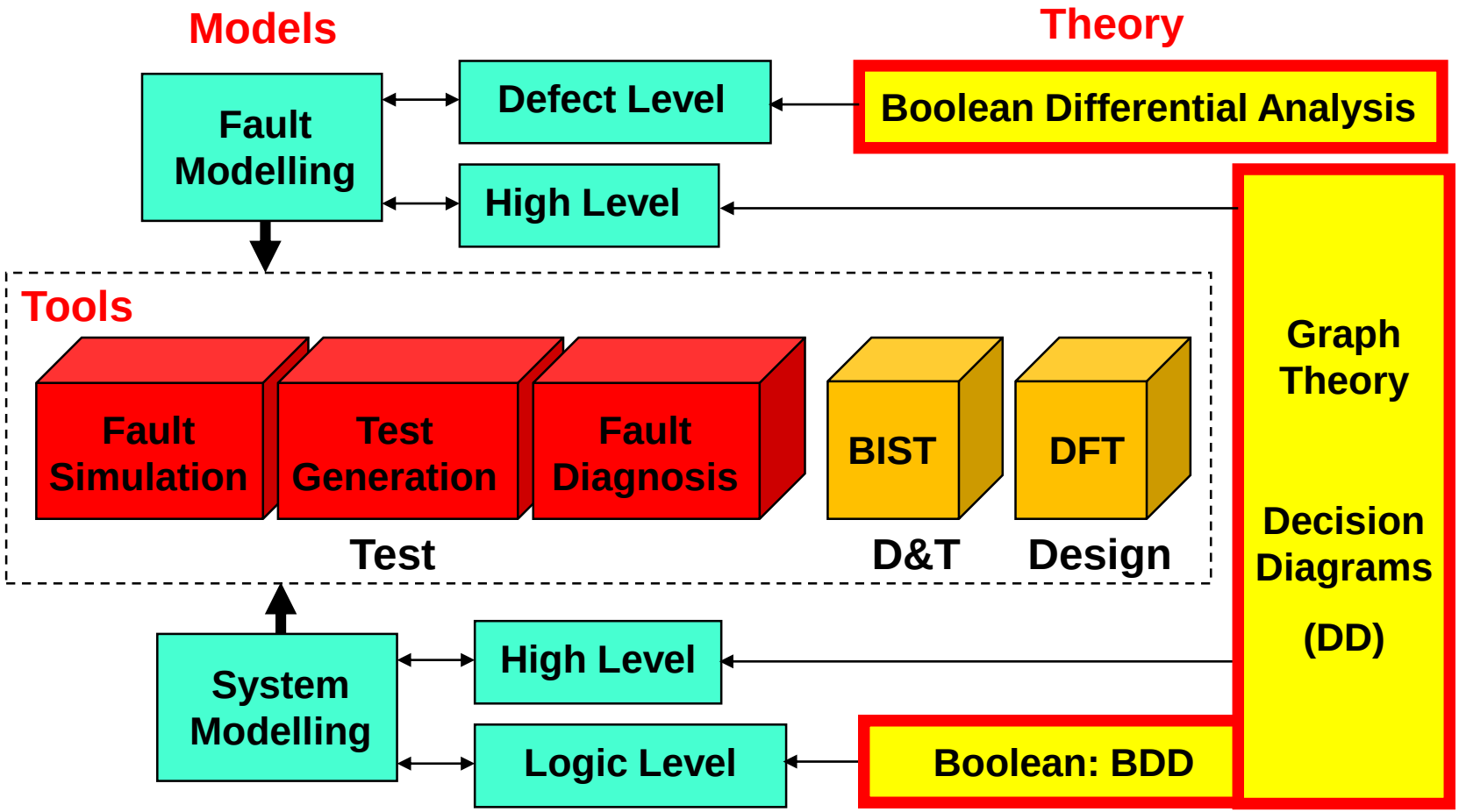
Süsteemide diagnostika

2. Teoreetilised alused

- 2.1. Boole'i differentsiaalalgebra
- 2.2. Binaarsed otsustusdiagrammid (BDD)
- 2.3. Kõrgtasandi otsustusdiagrammid

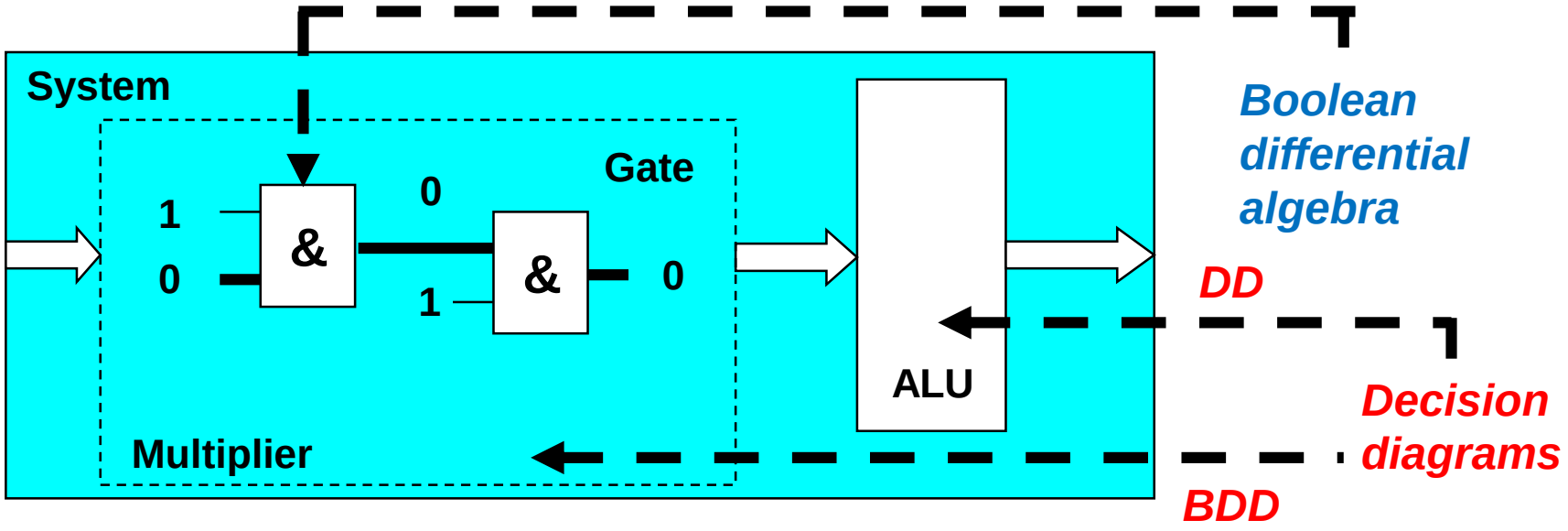
Introduction to Theories: The Course Map



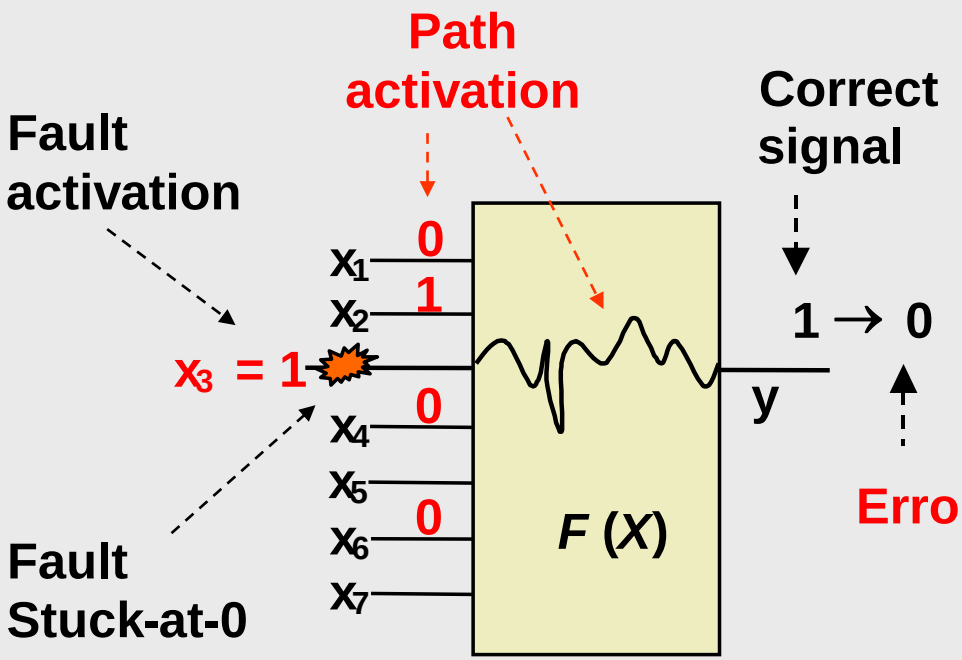
How to Go Beyond the Boolean World?

Two basic tasks:

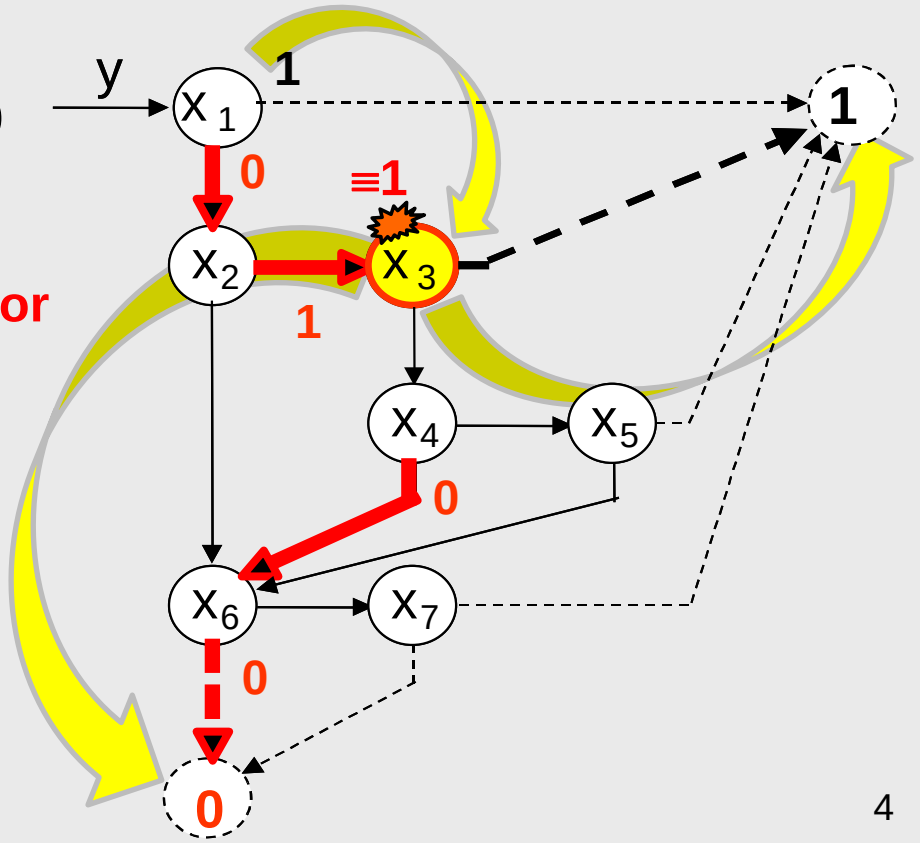
1. **Which test patterns are needed to detect a fault (or all faults)**
2. **Which faults are detected by a given test (or by all tests)**



BDDs and Testing of Logic Circuits



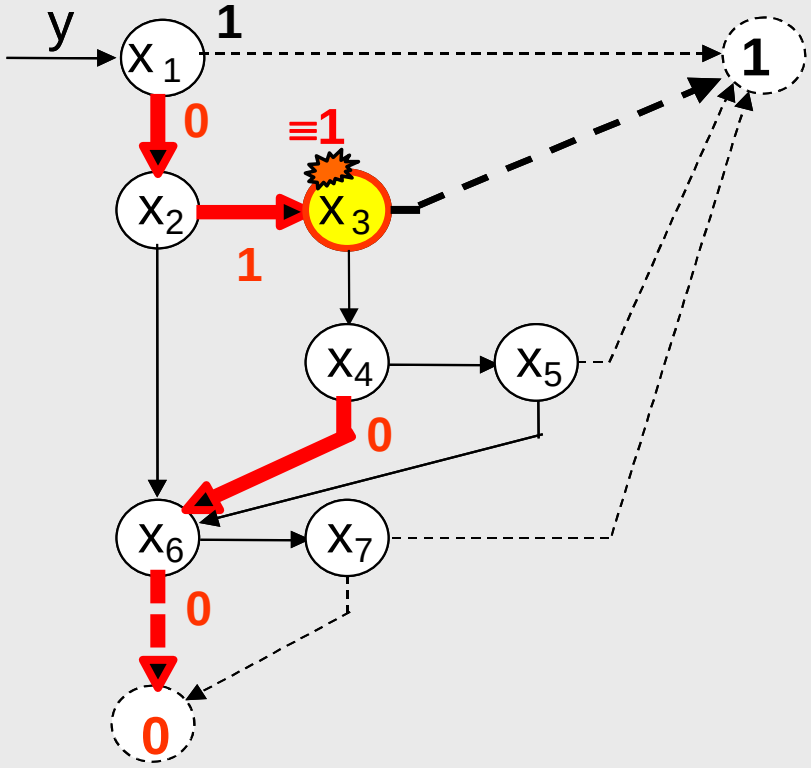
Topological view on Binary Decision Diagrams:



$$y = x_1 \vee x_2(x_3 \vee x_4 x_5) \vee x_6 x_7$$

Three interpretations of BDDs

1) BDD as a binary program:



Applicable only for simulation of input patterns

$$y = x_1 \vee x_2(x_3 \vee x_4 x_5) \vee x_6 x_7$$

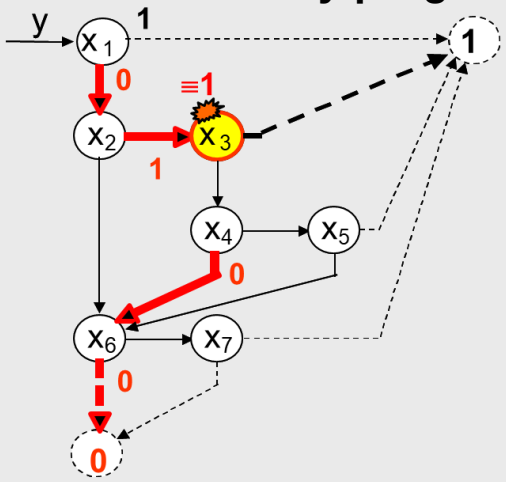
2) BDD as adata structure:

Components		Relations	
Node	**Var**	→	↓
1	x_1	**#1**	2
2	X_2	3	6
3	X_3	**#1**	4
4	X_4	5	6
5	X_5	**#1**	6
6	X_6	7	**#0**
7	x_7	**#1**	**#0**

Applicable for simulation, fault simulation, test generation, timing simulation, signal probability calculation… etc. for many other circuit analysis tasks

Three interpretations of BDDs

1) BDD as a binary program:

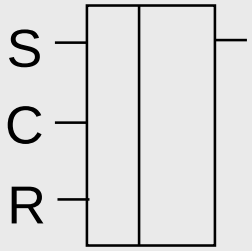


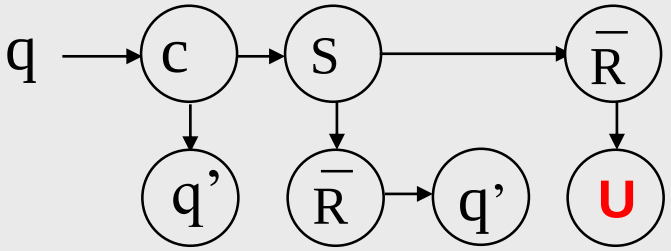
2) BDD as a data structure:

Node	Var	→	↓
1	x_1	#1	2
2	X_2	3	6
3	X_3	#1	4
4	X_4	5	6
5	X_5	#1	6
6	X_6	7	#0
7	x_7	#1	#0

3) BDD as knowledge presentation:

RS Flip-Flop





$$q = c(S \vee q'\bar{R}) \vee \bar{c}q'$$

$$SR = 0$$

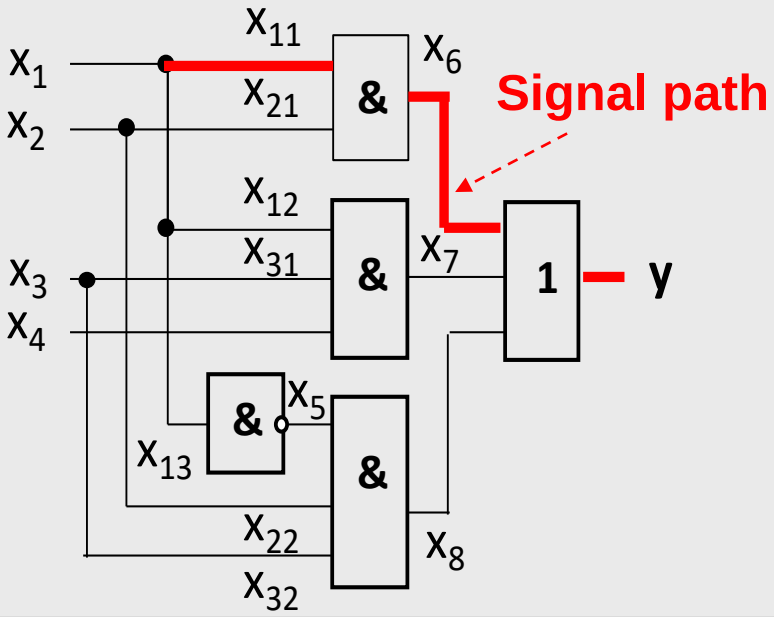
U - unknown value

The graph represents as much functional knowledge as we know about the circuit

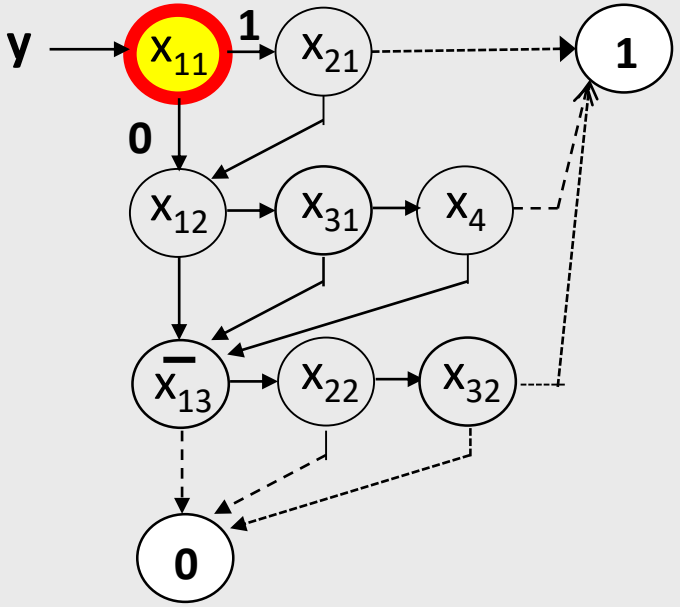
(U – indeterminism)

Mapping Between Circuit and SSBDD

4) BDD as a structural model of logic circuits



Each node in SSBDD represents a signal path:



Node x_{11} in SSBDD represents the **path** (x_1, x_{11}, x_6, y) in the circuit

The SAF-0(1) fault at the node x_{11} represents the **SAF faults** on the lines x_{11}, x_6, y in the circuit → **fault collapsing**

32 faults (16 lines) in the circuit → **16 faults** (8 nodes) in SSBDD

Test Generation with BD and BDD

BD:

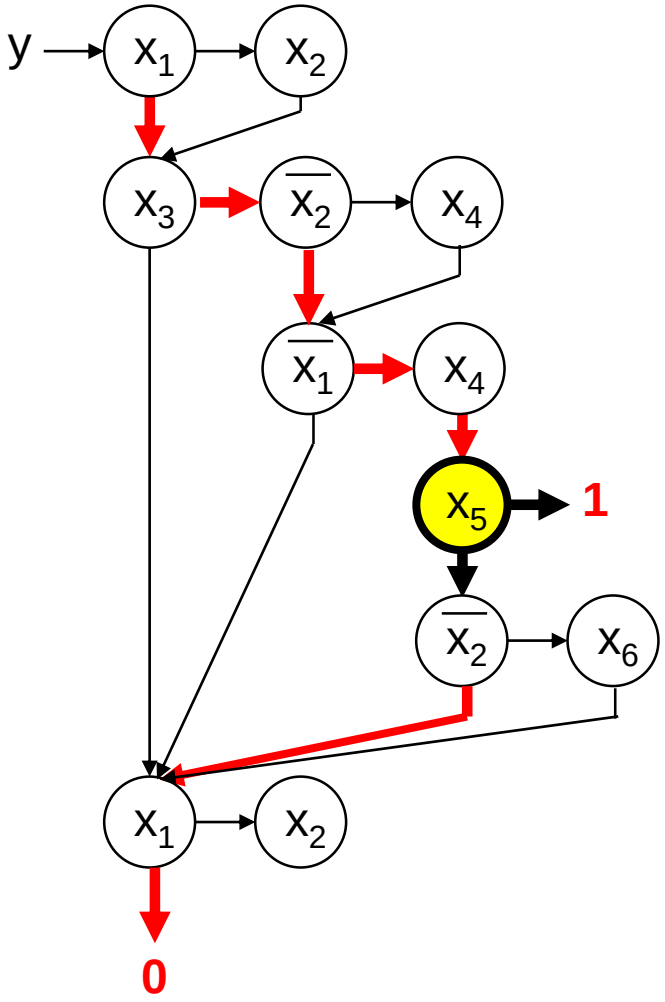
$$y = x_1x_2 \vee x_3(\overline{x_2}x_4 \vee \overline{x_1}(x_4 \vee (x_5 \vee \overline{x_2}x_6)) \vee x_1\overline{x_3}$$

$$\frac{\partial y}{\partial x_5} = \overline{(x_1x_2 \vee x_1\overline{x_3})}x_3(\overline{x_2}x_4)\overline{x_1}\,\overline{x_4}(\overline{x_2}x_6)\frac{\partial x_5}{\partial x_5} =$$

$$= (\overline{x_1} \vee \overline{x_2})(\overline{x_1} \vee x_3)x_3(x_2 \vee \overline{x_4})\overline{x_1}\,\overline{x_4}(x_2 \vee \overline{x_6}) =$$

$$= \overline{x_1}\,\overline{x_4}x_3x_2 \vee \ldots = 1$$

BDD:



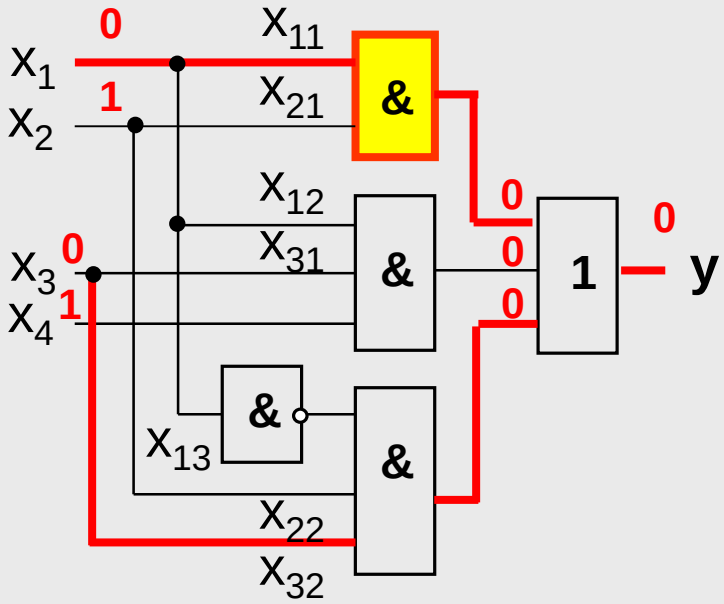
Test pattern:

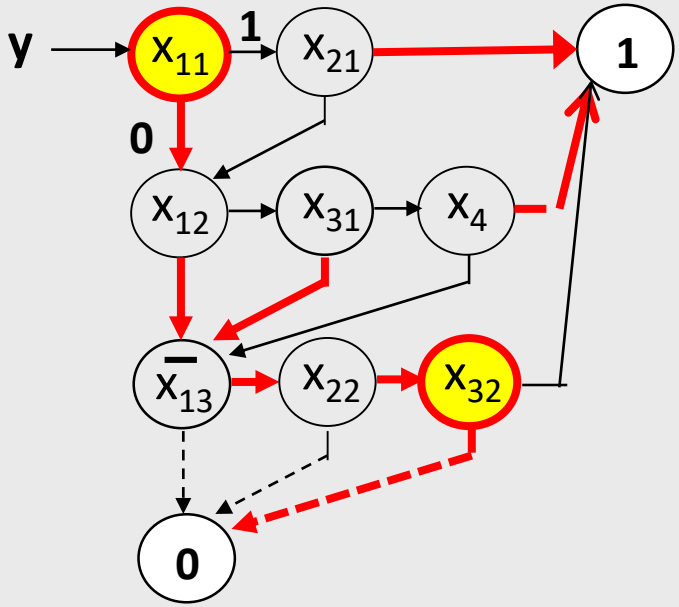
x_1	x_2	x_3	x_4	x_5	x_6	y
0	1	-	0	D	-	D

Fault Analysis with SSBDDs

Algorithm:

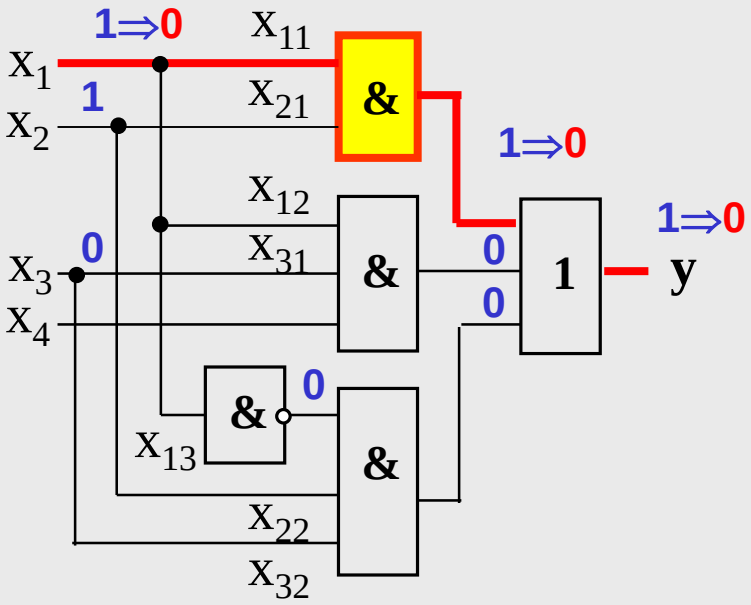
1. Determine the activated path to find the fault candidates
2. Analyze the detectability of the each candidate fault (each node represents a subset of real faults)



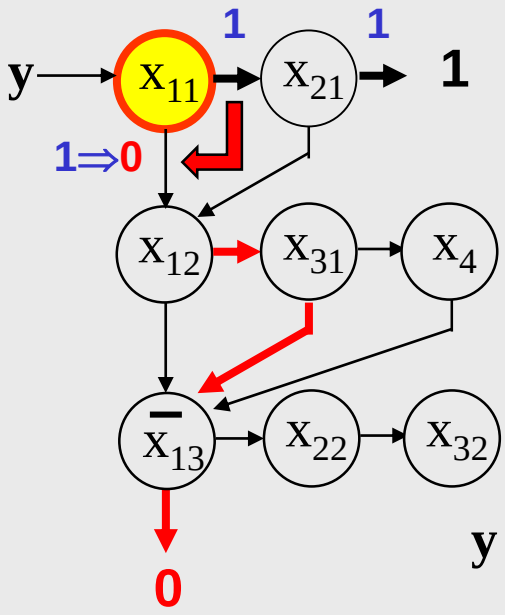


Test Generation with SSBDDs

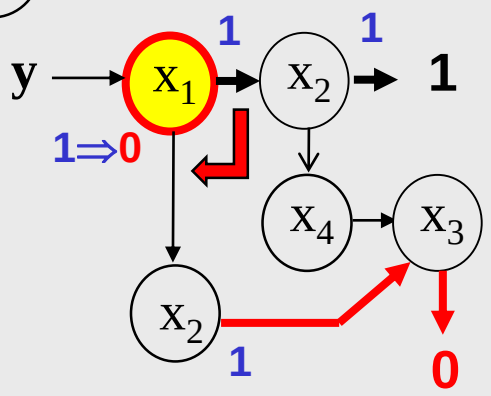
Test generation for: $x_{11}\equiv 0$



Structural BDD:



Functional BDD:



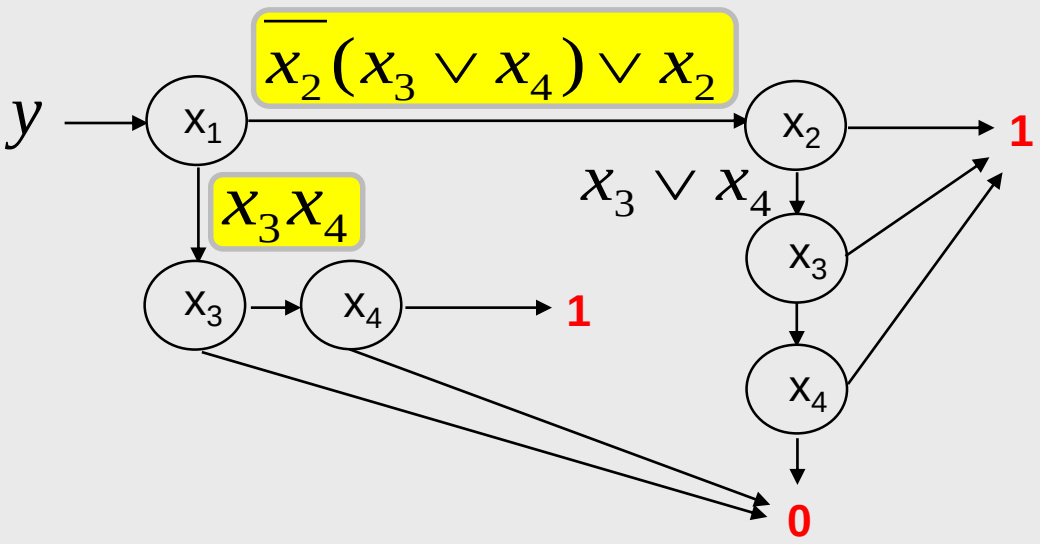
Test pattern:

x_1	x_2	x_3	x_4	y
1	1	0	-	1 ⇒0

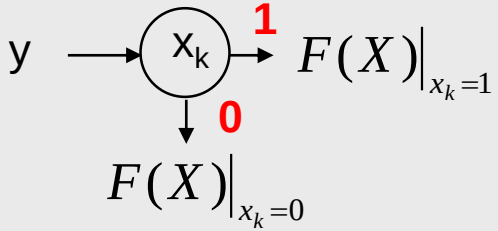
Functional Synthesis of BDDs

Shannon's Expansion Theorem: $y = F(X) = x_k F(X)\big|_{x_k=1} \vee \overline{x_k} F(X)\big|_{x_k=0}$

$$y = \underbrace{x_1\left(\overline{x_2}(x_3 \vee x_4) \vee x_2\right)}_{x_1 F(X)\big|_{x_1=1}} \vee \underbrace{\overline{x_1} x_3 x_4}_{\overline{x_1} F(X)\big|_{x_1=0}}$$



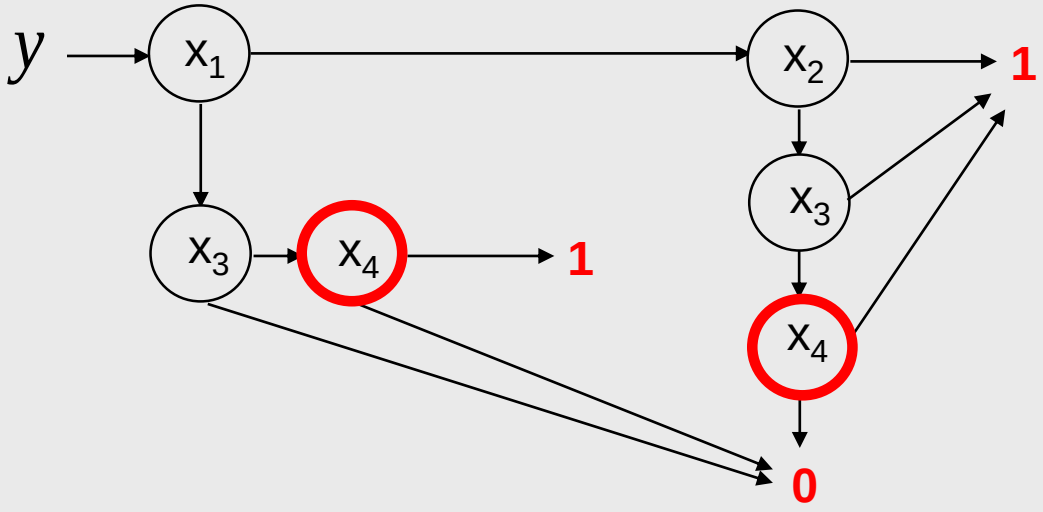
Using the Theorem for BDD synthesis:

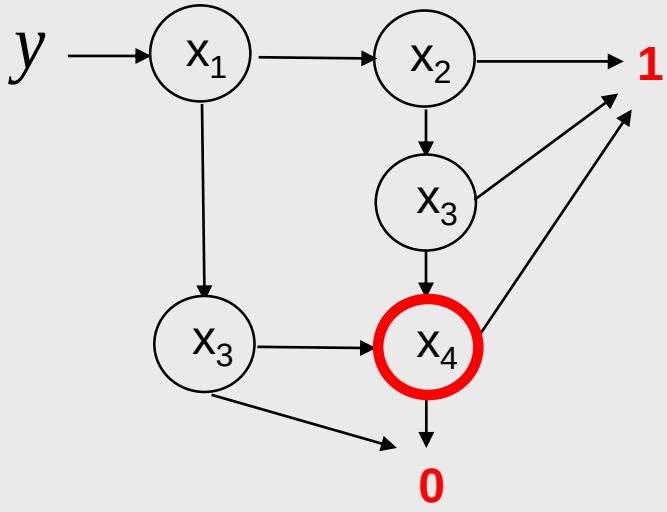


Functional Synthesis of BDDs

Shannon's Expansion Theorem: $y = F(X) = x_k F(X)\big|_{x_k=1} \vee \overline{x_k} F(X)\big|_{x_k=0}$

$$y = x_1(\overline{x_2}(x_3 \vee x_4) \vee x_2) \vee \overline{x_1} x_3 x_4$$





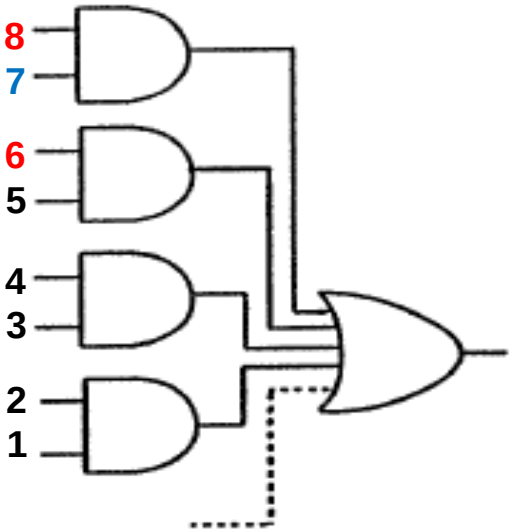
BDDs and Complexity

Optimization (by ordering of nodes): BDDs for a 2-level AND-OR circuit

BDD optimization:

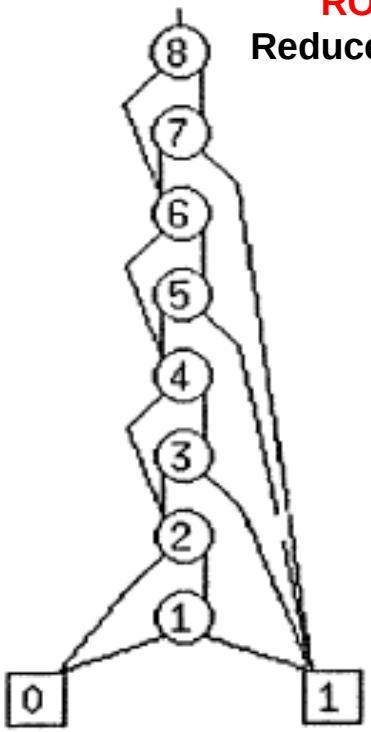
We start synthesis:

- from the most repeated variable



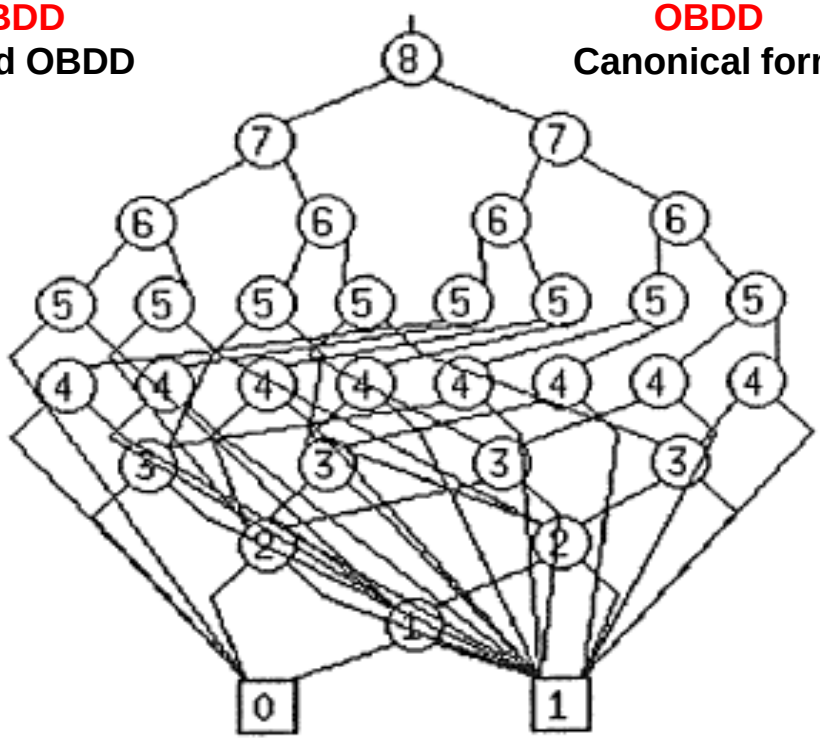
(a) Circuit.

ROBDD
Reduced OBDD



(b) In the best order.

OBDD
Canonical form

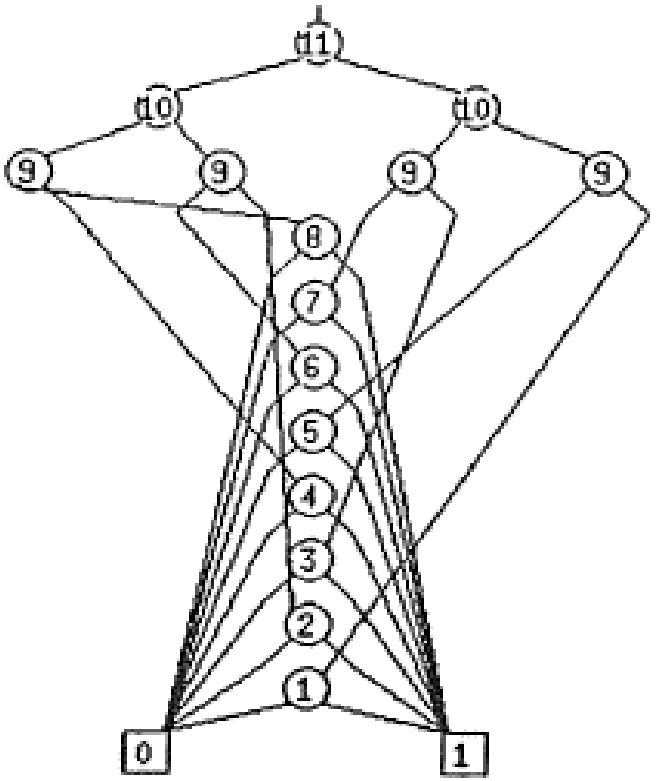


(c) In the worst order.

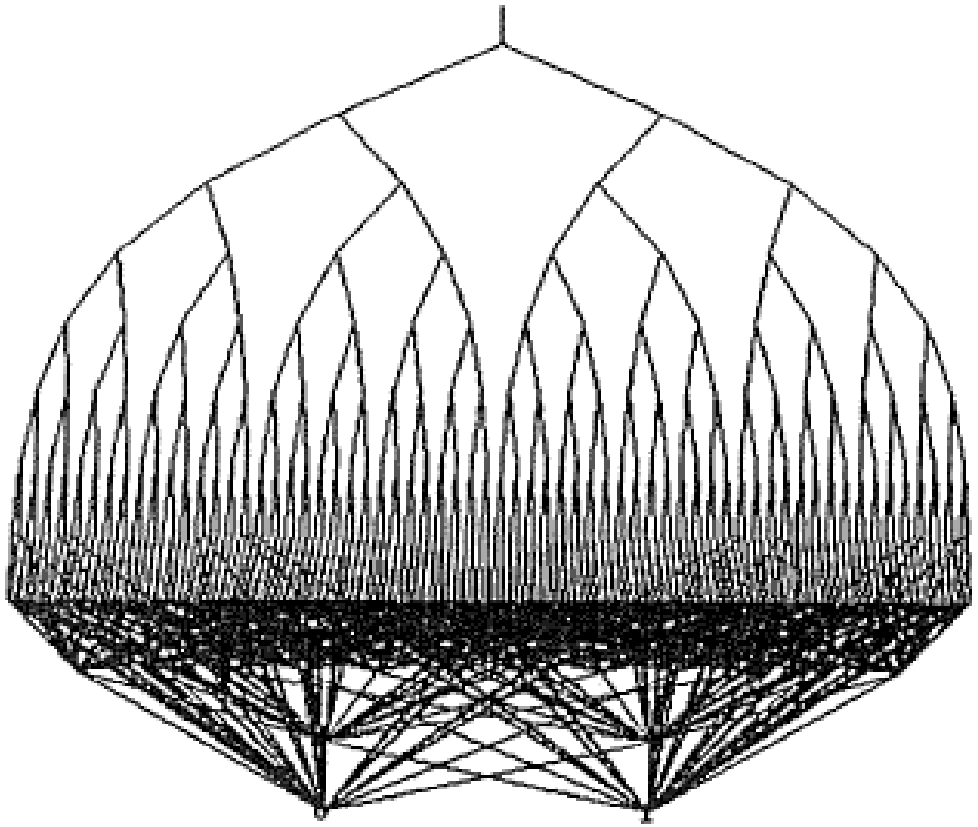
2n nodes

$2*2^n$ - 2 nodes

BDDs and Complexity



(a) In the best order.



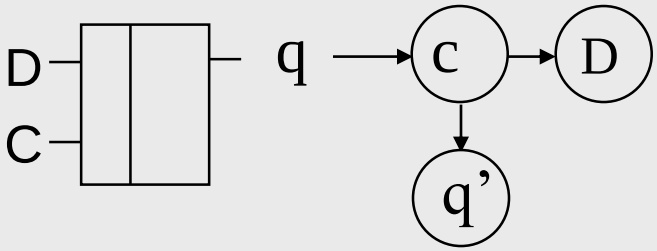
(b) In the worst order.

BDDs for an 8-bit data selector

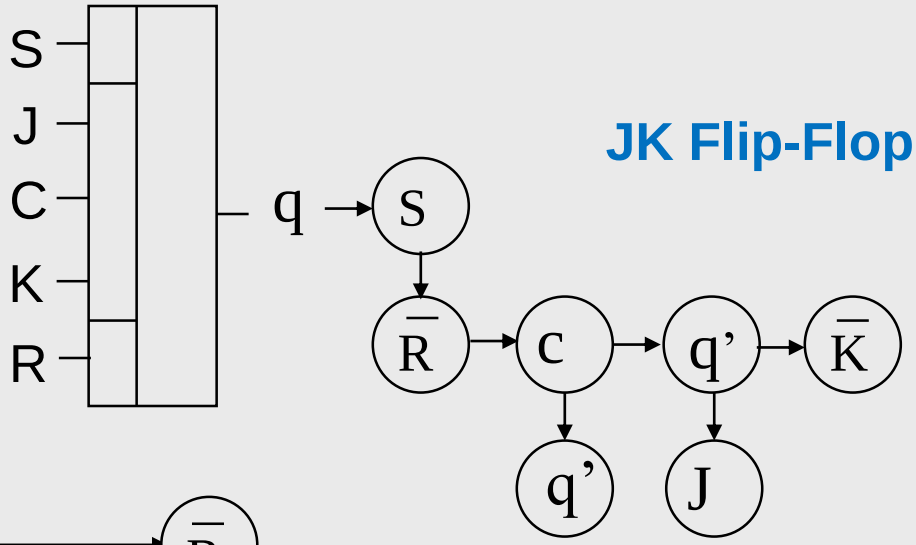
BDDs and Complexity

Elementary BDDs

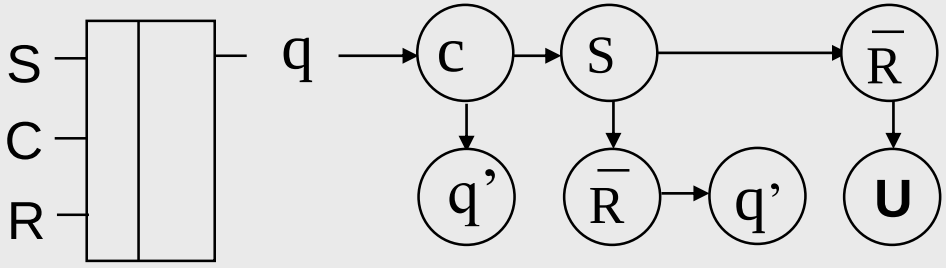
D Flip-Flop



JK Flip-Flop



RS Flip-Flop



$$q = c(S \vee q'\bar{R}) \vee \bar{c}q'$$

$$SR = 0$$

U - unknown value

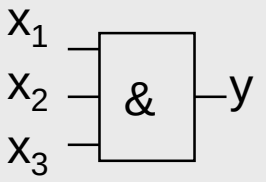
BDD optimization:

We may start synthesis:

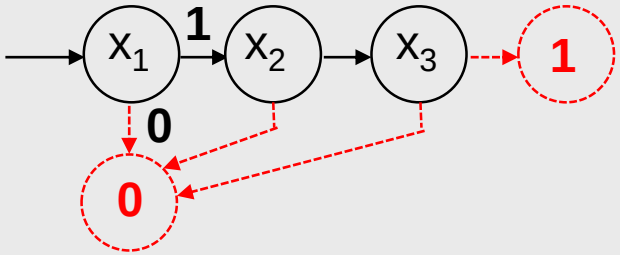
- from the **most important** variable, or
- from the **most repeated** variable

BDDs for Logic Gates

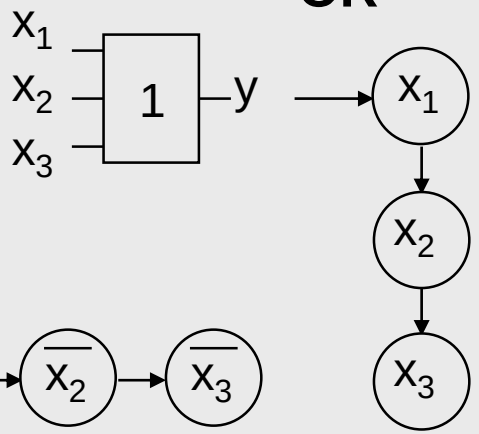
Elementary BDDs:



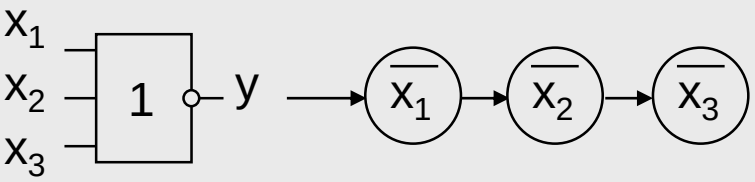
AND



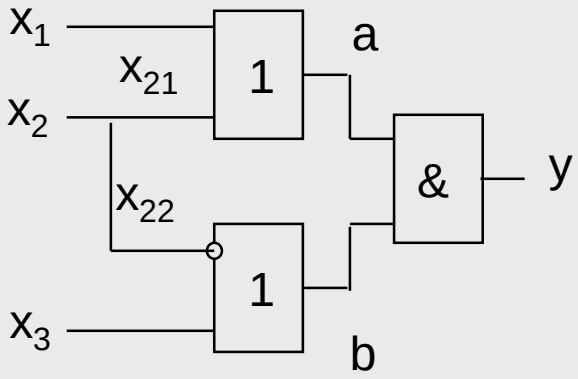
OR



NOR



Given circuit:

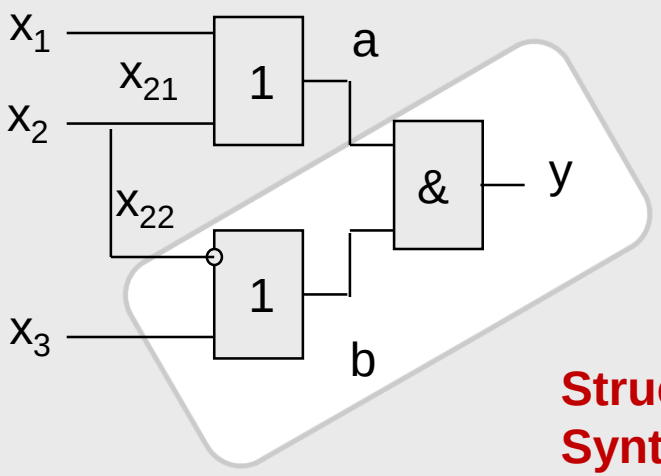


SSBDD synthesis:

SSBDDs for a given circuit are built by **superposition** of BDDs for gates

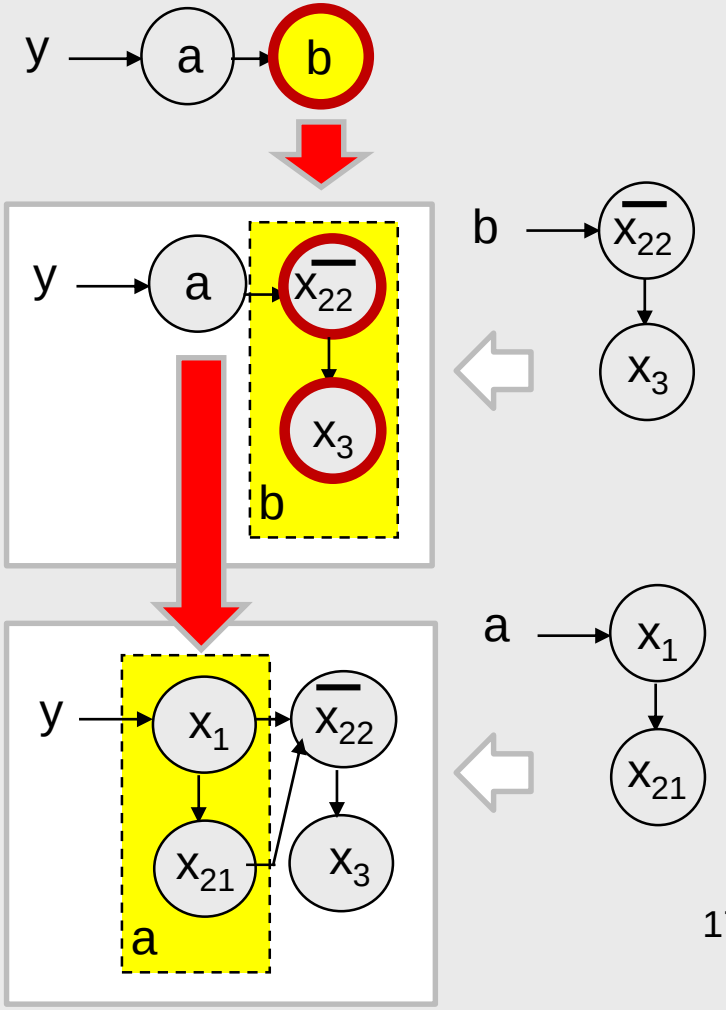
Synthesis of SSBDD for a Circuit

Given circuit:



Structurally Synthesized BDD:

Superposition of BDDs:

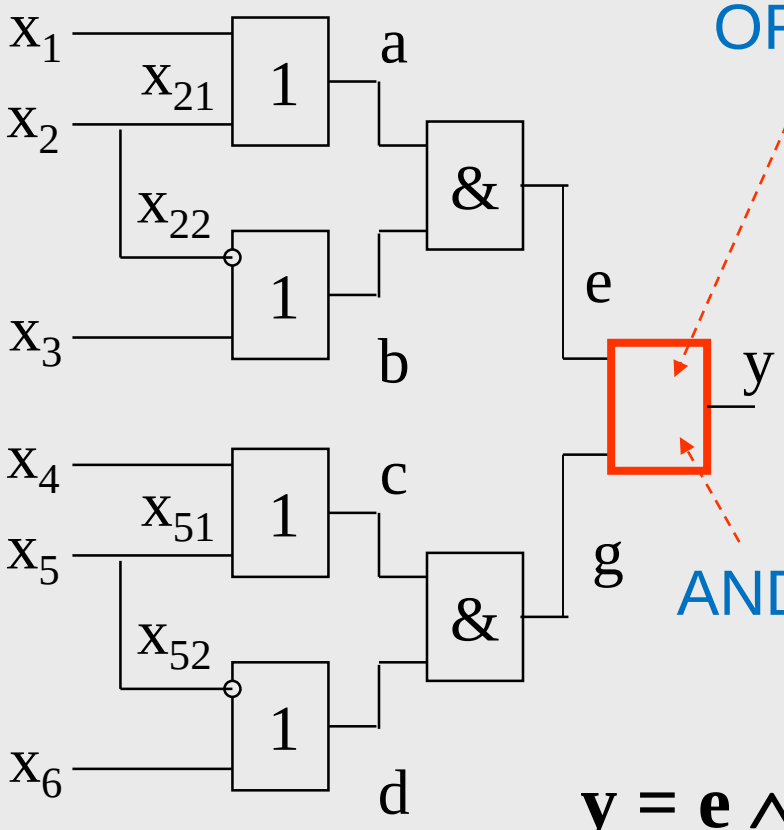


Compare to

Superposition of Boolean functions:

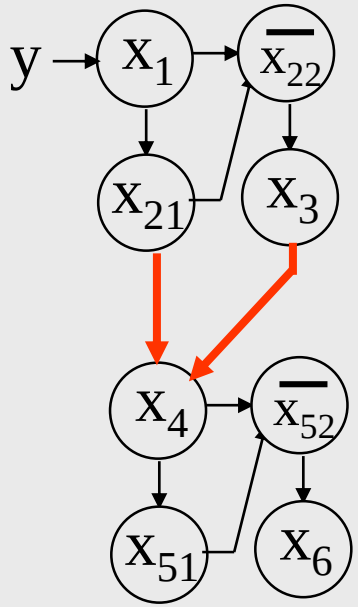
$$y = a\,\&\,b = (x_1 \vee x_{21})(\overline{x_{22}} \vee x_3)$$

Boolean Operations with SSBDDs



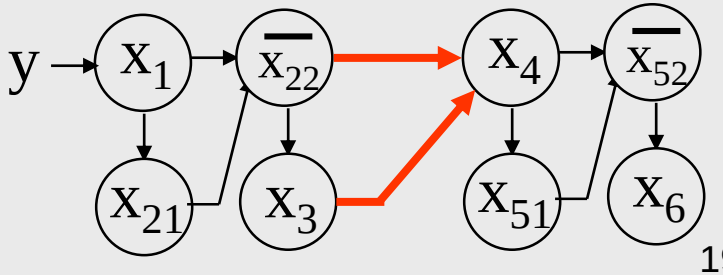
OR-operation:

$$y = e \vee g$$



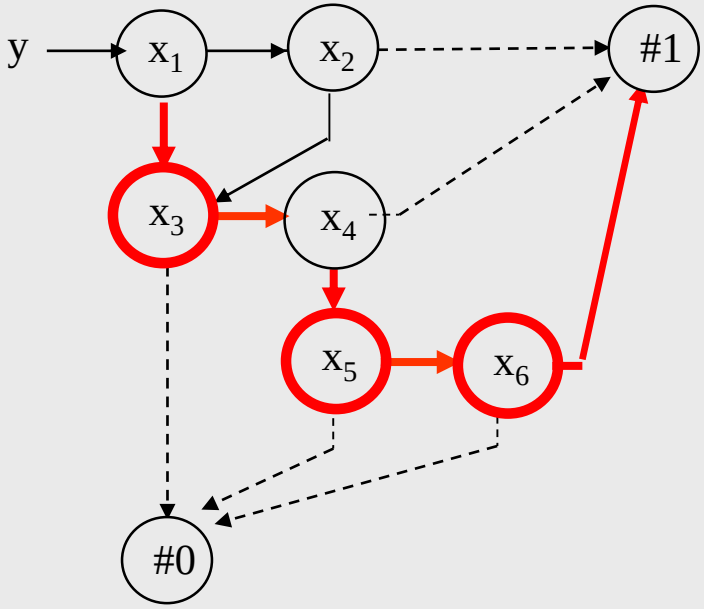
AND-operation:

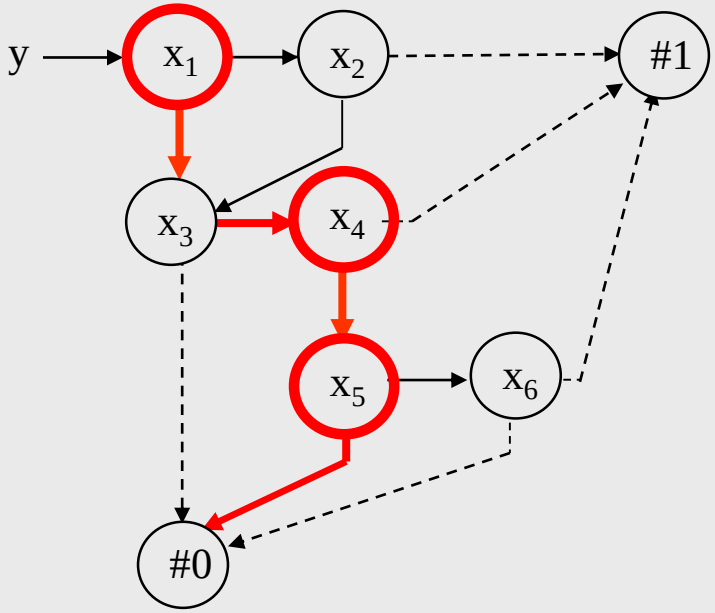
$$y = e \wedge g$$



Properties of SSBDDs

Boolean function: $y = x_1x_2 \vee x_3 (x_4 \vee x_5x_6) = x_1x_2 \vee x_3x_4 \vee \mathbf{x_3x_5x_6}$





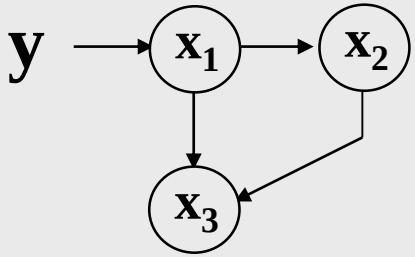
1-nodes of a **1-path** represent a term in the DNF: $\mathbf{x_3x_5x_6 = 1}$

0-nodes of a **0-path** represent a term in the CNF: $\mathbf{\overline{x_1 \vee x_4 \vee x_5} = 0}$

Boolean Operations with SSBDDs

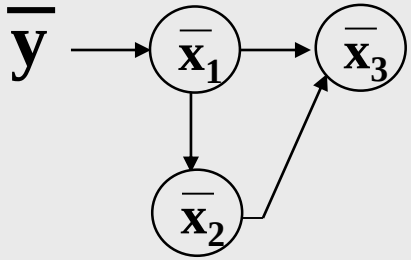
Boolean function:

$$y = x_1x_2 \vee x_3$$



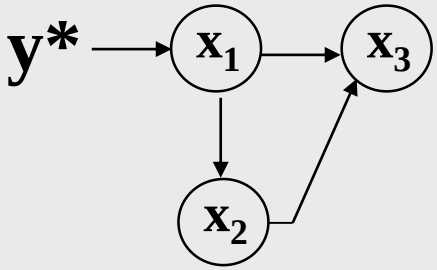
Inverted function (DeMorgan):

$$\overline{y} = \overline{x_1x_2 \vee x_3} = (\overline{x}_1 \vee \overline{x}_2)\,\overline{x}_3$$



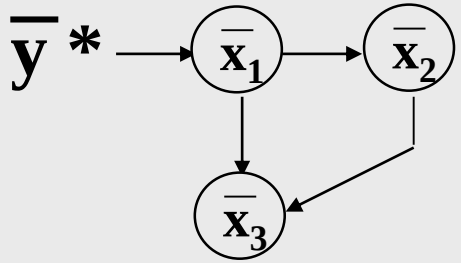
Dual function:

$$y^* = (x_1 \vee x_2)\,x_3$$



Inverted dual function:

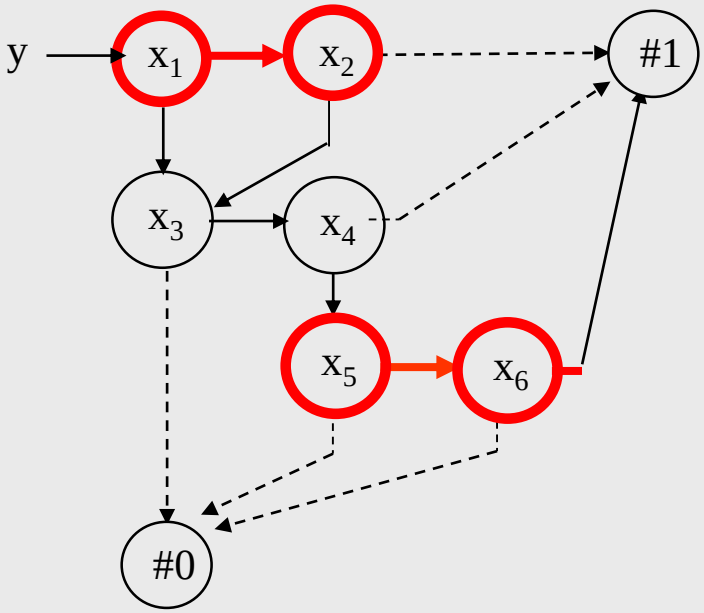
$$\overline{y}^* = \overline{x}_1\overline{x}_2 \vee \overline{x}_3$$



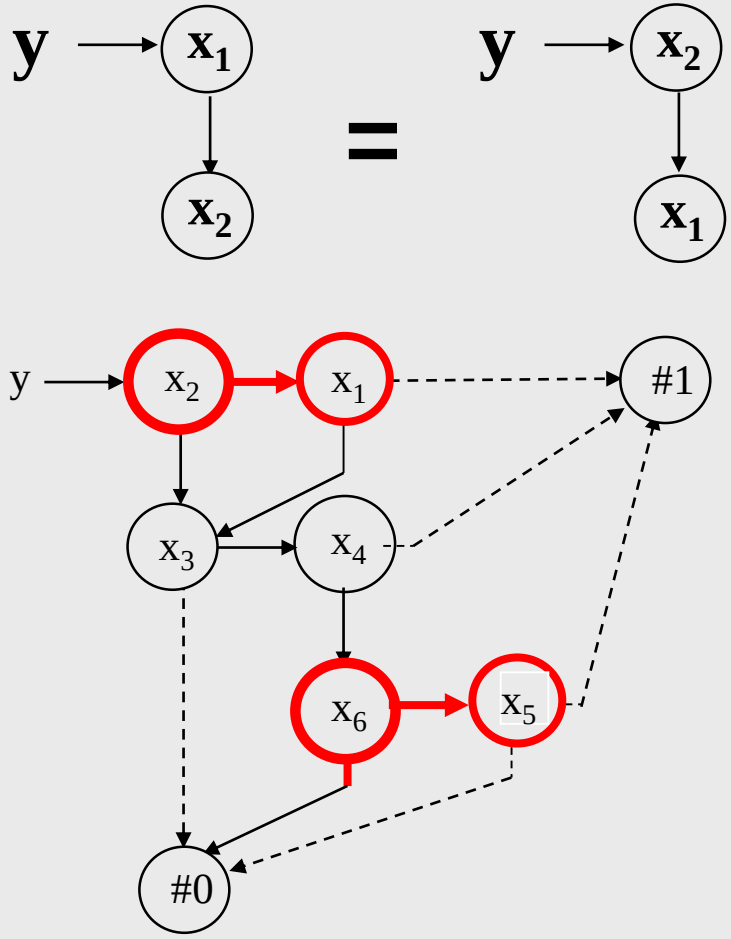
Properties of SSBDDs

Boolean function:

$y = x_1x_2 \vee x_3 (x_4 \vee x_5x_6)$



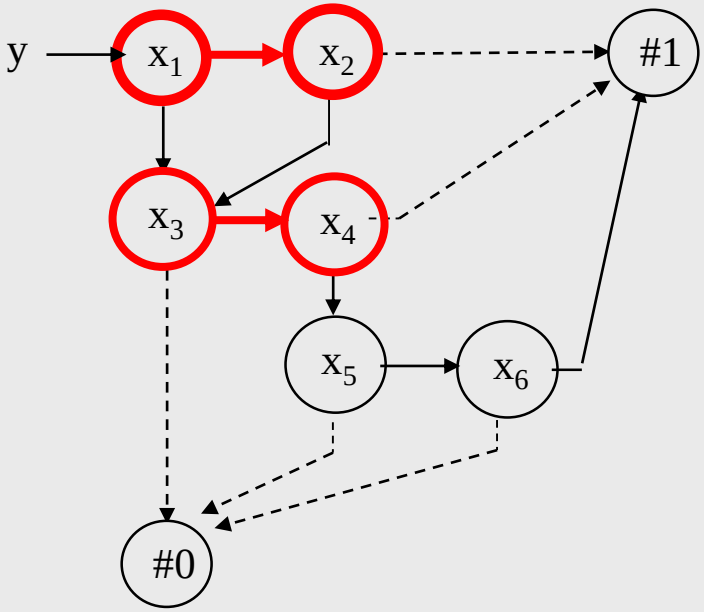
Exchange of nodes:



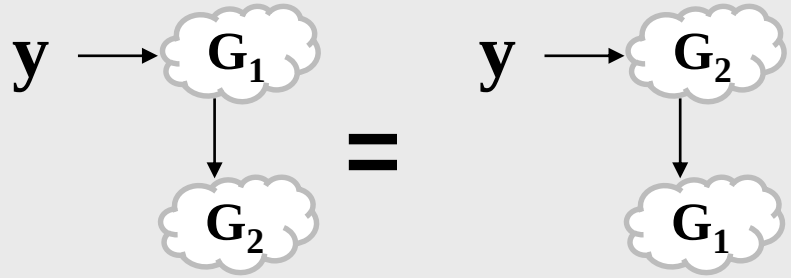
Properties of SSBDDs

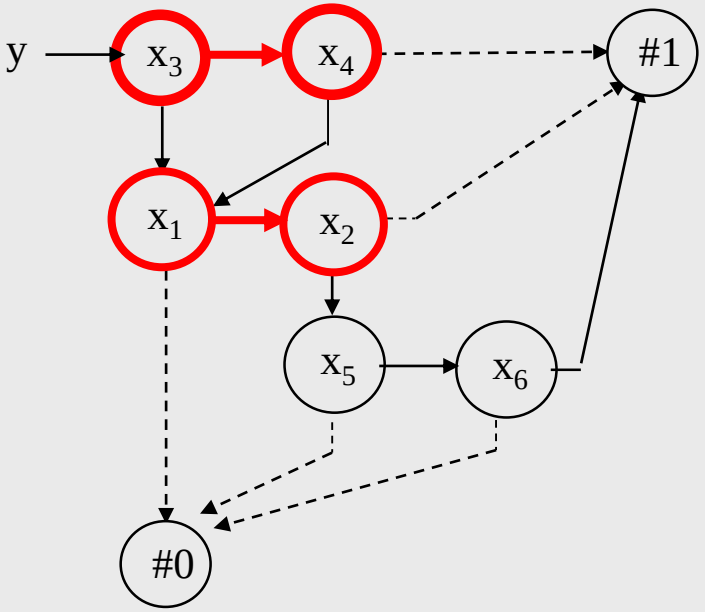
Boolean function:

$y = x_1x_2 \vee x_3 (x_4 \vee x_5x_6)$



Exchange of subgraphs:

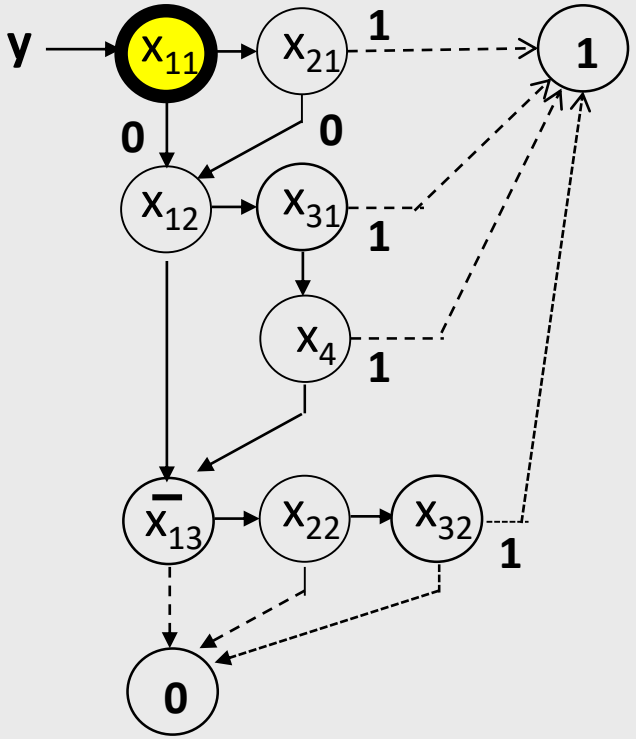


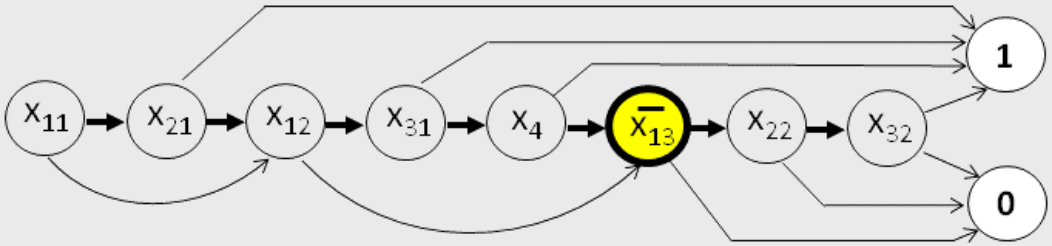


Properties of SSBDDs

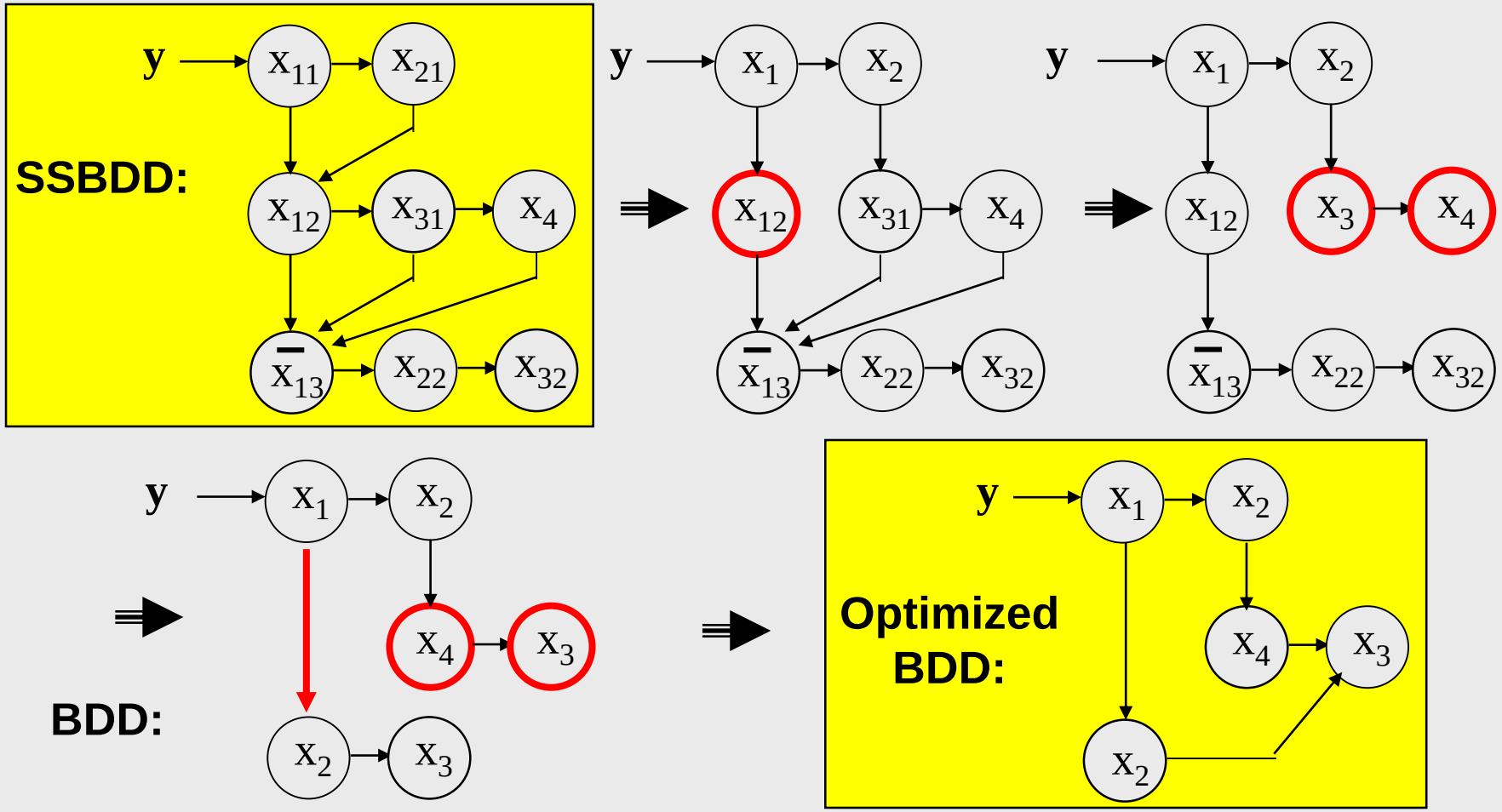
Graph related properties:

- ✓ SSBDD is
  - planar
  - asyclic
  - traceable (Hamiltonian path)
  - for every internal node there exists a 1-path and 0-path
  - homogenous



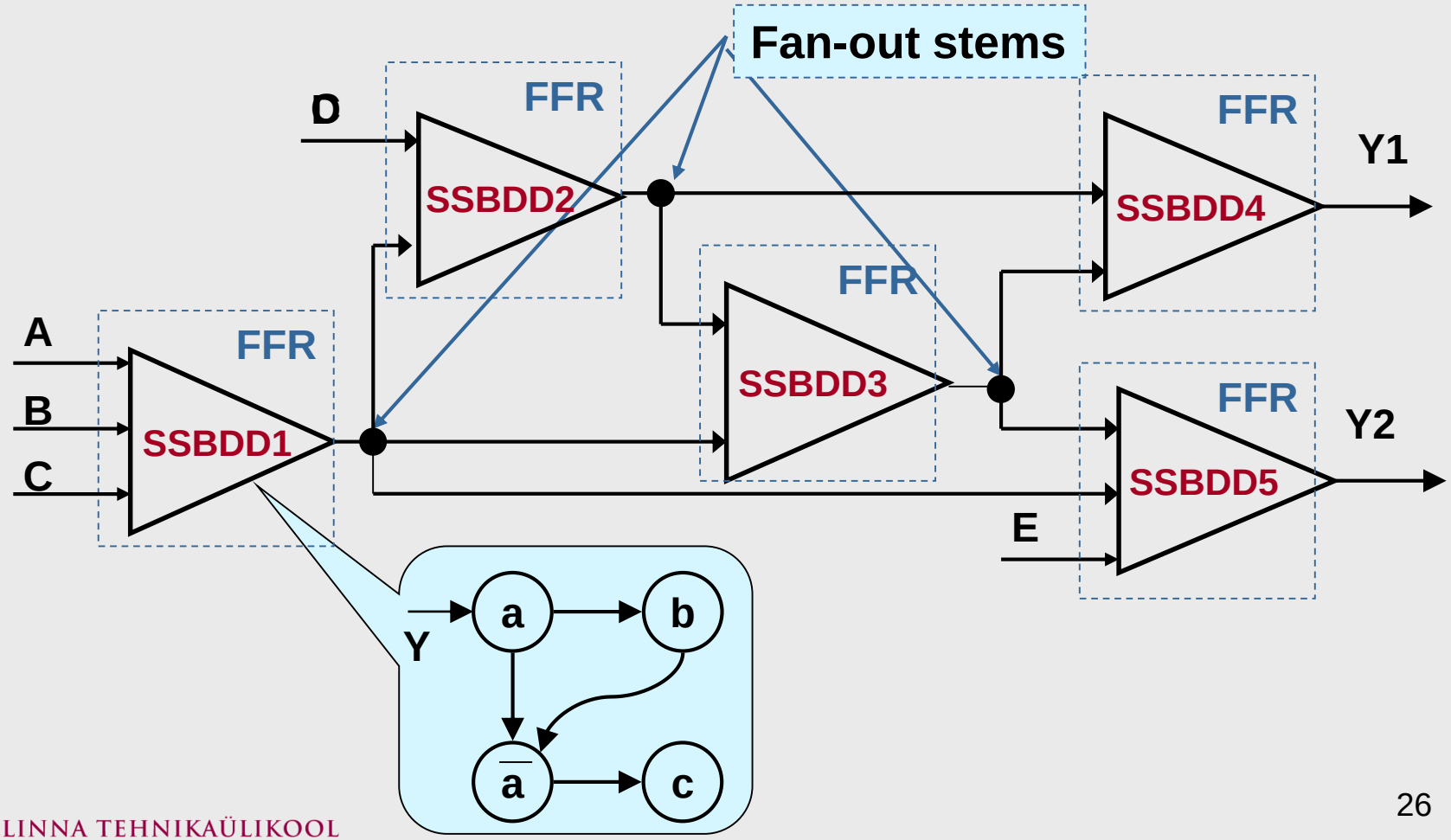


Transformation of SSBDDs to BDDs



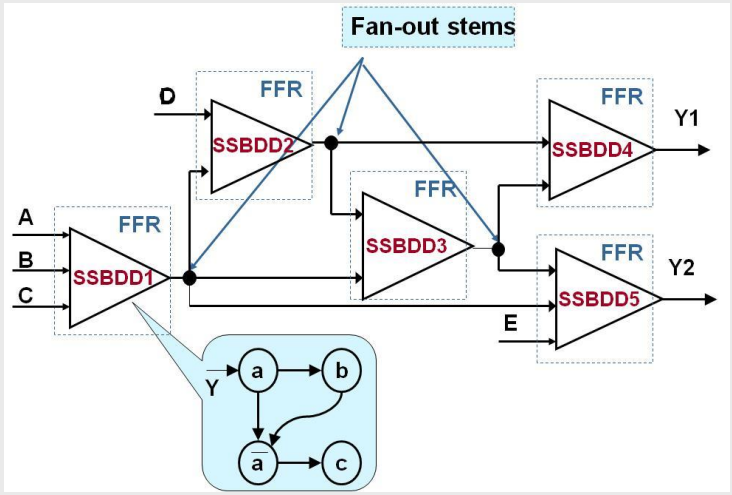
Mapping Between Circuit and SSBDD

From circuit to set of SSBDDs



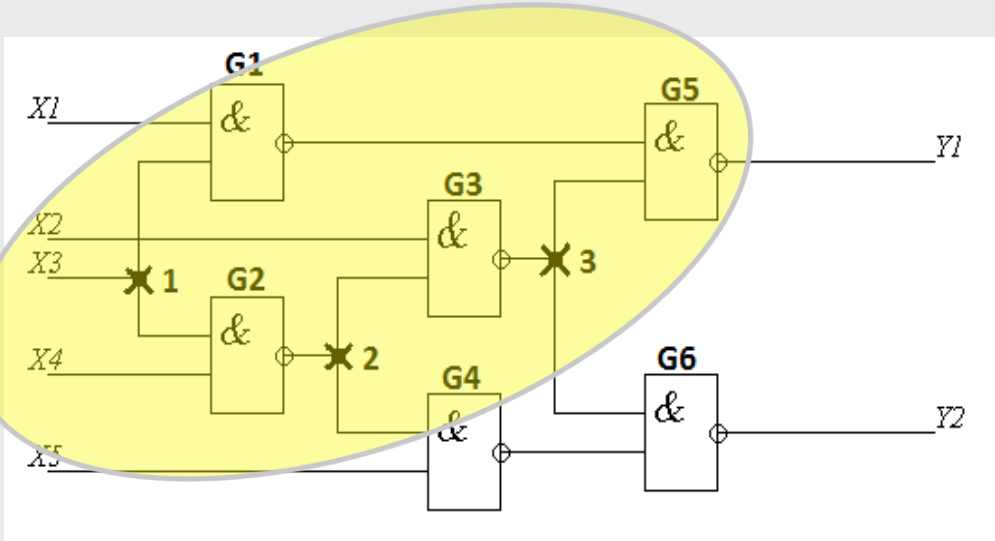
Advantages of SSBDDs

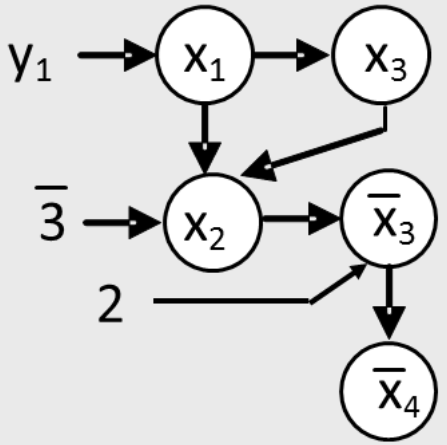
- ✓ **Linear complexity:** a circuit is represented as a system of SSBDDs, where each fanout-free region (FFR) is representred by a separate SSBDD
- ✓ **One-to-one correspondence** between the nodes in SSBDDs and signal paths in the circuit
- ✓ This allows easily to extend the logic simulation with SSBDDs to **simulation of faults** on signal paths



Shared SSBDDs - S³BDD

- ✓ Extension of superposition procedure **beyond the fan-out nodes** of the circuit
- ✓ Merging several functions in the same graphs by introducing **multiple roots**





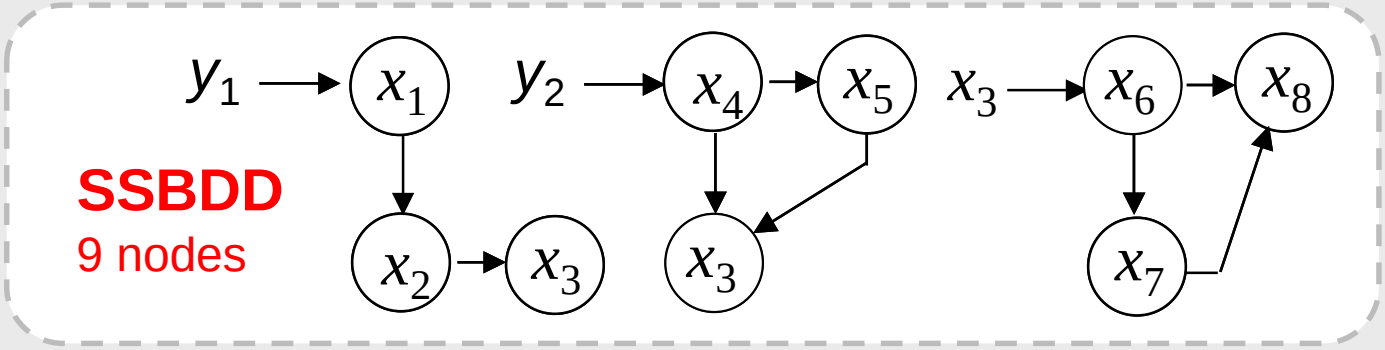
Superpositioning of FFRs

Node of SSBDD ⇒ signal path up to fan-out stem

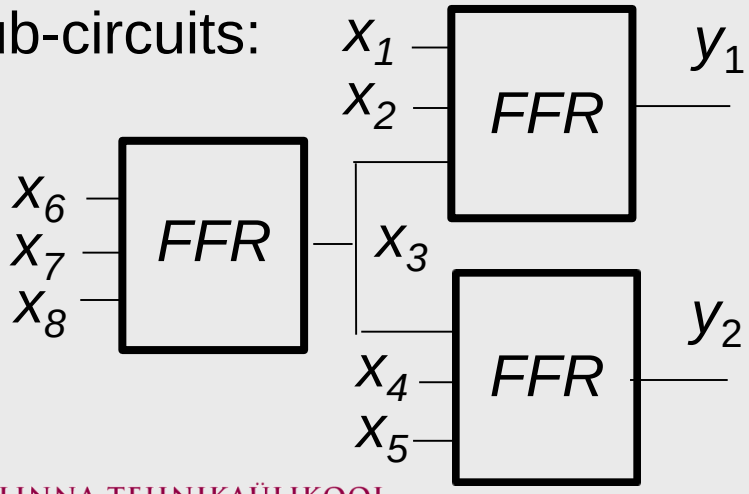
Input of SSBDD ⇒ circuit down to primary inputs

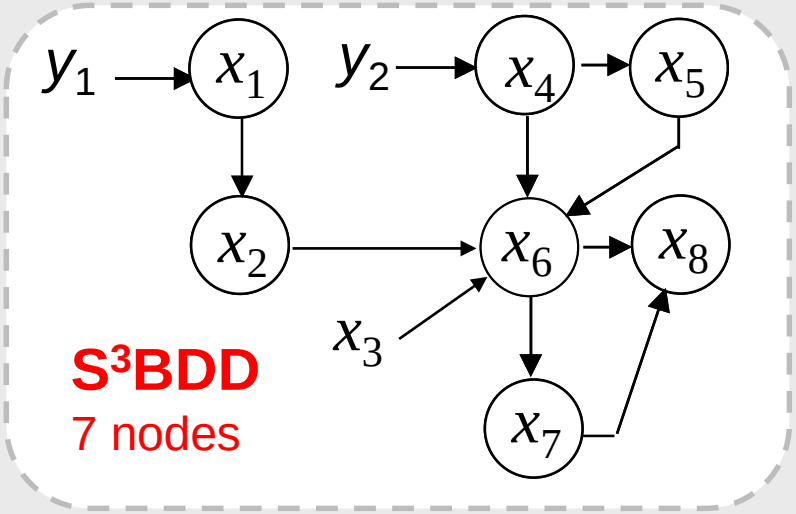
SSBDDs vs. S³BDDs

Example: Comparison of two models: SSBDD and S³BDD



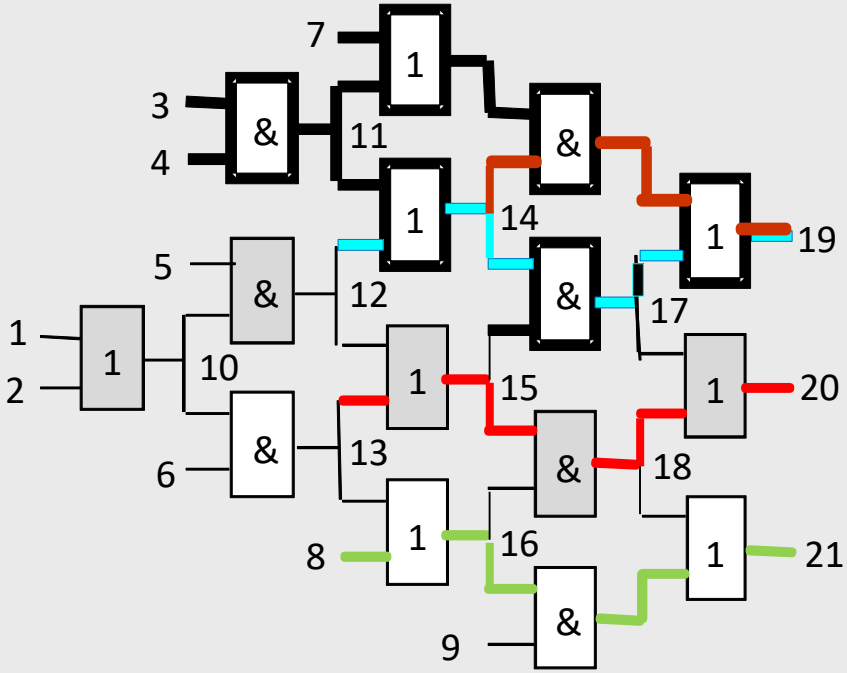
Network of
3 sub-circuits:

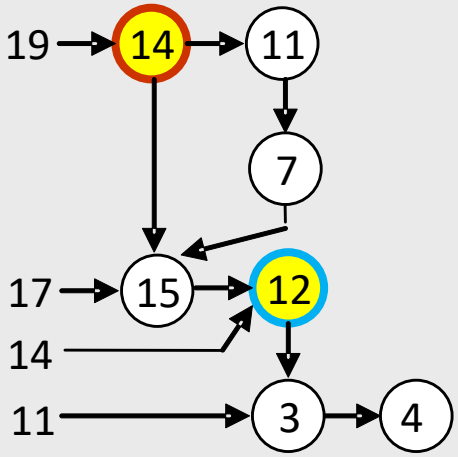


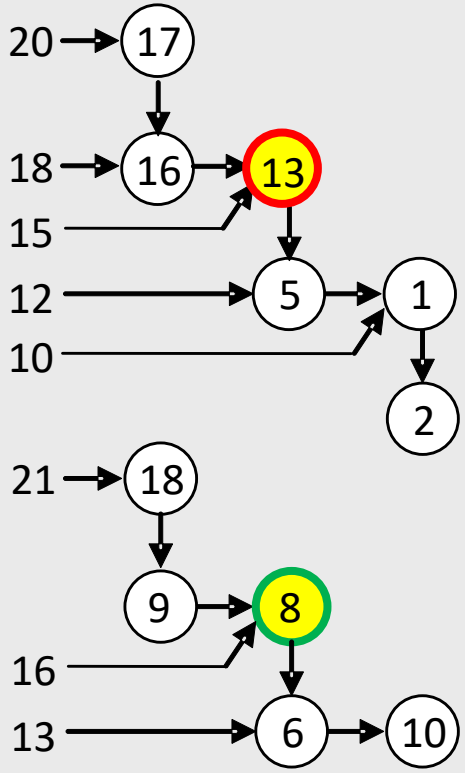


The whole circuit is represented by a single S³BDD

S³BDDs and Fault Collapsing





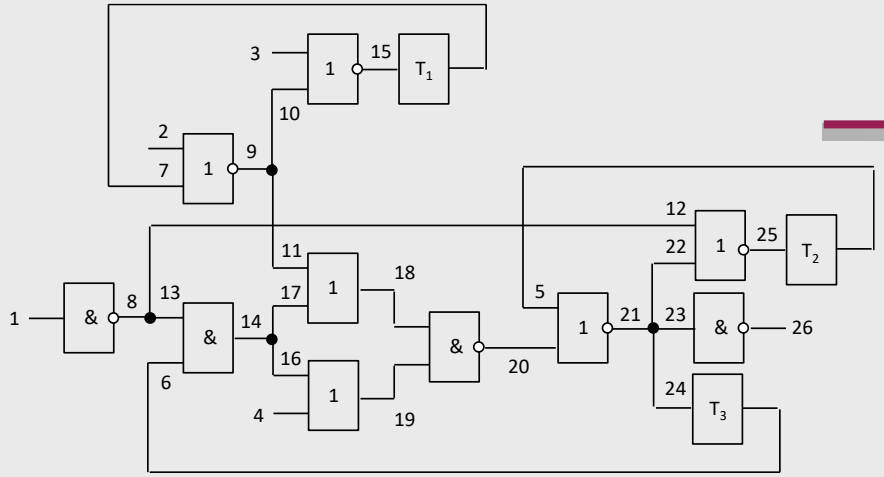


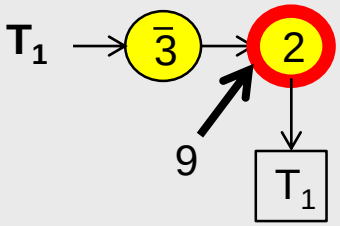
Fault collapsing:

From 84 faults to 36 faults

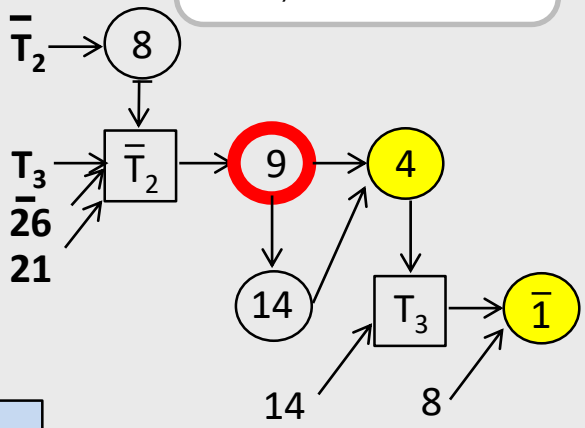
For SSBDD: 50 faults

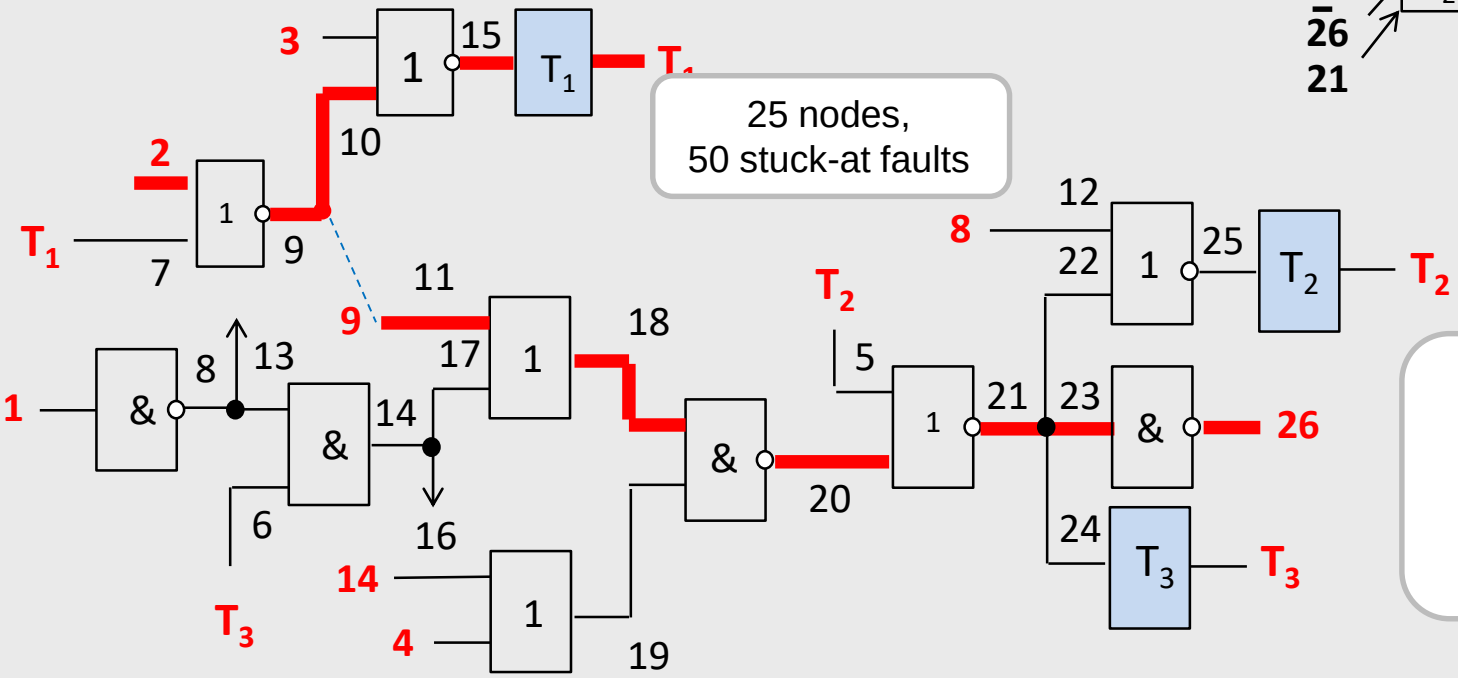
Sequentional S³BDDs





10 nodes,
20 stuck-at faults,
2,5 times less





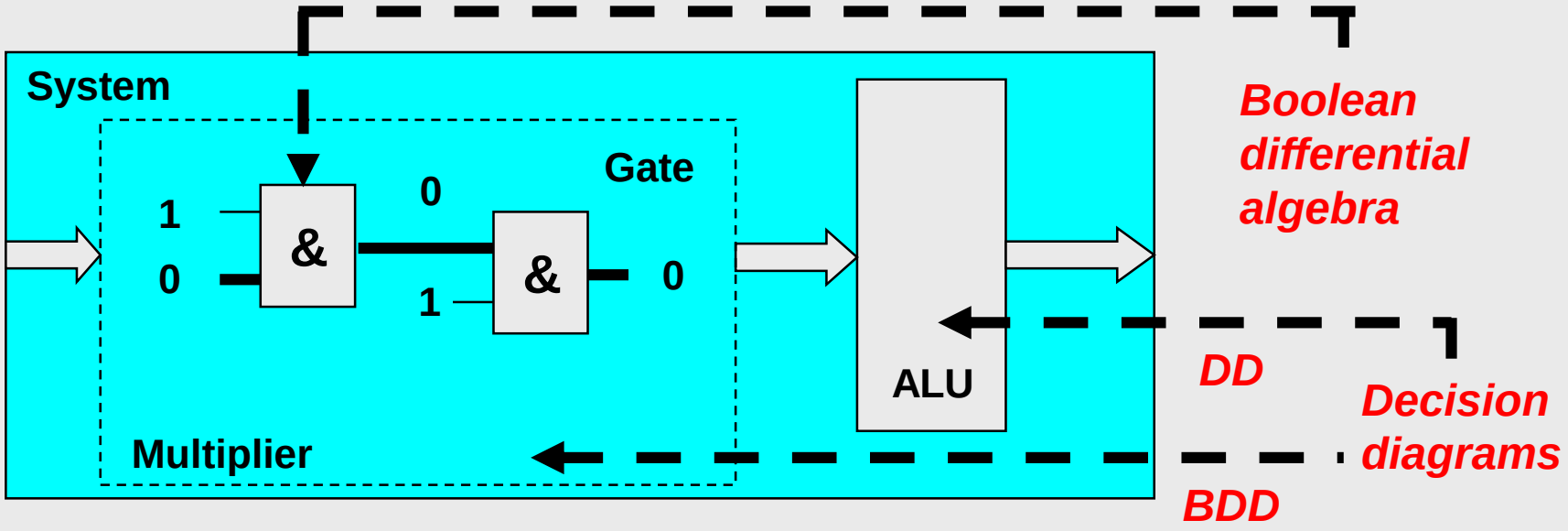
25 nodes,
50 stuck-at faults

GAIN:
2,5 times
in logic simulation,
$2{,}5^{2}$ = **6,25 times**
in fault simulation

How to Go Beyond the Boolean World?

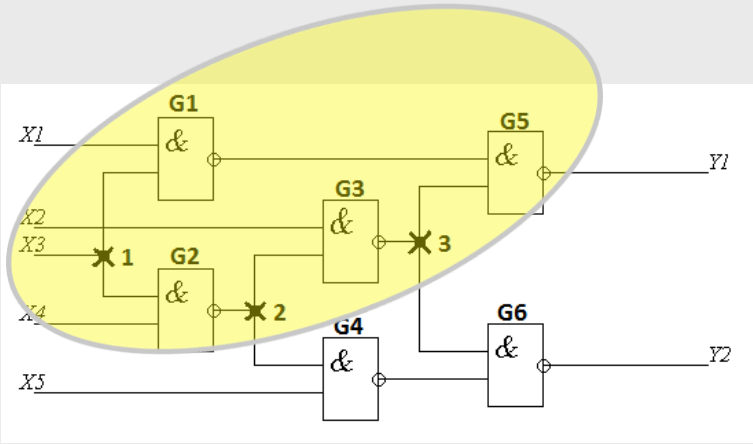
Two basic tasks:

1. **Which test patterns are needed to detect a fault (or all faults)**
2. **Which faults are detected by a given test (or by all tests)**



Synthesis of S³BDD for a Circuit

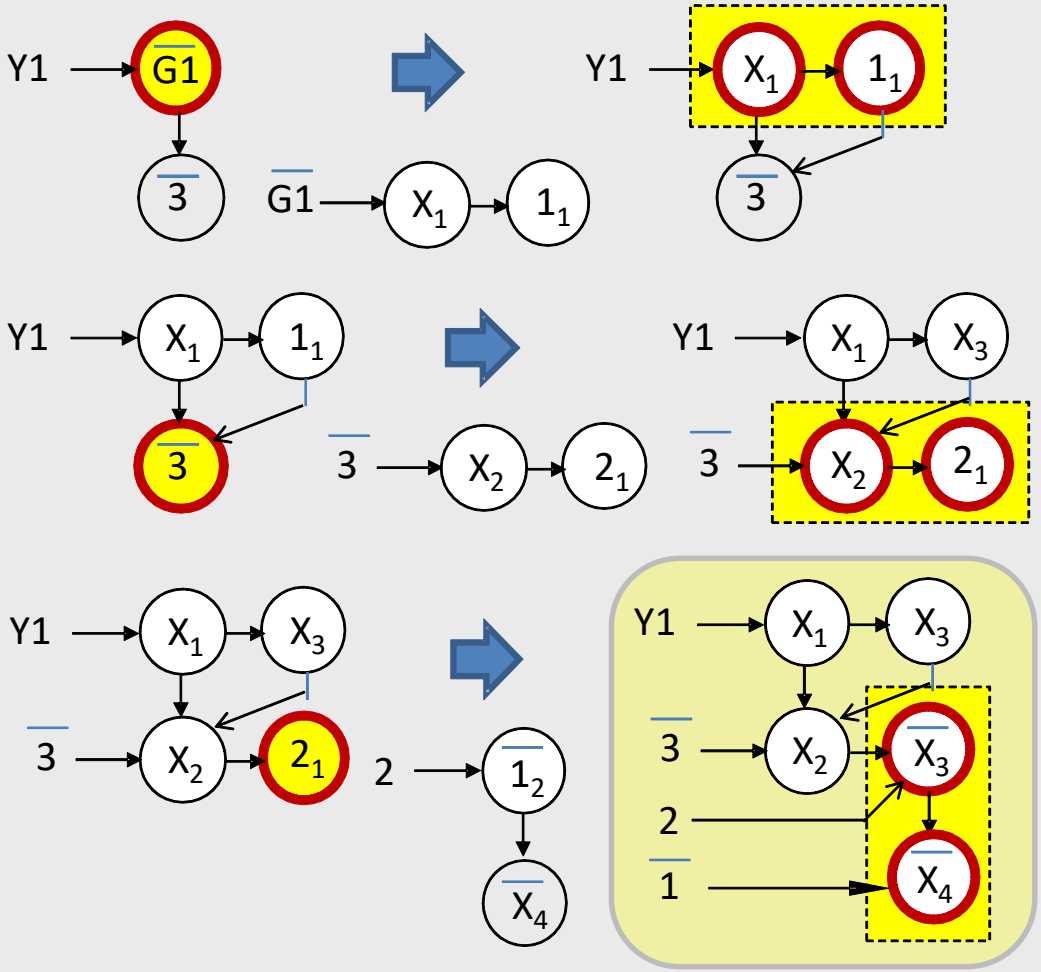
Given circuit C17



Each node in the S³BDD represents a **signal path** in the circuit

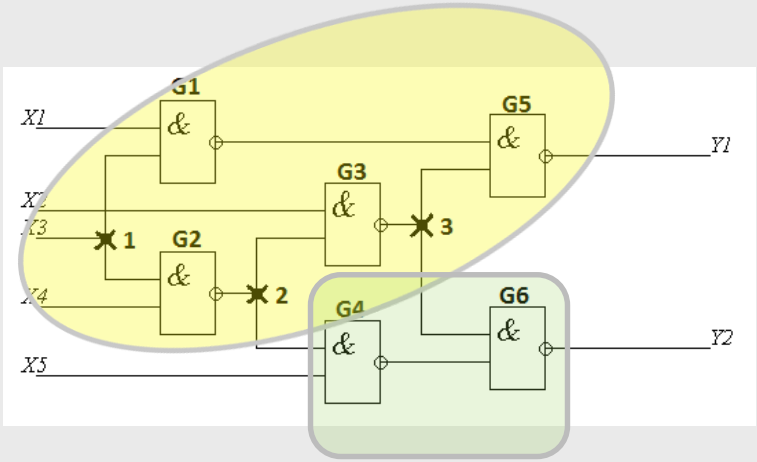
Testing a node in S³BDD means testing a signal path in the circuit

Superposition of BDDs:



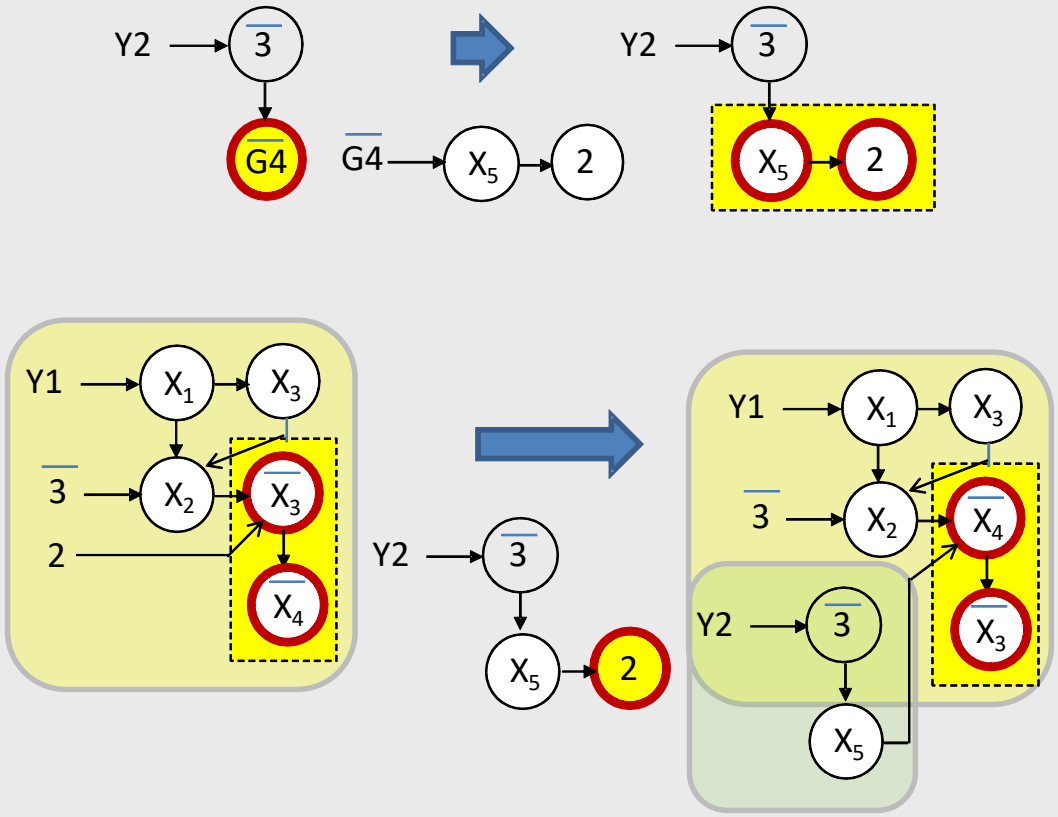
Synthesis of S³BDD for a Circuit

Given circuit C17



Two-output circuit is represented by a single SSBDD with shared subgraphs

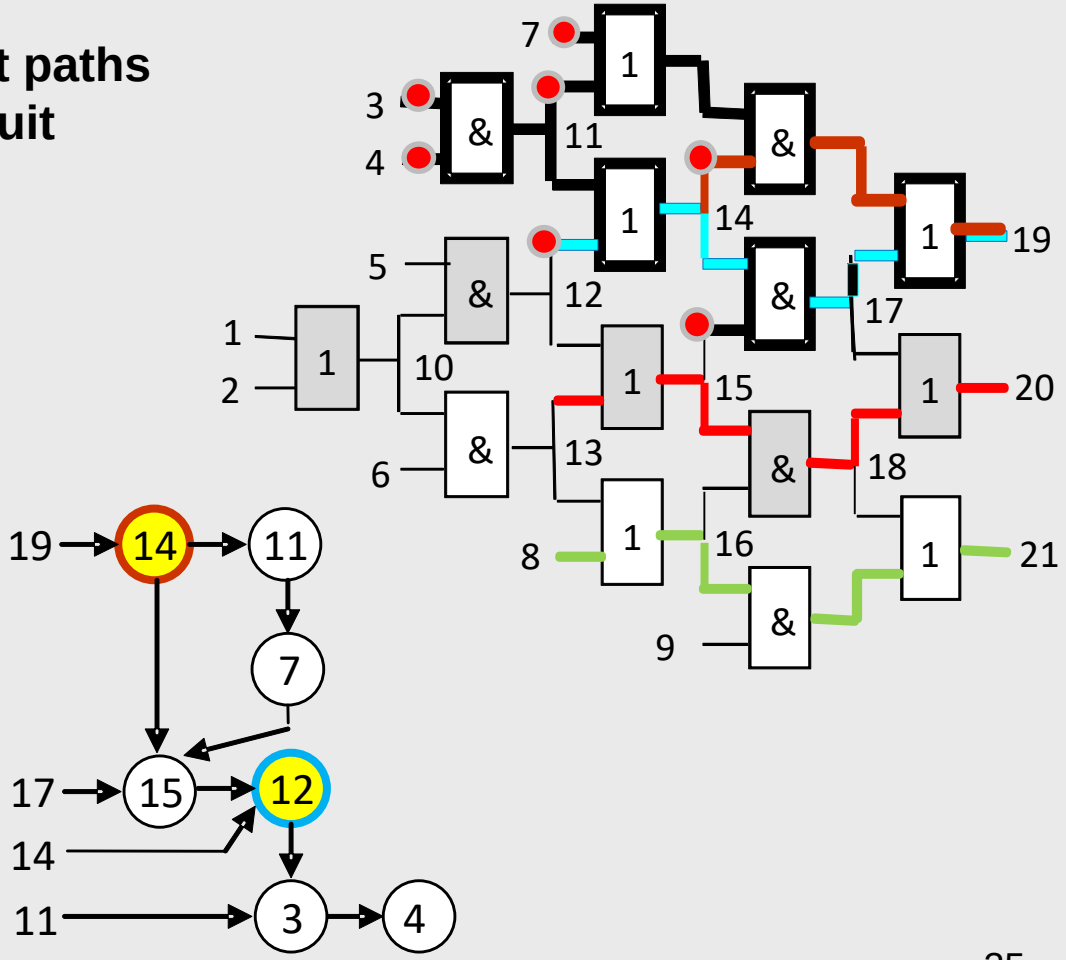
Superposition of BDDs:



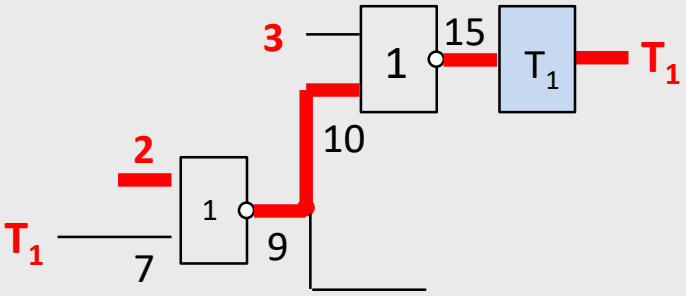
Shared SSBDDs

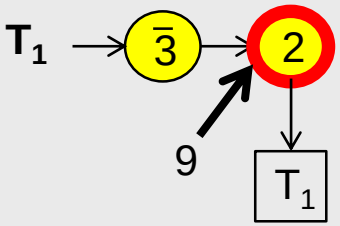
Each node represents different paths (path segments) in the circuit

SSSBDD for 19	
Node	Path
14	14_0-19 (3)
11	11_0-19 (4)
7	7-19 (4)
15	15_0-17_0-19 (3)
12	12_0-14_1-17_0-19 (4)
3	3-11_1-14_1-17_0-19 (5)
4	4-11_1-14_1-17_0-19 (5)



Structured Interpretation of S³BDDs



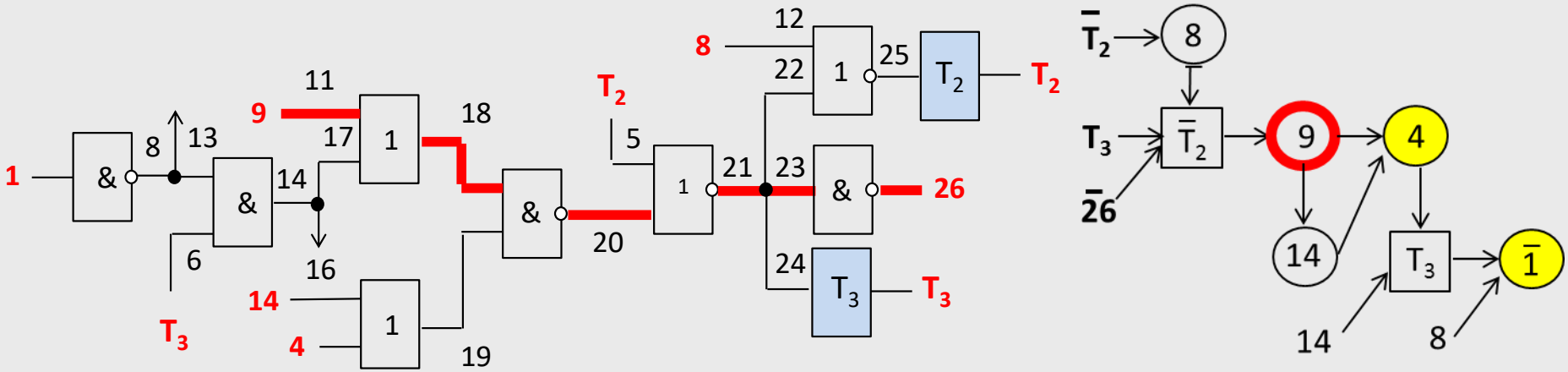


S3BDD represents two subcircuits

G	Nodes	Signal paths	L
G_{T1}	$\neg 3$	$3 - 15 - T_1$	3
	2	$2 - 9 - 10 - 15 - T_1$	5
	T_1	$T_1 - 7 - 9 - 10 - 15 - T_1$	6

Each node in the S3BDD represents a signal path in the circuit

Structured Interpretation of S³BDDs



G	Nodes	Signal paths	L
G_{T2}	8	8 – 12 – 25 – T_2	4
G_{26}	$\neg T_2$	T_2 – 5 – 21 – 23 – 26	5
	9	9 – 11 – 18 – 20 – 21 – 23 – 26	7
	14	14 – 17 – 18 – 20 – 21 – 23 – 26	7
	4	4 – 19 – 20 – 21 – 23 – 26	6
	T_3	T_3 – 6 – 14 – 16 – 19 – 20 – 21 – 23 – 26	9
	$\neg 1$	$\neg 1$ – 8 – 13 – 14 – 16 – 19 – 20 – 21 – 23 – 26	10