2019 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Connectivity and Beyond

Shahid Ikram[1], Joseph DErrico[1], Yasmin Farhan[1], Jim Ellis[1], Tushar Parikh[2]

[1]Marvell Semiconductor, Inc.

[2]Synopsys

accellera SYSTEMS INITIATIVE

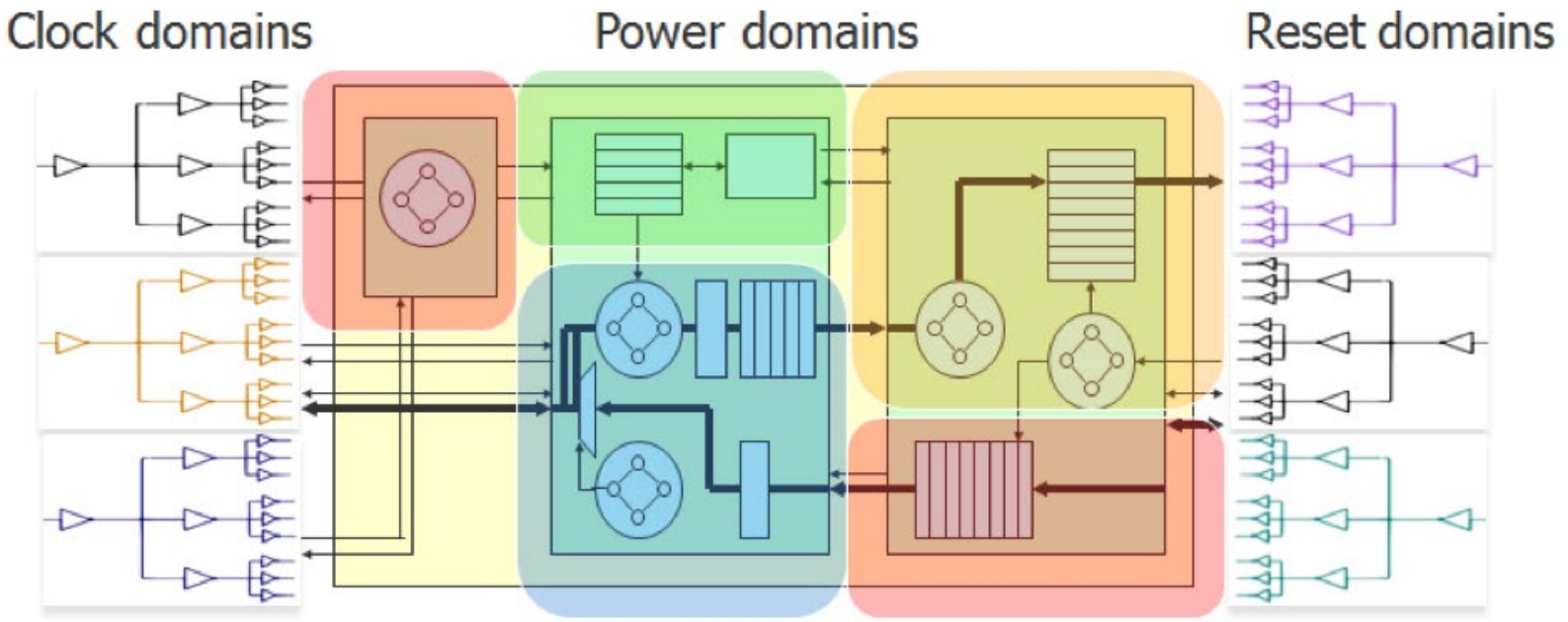
MARVELL

Outline

- Motivation
- Objective
- Connectivity
- Tools and Connectivity Flow
- Results
- Future
- Acknowledgments

Motivation

- **A Typical SOC**



- **Source:** Hardware and Software: Verification and Testing, 2015

SOC Connectivity Verification

- RTL connections verification.
  - One-to-one inter-module physical connections.
  - Logical equivalence among signals with some possible delay.
- Global connectivity rules are followed:
  - Reset
    - All instances of reset module are driven by reset signal from the reset block.
    - Only reset signal is driving reset type signals.
  - Clocks
    - All clock Gater test enable pins must not be tied off.
    - All internal clocks must be from clock_gater.
  - DFT, Debug, Fuse

Objective

- To find a comprehensive solution to connectivity verification at block-level, partition-level, as well as full-chip.
- Why?
  - Static and Formal tools provides orders of magnitude faster connectivity verification.
  - Higher degree of confidence(never say 100%).
  - Industry wide adaptation.

‘Trust but verify’ Ronald Reagan

CONNECTIVITY

Connectivity Check

- Structural Check:
  - There is a structural path from the ‘src’ to the ‘dest’. A structural path exists if:
    - There is a physical directional path from src to dest.
    - The bit width of the src and dest is equal.
    - The numbers of FFs between the src and dest is less than or equal to the path_delay.
- Functional Connectivity check:

```
enable_expr[*hold_time] |-> dest == $past(src, path_delay)
```

Thinking a Little Higher

- Beyond one-to-one connectivity, we can think of higher level properties:
  - One-to-one restricted
  - One-to-many
    - Examples: Clocks, Resets, DFT, Scan
  - Many-to-one
    - Examples: Bus, Debug
  - Many-to-Many

- Completeness of connectivity specification.

Thinking a Little Smarter

- Why only positive connectivity checks?
  - Existence of a potentially functional path between source and destination.
- Why not check for negative connectivity checks?
  - Non-existence of a path.
- Example cases:
  - No interference among the virtual channels.
  - No data propagation from any primary input to any primary output.
  - No data propagation from any register to the rd_data output.

Checking the Checks

- Completeness check.
  - What if the specification missing some signals present in RTL?
  - We will not know about it because there is no check for it.
- Mutation Analysis
  - Stuck_at_1, Stuck_at_0
  - Inversion
- 0-Delay Circular Connectivity Check
  - A is connected to B with a 0-delay.
  - B is connected to C with a 0-delay.
  - C is connected to A with a 0-delay.
  - We got a combinational loop.

‘Theory without practice is empty;
practice without theory is blind.’ **Kant**

TOOLS AND CONNECTIVITY FLOW

Specification Generation

- Designers' specify interconnection using YAML.
- The RTL infrastructure tools generate shells for the RTL blocks using these YAML specifications.
- The Formal connectivity specification was generated from the same YAML files.
- The specifications can be directly consumed by formal connectivity tool.

```
Name,Clock,Enable,Enable_hold,Source,Destination,Path_delay
hier1_in1_to_hier2_in1,clk,~hier1.sel&en&c_en,2,hier1.in2,hier1.inst.in1,0
hier1_in2_to_hier2_in1,clk,hier1.sel&en&c_en,2,hier1.in2,hier1.inst.in1,1
hier2_outv_to_hier1_outv,clk,hier1.en,1,hier1.inst.outv,hier1.outv,1
```

Formal Connectivity Checking(CC)

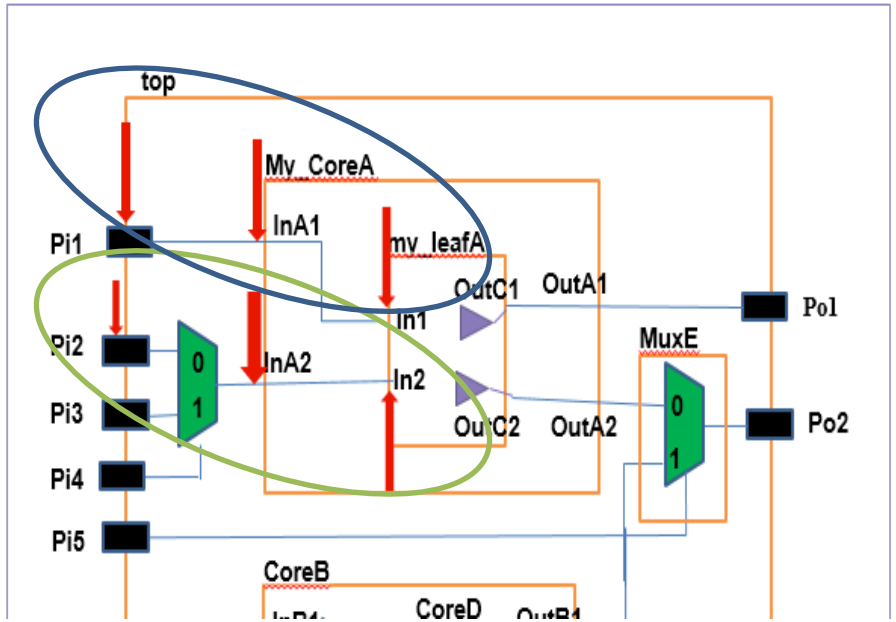
- An **app** in formal verification tool set.
- The CSV output is directly processed as connectivity assertions.
- These assertions are verified against RTL.
- The tool was used to verify Partition Builder(PB) specs for:
  - Reset
  - Full-Chip
  - Design Partitions

Toggle Coverage

- A by-product of the connectivity verification.
- Toggle coverage generation using connectivity
  - Uses the same toggle coverage goals as VCS.
  - Creates a coverage database that can be merged with simulation coverage data.
- Covers the following toggle coverage goals for connected checks:
  - Start and end points.
  - Module input and output ports on path between start and end points.
  - Flip flops in the path, for checks with path delay.
  - Nets in the path.

CC Toggle Coverage Example (© Synopsys)

- The design below have 2 connectivity checks that are proven connected:
  - `add_cc -from Pi1 -to my_coreA.my_leafA.In1 -enable 1`
  - `add_cc -from Pi2 -to my_coreA.my_leafA.In2 -enable ~Pi4`
- Toggle checks at the red arrows will be covered



Formal Testbench Analyzer(FTA)

- Checking the checks.
  - What if CSV missing some signals present in RTL?
  - Formal is only checking what is present in CSV.
- Mutation Analysis.
  - Only top-level connectivity fault inserted.
    - Stuck@1, Stuck@0, Negation.
  - Formal Connectivity tool is used to generate SVA.
    - Assert final (a == b)
  - Formally verify that all of these faults activated/detected if we have these assertions.
  - The thesis: Our connectivity list should be able to catch all these fault cases otherwise we have a missing connection.
- Ensures completeness of CSV.

Extracting High-Level Specs

- We wrote scripts to extract high-level specs from the Partition builder
  - The scripts generated CSV files.
- Each CSV file is divided into four exclusive subsets:
  - One-to-one connections.
  - One-to-many connections.
  - Many-to-one connections.
  - Many-to-many connections.
- Errors in specification:
  - Duplicated connections.
  - Loops

Verifying High-Level Specs

- Used a static verification tool(SpyGlass).
- One-to-one restricted connection.

```
Illegal_path –from {“alpha”} –except_to {“beta”}
```

- One-to-many connection.

```
illegal_path -from {“Reset_Driver.resets”} -except_to {”*resets“}
```

- Many-to-one connection.

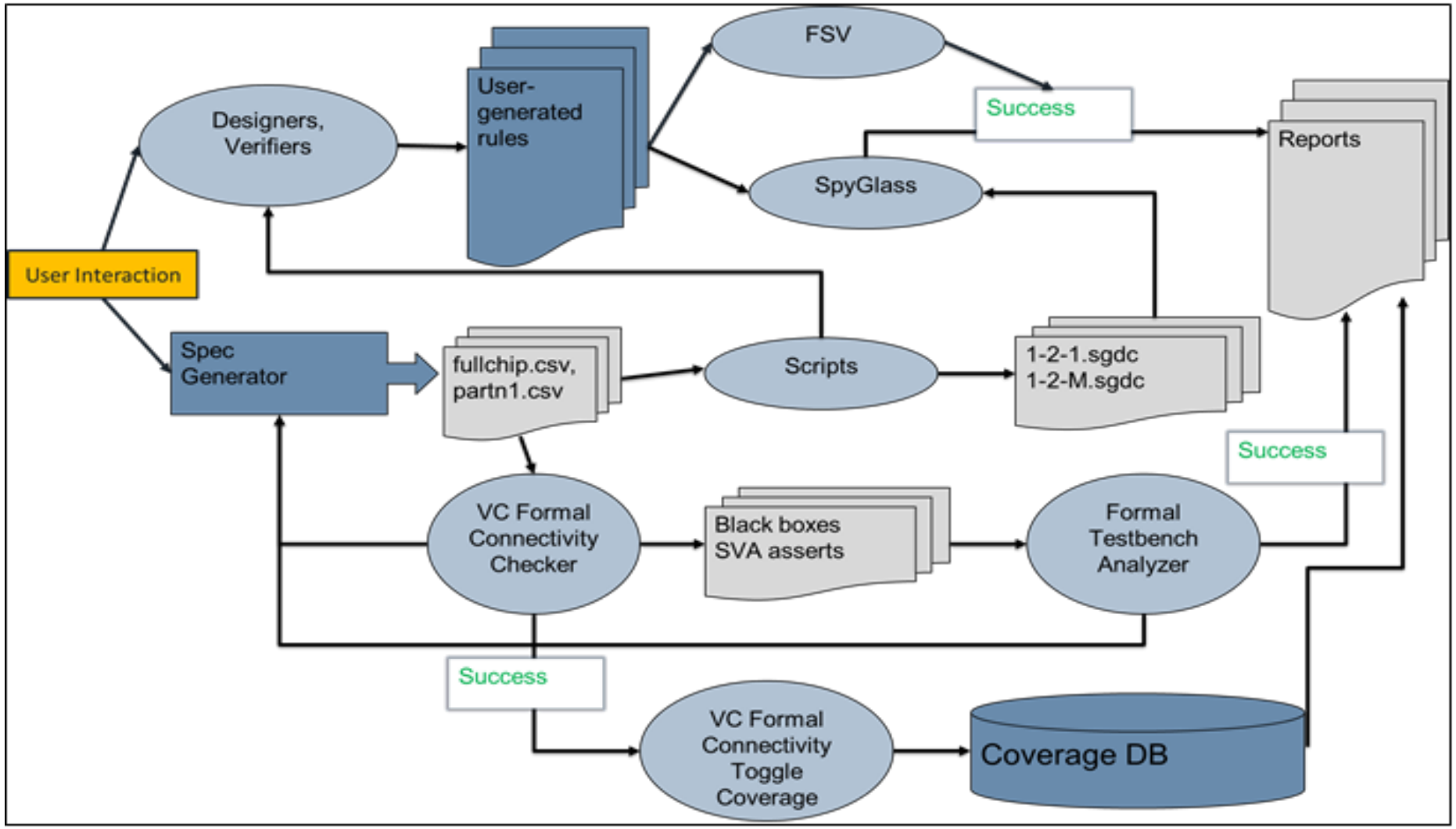
```
Illegal_path –from {“*bus_drivers”} –except_to {“bus”}
```

User-Defined Checks

- SOC integration checking.
- No interference among the virtual channels.
- No data propagation from any primary input to any primary output.
- No data propagation from any register to the rd_data output.
- Formal Security Verification App:
  - Jtag is not influencing the cold_reset behavior in anyway.

```
fsv_generate -name jtag_cold -src jtag  -dest cold_reset
```

The Regression Flow



‘To be or not to be, that is the question!’ Shakespeare

RESULTS

Connections

- Full-Chip
- Design Partitions
- Clocks
- Resets
- Design for Testability(DFT)/Scan
- Fuse
- Debug Bus connections
- Virtual channels

Results

- Found loops in the specification.
- Found missing signals in RTL which were part of the specification.
- Found bad-connections i.e. path does not exist or not properly specified in terms of delay etc.
- Verified 1-1, 1-M, and M-1 rules.
- Verified completeness of specification for multiple partitions as well as full-chip.
- Generated toggle coverage from connectivity proofs.
- Full-chip integration Verification.

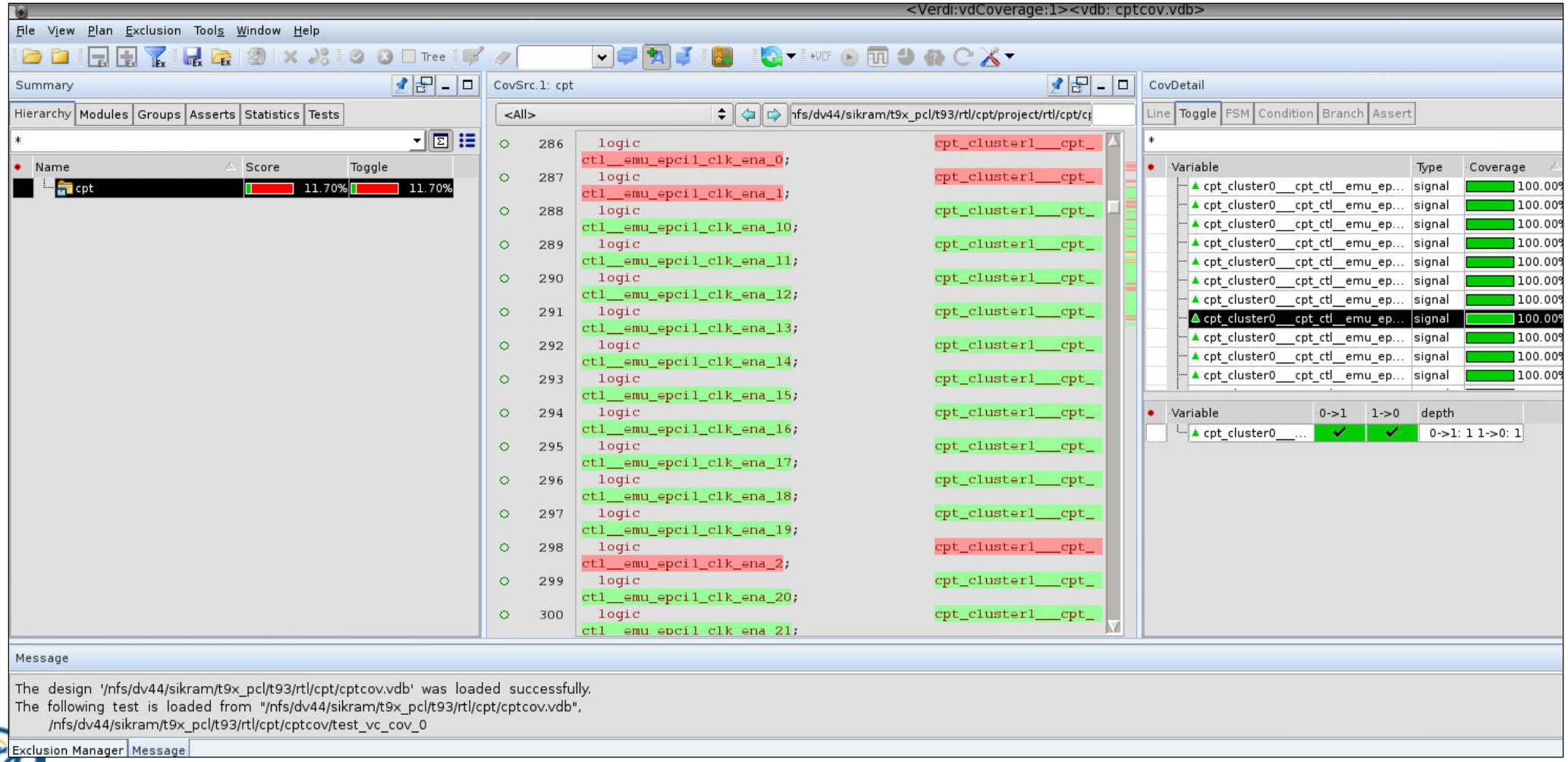
A Sample Bug

MODULE	Results	Non-Activated	Error Signal
C	FPV_FTA ---------- > Fault - # found : 243 - # non_activated: 3 - # detected : 240 > Disabled - # found : 5226 - # assert : 5226	[1062] non_activated - c.fault_id_13 (/rtl/c.v:68) [1173] non_activated - c.fault_id_14 (/rtl/c.v:68) [1284] non_activated - c.fault_id_15 (/rtl/c.v:68)	clk_obs “Present in C.v but missing from C_vcformal.csv” Fix: C_test_fta,clk,,,C.c_clk_wrap___cclk_out,c.clk_obs,0

Formal Vs. Static Tools

Partition	Specs	SpyGlass	VCFormal
A	Total=641 Passed=630 Failed=11	Total Time(S) :6131.36 CPU Time(S) :5960 Peak Memory(MB):23469	Total Time(S) :934.35 CPU Time(S) :216.85 Peak Memory(MB):2676
B	Total=670 passed=670	Total Time(S) :5233.59 CPU Time(S) :5167 Peak Memory(MB):16358	Total Time(S) :544.81 CPU Time(S) :420.78 Peak Memory(MB):7770
C	Total=247 Passed=247	Total Time(S) :256169 CPU Time(S) :251781 Peak Memory(MB):166524	Total Time(S) :11007.50 CPU Time(S) :8527.24 Peak Memory(MB):59224
D	Total=2400 Passed=x Failed = x Undecided=x	Total Time(S) :246990 CPU Time(S) :246236 Peak Memory(MB):102461	No convergence.

Toggle Coverage



Future Work

- Tighter Integration of tools.
- Integrating unreachability coverage analysis into the flow.
  - IP configurations.
  - False failures.
- More innovative use of Formal Security Tool.

Acknowledgments

- It takes a village.
  - Cameron Glass(Marvell)
  - Joel Longever(Marvell)
  - Eddie Langevin(Marvell)
  - Bill Dufresne(Synopsys)
  - Gregory Milano(Synopsys)
  - Anthony Joseph(Synopsys)
  - Anders Nordstrom(Synopsys)

2019 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

THANKS

accellera SYSTEMS INITIATIVE