2020 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Innovative Techniques to Solve Complex RDC Challenges

Rohit Kumar Sinha, Intel India

accellera SYSTEMS INITIATIVE

MOTIVATION – Why is RDC Important ?

Lack to FE-BE awareness

There is not much awareness on RDC issues and there is no robust flow to validate in FE and BE flows

Presence of Async and Sync reset

Different flavors of resets used in the low power design which increases the complexity of reset verification

Reconvergence Issues are similar to CDC

Complexity of reset structures leading to reconvegence risk. metastability and reconvergence issues similar to the failures seen in asynchronous CDC

Hierarchical Reset Arch

High Complexity
Reset Structure
Complex Reset Sequences lead to new domains

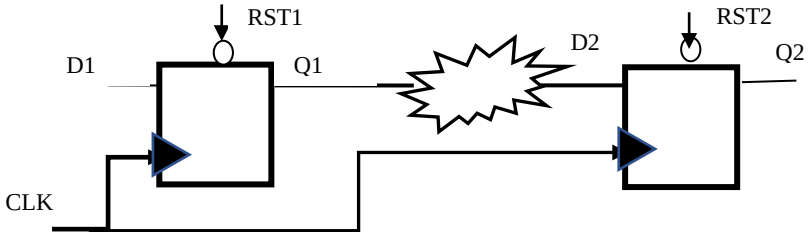
Multiple Reset Domain

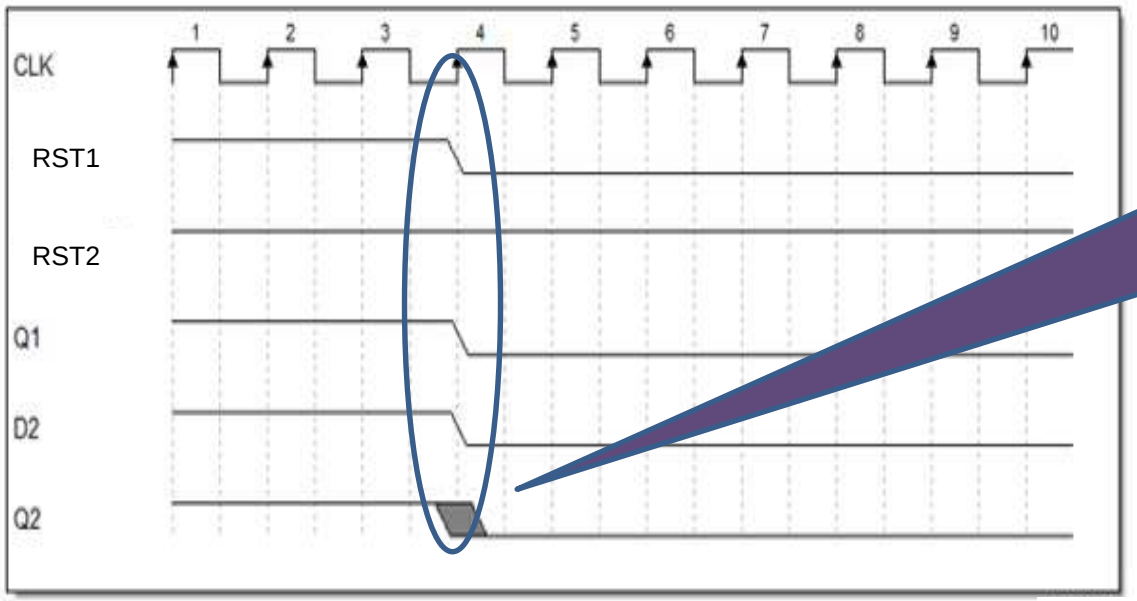
More than 20+ reset domains
Sometimes independent reset domains are created

RDC issues not addressed in simulations

RDC issues are not caught in functional simulation unless DUT covers such scenario.

RDC - An Introduction to the Problem



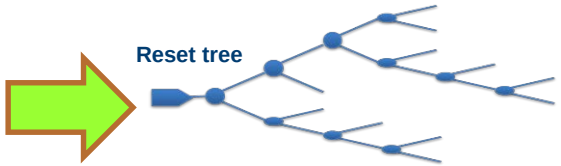


The assertion of RST1 while RST2 not being asserted can cause metastability on destination flop

RESET DOMAIN CROSSING FLOWS

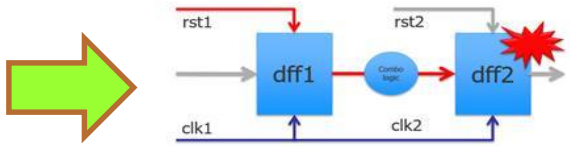
1. Reset Tree Evaluation

- Power On Reset, Watchdog Timeout reset
- Debug reset, Software reset, and Loss of Clock reset.
- Construction Issues



2. Reset Domain Crossing

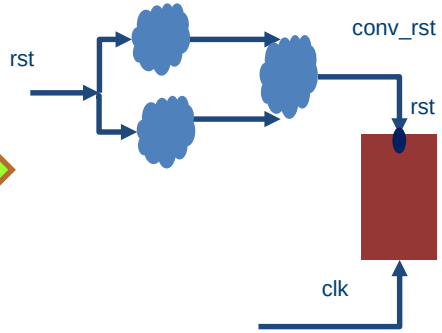
- Application of resets- Sync/Async Behaviour
- Same reset used as both sync/async in a single module leading to synth vs sim mismatches
- Combinational Logic in Reset Path

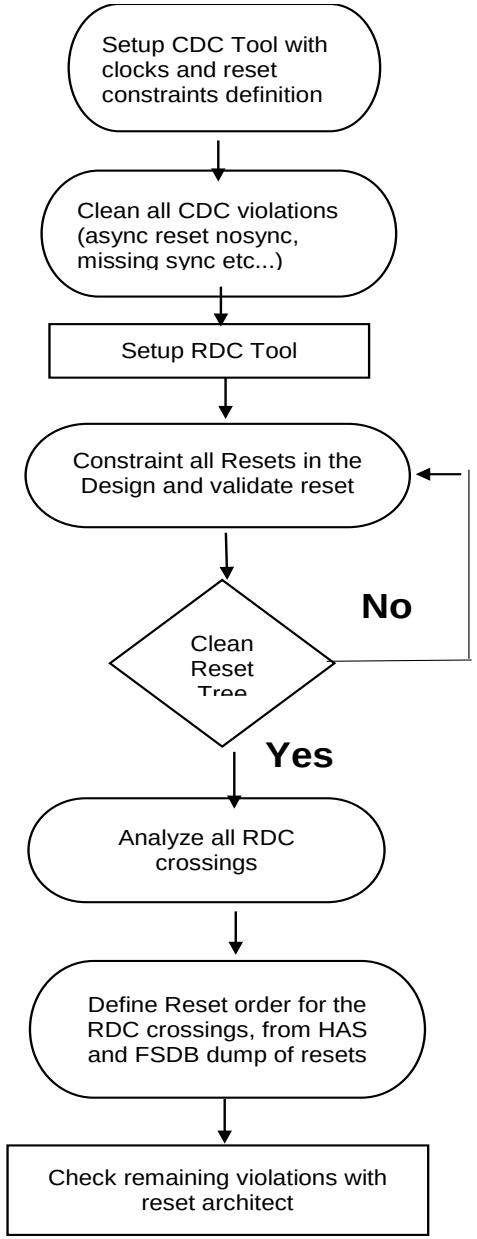


- If rst1 is asserted while rst2 is not asserted, the asynchronous data from dff1 will metastability on dff2

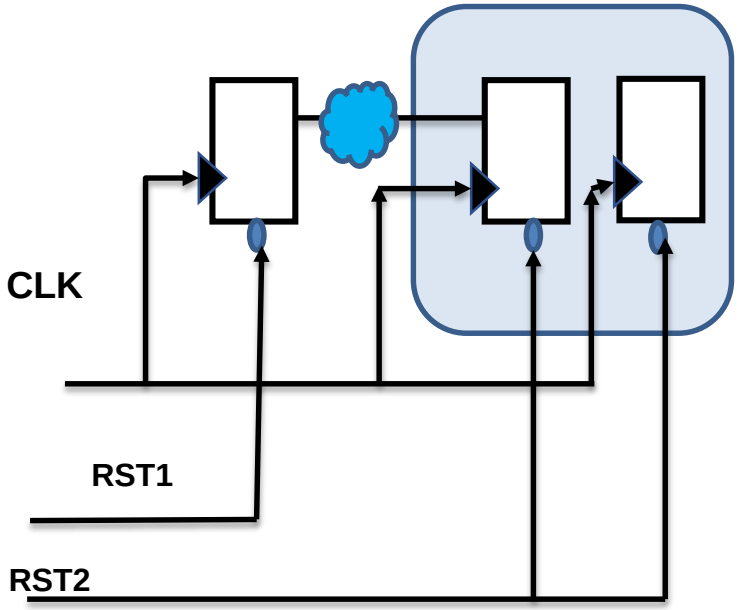
3. Reset Convergence & Reconvergence

- Reset splitting and converging
- Reset bus synchronizers reconverging

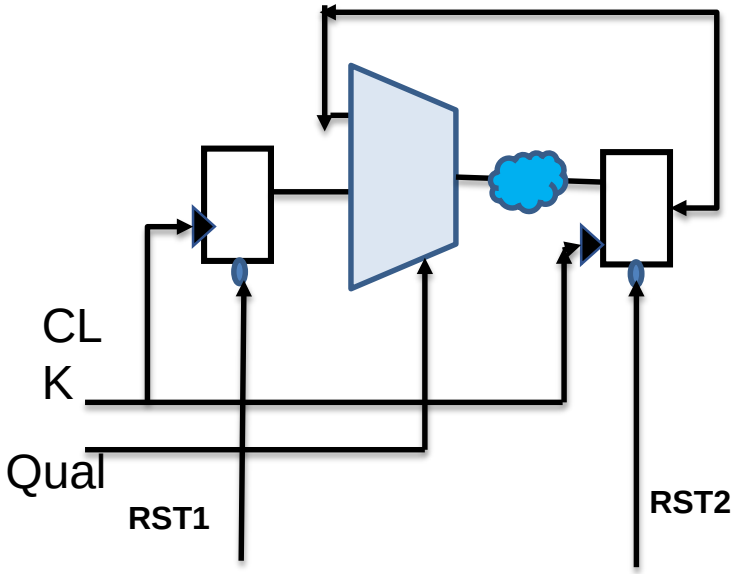




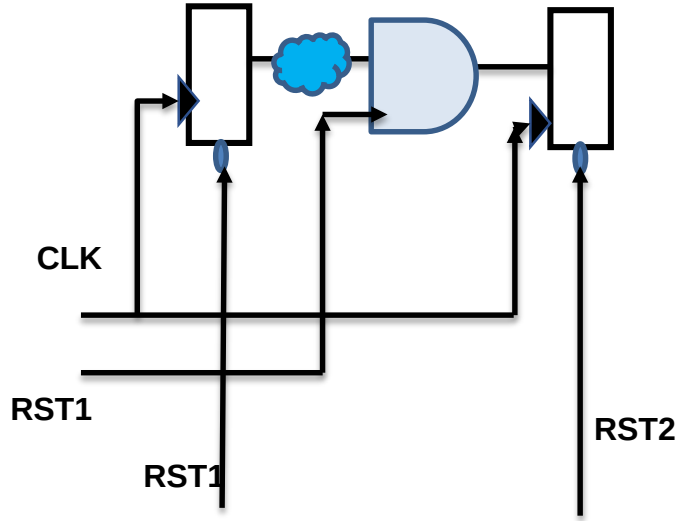
Reset Sync Techniques



Using Mutli-flop synccell



Source-Style Shut off Logic



Destination Style Shut off logic

Constraint “qualifier -rdc -name -from_rst-to_rst”

Reset Assertion Ordering 1/2

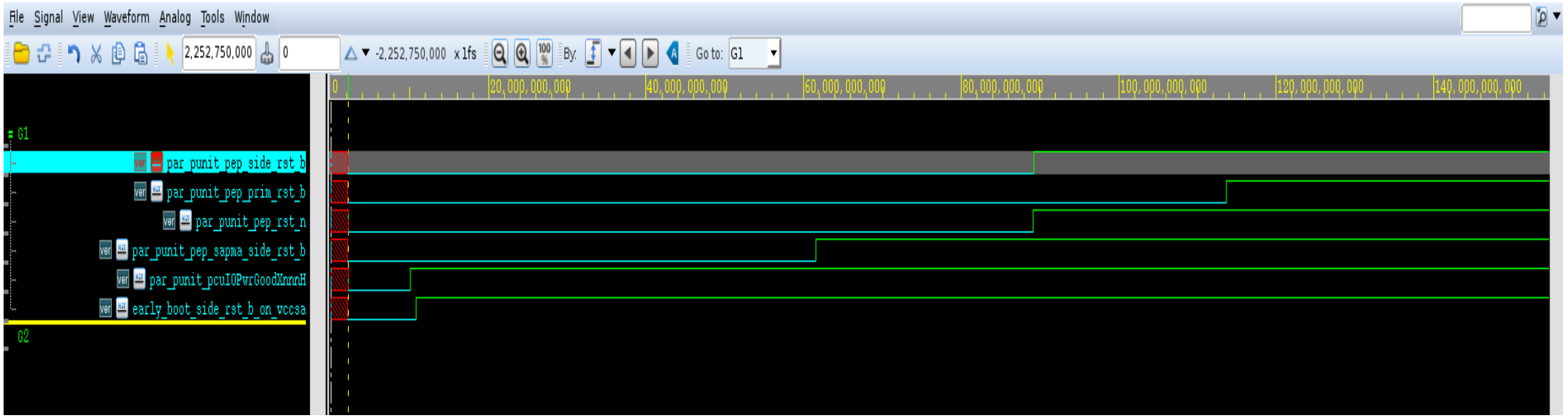
Reset Signal	Source		Destination	Clock	FLR, D3hot and hot-reset
PERST#	External	Platform***	P-Unit, SNPS CTRL, Custom Logic	Asynch	No. Asserted only before powerup and L2
R1		P-Unit	PEP entirely	Asynch	No. Asserted when power is unavailable.
R2		P-Unit	SAPMA	PMSB clock	No. Asserted only before powerup and L2
R3		P-Unit*	MGPHY	GPSB clock	No. Asserted only before powerup and L2
R4		P-Unit	Custom Logic / SBR	GPSB clock	No. Asserted only before powerup and L2
R5		P-Unit*	IOSF2AXI	GPSB clock	No. Asserted only before powerup and L2
R6		P-Unit*	Custom Logic	GPSB clock	No. Asserted only before powerup and L2
R7		P-Unit	IOSF2AXI	IOSF-PRI clock	No. Asserted only before powerup and L2
R8		P-Unit	Custom Logic	P-Unit	Yes. Asserted on all SoC resets
R9	Internal	SAPMA, Custom Logic	MGPHY	Asynch	No. Asserted only before powerup and L2
R10		SAPMA	MGPHY	CRI clock	No. Asserted only before powerup and L2
R11		Custom Logic	MGPHY	Asynch	No. Asserted only before powerup and L2
R12		Custom Logic	IOSF2AXI	AXI clock	No. Asserted only before powerup and L2
		Custom Logic	SNPS clkrst block	Asynch	No. Asserted only on powerup
		SNPS clkrst block	SNPS CTRL	aux_clk	No. Asserted only before powerup and L2
		SNPS clkrst block	SNPS CTRL	Relevant clock	No. Asserted only before powerup and L2

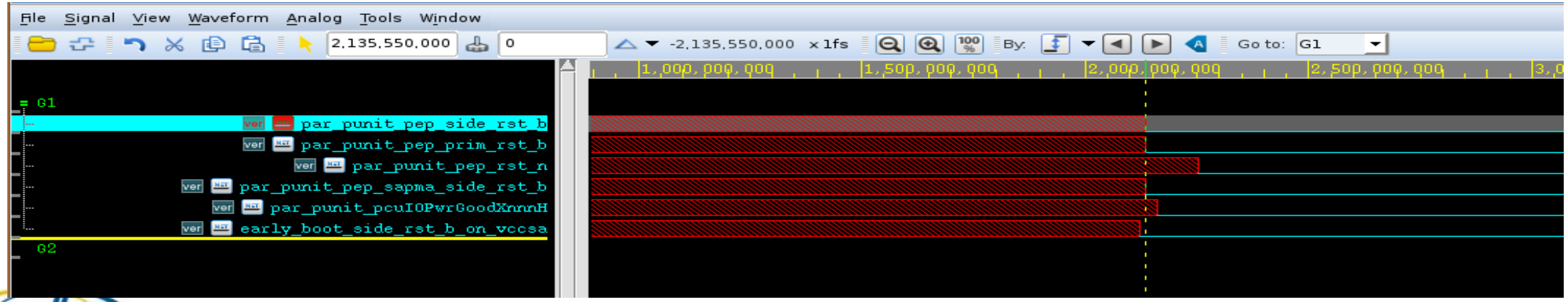
Pciess Resets	Soc Resets
R1	SoC R1_1
R2	SOC R2_1
R3	SOC R3_1
R4	SOC R4_1
R5	SOC R5_1
R6	SOC R6_1
R7	SOC R7_1

Mapping of Pciess resets with SOC

Reset Ordering from High Level Architecture Definition

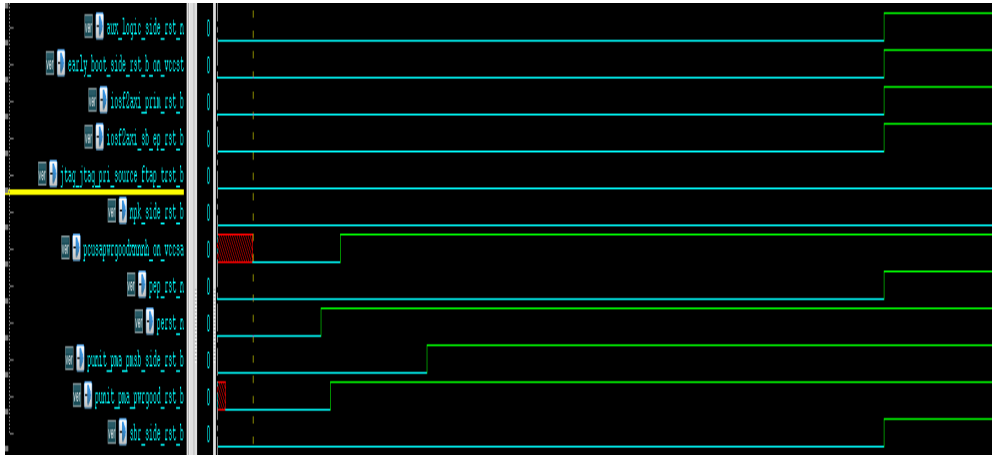
Reset Assertion Ordering 2/2

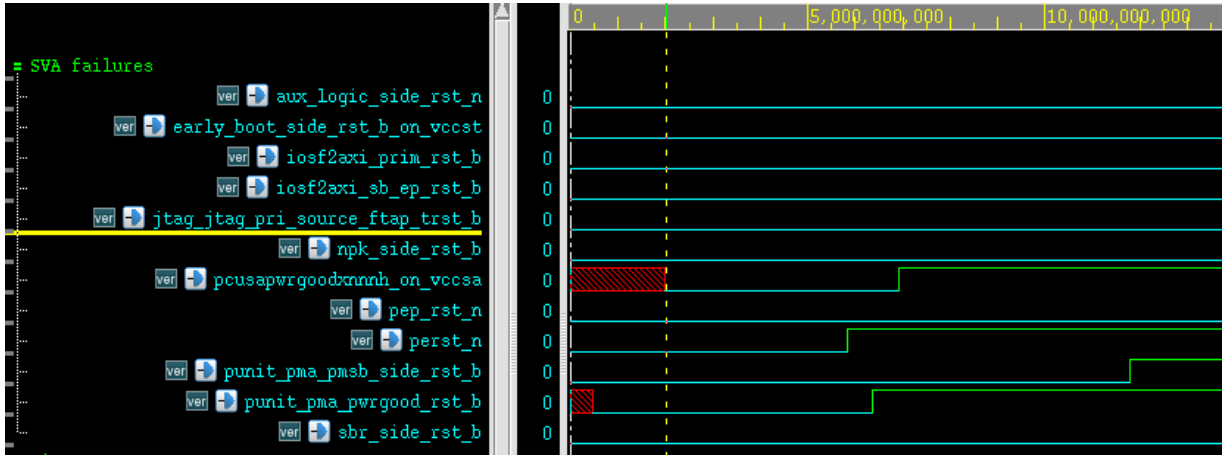




FSDB dump of Pciess resets from PUNIT

Reset Assertion Ordering





Mapping of Pciess resets with SOC

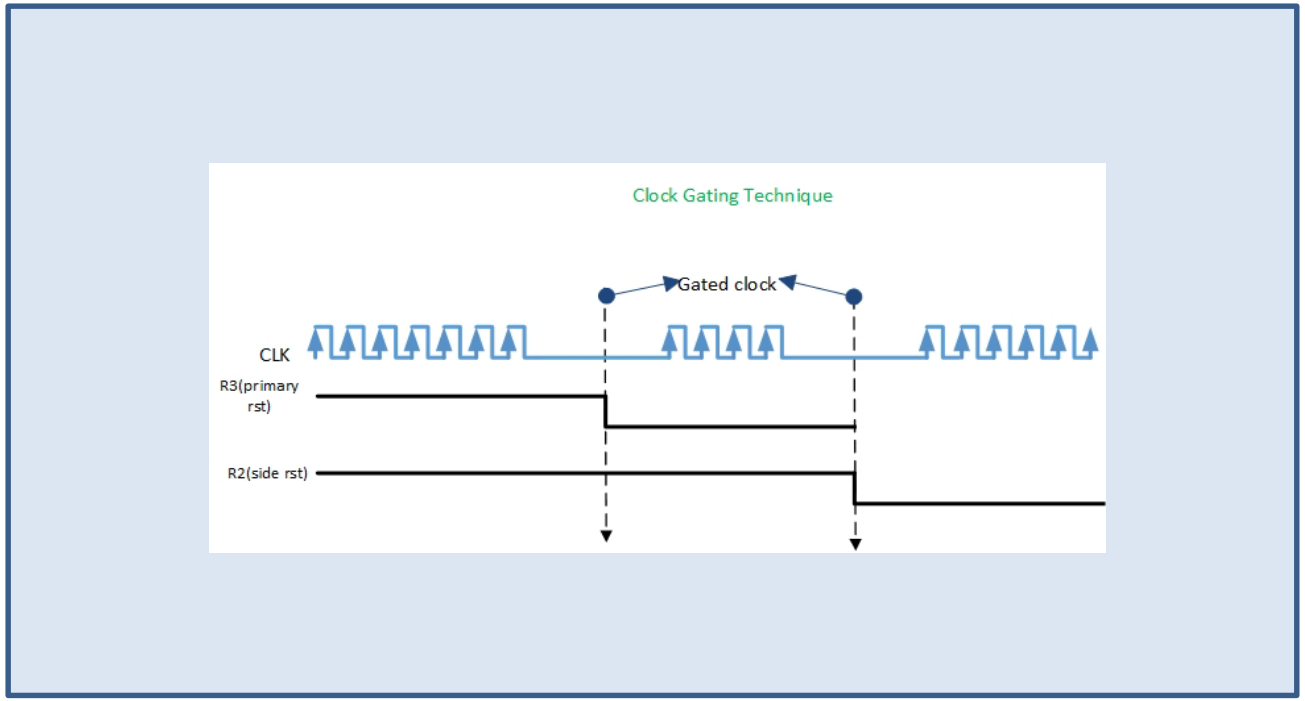
For ex- The TX reset punit_pma_pwrgood_rst_b from the waveform is being asserted after the assertion of the all the above RX resets, there by avoid metastability to propagate to the RX Flop.

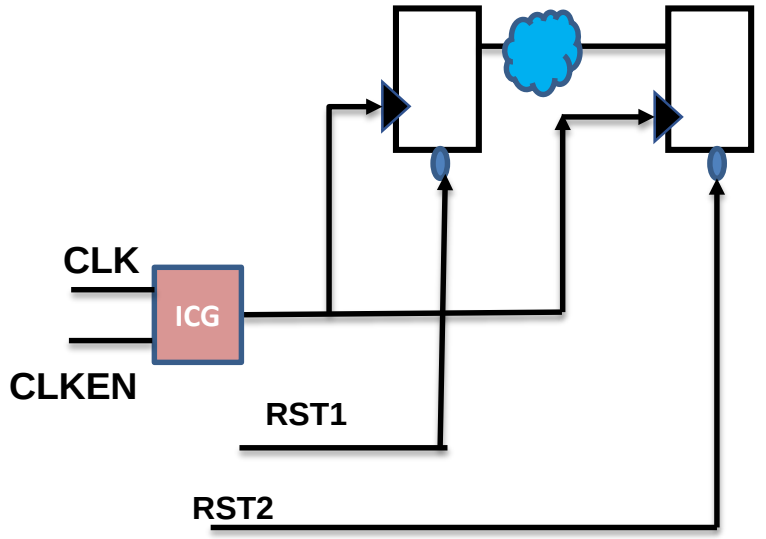
In Reset check tool, the assertion order of these resets can be given by the following directives.

resetcheck order assert -from R1 -to R2

resetcheck order assert -from R3 -to R4

Design Technique – Clock Gating

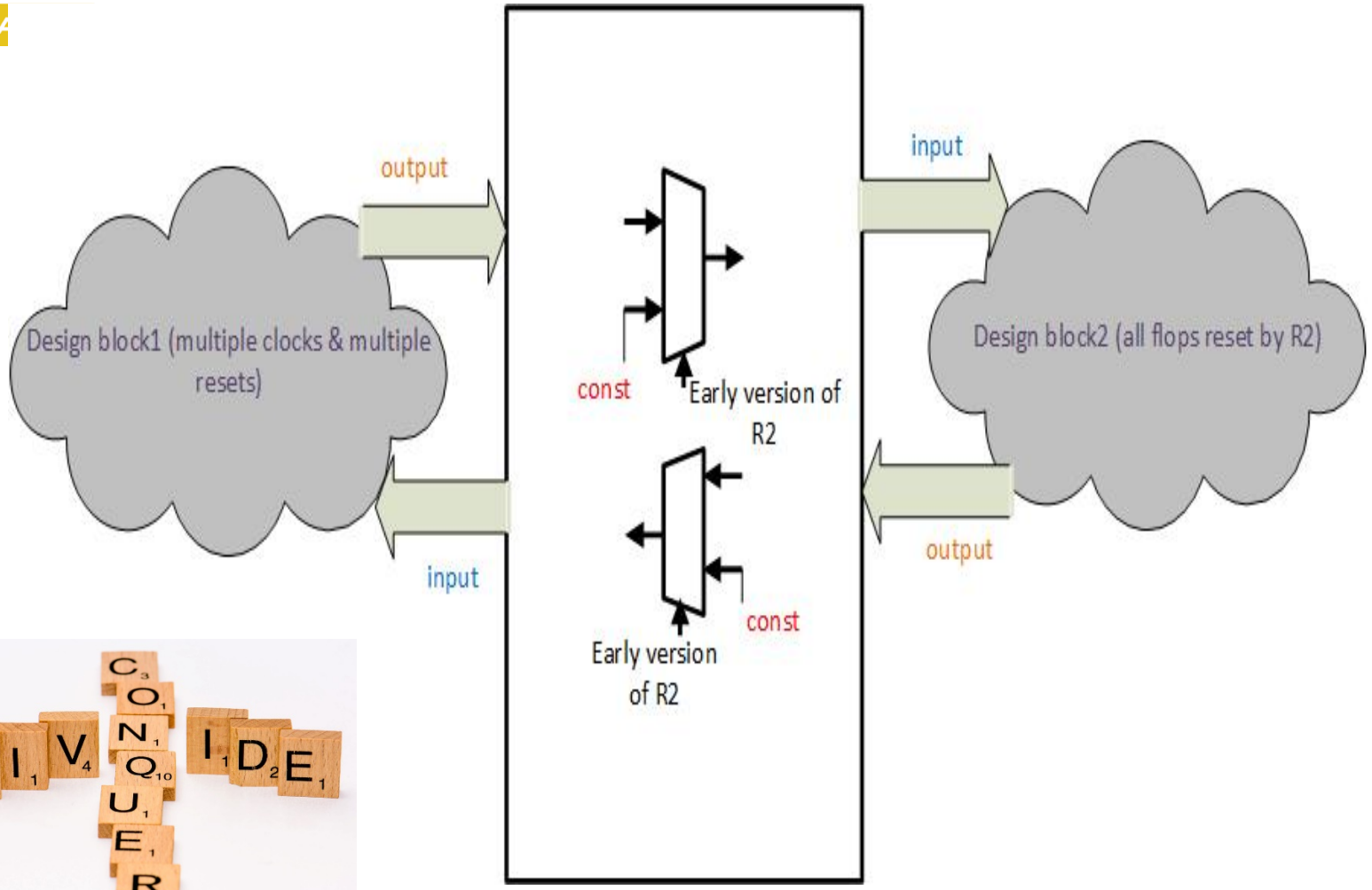




Clock Gating Technique for solving RDC

If the clock is gated before the assertion of the reset and is released after the logic settles down, then RDC can be avoided.

Design Technique – RDC Bridge



Design Block1 – Has multiple clocks and resets

Design Block 2 – Block2 has all flops on Reset R2.

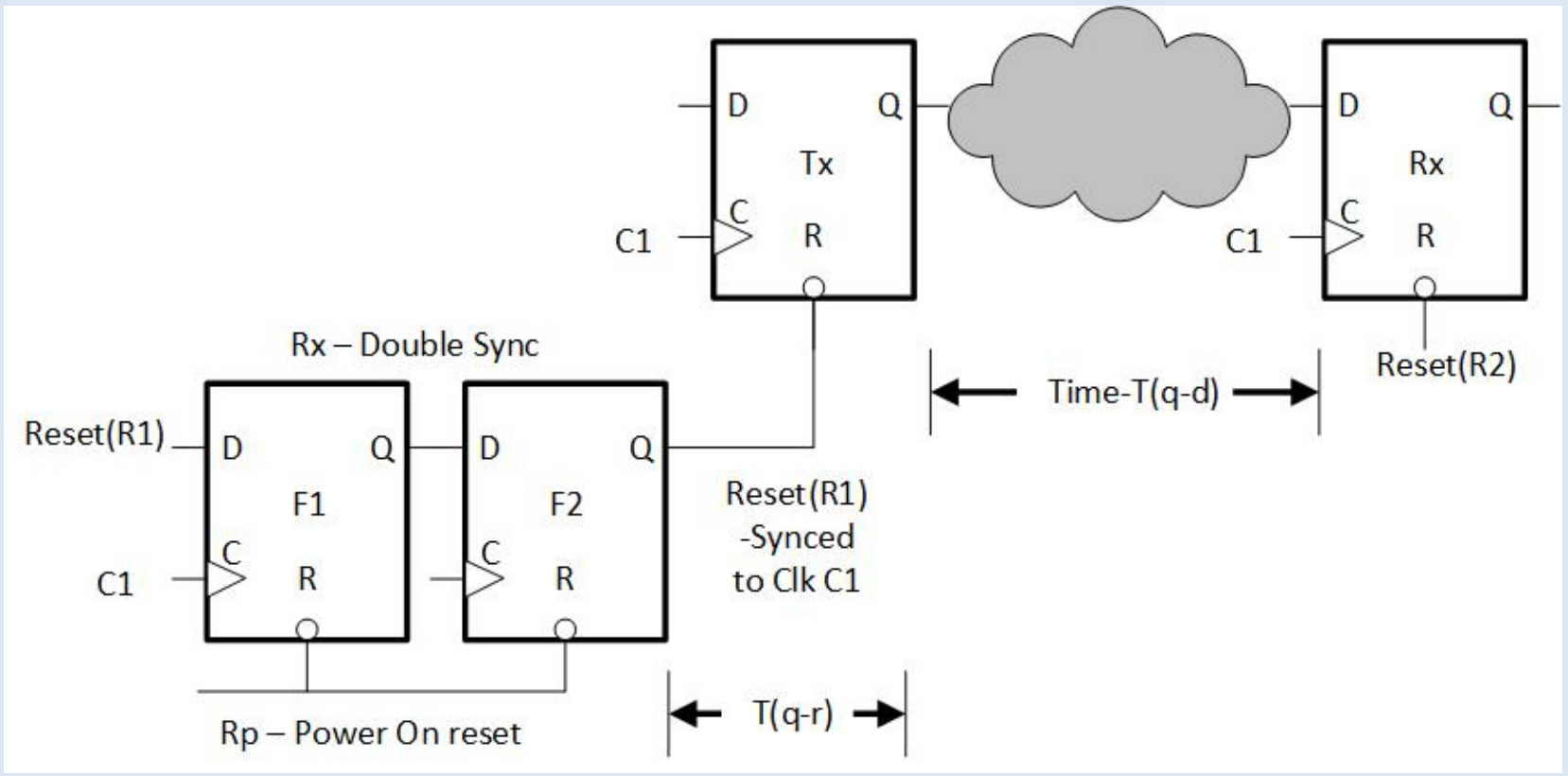
Can have multiple clocks, but the interface logic between the 2 blocks are common.

Create an RDC bridge to isolate the Design Block2 during warm reset and get rid of RDC at Block or even partition Level!

Introducing RDC bridge technique

FCT based RDC solution

Timing Paths in the RDC paths



Let Spyglass report all possible RDC violations paths

Let the STA tool time Only those paths that Spyglass reports.

We may see upto a 60% reduction In the number of RDC paths if the STA tool reports timing Closure.

The Main closure item is the timing path from Double Sync output F2.Q to Rx.D

Part1 of New Solution to RDC

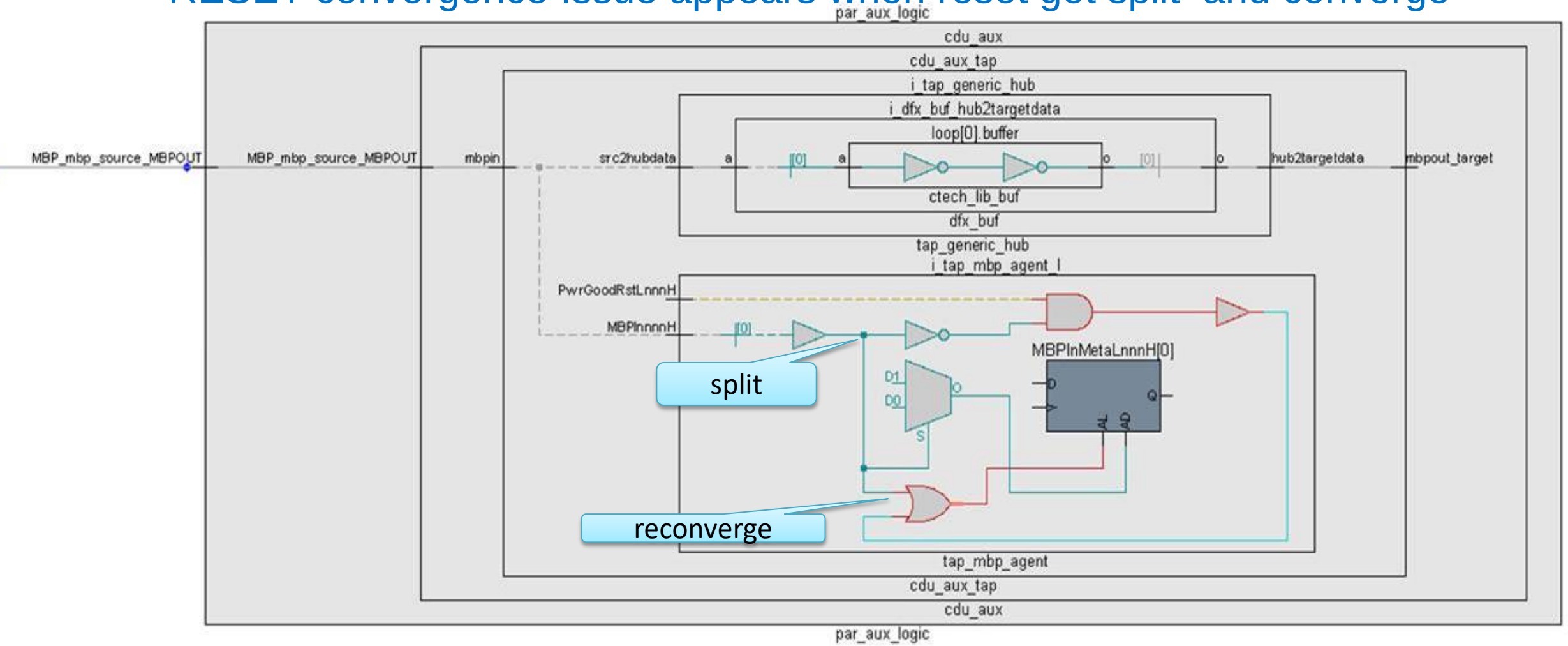
Summary: Bugs found in PCIE Subsystem

- First Implementation of RDC on PCIE Subsystem for Client SoC
- Additional checks improved the Quality Sign Off
- Bugs Uncovered after reset verification
  - Wrong Reset Propagation
  - Power Control logic interpreted as Reset
  - Reset is always active high
  - 2nd Flop with no set/reset pin
  - Reset wrongly used as reset pin
  - Set to Set domain crossing not synchronized
  - Set to Reset domain crossing not synchronized
  - Combinational Logic in reset path
  - Async reset used in sync mode

Partition	#Cell Count	Number of SIPs	Number of HIPs
aux_logic	2577	7	0
Fabric	3108	13	0
PHY	3696	9	5(clock compensator, phy, clock control unit)
Controller	17554	16	18(HIPs)

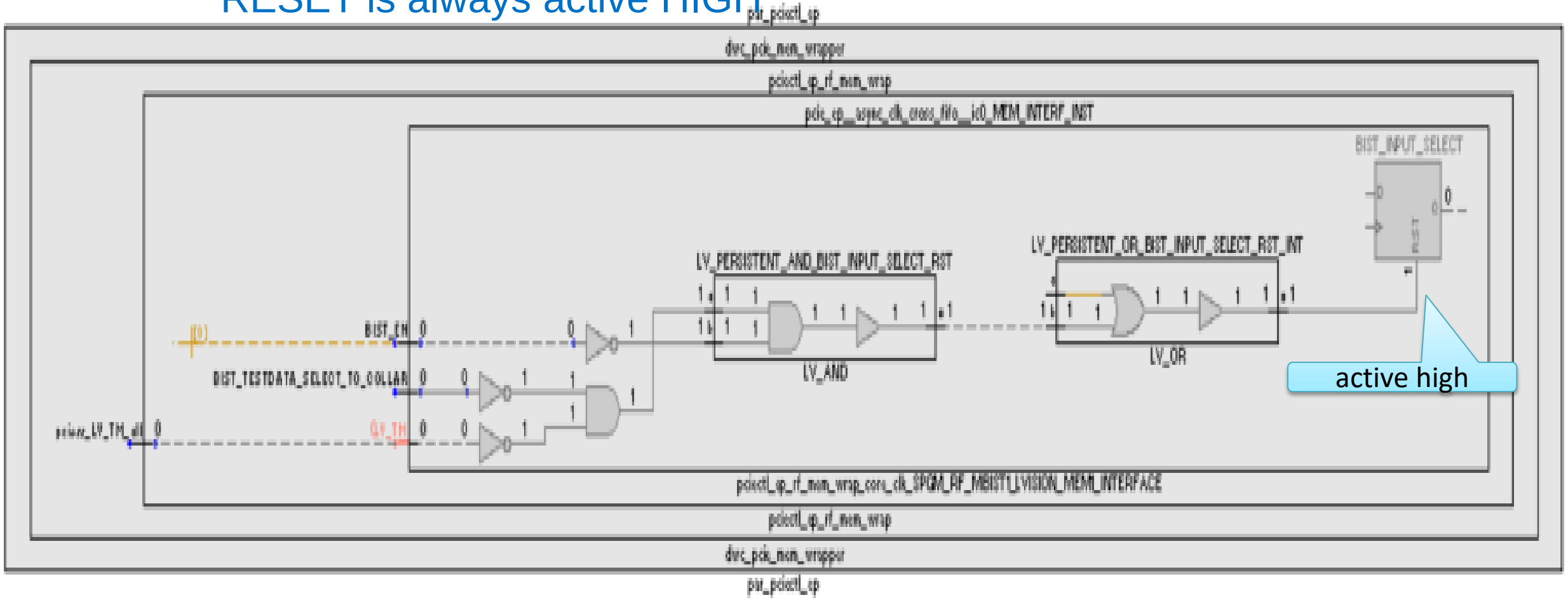
Evidence: Additional Metastability Risk Uncovered

RESET convergence Issue appears when reset get split and converge



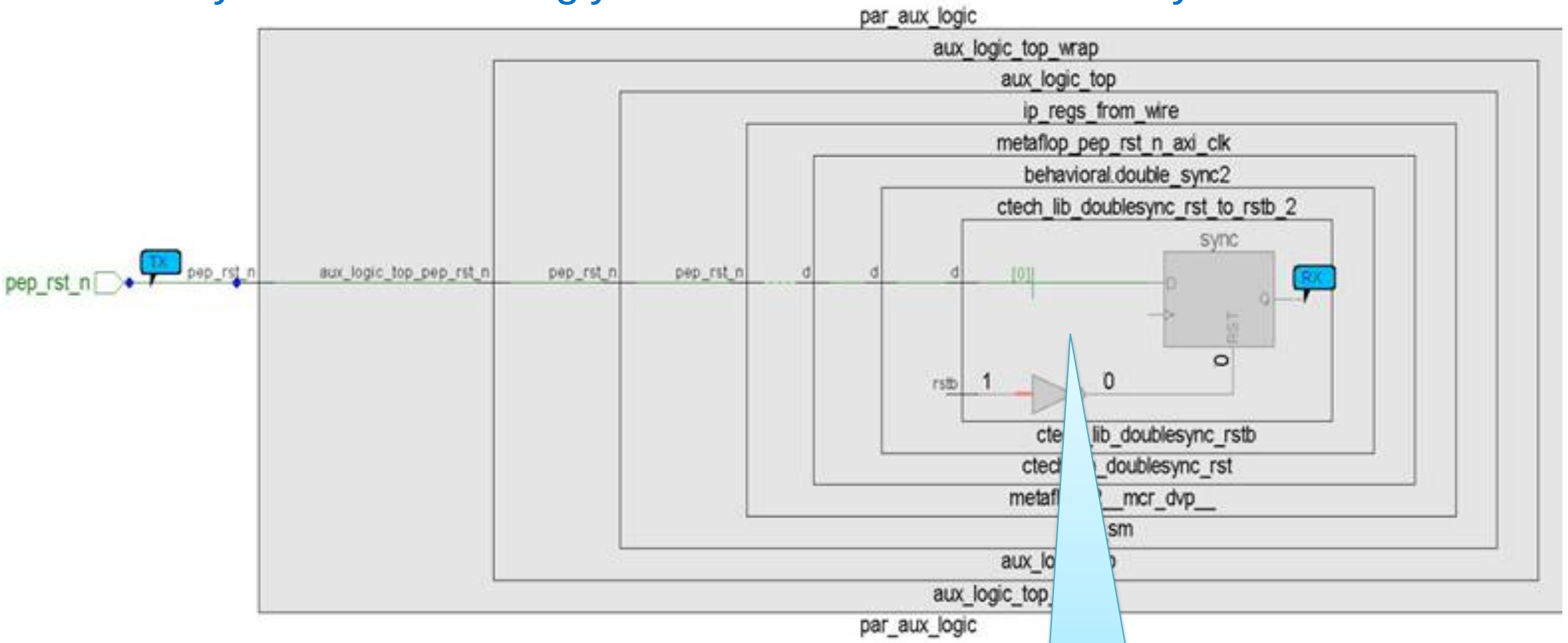
Evidence: Additional Metastability Risk Uncovered

RESET is always active HIGH



Evidence: Additional Metastability Risk To Be Uncovered

Asynchronous wrongly connected to data without synchronizer



2020 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Questions

accellera SYSTEMS INITIATIVE