Critical Area Weighted Pattern Optimization

Ron Press Director, Technology Enablement Tessent Test Solutions

February 2021

Restricted | © Siemens 2021 | R. Press | Tessent | 2021-01-11

Tessent TestKompress defect-oriented/automotive-grade ATPG

- New defect-oriented fault models detect defects missed by traditional methods
- New critical area based optimization uses defect likelihood & reduces pattern size by up to 40%



We have shown with test system results from a 14nm FinFET chip, that CAT patterns detect a huge amount of 4300 DPPM which are otherwise not detected with traditional SA and TDF patterns. We have shown on 150 SLT rejects that all passed traditional SA and TDF tests, but failed both SLT and CAT patterns. This resulted in the decision at Intel to make CAT patterns POR (Plan of Record)

W. Howell (Intel), et.al, ITC 2018



We have found that the bridge component alone reduces scan-related DPPMs by more than 700 compared even to cell-aware patterns P. Maxwell (ON Semiconductor), et.al, ETS 2017

Executive summary

- New defect-oriented fault models show significant DPM impact
- Traditional ATPG metrics use faults treated as equally likely
- Critical area weighted pattern selection provides a consistent metric based on likelihood of detecting physical defects:
 - -Choose the most effective patterns from multiple pattern sets
 - -Order patterns by most effective
 - -Pick the best sample of new pattern types to try in production
 - -Create a smaller pattern set for multiple fault models in one ATPG run
- Note: patterns such as timing-aware might be separately needed to target marginalities



Agenda

Choosing targets for pattern types and coverage

- -How do you know a pattern is going to be effective?
- Effectiveness of new pattern types
 - -How choose which to supplement existing pattern sets with?
- Grading pattern value by likelihood to detect physical defects
- Automatic pattern sorting and selection

NEW PATTERN TYPES WITH SIGNIFICANT SI RESULTS

Bestricted | © Siemens 2021 | R. Press | Tessent | 2021-01-1

Tessent TestKompress defect-oriented/automotivegrade ATPG

Creates fault models and test patterns that detect defects at the transistor and interconnect levels missed by traditional methods









We have shown with test system results from a 14nm FinFET chip, that CAT patterns detect a huge amount of 4300 DPPM which are otherwise not detected with traditional SA and TDF patterns. We have shown on 150 SLT rejects that all passed traditional SA and TDF tests, but failed both SLT and CAT patterns. This resulted in the decision at Intel to make CAT patterns POR (Plan of Record)

W. Howell (Intel), et.al, ITC 2018





We have found that the bridge component alone reduces scan-related DPPMs by more than 700 compared even to cell-aware patterns

> P. Maxwell (ON Semiconductor), et.al, ETS 2017



Timing-aware with cell-internal/cell-aware test



SIEMENS



Restricted | © Siemens 2021 | R. Press | Tessent | 2021-01-11

7

Timing-aware cell-aware test usage

Documented in "Scan and ATPG user's Manual"

- -Defect-Oriented Test/Timing-Aware Pattern Generation for cell-internal Defect
- -Example script to generate patterns for small delay defects with topoff patterns for remaining delay defects:

```
set fault type udfm -delay fault
read fault sites <tech>.udfm -filter {impact >= 50}
read sdf <design>.sdf.gz
set atpg timing on
set atpg timing -clock waveform <design settings>
add faults -all
create patterns
write patterns CAT timing.stil -stil -replace
set atpg timing off
delete fault sites -all
read fault sites<tech>.udfm
add fault sites -all
add faults -all
read patterns CAT timing.stil
simulate patterns -source external -store pattern none
create patterns
write patterns CAT delay.stil -stil -replace
```

Getting started with automotive-grade ATPG

• Tessent software tree contains two testcases for automotive-grade ATPG

1. Create an empty run directory in which you want to do the run. Then enter below command to check if all needed tools are accessible:

cellmodelgen -check_tools

 Extract a few Unix C-shell scripts to run the tool fully automated on the CellModelGen 24nm library: cellmodelgen -get script all



USING CRITICAL AREA FOR RELATIVE LIKELIHOOD

Restricted | © Siemens 2021 | R. Press | Tessent | 2021-01-1.

Overview critical area calculation



Dimensions distance (d), length (l) or width (w) are expressed in units of "technology length" (tl);

- 1 tl equates to the width of metal1 in a technology
- Critical areas are expressed in units of "technology squares" (ts)

Critical area calculation considers:

- the spot size
- the distance between adjacent objects (for bridges)
- the width of the net object (for opens and vias)
- the length of the adjacency

Lower distance and larger the bridging length equals higher probability of a bridge

Larger distance and smaller net width equals higher probability of an open

TOTAL CRITICAL AREA BASED PATTERN SORTING

Restricted | © Siemens 2021 | R. Press | Tessent | 2021-01-1.

Automotive-grade ATPG and diagnosis flow





SIEMENS

Complete DOT – Putting it all together Shortest test time & highest quality



- Just two ATPG runs will result in shortest test time with highest pattern quality
- The final production test patterns can now first time be optimized based on TCA

	#pat	#faults	TCA [ts]	FC [%]	TCA [%]	RE [%]	ТС [%]	UC+AU [%]
Delay	7405	25.9M	278M	88.58	91.10	2.44	91.01	8.99
Static	4782	22.9M	203M	96.27	96.61	2.76	99.16	0.84
		Source: F. Hapke, P. Maxwell, Mentor/ON Critical Area Based Test collaboration 2018						

SIEMENS

Defect-based pattern sorting

What

Ability to sort patterns based on Total Critical Area (TCA) values specified for UDFM faults

How

New option enables sorting of internal pattern sets by critical area

order_patterns -critical_area

External pattern sets must be simulated with UDFM before ordering

```
read_fault_sites cells.udfm
read_fault_sites bridge.udfm
read_patterns stuck_at.patdb
read_patterns static_bridge.patdb -append
set_critical_area_options -reporting ON
add_faults -all
simulate_patterns -source external -store_patterns all
order_patterns -critical_area
```

Value

- Easy way to determine the most effective patterns when truncating
- Faster defective part detection by ordering by likelihood of detecting silicon defects
- Any pattern set can be reordered based on critical area

Impact

Requires an Automotive-Grade ATPG license

Usage models

- 1. Pick a small sample of patterns for new fault model to try
- 2. Mix in new patterns with existing pattern set (shown in previous example)
- 3. Create patterns for all static or dynamic fault models in one ATPG run (recommended)
 - 1. Results in smaller pattern set than separate ATPG runs
 - 2. Method read UDFMs and create patterns
- NOTE: static (1-cycle) and dynamic pattern generation and sorting need to be performed as separate runs

Guidelines and example testcase

 Automotive-grade ATPG is included in the automotive reference flow in the Tessent Shell User's Manual

-Includes an open source testcase in the software tree

 Critical area weighted pattern optimization is included in the same automotive reference flow and testcase



- Critical area weighted pattern selection results in a major change in your ability to have a consistent metric to select or optimize patterns for the best DPM in the shortest test time
- The suggested process for generating high quality defect-oriented/automotive-grade tests and critical area pattern sorting is included in the Tessent Shell User's Manual
 - -The documented process has a corresponding open source testcase in the Tessent software



THANK YOU