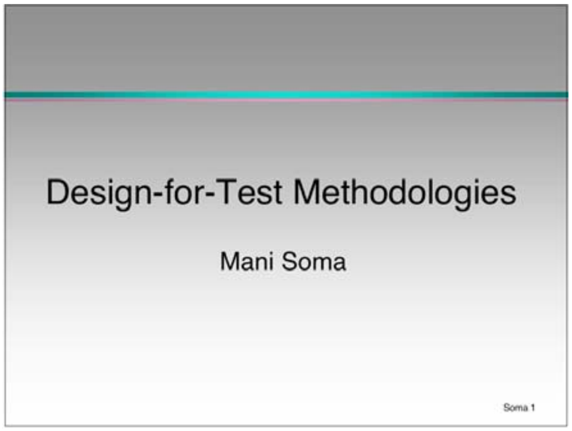
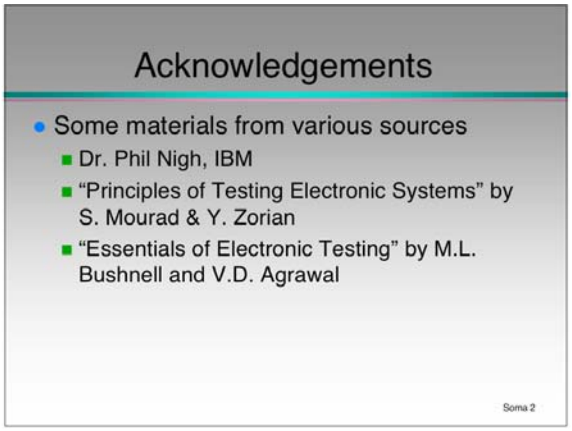


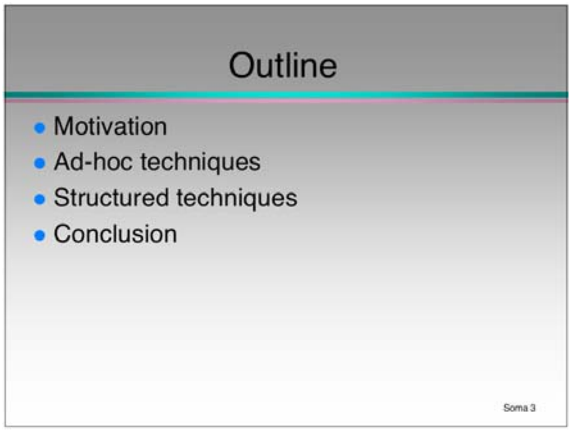
Design-for-Test Methodologies

Mani Soma



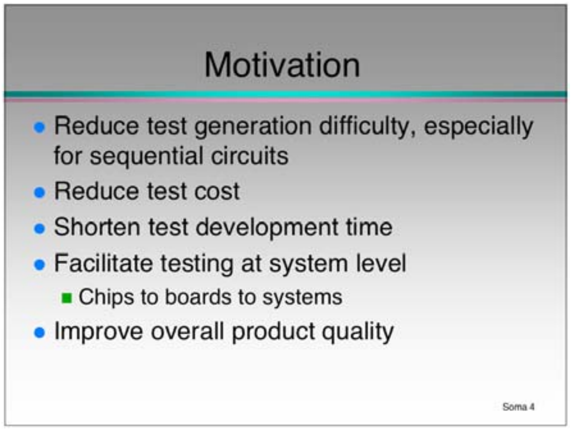
Acknowledgements

- Some materials from various sources
  - Dr. Phil Nigh, IBM
  - “Principles of Testing Electronic Systems” by S. Mourad & Y. Zorian
  - “Essentials of Electronic Testing” by M.L. Bushnell and V.D. Agrawal



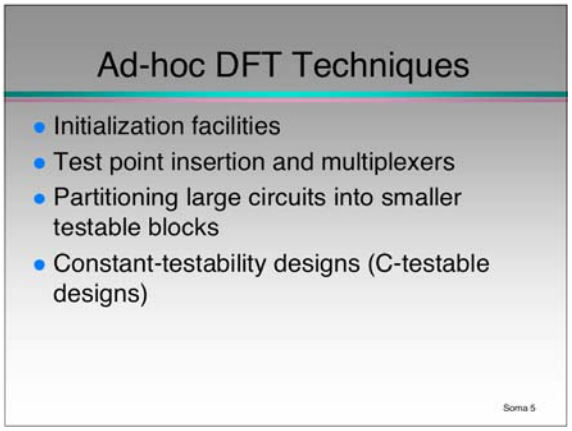
Outline

- Motivation
- Ad-hoc techniques
- Structured techniques
- Conclusion



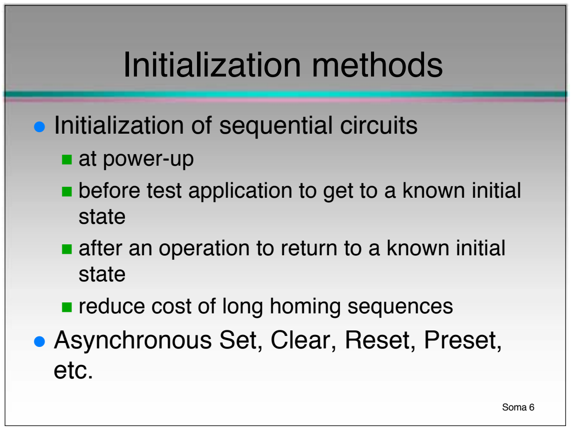
Motivation

- Reduce test generation difficulty, especially for sequential circuits
- Reduce test cost
- Shorten test development time
- Facilitate testing at system level
  - Chips to boards to systems
- Improve overall product quality



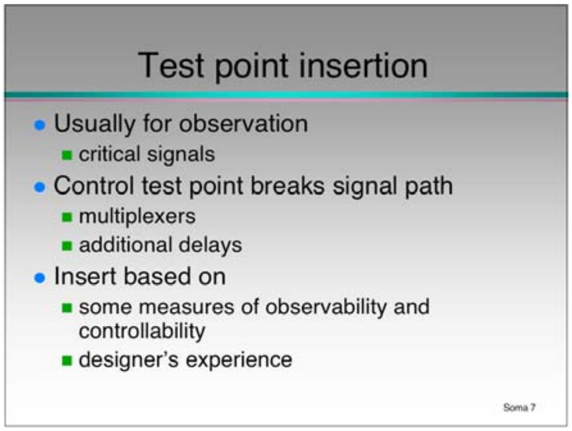
Ad-hoc DFT Techniques

- Initialization facilities
- Test point insertion and multiplexers
- Partitioning large circuits into smaller testable blocks
- Constant-testability designs (C-testable designs)



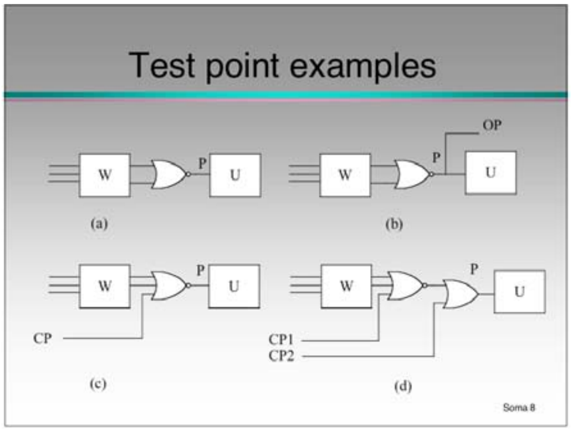
Initialization methods

- Initialization of sequential circuits
  - at power-up
  - before test application to get to a known initial state
  - after an operation to return to a known initial state
  - reduce cost of long homing sequences
- Asynchronous Set, Clear, Reset, Preset, etc.

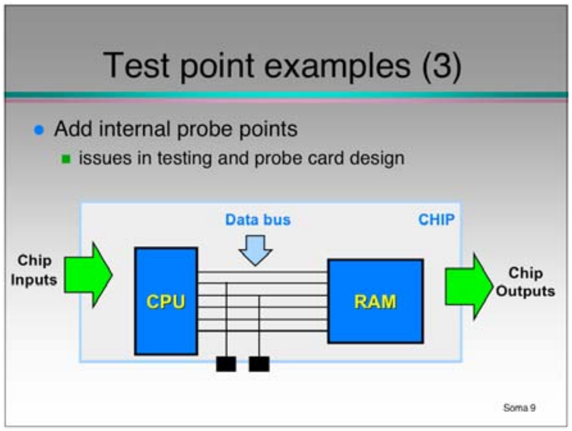


Test point insertion

- Usually for observation
  - critical signals
- Control test point breaks signal path
  - multiplexers
  - additional delays
- Insert based on
  - some measures of observability and controllability
  - designer's experience

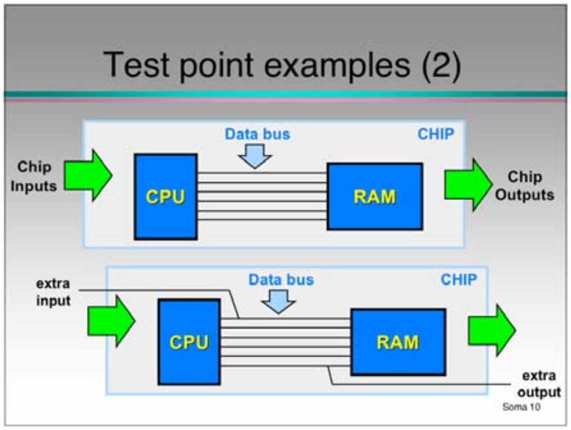


Test point examples

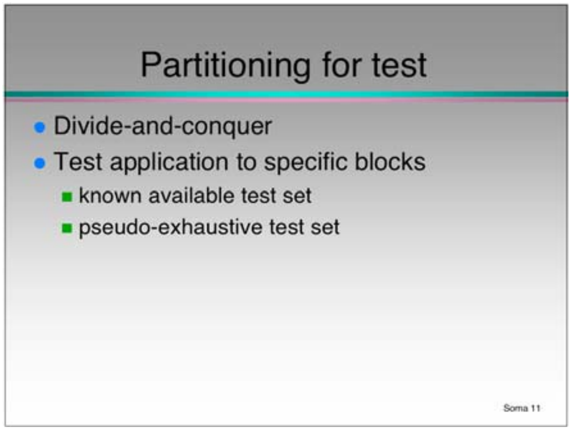


Test point examples (3)

- Add internal probe points
  - issues in testing and probe card design

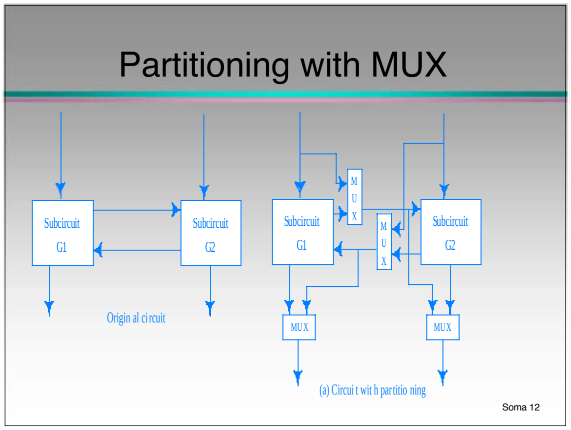


Test point examples (2)

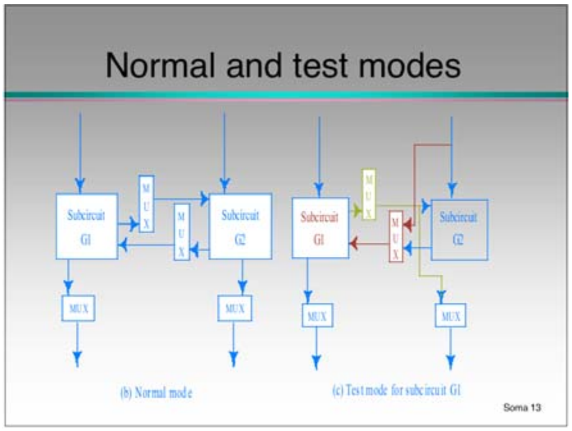


Partitioning for test

- Divide-and-conquer
- Test application to specific blocks
  - known available test set
  - pseudo-exhaustive test set



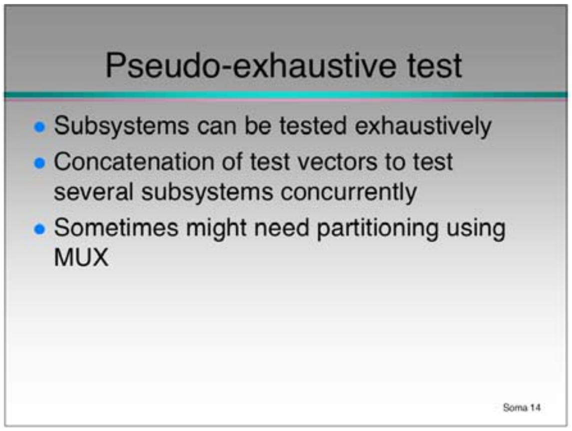
Partitioning with MUX



Normal and test modes

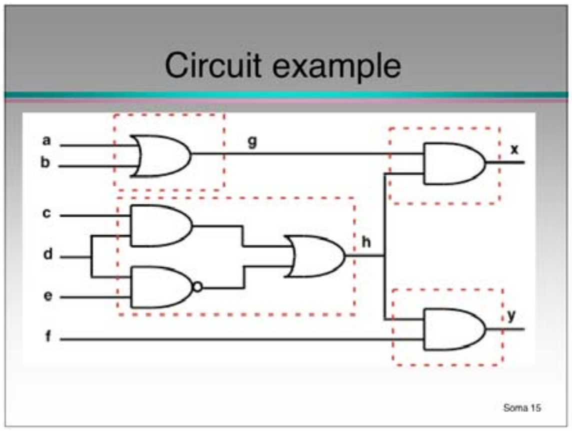
(b) Normal mode

(c) Test mode for subcircuit G1

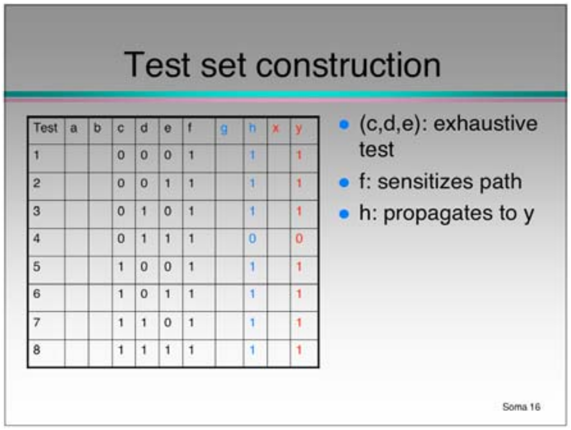


Pseudo-exhaustive test

- Subsystems can be tested exhaustively
- Concatenation of test vectors to test several subsystems concurrently
- Sometimes might need partitioning using MUX



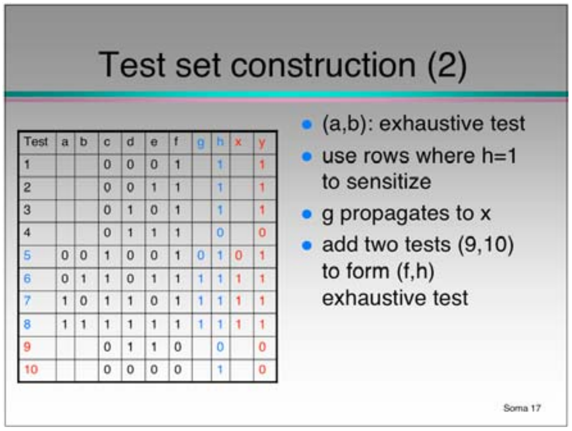
Circuit example



Test set construction

Test	a	b	c	d	e	f	g	h	x	y
1			0	0	0	1		1		1
2			0	0	1	1		1		1
3			0	1	0	1		1		1
4			0	1	1	1		0		0
5			1	0	0	1		1		1
6			1	0	1	1		1		1
7			1	1	0	1		1		1
8			1	1	1	1		1		1

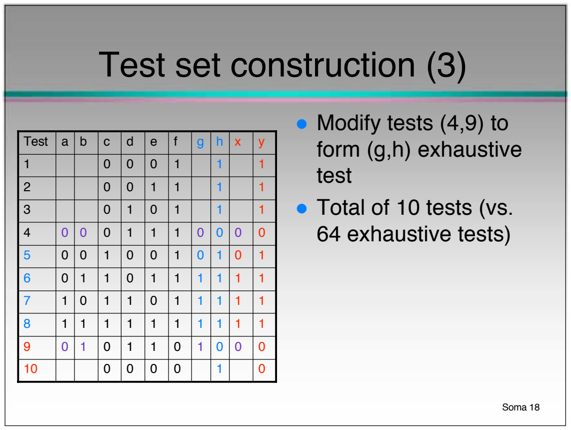
- (c,d,e): exhaustive test
- f: sensitizes path
- h: propagates to y



Test set construction (2)

Test	a	b	c	d	e	f	g	h	x	y
1			0	0	0	1		1		1
2			0	0	1	1		1		1
3			0	1	0	1		1		1
4			0	1	1	1		0		0
5	0	0	1	0	0	1	0	1	0	1
6	0	1	1	0	1	1	1	1	1	1
7	1	0	1	1	0	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1
9			0	1	1	0		0		0
10			0	0	0	0		1		0

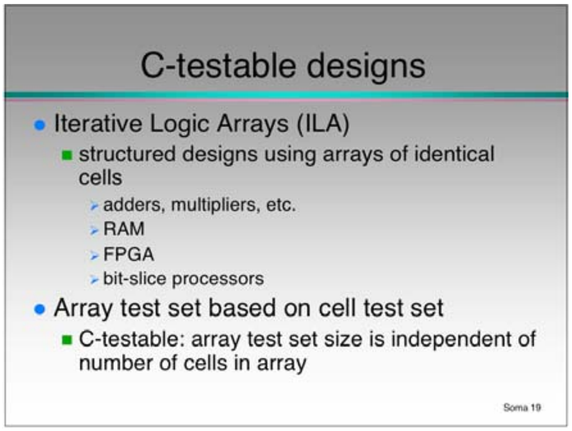
- (a,b): exhaustive test
- use rows where h=1 to sensitize
- g propagates to x
- add two tests (9,10) to form (f,h) exhaustive test



Test set construction (3)

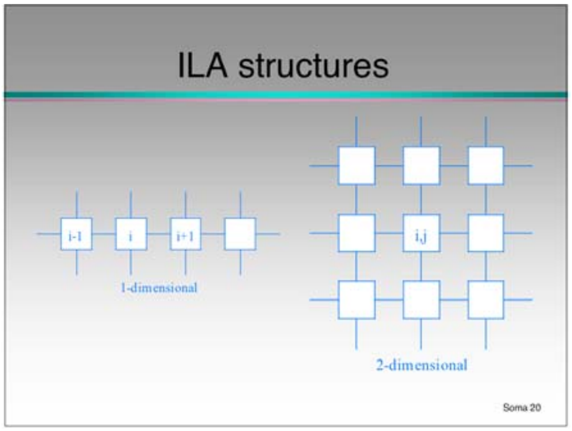
Test	a	b	c	d	e	f	g	h	x	y
1			0	0	0	1		1		1
2			0	0	1	1		1		1
3			0	1	0	1		1		1
4	0	0	0	1	1	1	0	0	0	0
5	0	0	1	0	0	1	0	1	0	1
6	0	1	1	0	1	1	1	1	1	1
7	1	0	1	1	0	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1
9	0	1	0	1	1	0	1	0	0	0
10			0	0	0	0		1		0

- Modify tests (4,9) to form (g,h) exhaustive test
- Total of 10 tests (vs. 64 exhaustive tests)

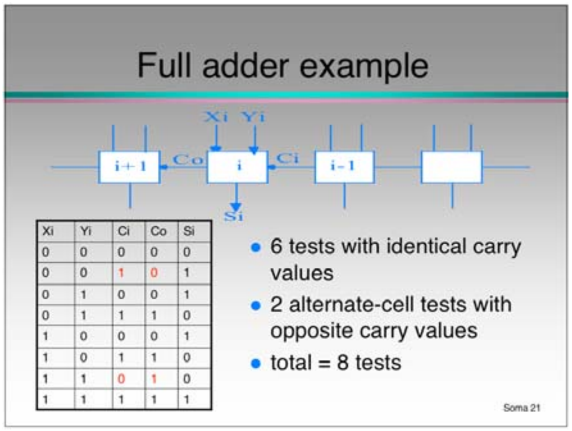


C-testable designs

- Iterative Logic Arrays (ILA)
  - structured designs using arrays of identical cells
    - adders, multipliers, etc.
    - RAM
    - FPGA
    - bit-slice processors
- Array test set based on cell test set
  - C-testable: array test set size is independent of number of cells in array



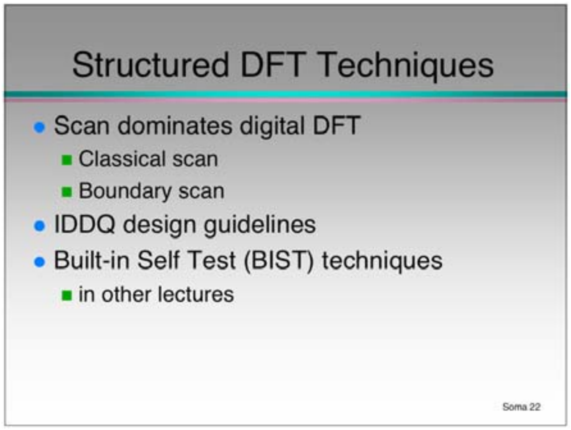
ILA structures



Full adder example

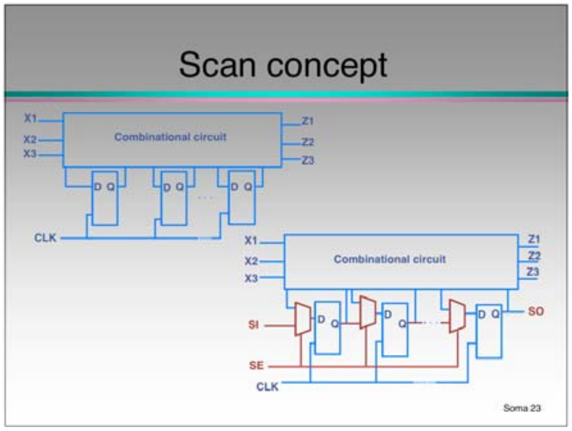
Xi	Yi	Ci	Co	Si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

- 6 tests with identical carry values
- 2 alternate-cell tests with opposite carry values
- total = 8 tests

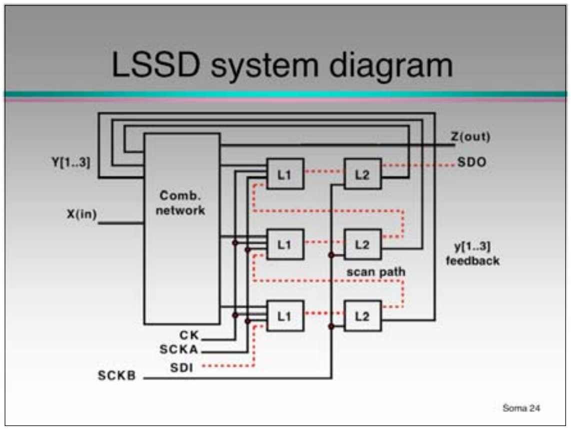


Structured DFT Techniques

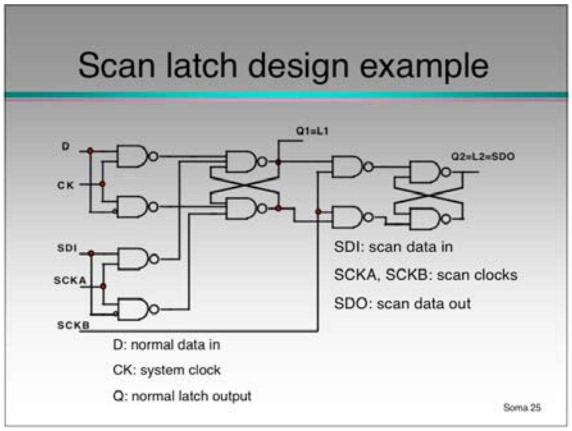
- Scan dominates digital DFT
  - Classical scan
  - Boundary scan
- IDDQ design guidelines
- Built-in Self Test (BIST) techniques
  - in other lectures



Scan concept



LSSD system diagram



Scan latch design example

SDI: scan data in

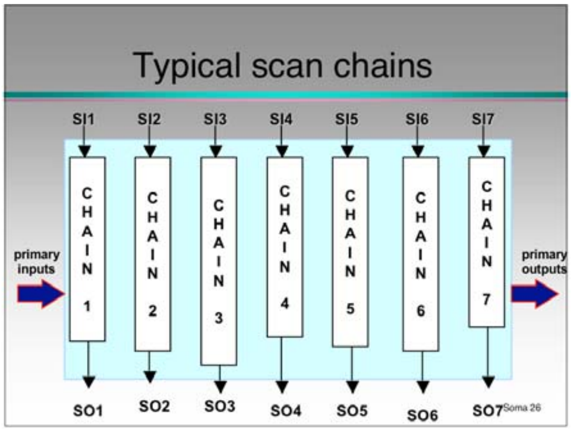
SCKA, SCKB: scan clocks

SDO: scan data out

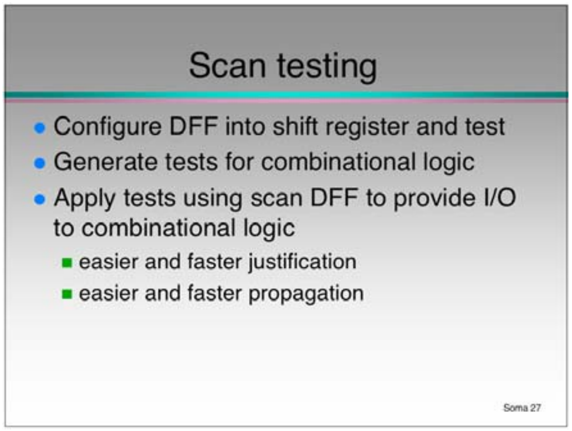
D: normal data in

CK: system clock

Q: normal latch output

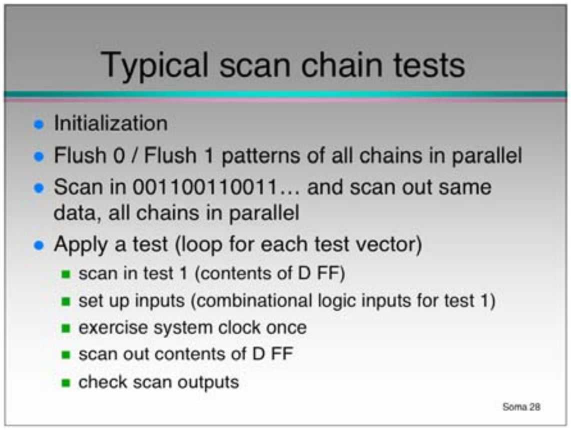


Typical scan chains



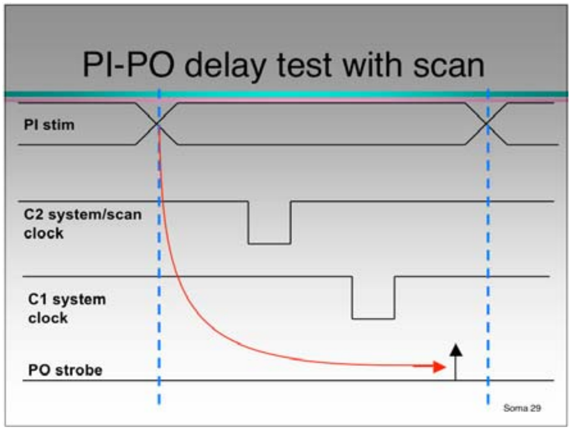
Scan testing

- Configure DFF into shift register and test
- Generate tests for combinational logic
- Apply tests using scan DFF to provide I/O to combinational logic
  - easier and faster justification
  - easier and faster propagation

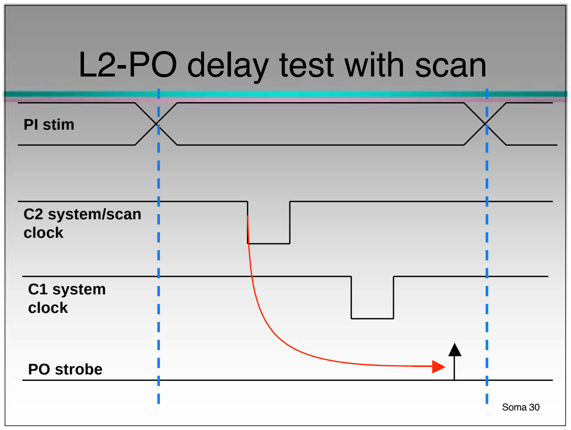


Typical scan chain tests

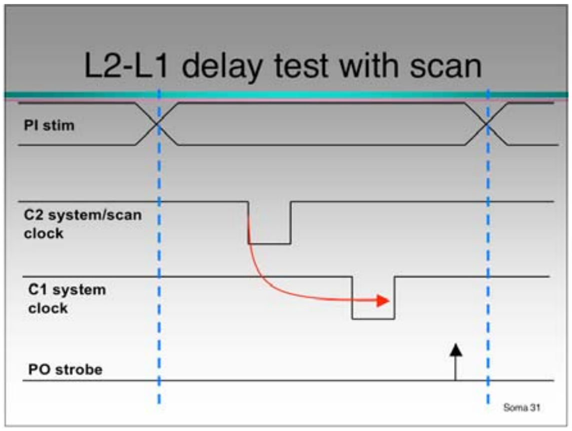
- Initialization
- Flush 0 / Flush 1 patterns of all chains in parallel
- Scan in 001100110011… and scan out same data, all chains in parallel
- Apply a test (loop for each test vector)
  - scan in test 1 (contents of D FF)
  - set up inputs (combinational logic inputs for test 1)
  - exercise system clock once
  - scan out contents of D FF
  - check scan outputs



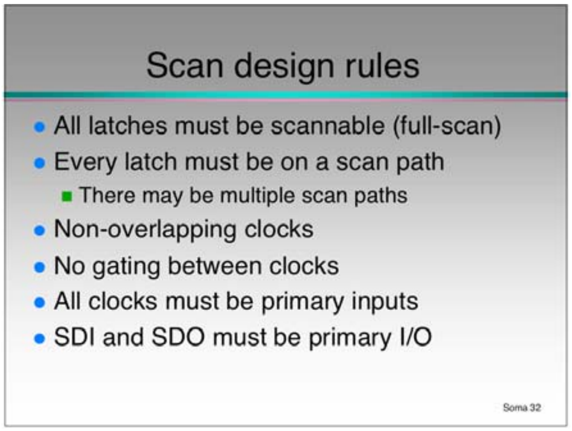
PI-PO delay test with scan



L2-PO delay test with scan

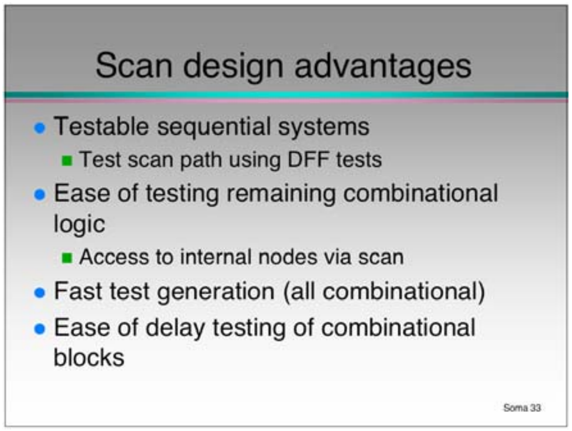


L2-L1 delay test with scan



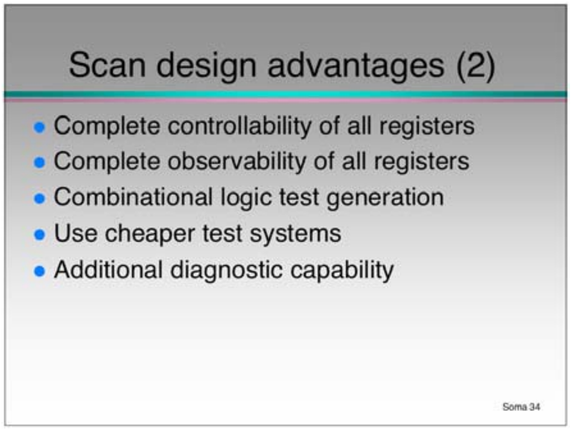
Scan design rules

- All latches must be scannable (full-scan)
- Every latch must be on a scan path
  - There may be multiple scan paths
- Non-overlapping clocks
- No gating between clocks
- All clocks must be primary inputs
- SDI and SDO must be primary I/O



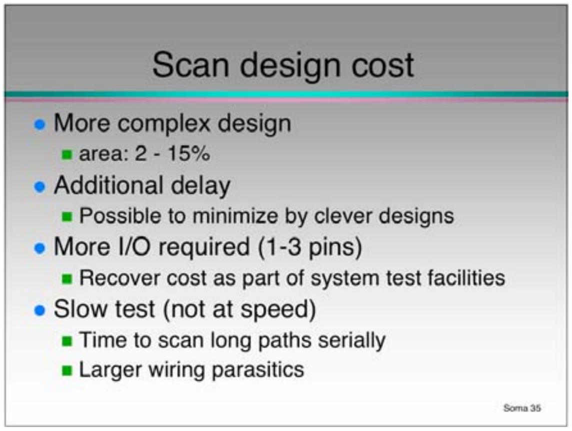
Scan design advantages

- Testable sequential systems
  - Test scan path using DFF tests
- Ease of testing remaining combinational logic
  - Access to internal nodes via scan
- Fast test generation (all combinational)
- Ease of delay testing of combinational blocks



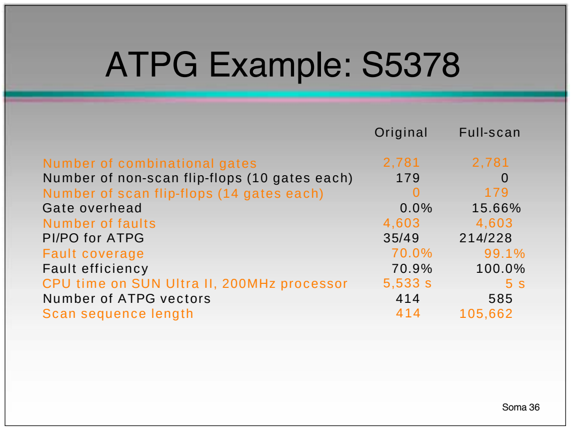
Scan design advantages (2)

- Complete controllability of all registers
- Complete observability of all registers
- Combinational logic test generation
- Use cheaper test systems
- Additional diagnostic capability



Scan design cost

- More complex design
  - area: 2 - 15%
- Additional delay
  - Possible to minimize by clever designs
- More I/O required (1-3 pins)
  - Recover cost as part of system test facilities
- Slow test (not at speed)
  - Time to scan long paths serially
  - Larger wiring parasitics



ATPG Example: S5378

	Original	Full-scan
Number of combinational gates	2,781	2,781
Number of non-scan flip-flops (10 gates each)	179	0
Number of scan flip-flops (14 gates each)	0	179
Gate overhead	0.0%	15.66%
Number of faults	4,603	4,603
PI/PO for ATPG	35/49	214/228
Fault coverage	70.0%	99.1%
Fault efficiency	70.9%	100.0%
CPU time on SUN Ultra II, 200MHz processor	5,533 s	5 s
Number of ATPG vectors	414	585
Scan sequence length	414	105,662

Scan varieties

- Full-scan: increasing use
  - partitioning issues in multiple scan chains
  - optimal placement of DFF in scan chain
- Partial-scan
- Random-access scan
  - Individual latch addressable
  - High overhead
- Level-Sensitive Scan Design
  - No races and timing problems

Partial Scan Example

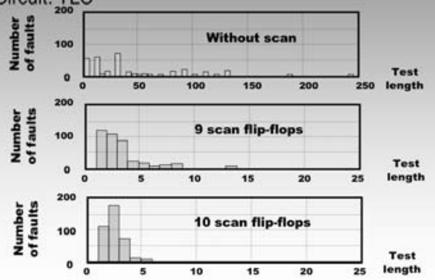
- Circuit: TLC, 355 gates, 21 flip-flops

Scan flip-flops	Max. cycle length	Depth*	ATPG CPU s	Fault sim. CPU s	Fault cov.	ATPG vectors	Test seq. length
0	4	14	1,247	61	89.01%	805	805
4	2	10	157	11	95.90%	247	1,249
9	1	5	32	4	99.20%	136	1,382
10	1	3	13	4	100.00%	112	1,256
21	0	0	2	2	100.00%	52	1,190

* Cyclic paths ignored

Test Length Statistics

- Circuit: TLC



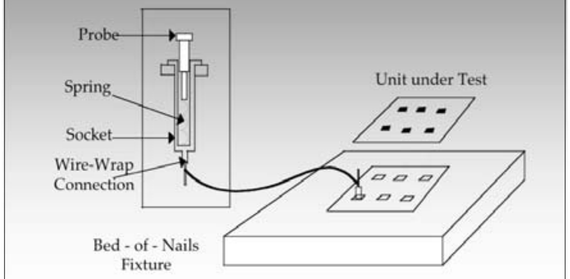
Partial vs. Full Scan: S5378

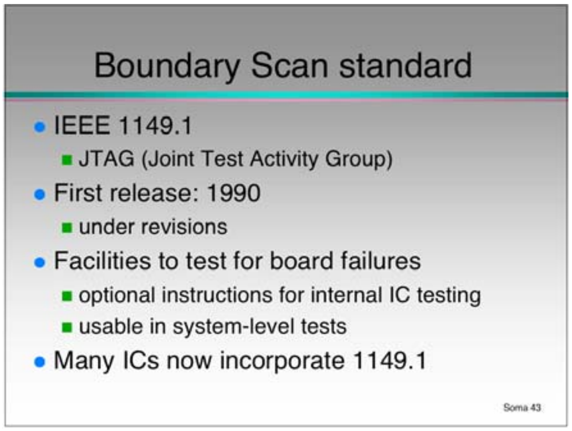
	Original	Partial-scan	Full-scan
Number of combinational gates	2,781	2,781	2,781
Number of non-scan flip-flops (10 gates each)	179	149	0
Number of scan flip-flops (14 gates each)	0	30	179
Gate overhead	0.0%	2.63%	15.66%
Number of faults	4,603	4,603	4,603
PI/PO for ATPG	35/49	65/79	214/228
Fault coverage	70.0%	93.7%	99.1%
Fault efficiency	70.9%	99.5%	100.0%
CPU time on SUN Ultra II 200MHz processor	5,533 s	727 s	5 s
Number of ATPG vectors	414	1,117	585
Scan sequence length	414	34,691	105,662

Motivation for board test

- High-density boards / Multichip Modules (MCM) not testable using bed-of-nails
- Board failure modes:
  - stuck-at faults
  - open
  - shorts
  - incorrect components
- Lack of test access

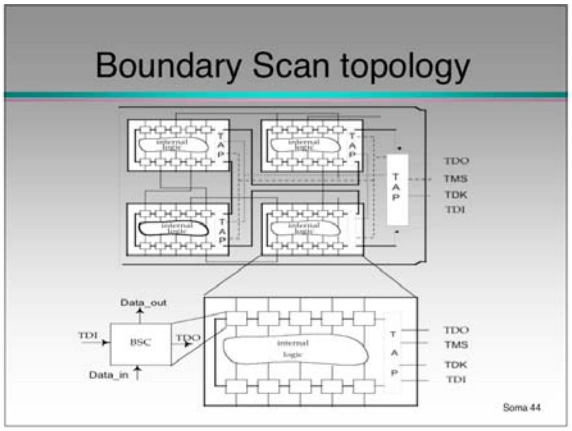
Bed-of-nails testing



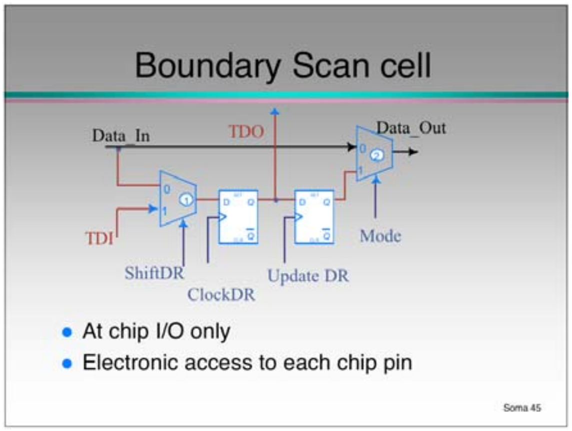


Boundary Scan standard

- IEEE 1149.1
  - JTAG (Joint Test Activity Group)
- First release: 1990
  - under revisions
- Facilities to test for board failures
  - optional instructions for internal IC testing
  - usable in system-level tests
- Many ICs now incorporate 1149.1

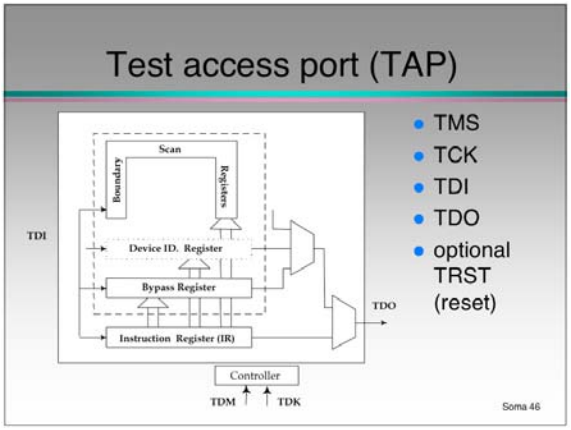


Boundary Scan topology



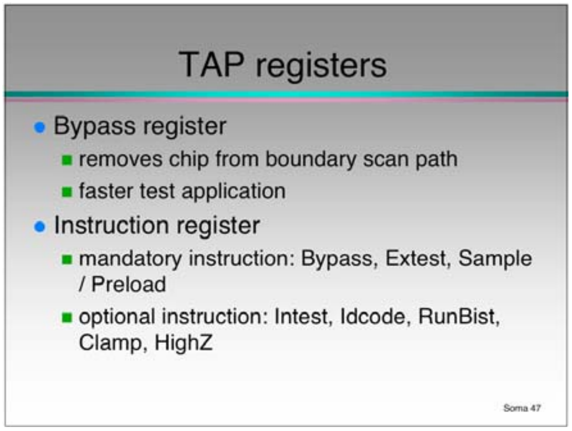
Boundary Scan cell

- At chip I/O only
- Electronic access to each chip pin



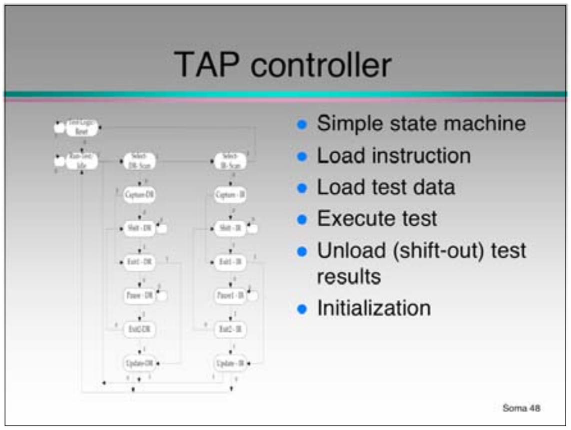
Test access port (TAP)

- TMS
- TCK
- TDI
- TDO
- optional TRST (reset)



TAP registers

- Bypass register
  - removes chip from boundary scan path
  - faster test application
- Instruction register
  - mandatory instruction: Bypass, Extest, Sample / Preload
  - optional instruction: Intest, Idcode, RunBist, Clamp, HighZ



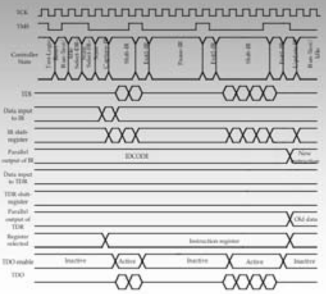
TAP controller

- Simple state machine
- Load instruction
- Load test data
- Execute test
- Unload (shift-out) test results
- Initialization

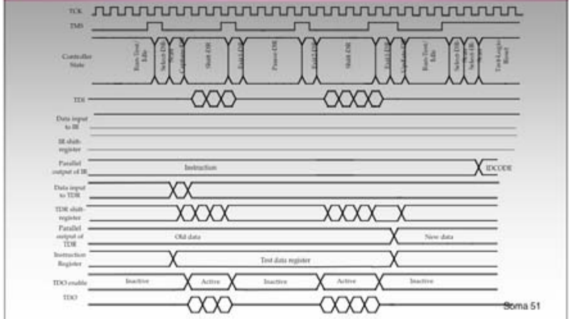
Register operations

- Instruction registers
  - loaded serially for all ICs
  - captured in parallel to each IC
- Data registers at chip pins
  - loaded serially, captured in parallel
  - test data and test results

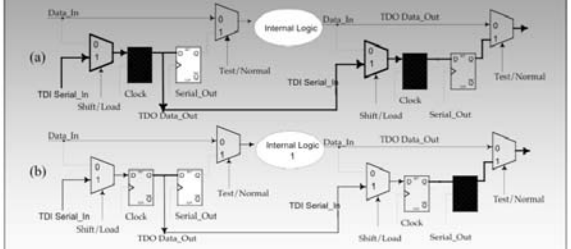
Instruction scan



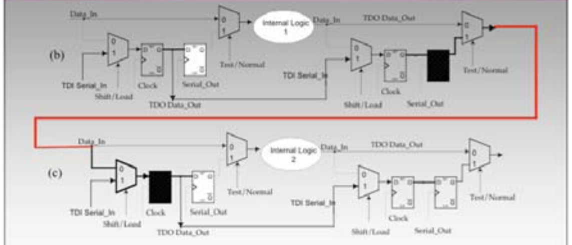
Data scan



EXTEST example (1)

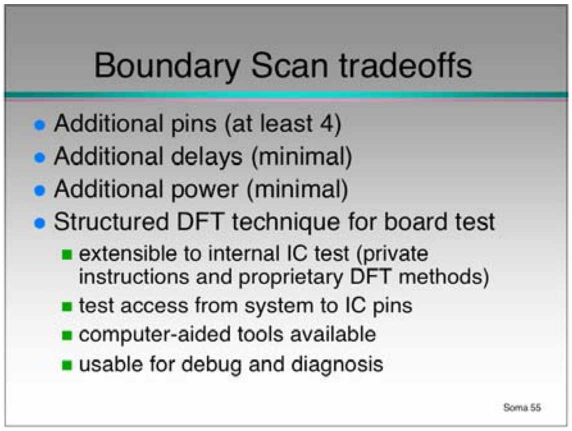


EXTEST example (2)



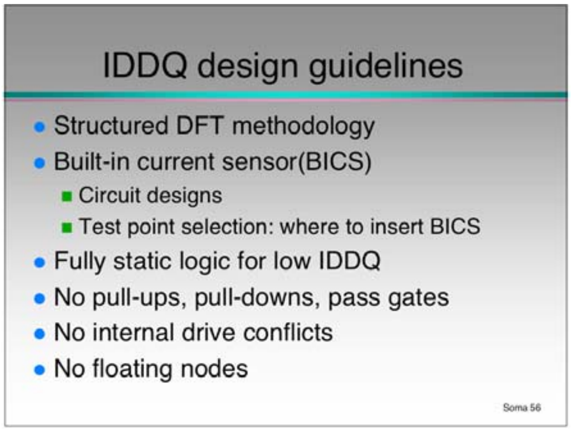
Boundary Scan design tools

- Boundary Scan Design Language
  - BSDL
  - facilitates incorporation and checking of correct BS designs
- Automatic synthesis of TAP controller
- Tools and automatic test generation available in most systems
  - supported by numerous testers



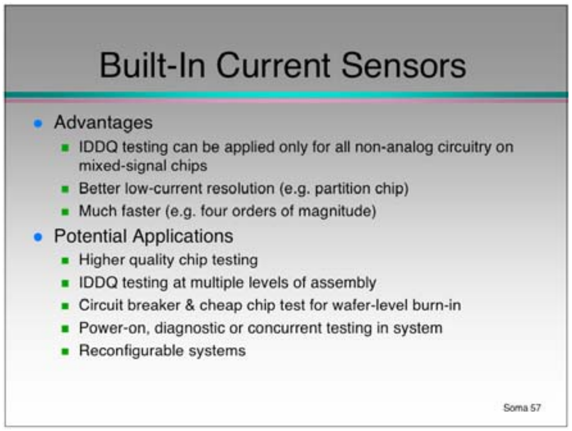
Boundary Scan tradeoffs

- Additional pins (at least 4)
- Additional delays (minimal)
- Additional power (minimal)
- Structured DFT technique for board test
  - extensible to internal IC test (private instructions and proprietary DFT methods)
  - test access from system to IC pins
  - computer-aided tools available
  - usable for debug and diagnosis



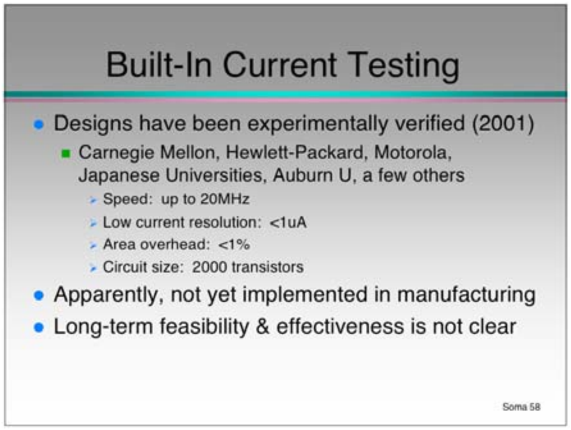
IDDQ design guidelines

- Structured DFT methodology
- Built-in current sensor(BICS)
  - Circuit designs
  - Test point selection: where to insert BICS
- Fully static logic for low IDDQ
- No pull-ups, pull-downs, pass gates
- No internal drive conflicts
- No floating nodes



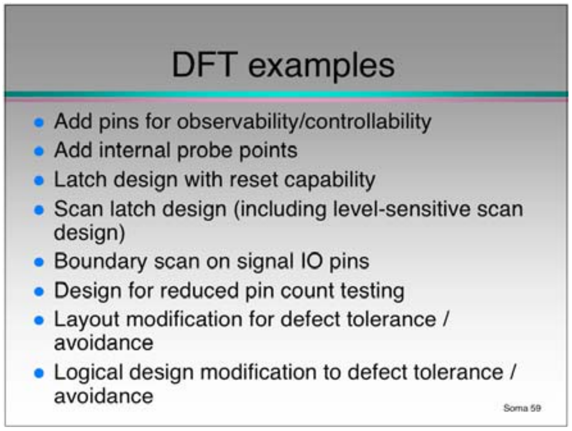
Built-In Current Sensors

- Advantages
  - IDDQ testing can be applied only for all non-analog circuitry on mixed-signal chips
  - Better low-current resolution (e.g. partition chip)
  - Much faster (e.g. four orders of magnitude)
- Potential Applications
  - Higher quality chip testing
  - IDDQ testing at multiple levels of assembly
  - Circuit breaker & cheap chip test for wafer-level burn-in
  - Power-on, diagnostic or concurrent testing in system
  - Reconfigurable systems



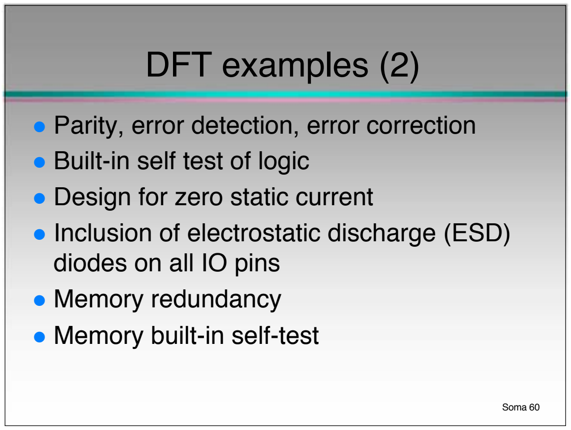
Built-In Current Testing

- Designs have been experimentally verified (2001)
  - Carnegie Mellon, Hewlett-Packard, Motorola, Japanese Universities, Auburn U, a few others
    - Speed: up to 20MHz
    - Low current resolution: <1uA
    - Area overhead: <1%
    - Circuit size: 2000 transistors
- Apparently, not yet implemented in manufacturing
- Long-term feasibility & effectiveness is not clear



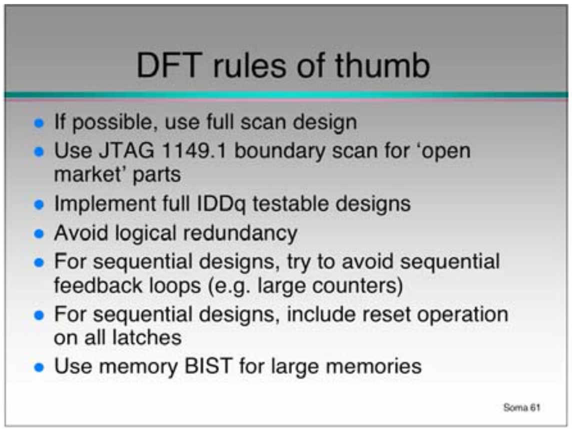
DFT examples

- Add pins for observability/controllability
- Add internal probe points
- Latch design with reset capability
- Scan latch design (including level-sensitive scan design)
- Boundary scan on signal IO pins
- Design for reduced pin count testing
- Layout modification for defect tolerance / avoidance
- Logical design modification to defect tolerance / avoidance



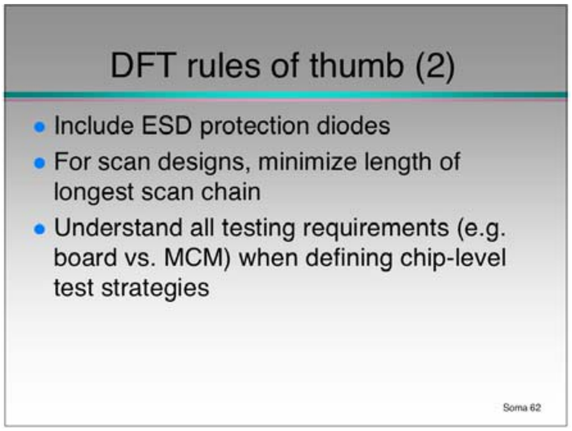
DFT examples (2)

- Parity, error detection, error correction
- Built-in self test of logic
- Design for zero static current
- Inclusion of electrostatic discharge (ESD) diodes on all IO pins
- Memory redundancy
- Memory built-in self-test



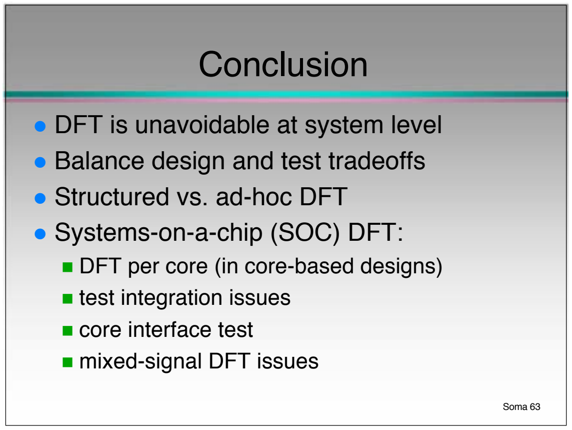
DFT rules of thumb

- If possible, use full scan design
- Use JTAG 1149.1 boundary scan for ‘open market’ parts
- Implement full IDDq testable designs
- Avoid logical redundancy
- For sequential designs, try to avoid sequential feedback loops (e.g. large counters)
- For sequential designs, include reset operation on all latches
- Use memory BIST for large memories



DFT rules of thumb (2)

- Include ESD protection diodes
- For scan designs, minimize length of longest scan chain
- Understand all testing requirements (e.g. board vs. MCM) when defining chip-level test strategies



Conclusion

- DFT is unavoidable at system level
- Balance design and test tradeoffs
- Structured vs. ad-hoc DFT
- Systems-on-a-chip (SOC) DFT:
  - DFT per core (in core-based designs)
  - test integration issues
  - core interface test
  - mixed-signal DFT issues