6. Rikete diagnoos

6.1. Kombinatoorne diagnoos (diagnostikasõnastikud)
6.2. Sekventsiaalne ehk adaptiivne diagnoos
6.3. Diagnostika resolutsioon
6.4. Diagnostika usaldatavus

Combinational Fault diagnosis

Fault localization by fault tables

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
T_1	0	1	1	0	0	0	0
T_2	1	0	0	1	0	0	0
T_3	1	1	0	1	0	1	0
T_4	0	1	0	0	1	0	0
T_5	0	0	1	0	1	1	0
T_6	0	0	1	0	0	1	1

E_1	E_2	E_3
0	0	1
0	1	0
0	1	0
1	0	1
1	0	1
0	0	0

Fault F_5 located

Faults F_1 and F_4 are not distinguishable

No match, diagnosis not possible

Combinational Fault Diagnosis

Fault localization by fault dictionaries

- **Fault dictionaries contain the sama data as the fault tables with the difference that the data is reorganised**
- **The column bit vectors can be represented by ordered decimal codes or by some kind of compressed signature**

No	Bit vectors	Decimal numbers	Faults
1	000001	01	F_7
2	000110	06	F_5
3	001011	11	F_6
4	011000	24	F_1, F_4
5	100011	35	F_3
6	101100	44	F_2

Test results:

$E_1 = 06$, $E_1 = 24$, $E_1 = 38$

No match

Combinational Fault Diagnosis

Minimization of diagnostic data

- To reduce the cost of building a fault table, the detected faults may be *dropped* from simulation
- All the faults detected for the first time by the same vector produce the same column vector in the table, and will included in the same equivalence class of faults
- Testing can stop after the first failing test, no information from the following tests can be used

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
T_1	**0**	**1**	**1**	**0**	**0**	**0**	**0**
T_2	**1**	0	0	**1**	**0**	**0**	**0**
T_3	0	0	0	0	**0**	**1**	**0**
T_4	0	0	0	0	**1**	0	**0**
T_5	0	0	0	0	0	0	**0**
T_6	0	0	0	0	0	0	**1**

With fault dropping, only 19 faults need to be simulated compared to the all 42 faults

The following faults remain not distinguishable:

{F_2, F_3}, {F_1, F_4}.

A tradeoff between computing time and diagnostic resolution can be achieved by dropping faults after k >1 detections

Fault Diagnosis Dilemmas

Diagnosis method	Fault table						Test result
Devil's advocate approach		Tested faults					Passed
				Tested faults			Failed
			Tested faults				Failed
Single fault assumption					Fault candi-dates		Diagnosis
Multiple faults allowed		?	Fault candidates				
Angel's advocate		Proved OK			Fault candidates		

6. Rikete diagnoos

- 6.1. Kombinatoorne diagnoos (diagnostikasõnastikud)
- 6.2. Sekventsiaalne ehk adaptiivne diagnoos
- 6.3. Diagnostika resolutsioon
- 6.4. Diagnostika usaldatavus

Sequential Fault Diagnosis

Sequential fault diagnosis by Edge-Pin Testing

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
T_1	0	1	1	0	0	0	0
T_2	1	0	0	1	0	0	0
T_3	1	1	0	1	0	1	0
T_4	0	1	0	0	1	0	0
T_5	0	0	1	0	1	1	0
T_6	0	0	1	0	0	1	1

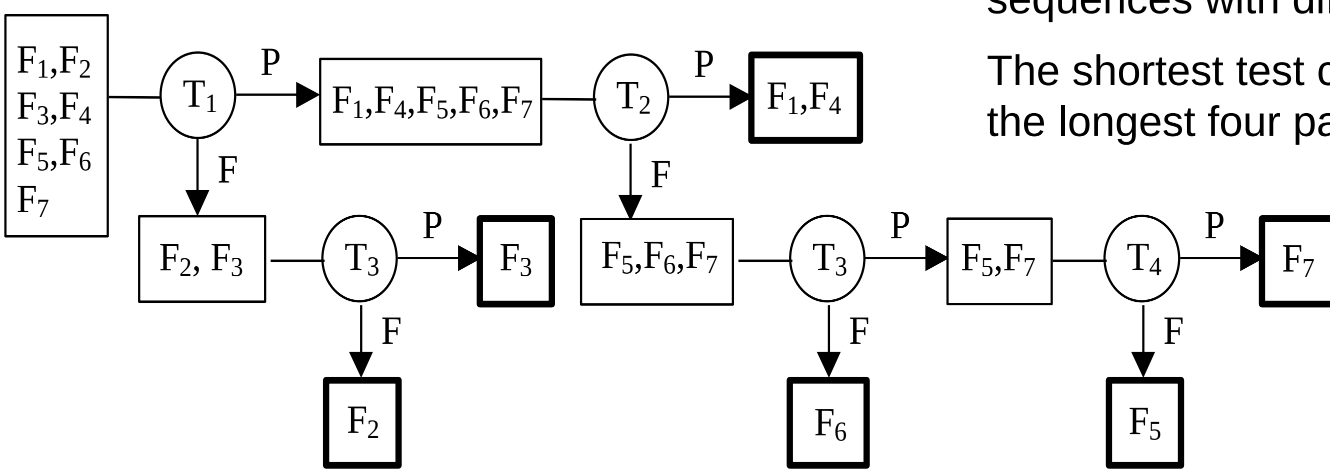
Diagnostic tree:

Two faults F_1,F_4 remain indistinguishable

Not all test patterns used in the fault table are needed

Different faults need for identifying test sequences with different lengths

The shortest test contains two patterns, the longest four patterns



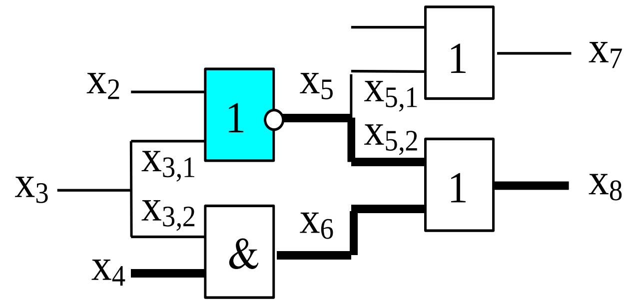
Questions:

1) How the tree Works if the system is fault-free?

2) What if a real fault is missing in the Fault Table?

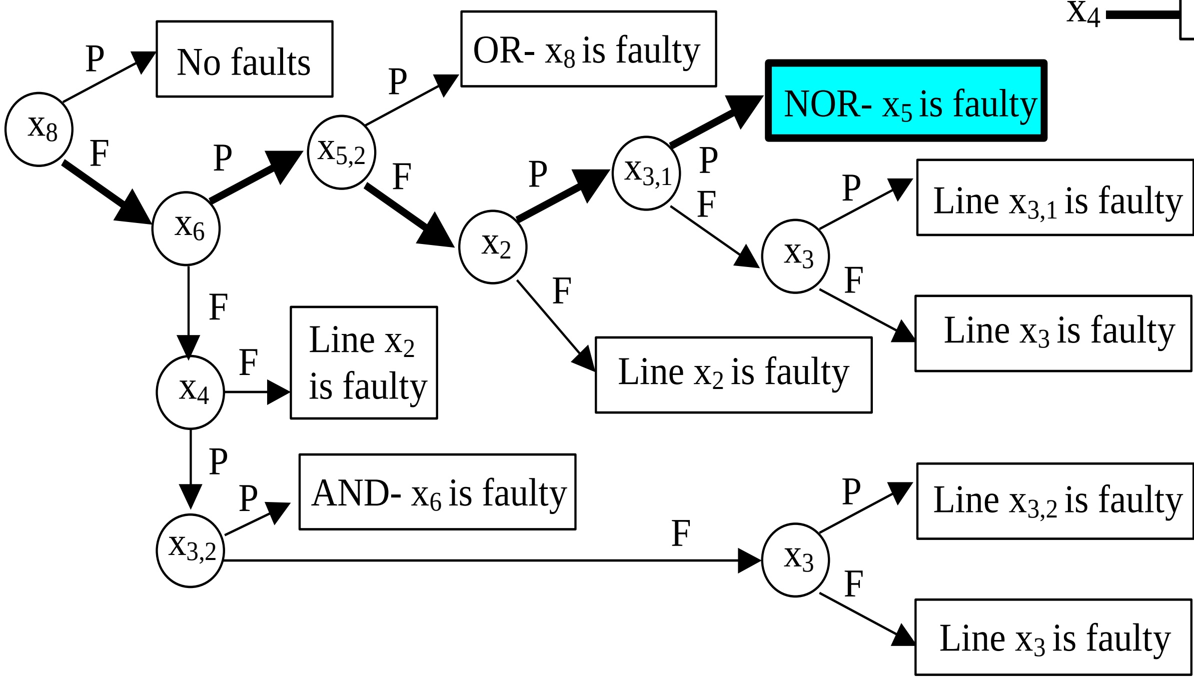
Sequential Fault Diagnosis

Guided-probe testing at the gate level



Faulty circuit

Searh tree:



Questions:

1) Is the order of measuring of signals $X_{5,2}$ and X_6 important?

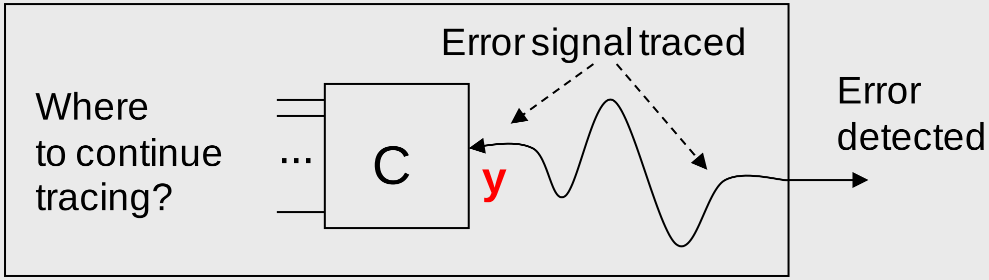
2) Need both $X_{5,2}$ and X_6 observation

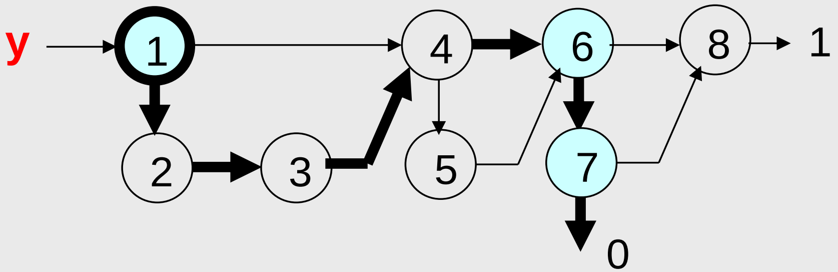
Optimized Critical Path Tracing with BDDs

Property 2:

If a test vector X activates in SSBDD a **0**-path (**1**-path) which travers a subset of nodes M, then only **0**-nodes (**1**-nodes) have to be considered as fault candidates

Fault diagnosis / Fault simulation:





Fault diagnosis and fault simulation can be speed-up by using Property 2

Speeding-up simulation:

M = {1,2,3,4,6,7}

M* = {1,6,7} – by Property 2

M** = {**6,7**} – by Property 1

Only **6** and **7** have to be considered

6. Rikete diagnoos

6.1. Kombinatoorne diagnoos (diagnostikasõnastikud)

6.2. Sekventsiaalne ehk adaptiivne diagnoos

6.3. Diagnostika resolutsioon

6.4. Diagnostika usaldatavus

About Diagnostic Resolution

Fault localization by fault tables at the single fault assumption

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
T_1	0	1	1	0	0	0	0
T_2	1	0	0	1	0	0	0
T_3	1	1	0	1	0	1	0
T_4	0	1	0	0	1	0	0
T_5	0	0	1	0	1	1	0
T_6	0	0	1	0	0	1	1

E_1	E_2	E_3
0	0	1
0	1	0
0	1	0
1	0	1
1	0	1
0	0	0

Fault F_5 located

Faults F_1 and F_4 are not distinguishable

No match, diagnosis not possible

About Diagnostic Resolution

Fault localization by fault tables at the multiple fault case, if fault masking takes place

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
T_1	0	1	1	0	0	0	0
T_2	1	0	0	1	0	0	0
T_3	1	1	0	1	0	1	0
T_4	0	1	0	0	1	0	0
T_5	0	0	1	0	1	1	0
T_6	0	0	1	0	0	1	1

Which faults are suspected if only T3 fails?

Fault F1 can be masked at T2, F2 - at T4, F4 at T2, and F6 – can be masked at both T5 and T6

Fault Diagnosis Dilemmas

Diagnosis method	Fault table						Test result
Devil's advocate approach		Tested faults					Passed
				Tested faults			Failed
			Tested faults				Failed
Single fault assumption					Fault candi-dates		Diagnosis
Multiple faults allowed		?	Fault candidates				
Angel's advocate		Proved OK			Fault candidates		

Improving Diagnostic Resolution

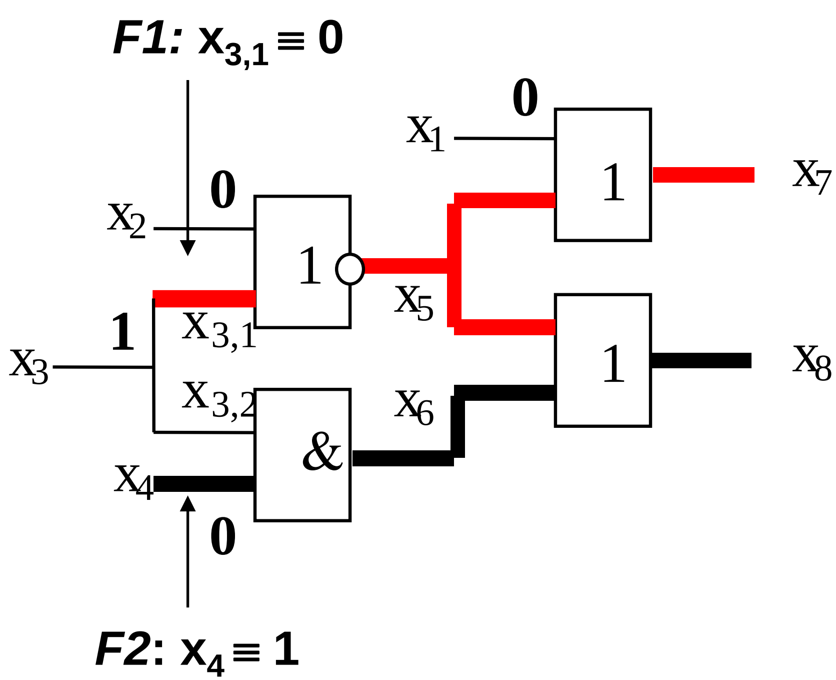
Generating tests to distinguish faults

- **To improve the fault resolution of a given test set *T*, it is necessary to generate tests to distinguish among faults equivalent under *T***
- **Consider the problem of distinguishing between faults *F1* and *F2.* A test is to be found which detects one of these faults but not the other**
- **The following cases are possible:**
  - ***F1* and *F2* do not influence the same outputs**
    - **A test should be generated for *F1 (F2)* using only the circuit feeding the outputs *influenced by F1 (F2)***
  - ***F1* and *F2* influence the same set of outputs.**
    - **A test should be generated for *F1* (F2) without activating *F2 (F1)***
- **How to activate a fault without activating another one?**

Improving Diagnostic Resolution

Generating tests to distinguish faults

Faults are influencing on different outputs:



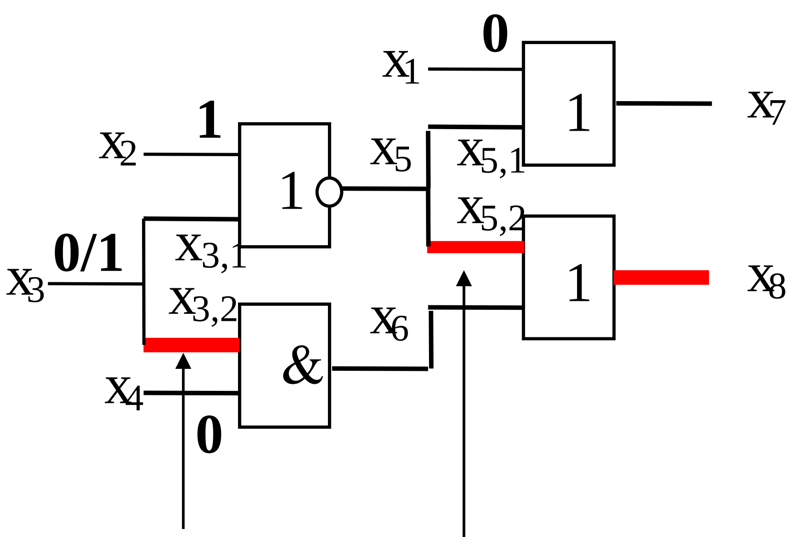
Method:

- ***F1*** **may influence both outputs,** ***F2*** **may influence only** x_8
- **A test pattern 0010 activates** ***F1*** **up to the both outputs, and** ***F2*** **only to** x_8
- **If both outputs will be wrong,** ***F1*** **is present**
- **If only** x_8 **will be wrong,** ***F2*** **is present**

Improving Diagnostic Resolution

Generating tests to distinguish faults

How to activate a fault without activating another one?



F1: $x_{3,2} \equiv 0$ ***F2***: $x_{5,2} \equiv 1$

Method:

- **Both faults influence the same output of the circuit**
- **One of them should be blocked**

Two possibilities:

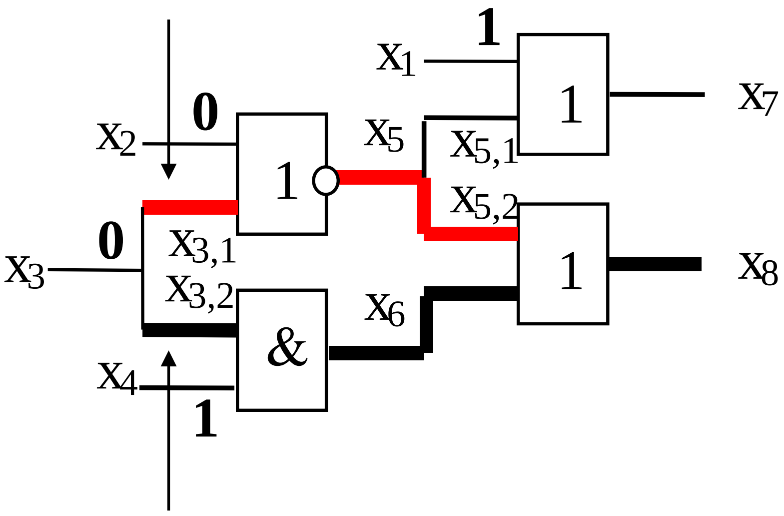
- **A test pattern 0100 activates the fault *F2*. *F1* is not activated: the line $x_{3,2}$ has the same value as it would have if *F1* were present**
- **A test pattern 0110 activates the fault *F2*. *F1* is now activated at his site but not propagated through the AND gate**

Improving Diagnostic Resolution

Generating tests to distinguish faults

How to activate a fault without activating another one?

F1: **$x_{3,1} \equiv 1$**



F2**:** **$x_{3,2} \equiv 1$**

Method:

- **Both of the faults may influence only the same output**
- **Both of the faults are activated to the same OR gate, none of them is blocked**
- **However, the faults produce different values at the inputs of the gate, they are distinguished**

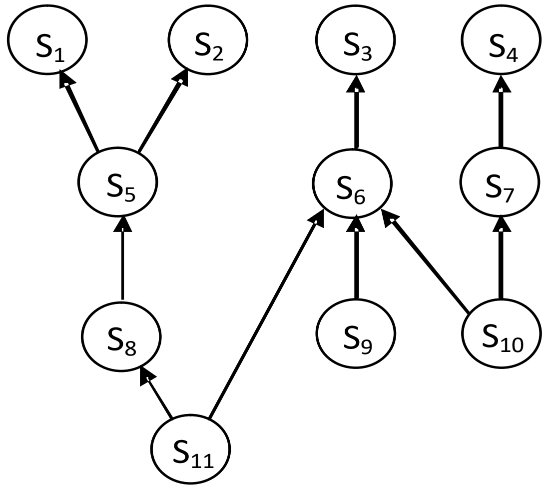
  if $x_8 = 0$, *F1* is present

  otherwise, if $x_8 = 1$ (OK value)

- **either *F2* is present**
- **or none of the faults are present**

Calculation of Diagnostic Resolution

Block-Level System Network



Diagnosability measure

$$D = \frac{\sum_{k=1}^{|M|} |M_k|}{|M|}$$

$$D = 1.57$$

Diagnostic matrix

B	Tests			
	T1	T2	T3	T4
1	1			
2		1		
3			1	
4				1
5	1	1		
6			1	
7				1
8	1	1		
9			1	
10			1	1
11	1	1	1	

Testability based partitioning of blocks:

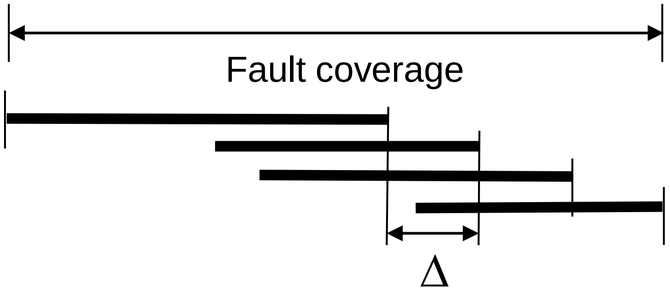
$$M = \{\{s_1\},\{s_2\},\{s_3,s_6,s_9\},\{s_4,s_7\},\{s_5,s_8\},\{s_{10}\},\{s_{11}\}\}$$

Random Generation of Diagnostic Tests

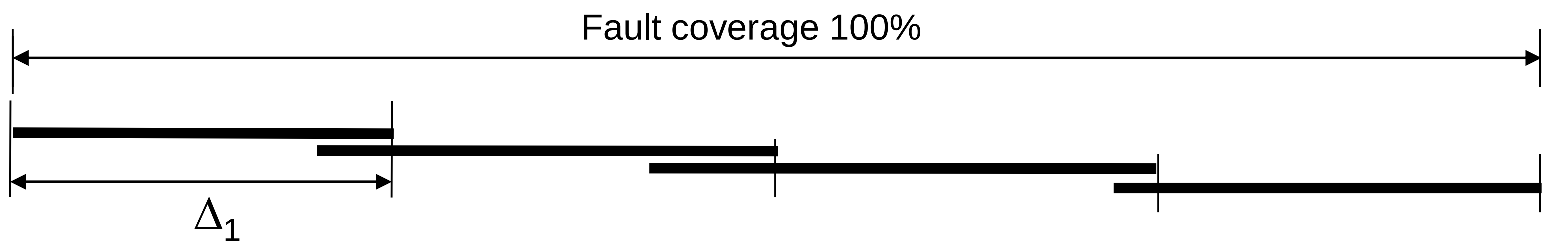
The main idea of the test generation method is to organize the test pattern selection process so that in each step the next test pattern is selected from a package of randomly generated patterns, which detects less or equal number of new not yet detected faults compared to the given criterion.

The algorithm can work as follows:

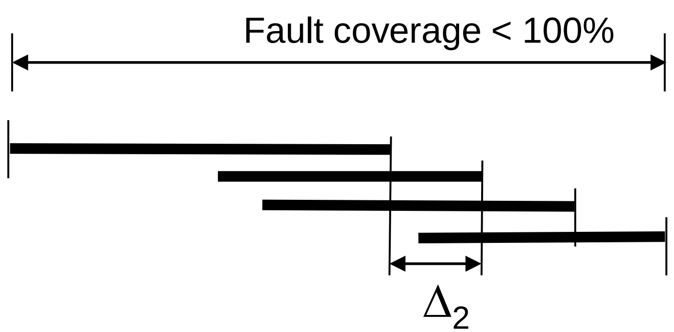
- Criterion Δ is defined
- A first pattern is selected which detects as less as possible faults
- Each next pattern is selected so that the number of new not yet detected faults were minimum for the current set of randomly generated patterns, and is less than Δ
- If such patterns can not be found the criterion Δ will be increased
- The procedure will finish when the number of unsuccessful trials will reach the predetermined criterion (value)



Random Generation of Diagnostic Tests



The criterion Δ_1 is equal to the min number of detectable faults. The probability of high average increment is big. Fault coverage is increasing fast, but the average diagnostic resolution remains bad (big average number of undistinguishable faults). The test length will be small.



The criterion Δ_2 is defined starting from 1 and will work only after fixing the first test pattern. The probability of high average increment is as low or less as the criterion. Fault coverage is increasing slowly, but this is not the goal. The average diagnostic resolution will be very good (small average number of undistinguishable faults). The length of the test will be big, but this is the normal cost of reaching the goal – good fault resolution.

6. Rikete diagnoos

6.1. Kombinatoorne diagnoos (diagnostikasõnastikud)

6.2. Sekventsiaalne ehk adaptiivne diagnoos

6.3. Diagnostika resolutsioon

6.4. Diagnostika usaldatavus

Diagnostic Simulation of multiple faults

State-of-the-art: Traditional methods of test generation can be trustworthy applied for single fault cases. If, however, there will be y present two or more faults, the faults can mask mutually each other

In testing, it is usually not the problem, because the same fault can be detected in general by several test patterns

In fault diagnosis, however, fault masking will cause wrong information from testing phase, and hence, the logic reasoning of faults for diagnostic purposes will be not any more possible.

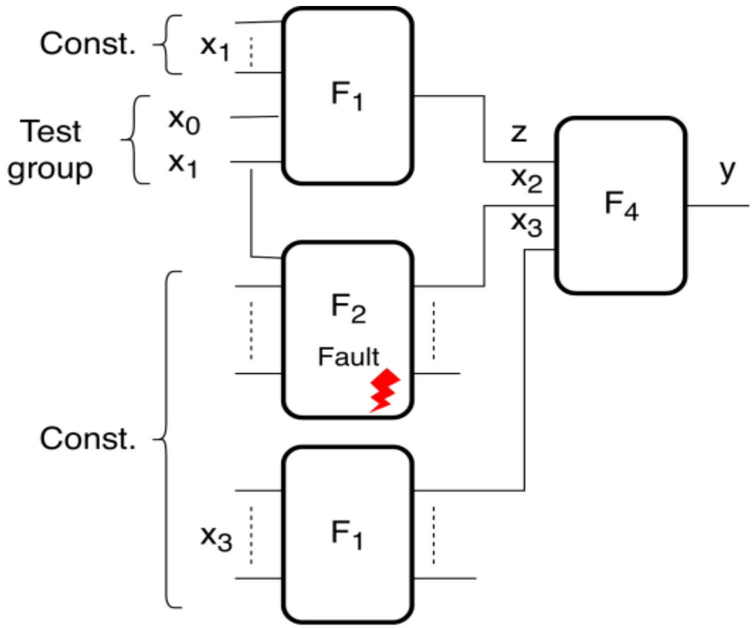
Diagnostic Simulation of multiple faults

Traditional methods are based on single fault assumption (advocate approach), and multiple fault mutual masking is not taken into account

In fault diagnosis, a fault masking produces wrong diagnostic information

The concept of iterative hierarchical fault diagnoosis:

- **The basis is the angel's advocate approach for generating test groups where each of them is able to prove the correctness of a related subcircuit**
- **A novel test strategy of „extending core" is proposed: if a core is proved to be fault-free, this knowledge can be used for**
  - **(1) extending step-by-step the core proved to be fault free, and**
  - **(2) if detecting an error the location of fault is specified as exactly as possible**



Example: Assume, the lines **z** and **x_3** (inputs of F4) are proved fault free If the test group for F2 fails, the faults in F2 can be diagnozed locally

Boolean Differentials and Fault Diagnosis

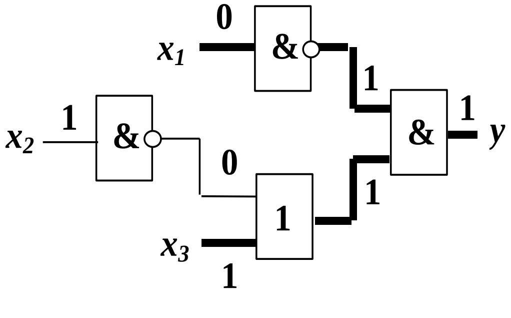
$$y = \overline{x}_1(\overline{x}_2 \vee x_3) \qquad dy = y \oplus (\overline{x}_1 \oplus dx_1)((\overline{x}_2 \oplus dx_2) \vee (x_3 \oplus dx_3))$$

Diagnostic experiment:

$x_1 = 0$ $x_2 = 1$ $x_3 = 1$ $dy = 0$

x_1, x_2, x_3 } **Test pattern**

dy - **Correct reaction**



Substitution of values:

$$dy = 1 \oplus \overline{dx}_1(dx_2 \vee \overline{dx}_3) = 0$$

$$dy = \overline{dx}_1(dx_2 \vee \overline{dx}_3) = 1$$

Adjusting for SAF faults:

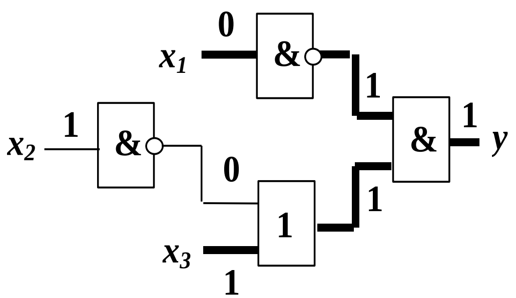
$$\overline{dx}_1^0(dx_2^1 \vee \overline{dx}_3^1) = 1$$

Partial diagnosis: $\overline{dx}_1^0 = 1$

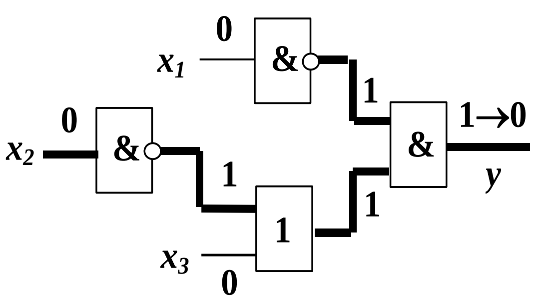
Boolean Differentials and Fault Diagnosis

$$y = \bar{x}_1(\bar{x}_2 \vee x_3) \qquad dy = y \oplus (\bar{x}_1 \oplus dx_1)((\bar{x}_2 \oplus dx_2) \vee (x_3 \oplus dx_3))$$

Two diagnostic experiments:



$x_1 = 0$
$x_2 = 1$
$x_3 = 1$
$dy = 0$



$x_1 = 0$
$x_2 = 0$
$x_3 = 0$
$dy = 1$

1) Correct output signal:

$$\overline{dx_1}^0 (dx_2^1 \vee \overline{dx_3}^1) = 1 \quad \rightarrow \quad \boxed{\overline{dx_1}^0 = 1}$$

2) Erroneous output signal:

$$dy = 1 \oplus \overline{dx_1}(\overline{dx_2} \vee dx_3) = 1$$

$$dx_1^0 \vee dx_2^0 \overline{dx_3}^0 = 1$$

Diagnosis from two experiments:

$$\overline{dx_1}^0 (dx_2^1 \vee \overline{dx_3}^1)(dx_1^0 \vee dx_2^0 \overline{dx_3}^0) = 1$$

Boolean Differentials and Fault Diagnosis

Diagnosis from two experiments:

$$\overline{dx_1}^0 (dx_2^1 \vee \overline{dx_3}^1)(dx_1^0 \vee dx_2^0 \overline{dx_3}^0) = 1$$

Rule: $\overline{dx_k}^0 dx_k^0 = 0$

$$\overline{dx_1}^0 (dx_2^1 dx_2^0 \overline{dx_3}^0 \vee dx_2^0 \overline{dx_3}^0 \overline{dx_3}^1) = 1$$

= 0

Rule: $dx_k^0 dx_k^1 = 0$

Final diagnosis:

$$\overline{dx_1}^0 dx_2^0 \overline{dx_3}^0 \overline{dx_3}^1 = 1$$

The line x_3 works correctly

There is a fault: $x_2 \equiv 1$

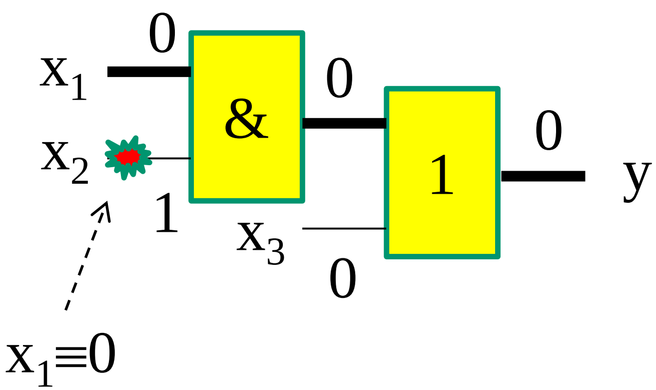
The fault $x_1 \equiv 1$ **is missing**

Question:

1) Which question regarding the possible present faults is still open?

Boolean Differentials and Fault Diagnosis

$$y = x_1 x_2 \vee x_3 \qquad dy = y \oplus (x_1 \oplus dx_1)(x_2 \oplus dx_2) \vee (x_3 \oplus dx_3)$$



We will test the fault $x_1 \equiv 1$

The test is successful

So, we conclude that x_1 is correct

Let us use now mathematics:

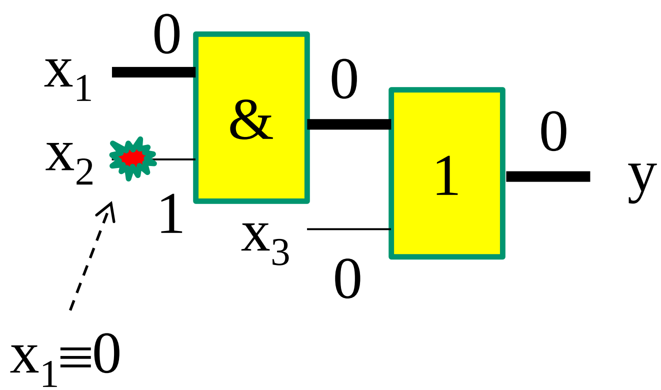
$$dy = dx_1 \overline{dx_2} \vee dx_3 = 0$$

$$(\overline{dx_1} \vee dx_2)\overline{dx_3} = 1$$

Now we are not any more very sure about x_1

Correctness Proof with Test Pairs

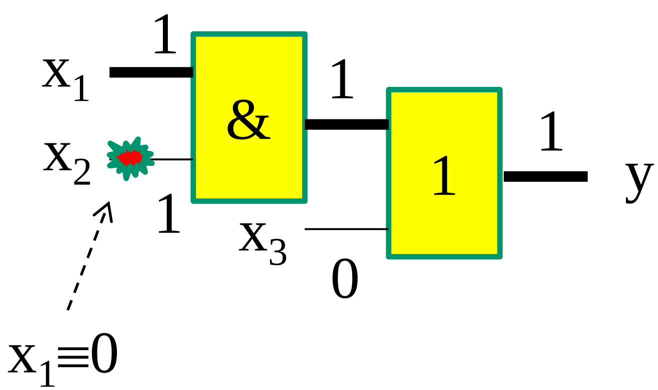
$$y = x_1 x_2 \vee x_3 \qquad dy = y \oplus (x_1 \oplus dx_1)(x_2 \oplus dx_2) \vee (x_3 \oplus dx_3)$$



We have: $dy = dx_1 \overline{dx_2} \vee dx_3 = 0$

$$(\overline{dx_1} \vee dx_2)\overline{dx_3} = 1$$

2. test: we change the value of x_1 to have a test pair



$$dy = 1 \oplus (\overline{dx_1}\,\overline{dx_2} \vee dx_3) = 0$$

$$(\overline{dx_1}\,\overline{dx_2} \vee dx_3) = 1$$

Both experiments together:

$$(\overline{dx^1{}_1}\,\overline{dx_2} \vee dx_3)(\overline{dx^0{}_1} \vee dx_2)\overline{dx_3} =$$

$$= \overline{dx^1{}_1}\,\overline{dx^0{}_1}\,\overline{dx^1{}_2}\,\overline{dx^0{}_3} = 1$$

Diagnostic Simulation of multiple faults

Algebra for diagnostic simulation of faulty signals

- **The diagnostic reasoning of the faulty subcircuit is processed using a system of Boolean differential equations (BDE), which in its turn dan be mapped into a set of the novel type of diagnostic BDDs (DBDD)**
- **The system of BDEs is solved by manipulations of DBDDs whereas the solution represents the set of candidate faults under suspicion**
- **For manipulations of Diagnostic BDDs, a 5-valued algebra was developed**

Boolean differential equation, as the model of exact fault diagnosis:

$$y = F(X)$$
$$dy = y \oplus F(X \oplus dX)$$

For finding solutions of the set of equations, novel 5-valued algebra is developed

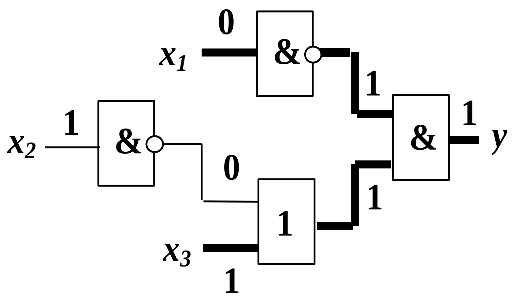
5-valued algebra					
dx	dx^0	$\overline{dx^0}$	dx^1	$\overline{dx^1}$	$\overline{dx}$
dx^0	dx^0	$\varnothing$	$\varnothing$	dx^0	$\varnothing$
$\overline{dx^0}$	$\varnothing$	$\overline{dx^0}$	dx^1	$\overline{dx}$	$\overline{dx}$
dx^1	$\varnothing$	dx^1	dx^1	$\varnothing$	$\varnothing$
$\overline{dx^1}$	dx^0	$\overline{dx}$	$\varnothing$	$\overline{dx^1}$	$\overline{dx}$
$\overline{dx}$	$\varnothing$	$\overline{dx}$	$\varnothing$	$\overline{dx}$	$\overline{dx}$

Diagnostic Simulation of multiple faults

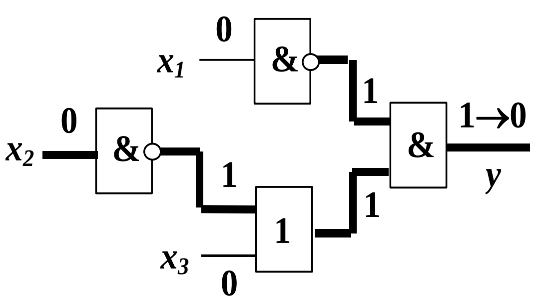
Example: $dy = y \oplus (\bar{x}_1 \oplus dx_1)((\bar{x}_2 \oplus dx_2) \vee (x_3 \oplus dx_3))$

Two diagnostic experiments:

$y = \bar{x}_1(\bar{x}_2 \vee x_3)$



$x_1 = 0$
$x_2 = 1$
$x_3 = 1$
$dy = 0$



$x_1 = 0$
$x_2 = 0$
$x_3 = 0$
$dy = 1$

1) Correct output $dy = 0$

$\overline{dx_1}^0 (dx_2^1 \vee \overline{dx_3}^1) = 1 \quad \rightarrow \quad \boxed{\overline{dx_1}^0 = 1}$

2) Error $dy = 0$

$dy = 1 \oplus \overline{dx_1}(\overline{dx_2} \vee dx_3) = 1$

$dx_1^0 \vee dx_2^0 \overline{dx_3}^0 = 1$

Diagnosis from two experiments:

$\overline{dx_1}^0 (dx_2^1 \vee \overline{dx_3}^1)(dx_1^0 \vee dx_2^0 \overline{dx_3}^0) = 1$

Algebra:

$\overline{dx_k}^0 dx_k^0 = 0$

$dx_k^0 dx_k^1 = 0$

Final solution:

$\overline{dx_1}^0 dx_2^0 \overline{dx_3}^0 \overline{dx_3}^1 = 1$

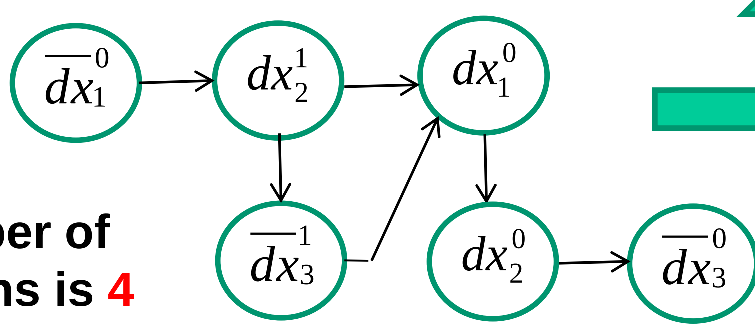
Fault: $x_2 \equiv 1$

Correct: x_3

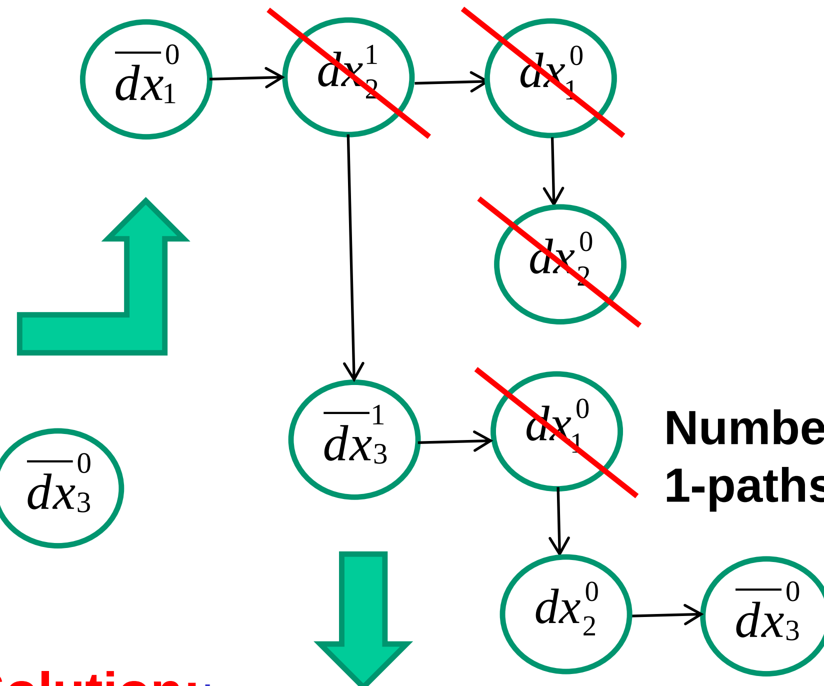
Diagnostic Simulation of multiple faults

Diagnosis from two test experiments:

$$\overline{dx_1}^0 (dx_2^1 \vee \overline{dx_3}^1)(dx_1^0 \vee dx_2^0 \overline{dx_3}^0) = 1$$



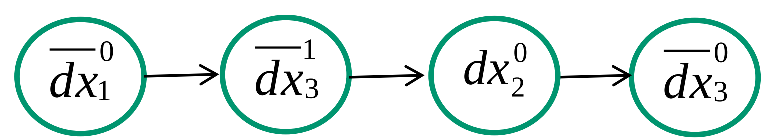
Number of 1-paths is 4



Number of 1-paths is 1

Solving a system of Boolean differential equations with SSBDDs

Solution::



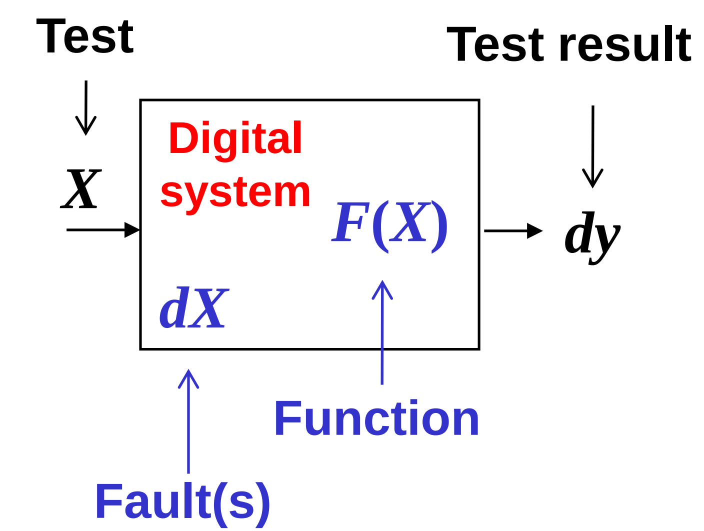
$$\overline{dx_1}^0 \overline{dx_3}^1 dx_2^0 \overline{dx_3}^0 = 1$$

Diagnostic Equation

Digital circuit: $X \Rightarrow$ $y=F(X)$ $\quad y = \overline{x}_1(\overline{x}_2 \vee x_3)$ $\Rightarrow y$

$X,dX \Rightarrow$ **Diagnostic model:** **Full diferential equation** $dy=F(X,dX)$ $\Rightarrow dy$

$$dy = y \oplus (\overline{x}_1 \oplus dx_1)((\overline{x}_2 \oplus dx_2) \vee (x_3 \oplus dx_3))$$

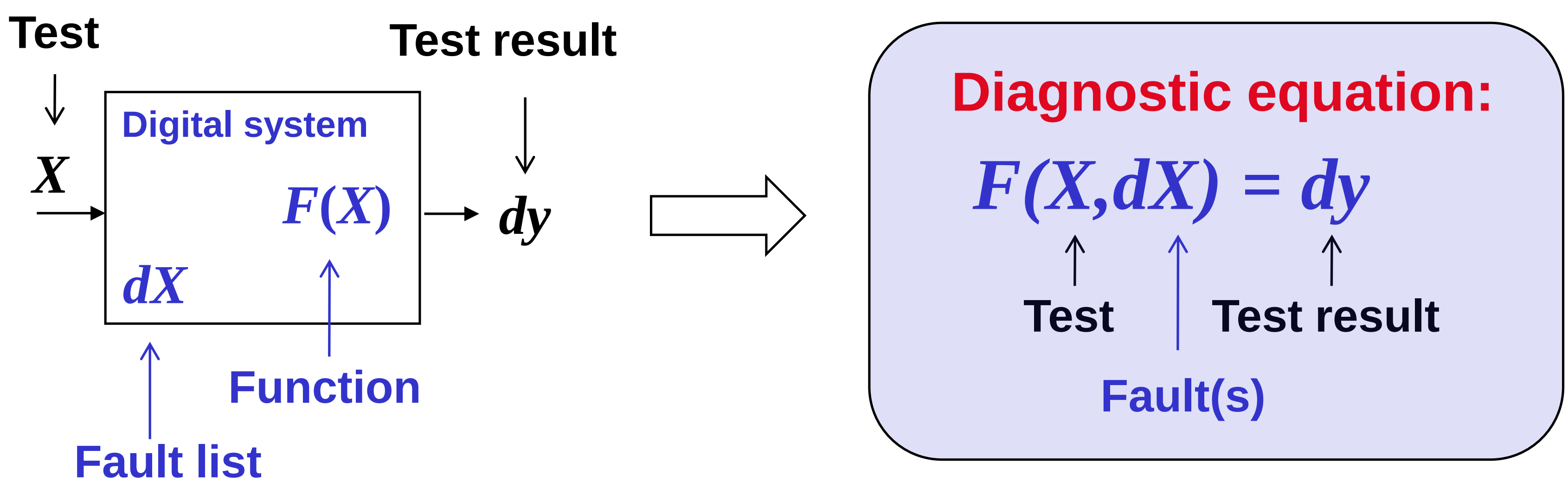


Diagnostic equation:

$$F(X,dX) = dy$$

X ↑ Test; dX ↑ Fault(s); dy ↑ Test result

How the Test Tasks are Related



Task	Given		Find
Test generation	dx	dy = 1	X
Fault diagnosis	X	dy	{dx}
Fault simulation	X	dy = 1	{dx}

Special case of fault diagnosis

Both equations can be presented as SAT tasks