Dynamic CDC Verification for Memory Design

Inryoul Lee, Jiyoung Ye, Gayeong You.

Memory Business Division, Samsung Electronics Co., Ltd.

Hyobeen Park, Nikhil Rahagude, Vikas Sachdeva

Real Intent, Inc.

Motivation

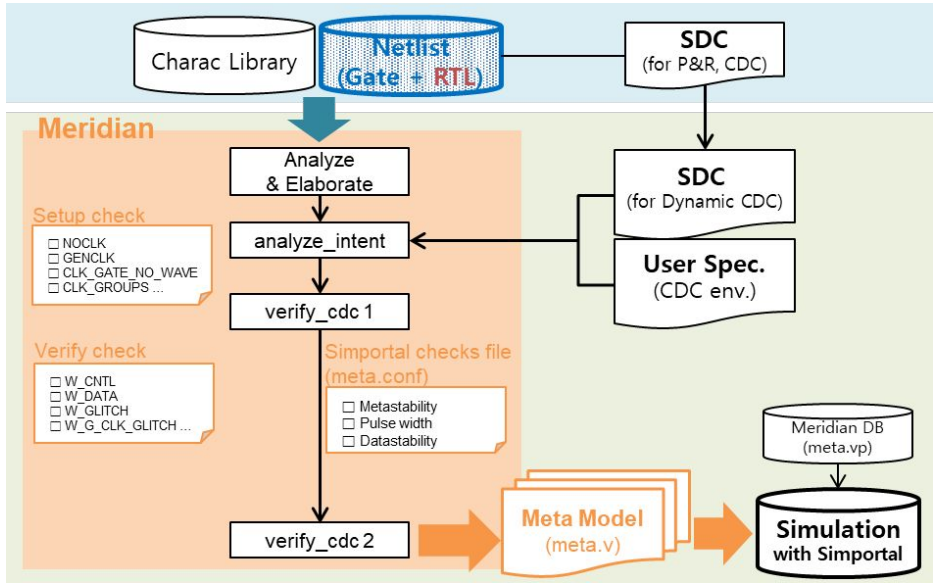
- **Memory design consists of a large number of asynchronous clock domains and interfaces.**
  - Timing on such paths can't be guaranteed, with the net effect that the destination flops may become Metastable.
  - These Metastable problems can be found through by several static CDC analysis tools for RTL or netlist designs.
  - There are so many issues reported by the static CDC. It is difficult to distinguish between false and true, resulting in a lot of human errors.

- **CDC issues are the common reason for silicon re-spins.**
  - A reason is missing correlation between static CDC verification and dynamic logic simulation.
  - Like some Issues of fast clock to slow clock.
  - To reduce the gap, we need to make the new CDC sign-off flow based on CDC data generated by up-front static analysis.

We develop the New Dynamic CDC methodology for Memory product by partnering with Real Intent(RI) to deliver required enhancements in Meridian CDC.

Main Idea

- **New Dynamic CDC(Simportal) Flow**
  - **Analyze and Elaborate –** In this step, we provide a design input which is a mix of RTL and gate-level netlist in memory design. The static CDC tool compiles and elaborates the design.
  - **Analyze intent –** This step reads constraints of SDC format. The static tool critiques whether setup is complete or if there are constraints missing in the input file.
  - **Verify CDC –** This step creates a structural CDC report and also generates a corresponding meta-stability-aware model to be used in simulation.
  - **Metastability-Aware Simulation –** The meta-stability-aware model is ported in simulation and run across the regression test cases.



- Figure 1 -

Main Idea (cont')

- **Optimization Items for Memory design**
  - The Dynamic CDC is RTL verification methodology and EDA tools only support RTL design.
    But our design consists of RTL and gate-level netlist. Thus, we need to enhance the method and tool for Memory design.

I. Application development for Post-Simulation environment. (w RI)

✔ **Need to cell delay of gate level F/F.**
: Gate level netlist has clock skew, it may malfunction due to hold violation at next FF, and result of simulation differs to normal CNTL function.

>> We make the setting option for F/F cell delay(Delay of Clock to Q).

✔ **Need to interconnect delay for Tx to Rx.**
: There is a timing latency between the Tx F/F Q stage and the Rx F/F DI.

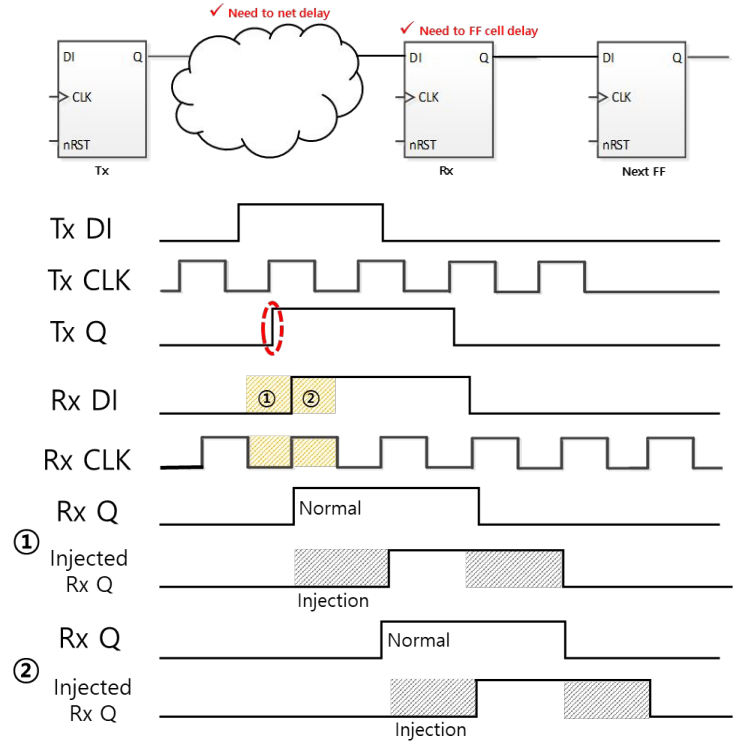
The arrival time of DI based on Rx Clock may be fast or late.

Meta injection considering this is required.

>> We make the Post-sim. mode. (Figure 2)

① If Rx DI transition faster than Rx CLK,
Simportal injects reverse signal to the Rx Q at first clock edge.

② If Rx DI transition slower than Rx CLK,
Simportal injects reverse signal to the Rx Q at next edge.



- Figure 2 -

Main Idea (cont’)

- **Optimization Items for Memory design**
  - The Dynamic CDC is RTL verification methodology and EDA tools only support RTL design.
    But our design consists of RTL and gate-level netlist. Thus, we need to enhance the method and tool for Memory design.

II. SDC(Synopsys Design Constraints) optimization.

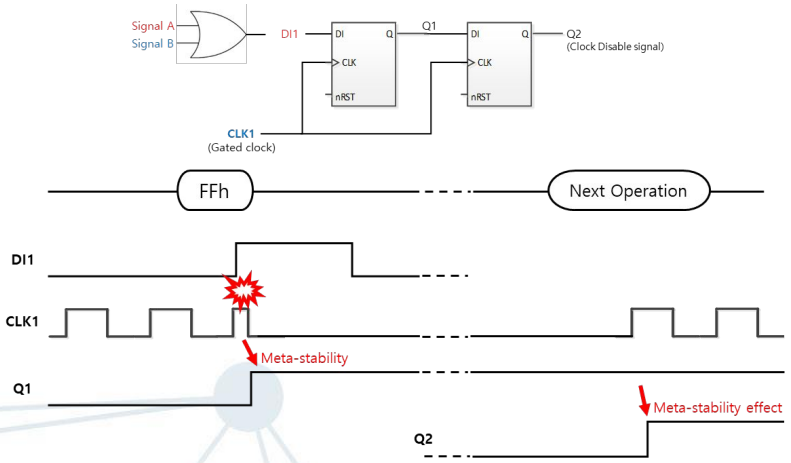
✔ SDC of Memory design has many clock defines because of the timing guarantee.
To reduce garbage fail, we optimize clock define & grouping in SDC file.
We used clock about the user spec only, not test mode and IO clocks.
>> 166 Clocks 69 Clocks & 42 Clock group

III. Simulation Runtime improvement in DATA path.

✔ Memory design has asynchronous data domain and interface.
As a result, many DATA paths are created, and printing large Meta-modeling file.
To reduce simulation TAT, we delete and split the DATA paths modeling files by function.
>> Simulation run time : x5 or more improvement

Evidence

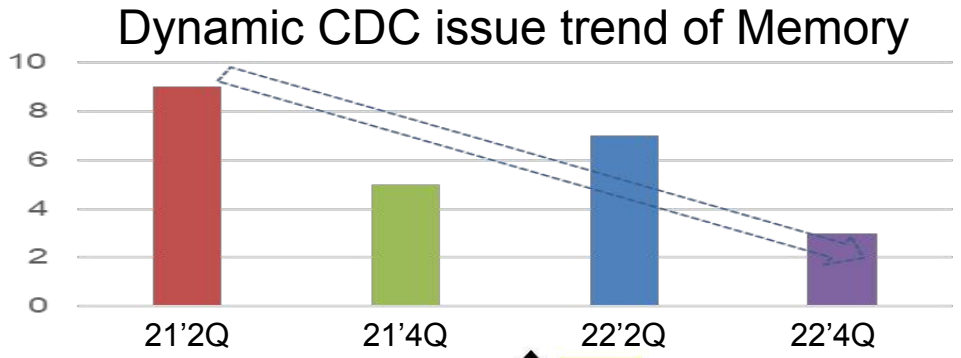
- **Mata-stability issue in Memory design**
  - Red and blue signal is different clock domains. Q2 signal affects the operation of the FSM.
  - If FFh(reset) Command is invoked simultaneously when the Flash is ready, the FSM is disabled.
  - In this case, by Meta, Q1 is high, but Q2 remains low.
  - And when the next operation is invoked, the CLK1 starts the operation again. But Q2 transitions to high so that the FSM is disabled. Eventually, the deadlock occurs in the Flash.
  - We were able to catch this failure with Simportal flow in our simulations.



- Figure 3 -

Summary

- **Conclusions**
  - Static analysis is very effective in this regard and reliably able to identify deviations from asynchronous interface mediation principles. While almost all CDC sign-off can be completed based on static analysis, a residual sign-off gap does remain. To close this gap, verification must functionally exercise the crossing including modeling meta-stability and its effects in simulation. Memory designs have some special characteristics like mix of RTL and gate-level descriptions, larger number of asynchronous paths and combinations of multiple asynchronous domain paths at several locations. The CDC sign-off flow must be adapted accordingly in light of these special characteristics. We presented memory design specific modified static flow with integration of Simportal technology. These changes allowed us to catch critical design bugs.



- Figure 4 -