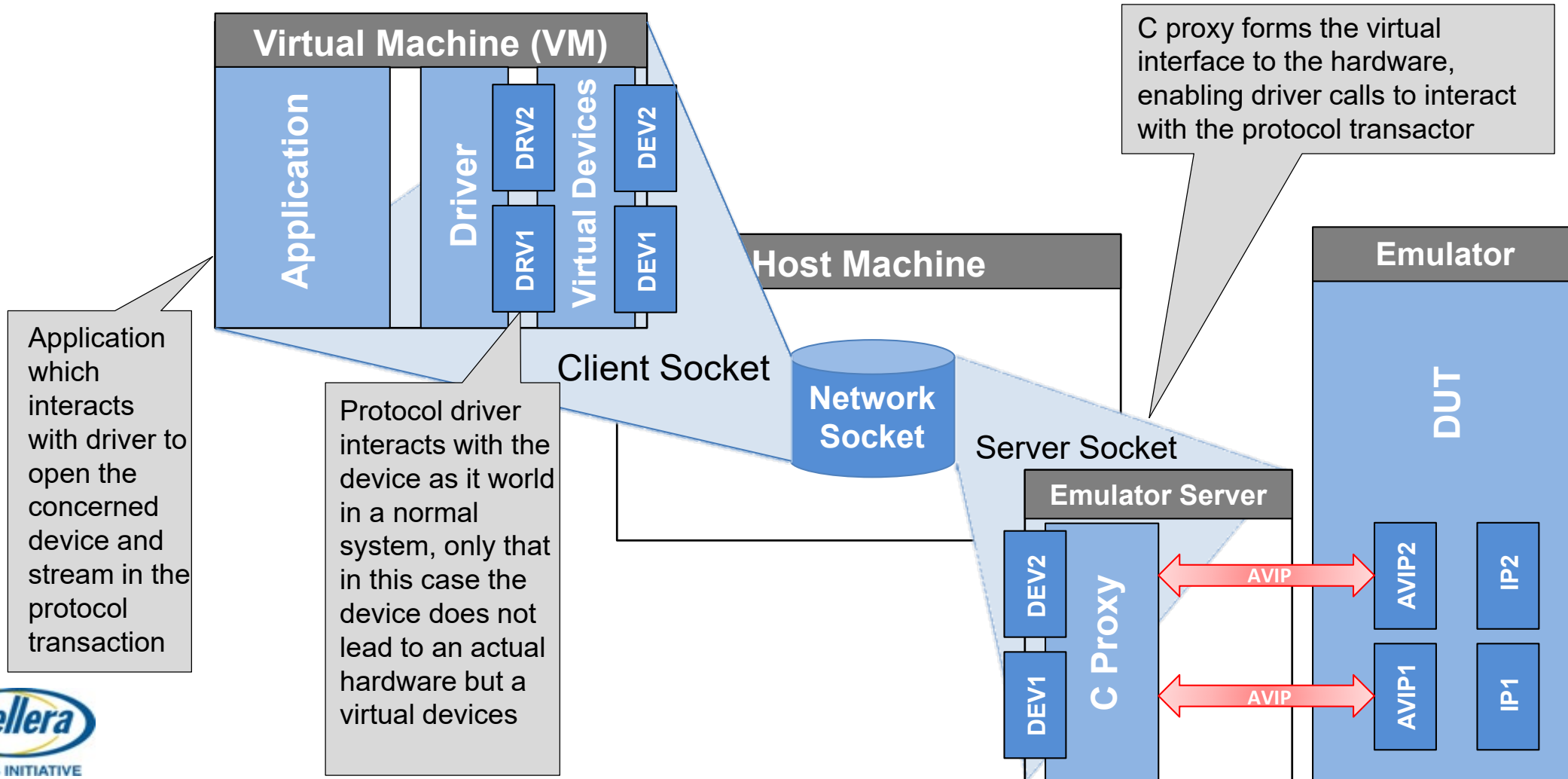
The concept of Virtual Emulation (2)

- Benefits
 - Flexibility to run jobs without physical constraints
 - Statically controlled runs allowing freedom to debug without timeouts
 - Validate system level S/W interoperability early in development cycle



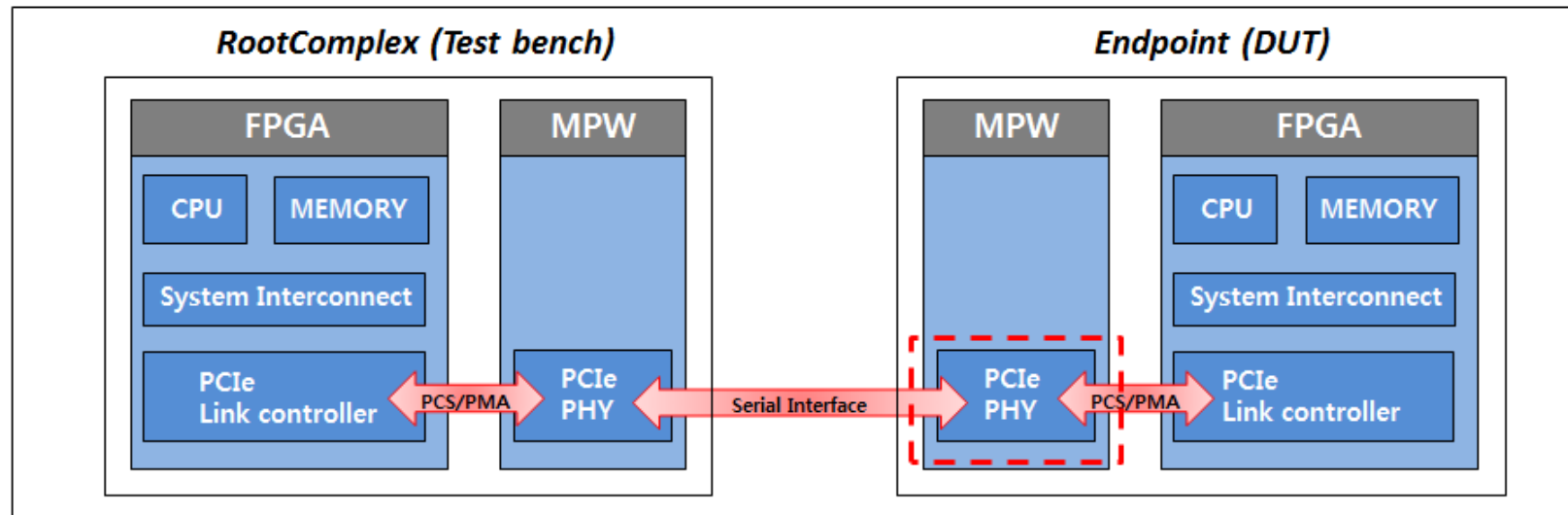
The concept of Virtual Emulation

- Architecture of Virtual Emulation Platform



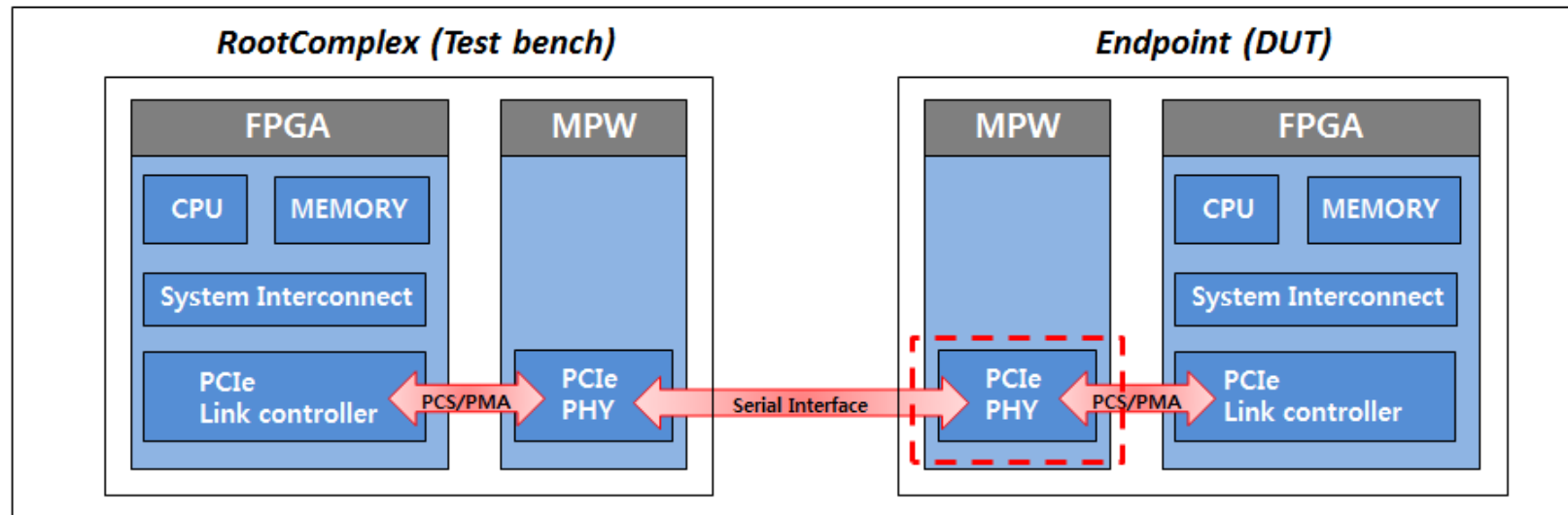
Existing HSI IP validation flow (1)

- HSI IP validation based on FPGA with MPW (Multi Project Wafer)
 - DUT: mini-SoC including PCIe Endpoint
 - Test bench: mini-SoC including PCIe Root Complex



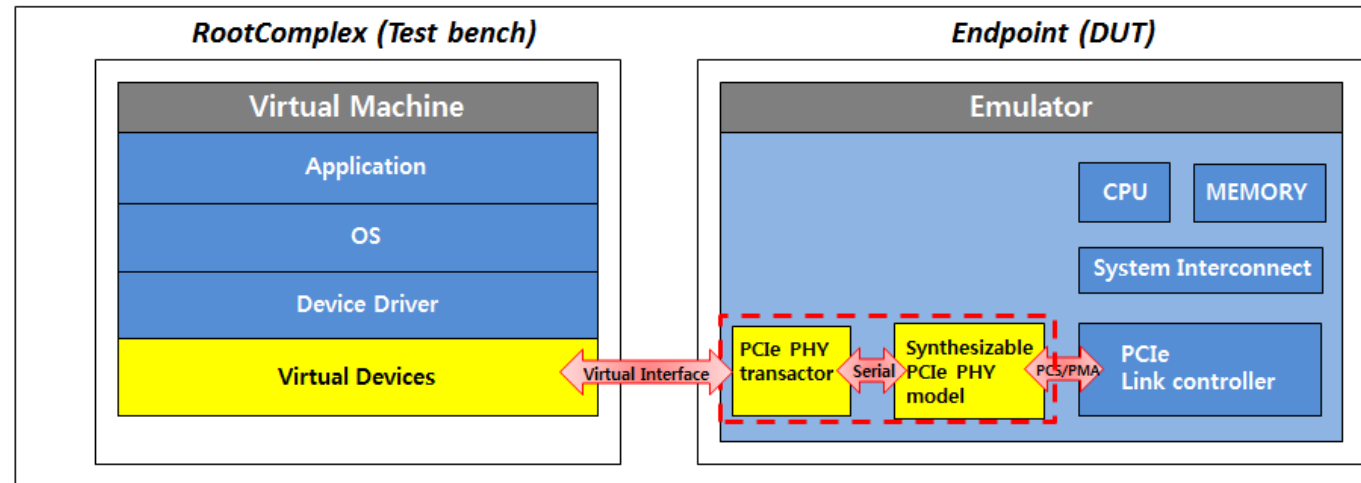
Existing HSI IP validation flow (2)

- Problems
 - Long TAT (Turn-Around-Time) because of MPW test chips
 - High effort to implement the DUT on FPGA



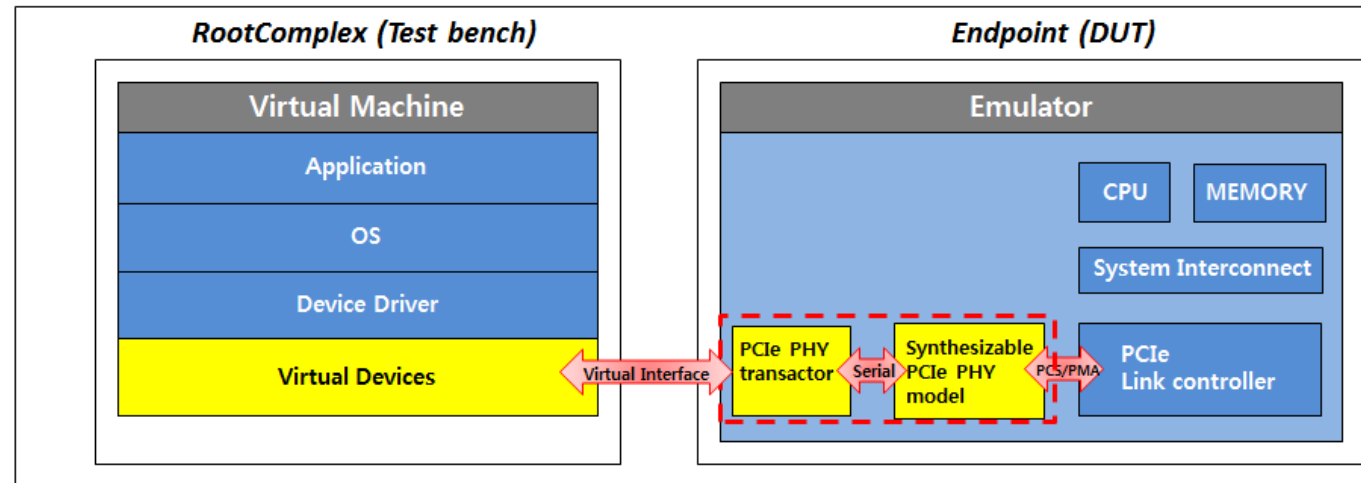
Proposed HSI IP validation flow (1)

- HSI IP validation based on virtual emulation platform
 - DUT: mini-SoC including PCIe Endpoint with PCIe transactor
 - Test bench: virtual devices with device driver on virtual machine



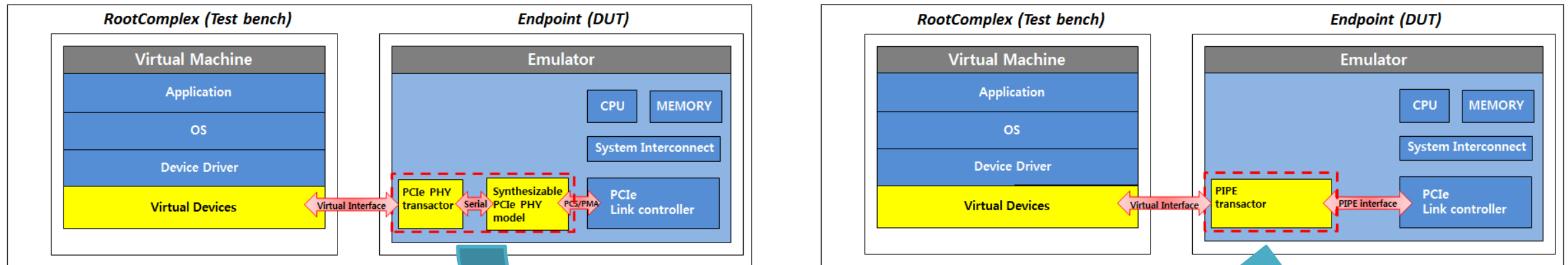
Proposed HSI IP validation flow (2)

- Benefits
 - Not require the MPW test chip and available at a pre-silicon stage
 - Reinforce the functional verification by performing OS-level scenarios







Method to accelerate pre-silicon validation (1)

- Using the PIPE (PHY interface for the PCI Express) interface
 - Replace PHY layer with PIPE transactor

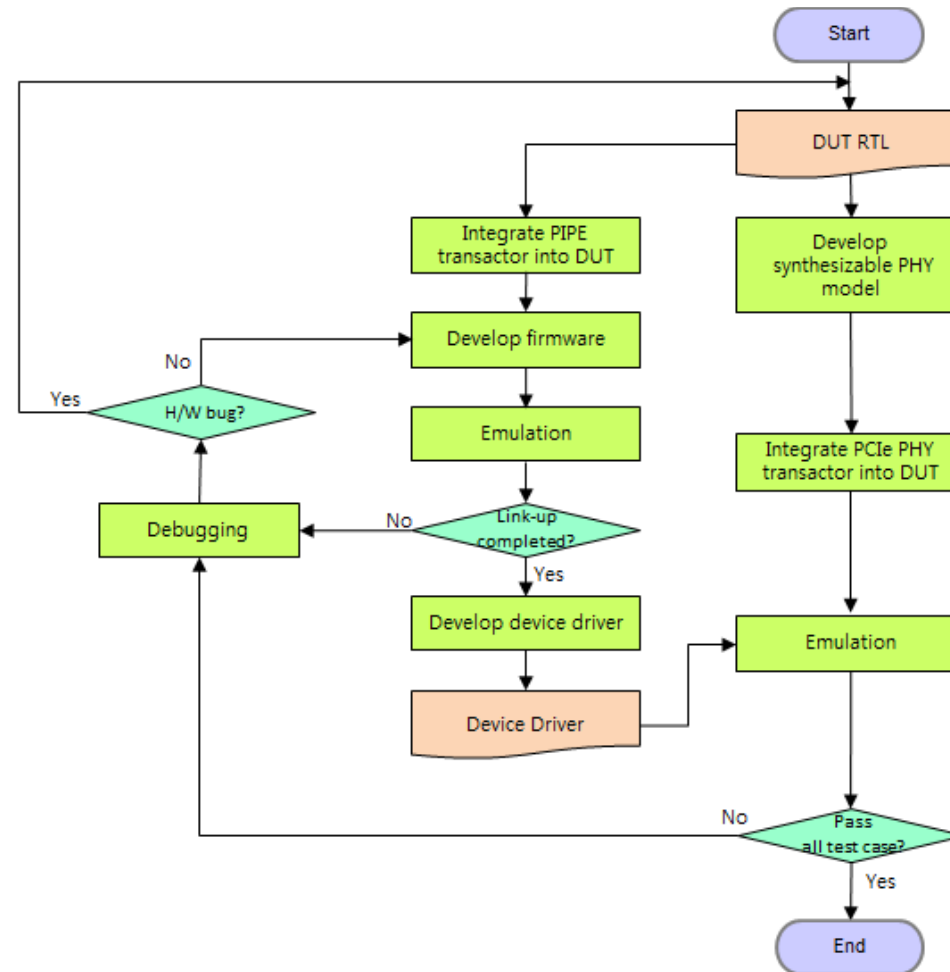


Method to accelerate pre-silicon validation (2)

Scenario / Interface	Gen3 Link-up		Loading VM and Transaction (4096 burst)	
	Simulation Time (ms)	Wall clock time (sec)	Simulation Time (ms)	Wall clock time (sec)
Serial Interface	62.7	3149.76	100	5006.75
PIPE Interface	2 	19.38 	18 	379.28 

Method to accelerate pre-silicon validation (3)

- Mixed interface flow for reduce development TAT (Turn-Around-Time)



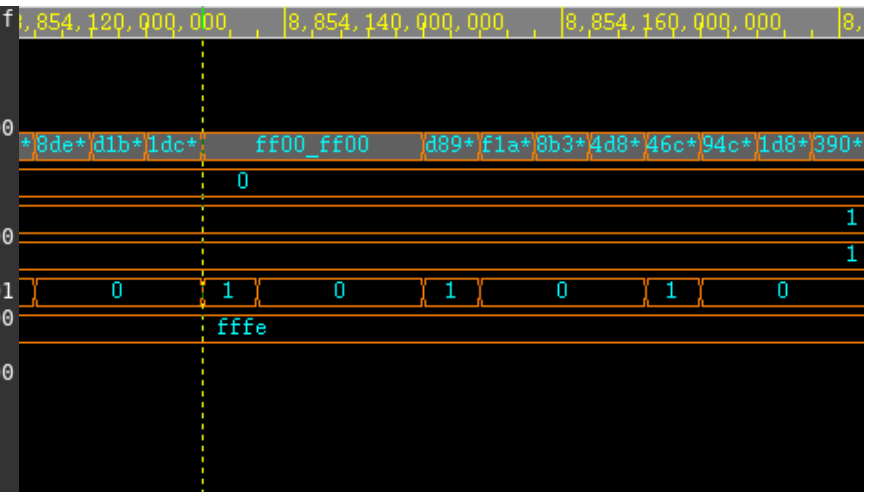
How to debug in Virtual Emulation Platform (1)

- H/W debugging method
 - Waveform dump with fully static environment
 - Transaction monitoring

```

=====START OF TRANSACTION LAYER PACKET=====
TX: CFG_WR0 Address: 30 Length: 1 Seq: fff Data: fffff800 TAG: 4
=====END OF TRANSACTION LAYER PACKET=====
=====START OF TRANSACTION LAYER PACKET=====
TX: CFG_RD0 Address: 30 Length: 1 Seq: fff TAG: 5
=====END OF TRANSACTION LAYER PACKET=====
=====START OF TRANSACTION LAYER PACKET=====
RX: CMPL_WITHOUT_DATA Requestor ID: 0 Completor ID: 0 Status : 0 Length: 0 Seq:
=====END OF TRANSACTION LAYER PACKET=====
=====START OF TRANSACTION LAYER PACKET=====
RX: CMPL_WITH_DATA Requestor ID: 0Completor ID: 0 Status : 0 Length: 1 Seq: fff
Data fields of this packet are-> ffff0000
=====END OF TRANSACTION LAYER PACKET=====
DEBUG_REMOVE Assigned local_id = 6
DEBUG_REMOVE Assigned local_id = 7
DEBUG_REMOVE Assigned local_id = 8
DEBUG_REMOVE Assigned local_id = 9
=====START OF TRANSACTION LAYER PACKET=====
TX: CFG_WR0 Address: 30 Length: 1 Seq: fff Data: ffff0001 TAG: 6
=====END OF TRANSACTION LAYER PACKET=====
=====START OF TRANSACTION LAYER PACKET=====
TX: CFG_WR0 Address: 10 Length: 1 Seq: fff Data: fe800000 TAG: 7
=====END OF TRANSACTION LAYER PACKET=====
60,WRITE_SLV_CFG_0,PCIE,0,0,56,20,f,1,416.882848612,fffffff
61,READ_SLV_CFG_0,PCIE,0,0,52,20,f,1,416.882909284,
62,WRITE_SLV_CFG_0,PCIE,0,0,56,20,f,1,416.885443729,0
63,READ_SLV_CFG_0,PCIE,0,0,52,24,f,1,416.885505469,
64,WRITE_SLV_CFG_0,PCIE,0,0,56,24,f,1,416.887329131,ff00ff00
65,READ_SLV_CFG_0,PCIE,0,0,52,24,f,1,416.887391100,
66,WRITE_SLV_CFG_0,PCIE,0,0,56,24,f,1,416.891628122,0
67,READ_SLV_CFG_0,PCIE,0,0,52,30,f,1,416.891695020,
68,WRITE_SLV_CFG_0,PCIE,0,0,56,30,f,1,416.893531233,fffffff800
69,READ_SLV_CFG_0,PCIE,0,0,52,30,f,1,416.893596693,
70,WRITE_SLV_CFG_0,PCIE,0,0,56,30,f,1,416.896182963,ffff0001
71,WRITE_SLV_CFG_0,PCIE,0,0,56,10,f,1,416.896273871,fe800000
72,WRITE_SLV_CFG_0,PCIE,0,0,56,14,f,1,416.896325877,0
73,WRITE_SLV_CFG_0,PCIE,0,0,56,30,f,1,416.896372162,fe970000
74,READ_SLV_CFG_0,PCIE,0,0,52,3c,2,1,416.998889817,
75,WRITE_SLV_CFG_0,PCIE,0,0,56,3c,1,1,417.000649482,a
76,READ_SLV_CFG_0,PCIE,0,0,52,4,3,1,417.000713440,
77,WRITE_SLV_CFG_0,PCIE,0,0,56,4,3,1,417.003050944,147

```



How to debug in Virtual Emulation Platform (2)

- S/W debugging method
 - S/W debugger
 - Transaction monitoring

```

ubuntu@ubuntu: ~/pcimem/pcimem
/sys/devices/pci0000:00/0000:00:05.0/resource0 opened.
Target offset is 0x0, page size is 4096
mmap(0, 4096, 0x3, 0x1, 4, 0x0)
PCI Memory Calls
PCI Memory mapped to address 0xbd57e000.
App Start Time : 674091182
MEM Written Value at offset 0x4E14 (0x7fd1bd57ee14): 0xDEADBEEF
App End Time : 674147442
NET Time : 56260
/sys/devices/pci0000:00/0000:00:05.0/resource0 opened.
Target offset is 0x0, page size is 4096
mmap(0, 4096, 0x3, 0x1, 4, 0x0)
PCI Memory Calls
PCI Memory mapped to address 0xbd57e000.
App Start Time : 674217193
MEM Written Value at offset 0x4E18 (0x7fd1bd57ee18): 0xDEADBEEF
App End Time : 674271200
NET Time : 54007
/sys/devices/pci0000:00/0000:00:05.0/resource0 opened.
Target offset is 0x0, page size is 4096
mmap(0, 4096, 0x3, 0x1, 4, 0x0)
PCI Memory Calls
PCI Memory mapped to address 0xbd57e000.
App Start Time : 674341197
MEM Written Value at offset 0x4E1C (0x7fd1bd57ee1c): 0xDEADBEEF
App End Time : 674363933
NET Time : 22736
App NET RATE(5000) : (2928528736) 585705

```

address	0	4	8	C 0123456
ZSD:B0010000	DEADBEEF	DEADBEEF	DEADBEEF	DEADBEEF i%-Pi%
ZSD:B0010010	DEADBEEF	DEADBEEF	DEADBEEF	DEADBEEF i%-Pi%

```

=====START OF TLP RAW HEADER=====
TX: DW0: Byte3(7:0)-Byte0(7:0)-> 01000040
TX: DW1: Byte3(7:0)-Byte0(7:0)-> 0f000000
TX: DW2: Byte3(7:0)-Byte0(7:0)-> 000001b0
TX: DW3: Byte3(7:0)-Byte0(7:0)-> deadbeef
=====END OF TLP RAW HEADER=====
TRANSMITTED PACKET RANGE 7:3 is 8and RANGE 2:1 is0and 40
RECEIVED PACKET FULL 1000040
RECEIVED PACKET FULL f000000
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL 1b0
RECEIVED PACKET FULL deadbeef
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL 0
RECEIVED PACKET FULL 0
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL 0
RECEIVED PACKET FULL 0
RECEIVED PACKET FULL ffffffff
RECEIVED PACKET FULL ffffffff
=====START OF TRANSACTION LAYER PACKET=====
TX: MEM_WR Address: b0010000 Length: 1 Seq: fff TC: 0
IN DATA PRINTING 0 10and 0
IN DATA PRINTING 1 1000040
IN DATA PRINTING 2 f000000
IN DATA PRINTING 0 10and 4
IN DATA PRINTING 1 1b0
IN DATA PRINTING 2 deadbeef
IN DATA PRINTING 0 10and 8
IN DATA PRINTING 1 0
IN DATA PRINTING 2 0
IN DATA PRINTING 0 10and c
IN DATA PRINTING 1 0
IN DATA PRINTING 2 0
Data fields of this packet are-> deadbeef
=====END OF TRANSACTION LAYER PACKET=====

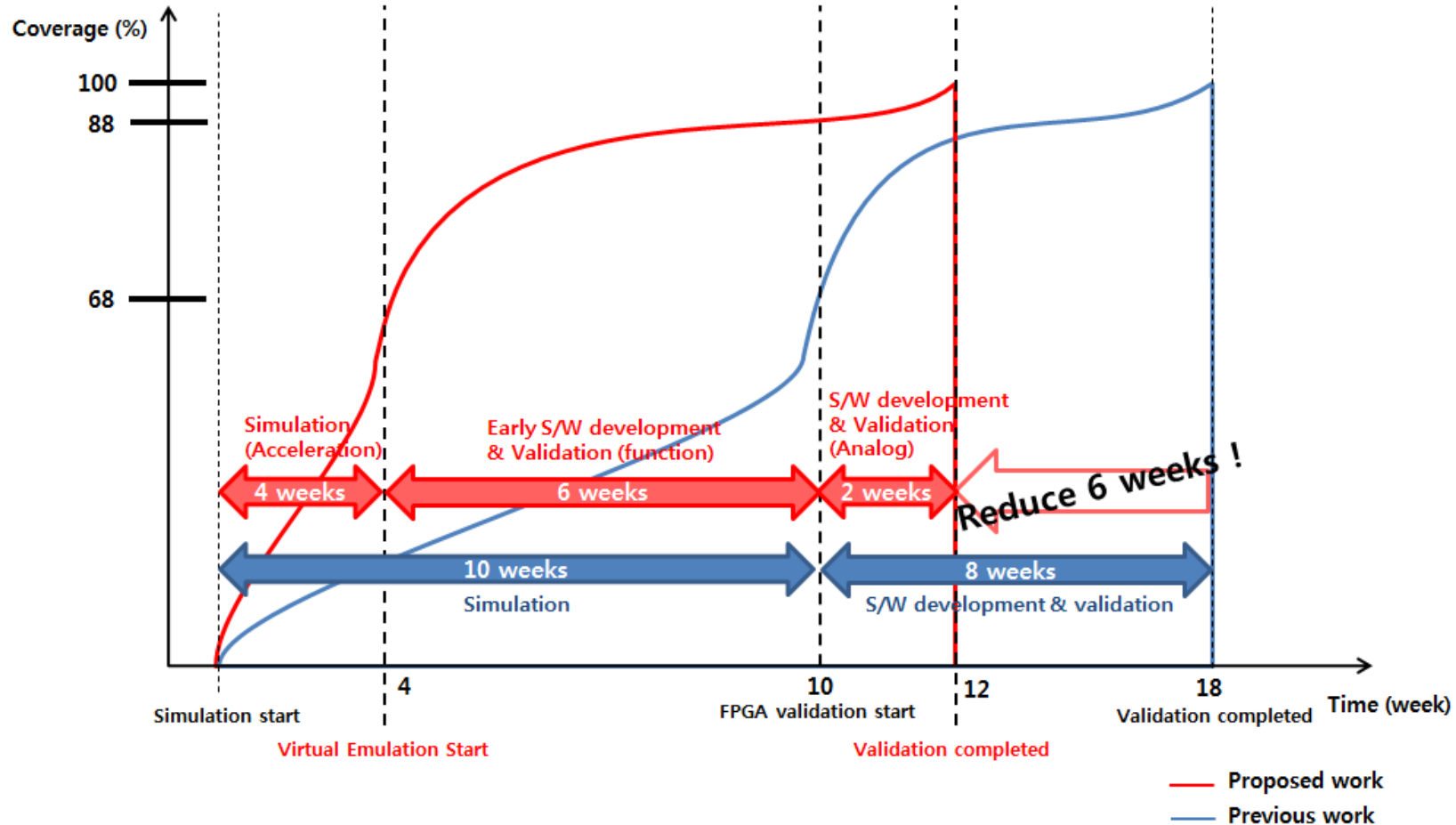
```

Experiment results (1)

Feature	Previous Work			Proposed Work		
	All	Completed	Coverage	All	Completed	Coverage
Normal operation (LTSSM, Speed Change, Interrupt, Bifurcation)	143	105	73.4%	143	120	83.9%
Transaction (Inbound, outbound, configuration, DMA)	38	23	60.5%	38	38	100%
Power Management (Aging, ASPM, PM)	33	11	33.3%	33	28	90.3%
Compliance	40	34	85%	40	37	92.5%
Total	254	173	68.1%	254	223	87.8%

19.7% ↑

Experiment results (2)



Conclusions

- The Virtual Emulation Platform (VEP) is proposed for the validation of High-Speed Interface IPs
 - Enable us to develop target S/W codes and take OS-level tests with the full speed of HSI IP at a pre-silicon stage
 - Reduce the TAT of a HSI IP validation
 - Have unique benefits such as flexibility, debug capability with static environment and accessibility at early stage
- For a industrial testcase, the proposed VEP able to achieve **19.7% coverage increase and 33.3% TAT (Turn-Around-Time) reduction**

Thank You