What can we expect from BIST?

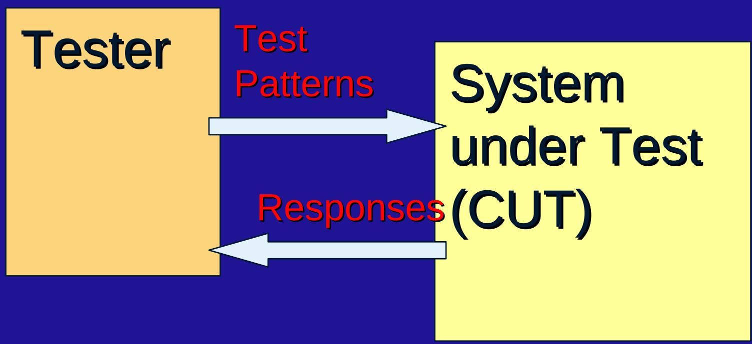
Ondřej Novák
Technická univerzita v Liberci
Česká republika

Outline

1. BIST structures
2. Random logic test pattern generators
   - Pseudoexhaustive test pattern generators
   - Pseudorandom test pattern generators
   - Mixed mode testing methods
   - Test Pattern Decompression Automata effectiveness
3. Memory BIST
4. Signature analysis
5. Conclusion

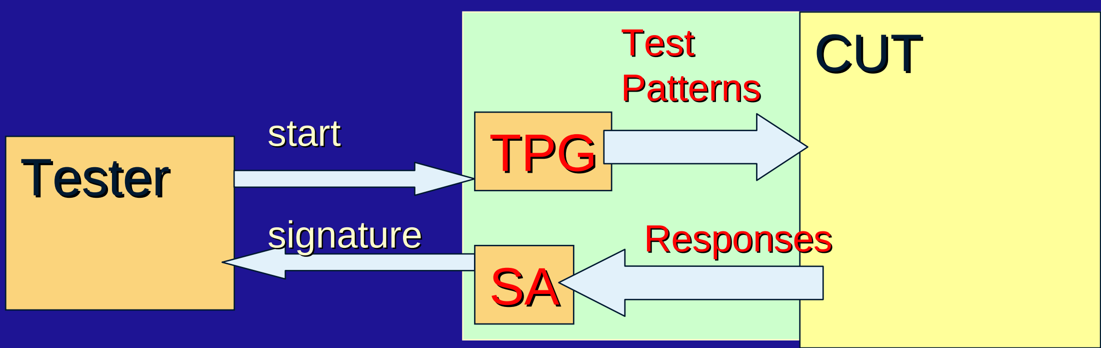
Digital Circuit Testing Problem

Classical diagnostic approach



Problem: hardware for diagnostic access, time

Built-In Test (BIST)



BIST has to be supported by DFT

- ❑ Partial scan – possible solution, not very often, multiple diagnostic approaches are needed
- ❑ Scan design – usual platform for BIST
- ❑ BILBO – enables TPC testing, big hardware overhead, big delays in functional paths
- ❑ Boundary Scan – enables us to incorporate TPGs and SAs, often used

BIST Simplifies Testing

- ❑ Simple Tester (lower memory, easier controlling and signature evaluation)
- ❑ lower amount of transferred data (usual request from silicon foundry)
- ❑ faster testing (higher system availability)
- ❑ disadvantage: higher IC area devoted to diagnostics

Two basical diagnostic approaches

- Test-Per-Clock (TPC) testing
  - faster testing
  - complicated hardware for catching responses
  - problems with effectivity of test patterns
- Test-Per-Scan (TPS) testing
  - shifting each new pattern through the chain
  - less hardware consuming
  - more time and energy consuming
  - cannot be performed at speed

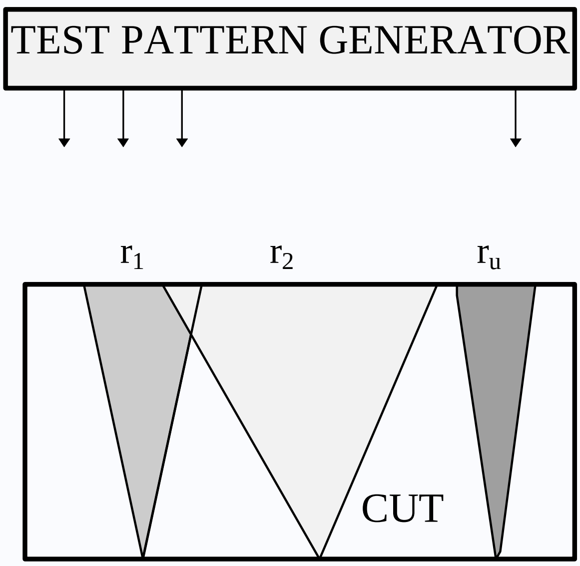
Test Patern Types Used in BIST

- deterministic patterns – stored in a special memory, possibility of pattern compression
- generated patterns – pseudoexhaustive, pseudorandom, deterministic
- mixed patterns – output sequence of an special decompressing automaton is modified with the help of stored patterns

Pseudoexhaustive Test Sets

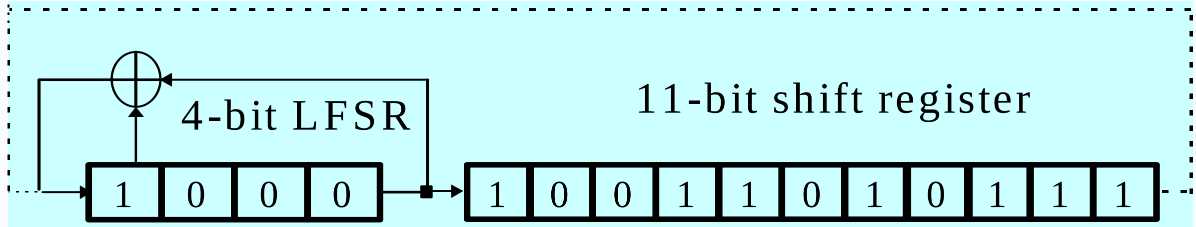
- 100 % probability of detecting considered faults
- relatively long test sequences
- for large systems not practical

(n,r) exhaustive testing



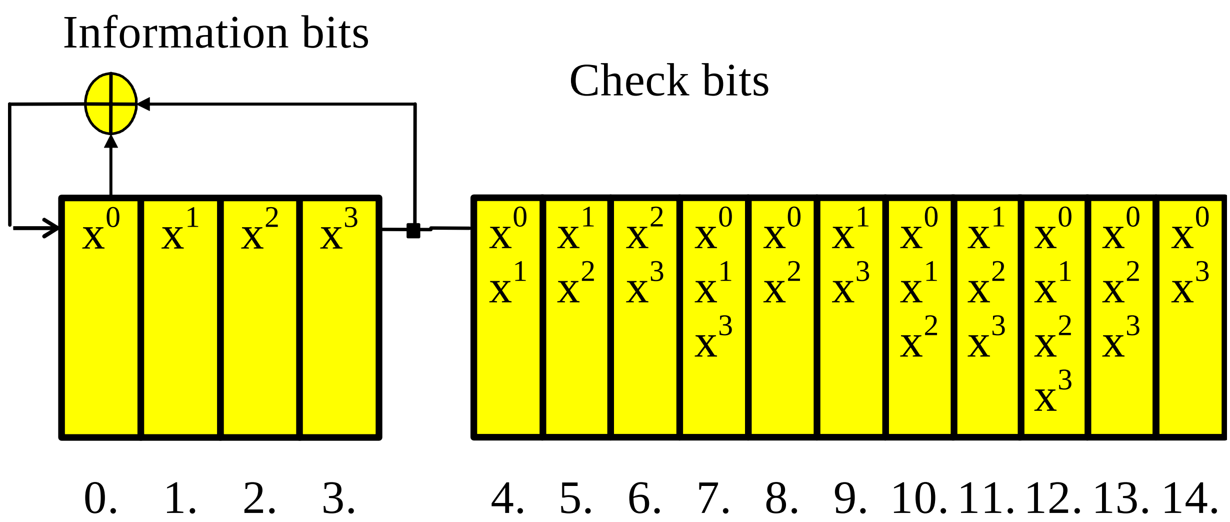
$r_i \leq r, \ i = 1, 2, 3, \ldots u$

fault coverage ≈ 100%



1	1	0	0	0	1	0	0	1	1	0	1	0	1	1
1	1	1	0	0	0	1	0	0	1	1	0	1	0	1
1	1	1	1	0	0	0	1	0	0	1	1	0	1	0
0	1	1	1	1	0	0	0	1	0	0	1	1	0	1
1	0	1	1	1	1	0	0	0	1	0	0	1	1	0
0	1	0	1	1	1	1	0	0	0	1	0	0	1	1
1	0	1	0	1	1	1	1	0	0	0	1	0	0	1
1	1	0	1	0	1	1	1	1	0	0	0	1	0	0
0	1	1	0	1	0	1	1	1	1	0	0	0	1	0
0	0	1	1	0	1	0	1	1	1	1	0	0	0	1
1	0	0	1	1	0	1	0	1	1	1	1	0	0	0
0	1	0	0	1	1	0	1	0	1	1	1	1	0	0
0	0	1	0	0	1	1	0	1	0	1	1	1	1	0
0	0	0	1	0	0	1	1	0	1	0	1	1	1	1

Linear Feedback Shift Register (LFSR)

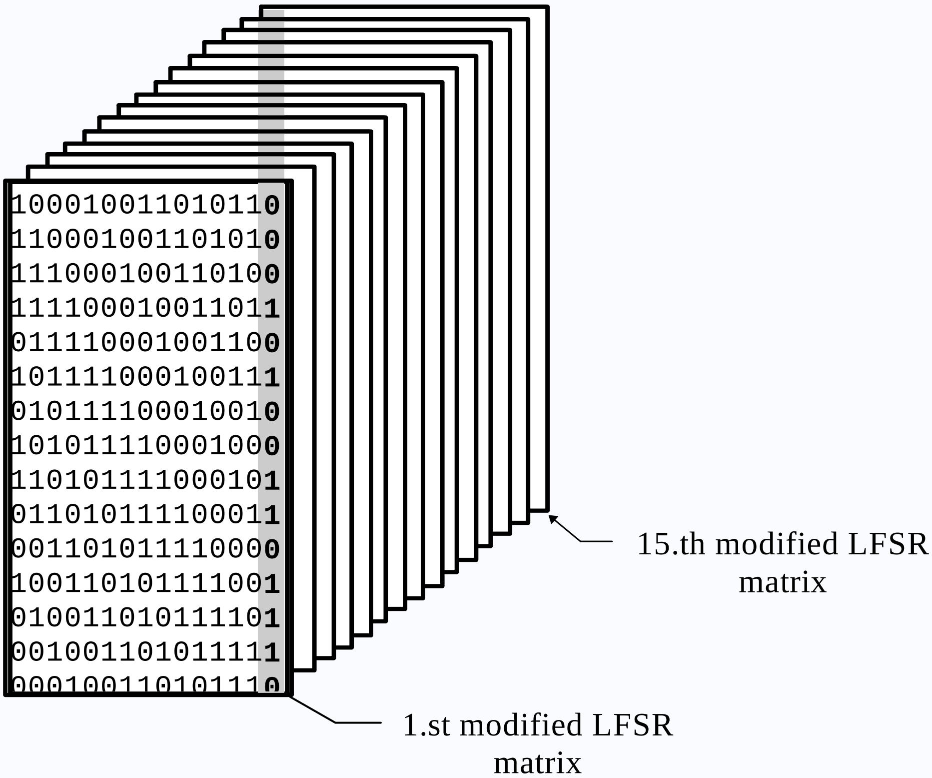


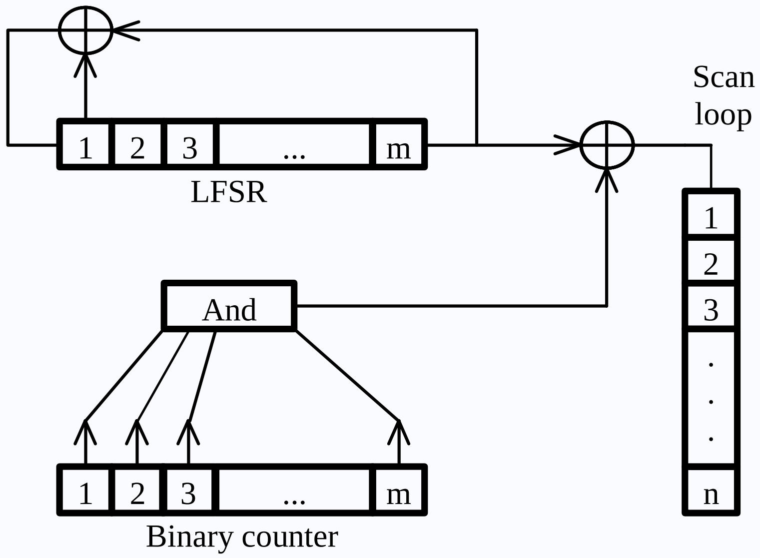
LFSR behavior example

Theorem:

The (n,r)-exhaustive test set with r < 3(w+1)/2 + 1 can be obtained by concatenation of n different code matrices of size n x n where each code matrix is derived from the matrix of the LFSR code by inverting one column.

Example: (15,5)-exhaustive test set, 225 t.



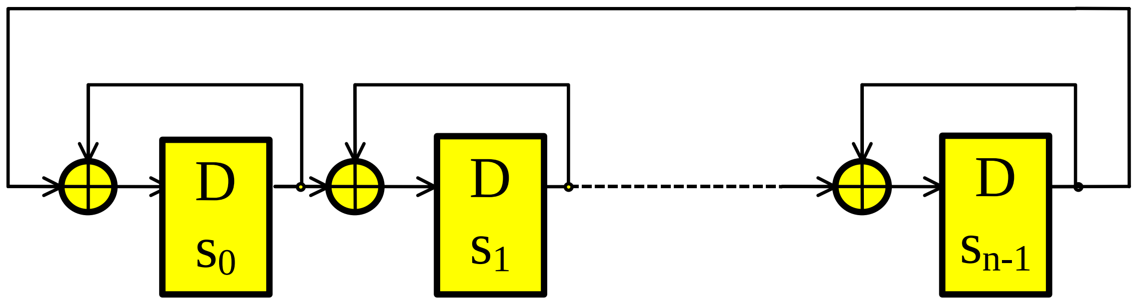


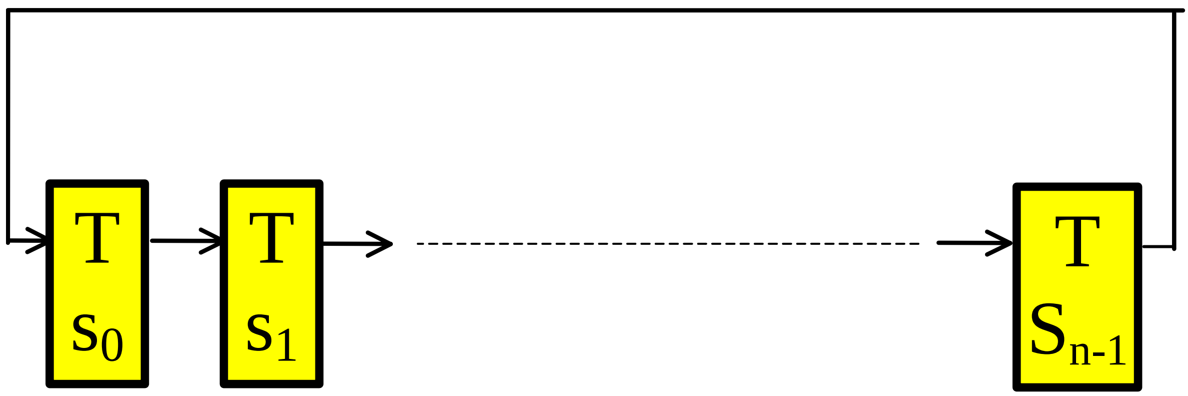
Best results:

Test type (n,r):	(15,5)	(15,6)	(15,8)	(31,5)	(63,5)	(225,5)
Test length:	225	480	3840	961	3969	65 010

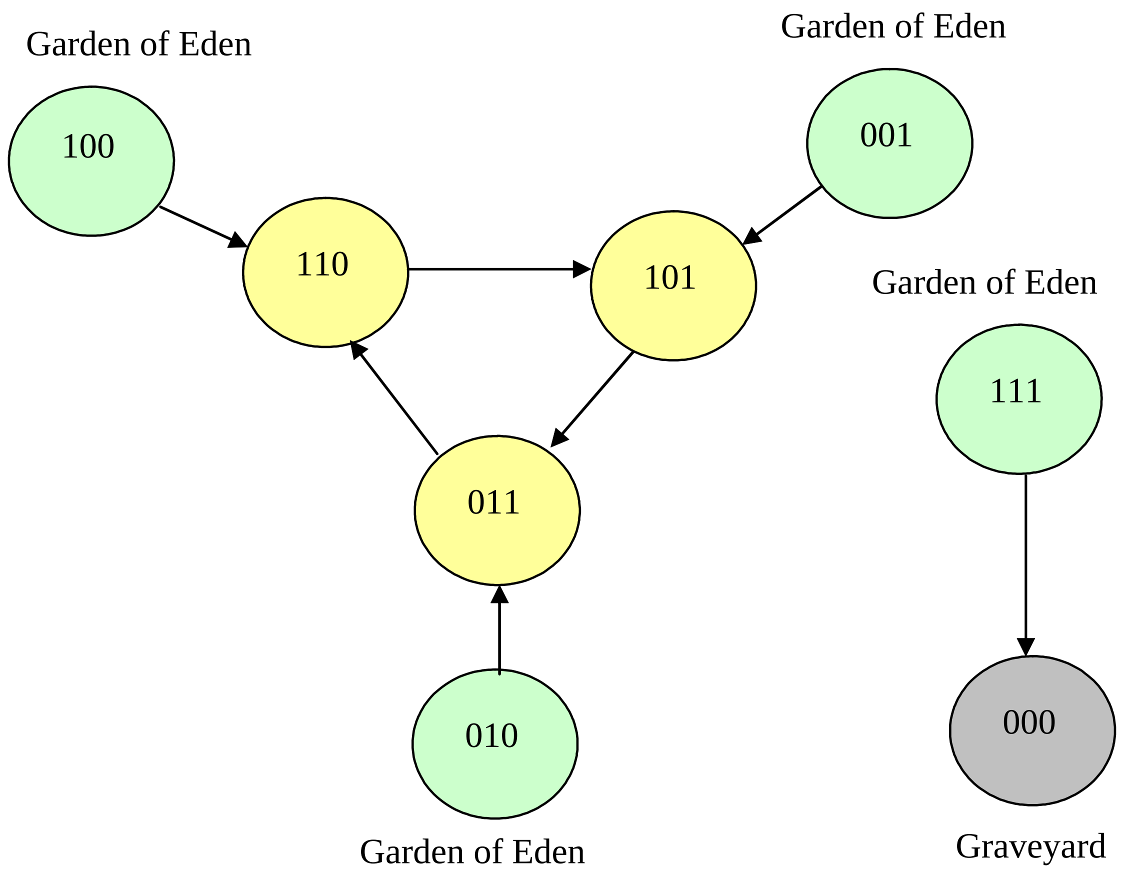
NOVAK, O. - HLAVICKA, J.: Enhancing fault coverage of pseudoexhaustive test sets. Proceedings of 1st IEEE International on-line testing workshop, Nice, France, July 4-6, 1995

Cellular Automata (CA)





CA behavior example



Comparison of LFSR and CA ability of (n,r) exhaustive test sets creating

demonstration of LFSR test pattern quality

demonstration of CA test pattern quality

LFSR and CA test pattern fault coverage

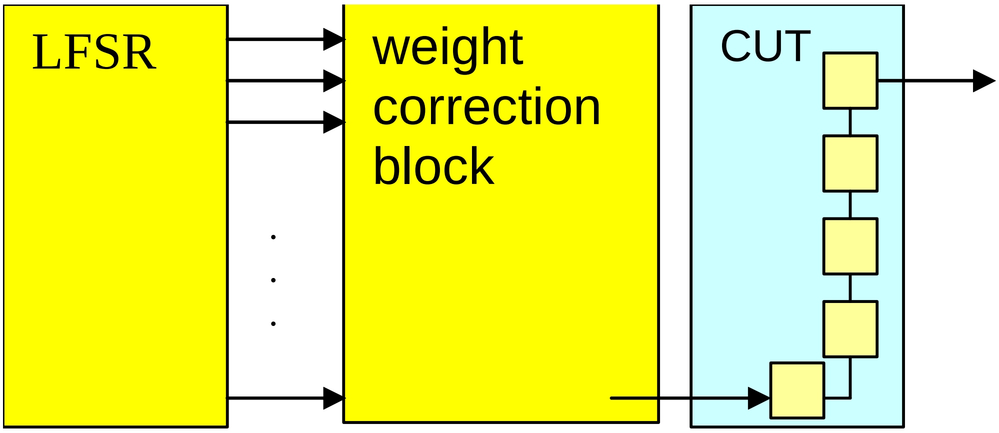
circuit	test cube length	32 bit LFSR : undetected faults	CA: undetected faults
s 9234	247	648	616
s 5378	214	38	27
s1196	32	17	11
s 15850	611	605	565
s 13207	700	624	472
c 7552	206	324	231

Novak, O.: Pseudorandom,Weighted Random and Pseudoexhaustive Test Patterns Generated in Universal Cellular Automata, Springer: Lecture Notes in Computer Science 1667, Sept. 1999

Weighted Random Testing

- ❑ improves random testing
- ❑ input-oriented weights
- ❑ pattern-oriented weights

weighted pseudorandom test patterns



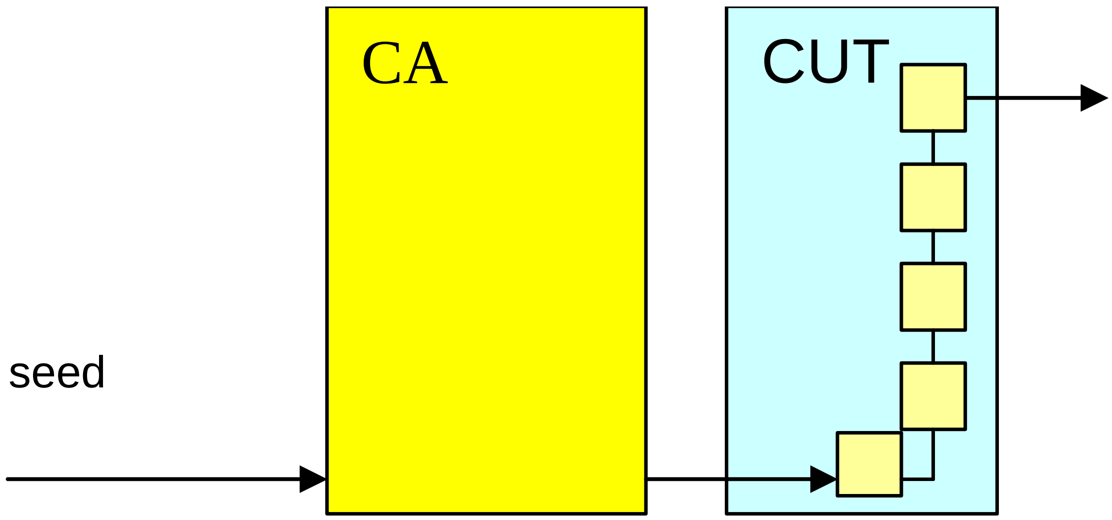
demonstrace LFSR

Pattern Oriented WeightedTest Set Effectiveness

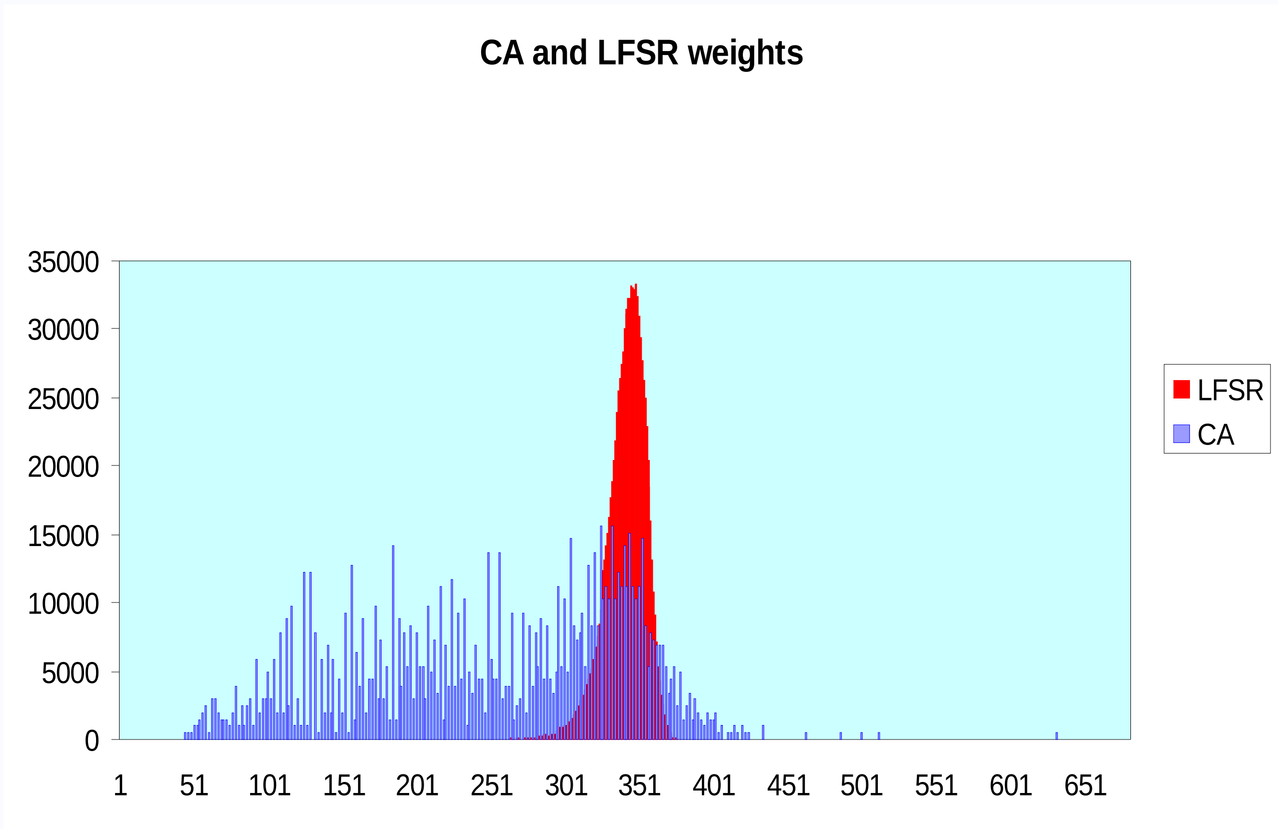
Benchmark Circuit	Fault Count (detectable)	Random Pattern Count	POWER-TEST: Pattern Oriented Weighted Patterns FC [%]	POWER-TEST: Pattern Oriented Weighted Patterns Undet. Faults
c2670	2,408	2,000	97.6	51
		10,000	99.5	6
		50,000	99.9	2
c5315	4,895	2,000	99.8	10
		10,000	100.0	0
		50,000		
c7552	6,876	2,000	96.8	212
		10,000	98.6	93
		50,000	99.5	29
s38584	30,237	2,000	96.8	844
		10,000	99.3	178
		50,000	99.9	31

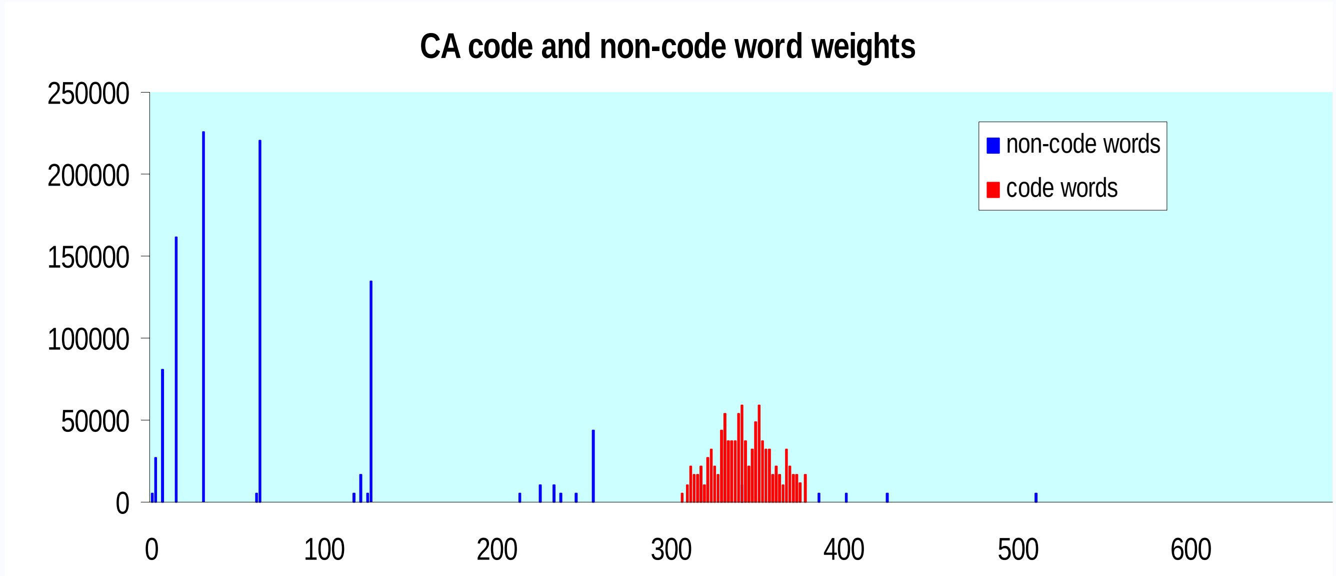
Kunzmann, A. : Efficient Random Testing with Global Weights. Proc. of IEEE EURO-DAC `96

CA as a source of weighted random test patterns



demonstrace





Mixed mode testing

Kinds of test sequences:

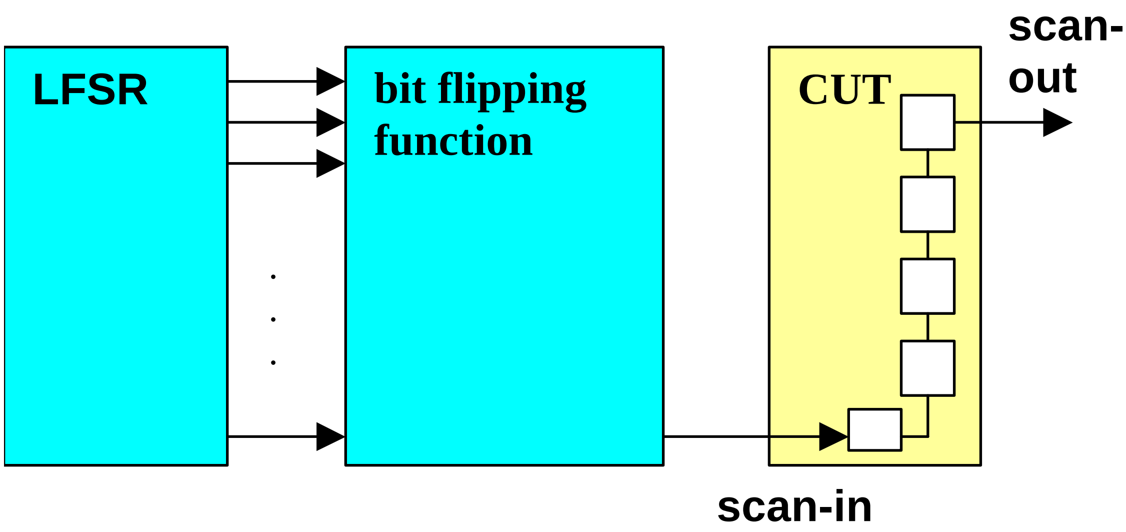
Using special comb. logic

- Automaton output bit modification (bit fixing, bit flipping)
- Using modified LFSR parallel output vectors [Chatterjee, Pradham 1995]

Using modification bits stored in a memory

- automaton sequence deflection
- pseudoexhaustive testing + supplementary pattern generation

Test-per-scan: bit flipping



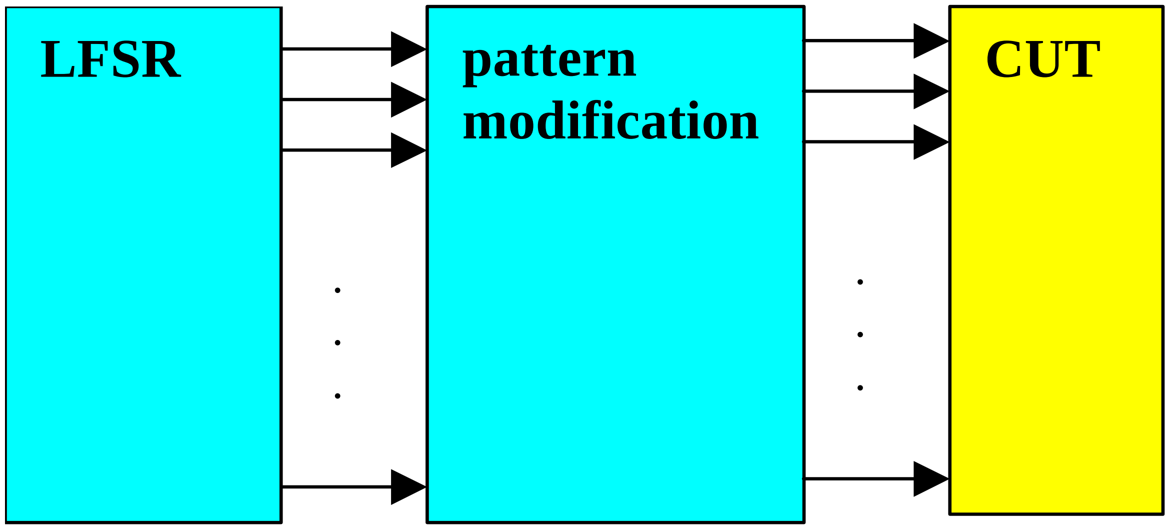
H.-J. Wunderlich, G. Kiefer: Bit-Flipping BIST, Proceedings ACM/IEEE International Conference on CAD-96 (ICCAD96), San Jose, California, November 1996 [WUN96]

IC area of MPLFSR reseeding and bit-flipping method [WUN96]

		Reseeding [HELL95]			Bit-flipping			
Circuit	n	LFSR	ROM	Area [μm^2]	LFSR	XORs	Terms	Area [μm^2]
s420	34	20	250	343,640	14	1	4	63,394
s641	54	22	183	344,013	14	1	3	62,544
s713	54	22	183	344,013	14	1	3	62,544
s838	66	36	1,623	533,077	14	2	37	99,566
s953	45	15	141	307,833	14	1	3	62,544
s1196	32	17	267	334,501	14	2	6	66,733
s1238	32	17	249	331,909	14	1	4	63,394
s5378	214	27	726	423,145	14	2	19	80,581
s9234	247	61	6,923	944,284	22	3	298	544,153
s13207	700	24	3,570	730,298	14	2	123	192,930
s15850	611	46	6,528	918,034	14	3	241	331,046
s38417	1,664	91	24,283	1,896,450	24	3	985	1,732,798
s38584	1,464	70	3,406	769,958	26	3	266	576,738
c2670	157	60	3,412	733,882	14	3	194	278,850
c7552	206	100	5,241	987,284	14	3	406	517,020

Results of the bit-flipping method and reseeding after 10,000 random patterns

Test-per-clock scheme, parallel LFSR outputs modification

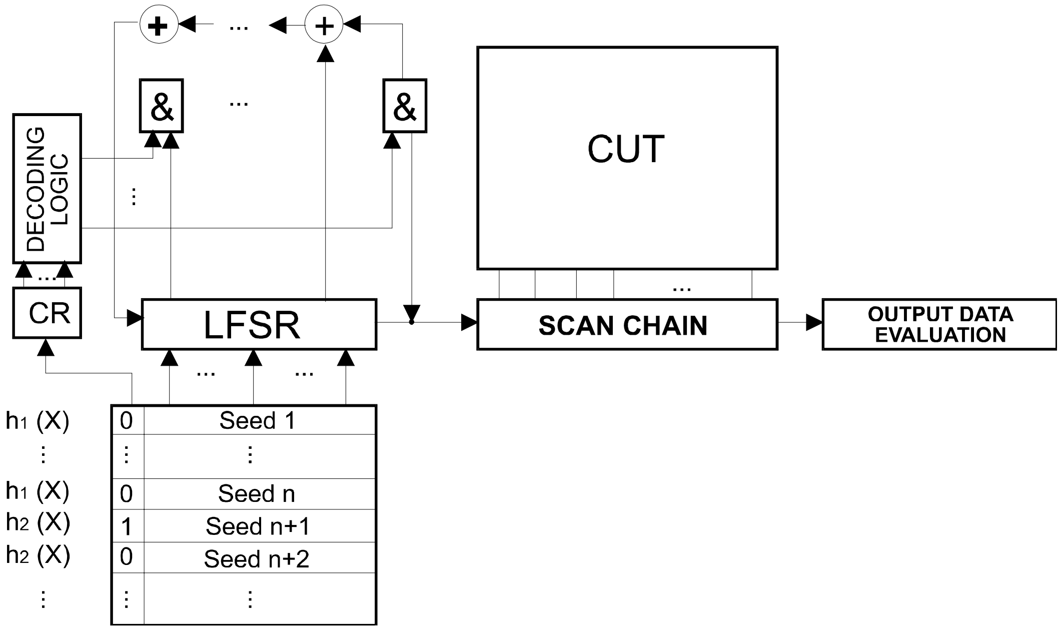


Chatterjee, M., Pradhan, D. K.: A Novel Pattern Generator for Near Perfect Fault-Coverage, IEEE VLSI Test Symposium, 1995

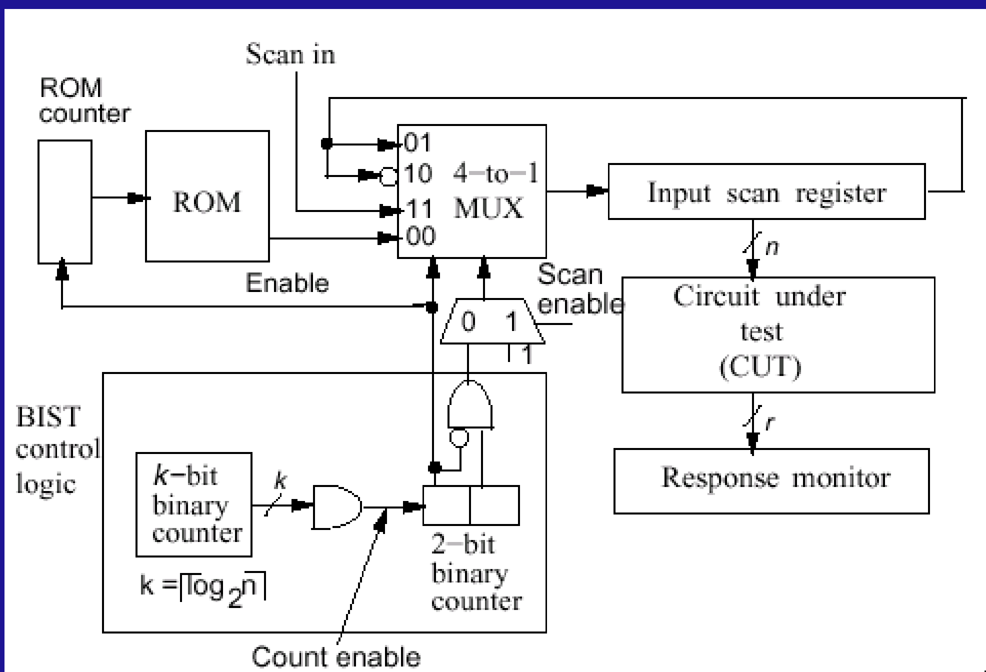
Methods using compressed test sequence in a memory

- Daen, W., Mucha, J,: Hardware Test Pattern Generation for Built-in Testing. Proc. of IEEE Test Conference, 1981
- Koenemann, B.: LFSR – coded test patterns for scan designs. Proc. Europ. Test Conf., Munich , Germany, 1991
- Hellebrand, S. - Rajski, J.- TarnickK, S.- Venkataraman, S. - Courtois, B.: Built-In Test for Circuits with Scan Based on Reseeding of Multiple-Polynomial Linear Feedback Shift Registers. IEEE Trans. on Comp., vol. 44, No. 2, February 1995 [HEL95]
- Chakrabarty, K. – Murray, B.T. – Iyengar, V.: Built-in Test Pattern Generation for High-Performance Circuits Using Twisted-Ring Counters. Proc. of IEEE VLSI Test Symp. 1999 [CHA99]
- Hellebrand, S., - Liang, H.G. – Wunderlich, H.J.: A mixed mode BIST scheme based on reseeding of folding counters. Proc. of IEEE ITC, 200 [HEL00]
- Novák, O., Hlawiczka, A. Garbolino, T, Guczwa, K. Plíva, Z. Nosek, J,: Low Hardware Overhead Deterministic Logic BIST with Zero-Aliasing Compactor. Proc. IEEE DDECS conf. Györ, Hungary 2001 [NOV01]
- Kaligeros,E., Kavousianos, X., Bakalis, D., Nikolos, D.: A New Reseeding Technique for LFSR-based Test Pattern Generation. Proc of 7th IOLTW On-Line Testing Workshop, 2001 [KAL01]
- Novák, O., Nosek, J. : Test Pattern Decompression Using a Scan Chain. Proc. of the 2001 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems [NOV01b]

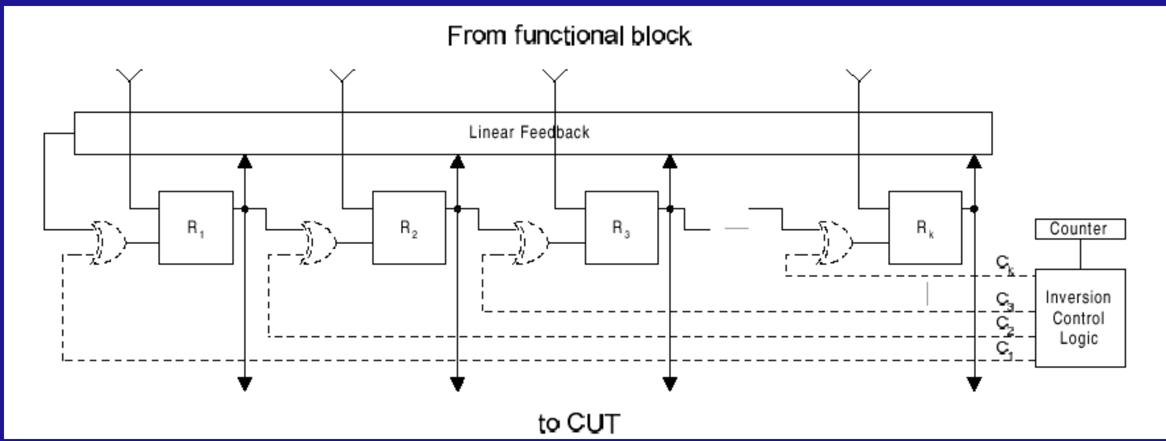
MP LFSR [HEL95]



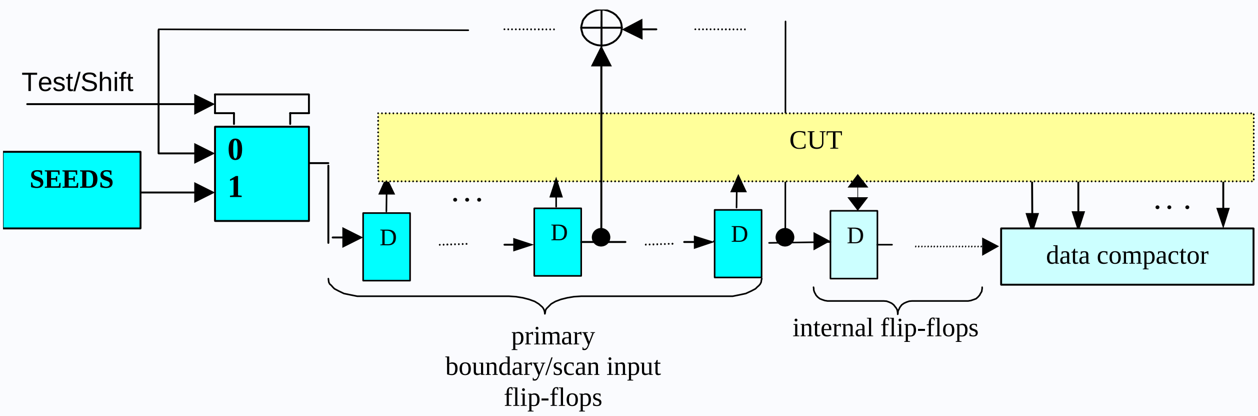
Twisted ring counter BIST [CHA99]



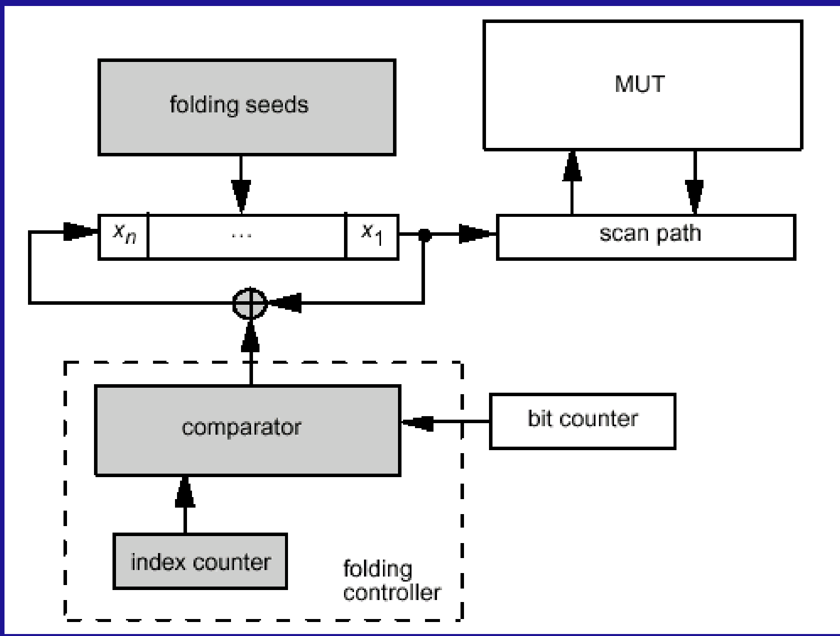
Reseeding Technique for LFSR-based Test Pattern Generation [KAL01]



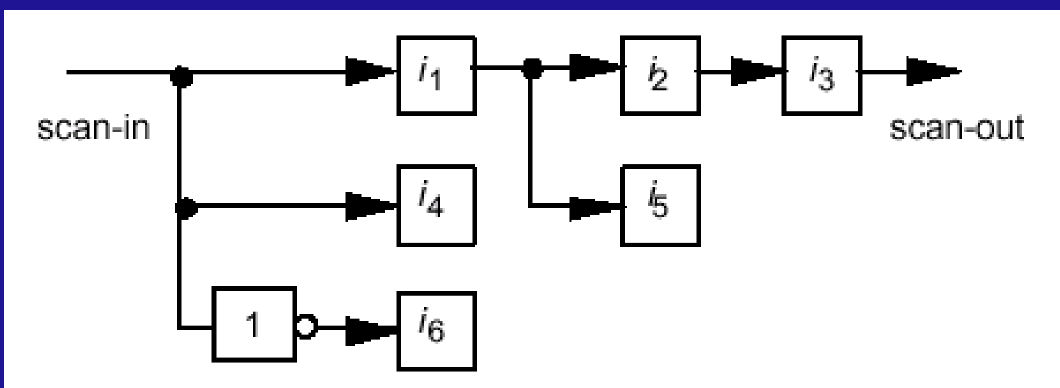
Long LFSR



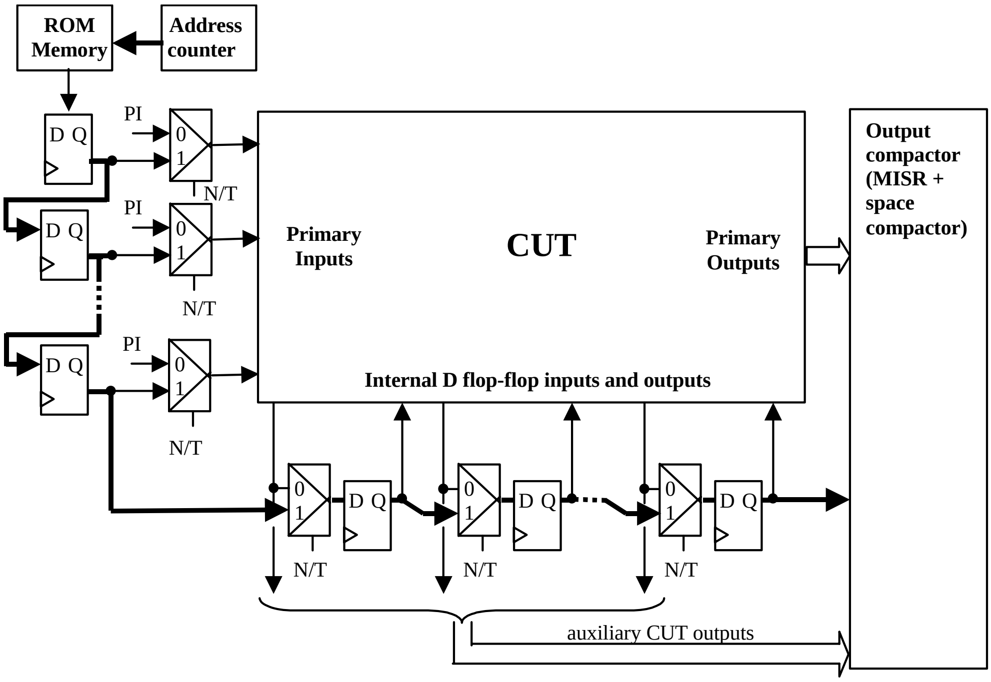
Folding counters BIST [HEL00]



Input reduction in scan-based BIST [HEL00]



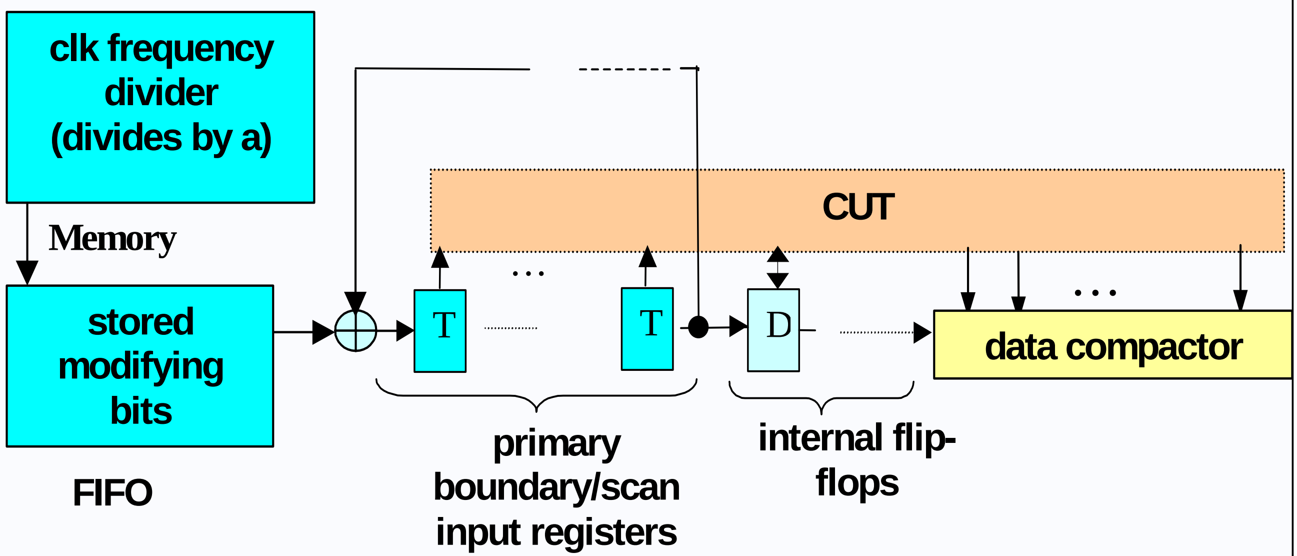
Using scan chain for test pattern decompression [NOV01b]



Scan Chain Decompression Ability [NOV01b]

Circuit	Stored bits	Scan chain/Mintest
c7552	7354	0,49
s953	715	0,21
s1196	800	0,22
s1238	832	0,21
s1423	659	0,33
s1488	439	0,31
s5378	2147	0,10
s9234	8960	0,35
s13207	4553	0,03
s15850	5158	0,09

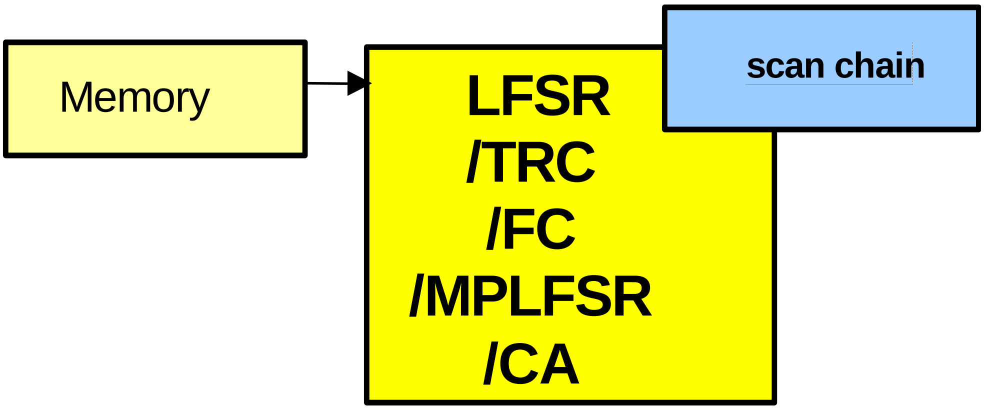
Using CA for test pattern decompression [NOV01]

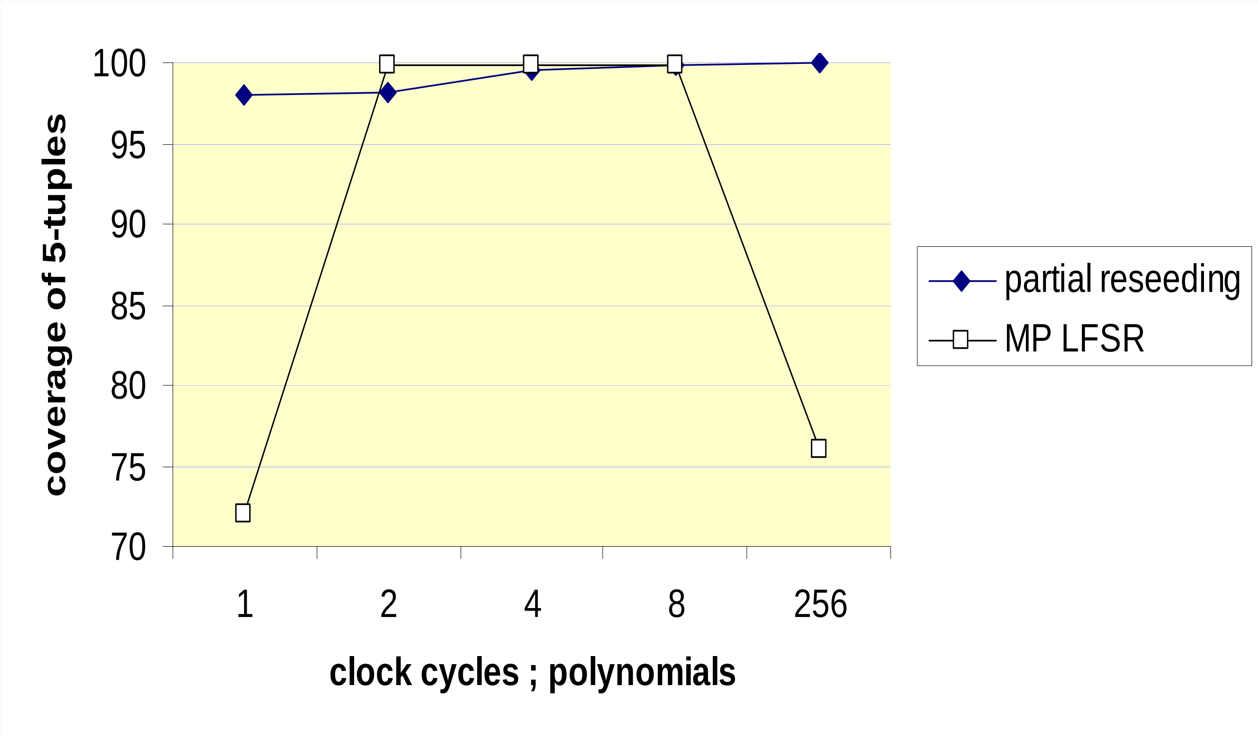


Decompression Automata

- scan chain
- binary counter
- LFSR with reseeding all internal bits simultaneously
- LFSR with output modification
- LFSR with sequence deflection
- Multi-Polynomial LFSR
- Cellular automaton
- Johnson (folding) counter

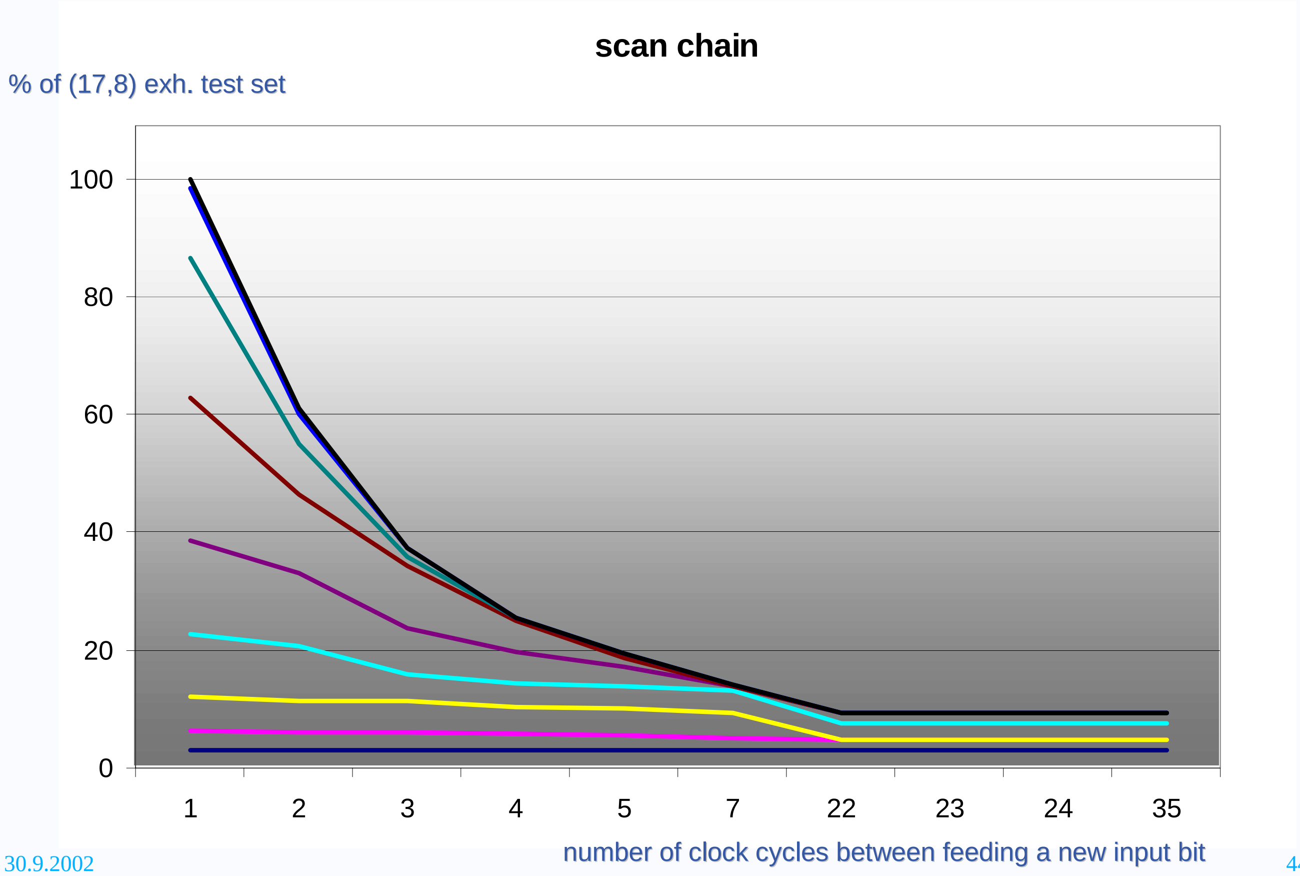
Hardware for test pattern decompression

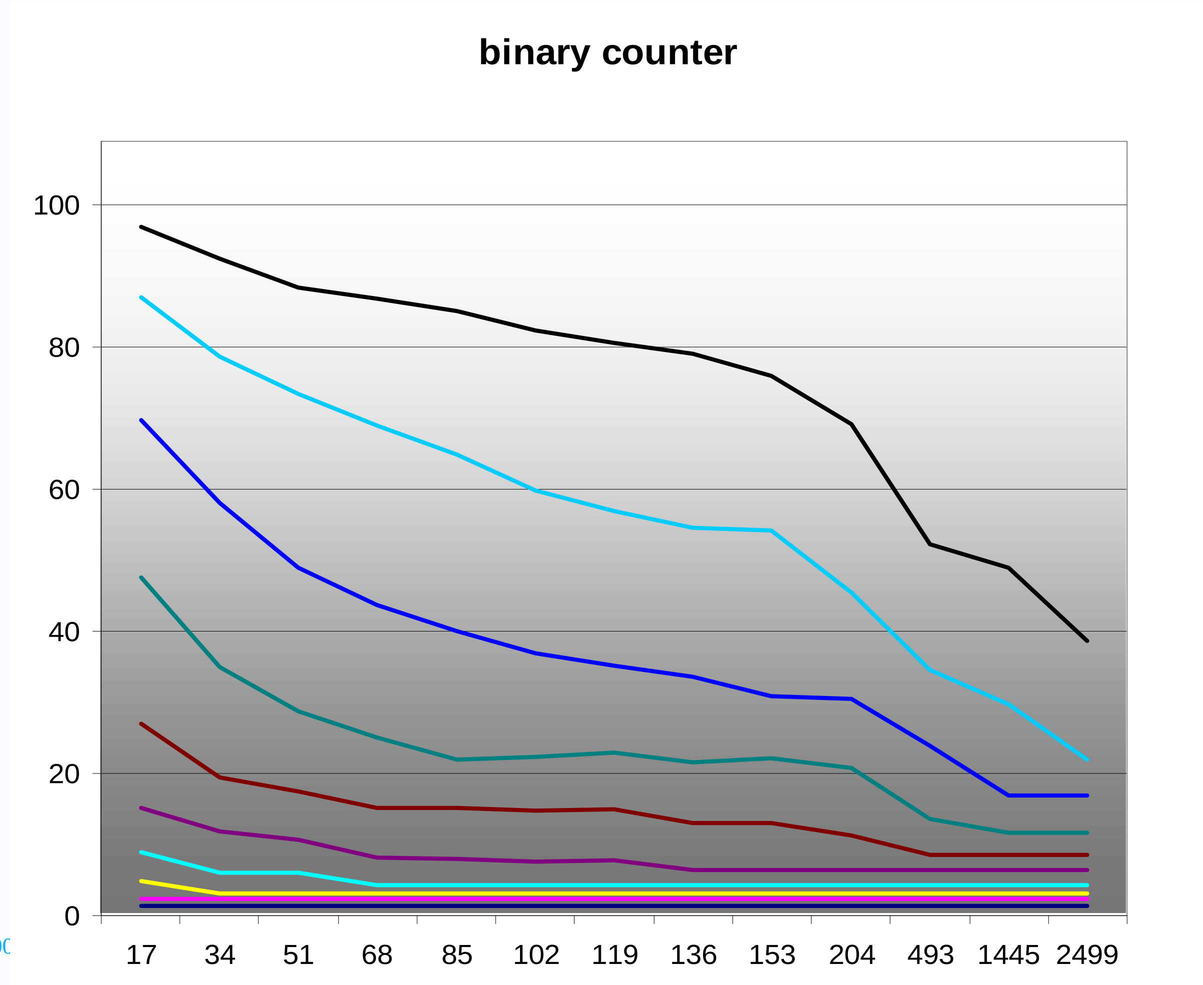


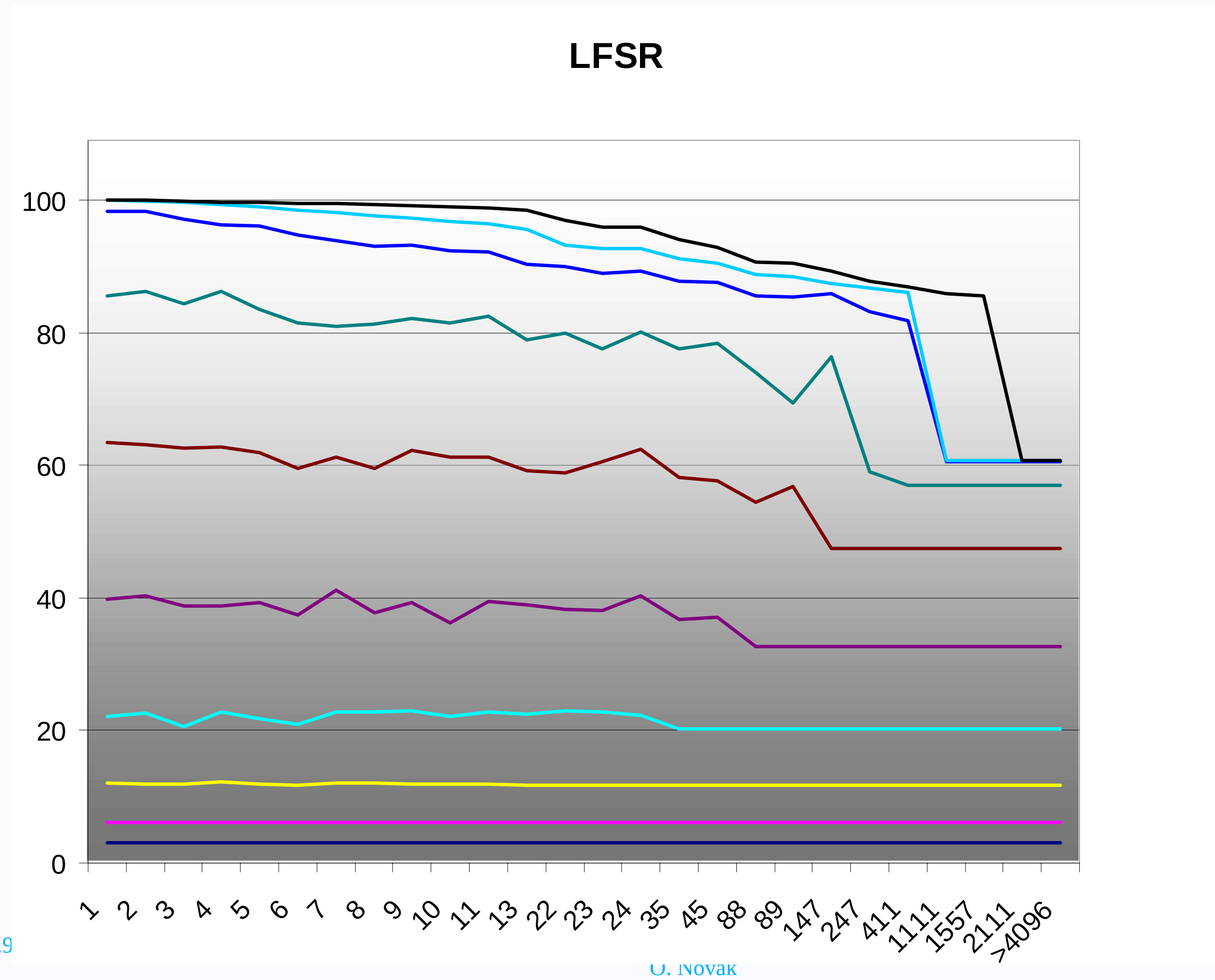


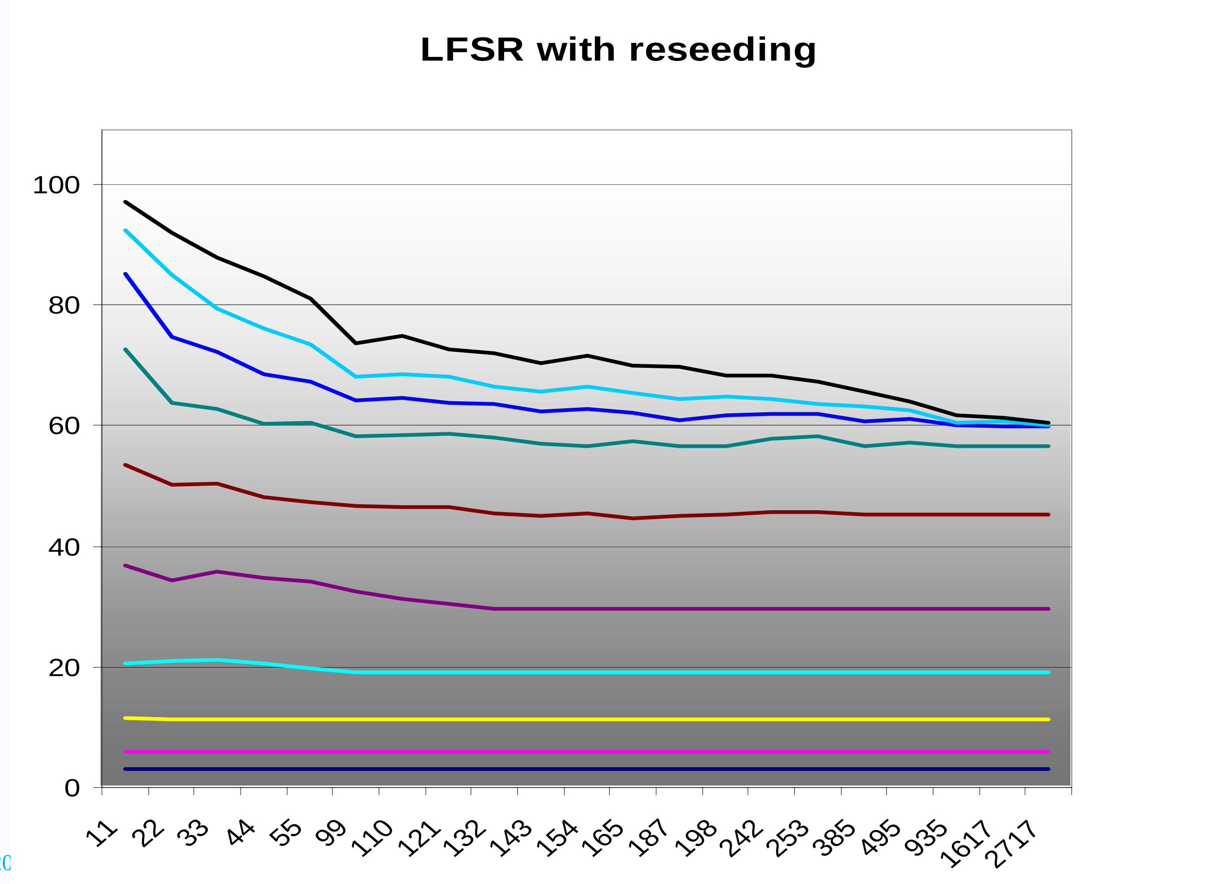
DA effectiveness: Experiment 1

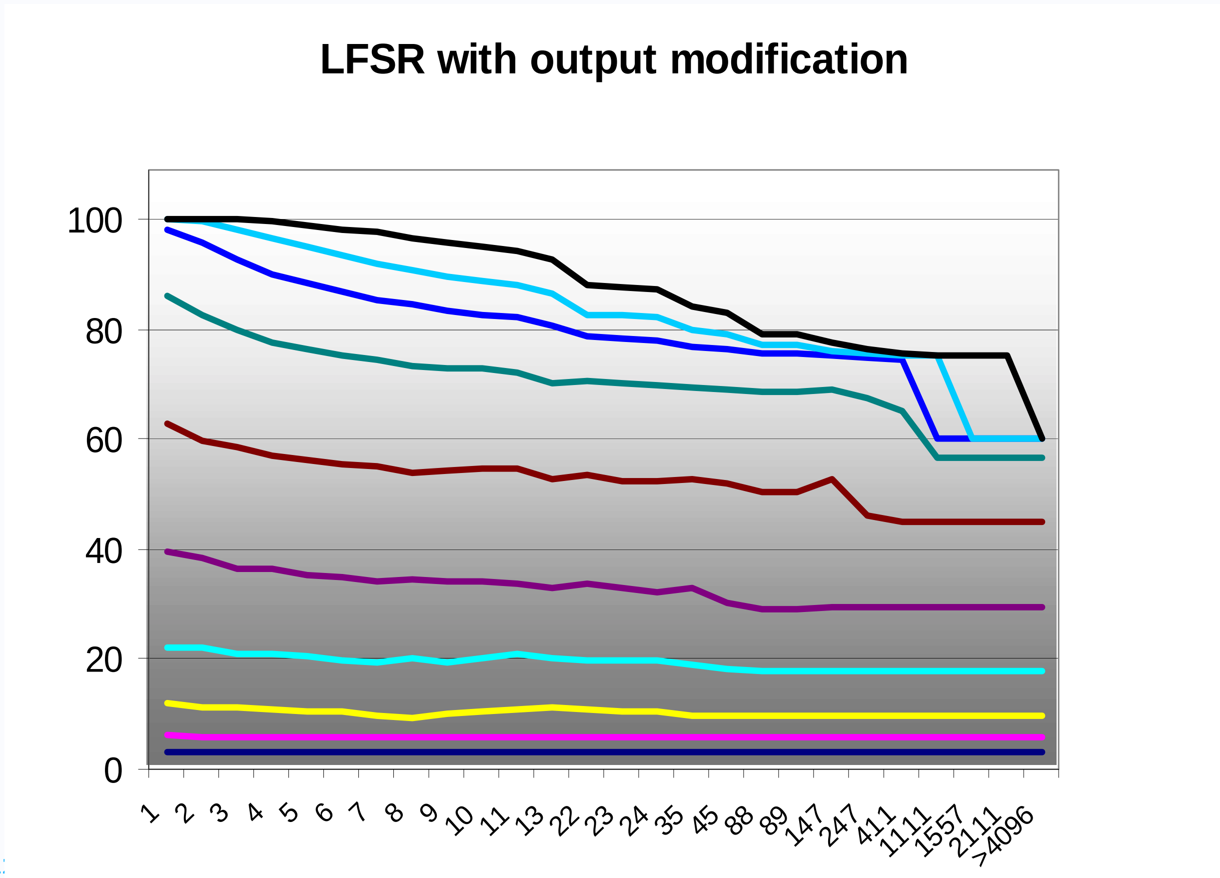
- Percentage of created (s,r) exhaustive test set against the number of clock cycles performed between feeding a new input random bit into the automaton. The number of automaton stages is fixed.

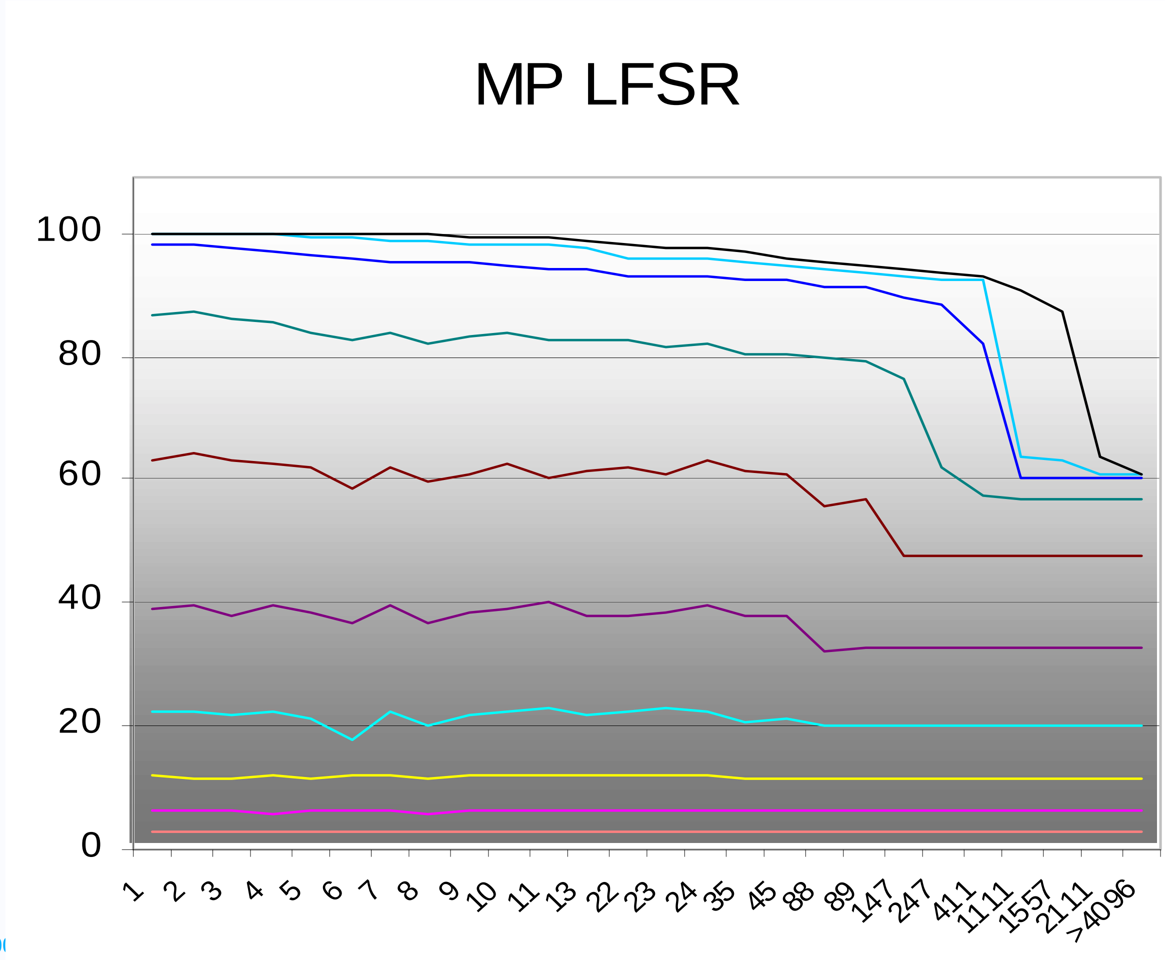


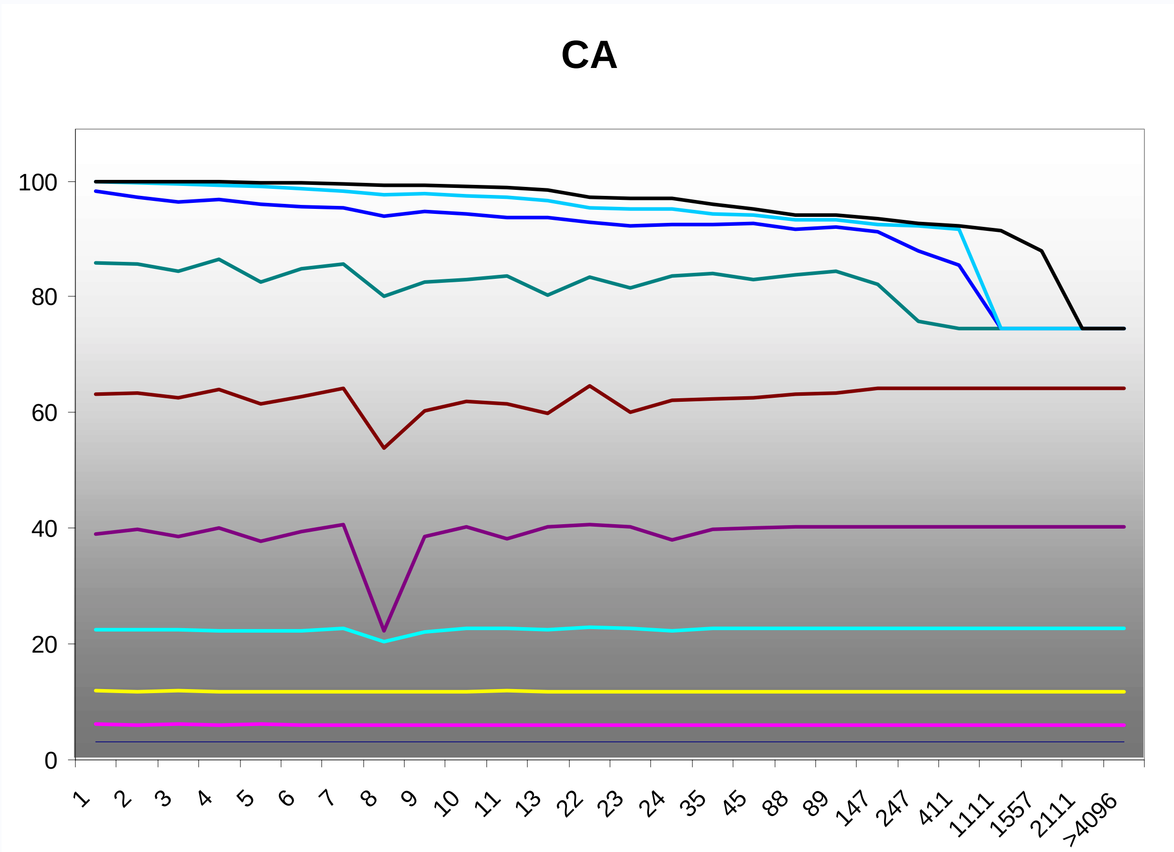

binary counter
100
80
60
40
20
0
17
34
51
68
85
102
119
136
153
204
493
1445
2499

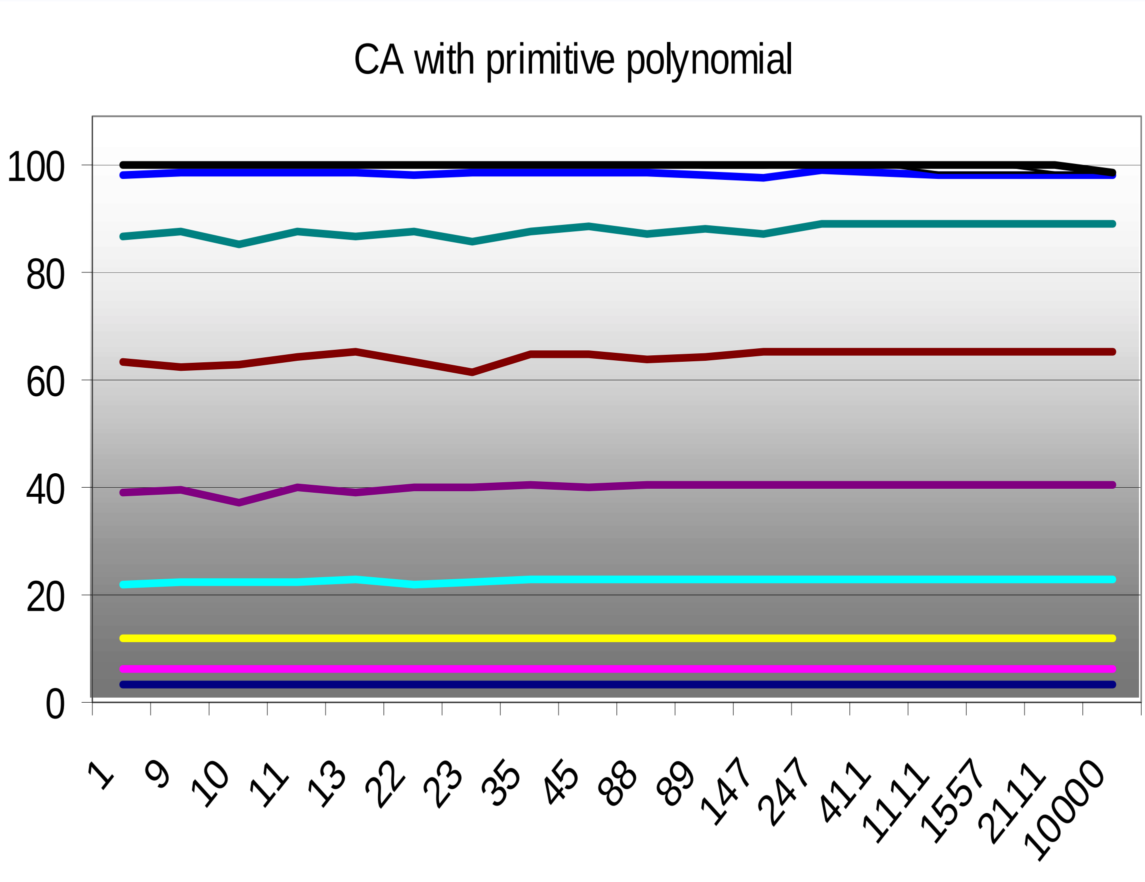

LFSR
100
80
60
40
20
0
1
2
3
4
5
6
7
8
9
10
11
13
22
23
24
35
45
88
89
147
247
411
1111
1557
2111
>4096

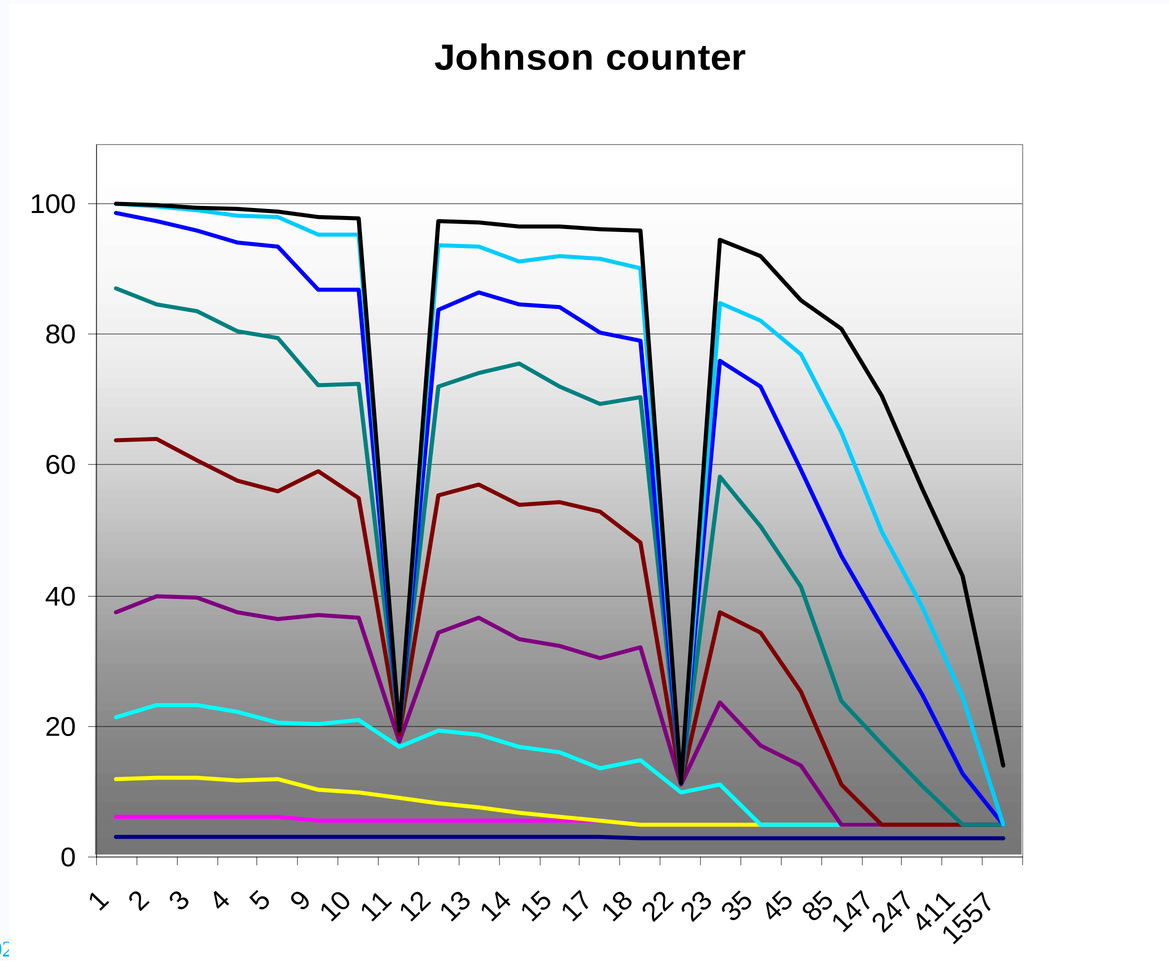

LFSR with reseeding
100
80
60
40
20
0
11 22 33 44 55 99 110 121 132 143 154 165 187 198 242 253 385 495 935 1617 2717


LFSR with output modification
100
80
60
40
20
0
1 2 3 4 5 6 7 8 9 10 11 13 22 23 24 35 45 88 89 147 247 411 1111 1557 2111 >4096


MP LFSR
100
80
60
40
20
0
1 2 3 4 5 6 7 8 9 10 11 13 22 23 24 35 45 88 89 147 247 411 1111 1557 2111 >4096

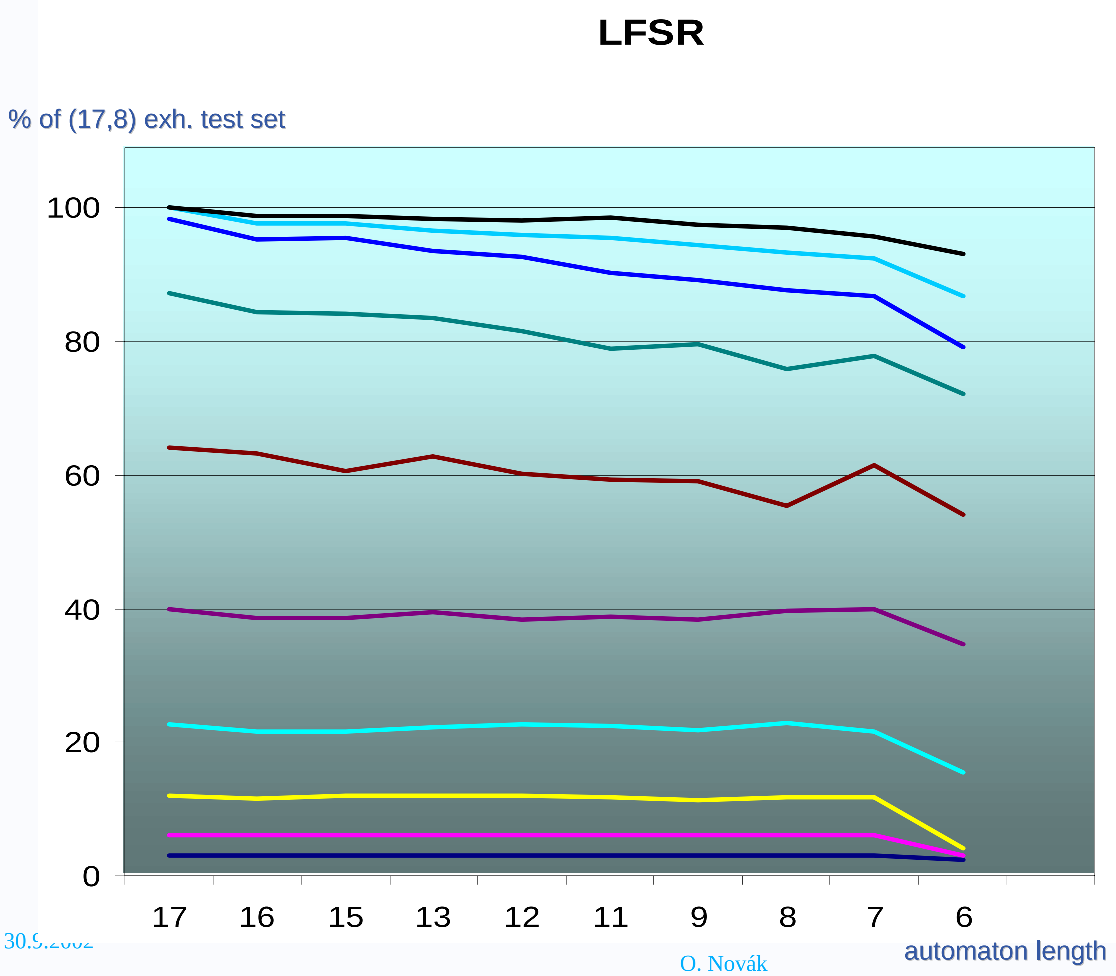

CA
100
80
60
40
20
0
1 2 3 4 5 6 7 8 9 10 11 13 22 23 24 35 45 88 89 147 247 411 1111 1557 2111 >4096

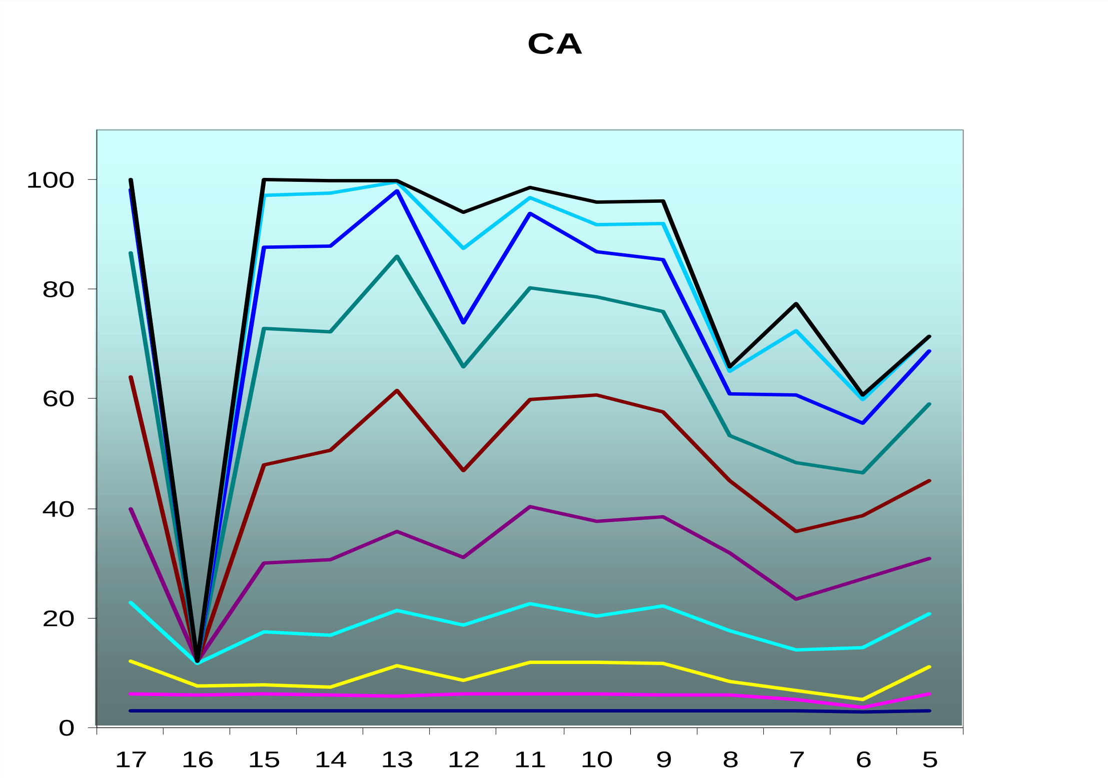

CA with primitive polynomial
100
80
60
40
20
0
1
9
10
11
13
22
23
35
45
88
89
147
247
411
1111
1557
2111
10000

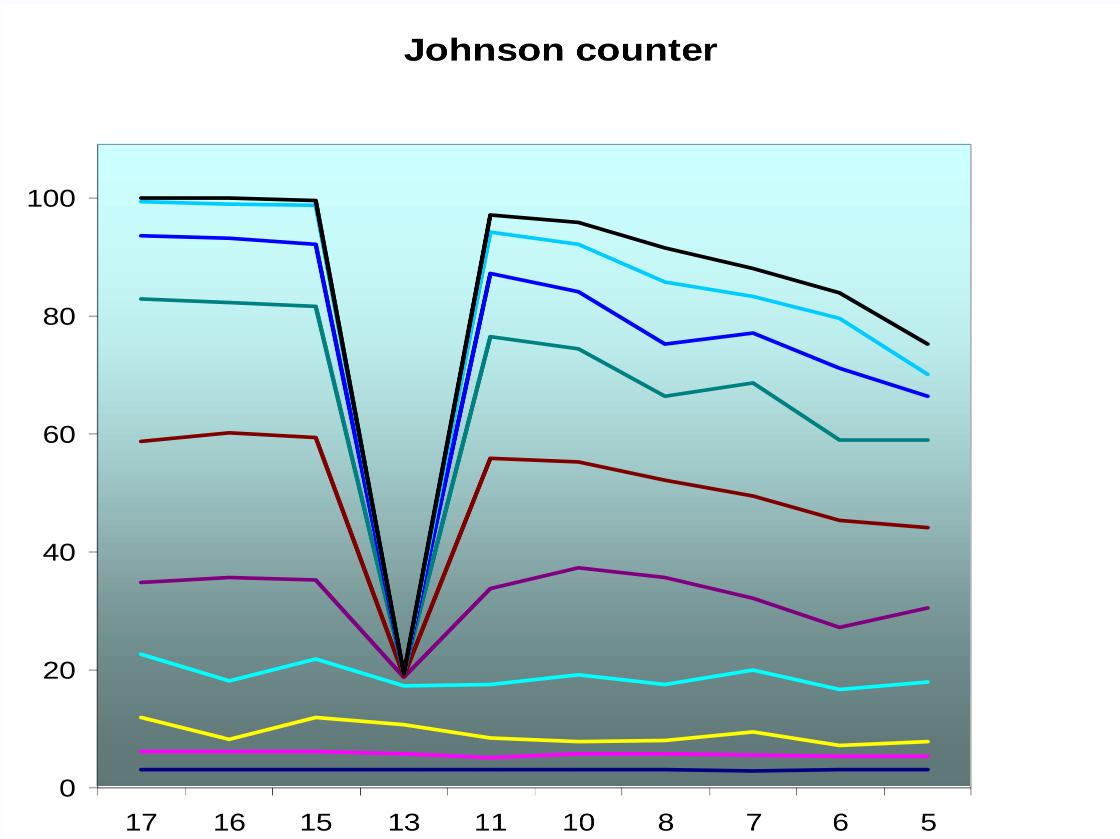

Johnson counter
100
80
60
40
20
0
1
2
3
4
5
9
10
11
12
13
14
15
17
18
22
23
35
45
85
147
247
411
1557

Experiment 2

- Percentage of created (s,r) exhaustive test set depending on the automaton dimensions. The number of clock cycles performed between feeding a new input random bit is fixed.

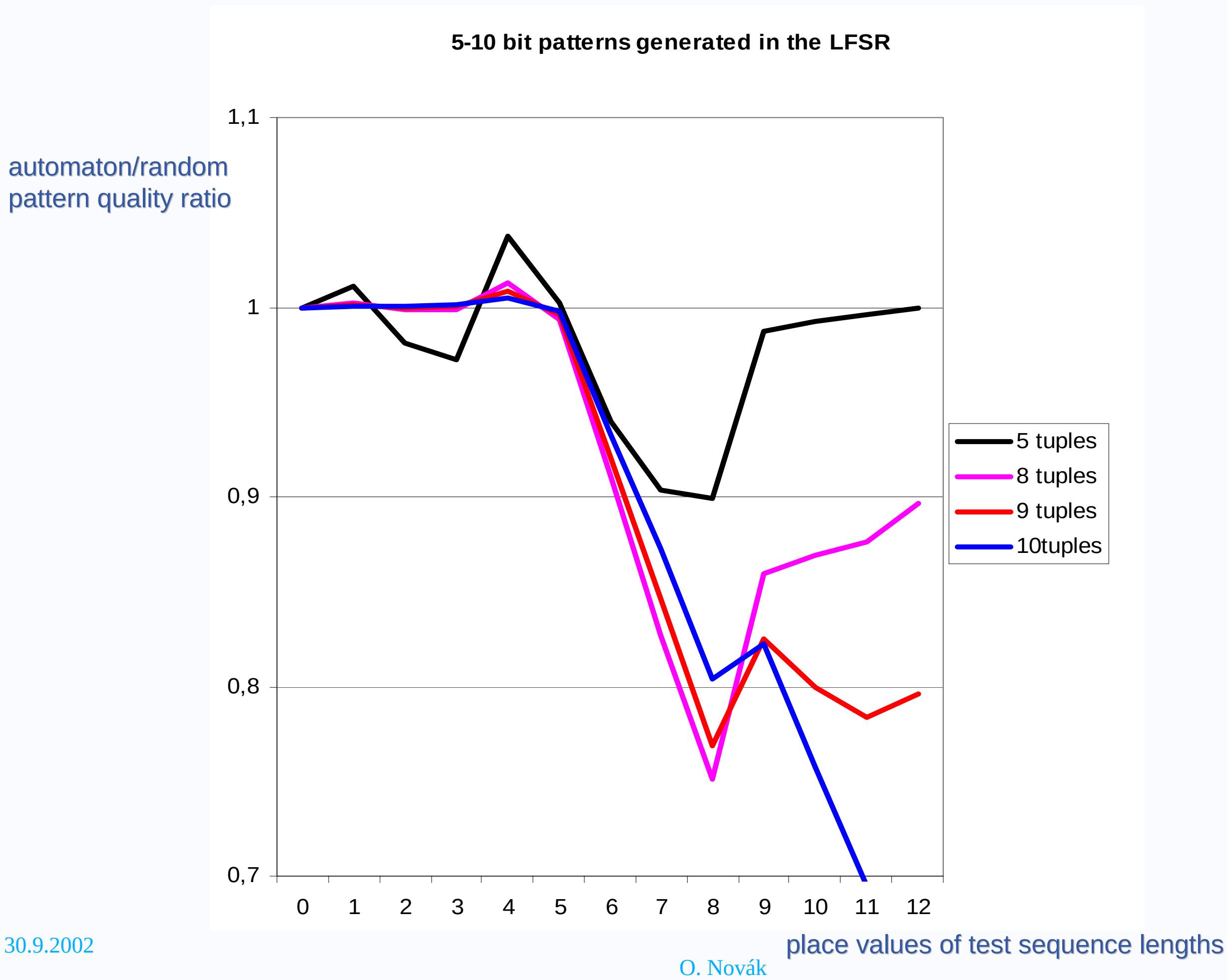


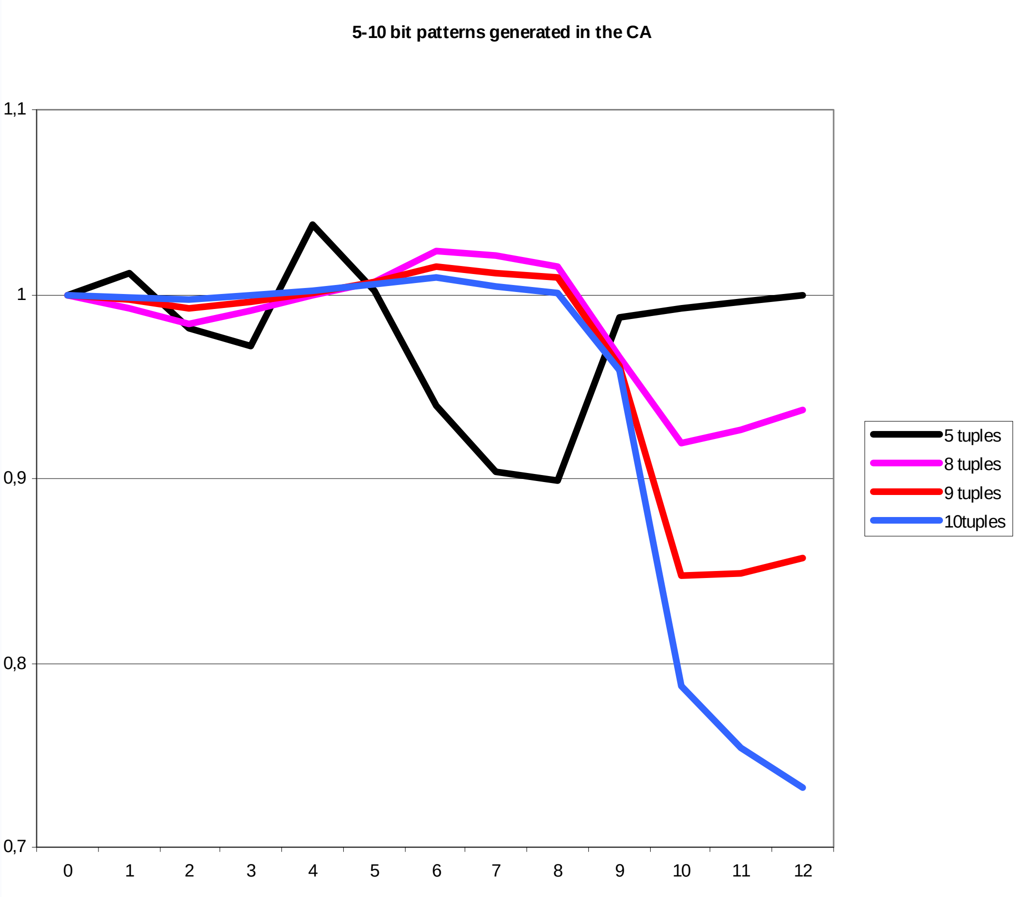

CA
100
80
60
40
20
0
17
16
15
14
13
12
11
10
9
8
7
6
5


Johnson counter
100
80
60
40
20
0
17
16
15
13
11
10
8
7
6
5

Experiment 3

- Relative part of (s,r) exhaustive test set, which was obtained by DA for several values r against the number of used test patterns. The number of clock cycles between feeding a new input bit is equal to 13.



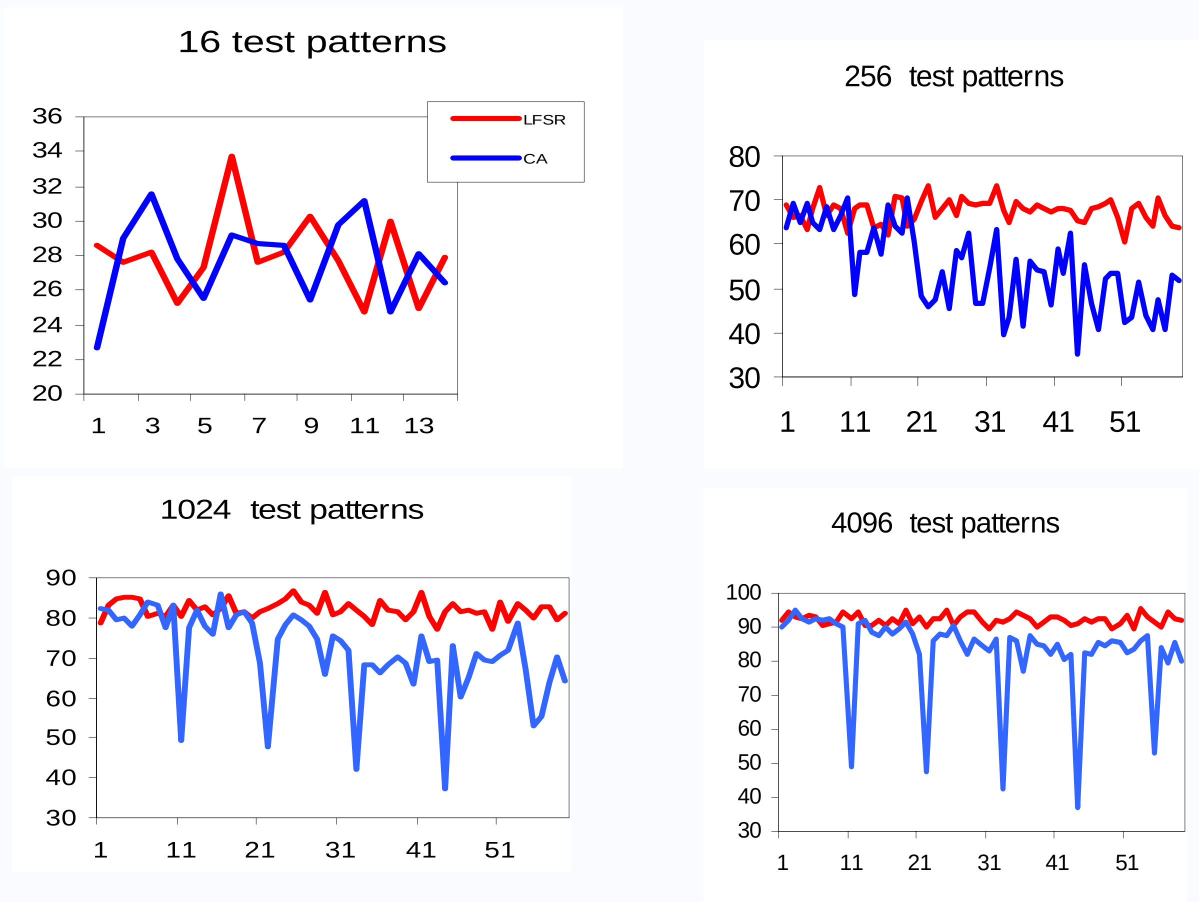


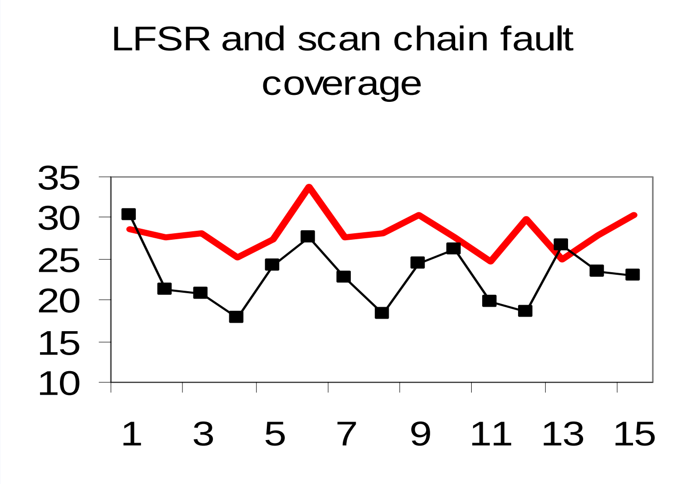
Experiment 4

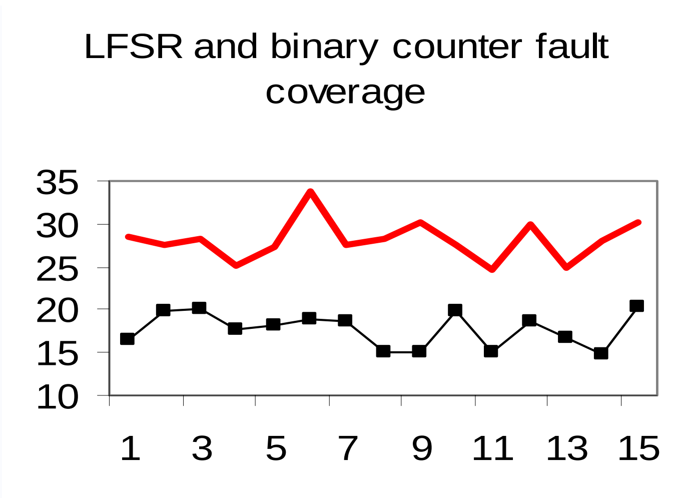
- Evaluation of the number of detected faults on ISCAS circuits. The automata are stimulated in the same way as in the previous experiments. Mathematical model is replaced with ISCAS circuits

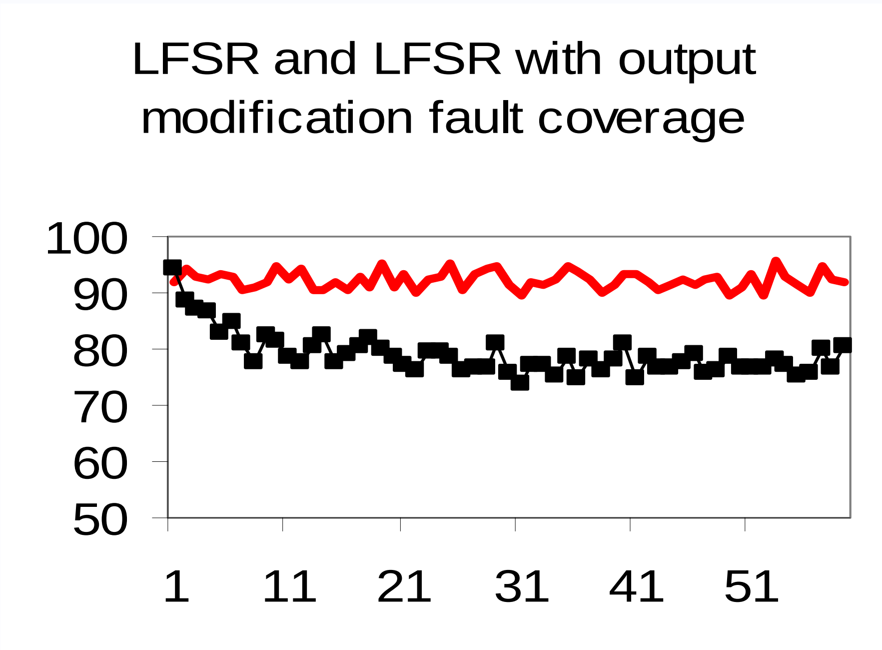
Average Decompression Effectiveness

- ❑CA with maximum period,
- ❑MP LFSR,
- ❑LFSR,
- ❑CA,
- ❑Johnson counter,
- ❑scan chain,
- ❑binary counter




LFSR and scan chain fault coverage
35
30
25
20
15
10
1
3
5
7
9
11
13
15


LFSR and binary counter fault coverage
35
30
25
20
15
10
1
3
5
7
9
11
13
15


LFSR and LFSR with output modification fault coverage
100
90
80
70
60
50
1
11
21
31
41
51

Conclusions from experiments with DAs

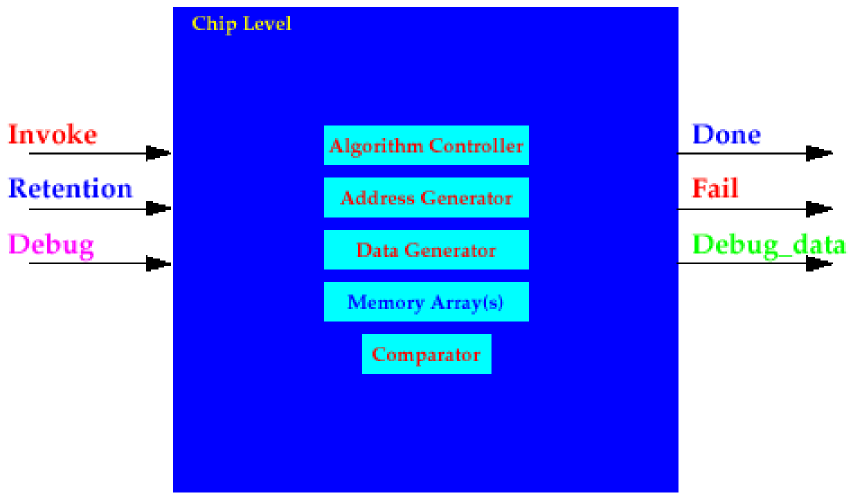
- Relative percentage of the (n,r) exhaustive test sets obtained from DAs, which are excited with regularly distributed random bits, gives us a general idea of the decompression quality of DAs
- Scan chain has similar pattern decompression effectiveness like other more complicated automata in case of using a new input bit every clock cycle

- Feeding the DA with one input bit in regular instants is very effective if a limited number of clock cycles is used between a new bit feeding
- For maximum memory savings (maximum numbers of autonomously performed cycles) we have to use more complicated hardware structures of DA in order to avoid the pattern quality degradation
- Mixed mode testing seems to be the most memory saving test approach
- Problem of difficult evaluation of the stored bits limits its usage

Memory BIST

- MBIST allows at-speed operation (access and recovery time can be ensured)
- BIST logic is developed as HDL and is brought through sysntesis with timing constrains to ensure that it operates at speed.

M BIST Requirements



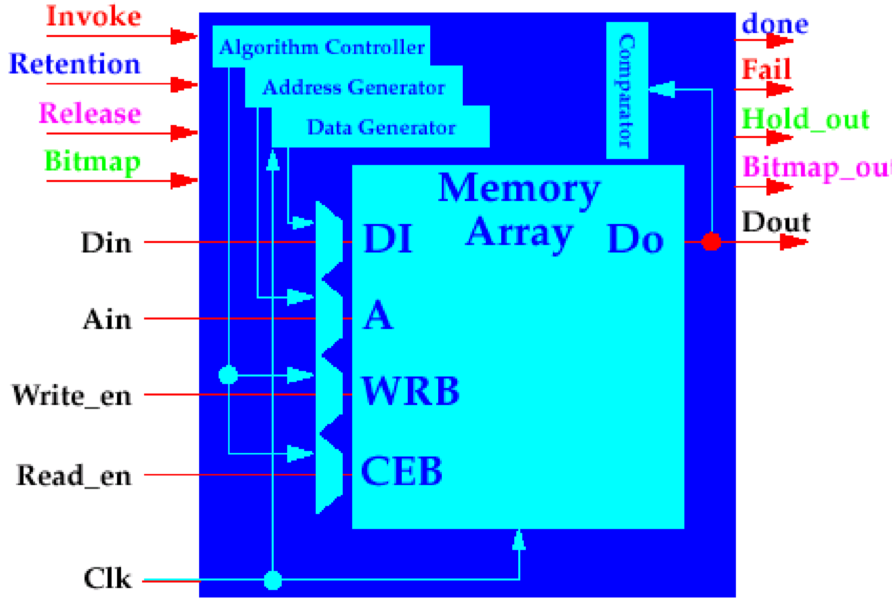
INPUTS
Invoke: Start BIST
Retention: Pause BIST and Memory Clocking
Debug: Enable BIST Bitmap Output

OUTPUTS
Fail: A Memory Has Failed a BIST Test
Done: Operation of BIST Is Complete
Debug_data: Debug Data Output

OPERATIONS
Address: Ability to Apply Address Sequences
Data: Ability to Apply Different Data Sequences
Algorithm: Ability to Apply Algorithmic Control Sequences
Comparator: Ability to Verify Memory Data

(from Crouch, A.L: Design for Test for Digital IC`s and Embedded Core Systems)

Example of M BIST



INPUTS

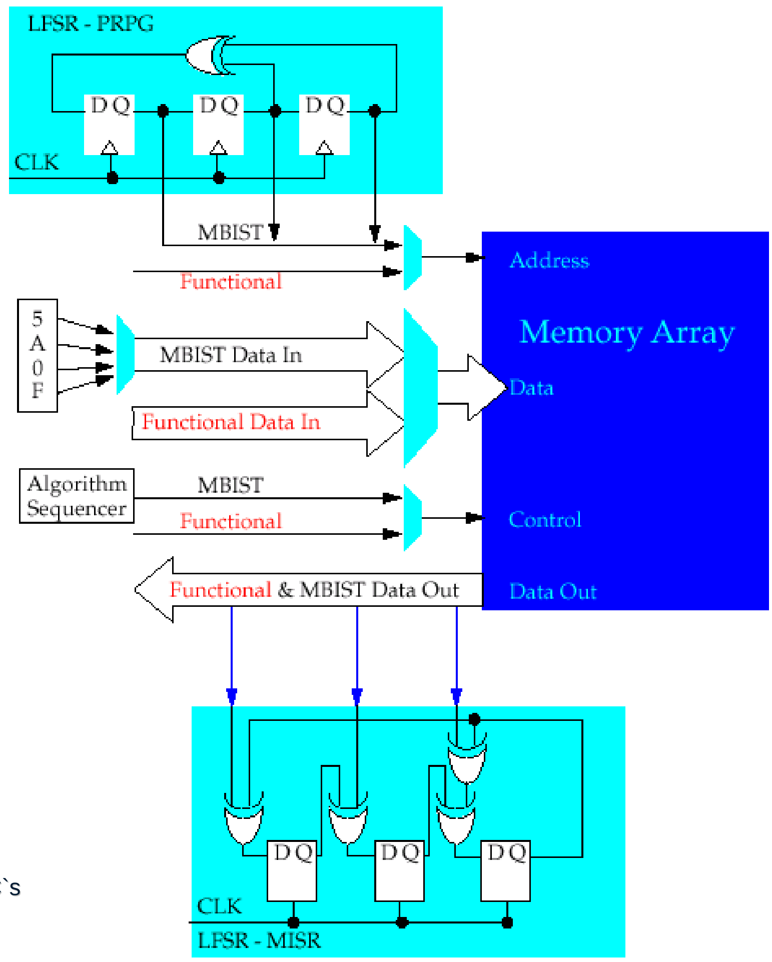
Invoke: invoke the BIST (apply muxes and release reset)
Retention: enable retention algorithm and pause
Release: discontinue and release pause
Bitmap: enable bitmap output on fail occurrence

OUTPUTS

Fail: sticky fail flag—dynamic under bitmap
Done: operation of BIST is complete
Bitmap_out: fail data under bitmap
Hold_out: indication of pause

(from Crouch, A.L: Design for Test for Digital IC`s and Embedded Core Systems)

LFSR – Based Memory BIST



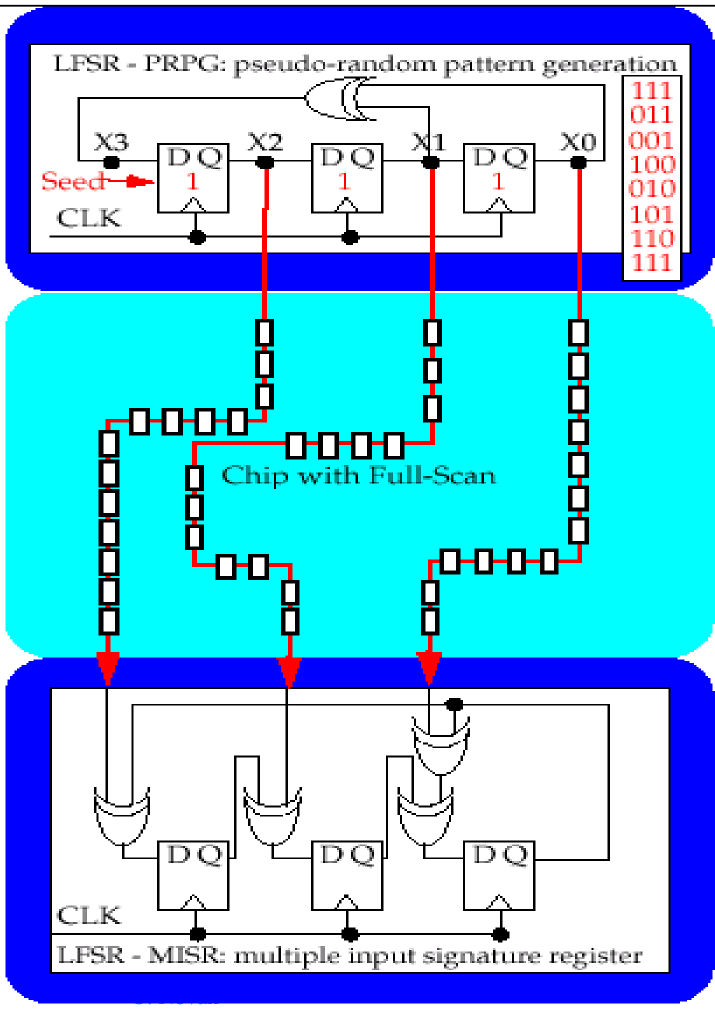
(from Crouch, A.L: Design for Test for Digital IC`s and Embedded Core Systems)

Signature analysers (SA)

- classically: 16 bit LFSR, primitive polynomial, 1 input
- MISR: additional XORed inputs
- problem of faulty CUT response masking (fault aliasing)
- BILBO, HILDO
- Circular BIST

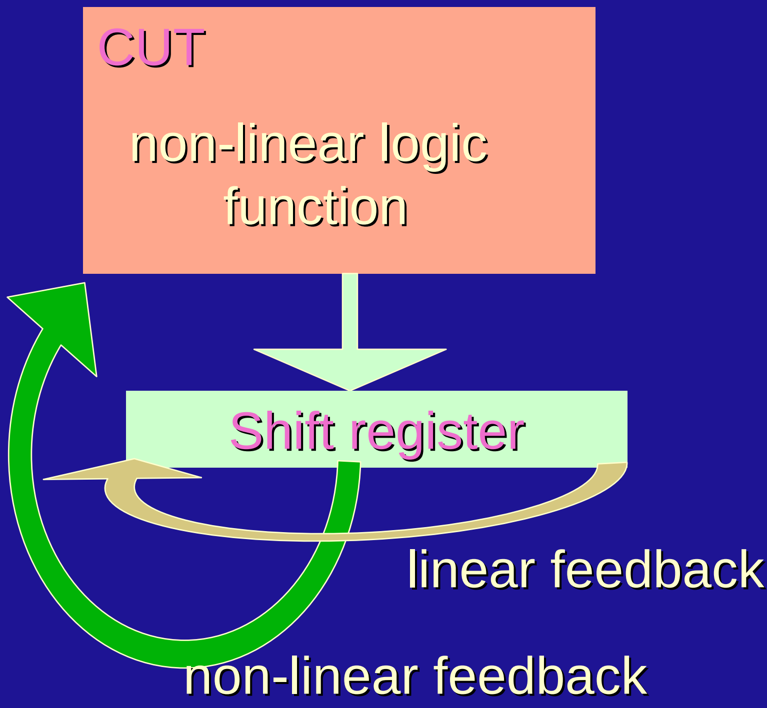
(fault aliasing = probability of two faults resulting in a correct signature)

CUT has to be designed with X- management



(from Crouch, A.L: Design for Test for Digital IC`s and Embedded Core Systems)

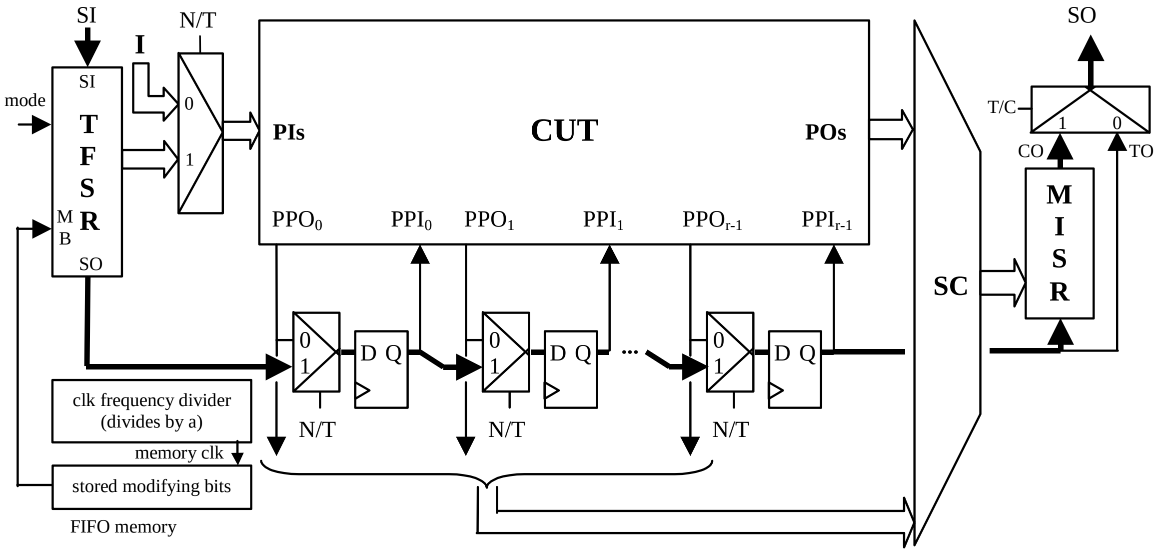
HILDO



Questions

- What is the number of undetectable faults for HILDO and C BIST?
- Is it possible to find a solution of HILDO for real circuits? (Problem of CPU time)
- In what extent do we have to take care about fault alising?

Test-per-clock BIST with zero fault aliasing [NOV01]



Conclusions

- BIST is commonly used in embedded memories (at speed testing)
- BIST becomes to be commonly accepted in core designs
- Mixed-mode BIST is a good compromise between low hardware overhead demand and the cost of testing (memory, time)