BIULETYN INSTYTUTU AUTOMATYKI I ROBOTYKI
NR 16, 2001

Algorytm generowania testu dla układów cyfrowych o nieznanej strukturze wewnętrznej

Artur ARCIUCH

Zak!ad Systemów Komputerowych, Instytut Automatyki i Robotyki WAT, ul Kaliskiego 2, 00-908 Warszawa.

STRESZCZENIE: W artykule przedstawiono metod" i algorytm do generowania testów dla uk!adów cyfrowych o nieznanej strukturze wewn"trznej. Cech# charakterystyczn# zaproponowanego algorytmu jest skrócenie sekwencji testowej bez pogorszenia jako$ci testowania. Test otrzymany w wyniku dzia!ania algorytmu jest nieredukowalnym testem kontrolnym.

1. Wstęp

Powsta!e w drugiej po!owie lat sze$%dziesi#tych metody generacji testów dla uk!adów cyfrowych bazuj# na znajomo$ci ich struktury wewn"trznej. Wykorzystuj# one algebr" Rooth'a i bardzo dobrze nadaj# si" do wyznaczania testów wykrywaj#cych niezdatno$ci sta!osygna!owe dla uk!adów cyfrowych, które sk!adaj# si" z bramek b"d#cych podstawowymi funktorami logicznymi. Obecnie wytwarzane s# scalone uk!ady cyfrowe wykonywane w technologiach submikronowych, które pozwalaj# realizowa% ca!e bloki systemów. Bardzo du&a skala integracji oraz du&a z!o&ono$% takich uk!adów powoduj#, &e rozpatrywanie ich na poziomie bramkowym jest trudne, a nawet niemo&liwe, np. zadanie zaprojektowania testów dla procesora rozpatrywanego na poziomie bramkowym jest zadaniem bardzo trudnym. Z tego powodu cz"$% wspó!cze$nie stosowanych metod generowania testów traktuje diagnozowany uk!ad jako obiekt o nie znanej strukturze wewn"trznej, który realizuje okre$lone przekszta!cenie. Opis systemu cyfrowego traktowanego jako czarna skrzynka realizuj#ca pewne przekszta!cenie mo&na uogólni% do opisu dowolnego obiektu (np.: procesora, programu komputerowego itp.), który powinien realizowa% okre$lone przekszta!cenie. Takie podej$cie pozwoli wype!ni% luk" mi"dzy poziomem rozwoju technologii, a rozwojem metod generowania testów. Do obiektów, dla których mo&na zastosowa% takie podej$cie mo&na zaliczy% równie& uk!ady PLD.

Uk!ady PLD podobnie jak matryce bramkowe (GA – ang. Gate Arrays), (S.C. – ang. Standard Cells), uk!ady projektowane ca!kowicie indywidualnie (FC – ang. Full Custom) s# zaliczane do klasy uk!adów ASIC (ang. Application – Specific Integrated Circuits). Coraz powszechniejsze zastosowanie tych uk!adów wynika z przyczyn ekonomicznych (obni&enie kosztów projektowania urz#dze') oraz technicznych (polepszenia niezawodno$ci, zmniejszenie rozmiarów, zwi"kszenie szybko$ci dzia!ania). Istotn# zalet# tych uk!adów jest równie& ich indywidualno$% i wewn"trzna „skryto$%”, gdy& producent urz#dze' wykorzystuj#cych uk!ady PLD zabezpiecza w ten sposób swoje wyroby przed na$ladowaniem i kradzie&# oryginalnych rozwi#za' technicznych. Podstaw# do oceny stanu funkcjonalnego uk!adu PLD jest wnioskowanie z charakteru reakcji tego uk!adu na okre$lone wymuszenia, którym jest on poddawany. W tym celu mo&na si" pos!u&y% zaprogramowanymi eksperymentami (z regu!y wcze$niej zaprojektowanymi) w stopniu umo&liwiaj#cym okre$lenie zwi#zków zachodz#cych mi"dzy okre$lonymi wymuszeniami podanymi na obiekt, a reakcjami obiektu na te wymuszenia, w zale&no$ci od rozpoznawanych niezawodno$ciowych stanów funkcjonalnych, w których obiekt mo&e si" znajdowa%. Wymuszenia maj# charakter okre$lonych zada' funkcjonalnych, do spe!nienia których dany obiekt jest przeznaczony. Oczywi$cie najprostsz# metod# przetestowania obiektu jest metoda porównania z odpowiedziami wzorcowymi odpowiedzi rzeczywistych uzyskanych w wyniku podania na obiekt pe!nego wymuszenia. Poni&ej opisano metod" oraz algorytm do generowania testu na podstawie znajomo$ci poprawnego przekszta!cenia realizowanego przez testowany uk!ad cyfrowy. Zalet# metody jest skrócenie sekwencji testowej bez obni&enia jako$ci testowania.

2. Opis metody do generowanie testu

W metodzie wykorzystano funkcjonalny model niezawodno$ciowy obiektu dyskretnego[1], w celu okre$lenia funkcjonalnego stanu niezawodno$ciowego diagnozowanego obiektu, a wi"c stanu, w którym obiekt realizuje okre$lone (jedno z mo&liwych), jednoznaczne przekszta!cenie sko'czonego zbioru jego produktów wej$ciowych w sko'czony zbiór jego produktów wyj$ciowych.

Funkcjonalnym modelem niezawodno$ciowym obiektu dyskretnego nazywamy par" uporz#dkowan#: $< R : S \to (N \equiv A);\ P(n = n'),\ n \in N >$ gdzie: S oraz N oznaczaj# odpowiednio zbiory rzeczywistych oraz funkcjonalnych stanów niezawodno$ciowych danego obiektu; $A,\ (A = \{X \to Y\})$ oznacza zbiór takich jednoznacznych przekszta!ce' zbioru produktów wej$ciowych X, $(1 \le Card X < \infty)$ w zbiór produktów wyj$ciowych Y, $(1 \le Card\ Y < \infty)$, &e funkcjonalny stan niezawodno$ciowy $n_0,\ (n_0 \in N)$ nazywany funkcjonalnym stanem zdatno$ci obiektu, uto&samiany jest z przekszta!ceniem $A_0,\ (A \in A)$ opisuj#cym poprawne dzia!anie obiektu, a ka&dy funkcjonalny stan niezawodno$ciowy $n'',\ (n'' \in N \setminus n_0)$ nazywany funkcjonalnym stanem niezdatno$ci obiektu z odpowiednim przekszta!ceniem $A'',\ (A'' \in A \setminus A_0)$ opisuj#cym okre$lone wadliwe dzia!anie obiektu, natomiast $P(n = n'),\ (n' \in N)$ oznacza rozk!ad prawdopodobie'stwa zmiennej losowej n.

Element zbioru $W(X),\ (W(X) = \{X' \subseteq X : X' \neq \varnothing\})$ nazywamy wymuszeniem funkcjonalnym, a warto$% $r(w,n)$, $(r\ (w,n) = \{ < x,\ r\ (x,n) > :$ $: x \in w \},\ r\ (x,\ n) \in Y)$ reakcj# (odpowiedzi# funkcjonaln#) obiektu znajduj#cego si" w stanie niezawodno$ciowym $n,\ (n \in N)$ na wymuszenie $w,\ (w \in W(X))$. Reakcj" $r(w,\ n)$ nazywamy reakcj# wzorcow#. Wymuszenie $w \in W$ nazywamy wymuszeniem elementarnym, wymuszenie $w \in W(X) \setminus X$ wymuszeniem kompleksowym, a wymuszenie $w = X$ - wymuszeniem pe!nym.

Test funkcjonalny jest prób# funkcjonaln# – okre$lonym zadaniem lub zbiorem zada', przewidzianych do wykonania przez diagnozowany obiekt, a wynik takiego testu – podstaw# wnioskowania o stanie niezawodno$ciowym obiektu.

Niech $P(N') = \{N_1', \ldots, N_K'\}$ oznacza K - dzielny, $(K \ge 2)$ podzia! zbioru $N',(N' \subseteq N^*,\ Card\ N' \ge 2,\ N^* = \{n' \in N : P(n = n') > 0\})$

Wymuszenie w nazywamy testem wzgl"dem podzia!u $P(N')$ zbioru N' je&eli istniej# stany niezawodno$ciowe $n',\ n''$ nale&#ce do ró&nych podzbiorów tego podzia!u, w których reakcje obiektu na to wymuszenie s# ró&ne, to jest je&eli: $\exists\ [\ n',\ n'' \in N' : (n' \in N_j') \to (n'' \notin N_j',\ 1 \le j \le K\ ]:$ $: r(w,\ n') \neq r(w,\ n'')$

Wymuszenie w jest testem wzgl"dem dowolnej pary (n',n'') stanów niezawodno$ciowych w przypadku gdy: $r(w,\ n') \neq r(w,\ n'')$.

Niech $T\ [X^*,\ P(N')]$ oznacza zbiór testów wzgl"dem podzia!u $P(N')$ istniej#cych w zbiorze wymusze' $W(X^*),\ (X^* \subset X)$.

Wymuszenie w nazywamy testem kompletnym wzgl"dem podzia!u $P(N')$ je&eli jest testem dla ka&dej pary stanów niezawodno$ciowych nale&#cych do ró&nych podzbiorów tego podzia!u.

Niech $T^K[X^*,\ P(N')]$ oznacza zbiór testów kompletnych wzgl"dem podzia!u $P(N')$ istniej#cych w zbiorze $W(X^*),\ (X^* \subseteq X)$.

Test kompletny nazywamy nieredukowalnym testem kompletnym, je&eli dowolny podzbiór jego wymusze' elementarnych nie jest ju& testem kompletnym, to jest je&eli: $\forall T' \subset T : T' \notin T^K[X^*,\ (N')]$.

Niech $T_N^K[X^*,\ P(N')]$ oznacza zbiór nieredukowalnych testów kompletnych podzia!u $P(N')$ istniej#cych w zbiorze $W(X^*),\ (X^* \subseteq X)$.

Test wzgl"dem podzia!u $P(N') = \{n_0,\ N^* \setminus n_0\}$ nazywamy testem kontrolnym, natomiast test wzgl"dem podzia!u $P(N^* \setminus n_0)$ - testem lokalizacyjnym wzgl"dem wymaganej wnikliwo$ci lokalizacji stanu niezdatno$ci okre$lonej przez ten podzia!.

Wymuszenie w jest testem kontrolnym wzgl"dem niezdatno$ci $n',\ (n' \in N^* \setminus n_0)$ je&eli $r(w,\ n') \neq r(w,\ n_0)$.

Mo&na wykaza%, &e test lokalizacyjny (wzgl"dem dowolnego podzia!u $P(N^* \setminus n_0)$ jest jednocze$nie testem kontrolnym.

Cz"sto, w praktyce test kontrolny w obiektu jest wyznaczony jako suma (unia) testów kontrolnych wyznaczonych wzgl"dem ka&dej niezdatno$ci z pewnego zbioru $N_w,\ (N_w \in N^* \setminus n_0)$.

2.1. Metoda wyznaczenia testu

Metoda

Krok 1. Okre$li% zbiór $\Phi\ [P(N')]$ par $(n',\ n'')$ taki, &e:

$$\Phi[P(N')] = \{(n', n'') \in N' \otimes N' : (n' \in N) \rightarrow (n'' \notin N)\},$$
$$(n', n'' \in N', N \in P(N'))$$
$$X^*(n', n'') = \{x \in X^* : r(x, n') \neq r(x, n'')\}, (X^* \subseteq X).$$

Zbiór $X^*(n',\ n'')$ jest zbiorem wymusze' elementarnych nale&#cych do X^*, które s# testem wzgl"dem pary stanów niezawodno$ciowych $(n',\ n'')$. Sposób okre$lenia tego zbioru zale&y od rodzaju poszukiwanego testu (dla testu kontrolnego rozpatrzy% podzia! $P(N') = \{n_0, N^* \setminus n_0\}$, dla testu lokalizuj#ce rozpatrzy% podzia! $P(N^* \setminus n_0)$) oraz od znajomo$ci zbioru reakcji wzorcowych obiektu.

Krok 2. Okre$li% funkcj":

$f(\bar{x} \mid X^*, P(N')) = \forall\ (n', n'') \in \Phi\ [P(N')\ \exists\, x' \in X^*(n', n''): x'\ ]$.

Funkcja $f(\bar{x} \mid X^*, P(N'))$ oznacza tak# funkcj" bulowsk#, która przyjmuje warto$% 1 wtedy i tylko wtedy, gdy zbiór sk!adowych wektora $\bar{x}$ o warto$ci 1 jest elementem zbioru $T^K[X^*, P(N')]$. Jest ona przedstawiona w postaci normalnej formu!y koniunkcyjnej (nfk).

Krok 3. Przekszta!ci%, zgodnie z algebr# Boole'a, normaln# formu!" koniunkcyjn# funkcji („iloczyn sum" okre$lony w kroku 2) $f(\bar{x} \mid X^*, P(N'))$ do postaci minimalnej normalnej formu!y alternatywnej (mnfa – „sumy iloczynów").

Krok 4. Je&eli liczba termów minimalnej normalnej formu!y alternatywnej jest wi"ksza od 1 – wybra% term o najmniejszej liczbie elementarnych wymusze'.

Algorytm

W oparciu o opisan# metod" opracowano algorytm wyznaczaj#cy kompletny test kontrolny b"d#cy elementem zbioru $T^K[X^*, P(N')]$ dla pojedynczych niezdatno$ci sta!osygna!owych obserwowanych na wyj$ciach uk!adu kombinacyjnego.

Za!o&enia:

- diagnozowany obiekt traktowany jest jako czarna skrzynka;
- znany jest pe!ny wzorzec kontrolny obiektu;

Kroki algorytmu:

Krok 1. Poda% reakcj" (odpowied() wzorcow# uk!adu kombinacyjnego to jest: $r(w, n_0)$, $(w = X^*)$.

Krok 2. Poda% reakcje uk!adu kombinacyjnego dla stanów niezdatno$ci sta!osygna!owych wyj$% uk!adu. W ten sposób utworzony zostanie podzia! $P(N')$, zbioru N', taki &e stany niezdatno$ci n', n'' nale&#ce do ró&nych podzbiorów tego podzia!u odpowiadaj# uszkodzeniu: $\bar{s}_i(a)$, $(i \in \{1, 2, ..., Z\}, a \in \{0, 1\}$ - oznacza fizykalne stany niezawodno$ciowe uk!adu polegaj#ce na tym, &e uk!ad funkcjonuje tak, jak gdyby jego i – te wyj$cie mia!o stale stan logiczny a; Z – liczba wyj$%.

Krok 3. Utworzy% zbiór $\Phi\ [P(N')]$ o podziale:

- $P(n_0,\ N' \setminus n_0)$ dla wyznaczenie kompletnego nieredukowalnego testu kontrolnego;
- $P(N' \setminus n_0)$ dla wyznaczenie kompletnego nieredukowalnego testu lokalizacyjnego o wymaganej wnikliwo$ci stanu lokalizacji okre$lonego przez ten podzia!.

Krok 4. Dla ka&dego wymuszenia elementarnego x' okre$li% wszystkie pary niezdatno$ci (n', n'') ze zbioru $\Phi\ [P(N')]$, dla których to wymuszenie jest testem: $[\delta(x'(n'\ , n''))=1] \rightarrow [x' \in X^*(n',\ n'')]$. W ten sposób utworzono zbiór $T\ [X^*, P(N')]$

Krok 5. Nast"pnie w ka&dym kroku k procedury, wybra% do testu takie

$$\Phi_k[(P(N')] = \begin{cases} \Phi[P(N')],\ \text{dla k} = 1 \\ \Phi_{k-1}[P(N') \setminus \{n',\ n'') \in \Phi_{k-1}[P(N')]: \\ x(k) \in X^*(n',\ n'')\},\ \text{dla k} \geq\ 2 \end{cases}$$

$$X^*(k) = \begin{cases} X^*\ ,\ \text{dla k}\ = 1 \\ X^* \setminus \{x(1), ..., x(k-1),\ \text{dla}\ 2\ \leq\ \text{k}\ \leq\ \rho \end{cases}$$

wymuszenie elementarne $x(k)$, $(x(k) \in X^*(k)$, $X^*(k) \subseteq X^*$, $X^*(k)$ – zbiór wymusze' elementarnych dopuszczalnych do wybrania w k -tym kroku procedury, które jest testem wzgl"dem maksymalnej liczby par $(n',\ n'')$ stanów niezawodno$ciowych ze zbioru $\Phi_K[P(N')]$, przy czym:

$k, (k = 1,\ ...,\ \rho;\ 1 \leq \rho \leq \min\{Card\ \Phi\ [P(N')],\ Card\ X^*\})$,

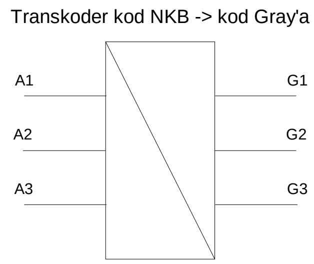
Krok 6. Zako' czy% wyznaczenie testu w kroku ρ takim, &e:

$$\{(n',\ n'') \in \Phi_\rho[P(N')]: x(\rho) \in X^*(n',\ n'')\} = \Phi_\rho[P(N')].$$

Zbiór $\{x(1)\ , ...,\ x(\rho)\}$, tak wybranych (ze zbioru X^*) wymusze' elementarnych, jest nieredukowalnym testem kompletnym wzgl"dem podzia!u $P(N')$.

Przykład 1

Wyznaczy%, w zbiorze $W(X)$, kompletny test kontrolny dla uk!adu cyfrowego przedstawionego na rysunku 1 wykrywaj#cy niezdatno$ci sta!osygna!owe pojedyncze obserwowane na wyj$ciach uk!adu.



Rysunek 1. Układ cyfrowy, dla którego wyznaczamy kompletny nieredukowalny test kontrolny

Sposób postępowania:

1) Poda% pe!ny wzorzec kontrolny obiektu, (tabela 1).

Tabela 1.

	$r(x_i,n_0)$		
	x_i	A3,A2,A1	G3,G2,G1
X	x_1	000	000
	x_2	001	001
	x_3	010	011
	x_4	011	010
	x_5	100	110
	x_6	101	111
	x_7	110	101
	x_8	111	100

2) Okre$li% podzia! $P(N')$ ze wzgl"du na pojedyncze niezdatno$ci sta!osygna!owe, (tabela 2).

Tabela 2.

$r(x_i,n_0)$			n_0	n_1	n_2	n_3	n_4	n_5	n_6
		A3,A2,A1							
X	x_1	000	000	00**0**	0**0**0	**0**00	00**1**	0**1**0	**1**00
	x_2	001	001	00**0**	0**0**1	**0**01	00**1**	0**1**1	**1**01
	x_3	010	010	01**0**	0**0**0	**0**10	01**1**	0**1**0	**1**10
	x_4	011	011	01**0**	0**0**1	**0**11	01**1**	0**1**1	**1**11
	x_5	100	100	10**0**	1**0**0	**0**00	10**1**	1**1**0	**1**00
	x_6	101	101	10**0**	1**0**1	**0**01	10**1**	1**1**1	**1**01
	x_7	110	110	11**0**	1**0**0	**0**10	11**1**	1**1**0	**1**10
	x_8	111	111	11**0**	1**0**1	**0**11	11**1**	1**1**1	**1**11
Uszkodzenia sta!osygna!owe				**G1(0)**	**G2(0)**	**G3(0)**	**G1(1)**	**G2(1)**	**G3(1)**

3) Dla ka&dego wymuszenia elementarnego x' okre$li% wszystkie pary niezdatno$ci (n', n'') ze zbioru $\Phi\ [P(N')]$, dla których to wymuszenie jest testem (tabela 3). Utworzono zbiór $T\ [X^*, P(N')]$

Tabela 3.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\ \{N'\setminus n_0\}\}]$					
		(n_0,n_1)	(n_0,n_2)	(n_0,n_3)	(n_0,n_4)	(n_0,n_5)	(n_0,n_6)
X	x_1				1	1	1
	x_2	1				1	1
	x_3	1	1				1
	x_4		1		1		1
	x_5		1	1	1		
	x_6	1	1	1			
	x_7	1		1		1	
	x_8			1	1	1	
$[\delta(x'(n',\ n''))=1]\rightarrow[x'\in X^{*}(n',\ n'')]$							

4) Wyznaczy% $x(k)$ dla kolejnych k, (tabela 4).

Tabela 4.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\ \{N'\setminus n_0\}\}]$		
		Liczba jedynek dla x(0)	**Liczba jedynek dla x(1)**	**Liczba jedynek dla x(2)**
X	x_1	3	0	0
	x_2	3	1	0
	x_3	3	2	0
	x_4	3	1	0
	x_5	3	2	0
	x_6	3	3	0
	x_7	3	2	0
	x_8	3	1	0

5) Kolejne wyznaczone wymuszenia przedstawiono poni&ej dla ka&dego kroku.
$\Phi_1[(P(N')]=x_1$; $\Phi_2[(P(N')]=x_6$

Dla wierszy z tabel: 3, 4 o równej liczbie jedynek w kroku k-tym algorytm wybiera wymuszenie x o najni&szym indeksie. Jednym z kompletnych testów kontrolnych dla tej klasy uszkodze' jest:

$\{x_1,\ x_6\}\in T_N^K[X,\ \{\{n_0\},\ \{n_1,\ ...,\ n_6\}\}]$ gdzie: $x_1=\{0,0,0\}$; $x_6=\{1,0,1\}$.

Rozwi#zanie analityczne:

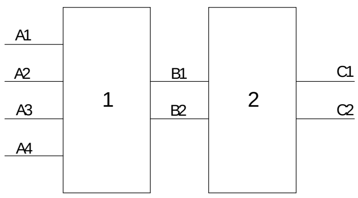
Na podstawie tabeli 3 wyznaczy% normaln# formu!" koniunkcyjn# *nfk* i przeksztа!ci% j# do postaci minimalnej normalnej formu!y alternatywnej mnfa.

$$(x_2+x_3+x_6+x_7)\cdot(x_3+x_4+x_5+x_6)\cdot(x_5+x_6+x_7+x_8)\cdot(x_1+x_4+x_5+x_8)\cdot(x_1+x_2+x_7+x_8)\cdot$$
$$\cdot(x_1+x_2+x_3+x_4)=x_1x_3x_5+x_1x_3x_7+x_1x_5x_7+\mathbf{x_1x_6}+x_2x_4x_6+x_2x_4x_8+x_2x_5+x_2x_6x_8+x_3x_5x_7+$$
$$+x_3x_8+x_4x_6x_8+x_4x_7$$

Wynik dzia!ania algorytmu (w tym przypadku) jest jednym z dwunastu mo&liwych nieredukowalnych kompletnych testów kontrolnych:

Przykład 2

Wyznaczy%, w zbiorze $W(X)$, nieredukowalny test kompletny lokalizuj#cy z maksymaln# wnikliwo$ci# pojedyncze niezdatno$ci sta!osygna!owe obserwowane na wyj$ciach uk!adów 1, 2 wchodz#cych w sk!ad uk!adu cyfrowego o pe!nym wzorcu kontrolnym opisanym w tabeli 5. Uk!ad przedstawiono na rysunku 2.



Rysunek 2. Układ cyfrowy, dla którego wyznaczamy nieredukowalny test kompletny o lokalizujący z maksymalną wnikliwością niezdatności stałosygnałowe na wyjściach układów: 1,2

Tabela 5.

$r(x_i,n_0)$				
	x_i	A4,A3,A2,A1	B2,B1	C2,C1
X	x_1	0001	00	00
	x_2	0010	01	01
	x_3	0100	10	11
	x_4	1000	11	10

Tabela 6.

$r(x_i,n_0)$		A4,A3,A2,A1	n_0	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8
X	x_1	0001	00	00	00	01	10	00	00	01	10
	x_2	0010	01	00	01	01	11	00	01	01	11
	x_3	0100	11	10	01	11	11	10	01	11	11
	x_4	1000	10	10	00	11	10	10	00	11	10
Uszkodzenia sta!osygna!owe				**B1(0)**	**B2(0)**	**B1(1)**	**B2(1)**	**C1(0)**	**C2(0)**	**C1(1)**	**C2(0)**

Rozwi#zanie dla przypadku wyznaczenia kompletnego testu lokalizacyjnego:

1) Utworzy% podzia! $P(N')$ ze wzgl"du na pojedyncze niezdatno$ci sta!osygna!owe, (tabela 6).

2) Dla ka&dego wymuszenia elementarnego x' okre$li% wszystkie pary niezdatno$ci (n', n'') ze zbioru $\Phi\,[P(N')]$, dla których to wymuszenie jest testem (tabela 7). Utworzono zbiór $T\,[X^*, P(N')]$

Tabela 7

$\delta(x'(n',n''))$		$\Phi\,[\{\{n_1\},\ ...,\ \{n_8\}\}]$										
		(n_1,n_2)	(n_1,n_3)	(n_1,n_4)	(n_1,n_5)	(n_1,n_6)	(n_1,n_7)	(n_1,n_8)	(n_2,n_3)	(n_2,n_4)	(n_2,n_5)	(n_2,n_6)
X	x_1		1	1		1		1	1	1		1
	x_2	1	1	1		1	1	1		1	1	
	x_3	1	1		1		1		1	1	1	1
	x_4	1	1	1	1		1	1	1	1	1	1
$[\delta(x'(n',\ n'')) = 1] \rightarrow [x' \in X^*(n',\ n'')]$												

$\delta(x'(n',n''))$		$\Phi\,[\{\{n_1\},\ ...,\ \{n_8\}\}]$										
		(n_2,n_7)	(n_2,n_8)	(n_3,n_4)	(n_3,n_5)	(n_3,n_6)	(n_3,n_7)	(n_3,n_8)	(n_4,n_5)	(n_4,n_6)	(n_4,n_7)	(n_4,n_8)
X	x_1		1	1	1		1	1	1	1	1	1
	x_2		1	1	1			1	1	1	1	1
	x_3	1	1	1		1	1	1	1		1	
	x_4	1	1			1	1			1	1	
$[\delta(x'(n',\ n'')) = 1] \rightarrow [x' \in X^*(n',\ n'')]$												

$\delta(x'(n',n''))$		$\Phi\,[\{\{n_1\},\ ...,\ \{n_8\}\}]$					
		(n_5,n_6)	(n_5,n_7)	(n_5,n_8)	(n_6,n_7)	(n_6,n_8)	(n_7,n_8)
X	x_1	1		1	1	1	1
	x_2	1	1	1		1	1
	x_3	1	1	1	1		1
	x_4	1	1		1	1	1
$[\delta(x'(n',\ n'')) = 1] \rightarrow [x' \in X^*(n',\ n'')]$							

3) Wyznaczy% $x(k)$ dla kolejnych k, (tabela 8).

4) Kolejne wyznaczone wymuszenia przedstawiono poni&ej dla ka&dego kroku.

$$\Phi_1[(P(N')] = x_1 \quad \Phi_2[(P(N')] = x_3$$

Dla wierszy z tabel: 7, 8 o równej liczbie jedynek w kroku k – tym algorytm wybiera wymuszenie x o najni&szym numerze.

Jednym z kompletnych testów kontrolnych (b"d#cym jednocze$nie nieredukowalnym kompletnym testem kontrolnym) dla tej klasy uszkodze' jest test: $\{x_1, x_3\} \in T_N^K[X,\ \{\{n_1,\ ...,\ n_8\}\}]$, gdzie: $x_1 = \{0,0,0,1\}$; $x_3 = \{0,1,0,0\}$. Mo&na zauwa&y%, &e uszkodzenie B1(0) b"dzie uto&samiane poprawn# prac# uk!adu, wi"c przed przeprowadzeniem testu lokalizuj#cego, w tym przypadku nale&y przeprowadzi% nieredukowalny kompletny test kontrolny dla sprawdzenie czy uk!ad dzia!a poprawnie. Je&eli uk!ad dzia!a niepoprawnie to mo&na okre$li%, w którym miejscu nast#pi!o uszkodzenie.

Tabela 8.

$\delta(x'(n',n''))$		$\Phi\ [\{\{n_1\},\ ...,\ \{n_8\}\}]$		
		Liczba jedynek dla x(0)	**Liczba jedynek dla x(1)**	**Liczba jedynek dla x(2)**
X	x_1	21	0	0
	x_2	21	4	0
	x_3	21	7	0
	x_4	21	7	0

Rozwi#zanie dla przypadku wyznaczenia kompletnego testu kontrolnego:

1) Utworzy% podzia! $P(N')$ ze wzgl"du pojedyncze uszkodzenia sta!osygna!owe, (tabela 6).
2) Dla ka&dego wymuszenia elementarnego x' okre$li% wszystkie pary niezdatno$ci (n', n'') ze zbioru $\Phi\ [P(N')]$, dla których to wymuszenie jest testem (tabela 9). Utworzono zbiór $T\ [X^*, P(N')]$

Tabela 9.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\{N'\setminus n_0\}\}]$							
		(n_0,n_1)	(n_0,n_2)	(n_0,n_3)	(n_0,n_4)	(n_0,n_5)	(n_0,n_6)	(n_0,n_7)	(n_0,n_8)
X	x_1			**1**	**1**		**1**		**1**
	x_2	**1**			**1**	**1**			**1**
	x_3		**1**	**1**		**1**		**1**	
	x_4	**1**	**1**				**1**	**1**	
$[\delta(x'(n',\ n'')) = 1] \rightarrow [x' \in X^*(n',\ n'')]$									

3) Wyznaczy% $x(k)$ dla kolejnych k, (tabela 10).

Tabela 10.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\{N'\setminus n_0\}\}]$			
		Liczba jedynek dla x(0)	**Liczba jedynek dla x(1)**	**Liczba jedynek dla x(2)**	**Liczba jedynek dla x(3)**
X	x_1	4	0	0	0
	x_2	4	2	1	0
	x_3	4	3	0	0
	x_4	4	3	1	0

Kolejne wyznaczone wymuszenia przedstawiono poni&ej dla ka&dego kroku algorytmu: $\Phi_1[(P(N')] = x_1$; $\Phi_2[(P(N')] = x_3$; $\Phi_3[(P(N')] = x_2$

Dla wierszy z tabel: 9, 10 o równej liczbie jedynek w kroku k – tym algorytm wybiera wymuszenie x o najni&szym indeksie. Jednym z nieredukowalnych kompletnych testów kontrolnych dla tej klasy uszkodze' jest test: $\{x_1, x_2, x_3\} \in T^K[X, \{\{n_0\}, \{n_1, ..., n_8\}\}]$

gdzie: $x_1 = \{0,0,0,1\}$; $x_2 = \{0,0,1,0\}$; $x_3 = \{0,1,0,0\}$.

Mo&na zauwa&y%, &e test $\{x_1, x_3\}$ jest podzbiorem nieredukowalnego kompletnego testu kontrolnego.

Wynikiem dzia!ania algorytmu w przedstawionych dotychczas przyk!adach by! kompletny test kontrolny, który jednocze$nie by! nieredukowalnym kompletnym testem kontrolnym. Mo&e si" jednak zdarzy%, &e otrzymana sekwencja testowa nie b"dzie nieredukowalnym kompletnym testem kontrolnym. Przyk!ad 3 pokazuje jak w takim przypadku przekszta!ci% otrzyman# sekwencj" testow# do postaci nieredukowalnego kompletnego testu kontrolnego.

Przykład 3

Wyznaczy% kompletny test kontrolny dla uk!adu cyfrowego realizuj#cego przekszta!cenie zgodnie z tablic# wzorcow# (tabela 11) dla pojedynczych niezdatno$ci sta!osygna!owych na wyj$ciach.

Tabela 11.

$r(x_i, n_0)$			
	x_i	A2,A1	G2,G1
X	X_1	00	00
	X_2	01	10
	X_3	10	10
	X_4	11	01

1) Poda% podzia! $P(N')$ ze wzgl"du na pojedyncze niezdatno$ci sta!osygna!owe, (tabela 12).

Tabela 12.

$r(x_i, n_0)$			n_0	n_1	n_2	n_3	n_4
		A2,A1					
X	x_1	00	00	**00**	**1**0	0**0**	0**1**
	x_2	01	10	**00**	10	10	1**1**
	x_3	10	10	**00**	10	10	1**1**
	x_4	11	01	**0**1	**1**1	0**0**	01
Uszkodzenia sta!osygna!owe				**G2(0)**	**G2(1)**	**G1(0)**	**G1(1)**

2) Dla ka&dego wymuszenia elementarnego x' okre$li% wszystkie pary niezdatno$ci (n', n'') ze zbioru $\Phi\,[P(N')]$, dla których to wymuszenie jest testem, (tabela 13). Utworzono zbiór $T\,[X^*, P(N')]$

Tabela 13.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\ \{N'\setminus n_0\}\}]$			
		(n_0,n_1)	(n_0,n_2)	(n_0,n_3)	(n_0,n_4)
X	x_1		**1**		**1**
	x_2	**1**			**1**
	x_3	**1**			**1**
	x_4		**1**	**1**	
$[\delta(x'(n',\ n''))=1] \rightarrow [x' \in X^*(n',\ n'')]$					

3) Wyznaczy% $x(k)$ dla kolejnych k, (tabela 14).

Tabela 14.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\{N'\setminus n_0\}\}]$			
		Liczba jedynek dla x(0)	**Liczba jedynek dla x(1)**	**Liczba jedynek dla x(2)**	**Liczba jedynek dla x(3)**
X	x_1	2	0	0	0
	x_2	2	1	0	0
	x_3	2	1	0	0
	x_4	2	1	1	0

4) Kolejne wyznaczone wymuszenia przedstawiono poni&ej dla ka&dego kroku.
$\Phi_0[(P(N')] = x_1$; $\Phi_1[(P(N')] = x_2$; $\Phi_2[(P(N')] = x_4$

Dla wierszy z tabel: 13, 14 o równej liczbie jedynek w kroku k – tym algorytm wybiera wymuszenie x o najni&szym indeksie. Jednym kompletnych testów kontrolnych dla tej klasy uszkodze' jest test: $\{x_1, x_2, x_4\}$, gdzie: $x_1 = \{0,0\}$; $x_2 = \{0,1\}$; $x_4 = \{1,1\}$.

Rozwi#zanie analityczne pokazuje, &e otrzymany wynik nie jest nieredukowalnym testem kontrolnym:

$$(x_2 + x_3)\cdot(x_1 + x_4)\cdot x_4 \cdot (x_1 + x_2 + x_3) = (x_2 x_4 + x_3 x_4)$$

W celu otrzymania nieredukowalnego testu kontrolnego nale&y dokona% redukcji otrzymanego wyniku. Sposób modyfikacji przedstawiono w tabeli 15. Analizuj#c wiersze tabeli 15 mo&na zauwa&y%, &e wymuszenie x_1 jest tu nadmiarowe, poniewa& testem wzgl"dem pary niezdatno$ci (n_0, n_2) jest równie& wymuszenie x_4, a testem wzgl"dem pary niezdatno$ci (n_0, n_3) jest równie& wymuszenie x_2. Na podstawie tej analizy mo&na z sekwencji testowej usun#% wymuszenie x_1, a otrzymana w ten sposób sekwencja testowa jest nieredukowalnym kompletnym testem kontrolnym.

Tabela 15.

$\delta(x'(n',n''))$		$\Phi[\{\{n_0\},\ \{N'\setminus n_0\}\}]$			
		(n_0,n_1)	(n_0,n_2)	(n_0,n_3)	(n_0,n_4)
X	x_1		1		1
	x_2	1			1
	x_4		1	1	
$[\delta(x'(n',\ n''))=1]\rightarrow[x'\in X^*(n',\ n'')]$					

3. Podsumowanie

W artykule przedstawiono algorytm wyznaczania zbioru nieredukowalnych testów kompletnych, który opiera si" na metodach s!ownikowych przedstawionych w [1]. W rzeczywisto$ci pewne rodzaje niezdatno$ci wyst"puj" cz"$ciej ni& inne. Znajomo$% prawdopodobie'stw wyst"powania poszczególnych niezdatno$ci jest przydatna przy projektowaniu testów, poniewa& pozwala dobra% wymuszenia o najwi"kszej skuteczno$ci kontrolnej i oszacowa% b!#d testu. Oczywi$cie dobór sekwencji wymuszenia nie jest uwarunkowany tylko rozk!adem prawdopodobie'stwa stanu niezawodno$ciowy obiektu traktowanego jako zmienna losowa. Bardzo wa&n# rol" w procesie generowania testu odgrywa koszt wysterowania okre$lonego wymuszenia. Ma to istotne znaczenie przy testowaniu obiektów, których praca zale&y od kolejno$ci u&ycia poszczególnych wymusze' elementarnych wchodz#cych w sk!ad testu.

Literatura:

[1] Kulesza R.: *Podstawy diagnostyki sieci logicznych i komputerowych*, IAiR, WAT, Warszawa, 2000

[2] Kulesza R.: *Niektóre teoretyczne podstawy tablicowych metod wyznaczania testów diadnostycznych*, Prace Przemys!owego Instytutu Elektroniki nr 106, Warszawa, 1988

[3] Sapiecha K.: *Testowanie i diagnostyka systemów cyfrowych*, PWN, Warszawa, 1987

[4] Arciuch A.: *Stanowisko laboratoryjne do testowania układów PLD z wykorzystaniem magistrali diagnostycznej IEEE 1149.1 (praca magisterska)* WAT, Warszawa, 1999

Recenzent: dr hab.inż. Roman Kulesza
Praca wpłynęła do redakcji: 10.10.2001