Shift Left DFT Sign-off Methodology for Edge AI Processor

DAC Case Study by Parag Dass & Venkat Busani of Kinara

Executive Summary: DAC 2022 Kinara case study (presented by Kanad Chakraborty of Real Intent)

Case Study Overview

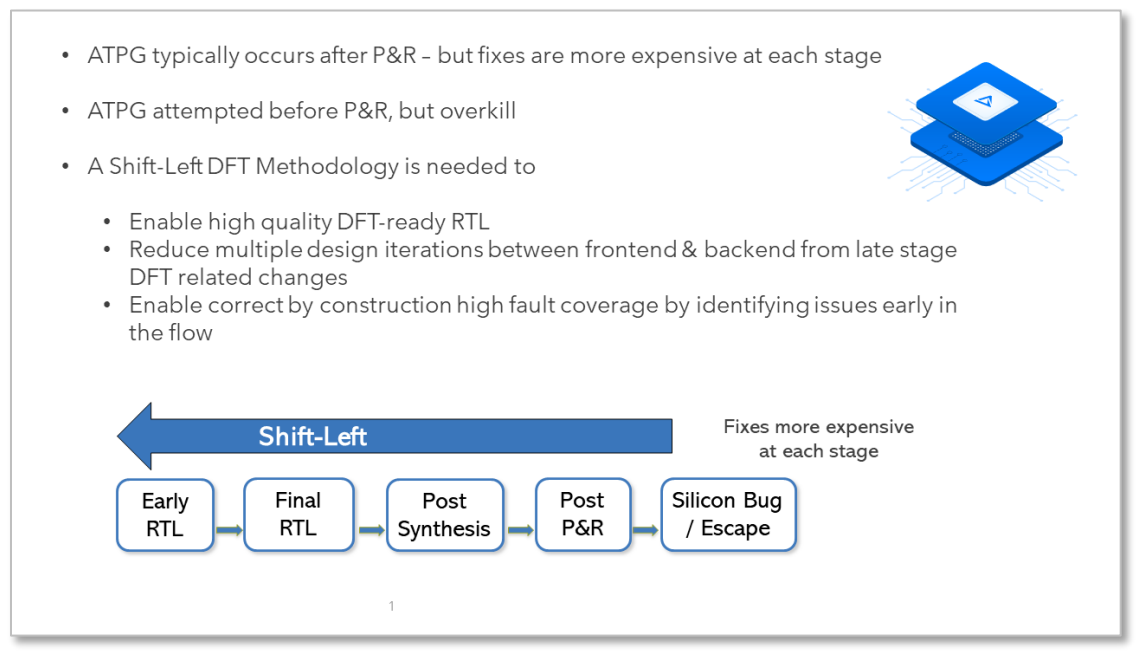
This case study covers Kinara's advanced DFT sign-off methodology to ensure their AI processor has high quality DFT-ready RTL and high fault coverage, while meeting tight delivery timeframes. Kinara deploys Real Intent Meridian DFT in this flow.

DFT Shift Left: Quality, Fault Coverage, & Faster Sign-Off

Kinara designs its ARA-1 AI processor with a specialized architecture for edge applications, including replicated cores and distributed memories.

The complex architecture of Kinara's ARA-1 AI processor makes it important to ensure high-quality DFT-friendly RTL, so as to minimize design iterations between RTL and netlist for faster sign-off.

Kinara's emphasis is on "shift-left" DFT multi-test-mode static sign-off. Fixes are less expensive when DFT bugs are found early. These bugs would likely be found at ATPG; however, fixing them after ATPG involves costly design iterations.



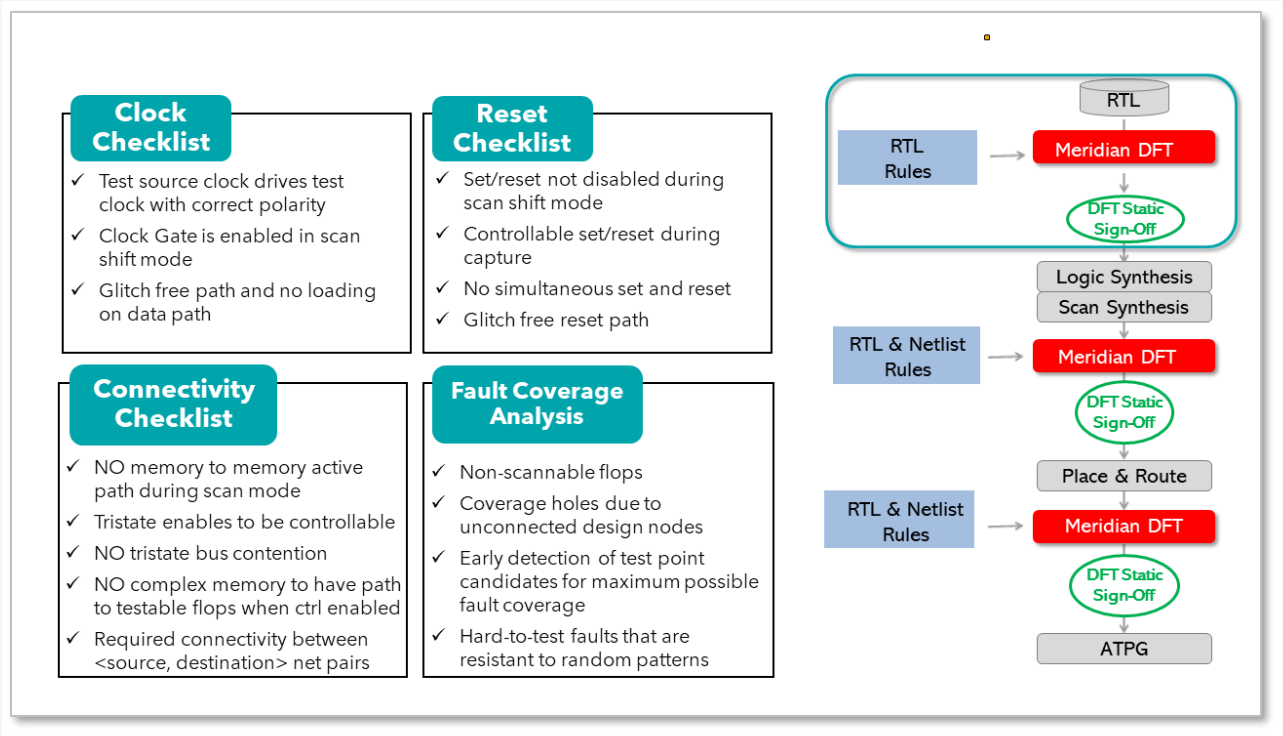
Kinara's early DFT sign-off involves

- Early checking of DFT DRC rules exhaustively for clocks, resets, & connectivity
- Enabling fault coverage analysis at RTL via early design exploration, e.g., early test point candidate detection, to achieve excellent correlation with ATPG fault coverage results

RTL DFT Sign-Off with Fault Coverage Analysis

Kinara performs early DFT sign-off using checklists of test clock, reset and connectivity rules. Using the clock and reset checks, Kinara identifies clock and reset controllability and glitch violations.

With connectivity checks, tristate bus contention and other such issues are also identified early.



Early fault coverage analysis allows them to find sequential capture depth through non-scannable flops, coverage holes due to undriven or unread nets, detect test point candidates, and perform optional analysis of random-pattern resistant faults.

This early detection of test point candidates uses information computed by Real Intent Meridian DFT and pre-fetched from test synthesis; it is applied at RTL for accurate correlation to ATPG fault coverage.

Multimode DFT Sign-Off with Bi-Polar Resets

Kinara has a novel use model for multi-test-mode DFT analysis at RTL. The chip has two blocks that are fed by a single, bipolar reset source. When the reset is active in one block, it is inactive in the second block.

This architecture lends itself naturally to two test modes, where the reset constraint is toggled between the two test modes for shift.

Problem: Asynchronous RST/SET should not be active during shift scan mode

- Cause:
  - Few blocks are working on active high and others are on active low reset for same reset source.
  - Example: B1 is driven with expected polarity hence disabled in shift mode whereas B2 is reset during shift which can erase data to be shifted.
- Fix:
  - Due to continuous maturing RTL, design logic is restructured to have all active low polarity.

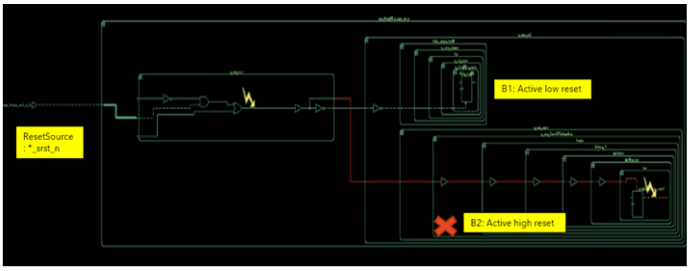
How to catch:

- Take advantage of multi-mode capability in Real Intent Meridian DFT
- Thus one two-mode set up for both polarities exercises both logic scenarios in single run

```
....
source global_constraints.tcl
enable_rules RESET_NOT_DISABLED -global

set_dft_test_mode M_ActiveHigh
set_dft_const srst_n -value 1 -scan_mode shift

set_dft_test_mode M_ActiveLow
set_dft_const srst_n -value 0 -scan_mode shift
....
```



With multimode DFT Kinara can verify both blocks with opposite constraints on the reset signal, along with their other DFT constraints, in the same run.

Errors Identified & Corrected Early

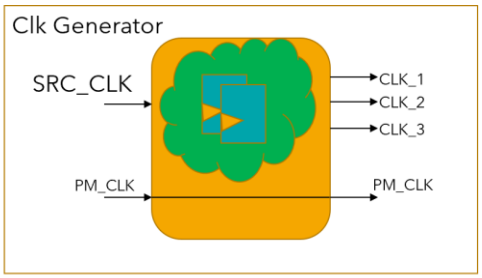
Case 1: Kinara identified multiple test clocks that were not controllable from the test source clock through the clock generator block. They then modified the RTL to enable control paths from the test source clock to every test clock.

CASE-1

Problem: Test clocks were not getting driven by source clock

Cause:
- Incorrect source clock driving child test clocks

Fix:
- Clock generator block is having SRC_CLK as source input which is generating test clocks internally
- Corrected the constraint for source clock to test clock (or RTL change)



Case 2: Kinara discovered that not all flip-flops were driven by a test clock. They changed the RTL to route the test clock to every flip-flop.

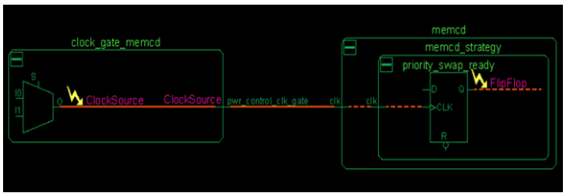
CASE- 2

Problem: During testability analysis, all FFs are not driven by test clock

Cause:

- Scan EN not connected at instance level
- Wrong clock connections i.e. clkA connected instead of clkB
- Typos regACLK instead of reg_ACLK

- Fix:
  - Expect RTL changes
  - As a workaround to proceed, added constant on enable signal



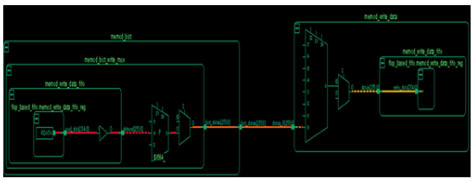
Case 3: Kinara found high sequential depth paths traversing through memory blocks that caused loss of fault coverage. Kinara inserted scan flip-flops in the paths to reduce the sequential depth.

They also identified sequential propagation issues through non-scannable flip-flops due to sequential loops and deep sequential paths through such flip-flops. Kinara broke up the paths and loops with scannable flip-flops.

CASE- 3

Problem: High sequential depth in TEST mode

- Cause:
  - Certain memory cells should not be actively connected with another memory cell during capture test mode
  - Or mem to flops in test hierarchy
- Fix:
  - RTL change is required to disable the path from one MEM to another as such sequential depth is undesired.



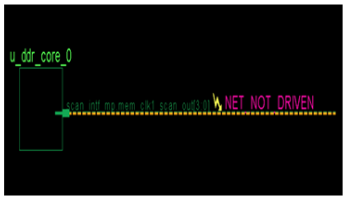
Memory Cell	Active State
mem1w1r	{ wen_a_i 1 }
sram2	{ CENA 1 CENB 1 WENB 1}
write_data_fifo_reg	{write_enable 1

Case 4: Undriven nets created inaccurate coverage estimates, because following synthesis, the undriven nets would be connected to DFT logic. Kinara was able to improve fault coverage estimation by applying user-specified test points to the nets.

CASE- 4

Problem: Targeted ~99% coverage not reached

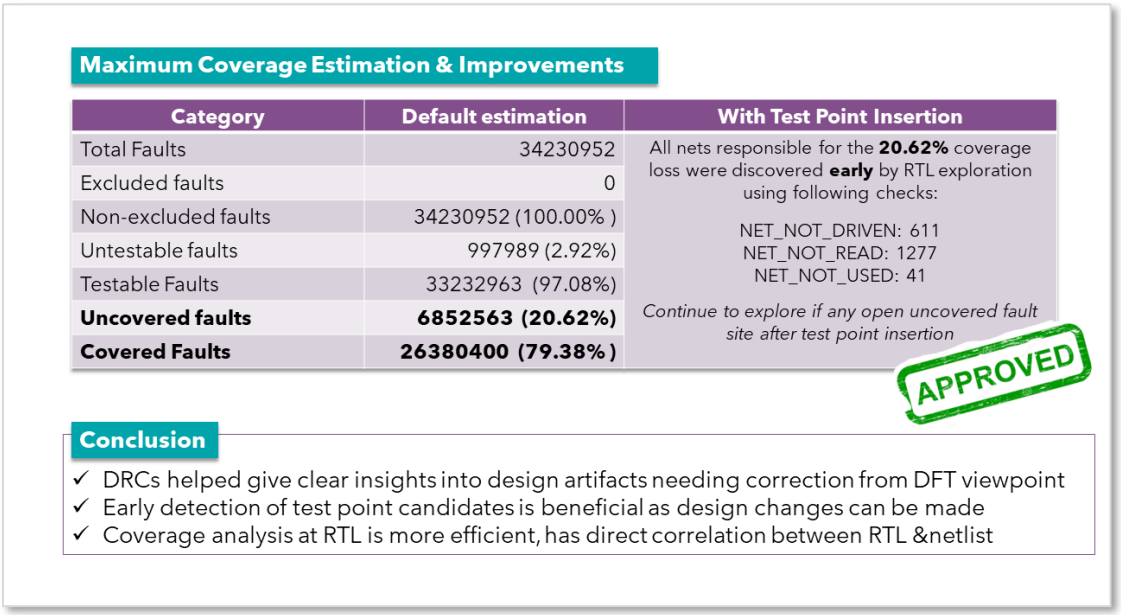
- Cause:
  - Major coverage hole is due to undriven nets causing controllability issue
  - And side inputs causing unobservability of unused nets
- Fix:
  - Due to continuous maturing RTL, test point is inserted at all such points



"Shift Left" DFT Sign-Off Methodology -- Coverage Results

To achieve DFT sign-off with high fault coverage for their RTL, Kinara 1) applied DFT rules and fixed errors, 2) identified test point candidates, and 3) performed coverage analysis.

1. Fixing the errors found by DFT rule checking achieved an estimated fault coverage of 79%. The integrated, interactive debug environment of iDebug provided a focused schematic that labeled the violations and improved debug efficiency.
2. Identifying test point candidates on undriven nets and adding test points to the fault coverage analysis (without actually adding them to the design) raised the total fault coverage estimate to 97%.
3. Coverage analysis allowed Kinara to visualize approximately 3% untestable faults in the schematic and cross-probe to the schematic source for the test points, violating flip-flops, and instances with highest number of uncovered faults.



Maximum Coverage Estimation & Improvements

Category	Default estimation	With Test Point Insertion
Total Faults	34230952	All nets responsible for the **20.62%** coverage loss were discovered **early** by RTL exploration using following checks: NET_NOT_DRIVEN: 611 NET_NOT_READ: 1277 NET_NOT_USED: 41 *Continue to explore if any open uncovered fault site after test point insertion*
Excluded faults	0	
Non-excluded faults	34230952 (100.00%)	
Untestable faults	997989 (2.92%)	
Testable Faults	33232963 (97.08%)	
Uncovered faults	**6852563 (20.62%)**	
Covered Faults	**26380400 (79.38%)**	

Conclusion

- ✓ DRCs helped give clear insights into design artifacts needing correction from DFT viewpoint
- ✓ Early detection of test point candidates is beneficial as design changes can be made
- ✓ Coverage analysis at RTL is more efficient, has direct correlation between RTL &netlist

Additionally, color-coding for uncontrollable, unobservable and constant nets assisted in identifying the coverage loss sources.

Kinara continued to perform RTL exploration using Meridian DFT rules to refine the design and reduce the percentage of untestable faults further.

###