2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION INDIA
Sept 25-26, 2014
Hotel Park Plaza
Bangalore, India

Low power Verification challenges and coverage recipe to sign-off Power aware Verification

Deepmala Sachan , Intel technology India Pvt Ltd , Bangalore , India (deepmala.sachan@intel.com)

Thameem Syed S, Intel technology India Pvt Ltd , Bangalore , India (syed.s.thameem@intel.com)

Raghavendra Prakash, Intel technology India Pvt Ltd , Bangalore , India (raghavendra.prakash@intel.com)

Venugopal Jennarapu,, Intel technology India Pvt Ltd , Bangalore , India (venugopal.jennarapu@intel.com)

The complexity of low-power designs has been going up drastically. Verification is complex enough even without power, adding power management into the fusion makes things much harder using traditional verification approaches. The fundamental problem with low-power verification is the explosion in scope and complexity caused by low-power design techniques such as clock gating, power gating, state retention strategies and dynamic voltage scaling. These new features increase the overall verification effort hence the functionality of the design must be verified for all added power modes, power sequences, and power-mode along with power management logic. This requires an appropriate coverage analysis of low power objects (Power state Table (PST), Power switches & Protection policies etc).In this paper we will also discuss how we have done the detailed coverage analysis on low power objects and faced series of challenges/setbacks and debug solutions which can help folks across the industry.

I. Introduction

Today’s most system - on-chip (SoC) designs are composed of different blocks running multiple applications with varying power requirements. This demands SoCs to have multiple power domains as well as various operating modes with different and dynamically variable voltage levels. The fundamental problem with low-power verification is the explosion in scope and complexity caused by low-power design techniques as designer captures the power intent such as power switch rules, retention, isolation, and level-shifter rules in UPF (Unified Power Format) by separating from the RTL code. Addition of power format files along with RTL increase the low power verification effort hence as a part of complete verification process all the added power modes, power sequences, and power-mode transitions need to be covered such that the design meets both functional and power intent.

Until recently, SoC and IP designs were either single power domain or power intent embedded in the RTL. Focus was on functional verification only and least on anything specific to power. However, with the need for a consistent way to describe power strategies in the design led to the IEEE 1801 standard, UPF. With the introduction of UPF as a separate and standard power specification format, it is essential that the UPF be verified along with the RTL such that the design meets both functional and power intent.

Low Power aware (PA) verification ensures

- Correctness of design when Power Domains (PD) are OFF/ design is Powered-down
- Power domains come up in good known states by signal restoration of retained values for the sequential elements

We have enabled the low-power coverage driven verification by analyzing the coverage on low power objects to modify the test bench and addition of the new test sequences by isolating the uncovered low power scenarios. As methods like static and dynamic checks are not sufficient enough hence it is extremely important to plan low power specific coverage in test plan from start and track till verification.

In the later sections we will discuss on how we have done the detailed coverage analysis for Power state tables (PST), Power switches & Protection (Retention /Isolation) policies to adapt low power coverage and faced series of challenges/ setbacks which can help other folks across the industry as The Quality of Power Aware verification is measured through “Low Power coverage” and it improves the power aware verification sign-off confidence.

This paper will also unveils the challenges and methodology used for efficient power aware verification, low power debug capabilities and systematic approach to share the findings/limitations/recommendation and methodology to plug-in low power coverage to functional coverage for bug-free power aware verification.

II. LOW POWER VERIFICATION CHALLENGES

We have enabled the power aware simulator to enable the verification of power managed designs at the RTL and stumbled upon various power-related errors during low power simulation by mimicking real silicon behavior. It helped us to comprehend the correctness of the design with respect to the multiple voltages used across several blocks at a very early stage in the design flow. Despite using Power aware simulator which has built-in automated assertions to monitor power on/off sequences and automated coverage of power objects to measure the LP verification progress for all LP designs we realized that the process of debugging in power aware simulation is much more complex as an error found in power-aware designs may well be caused by RTL functions or by the behavior defined in the power-specification.
Verification of “Power intents” in aggregation with RTL remains a challenge with sprouting power format. One of the Major challenge in Power aware verification is to validate all valid states /Transitions and report invalid state/ transitions early in design cyle.
Refer Fig-2
Lets say we have 2 supplies- V1 & V2
Possible states: 2^2 (s1, s2, s3 & S4)
Illegal transition :“s1->S4” and “s4 ->s1”
Power Aware verification should be able to
Key Low power verification challenges are:

- To retain sufficient state information to enable restoration of functionality when power is restored
- Power switch off/on duration
- Transitions between different power modes and states
- Interface between power domains
- Missing of level shifters, isolation and retention
- Legal and illegal transitions
- Clock enabling/toggling
- Verifying retention registers and isolation cells and level shifter Strategies

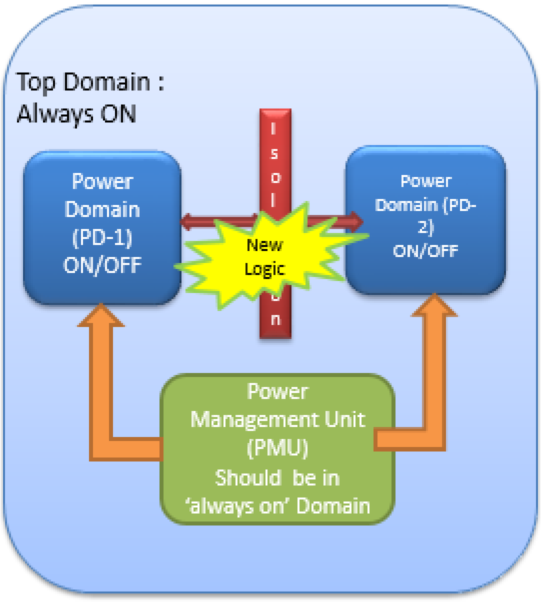
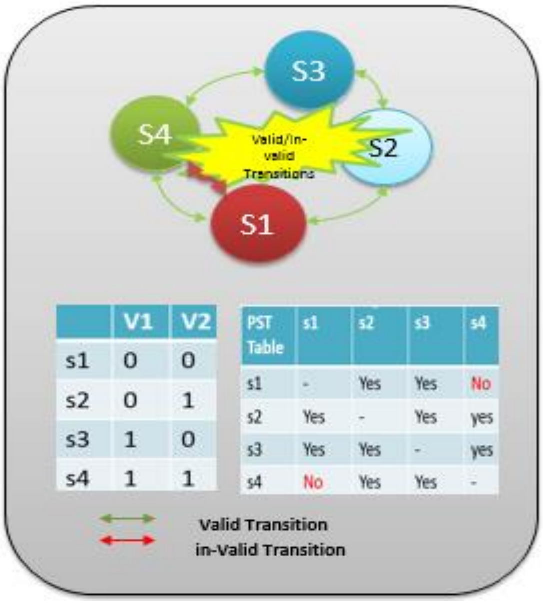

Top Domain :
Always ON
Power Domain (PD-1) ON/OFF
I s o l
n
New Logic
Power Domain (PD-2) ON/OFF
Power Management Unit (PMU) Should be in 'always on' Domain

FIG-1 POWER AWARE DESIGN



	V1	V2
s1	0	0
s2	0	1
s3	1	0
s4	1	1

PST Table	s1	s2	s3	s4
s1	-	Yes	Yes	No
s2	Yes	-	Yes	yes
s3	Yes	Yes	-	yes
s4	No	Yes	Yes	-

🟩 Valid Transition
🟥 in-Valid Transition

FIG-2 POWER AWARE VERIFICATION

III. TYPICAL DEBUG SCENARIOS IN POWER AWARE (PA) VERIFICATION

Figure 3 is a representative version of the design with low power features implemented. Power management unit (PMU) represented in green box implements the power state machines to control the complete power sequencing. This block is in 'always ON' domain i.e. there is no case when the chip will be powered but this logic will be shut down. The blue block is the UnitA while the red block is UnitB. There are 2 power domains (PD1 & PD2) defined wherein the PD1 can be shutdown independently of PD2 (Sleep mode) or both PD1 and PD2 can be shut down together (Deep Sleep mode). In absence of qualified signals from UnitB for a defined period, the power manager initiates Sleep mode. During Sleep mode, if there is a wake up interrupt (from different sources with one represented here), the power manager initiates the wake up cycle. To avoid re-initialization of the UnitA, it is expected that the last functional state of the PD1 be retained. State retention cells help in saving the state of the block and restoring it when required.

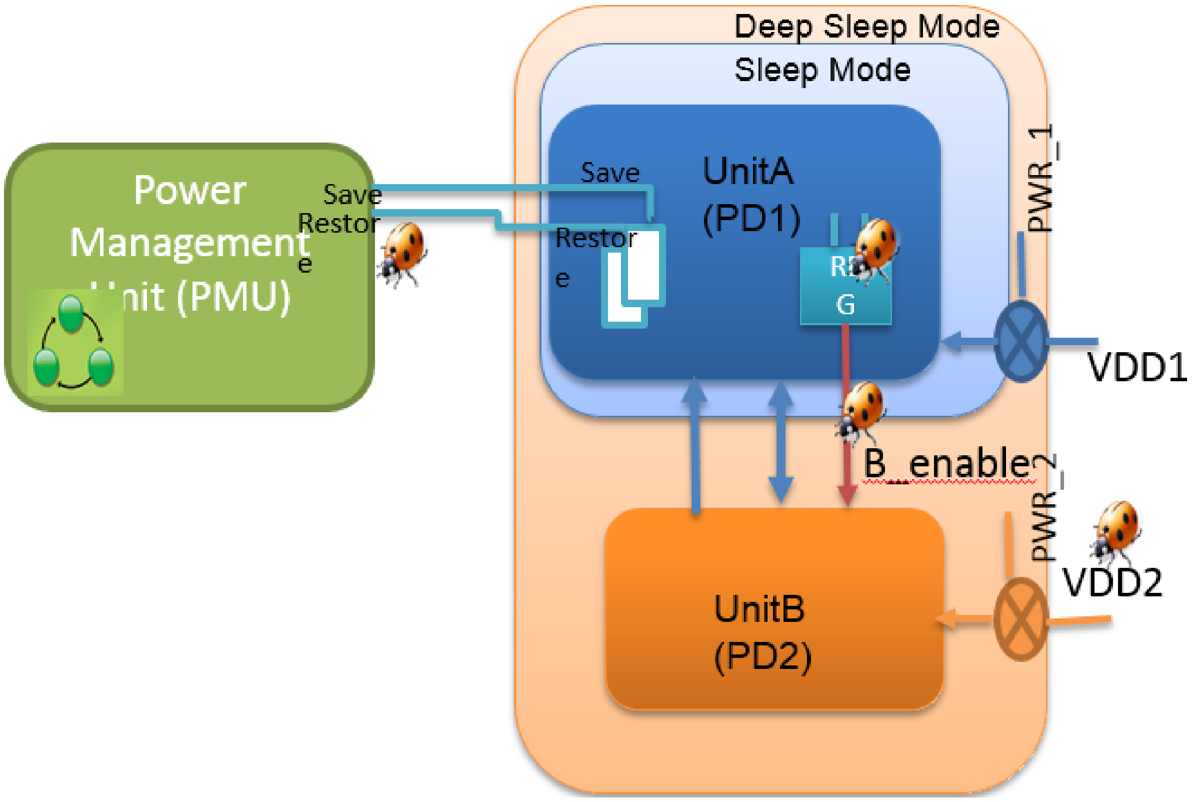


FIG-3 DUT representing the source of power related bugs

A. Case-1 Missing isolation cells

Let’s Say design is in Sleep mode (Unit A is Power OFF) and PD2 is in Power ON. In the absence of isolation strategy between PD1 and PD2 “X” or “Unknown” values will propagated from the output ports of Unit A to Unit B

B. Case-2 Incorrect Specification of Save Signal in UPF

States of PD1 did not get restored instead the signals shows “X” When the PD1 is powered up after the sleep mode is due to incorrect Save & Re-store

Typically this kind of bug can be root caused by UPF modification: UPF developer who defined LOW assuming that the SAVE should happen when there is a transition from HIGH to LOW while the correct interpretation should have been that when there is an active HIGH pulse on SAVE the values should be preserved. By replacing LOW by HIGH in UPF the problem was resolved

```
set_retention        retention_latch -domain PD1 -retention_ground_net vss -retention_power_net  vdd
set_retention_control retention_latch -domain PD1 -save_signal {isuspend_ret low} -restore_signal {isuspend_ret
high}
```

C. Case-3 Missing Retention cell between Power Domain

B_enable input is driven from the Unit A. During Normal mode of operation, the UnitA would drive this signal based on the value programmed in the register. During Sleep mode, the PD1 can be off while the PD2 continues to be on. an isolation cell is declared at the output of unitA that clamps the value of B_enable to HIGH.

During NORMAL operation, the value of B_enable is HIGH. Once isolated, the value continues to be HIGH while the PD1 is powered off. When the UnitB is powered on, the isolation is removed and after the clock is enabled, the default value of the register is driven. Since the default value is ZERO, the UnitB block actually gets functionally disabled when the UnitA is fully functional. After adding a state retention cell to the UPF for this register, desired functionality is achieved.

D. Case-4 Due to incorrect save & restore power sequence

Some time there is 'X' in a waveform could be a bug due to incorrect save and restore sequence triggered by Power management unit.

E. Case-5 Incorrect connections of Power supply to power domains

there are 2 voltage supplies each connected to the baseband UnitA and RF UnitB blocks. These supplies aren't present in RTL and specified in the UPF. Corresponding to these supplies there are power switches with the enable to the switches coming from power manager based on the Sleep or Deep sleep mode. In this case, if the UPF definition had an incorrect connection to the power switch for UnitB. As a result, during power aware simulations, in Sleep mode, instead of turning off the UnitA alone, the UnitB will also turned off i.e. instead of Sleep mode, the chip actually will go to Deep sleep mode

IV. Debug solution: Features requisite for LP debug capabilities

Power–related error during verification can be due to *Error* found in PA designs ***"caused by RTL"*** *or* the behavior defined in the ***"power specification code (UPF)"*** hence it is critical need for debug tools to understand the power specifications and ease the verification effort by providing additional capabilities specified in FIG 4.

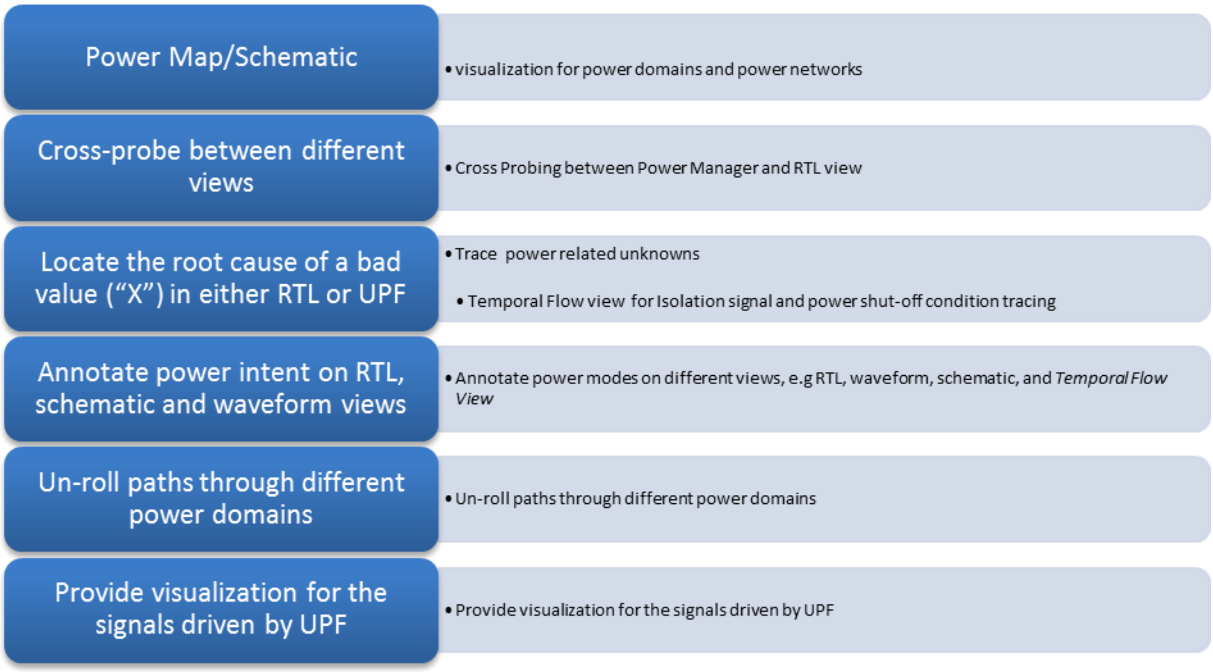

Power Map/Schematic
• visualization for power domains and power networks
Cross-probe between different views
• Cross Probing between Power Manager and RTL view
Locate the root cause of a bad value ("X") in either RTL or UPF
• Trace power related unknowns
• Temporal Flow view for Isolation signal and power shut-off condition tracing
Annotate power intent on RTL, schematic and waveform views
• Annotate power modes on different views, e.g RTL, waveform, schematic, and *Temporal Flow View*
Un-roll paths through different power domains
• Un-roll paths through different power domains
Provide visualization for the signals driven by UPF
• Provide visualization for the signals driven by UPF

FIG4: Power aware Debug capabilities

Using these debug features accelerate the PA verification and hence improve the overall efficiency of verification environment. This features helps validator to identify:

- Schematic representation of power intents
- The power aware signal types (Retention signals, Isolation signals or level shifter signals) in waveform
- Trace power related unknown or "X" and help validator to know specific signals belongs to which Power domain

V. Sign-off Criteria for PA Verifcation

Just as functional coverage is an important part of functional testing, coverage of low power objects is equally an important part of Power aware testing but as the number of power-domains increases and chip complexity grows, scope and coverage of power aware (PA) verification becomes complex but a critical necessity as we want to find the critical bugs at early stages in design cycle rather than wait for silicon results. Low power coverage methodology helped us to identify the holes/un-covered test – sequences during power aware verification without compromising on runtime /memory overhead during RTL simulation. We have enabled the low-power coverage driven verification by analysing the coverage on low power objects to modify the test bench and addition of the new test sequences by isolating the uncovered low power scenarios.

In a Power aware verification what is "not verified" by static and dynamic checks can be validated ONLY by Low power coverage.

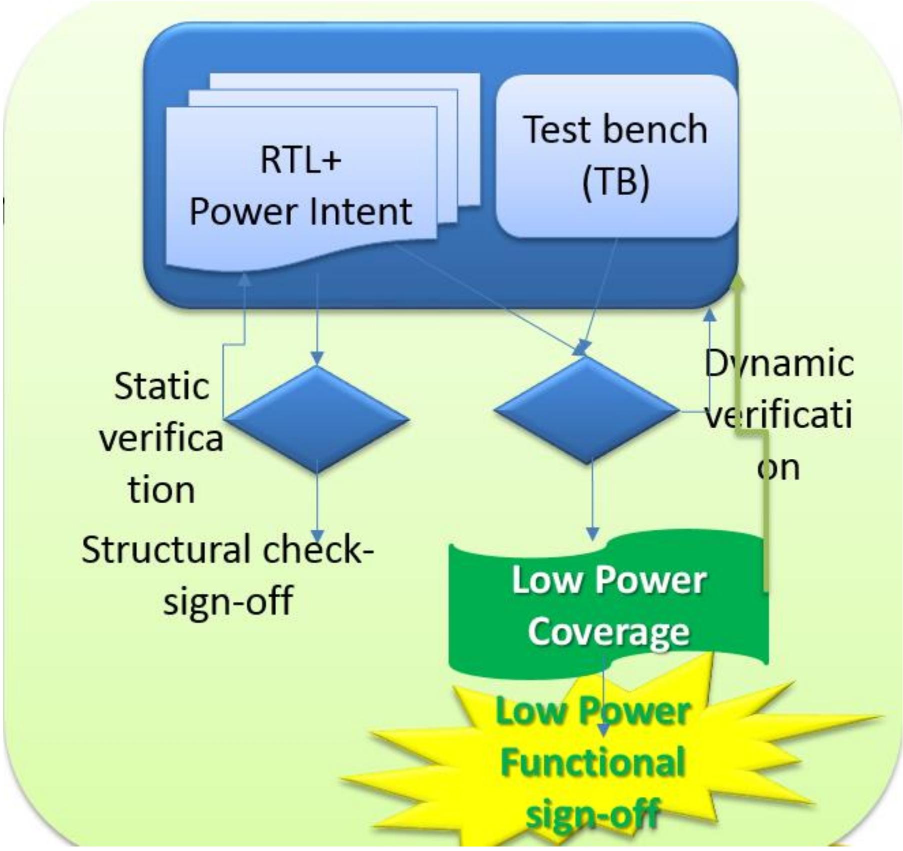


FIG5: LP coverage intercept during PA verification

Other methods like static and dynamic checks are not sufficient enough hence it is extremely important to plan low power specific coverage in test plan from start and track till verification closure.
Key benefits of low power coverage:

- It provides dynamic approach to validate low power implementation
- Identify Uncovered gaps in the low power verification test plan
- Easy bucketization of the low power checks for improving coverage
- As part of regression, the verification engineers gets to know both his functional and low power coverage with zero impact

Hence we recommend low power coverage is one of the major sign-off criteria for low power functional validation.

VI. COVERAGE ANALYSIS FOR LOW POWER OBJECTS

Low power coverage identifies the coverage of low power objects for sign-off confidence and Assist validation engineers to identify the holes/un-covered test – sequences during power aware verification

Low power coverage approach adopted by us:

- Ensure all states in Power state table (PST) has been exercised

* Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors).*

- Ensure all transitions in Power state transition table has been occurred as per power spec (HAS)
- Ensure power switch switched between ON and OFF states
- Ensure Control signals toggle
- Full power-up /power down sequence
- Do we see illegal states/transitions?

A. Detailed coverage analysis on Power State Table (PST)

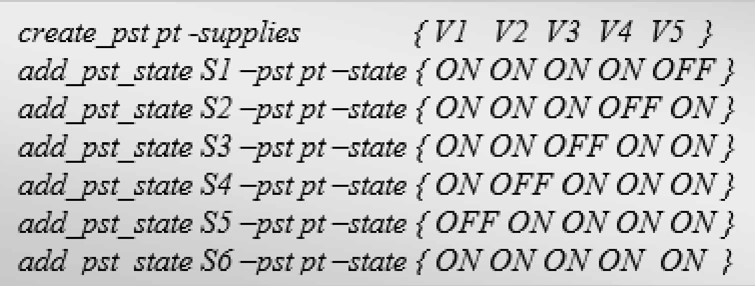
The use of the Power State Table (PST)coverage helps identifying which power states have not been verified yet and hence identify which test sequence has to be developed to exercise the unverified power states/transitions.

Power State Table (PST)coverage consist of states and transition coverage , By default simulator assumes that all state transitions between legal and user-defined states are valid transitions and the rest of them are illegal states and transitions.

Example:

Consider a design having Five supply nets , let's say "V1" ,"V2", " V3", "V4" and "V5"

Hence there can be 2^5 states in Power State Table (PST). However if power architect has specified only six states S1 S2 S3 S4 S5 S6 in UPF. So simulator will consider remaining other 26 states (let's say S7 to S32) as illegal states by default.



```
create_pst pt -supplies           { V1  V2  V3  V4  V5  }
add_pst_state S1 –pst pt –state { ON ON ON ON OFF }
add_pst_state S2 –pst pt –state { ON ON ON OFF ON }
add_pst_state S3 –pst pt –state { ON ON OFF ON ON }
add_pst_state S4 –pst pt –state { ON OFF ON ON ON }
add_pst_state S5 –pst pt –state { OFF ON ON ON ON }
add  pst  state S6 –pst pt –state { ON ON ON ON  ON  }
```

FIG-6 Legal Power State Table (PST) entries in UPF

Legal states: S1, S2, S3, S4, S5 and S6
Legal Transitions: Any transitions between S1 to S6
Illegal states: S7-S32
Illegal transitions: Any transition between Legal states to illegal states
S1->(S7-S32), S2->(S7-S32), S3->(S7-S32), S4->(S7-S32), S5->(S7-S32),S6->(S7-S32),

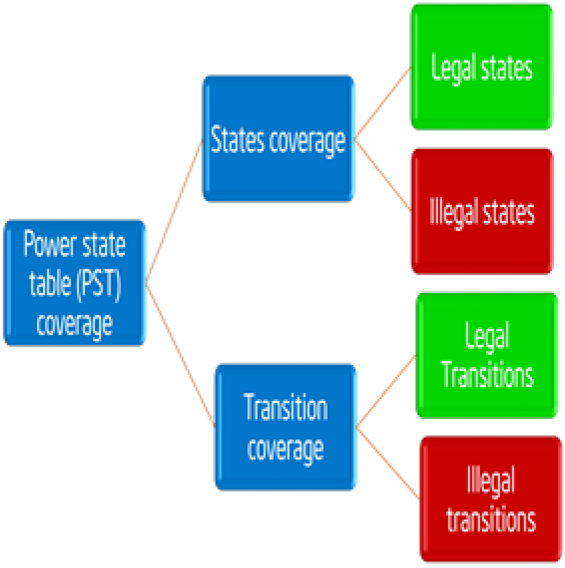
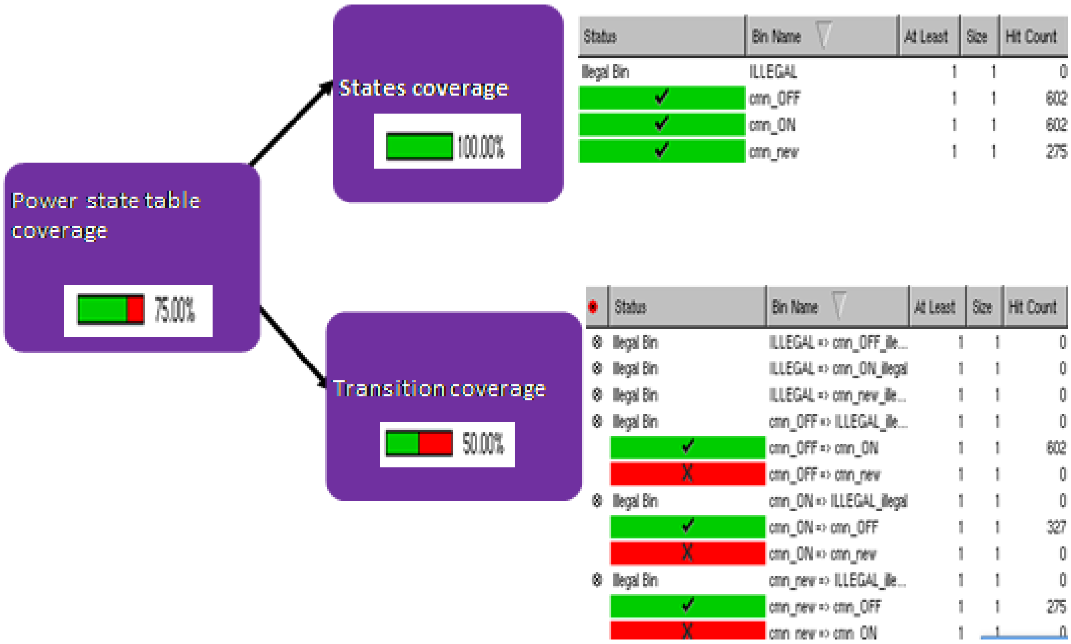


FIG -7 Power State Table (PST) Coverage

During the Low Power coverage analysis PA simulator generates the default cover Bins for each legal /Illegal states and transitions hence the overall Power state Table (PST) coverage is the accumulative coverage of state and transition coverage.



Status	Bin Name	At Least	Size	Hit Count
Illegal Bin	ILLEGAL	1	1	0
✓	cmn_OFF	1	1	602
✓	cmn_ON	1	1	602
✓	cmn_new	1	1	275

●	Status	Bin Name	At Least	Size	Hit Count
⊗	Illegal Bin	ILLEGAL => cmn_OFF_ille...	1	1	0
⊗	Illegal Bin	ILLEGAL => cmn_ON_illegal	1	1	0
⊗	Illegal Bin	ILLEGAL => cmn_new_ille...	1	1	0
⊗	Illegal Bin	cmn_OFF => ILLEGAL_ille...	1	1	0
	✓	cmn_OFF => cmn_ON	1	1	602
	X	cmn_OFF => cmn_new	1	1	0
⊗	Illegal Bin	cmn_ON => ILLEGAL_illegal	1	1	0
	✓	cmn_ON => cmn_OFF	1	1	327
	X	cmn_ON => cmn_new	1	1	0
⊗	Illegal Bin	cmn_new => ILLEGAL_ille...	1	1	0
	✓	cmn_new => cmn_OFF	1	1	275
	X	cmn_new => cmn_ON	1	1	0

FIG-8 Power State Table (PST) Coverage analysis

At the end of the Power State Table (PST) coverage analysis, Power aware simulator pin point the un-verified states /transitions , that helped validation team to add/modify new test sequences to improve overall functional coverage during Low Power verification.

B.   Detailed Coverage Analysis on Protection policies

Just like Power state Table, coverage analysis on protectin polices are extremely important to analyze the overall all low power coverage and to ensure all the Retention and Isolation policies are exercised during the power aware verification.
Simulator generates the automated cover groups for isolation and Retention control signals and create bin for each states (Active, Low) and transitions (Active-> Low and Low->Active).

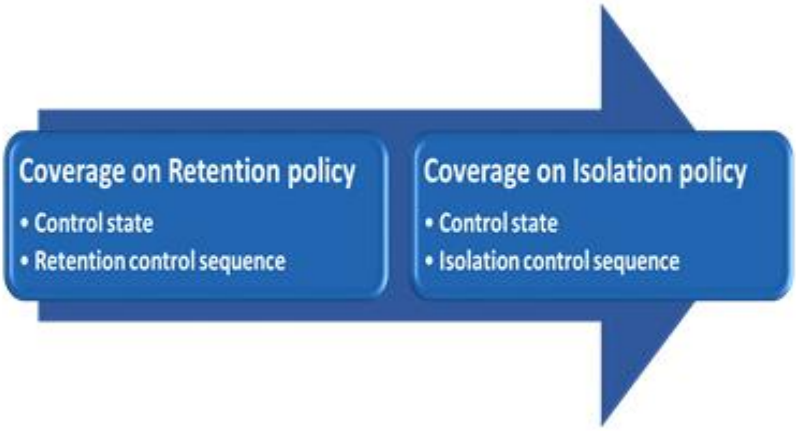
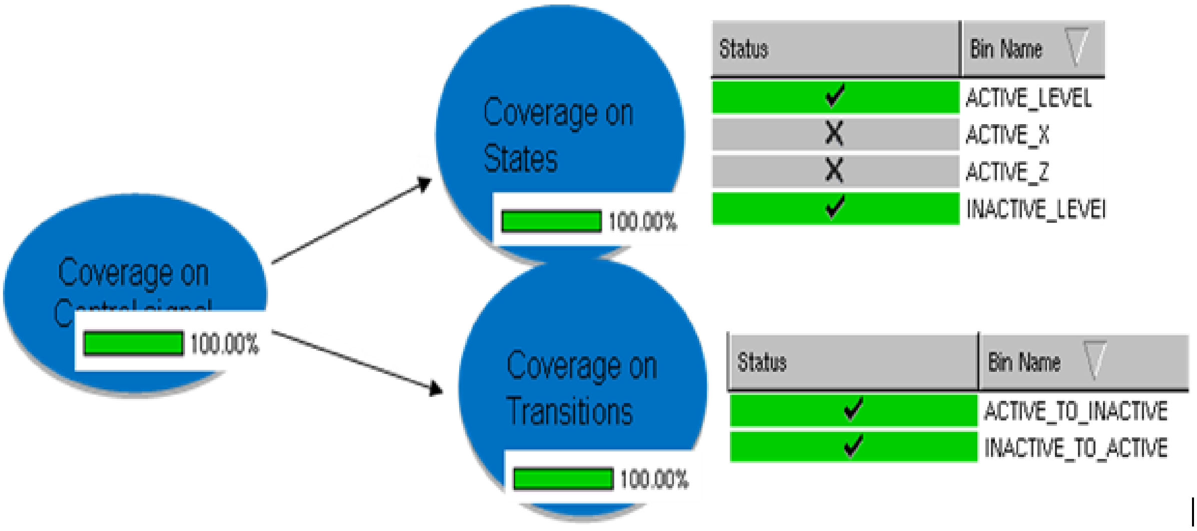

Coverage on Retention policy
• Control state
• Retention control sequence
Coverage on Isolation policy
• Control state
• Isolation control sequence

FIG-9 Coverage on Protection policies

Overall coverage on Isoilation and Retention polices are measured based on bins hit during control signal’s state (Active, Low) and transition (Active ->Low and Low->Active).



Status	Bin Name
✓	ACTIVE_LEVEL
X	ACTIVE_X
X	ACTIVE_Z
✓	INACTIVE_LEVEI

Status	Bin Name
✓	ACTIVE_TO_INACTIVE
✓	INACTIVE_TO_ACTIVE

FIG-10 Retention/Isolation coverage analysis

At the end of the Retention/Isolation coverage analysis, coverage analysis pin point the un-exercised control signals state/transition , that helped us to identify/add/modify new test sequences to improve overall functional coverage during Low Power verification.

C. Coverage analysis on Power Switches:

Like Power State Tables (PST) and Protection polices, Coverage analysis on Power Switches is an essential part of overall all low power coverage in Power aware validation. It help us to identify if we are switching to all the valid states (ON and OFF) and if testplan is covering all the valid transitions (ON-> OFF and OFF-> ON) of Power switch.

VII. RESULTS AND KEY TAKEAWAYS

This paper exhibited a way to speed up the low power validation using power aware simulator and facilitate/recommend the power-aware debug methodology using power-aware debugger tool to find various critical errors in design which resulted in more than 3X improvement in power-aware simulation debug compared to traditional debug approach.

Key takeaways:

- LP verification must merge with functional verification methodologies for assertions, testbench and coverage
- Static and dynamic checking of RTL & Power-intent for complete power aware verification
- Need for Low power debugger capabilities to reduce the validation effort.
- Low power coverage should be a part of overall verification test plan.
- Run the LP tests along with functional tests -It confirm the functionality of the design doesn't get impacted by enabling power-saving features.
- LP tests should be checked with coverage numbers in addition to traditional pass/fail mechanism.
- As you start running the power regressions, enable automatic low power coverage.
- It is extremely important to plan low power specific coverage from start and track it till design closure

This Paper recommend users to use the Low Power coverage as a part of "complete" low power verification process, as Low power coverage is a crucial requirement in addition to static & dynamic checks for low power functional sign-off because what is "not verified" by static and dynamic checks can be validated by Low power coverage.

REFERENCES

[1] VCSMX user guide from Synopsys.

[2] DAC paper "Verification Challenges in a Power Hungry SmartWorld " by Deepmala Sachan & Bharathi, V

[3] White Paper on "Challaneges and requiremnets for power aware debug" from springsoft