Left Shift Mechanism to Mitigate Gate Level Asynchronous Design Challenges

Kotha Kavya ,Sinha Rohit Kumar

Intel Technology Private Ltd, Bangalore, India -560103

2022 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES VIRTUAL FEBRUARY 28 - MARCH 3, 2022

INTRODUCTION

Advent of complex clock-reset architecture and the increase in reset signaling complexity with the emergence of multiple reset domains, gate level clock and reset domain crossing verification becomes an absolute need to ensure glitch free reset assertions during various power states.
Many Challenges leading to potential metastability at gate level Design
Synthesis changes such as optimization as sequential optimization, merging, duplication etc done with respect to the RTL
Clock-Reset paths changes during synthesis consist CTECHs(Component TECHnologies)
Synthesis/PNR(Place and Route) introduces any new glitches in the async paths
ECO(Engineering change order) changes leading to existing or broken CDC paths
DFT(Design for Testability) Scan insertion changes during synthesis flows
Introduction of isolation logic can impact clock and reset paths leading to glitches
Isolation logic on a synchronized CDC path can result in combinational logic in the data path that should be guarded against glitches.
Enable path of the ICGs can create a new asynchronous path
Introduction of isolation logic can create new CDC paths if the isolation enable signal is driven by an asynchronous clock domain.
Retention Save & Restore logic can have asynchronous signals driving save and restore, leading to additional CDC paths.
Need to ensure all the CDC or RDC issues post low power cell insertions, scan insertion and synthesis optimization are left-shifted in the Front-End CDC or RDC.
So, we have proposed few simple techniques in the front-end verification to address issues which are mainly related to implementation flows.

Low power cell instrumentation leading to New or Broken CDC or RDC paths

Absence of power intent in the RTL and extraction from the UPF file during synthesis.
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.
Clock and CDC errors can occur when the power elements are incorrectly inserted in the clock tree, the reset tree, or the CDC paths.
New CDC paths introduced in the design, which may lead to multiple iterations to fix the RTL/UPF.
Identification of critical path late in the design flow.
Results in incorrect functionality that may not be reproduced in simulation and are extremely difficult to debug in silicon

Traditional design architectural techniques to handle low power instrumentation related CDC/RDC issues

Clock -Assertions Used to validate to both source & destination side are off while switching on the power : This mechanism is a standard technique to ensure the safe asynchronous crossing by the means of design architecture
Synthesis- Launch and Capture clock for signals going through isolation cells : As per the design review, need to ensure the power control signal crossing one domain to another, and is signed off through manual reviews.
At netlist level, need to ensure clock are gated during transition
Verification - Review of default isolation value and value after reset deassertion need to validate between Integration and Verification Team
DFT – Scan review post scan-insertion flow
As per the standard design paranoia review, all the scan paths are reviewed post scan insertion and potential metastability issues are checked accordingly.

Proposed Flow for Low power cell instrumentation

As per the new proposed flow, we need to ensure that post Front end CDC and structural Low power signoff
UPF aware CDC is enabled, and all the violations post low power cell instrumentation is reviewed and signed off.

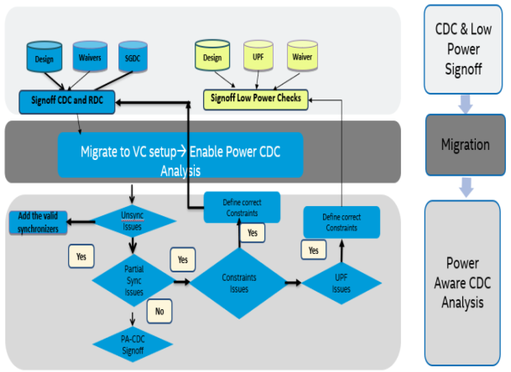


Figure 1: proposed flow to handle low power Instrumentation

Glitches Created by Synthesis Optimization

1) Insertion of Clock Gating Logic:
Issue related to clock-gating insertions can only be detected at the netlist level CDC verification, but it is mainly ensured using the design architecture that asynchronous path from clock enable circuitry is not prone to metastability threats

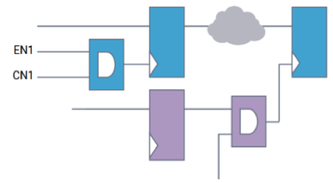


Figure 2: Clock gating logic Insertion

2) Retiming during synthesis/APR
Issue related to retiming can only be analyzed at the gate-level, however retiming is only done for the synchronous
paths and movement of logic across asynchronous paths are not done during the retiming process.

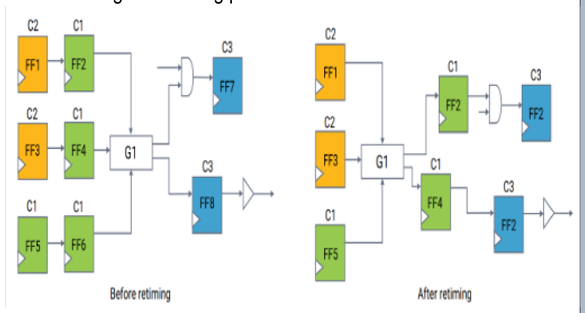


Figure 3: Before and After retiming

3) Synthesis Optimization of muxes in the async path
Used to identify potential glitches source in the async paths and providing the utility to RTL designer to follow design practices which will prevent the mux decomposition which could potentially lead to glitches at the gate level netlist

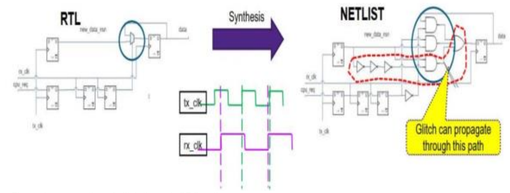


Figure 4: propagation of glitch through CDC path

Proposed Flow to prevent Mux decomposition during Synthesis Optimization

Identify the Muxes in CDC path using vendor tool schemes :

i. Recirculation MUX Synchronization Scheme
ii. MUX-Select Sync (Without Recirculation) Synchronization Scheme

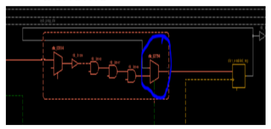


Figure 5: Recirculation MUX Synchronization Scheme

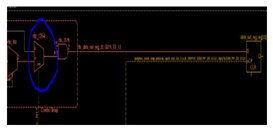


Figure 6: MUX-Select Synchronization Scheme

Add //infer_mux_override in the RTL of the MUX that were reported by above two schemes

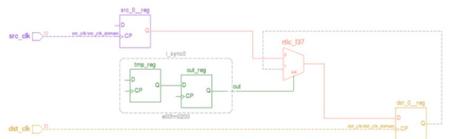
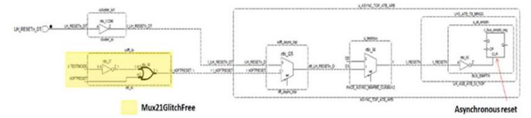


Figure 7: pragma to identify muxes in CDC paths

EDA tools will check mux_inferencing RTL code to determine if an pragma //infer_mux_override exists via Ac_cross_analysis rule
Seeing this pragma synthesizable tool can distinguish the CDC muxes

Glitch in Reset Paths

Introduced due to incorrect modelling in the RTL
Synthesis tool instantiates regular mux instead of the glitch-free mux
Result unexpected glitches in the reset path



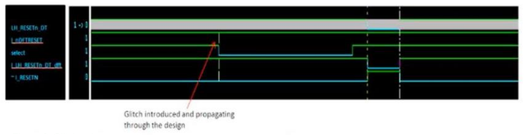


Figure 8: Glitches in reset path

```
201 //`MUX51_ONEH0T (I_nDFTRESET, 1'b1, nDFTRESET, 1'b1, 1'b1, 1'b1, select
202 `MUX21GLITCHFREE(I_nDFTRESET, nDFTRESET, 1'b1, TESTMODE)
203
204
```

Figure 9: The commented line is the buggy(usage of glitchy mux on the signal)logic.The uncommented line is the fix(glitch free mux)

```
383
384 `MUX21 (I_LH_RESETn_DT_dft, nDFTRESET, I_LH_RESETn_DT, TESTMODE)
385
```

Figure 10: The reset propagation across another mux

Design Practices to avoid Metastability Issues

Following are some of the design practices which are adopted to ensure that metastability issues are not specifically introduced after synthesis and APR stage and front-end CDC analysis covers all the asynchronous design challenges.
Avoiding complex synthesis change done with respect to the RTL: This is the standard step that we should not be enabling any complex synthesis optimization.
Clock-Reset paths only consist of CTECHs
Synthesis/PNR doesn't introduce any new glitches
Ensuring ECO changes don't introduce any new glitches
DFT Full-Chip GLS is done with Test Enable
Val Full-Chip Power Aware GLS with zero-delay, SDF-max, SDF min

RESULTS

The techniques mentioned were first implemented in ARM-based SoC design and it is currently in the POR stage.
In the hierarchical SoC design, the flow was integrated at p05 milestone itself and all the CDC/RDC related issues that are Introduced during the implementation flows are analyzed in the front-end asynchronous
Analysis using techniques such as power aware CDC, glitch Analysis, following standardized design practices or extensive design reviews

SUMMARY

Presented several gate level CDC/RDC verification challenges and proposed various solution to address the problems.
Demonstrated the solution through verifying one of our SoC design, and presented the issues discovered.
Left shifting gate level challenges and ensuring all aspects of gate level CDC/RDC verification is covered, requires a full suite of methods involving enabling various flows, design practices, RTL change etc.