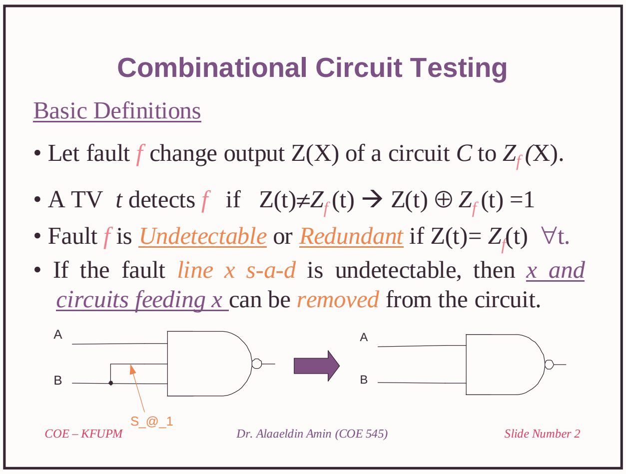
DIGITAL SYSTEM TESTING
COE -545

Lecture – 05

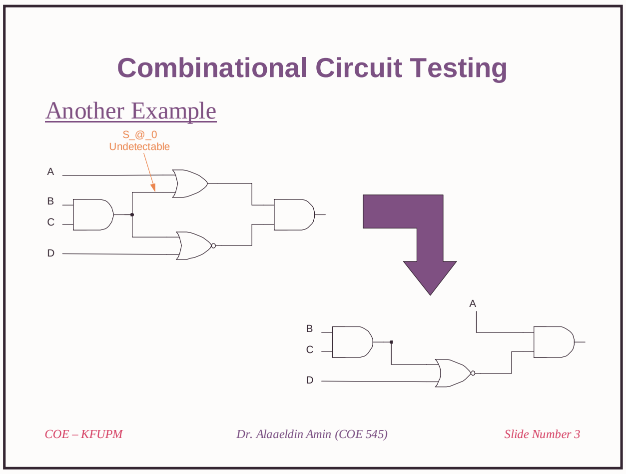
Test Generation (Boolean Difference)



Combinational Circuit Testing

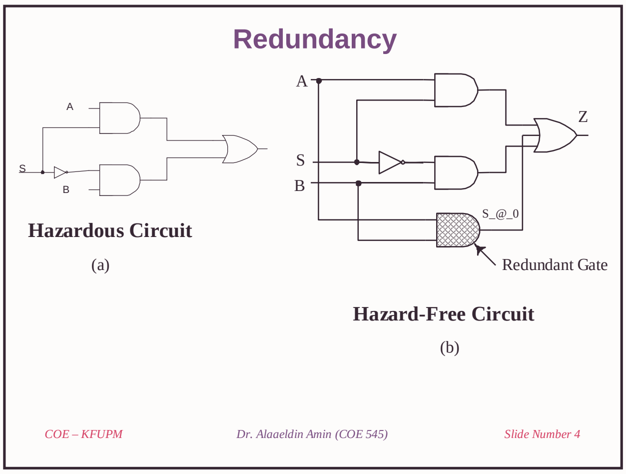
Basic Definitions

- Let fault *f* change output Z(X) of a circuit *C* to $Z_f(X)$.
- A TV *t* detects *f* if $Z(t) \neq Z_f(t) \rightarrow Z(t) \oplus Z_f(t) = 1$
- Fault *f* is *Undetectable* or *Redundant* if $Z(t) = Z_f(t)$ $\forall t$.
- If the fault *line x s-a-d* is undetectable, then *x and circuits feeding x* can be *removed* from the circuit.



Combinational Circuit Testing

Another Example



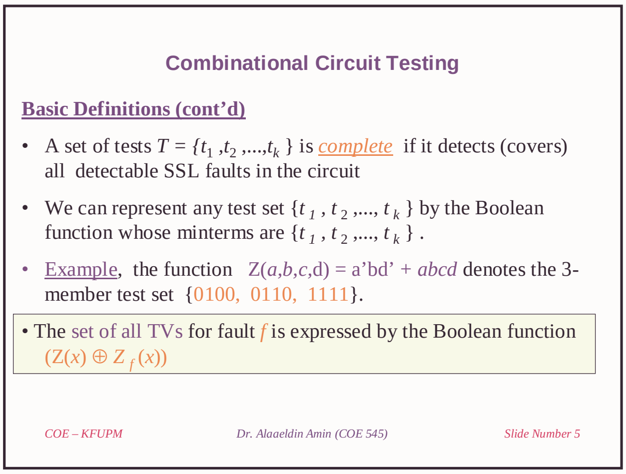
Redundancy

Hazardous Circuit

(a)

Hazard-Free Circuit

(b)

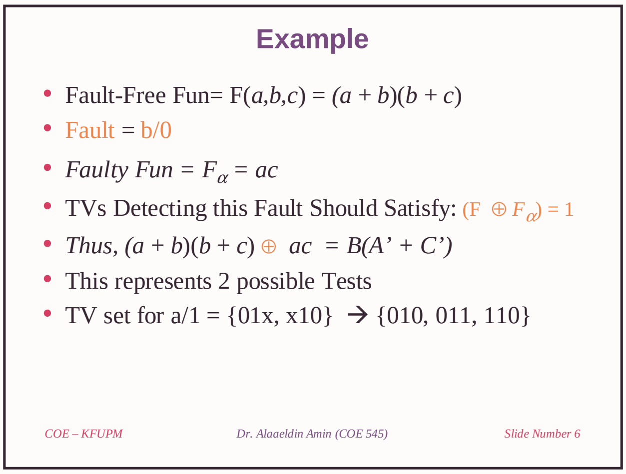


Combinational Circuit Testing

Basic Definitions (cont'd)

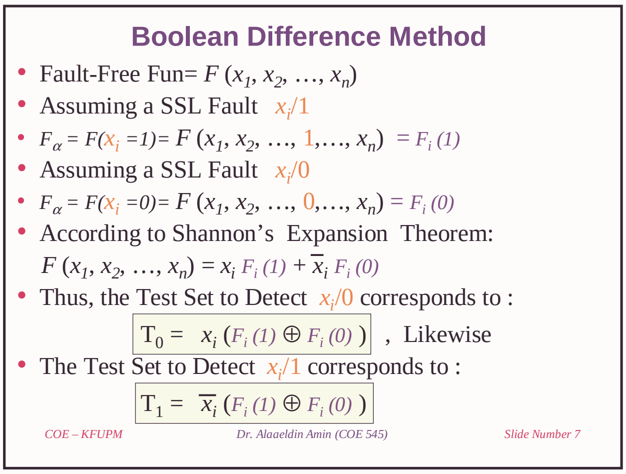
- A set of tests $T = \{t_1, t_2, ..., t_k\}$ is *complete* if it detects (covers) all detectable SSL faults in the circuit
- We can represent any test set $\{t_1, t_2, ..., t_k\}$ by the Boolean function whose minterms are $\{t_1, t_2, ..., t_k\}$.
- Example, the function $Z(a,b,c,d) = a'bd' + abcd$ denotes the 3-member test set {0100, 0110, 1111}.

- The set of all TVs for fault *f* is expressed by the Boolean function $(Z(x) \oplus Z_f(x))$



Example

- Fault-Free Fun= $F(a,b,c) = (a + b)(b + c)$
- Fault = b/0
- *Faulty Fun* $= F_\alpha = ac$
- TVs Detecting this Fault Should Satisfy: $(F \oplus F_\alpha) = 1$
- *Thus,* $(a + b)(b + c) \oplus ac = B(A' + C')$
- This represents 2 possible Tests
- TV set for a/1 = {01x, x10} → {010, 011, 110}

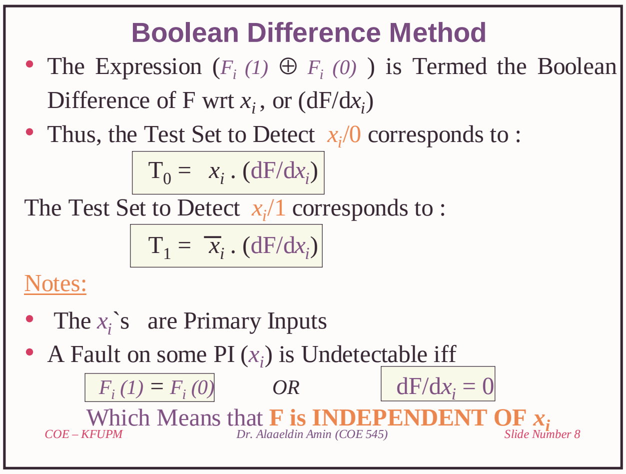


Boolean Difference Method

- Fault-Free Fun= $F(x_1, x_2, \ldots, x_n)$
- Assuming a SSL Fault $x_i/1$
- $F_\alpha = F(x_i = 1) = F(x_1, x_2, \ldots, 1, \ldots, x_n) = F_i(1)$
- Assuming a SSL Fault $x_i/0$
- $F_\alpha = F(x_i = 0) = F(x_1, x_2, \ldots, 0, \ldots, x_n) = F_i(0)$
- According to Shannon's Expansion Theorem: $F(x_1, x_2, \ldots, x_n) = x_i F_i(1) + \overline{x}_i F_i(0)$
- Thus, the Test Set to Detect $x_i/0$ corresponds to :

  $\boxed{T_0 = x_i (F_i(1) \oplus F_i(0))}$, Likewise
- The Test Set to Detect $x_i/1$ corresponds to :

  $\boxed{T_1 = \overline{x}_i (F_i(1) \oplus F_i(0))}$



Boolean Difference Method

- The Expression $(F_i(1) \oplus F_i(0))$ is Termed the Boolean Difference of F wrt x_i, or (dF/dx_i)
- Thus, the Test Set to Detect $x_i/0$ corresponds to :

$\boxed{T_0 = x_i \,.\, (dF/dx_i)}$

The Test Set to Detect $x_i/1$ corresponds to :

$\boxed{T_1 = \overline{x}_i \,.\, (dF/dx_i)}$

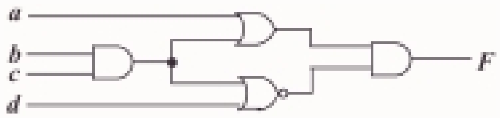
Notes:

- The x_i`s are Primary Inputs
- A Fault on some PI (x_i) is Undetectable iff

$\boxed{F_i(1) = F_i(0)}$ *OR* $\boxed{dF/dx_i = 0}$

Which Means that **F is INDEPENDENT OF x_i**

Example



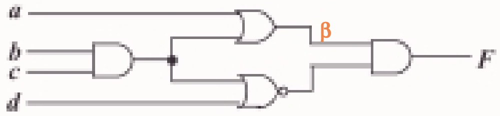
- Fault = a/1
- Fault-Free Fun= $F(a,b,c,d) = (a + bc)(b' + c')d'$
- **T**(a/1) = $a'.dF/da$
- $dF/da = [1(b' + c')d' \oplus bc(b' + c')d']$
  $= b'd' + c'd'$
- So **T**(a/1) = $a'b'd' + a'c'd'$
- Test set for a/1 = {00x0, 0x00} = {0000, 0010, 0100}

Boolean Difference Method

- Boolean Difference (BD) has the following Properties:

1. $d\overline{F}/dx_i = dF/dx_i$
2. $d/dx_i[dF/dx_j] = d/dx_j[dF/dx_i]$
3. $d[F(X) \oplus G(X)]/dx_i = dF/dx_i \oplus dG/dx_i$
4. $d[F(X).G(X)]/dx_i = F.dG(X)/dx_i \oplus G.dF(X)/dx_i \oplus dF(X)/dx_i .dG(X)/dx_i$
5. $d[F(X) + G(X)]/dx_i = \overline{F}.dG/dx_i \oplus \overline{G}.dF/dx_i \oplus (dF/dx_i).(dG/dx_i)$

Example



- Fault = β/0
- **T** (β /0) = $\beta(X).dF(x, \beta)/d\beta$
  - $\beta(X) = a + bc$ $F(x, \beta) = (b' + c')d'\beta$
- $dF/d\beta = (b' + c')d'$
- So **T**(β /0) = $ab'd' + ac'd'$
- Test set for β/0 = {1000,1010,1100}

Test Generation Schemes

- BD method is more Theoretical → Too Complex to Use for Practical Circuits
- Need a More Practical Approach
- **Deterministic Test Generation**
  - **Fault- Oriented ATG**
  - **Fault Independent ATG**
- **Random Test Generation**
  - **Combined Deterministic and Random Test Generation**
- **ATG Systems**
- **Conclusions**

Classification & Problems

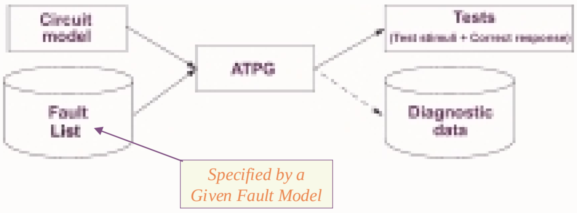
Testing

- **Off- Line**
- **Edge- Pin**
- **Stored- Pattern**
- **Full Comparison of the Output Results**

General Problems for TG:

- **Cost of Generating the Test**
- **Quality of the Generated Test**
- **Cost of Applying the Test**

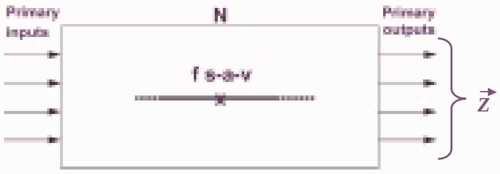
Deterministic Test Generation



Specified by a Given Fault Model

- **Manual / Automatic**
  - **Automatic Test Pattern Generation (ATPG)**
- **Deterministic ATPG Can Be Fault- Oriented / Fault- Independent**

Fault-Oriented ATPG



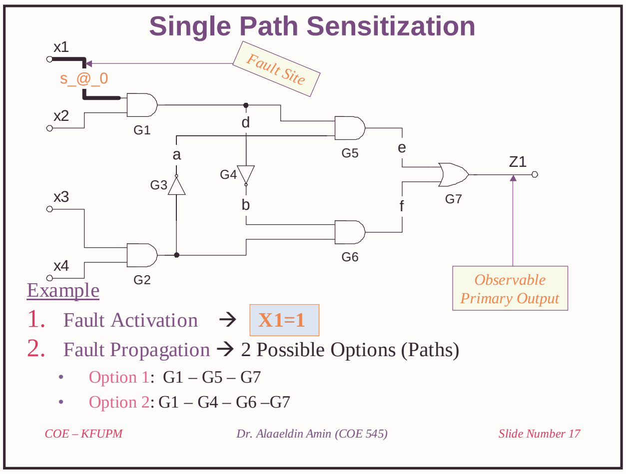
- *Targets* a Particular Fault $f \in \{ \textit{Fault List} \}$
  - Finds a TV t which *Detects* f
    - ❑ t *Detects* f <=> $Z(t) \neq Z_f(t)$ <=> $Z(t) \oplus Z_f(t) = 1$

TV Generation

- Two Steps Are Necessary to Generate TV t which Detects Fault f:
  - Fault Activation/excitation/provoking: Effecting a Different Value $\bar{v}$ at the Faulty Line x When x Is Stuck at v.
  - Fault Propagation: Propagating Error to an Observable PO

Definitions

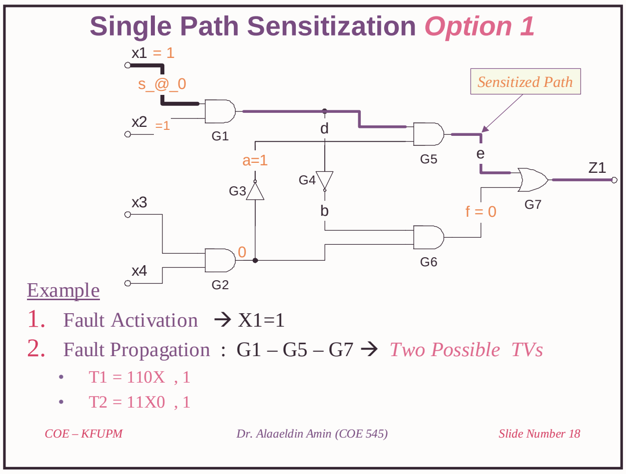
- A *Line* whose Value under TV t Changes in the Presence of Fault f is Said to Be Sensitized to the Fault f by the Test t
- A Path Composed of Sensitized Lines Is Called a Sensitized Path



Single Path Sensitization

Example

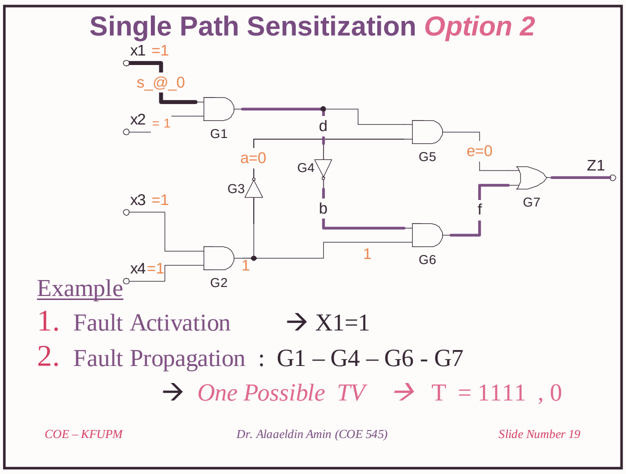
1. Fault Activation → **X1=1**
2. Fault Propagation → 2 Possible Options (Paths)
    - Option 1: G1 – G5 – G7
    - Option 2: G1 – G4 – G6 –G7



Single Path Sensitization *Option 1*

Example

1. Fault Activation → X1=1
2. Fault Propagation : G1 – G5 – G7 → *Two Possible TVs*
    - T1 = 110X , 1
    - T2 = 11X0 , 1

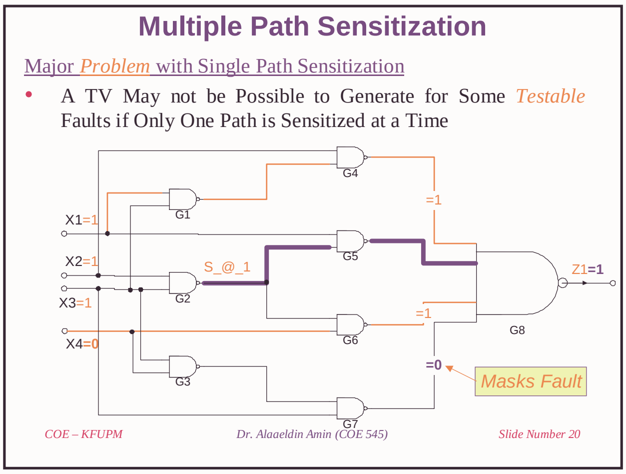


Single Path Sensitization *Option 2*

Example

1. Fault Activation → X1=1
2. Fault Propagation : G1 – G4 – G6 - G7

→ *One Possible TV* → T = 1111 , 0

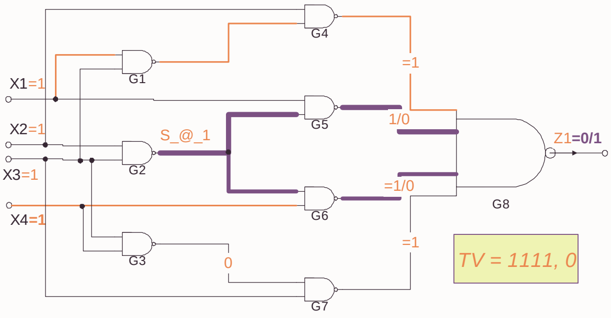


Multiple Path Sensitization

Major *Problem* with Single Path Sensitization

- A TV May not be Possible to Generate for Some *Testable* Faults if Only One Path is Sensitized at a Time

Multiple Path Sensitization



TV = 1111, 0

The D-Algorithm

- An Algorithmic Approach which Generates a TV for a Given Fault if One Exists
- Multiple Path Sensitization
- 5-Valued Logic { 0, 1, X, D, $\overline{D}$}
- **D**: *A Line is Assigned a* **D** *value if it has a value* **1** *in the* ***Fault-Free*** *Circuit but has a 0 Value in the Faulty Circuit.* (**D** ≈ S_@_0)
- $\overline{D}$: *A Line is Assigned a* $\overline{D}$ *value if it has a value* **0** *in the* ***Fault-Free*** *Circuit but has a 1 Value in the Faulty Circuit.* ($\overline{D}$ ≈ S_@_1)
- D / $\overline{D}$ Follow Rules of Boolean Algebra

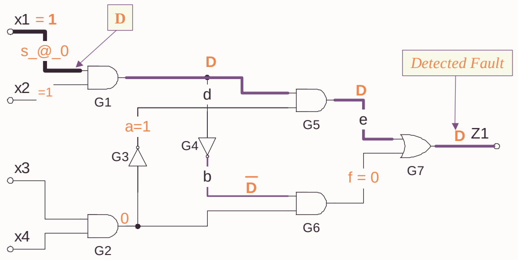
The D-Algorithm

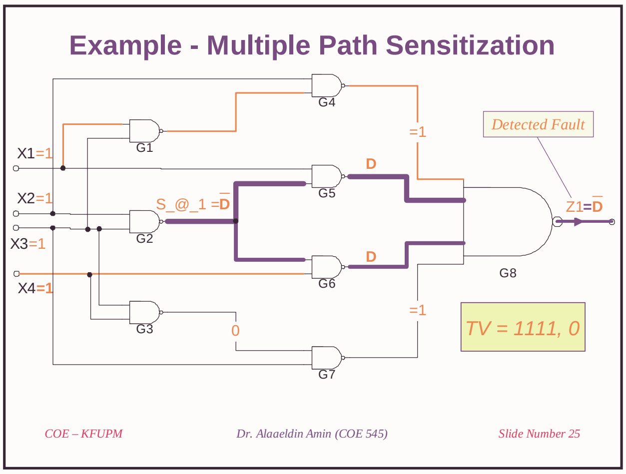
D + D = D	**$\overline{D}$ + $\overline{D}$ = $\overline{D}$**
D . D = D	**$\overline{D}$. $\overline{D}$ = $\overline{D}$**
$\overline{D}$ + D = 1	**D . $\overline{D}$ = 0**
D . 1 = D	**$\overline{D}$. 1 = $\overline{D}$**
D . 0 = 0	**$\overline{D}$. 0 = 0**
D + 1 = 1	**$\overline{D}$ + 1 = 1**
D + 0 = D	**$\overline{D}$ + 0 = $\overline{D}$**

•	**0**	**1**	**D**	**$\overline{D}$**	**x**
0	0	0	0	0	0
1	0	1	D	$\overline{D}$	x
D	0	D	D	0	x
$\overline{D}$	0	$\overline{D}$	0	$\overline{D}$	x
x	0	X	x	x	x

AND Operation of the 5-Valued D-Calculus

Example - Single Path Sensitization



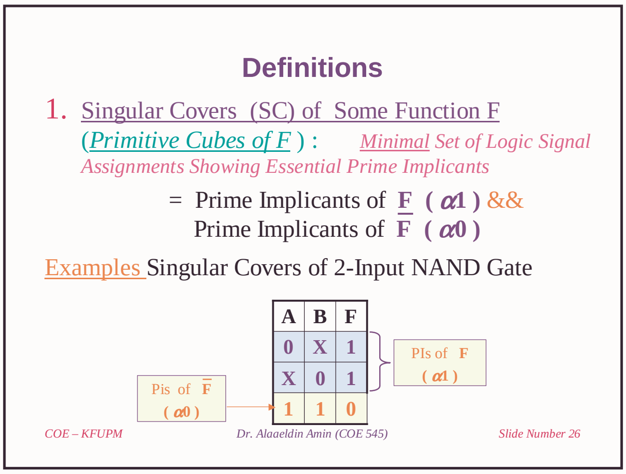


Example - Multiple Path Sensitization

Detected Fault

TV = 1111, 0

COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 25*



Definitions

1. Singular Covers (SC) of Some Function F (*Primitive Cubes of F*) : *Minimal Set of Logic Signal Assignments Showing Essential Prime Implicants*

   = Prime Implicants of **F** (α1) &&
   Prime Implicants of **F** (α0)

Examples Singular Covers of 2-Input NAND Gate

A	B	F
0	X	1
X	0	1
1	1	0

PIs of **F** (α1)

Pis of $\overline{F}$ (α0)

COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 26*