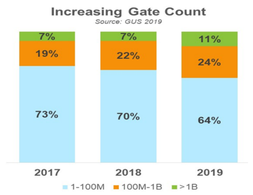
# “Shift left” Hierarchical Low-Power Static Verification Using SAM

Bharani Ellore bharani.ellore@amd.com, Parag Mandrekar parag.mandrekar@amd.com, Himanshu Bhatt himanb@synopsys.com,
Susantha Wijesekara susantha@synopsys.com, Bhaskar Pal bpal@synopsys.com

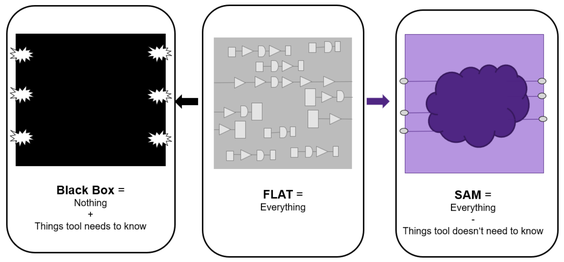
## Why is Signoff Abstract Model (SAM) Flow Critical?

- **Increasing design sizes**
  - Considerable number of designs are > 100M
- **LP Signoff in Flat SoC takes a long time**
  - For some designs more than 24 hours
- **Next generation SoCs growing**
  - Expected to be 10x bigger & can’t rely on full-flat run
- **Integration of large number of IPs with standalone UPF files validated at block level**
- **Existing hierarchical flows do not cover all aspects of Signoff**

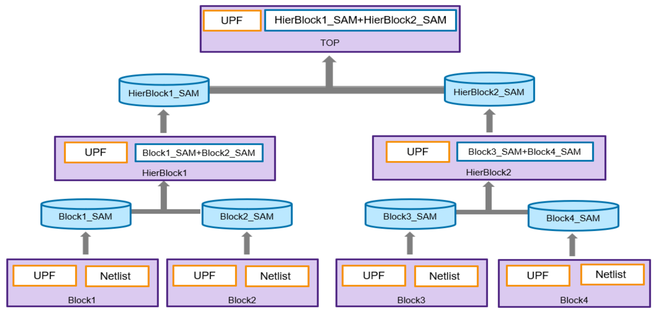



Methodology and tools needed to handle next generation SoC signoff with considerably better performance and capacity, and **NO** loss in QoR or signoff confidence

## SAM vs Blackbox




## SAM Hierarchical Flow Methodology




## Creating and Reading a SAM Model

```
source synopsys_vcst.setup
set search_path "."
set link_library "a.db b.db"

configure_lp_abstraction

read_file -verilog -netlist BlockA.vg -top BlockA
read_upf ./src/BlockA_gate.upf

infer_source   //enable only UPF roots as reqd
foreach f [ glob [runtime_db]/reports/*.tcl ] {
    source $f
}

mark_lp_abstraction

write_verilog_abstract_model -path foo/bar  // -file
option is also supported to give direct file path

report_lp
quit
```

### Writing SAM Model

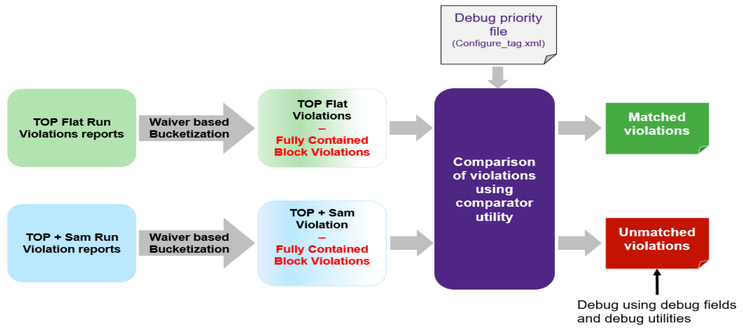
### Reading SAM Model

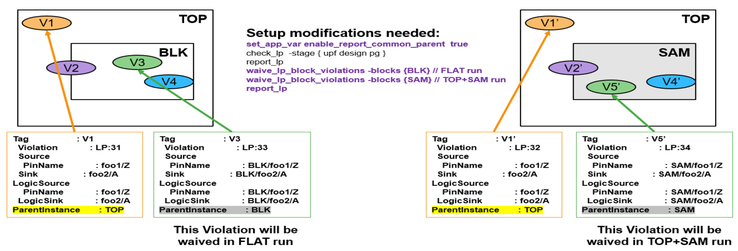
```
source synopsys_vcst.setup
set search_path "."
set link_library "a.db b.db"

configure_lp_tag -tag * -enable  //enable your rule set

set_verilog_abstract_model -module BlockA
set_verilog_abstract_model -module BlockB

read_file -verilog -netlist "
foo/bar/BlockA/Verilog/BlockA_SNPS_VCSTATIC_INM_abstract.
v
foo/bar/BlockB/Verilog/BlockB_SNPS_VCSTATIC_INM_abstract.
v
top.vg"
-top top

load_upf ./src/top.upf
infer_source
foreach f [ glob [runtime_db]/reports/*.tcl ] {
    source $f
}
check_lp -stage {upf design pg}

report_lp
quit
```

## How is QoR Guaranteed?

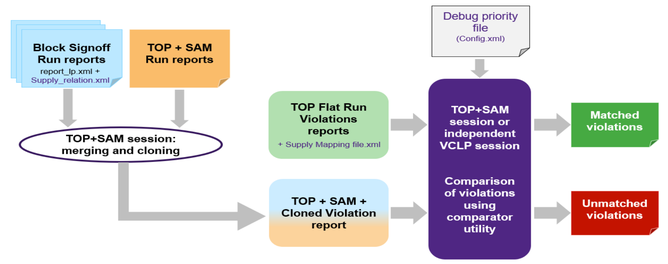
- QoR guarantee is MUST for Signoff confidence
- Multiple QoR validation flows are developed
  - Subtractive QoR Flow:
    - “TOP+SAM” Violations >= Full Flat Violations – Block Violations
  - Additive QoR Flow:
    - Block Violations + “TOP+SAM” Violations >= Full Flat Violations

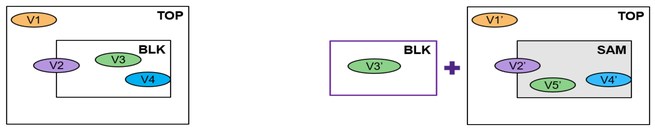
## Subtractive QoR




## Additive QoR




| TOP FLAT | BLK | TOP+SAM | (TOP+SAM) + BLK |
|---|---|---|---|
| V1 : ISO_INST_MISSING | NA | V1' : ISO_INST_MISSING | V1' : ISO_INST_MISSING |
| V2: ISO_BUFINV_STATE | V2' : ISO_BUFINV_STATE | V2' : ISO_BUFINV_STATE | V2' : ISO_BUFINV_STATE |
| V3: DESIGN_PIN_SCMR | V3' : DESIGN_PIN_SCMR | NA | V3' : DESIGN_PIN_SCMR |
| V4 : LS_INST_MISSING | V4' : LS_INST_MISSNG | V4' : LS_INST_MISSING | V4' : LS_INST_MISSING |
| | | V5' : ISO_OUTPUT_UNCONN (Noise due to Abstraction) | V5': ISO_OUTPUT_UNCONN (Noise due to Abstraction) |

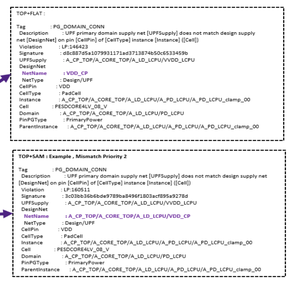
## Compare Violation Utility Usage and Outputs

```
compare_violations      -reference_run report_lp_flat.xml \
                        -implementation_run report_lp_top_sam.xml \
                        -output_dir data \
                        -debug_priority configure.xml \
                        -split
```

| File | Content |
|---|---|
| Summary.csv | Summary of the violation match, mismatch and total counts |
| IgnoredTag.rpt | Tags which are ignored by compare utility. It can be due to custom tags or missing configuration in configulre_lp.xml |
| IgnoredDebugField.rpt | Debug fields of violations which are ignored during comparison. It can be due to custom debug fields or missing configuration in configulre_lp.xml |
| TopFlat_Primary1DebugField_Multi.rpt | List of violations for which reference run has same priority one debug fields |
| OneToOne_Primary1DebugFiled_Matching.csv | Contains pairs of violation ID s from TOP + SAM run and from full flat run in which all the debug fields in priority 1 are matched. It also contains the violations matched due to Priority 2 and Priority 3 debug fields |
| OneToOne_Primary1DebugFiled_MisMatch.csv | Contains violation ID s for which no priority 1 debug fields match found. |
| OneToOne_Primary1DebugFiled_MisMatch.rpt | Contains violation details for which no priority 1 debug fields match found. |
| OneToMany_Primary1DebugFiled_Matching.csv | The implementation (TOP+SAM ) run has more than one violation which matches to a corresponding reference run violation |

## QoR: LP Violations Summary

| Violation Tag | Mismatch Only TOP_FLAT | Mismatch Only TOP_SAM | Exact Match | Mismatch Priority 2 | Mismatch Priority 3 | 1-Many Exact Match | 1-Many Mismatch Priority 2 | 1-Many Mismatch Priority 3 | Total TOP Flat Count | Total TOP SAM Count |
|---|---|---|---|---|---|---|---|---|---|---|
| LS_OUTPUT_UNCONN | 0 | 0 | 33 | 0 | 0 | 0 | 0 | 0 | 33 | 33 |
| LS_INST_REDUND | 0 | 0 | 193 | 0 | 0 | 0 | 0 | 0 | 193 | 193 |
| LS_INPUT_TIEHI | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 1 | 2 |
| ISO_STRATEGY_UNUSED | 0 | 0 | 285 | 0 | 0 | 0 | 0 | 0 | 285 | 285 |
| ISO_OUTPUT_UNCONN | 0 | 0 | 575 | 0 | 0 | 150 | 0 | 0 | 650 | 725 |
| ISO_DATA_CONSTANT | 0 | 0 | 160 | 0 | 0 | 1124 | 0 | 0 | 722 | 1284 |
| ISO_BUFINV_FUNC | 0 | 0 | 8594 | 0 | 0 | 0 | 0 | 0 | 8594 | 8594 |
| UPF_CSN_SRM | 0 | 0 | 6 | 0 | 0 | 6 | 0 | 0 | 6 | 12 |
| ISO_DATA_BLOCKED | 0 | 0 | 0 | 0 | 0 | 24 | 0 | 0 | 12 | 24 |
| UPF_CSN_LS | 0 | 0 | 1734 | 0 | 0 | 264 | 0 | 0 | 1866 | 1998 |
| ISO_STRATEGY_CONFLICT | 0 | 0 | 38 | 0 | 0 | 0 | 0 | 0 | 38 | 38 |
| ISO_STRATCLAMP_MISMATCH | 0 | 0 | 11 | 0 | 0 | 0 | 0 | 0 | 11 | 11 |
| PSW_MAP_WRONG | 0 | 0 | 62 | 0 | 0 | 2 | 0 | 0 | 63 | 64 |
| PG_DOMAIN_CONN | 0 | 0 | 7 | 3 | 0 | 63 | 63 | 0 | 73 | 136 |
| UPF_HIERSRSN_CONN | 0 | 0 | 18 | 0 | 0 | 22 | 0 | 0 | 40 | 40 |
| LS_INST_MISSING | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 2 | 2 |
| ISO_MAP_WRONG | 0 | 0 | 6 | 0 | 0 | 2 | 2 | 0 | 10 | 10 |




## Results on Customer Designs

| Design | Runtime Flat | Runtime TOP+SAM | Speed-up | Peak Memory Flat | Peak Memory TOP+SAM | Improvement |
|---|---|---|---|---|---|---|
| **Design 1** (Full Chip Netlist) | 39641 | 4778.23 | 8.3x | 194155 | 18651 | 10.4x |
| **Design 2** (Full Chip Netlist) | 68166 | 4620 | 14.8x | 388751 | 25694 | 15.1x |
| **Design 3** (Full Chip PG Netlist) | 241370 | 29834 | 8.1x | 572142 | 64883 | 8.8x |
| **Design 4** (Full Chip Netlist) | 24280 | 6653 | 3.6x | 106983 | 22238 | 4.8x |
| **Design 5** | 14212 | 2591 | 5.5x | 109622 | 17294 | 6.4x |
| **Design 6** | 8884 | 1900 | 4.7X | 94451 | 14923 | 6.4X |
| **Design 7** | 2415 | 1184 | 2x | 25192 | 11679 | 2.15x |
| **Design 8** | 1704 | 1120 | 1.5x | 12533 | 9258 | 1.4x |
| **Design 9** | 67988 | 48510 | 1.4X | 414330 | 277061 | 1.5X |
| **Design 10** | 87750 | 38998 | 2.25X | 653185 | 317904 | 2X |

## VC LP Hierarchical Verification Flows and Tradeoffs