

## Sunkiai patikrinamų gedimų testavimas

M. Jančiukas, E. Bareiša

Programų inžinerijos katedra, Kauno technologijos universitetas

Studentų g. 50, LT-3031 Kaunas, Lietuva, tel.: +370 7 300629, 300361, el. paštas: minde@elen.ktu.lt, edas@soften.ktu.lt

### Įvadas

Sparčiai tobulėjant elektronikos technologijoms, schemas ir jų testavimas darosi vis sudėtingesni. Testų generatorių nuoseklioms schemoms algoritmai nuolatos tobulinami bei kuriami nauji. Nepaisant to, testų generatoriai pasiekia prastesnius rezultatus didėjant trigerių skaičiui nuoseklioje schemose. Tais atvejais, kai reikia patikrinti visus gedimus, neišvengiamai tenka naudoti pilnutinio ar dalinio skenavimo technologiją. Tačiau šios technologijos turi nemažų trūkumų: reikia nemažai papildomos aparatūros, prarandama galimybė nuoseklią schemą testuoti darbinio greičiu, be to, didžiąją testavimo trukmės dalį užima vien pasirinktų padėčių įstūmimas ir reakcijų į jas išstūmimas iš atminties taškų grandinės. Be abejo, siūloma nemažai šios technologijos patobulinimų – naudoti genetinius testų generavimo algoritmus bei kelias lygiagrečias trigerių grandines [1,3]. Tačiau šiais patobulinimais tik siekiama sumažinti schemų testavimo laiko sąnaudas.

Ankstesniuose tyrimuose testas nuosekliai sinchroninei schemai buvo gaunamas naudojant iteracinį modelį (IM) ir kombinacinį automatinio testinių rinkinių generatorių (Automatic Test Patterns Generator - ATPG) [5]. Taip gauti testai yra gana konkurencingi, tačiau, kaip ir kitų sistemų testai, palieka iki 50 proc. nepatikrintų gedimų.

Šiame straipsnyje testo nepatikrintų gedimų aibėje siūloma išskirti nepatikrinamų gedimų aibę, likusius įvardijant kaip sunkiai patikrinamus. Gedimams priskirti prie nepatikrinamų gedimų aibės bus taikoma metodika, pagrįsta IM modifikacijomis, legalių ir nelegalių padėčių aibių panaudojimu, apribojant grįžtamųjų ryšių padėčių aibę [2]. Sunkiai patikrinamiems gedimams testą siūloma kurti papildžius nuoseklią schemą aparatūra, didinančia atminties taškų kontroliuojamumą. Siūlant papildomos aparatūros struktūrą, nemažai dėmesio skiriama jos kiekio mažinimui. Toks *Scan* metodikai alternatyvus nuoseklių schemų testavimo būdas, garantuojantis 100 proc. testo pilnumą, susideda iš dviejų etapų: atminties taškų parinkimo stebėjimui ir nustatymui ir stebėjimo bei nustatymo būdų parinkimo.

Ekspperimentai atlikti su ITC'99 etaloninėmis schemomis, naudojantis Synopsys kombinaciniu ATPG.

### Reikiamų atminties taškų radimas schemos kontroliuojamumui padidinti

Nuosekliai schemai sudarius IM ir gavus testą, patikrinti gedimai  $P$  atmetami. Toliau analizuojami tik nepatikrinti gedimai. Pagal nepatikrinamų gedimų įrodymo metodiką [2] nustatyti gedimai  $U$  taip pat atmetami. Po šių dviejų veiksmų likę gedimai apibrėžiami kaip sunkiai patikrinami.

*1 apibrėžimas.* Nepatikrinamas gedimas – tai toks gedimas, kai nėra tokios testinės sekos, kuria būtų galima stebėti reakciją į jį.

*2 apibrėžimas.* Sunkiai patikrinamas gedimas – tai toks gedimas, kuriam dėl schemos sudėtingumo ar testavimo sistemos resursų stokos testų generatorius nesugeba rasti testinės sekos bei negalima įrodyti, kad tokia seka neegzistuoja.

Pradinė situacija – turime nuoseklią schemą ir sunkiai patikrinamų gedimų aibę  $G$ . Nuosekliai schemai galime sudaryti IM. Jis susideda iš nuoseklios schemos

kombinacinės logikos kopijų  $\Rightarrow K_i$ , sujungtų taip, kad

imituotų nuoseklios schemos veikimą, išskleistą laiko atžvilgiu.  $\Rightarrow$  žymi kopijų tarpusavio sujungimą grįžtamaisiais ryšiais. Kai iš nuosekliosios sinchroninės schemos atminties taškų reikšmės su kiekvienu taktu grįžta atgal į schemą, tai iteraciniame modelyje jos pereina iš  $K_i$  į

$K_{i+1}$ . Tiksliau IM galime apibrėžti taip:  $\Rightarrow K_1 \Rightarrow K_g \xrightarrow{m} K_j$ ,

kur  $K_i$  – kopijos, nustatančios reikiamas padėtis gedimams patikrinti;  $K_g$  – kopija, kurioje įvedami gedimai;  $K_j$  – kopijos, transportuojančios reakcijas į gedimus iki tikrųjų išėjimų. Bendras kopijų skaičius  $k=n+l+m$ . IM yra visiškai kombinacinė schema, kuriai taikomas kombinacinis ATPG.

Tikslas – nustatyti atminties taškus (AT), kuriuos kontroliuojant  $G$  aibės gedimai taptų patikrinami, t.y. ATPG šioms gedimams rastų testines sekas. Taip pasiekiamas 100 proc. testo pilnumas (TP).

$$TP = \frac{|P| + |G|}{\text{visi gedimai} - |U|} \times 100. \quad (1)$$

Atminties taškai stebėjimui ir nustatymui bus renkami

atskirai. Bus gautos dvi aibės: atminties taškų, kuriuos reikia stebėti, aibė  $S$  ir taškų, kuriuos reikia nustatyti, aibė  $N$ . Remiantis praktiniais tyrimais, aibę  $G$  galima suskirstyti į keturis poaibius. Jie skirstomi pagal patikrinimo reikalavimus, t.y. juos galima patikrinti:

- tik stebint tam tikrus AT,
- tik nustatant tam tikrus AT,
- stebint ar nustatant tam tikrus AT,
- stebint ir nustatant tam tikrus AT.

Papildoma aparatūra (PA) taip pat bus konstruojama stebėjimui ir nustatymui atskirai. Remiantis teoriniais apskaičiavimais, vidutiniškai PA, skirta vienam AT papildomai stebėti, bus mažesnė nei PA, skirta vienam AT nustatyti. Todėl, siekiant c) gedimų grupę patikrinti didinant stebimumą,  $S$  ir  $N$  aibių rinkimas bus pradedamas nuo aibės  $S$ . Be IM, bus naudojama logika  $SL$ , padedanti nustatyti, kurie AT buvo stebimi generuojant testą. Atsižvelgiant į šiuos teiginius sudaroma  $S, N$  aibių rinkimo procedūra:

(1)  $G := T \setminus P \cup U$ , kur  $T$  – visa gedimų aibė,  $S := \emptyset$ ,  $N := \emptyset$ ,  $r$  – bendras AT kiekis.

(2) Sudaromas IM, kurio struktūra apibrėžiama taip:  $\Rightarrow K_i \Rightarrow K_g$ . Taip pat sudaroma galimybė ATPG

stebėti po vieną  $K_g$  kopijos grįžtamųjų ryšių išėjimus  $RO_u$ ,  $u=1\dots r$ , atitinkančius nuoseklios schemos AT.  $u$  nusako, kuris AT bus stebimas.

(3)  $G'_u := \emptyset$ ,  $u=1\dots r$  ( $G'_u$  – aibių masyvas, kuriame bus saugomos patikrintų gedimų aibės, esant kiekvienam atviram  $RO_u$ ).

(4)  $u := 1$ .

(5) Jei  $RO_u \in S$ , einame į (7) žingsnį.

(6) Paleidžiamas ATPG IM-iui su stebimu  $RO_u$ .

Gaunama:  $G'$ ,  $S'$ .  $G'_u := G'$ .

(7)  $u := u + 1$ ; jei  $u \leq r$ , grįžtame į (5) žingsnį.

(8) Randame  $u$ , kuriam esant  $|G'_u|$  turi didžiausią vertę. Jei  $|G'_u| \neq \emptyset$ , tai  $G := G \setminus G'_u$ ,  $S := S \cup RO_u$ , kitu atveju pereiname į (9) žingsnį.

(9) Jei  $G = \emptyset$ , procedūra baigta.  $S$  – stebimų AT aibė;  $N$  – nustatomų AT aibė.

(10) Jei  $|G'_u| \neq \emptyset$ , IM modifikuojamas – atidaromas grįžtamojo ryšio išėjimas  $RO_u$  (toks IM bus naudojamas kituose žingsniuose) ir grįžtama į (3) žingsnį.

(11) Sudaromas IM, kurio struktūra apibrėžiama taip:  $\Rightarrow K_i \Rightarrow K_g \Rightarrow K_j$ . Sudaroma galimybė ATPG

nustatyti reikšmę pasirinktinai vienam iš  $K_g$  kopijos grįžtamųjų ryšių įėjimų  $RI_u$ ,  $u=1\dots r$ , atitinkančių nuoseklios schemos AT.  $u$  nusako, kuris AT bus nustatomas.

(12)  $G'_u := \emptyset$ ,  $S'_u := \emptyset$ ,  $u=1\dots r$  ( $G'_u$ ,  $S'_u$  – aibių masyvai, kuriuose bus saugomos patikrintų gedimų ir stebėtų grįžtamųjų ryšių išėjimų aibės, esant kiekvienam atviram  $RI_u$ ).

(13)  $u := 1$ .

(14) Jei  $RI_u \in N$ , einame į (16) žingsnį.

(15) Paleidžiamas ATPG IM-iui su nustatomu  $RI_u$ .

Gaunama:  $G'$ ,  $S'$ .  $G'_u := G'$ ,  $S'_u := S'$ .

(16)  $u := u + 1$ , jei  $u \leq r$ , grįžtame į (14) žingsnį.

(17) Randame  $u$ , kuriam esant  $|G'_u|$  turi didžiausią vertę.  $S := S \cup S'_u$ . Jei  $|G'_u| \neq \emptyset$ , tai  $G := G \setminus G'_u$ ,  $N := N \cup RI_u$ , kitu atveju pereiname į (18) žingsnį.

(18) Jei  $G = \emptyset$ , procedūra baigta.  $S$  – stebimų AT aibė,  $N$  – nustatomų AT aibė.

(19) Jei  $|G'_u| \neq \emptyset$ , IM modifikuojamas – atidaromas  $K_g$  kopijos grįžtamojo ryšio įėjimas  $RI_u$ ; tokia modifikacija išliks kituose žingsniuose ir grįžtama į (12) žingsnį.

(20) Sudaromas IM, kurio struktūra apibrėžiama taip:

$\Rightarrow K_i \Rightarrow K_g \Rightarrow SL$ . Sudaroma galimybė ATPG nustatyti

reikšmę pasirinktinai vienam iš  $K_g$  kopijos grįžtamųjų ryšių įėjimų  $RI_u$ , kur  $u=1\dots r$ , atitinkančių nuoseklios schemos AT.  $u$  nusako, kuris AT bus nustatomas. Grįžtame į (12) žingsnį.

Eksperimentas buvo atliekamas šiek tiek supaprastinus šią procedūrą. Renkant reikiamų stebėti AT aibę  $S$  ir nustatyti AT aibę  $N$  atitinkamai a), c) ir b), d) gedimų poaibiams, nebuvo naudojamos iteracijos, t.y. į šias aibes buvo įtraukiami tie AT, kuriuos parinko ATPG. IM buvo sudaromas prie  $K_g$  kopijos naudojant pagalbinę aparatūrą, leidžiančią ATPG išrinkti visus reikiamus AT vienu metu.

### Atminties taškų aibės stebėjimas

Atminties taškų aibėms stebėti siūlomi du būdai:

- stebėjimas naudojant esamus išėjimus (tikruosius);
- stebėjimas naudojant papildomą išėjimą per logiką „suma moduliu 2”.

Naudojant pirmąjį būdą reikalinga papildoma logika, valdoma papildomo valdymo įėjimo, užtikrina stebėjimą per jau esamus nuoseklios schemos išėjimus. Taigi schemoje taikomas testavimo režimas, kuris įjungiamas ar išjungiamas per valdymo įėjimą ('0' – išjungtas, '1' – įjungtas). Jei testavimo režimas išjungtas, schema veikia įprasto darbo režimu; įsijungus testavimo režimui tikrieji išėjimai naudojami atminties taškų reikšmėms išvesti.

Papildomos logikos kiekį ( $PK$ ) galime įvertinti ventilių kiekiu ( $V$ ), išvadų kiekiu ( $I$ ), laidų kiekiu ( $L$ ) priklausomai nuo stebimų atminties taškų kiekio ( $A$ ) ir schemos išėjimų kiekio ( $O$ ). Taigi

$$PK = V + L + I; \quad (2)$$

$$\text{čia } V = 4 \cdot A, L = 6 \cdot A, I = \begin{cases} 1, & \text{jei } A \leq O, \\ 1 + (A - O), & \text{jei } A > O. \end{cases} \quad (3)$$

Šio būdo naudojimas pasiteisina tik tuo atveju, jei atminties taškų, kuriuos reikia stebėti, skaičius neviršija schemos tikrųjų išėjimų skaičiaus.

Naudojant antrąjį būdą, reikalinga papildoma logika, kuri atlieka funkciją „suma moduliu 2” ( $XOR$ ) ir papildomas išėjimas. Papildomos logikos kiekis apskaičiuojamas pagal (2) formulę; čia

$$V = A - 1, L = 2 \cdot A - 1, I = 1. \quad (4)$$

Šiuo būdu papildomos aparatūros kiekis gaunamas mažesnis, tačiau išlieka galimybė, kad gali atsirasti gedimų maskavimas papildomame išėjime. Tai vyksta, kai reakcija į gedimą stebima keliuose atminties taškuose. Šios reakcijos maskuoja viena kitą papildomoje logikoje ir

išėjime nebematyti signalo pokyčio esant šiam gedimui. Ši problema gali būti sprendžiama XOR logikos medžių skaidant į kelis medžius, dubliuojant vieną iš atminties taškų, kuriame stebima reakcija į gedimą arba XOR logikos medyje padidinus įėjimų skaičių [4].

### Atminties taškų nustatymas į reikiamas padėtis

Atminties taškų nustatymo esmė – reikiamu momentu atminties taškams priskirti reikiamas reikšmes, kurios leistų aptikti likusius sunkiai patikrinamus gedimus. Šį reikalavimą siūloma realizuoti vienu iš dviejų būdų:

1. atminties taškus nustatyti per esamus (tikruosius) įėjimus;

2. naudoti papildomą aparatūrą, minimizuojančią naudojamų tikrųjų įėjimų skaičių.

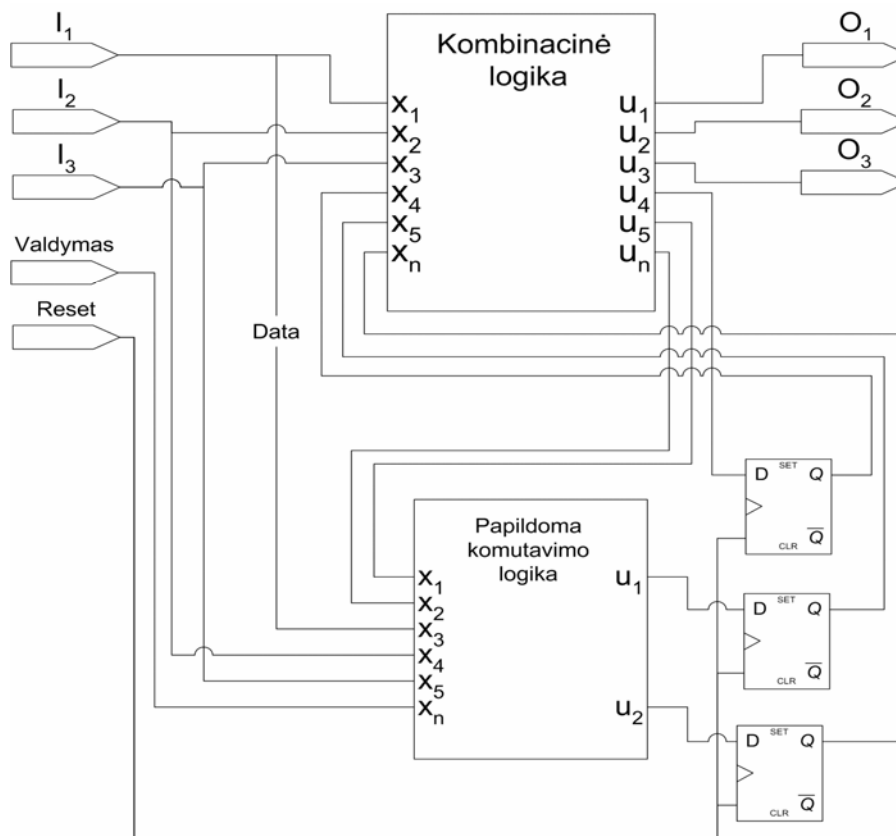
Pirmasis būdas yra lygiagretaus dalinio skenavimo atmaina, kai naudojama tik reikšmių ištūmimas į reikiamus atminties taškus. Naudojantis papildoma logika ir papildomu valdymo įėjimu, atminties taškai nustatomi į reikiamas padėtis per tikruosius įėjimus. Atsiranda papildomas testavimo režimas, kuris įjungiamas per valdymo įėjimą. Remiantis papildoma logika testavimo režimu reikiami atminties taškai lygiagrečiai nustatomi į reikiamas padėtis per tikruosius įėjimus. Papildomos logikos kiekį galime nusakyti tomis pačiomis formulėmis (2), (3), kurios naudojamos atminties taškų stebėjimo pirmo būdo logikos kiekiui įvertinti. Tik šiuo atveju vietoj schemos išėjimų skaičiaus reikėtų imti schemos įėjimų skaičių. Šį būdą būtų tikslinga naudoti tik tais

atvejais, jei reikiamų nustatyti atminties taškų yra ne daugiau nei tikrųjų įėjimų arba esant specifinei užduočiai, kai geriau naudoti didesnę papildomų įėjimų skaičių nei didesnę papildomos logikos kiekį.

Antrojo būdo esmė – reikiamų atminties taškų nustatymas remiantis papildoma logika ir papildomu valdymo įėjimu, tikruosius įėjimus naudojant atminties taško numeriui ir reikšmei perduoti (1 pav.). Jei schema veikia įprastiniu darbo režimu, t.y. valdymo signalas yra '0', komutavimo logika veikia kaip paprasčiausi laidai, t.y. įėjimų reikšmės perduodamos tiesiogiai į išėjimus. Jei valdymo signalas yra '1', schema persijungia į testavimo režimą. Komutavimo logika per tikruosius įėjimus gauna reikiamo atminties taško kodą, pagal kurį atrenkamas reikiamas trigeris, į kurį įrašoma duota reikšmė. Papildomos logikos elementų skaičių bus galima nusakyti tik praktiškai atlikus eksperimentą. Komutacinę logiką konkrečiu atveju, kai reikia nustatyti du atminties taškus, galima aprašyti taip, kaip parodyta 1 lentelėje.

1 lentelė. Komutavimo logikos veikimas

$X_n$	$X_4 X_5$	$U_1 U_2$
0	0 0	$X_1 X_2$
0	0 1	
0	1 0	
0	1 1	
1	0 0	$X_1 X_2$
1	0 1	$X_3 X_2$
1	1 0	$X_1 X_3$
1	1 1	$X_1 X_2$



1 pav. Atminties taškų nustatymas per tikruosius įėjimus, kuo mažiau naudojant tikruosius įėjimus

## Eksperimentas

Eksperimente buvo naudojamos ITC'99 etaloninės schemos. Informacija apie jas pateikiama 2 lentelėje.

**2 lentelė.** ITC'99 etaloninės schemos ir jų gedimų aibės  $P$ ,  $U$ ,  $G$

Schema	Ventilių skaičius, be trigerių	Trigerių skaičius	Iėjimų skaičius	Išėjimų skaičius	$P$	$U$	$G$	$T$
b01	40	5	4	2	268	0	0	268
b02	18	4	3	1	127	0	1	128
b03	111	30	6	4	615	199	8	822
b04	394	66	13	8	2395	201	44	2640
b05	570	34	3	36	1658	128	1576	3362
b06	48	9	4	6	326	20	0	346
b07	321	51	3	8	1574	49	575	2198
b08	133	21	11	4	898	0	20	918
b09	100	28	3	1	646	20	70	736
b10	137	17	13	6	889	27	36	952

Šioms schemoms pritaikius supaprastintą nustatomų ir stebimų atminties taškų radimo procedūrą gauti rezultatai pateikiami 3 lentelėje.

**3 lentelė.** Stebimi ir nustatomi AT

Schema	Stebimi AT, $S$	Nustatomi AT, $N$	$S/N$
b02	1	0	1
b03	4	0	4
b04	34	2	35
b05*	19	28	32
b07**	39	34	49
b08	2	4	4
b09	8	8	14
b10	6	4	10

\* - testo pilnumas 92,27 proc.

\*\* - testo pilnumas 97,63 proc.

$S/N$  bus naudojama įvertinti papildomai aparatūrai, apskaičiuotai pagal dalinio skenavimo metodologiją.

Toliau pateikiami siūlomos papildomos aparatūros, naudojamos aibės  $S$  atminties taškams stebėti (4 lentelė) ir aibės  $N$  atminties taškams nustatyti (5 lentelė), įvertinimai etaloninėje PdT elementinėje bazėje.

**4 lentelė.** Papildomos aparatūros kiekis AT-ams stebėti

Schema	1 būdas				2 būdas			
	V	I	L	Area	V	I	L	Area
b02	4	1	6	7	0	1	1	0
b03	16	1	24	25	3	1	7	6
b04	136	27	204	205	33	1	67	63
b05	76	1	114	115	18	1	37	33
b07	156	32	234	235	38	1	77	71
b08	8	1	12	13	1	1	3	4
b09	32	8	48	49	7	1	15	13
b10	24	1	36	37	5	1	11	11

**5 lentelė.** Papildomos aparatūros kiekis AT-ams nustatyti

Schema	1 būdas				2 būdas			
	V	I	L	Area	V	I	L	Area
b02	0	0	0	0	0	0	0	0
b03	0	0	0	0	0	0	0	0
b04	8	1	12	13	6	1	11	10
b05	112	26	168	169	153	4	170	269
b07	136	32	204	205	173	5	215	342
b08	16	1	24	25	16	1	24	24
b09	32	6	48	49	29	2	42	45
b10	16	1	24	25	16	1	24	24

Iš šių lentelių matome, kad stebėjimui visais atvejais reikia mažiau papildomos aparatūros taikant antrąjį būdą. Nustatymui mažiau logikos reikia taikant pirmąjį būdą tik sudėtingoms  $b05$ ,  $b07$  schemoms, tačiau antras būdas reikalauja mažiau papildomų išvadų. Taigi pasirinkę koks būdas labiau tinka stebėjimui ir nustatymui, jį galime realizuoti nuoseklioje schemoje. Siūlomos papildomos aparatūros kiekį palyginame su *Full-scan* ir *Partial-scan* papildomos aparatūros kiekiais (6 lentelė).

**6 lentelė.** Papildomos aparatūros rezultatai

Schema	Pilnutinio skenavimo aparatūra	Dalinio skenavimo aparatūra	Siūloma aparatūra			
			1,1	1,2	2,1	2,2
b02	35,00%	8,75%	8,75%	8,75%	0,00%	0,00%
b03	38,89%	5,19%	4,63%	4,63%	1,11%	1,11%
b04	31,09%	16,49%	14,67%	14,47%	5,11%	4,91%
b05	15,17%	14,28%	18,10%	24,47%	12,87%	19,25%
b07	28,70%	27,57%	35,37%	46,38%	22,19%	33,20%
b08	29,76%	5,67%	7,69%	7,49%	5,87%	5,67%
b09	38,58%	19,29%	19,29%	18,50%	12,20%	11,42%
b10	24,64%	14,49%	12,84%	12,63%	7,45%	7,25%

Iš rezultatų matome, kad tik  $b05$  ir  $b07$  schemoms eksperimentas nepasitvirtino. Tai galėjo sąlygoti  $S$ ,  $N$  aibių rinkimo supaprastinimas, dėl kurio sumažėjo sprendinio optimalumas. Dėl šio supaprastinimo  $S$ ,  $N$  aibių rinkimo uždavinys tapo per sudėtingas ATPG su turimais resursais ir šioms dviem schemoms nebuvo pasiektas 100 proc. testo pilnumas.

## Išvados

1. Pasiūlyta procedūra stebimų ir nustatomų atminties taškų aibėms rasti. Procedūros uždavinys: rasti kuo mažesnes šių atminties taškų aibes, leidžiančias tikrinti sunkiai patikrinamus gedimus.

2. Pasiūlyti atminties taškų stebėjimo ir nustatymo realizavimo būdai. Apibrėžta jų naudojimo specifika ir papildomos aparatūros sąnaudos.

3. Šių būdų pranašumai, palyginti su pilnutinio ir dalinio skenavimo metodologija, yra trumpesnis testas ir testavimas nuoseklios schemos veikimo greičiu. Papildomos aparatūros kiekis neturėtų būti didesnis nei taikant skenavimo technologiją.

4. Papildoma atminties taškų nustatymo logika įneša vėlinimus, kurie gali turėti įtakos schemos veikimui.

#### Literatūra

1. **Hamzaoglu I., Patel J. H.** Reducing Test Application Time for Full Scan Embedded Cores // Int. Symp. on Fault-Tolerant Computing. – June 1999. – P. 260-267.
2. **Jančiukas M.** Sunkiai patikrinamų gedimų aibės išskyrimas ir tikrinimas // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2002. – Nr.4(39) – P. 79-83.
3. **Rudnick E. M., Patel J. H.** A Genetic Approach to Test Application Time Reduction for Full Scan and Partial Scan Circuits // Proceedings of the Eighth International Conference on VLSI Design. – January 1995. – P. 288-293.
4. **Rudnick E. M.** Simulation-based techniques for sequential circuit testing // Thesis. – University of Illinois, 1994. – P. 119.
5. **Šeinauskas R., Bareiša E.** Test Generation for sequential circuit by means of combinational ATPG // Informacinės technologijos ir valdymas. – 1999. – Nr. 2(11) – P. 37-42

Pateikta spaudai 2003 06 23

**M. Jančiukas, E. Bareiša. Sunkiai patikrinamų gedimų testavimas // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2003. – Nr. 6(48). – P. 13-17.**

Ankstesniuose tyrimuose buvo išnagrinėtas sunkiai patikrinamų gedimų aibės išskyrimo metodas. Šiame straipsnyje pristatomas nuoseklių sinchroninių schemų testavimo būdas, užtikrinantis 100 proc. testo pilnumą. Pirmas šio būdo etapas – atminties taškų parinkimas stebėjimui ir nustatymui. Antrajame etape naudojami būdai, alternatyvūs skenavimo metodologijai, papildomo atminties taškų stebėjimo ir nustatymo realizavimui. Šio būdo pranašumai, palyginti su skenavimu, yra trumpesni testai bei schemos testavimas visu jos veikimo greičiu, esant tiek pat, o kartais mažiau papildomos aparatūros. Eksperimente naudojamos ITC'99 etaloniškos schemos. Darbo rezultatai apibendrinami išvadomis. Il. 1, bibl. 5 (lietuvių kalba; santraukos lietuvių, anglų ir rusų k.).

**M. Jančiukas, E. Bareiša. Testing of hard to detect faults // Electronics and Electrical Engineering. - Kaunas: Technologija, 2003. - No. 6(48). - P. 13-17.**

In previous researches method of selection of hard to detect faults was analyzed. In this article method of the synchronous sequential circuit's testing are discussed. Presented method guarantees 100% test coverage. The first stage of method is collecting sets of flip-flops for observing and controlling. In the second stage we use scan alternative way of implementation of additional flip-flop's controlling and observing. Proposed method's advantage over scan is less test application time and at-speed testing using the same and in some cases the less area overheads. ITC'99 benchmarks are used in experiment. Results of work are summarized by conclusions. Ill. 1, bibl. 5 (in Lithuanian; summaries in Lithuanian, English, Russian).

**М. Янчюкас, Э. Барейша. Тестирование сложно обнаруживаемых неисправностей // Электроника и электротехника. - Каунас: Технология, 2003. – №. 6(48). - С. 13-17**

В предыдущих исследованиях был анализирован метод выделения множества сложно обнаруживаемых неисправностей. В этой статье представлен метод тестирования синхронных последовательных схем, гарантирующий 100% полноту теста. В первом этапе производится сбор множеств элементов памяти для установления и наблюдения. Во втором этапе используются способы альтернативные сканированию для реализации дополнительного установления и наблюдения элементов памяти. Преимущество этого метода перед сканированием – более короткое время тестирования и скоростное тестирование при том же а в некоторых случаях и меньшем количестве дополнительной логики. Эксперимент проведен над ITC'99 эталонными схемами. Ил. 1, библи. 5 (на литовском языке; рефераты на литовском, английском и русском яз.).

DOI: 10.5755/j02.eie.11118