Accelerating Pre-silicon Verification Efficiency through Innovative AI-Powered Debug Automation

Abhishek Jain
Hani J Poly

Intel Corporation

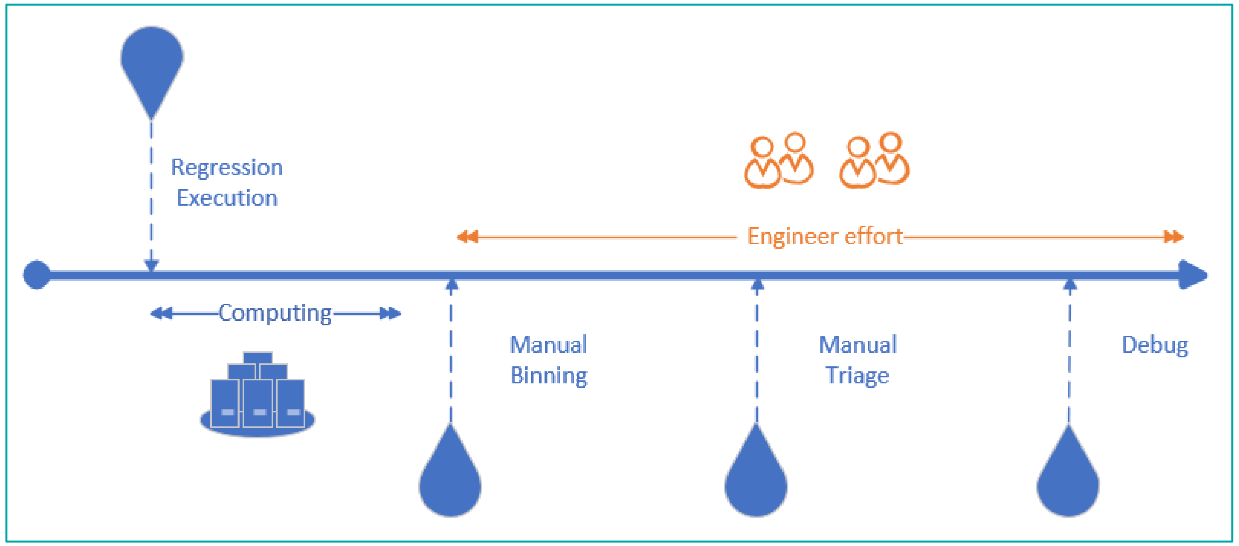
Agenda

- Motivation for RCA
- DUTRCA & TBRCA
- Example case study
- Challenges & Recommendation
- Results & Conclusion

Introduction

- Motivation

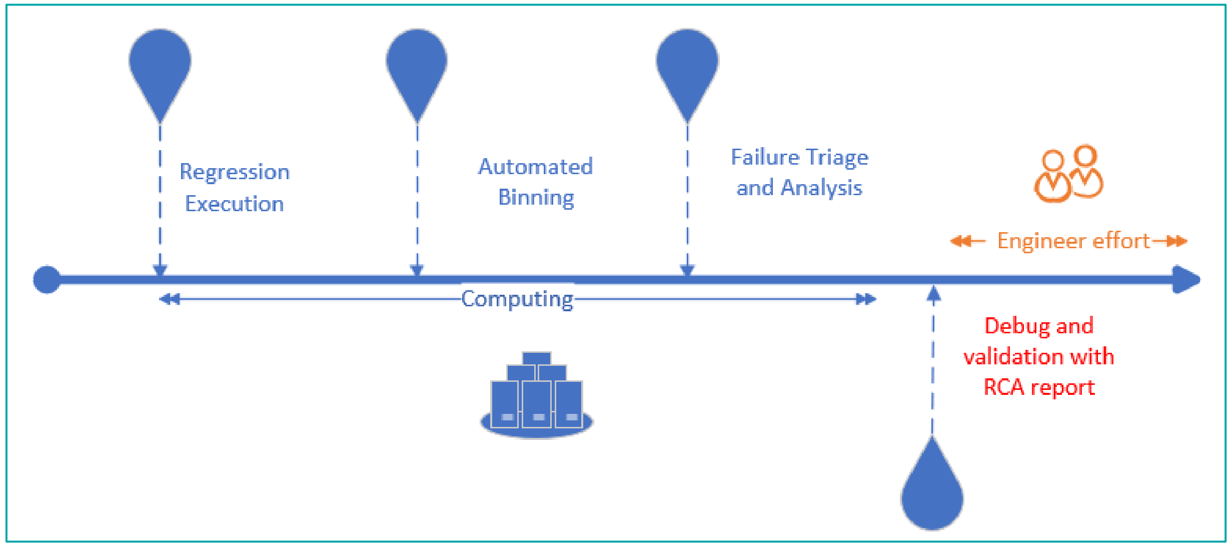
Introduction



- High Engineering effort for Regression Closure.
- Lengthy Manual Debug cycles.
- Debug complexity increases for complex designs, diverse testbench, parsing logs and many more.

- Is there a better approach?.

Motivation


Regression Execution
Automated Binning
Failure Triage and Analysis
Engineer effort
Computing
Debug and validation with RCA report

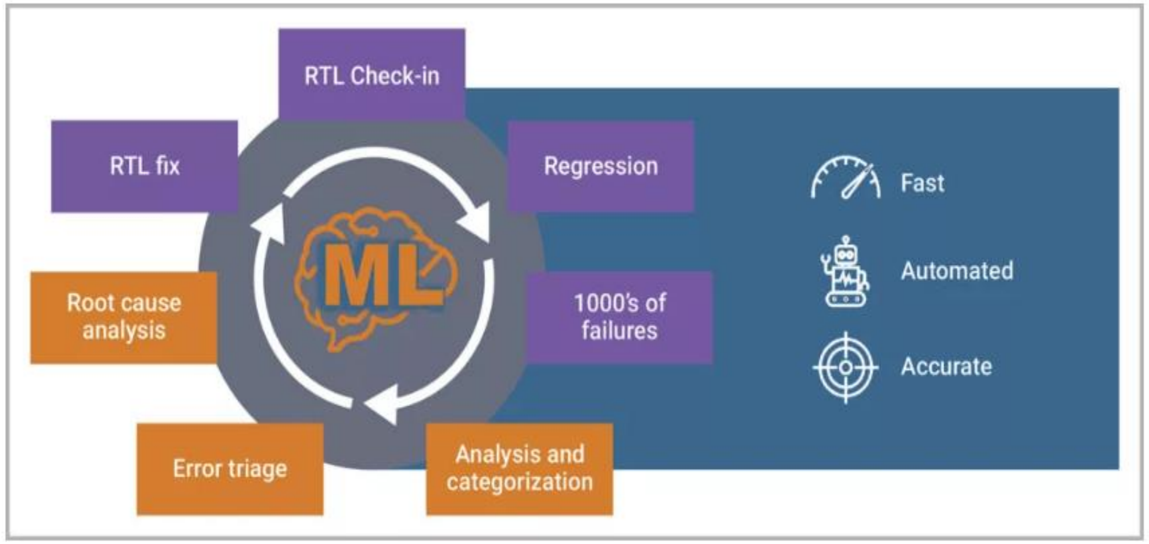
- Effective and assisted debug mode:
  - Speed up debug process with guided debug
  - Reduce/eliminate multiple iterations
- Verification productivity improvement
  - Reduction in turn around time to root cause issues
- Easy to adopt
  - Simple setup guide
  - Quick learning curve for Verdi users

AI powered Root Cause Analysis

- What if one of the most laborious, time-consuming steps in developing a chip could get a jolt of intelligence for faster first-time-right silicon?

- Imagine the opportunities of integrating AI into the chip verification and debugging, particularly in the light of escalating complexity of chips.

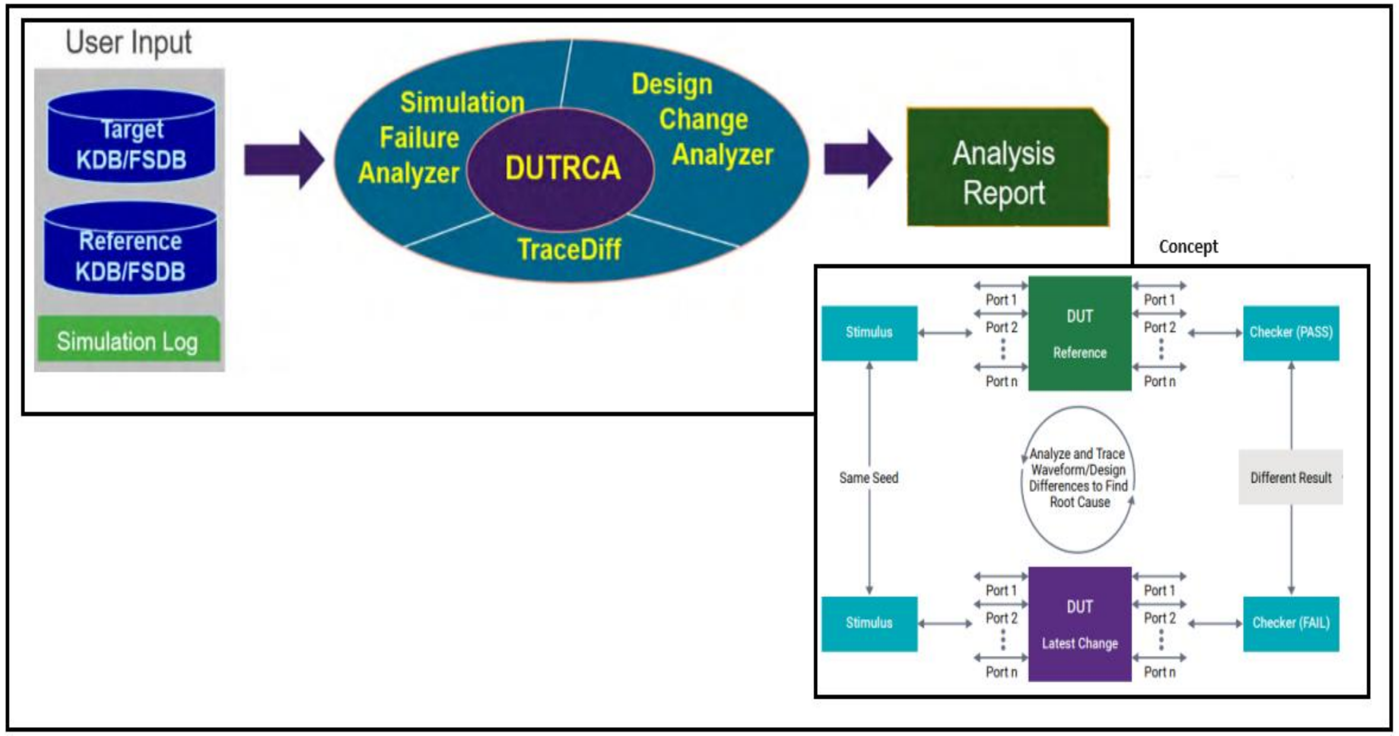
RDA Flow



Source: ai-driven-debug-automation-speeds-root-cause-analysis.html

- Synopsys Regression Debug Automation (RDA) is an advanced solution that uses machine learning, AI framework.
- Uses auto-trace technologies to obtain automatic binning of the failure cases of a regression.
- Automate the process of root cause analysis and tracing.
- Removes manual comparisons of waveforms, source code, and log files

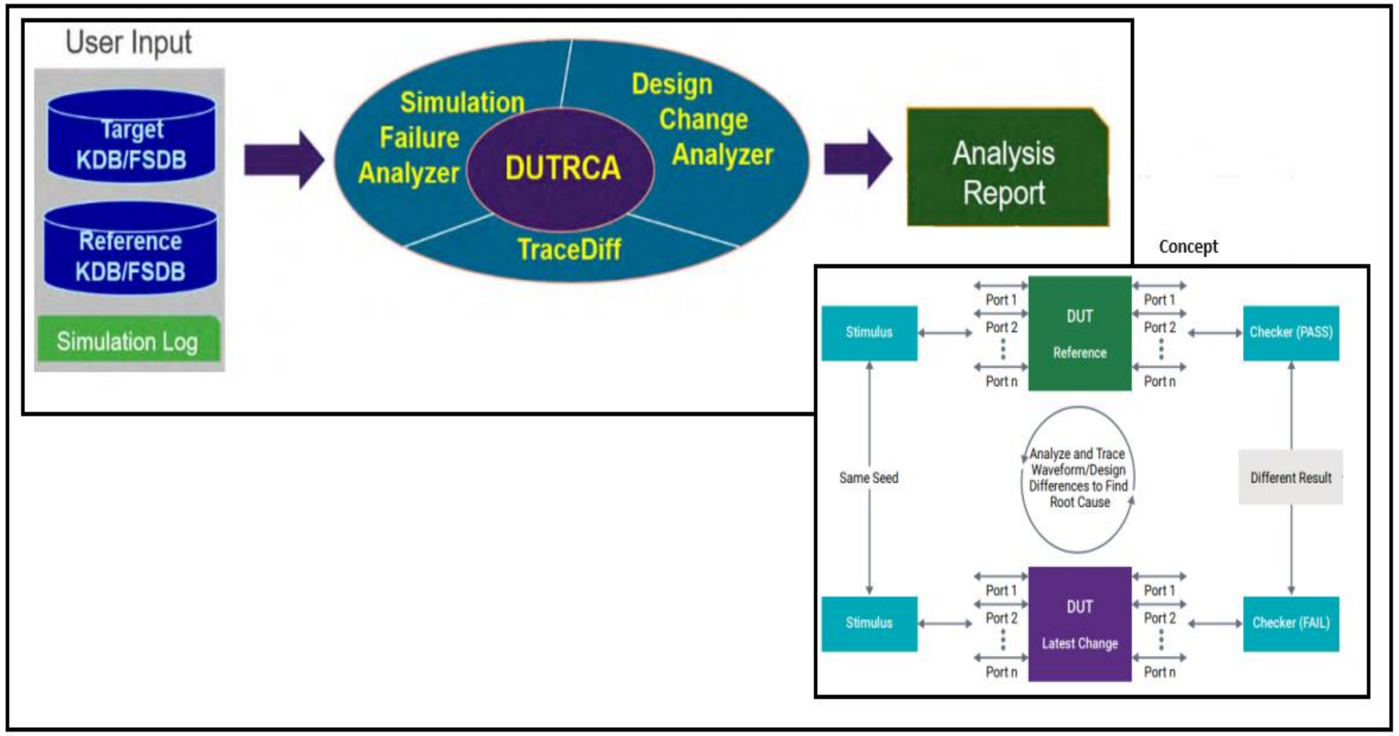
DUT Root Cause Analysis (DUTRCA)



Source: [synopsys/white-papers/verdi-rda-wp]

- Auto Root cause for failure cause by design change via integrated AI solution with following Key engines.
  - Log Analyzer
  - Design difference analyzer
  - TraceDiff & nCompare

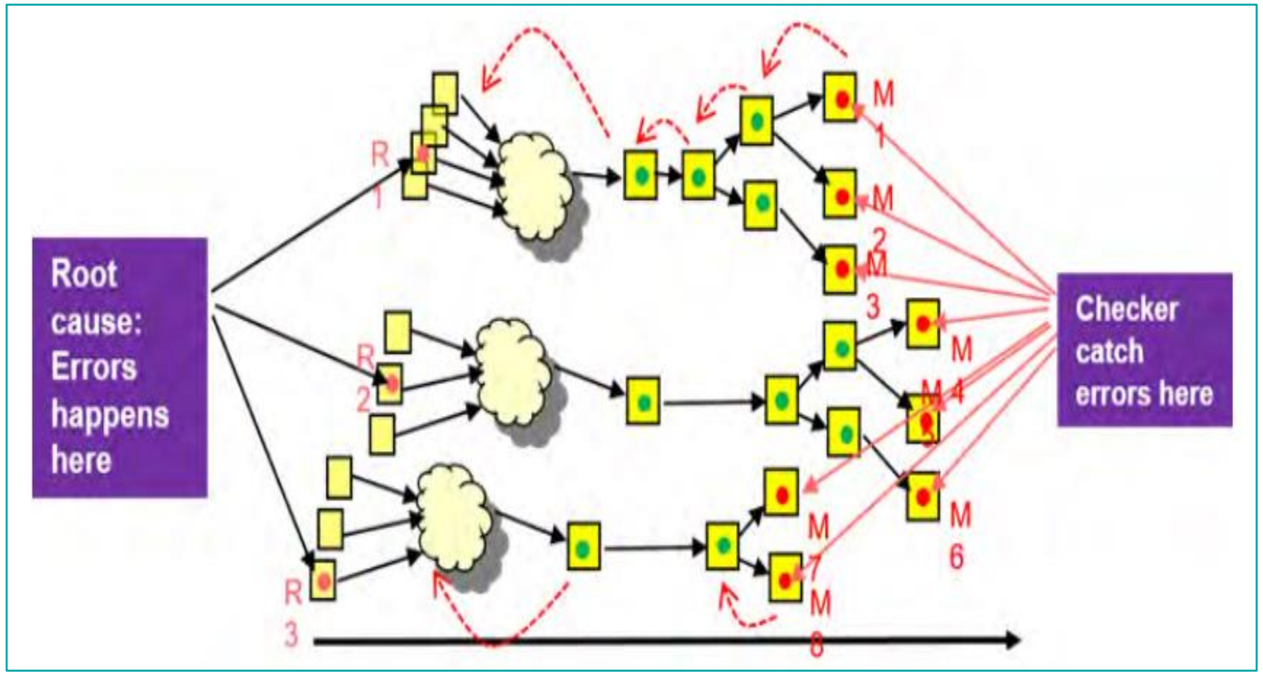
DUT Root Cause Analysis (DUTRCA)



Source: [synopsys/white-papers/verdi-rda-wp]()

- DUTRCA reports in RCA manager.
  - Simulation log Analysis
  - Trace Analysis
  - Design change list.
  - Debug entries with possible root causes.
- Comparative debug mode
  - Dual source code viewer

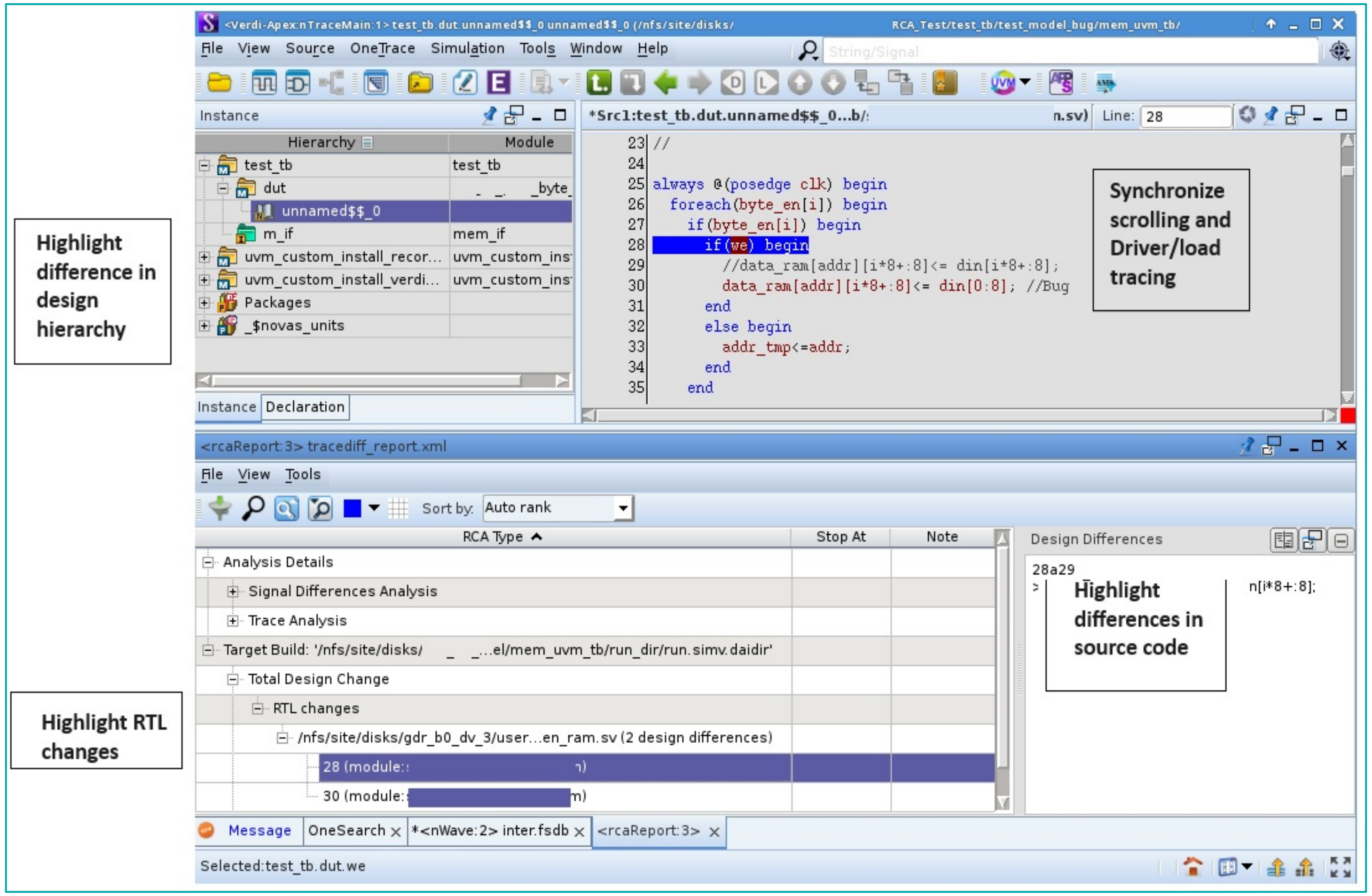
DUTRCA



Source: RDA_UserGuide.pdf

- DUTRCA Automate Value Difference Tracing
  - Detects the waveform shift between the reference and the target signals.
- RCA on Boundary solution
  - Reports that helps user to judge if the problem is due to the testbench changes or DUT changes.

DUTRCA



Highlight difference in design hierarchy
Highlight RTL changes
Synchronize scrolling and Driver/load tracing
Highlight differences in source code
File View Source OneTrace Simulation Tools Window Help
String/Signal
Instance
Declaration
File View Tools

<Verdi-Apex:nTraceMain:1> test_tb.dut.unnamed$$_0 unnamed$$_0 (/nfs/site/disks/ RCA_Test/test_tb/test_model_bug/mem_uvm_tb/

Hierarchy	Module
test_tb	test_tb
dut	_ _. _byte_
unnamed$$_0	
m_if	mem_if
uvm_custom_install_recor...	uvm_custom_ins
uvm_custom_install_verdi...	uvm_custom_ins
Packages	
_$novas_units	

*Src1:test_tb.dut.unnamed$$_0...b/ n.sv) Line: 28

```
23 //
24
25 always @(posedge clk) begin
26   foreach(byte_en[i]) begin
27     if(byte_en[i]) begin
28       if(we) begin
29         //data_ram[addr][i*8+:8]<= din[i*8+:8];
30         data_ram[addr][i*8+:8]<= din[0:8]; //Bug
31       end
32       else begin
33         addr_tmp<=addr;
34       end
35     end
```

<rcaReport:3> tracediff_report.xml

Sort by: Auto rank

RCA Type	Stop At	Note
Analysis Details		
Signal Differences Analysis		
Trace Analysis		
Target Build: '/nfs/site/disks/ _ _...el/mem_uvm_tb/run_dir/run.simv.daidir'		
Total Design Change		
RTL changes		
/nfs/site/disks/gdr_b0_dv_3/user...en_ram.sv (2 design differences)		
28 (module:[redacted])		
30 (module:[redacted]m)		

Design Differences

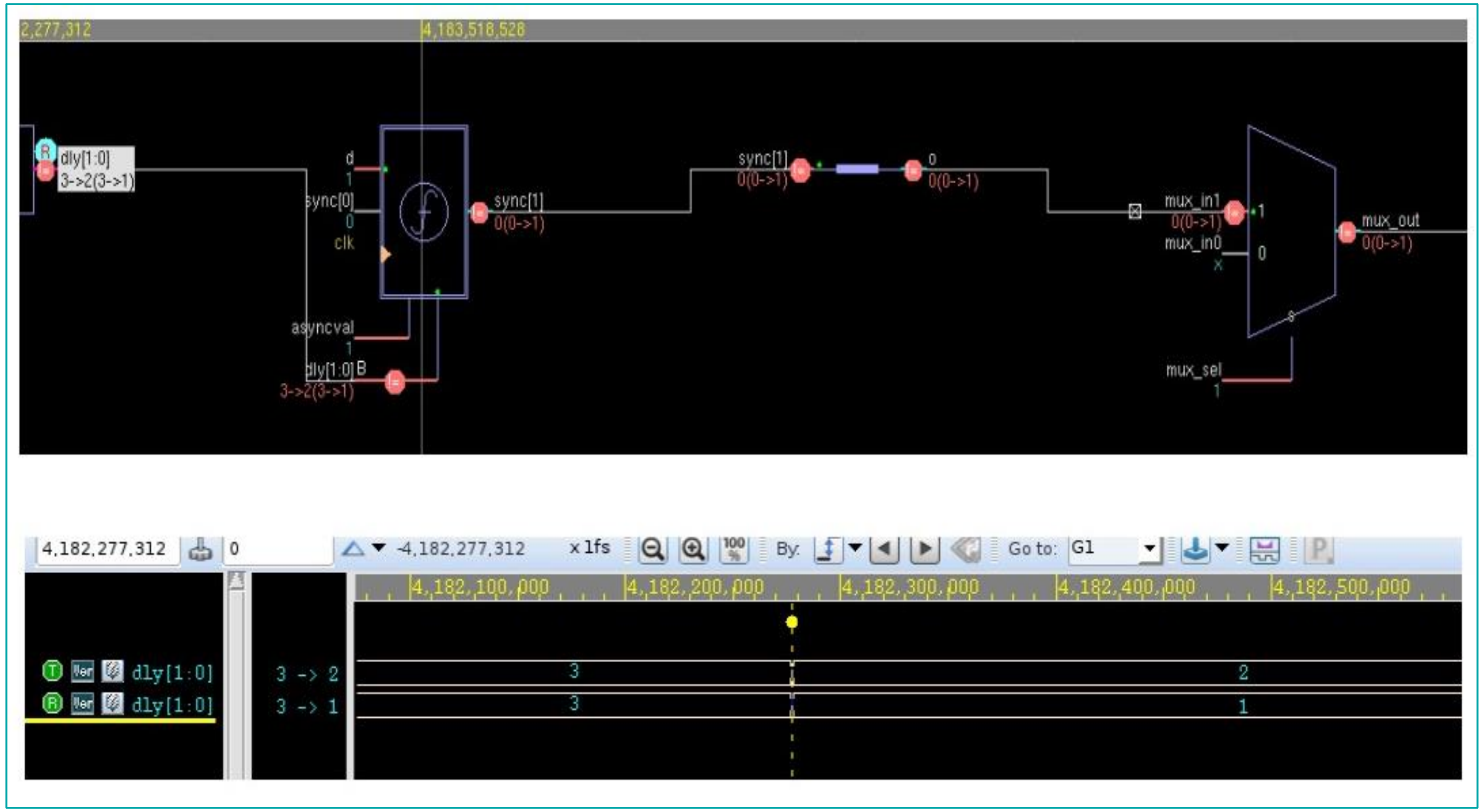
```
28a29
>                n[i*8+:8];
```

Message | OneSearch | *<nWave:2> inter.fsdb | <rcaReport:3>

Selected:test_tb.dut.we

- Repost debug assistance with guided debug entry
  - Side by side RTL change viewer
  - Debug entry list
  - One click access to TFV

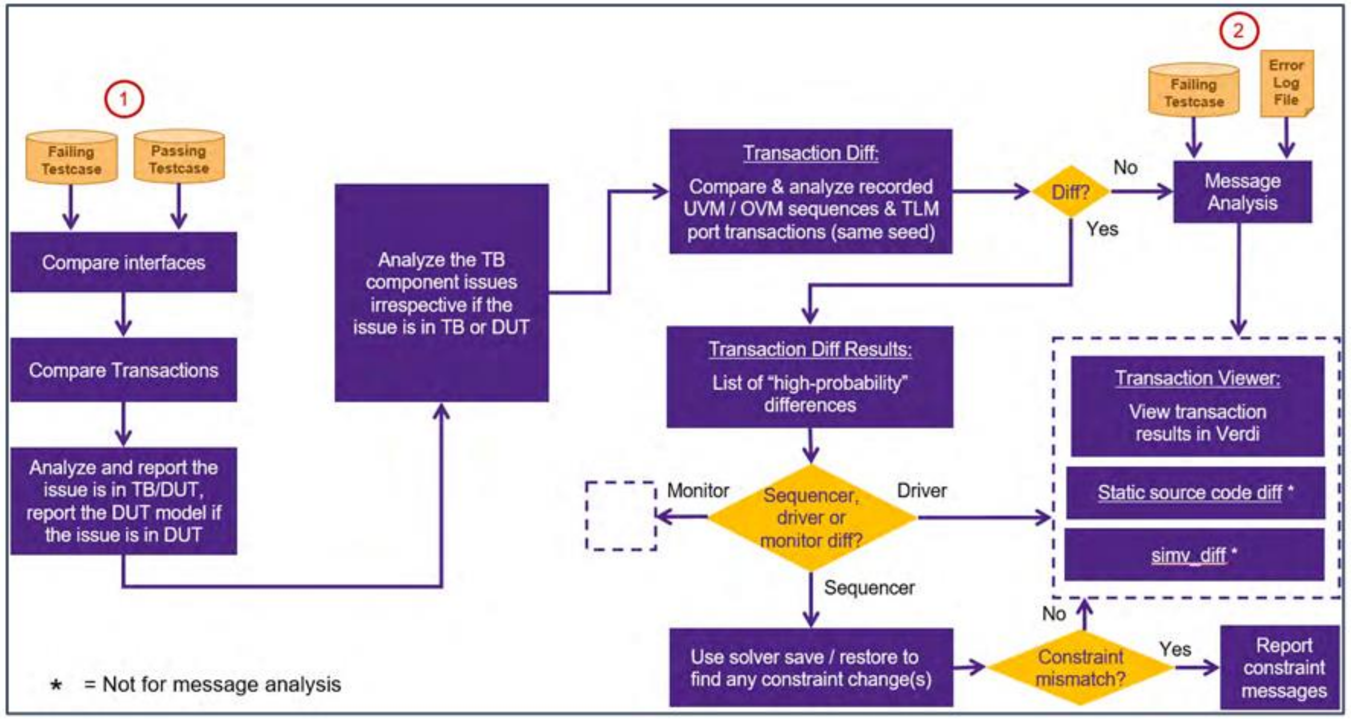
DUTRCA



- Linked Icon to show signal difference between reference and target waveforms.

- Signal mismatches in a single window

Testbench Root Cause Analysis (TBRCA)

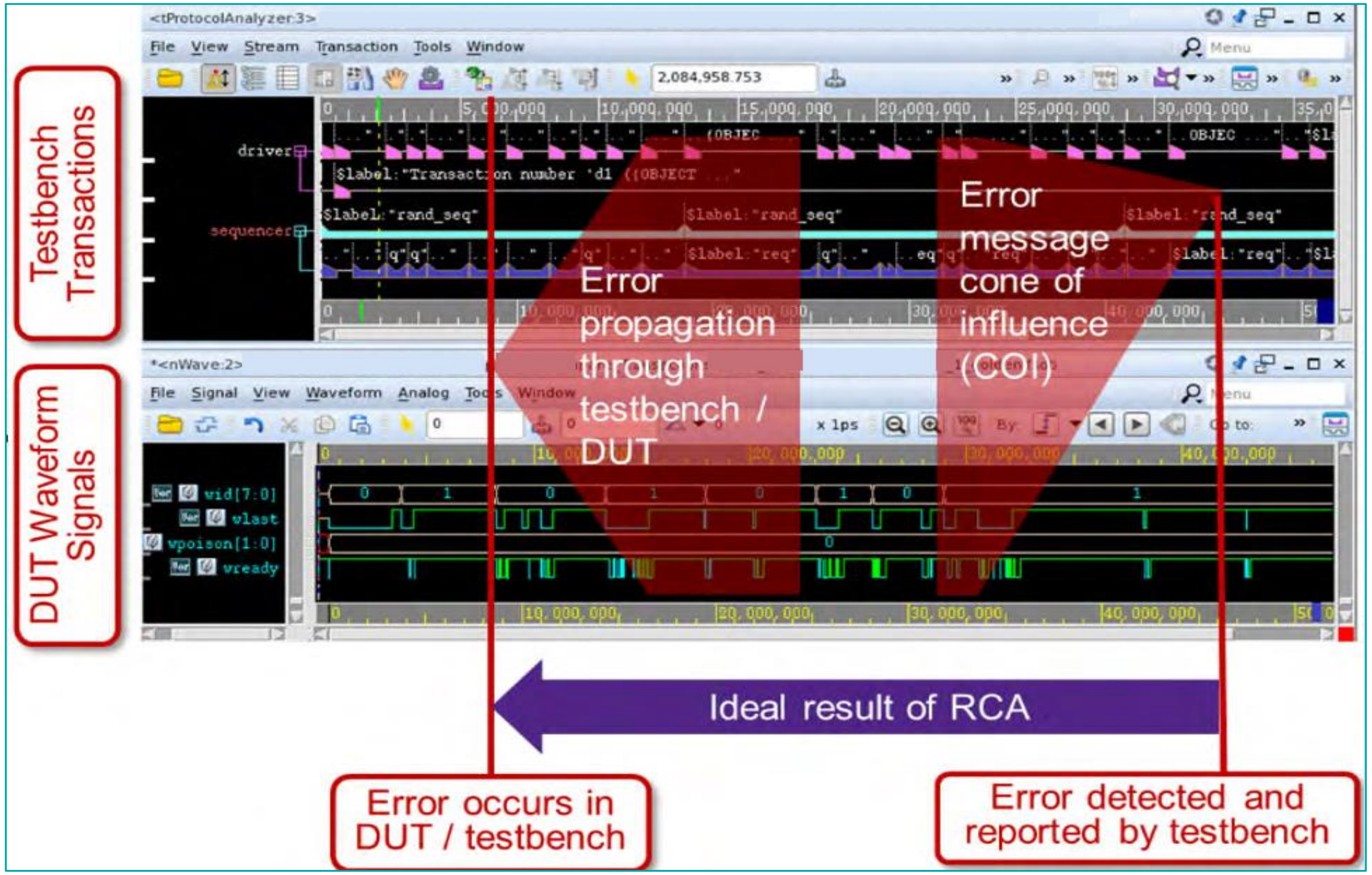


1
Failing Testcase
Passing Testcase
Compare interfaces
Compare Transactions
Analyze and report the issue is in TB/DUT, report the DUT model if the issue is in DUT
Analyze the TB component issues irrespective if the issue is in TB or DUT
Transaction Diff:
Compare & analyze recorded UVM / OVM sequences & TLM port transactions (same seed)
Diff?
No
Yes
2
Failing Testcase
Error Log File
Message Analysis
Transaction Diff Results:
List of "high-probability" differences
Transaction Viewer:
View transaction results in Verdi
Static source code diff *
simv_diff *
Monitor
Sequencer, driver or monitor diff?
Driver
Sequencer
Use solver save / restore to find any constraint change(s)
Constraint mismatch?
No
Yes
Report constraint messages
* = Not for message analysis

Source: RDA_UserGuide.pdf

- Auto Root cause for the failure cause by testbench errors.
- Encompassing the identification of the specific testbench component linked to the error and the precise location within that component where the error originates.
- TBRCA support following approach
  - Transaction Difference Analysis
  - Message Analysis

TBRCA



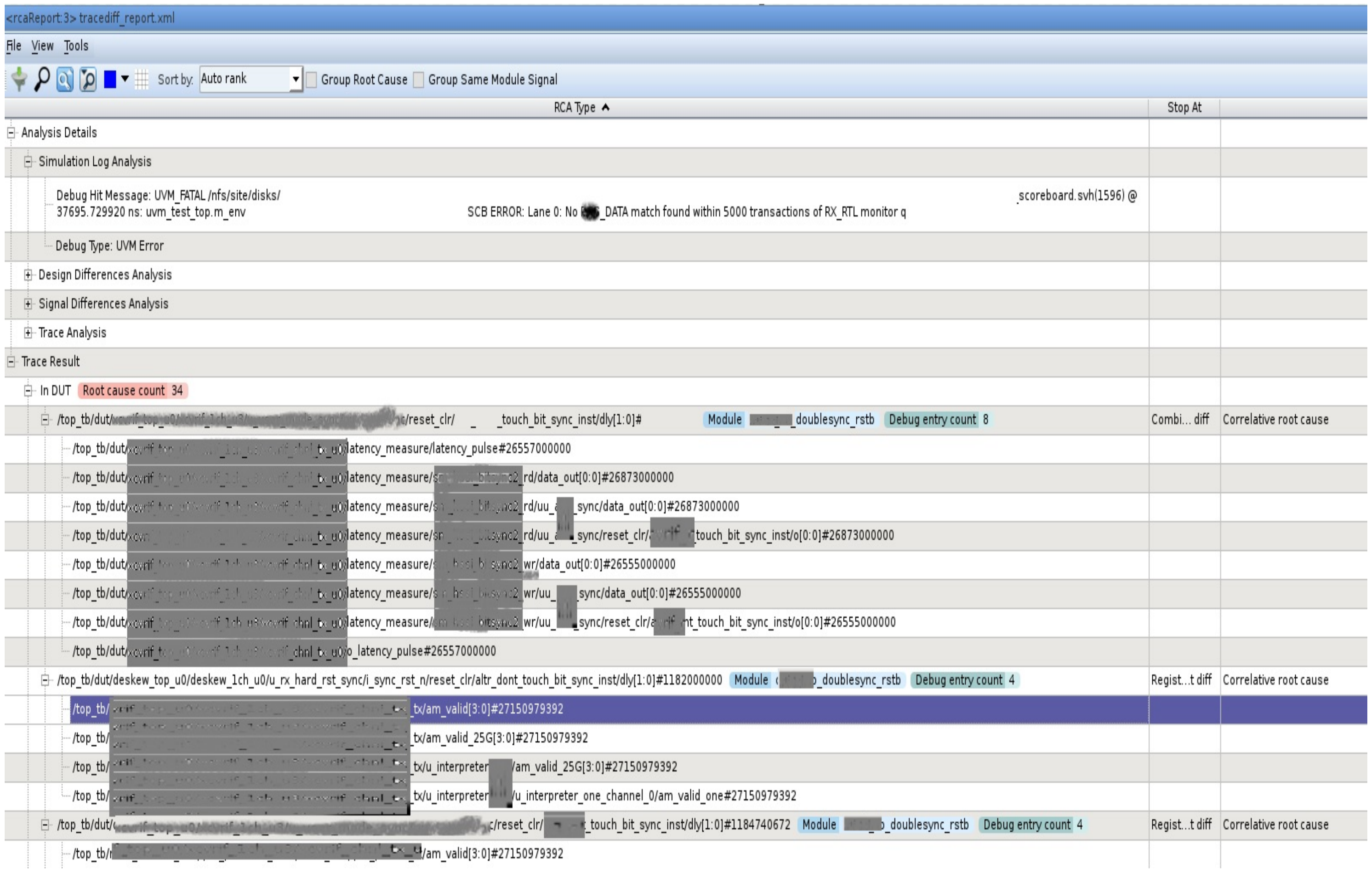
Testbench Transactions
DUT Waveform Signals
Error propagation through testbench / DUT
Error message cone of influence (COI)
Ideal result of RCA
Error occurs in DUT / testbench
Error detected and reported by testbench

Source: [RDA_UserGuide.pdf]

- The cause of the error and the detection of the error.
- Transaction difference Analysis flow.
  - Interface Analysis
  - UVM/OVM characteristic analysis
  - Transaction comparison
  - Verdi GUI debug

Example case study

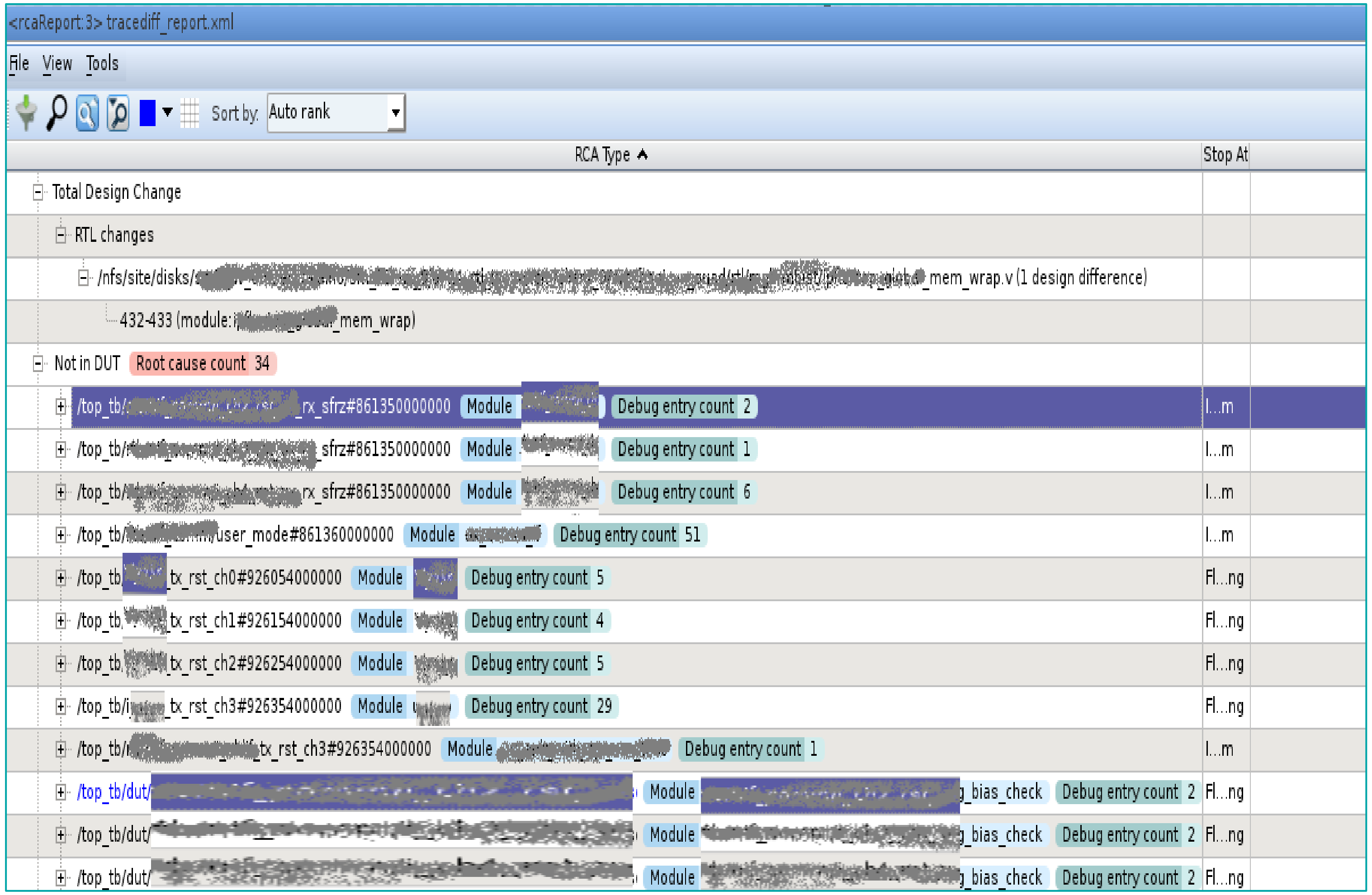
Case Study #1



```
<rcaReport:3> tracediff_report.xml
File  View  Tools
Sort by: Auto rank   Group Root Cause   Group Same Module Signal
                                   RCA Type ^                                                    Stop At
⊟ Analysis Details
  ⊟ Simulation Log Analysis
      Debug Hit Message: UVM_FATAL /nfs/site/disks/                                          scoreboard.svh(1596) @
      37695.729920 ns: uvm_test_top.m_env          SCB ERROR: Lane 0: No [redacted]_DATA match found within 5000 transactions of RX_RTL monitor q
      Debug Type: UVM Error
  ⊞ Design Differences Analysis
  ⊞ Signal Differences Analysis
  ⊞ Trace Analysis
⊟ Trace Result
  ⊟ In DUT  Root cause count 34
    ⊟ /top_tb/dut/[redacted]c/reset_clr/ _ _touch_bit_sync_inst/dly[1:0]#   Module [redacted]_doublesync_rstb  Debug entry count 8   Combi... diff   Correlative root cause
        /top_tb/dut/[redacted]latency_measure/latency_pulse#26557000000
        /top_tb/dut/[redacted]latency_measure/[redacted]rd/data_out[0:0]#26873000000
        /top_tb/dut/[redacted]latency_measure/[redacted]rd/uu_[redacted]sync/data_out[0:0]#26873000000
        /top_tb/dut/[redacted]latency_measure/[redacted]rd/uu_[redacted]sync/reset_clr/[redacted]touch_bit_sync_inst/o[0:0]#26873000000
        /top_tb/dut/[redacted]latency_measure/[redacted]wr/data_out[0:0]#26555000000
        /top_tb/dut/[redacted]latency_measure/[redacted]wr/uu_[redacted]sync/data_out[0:0]#26555000000
        /top_tb/dut/[redacted]latency_measure/[redacted]wr/uu_[redacted]sync/reset_clr/[redacted]nt_touch_bit_sync_inst/o[0:0]#26555000000
        /top_tb/dut/[redacted]o_latency_pulse#26557000000
    ⊟ /top_tb/dut/deskew_top_u0/deskew_1ch_u0/u_rx_hard_rst_sync/i_sync_rst_n/reset_clr/altr_dont_touch_bit_sync_inst/dly[1:0]#1182000000   Module [redacted]_doublesync_rstb  Debug entry count 4   Regist...t diff   Correlative root cause
        /top_tb/[redacted]tx/am_valid[3:0]#27150979392
        /top_tb/[redacted]tx/am_valid_25G[3:0]#27150979392
        /top_tb/[redacted]tx/u_interpreter[redacted]/am_valid_25G[3:0]#27150979392
        /top_tb/[redacted]tx/u_interpreter[redacted]/u_interpreter_one_channel_0/am_valid_one#27150979392
    ⊟ /top_tb/dut/[redacted]c/reset_clr/[redacted]_touch_bit_sync_inst/dly[1:0]#1184740672   Module [redacted]_doublesync_rstb  Debug entry count 4   Regist...t diff   Correlative root cause
        /top_tb/[redacted]/am_valid[3:0]#27150979392
```

- RCA reports based on UVM Fatal/error in log.
- Provided 34 possible DUT root cause for one issue.
  - Each root cause further provide multiple debug entry
  - The second underlying cause actually indicated the necessary debug entry.
  - Among the 34 instances, several root causes lead to irrelevant debug entries.
  - Reference and Target waves with different seeds create considerable noise, necessitating thorough filtering.

Case Study #2



<rcaReport:3> tracediff_report.xml

File View Tools

Sort by: Auto rank

RCA Type	Stop At
Total Design Change	
RTL changes	
/nfs/site/disks/[redacted]_mem_wrap.v (1 design difference)	
432-433 (module:[redacted]_mem_wrap)	
Not in DUT Root cause count 34	
/top_tb/[redacted]rx_sfrz#861350000000 Module [redacted] Debug entry count 2	I...m
/top_tb/[redacted]sfrz#861350000000 Module [redacted] Debug entry count 1	I...m
/top_tb/[redacted]rx_sfrz#861350000000 Module [redacted] Debug entry count 6	I...m
/top_tb/[redacted]user_mode#861360000000 Module [redacted] Debug entry count 51	I...m
/top_tb[redacted]tx_rst_ch0#926054000000 Module [redacted] Debug entry count 5	Fl...ng
/top_tb[redacted]tx_rst_ch1#926154000000 Module [redacted] Debug entry count 4	Fl...ng
/top_tb[redacted]tx_rst_ch2#926254000000 Module [redacted] Debug entry count 5	Fl...ng
/top_tb/[redacted]tx_rst_ch3#926354000000 Module [redacted] Debug entry count 29	Fl...ng
/top_tb/[redacted]tx_rst_ch3#926354000000 Module [redacted] Debug entry count 1	I...m
/top_tb/dut/[redacted] Module [redacted]_bias_check Debug entry count 2	Fl...ng
/top_tb/dut/[redacted] Module [redacted]_bias_check Debug entry count 2	Fl...ng
/top_tb/dut/[redacted] Module [redacted]_bias_check Debug entry count 2	Fl...ng

- RCA reports based on Hung/Timeout scenario. No UVM fatal/error in logs.
- Provided 34 possible root causes .
  - The root cause of the Hung scenario poses a greater challenge for the tool.
  - It compare all signals based on their value differences and offers some analysis with RCA which can be helpful for initial debug.
  - Despite setting the target scope to encompass all hierarchy depths at the top level, the RCA won't display all debug entries compared to the RCA results obtained after manual debugging with a specific target scope.

Challenges & Recommendation

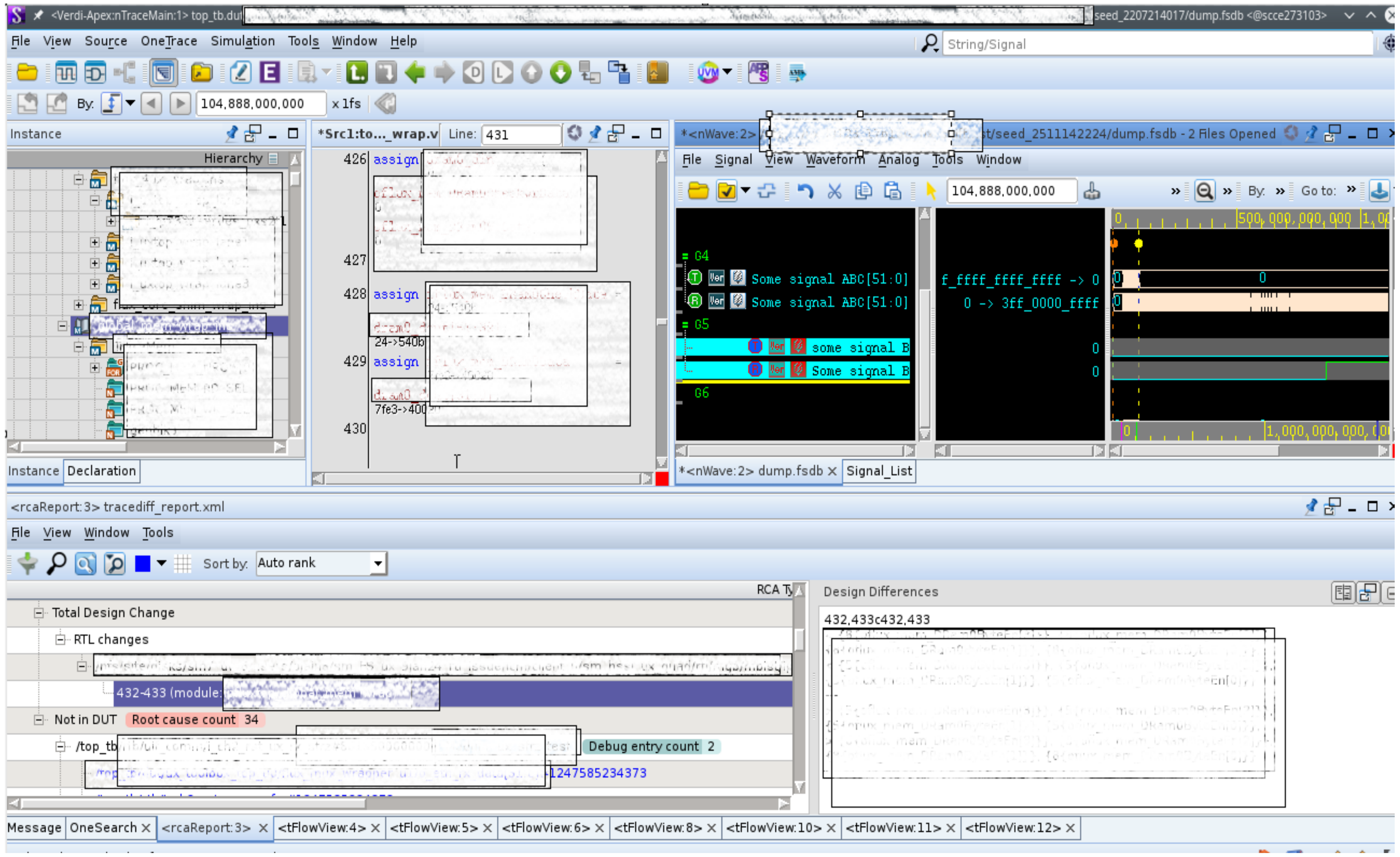
Challenges & Recommendation

- To navigate through multiple debug entries and potential root causes, prior knowledge of DUT and TB is essential to prevent misinterpretation.
- The requirement for both target and reference database, logs, and waveforms is not always feasible.
- Message analysis results depend on the quality of the error message in simulation logs which might mislead root cause analysis.
- Flow recommended to use same seed, although that might not always be feasible.
- Latest vcs/verdi version are needed, and both Target and reference database need to be with same version

Results & Conclusion

Results

DUTRCA : User Interaction

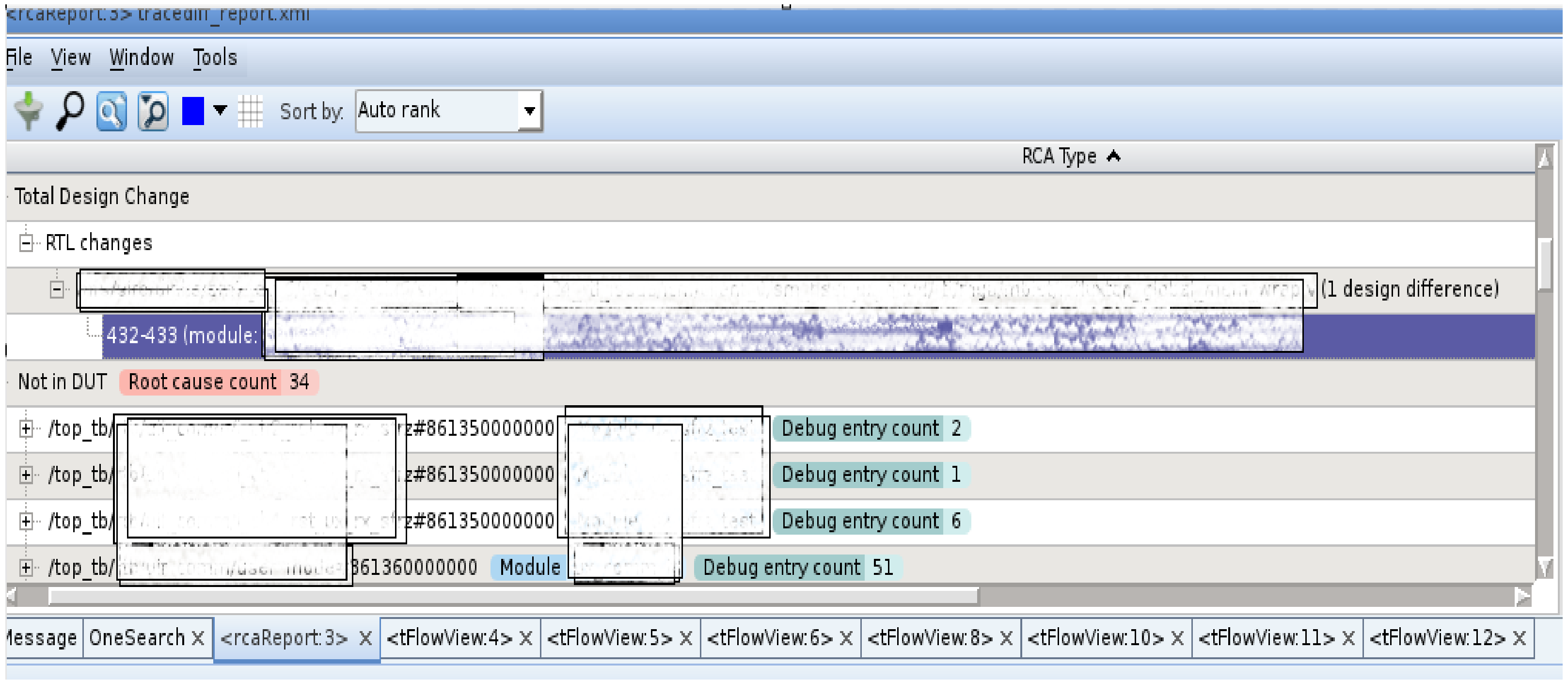


File View Source OneTrace Simulation Tools Window Help
String/Signal
104,888,000,000
Instance
Hierarchy
*Src1:to..._wrap.v Line: 431
*<nWave:2>
File Signal View Waveform Analog Tools Window
Some signal ABC[51:0]
Some signal ABC[51:0]
some signal B
Some signal B
f_ffff_ffff_ffff -> 0
0 -> 3ff_0000_ffff
Instance Declaration
*<nWave:2> dump.fsdb Signal_List
<rcaReport:3> tracediff_report.xml
File View Window Tools
Sort by: Auto rank
Total Design Change
RTL changes
432-433 (module:
Not in DUT Root cause count 34
Debug entry count 2
Design Differences
432,433c432,433
Message OneSearch <rcaReport:3> <tFlowView:4> <tFlowView:5> <tFlowView:6> <tFlowView:8> <tFlowView:10> <tFlowView:11> <tFlowView:12>

- Easy to setup
  - Similar to Verdi
- Highlights all Design changes
  - View side by side design difference
  - High light difference in source code
- RCA reports
  - Root cause entries
  - Easy to navigate
  - Link access to waveform and TFV.

Results

DUTRCA : Accuracy



<rcaReport:3> tracediff_report.xml

File View Window Tools

Sort by: Auto rank

RCA Type

Total Design Change
- RTL changes
  - [redacted] (1 design difference)
    - 432-433 (module: [redacted]

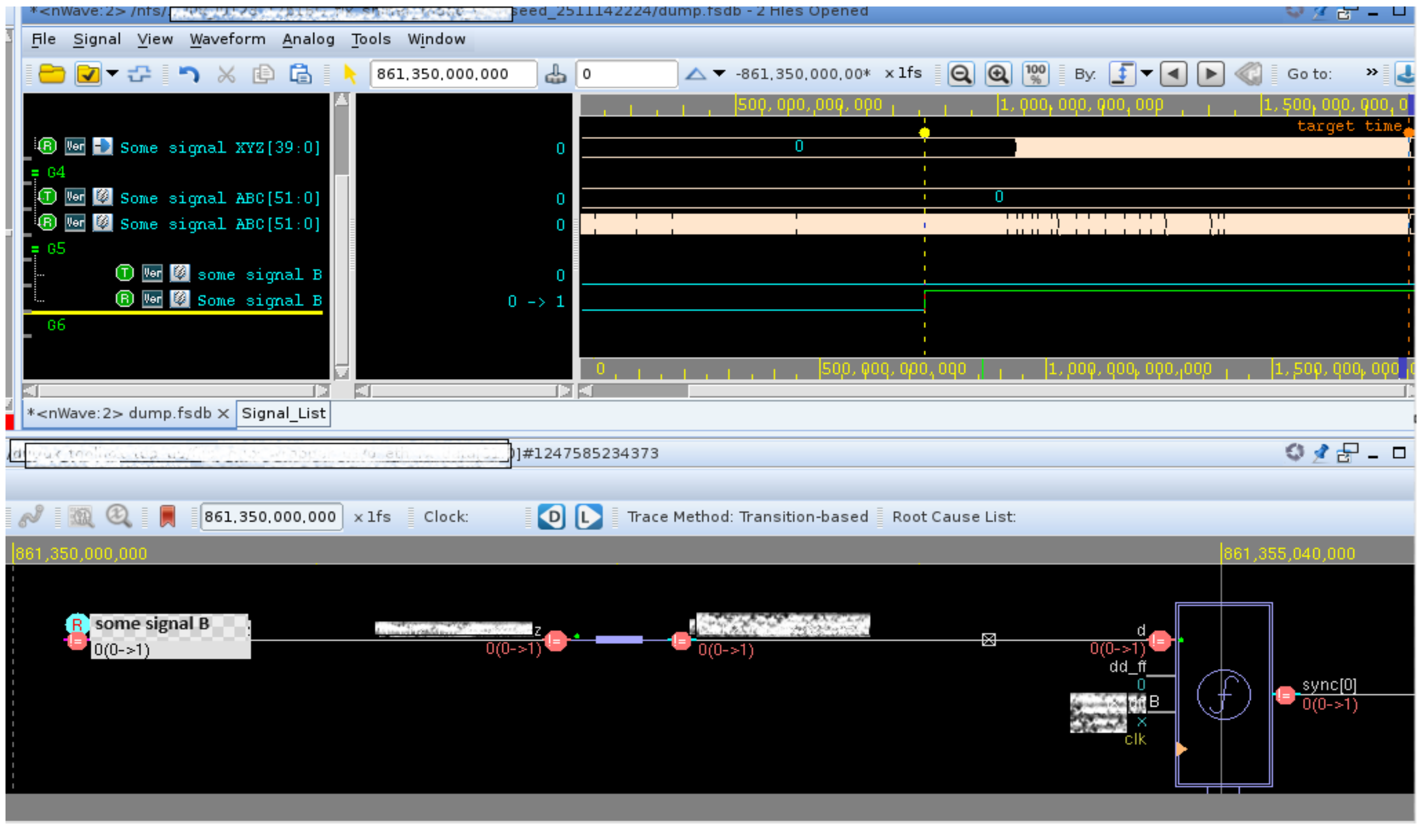
Not in DUT Root cause count 34
- /top_tb/ [redacted] z#861350000000 [redacted] Debug entry count 2
- /top_tb/ [redacted] z#861350000000 [redacted] Debug entry count 1
- /top_tb/ [redacted] z#861350000000 [redacted] Debug entry count 6
- /top_tb/ [redacted] 861360000000 Module [redacted] Debug entry count 51

Message | OneSearch | <rcaReport:3> | <tFlowView:4> | <tFlowView:5> | <tFlowView:6> | <tFlowView:8> | <tFlowView:10> | <tFlowView:11> | <tFlowView:12>

- Successfully identify design differences.
- Suggest multiple debug entries for a single root cause in DUT
  - Multiple root causes sometimes hinders the accuracy.
  - Encountering Hung scenarios can be challenging.

Results

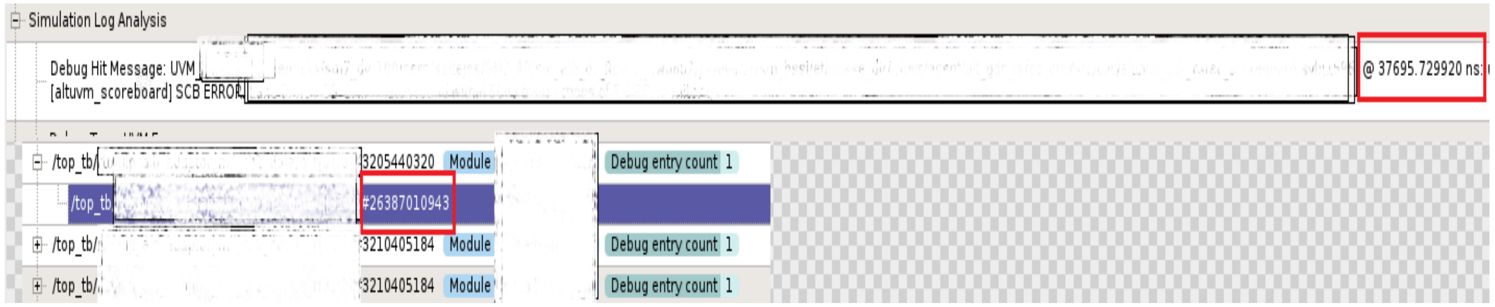
DUTRCA : Debug Acceleration



- Runtime: Highly depends on design complexity, FSDB size and other parameters including hierarchy depth, scope etc.
  - From minutes to hours.
- Compare target & reference designs
- TFV shows the propagation path(s) of the root cause

Results

DUTRCA : Debug Acceleration



```
Simulation Log Analysis
  Debug Hit Message: UVM [redacted] @ 37695.729920 ns:
  [altuvm_scoreboard] SCB ERROR [redacted]
  /top_tb/[redacted] 3205440320  Module [redacted]  Debug entry count 1
    /top_tb[redacted] #26387010943
  /top_tb/[redacted] 3210405184  Module [redacted]  Debug entry count 1
  /top_tb/[redacted] 3210405184  Module [redacted]  Debug entry count 1
```

- Identifies the time gap between the occurrence of the error and its detection which is a need of time and fast track the debug
  - UVM Error flagged at 37695.729920ns
  - Root Cause found at 26387.010943ns
- Analyzes logs for target time

Results

DUTRCA : Analysis Results

- With the utilization of Synopsys AI-powered debug RDA-RCA, it was observed that it helps project teams to achieve
  - Higher levels of verification productivity.
  - Greater accuracy in bug detection with guided debug entry.
  - Root-cause analysis which accelerate the debug turnaround time.

Conclusion

- Revolutionary AI power debug with easy setup guide and simple UI to navigate and single command to remember.
- Saves significant debug time and efforts with well guided debug entry and possible root causes.
- Observe much room for improvement as the tool is currently under active development.
- Can be improved to provide guidance and identify root causes in the absence of a reference scenario, particularly during the initial stages of the project

Acknowledgement

Special thanks to Synopsys R&D team and Application engineer.

- Katike, Rahul (AE)

snug

THANK YOU

Our Technology, **Your Innovation**™