



Logic BIST Observation Scan Technology (OST) and x-tolerance for BIST designs

Oct 2023

Agenda

Observation Scan Technology (OST) - overview

Functionality

Results

Logic BIST X-tolerance with OST

Key Metrics for In-System Testing

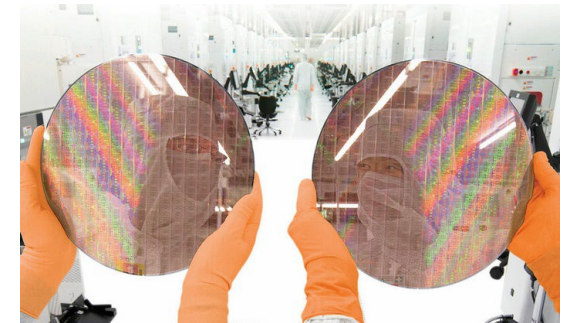
Test time

- Time required to execute the test in-field



Test quality

- Desired test coverage levels dictated by ISO 26262



Test cost

- Silicon area overhead to implement desired solution

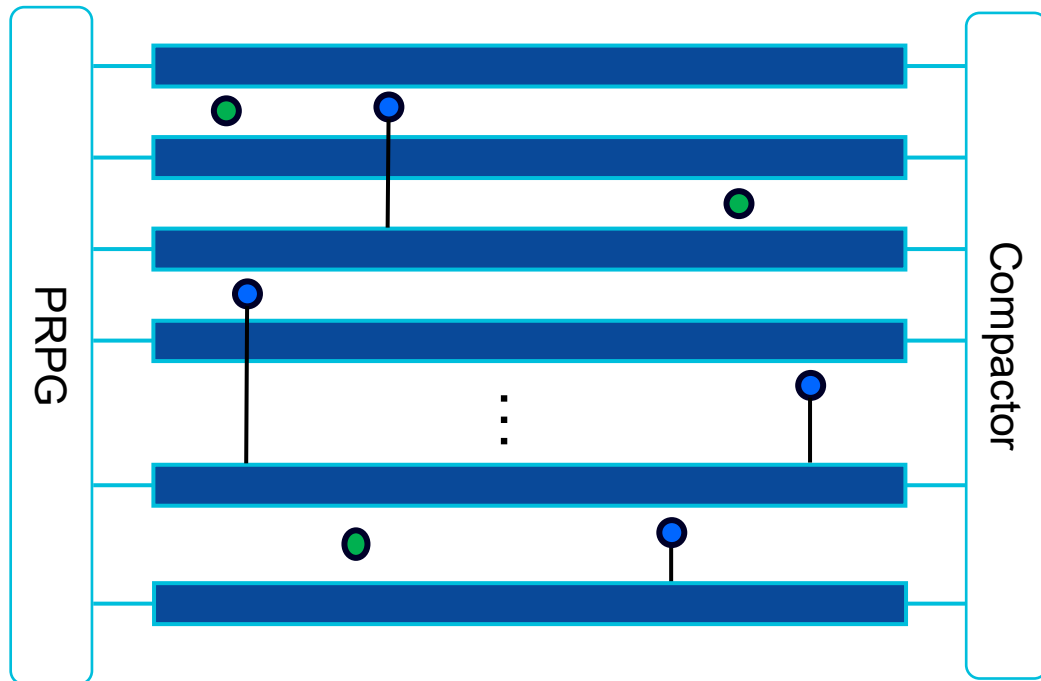


Up to 10x Fewer Patterns – Reduced DTI and Aging Impact

Tessent LogicBIST with Observation Scan Technology

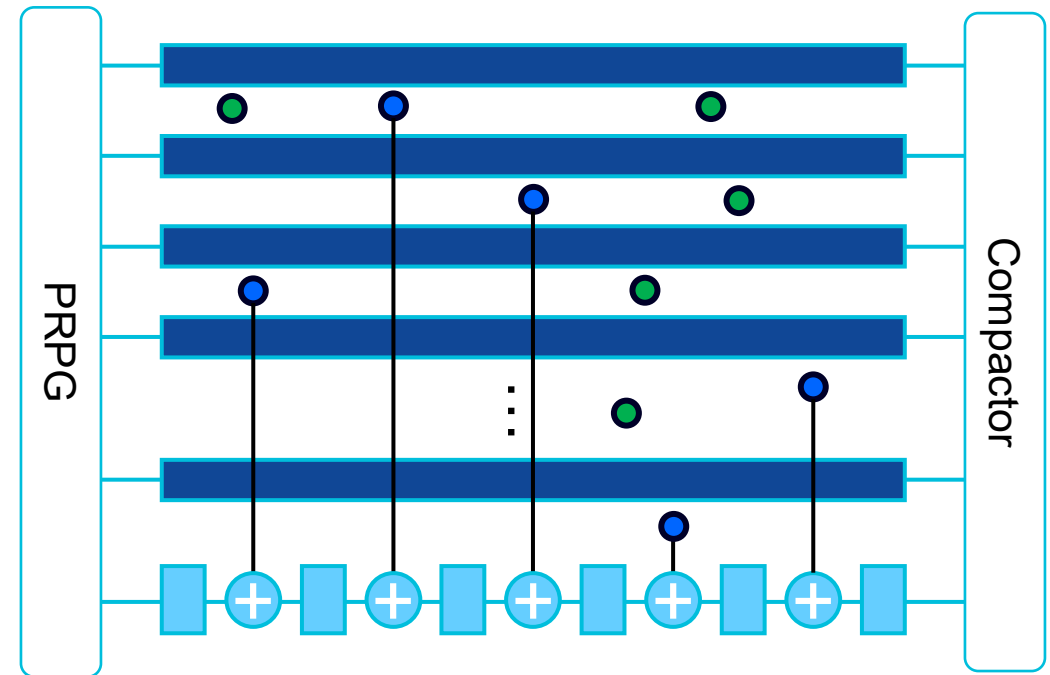
LBIST

(per pattern observation)

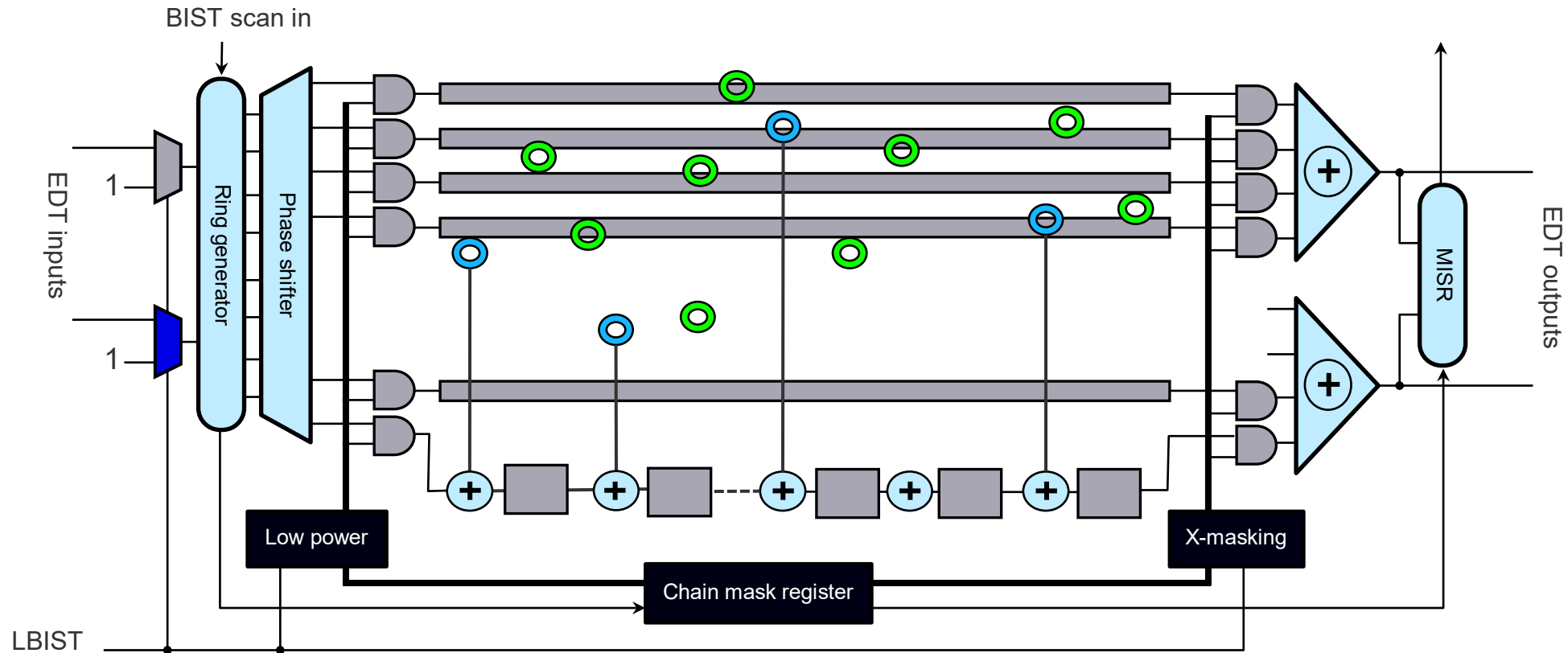


LBIST-OST

(per cycle observation)



Observation Scan Technology (OST) for LBIST designs



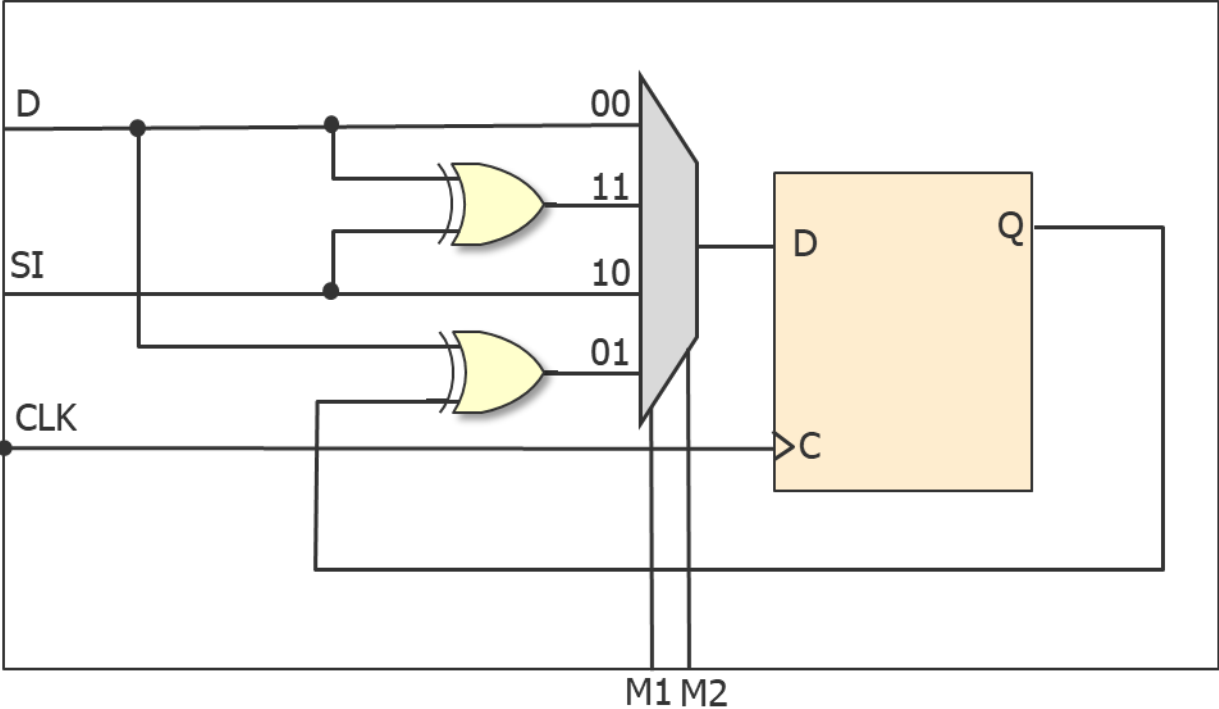
Observe points are observed every shift cycle

Example: chain length of 100 with 10K pattern – observed for 1M cycles

Key benefits – significant reduction in patterns needed to achieve a target test coverage

Observation Scan Cell Structure

Scan Cell



Modes	M1 (=SE)	M2	D	SI	Q
Shift	1	0	d	s	s
Shift & Capture	1	1	d	s	d + s
Capture	0	0	d	s	d
Capture & Accumulate	0	1	d	s	d + q

Has two modes of operation during shift – regular shift or shift_and_capture

Has two modes of operation during capture – functional (mission) or capture_and_accumulate

Results on Micron design used for OST implementation



Design	Technology	# Flops	Chain Length	#TEST POINT	Area(sqmm)	#Test Pattern	Coverage Achieved	Test Time(100MHz)
Design_wo_ost	12nm	~225K	570	2500	~.23	15000	88.54%	85.65ms
Design_w_ost	12nm	~225K	570	2500	~.235	3072	90%	17.541ms
Difference	-	-	-	-	-	~5X reduction	1.5% Better	~5X reduction

Design	Coverage with NCP
w/o_ost	84.85%(15K pat)
w_ost	90%(15K pat)
Difference	5.15% Better

Logic BIST X-tolerance with OST

X tolerance techniques

Static Chain Masking

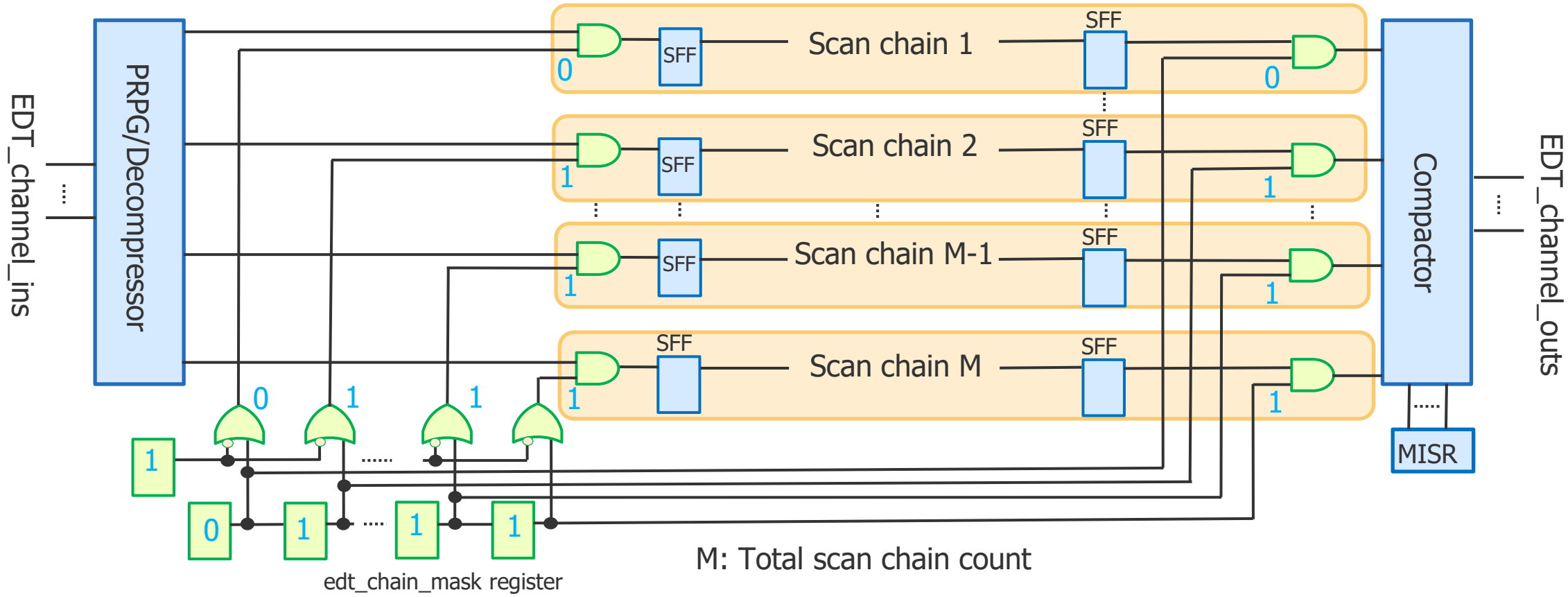
- Mask the entire chain with X sources
- Implementation is straightforward (static chain masking register available with Hybrid TK/LBIST IP)
- Might result in loss of observability when scan chain outputs are masked
 - Compromise between test patterns and coverage
- Higher flexibility requires one bit per chain resulting in higher silicon overhead

Cycle based X-tolerance

- Xs reported are used to generate masking data that contains information on which scan chain outputs should be masked in each clock cycle
- Higher implementation effort
- Additional hardware and computation overhead
- Impact on pattern count to achieve target test coverage

Static chain masking

- In Hybrid TK/LBIST flow – there is a static masking register for masking scan chains
- The register can be loaded dynamically during LBIST setup

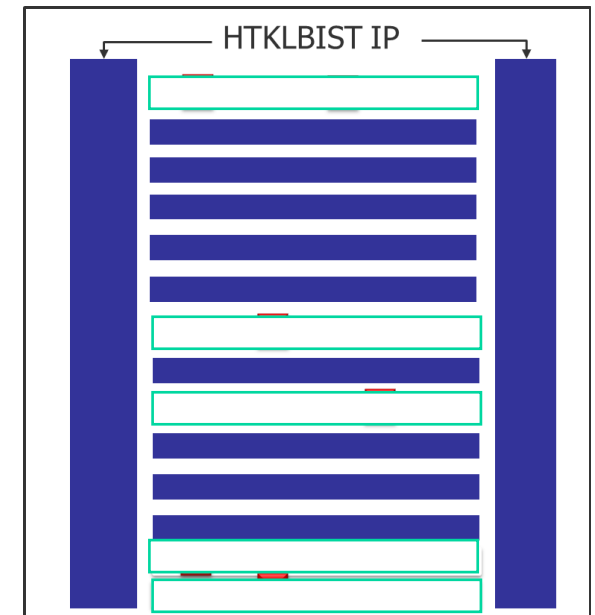


X Tolerant LBIST-OST

When Observation Scan Technology (OST) is enabled, even a single X captured will result in loss of observability for that OST chain

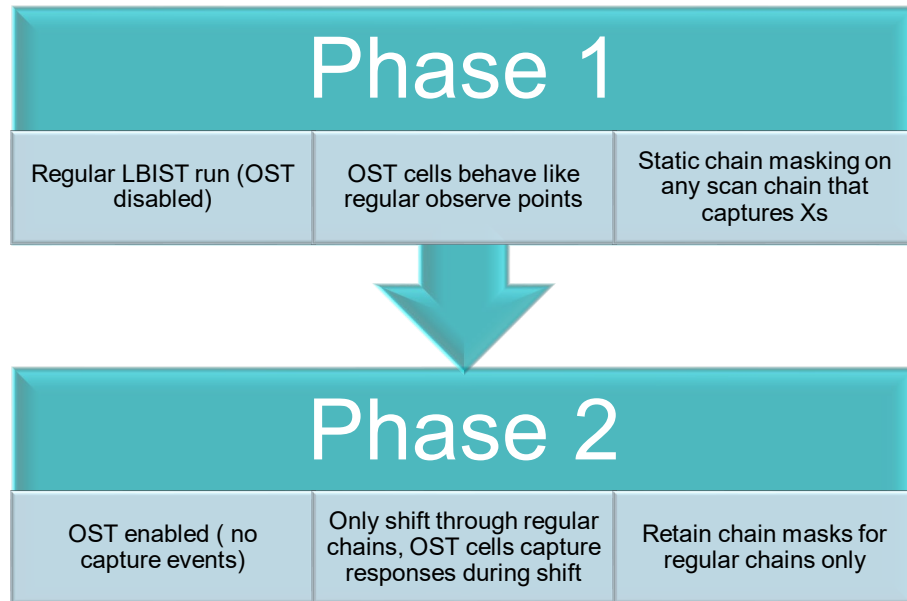
Proposed solution involves two phase fault simulation

- Building on static chain masking
- Exploit the pattern reduction capabilities of OST

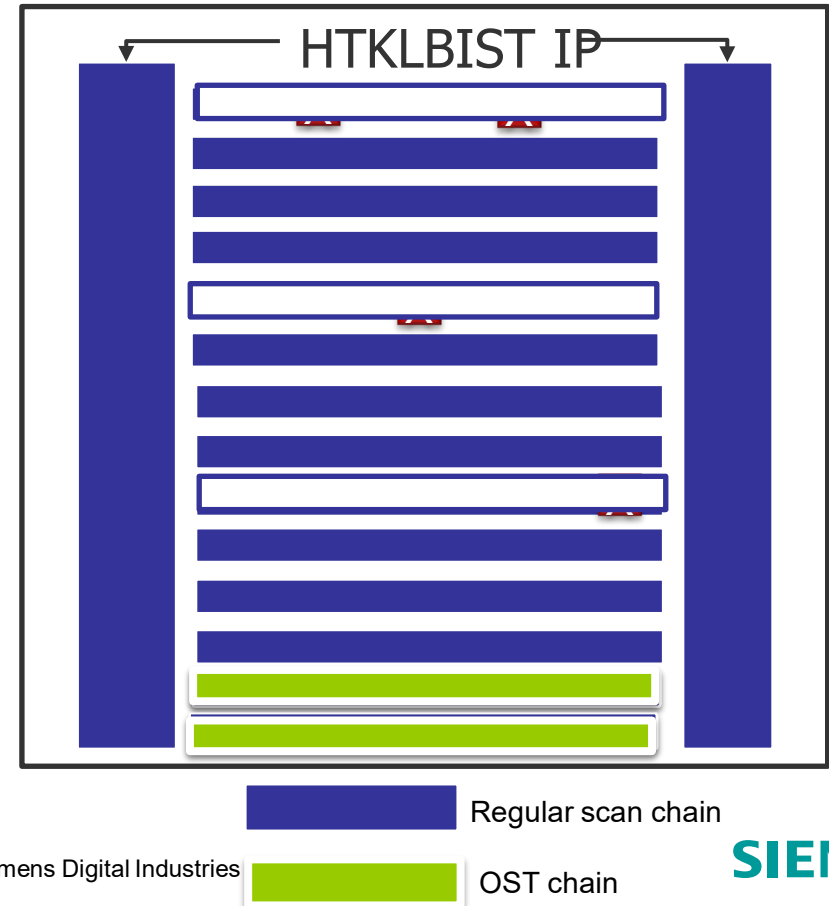


Functionality

When enabled, the solution performs LBIST fault simulation in 2 phases
Optimizes pattern count v/s test coverage and still recover some benefits of using OST



*No Xs are generated during shift



LBIST-OST – Experimental Results

Design	#SCs	#Chains	Chain Length	#Xs	Xs in Reg Chains	Xs in OBS Chains
D1	80,270	1254	65	41	30/1221	32/34
D2	81,546	1255	66	41	30/1216	25/32
D3	151,042	2502	64	74	61/2428	51/74
D4	94,411	528	180	48	5/521	4/7

Design	Pattern Count at 90% Test Coverage					
	Reg BIST	LBIST-OST	PC Red.	LBIST + X- masking	LBIST-OST + X- masking (2-phase)	PC Red. (w.r.t. Reg BIST)
D1	5312	512	10.4X	8,384	1152+1024	2.4X
D2	6912	576	12X	13,248	1344+1216	2.7X
D3	3200	448	7.1X	3,456	512+576	2.9X
D4	7040	4672	1.51X	14,400	7500 + 576	0.87X

Summary

OST two-phase run is designed for supporting X-tolerance for observation scan chains that may capture Xs during shift cycles

Loss of coverage can be minimized using static chain masking in addition to using 2 phase OST fault simulation

User may need to perform a prior fault simulation run with `observation_scan` disabled to collect all the X locations

Thank you