

Advanced Techniques to Accomplish Power Aware CDC Verification

Rohit K Sinha, Intel Corporation
Ashish Hari, Mentor- A Siemens Business
Sulabh Kumar Khare, Mentor – A Siemens Business

Abstract

The focus on low power devices has made it essential to implement static and dynamic power reduction techniques in every modern era SoC. Reduced power consumption is typically achieved by partitioning a SoC into multiple power domains and then controlling these domains by switching off power selectively, or by reducing voltage levels. The connections across power domains need to be managed, which require designers to add power elements such as Isolation cells, Retention logic, and Voltage shifter components at power domain interfaces. The Unified Power Format (UPF), which is the IEEE standard format for specifying power domains and power control logic, is used to list these power elements. Traditionally, Clock Domain Crossing (CDC) verification is achieved with RTL as input, however, the presence of power components poses new challenges for CDC verification. UPF intent becomes a crucial factor to attain power aware CDC verification closure.

In this paper, we will present the four most prominent challenges posed by low power design on CDC paths and the solutions that can help overcome them.

Following are the key challenges that can be resolved by utilizing a combination of advanced techniques provided by CDC verification tools along with the power aware CDC verification methodology:

1. Introduction of isolation logic can impact clock and reset paths. Clock and reset paths may become prone to glitches due to the presence of combinational logic.
2. Introduction of isolation logic can create new CDC paths if the isolation enable signal is driven by an asynchronous clock domain.
3. Isolation logic on a synchronized CDC path can result in combinational logic in the data path that should be guarded against glitches.
4. Retention Save & Restore logic can have asynchronous signals driving save and restore, leading to additional CDC paths.

CDC Verification tools understand UPF intent and integrate it with RTL to illustrate the complete picture at the RTL stage. When CDC verification is performed on an integrated design that captures power intent, it leads to creating new CDC paths for which synchronizers must be added. Hierarchical flows are used for CDC verification of SoC designs, similarly, UPFs are used in hierarchical flows with a separate UPF for each block and top level. Hierarchical CDC verification must handle block and top level UPFs without any additional overhead to the designers. Therefore, CDC verification tools must include the ability to analyze hierarchical flows with UPFs.

In this paper, the key challenges are described along with examples of how they impact the CDC verification. The paper also details the modeling of different power elements from UPF and how they can be handled in power aware CDC verification. Paper concludes with a case study on production designs that include UPF to illustrate the critical need and benefits of power aware CDC verification techniques.

I. INTRODUCTION

The Unified Power Format (UPF) is the standard for specifying the power control logic and its design connections. When the power logic is not instantiated in a design, the power intent is absent in the RTL and is extracted from the UPF file during synthesis. The late implementation of the power intent information into the gate-level design may delay the start of power verification until after the gate-level representation is available, and the power verification may create a critical path late in the design flow.

Most design teams are aware of possible design problems that can be introduced by the power control logic implementation. Clock and CDC errors can occur when the power elements are incorrectly inserted in the clock tree, the reset tree, or the CDC paths. These errors may result in incorrect functionality due to data loss or data corruption. In other cases, these errors may result in intermittent errors that may not be reproduced in simulation and are extremely difficult to debug in silicon. These intermittent errors may appear or disappear when the operating conditions, such as temperature, voltage, or frequency change.

Figure 1 shows a typical power aware design flow with UPF. UPF is available at RTL level and is not part of the design. UPF contains design's power architecture and additional power elements. Power elements become part of the design after synthesis is completed.

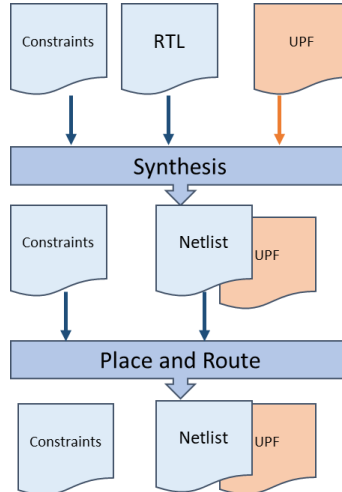


Figure 1. Power aware design flow with UPF

CDC verification involves understanding clocks, clock trees, and CDC paths in a design. CDC verification at RTL stage identifies synchronized paths and paths with missing or incorrect synchronizers. Power aware CDC verification involves understanding how the power domains and power control logic affect clocks, clock trees, and CDC paths. Power aware CDC analysis compiles UPF and adds the power elements to the RTL design. The added power elements may break already synchronized paths or create new unsynchronized CDC paths. Power aware CDC verification must identify such issues. Advanced techniques can be applied to further validate the CDC path correctness and adherence to protocols.

II. POWER AWARE CDC VERIFICATION CHALLENGES

Power aware designs specify power intent through UPF and rely on synthesis to correctly insert the power elements in design. All power elements are available in the netlist at the gate level and allows to perform CDC verification. However, CDC verification can be considered overdue at the gate level stage because there might be new CDC paths introduced in the design, which may lead to multiple iterations to fix the RTL/UPF. The time it takes to fix the RTL and run the verification again can impact the design schedules. Thus, it is highly recommended to verify power aware CDC at RTL level.

Following are the challenges to perform power aware CDC verification:

A. Isolation logic – Logic on clock and reset paths

Isolation controls are used to isolate the clock when it crosses power domains. In regular flows, clock gating is done with a latch which ensures that a glitch does not happen due to clock enable changing very close to clock edge. When isolation control is inserted through UPF, it may be connected directly to the clock without going through the clock gating latch. Figure 2 shows an example of modeling isolation control on the clock in UPF.

```

set_isolation_control D_mem_low_clk_els_iso \
  -domain "PD_mem" \
  -isolation_signal "mem/ctrl/do_iso_en" \
  -isolation_sense "low" \
  -location "self"
  
```

Figure 2. Isolation logic on clock paths in UPF

In another case, if the isolation signal is driven from an asynchronous clock domain, a glitch might be created on the clock path. A glitch on clock can be dangerous and can lead to metastable or undefined behavior. Figure 3 below shows an example where an isolation cell inserted at the clock path creates a combinational logic in the clock path, which can result in glitches on the clock path.

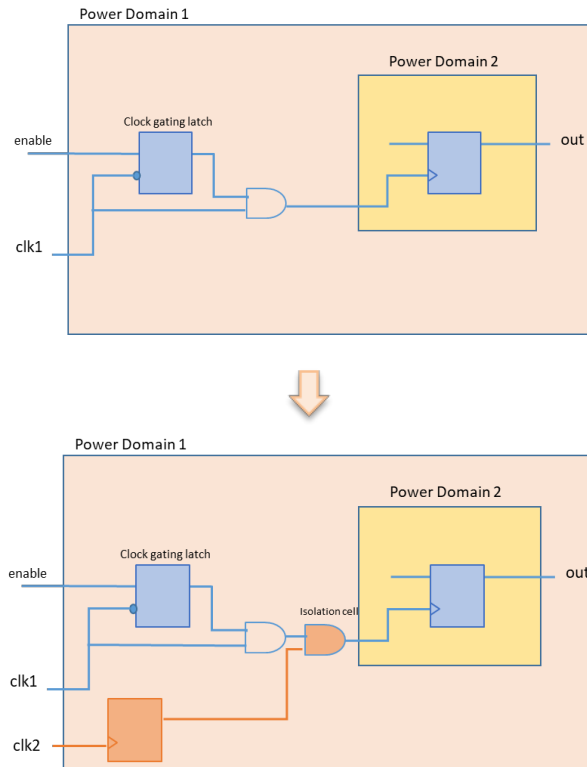


Figure 3. Glitch prone combinational logic on clock path

B. Isolation logic – New CDC paths

Isolation cells are inserted at the boundary of a power domain to ensure that the receiving logic always obtain an unambiguous 1 or 0 value. Isolation may be inserted for an input or for an output of the power domain. An isolation cell operates in two modes: normal mode, in which it acts like a buffer; and isolation mode, in which it clamps its output to a defined value. An isolation enable signal determines the operational mode of an isolation cell at any given time. The isolation cells are modeled as combinational logic and may create combinational logic or missing synchronizer violations.

```
## Isolation of outputs
set_isolation do_iso \
-domain PD_mem \
-isolation_supply_set PD_mem.isolation \
-elements { fifo_l_d } \
-clamp_value 0 \
-applies_to outputs \
-isolation_signal do_iso_en \
```

Figure 4. Modeling of isolation logic in UPF

An isolation cell introduces a new path from the enable signal to the destination register and may also introduce a missing synchronizer violation between the isolation enable signal and the destination register (see Fig. 5). If the isolation enable signal is in a clock domain that is asynchronous to the destination register's clock domain, the assertion and de-assertion of the isolation cell enable may cause an asynchronous edge at the output register. When a missing synchronizer is detected, designers should correct the power control logic by synchronizing the isolation enable signal to the destination register's clock domain.

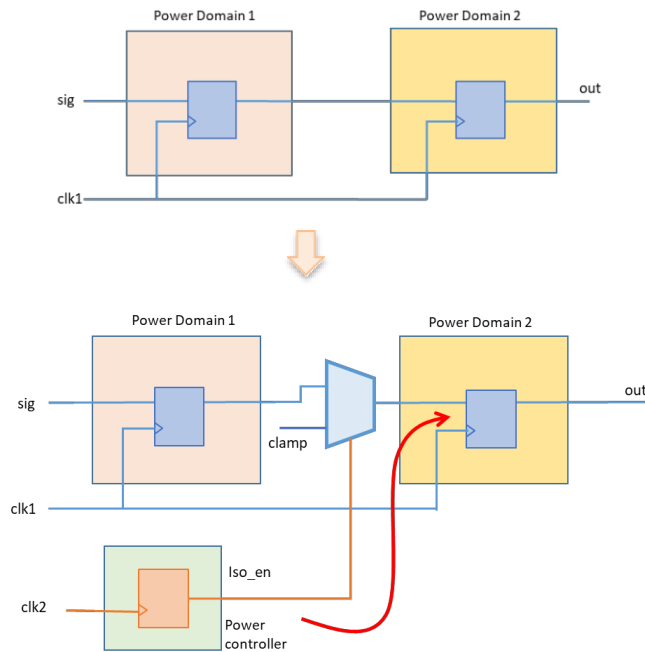


Figure 5. Isolation cell introduces a CDC path between synchronous registers

C. Isolation logic – Broken CDC synchronizer

Isolation cells may introduce additional combinational logic, which is prone to create glitches, on properly synchronized CDC paths. Glitches may be captured at the asynchronous boundary and can cause critical design errors. Moreover, incorrect logic value can be propagated when no design changes were expected (see Figure 6). Power aware CDC verification tools work on the RTL design and UPF, and detect combinational logic violations along with new CDC paths created between the isolation cell enable and the destination register. Designers must review the paths and fix them by either adding additional logic to ensure that unwanted glitches do not propagate, or by updating the UPF to validate that the isolation cell inserted between flops does not create additional logic.

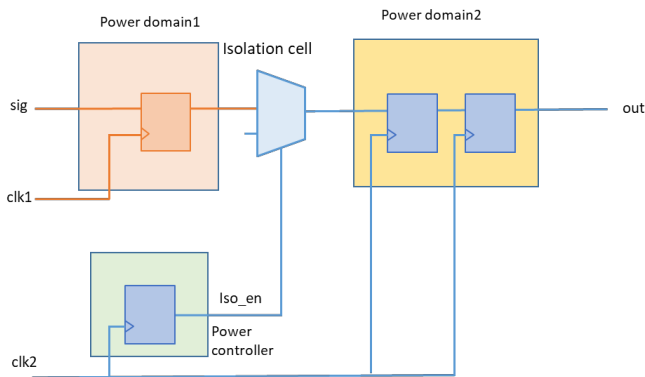


Figure 6. Isolation cell introduces additional logic on synchronized CDC path

D. Retention logic

State retention is the ability to retain the value of a state element in a power domain while switching off the primary power to that element; and being able to use the retained value as the functional value of the state element upon power-up. State retention can enable a power domain to quickly return to operational mode after a power-down/power-up sequence. State retention can also be used to maintain state values that cannot be easily recomputed on power-up. State retention is implemented using retention memories or retention registers. Retention registers are sequential elements (latches or flip-flops) that have state retention capability. To create retention capability registers, add additional save and restore pins.

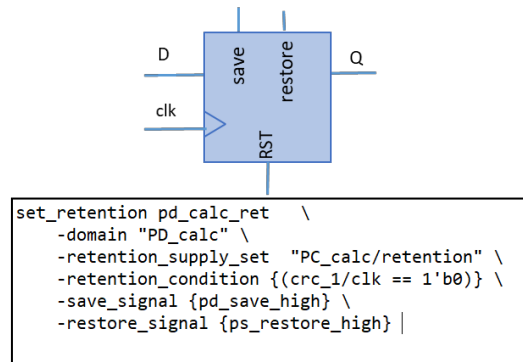


Figure 7. Retention register with additional pins

Retention Save and Restore paths must be synchronized before reaching the retention registers. Designers must identify the Save and Restore signals generated in the asynchronous clock domain (Figure 8) and address them to ensure that every path through retention registers are synchronized.

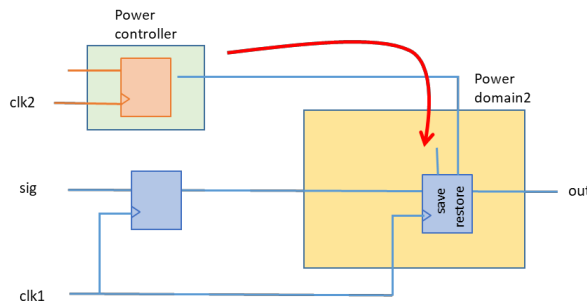


Figure 8. Additional CDC paths created due to retention restore signal

E. Hierarchical UPF

Due to the increasing design sizes, more and more teams are adapting the hierarchical flow. This flow applies to power aware verification as well. In hierarchical UPF flow, block level power intent is described within the scope construct of the block level UPF. Block UPF can be used to synthesize block RTL. Designers can then create a merged power aware netlist at the top level by using block UPFs from different blocks. Advanced CDC tools have one to one mapping with hierarchical flow. CDC tool can capture the CDC paths and relevant information in the block hierarchical model. Block hierarchical models are then used at top level to complete the CDC verification. At block level, only the intra-block CDC paths are analyzed and at top level inter block CDC paths are analyzed.

III. CASE STUDY ON ACTUAL DESIGN

The power aware CDC analysis flow was tried on a real SoC which contained multiple blocks. Blocks are the subsystems that contain their own power management logic. A 3-step hierarchical methodology was followed as explained in Figure 9. First step in this methodology was to generate CDC constraints for blocks. Step 2 was to run power aware CDC analysis on different blocks with block level UPF and generated constraints from Step 1. At the subsystem level, results of power aware CDC analysis were examined and issues were fixed, wherever required. After the subsystem level results were clean, top level power aware CDC analysis was performed with a hierarchical model for the blocks and top level UPF in Step 3.

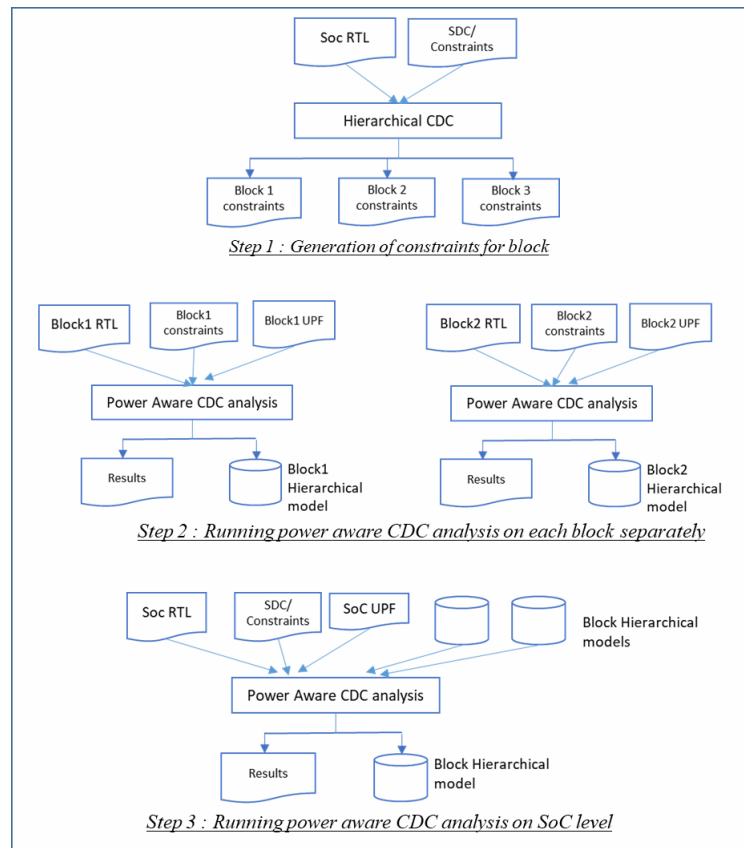


Figure 9. Hierarchical power aware CDC verification flow

Figure 10 shows the example of a broken clock issue found in the design. A clock was originally gated with latch-based clock gating. When the clock was crossing the power domain, an isolation control signal was added. Due to the isolation control signal, an additional gating signal was created in the clock path when it reaches to the second power domain, and became prone to glitch.

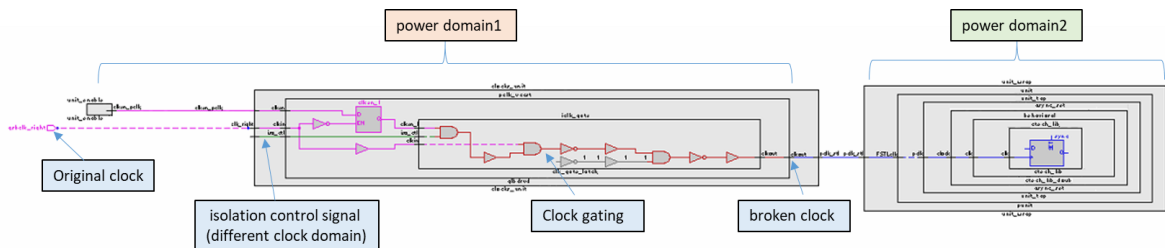


Figure 10. Broken clock path due to isolation control signal

Figure 11 captures an example of new CDC path creation due to the presence of isolation control signal. Here, when the signal was crossing the power domain, an isolation MUX was added to the path. Due to this addition, UPF inserts an isolation MUX that is driven from an asynchronous clock domain. The additional path between asynchronous clock domains is unexpected in the design and can cause meta-stability on the receiver side.

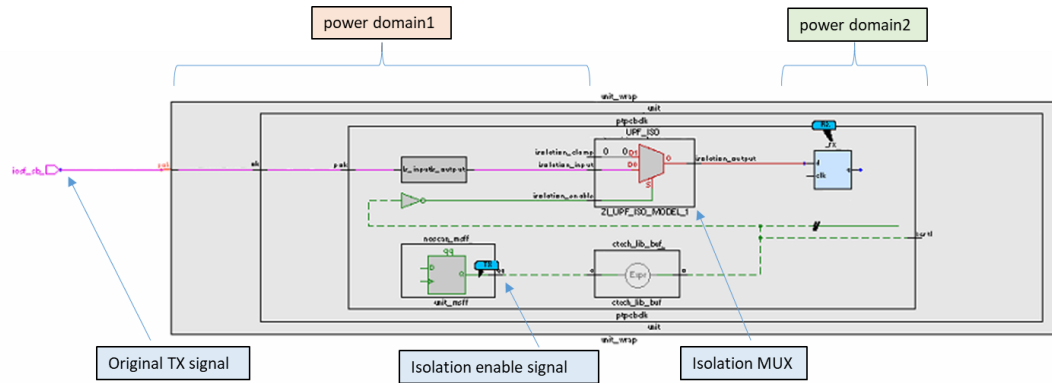


Figure 11. New CDC crossing created due to isolation control signal

Figure 12 shows an example of a broken CDC synchronizer due to the presence of power logic in the CDC path. A 2-DFF synchronizer was introduced in the CDC path between known asynchronous clocks. UPF inserted an isolation MUX between Tx and Rx to the isolation signal crossing power domain. UPF also inserted an unexpected isolation MUX that created a combinational logic which was driven from multiple clock domains. Such combinational logics are undesirable on the CDC paths. Designers must update the UPF to ensure extra isolation from asynchronous clock domain is not inserted in the CDC path.

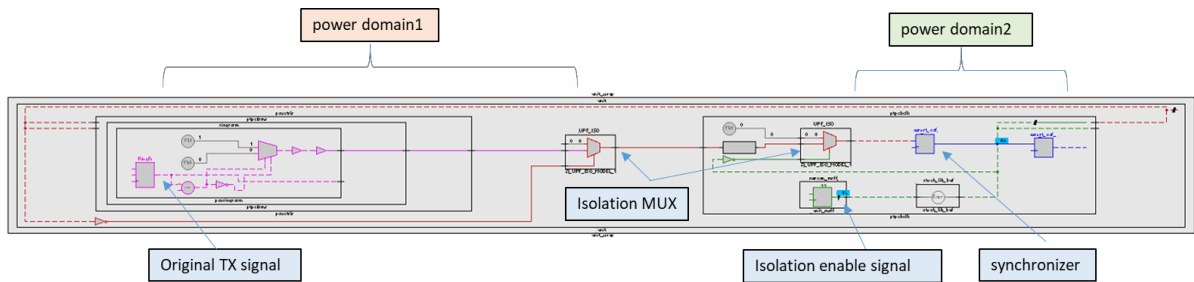


Figure 12. Two-DFF CDC synchronizer broken due to additional combinational logic in the TX-Rx path

Table 1 summarizes the results of the power aware CDC analysis on a real design. Power aware CDC analysis with mentioned methodology helped identify some critical issues in the design, which could have been missed otherwise.

Table 1: Power aware CDC analysis results on a real design

Design	Power domains	Isolation cells	Retention registers	Clocks broken	New CDC paths (isolation)	New CDC paths (retention)	Broken CDC synchronizer
Subsystem 1	3	329	0	25	1248	0	13
Subsystem 2	15	131	96	8	1236	1536	1



IV. SUMMARY

With advances in low power design methodologies and techniques, designers are facing new challenges related to the impact of power elements on CDC paths. By using the advanced techniques proposed in this paper, designers can start power aware CDC verification early in the design cycle. Performing power aware CDC verification at an early stage, such as RTL, identifies and enables designers to fix issues that would have been otherwise discovered post synthesis. Our validation on the actual design suggests that power aware verification with advanced technique can help close the CDC verification much faster and save from unexpected chip killing CDC issues.

V. REFERENCES

- [1] C.E.Cummings, "Clock Domain Crossing (CDC) Design and Verification Techniques using System Verilog," SNUG 2008.
- [2] A. Chen, et. al., "Power Aware Clock Domain Crossing Verification", DAC 2014.
- [3] P. Yeung, "Multi-Domain Verification: When Clock, Power and Reset Domains Collide", DVCon, March 2015.
- [4] E. Marschner, December 6, 2012, "The Next UPF", Semiconductor Engineering, <http://semiengineering.com/the-next-upf>.
- [5] P. Kothanath, "Clock Domain Crossing Verification for an SoC : Beyond The Usual Suspects", Mentor User2User, October 2014.