2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION EUROPE

Complex Low Power Verification Challenges in NextGen SoCs: Taming the Beast!

Abhinav Nawal, Freescale Semiconductors India Pvt Ltd, abhinav.nawal@freescale.com

Gaurav Jain, Freescale Semiconductors India Pvt Ltd, gaurav.jain@freescale.com

***Abstract* — Next Generation Low Power SoCs have become increasingly complex due to multiple low power techniques such as fine grained clock gating , power shutoff involving multiple low power domains, body or well biasing , dynamic frequency and voltage scaling, etc being deployed to meet aggressive low power targets in each device mode. The complex Power Management Controller and power switch fabric integration, Multi-Core Design , Low Power Subsystem with Analog and Digital low power peripherals, Memory types and Retention schemes, multiple clock and reset sources etc, lead to significant increase in the verification scope and debug challenges for functional verification at RTL level and timing implications for Gate Level simulations. This paper showcases a holistic Low Power Verification Methodology employed for verifying the power intent in a next generation SoC, using advanced verification techniques. The paper discusses the generic low power design areas which can be identified as sweet spots for Low Power Assertions and Cover-groups. This paper also discusses the hazards of X-Optimism in context of Low Power Simulations and a new methodology to catch these issues early at RTL level. The paper presents a case study on variety of low power design issues caught using advanced low power verification techniques during various stages of SoC development life cycle.**

***Keywords—CPF*, Low Power Verification, CPF Enabled Gate Level Simulations, Assertions, Coverage, X-Optimism, X-Prop, SoC, Macro Models, vPlan, Power domain, Power Shutoff, Power Mode**

I. INTRODUCTION

Low power design complexity presents numerous new verification challenges. The low power verification scenarios have increased tremendously due to multiple power domains and power modes coupled with multiple clock options and clock ratios, reset domains and reset types and wakeup sources. Low Power SoC Designs also have high performance domains having multi-core processors along with various boot options to choose from upon exit from low power modes.

Low Power Automotive SoC designs have significant analog content like on-chip voltage regulators, power switches and GPIOs with multiple supplies, precision ADCs, DACs, low power flash etc., and provide significant challenge to verify analog-digital low power integration and low power operation of analog macro models. The low power peripherals' seamless operation across power modes and various memory retention modes needs significant verification effort. Low Power design for Test and testability, Reset controllers, logic BIST controllers in each power domain further increase the scope and debug challenges for functional verification at RTL level.

Performance verification needs to be performed to measure mode entry and exit times for various configurations and low power mode exit schemes needs to be verified. Mode entry aborts scenarios due to early wakeup or due to various reset events need to be thoroughly verified to check all possible abort windows such that there are no dead-end states in multiple FSMs which work in tandem for power mode control and generate the critical low power control signals. The debug infrastructure too needs to be verified for each core and also in-case there is a core in a low power subsystem.

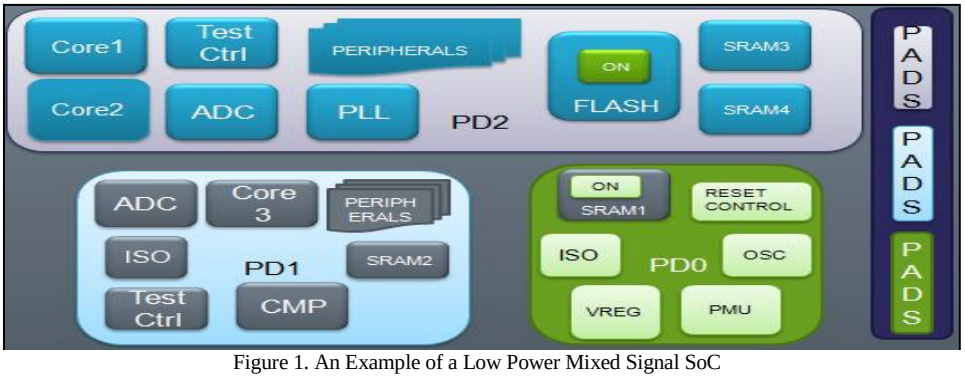

Figure 1. An Example of a Low Power Mixed Signal SoC

An idea about total verification scenarios can be gauged from below illustrations of the combinations possible for verification .With increase in low power design components, the scenarios increase exponentially.

Low Power Design Features	Abbreviations
Number of power domains	PDn
Number of Power Modes	PMn
Number of clock sources (IRC/OSC/PLL)/clock ratios(core, platform, bus, flash)	CSn
Number of reset domains and reset types	RDn
Number of Wakeup sources (GPIO, NMI, Resets, Interrupts etc)	WSn
Number of Retention Memories	RMn
Number of Analog-Digital Interfaces and Macro Models	ADn
Number of Low Power Peripherals	LPn
Number of Abort Scenarios(early wakeup, resets)	ASn

Table 1. Low Power Verification Scenarios

Total Verification Scenarios = PDn X PMn X CSn X RDn X WSn X RMn X ADn X LPn X ASn..

In order to mitigate the above low power verification challenges, it is imperative to have a comprehensive and multi-faceted low power verification methodology based on latest verification techniques and flows.

II. LOW POWER VERIFICATION METHODOLOGY

The significant verification challenge presented by introduction of complex low power design techniques in SoCs needs a comprehensive low power verification methodology to ensure robust low power verification with coverage closure. The Low Power Verification Methodology components are discussed below;

- CPF enabled power aware simulations leveraging multi-core testbench setup and C-based stimulus with message portal to SystemVerilog testbench for randomizing various clock configurations, mode transitions, wakeup sources, etc.
- Low Power Assertions (user as well as tool generated) and SystemVerilog Covergroups defined for verification and functional coverage generation for low power protocol and integration checks. The additional ROI from various types of assertions is reduced debug time and coverage with re-usability across NPIs and low power architectures.
- CPF simulations with X-Propagation enabled to catch x-optimism issues early at RTL Level.
- CPF Enabled Gate Level Simulations to validate the integrity of key low power signals and protocols and timing implications in worst case timing for final sanity and verification signoff along with coverage results from RTL level verification.

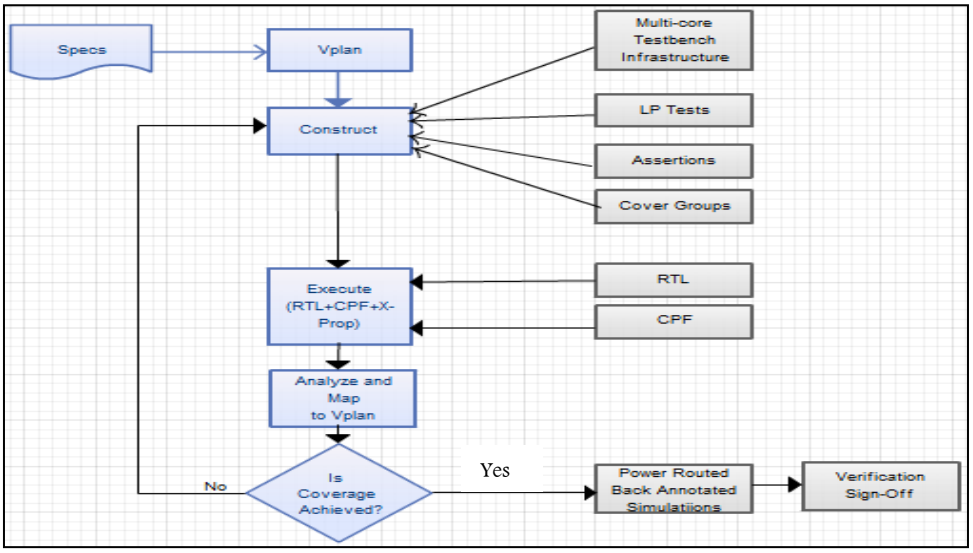

Figure 2. Low Power Verification Methodology Flow

These topics are covered in detail with code illustrations where ever applicable to highlight rationale and re-usability of the best practices.

A. Comprehensive Low Power Verification Plan Creation & Multi-core Test bench Infrastructure Setup

The Low Power Verification starts with creation of a comprehensive vPlan which indentifies the key low power design features for verification and maps them with corresponding directed or controlled random tests, assertions and coverage items to be implemented for covering the same. The verification results are mapped back to the executable Low Power vPlan with specification tags to ensure robust verification via coverage closure.

The next step is to create the testbench infrastructure which provides features to test various low power modes and system level scenarios. We created an embedded SystemVerilog and C based testbench infrastructure with support for multi-processor tests running in parallel and randomized selection of booting core with application code residing in system RAM or Flash for execution after low power mode exit. The testbench uses C based stimulus for device configuration and a message portal to SystemVerilog classes using mailboxes to transfer configuration data like clock configurations, clock ratios, mode transitions etc, randomized in SystemVerilog side and for test flow synchronization. Generic Power APIs are developed for low power modes entry and exit to be reused in all low power tests covering all possible mode transitions, wake up sources, reset sources, low power peripheral functionality, clock configurations, debug, multiple power modes' entry/exit and abort scenarios. This kind of setup allowed us to take advantages of a structured UVM based testbench and verification features like Assertions and Coverage while allowing us to simulate real software based device scenarios.

B. CPF enabled Low Power Verification

The Common Power Format (CPF) is used to define the complete information related to SoC power intent like power domains with shut off conditions where applicable , macro models power domain integration with top level domains , legal power modes and power mode transitions with entry and exit conditions , isolation and state retention rules as applicable and top level power supply signals. The CPF enabled simulations are run with all low power test cases at RTL stage to cover all power modes, mode transitions with various wakeup sources, clocking and reset scenarios. CPF simulations are run with low power assertions and cover-groups to check low power operation in each mode and generate coverage information to map back to the executable vPlan. The assertions belonging to shutoff domains are suspended by the simulator during power gating of the domain via CPF.

C. Low Power Assertions

The CPF enabled low power simulations can be cycle time and debug effort intensive due to extra effort in debugging of X-propagation in the design. It also depends on test code and manual checking of waves to uncover issues in the design which can lead to verification holes. Assertions targeted towards low power design features augment the power aware verification by considerably reducing the cycle time and debug effort as they highlight the type and source of the issue. The assertions based low power protocol and integration checkers embedded in the design also provide crucial low power coverage information along with system level cross coverage on key signals, to measure verification completeness. These low power assertions can also be re-used by other projects by creating template properties.

We identified some sweet spots in low power design architecture as key areas for deployment of assertions. Following assertions categories were deployed to target verification of low power design features, clocking and reset schemes, sequencing and connectivity of critical control signals for low power mode entry and exit.

- Reset Mapping Assertions

One of the most common issues we found was related incorrect re-initialization of shutoff domain instance due to wrong domain specific reset connected to the instance. The reset inputs of each IP in shutoff domains were identified and assertions were added to check connectivity to correct domain specific reset.

- Low Power Mode Entry/Exit Protocol Assertions

The low power mode entry and exit protocol involves complex set of signals like power mode entry condition, isolation enable, power switch open signal, power switch status signal, voltage regulator turnoff and status signals with multiple set of conditions as per each low power mode. An extensive protocol checker which checked signal connectivity and protocol sequence between power controller and power switch fabric was added.

- Macro Model Low Power Protocol Assertions

Assertions to check glitches on critical signals like power-down and status signals of clock sources like crystal oscillators, flash, memories were added

- Isolation –Reset sequencing checks

For latch based isolations to latch correct values, reset to the switchable domain should be asserted after assertion of isolation enable signal

- Low Power Peripheral Clock Gating Assertions

Fine Grained clock gating in low power mode can be good target for assertions. We added assertions to check IP specific clock gating happens as per system clock configurations in low power modes wherever IP is not operational.

- Low Power Peripheral Operation Assertions

Powered ON Low Power Peripherals can do seamless operation with no reset applied across mode transitions where other domains are being reset.

The below table illustrates some of the property templates and assertion examples for Low Power Verification. We recommend extensive use of assertions to verify low power scenarios as they generally include temporal behavior of control signals, one hot conditions and relationship based values between design signals , which can give very good ROI in terms of finding corner case bugs and coverage.

```
//Reset Mapping Check
property prop_ip_reset_sig_chk (ip_reset_sig , domain_reset_sig ,power_domain);
@(posedge simulation_clk)   !ip_reset_sig  |-> !domain_reset_sig  ;
endproperty

//VREG and PMU Low Power Protocol Checker
standby_to_Run_when_sys_wakeup : assert property (@(posedge simulation_clk)
$rose(sys_wakeup) & curr_mode==`STBY  |-> ##[0:100] (sw1_done & sw2_done ) ##[0:100] !iso1
& !iso2 ##[1:10] curr_mode==`RUN);

//Macro Model Low Power Protocol checker
sleep_asserted_before_rd_sw_open : assert property (@(posedge simulation_clk)
$fell(`sw_rd.sw_open) |-> $past(`sram.sleep,2) &&  $past(`sram.sleep,1) && `sram.sleep);

//Isolation Reset Sequence check
iso_asserted_before_reset : assert property (@(posedge simulation_clk) `pmu.target_mode==`STBY
&& !`rst_ctrl.pd2_dest_reset   |-> $past(`pmc.pd2_iso ,2) &&  $past(`pmc.pd2_iso,1) &&
`pmc.pd2_iso );

// Low Power Peripheral Clock Gating Check
Lp_mode_periph_clk_gated_when_module_disabled : assert property (@(posedge simulation_clk)
`pmu.target_mode==`LP_STOP && `pmu.periph_stop_reg[0] |-> ##[40:100] `LP_Periph_num0.clk
== 1'b0 ##1 $stable(`LP_Periph_num0.clk) );
```

Figure 3. Low Power Assertions Examples

*D. **Low Power Coverage***

The SystemVerilog Covergroups and cross coverpoints provide a good way to judge if low power mode design scenarios are covered by the test suite. The key design signals are mirrored in the testbench via "Out of Module References" (OOMRs) and then covergroups and cross coverpoints are added to these signals to check coverage for various device use-cases and corner case scenarios. The simulator also provided CPF derived automatic coverage for control signals like shutoff conditions, isolation enables, power mode transitions starting conditions. An illustration of the user written system level low power scenario coverage is shown below.

The lowest power mode is described as STANDBY mode in where only keep alive logic is powered and after mode entry is complete, the design is configured to turn off all clocks. In this scenario it is important for device to exit the STANDBY mode by an asynchronous external reset or an asynchronous external wakeup event. A covergroup is created to target this scenario.

```
// Standby Mode Exit via External Reset toggle with no System clock present
bit ext_reset_b , irc_clk_en;
bit[3:0] current_mode ;

covergroup  STBY_MODE_EXT_RST_IRC_OFF @(posedge simulation_clk);

EXT_RST : coverpoint ext_reset_b {
bins ext_reset_b_low = (1=>0);
bins ext_reset_b_high= (0=>1) ;
}

IRC_EN : coverpoint irc_clk_en {
bins irc_clk_en_high = {1};
bins irc_clk_en_low = {0};
}

CURR_MODE : coverpoint current_mode {
bins STBY = {4'hD};
bins RUN = {4'h3} ;
bins STOP = {4'hA};
}

stby_rst_irc_en : cross EXT_RST , IRC_EN , CURR_MODE {
bins stby_ext_rst_exit_irc_off = binsof(CURR_MODE.STBY) &&
binsof(IRC_EN.irc_clk_en_low) && binsof(EXT_RST.ext_reset_b_low);
}

endgroup

STBY_MODE_EXT_RST_IRC_OFF stby_mode_ext_rst_irc_off_inst = new;
```

Figure 4. Cover groups

E. Uncovering Power Up issues hidden by X-Optimism in RTL

The Low Power simulations rely on corrupting the shutoff domain by driving unknown values and X propagation in the powered domain to un-cover the low power design issues. RTL coding styles having inherent X-optimism, which does not propagate the X values, can mask low power issues in low power simulations. We have observed that power-up protocol issues like incorrect domain reset connectivity or restore event generation issues have been masked due to X-Optimism in the RTL code. These issues are either caught very late during CPF enabled gate level simulations due to X propagation in logic gates or sometimes missed if not present in data path being checked by the test, leading to Post-Silicon issues. Since elaborate checkers may not be present on each signal crossing, these issues are almost impossible to catch at the RTL stage. Here comes the need of some enhanced methodology that caters to these issues. An example of this kind of issue is shown below.

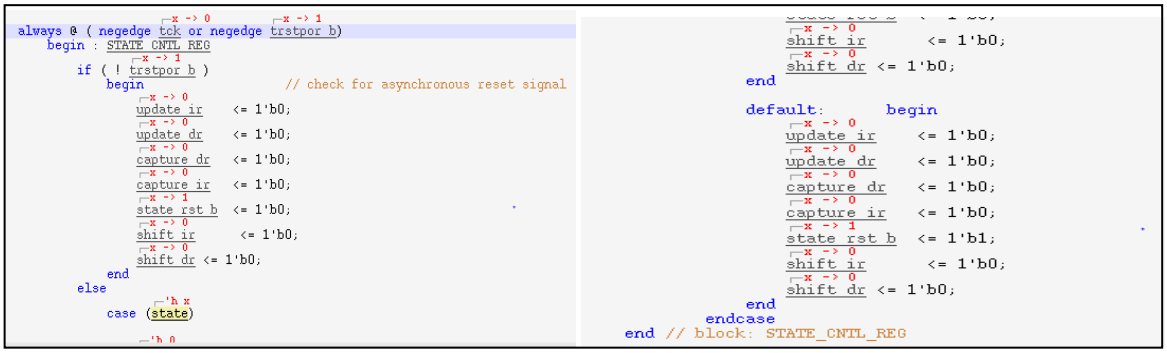

Figure 5. X-Optimism code example

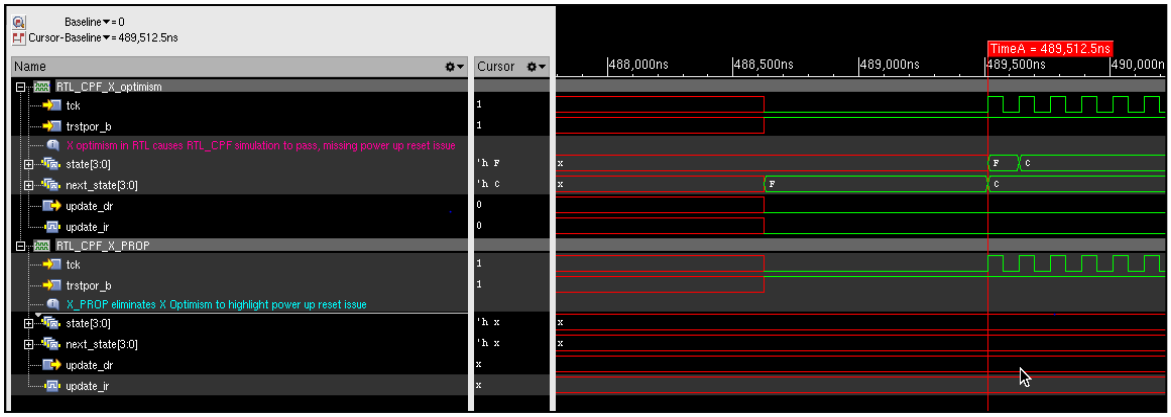

Figure 6. X-Optimism v/s Low Power X-Prop Waves

We created an Assertion based solution to detect uninitialized flops in the switchable domain after power up. This solution relies on getting list of flops in shutoff domain from simulator or synthesis engine. The Reset Mapping assertions were also added to check that correct the domain reset is connected to each shut-down instance re-initialize the outputs.

```
property reg_not_x_post_power_up(shutoff_condition,signal);
   @(posedge sim_clk) $fell(shutoff_condition)|=> !$isunknown(signal);
 endproperty
assert property (reg_not_x_post_power_up (`PCU.pwr_down , `PD2_inst.state));
```

Figure 7. X-Optimism Assertion Checker Example

A recent update in simulator's X propagation feature enabled with CPF where-in X propagation analysis in a domain is suspended when the domain is shutoff and enabled automatically after domain power-up was also successfully checked to catch incorrect initialization issues.

F. Power aware Gate Level Simulations

The final placement and routed netlist can have low power issues like incorrect placement of switchable buffers on always-on nets and un-buffered or don't touch analog nets due to power domain merging and placement of buffers to meet timing in all corners. Further it is important to check that design is able to enter and exit key low power modes with correct low power entry exit protocol sequence under worst case timing corner. Power aware Gate Level Simulations are run to validate the integrity of key low power control signals and protocols.

III. DESIGN ISSUES CAUGHT USING LOW POWER VERIFICATION METHODOLOGY

We found several low power design issues related to connectivity, protocol sequence and functional behavior of low power control signals, clocking and reset scheme issues, which we have categorized below

A. Wrong Isolation Type Bugs

We found that design had used latch based isolations in a power domain which was alive in one mode and switchable in another mode. When the mode transition was done from latter mode to the first one, these latch based isolation remained uninitialized causing device operation to fail.

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

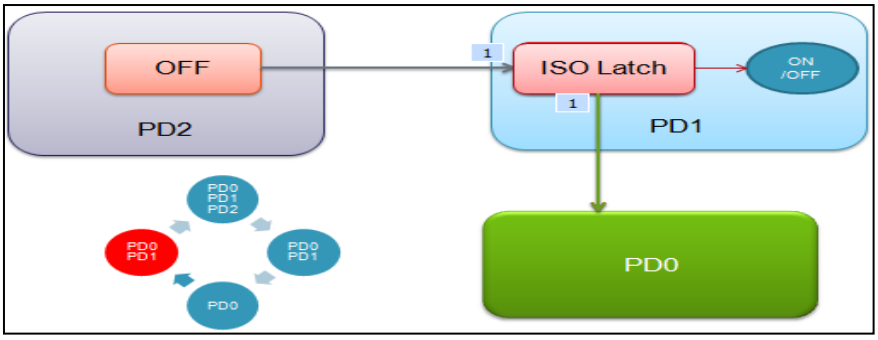

Figure 8.Wrong Isolation Type

B. *Low Power Protocol Sequence Bugs*

We found that there was a bug in the integration of power switch matrix with power mode controller where the correct sequence of removing isolation after closing of power switches of both PD1 and PD2 domains was not being followed. The power mode controller was de-asserting the isolation without taking into account the switch close signal of PD1 domain resulting in system hang scenario at the low power mode exit as seen in figure 8.

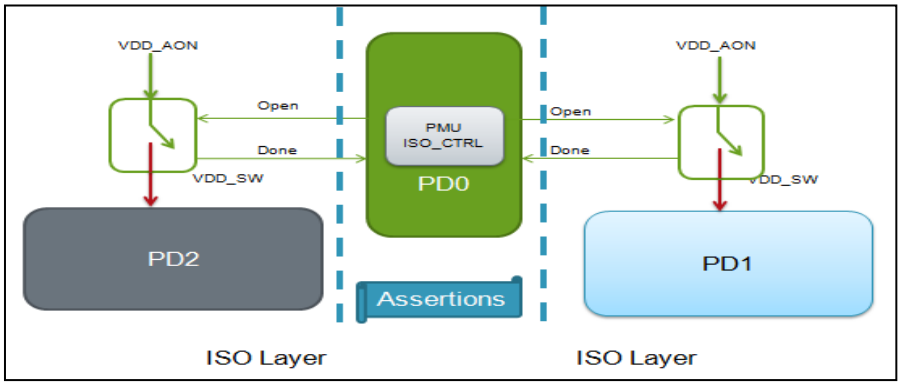

Figure 9.Low Power Protocol Issue

Another issue that was caught was of unexpected toggle on power shutoff signal of a power domain as seen in figure 9. PMC Low Power Protocol assertions were key to uncovering many such issues.

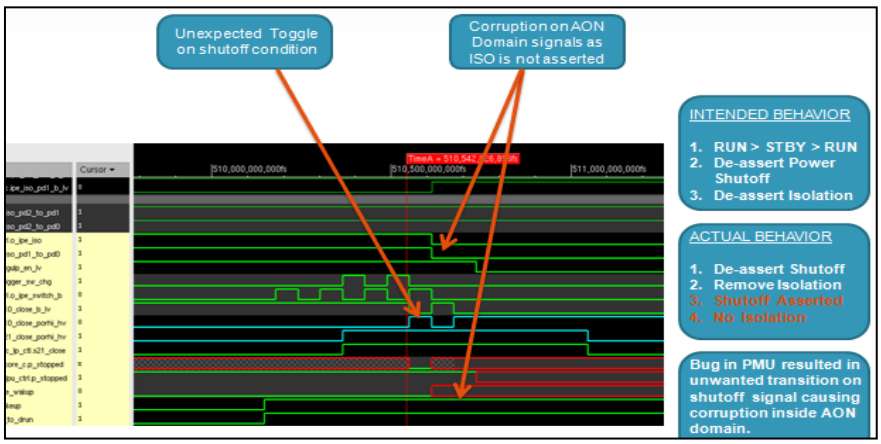

Figure 10. Low Power Protocol Issue

C. *Reset Power Domain Connectivity Bugs*

Incorrect domain reset connections at the input of instances which are powered off result in re-initialization issues after domain is powered up. These causes unknown values to cross to into alive or powered domain when isolation is disabled leading to a system hang scenario.

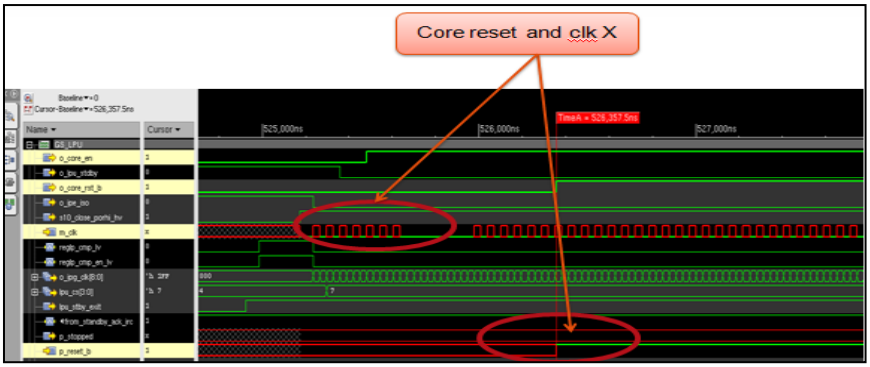

Figure 11. Reset Mapping Connectivity Bug

D. Macro Model Low Power Protocol Bugs

Memory Retention protocol required that memory input signal "SLEEP" was asserted before periphery power supply was removed. The protocol got violated where in SLEEP was asserted after opening the power switch leading to corruption of retained SRAM. Assertions were added to act as checkers for all such hard macros.

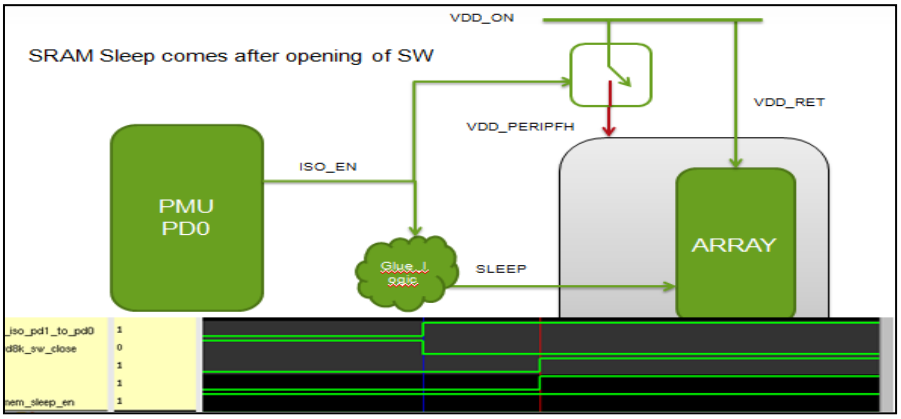

Figure 12. Memory Sleep Late Assertion Bug

E. Hard Macro Low Power Design Issues

A unique issue was discovered in the design of the analog power switch where the switch status pin "done" was driven by the switch output supply which would be low when switch opens. This lead to the switch "done" signal to be in unknown state, which caused corruption in the Power Management unit when low power mode was entered. This issue was discovered due to macro model CPF, where as per the circuit, the close signal was mapped to switched supply, instead of always on supply.

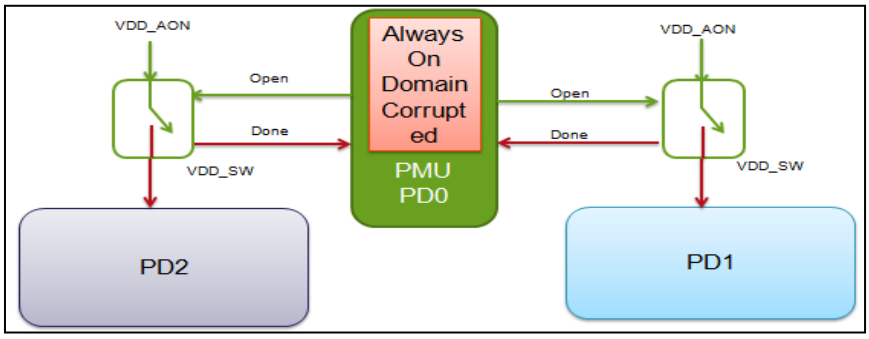

Figure 13. Hard Macro Low Power Design Bug

F. Low Power Issues caught in CPF enabled Gate Level Simulations

CPF enabled Gate Level Simulations to check the mode transitions at worst case timing corner helped in catching a memory retention issue related to timing sequence between isolation enable and proper input value of sleep pin of memory.

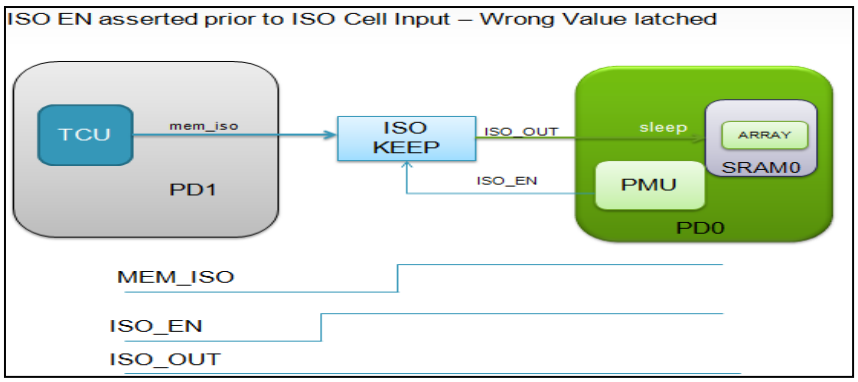

Figure 14. Memory Retention Issue

Another issue that was caught in CPF enabled Gate Level Simulations was that switchable buffers were inserted on the analog nets by the Placement and Routing tool causing corruption in the analog behavioral models where these nets were connected.

IV. CONCLUSION

This paper has discussed a comprehensive Low Power Verification Methodology deployed for functional verification of Next Generation Low Power SoC designs at various stages of SoC development cycle. The paper highlights how extensive verification of SoC power intent can be done RTL level, using CPF enabled Low Power Simulations, Low Power Assertions and Cover-groups. The paper provides guidelines and key areas where Low Power Assertions and Covergroups can be added for generating important system level coverage to achieve verification closure as per the low power verification plan. The paper presents the hazards of X-optimism in traditional CPF enabled RTL simulations flow due to masking of low power design issues and introduces X-Prop flow along with user assertions to uncover these issues at RTL verification stage. The paper discussed the CPF enabled Gate Level Simulations at worst case timing corner to provide final sanity checks for key low power mode entry exit scenarios. The paper also illustrated real low power design issues which we were able to catch using this flow and can be referenced as potential verification target areas.

V. ACKNOWLEDGMENT

We would like to acknowledge the contributions of Yogesh Mittal and Garima Sharda from Freescale Semiconductors India, for valuable suggestions and inputs related to the work we have presented in this paper. We would also like to express our gratitude to our families and colleagues for their continuous support and inspiration.

VI. REFERENCES

[1] Michael Keating,David Flynn,Robert Aitken,Alan Gibbons,Kaijian Shi, Low Power Methodology Manual For System-On-Chip Design. Mill Valley, Springer.

[2] Neyaz Khan, Yaron Kashai , From Spec to Verification Closure: a case study of applying UVM-MS for first pass success to a complex Mixed-Signal SoC design . DVCON 2012

[3] Stuart Sutherland, I'm still in love with my X! . DVCON 2013

[4] Luke Lang , Christina Chu, New Challenges in Verification of Mixed-Signal IP and SoC Design. DVCON 2012