Tema 3.2. Algoritmul D

Algoritmul D este o metodă structurală de elaborare a testelor, care conduce la obţinerea unui test de diagnosticare a unui defect în termenii intrărilor şi ieşirii porţii defecte, generând simultan toate căile posibile de propagare a defectului la toate ieşirile primare ale circuitului. La fiecare pas al algoritmului se verifică convergenţa căilor, renunţându-se la caile care nu sunt divergente. În final, se urmăresc până la intrările primare toate căile de propagare generate, căutându-se în mod automat valorile logice ale semnalelor de intrare, care evidenţiază manifestarea defectului la ieşirile primare.

Algoritmul D utilizează noţiunile de *cub singular* sau *cub de definiţie* şi *cub* D pentru descrierea porţilor logice din circuit.

Cuburi singulare

Funcţionarea porţilor logice poate fi descrisă prin tabele de adevăr. Dacă în aceste tabele valorile de intrare, ce nu influenţează valoarea semnalului de ieşire, se vor considera valori indiferente (notate prin simbolul *), vom obţine un nou tabel. Acest tabel va fi format din cuburi singulare, totalitatea cărora va forma *acoperirea singulară a funcţiei logice,* care descrie comportamentul porţii respective.

De exemplu, pentru determinarea cuburilor singulare a porţii logice ŞI cu două intrări se ţine seama de următoarele considerente:

- este de ajuns ca o singură intrare să fie egală cu 0 pentru a avea la ieşire valoarea 0;

- este necesar ca ambele intrări să fie egale cu 1 pentru a avea la ieşire valoarea 1.

Obţinerea cuburilor singulare din tabelul de adevăr al porţii logice ŞI cu două intrări :

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

$\Rightarrow$

x_1	x_2	y	
0	*	0	CS_1
*	0	0	CS_2 $\leftarrow$cuburi singulare
1	1	1	CS_3

Elementele cubului singular sunt denumite coordonate sau noduri. De exemplu, cubul singular 0*0 are nodurile 000 şi 010 cu coordonatele x1, x2 şi y.

Cuburile singulare ale porţilor logice ŞI, SAU, ŞI-NU şi SAU-NU cu două intrări:

	ŞI			ŞI-NU			SAU			SAU-NU		
x_1	0	*	1	0	*	1	1	*	0	1	*	0
x_2	*	0	1	*	0	1	*	1	0	*	1	0
y	0	0	1	1	1	0	1	1	0	0	0	1

Cuburi $\boldsymbol{D}$ de propagare

Pentru a obţine un cub $\boldsymbol{D}$ pentru o poartă logică se intersectează două cuburi singulare cu valori diferite ale ieşirii conform următoarelor reguli:

$$\begin{aligned}
&0 \cap 0 = 0 \cap * = * \cap 0 = 0 \\
&1 \cap 1 = 1 \cap * = * \cap 1 = 1 \\
&* \cap * = * \\
&1 \cap 0 = \boldsymbol{D} \\
&0 \cap 1 = \bar{\boldsymbol{D}}
\end{aligned} \qquad (2.1)$$

De exemplu, pentru a obţine cuburile $\boldsymbol{D}$ ale porţii logice ŞI (vezi figura 2.9) putem intersecta CS_3 cu CS_1 şi CS_3 cu CS_2:

$$\begin{aligned}
\boldsymbol{CS}_3 &= 1\ 1\ 1 \\
\boldsymbol{CS}_1 &= 0\ *\ 0 \\
\boldsymbol{CS}_3 \cap \boldsymbol{CS}_1 &= \boldsymbol{D}1\boldsymbol{D}
\end{aligned}
\qquad\qquad
\begin{aligned}
\boldsymbol{CS}_3 &= 1\ 1\ 1 \\
\boldsymbol{CS}_2 &= *\ 0\ \ 0 \\
\boldsymbol{CS}_3 \cap \boldsymbol{CS}_2 &= 1\boldsymbol{DD}
\end{aligned}$$

Cubul $\boldsymbol{D}$ exprimă dependenţa semnalului de la ieşirea porţii logice faţă de cea aplicată la una din intrările ei. $\boldsymbol{D}$ poate avea două valori: 0 sau 1. De exemplu, cubul

$D1D$ are nodurile 010 şi 111. Aceasta mărturiseşte despre faptul că, dacă poarta e corectă, atunci semnalul de la ieşirea *y* este determinat doar de semnalul de la intrarea x_1, x_2 fiind fixat în 1 logic. Astfel, apariţia defectului $x_1 \equiv 1$ ($x_1 \equiv 0$) este detectată la ieşirea porţii respective.

Dacă semnalul oricărei coordonate a cubului D, notat prin D, are valoarea 1(0), atunci toate celelalte coordonate, notate prin D, vor fi egale cu 1(0), iar semnalele coordonatelor, notate prin $\bar{D}$, vor fi egale cu 0(1).

Se deosebesc *cuburi* D *singulare* şi *cuburi* D *multiple*. În cuburile D singulare doar o singură intrare e notată prin D sau $\bar{D}$. În cuburile D multiple două sau mai multe intrări sunt notate prin simbolurile D sau $\bar{D}$. Necesitatea utilizării cuburilor D multiple se explică prin faptul că la intrările unei porţi logice se pot întâlni mai multe semnale D, atunci când în circuit sunt prezente ramificaţii.

Cuburile D ale porţilor logice se mai numesc *cuburi* D *de propagare* (CDP). CDP ale porţilor logice ŞI, SAU, ŞI-NU şi SAU-NU cu două intrări.

	ŞI				ŞI-NU				SAU				SAU-NU			
x_1	D	1	D	D	D	1	D	D	D	0	D	D	D	0	D	D
x_2	1	D	D	$\bar{D}$	1	D	D	$\bar{D}$	0	D	D	$\bar{D}$	0	D	D	$\bar{D}$
y	D	D	D	0	$\bar{D}$	$\bar{D}$	$\bar{D}$	1	D	D	D	1	$\bar{D}$	$\bar{D}$	$\bar{D}$	0

Cuburi D ale defectelor (CDD)

Cuburile D ale defectelor permit exprimarea testelor în termenii intrării şi ieşirii porţii defecte. Ele se utilizează atunci când este necesar a testa nodurile interne ale circuitului. În CDD simbolul D e interpretat ca semnal 1 logic pentru starea corectă şi ca semnal 0 logic pentru starea defectă. Simbolul $\bar{D}$ e interpretat invers – 0 logic pentru starea corectă şi 1 logic pentru starea defectă.

CDD pentru un nod blocat la 0 poate fi obţinut prin intersecţia fiecărui cub singular cu valoarea ieşirii egală cu 1 logic pentru poarta corectă cu fiecare cub singular al cărei ieşire este egală cu 0 logic pentru o poartă defectă. În mod similar, CDD pentru un nod blocat la 1 poate fi obţinut prin intersecţia fiecărui cub singular cu valoarea ieşirii egală cu 0 logic pentru poarta corectă cu fiecare cub singular al cărei ieşire este egală cu 1 logic pentru o poartă defectă.

CDD ale porţilor logice ŞI, SAU, ŞI-NU şi SAU-NU.

<table>
<tr><th rowspan="2">Defecte</th><th colspan="3">ŞI</th><th colspan="3">ŞI-NU</th><th colspan="3">SAU</th><th colspan="3">SAU-NU</th><th colspan="3">XOR</th></tr>
<tr><th>x_1</th><th>x_2</th><th>y</th><th>x_1</th><th>x_2</th><th>y</th><th>x_1</th><th>x_2</th><th>y</th><th>x_1</th><th>x_2</th><th>y</th><th>x_1</th><th>x_2</th><th>y</th></tr>
<tr><td rowspan="2">≡ 0</td><td rowspan="2">1</td><td rowspan="2">1</td><td rowspan="2">D</td><td>0</td><td>*</td><td>D</td><td>1</td><td>*</td><td>D</td><td rowspan="2">0</td><td rowspan="2">0</td><td rowspan="2">D</td><td>1</td><td>0</td><td>D</td></tr>
<tr><td>*</td><td>0</td><td>D</td><td>*</td><td>1</td><td>D</td><td>0</td><td>1</td><td>D</td></tr>
<tr><td rowspan="2">≡ 1</td><td>0</td><td>*</td><td>$\bar{D}$</td><td rowspan="2">1</td><td rowspan="2">1</td><td rowspan="2">$\bar{D}$</td><td rowspan="2">0</td><td rowspan="2">0</td><td rowspan="2">$\bar{D}$</td><td>1</td><td>*</td><td>$\bar{D}$</td><td>0</td><td>0</td><td>$\bar{D}$</td></tr>
<tr><td>*</td><td>0</td><td>$\bar{D}$</td><td>*</td><td>1</td><td>$\bar{D}$</td><td>1</td><td>1</td><td>$\bar{D}$</td></tr>
</table>

Intersecţia D

Generarea unei căi de propagare a defectului spre ieşirea primară a circuitului este realizată prin intermediul intersecţiei D.

Fie date două cuburi D:

$A = (a_1, a_2, \ldots, a_n)$

$B = (b_1, b_2, \ldots, b_n)$,

unde $a_i, b_i \in \{1, 0, *, D, \bar{D}\}, \quad i = \overline{1, n}$.

Intersecţia D se efectuează doar pentru coordonate identice conform următoarelor reguli:

1) $* \underset{D}{\cap} a_i = a_i$

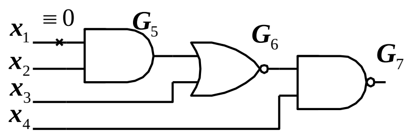
$* \underset{D}{\cap} b_i = b_i$

2) Dacă $a_i \neq *$ şi $b_i \neq *$, atunci

$$a_i \underset{D}{\cap} b_i = \begin{cases} a_i, & pentru \quad a_i = b_i \\ \varnothing, & pentru \quad a_i \neq b_i \end{cases}.$$

Intersecţia $\boldsymbol{D}$ reprezintă o formă de descriere a propagării defectului de la nodul analizat spre ieşirea primară a circuitului.

Să analizăm o cale arbitrară, selectată pentru propagarea defectului $x_1 \equiv 0$



Utilizând intersecţia $\boldsymbol{D}$ dintre cuburile $\boldsymbol{D}$ ale porţilor logice din calea respectivă vom obţine cubul $\boldsymbol{D}$ al circuitului .

Formarea cubului $\boldsymbol{D}$ al circuitului

Cuburi	Coordonate						
	x_1	x_2	x_3	x_4	G_5	G_6	G_7
CDP5	$\boldsymbol{D}$	1	*	*	$\boldsymbol{D}$	*	*
CDP6	*	*	0	*	$\boldsymbol{D}$	$\bar{\boldsymbol{D}}$	*
CDP7	*	*	*	1	*	$\bar{\boldsymbol{D}}$	$\boldsymbol{D}$
CD al circuitului	$\boldsymbol{D}$	1	0	1	$\boldsymbol{D}$	$\bar{\boldsymbol{D}}$	$\boldsymbol{D}$

Cubul $\boldsymbol{D}$ obţinut ($\boldsymbol{D}$, 1, 0, 1, $\boldsymbol{D}$, $\bar{\boldsymbol{D}}$, $\boldsymbol{D}$) verifică conexiunile x_1, G_5, G_6 în baza ieşirii G_7, fiind fixate valorile semnalelor intrărilor primare x_2, x_3, x_4.

Deoarece $\boldsymbol{D}$ poate lua două valori logice – 0 şi 1, cubul $\boldsymbol{D}$ al circuitului se foloseşte la detectarea a două defecte: $x_1 \equiv 0$ şi $x_1 \equiv 1$.

Pentru $x_1 \equiv 0$, considerăm $\boldsymbol{D}=1$ şi obţinem testul:

$$T_{x1\equiv0}=(x_1, x_2, x_3, x_4; G_7)=(1, 1, 0, 1; 1).$$

Pentru $x_1 \equiv 1$, considerăm $\boldsymbol{D}=0$ şi obţinem testul:

$$T_{x1\equiv1}=(x_1, x_2, x_3, x_4; G_7)=(0, 1, 0, 1; 0).$$

Etapele algoritmului *D*

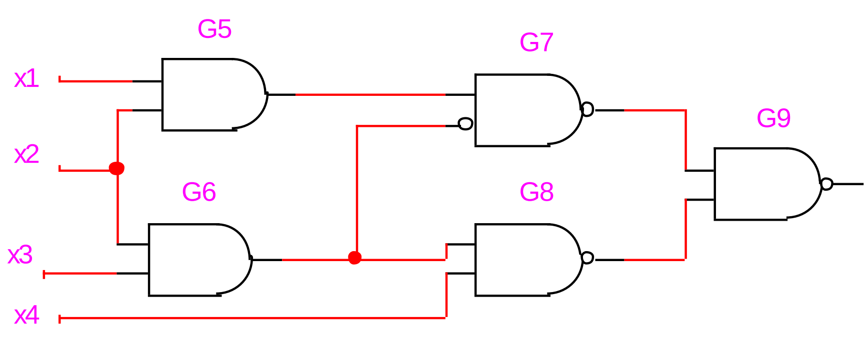
Pentru început se determină cuburile singulare (CS) şi cuburile ***D*** de propagare (CDP) a fiecărei porţi logice din circuitul logic combinaţional (CLC).

Generarea testelor prin metoda algoritmului ***D*** constă din următoarele etape:

1) Construirea cubului ***D*** al defectului (CDD);
2) Propagarea defectului prin efectuarea intersecţiei CDD cu CDP a porţilor logice de pe calea aleasă până la ieşirea primară;
3) Verificarea consistenţei (trecerea înapoi) prin intersecţia cubului rezultant cu CS ale porţilor logice, care nu au fost utilizate la prima intersecţie;
4) Repetarea etapelor 1-3 până când se obţin testele pentru toate defectele analizate pe toate căile singulare şi multiple;
5) Minimizarea testelor.

La etapa de verificare a consistenţei se pot obţine elemente vide pe anumite coordonate. În acest caz întreaga intersecţie se consideră vidă şi se renunţă la calea dată, deoarece ea este inconsistentă.

Să analizăm mai detaliat primele trei etape de generare a testelor pentru circuitul logic prezentat.



Testul pentru detectarea defectelor intrării primare $x_1 \equiv 0$ şi $x_1 \equiv 1$ pe calea (5,7,9):

Et.	Explicaţii	Cub	Coordonate		
			1 2 3 4	5 6 7 8	9
Et. 1	CDD	C_1	D * * *	* * * *	*
Et. 2	Intersectăm C_1 cu CDP al G_5	C_2	D 1* *	D * * *	*
	Intersectăm C_2 cu CDP al G_7	C_3	D 1 * *	D 0 $\bar{D}$ *	*
	Intersectăm C_3 cu CDP al G_9	C_4	D 1 * *	D 0 $\bar{D}$ 1	D
Et. 3	Intersectăm C_4 cu CS al G_8	C_5	D 1 * *	D 0 $\bar{D}$ 1	D
	Intersectăm C_5 cu CS al G_6	C_6	D 1 0 *	D 0 $\bar{D}$ 1	D

Din cubul rezultant C_6 se obţin două teste:

D=1 $T_{x2\equiv 0}=(x_1,x_2,x_3,x_4;8)=(1,1,0,*;1)$ şi

D=0 $T_{x2\equiv 1}=(x_1,x_2,x_3,x_4;8)=(0,1,0,*;0)$.

Testul pentru detectarea defectului conexiunii interne $G_6\equiv 0$ pe calea (8,9). Deoarece în CDD ale porţilor logice simbolul D poate lua doar valoarea 1, iar $\bar{D}$ doar valoarea 0, nu este posibil a detecta printr-un singur test defectele $G_6\equiv 0$ şi $G_6\equiv 1$, fiind necesare două CDD diferite.

Et.	Explicaţii	Cub	Coordonate		
			1 2 3 4	5 6 7 8	9
Et. 1	CDD C1=CDD	C_1	* 1 1 *	* D * *	*
Et. 2	Intersectăm C_1 cu CDP al G_8	C_2	* 1 1 1	* D * $\bar{D}$	*
	Intersectăm C_2 cu CDP al G_9	C_3	* 1 1 1	0 D 1 $\bar{D}$	D
Et. 3	Intersectăm C_3 cu CS al G_7	C_4	* 1 1 1	0 D 1 $\bar{D}$	D
	Intersectăm C_4 cu CS al G_5	C_5	0 1 1 1	0 D 1 $\bar{D}$	D

Din cubul rezultant C_4 se obţine următorul test:

$T_{G5\equiv 0}=(x_1,x_2,x_3,x_4;8)=(0,1,1,1;1)$.

Pentru a obţine testul detectării defectului $G_6 \equiv 1$ vom considera $CDD=(x_2,x_3,6)=(0,*,\ D)$ sau $CDD=(x_2,x_3,6)=(*,0,\ D)$.

Algoritmul D este o metodă bine formalizată şi poate fi uşor programată, ceea ce permite automatizarea procesului generării testelor. În prezent există mai multe modificaţii ale algoritmului D, care permit accelerarea procesului de generare a testelor. De exemplu, procedura PODEM, în care fiecare pas de propagare este urmat de paşii de trecere înapoi până la intrările primare ale circuitului, depistându-se mai repede căile inconsistente. Pe lângă aceasta, spre deosebire de metoda activării unei căi, algoritmul D garantează obţinerea testului, dacă acesta există, datorită faptului că sunt analizate toate căile de propagare a defectului, inclusiv şi cele multiple.

Exemplu. Vom obţine testele pentru intrările primare pe toate căile posibile, inclusiv şi cele multiple.

Nr	Def.	Intrări primare				Conexiuni interne				Ieş.	Calea
		x1	x2	x3	x4	5	6	7	8	9	
1	x_1	D	1	0	*	D	0	$\bar{D}$	1	D	5,7,9
2	x_2	1	D	0	*	D	0	$\bar{D}$	1	D	5,7,9
3	x_2	0	D	1	1	0	D	1	$\bar{D}$	D	6,8,9
4	x_2	1	D	1	1	D	D	1	$\bar{D}$	D	5-6,7-8,9
5	x_2	 1	D 1	1 $1 \cap D = \varnothing$	1	1 Cale inconsistentă	D				6,7,9
6	x_3	1	1	D	0	1	D	D	1	$\bar{D}$	6,7,9
7	x_3	0	1	D	1	0	D	1	$\bar{D}$	D	6,8,9
8	x_3		1	D	1	1 Cale convergentă	D	D	$\bar{D}$	1	5,7,9
9	x_4	*	1	1	D	*	1	1	$\bar{D}$	D	8,9

Din cele 7 cuburi rămase, pot fi obţinute 14 teste, înlocuind simbolul ***D*** cu valorile 0 şi 1, în dependenţă de defectul analizat .

Tabelul iniţial al testelor

Nr.	Def	Testul x_1, x_2, x_3, x_4 ; 9
1	$x_1 \equiv 0$	1 1 0 * 1
2	$x_1 \equiv 1$	0 1 0 * 0
3	$x_2 \equiv 0$	1 1 0 * 1
4	$x_2 \equiv 1$	1 0 0 * 0
5	$x_2 \equiv 0$	0 1 1 1 1
6	$x_2 \equiv 1$	0 0 1 1 0
7	$x_2 \equiv 0$	1 1 1 1 1

Nr.	Def	Testul x_1, x_2, x_3, x_4; 9
8	$x_2 \equiv 1$	1 0 1 1 0
9	$x_3 \equiv 0$	1 1 1 0 0
10	$x_3 \equiv 1$	1 1 0 0 1
11	$x_3 \equiv 0$	0 1 1 1 1
12	$x_3 \equiv 1$	0 1 0 1 0
13	$x_4 \equiv 0$	* 1 1 1 1
14	$x_4 \equiv 1$	* 1 1 0 0

Tabelul testelor minimizate

Nr	Nr. t.iniţiale	x_1, x_2, x_3, x_4; 9	Defectele detectate
1	1,3,10	1 1 0 0 1	$x_1 \equiv 0$, $x_2 \equiv 0$, $x_3 \equiv 1$
2	2,12	0 1 0 1 0	$x_1 \equiv 1$, $x_3 \equiv 1$
3	3,10	1 1 0 0 1	$x_2 \equiv 0$, $x_3 \equiv 1$
4	4	1 0 0 0 0	$x_2 \equiv 1$
5	5,11,13	0 1 1 1 1	$x_2 \equiv 0$, $x_3 \equiv 0$, $x_4 \equiv 0$
6	6	0 0 1 1 0	$x_2 \equiv 1$
7	8	1 0 1 1 0	$x_2 \equiv 1$
8	9,14	1 1 1 0 0	$x_3 \equiv 0$, $x_4 \equiv 1$