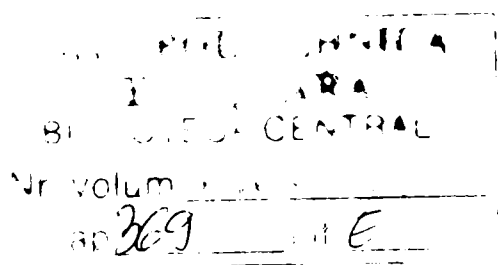


# Cercetări privind fenomenele termice la plăcile cu circuite integrate folosind imaginile termografice și senzorii de temperatură

(Teza doctorat)



Coordonator științific:  
prof. dr. ing. Mircea VLĂDUȚIU

Doctorand:  
Marius MARCU

2005



Dedicată mamei mele.



# Cuprins

1. Introducere .....	7
1.1 Importanța și necesitatea testării în contextul dezvoltării tehnologice .....	7
1.2 Temperatura și puterea consumată în contextul evoluției tehnologice .....	10
1.3 Clasificarea metodelor actuale de testare .....	16
1.4 Temperatura în circuitele integrate VLSI. Tendințe și probleme .....	21
1.5 Obiectivele tezei.....	22
1.6 Organizarea tezei.....	23
2 Testarea termică a circuitelor integrate .....	24
2.1 Tipuri de defecte fizice.....	24
2.1.1 Tehnologia de fabricare a circuitelor integrate VLSI.....	24
2.1.2 Defecte în straturile de metal .....	26
2.1.3 Defecte în substratul de siliciu .....	27
2.1.4 Defecte în stratul de oxid .....	27
2.2 Modelarea defectelor fizice.....	28
2.2.1 Defecte de scurt-circuit în circuitele combinaționale.....	29
2.2.2 Defecte de scurt-circuit în circuitele secvențiale.....	42
2.2.3 Defecte de întrerupere .....	44
2.3 Studiul transferului de căldură în circuitele integrate .....	50
2.3.1 Baza axiomatice a termodinamicii clasice .....	50
2.3.2 Generarea căldurii în dispozitivele electronice .....	51
2.3.3 Modurile elementare de transfer termic .....	52
2.3.4 Modelarea disipării căldurii .....	59
2.3.5 Modelarea disipării căldurii în circuitele integrate .....	61
2.4 Simularea electro-termică a circuitelor integrate .....	65
2.5 Testarea termică a circuitelor integrate .....	69
2.5.1 Aplicabilitatea testării termice .....	69
2.5.2 Testarea termo-optică.....	71
2.5.3 Testarea termo-electrică .....	74
2.5.4 Metoda de testare termică .....	76
2.5.5 Localizarea și diagnosticarea defectelor prin testarea termică.....	80
2.6 Probleme ale testării termice .....	81
2.6.1 Mecanismele de scurgere într-un tranzistor MOS.....	81
2.6.2 Soluții tehnologice.....	84
2.6.3 Semnătura termică.....	86
2.7 Generarea a vectorilor de test pentru testarea termică .....	89
2.8 Simularea defectelor de scurt-circuit rezistiv.....	93
2.9 Aplicația de studiu și simulare a defectelor de scurt-circuit rezistiv.....	102
2.9.1 Selecția defectelor și a vectorilor de test.....	102
2.9.2 Simularea defectelor circuitelor combinaționale.....	105
2.10 Concluzii .....	112
3 Testarea termică a plăcilor cu circuite integrate.....	115
3.1 Modelarea disipării temperaturii în plăcile cu circuite integrate.....	115
3.1.1 Modelul rezistențelor termice.....	117
3.1.2 Analiza numerică a câmpurilor staționare de temperaturi .....	124
3.1.3 Estimarea puterii disipate de circuitele integrate pe baza termogramei.....	126

3.1.4	Generarea imaginilor termice ideale .....	129
3.2	Deconvoluția imaginilor termice .....	134
3.2.1	Algoritm pseudo-genetic pentru deconvoluția imaginilor termice .....	135
3.2.2	Deconvoluția imaginilor termice cu surse de căldură neuniforme.....	148
3.3	Inspecția vizuală a plăcilor cu circuite integrate .....	150
3.4	Studiul imaginilor termice ale plăcilor cu circuite integrate.....	151
3.4.1	Metoda de abordare a lucrării.....	152
3.4.2	Analiza imaginilor termice.....	153
3.4.3	Analiza fenomenelor termice .....	156
3.5	Testarea pe bază de semnătură termică.....	157
3.6	Aplicația pentru studiul imaginilor în infraroșu .....	165
3.7	Concluzii .....	169
4	Monitorizarea și controlul temperaturii în sistemele de calcul .....	171
4.1	Metode electrice de măsurare a temperaturii .....	171
4.1.1	Tipuri de senzori de temperatură.....	171
4.2	Metode de conectare a senzorilor .....	174
4.2.1	Magistrala SMBus.....	174
4.2.2	Magistrala 1-Wire Bus .....	176
4.3	Semnătura termică a unui sistem de calcul.....	179
4.3.1	Implementarea aplicațiilor de timp real sub sistemul de operare Windows ..	179
4.3.2	Metode de măsurare a temperaturii folosind senzori de temperatură externi	186
4.3.3	Aplicația de monitorizare a temperaturii componentelor unui sistem .....	189
4.3.4	Studiul semnăturii termice a sistemelor de calcul.....	196
4.4	Semnătura termică a programelor .....	200
4.4.1	Mecanisme de reducere a temperaturii și puterii disipate .....	200
4.4.2	Proiectarea sistemelor cu consum optim de energie .....	204
4.4.3	Benchmark termic .....	206
4.4.4	Controlul puterii și temperaturii la nivelul aplicațiilor utilizator .....	210
4.5	Concluzii .....	221
5	Concluzii .....	223
Anexa 1	.....	228
Anexa 2	.....	229
Anexa 3	.....	231
Anexa 4	.....	232
Anexa 5	.....	233
Bibliografie.....	.....	234
Articole publicate de autor .....	.....	245

# 1. Introducere

## 1.1 Importanța și necesitatea testării în contextul dezvoltării tehnologice

În procesul de dezvoltare al unui sistem digital complex sau al unui simplu circuit integrat, testarea este extrem de importantă pentru obținerea unor produse de o înaltă fiabilitate, disponibilitate, siguranță și mentenabilitate. Firmele producătoare trebuie să garanteze funcționarea corectă a produselor lor și trebuie să furnizeze anumite date conținând caracteristicile de performanță ale produselor. Dacă o firmă producătoare livrează produse insuficient sau inadecvat testate, s-ar putea ca rata de produse defecte ajunse la utilizatori (client) să fie mare, ceea ce duce la creșterea nemulțumirii acestora și scăderea reputației producătorului, deci implicit la scăderea vânzărilor. Astfel că este necesară și foarte importantă utilizarea corectă a unor tehnici de testare corespunzătoare precum și înțelegerea și actualizarea lor pe măsură ce produsele și tehnologiile de fabricație evoluează în timp.

Testarea a rămas în actualitate însă a devenit o etapă din ce în ce mai costisitoare în procesul de producție al circuitelor integrate, al plăcilor cu circuite integrate și al sistemelor digitale. Costul optim al testării, într-un proces de fabricație complet, trebuie să nu depășească costul necesar înlocuirii sau reparării dispozitivelor defecte dar trebuie să asigure un procentaj de peste 99% al circuitelor fără defecte (Figura 1.1). [Abramovici1990]. Tendința costului testării este una crescătoare în timp ce prețurile de vânzare au o tendință descrescătoare, astfel că firmele producătoare sunt nevoite să investească mai mult în procesul de testare pentru a menține nivelul de calitate al produselor.

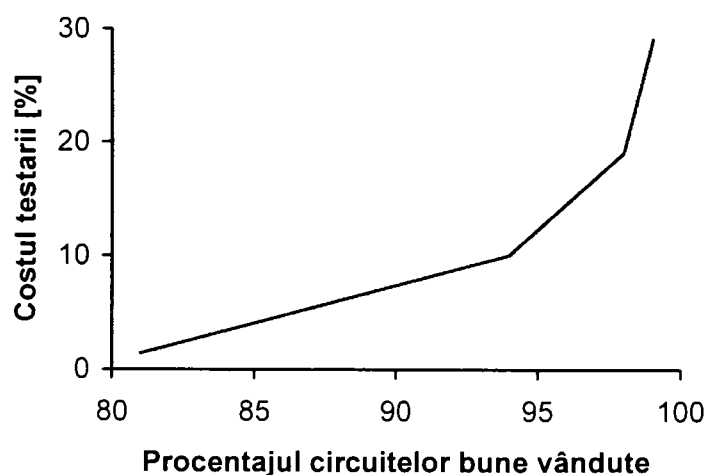


Figura 1.1 – Costul testării funcție de procentajul de circuite bune vândute [Abramovici1990]

Nivelul calității unui circuit integrat depinde de numărul de defecte apărute în timpul fabricației, de acoperirea vectorilor test aplicați, de gradul de utilizare al ariei de siliciu din chip și de viteza pentru care acesta este proiectat. Pentru a menține procentajul circuitelor vândute fără defecte peste o anumită limită trebuie investit tot mai mult în procesul de testare deoarece elementele care stau la baza evoluției tehnologiilor de fabricație au efect direct asupra calității produsului final.

Evoluția circuitelor integrate a fost determinată, datorită concurenței și cerințelor pieței, de două mari deziderate: acela de a oferi putere de calcul tot mai mare (mai multe funcționalități, complexitate ridicată) și viteză de procesare tot mai mare. În continuare, în

acest paragraf introductiv, va fi prezentat modul în care complexitatea și viteza circuitelor integrate influențează calitatea și fiabilitatea și determină procesul de testare.

Integrarea unui număr tot mai mare de funcții în circuitele integrate a fost posibilă prin creșterea ariei și a densității de integrare. Creșterea ariei de integrare influențează însă în mod dramatic calitatea circuitelor, figura 1.2 prezintă relația ce există între complexitate, cost și procentajul circuitelor considerate fără defecte din numărul total de circuite produse [Mazor1995].

Dacă examinăm figura 1.2 se poate observa că dacă pastila de siliciu este prea mică, pe ea nu se vor putea integra suficiente funcții pentru a se justifica prețul; iar dacă pastila de siliciu este foarte mare, factorul de producție va fi foarte mic iar circuitele rezultate vor fi foarte scumpe.

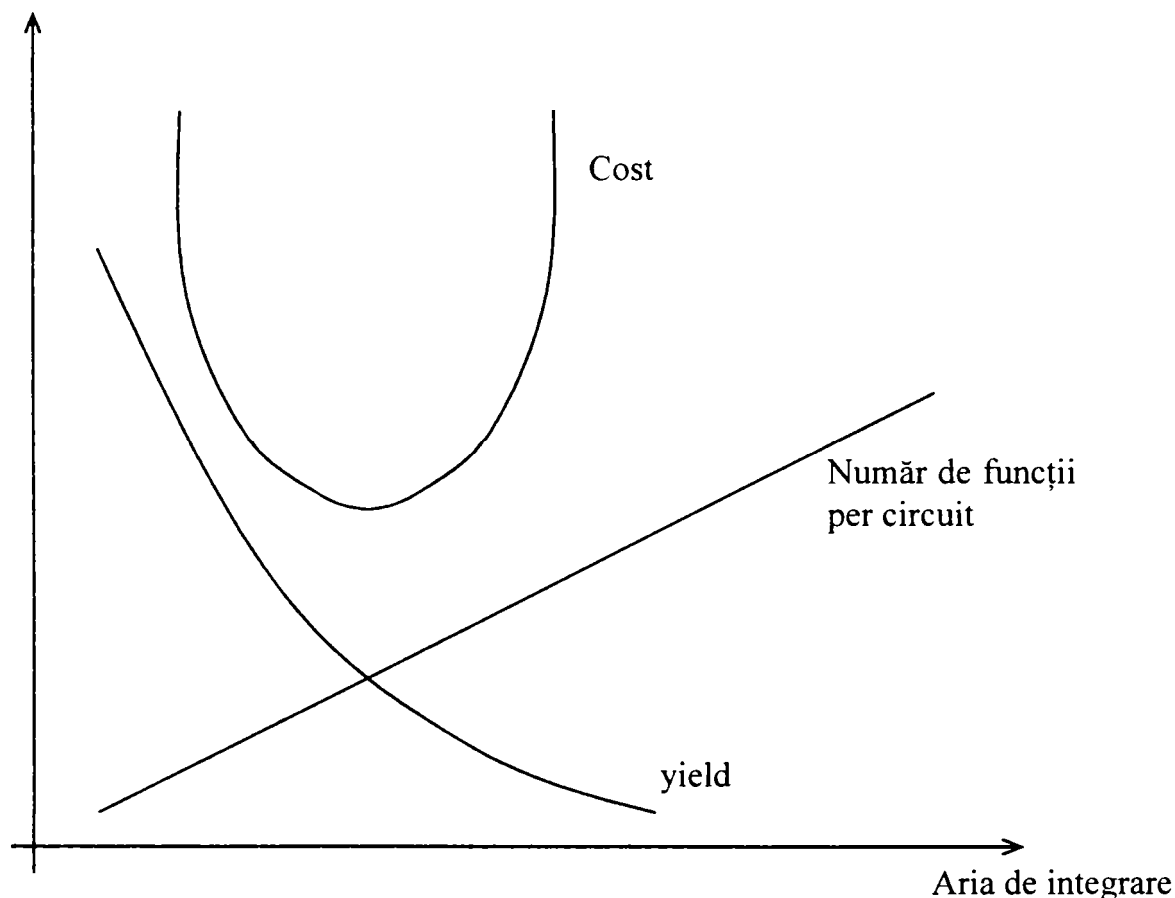


Figura 1.2 – Complexitate, cost, yield [Mazor1995]

Pentru estimarea calității circuitelor integrate se folosește rata de returnare (reject ratio) sau nivelul de defecte (defect level) definite ca raportul dintre numărul de circuite integrate defecte vândute ca și bune (numărul de circuite integrate defecte ce trec de faza de testare) și numărul de circuite integrate vândute [Chakravarty1997]. Unitatea de măsură pentru  $RR$  este ppm (parts per million).

$$RR = \frac{N_d}{N} \quad (1.1)$$

$N_d$  – numărul de circuite integrate defecte ce trec de faza de testare;  
 $N$  – numărul de circuite integrate ce trec de faza de testare.

Pentru estimarea  $RR$  există mai multe formule de calcul, printre care și cea prezentată în [Chakravarty1997].



$$RR = 1 - \left[ \frac{1 + Ab(1 - e^{-c})}{1 + Ab(1 - e^{-cf})} \right]^{-a} \quad (1.2)$$

în care s-a luat în calcul un factor de producție determinat cu formula [Chakravarty1997]:

$$y = [1 + Ab(1 - e^{-cf})]^{-a} \quad (1.3)$$

unde:

- $A$  – aria de integrare a circuitului;
- $f$  – procentajul de acoperire a defectelor a modelului de testare utilizat;
- $a, b, c$  – parametrii determinați empiric pentru calcul factorului de producție;
- $y$  – factorul de producție;
- $RR$  – rata de respingere.

Dacă examinăm ecuația de calcul pentru  $RR$  (1.2) se poate observa că pentru anumite valori ale parametrilor  $a$  și  $c$ ,  $RR$  este o funcție crescătoare de  $Ab$ . Pentru  $a = 1.636$ ,  $c = 0.67$  și un procentaj de acoperire a defectelor de 0.9, dependența dintre  $RR$  și  $Ab$  este cea prezentată în figura 1.3 [Chakravarty1997]. După cum se observă  $RR$  crește o dată cu creșterea ariei de integrare. Pentru a menține calitatea circuitelor la creșterea ariei de integrare trebuie să crească și procentajul de acoperire a defectelor a setului de teste utilizat sau adăugarea de noi teste cu un procentaj de acoperire mai mare. În exemplul prezentat, pentru a menține rata de returnare din cazul  $Ab = 1.5$  la nivelul cazului în care  $Ab$  era 0.25, ar trebui ca procentajul de acoperire a defectelor să crească de la 0.9 la peste 0.97.

$Ab$	0.25	0.75	1.0	1.25	1.5
$RR$ [ppm]	12907	31684	38726	44685	49792

Tabelul 1.1 – Calitatea funcție de aria de integrare [Chakravarty1997]

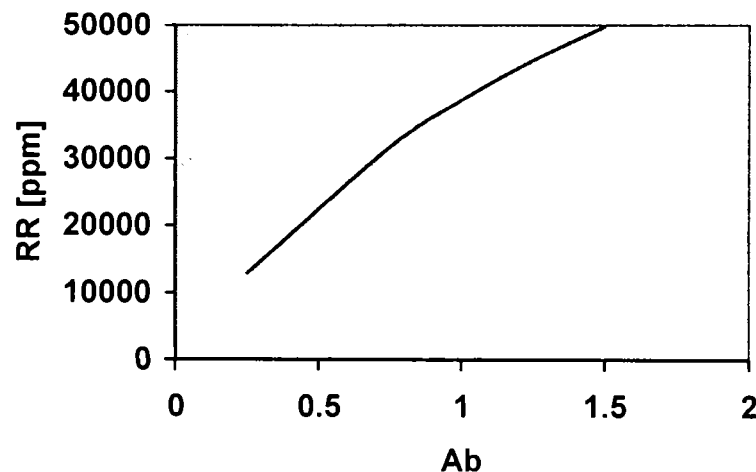


Figura 1.3 – Dependența calității funcție de aria de integrare

Pentru circuite de înaltă calitate,  $RR$  ar trebui să fie sub 500 dpm, iar acoperirea defectelor pentru setul de teste aplicat este de ~99%.

Viteza, pe de altă parte, este un alt factor ce influențează calitatea și fiabilitatea circuitelor integrate și a plăcilor cu circuite integrate. La frecvențe de sub 1 GHz, la

proiectarea plăcilor cu circuite integrate se pot ignora efectele electromagnetice ale semnalelor. Dar la frecvențe de peste 1 GHz, aceste efecte nu mai pot fi ignorate [Ind2002], deoarece apar rezonanțe, intercuplări, zgomote ce pot duce la funcționarea eronată sau instabilă a sistemului. Reducerea semnificativă a marginii de zgomote împreună cu frecvența foarte mare cresc dramatic impactul defectelor tranzitorii. Din această cauză este foarte utilă, aplicându-se tot mai mult, dotarea circuitelor (în special circuitele VLSI) cu facilități integrate de testare și autotestare, pentru testarea on-line (în timpul funcționării) a acestora.

Temperatura este un alt parametru influențat direct de frecvența de lucru iar temperatura influențează la rândul ei fiabilitatea circuitelor. Influența temperaturii asupra calității circuitelor este discutată în paragraful următor.

După cum se poate acum observa, cele două mari caracteristici care au influențat evoluția circuitelor integrate (densitatea de integrare și viteza) influențează în mod direct calitatea circuitelor. În plus, densitatea circuitelor integrate va continua să crească, în timp ce numărul de pini va rămâne relativ constant (tabelul 1.4). Acest lucru induce o escaladare a complexității și testarea devine unul din costurile majore ale industriei (estimată la aproximativ 30% din întregul cost de producție dar poate ajunge chiar până la 50% din costul circuitului).

Circuitele integrate trebuie testate înainte și după împachetare, după montarea pe o placă și periodic în timpul funcționării. Pentru fiecare caz sunt necesare metode diferite de testare. Tehnicile de testare trebuie să se dezvolte și ele pe măsură ce densitatea de integrare și viteza continuă să crească, pentru a se menține calitatea acestora.

## 1.2 Temperatura și puterea consumată în contextul evoluției tehnologice

În ultimii ani, progresele realizate în domeniul microelectronicii au fost foarte mari și foarte rapide, în special în ceea ce privește tehnologia de fabricație a circuitelor integrate VLSI. Pentru a oferi o putere de calcul tot mai mare (mai multe funcționalități, complexitate ridicată) și viteză de procesare tot mai mare, la baza evoluției circuitelor integrate a stat așa numitul proces de scalare. Scalarea este procesul prin care se poate crește densitatea de integrare și viteza de operare, prin micșorarea dimensiunilor tranzistorilor, scăderea tensiunii de prag și a tensiunii de alimentare (tabelul 1.2). Pe un plan secund au fost lăsate alte caracteristici importante cum ar fi puterea consumată, temperatura disipată și în unele cazuri chiar calitatea și fiabilitatea circuitelor, care au fost în mod direct afectate de procesul de scalare. Însă, în ultimul timp, datorită problemelor care apar, tot mai mult se discută despre controlul puterii consumate și scăderea temperaturii disipate de circuitele integrate VLSI. În continuare, în acest paragraf, va fi prezentat modul în care evoluția a influențat puterea consumată și temperatura disipată de circuitele integrate, precum și efectul acestora asupra fiabilității circuitelor.

Anul	2000-2006	2020
Dimensiunea canalului	0.13-0.10 $\mu\text{m}$	0.018 $\mu\text{m}$
Grosimea stratului de oxid	$\sim 20\text{\AA}$	$< 10\text{\AA}$
Curentul de scurgere	$\sim 0.05 \text{ nA}/\mu\text{m}^2$	$\sim 0.05 \mu\text{A}/\mu\text{m}^2$
Tensiunea de prag	0.3V	$< 0.1\text{V}$
Tensiunea de alimentare	1V	$< 0.6\text{V}$

Tabelul 1.2 – Caracteristicile procesului de scalare [Chung2001]

Urmărind și extrapolând evoluția istorică a tehnologiilor de fabricație a circuitelor integrate (tabelul 1.3) și în special a procesoarelor (tabelul 1.4), cunsumul unui procesor va ajunge la 200W în următorii 10 ani, cu o medie a densității de putere pe unitatea de suprafață de 125 W/cm<sup>2</sup>. Principalele cauze ale creșterii puterii consumate sunt creșterea densității de integrare (în special datorită micșorării grosimii stratului de oxid) și a dimensiunilor pastilei de siliciu, numărul tot mai mare de tranzistoare integrate într-un circuit și frecvența de operare tot mai mare. Pentru a limita creșterea consumului de putere datorită procesului de scalare, o dată cu scăderea dimensiunilor tranzistoarelor s-a impus și micșorarea tensiunii de alimentare.

Anul	1997	1999	2001	2003	2006	2009	2012
Tehnologia [ $\mu\text{m}$ ]	0.25	0.18	0.15	0.13	0.10	0.07	0.05
Număr tranzistoare [ $\text{mil}/\text{cm}^2$ ]	3.7	6.2	10	18	39	84	180
Frecvența [MHz]	750	1250	1500	2100	3500	6000	10000
Dimensiunea pastilei [ $\text{mm}^2$ ]	280	400	445	560	790	1120	1580
Tensiunea de alimentare [V]	2.5-1.8	1.8-1.5	1.5-1.2	1.5-1.2	1.2-0.9	0.9-0.6	0.6-0.5
Puterea consumată [W]	70	90	110	130	160	170	175
$T_J/T_A$ [ $^{\circ}\text{C}$ ]	125/55	125/55	125/55	125/55	125/55	125/55	125/55

Tabelul 1.3 – Evoluția tehnologică [Sofia2000]

Cerințele pieței pentru microprocesoare tot mai performante (tabelul 1.4) a condus la creșterea continuă și accentuată a puterii consumate de acestea și deci și a aplicațiilor în care sunt folosite [Gunther2001, Viswanath2000, Chung2001]. Creșterea consumului de putere nu afectează doar costul de utilizare al produsului ci și performanța și durata de viață a acestuia.

Procesor	Anul	Număr de tranzistoare	Număr de pini	Tehnologie [ $\mu\text{m}$ ]	$V_{DD}$ [V]	$F_{max}$ [MHz]	$P_{max}$ [W]
4004	1971	2250	16	10	12	0.1	
8080	1974	4500	40	6	5	3	
8086	1978	29000	40	2	5	12	
80286	1982	134000	68	1.5	5	20	
80386	1985	275000	132	1	5	33	1.9
80486	1989	1200000	168	0.8	5 - 3.3	66	6
Pentium	1994	3100000	320	0.6 - 0.35	5 - 3.3	200	15
Pentium Pro	1997	5500000	387	0.35	3.3 - 2.9	300	47
Pentium 2	1998	7500000	242	0.25	3.3 - 2.1	500	27
Pentium 3	1999	9500000	242	0.18	3.3 - 1.3	1200	37
Pentium 4	2000	55000000	478	0.13	1.75 - 1.5	1500	57
Pentium 4	2001	55000000	478	0.13	1.75 - 1.5	2000	75
Itanium	2001	25000000		0.18	3.3	800	116
Pentium 4	2002	55000000	478	0.13	1.75 - 1.5	3000	81
Itanium 2	2002	220000000		0.18	3.3	1000	130

Tabelul 1.4 – Evoluția procesoarelor Intel

O tendință similară de creștere o are și fluxul de putere (puterea disipată pe unitatea de suprafață) (figura 1.4). Această creștere se datorează faptului că mecanismele de reducere a

puterii implementate prin schimbări arhitecturale și tehnologice nu pot compensa creșterea consumului de putere datorată creșterii numărului de tranzistoare și a frecvenței de lucru. Pentru a avea o referință pentru valorile din figura 1.4, la periferia Soarelui, unde temperatura este de aproximativ 6000 °C, fluxul de căldură e în jur de  $10^7 \text{ W/m}^2$ , iar în chip-ul unui circuit poate ajunge la  $100 \text{ W/cm}^2$  adică  $10^6 \text{ W/m}^2$ , doar cu un ordin de mărime mai mic decât fluxul de căldură de la periferia Soarelui.

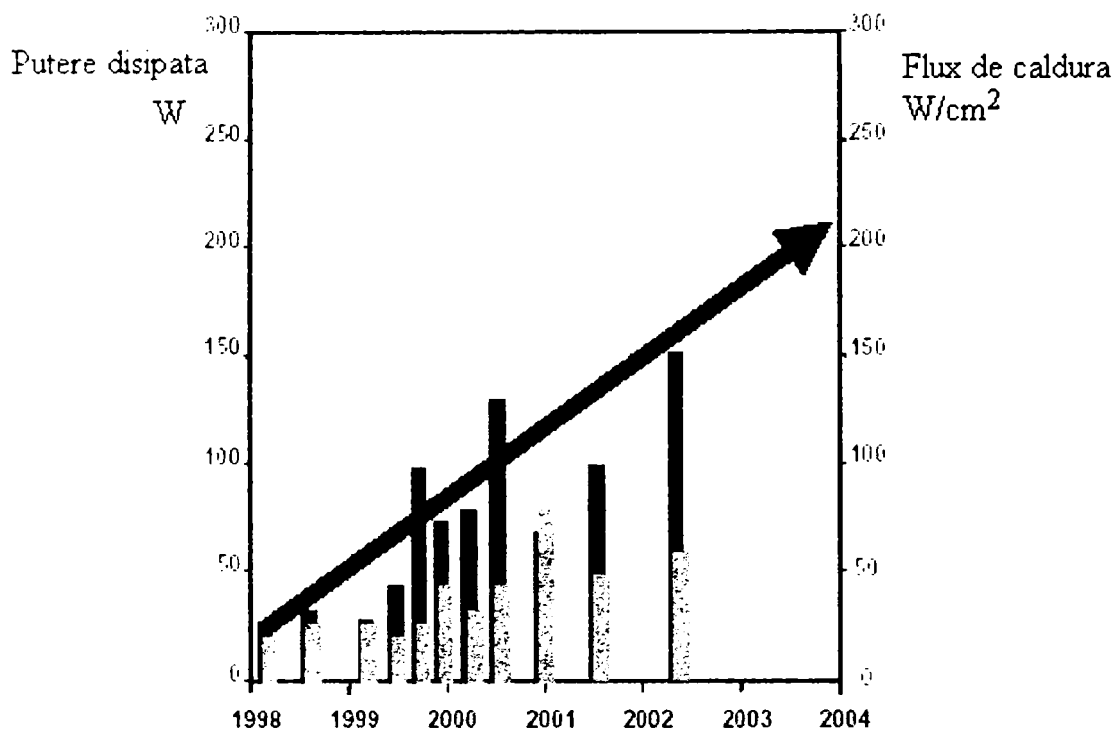


Figura 1.4 – Creșterea puterii consumate și a fluxului de căldură

Pentru a evidenția distribuția consumului între componentele unui sistem se poate urmări tabelul 1.5. Se poate observa că o pondere foarte mare din consumul total al unui sistem o are microprocesorul.

Componenta	Putere consumată [W]	Pondere [%]
Microprocesor	60-105	30
Monitor	55-110	30
HDD/CD-ROM	10-15	7
Boxe	10	3.5
Modem/Rețea	10	3.5
Imprimantă	75	26

Tabelul 1.5 – Distribuția consumului de putere într-un sistem [Viswanath2000]

Un al doilea exemplu de distribuție a consumului de putere între componentele unui sistem portabil se poate vedea în tabelul 1.6. Componentele în acest al doilea exemplu sunt de consum redus.

Componenta	Putere consumată [W]	Pondere [%]
Microprocesor	2.24	28
Ecran	2.80	35
HDD	0.64	8
Video	0.80	10
Memorie	0.16	2
Altele	1.36	17

Tabelul 1.6 – Distribuția consumului de putere într-un sistem portabil [Lorch98]

Principalul parametru care este influențat în mod direct de creșterea puterii consumate de un circuit integrat este temperatura [Wiklund1998]. Temperatura este un parametru extrem de important în contextul circuitelor integrate deoarece creșterea acesteia influențează negativ atât performanța circuitului cât și fiabilitatea acestuia.

Efectele creșterii temperaturii sunt:

- scăderea vitezei – viteza scade cu 0.15% pentru fiecare creștere a temperaturii cu 1 °C. Creșterea temperaturii afectează timpii de comutare la nivel de poartă logică (vezi figura 1.5);
- scăderea fiabilității – durata de viață se diminuează cu 50% la o creștere a temperaturii cu 10-15 °C. Creșterea temperaturii afectează și degradează fizic structura de siliciu ducând la defectarea prematură a circuitului;
- creșterea curenților reziduali –  $I_{OFF}$  crește cu temperatura. Poate apare fenomenul de fugă termică (thermal runaway) datorită creșterea curenților reziduali implică o încălzire și mai puternică a circuitului, care la rândul ei produce un curent mai mare, ș.a.m.d.

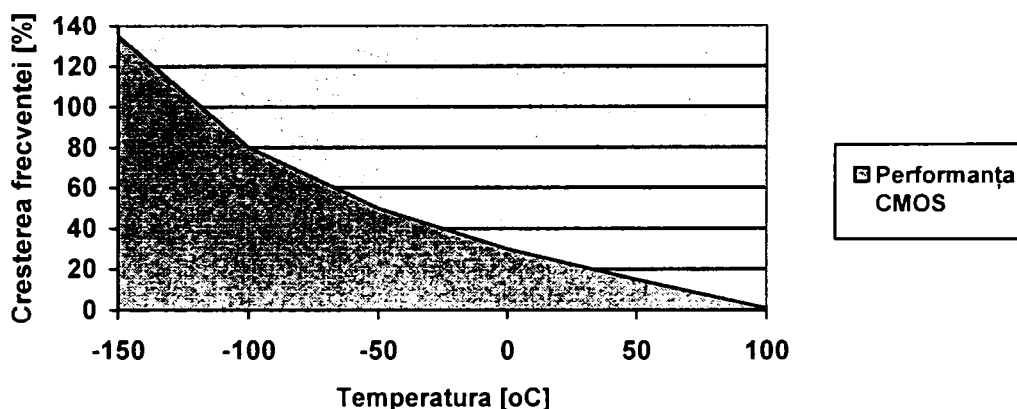


Figura 1.5 – Performanța vs. temperatură [Sofia2000]

Efectul cel mai grav și distructiv al creșterii temperaturii este fenomenul de fugă termică [Krauss1987, Sony2001]. La echilibru termic, puterea disipată de joncțiunea tranzistorului este egală cu căldura transmisă de la joncțiune la mediul înconjurător. Se știe că la creșterea temperaturii joncțiunii va crește și curentul prin tranzistor. Această creștere a curentului cauzează o creștere a puterii disipate, care la rândul ei, datorită rezistenței termice fixe dintre joncțiune și mediu, duce la o creștere a temperaturii joncțiunii. Din nou curentul prin joncțiune crește, la fel și puterea disipată, la fel și temperatura joncțiunii, ș.a.m.d. Se observă că poate apărea un ciclu regenerativ de căldură. Acest fenomen poate cauza defecțiuni catastrofice ale dispozitivului, dacă acesta nu este proiectat și răcit corespunzător.

Din practică s-a constatat că cea mai întâlnită cauză de defectare la circuitele integrate este temperatura, 55% din totalul cauzelor de defectare se datorându-se acesteia (vezi figura 1.6). Circuitele integrate au specificate ca și date de catalog și valorile maxime admisibile pentru temperatura joncțiunii. Dacă se depășește această valoare circuitul riscă să se defecteze și se degradează mai repede. Din acest motiv, chiar dacă puterea consumată de un circuit integrat crește, temperatura acestuia trebuie păstrată sub valoarea maximă admisibilă, prin utilizarea mecanismelor de răcire corespunzătoare.

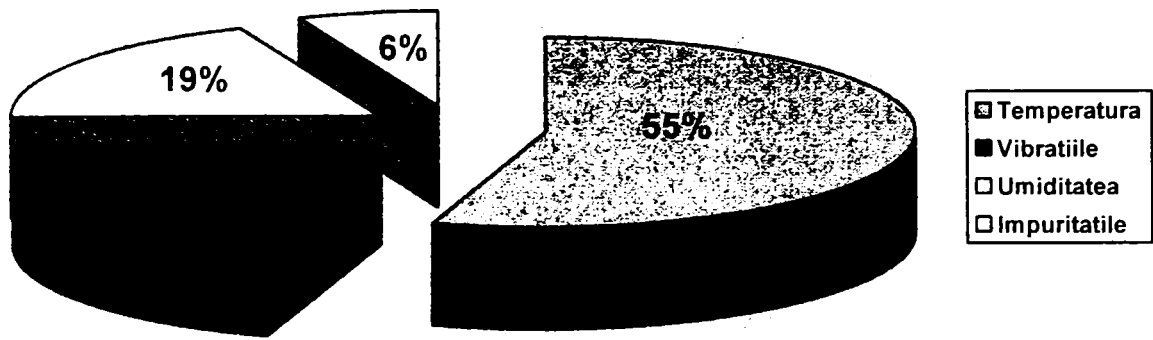


Figura 1.6 – Principalele cauze de defectare ale circuitelor integrate [Sofia2000]

Durata de viață a dispozitivelor semiconductoare este sensibil dependentă de temperatură. Reacția la stresul termic a fost standardizată de Arrhenius, iar modelul propus de el este utilizat pentru prezicerea duratei de viață a dispozitivelor semiconductoare (Ecuația 1.4). [Wiklund1998. Sony2001]

$$R(T) = A \cdot e^{-\frac{E_A}{kT}} \quad (1.4)$$

unde:

- R – durata de viață
- A – constantă ce nu depinde de temperatură
- $E_A$  – energia de activare [eV]
- T – temperatura absolută [K]
- k – constanta lui Boltzmann

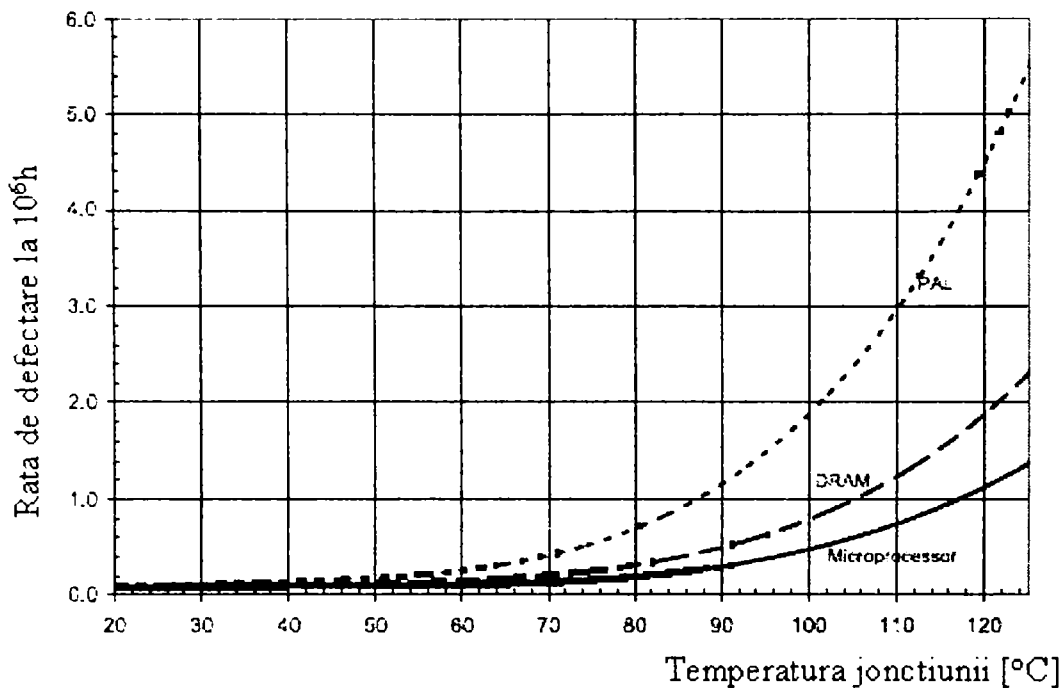


Figura 1.7 – Rata de defectare a diferitelor tipuri de componente în funcție de temperatura jonctiunii [Remsburg2001]

Energia de activare este energia necesară moleculelor pentru a participa la reacție. În cazul circuitelor integrate participarea moleculelor la o reacție termică duc la apariția

anumitor defecte (Tabelul 1.7). Energia de activare este o măsură a efectului pe care temperatura o are asupra duratei de viață a circuitelor.

Defect	$E_A$ [eV]
Scurt-circuite poartă-oxid	0.3
Contaminare	1.0
Defecte în substratul de siliciu	0.5
Migrația electronilor la nivelul traseelor	0.5
Migrația electronilor la nivelul contactelor	0.9
Pierderea capacității de memorare pentru DRAM	0.6

Tabelul 1.7. – Energia de activare termică tipică pentru diferite tipuri de defecte [Wiklund1998]

Pentru microprocesoare s-a constatat că puterea consumată depinde de tipul aplicației executate (figura 1.8), astfel că în timpul funcționării puterea variază în funcție de aplicațiile executate la un moment dat. Sistemul trebuie astfel proiectat încât să asigure funcționarea procesorului sub limita maximă specificată pentru temperatură, chiar dacă puterea consumată crește până la valoarea maximă. Pentru a satisface acest deziderat trebuie utilizate sisteme de răcire tot mai performante, la prețuri cât mai mici.

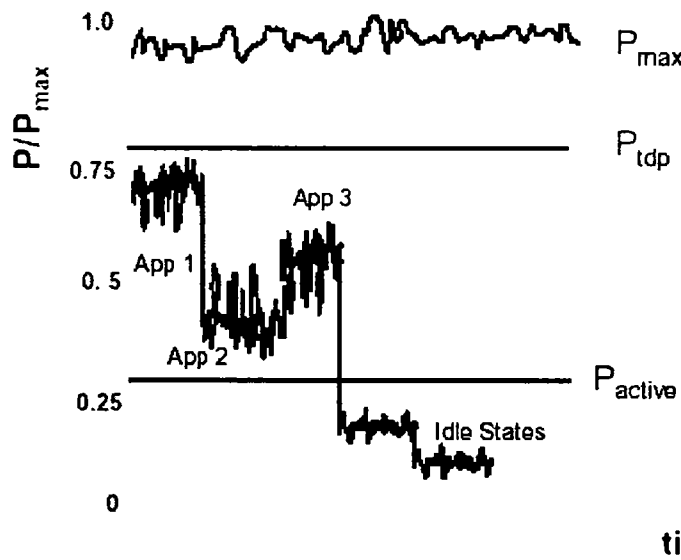


Figura 1.8 – Variația puterii consumate funcție de aplicație [Viswanath2000]

Un al doilea parametru afectat de creșterea puterii consumate este curentul absorbit de circuit. Curentul absorbit de circuit crește atât datorită creșterii puterii consumate cât și datorită micșorării tensiunii de alimentare. De exemplu, pentru un procesor Pentium 4 ce consumă 75W, la o tensiune de alimentare de 1.5V, va absorbi un curent de ~50A.

Efectele creșterii curentului sunt:

- apariția căderilor de tensiune pe liniile de alimentare;
- afectarea tranzițiilor, fronturile semnalului de tact;
- scăderea fiabilității și apariția defectelor (scurt-circuite și întreruperi datorită migrării electronice).

În al treilea rând, costul soluțiilor de răcire depinde și el de temperatură. În ultimii ani, în primul rând datorită concurenței, tendința prețurilor de livrare a dispozitivelor electronice a fost într-o continuă scădere în ciuda creșterii performanțelor acestora. Creșterea puterii consumate de dispozitivele electronice și a temperaturii disipate de acestea impun însă costuri suplimentare în direcția reducerii lor și a testării.

## 1.3 Clasificarea metodelor actuale de testare

În procesul clasic de testare se încearcă a se răspunde la două întrebări foarte simple [Johnson1989]:

- Dispozitivul testat funcționează corect?
- Dispozitivul testat dispune de capacitatea lui de funcționare completă?

În plus, în ultimul timp, în domeniul testării, se vehiculează tot mai des și o cea de-a treia întrebare:

- Dispozitivul testat funcționează la parametrii normali?

Răspunsul primei întrebări este în mod evident necesar pentru a asigura corectitudinea dispozitivului înainte de a fi folosit sau livrat clienților. A doua și a treia întrebare pot fi considerate puțin comfuz. S-ar putea spune că dacă un sistem funcționează corect, acesta dispune și de capacitatea lui de funcționare completă, însă acest lucru nu este valabil în cazul sistemelor redundante. Un sistem redundant poate să funcționeze corect, însă datorită existenței deja în sistem a unui defect se poate să-și fi pierdut capacitatea de a tolera defectele [Johnson1989].

La aceste întrebări critice se poate răspunde prin aplicarea unui număr de teste, într-unul sau mai multe moduri, iar plecând de la aceste întrebări se va prezenta în continuare o taxonomie a metodelor și tehnicilor actuale de testare.

Testarea poate fi realizată prin unul din următoarele două mecanisme: testare internă (integrată, built-in) sau testare externă [Johnson1989]. Tehnicile de testare externe în mod curent necesită scoaterea dispozitivului testat (DUT – Device Under Test) din mediul lui de operare normal pentru a fi supus testelor, folosind un echipament de testare extern lui. Tehnicile de testare interne presupun includerea echipamentului de testare în interiorul dispozitivului, astfel că testarea se poate face fără nici un fel de echipament extern.

Din punctul de vedere al modului de funcționare al DUT, tehnicile de testare interne se împart în: concurente (on-line) și neconcurente (off-line). Tehnicile de testare concurente permit testarea dispozitivului în timpul funcționării normale a acestuia. Tiparele de date ce se execută servesc și ca vectori de test iar defectele sunt puse în evidență prin circuitele de monitorizare interne. Tehnicile neconcurente necesită scoaterea dispozitivului din modul de funcționare normală și trecerea lui în modul de funcționare în vederea testării, testarea făcându-se de către modulele interne DUT dedicate acestui proces.

Independent de mecanismul de testare disponibil, există trei tipuri de teste [Johnson1989]:

- funcționale – Testarea funcțională încearcă să verifice dacă DUT posedă caracteristicile funcționale pe care ar trebui să le aibă. De exemplu testele funcționale pentru un circuit combinațional vor verifica dacă acesta funcționează conform tabelului de adevăr al acestuia; iar pentru un circuit secvențial un test funcțional va trebui să verifice dacă acesta funcționează în concordanță cu diagrama de tranziții a stărilor și cu tabelul de adevăr al ieșirilor. Scopul principal al unui test funcțional este determinarea greșelilor de proiectare;
- structurale – Testarea structurală încearcă să determine dacă există vre-un defect în DUT vizând o anumită clasă de defecte. Testarea structurală presupune modelarea clasei de defecte vizate, generarea setului de vectori de test în funcție de modelul ales și abia apoi aplicarea pe DUT. În plus, multe teste structurale nu se rezumă doar la punerea în evidență a defectelor ci încearcă și localizarea acestora;
- parametrice – Scopul testării parametrice este acela de a verifica dacă un anumit parametru al DUT se încadrează în intervalul de variație specificat.

Fiecare din aceste trei tipuri de testări pot fi efectuate în mod static sau dinamic. În cazul testării statice sunt evaluate caracteristicile dispozitivului aflat în starea stabilă, astfel că acest tip de testare permite detectarea defectelor care modifică comportamentul static al DUT.

•••



Testarea statică presupune aplicarea stimulilor la intrarea dispozitivului după care se așteaptă până se stabilizează ieșirile iar apoi acestea sunt măsurate și evaluate (figura 1.9 a). În esență timpii de răspuns la ieșirile DUT nu sunt evaluați în testarea statică.

Pentru a detecta defectele ce sunt puse în evidență doar de comportamentul dinamic al DUT, vectorii de test trebuie aplicați la intervale de timp mult mai mici decât în cazul testării statice. Aceste intervale de timp sunt de ordinul perioadei de tact din regimul de funcționare normală a DUT. Două dintre variantele testării dinamice sunt slow-fast [Chakravarty1997] și at-speed [Chakravarty1997, Johnson1989]. În testarea slow-fast (figura 1.9 b) vectorii de test se aplică în perechi, permițând într-o primă fază inițializarea DUT și apoi testarea lui. În exemplul din figura 1.9 b, după aplicarea vectorului  $T_1$  și trecerea fenomenelor tranzitorii se aplică cel de-al doilea vector de test  $T_2$  iar măsurarea ieșirilor se face după un timp  $t_0$ , mult mai mic decât timpul de stabilizare,  $t_s$ . Perioada de timp  $t_0$  a tactului de funcționare normală determină rata la care sunt aplicați vectorii de test la intrările DUT în cazul testării at-speed. Ieșirile sunt măsurate chiar înainte de aplicarea următorului vector de test.

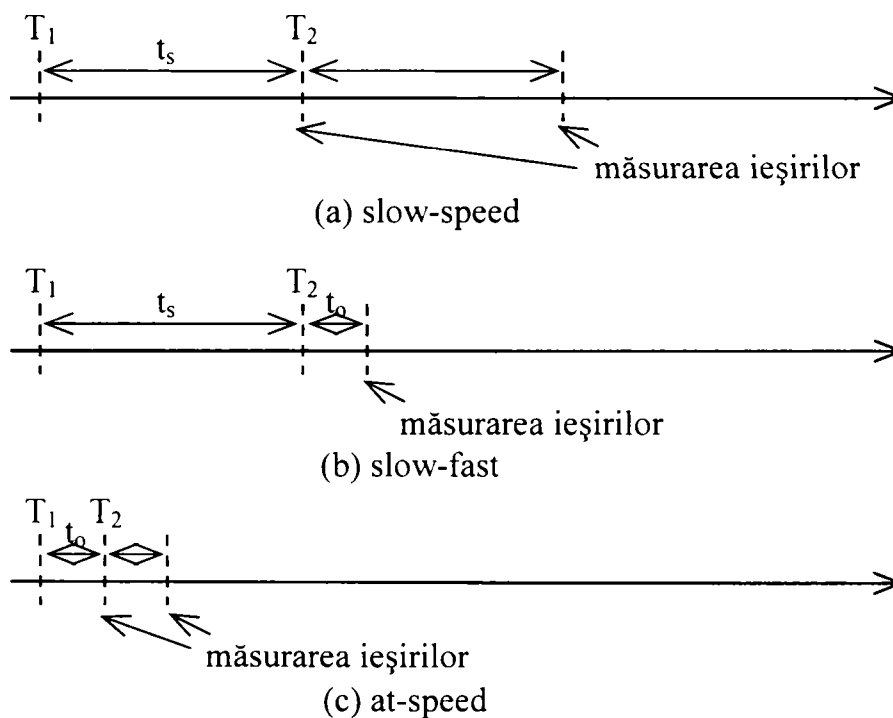


Figura 1.9 – Viteza testării [Chakravarty1997]

Din punctul de vedere al metodei de aplicare a testării toate metodele de testare presupun stimularea DUT folosind un anumit set de vectori de test urmând ca mai apoi să se monitorizeze și analizeze răspunsul DUT (valorile ieșirilor – testare logică, valorile unuia din parametrii continui – testare parametrică) (figura 1.10). Totuși, și din acest punct de vedere metodele de testare se pot clasifica în funcție de importanța alocată sau stimulării intrărilor (generarea vectorilor de test) sau analizei ieșirilor [McCluskey1986].

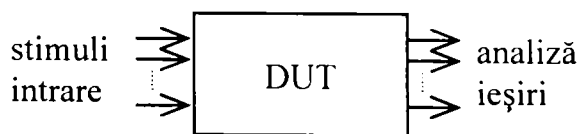
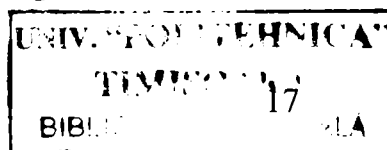


Figura 1.10 – Aplicarea unui test



Generarea vectorilor de test este procesul de determinare a setului minimal de vectori de intrare pentru un DUT, astfel încât să pună în evidență defectele existente (dacă acestea există) [Abramovici1990]. Acest proces poate fi împărțit în două faze: derivarea (obținerea) setului de vectori de test și aplicarea testului. Pentru generarea setului de vectori de test se pleacă pornind de la modelul DUT (la nivel de tranzistor sau la nivel de poartă logică) și de la modelul de defectare și se determină acele combinații de la intrarea DUT care produc răspunsuri diferite ale acestuia pentru cazul în care există defecte față de cazul fără defecte.

Analiza răspunsurilor DUT conține metodele orientate doar pe captura valorilor monitorizate (parametrice sau logice), presupunând că DUT este stimulat sau cu un set de test exhaustiv sau cu unul aleator. Această metodă este utilizată mai ales în cazul mecanismului de testare integrată și al testării on-line, când se compară răspunsul DUT cu o valoare așteptată determinată pentru circuitul fără defecte. Însă este foarte ineficient să se salveze toate răspunsurile posibile ale unui DUT, astfel că o variantă alternativă ar fi metodele de testare bazate pe analiza semnăturilor.

Conceptul de analiză a semnăturilor este foarte simplu. Dacă la intrările DUT se aplică o secvență de vectori de test  $\{T_1, T_2, \dots, T_k\}$  atunci fiecare punct din circuit răspunde cu o anumită secvență de valori. Punctele critice dintr-un DUT pot fi selectate ca puncte de test și pot fi monitorizate. Pe baza valorilor monitorizate se pot pune în evidență defectele din DUT, dacă acestea există. Datele urmărite în punctele de test sunt însă foarte mari, memorarea și procesarea lor necesitănd resurse foarte mari. De aceea se recurge la compresia sau codarea acestor date în anumite entități denumite semnături. Astfel că semnăturile unui circuit trebuie să caracterizeze răspunsul circuitului la vectorii de test aplicați la intrare. Un exemplu de sistem testat pe baza analizei semnăturilor din două puncte de test este prezentat în figura 1.11.

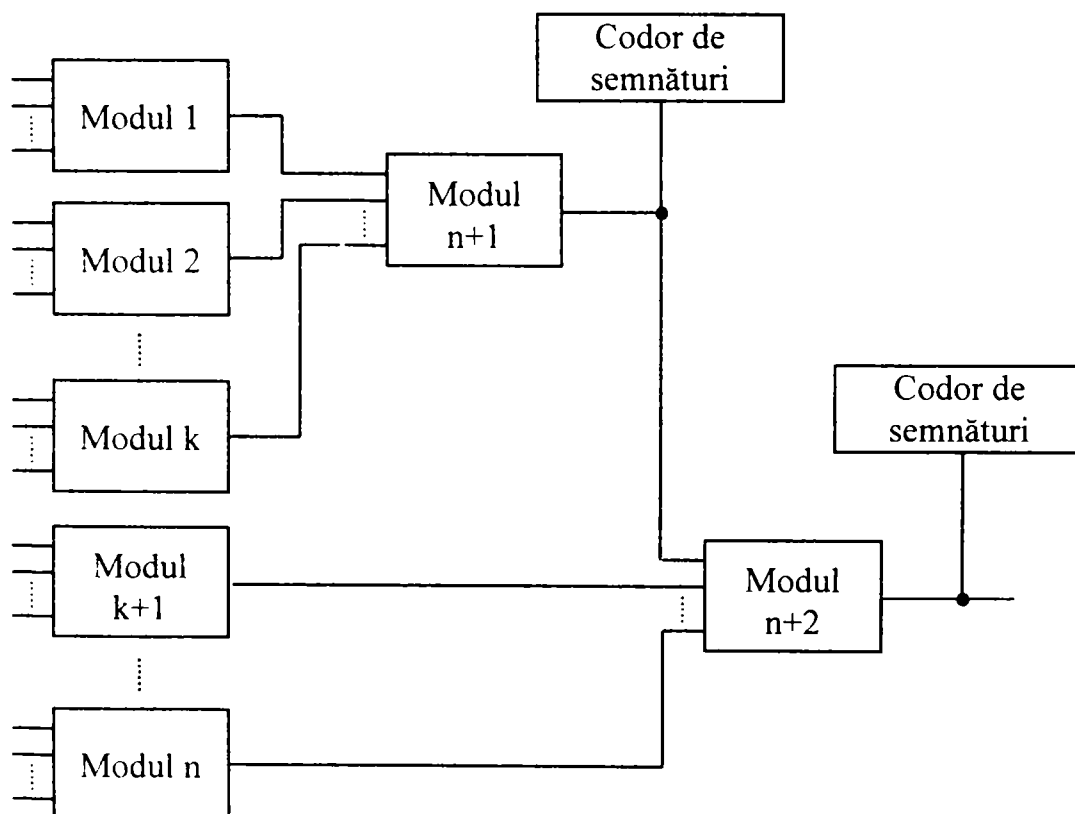


Figura 1.11 – Analiza semnăturilor

Valorile monitorizate în analiza de semnături nu trebuie să fie neapărat valori logice, se mai pot monitoriza curentul (semnătură de curent), temperatura (semnătură termică), etc. Mai târziu în această lucrare se va reveni asupra acestor metode de testare.

Figura 1.12 sintetizează cele prezentate până aici, prezentând o taxonomie a testării. Pentru orice strategie de testare trebuie ales tipul testării, un mecanism al testării, viteza testării și un mecanism de aplicare al testelor. O detaliere a clasificării din figura 1.12 este prezentată în Anexa 1.

De exemplu testarea  $I_{DDQ}$  face parte din categoria testelor parametrice, parametrul urmărit fiind curentul de scurgere (leakage), mecanismul testării  $I_{DDQ}$  poate fi intern sau extern, însă în cele mai multe cazuri se folosește o testare internă folosind senzori integrați de curent. Testarea  $I_{DDQ}$  este o testare statică iar aplicarea testelor presupune un proces de generare al vectorilor de test și monitorizarea curentului absorbit de circuit.

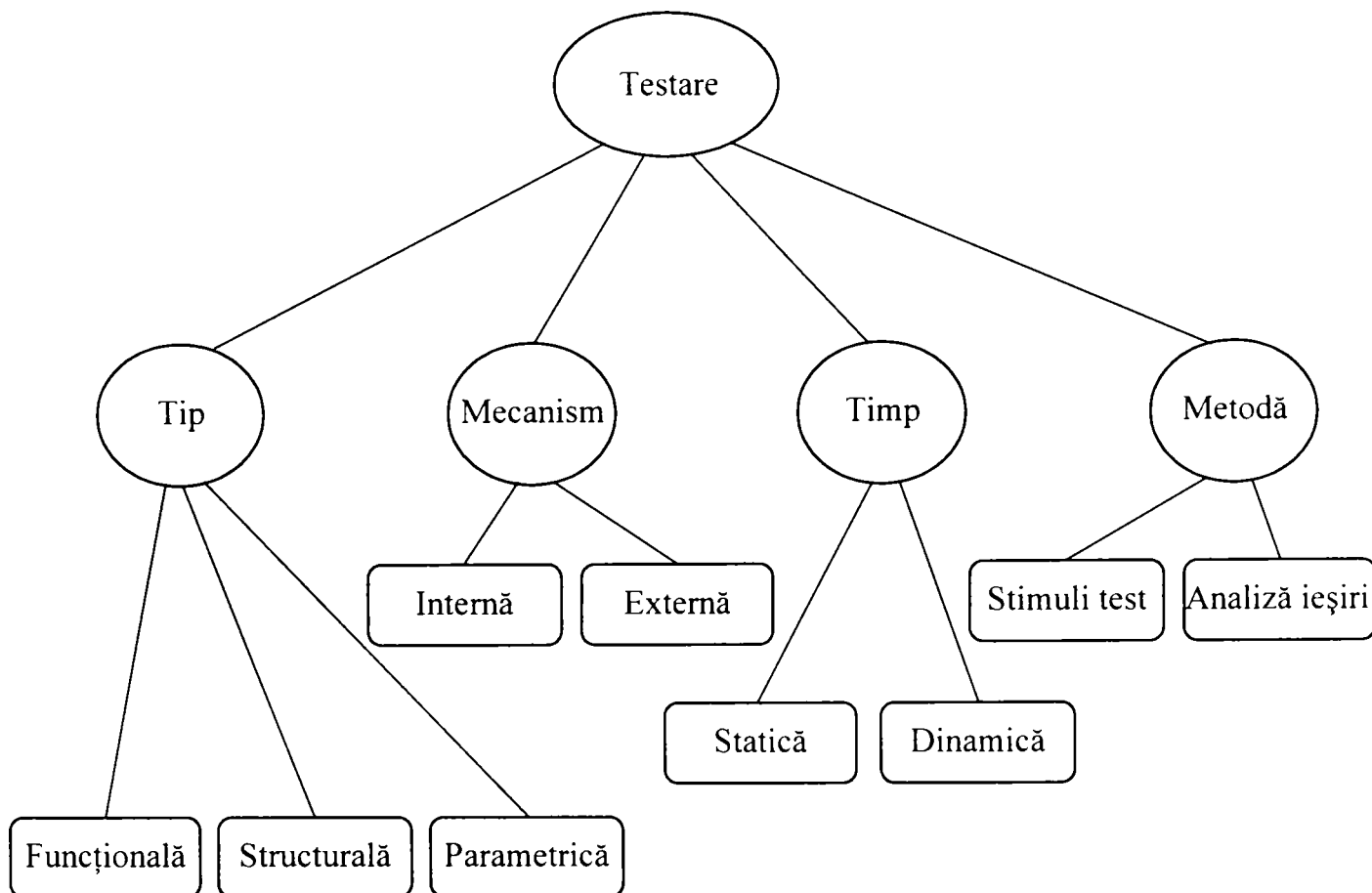


Fig. 1.12 – Clasificarea metodelor de testare

### Testarea logică

Un defect ce cauzează modificarea funcționării logice a DUT poate fi reprezentat și modelat printr-un nivel logic de abstractizare și se numește defect logic. Metoda de testare care vizează detecția defectelor logice se numește testare logică.

### Testarea VLV (Very Low Voltage Testing)

Testarea la o tensiune de alimentare redusă este o tehnică nedestructivă de testare folosită pentru detecția circuitelor slabe (cele care funcțional sunt considerate bune însă se defectează foarte repede în timpul funcționării). În această tehnică de testare, DUT este testat fiind alimentat la o tensiune redusă – aproximativ de 2 sau 2.5 ori tensiunea de prag a tranzistorului. De exemplu pentru o tensiune de prag de 0.7 V un test VLV ar putea folosi 1.7V pentru tensiunea de alimentare. Testarea VLV se aplică la viteză mică.

### **Testarea parametrică**

Un defect ce produce o modificare a unui parametru continuu al dispozitivului testat se numește defect parametric. Metodele de testare care vizează detecția defectelor parametrice se numesc metode de testare parametrică. Unele defecte pot fi modelate printr-un defect logic, altele printr-un defect parametric iar altele prin ambele tipuri de defecte.

În continuare sunt prezentate câteva exemple de testare parametrică:

### **Testarea $\min V_{DD}$**

Această tehnică de testare își propune să determine valoarea minimă a tensiunii de alimentare  $V_{DD}$  pentru care DUT mai funcționează. Valoarea minimă pentru  $V_{DD}$  se poate obține variind valoare tensiunii de alimentare în timp ce se aplică anumiți stimuli la intrări. Circuitele ce au valori anormale pentru  $\min V_{DD}$  sunt considerate defecte.

### **Testarea întârzierii de propagare (Delay Fault Path Testing)**

Un defect de întârziere (delay fault) este un defect care afectează timpul de propagare al trazițiilor logice. O traziție logică trebuie propagată până la ieșirile observabile ale DUT după care se măsoară întârzierea între momentul aplicării vectorului de test și obținerea rezultatului pe ieșiri. O valoare mare a acestei întârzieri detectează un circuit cu defect de întârziere. Testarea pe baza timpului de propagare este o testare parametrică în care vectorii de test se aplică la viteză mare, punându-se în evidență defectele ce afectează comportamentul dinamic al DUT. Defectele de întârziere se activează destul de repede în timpul funcționării și pot provoca erori intermitente.

### **Testarea $I_{DDQ}$**

Tehnicile de testare pe bază de curent aparțin clasei de teste parametrice în care se monitorizează curentul absorbit de DUT. În cazul testării  $I_{DDQ}$  se monitorizează curentul de pe liniile de alimentare în starea statică a circuitului. Această metodă de testare este folosită la testarea circuitelor CMOS deoarece pentru această familie de circuite curentul de scurgere în starea statică este foarte mic. O valoare mare a curentului  $I_{DDQ}$  (de obicei cu câteva ordine de mărime mai mare decât cazul fără defecte) denotă existența unui defect.

### **Testarea termică**

Este cunoscut faptul că în destul de multe cazuri circuitele defecte se încălzesc mai mult decât circuitele funcționale. Un defect care afectează temperatura disipată de un DUT se numește defect termic. De obicei defectele de scurt-circuit produc o încălzire suplimentară a DUT. Testarea termică își propune monitorizarea temperaturii disipate de un circuit considerând ca defecte circuitele care au o temperatură disipată mai mare decât cea normală a circuitelor fără defecte. Măsurarea temperaturii pentru un anumit vector de test trebuie efectuată după stabilirea echilibrului termic astfel că această metodă de testare este statică.

Există două metode de testare termică: testarea în infraroșu (sau termografică) și testarea pe bază de senzori de temperatură. Testarea termică în infraroșu se bazează pe imaginea termică a DUT, imagine ce redă distribuția termică din DUT. Pe baza termogramei se poate lua o decizie cu privire la existența defectelor, localizarea defectelor și chiar informații despre corectitudinea proiectării DUT.

Pentru cea de-a doua metodă de testare termică, DUT sunt prevăzute cu senzori de temperatură integrați. iar testarea se face pe baza valorilor măsurate de acești senzori la aplicarea anumitor stimuli de test.

## 1.4 Temperatura în circuitele integrate VLSI. Tendențe și probleme

Căldura disipată de circuitele integrate VLSI moderne a crescut rapid o dată cu creșterea numărului de tranzistoare integrate și cu creșterea frecvenței. De aceea, în ultimii ani, eforturile producătorilor de circuite integrate și a cercetătorilor din acest domeniu au fost direcționate înspre reducerea puterii consumate și scăderea temperaturii disipate de către circuitele integrate.

Tendențele tehnologice pentru următorii ani vor rămâne aceleași:

- CMOS va rămâne încă tehnologia de fabricație atât pentru circuitele logice cât și pentru memorii;
- dimensiunile ariei de integrare și densitatea de integrare vor continua să crească generând un flux tot mai mare de căldură;
- costul va continua (va trebui) să scadă, chiar dacă apar problemele de răcire și control al puterii consumate;
- reducerea timpului de proiectare.

Dar va fi totuși o redirecționare a eforturilor către:

- găsirea unor noi soluții de răcire mai eficiente și mai ieftine;
- proiectarea se va face în funcție de puterea consumată și temperatură pentru minimizarea acestora;
- creșterea fiabilității circuitelor și găsirea unor noi metode de testare a fiabilității;
- controlul puterii și a temperaturii disipate.

În ceea ce privește studiul și aplicabilitatea temperaturii în sistemele digitale, domeniul a devenit foarte larg, distingându-se însă câteva direcții mai importante și focalizate pe o anumită caracteristică. Principalele direcții de studiu a temperaturii sistemelor electronice sunt următoarele (figura 1.13):

- modelarea disipării căldurii într-un sistem – este necesară pentru estimarea căldurii disipate și are la bază noțiuni din termotehnică și termodinamică. Modelarea este abordată la mai multe niveluri de abstractizare, astfel că există modele pentru disiparea căldurii într-un circuit integrat, placă cu circuite integrate și sistem.
- metode de răcire a componentelor electronice – este foarte importantă în contextul în care puterea disipată crește iar temperatura la nivel de joncțiune trebuie limitată la o anumită valoare.
- metode de testare termică – temperatura este un parametru pe baza căruia se poate estima corectitudinea și funcționalitatea unui dispozitiv electronic. Cunoscând harta termică (distribuția de căldură) a unui dispozitiv se pot detecta și localiza defectele ce apar în respectivul dispozitiv. Testarea termică poate fi utilizată atât în faza de testare din procesul de producție al dispozitivului, cât și în timpul utilizării acestuia.
- metode de control ale temperaturii disipate și a puterii consumate de un dispozitiv electronic – există destul de multe situații în care nu se poate recurge la metodele de răcire cele mai eficiente datorită costului, dimensiunilor sistemului, consumului suplimentar, etc. Câteva exemple ar fi calculatoarele portabile, telefoanele mobile, etc. De exemplu durata uzuală de funcționare pentru un calculator portabil este de 2-4 ore. În cazul dispozitivelor ce necesită un consum redus este foarte utilă introducerea unor mecanisme de control al puterii consumate și a temperaturii disipate, efectuându-se o balansare dinamică și controlabilă a raportului performanță oferită și consum.

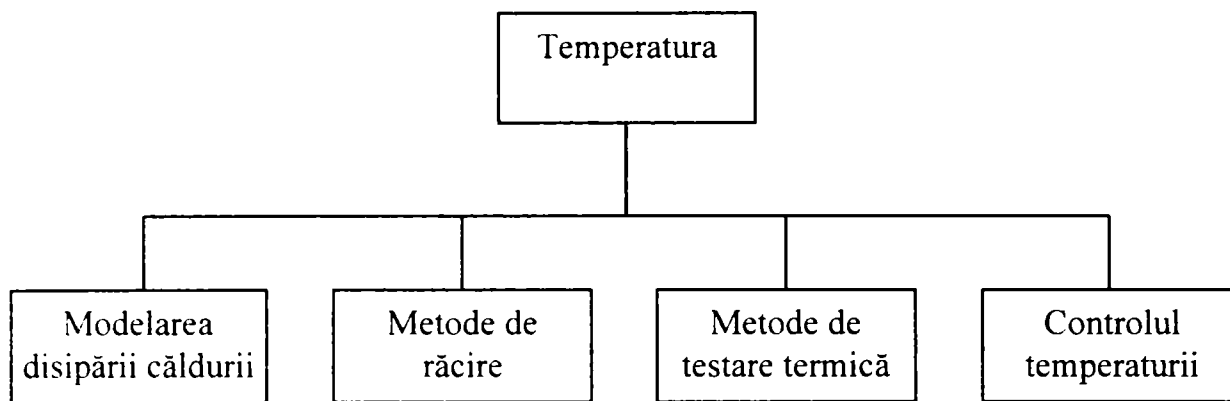


Figura 1.13 – Direcții de cercetare în domeniul temperaturii

## 1.5 Obiectivele tezei

Concluzionând cele spuse în prima parte a acestui capitol, pe măsură ce tehnologia de fabricație a circuitelor integrate evoluează în timp s-a observat că metodele clasice de testare nu mai fac față cerințelor de calitate impuse, astfel că este necesară și foarte importantă utilizarea corectă a unor tehnici de testare corespunzătoare noilor tehnologii precum și înțelegerea și actualizarea lor. Pe de altă parte temperatura a devenit un parametru extrem de important în toate fazele de producție și utilizare precum și la toate nivelurile de componente ale unui sistem de calcul.

Tema de doctorat s-a axat pe studiul fenomenelor termice ce apar la diferitele niveluri de detaliu dintr-un sistem de calcul: circuit integrat, placă echipată cu circuite integrate și sistem. S-a studiat posibilitatea de utilizare a testării termice precum și evaluarea ei comparativ cu celelalte metode de testare existente, în special testarea logică și testarea pe bază de curent. Fenomenele termice ce apar într-un dispozitiv electronic sunt efectul trecerii curentului prin respectivul dispozitiv.

În cazul circuitelor integrate s-au studiat modurile de defectare fizică, efectul acestor defecte asupra temperaturii și posibilitatea de detecție a diferitelor metode de testare (testare logică, testare  $I_{DDQ}$  și testare termică). Efectele defectelor de scurt-circuit dintr-un circuit integrat au fost simulate și studiate într-un mediu de simulare electric (SPICE). Pe baza simulărilor și a testelor efectuate s-au prezentat și câteva rezultate comparative între testarea termică, testarea pe bază de curent și testarea logică.

Studiul fenomenelor termice la nivelul plăcilor echipate cu circuite integrate a fost o a doua direcție de cercetare în cadrul tezei. Aici s-au folosit de imaginile în infraroșu ale unei plăci cu 66 de circuite integrate și de niște programe de simulare termică. Rezultatele obținute sunt concretizate în implementarea a doi algoritmi genetici de deconvoluție a imaginilor termice și o metodă de testare și clasificare a plăcilor cu circuite integrate pe bază de semnătură termică.

A treia direcție de cercetare a tezei a avut în vedere studiul temperaturii într-un sistem de calcul ca ansamblu. Într-un sistem, pe lângă utilizarea temperaturii ca metodă de testare, sunt foarte importante și mecanismele de monitorizare, control și reducere a temperaturii. În cadrul cercetărilor efectuate s-a introdus, prezentat și implementat noțiunea de benchmark termic al unui sistem, s-a implementat o aplicație de monitorizare și control d.p.d.v. termic pentru un sistem și s-a propus și implementat o interfață de programare (API) pentru managementul și reducerea temperaturii la nivelul aplicațiilor utilizator.

Sintetizând cele spuse, în cadrul tezei s-a abordat problema temperaturii așa cum apare ea la diferitele niveluri de detaliu ale unui sistem de calcul iar obiectivele principale cu care a fost lansată această temă de doctorat au fost:

- Să se cerceteze fenomenele termice ce se stabilesc la nivel de circuite integrate și de plăci cu circuite integrate;
- Să se conceapă și să se dezvolte o nouă metodă de testare termică a plăcilor cu circuite integrate bazată pe conceptul de semnătură termică;
- Să se valideze din punct de vedere experimental metoda de testare și să o compare cu celelalte metode de testare aplicate la momentul actual.

În ultimii ani domeniul termic a fost direcționat spre a oferi soluții de control și reducere a temperaturii produse de microprocesoare și sistemele portabile, astfel că tema de cercetare a fost extinsă și asupra acestor noi probleme apărute, lucrarea de față conținând un capitol dedicat reducerii și controlului temperaturii în sistemele de calcul.

## 1.6 Organizarea tezei

Structura acestei teze este mapată pe cele trei direcții de studiu prezentate în subcapitolul precedent.

Capitolul 2 tratează defectele fizice ce pot apare în circuitele integrate CMOS, modelarea lor și modalitățile de detectare a acestora folosind testarea logică, testarea pe bază de curent și testarea termică. În a doua parte a capitolului este prezentată metoda de testare termică împreună cu importanța, avantajele și dezavantajele ei, precum și o comparație între metoda de testare termică și testarea logică și testarea  $I_{DDQ}$ .

În capitolul 3 este prezentată aplicabilitatea metodei de testare termică la plăcile cu circuite integrate. În cadrul capitolului sunt prezentate mecanismele de transfer termic ce apar la nivelul unei plăci cu circuite integrate, interfațarea termică între circuite și placă, metodele de simulare termică a plăcilor cu circuite integrate și problemele ce apar la analiza termică a plăcilor. În continuare sunt prezentate câteva posibile soluții pentru testarea plăcilor cu circuite integrate pe baza de imagini termice în infraroșu.

Capitolul 4 conține o prezentare unitară a mecanismelor de control al puterii și temperaturii disipate de un sistem de calcul, problemele ce apar la aplicarea acestor mecanisme și în final, este prezentată o soluție de reducere a puterii și temperaturii disipate de un sistem prin monitorizarea parametrilor sistemului și auto-adaptarea aplicațiilor la cerințele de consum dorite de utilizator. În a doua parte a capitolului sunt prezentate și rezultatele cercetărilor obținute pe această direcție.

În ultimul capitol sunt prezentate concluziile acestei teze și perspectivele pentru continuarea cercetării.

# 2 Testarea termică a circuitelor integrate

## 2.1 Tipuri de defecte fizice

În timpul procesului de fabricație al unui circuit integrat pot apare mai multe fenomene care au ca și rezultat modificarea circuitului integrat obținut față de cel ideal specificat. Orice astfel de deviație apărută într-un circuit integrat se numește defect fizic de fabricație. Defectele fizice apărute în procesul de fabricație pot cauza o funcționare incorectă a circuitului sau pot modifica cel puțin unul din parametrii măsurabili ai circuitului.

Defectele fizice se pot împărți din punctul de vedere al localizării lor fizice în două mari clase:

- defecte globale – sunt defectele ce afectează o zonă mai întinsă din circuit. Aceste defecte au manifestări imediate și evidente în funcționarea logică a circuitului. Exemple: zgârieturi în pastila de siliciu, alinierea greșită a măștilor, dizlocări ale liniilor de conductor sau erori majore în controlul procesului de fabricație.
- defecte locale – sunt defectele care afectează o zonă mică sau foarte mică din circuit. Aceste defecte trebuie studiate, modelate și generate secvențe de testare pentru detecția lor. Exemple: puncte de extramaterial, material lipsă sau măști impropriu poziționate.

Pe de altă parte, defectele fizice se mai pot clasifica în funcție de faza din procesul de fabricație în care apar:

- defecte apărute în procesul de fabricație al pastilei de siliciu – imperfecțiuni ale materialului, praful sau alte impurități, etc.;
- defecte apărute în faza de împachetare a circuitului;
- defecte apărute în faza de montare pe placă;
- defecte apărute în faza de asamblare a sistemului;
- defecte apărute în faza de utilizare.

Există o multitudine de cauze posibile pentru defectele fizice ce pot apare în circuitele integrate VLSI, în fiecare din fazele prezentate mai sus și ar lua destul de mult timp prezentarea fiecăreia în lucrarea de față. În acest capitol se vor prezenta cauzele de defectare cele mai des întâlnite, modul în care aceste defecte pot fi modelate, influența lor asupra funcționării normale a circuitului și capacitatea de detecție a acestor defecte de către testarea termică.

### 2.1.1 Tehnologia de fabricare a circuitelor integrate VLSI

Procesul de fabricație al circuitelor integrate VLSI (Very Large Scale Integration) este unul dintre cele mai incredibile dintre realizările științei și tehnologiei actuale, datorită dimensiunilor foarte mici cu care se lucrează și datorită numărului foarte mare de unități integrate.

În procesul de fabricație se face apel la operații din mai multe domenii: procese de spălare chimică, difuzie, fotolitografie, gravare uscată, metale și implantare. Toate aceste operații tehnologice au diferite finalități în procesul de fabricație al circuitului. Un circuit integrat este realizat din mai multe straturi semiconductoare suprapuse, fiecare trecere prin aceste operații tehnologice diferite creând un nou strat. Toate aceste operații tehnologice sunt



și posibile surse de defecte, astfel că în fiecare din fazele procesului tehnologic de fabricație se pot introduce defecte în circuitul integrat rezultat.

Faza preliminară procesului de fabricație este obținerea pastilei de siliciu monocristalin cu gradul de puritate dorit ("Wafer"). Actualmente acest proces de obținere a pastilei de siliciu este bine pus la punct, ieftin și nu prezintă complicații deosebite. Majoritatea companiilor folosesc pastile de siliciu (wafer) cu diametru de 200 mm. Cu cât pastila de siliciu este mai mare, cu atât încap mai multe circuite (die) pe o pastilă (Figura 2.1). Pe aceste circuite (die) sunt realizate, prin anumite operații, mai multe tranzistoare. Cu cât dimensiunile tranzistoarelor sunt mai mici, cu atât densitatea de integrare este mai mare (se pot integra mai multe tranzistoare într-un chip).

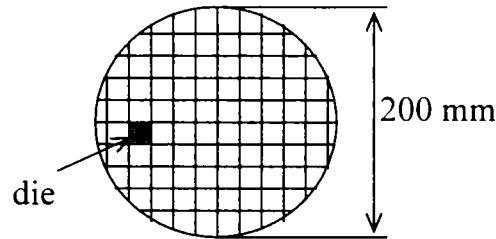


Figura 2.1

La ora actuală procesoarele sunt realizate în tehnologia de 0.25 microni, 0.18 microni și 0.13 microni, tinzându-se spre dimensiuni cât mai mici. Din punct de vedere logic un tranzistor MOS poate fi văzut ca în figura 2.2, iar în mod real cel mai simplu tranzistor MOS este prezentat în figura 2.3.

### Tranzistor

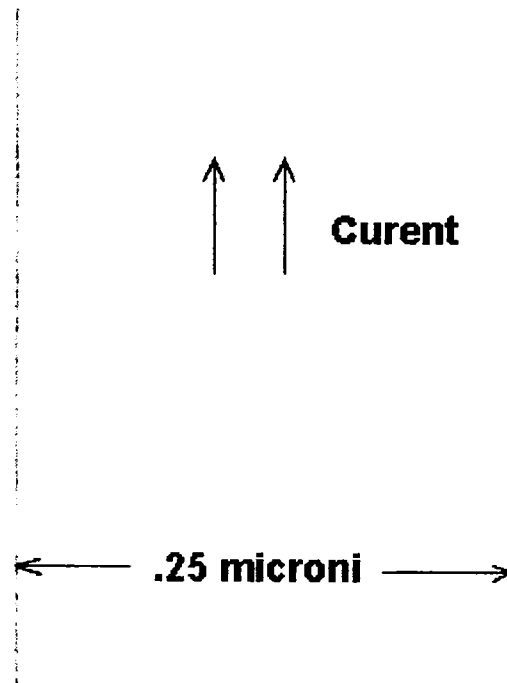


Figura 2.2 – O vedere abstractizată a unui tranzistor MOS

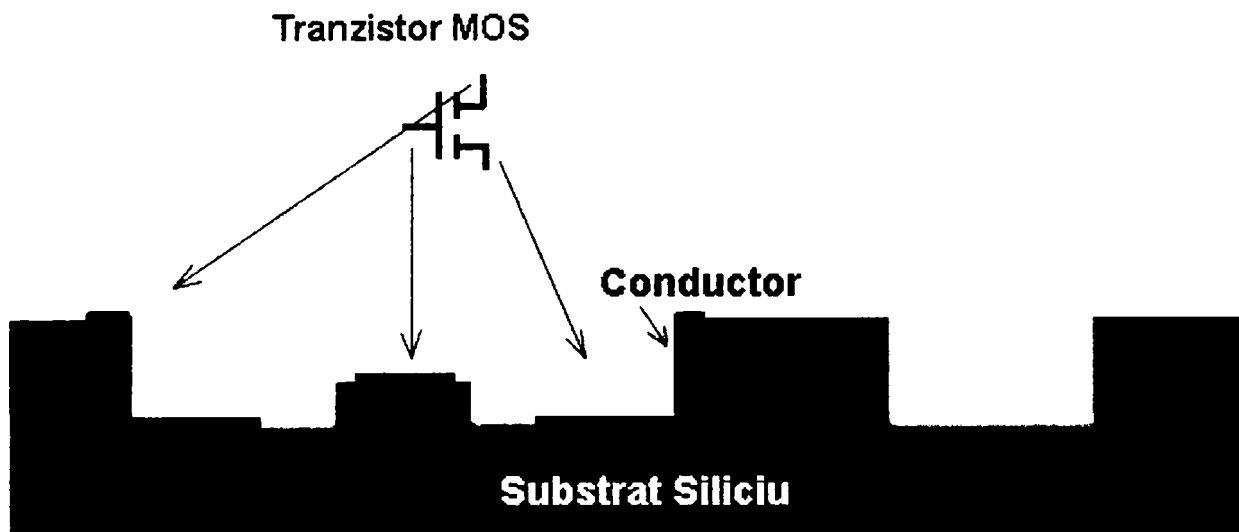


Figura 2.3 – Implementarea unui tranzistor MOS

### 2.1.2 Defecte în straturile de metal

Traseele de material conductor, numite și metalizare, sunt straturile subțiri de metal (de obicei aluminiu) ce sunt plasate într-un circuit integrat pentru conectarea diferitelor puncte ale circuitului [Abramovici1990]. O dată cu creșterea complexității și densității de integrare a circuitelor, a fost necesară introducerea mai multor straturi de material conductor pentru a se putea implementa numărul mare de conexiuni. Straturile de metal sunt separate printr-un material izolator, iar contactele dintre straturi se realizează prin îndepărtarea materialului izolator.

Cauzele cele mai des întâlnite care stau la baza defectelor apărute în straturile de metalizare sunt [Abramovici1990, Rajsuman2000, Sony2001, Henderson2001]:

- două sau mai multe linii de metal pot fi scurt-circuitate datorită impurităților sau porțiunilor de extra-material;



Figura 2.4 – Defect de tip scurt-circuit între două linii de interconectare

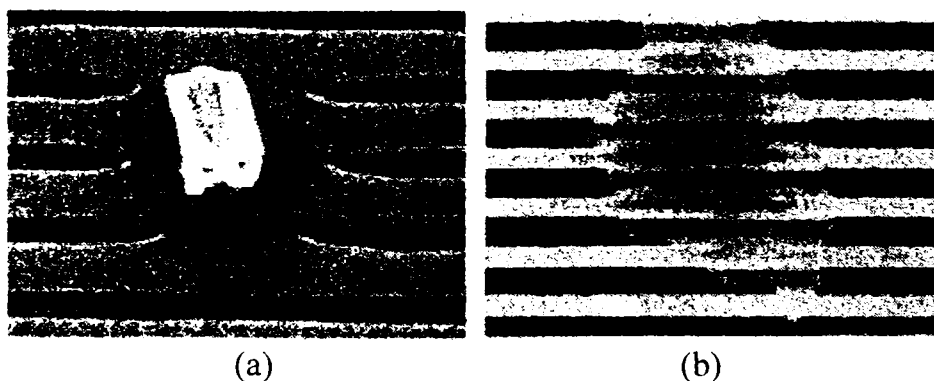


Figura 2.5 – (a) Defect de scurt-circuit între patru linii de metalizare datorită unei impurități; (b) Scurt-circuit datorat unei slabe expuneri a stratului de fotorezist [Rajsuman2000]

- întreruperi ale liniilor de metal datorate zgârieturilor, imperfecțiunilor în material sau impurităților:



Figura 2.6 – Defect de tip întrerupere

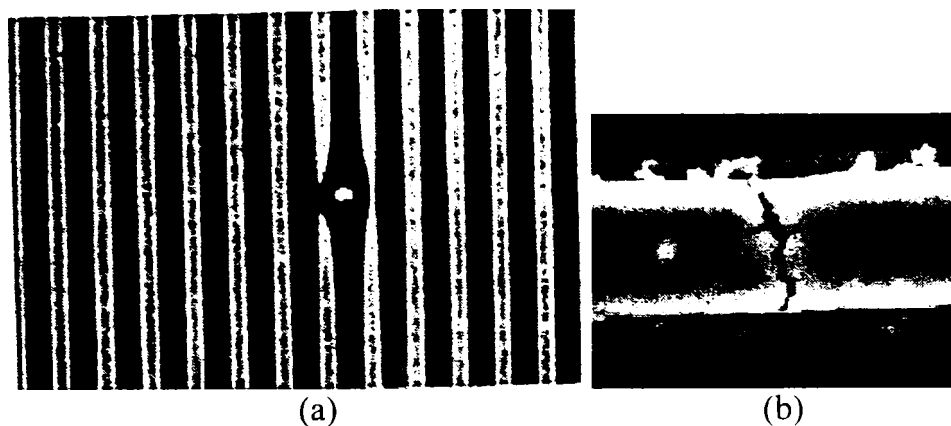


Figura 2.7 – (a) Defect de întrerupere și subțierea unei linii datorate unei impurități [Rajsuman2000]; (b) Defect de întrerupere datorat unei imperfecțiuni în material [Henderson2001]

- potrivirea greșită a liniilor de metal peste trecerile din stratul izolator.

Cele mai importante moduri de defectare în funcționare a structurii de metalizare sunt:

- migrația electronilor – constă în deplasarea masei de metal ce formează liniile conductoare la trecerea unor curenți mari;
- corodarea metalului – poate produce diminuarea conductivității liniilor de conectare sau chiar întreruperea acestora.

### 2.1.3 Defecte în substratul de siliciu

Difuzia este procesul prin care se implantează impurități în anumite regiuni ale substratului pentru a forma drena și sursa tranzistoarelor MOS. Procesele de difuzie și implantare ionică introduc și ele, la rândul lor o serie de defecte, dintre care cele mai importante și mai răspândite sunt:

- doparea improprie (insuficientă sau prea accentuată) a anumitor zone – sunt afectate negativ caracteristicile dispozitivului;
- implantarea impurităților în zone greșite în substrat.

### 2.1.4 Defecte în stratul de oxid

Depunerea straturilor de oxid este folosită în mai multe faze ale procesului de fabricație și are ca scop protejarea zonelor care nu sunt dopate în procesul de difuzie și izolarea straturilor de metal.

Principalele cauze ale defectelor apărute în stratul de oxid sunt:

- impuritățile din material pot afecta capacitatea de izolare a acestuia sau caracteristicile electrice ale dispozitivului;



Figura 2.8 – Defect de tip scurt-circuit între două straturi de metalizare

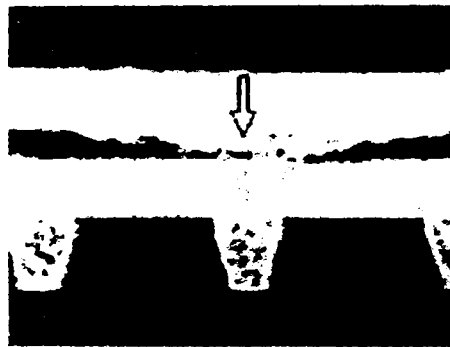


Figura 2.9 – Defect de scurt-circuit între două straturi de metalizare [Rajsuman2000]

- grosimea stratului de oxid poate afecta capacitatea de izolator a acestuia;
- variațiile în grosimea stratului de oxid pot cauza întreruperea liniilor de metal plasate peste stratul de oxid;
- trecerile în stratul de oxid pot fi poziționate greșit.

## 2.2 Modelarea defectelor fizice

Majoritatea defectelor fizice prezentate pot fi modelate la nivelul electric de abstractizare prin:

- defect de scurt-circuit (rezistență nulă);
- defect de întrerupere (rezistență infinită);
- defecte de scurt-circuit rezistiv (rezistență diferită de zero);
- defect de tranzistor blocat (stuck-open) sau saturat (stuck-on);
- modificarea nivelului de prag pentru tranzistor.

La nivel logic defectele pot fi modelate în mai multe moduri, cele mai răspândite fiind: defectele de blocare pe 0 logic sau pe 1 logic, tranzistor blocat sau tranzistor saturat, etc.

Exemplu:

- un anumit defect fizic – extramaterial între două linii de metal:

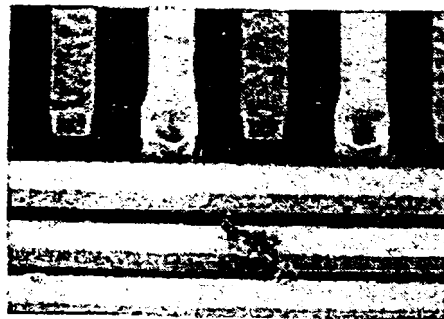


Figura 2.10 – Scurt-circuit între două linii datorită unui defect în stratul de metalizare [Rajsuman2000]

- la nivel electric este considerat un defect de scurt-circuit între liniile 1 și 2:

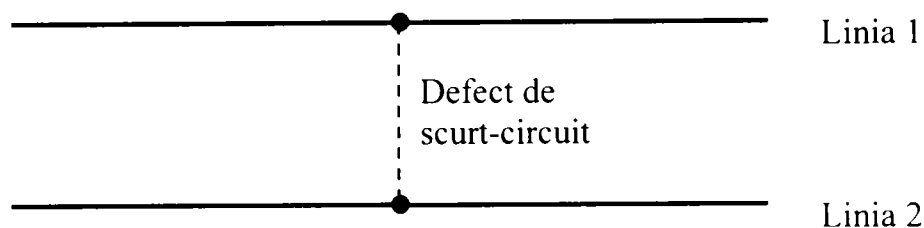


Figura 2.11 – Modelul electric al defectului

- modelarea defectului la nivel logic depinde de poziția acestuia relativă la porțile ce au contact cu liniile 1 și 2:

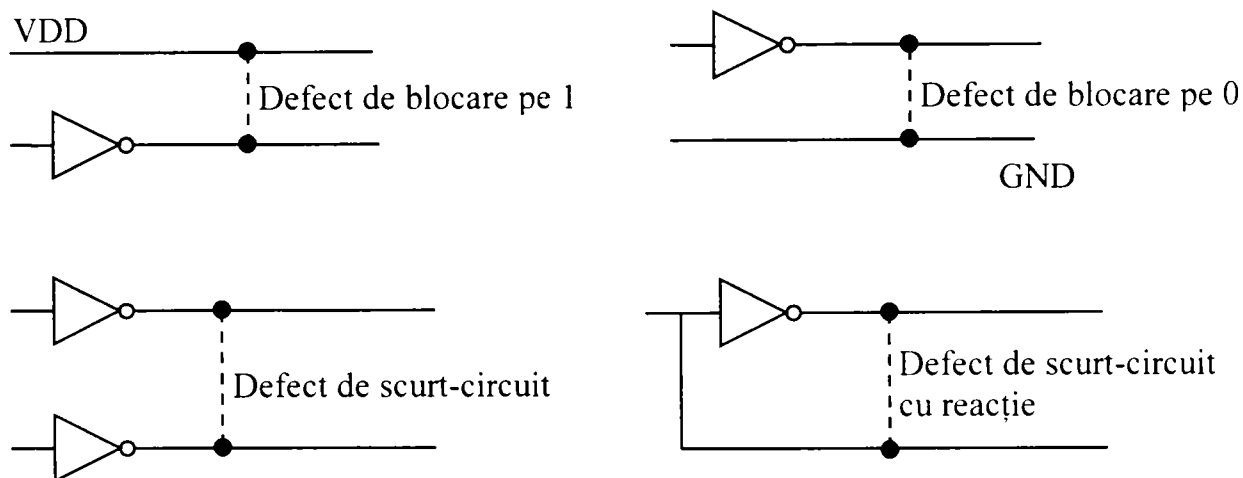


Figura 2.12 – Modele logice al defectului

În continuare sunt prezentate în detaliu defectele de scurt circuit și modificările aduse comportamentului termic al circuitelor. În același timp sunt prezentate și posibilitățile de detecție ale fiecărui tip de defect de testarea logică, testarea  $I_{DDQ}$ , testarea termică și testarea întârzierii.

### 2.2.1 Defecte de scurt-circuit în circuitele combinaționale

Un defect de scurt-circuit poate apare datorită existenței unui material extra-conductor sau datorită lipsei izolației între două linii conductoare rezultând conectarea acestora. Un defect de scurt-circuit va cauza un comportament eronat doar dacă cele două noduri cuplate prin defect au valori logice diferite în același timp. Când cele două noduri au valori logice diferite atunci defectul poate fi detectat prin:

- testarea logică – pentru ca defectele de scurt-circuit să fie detectate prin testarea logică, nu e suficient ca nodurile să aibă valori logice diferite, ci trebuie ca unul din ele să domine valoarea logică a celuilalt și să existe o cale de la nodul dominant la o ieșire primară, astfel încât eroarea să poată fi observată și astfel detectată;
- testarea pe baza întârzierii – chiar dacă valoarea logică a unui nod nu domină valoarea logică a celuilalt nod, defectul poate însă încetini tranziția de la o valoare logică la alta, a celui de-al doilea nod;

- testarea  $I_{DDQ}$ : când cele două noduri conectate prin defectul de scurt-circuit sunt conduse spre valori logice diferite, există o cale de legătură electrică între  $V_{DD}$  și  $V_{SS}$  conducând la creșterea curentului  $I_{DDQ}$ ;
- testarea termică: când cele două noduri conectate prin defectul de scurt-circuit sunt conduse spre valori logice diferite, există o cale de legătură electrică între  $V_{DD}$  și  $V_{SS}$  conducând la creșterea puterii consumate și a temperaturii disipate.

Pentru tehnologiile ECL și TTL defectele de scurt-circuit se comportă ca un SAU-cablat sau ca un SI-cablat. Comporatmentul logic al unei porți CMOS este mai complicat și este afectat de dimensiunea tranzistorului, topologie (tranzistoarele în paralel sunt mai puternice decât cele în serie), variațiile procesului de fabricație, etc. Oricum, funcția logică a defectului de scurt-circuit poate să nu îndeplinească cerințele de detectare a erorii logice cu un singur test, chiar dacă circuitul nu este redundant. În acest caz, detectarea acestui defect necesită un test parametric. Testarea  $I_{DDQ}$  și testarea termică sunt modurile de testare cu cea mai mare rată de detecție a defectelor de scurt-circuit din circuitele integrate CMOS.

Chiar dacă un scurt-circuit nu cauzează schimbări în comporatmentul logic, este probabil să cauzeze erori de întârziere. Tranziția de la un nivel logic la altul a unui anumit nod va fi întârziată de existența defectului. Această întârziere se va propaga până la ieșirile observabile ale circuitului. Detectarea erorilor de întârziere este, însă, considerabil mai dificilă decât detecția erorilor logice și a celor bazate pe curent și temperatură.

O strategie alternativă de testare se bazează pe disiparea suplimentară de putere datorită defectului, atunci când cele două noduri ce compun defectul de scurt-circuit sunt aduse la valori logice diferite. În continuare se va prezenta capacitatea de detecție a defectelor de scurt-circuit de către testarea termică.

Nodurile care sunt intrări sau ieșiri pentru o poartă logică sunt considerate noduri externe (de exemplu a, b, c, z în poarta  $P_1$  din figura 2.13). Celelalte noduri dintr-o poartă logică sunt considerate noduri interne (de exemplu x și y din  $P_1$ , respectiv  $P_2$ ) [Chakravarty1997].

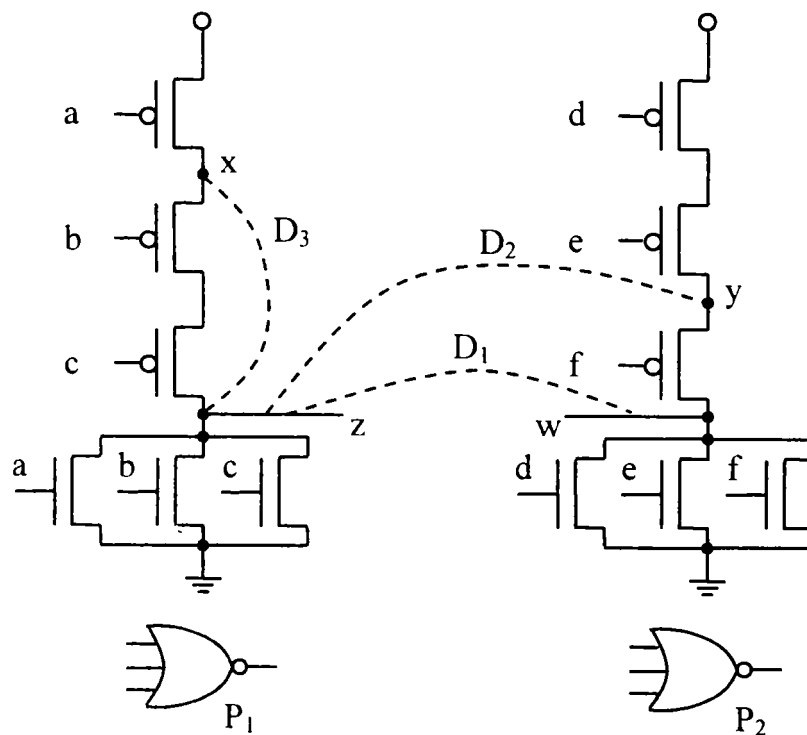


Figura 2.13 – Defecte de scurt-circuit la nivel logic

**Definiția 2.1:** Defectele de scurt-circuit între nodurile a două porți logice diferite se numesc defecte de scurt-circuit inter-poți ( $D_1$  și  $D_2$ ).

**Definiția 2.2:** Defectele de scurt-circuit inter-poți care implică doar nodurile externe se numesc scurt-circuite externe ( $D_1$ ).

**Definiția 2.3:** Defectele de scurt-circuit între nodurile unei singure porți se numesc scurt-circuite interne sau intra-poartă ( $D_3$ ).

În funcție de localizarea nodurilor conectate printr-un defect de scurt-circuit există următoarele tipuri de defecte:

- scurt-circuite de tip grilă-oxid (gate-oxide shorts);
- scurt-circuite inter-poți (inter-gate shorts);
  - scurt-circuite externe (external shorts);
  - scurt-circuite interne (intra-gate shorts).
    - scurt-circuite între terminalele unui tranzistor MOS
- scurt-circuite extrase (cele posibil să apară într-un anumit design).

### 2.2.1.1 Scurt-circuite grilă-oxid

Scurt-circuitele de tip grilă-oxid sunt modul dominant de defectare al circuitelor integrate CMOS [Chakravarty1997, Rajsuman2000]. Structura abstractizată a unui tranzistor MOS este cea prezentată în figura 2.14 (a). Într-un astfel de tranzistor, ce nu conține defecte, nu există nici o cale de curent între grilă și celelalte terminale ale tranzistorului, datorită stratului de oxid de sub grilă ce are rol de izolator. Defectele de tip scurt-circuit grilă-oxid pot apărea în procesul de fabricație datorită imperfecțiunilor în substratul de siliciu sau în oxid, sau în exploatare în special ca rezultat la stresul electric (figura 2.14 b).

Modelul pentru un tranzistor MOS cu un defect de scurt-circuit grilă-oxid e reprezentat în figura 2.14 (c) ca o pereche de două tranzistoare parazite și rezistența defectului  $R_d$ . Caracteristicile celor două tranzistoare obținute datorită defectului depind de localizarea defectului reprezentată prin constanta  $k$ , definit ca raportul dintre distanța defectului față de drenă și lungimea totală a canalului.

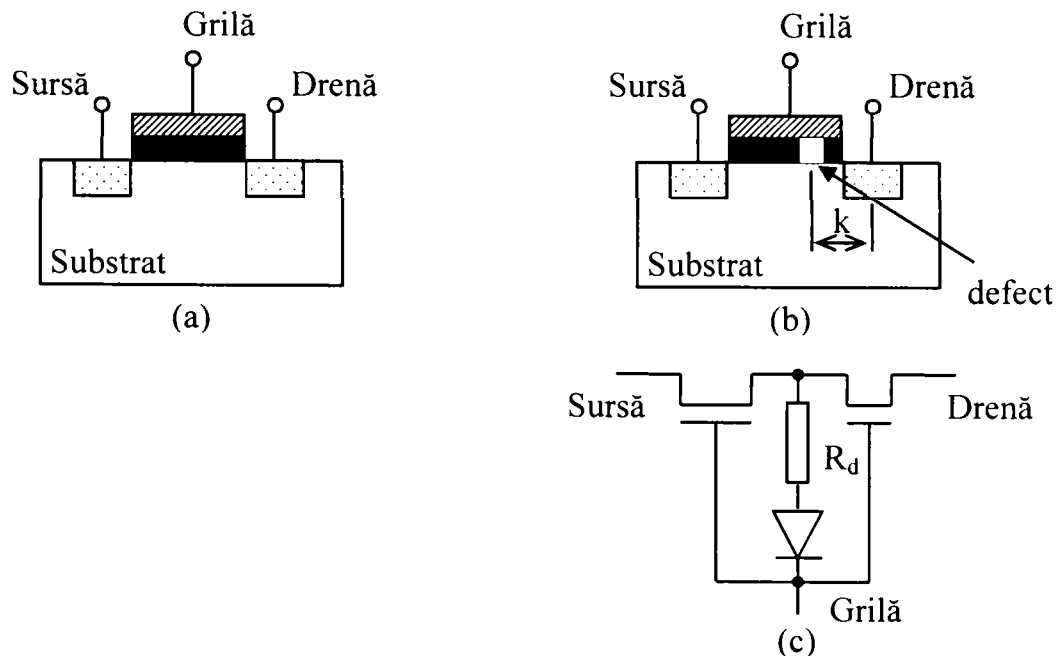


Figura 2.14 – Defect de scurt-circuit grilă-oxid

Pentru a înțelege comportamentul circuitelor CMOS în prezența defectelor de scurt-circuit grilă-oxid și capacitatea de detecție a testării termice, voi folosi schema cu cele trei inversoare din figura 2.15, în care tranzistorul nMOS din inversorul  $P_2$  conține defectul.

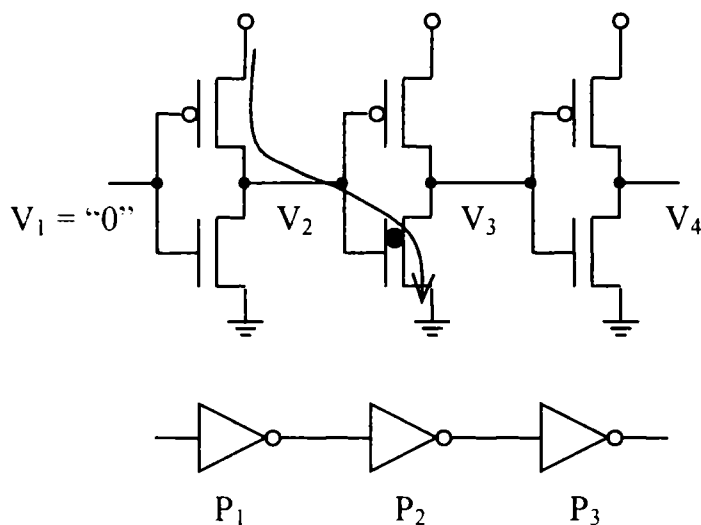


Figura 2.15 – Trei inversoare inseriate

Pentru a detecta acest defect trebuie ca la intrarea inversorului ce conține defectul să se aplice “1”, deci  $V_1 = “0”$ . În acest caz defectul introduce o cale de curent de la  $V_{DD}$  la masă prin tranzistorul pMOS al porții  $P_1$  și defectul din tranzistorul nMOS al porții  $P_2$ . Acest curent  $I_{DDQ}$  suplimentar este de câteva ordine de mărime mai mare decât curentul circuitului fără defect, permițând astfel detecția defectului printr-un test  $I_{DDQ}$ . În același mod defectul prezent în poarta  $P_2$  duce și la creșterea puterii disipate de circuit, astfel că defectul se poate evidenția și prin testarea termică.

S-a simulat în SPICE funcționarea circuitului din figura 2.15 atât pentru cazul fără defect, cât și pentru diferite caracteristici ( $k$  și  $R_d$ ) ale defectului de scurt-circuit grilă-oxid. Implementarea circuitului s-a realizat folosind tehnologia 0.18 microni:

```
*TRANSISTORS NETWORK
*Mname DRAIN GATE SOURCE SUBSTRATE MODEL WIDTH LENGTH
*      NODE  NODE  NODE  NODE      NAME  MICRONS MICRONS
*-----
MP1    2     1     5     5      PMOS18  W=1.4u  L=0.2U
MN1    2     1     0     0      NMOS18  W=0.7U  L=1.2U
C1     1     0     .1pF
MP2    3     2     5     5      PMOS18  W=1.4U  L=0.2U
MN2    3     2     0     0      NMOS18  W=0.7U  L=0.2U
C2     2     0     .1pF
MP3    4     3     5     5      PMOS18  W=1.4U  L=0.2U
MN3    4     3     0     0      NMOS18  W=0.7U  L=0.2U
C3     3     0     .1pF
```

Puterea disipată de circuitul fără defecte la aplicarea unui semnal rectangular ( $v(1)$  din Figura 2.17) la intrarea  $V_1$  este prezentată în figura 2.16. Se poate observa că există un consum de curent doar în timpul tranzițiilor circuitului, iar în perioada de funcționare statică, circuitul fără defecte disipă o cantitate infimă de putere. Puterea statică disipată de circuit după terminarea fenomenelor tranzitorii este foarte mică, de câteva zeci de pA (Tabelul 2.1).



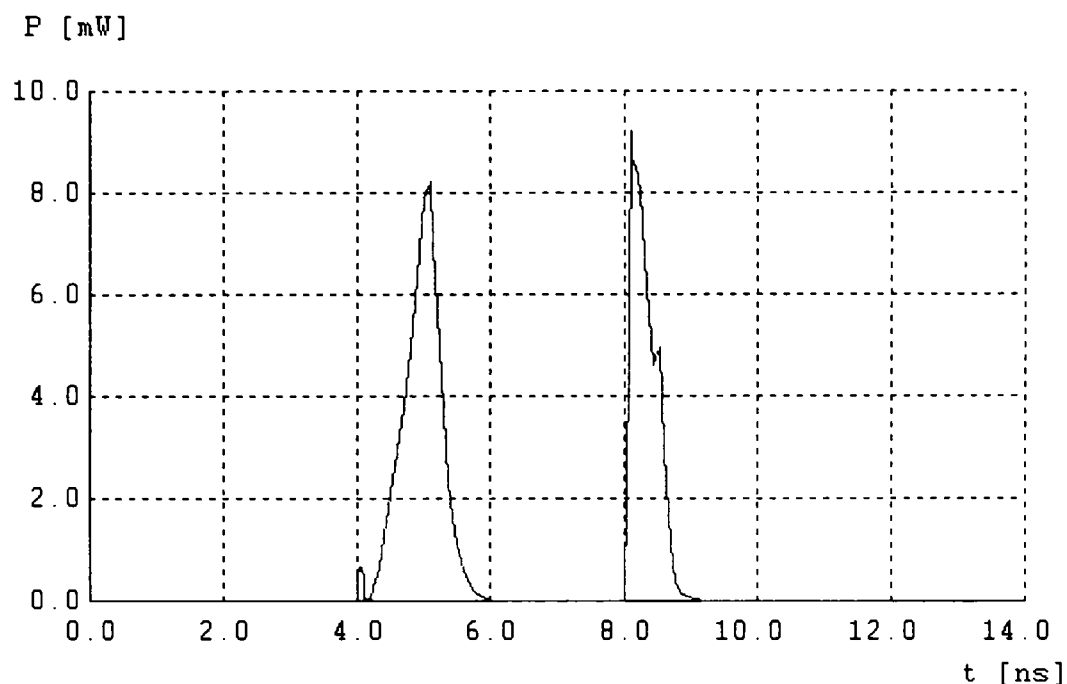


Figura 2.16 – Puterea disipată de circuitul fără defecte

	$V_1 = 0 \text{ V}$	$V_1 = 5 \text{ V}$
$I_{DDQ}$ [pA]	96	101
$P_s$ [pW]	480	505

Tabelul 2.1 – Curentul  $I_{DDQ}$  și puterea disipată de circuitul fără defecte

În continuare s-au făcut mai multe simulări pentru diferite caracteristici ale defectului de scurt-circuit grilă-oxid din poarta  $P_2$ . Tranzistorul NMOS al porții  $P_2$  ( $N_2$ ) a fost înlocuit cu modelul pentru defectul de scurt-circuit grilă-oxid prezentat în figura 2.14 (c), având descrierea pentru SPICE prezentată în continuare:

```
* GATE-OXIDE SHORT (k = 0.65, Rd = 1K)
MN2S  6    2    0    0    NMOS18  W=0.7U  L=0.07U
MN2D  3    2    6    0    NMOS18  W=0.7U  L=0.13U
Rd    6    7    1k
Dd    2    7    DMOD
```

Defectul din poarta  $P_2$  este pus în evidență de testarea termică dacă la intrarea  $V_1$  a circuitului se aplică "0". Puterea statică disipată de circuitul defect are valori de câteva ordine de mărime mai mare decât cea disipată de circuitul fără defecte. Această valoare mare a puterii disipate apare datorită defectului, deoarece se stabilește o cale de curent între  $V_{DD}$  și masă prin tranzistorul PMOS al porții  $P_1$  și defectul de scurt-circuit din tranzistorul NMOS al porții  $P_2$  (Figura 2.15).

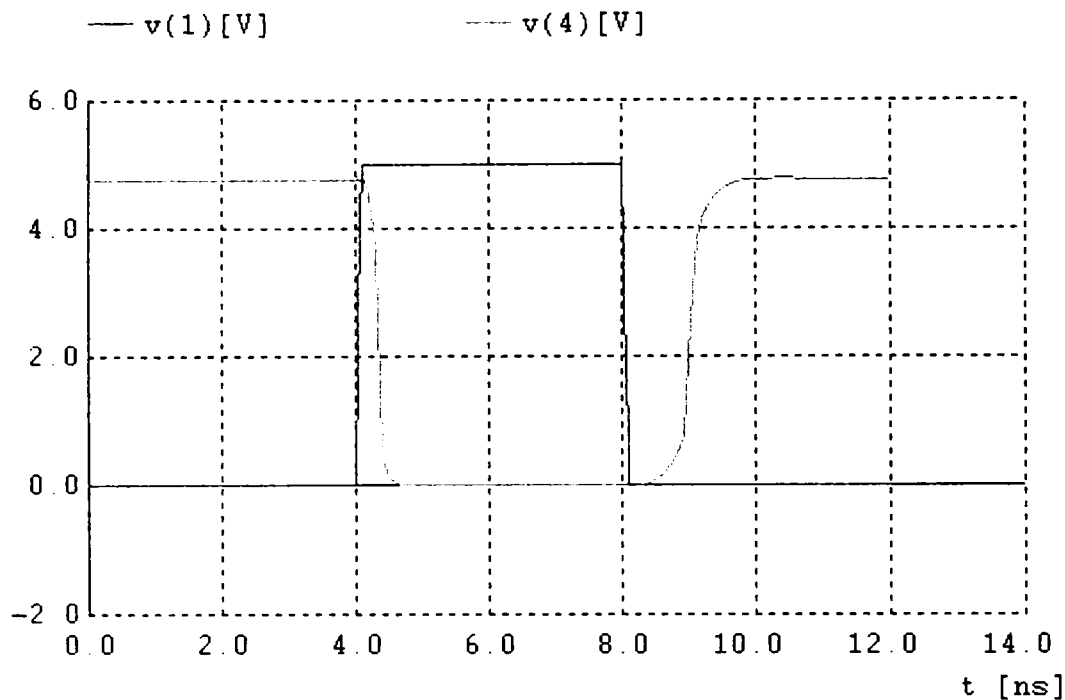


Figura 2.17 – Comportamentul logic al circuitului defect

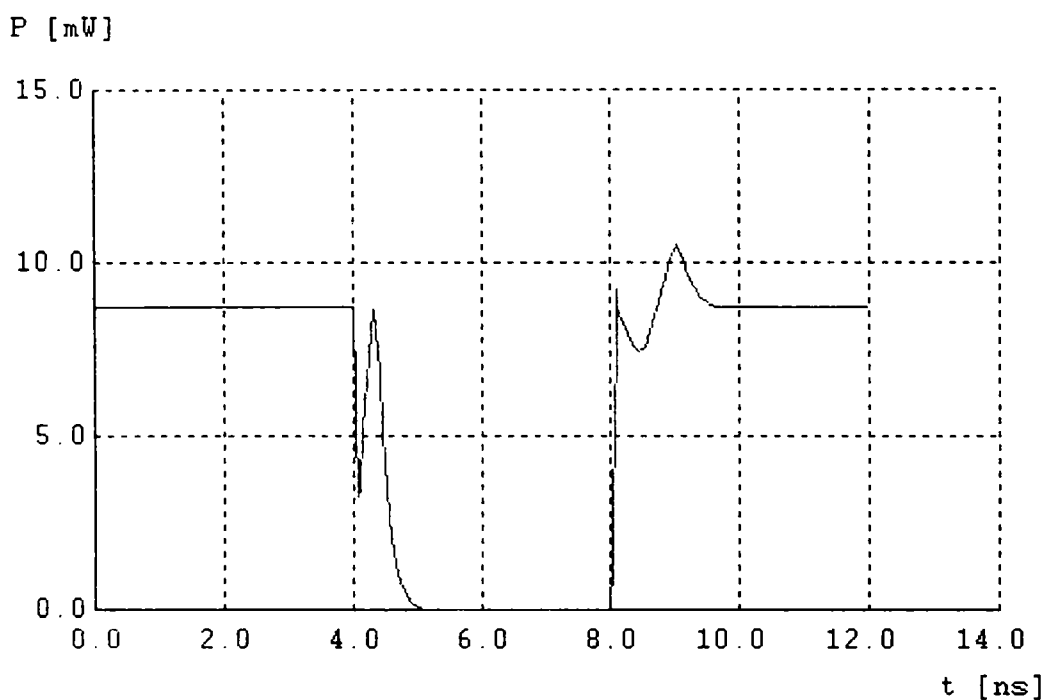


Figura 2.18 – Puterea disipată de circuitul defect

Din simulări s-a determinat că puterea disipată crește cu creșterea lui  $k$  și scade cu creșterea rezistenței defectului,  $R_d$  (Tabelul 2.2, Figura 2.19 și Tabelul 2.3). Chiar dacă, pentru valori mici ale rezistenței defectului  $R_d$  ( $< 500$  ohmi) acest tip de defect devine defect de tip blocare la “0” a intrării  $P_2$ , astfel că poate fi detectat prin testarea logică, majoritatea acestor defecte au valori mari pentru  $R_d$  și nu pot fi detectate prin testarea logică. Pentru valori ale  $R_d$  între 500 și 1k ohmi defectul poate fi detectat prin testarea pe baza întârzierii de propagare, ca defect de întârziere la tranziția “1”-“0” la ieșirea porții  $P_2$ . Majoritatea defectelor de scurt-circuit grilă-oxid pot fi însă detectate cu ajutorul testării  $I_{DDQ}$  și a testării termice.

$R_d$ [k $\Omega$ ]	0.1	0.5	1	1.5	2	5	10	50	100	1000
$I_{DDQ}$ [mA]	1.40	1.54	2.02	1.69	1.44	0.72	0.38	0.085	0.044	0.004
$P_s$ [mW]	7.00	7.70	10.11	8.45	7.20	3.60	1.90	0.427	0.217	0.022

Tabelul 2.2 – Curentul  $I_{DDQ}$  și puterea disipată pentru diferite valori ale  $R_d$ ,  $k = 0.65$

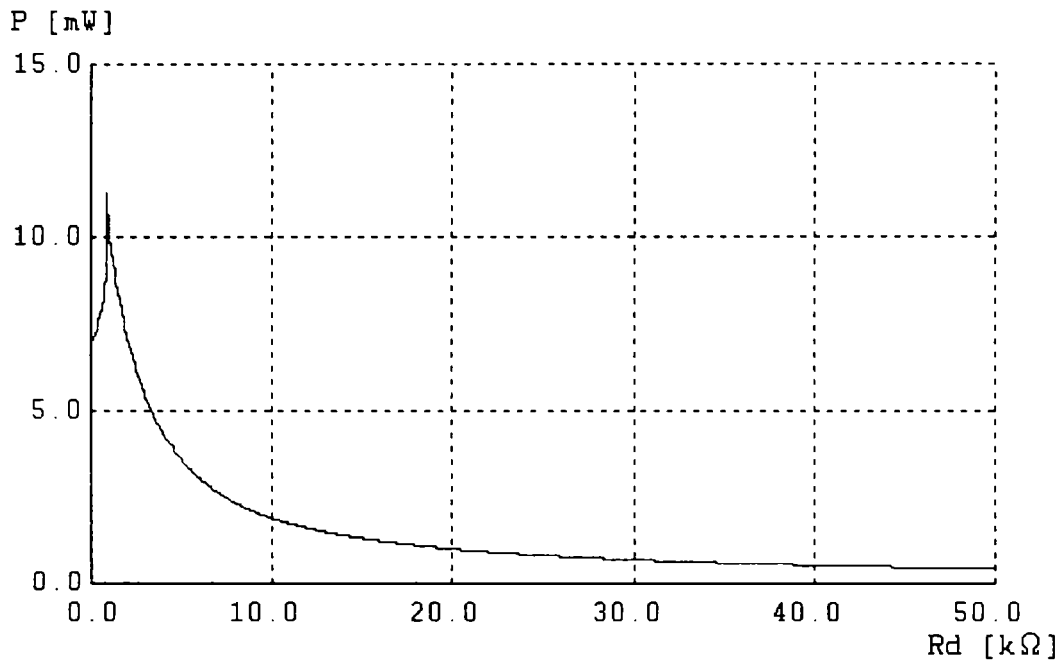


Figura 2.19 – Puterea disipată funcție de  $R_d$ ,  $k = 0.65$

$k$	0	0.2	0.3	0.4	0.5	0.6	0.7	0.8	1
$I_{DDQ}$ [mA]	1.538	1.656	1.685	1.740	1.813	1.926	2.260	2.463	2.050
$P_s$ [mW]	7.690	8.278	8.428	8.700	9.067	9.632	11.300	12.319	10.300

Tabelul 2.3 – Curentul  $I_{DDQ}$  și puterea disipată pentru diferite valori ale  $k$ ,  $R_d = 1k$

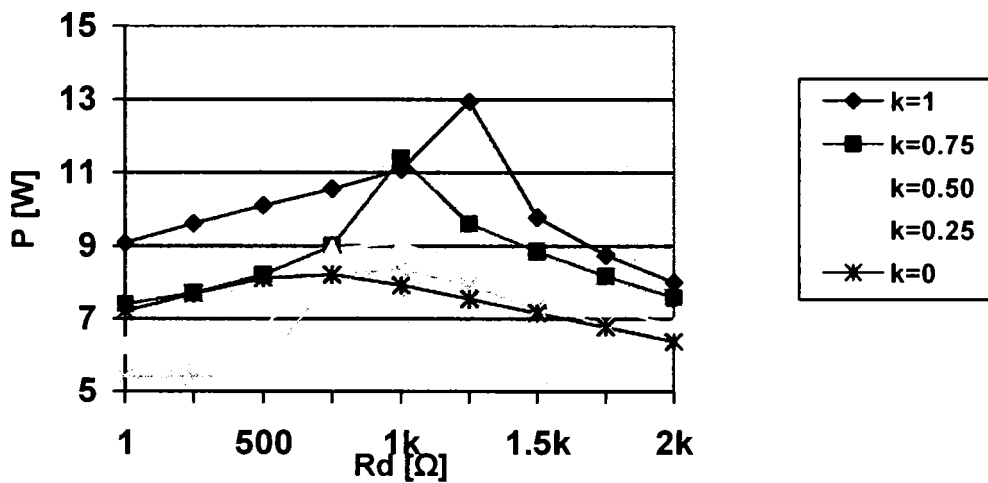


Figura 2.20 – Puterea disipată de sursele de căldură funcție de  $R_d$  și  $k$

O posibilă problemă care se ridică e următoarea: dacă majoritatea defectelor de scurt-circuit grilă-oxid nu se manifestă la nivel logic, de ce mai sunt considerate ca și defecte? Răspunsul la această întrebare vine tot din practică, deoarece s-a observat că circuitele ce aveau defecte de scurt-circuit grilă-oxid ce aveau o funcționare logică corectă, erau puțin fiabile, deoarece defectul se degradează în timp datorită stresului electric și se ajunge să cauzeze și erori logice. În plus defectele de scurt-circuit grilă-oxid afectează și comportamentul dinamic al circuitului.

### 2.2.1.2 Scurt-circuite externe

Pentru a înțelege modul de manifestare al defectelor de scurt-circuit externe din punct de vedere termic, mă voi folosi de exemplul din figura 2.21, în care există un defect de scurt-circuit între ieșirile porților  $P_1$  și  $P_2$ .

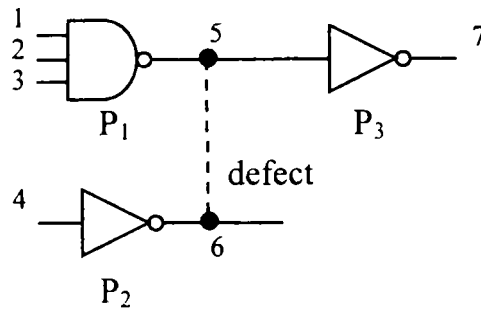


Figura 2.21 – Defect de scurt-circuit extern

La simularea în SPICE a circuitului implementat în tehnologia 0.18 microni, fără existența defectului s-a obținut consumul de putere din Figura 2.22 și Tabelul 2.4. Pentru figurile 2.22 și 2.23 circuitul a fost stimulat cu următoarea secvență de vectori:  $T_1 = \langle 0001 \rangle$ ,  $T_2 = \langle 0011 \rangle$ ,  $\langle 1111 \rangle$ ,  $T_8 = \langle 1110 \rangle$ ,  $\langle 0000 \rangle$  (Tabelul 2.4).

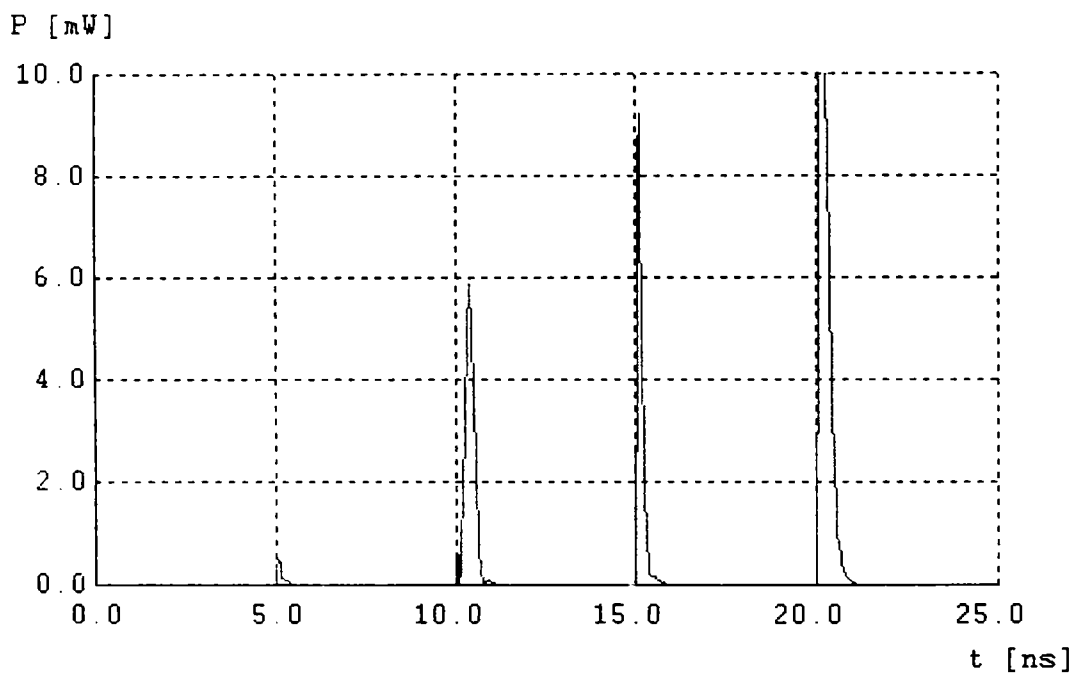


Figura 2.22 – Puterea disipată de circuitul fără defect

Defectul de scurt-circuit din figura 2.21 poate fi detectat atunci când nodurile 5 și 6 au valori logice diferite, astfel că se poate alege următorul vector de test  $T_1 = \langle 0, 0, 0, 1 \rangle$  (care se mai poate scrie  $\langle 0001 \rangle$ ). La aplicarea vectorului  $T_1$ , valorile logice ale ieșirilor porților  $P_1$  și  $P_2$  sunt "1" și respectiv "0", iar datorită defectului va exista o cale de curent între  $V_{DD}$  și masă (figura 2.24) prin cele trei tranzistoare pMOS ale porții  $P_1$  și prin tranzistorul nMOS al porții  $P_2$ . Datorită existenței acestei căi de curent între  $V_{DD}$  și masă în prezența defectului, acesta poate fi detectat printr-o testare termică la aplicarea vectorului de test  $T_1$ . Puterea disipată de circuit crește cu scăderea rezistenței defectului.

Vectori intrare	Fără defect		Cu defect		Vectori test
	$I_{DDQ}$ [pA]	$P_s$ [pW]	$I_{DDQ}$	$P_s$	
<0000>	112	561	112 pA	561 pW	
<0001>	93	466	1.48 mA	7.41 mW	T1
<0010>	136	680	136 pA	681 pW	
<0011>	117	585	1.56 mA	7.84 mW	T2
<0100>	116	578	116 pA	578 pW	
<0101>	97	484	1.56 mA	7.84 mW	T3
<0110>	206	1030	206 pA	1030 pW	
<0111>	187	936	1.43 mA	7.18 mW	T4
<1000>	115	577	115 pA	577 pW	
<1001>	96	482	1.56 mA	7.84 mW	T5
<1010>	186	931	186 pA	932 pW	
<1011>	167	836	1.43 mA	7.18 mW	T6
<1100>	188	941	188 pA	941 pW	
<1101>	169	846	1.43 mA	7.18 mW	T7
<1110>	151	757	1.63 mA	8.13 mW	T8
<1111>	133	663	133 pA	663 pW	

Tabelul 2.4 – Curentul  $I_{DDQ}$  și puterea consumată de circuitul fără defect și cu defect

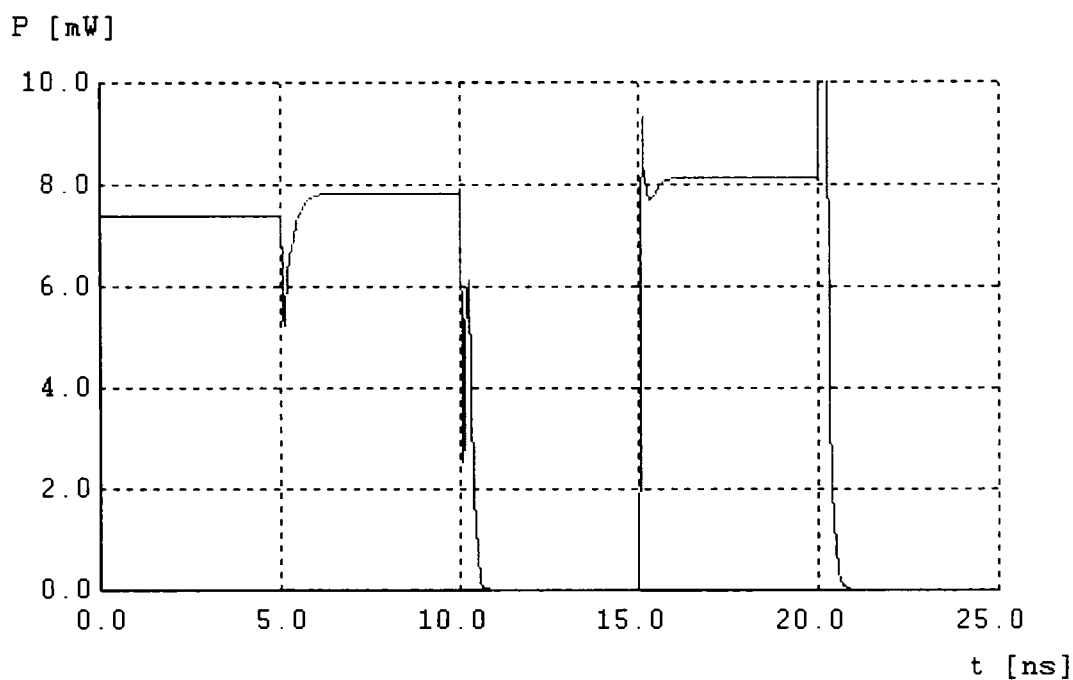


Figura 2.23 – Puterea consumată pentru circuitul defect

Un alt vector de test poate fi  $T_2 = \langle 0011 \rangle$ . Dacă la intrările circuitului se aplică  $T_2$ , căile de curent activate de acest vector de test sunt așa cum arată figura 2.25, prin cele două tranzistoare pMOS comandate cu  $V_1=0$  și  $V_2=0$  ale porții  $P_1$  și tranzistorul nMOS al porții  $P_2$ . În acest caz doar două din cele trei tranzistoare pMOS ale porții  $P_1$  conduc, deci rezistența între ieșirea porții și  $V_{DD}$  e mai mare decât în cazul aplicării vectorului  $T_1$ . Rezistența între ieșirea porții  $P_1$  și masă fiind aceeași (cele două tranzistoare nMOS ale porții  $P_2$  și rezistența defectului) ar trebui ca la aplicarea  $T_2$  să se obțină un curent mai mic sau cel mult egal cu cel obținut la aplicarea vectorului  $T_1$ . În realitate, însă se obține un curent mai mare la aplicarea vectorului de test  $T_2$  (Tabelul 2.4), ceea ce înseamnă că mai există o sursă suplimentară de curent. Această sursă suplimentară este inverterul  $P_3$ , datorită faptului că nivelul de tensiune la intrarea inverterului se degradează datorită defectului și se stabilește undeva între “0” și “1”, astfel că ambele tranzistoare din  $P_3$  vor conduce. Această cale de curent exista și pentru  $T_1$ , însă nu a fost atât de mare deoarece nivelul de tensiune de la intrarea  $P_3$  era mai apropiat

de "1" (datorită rezistenței mai mici între intrarea  $P_3$  și  $V_{DD}$ ) decât în cazul lui  $T_2$ . Această cale suplimentară de curent se numește sursa secundară pentru puterea disipată în timp ce sursa primară este stabilită de calea de curent prin defect.

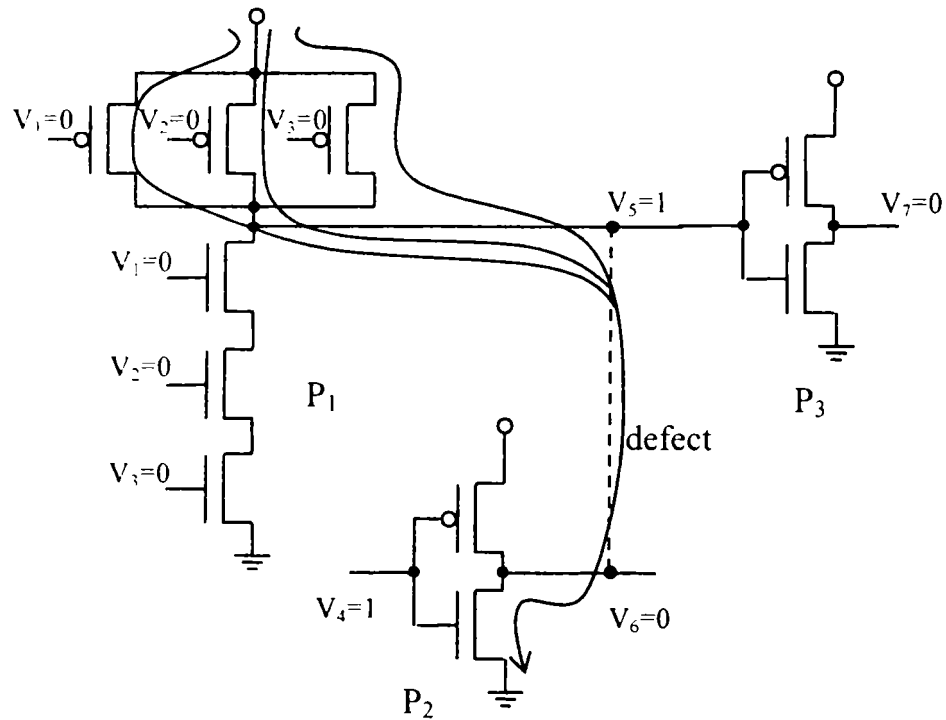


Figura 2.24 – Detecția defectului prin testarea termică (sursa de putere primară)

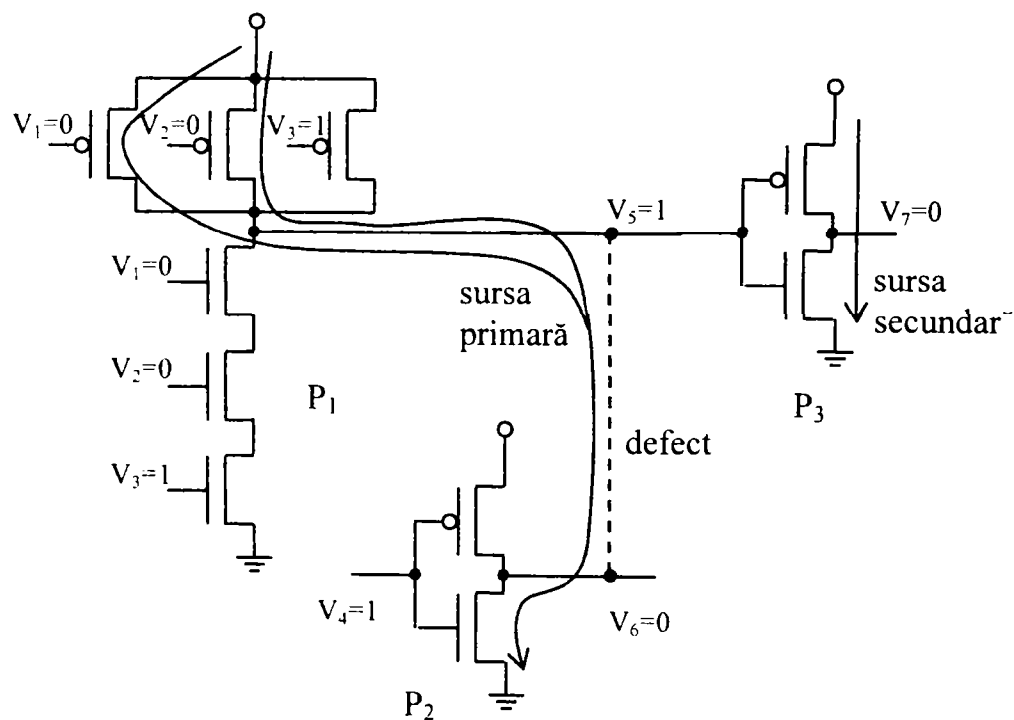


Figura 2.25 – Detecția defectului prin testarea termică (sursa primară și sursa secundară)

Din vectorii de test din Tabelul 2.4 doar  $T_4$  și  $T_6$  detectează defectul din Figura 2.21 prin testarea logică. Testarea logică poate detecta defectele de scurt-circuit rezistiv până la o anumită valoare a rezistenței defectului (Figura 2.26). În schimb testarea  $I_{DDQ}$  și testarea termică au o acoperire mai mare a defectelor de scurt-circuit (Figura 2.27).

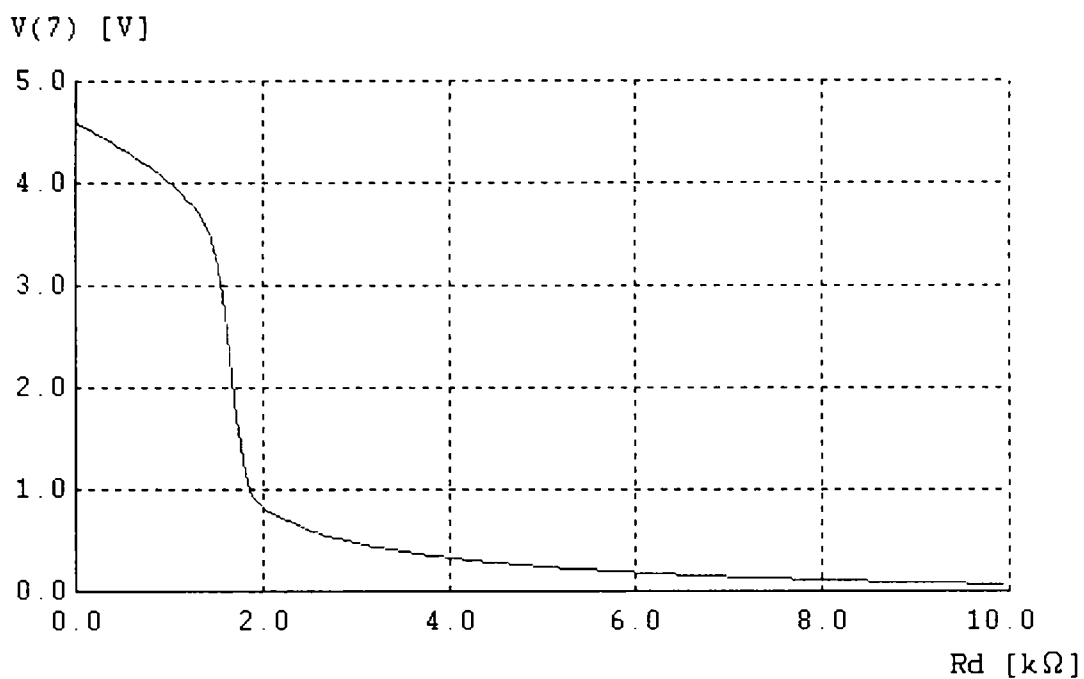


Figura 2.26 – Valoarea logică a ieșirii observabile a circuitului funcție de  $R_d$

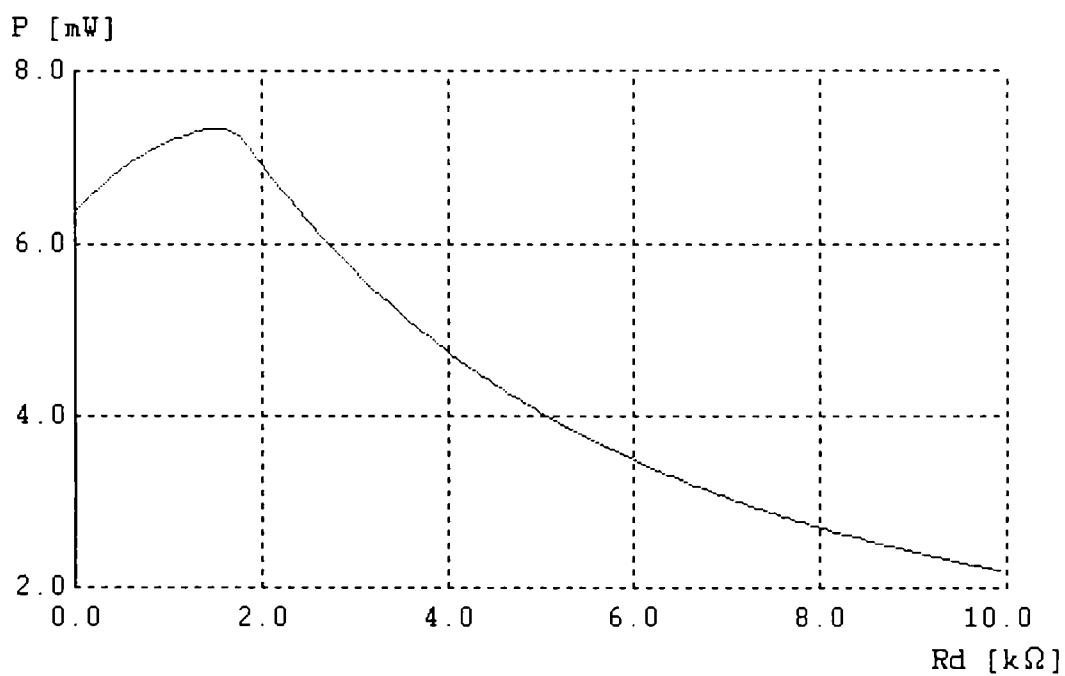


Figura 2.27 – Puterea disipată de circuit funcție de rezistența defectului  $R_d$  la aplicarea vectorului de test  $T_4$

### 2.2.1.3 Scurt-circuite interne

O subclasă a defectelor de scurt-circuit interne sunt scurt-circuitele ce apar între terminalele unui tranzistor MOS. Defectele de scurt-circuit ce pot apare între terminalele unui tranzistor MOS sunt următoarele:

- scurt-circuit grilă-sursă ( $d_1$ );
- scurt-circuit grilă-drenă ( $d_2$ );
- scurt-circuit sursă-drenă ( $d_3$ );
- scurt-circuit grilă-substrat ( $d_4$ );
- scurt-circuit sursă-substrat ( $d_5$ );
- scurt-circuit drenă-substrat ( $d_6$ ).

Aceste defecte de scurt-circuit conectează terminalele tranzistorului MOS fără a modifica structura și funcționarea acestuia. Astfel că defectul de scurt-circuit grilă-drenă este diferit de defectul de scurt-circuit grilă-oxid, în sensul că acesta din urmă apare în structura tranzistorului, modificând și funcționarea acestuia.

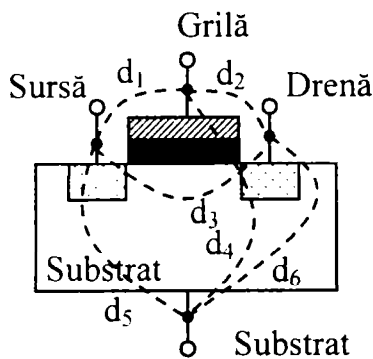


Figura 2.28 – Defectele de scurt-circuit pentru un tranzistor MOS

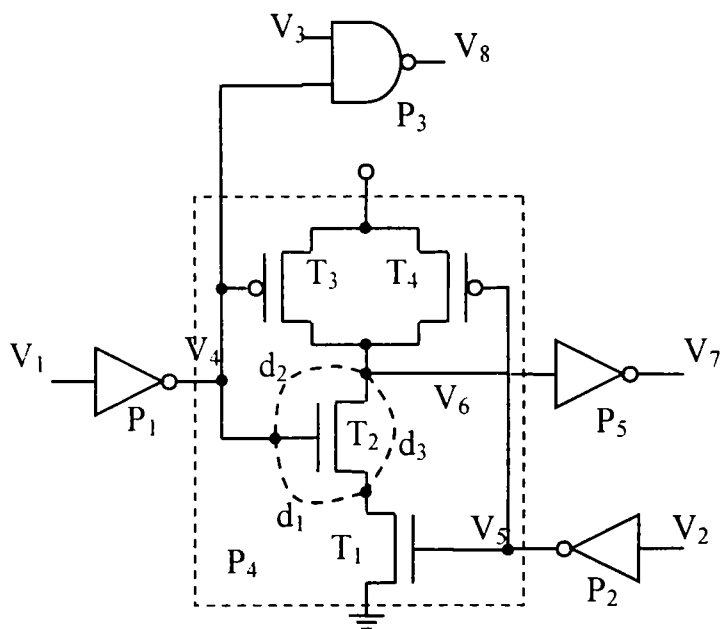


Figura 2.29 – Exemplu pentru prezentarea defectelor de scurt-circuit interne

### Scurt-circuit grilă-sursă

Pentru detecția defectului  $d_1$  trebuie ca nodurile  $V_4$  și  $V_5$  din figura 2.29 să aibă valori logice diferite. Un astfel de vector de test poate fi ales  $T_1 = \langle 00x \rangle$ . La aplicarea acestui vector tranzistorul  $T_1$  este deschis, deci sursa tranzistorului  $T_2$  va avea valoarea logică "0", iar grila lui  $T_2$  comandată de inversorul  $P_1$  va avea valoarea logică "1". Deci, la aplicarea vectorului  $T_1 = \langle 00x \rangle$ , se va stabili o cale de curent între  $V_{DD}$  și masă prin tranzistorul pMOS al porții  $P_1$ , defectul  $d_1$  și tranzistorul nMOS  $T_1$  al porții  $P_4$ , ce duce la creșterea semnificativă a puterii disipate de circuit, permițând astfel detecția defectului. Această cale de curent este calea primară de activare a defectului. Și în acest caz poate exista o cale secundară de curent între  $V_{DD}$  și masă. În funcție de rezistența defectului, tensiunea în nodul  $V_4$  se poate stabiliza la o valoare intermediară ce va deschide ambele grupuri de tranzistoare din poarta  $P_3$ , dacă  $V_3=1$ .

Puterea disipată de circuitul din figura 2.29 în prezența defectului  $d_1$ , la aplicarea secvenței de vectori:  $\langle 001 \rangle$ ,  $\langle 101 \rangle$ ,  $\langle 111 \rangle$ ,  $\langle 000 \rangle$ , este prezentată în figura 2.30. De aici se poate observa că  $T_1 = \langle 00x \rangle$  pune în evidență defectul  $d_1$  pentru testarea termică. Defectele de scurt-circuit rezistive pot fi detectate și prin testarea logică până la valoarea de  $2k$  a rezistenței defectului.



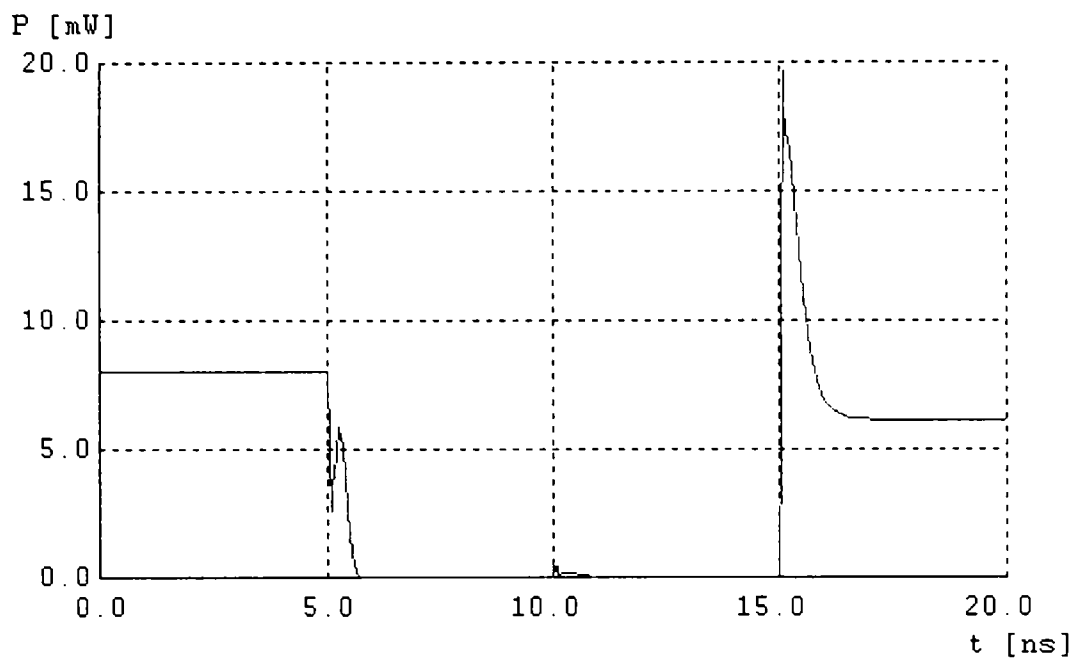


Figura 2.30 – Puterea disipată datorată prezenței defectului  $d_1$

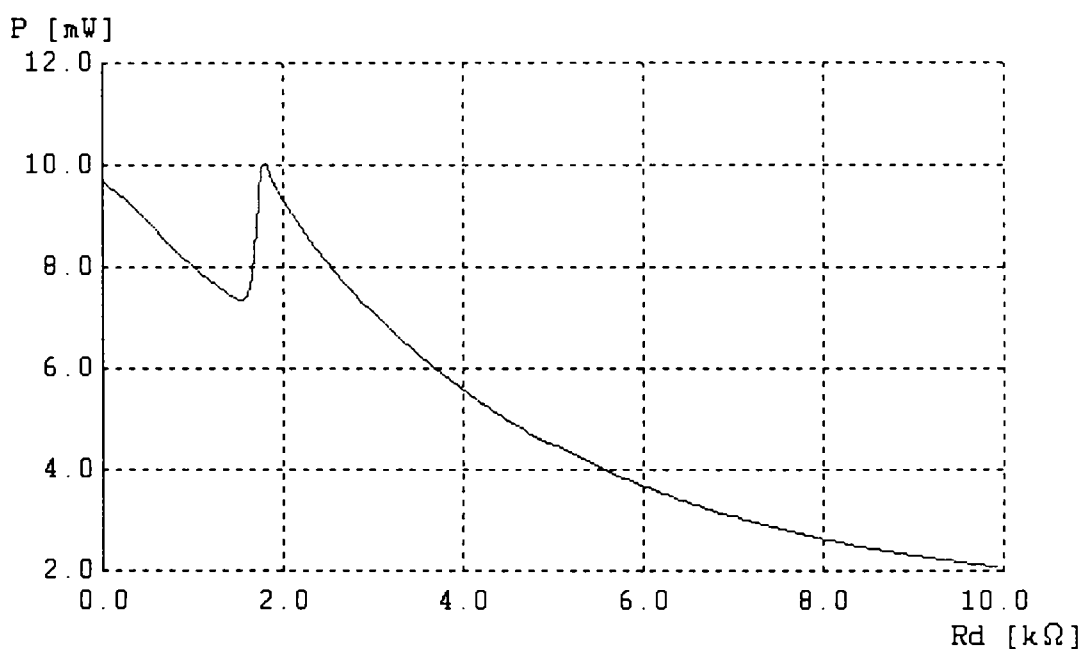


Figura 2.31 – Puterea disipată funcție de rezistența defectului  $d_1$

### Scurt-circuit grilă-drenă

Defectul  $d_2$  poate fi detectat aplicând la intrările circuitului  $T_2 = \langle 11x \rangle$ , prin stabilirea unei căi de curent între  $V_{DD}$  și masă prin cele două tranzistoare pMOS ( $T_3$  și  $T_4$ ) ale porții  $P_4$ , defectul  $d_2$  și tranzistorul nMOS al inversorului  $P_1$ . Și acest defect poate activa o cale secundară de curent datorită degradării nivelului logic al nodului  $V_4$ .

### Scurt-circuit sursă-drenă

Defectul  $d_3$  se poate activa folosind  $T_3 = \langle 10x \rangle$  prin creșterea curentului și a puterii disipate datorită existenței unei căi de curent între  $V_{DD}$  și masă prin tranzistorul pMOS  $T_3$  al porții  $P_4$ , defectul  $d_3$  și tranzistorul nMOS  $T_1$  tot al porții  $P_4$ .

## 2.2.2 Defecte de scurt-circuit în circuitele secvențiale

Deși testarea termică poate detecta un număr foarte mare din defectele de scurt-circuit din circuitele combinaționale, această afirmație nu este întru-totul adevărată în ceea ce privește circuitele secvențiale. Datorită buclelor de memorare din circuitele secvențiale anumite defecte de scurt-circuit nu pot fi puse în evidență nici de testarea  $I_{DDQ}$  nici de testarea termică [Chakravarty1997, Yamazaki1996]. Există două tipuri de defecte de scurt-circuit în circuitele secvențiale ce nu pot fi detectate de testarea termică:

- scurt-circuit ce conectează o buclă de memorare de un subcircuit combinațional ( $d_1$  din Figura 2.32);
- scurt-circuit ce conectează două bucle de memorare ( $d_2$  din Figura 2.32).

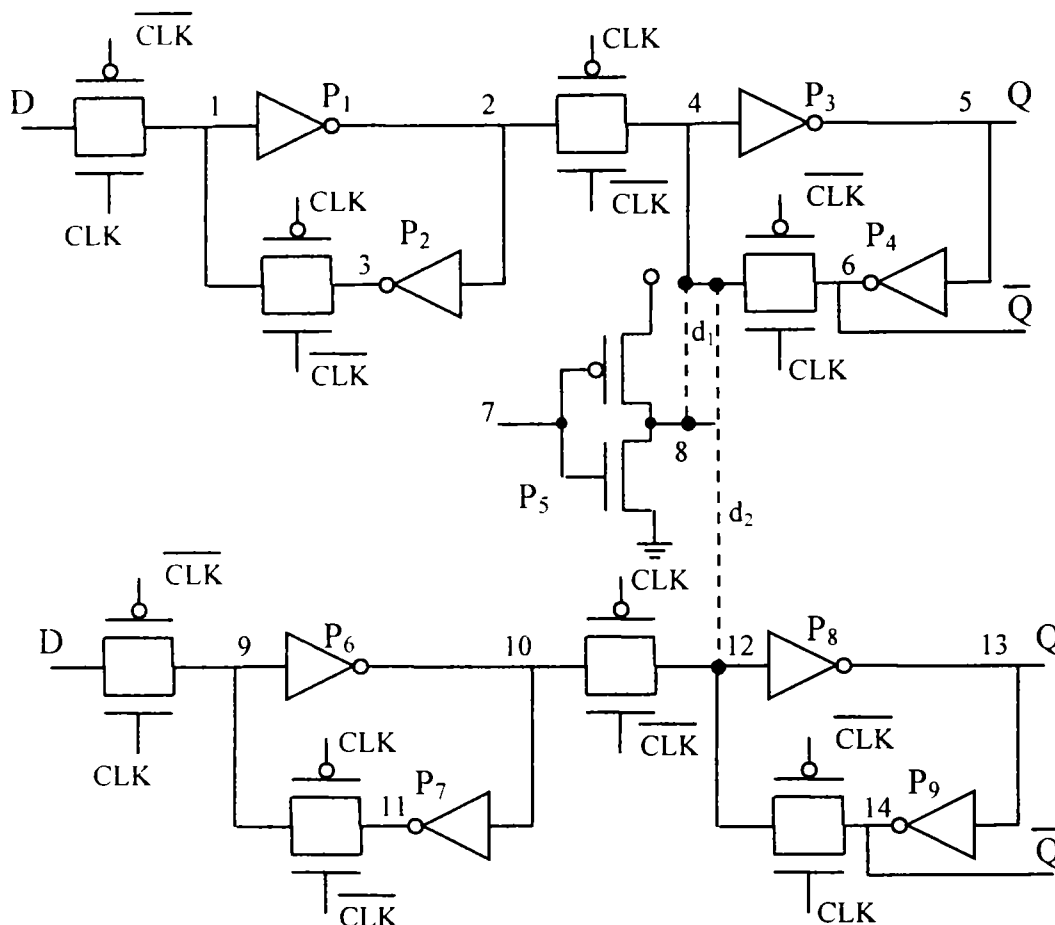


Figura 2.32 – Defecte de scurt-circuit în circuitele combinaționale ce nu pot fi detectate de testarea termică (bistabil D master-slave)

Pentru primul tip de defect de scurt-circuit ( $d_1$  din figura 2.32), indiferent de valoarea aplicată la intrarea de tact (CLK), defectul leagă nodul 8 de una din cele două bucle cu reacție din primul bistabil (nodul 4). În cazul în care rezistența defectului este mică, nodurile 8 și 4 au aceeași valoare logică, chiar dacă vectorul de intrare sau secvența de vectori de intrare încearcă să le seteze pe valori logice diferite. Valoarea logică ce se stabilește la nivelul defectului este dependentă de parametrii tranzistoarelor ce formează porțile  $P_5$  și  $P_1$  sau  $P_4$ , luând valoarea porții dominante. Valoarea nouă va fi memorată de bucla de memorare din bistabil, astfel că nu va exista nici o cale de curent de la  $V_{DD}$  la masă, care să ducă la creșterea puterii consumate (Figura 2.33). Deci acest tip de defect nu poate fi detectat de testarea termică sau testarea  $I_{DDQ}$ , însă datorită modificării valorii logice la nivelul unui nod, acest defect este posibil a fi detectat de testarea logică. Dacă rezistența defectului este mare ( $> 2k\Omega$ ) va exista o cădere de tensiune pe rezistență, iar cele două noduri ale defectului vor avea valori logice diferite. Din acest motiv defectul nu poate fi detectat prin testarea logică, însă deoarece

există o cale de curent de la  $V_{DD}$  la masă (Figura 2.34), el poate fi detectat prin testarea  $I_{DDQ}$  sau testarea termică.

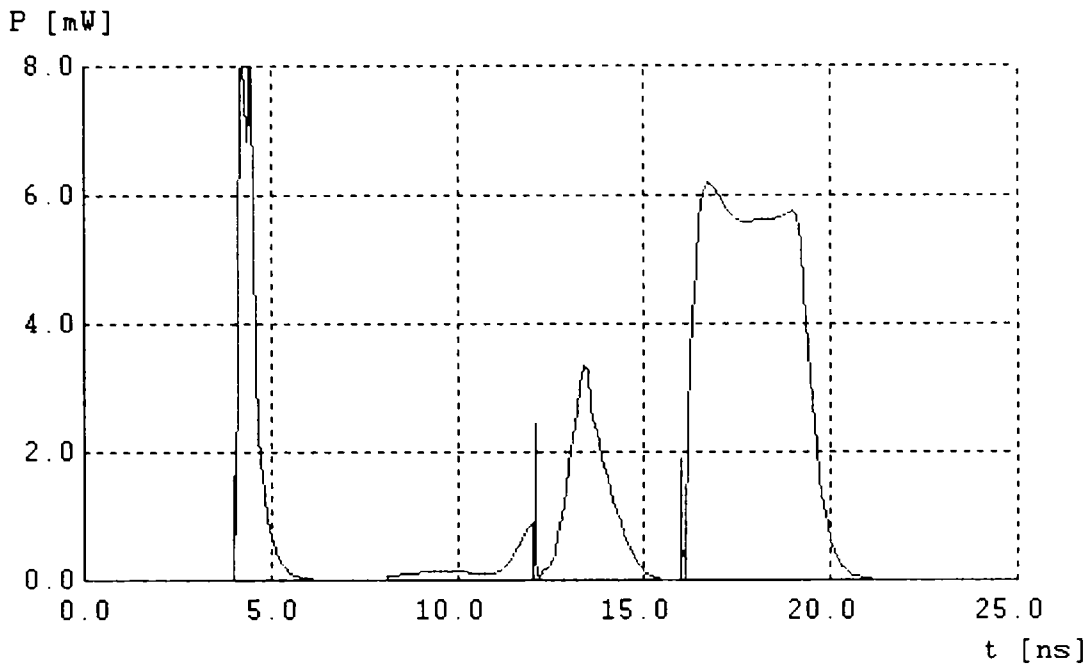


Figura 2.33 – Puterea disipată de circuit pentru  $R_d = 100\Omega$

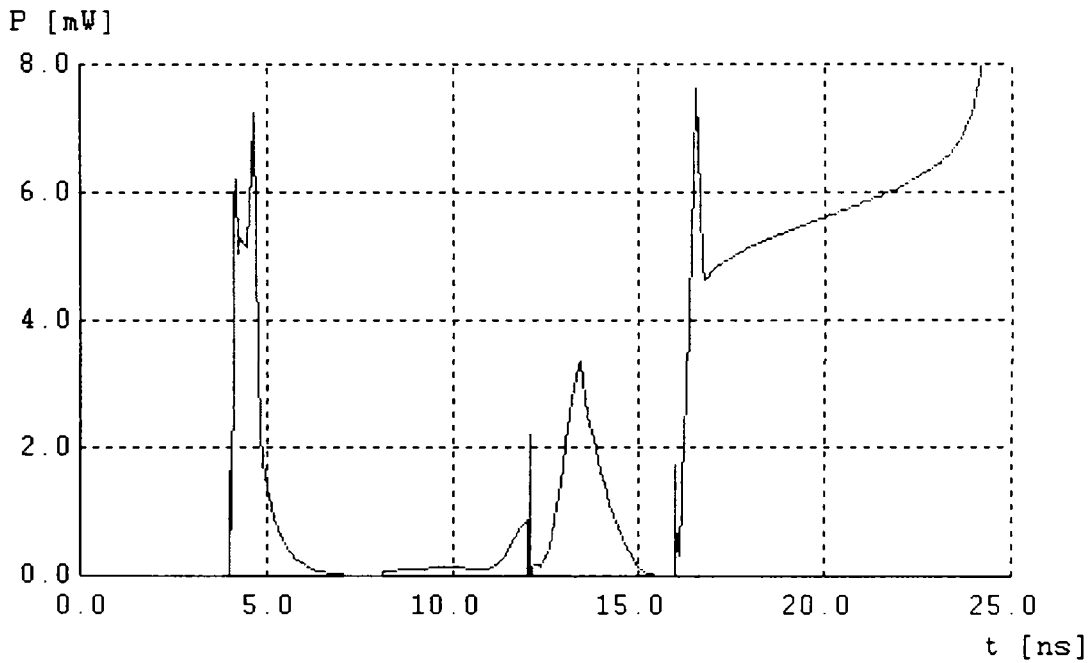


Figura 2.34 – Puterea disipată de circuit pentru  $R_d = 5K\Omega$

În cazul celui de-al doilea tip de defect de scurt-circuit ( $d_2$  din Figura 2.32), indiferent de valoarea logică a intrărilor CLK, acesta face legătura între câte o buclă de reacție din fiecare bistabil. La fel ca în cazul precedent, dacă defectul de scurt-circuit are o rezistență mică, atunci el nu poate fi pus în evidență prin testarea termică, însă este posibil să fie detectat prin testarea logică. Defectele de scurt-circuit rezistiv pot fi detectate atât prin testarea  $I_{DDQ}$  cât și prin testarea termică.

### 2.2.3 Defecte de întrerupere

Un defect de întrerupere poate fi cauzat de lipsa unei porțiuni a materialului conductor (stratul de metalizare) sau a unei izolări greșite a două noduri. Din punctul de vedere al localizării defectelor, întreruperile se împart în două categorii, fiecare având comportamente diferite:

- întreruperi interne (care implică nodurile interne unei porți logice);
- întreruperi externe (care implică liniile de semnal de conectare a porților).

Dacă o întrerupere internă cauzează despărțirea tuturor căilor de mică impedanță de la oricare din  $V_{SS}$  sau  $V_{DD}$  la ieșirea porții, aceasta va fi blocată pe o valoare logică. Dacă defectul întrerupe câteva, dar nu toate căile de la  $V_{SS}$  sau  $V_{DD}$  la ieșirea porții, aceasta se va comporta ca un circuit secvențial [Ferguson1990].

Întreruperile externe din circuitele CMOS statice cauzează porți flotante. Există două tipuri de întreruperi externe:

- întreruperile care cauzează instabilitate numai într-una rețelele de tranzistoare pFET sau nFET ale porții logice (figura 2.35 a);
- întreruperile care cauzează instabilitate pentru ambele rețele de tranzistoare pFET și nFET ale porții logice (figura 2.35 b).



Figura 2.35 – Defecte de întrerupere

Comportamentul porții logice afectată de o întrerupere a liniei de semnal este determinată de starea (conductoare – neconductoare) a tranzistoarelor conectate la un nod flotant. Dacă tranzistoarele cu porți flotante sunt permanent neconductoare, comportarea circuitului va fi echivalentă cu comportarea întreruperilor interne. Dacă tranzistoarele cu porți flotante sunt permanent conductoare, comportarea porții va fi aceeași cu a unui tranzistor saturat (stuck-on). Pentru cazul din figura 2.35 (b) în care perechea de tranzistoare flotante, va avea cel mai probabil un tranzistor conductor și unul neconductor [Ferguson1990].

Din punctul de vedere al dimensiunii defectului întreruperile pot fi [Chakravarty1997]:

- întreruperi mari (largi) – la aplicarea unei tensiuni între cele două capete ale defectului nu trece deloc curent prin acesta;
- întreruperi mici ( $< 100\text{nm}$ ) – la aplicarea unei tensiuni între cele două capete ale defectului se stabilește un curent de scurgere datorită efectului de tunel.

Impactul defectelor de întrerupere asupra funcționării și performanței unui circuit depinde de localizarea și dimensiunea defectului, precum și de parametrii electrice ai circuitului. Majoritatea defectelor de întrerupere nu pot fi detectate de testarea termică.

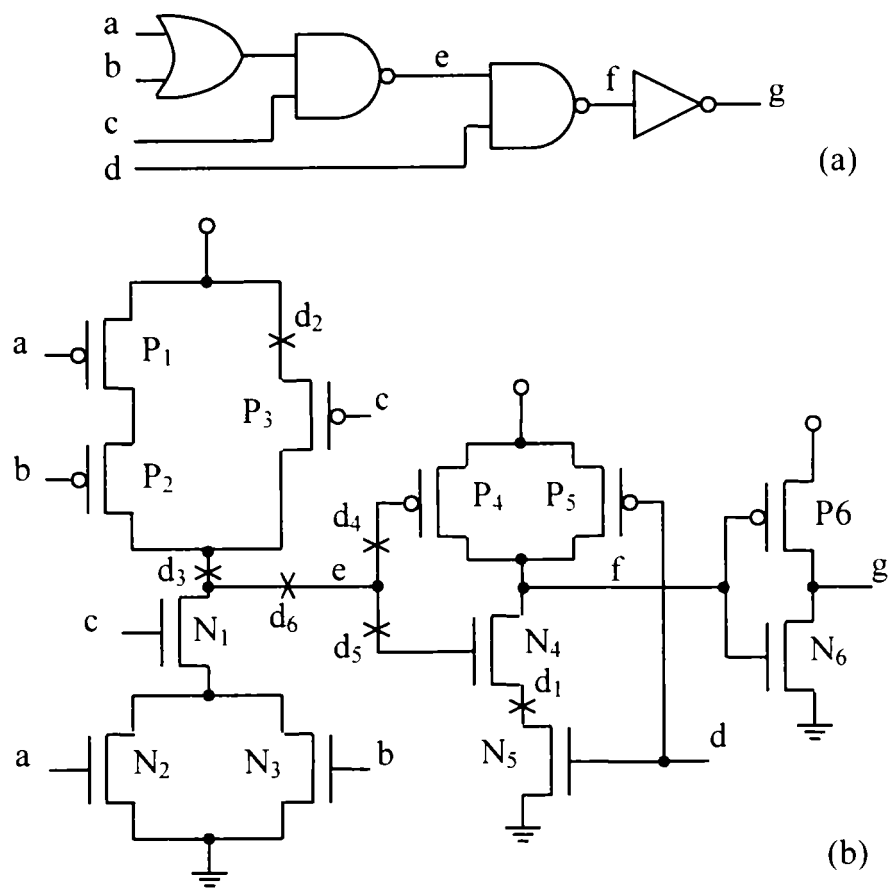


Figura 2.36 – Tipuri de defecte de întrerupere

### Întreruperea sursei sau drenei

În funcție de localizarea defectului există două cazuri:

- un defect de întrerupere a sursei unui tranzistor MOS poate apare pe unica cale către  $V_{DD}$  sau masă a unei porți logice. Un exemplu pentru acest caz este defectul  $d_1$  ce poate fi considerat ca defect de întrerupere al sursei tranzistorului nMOS  $N_4$  sau ca un defect de întrerupere a drenei tranzistorului nMOS  $N_5$ . Dacă acest defect este larg atunci nu există nici o cale de curent de la f la masă, astfel că acest defect se manifestă ca un defect de stuck-open al tranzistorului nMOS  $N_4$ . Aceste defecte se pot detecta folosind perechi de vectori de test pentru defectele de tip stuck-open. Pentru  $d_1$  perechea de vectori de test  $T_1 = \langle 1110 \rangle$  urmat de  $T_2 = \langle 0001 \rangle$  vor menține ieșirea f la "1" și defectul va fi detectat ca un defect de stuck-open al tranzistorului  $N_4$  sau  $N_5$ . Dacă defectul de întrerupere este mic atunci există un curent infim care va împiedica detecția acestor defecte prin testarea logică. Aceste defecte pot fi, însă detectate ca un defect de întârziere pentru o tranziție "1"- "0" la nivelul nodului f. Există cazuri în care aceste defecte pot fi detectate și de testarea  $I_{DDQ}$  și testarea termică, deoarece există un interval de timp în care nivelul logic al nodului f este apropiat (datorită tranziției lente) de nivelul de prag, astfel că cele două tranzistoare ale porții inversoare sunt în conducție și se produce un curent suplimentar. Probabilitatea de detecție a acestor defecte prin testarea termică este însă mică.
- defectul nu întrerupe unica cale către  $V_{DD}$  sau masă unei porți logice ( $d_2$ ). Pentru o întrerupere mare comportamentul este cel specific pentru un defect de stuck-open al tranzistorului pMOS  $P_3$ . În testarea logică acest defect poate fi detectat cu secvența de test:  $T_3 = \langle 1110 \rangle$ ,  $T_4 = \langle 1101 \rangle$  care va menține nodul e la "0" și deoarece  $d=1$  valoarea nodului e va ajunge la ieșirea observabilă g a circuitului. În cazul când întreruperea este mică ea poate fi detectată ca un defect de întârziere pentru tranziția "0"- "1" a nodului e, la aplicarea acelorași vectori de test ( $T_3$  urmat de  $T_4$ ). Interesant este faptul că aceste defecte

de întrerupere sunt detectate în proporție mare de testarea  $I_{DDQ}$  [Chakravarty1997]. Explicația constă în faptul că valoarea tensiunii în nodul e nu rămâne pe valoarea "0". Datorită tranzițiilor de la intrările circuitului, "0" pe nodul e se degradează iar tensiunea în acest punct se stabilește undeva în jurul nivelului de prag. Astfel că dacă  $d=1$  în poarta a treia se va stabili o cale de curent între  $V_{DD}$  și masă prin tranzistoarele parțial deschise  $P_4$  și  $N_4$ .

Pentru a verifica posibilitatea de detecție a defectelor de întrerupere prin testarea termică, s-a simulat în SPICE circuitul din figura 2.36 pentru o implementare în tehnologia 0.18 microni. Pentru detecția defectelor de întrerupere se aplică o secvență de doi vectori de test, primul pentru inițializarea circuitului iar cel de-al doilea pentru detecția defectului.

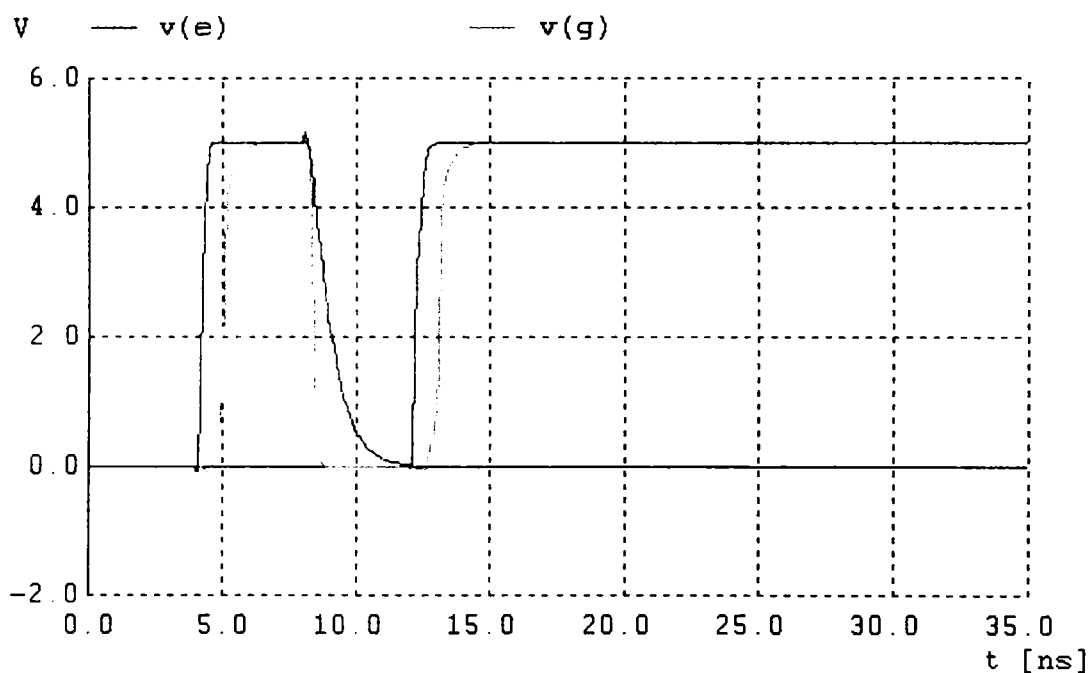


Figura 2.37 – Valoarea logică a nodurilor e și g la aplicarea secvenței  $T_1, T_2, T_3, T_4$  la circuitul fără defecte

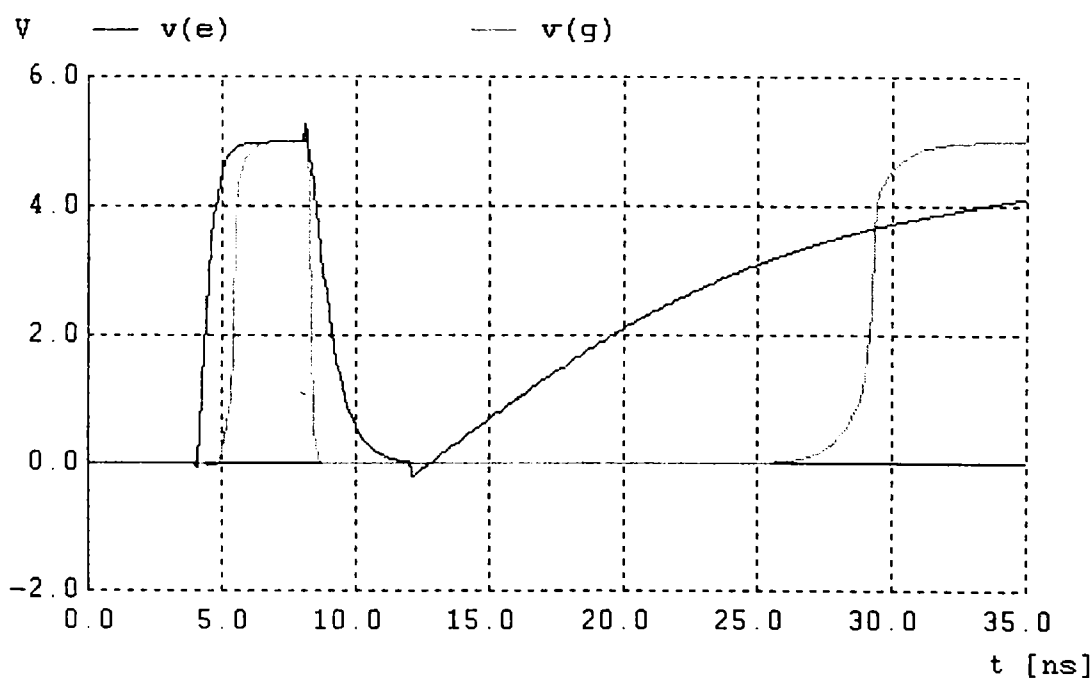


Figura 2.38 – Valoarea logică a nodurilor e și g la aplicarea secvenței  $T_1, T_2, T_3, T_4$  la circuitul din figura 2.36 cu defectul  $d_2$

În graficele din figurile 2.37 și 2.38 este prezentată variația tensiunii în nodurile e și g la aplicarea secvenței de vectori de test  $T_1, T_2, T_3, T_4$ . În prezența defectului de întrerupere  $d_2$ , având o rezistență de  $500\text{ k}\Omega$ , defectul se manifestă prin întârzierea tranziției de la “0” la “1”, a ieșirii e a primei porți, astfel poate fi detectat prin testarea pe baza întârzierii. În graficele din figurile 2.39 și 2.40 s-a prezentat variația puterii disipate de circuit la aplicarea secvenței de vectori de test  $T_1, T_2, T_3, T_4$ . Datorită defectului, circuitul disipă o cantitate mai mare de căldură la aplicarea vectorului  $T_4$ , însă pe o perioadă limitată de timp, dar suficient de mare pentru a putea fi detectat. În funcție de structura defectului de întrerupere circuitul se comportă în mod diferit, astfel că nu orice astfel de defect poate fi detectat de testarea termică însă nici de testarea logică sau testarea  $I_{DDQ}$ .

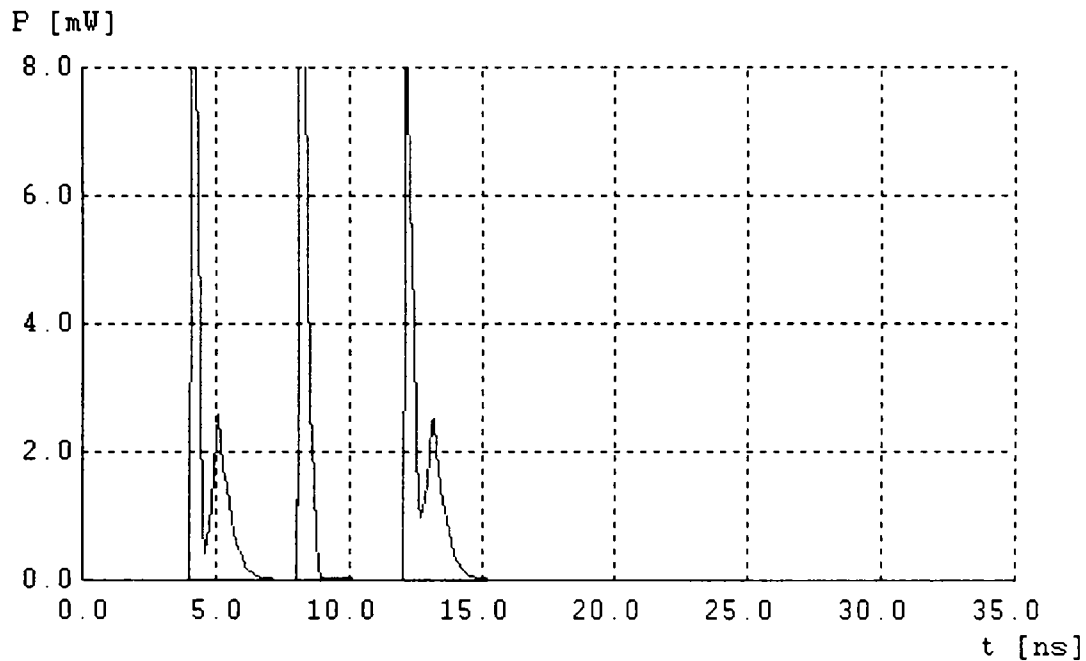


Figura 2.39 – Puterea disipată de circuitul fără defecte la aplicarea secvenței  $T_1, T_2, T_3, T_4$

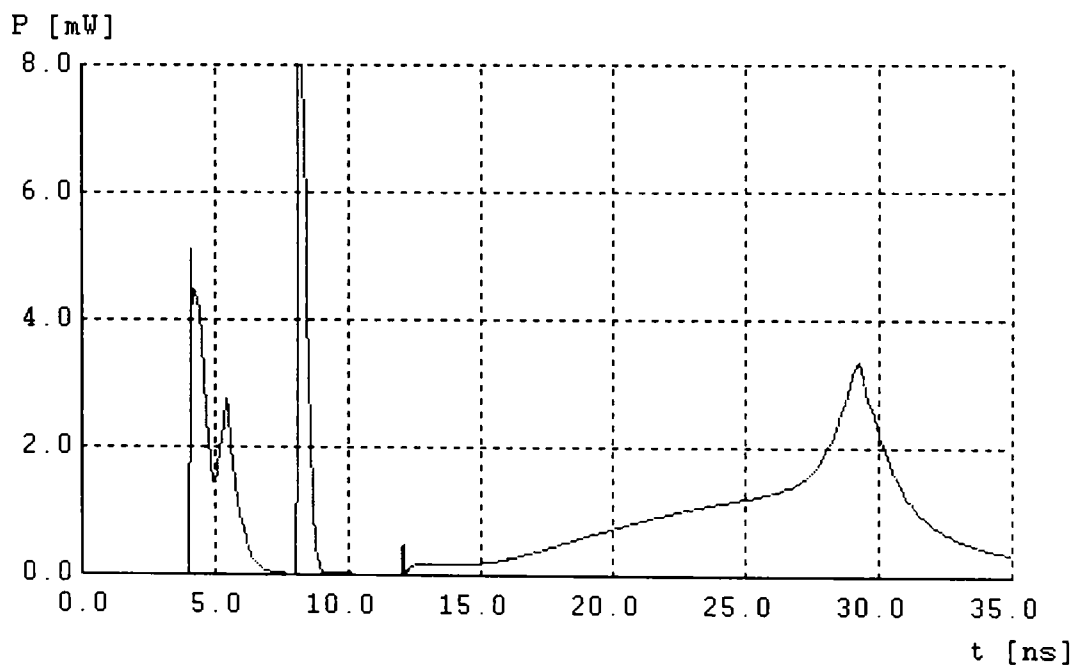


Figura 2.40 – Puterea disipată de circuit în prezența defectului  $d_2$

### Înteruperea rețelei de tranzistoare nMOS sau pMOS

Un exemplu de astfel de defect este  $d_3$  din figura 2.36. Defectul  $d_3$  decuplează întreaga rețea de tranzistoare pMOS de nodul e. Această clasă de defecte se comportă în mod asemănător cu defectele de întrerupere a sursei sau drenei pentru cazul apariției pe calea unică către  $V_{DD}$  sau masă. Și pentru aceste defecte probabilitatea de detecție a testării termice este mică.

### Înteruperea grilei

Defectele de întrerupere a grilei au ca efect obținerea unui tranzistor MOS a cărui grilă nu se setează la nici o valoare logică de nici o ieșire. Acești tranzistori poartă numele de tranzistori cu grila flotantă. Această clasă de defecte este cea mai dificilă de modelat datorită complexității ei și datorită faptului că depinde foarte mult de mediu [Chakravarty1997]. Întreruperea grilei poate cauza blocarea tranzistorului, deschiderea slabă a acestuia sau chiar saturarea tranzistorului.

Defectele  $d_5$  și  $d_6$  din figura 2.36 fac parte din această categorie. Aceste defecte, în funcție de modul în care se stabilește starea tranzistorului, pot fi detectate ca defecte de stuck-open (dacă tranzistorul e blocat), defecte de întârziere (tranzistorul e deschis slab) sau defect  $I_{DDQ}$  dacă tranzistorul e saturat.

Există unele situații în care defectele din această categorie pot fi detectate și de testarea termică. În figura 2.41 este prezentată variația puterii disipate de circuit la aplicarea aceleiași secvență de vectori de test  $T_1, T_2, T_3, T_4$ , în prezența defectului  $d_5$ , având o rezistență de  $100k\Omega$ .

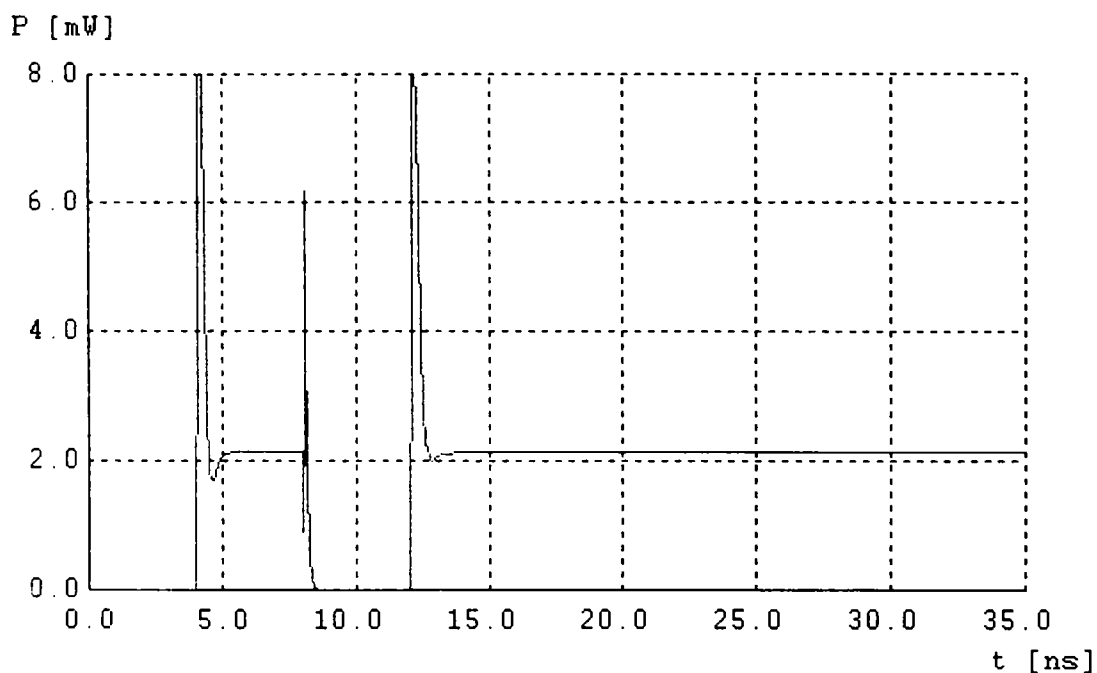


Figura 2.41 – Puterea disipată de circuit în prezența defectului  $d_4$

### Înteruperea ieșirii

Defectul  $d_6$  din figura 2.36 intră în această clasă de defecte și are ca efect obținerea a două tranzistoare complementare cu grila flotantă. În acest caz ambele tranzistoare conduc, astfel că acest tip de defecte pot fi detectate prin testarea  $I_{DDQ}$  și testarea termică

În figura 2.42 este trasată variația puterii disipate la aplicarea secvenței de vectori de test  $T_1, T_2, T_5 = \langle 1111 \rangle$  în prezența defectului  $d_6$ . Defectul este considerat scurt și are o rezistență de  $100k\Omega$ . Pentru un anumit interval de timp, limitat, defectul poate fi pus în evidență prin testarea termică.



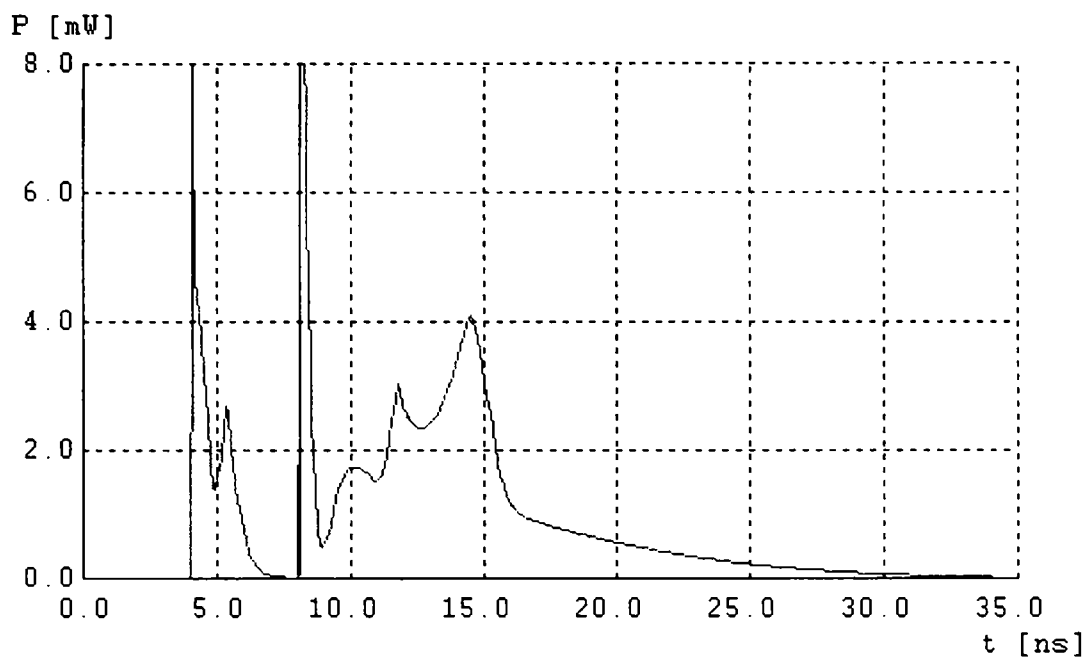


Figura 2.42 – Puterea disipată de circuit în prezența defectului  $d_6$  la aplicarea  $T_1, T_2, T_3$

Din punct de vedere logic circuitul cu defectul  $d_6$  se comportă la ieșirea observabilă  $g$  ca în figura 2.43. Se observă apariția unui salt de tensiune parazit ce este destul de complicat de detectat prin testarea logică, însă acesta poate să aibă efecte tranzitorii în funcționarea circuitului.

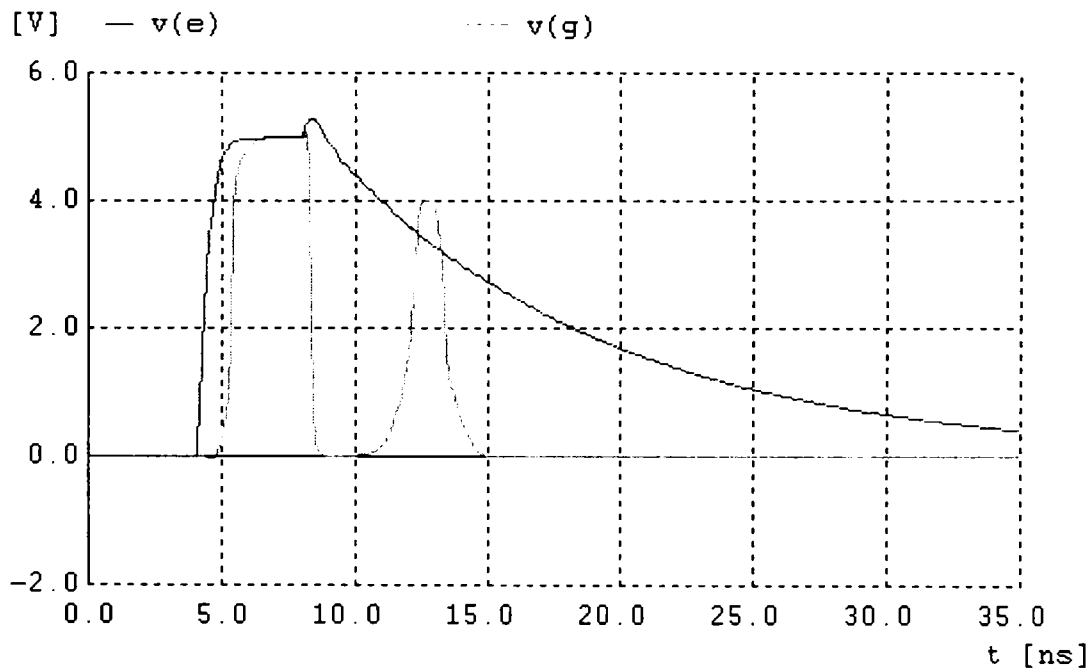


Figura 2.43 – Valorile logice în punctele  $e$  și  $g$  ale circuitului din figura 2.36 în prezența defectului  $d_6$  la aplicarea secvenței  $T_1, T_2, T_3$

## 2.3 Studiul transferului de căldură în circuitele integrate

Termotehnica este o parte a fizicii tehnice care se ocupă cu studiul legilor de obținere a căldurii din alte forme de energie, cu transformarea, propagarea și utilizarea căldurii. Obținerea și transformarea căldurii din și în alte forme de energie sunt procese în care este respectat principiul conservării și transformării energiei. Pentru a se obține căldură trebuie transformată, în cantitate echivalentă, energie care se prezintă sub alte forme: mecanică, electrică, chimică, etc. [Lienhard2004, Remsburg2001, Krauss1987, Negru1990]

Propagarea căldurii are loc în medii cu temperaturi neuniforme și se desfășoară de la sine de la temperaturi ridicate la temperaturi scăzute. Studiul fenomenelor de propagare a căldurii permite să se determine relații de calcul privind optimizarea structurilor, dimensionarea schimbătoarelor de căldură, dirijarea unor procese tehnologice, etc.

### 2.3.1 Baza axiomatică a termodinamicii clasice

Termodinamica clasică este construită pe două postulate și trei principii, care alcătuiesc baza sa axiomatică. Cele cinci afirmații cuprinse în aceste postulate și principii reprezintă generalizarea observațiilor macroscopice și sunt acceptate fără a putea fi demonstrate matematic [Lienhard2004, Leonachescu1981, Negru1990].

**Postulatul echilibrului** - Un sistem izolat atinge starea de echilibru după un anumit interval de timp, stare din care nu poate ieși de la sine.

**Postulatul al doilea al termodinamicii** - Parametrii interni ai unui sistem aflat în echilibru depind de parametrii externi și de temperatură.

**Primul principiu al termodinamicii** - Într-un sistem izolat energia rămâne constantă indiferent de transformările care au loc în interior.

**Principiul al doilea al termodinamicii** - Într-un sistem izolat fenomenele se desfășoară de la sine cu maximum de probabilitate de la potențiale ridicate la potențiale scăzute.

**Principiul al treilea al termodinamicii** - Izoterma de zero absolut coincide cu adiabata.

Transmiterea căldurii reprezintă procesul de transfer al energiei interne în interiorul aceluiași sistem termodinamic, din zonele cu temperaturi mai mari spre zonele cu temperaturi mai mici, sau între sisteme diferite, aflate la temperaturi diferite, de la sistemul cu temperatură mai mare spre cel cu temperatură mai mică. Transmiterea căldurii este un fenomen ireversibil și dependent de timp, care respectă principiile termodinamicii. Principiul I este respectat prin faptul că energia internă cedată de către sistemul mai cald este egală cu energia internă primită, de către sistemul mai rece. Principiul al II-lea este de asemenea respectat având în vedere formularea lui Clausius referitoare la sensul spontan de trecere a căldurii: "căldura trece spontan numai de la corpul mai cald la corpul mai rece". [Leonachescu1981, Negru1990]

Cercetările teoretice și experimentale au pus în evidență caracterul complex al acestui fenomen. Din această cauză studiul transmiterii căldurii se împarte în studiul mai multor fenomene simple de transmitere a căldurii. Se deosebesc trei fenomene simple principial diferite, de transmitere a căldurii: conducția, convecția și radiația. Legile de bază ale acestor moduri de transmitere a căldurii sunt complet diferite. În realitate însă aceste trei fenomene simple nu se întâlnesc separat decât în cazuri particulare, așa încât, de obicei, se pune problema asamblării lor.

## 2.3.2 Generarea căldurii în dispozitivele electronice

În timpul funcționării dispozitivele electronice, atât cele active cât și cele pasive, degajă căldură. La trecerea curentului printr-un dispozitiv electric, o parte din putere este disipată ca și energie termică, ceea ce duce la încălzirea respectivului dispozitiv. Puterea disipată de un dispozitiv electronic este dată de ecuația 2.1, dacă tensiunea și curentul sunt constante:

$$P = V \cdot I \quad (2.1)$$

sau puterea disipată medie e dată de ecuația 2.2 în cazul când variază tensiunea sau curentul în funcție de timp:

$$P = \frac{1}{t} \int_0^t V(t) \cdot I(t) dt \quad (2.2)$$

unde:

P – puterea disipată [W]

V – tensiunea de alimentare a dispozitivului [V]

I – curentul absorbit de dispozitiv [A] sau variația curentului în funcție de timp I(t).

Puterea disipată de circuitele integrate CMOS are două componente (Ecuația 2.3):

- puterea statică – puterea disipată de circuit în după terminarea fenomenelor tranzitorii (Ecuația 2.4);
- puterea dinamică – puterea disipată în timpul tranzițiilor. Această componentă a puterii are și ea la rândul ei două subcomponente (Ecuația 2.5):
  - puterea consumată pentru încărcarea și descărcarea capacităților parazite în timpul tranzițiilor (primul termen din Ecuația 2.5). Aceasta este componenta dominantă a puterii disipate, având în jur de 90% din total;
  - puterea de scurt-circuit este cauzată de tranzistorii ce sunt deschiși în timpul schimbărilor de stare (al doilea termen din Ecuația 2.5).

$$P = P_S + P_D \quad (2.3)$$

Componenta statică a puterii disipate de un circuit CMOS depinde de curentul de scurgere  $I_{DDQ}$ :

$$P_S = I_{DDQ} \cdot V_{DD} \quad (2.4)$$

Componenta dinamică a puterii disipate de un circuit CMOS depinde de frecvența și geometria dispozitivului:

$$P_D = \frac{CV_{DD}^2}{2} f + Nqf \quad (2.5)$$

unde:

$V_{DD}$  – tensiunea de alimentare [V];

C – capacitatea de intrare a circuitului [F];

f – frecvența [Hz];

N – numărul de porți ce conduc;

q – puterea de scurt-circuit per poartă [W/Hz].

Puterea disipată de traseele de metalizare din circuitele integrate, precum și puterea disipată datorită defectelor de scurt-circuit este dată de legea lui Jule [Reensburg2001]:

$$P = I^2 R = I^2 \cdot \rho \frac{L}{A} \quad (2.5)$$

unde:

q – rezistivitatea materialului [ $\Omega/m$ ];

L – lungimea conductorului;

A – aria secțiunii transversale a conductorului.

## 2.3.3 Modurile elementare de transfer termic

Propagarea căldurii este un fenomen complex în care se disting trei moduri de transfer termic: [Lienhard2004, Remsburg2001, Krauss1987, Negru1990, Leonachescu1981]

**Conducția termică** este fenomenul de transfer termic prin corpuri fără mișcări aparente, caracteristic solidelor și straturilor subțiri de fluid în repaus. Conducția termică depinde de natura corpului și de spectrul de izoterme.

**Convecția termică** este fenomenul de transfer termic prin corpuri cu mișcări aparente. Se manifestă la fluide în mișcare și în procesele de schimbare de stare de agregare. După cauza care generează mișcarea fluidului, fenomenele convective se clasifică în două mari categorii:

- convecția liberă sau naturală când fluidul se mișcă pe baza forțelor arhimedice generate de neuniformitatea câmpului de temperaturi, vitezele de mișcare fiind în general mici;
- convecția forțată când fluidul se mișcă pe baza forțelor rezultate din diferențe de presiuni statice sau dinamice create artificial sau natural. În acest caz vitezele de mișcare sunt sensibil mari.

Convecția căldurii se intensifică simultan cu creșterea vitezelor de mișcare sau cu accentuarea neuniformității câmpului de temperaturi. Ea se manifestă la suprafața de separație dintre un solid și un fluid, dar și la suprafața de separație a fazelor componente ale fluidelor polifazice.

**Radiația termică**, ca fenomen de transport de energie care utilizează drept suport material undele electromagnetice, se manifestă termic la contactul cu corpurile pe care cade, fiind absorbită, reflectată sau străbătând corpurile prin transparentă în proporții ce depind de natura acestor corpuri. În același timp corpurile emit energie radiantă direct proporțional cu puterea a patra a temperaturii absolute a acestora, cu suprafața și timpul de emisie. Mecanismul emisiei și absorbției energiei radiante se aplică prin teoria cuantică, cu ajutorul modelelor atomice create de Bohr și Sommerfeld.

### 2.3.3.1 Conducția termică

Propagarea căldurii prin conducție apare ca fenomen determinat la solide și la straturi subțiri de fluide în repaus, care prezintă neuniformități ale câmpului de temperaturi. Transferul de căldură prin conducție apare pe două căi distincte: prin interschimbarea energiei cinetice între moleculele substanței și prin agitația electronică. Conducția apare când există o diferență de temperatură între punctele unui material sau când două materiale la temperaturi diferite sunt puse în contact. Transferul de căldură prin conducție se poate imagina ca interschimbarea energiei cinetice între vibrarea rapidă a moleculelor unei substanțe și mai puțin rapidă vibrație a moleculelor vecine. Este bine cunoscut faptul că bunii conducători de electricitate sunt și buni conducători de căldură. [Krauss1987]

Este de reținut faptul că în cele două mecanisme descrise pentru a explica transferul de căldură prin conducție, transferul de energie apare ca fiind un fenomen microscopic. Aceasta spre deosebire de transferul de căldură prin convecție, în care efectele macroscopice joacă un rol important.

În studiul conducției căldurii se face distincție între fluxul de căldură uniform, neuniform, regulat și neregulat (tranzitoriu). Fluxul de căldură în stare regulată există când temperatura fiecărui punct din calea fluxului de căldură nu se modifică cu timpul. În schimb, fluxul de căldură tranzitoriu apare când temperatura oricărui punct din fluxul de căldură se modifică cu timpul. În continuare se va discuta despre fluxul de căldură regulat.

### Coefficientul de conducție termică

Propagarea căldurii prin conducție este un fenomen difuziv asigurat prin mecanisme complexe: contribuția electronilor și a ionilor, transferul de energie prin fotoni, prin câmpul

magnetic, etc. Electronii se deplasează din zonele cu temperaturi ridicate spre zonele cu temperaturi scăzute transferând cu ei energie termică. Ionii corpului efectuează mișcări în timpul cărora au loc ciocniri cu transfer de energie de la unul la altul pe baza legilor de conservare. În acest fel energia este transferată în timpul ciocnirilor din zonele cu temperaturi mai mari în zonele cu temperaturi mai mici, transfer efectuat din aproape în aproape, difuziv. Contribuția ionilor la transferul termic este modelată prin introducerea unui gaz imaginar imponderabil numit gaz fononic care străbate corpurile de la temperaturi ridicate la temperaturi scăzute transferând energie fără să modifice proprietățile corpului. Astfel coeficientul de conducție termică are mai multe componente [Krauss1987]

$$k = k_{el} + k_{fon} + k_{fot} + k_m + \dots \quad (2.6)$$

care marchează contribuția electronilor, ionilor, fotonilor, a câmpului magnetic etc. la fenomenele de propagare a căldurii. Participările însă sunt procentual diferite, după natura corpului.

La metale, contribuția electronilor este fundamentală, putându-se aproxima astfel

$$k \approx k_{el}$$

și enunță o strictă dependență între conducția termică și conducția electrică. Această dependență este pusă în evidență de către legea lui Wiedemann - Franz - Lorenz

$$\frac{k_{el}}{\sigma} = const \cdot T \quad (2.7)$$

$\sigma$  fiind coeficientul de conducție electrică și  $T$  temperatura absolută. Corpurile bune conducătoare de căldură sunt deci în același timp și bune conducătoare de electricitate.

La nemetale contribuția ionilor este covârșitoare

$$k \approx k_{fon}$$

iar dependența de temperatură absolută este exprimată prin Legea lui Eucken

$$k_{fon} \cdot T = k_0 \cdot T_0 = const. \quad (2.8)$$

Valorile pe care le ia coeficientul de conducție termică depind de foarte mulți factori, valorile mari caracterizând corpurile bune conducătoare de căldură, iar valorile mici caracterizând corpurile izolatoare din punct de vedere termic.

Temperatura influențează în mod hotărâtor valoarea coeficientului de conducție termică al solidelor. Legile de variație sunt liniare, parabolice

$$k = a \pm bT$$

$$k = a \pm bT \pm cT^2$$

și chiar exponențiale în cazul grafitului:

$$k = 228.26 \cdot 10^{-\frac{T}{2000}}$$

Aceste funcții au fost precizate experimental, iar în unele cazuri dependența lui  $k$  de temperatură este prezentată sub formă tabelară. În cazul unor materiale (argint), simultan cu creșterea temperaturii,  $k$  scade; iar în cazul altor materiale (aluminiu)  $k$  crește cu temperatura; iar la altele (fontă) se semnalează ambele sensuri de variație.

Umiditatea materialului are influență asupra lui  $k$ , acesta crescând pe măsură ce umiditatea crește. Materialele izolatoare termic își pierd proprietățile izolante în momentul în care sunt îmbibate cu apă.

Porozitatea materialelor este un factor de care coeficientul de conducție depinde sensibil. Materialele care au pori de mici dimensiuni și care nu comunică între ei, iar porii conțin aer, sunt materiale izolatoare termic și au  $k$  mic. Pe măsură ce dimensiunile porilor cresc iar aceștia comunică între ei, apar fenomene convective care conduc la mărirea

coeficientului de conducție termică. În multe cazuri această dependență este prezentată sub forma unei relații de calcul pentru  $k$  în funcție de densitatea aparentă a materialului.

Presiunea la care sunt supuse unele materiale poroase și fibroase influențează valoarea coeficientului de conducție termică.

### Ecuția generală a conducției termice

Viteza fluxului de căldură prin conducție printr-un material este proporțională cu aria secțiunii normale pe direcția fluxului de căldură și cu gradientul de temperatură de-a lungul căii fluxului de căldură. Aceasta este reprezentată matematic astfel [Lienhard2004]:

$$q \approx -A \frac{dT}{dx}$$

unde semnul minus este folosit pentru a permite un flux de căldură pozitiv în prezența unui gradient de temperatură negativ (legea a doua a termodinamicii spune că, căldura nu poate curge de la o regiune cu temperatura mai mică la o regiune cu temperatura mai mare, fără intervenție din exterior). Prin adăugarea unei constante de proporționalitate obținem legea lui Fourier:

$$q = -kA \frac{dT}{dx} \quad (2.9)$$

care servește la a defini constanta de proporționalitate  $k$ :

$$k \equiv \frac{q}{A(dT / dx)}$$

$k$  este o constantă de material, și este numită conductivitatea termică.

Folosind legea lui Fourier se poate obține ecuația generală a conducției căldurii. Vom lua în considerare volumul infinitesimal din figura 2.44. Balanța energetică este scrisă egalizând viteza fluxului de căldură ce intră în respectivul subvolum plus viteza generării căldurii în interiorul subvolumului, cu viteza fluxului de căldură ce părăsește subvolumul plus viteza de schimbare a energiei interne (căldura înmagazinată) în subvolum [Krauss1987, Remsburg2001].

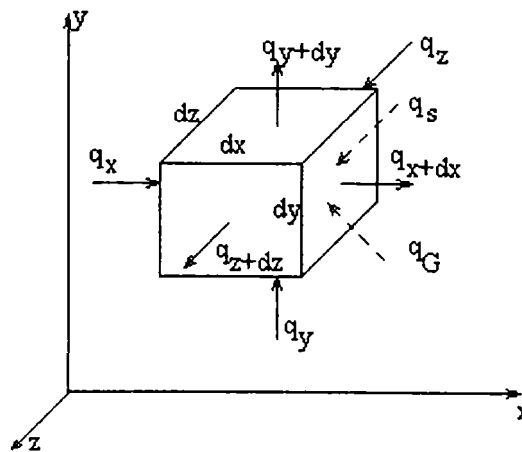


Figura 2.44 – Transferul de căldură în unitatea de volum [Lienhard2004, Krauss1987]

Astfel se poate scrie:

$$q_{in} + q_G = q_{out} + q_s \quad (2.10)$$

sau

$$q_{in} - q_{out} + q_G = q_s$$

în care  $q_{in}$  este fluxul de căldură ce intră în subvolum,  $q_{out}$  fluxul de căldură care iese din subvolum, iar  $q_G$  și respectiv  $q_s$  sunt căldura generată în interiorul volumului și căldura stocată în interiorul volumului.

Considerăm mai întâi direcția axei Ox. Căldura va intra prin fața din stânga în concordanță cu ecuația 2.9.

$$q_x = -kA \frac{\partial T}{\partial x} = \left(-k \frac{\partial T}{\partial x}\right) dy dz$$

unde gradientul de temperatură este exprimat ca derivată parțială, deoarece temperatura  $T$  este funcție de  $y$  și  $z$  și de timpul  $t$ . Căldura va părăsi volumul prin fața dreaptă la  $x+dx$ ,

$$q_{x+dx} = \left[ \left(-k \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial x} \left(-k \frac{\partial T}{\partial x}\right) dx \right] dy dz$$

astfel încât diferența dintre căldura ce intră prin planul  $x$  și căldura ce iese prin planul  $x+dx$  va fi:

$$q_x - q_{x+dx} = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) dx dy dz$$

La fel se calculează diferențele de căldură pe direcțiile Oy și Oz și se obține:

$$q_y - q_{y+dy} = \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) dx dy dz$$

și

$$q_z - q_{z+dz} = \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) dx dy dz$$

Astfel putem calcula  $q_{in} - q_{out}$ , iar apoi se înlocuiește în ecuația 2.5.

$$q_{in} - q_{out} = \left[ \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) \right] dx dy dz$$

Căldura generată în interiorul subvolumului poate proveni dintr-o disipare electrică sau electronică, cum ar fi o căldură generată de forma  $I^2 R$  sau dintr-o reacție chimică. Dacă  $q_i$  este viteza de generare a căldurii pe unitatea de volum, atunci:

$$q_G = q_i (dx dy dz)$$

Căldura acumulată în interiorul subvolumului este direct proporțională cu viteza de modificare a energiei interne,

$$\frac{du}{dt} = c_v dm \frac{dT}{dt}$$

unde  $u$  este energia internă, iar  $c_v$  este căldura specifică la volum constant. Dacă  $c_v$  este înlocuit cu  $c$  (deoarece pentru un solid  $c_v = c_p = c$ ) și  $dm = \rho dx dy dz$ , unde  $\rho$  este densitatea materialului, atunci vom avea:

$$q_s = \frac{du}{dt} = \rho c \frac{\partial T}{\partial t} dx dy dz$$

Înlocuind toate valorile obținute în ecuația 2.5. vom avea:

$$\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) + q_i = \rho c \frac{\partial T}{\partial t}$$

Dacă se presupune că  $k$ ,  $c$ , și  $\rho$  sunt independente de temperatură, direcție și timp, obținem ecuația generală a conducției căldurii:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} + \frac{q_i}{k} = \frac{1}{\alpha} \frac{\partial T}{\partial t} \quad (2.11)$$

unde  $\alpha$  este cunoscut ca fiind dispersia termică a materialului:

$$\alpha \equiv \frac{k}{c\rho}$$

Dacă sistemul nu conține surse de căldură ecuația 2.11. devine *ecuația Fourier*,

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} = \frac{1}{\alpha} \frac{\partial T}{\partial t}$$

Dacă considerăm starea staționară, în care distribuția de temperatură nu variază cu timpul, obținem ecuația Poisson:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} + \frac{q_i}{k} = 0$$

În final, în absența surselor de căldură și considerând starea staționară obținem:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} = 0$$

care este numită ecuația Laplace, ce este frecvent notată:

$$\nabla^2 T = 0$$

### 2.3.3.2 Convecția termică

Prin convecție termică se înțelege transmiterea căldurii între un fluid în mișcare și un perete care delimitează mișcarea fluidului [Krauss1987]. Pentru producerea schimbului de căldură este necesară existența unei diferențe de temperatură între fluidul în mișcare și perete. Se va nota în cele ce urmează cu  $t_f$  temperatura fluidului și cu  $t_p$  temperatura peretelui.

Când mișcarea fluidului se produce liber, ca urmare a diferențelor din masa fluidului, se vorbește de convecție liberă, iar când mișcarea fluidului se produce forțat datorită diferențelor de presiune create în mod artificial, se vorbește de convecție forțată. La convecția liberă curentul predominant de fluid se produce pe direcția verticală, pe când la convecția forțată poate avea orice direcție.

Indiferent de natura mișcării fluidului se deosebesc două tipuri de curgere. Curgerea poate să fie laminară când straturile de fluid în mișcare nu se amestecă între ele, sau turbulentă, când se amestecă. La curgerea laminară spre deosebire de cea turbulentă viteza de curgere nu are componente perpendiculare pe direcția de curgere.

Indiferent de natura mișcării fluidului și de caracterul curgerii, datorită forțelor de atracție moleculară, la perete se formează un strat aderent de fluid care se numește strat limită. În acest strat limită curgerea este întotdeauna laminară, iar viteza limită variază liniar, de la zero la perete până la o viteză oarecare. Prin stratul limită transmiterea căldurii nu se poate realiza decât prin conducție. O importanță deosebită în convecția termică o are deci coeficientul de conducție termică al fluidului  $\lambda_f$  și grosimea stratului limită  $\delta$ . Grosimea stratului limită depinde de proprietățile fizice ale fluidului, deci de natura fluidului, dar în măsură mult mai mare depinde de caracterul curgerii. Stratul limită este mult mai gros la curgerea laminară decât la cea turbulentă, iar la curgerea turbulentă se subțiază cu atât mai



mult cu cât turbulența este mai mare. Este de așteptat ca la curgerea turbulentă schimbul de căldură între fluid și perete să fie mult mai intens decât la curgerea laminară.

Pentru calculul căldurii schimbate prin convecție  $Q(J)$  între un perete cu aria  $A (m^2)$  având temperatura  $t_p (^\circ C)$  și un fluid cu temperatura (neinfluențată de temperatura peretelui)  $t_f (^\circ C)$  în timpul  $\tau(s)$  se utilizează formula lui Newton:

$$Q = \alpha A(t_p - t_f)\tau \quad (2.12)$$

unde:  $\alpha (W/m^2k)$  reprezintă coeficientul de transmitere a căldurii prin convecție, sau pe scurt, coeficientul de convecție.

Formula anterioară este foarte simplă ca formă. În realitate însă, coeficientul de convecție  $\alpha$  este o funcție complicată și dificil de determinat, de multe mărimi fizice ale fluidului, cum ar fi: viteza, densitatea, temperatura, presiunea, căldura specifică, coeficientul de conductivitate termică, vâscozitatea, etc. Toate aceste mărimi variază în lungul curentului, datorită schimbului finit de căldură, care duce în special la modificarea temperaturii atât în fluid cât și în stratul limită. În cazul unui regim nestaționar de schimb de căldură aceste mărimi variază și în timp. În aceste condiții foarte dificile se pune problema determinării coeficientului de convecție. Pentru aceasta trebuie să se facă apel la ecuațiile diferențiale care descriu fenomenul de convecție termică.

Transmiterea căldurii prin convecție depinde de mișcarea fluidului, de modul cum se transmite căldura prin conductivitate prin straturile de fluid și de condițiile de contur. Conductivitatea prin straturile de fluid este descrisă de ecuația lui Fourier generalizată pentru fluidele în mișcare. Ecuația de contur ia în considerare modul în care se produce transferul de căldură între fluid și perete prin stratul limită.

### Ecuația de continuitate a convecției termice

Se consideră un volum elementar  $dV$  prin care curge tridimensional un fluid cu densitatea  $\rho$  având la intrare în element vitezele  $w_x, w_y, w_z$ .

Masele de fluid care intră în element după cele trei axe de coordonate sunt [Lienhard2004]:

$$dm_x = \rho \cdot w_x (dy \cdot dz) d\tau$$

$$dm_y = \rho \cdot w_y (dx \cdot dz) d\tau$$

$$dm_z = \rho \cdot w_z (dx \cdot dy) d\tau$$

Masele de fluid care părăsesc elementul de volum dacă densitatea  $\rho$  este constantă, sunt:

$$dm_{x+dx} = \rho \left( w_x + \frac{\partial w_x}{\partial x} \cdot dx \right) dy dz d\tau$$

$$dm_{y+dy} = \rho \left( w_y + \frac{\partial w_y}{\partial y} \cdot dy \right) dx dz d\tau$$

$$dm_{z+dz} = \rho \left( w_z + \frac{\partial w_z}{\partial z} \cdot dz \right) dx dy d\tau$$

Variațiile masei după cele trei direcții sunt :

$$dm_{x+dx} - dm_x = \rho \frac{\partial w_x}{\partial x} \cdot dx dy dz d\tau$$

$$dm_{y+dy} - dm_y = \rho \frac{\partial w_y}{\partial y} \cdot dx dy dz d\tau$$

$$dm_{z+dz} - dm_z = \rho \frac{\partial w_z}{\partial z} \cdot dx dy dz d\tau$$

Pe baza legii conservării masei suma variațiilor trebuie să fie nulă. Rezultă :

$$\rho \left( \frac{\partial w_x}{\partial x} + \frac{\partial w_y}{\partial y} + \frac{\partial w_z}{\partial z} \right) dV d\tau = 0$$

care reprezintă ecuația de continuitate. Pentru densitate constantă ecuația anterioară devine :

$$\frac{\partial w_x}{\partial x} + \frac{\partial w_y}{\partial y} + \frac{\partial w_z}{\partial z} = 0$$

În cazul în care densitatea fluidului nu este constantă relațiile variațiilor masei devin :

$$dm_{x+dx} - dm_x = \frac{\partial(\rho \cdot w_x)}{\partial x} \cdot dV d\tau$$

$$dm_{y+dy} - dm_y = \frac{\partial(\rho \cdot w_y)}{\partial y} \cdot dV d\tau$$

$$dm_{z+dz} - dm_z = \frac{\partial(\rho \cdot w_z)}{\partial z} \cdot dV d\tau$$

Din cauza variației densității fluidului în timp se produce o variație a masei volumului elementar.

$$dm_0 = \frac{\partial \rho}{\partial \tau} dV d\tau$$

Din legea conservării masei va rezulta că:

$$\frac{\partial(\rho \cdot w_x)}{\partial x} + \frac{\partial(\rho \cdot w_y)}{\partial y} + \frac{\partial(\rho \cdot w_z)}{\partial z} = - \frac{\partial \rho}{\partial \tau}$$

Efectuând derivatele din relația de mai sus rezultă :

$$\rho \left( \frac{\partial w_x}{\partial x} + \frac{\partial w_y}{\partial y} + \frac{\partial w_z}{\partial z} \right) + w_x \cdot \frac{\partial \rho}{\partial x} + w_y \cdot \frac{\partial \rho}{\partial y} + w_z \cdot \frac{\partial \rho}{\partial z} = - \frac{\partial \rho}{\partial \tau}$$

Termenii de forma  $w_x \cdot \frac{\partial \rho}{\partial x}$  reprezintă evident modificarea densității fluidului pe

drumul  $dx = w_x \cdot d\tau$ , provocată de fenomenul de convecție, pe când termenul  $\frac{\partial \rho}{\partial \tau}$  reprezintă

modificarea locală a densității. Modificarea în timp sau în spațiu a densității fluidului se poate produce ca urmare a modificării presiunii sau temperaturii. Suma celor două modificări ale densității reprezintă variația totală a densității numită și diferențiala substanțială a densității, care se scrie sub forma:

$$\frac{D\rho}{d\tau} = \frac{\partial \rho}{\partial \tau} + w_x \cdot \frac{\partial \rho}{\partial x} + w_y \cdot \frac{\partial \rho}{\partial y} + w_z \cdot \frac{\partial \rho}{\partial z} \quad (2.13)$$

## 2.3.4 Modelarea disipării căldurii

Există trei niveluri de transmitere a căldurii pornind de la joncțiune către mediu: la nivel de componentă, la nivel de placă și la nivel de sistem. Acestea sunt toate de mare importanță atât pentru analist cât și pentru proiectant. Rezistența termică la nivel de componentă (circuit integrat) este rezistența internă a entității microelectronice care există între joncțiune și suprafața exterioară a circuitului. Rezistența termică la nivel de placă (PCB) constă în rezistența întâmpinată de fluxul de căldură de la suprafața circuitului la un punct de referință pentru întregul sistem. Acest punct de referință poate fi temperatura aerului ce înconjoară componenta, marginea cablajului imprimat în care diferitele componente sunt fixate sau chiar un element de răcire care de obicei este prevăzut la unele circuite. Rezistența termică a sistemului se referă la rezistența întâmpinată de fluxul de căldură de la punctul de referință de pe placă la mediul înconjurător.

Pentru modelarea, simularea și estimarea transferului de căldură la nivel de componentă, placă și sistem se utilizează metode numerice:

- metoda diferențelor finite (Finite Differences Method – FDM);
- metoda elementelor finite (Finite Elements Method – FEM);
- metoda elementelor marginale (Boundary Elements Method – BEM);

Toate aceste metode pot rezolva cu o acuratețe foarte bună ecuațiile de tip Laplace/Poisson, care stau la baza disipării căldurii. În fiecare din cele trei metode, pentru a se obține soluțiile cu o anumită precizie sunt necesare resurse computaționale și de memorare foarte mari, astfel că de cele mai multe ori se folosesc doar în probleme locale.

### 2.3.4.1 Modelul electrotermic

O metodă mai simplă se bazează pe similitudinea electrotermică [Krauss1987]. Pentru o placă plană ale cărei fețe sunt menținute la  $T_1$  și  $T_2$  ca în figura 2.45, se folosește ecuația 2.11, doar pentru coordonata  $x$ :

$$\frac{d^2 T}{dx^2} = 0$$

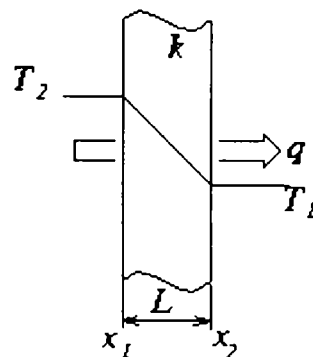


Figura 2.45 – Transferul de căldură printr-o suprafață plană [Remsburg2001, Krauss1987]

Integrând de două ori această ecuație și folosind condițiile la limită se obține soluția:

$$T = T_1 - \frac{x}{L}(T_1 - T_2)$$

unde  $L = x_2 - x_1$ .

Din ecuația de mai sus se obține distribuția temperaturilor în plan iar căldura poate fi determinată folosind ecuația 2.10.

$$q = -kA \frac{dT}{dx}$$

sau

$$q = -kA \left[ \frac{-(T_1 - T_2)}{L} \right] = \frac{kA}{L} (T_1 - T_2)$$

Forma rezultatului de mai sus indică analogia directă dintre transferul căldurii și curentul electric, guvernate de legea lui Ohm.

$$I = \frac{V}{R}$$

În această analogie, entitățile similare sunt:

Curentul  $I$  - fluxul de căldură  $q$

Potențialul  $V$  - diferența de temperatură  $\Delta T$

Rezistența  $R$  - rezistența termică  $R$

Este ușor de văzut că, pentru un plan subțire, rezistența termică este:

$$R = \frac{\Delta T}{q} = \frac{L}{kA} \quad (2.14)$$

Modelul electrotermic analog planului prezentat în secțiunea precedentă este arătat în figura de mai jos:

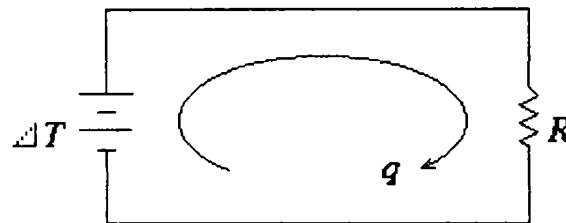


Figura 2.46 – Modelul electrotermic [Lienhard2004]

Conducția este adesea însoțită de alte modele de transfer al căldurii. Un caz important este cel al apariției simultane a conducției și convecției. Fiecare mod de transfer are un efect asupra transferului total de căldură, și este important a obține o singură expresie care încorporează ambele modele.

Convecția este direct proporțională cu aria suprafeței normale pe direcția fluxului de căldură și cu diferența de temperatură dintre fluidul convectiv și suprafața de limită:

$$q \approx S\Delta T$$

Constanta de proporționalitate este  $h$ , coeficientul transferului de căldură,

$$q = hS\Delta T$$

$$h \equiv \frac{q}{S\Delta T} \quad (2.15)$$

Figura 2.47 prezintă un perete plan compus din două plăci,  $T_m$  fiind temperatura interfeței. Acest perete este supus unui fluid cald în stânga și unul rece la dreapta, iar coeficienții de transfer sunt  $h_h$  și  $h_c$ . Căldura curge de la fluidul cald spre cel rece prin seria de plăci.

$$q = h_h S (T_h - T_1) = \frac{k_1 A}{L_1} (T_1 - T_m) = \frac{k_2 A}{L_2} (T_m - T_2) = h_c S (T_2 - T_c)$$

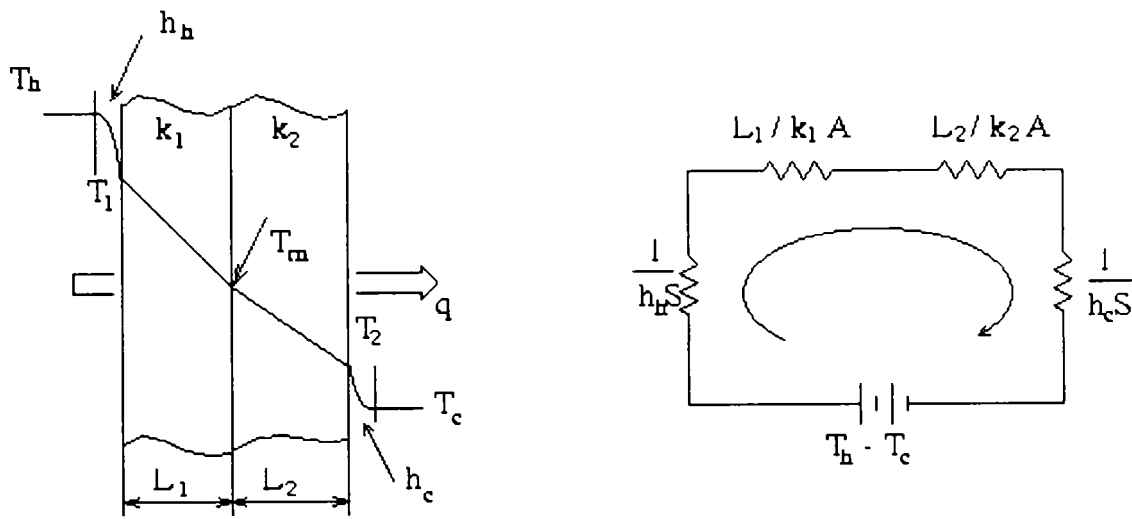


Figura 2.47

De aici rezultă diferențele de temperatură:

$$T_h - T_1 = \frac{q}{h_h S}$$

$$T_1 - T_m = \frac{q L_1}{k_1 A}$$

$$T_m - T_2 = \frac{q L_2}{k_2 A}$$

$$T_2 - T_c = \frac{q}{h_c S}$$

care adunate dau:

$$T_h - T_c = \frac{q}{A} \left( \frac{1}{h_h} + \frac{L_1}{k_1} + \frac{L_2}{k_2} + \frac{1}{h_c} \right)$$

Putem deduce rezistența termică globală:

$$R_T = \sum R = \frac{T_h - T_c}{q} = \frac{1}{h_h A} + \frac{L_1}{k_1 A} + \frac{L_2}{k_2 A} + \frac{1}{h_c A}$$

Inversa rezistenței este conductanța termică:

$$\theta_T = \frac{1}{R_T}$$

### 2.3.5 Modelarea disipării căldurii în circuitele integrate

Îmbunătățirea performanțelor dispozitivelor semiconductoare a dus la creșterea puterii disipate de dispozitive, în timp ce creșterea temperaturilor pe joncțiune nu este permisă și trebuie limitată la o anumită valoare maximă. Valorile maxime admisibile pentru temperatura la nivel de joncțiune sunt date de catalog pentru fiecare circuit sau familie de circuite integrate. Aceste valori trebuie să poată fi utilizate și la compararea performanțelor relative ale soluțiilor de împachetare utilizate de diferiți producători. Compararea se poate face doar dacă se folosește același standard pentru efectuarea testelor și măsurarea valorilor de temperatură. Din punct de vedere istoric, măsurătorile termorezistente au fost făcute într-o varietate de moduri, dar s-au impus două standarde industriale acceptate: JEDEC (Joint

Electron Device Engineering Council) și SEMI (Semiconductor Equipment and Materials International). La momentul actual se utilizează metoda JEDEC 51 (JESD 51) pentru determinarea rezistenței termice a dispozitivelor semiconductoare.

Standardul JEDEC este dezvoltat pentru a crea o metodă uniformă de caracterizare a metodelor de împachetare a circuitelor integrate pe baza căreia să se poată evalua și compara performanțele diferitelor metode de împachetare pentru dispozitive similare sau dispozitive similare împachetate la fel [TI1999, Colquitt2000, JESD51-4, Sofia1995].

Chiar și la nivelurile de tensiune de alimentare joase, creșterea continuă a frecvențelor de lucru și a numărului de porți au ca rezultat disiparea de căldură parazită, acestea putând avea valori incredibile pentru tehnologia anilor trecuți. Unele procesoare bazate pe tehnologie de vârf (avansat submicronică) trebuie să disipe căldură echivalentă cu cea a unui bec incandescent de cameră. Aceasta oferă provocări semnificative proiectanților de sisteme, în special datorită faptului că temperaturile permise la nivelul joncțiunilor au rămas neschimbate. Convergent, în unele tehnologii, temperatura joncțiunilor trebuie menținută mai jos decât cea permisă înainte datorită siguranței pe care trebuie să o prezinte în cadrul noilor sisteme metalice și a geometriilor tot mai mici.

În trecut, un parametru folosit de proiectanții de sisteme pentru a compara dispozitivele și a determina temperaturile funcționale ale joncțiunilor sunt caracteristicile publicate de către producătorul dispozitivului, care specifică capacitatea ansamblului de dispozitive de a disipa căldura generată de joncțiuni în exteriorul suprafeței de siliciu. De altfel, destul de simplu ca și concept, aceste valori ale rezistențelor termice sunt uneori greșit înțelese și folosite, în parte datorită faptului că producătorii publică foarte rar modul de măsurare al acestor valori și ce reprezintă ele fizic [TI1999].

Modelarea fluxurilor de căldură în circuitele integrate este un proces dificil în primul rând datorită complexității fizice a acestora: pastila de siliciu în care sunt plasate tranzistoarele, traseele de metal, straturile de oxid, pinii circuitului, capsula circuitului, elementele de interfață și răcire. Există zeci de studii ale transferului de căldură pentru fiecare din elementele componente ale unui circuit ceea ce denotă din nou complexitatea și importanța acestui subiect [Malhammar2002\_4].

Căile de disipare a căldurii într-un circuit integrat sunt prezentate schematizat în figura 2.48. Mecanismele de transfer de căldură într-un circuit integrat sunt conducția în interiorul circuitului și convecția, radiația și conducția în exteriorul circuitului.

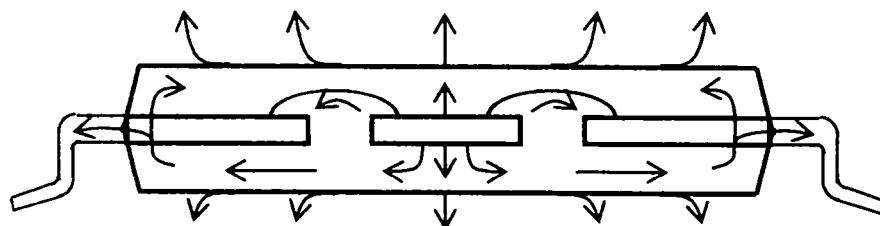


Figura 2.48 – Căile de disipare a căldurii într-un circuit integrat

Figura 2.49 arată în mod simplificat diagrama bidimensională a rezistenței termice, pornind de la joncțiune până la suprafața dispozitivului.

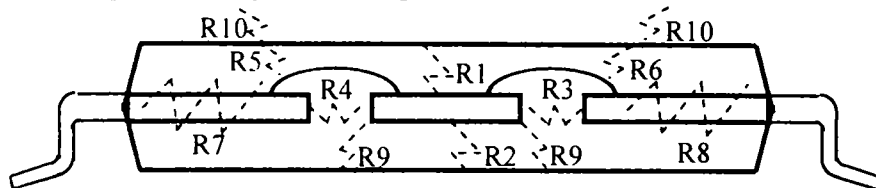


Figura 2.49 – Reprezentarea distribuției rezistenței termice la disiparea căldurii într-un circuit integrat

$R_1, R_2, \dots, R_{10}$  sunt echivalentul rezistențelor termice de la suprafața de siliciu la o secțiune a modulului. Deoarece rezistența termică în solid depinde de proprietățile materialului, lungimea și secțiunea căii de disipare a căldurii, fiecare secțiune a modulului contribuie cu o rezistență termică unică. Puterea termică disipată de la siliciu la suprafața circuitului poate fi calculată prin formula:

$$P = (T_j - T_{s1})/R_1 + (T_j - T_{s7})/(R_3 + R_7) + \dots + (T_j - T_{sn})/R_n$$

Unde:

$T_{s1} - T_{sn}$  – temperatura la suprafața modulului în locațiile exterioare indicate;

$T_j$  – temperatura medie a joncțiunii la suprafața pastilei de siliciu;

$R_1 - R_n$  – rezistențele termice distribuite pentru fiecare secțiune a modulului.

Cu cât rezistența termică a modulului se micșorează, cu atât modulul va conduce mai bine căldura în afară. Dacă abilitatea modulului de a conduce căldura din interior la suprafața exterioară ar fi tot ce e necesar pentru a putea prezice temperaturile joncțiunilor, producătorii dispozitivelor ar putea să furnizeze valori exacte ale rezistențelor termice folosind analiza elementelor finite, sau modele discrete echivalente. Din nefericire, factorii externi cum ar fi curenții de aer și spațiile dintre cablaj (PCB) pot afecta rezistența termică a modulului mult mai mult decât construcția modelului.

Pentru a simplifica lucrurile, modelul unui circuit integrat e prezentat în figura 2.50:

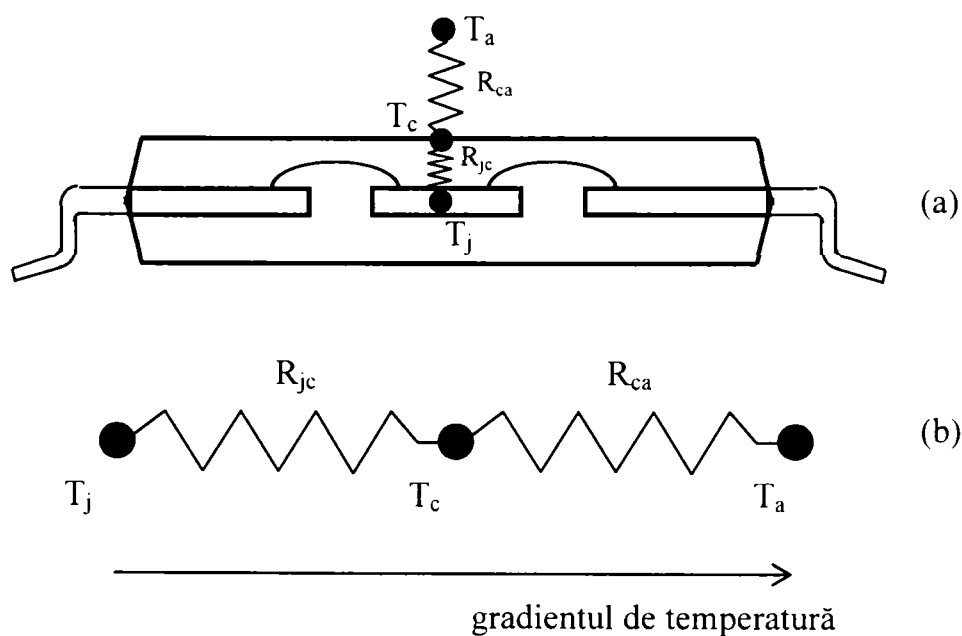


Figura 2.50 – Caracteristica termică simplificată [JESD51-4]

unde:

$T_j$  – temperatura la nivelul joncțiunii;

$T_c$  – temperatura materialului în care e împachetată pastila de siliciu (capsula circuitului);

$T_a$  – temperatura mediului ambiant;

$$T_j > T_c > T_a$$

Temperatura joncțiunii se poate estima pe baza relației 2.16. Dacă circuitul este prevăzut și cu elemente de răcire atunci  $R_{ca}$  este compusă din rezistența termică a elementului de răcire și rezistența termică de interfață între capsula circuitului și elementul de răcire.

$$T_j = T_a + P(R_{jc} + R_{ca}) = T_a + P(R_{jc} + R_{cs} + R_{sa}) \quad (2.16)$$

$R_{jc}$  – rezistența termică între pastila de siliciu și circuit;

$R_{ca}$  – rezistența termică între capsula circuitului și aerul înconjurător;

$R_{cs}$  – rezistența termică între capsula circuitului și elementul de răcire (dacă există);

$R_{sa}$  – rezistența termică între răcitor și aerul înconjurător.

Pe măsură ce puterea disipată de un circuit integrat crește trebuie menținută temperatura la nivelul joncțiunii constantă. Acest lucru se poate realiza doar dacă se intervine la nivelul rezistențelor termice ale capsulei circuitului și a mediului ambiant. Există două direcții principale în care se fac la ora actuală eforturi: găsirea de noi materiale și metode de împachetare a pastilei de siliciu și în al doilea rând găsirea unor noi metode de răcire ieftine și eficiente. Prima direcție are ca scop scăderea rezistenței termice a circuitului și chiar introducerea răcirii direct la nivelul circuitului integrat, iar a doua direcție își propune scăderea rezistenței termice dintre circuitul integrat și mediul ambiant prin mărirea suprafeței de disipare a căldurii și ventilație.

Valoarea maximă pentru temperatura joncțiunii este de 150 °C sau 175 °C [Electus2001, TI1999]. Rezistența termică întâmpinată de fluxul de căldură în capsula circuitului ( $R_{jc}$ ) este relativ mică și depinde de materialul din care este realizat: uzual valorile sunt între 0.7 și 4.5 °C/W [Electus2001]. Dacă circuitul nu este prevăzut cu elemente de răcire, atunci rezistența termică dintre capsulă și mediu ( $R_{ca}$ ) este foarte mare, între 35 și 100°C/W. Pentru a reduce această valoare a rezistenței  $R_{ca}$ , trebuie prevăzute elemente de răcire pasive sau active. În prezența elementelor de răcire și a unei paste termo-conductoare între circuit și răcitor,  $R_{ca}$  poate ajunge sub 1°C/W [Electus2001].

La un microprocesor ce disipă o putere de 100W, pentru ca acesta să poată funcționa într-un mediu cu temperatura de până la 75°C, rezistența termică între joncțiune și mediu,  $R_{ja}$  ar trebui să fie cel mult:

$$R_{ja} = (T_j - T_a) / P = (150^\circ\text{C} - 75^\circ\text{C}) / 100\text{W} = 0.75^\circ\text{C/W}$$

Pentru o capsulă de plastic cu rezistența termică  $R_{ja}$  de 50 °C/W, puterea maximă pe care o poate disipa circuitul încapsulat într-o astfel de capsulă într-un mediu cu temperatura de până la 75°C, este:

$$P = (T_j - T_a) / R_{ja} = (150^\circ\text{C} - 75^\circ\text{C}) / 50^\circ\text{C/W} = 1.5\text{ W}$$

Din exemplele precedente se observă necesitatea proiectării din punct de vedere termic atât a pastilei de siliciu, cât și a capsulei în care este împachetată aceasta. În cazul circuitelor VLSI care disipă o cantitate mare de putere, trebuie prevăzute și elemente de răcire pentru a asigura funcționarea corectă a acestora. Astfel că proiectarea circuitelor integrate trebuie să aibă în vedere, pe lângă funcționalitatea propriuzisă și asigurarea condițiilor termice de funcționare a circuitului. Acest lucru presupune alegerea tipului de încapsulare și elementelor de răcire necesare pentru a obține o performanță maximă de la circuit.

Tot din această cauză se studiază și se aplică tot mai mult tehnicile de control al puterii consumate și a temperaturii disipate de sistemele digitale, subiect ce va fi abordat în Capitolul 4.



## 2.4 Simularea electro-termică a circuitelor integrate

Datorită creșterii densității componentelor și a vitezei de operare și a dimensiunilor tot mai reduse de integrare, continuă să crească și puterea disipată și temperatura în circuitele integrate. Diferența de temperatură între centrul circuitului și marginile acestuia poate ajunge la câteva zeci de grade. Datorită problemelor ridicate de creșterea temperaturii, așa cum s-a prezentat și în paragraful precedent, este foarte important procesul de proiectare al circuitelor integrate atât din punct de vedere logic și electric cât și termic. Astfel că verificarea performanțelor și fiabilității circuitelor necesită unelte speciale de simulare înainte de începerea procesului de fabricare. [Miao2000, Milanovic2000, Klaassen1997]

La dispoziția proiectanților stau mai multe aplicații software, comerciale sau gratuite, dedicate pentru simularea logică, electrică sau termică a circuitelor integrate. Aceste aplicații au o anumită maturitate și sunt folosite chiar de producătorii mari de circuite integrate. Una din problemele cu care se confruntă proiectanții este însă, lipsa unei aplicații de simulare care să permită atât verificarea electrică a circuitelor cât și verificarea termică pe aceleași date de descriere. În plus nu există nici o unealtă consacrată pentru simularea cuplată electro-termică. Necesitatea unei astfel de aplicații apare în contextul creșterii puterii disipate de circuitele integrate.

Curenții de scurgere dintr-un circuit aflat în faza statică sunt dependenți exponențial de temperatură. Aplicațiile de simulare electrică (SPICE) iau în calcul temperatura dar doar ca pe un parametru static. În realitate însă, în timpul funcționării circuitului se degajă căldură datorită puterii disipate, astfel că temperatura circuitului crește, fapt ce duce la creșterea curenților de scurgere. Creșterea curenților consumați duce la rândul ei din nou la creșterea temperaturii, iar procesul se repetă până se ajunge la echilibru termic. Aplicațiile de simulare termică (FEMAP, ANSYS, THERMAN, WINTHERM), la rândul lor, iau în calcul, ca parametri de intrare doar puterea disipată de elementele active din circuite, fără a mai exista o cale de reactualizare a puterii după estimarea temperaturii.

O aplicație de simulare electro-termică ar trebui să permită atât simularea din punct de vedere electric al circuitului cât și simularea din punct de vedere termic. O simulare cuplată (simultană) electro-termică ar trebui să actualizeze în fiecare pas de simulare atât mărimile electrice cât și cele termice, iar simularea electrică și termică în pasul curent depinde de mărimile termice și respectiv electrice obținute în pasul precedent, până se ajunge la echilibrul electro-termic.

Încă de la începutul anilor '90 există mai multe propuneri pentru simularea electro-termică a circuitelor integrate însă niciuna din acestea nu s-a impus până în prezent. Dovadă este faptul că încă se mai publică articole pe această temă [Miao2000, Milanovic2000, Korvink2003, Liao2003, Wang2004]. Dificultatea la o simulare electro-termică constă în necesitatea existenței dispunerii în pastila de siliciu a componentelor electronice ce formează circuitul integrat. Această dispunere în siliciu a componentelor circuitului se generează într-o fază ulterioară simulării, și chiar mai mult, ar avea nevoie de rezultatele unei simulări termice preliminare, pentru a putea plasa componentele în funcție de temperatura lor.

Există două categorii de soluții propuse pentru simularea electro-termică, fiecare având avantajele și dezavantajele lor:

- problema termică este mapată într-un circuit electric echivalent care este simulat în același timp cu simularea electrică de către o aplicație de simulare la nivel electric a circuitului integrat;
- folosirea a două simulatoare independente, unul electric și celălalt termic, care sunt executate în paralel, comunică între ele și converg spre soluția finală.

Pentru simularea termică la nivel de microarhitectură, s-a folosit modelul electric echivalent ce a fost prezentat în paragraful precedent. Acest model a fost implementat în

SPICE ca un subcircuit ce are ca și nod de intrare sursa de putere monitorizată, iar ca ieșiri are temperatura medie a joncțiunii și a capsulei. Subcircuitul primește ca parametrii rezistențele termice  $R_{jc}$  și  $R_{ca}$ , capacitățile termice  $C_j$  și  $C_c$  și temperatura mediului ambiant  $T_a$ . Acest circuit estimează media variațiilor de temperatură pentru circuit atunci când acesta este stimulat de o anumită secvență de vectori de intrare. Modelul acesta a fost implementat și în [Veijola1995, Dashney1999] (Figura 2.51).

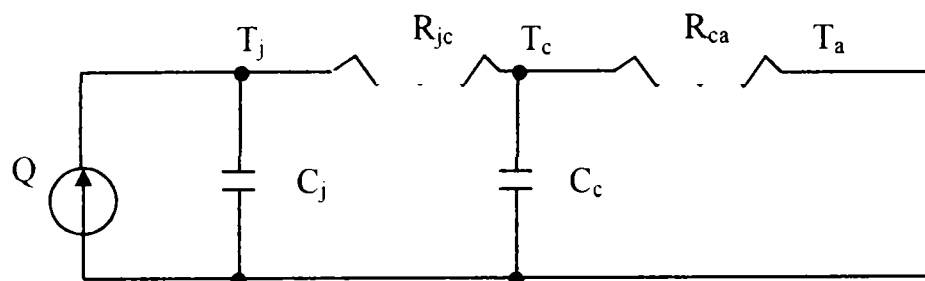


Figura 2.51 – Rețeaua termică ce descrie transferul de căldură de la joncțiune prin capsula la mediul ambiant

Temperatura medie internă a circuitului este:

$$T_j = T_a + P_{med} \cdot R_{ja} \cong T_a + \frac{P_{med}}{10^5 \cdot die\_size}$$

Rezistența termică poate fi estimată cu ajutorul ecuației 2.14, iar capacitatea termică cu formula:

$$C = \rho Vc$$

unde:

- $\rho$  – densitatea materialului;
- $V$  – volumul
- $c$  – căldura specifică.

Dacă considerăm circuitul c432 din circuitele ISCAS85, puterea disipată de acesta la aplicarea secvenței de vectori de test: <000001...1>, <010001...1>, <101011...1> este prezentată în figura 2.52. Variația temperaturii la aplicarea aceleași secvențe de intrare este prezentată în figura 2.53.

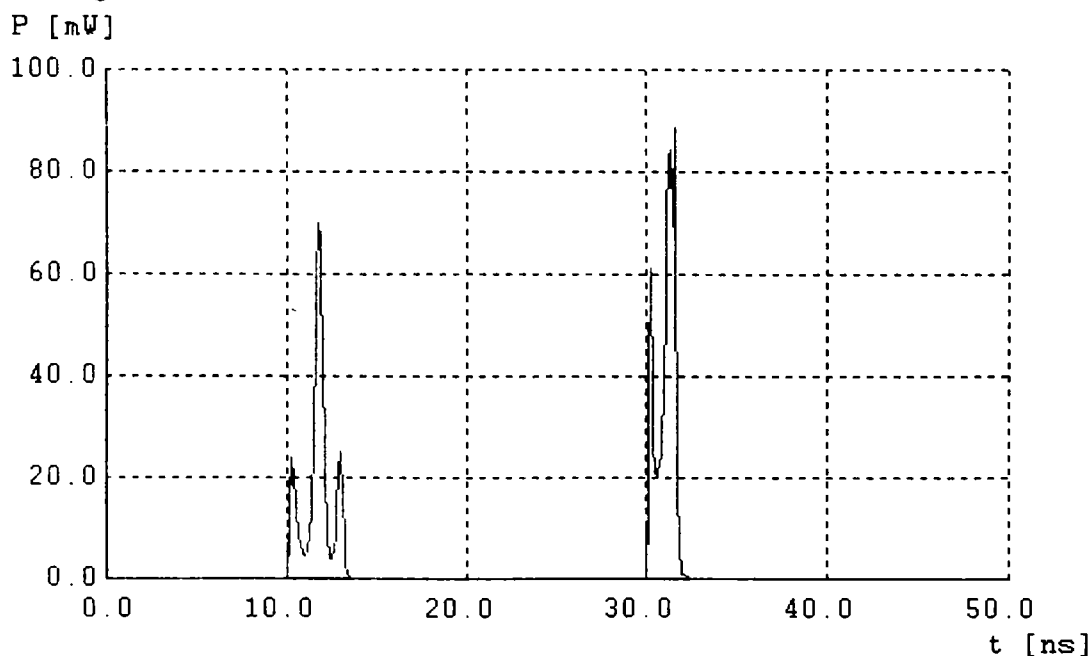


Figura 2.52 – Puterea disipată de c432 fără defecte

După cum se observă în figura 2.53, temperatura la nivelul pastilei de siliciu variază odată cu variația puterii disipate, în schimb temperatura la nivelul capsulei are o latență termică mai mare. Acest lucru a fost prezentat și în [Altet2001].

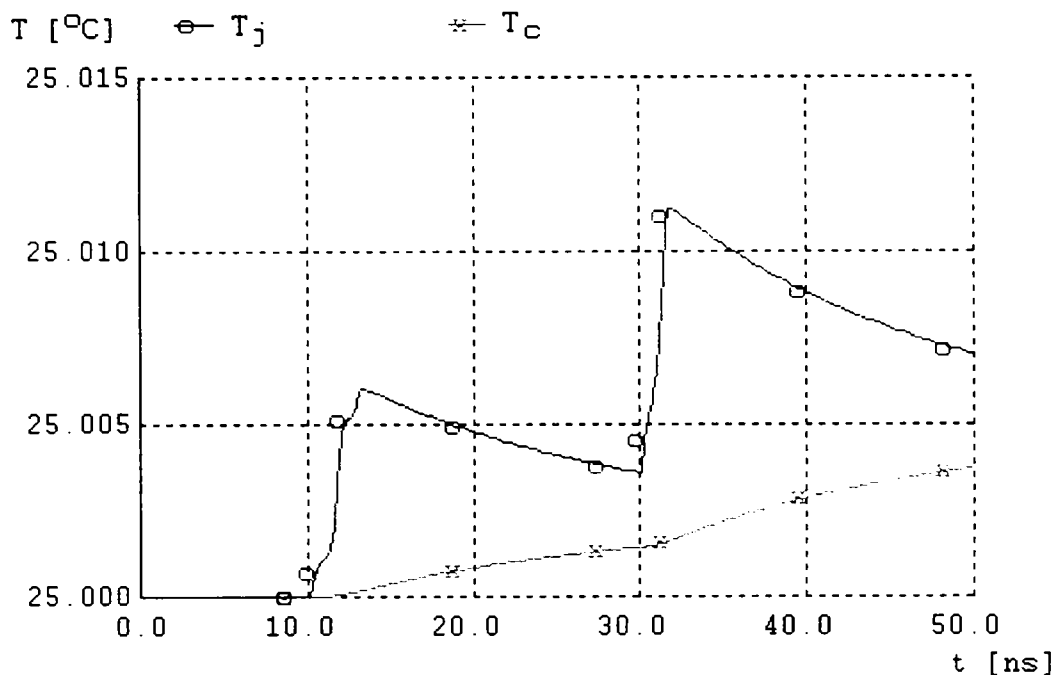


Figura 2.53 – Variația temperaturii pentru c432 fără defecte

În prezența unui defect de scurt-circuit între ieșirile porților 118gat și 119gat, puterea disipată de circuit este prezentată în figura 2.54 iar temperatura în figura 2.55.

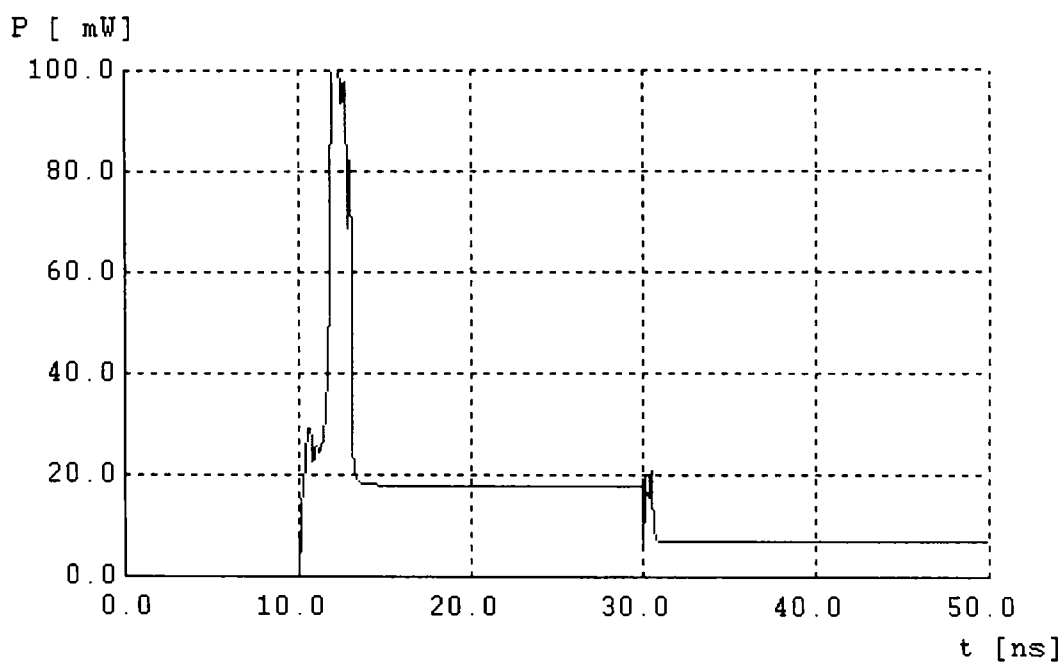


Figura 2.54 – Puterea disipată de circuitul defect

Puterea disipată de circuitul defect la aplicarea vectorului de intrare <010001...1> este cu câteva ordine de mărime mai mare decât în cazul circuitului fără defecte. Temperatura joncțiunii crește însă doar cu 0.03 °C în 20 ns iar după un interval mult mai mare de timp (10 μs) temperatura ajunge la aproximativ 0.9 °C.

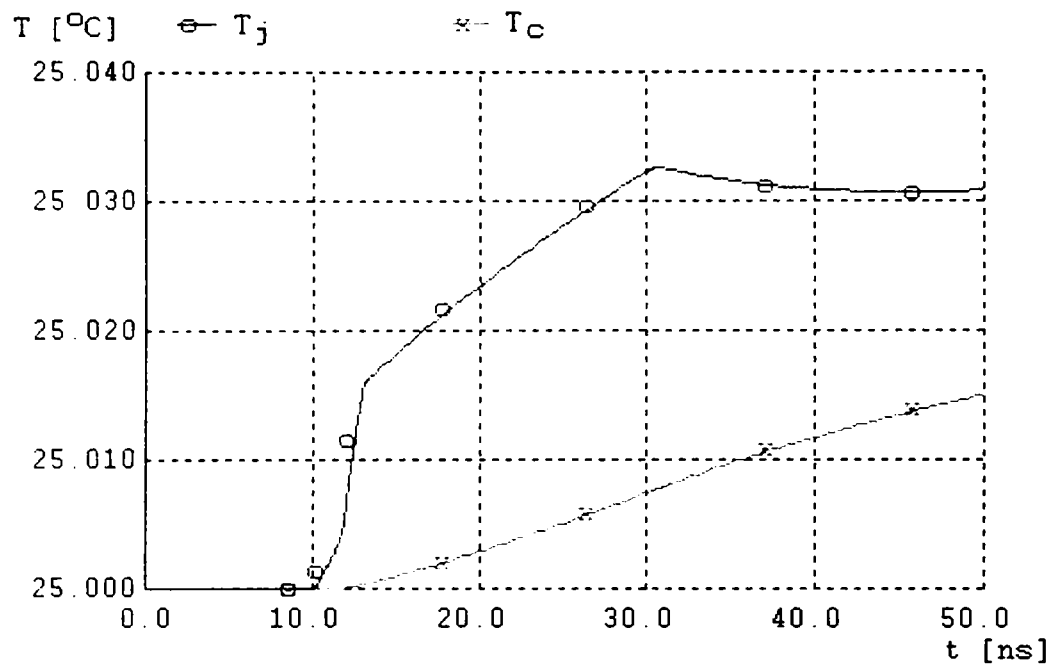


Figura 2.55 – Variația temperaturii pentru circuitul defect

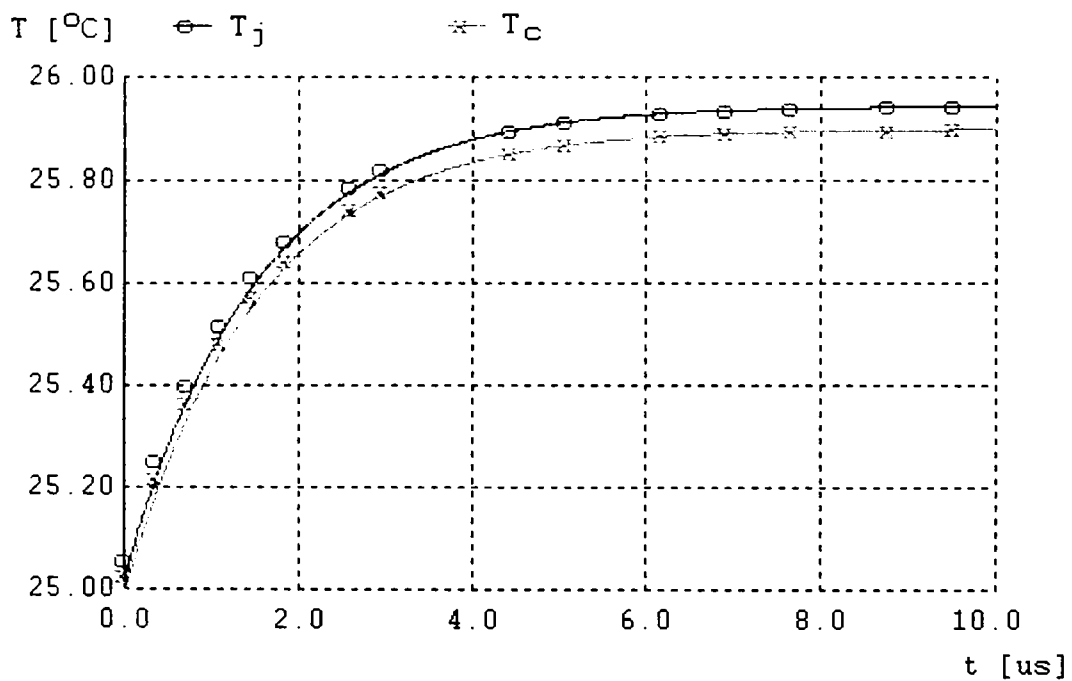


Figura 2.56 – Temperatura la echilibru termic pentru circuitul defect la aplicarea vectorului de intrare <010001...1>

Fluxul de temperatură dispat de circuit este dependent de caracteristicile termice ale capsulei în care este împachetat circuitul și de tipul de elemente de răcire utilizate. Pentru un circuit fără răcire, așa cum s-a considerat în exemplele anterioare, rezistența termică este mare, astfel că circuitul defect se încălzește mai mult astfel că defectul poate fi detectat prin testarea termică. În cazul în care sunt folosite însă elemente de răcire, încălzirea circuitului nu mai este atât de semnificativă, iar defectele sunt detectate doar dacă consumă mult mai mult curent.

## 2.5 Testarea termică a circuitelor integrate

Este cunoscut faptul că în destul de multe cazuri circuitele defecte se încălzesc mai mult decât circuitele funcționale. Un defect care afectează temperatura disipată de un DUT (Device Under Test) se numește defect termic. De obicei defectele de scurt-circuit produc o încălzire suplimentară a DUT. Testarea termică își propune monitorizarea temperaturii disipate de un circuit considerând ca defecte circuitele care au o temperatură disipată mai mare decât cea normală, a circuitelor fără defecte. Măsurarea temperaturii pentru un anumit vector de test trebuie efectuată după stabilirea echilibrului termic astfel că această metodă de testare este statică.

Tehnicile de testare termică se împart în două mari clase în funcție de metoda de măsurare (dectție) a temperaturii:

- metoda termo-electrică – constă în aplicarea unui curent constant la nivelul unei joncțiuni și compararea tensiunii pe joncțiune cu o anumită valoare de referință. Tensiunea măsurată la nivelul joncțiunii depinde de temperatură după relația ce descrie variația curentului printr-o joncțiune:

$$I = I_s (e^{qV/kT} - 1)$$

- metoda termo-optică – se bazează pe proprietățile optice ale corpurilor ce emit căldură, proprietăți care servesc la obținerea unei imagini termice a distribuției temperaturilor la nivelul dispozitivului. Metodele termo-optice se clasifică în:

- metode cu contact direct: cristale lichide;
- metode fără contact: infraroșu, reflectometric, interferometric.

Cele mai răspândite metode de testare termică sunt testarea pe baza imaginilor termice (sau termografică) și testarea pe bază de senzori de temperatură (termo-electrică). Testarea termică în infraroșu se bazează pe imaginea în infraroșu a DUT, imagine ce redă distribuția termică din DUT. Pe baza termogramei se poate lua o decizie cu privire la existența defectelor, localizarea defectelor și chiar informații despre corectitudinea proiectării DUT. Pentru cea de-a doua metodă de testare termică, DUT sunt prevăzute cu senzori de temperatură integrați sau măsurarea temperaturii se mai poate face cu ajutorul unor senzori termici externi, iar testarea se face pe baza valorilor măsurate de acești senzori la aplicarea anumitor stimuli de test.

### 2.5.1 Aplicabilitatea testării termice

Caracterizarea, controlul și testarea termică sunt utile pe de-o parte în fiecare pas al procesului de producție al unui circuit, al unei plăci sau al unui sistem și pe de altă parte atât la proiectarea acestora (thermal codesign) cât și în timpul funcționării acestora (auto-testare și auto-control pe bază de temperatură). Controlul și testarea termică au ca și scop atât reducerea costurilor de testare și utilizare (prin controlul puterii consumate, minimizarea numărului de defecte returnate, etc.) cât și creșterea performanței și fiabilității dispozitivelor (scăderea temperaturii disipate).

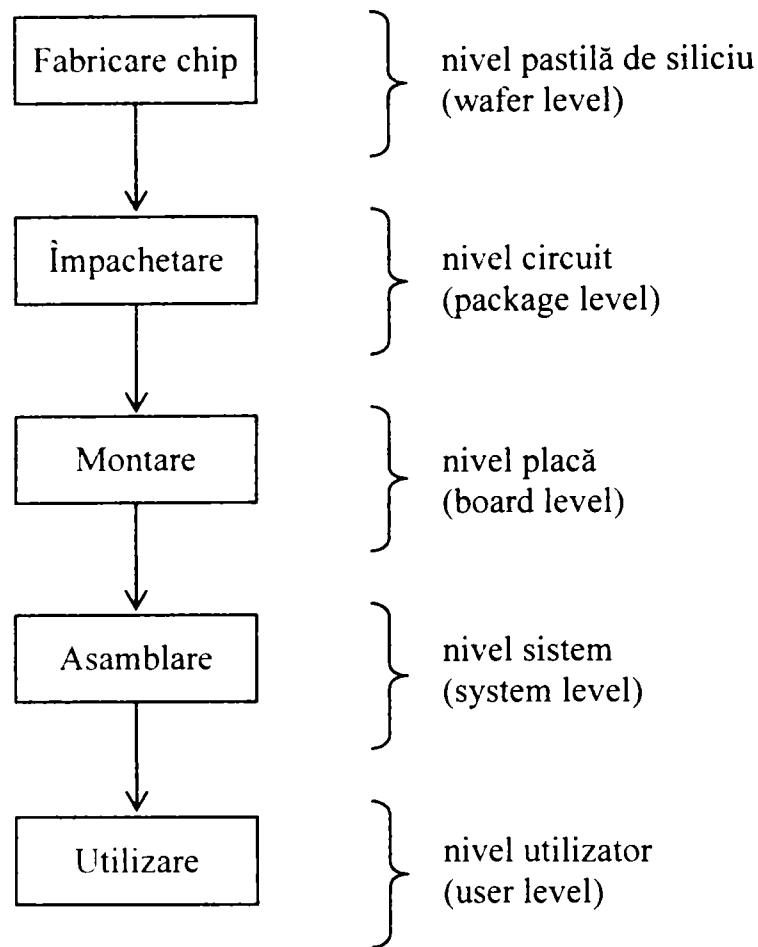


Figura 2.57 – Fazele prin care trece un dispozitiv

Fiecare din stadiile prezentate în figura 2.57 presupune una sau mai multe operații de testare asupra produsului obținut în stadiul respectiv. În faza de testare, în fiecare din stadiile procesului de fabricație, temperatura are un rol foarte important atât în ceea ce privește controlul ei cât și ca metodă specifică de testare parametrică.

Faza de testare din procesul de fabricație a pastilei de siliciu se numește sortare și este primul pas din procesul de testare. Sortarea are ca scop reducerea costurilor de împachetare prin identificarea chip-urilor defecte. În trecut sortarea se efectua la temperatura camerei, acum însă, testele pentru sortare se aplică la temperatură scăzută, la limita inferioară de temperatură pentru care chip-ul este funcțional. S-a observat că aceste teste aplicate la temperatura scăzută detectează o subclasă de defecte care nu ar fi detectate altfel decât în faza de testare structurală aplicată circuitului integrat [Tadayon2000]. În această fază este foarte importantă reducerea temperaturii joncțiunii, investindu-se tot mai multe resurse în această direcție. Temperatura joncțiunii se poate reduce pe două căi: prin scăderea temperaturii mediului în care se face testarea sau prin creșterea conductivității termice (micșorarea rezistenței termice) a chip-ului (vezi formula 2.16). Scăderea rezistenței termice dintre joncțiune și mediu se realizează intervenind la interfațarea pastilei de siliciu cu capsula circuitului prin folosirea unor materiale cu conductivitate mare (grăsimi termice) și la tipul de materiale folosite în circuit (de exemplu cuprul are o conductivitate termică de două ori mai mare ca aluminiul).

Testarea circuitelor integrate împachetate cuprinde cel puțin doi pași: aplicarea uneia sau a mai multor clase de teste logice, structurale și/sau parametrică și aplicarea unui set de teste de stresare (burn-in). O caracteristică a testelor aplicate în această fază este aceea că trebuie să fie aplicate la temperatura sau peste temperatura specificată pentru client în datele de catalog. Temperatura joncțiunii trebuie păstrată însă sub valoarea maximă admisibilă. La

acest nivel se caută soluții mai eficiente de răcire a circuitelor integrate (în special a microprocesoarelor) prin îmbunătățiri aduse sistemului de încapsulare, utilizarea de materiale cu conductivitate termică mare folosirea de elemente de răcire externe (cu și fără ventilație).

Testele burn-in se aplică la mai multe (în jur de 1000) unități asamblate în același timp fiind stresate la temperaturi și tensiuni ridicate, pentru a activa defectele latente ce nu produc modifică logice sau parametrice în cazul funcționării normale. Unul din scopurile acestei faze de testare este reducerea timpului de testare. Timpul testării se reduce dacă se menține temperatura joncțiunii la o valoare ridicată dar sub valoarea maximă admisibilă deoarece defectele latente sunt activate mai repede la o temperatură mai ridicată.

După cum se poate observa temperatura este foarte importantă în procesul de testare în primul rând datorită informațiilor ce le furnizează despre dispozitivul testat, date pe baza cărora se poate stabili corectitudinea dispozitivului atât din punct de vedere funcțional cât și al proiectării. În al doilea rând temperatura este importantă pentru reducerea costurilor testării și al defectelor returnate, prin utilizarea de metode eficiente de răcire și control termic.

## 2.5.2 Testarea termo-optică

Analiza imaginilor termice se bazează pe faptul că fiecare componentă electronică are, ca o caracteristică principală, puterea și, după cum s-a prezentat, aceasta se transformă în căldură disipată, care este o altă caracteristică a fiecărui circuit. Deviația de la modelul de disipare normală a căldurii constituie un indiciu al apariției unui defect.

Așa cum s-a prezentat într-un paragraf precedent (2.2) defectele de scurt-circuit activate de un anumit set de vectori produc mai mult curent decât în absența defectului, curent care produce o cantitate de căldură semnificativă. Tehnicile de testare termică folosesc aceste efecte pentru a detecta și localiza defectul. Există mai multe tehnici de testare pe baza imaginilor termice acestea fiind clasificate în funcție de modul de obținere a imaginii termice:

- cu contact fizic direct:
  - testarea termică pe baza imaginilor obținute cu cristale lichide;
  - testarea termică pe baza imaginilor microtermice fluorescente;
- fără contact fizic:
  - testarea termică pe baza imaginilor în infraroșu;
  - analiza termică pe baza imaginilor determinate reflectometric și interferometric.

Achiziția imaginilor în infraroșu se bazează pe radiația emisă sau reflectată de un dispozitiv care nu se găsește la temperatura de zero absolut. Porțiunile mai fierbinți dintr-o imagine termică sunt reprezentate prin culori mai deschise, în timp ce zonele mai reci sunt reprezentate prin culori mai închise. Fiecare imagine termică are atașată și o scală de culori împreună cu echivalența lor în temperatură (figura 2.58). Pe baza unor astfel de imagini termice se pot determina și localiza punctele calde din dispozitiv, analiza acestora stabilind dacă dispozitivul este corect sau nu. Imaginile termice în infraroșu au o foarte bună rezoluție termică însă rezoluția lor spațială și temporală este limitată. Pata de temperatură cauzată de conductibilitatea termică este difuză și se întâmpină dificultăți la analiză atunci când acestea sunt mai mici de 5  $\mu\text{m}$ .

Rezoluția termică a imaginilor în infraroșu este de ordinul 0.1  $^{\circ}\text{C}$  însă acuratețea imaginilor în mod real se încadrează între 0.2  $^{\circ}\text{C}$  și 0.3  $^{\circ}\text{C}$ . Rezoluția spațială este de ordinul a 10  $\mu\text{m}$ . Metoda este destul de rapidă putându-se investiga și fenomenele tranzitorii [Szekely1998]. Principalul dezavantaj al acestei metode este prețul foarte ridicat al unei camere de achiziție a imaginilor în infraroșu, costul fiind în jur de 100000USD.

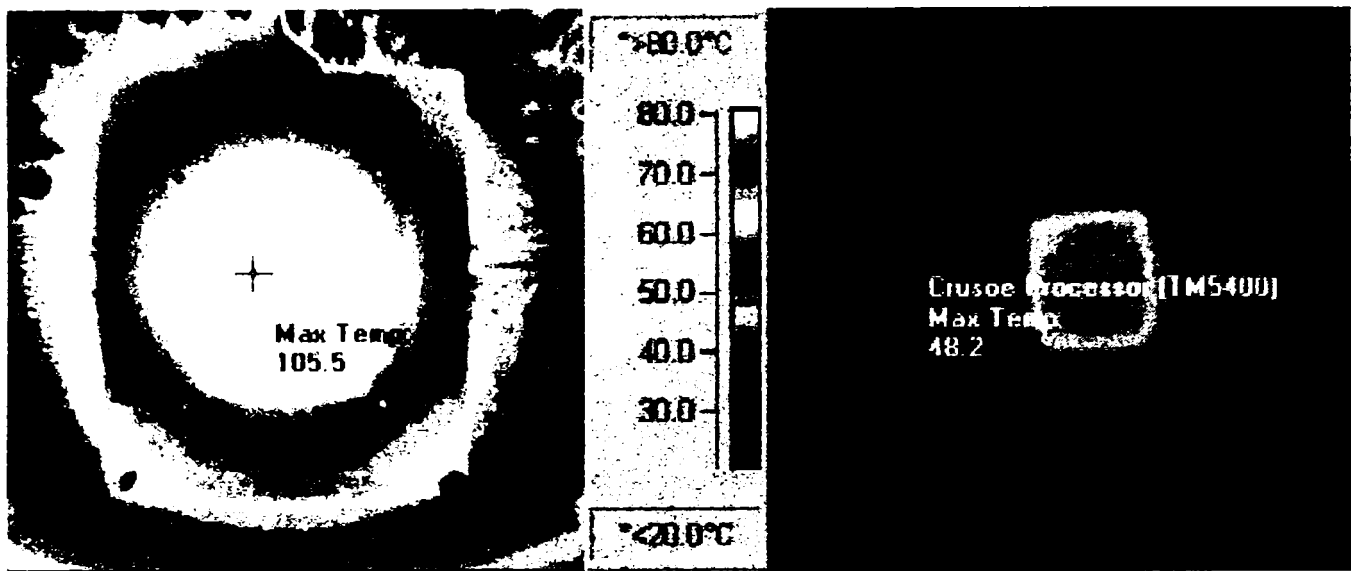


Figura 2.58 – Imagini termice pentru două microprocesoare [Transmeta2002]

Metodele de achiziție a imaginilor termice bazate pe cristale lichide fac parte din clasa metodelor cu contact direct cu sursa de căldură. Cristalele lichide sunt substanțe care își schimbă starea, din starea solidă cristalină în starea lichidă opacă și în starea lichidă transparentă, pe măsură ce temperatura crește (figura 2.59) [Farino1995]. Această metodă presupune aplicarea unui strat subțire și uniform de folie cu cristale lichide pe suprafața circuitului cărui i se aplică vectorii de test. Zonele calde de pe suprafața circuitului vor fi puse în evidență de zonele mai transparente din folia cu cristale lichide iar zonele mai reci se vor transpune în zonele opace din cristalele lichide (figura 2.59) [Hora2002].

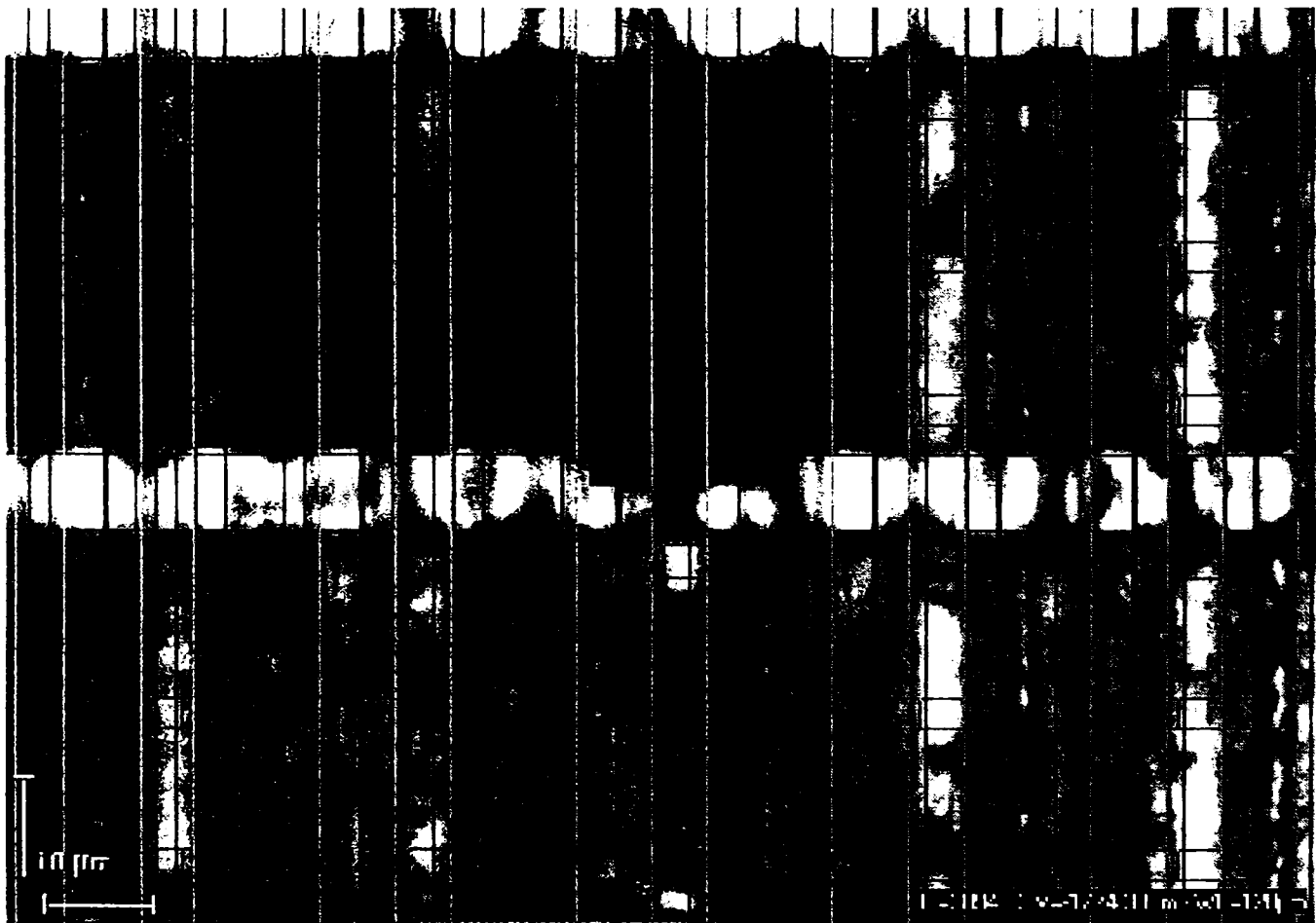


Figura 2.59 – Imagine termică folosind metoda cristalelor lichide [Hora2002]



Rezoluția termică a acestei metode este mai bună de  $0.1\text{ }^{\circ}\text{C}$  iar rezoluția spațială este între  $1\text{ }\mu\text{m}$  și  $3\text{ }\mu\text{m}$ , în funcție de grosimea stratului de cristal lichid [Szekely1998]. Costurile implicate de această metodă sunt mult mai mici decât în cazul imaginilor în infraroșu. Dezavantajul metodei constă în timpul mare de măsurare, între 10 și 20 de minute.

Metoda imaginilor microtermice fluorescente, este o altă tehnică de analiză fizică care detectează punctele temperaturilor ridicate dintr-un dispozitiv defect. În acest procedeu se folosește un film termorezistiv ce produce o hartă termică de o înaltă rezoluție.

Un alt important efect secundar al corpurilor calde este emisia de lumină, fenomen ce este exploatat de o altă categorie de metode de testare termică. Un circuit integrat în timpul operațiilor electrice emite lumină. Joncțiunile tranzistoarelor emit fotoni în spectrul vizibil și infraroșu. Prin analiza lungimii de undă a luminii emise de un circuit se poate localiza și clasifica defectul (figura 2.60). Această ultimă metodă se aplică la analiza defectelor deoarece necesită acces optic direct la pastila de siliciu.

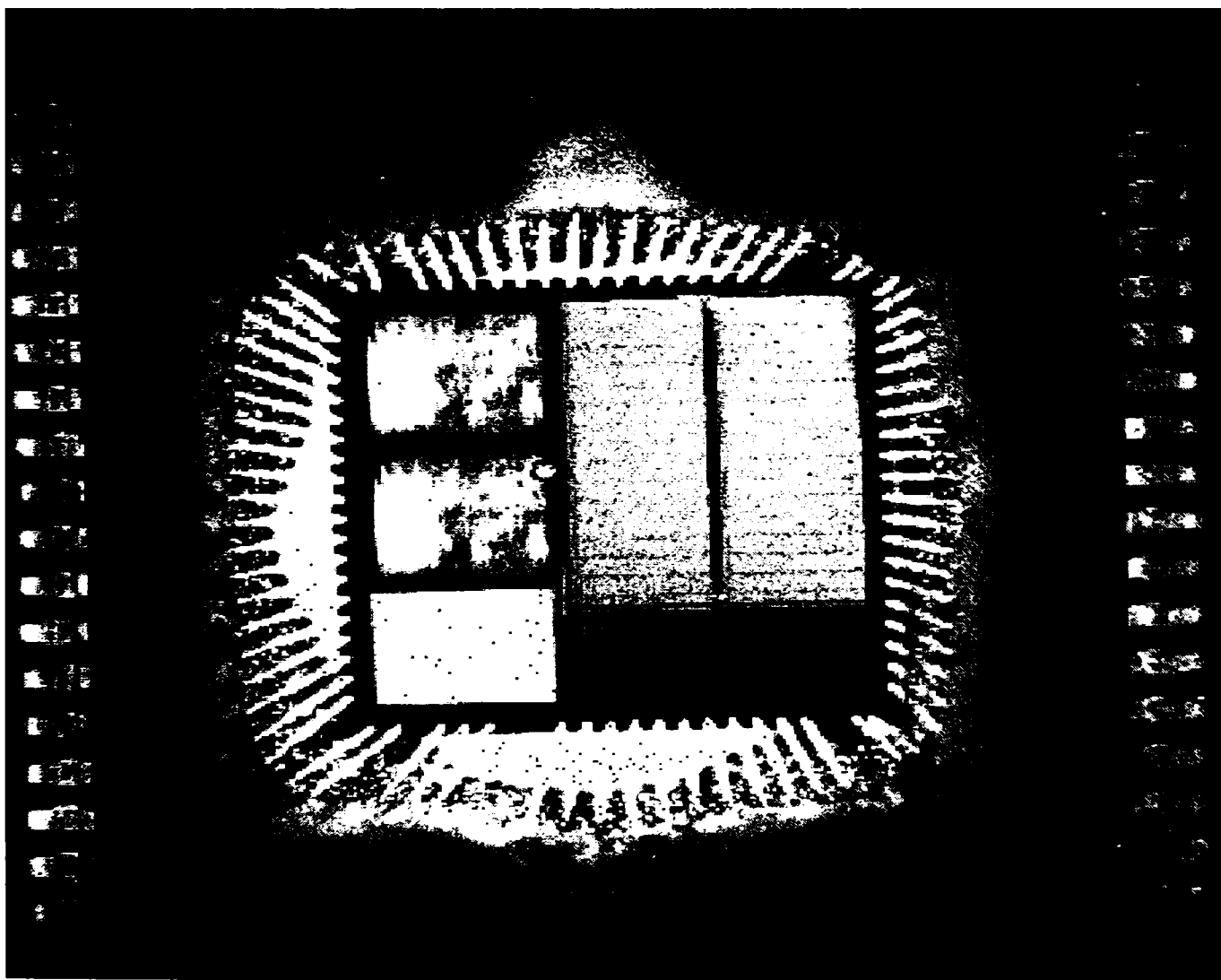


Figura 2.60 – Emisia de lumină a unui circuit defect [Hora2002]

Testarea dispozitivelor electronice pe baza imaginilor termice este utilizată de tot mai mulți producători pentru îmbunătățirea calității circuitelor și ansamblelor produse. În mod tradițional, inspecția vizuală era realizată manual de muncitori experimanțați care observă modificările față de imaginea corectă. Inspecția automată a dispozitivelor electronice pe baza imaginilor termice se realizează folosind algoritmi specifici, care însă folosesc fiecare anumite cunoștințe a priori despre dispozitivul testat. Aceste cunoștințe a priori se referă la strategii și standarde necesare procesului de testare.

Ca metode de testare automată pe baza imaginilor termice se pot aminti:

- compararea (scăderea) imaginilor – este o metodă simplă și imediată de testare ce presupune compararea imaginii termice a dispozitivului testat (DUT) cu o imagine martor a dispozitivelor fără defecte. Dacă există diferențe semnificative între cele două imagini atunci DUT se consideră defect. Ca dezavantaje trebuie amintite: cantitatea mare de date de referință stocate, alinierea între imagini trebuie să fie foarte exactă, condițiile de achiziție a imaginilor trebuie să rămână nemodificate, etc.;
- potrivirea caracteristicilor – este o extensie a metodei de scădere a imaginilor în care nu se compară imaginile ci doar caracteristicile lor. Pentru aceasta trebuie definit un set de mărimi caracteristice circuitelor fără defecte mărime care vor avea valori diferite pentru circuitele cu defecte. Ca mărimi caracteristice se pot folosi: conturul, culori, distanțe, histogramă, etc.;
- metode sintactice – oferă posibilitatea descrierii unui set larg de obiecte complexe utilizând un set restrâns de primitive și un set de reguli structurale. Caracteristicile locale pot fi descompuse într-un număr relativ mic de elemente unice numite primitive, cum ar fi colțuri, linii, cercuri, etc. Astfel că imaginea este descrisă pe baza acestor primitive combinate pe baza setului de reguli și defectele, la rândul lor, sunt descrise la fel. La testare se verifică dacă există vreun șir corespunzător defectelor în imaginea testată.

### 2.5.3 Testarea termo-electrică

În circuitele digitale și analogice o anumită gamă de defecte fizice (scurt-circuitele) pot genera o cantitate neasteptată de căldură. Când un vector de intrare activează un defect, anumite părți ale circuitului pot disipa o putere mai mare cât timp vectorul de intrare este aplicat. Datorită acestei disipări de putere, va crește și temperatura suprafeței de siliciu din vecinătatea tranzistorilor afectați de defect.

Scopul testului termo-electric este, ca prin măsurarea temperaturii în câteva puncte ale suprafeței de siliciu (puncte de monitorizare), folosind niște senzori incorporați sau a unor senzori externi comandați (sensibili la radiația în infraroșu), de a detecta aceste creșteri de temperatură și prin acestea de a detecta activarea unui defect.

Mai mult, datorită naturii difuzive a mecanismului de transfer termic și a materialelor și structurii unui circuit integrat, simpla procesare de semnal a reprezentării grafice a temperaturii sesizate în punctele de monitorizare poate să ofere informații despre distanța între sursa de căldură (defectul activat) și punctul de monitorizare. În termeni de timpi de testare, tehnica este foarte interesantă, cel puțin în comparație cu metodele de testare  $I_{DDQ}$ . De obicei se crede că difuzia căldurii și evoluția temperaturii sunt mecanisme lente. Acest lucru este adevărat cu privire la observarea sistemelor macroscopice, dar în cazul circuitelor integrate nu se întâmplă la fel. Conductivitatea termică a siliciului este mai mare decât cea corespunzătoare oțelului și în aceeași măsură decât cea a tungstenului. Acest fapt împreună cu materialul în cantități foarte mici considerat în analiză (circuite integrate miniaturizate) implică o rezistență termică și o capacitate termică reduse, rezultând un timp mic constant (în suprafața circuitelor integrate o creștere de  $0.1\text{ }^{\circ}\text{C}$  poate fi propagată la  $50\text{ }\mu\text{m}$  distanță în mai puțin de  $1\text{ }\mu\text{s}$ ).

Principiul laserului termorelectometru este prezentat în continuare: când temperatura semiconductorului sau a metalului se schimbă, coeficientul lui de reflexie a luminii (pentru incidența normală și lungime de undă fixată) suferă o schimbare. Pentru temperaturi joase sub un anumit prag, relația între ambele schimbări poate fi considerată liniară. Focalizând o rază laser pe suprafața pentru care trebuie determinată temperatura ( $3\text{ }\mu\text{m}$  diametru) și măsurând lumina reflectată cu o fotodiodă, se obține un curent electric proporțional cu temperatura suprafeței aflată sub acțiunea laser-ului. Caracteristicile sistemelor externe de măsură a

temperaturii sunt: măsurători ale schimbărilor dinamice de temperatură cu intervale de frecvență de la DC la 10 MHz, cu rezoluție laterală de 1  $\mu\text{m}$  și o sensibilitate de 1  $\text{m}^\circ\text{C}$ . Intervalul măsurabil al lui  $\Delta T$  merge de la  $10^2$  până la  $10^{-3}^\circ\text{C}$  [Altet1997, Szekely1998].

Cel mai potrivit loc pentru senzorii de temperatură este însă în interiorul capsulei, ideal ar fi pe suprafața pastilei de siliciu. Cerințele pentru senzorii de temperatură integrați sunt următoarele:

- compatibilitate cu procesul tehnologic;
- dimensiune redusă;
- consum redus;
- interfață în format digital.

În continuare sunt descrise pe scurt elementele care stau la baza senzorilor de temperatură integrați. Temperatura în orice punct de pe suprafața de siliciu poate fi măsurată folosind o diodă și o sursă stabilă de curent. Căderea de tensiune pe diodă este dependentă de curentul care trece prin ea și temperatura joncțiunii diodei. Prin menținerea unui curent constant și stabil prin diodă, tensiunea pe diodă poate fi utilizată ca o măsură exactă a temperaturii de operare a diodei [Gunther2001].

Senzorii de temperatură diferențiali produc o tensiune proporțională cu diferența de temperatură a celor două puncte de monitorizare plasate pe suprafața de siliciu. Avantajul acestei strategii de măsurare este obținerea unui grad mai mare de imunitate la variațiile de deplasare ale hărții termice la suprafața siliciului [Altet1997]. Aceste variații ar putea proveni de la schimbări ale temperaturii ambientului sau de la componentele statice ale puterii disipate de circuitul integrat.

Acuratețea senzorilor prezentați este destul de slabă: între 1  $^\circ\text{C}$  și 3  $^\circ\text{C}$ . Aria ocupată de acești senzori e de aproximativ 0.5  $\text{mm}^2$  în procesul tehnologic de 0.7 microni. Puterea disipată este însă foarte mică, de aproximativ 7  $\mu\text{W}$  [Szekely1998]. Ieșirea acestor senzori constă într-o valoare analogică, care ar trebui digitizată într-o valoare numerică. Un alt exemplu de senzor, prezentat în [Szekely1998\_1], se bazează pe dependența de temperatură a parametrilor unui tranzistor MOS. Acest senzor are curentul ca și componentă analogică dependentă de temperatură, iar ieșirea digitală este dată de un convertor curent-frecvență. Ieșirea în frecvență poate fi introdusă într-un numărător, existând astfel posibilitatea de a oferi o valoare numerică digitală dependentă liniar de temperatură. Acuratețea măsurării e de 0.35  $^\circ\text{C}$ , aria ocupată de senzor este 0.005  $\text{mm}^2$  pentru procesul tehnologic de 0.8 microni. Puterea disipată este însă destul de mare: 200  $\mu\text{W}$ .

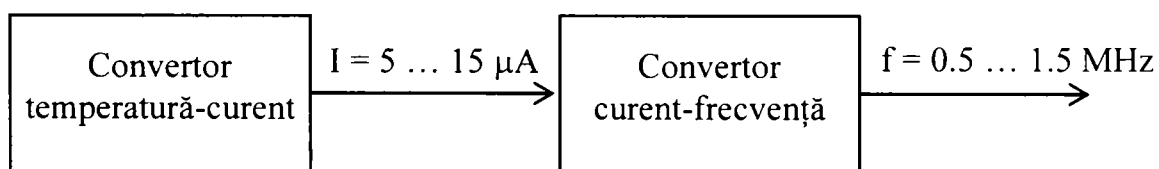


Figura 2.61 – Schema bloc a unui senzor cu ieșire în frecvență

În ultimii ani au fost dezvoltate foarte multe soluții de senzori de temperatură pentru circuitele integrate CMOS [Szekely1998\_1, Lopez2000, Altet2000]. Acestea nu sunt prezentate în detaliu în această lucrare deoarece nu au făcut parte din scopul acesteia.

## 2.5.4 Metoda de testare termică

Pentru a înțelege ce este și cum se aplică testarea termică la circuitele integrate CMOS statice este necesară cunoașterea modului de implementare și funcționare a acestora. Figura 2.62 (a) prezintă o poartă ȘI-NU cu două intrări  $a$ ,  $b$  și o ieșire  $c$ . Această poartă logică este implementată în tehnologia CMOS folosind tranzistoare nMOS și pMOS așa cum este arătat în figura 2.62 (b).

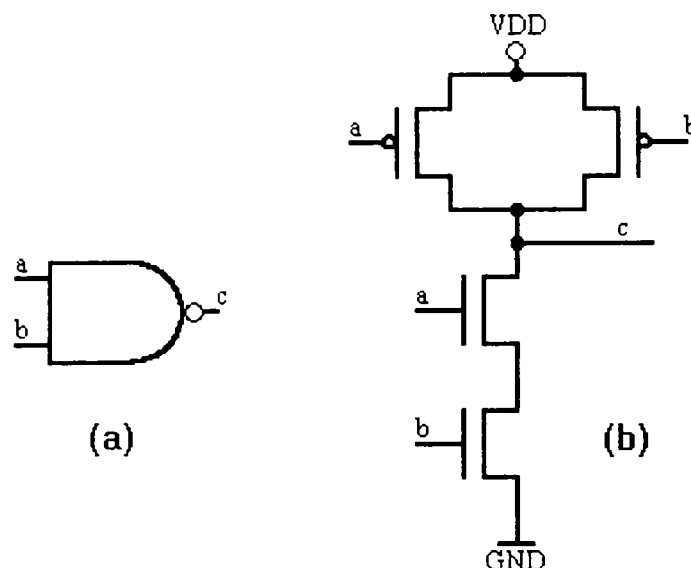


Figura 2.62 – Circuit ȘI-NU implementat în tehnologia CMOS static

Dacă intrările  $a$  și  $b$  sunt ambele poziționate pe 1 logic ("1") atunci cele două tranzistoare nMOS conduc iar ieșirea  $c$  este conectată la masă, deci va avea valoarea 0 logic ("0"). Trebuie observat că pentru această combinație de valori de la intrarea circuitului nici unul din tranzistoarele pMOS nu conduce, și deci, nu există nici o cale de conducție între  $V_{DD}$  și GND. În timpul funcționării statice (s-au încheiat fenomenele tranzitorii) curentul absorbit de la sursă este format doar din curenți de scurgere ce trec prin tranzistoarele MOS. Acest curent static de la  $V_{DD}$  la GND, referit prin  $I_{DDQ}$ , este foarte mic la circuitele fără defecte.

Aplicând la intrări vectorul  $\langle a=0, b=1 \rangle$  se crează o cale de conducție de la  $V_{DD}$  la  $c$ , prin tranzistorul pMOS legat la intrarea  $a$ , iar între  $c$  și GND nu există nici o cale de conducție. Astfel,  $c=1$  și  $I_{DDQ}$  este foarte mic.

Așa cum este prezentat în figura 2.63, fiecare poartă CMOS este compusă din două rețele de tranzistoare MOS: o rețea de tranzistoare pMOS ("trage" ieșirea porții la  $V_{DD}$ ) și o rețea de tranzistoare nMOS ("trage" ieșirea la masă)

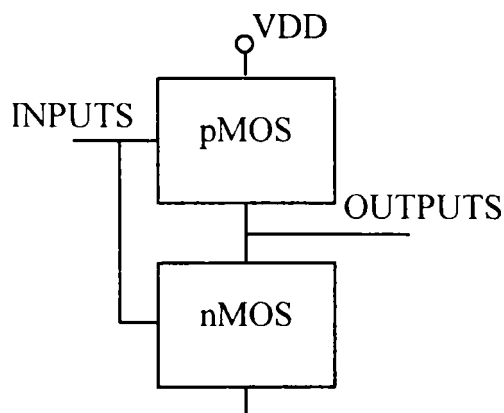


Figura 2.63 – Structura unei porți CMOS

Pentru orice combinație la intrările unei porți logice CMOS, în cazul fără defecte, există sau o cale de conducție de la  $V_{DD}$  la ieșirea porții sau de la ieșire la GND. Astfel, pentru orice combinație a intrărilor,  $I_{DDQ}$  este neglijabil. O altă implicație a acestui fapt este că puterea consumată în starea statică este foarte mică.

În cazul în care într-un circuit CMOS este prezent un defect de scurt-circuit, după cum s-a prezentat în subcapitolul 2.2, acesta duce la creșterea puterii disipate de circuit în faza statică cu câteva ordine de mărime peste cea disipată de circuitul fără defecte.

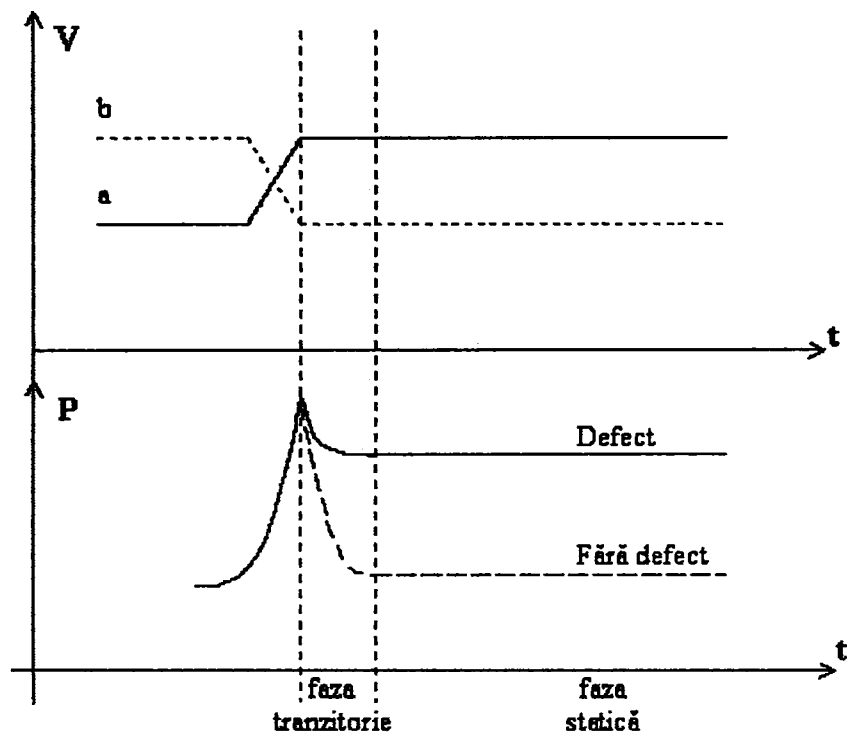


Figura 2.64 – Variația puterii disipate în cazul circuitelor defecte

În testarea termică se aplică un vector de test sau un set de vectori de test la intrările circuitului de testat (CUT – Circuit Under Test), se așteaptă terminarea fazei tranzitorii și apoi se măsoară temperatura circuitului. Dacă cel puțin una din temperaturile măsurate este mai mare decât o anumită valoare de prag, circuitul este considerat defect.

Cu cât dimensiunile sursei de căldură sunt mai mici în comparație cu substratul de siliciu cu atât problema tridimensională a ecuației generale a conducției (ecuația 2.10) poate fi transformată într-una unidimensională în coordonate sferice. Utilizând metoda diferențelor finite, problema transferului de căldură poate fi reprezentată printr-o rețea RC. Cu această tehnică este posibil a se determina creșterea temperaturii generată de sursa de căldură în orice punct de pe substratul de siliciu. De exemplu, figura 2.65 ilustrează creșterea temperaturii generată de un tranzistor PMOS al unei porți NAND cu o celulă standard de  $1.2 \mu\text{m}$ , care disipă  $13 \text{ mW}$ , ca funcție de distanța de la sursa de căldură și timpul cât această sursă de căldură este activă [Altet1997].

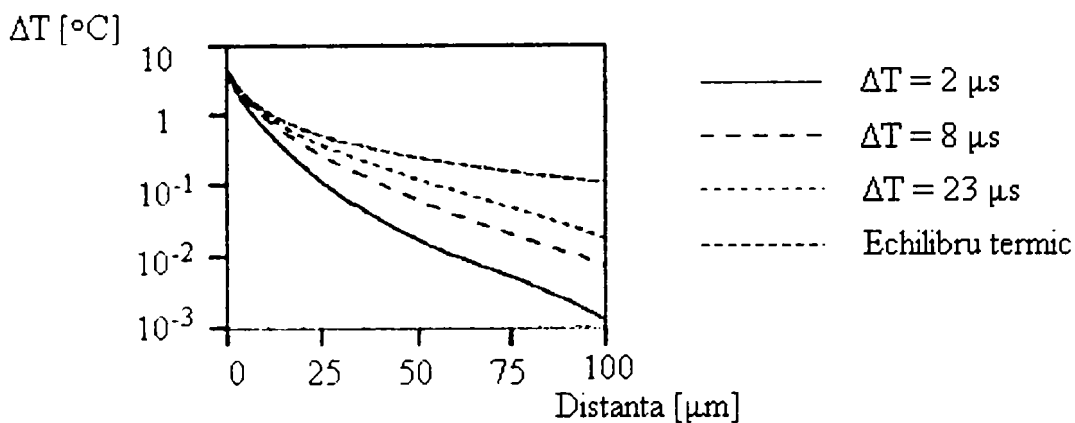


Figura 2.65 – Creșterea temperaturii generate de un tranzistor MOS [Altet1997]

Creșterea maximă a temperaturii este atinsă într-un anumit punct după o întârziere față de momentul generării căldurii la nivelul sursei (defectului). Această întârziere depinde de distanța dintre locația sursei de căldură și de punctul sensibil și de intervalul de timp cât sursa de căldură este activă. De aceea informația despre locul și magnitudinea sursei de căldură poate fi extrasă prin analizarea hărții termice a circuitului în timpul testării.

Ca și concluzie, dacă există un defect fizic în structura circuitului și acest defect este activat de un vector de test atunci se stabilește o cale de curent între alimentare și masă. Această cale de curent suplimentară are ca efect creșterea puterii consumate de circuit și a căldurii disipate de acesta. Astfel că aceste defecte din circuitele CMOS disipă o cantitate suplimentară de căldură, care are ca efect creșterea temperaturii în jurul defectului. Magnitudinea acestei surse de căldură, și implicit și a temperaturii, depinde de dimensiunile tranzistoarelor (vezi figura 2.66), tensiunea de alimentare și topologia defectului (vezi subcapitolul 2.2).

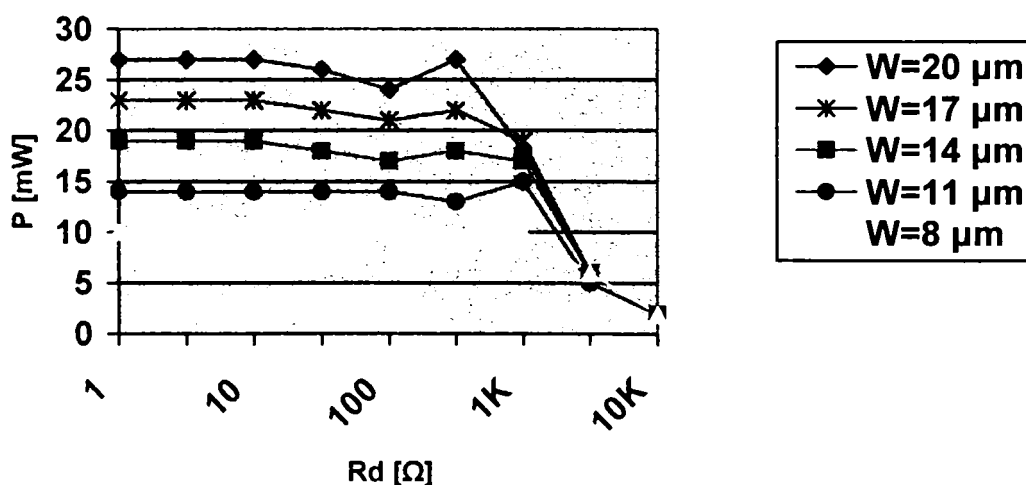


Figura 2.66 – Puterea disipată de sursele de căldură din circuitul din figura 2.15 funcție de  $R_d$  și  $W$

În procesul de testare se măsoară temperatura fiecărui circuit și în funcție de valoarea obținută se ia o decizie asupra corectitudinii din punct de vedere termic a respectivului circuit. În mod uzual temperatura măsurată pentru fiecare circuit se compară cu o anumită valoare de prag și dacă este mai mare atunci circuitul se consideră defect. Determinarea valorii de prag se poate face considerând un lot martor  $S$  ce conține un număr rezonabil de mare de circuite integrate. Setul de vectori de test termic  $T$  folosit se consideră cunoscut. Pentru fiecare circuit

integrat (IC)  $X \in S$ , se aplică vectorii din  $T$  și se măsoară temperatura prin una din metodele prezentate, după aplicarea fiecărui vector. Maximul temperaturilor măsurate pentru un circuit se notează cu  $T_X$  și va fi folosit în continuare pentru determinarea valorii de prag.

Distribuția valorilor  $T_X$  măsurate pentru toate circuitele din  $S$  poate fi aproximată printr-o distribuție gaussiană ca cea din figura 2.67. [Chakravarty1997]

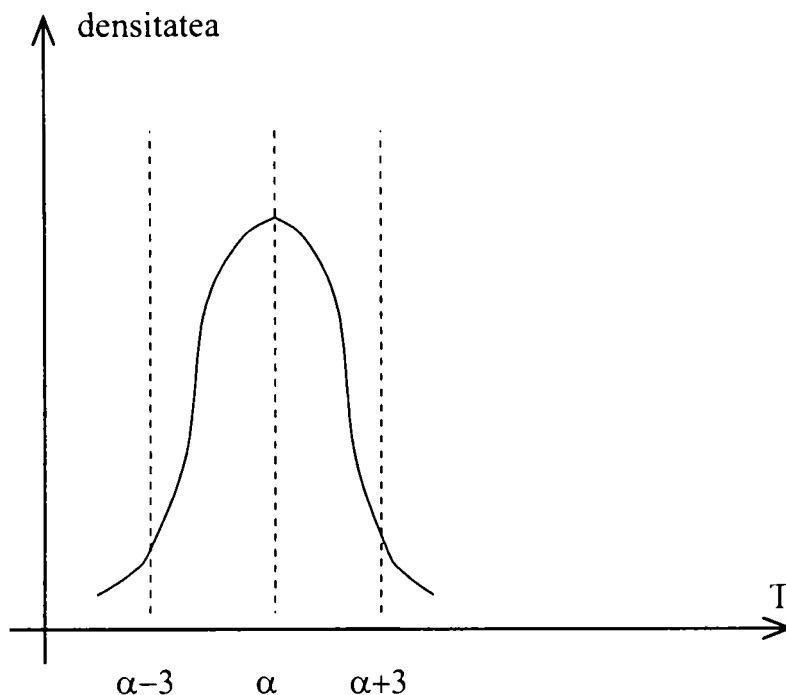


Figura 2.67 – Distribuția normală a temperaturilor pentru circuitele fără defect

Nivelul de prag se calculează în felul următor. Fie  $n$  dimensiunea lotului martor  $S$ , și  $T_{X1}, T_{X2}, \dots, T_{Xn}$  valorile maxime măsurate ale temperaturii pentru fiecare IC din lot. În continuare se calculează media aritmetică  $\alpha$  și deviația standard  $\sigma$ :

$$\alpha = \frac{1}{n} \sum_{i=1}^n T_{Xi}$$

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n (T_{Xi} - \alpha)^2$$

După cum s-a arătat și în figura 2.67 intervalul  $[\alpha-3\sigma, \alpha+3\sigma]$  pentru o distribuție normală conține partea cea mai semnificativă din distribuție. Iar nivelul de prag se ia  $\alpha+3\sigma$ .

Această metodă este însă corectă dacă lotul martor  $S$  nu conține circuite defecte. Această presupunere nu este însă validă și conduce la o valoare mai mare a valorii de prag datorită a doi factori: (i) procentajul circuitelor defecte din  $S$  (acesta este destul de mare în special dacă procesul de fabricație nu este unul matur); (ii) valorile mari ale temperaturilor pentru circuitele defecte.

Fie  $T_A$  valoarea temperaturii astfel încât 25% din circuitele din  $S$  au  $T_X < T_A$ ,  $T_B$  valoarea temperaturii pentru care 75% din circuitele din  $S$  au  $T_X < T_B$ ,  $R_{1-3} = T_B - T_A$  și  $M$  mediana distribuției de curent. Cu aceste valori se poate atribui nivelului de prag valoarea  $M + 3R_{1-3}$ .

Dacă se folosește metoda experimentală de determinare a valorii de prag trebuie ca această valoare să fie recalculată pe măsură ce procesul de fabricație se maturizează. Inițial, procentajul circuitelor defecte din lotul martor este mai mare, deci va conduce la o valoare de prag mai mare. Însă, în timp, procesul de fabricație se dezvoltă și procentajul circuitelor defecte scade, de aceea valoarea de prag trebuie recalculată pentru a se obține o separare mai bună a circuitelor.

## 2.5.5 Localizarea și diagnosticarea defectelor prin testarea termică

Localizarea defectelor este cea mai critică fază în procesul de analiză a defectelor. Localizarea poate fi făcută la nivel de bloc logic sau la nivel de rețea de conexiuni, sau în anumite cazuri direct pe locația defectă. Aici avem două grupuri de tehnici principale prin care se face localizarea: fizică (bazat pe hardware) și electric (bazat pe software). Primul grup necesită o importantă investiție în echipament, unelte și personal calificat în testare, arhitectura dispozitivului și lucrul cu echipamentul. Tehnica de localizarea fizică este de asemenea consumatoare de timp dar în cel mai multe cazuri izolează defectul foarte exact. Tehnicile de diagnoză electrice combină rezultatele simulării cu datele ce vin de la tester, pentru a identifica defectul. Simularea folosește modele cu erori și de aceea nu sunt întotdeauna precise și câteodată rezultatul este imprecis iar locația este cea nedorită.

Metodele termice de testare pot fi utilizate și în diagnosticarea defectelor și localizarea acestora. În majoritatea firmelor producătoare de dispozitive electronice defectele returnate de clienți sunt foarte atent analizate pentru a se detecta cauza defectului și motivul pentru care acesta a trecut de faza de testare. Circuitelor integrate defecte returnate sunt analizate printr-o metodă fizică sau electrică, un exemplu de metodă fizică termică ar fi cea pe baza emisiei de lumină prezentată în figura 2.60.

Localizarea defectelor termice se poate estima și dacă se utilizează senzorii integrați de temperatură. Astfel dacă sunt prevăzuți într-un circuit integrat mai mulți senzori termici, dispuși conform unor reguli de acoperire (câte unul în fiecare bloc funcțional sau câte unul în fiecare colț, etc.) atunci în funcție de valorile fiecărui sensor se poate estima poziția fizică a defectului în pastila de siliciu.



## 2.6 Probleme la testarea termică a circuitelor realizate în tehnologie submicronică

Testarea termică se bazează pe presupunerea că la circuitele fără defecte puterea disipată statică este foarte mică. Tendințele tehnologice actuale de proiectare și implementare a circuitelor integrate duc la creșterea puterii disipate și a temperaturii, astfel devine dificilă folosirea testării termice pentru aceste circuite, deoarece nu se mai poate face distincție pe baza temperaturii între circuitele defecte și cele fără defecte.

Pentru a înțelege aceste tendințe de creștere ale puterii statice disipate este necesară o descriere a componentelor ce formează curentul de blocare ( $I_{OFF}$ ) al unui tranzistor MOS. Tabelul 2.5 compară curenții  $I_{OFF}$  pentru mai multe procese tehnologice și pune în evidență creșterea acestora o dată cu scăderea dimensiunii tranzistoarelor [Keshavarzi1998].

Proces tehnologic [ $\mu\text{m}$ ]	$V_{DD}$ [V]	$T_{OX}$ [ $\text{\AA}$ ]	$V_T$ [V]	$L_{eff}$ [ $\mu\text{m}$ ]	$I_{OFF}$ [pA/ $\mu\text{m}$ ]
1.0	5	200	-	0.80	4.1E-4
0.8	5	150	0.60	0.55	5.8E-2
0.6	3.3	80	0.58	0.35	0.15
0.35	2.5	60	0.47	0.25	8.9
0.25	1.8	45	0.43	0.15	24
0.18	1.6	30	0.40	0.10	86

Tabelul 2.5 – Compararea proceselor tehnologice

unde:

- $V_{DD}$  – tensiunea de alimentare;
- $T_{OX}$  – grosimea peliculei de oxid;
- $V_T$  – tensiunea de prag;
- $L_{eff}$  – lungimea efectivă a canalului;
- $I_{OFF}$  – curentul de blocare al tranzistorului MOS.

Curentul de blocare  $I_{OFF}$  al unui tranzistor MOS este curentul de drenă în cazul  $V_{GS}=0$ . Puterea statică disipată de un circuit integrat CMOS este dependent de curenții  $I_{OFF}$  ce apar în tranzistoarele MOS blocate.

### 2.6.1 Mecanismele de scurgere într-un tranzistor MOS

Mecanismele de scurgere într-un tranzistor MOS (figura 2.68) [Keshavarzi1998] sunt următoarele:

- $I_1$  curentul de scurgere într-o joncțiune  $pn$  polarizată invers;
- $I_2$  curentul de inversiune;
- $I_3$  micșorarea barierei de potențial indusă de drenă (DIBL – Drain Induced Barrier Lowering);
- $I_4$  curentul de scurgere din drenă indus de grilă (GIDL – Gate-Induced Drain Leakage);
- $I_5$  străpungerea canalului (punchthrough);
- $I_6$  efectul de apropiere;
- $I_7$  curentul de scurgere prin stratul de oxid; (pierderile prin stratul de oxid)
- $I_8$  curentul de grilă datorat injecției de purtători.

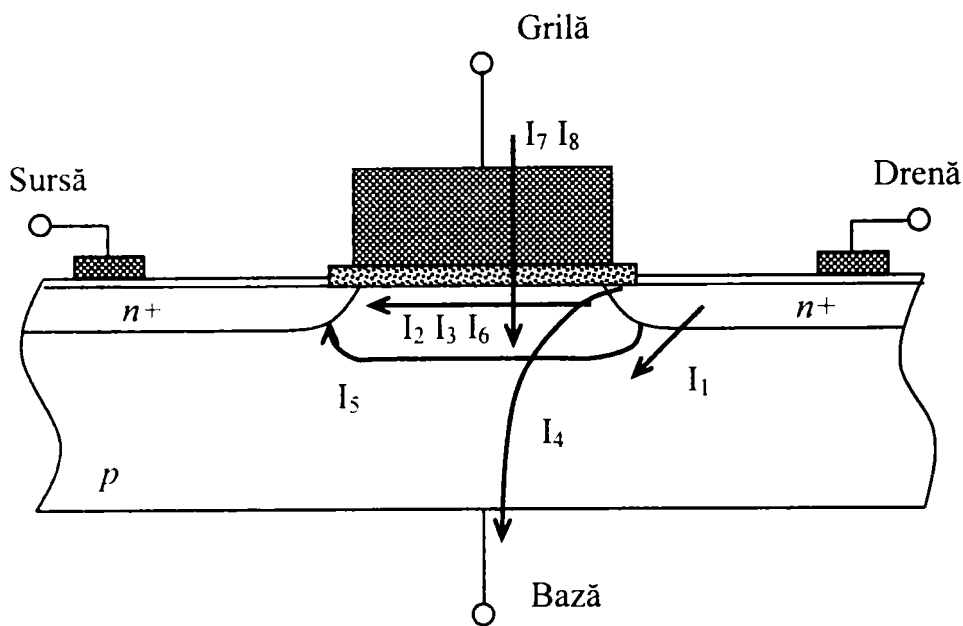


Figura 2.68 – Mecanismele de scurgere în tranzistoarele MOS implementate în tehnologiile submicronice

1. Curentul la polarizarea inversă a unei joncțiuni  $pn$  ( $I_1$ ) are două componente principale:
  - mișcarea de drift a purtătorilor minoritari din regiunea de difuzie;
  - generarea de perechi electron-gol în regiunea de difuzie a joncțiunii polarizate invers.

Curentul  $I_1$  depinde de aria joncțiunii și de concentrația dopării.

2. Curentul de inversiune ( $I_2$ ) între sursă și drenă într-un tranzistor MOS apare atunci când tensiunea aplicată grilei este inferioară tensiunii de prag necesare deschiderii acestuia. Acest curent corespunde porțiunii liniare din graficul prezentat în figura 2.69. Canalul nefiind supus nici unui câmp electric, purtătorii se mișcă prin difuzie, existând o relație exponențială între tensiune ( $V_{GS}$ ) și curentul de drenă ( $I_D$ ). Curentul  $I_2$  este dominant în componentele MOS actuale datorită valorilor mici ale tensiunilor de prag ( $V_T$ ).

3. Curentul  $I_3$  datorat micșorării barierei de potențial indusă de drenă apare atunci când regiunea de difuzie a drenei interacționează cu sursa la nivelul canalului. Sursa injectează purtători în suprafața canalului indiferent de potențialul aplicat la grilă. Acest curent crește cu creșterea  $V_D$  și scăderea  $L_{eff}$  și poate fi scăzut prin creșterea suprafeței și a dopării canalului.  $I_3$  influențează în mod direct tensiunea de prag  $V_T$ .

4. Curentul de scurgere indus de grilă ( $I_4$ ), apare ca produs al câmpului electric din regiunea de suprapunere între grilă și drenă, acest câmp producând difuzie în adâncimea substratului. Acest curent apare la valori mici ale tensiunii de grilă  $V_G$  împreună cu valori mari ale  $V_D$ .

5. Curentul de străpungere a canalului ( $I_5$ ) apare atunci când regiunile de difuzie induse de sursă și drenă se apropie una de cealaltă, ajungând să se „atingă” în substrat. Efectul acestei apropieri este pierderea controlului de către grilă asupra canalului.  $I_5$  variază și crește cu creșterea tensiunii din drenă  $V_D$ .

6. Curentul  $I_6$  este dependent de geometria canalului și are ca și efect micșorarea tensiunii de prag  $V_T$ .

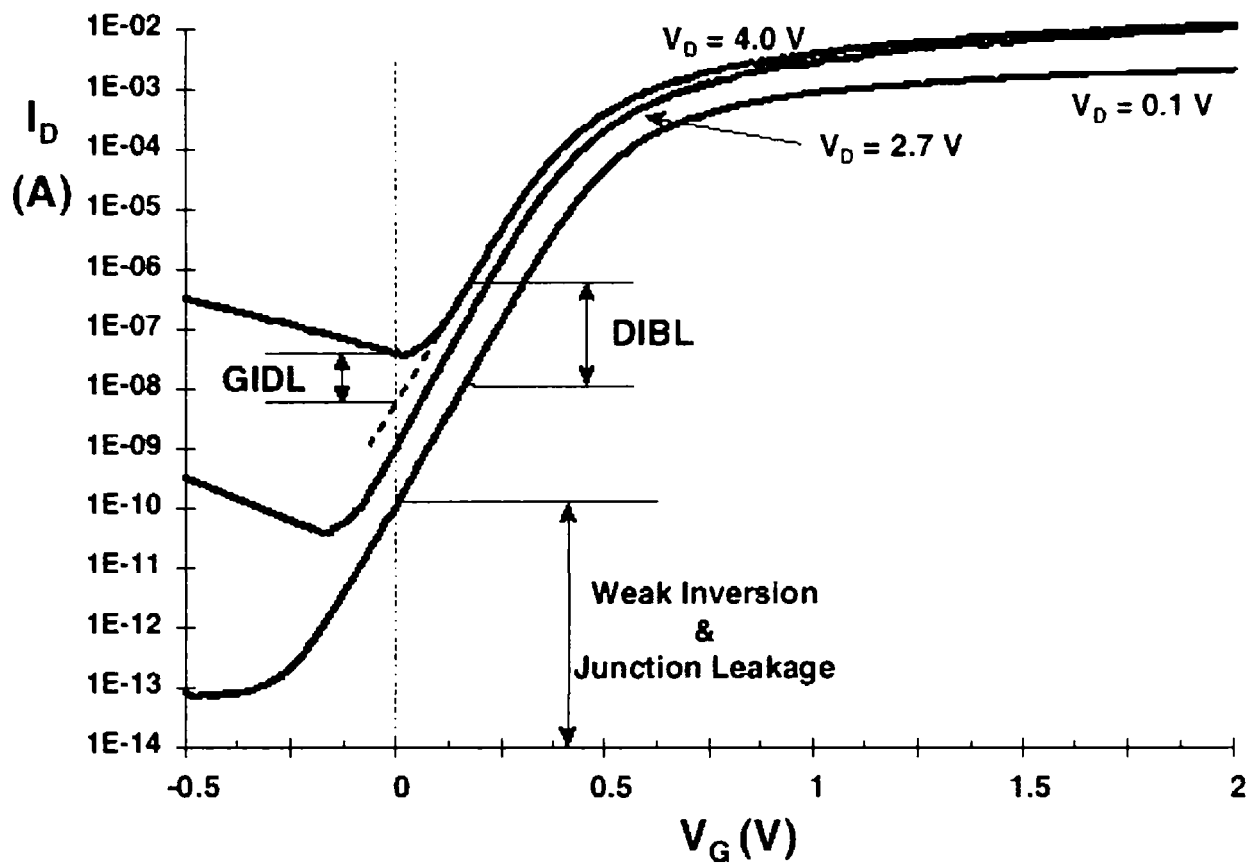


Figura 2.69 – Curenții de scurgere într-un tranzistor MOS [Sachdev1997]

7. Câmpul electric între grilă și substrat poate induce efectul de tunel prin grilă și stratul de oxid. Curentul indus de acest câmp electric este nesemnificativ în tranzistoarele actuale dar ar putea crește dacă se micșorează grosimea stratului de oxid.

8. Curentul  $I_8$  apare datorită injecției de purtători (electroni sau goluri) în stratul de oxid. Acest curent depinde de dimensiunea canalului și crește cu scăderea lui  $L_{eff}$ . Pentru a nu crește se micșorează  $V_{DD}$ .

Figura 2.70 rezumă contribuția relativă a mecanismelor de scurgere la curentul total de scurgere  $I_{OFF}$  pentru tehnologia CMOS de 0.35 microni [Keshavarzi1998].

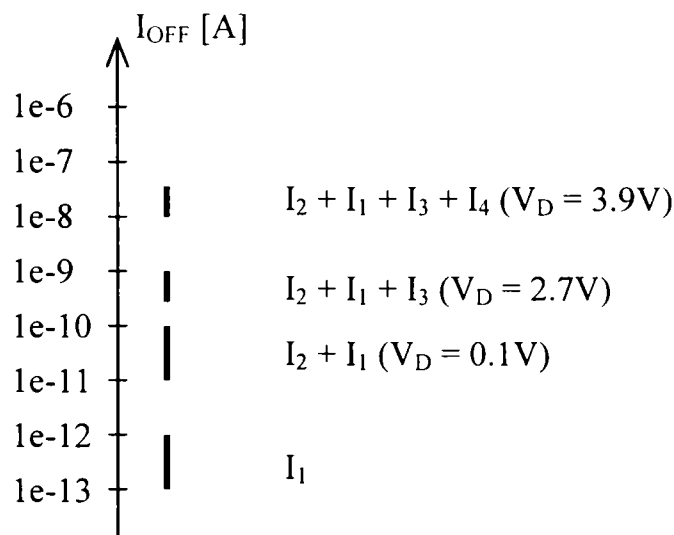


Figura 2.70 – Componentele curentului de scurgere  $I_{OFF}$

Ca și concluzie, din punct de vedere teoretic ar trebui ca pentru  $V_{GS} < V_T$  curentul  $I_{DS}$  să fie zero, în realitate însă curentul din drenă scade exponențial cu micșorarea tensiunii din grilă. Acest curent de scurgere se datorează purtătorilor minoritari și curenților de difuzie într-un tranzistor MOS. Pentru estimarea curentului de scurgere (pentru componentele cele mai semnificative din  $I_{OFF} = I_1$  și  $I_2$ ) se folosește raportul dintre tensiunea din grilă și logaritmul curentului din drenă. Acest raport se numește coeficient unghiular și se notează cu S (Slope). Coeficientul unghiular S depinde de grosimea stratului de oxid și de gradul de dopare al substratului și este exprimat în ecuația următoare:

$$S = \frac{dV_{GS}}{d(\log I_{DS})}$$

Pentru circuitele CMOS actuale S se situează în jurul valorii de 80 mV/decadă și a rămas constant la evoluția tehnologică (tabelul 2.6). Circuitele care au valori ale lui S peste 100 mV/decadă sunt considerate slabe.

Tehnologia CMOS	S [mV/decadă]
0.8 $\mu\text{m}$ 5V	86
0.6 $\mu\text{m}$ 5V	80
0.35 $\mu\text{m}$ 3.3V	80
0.35 $\mu\text{m}$ 2.5V	78
0.25 $\mu\text{m}$ 1.8V	85

Tabelul 2.6 – Variația lui S cu evoluția tehnologică

Pentru estimarea curentului de scurgere se poate folosi formula:

$$I_{off} = \frac{a \cdot W}{L} \cdot e^{b \frac{V_T}{T}}$$

W – lățimea canalului;  
L – lungimea canalului;  
a, b – constante;  
T – temperatura.

## 2.6.2 Soluții tehnologice

După cum s-a arătat, tendința generală este aceea de creștere a curenților de scurgere în circuitele integrate CMOS, astfel că este periclitată utilitatea și eficiența atât a testării termice cât și a testării  $I_{DDQ}$ . Există însă, mai multe metode de reducere a curenților de scurgere intrinseci pentru micșorarea puterii consumate și eficientizarea testabilității  $I_{DDQ}$ .

1. Tehnologii alternative. SOI (Silicon On Insulator) este o tehnologie nouă și promițătoară pentru obținerea unor circuite integrate foarte rapide, cu densitate de integrare mare și consum redus [Sachdev1997, Keshavarzi1998]. Îmbunătățirea în performanță adusă de această tehnologie se datorează reducerii capacitanților parazite și astfel s-au obținut valori pentru S mai mici de 60 mV/decadă. Pentru o aceeași tensiune de prag la circuitele realizate în tehnologie SOI s-a redus curentul de scurgere cu un ordin de mărime comparat cu circuitele implementate în tehnologia CMOS clasică.

2. Temperatura: Factorul de sensibilitate  $S$  scade o dată cu scăderea temperaturii în timp ce tensiunea de prag crește cu scăderea temperaturii – iar acești doi factori contribuie la scăderea  $I_{OFF}$ . Din testele efectuate [Keshavarzi1998] la temperatura de  $-50^{\circ}\text{C}$ ,  $I_{OFF}$  a fost diminuat de 350 de ori.
3. Polarizarea inversă a substratului: Dacă se aplică substratului o polarizare inversă de 2-3V se obține o diminuare cu un factor între 2500 și 4400 de ori a curentului de scurgere  $I_{OFF}$ . Se aplică doar în timpul de funcționare statică nu și pe perioada tranzitorie din ciclul de funcționare al dispozitivului. Prin polarizarea inversă a substratului se reduce componenta GIDL din  $I_{OFF}$ , însă duce la creșterea câmpului electric între grilă și substrat, producând o creștere a curentului de scurgere prin stratul de oxid. Astfel că acest mecanism trebuie aplicat cu grijă și pentru fiecare tehnologie în parte trebuie evaluate consecințele introducerii polarizării inverse a substratului.
4. Tensiunea de alimentare: Curenții de scurgere scad pe măsură ce este micșorat  $V_{DD}$ .
5. Tensiuni de prag multiple: Această tehnică se bazează pe faptul că curenții reziduali scad exponențial o dată cu creșterea tensiunii de prag  $V_T$ . Astfel că într-un circuit vor exista tranzistoare cu nivele de prag diferite, subcircuitul ce necesită viteză sporită vor avea  $V_T$  mic iar pentru blocurile din circuit lente sau sensibile la marginea de zgomot se va alege o tensiune de prag mai mare.  $I_{OFF}$  poate fi redus [Keshavarzi1998] cu patru ordine de mărime la o creștere cu doar 150 mV a tensiunii de prag.

Sachdev [Sachdev1997] a propus o modificare a tranzistorului MOS în vederea testării  $I_{DDQ}$ , stabilindu-se două moduri de lucru pentru circuitele bazate pe acest tip de tranzistor: modul de funcționare normală și modul de testare, în care se încearcă a se stabili condițiile optime pentru testarea  $I_{DDQ}$ . Aceste două moduri de lucru au fost obținute prin separarea contactelor pentru sursă și substrat și alimentarea lor la tensiuni diferite în timpul testării, pentru a se permite polarizarea inversă a substratului (figura 2.71). Din figură se poate observa că  $V_{DD}$  (respectiv  $V_{SS}$ ) a fost împărțit în  $V_{DD \text{ logic}}$  ( $V_{SS \text{ logic}}$ ) la care este conectată drena (sursa) și  $V_{DD \text{ substrat}}$  ( $V_{SS \text{ substrat}}$ ) conectată la substrat. Această separare permite măsurarea individuală și fără ambiguități a diferitelor componente ale curentului de scurgere.

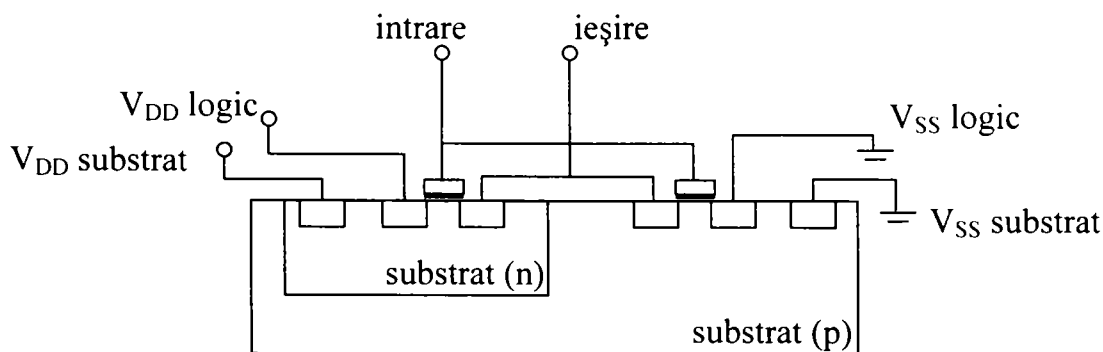


Figura 2.71 – Structura unui tranzistor MOS modificat pentru a facilita testarea [Sachdev1997]

Principalele defecte de scurt-circuit într-un tranzistor MOS sunt (figura 2.72):

- $f_{GS}$  – defect de scurt-circuit între grilă și sursă (1);
- $f_{GD}$  – defect de scurt-circuit între grilă și drenă (2);
- $f_{SD}$  – defect de scurt-circuit între sursă și drenă (3);
- $f_{BS}$  – defect de scurt-circuit între substrat și sursă (4);

- $f_{BD}$  – defect de scurt-circuit între substrat și drenă (5);
- $f_{BG}$  – defect de scurt-circuit între substrat și grilă (6).

Pe lângă acestea mai sunt vizate:

- defectele de scurt-circuit între substratul n și substratul p (7);
- defectele de scurt-circuit inter-poți logice (8 – nereprezentat însă în figură).

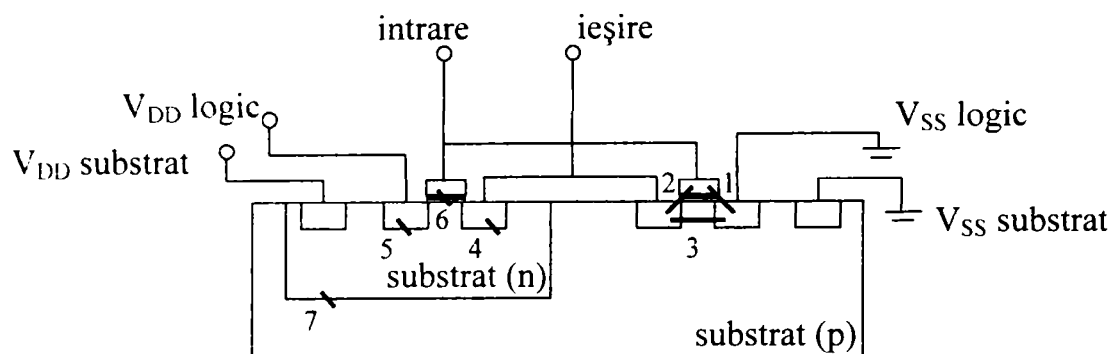


Figura 2.72 – Defectele de scurt-circuit

Din punctul de vedere al necesității aplicării polarizării inverse defectele prezentate mai sus se pot grupa în trei clase:

- defecte ce nu necesită polarizare inversă: – din această categorie fac parte defectele de scurt-circuit substrat-substrat. Acest tip de defect nu depinde nici de starea circuitului și nici de vectorii de test aplicați, deoarece defectul introduce o legătură directă între  $V_{DD}$  și  $V_{SS}$ . Acest curent se poate măsura individual (pentru detalii vezi [Sachdev1997]), iar o valoare ridicată a acestui curent este o dovadă certă pentru defectele din această categorie, deoarece în cazul circuitelor fără defecte acest curent este foarte mic;
- defecte pentru care polarizarea inversă este opțională: – defectele din această categorie nu necesită polarizarea inversă a substratului, însă dacă polarizarea inversă se aplică atunci crește capacitatea de diagnosticare a acestor defecte. Această categorie conține defectele de scurt-circuit dintre substrat și oricare din celelalte trei terminale ale tranzistorului MOS (vezi defectele 4, 5 și 6 din figura 2.72). Ca și exemplu, pentru defectul 6 ( $f_{BG}$ ), dacă la intrare se introduce '0', defectul va produce creșterea curentului  $I(V_{DD\text{ substrat}})$  sau a curentului  $I(V_{SS\text{ logic}})$ .
- defecte pentru care polarizarea inversă este necesară: – această categorie conține defectele ce necesită aplicarea polarizării inverse a substratului pentru a reduce curentul de scurgere pentru circuitele fără defect și pentru a putea pune în evidență circuitele defecte. Defectele din această categorie (1, 2, 3) duc la creșterea curenților  $I(V_{DD\text{ logic}})$  și  $I(V_{SS\text{ logic}})$ .

### 2.6.3 Semnătura termică

Pentru a explica metoda de testare pe baza semnăturii termice mă voi folosi de exemplul din figura 2.73. În figura 2.73 (a) este prezentat circuitul logic iar în (b) este prezentată implementarea acestui circuit în tehnologie CMOS. Tot în figura 2.73 (b) sunt prezentate și două defecte de scurt-circuit pentru care se vor studia semnăturile termice.

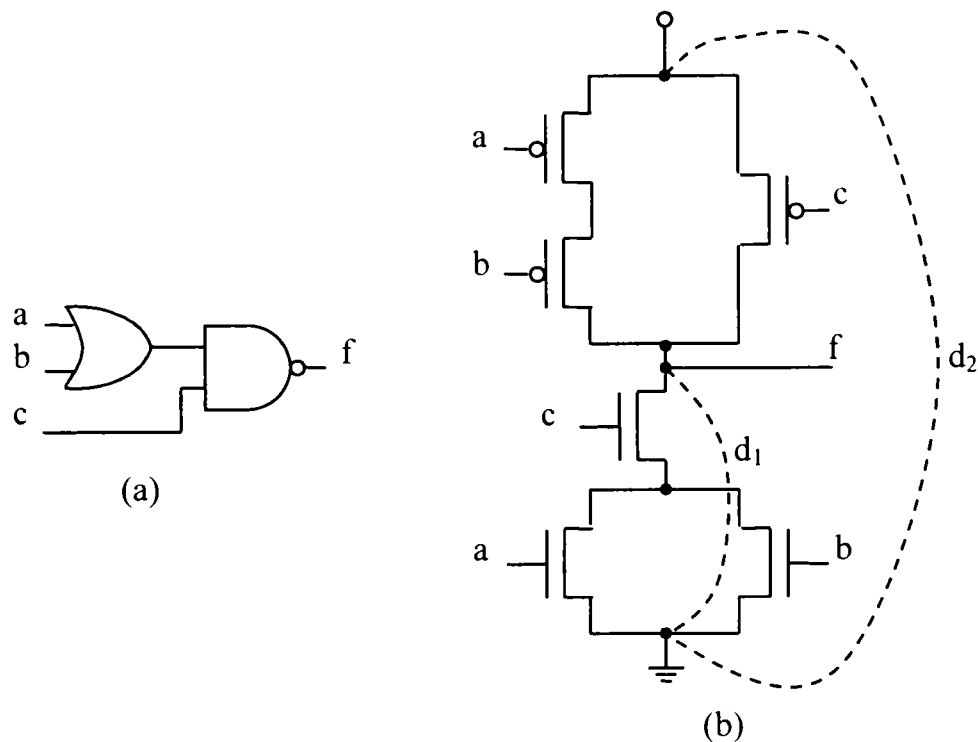


Figura 2.73 – Defecte de scurt-circuit ce afectează diferit semnătura termică

Tabelul de adevăr pentru circuitul din figura 2.73 este prezentat în tabelul 2.7. Se consideră setul de vectori de test  $\{T_1, T_2, T_3, T_4\}$  prezentat tot în tabelul 2.7, împreună cu valorile (calitative doar) temperaturilor pentru circuitul fără defecte, circuitul în care este prezent  $d_1$  și respectiv  $d_2$ .

	a	b	c	f	Temp (fără defect)	Temp (defect $d_1$ )	Temp (defect $d_2$ )
$T_4$	0	0	0	1	$t_2$	$t_4$	$t_5$
	0	0	1	1			
	0	1	0	1			
	0	1	1	1			
	1	0	0	1			
$T_3$	1	0	1	0	$t_1$	$t_1$	$t_5$
$T_1$	1	1	0	0	$t_1$	$t_3$	$t_5$
$T_2$	1	1	1	0	$t_2$	$t_2$	$t_5$

Tabelul 2.7

Reprezentând grafic temperaturile în funcție de vectorii de test se obține graficul din figura 2.74 (a). Dacă se sortează în ordine crescătoare vectorii de test după valoarea temperaturilor, se obține o curbă care caracterizează acest circuit. Această curbă se numește semnătură termică. Considerând  $t_1 < t_2$ , semnătura de curent a circuitului pentru vectorii de test specificați, este prezentată în figura 2.73 (b).

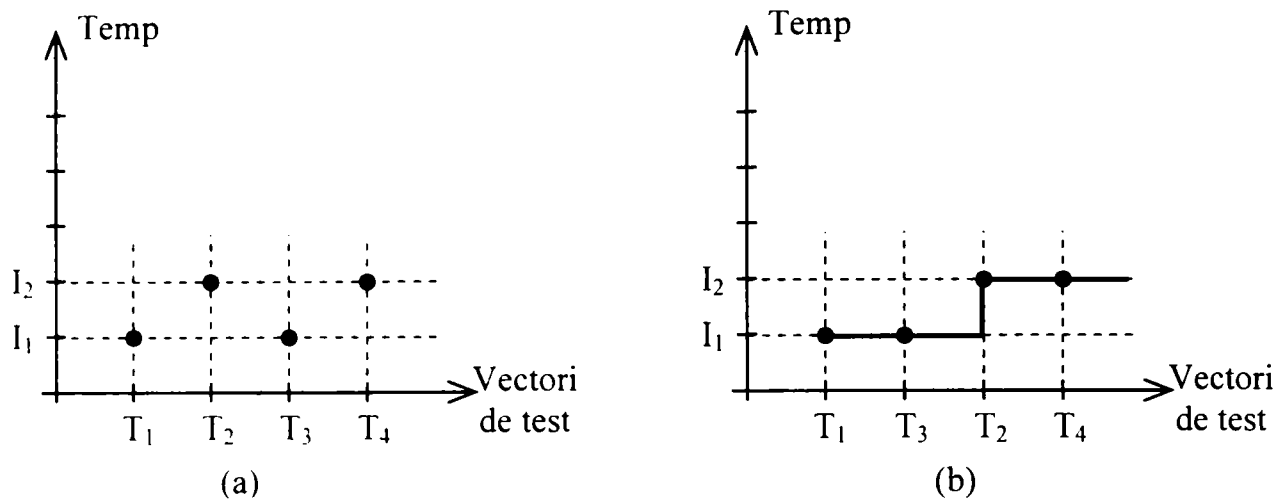


Figura 2.74 – Semătura termică a circuitului fără defecte

Răspunsul circuitului la aplicarea unei secvențe de vectori de test poartă numele de semnătură. Deoarece în cazul testării termice valorile monitorizate și măsurate sunt temperaturi, răspunsul circuitului la aplicarea vectorilor de test se numește semnătură termică. Semnătura termică este o caracteristică a circuitului fără defecte ce poate fi utilizată în testarea circuitelor.

Semnăturile obținute în cazul defectelor  $d_1$  și  $d_2$  sunt prezentate în figura 2.75. Defectul  $d_1$  (figura 2.75 a) depinde de vectorul de intrare aplicat circuitului. Acest tip de defecte se numesc defecte active. Cel de-al doilea defect ( $d_2$ , figura 2.75 b) are același comportament independent de stimulii aplicați la intrările circuitului. Aceste defecte se numesc defecte pasive.

Dacă se dorește detecția celor două defecte folosind metoda nivelului de prag, valoarea acestui nivel de prag trebuie aleasă astfel încât să fie mai mare decât maximum temperaturilor măsurate pentru fiecare vector de test și trebuie să fie mai mică decât valorile temperaturilor ce pun în evidență defectul. Pentru defectul  $d_1$ , dacă  $t_3$  și  $t_4$  sunt mult mai mari decât  $t_1$  și  $t_2$  atunci se poate alege o valoare a nivelului de prag undeva între  $t_1 < t_2$  și  $t_3 < t_4$  și se poate utiliza și metoda nivelului de prag. Și în cazul defectului  $d_2$  dacă  $t_5 \gg t_2$  și  $t_1$  se poate utiliza și metoda nivelului de prag. În schimb, dacă  $t_5$  este apropiat ca valoare de  $t_2$  și  $t_1$ , acest caz poate apare în cazul unui scurt-circuit rezistiv ce are o rezistență mare, atunci acest defect nu poate fi detectat nici prin metoda nivelului de prag și nici prin testarea logică, dar datorită diferenței de semnături, poate fi detectat prin analiza semnăturilor.

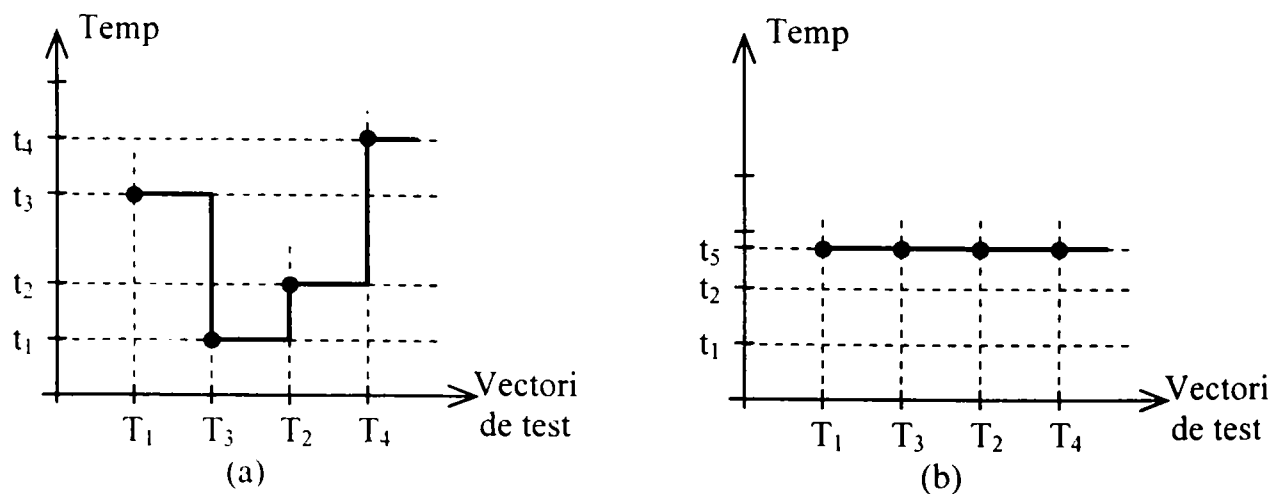


Figura 2.75 – Semnăturile circuitelor defecte



Testarea prin metoda analizei semnăturii termice presupune, într-o primă fază, stabilirea semnăturii termice care caracterizează circuitele fără defecte. Mai apoi, în faza de testare, pentru fiecare circuit testat se determină semnătura termică corespunzătoare și se compară cu semnătura standard a circuitelor fără defecte. Dacă există diferențe majore între cele două semnături se consideră că circuitul testat este defect.

Pe lângă faptul că prin metoda analizei semnăturii de curent se detectează anumite defecte care scapă testării logice sau a testării  $I_{DDQ}$  prin metoda nivelului de prag, această metodă poate fi utilizată cu succes și în cazul circuitelor implementate în tehnologie submicronică ce au valori mari ale curenților  $I_{DDQ}$  pentru circuitele fără defecte.

Ca și dezavantaj al semnăturilor termice trebuie amintit faptul că această metodă necesită calcule mai multe iar timpul de testare este mai mare.

## 2.7 Generarea a vectorilor de test pentru testarea termică

Procesul de generare a vectorilor de test pentru testarea termică constă în căutarea vectorilor de intrare care activează defectele de scurt-circuit ce pot apare în circuit, astfel încât în faza de testare, la aplicarea acestor vectori să se poată face distincția între circuitele defecte și cele fără defect. Algoritmii de generare automată a vectorilor de test pentru testarea termică sunt identici cu cei pentru testarea  $I_{DDQ}$ , deoarece cele două metode de testare vizează și detectează același tip de defecte: defectele de scurt-circuit. Din acest motiv, nu există algoritmi specifici de generare a vectorilor de test pentru testarea termică. Vectorii de test pentru testarea termică se generează folosind unul din algoritmii pentru defectele de scurt-circuit sau pentru testarea  $I_{DDQ}$ . Acești algoritmi au deja o anumită maturitate, existând articole despre ei încă de la începutul anilor '90. [Nigh1990, Mao1990, Mahlstedt1992]

Generarea automată a setului de vectori de test pentru testarea termică se bazează pe modelul ales pentru defecte, și presupune, pe baza descrierii circuitului, găsirea cel puțin a unui vector de test pentru fiecare defect din model. Numărul de defecte de scurt-circuit posibile într-un circuit este uneori considerabil de mare pentru a fi luat în considerare fiecare la generarea vectorilor de test sau pentru simularea erorii. Astfel, pentru un număr de  $N$  noduri într-un circuit vor exista  $N(N-1)/2$  perechi de noduri între care poate exista un defect de scurt-circuit. Generarea unui set de vectori de test pentru toate defectele de scurt-circuit (setul exhaustiv de defecte de scurt-circuit) devine consumatoare de timp pentru circuitele de mari dimensiuni (peste 10000 de porți). În plus, secvența de vectori de test generată este lungă, ceea ce implică un timp foarte mare de testare. Se poate utiliza cu rezultate bune și un subset al defectelor de scurt-circuit. Astfel se poate folosi unul din modelele de mai jos:

- defectele de scurt-circuit la nivel de tranzistor – sunt defectele de scurt-circuit ce pot apare între cele patru terminale ale unui tranzistor MOS: grila (G), sursa (S), drena (D) și substratul (B). (scurt-circuit grilă-sursă – GS, scurt-circuit grilă-drenă – GD, scurt-circuit sursă-drenă – SD, scurt-circuit grilă-substrat – GB, scurt-circuit sursă-substrat – SB, scurt-circuit drenă-substrat – DB) [Nigh1990, Mao1990];
- defectele de scurt-circuit la nivel de poartă – sunt toate defectele de scurt-circuit ce pot apare între fiecare intrările, ieșirile, nodurile interne și nodurile de alimentare ale porții [Mahlstedt1992];
- defectele de scurt-circuit la nivel logic (inter-porți) – sunt toate defectele de scurt-circuit ce pot apare între nodurile externe ale porților ce formează circuitul [Thadikaran1993];
- defectele de scurt-circuit între liniile de metal – sunt toate defectele de scurt-circuit ce pot apare între traseele din straturile de metalizare;

- defectele de scurt-circuit realiste (extrase) – se utilizează informațiile disponibile despre localizarea fizică a structurilor din circuit pentru a selecta un set mai redus de defecte [Glaser1994, Chen1997].

Un defect de scurt-circuit între două noduri poate fi detectat (este activat) atunci când cele două noduri sunt aduse la valori logice complementare. Un vector de intrare care setează cele două noduri conectate de defect la valori logice complementare este un vector de test termic pentru defectul respectiv. Pe de altă parte, în testarea termică un defect trebuie doar activat și nu este necesară propagarea efectului defectului la ieșirile primare ale circuitului, datorită faptului că acest efect este sesizat la nivelul senzorului de temperatură. Astfel că metoda de testare termică are o observabilitate foarte bună.

Algoritmii de generare automată a vectorilor de test selectează câte un defect din setul de defecte specificat prin modelul ales, după care este executată o procedură recursivă pentru activarea defectului, pe baza descrierii circuitului. În cazul testării termice este important modul în care sunt selectate defectele. La fiecare pas se alege acel defect pentru care numărul de tranziții la aplicarea vectorilor de test consecutivi să fie cât mai mic posibil, deoarece trebuie reduse activitățile de comutare care generează o cantitate mai mare de căldură care poate masca căldura degajată de defect.

În [Nigh1990] sunt introduse două funcții logice  $N_i^{VDD}$  și  $N_i^{GND}$  ce reprezintă vectorii de intrare ce crează o cale de curent de la nodul  $i$  la  $VDD$  și respectiv  $GND$ . Folosind aceste funcții pentru fiecare defect de scurt-circuit  $\langle i,j \rangle$  din lista de defecte se calculează funcția de test  $T_{\langle i,j \rangle} = N_i^{VDD}N_j^{GND} + N_i^{GND}N_j^{VDD}$ . Un defect de scurt-circuit este detectat dacă funcția de test corespunzătoare este adevărată (are valoarea logică "1"). Această abordare este limitată la circuitele combinaționale iar complexitatea algoritmului depinde de funcțiile logice ce trebuie generate și rezolvate.

Deoarece testarea  $I_{DDQ}$  și testarea termică sunt metode de testare statice foarte lente, în [Mao1990] autorii au propus o metodă euristică de selecție a vectorilor de test pentru reducerea numărului lor, păstrând procentajul de acoperire a defectelor. Acest articol vizează doar defecte de scurt-circuit intra-poartă din circuitele combinaționale. În [Chakravarty1994] algoritmul a fost extins și la circuitele secvențiale.

În [Glaser1994] s-a folosit atât setul exhaustiv de defecte cât și subsetul de defecte extrase. Acest articol arată că folosirea setului exhaustiv de defecte de scurt-circuit duce la creșterea setului de vectori obținut și implicit la creșterea duratei de testare. Setul de vectori de test pentru detecția defectelor de scurt-circuit este mult mai mic decât cel pentru testarea logică, iar în anumite cazuri particulare este un subset al acestuia din urmă [Mahlstedt1992]. Și aceste articole discută circuitele combinaționale.

[Isern1993] folosește diagramele de decizie binară pentru identificarea scurt-circuitelor echivalente pentru a reduce numărul de defecte de scurt-circuit. După faza de reducere se folosește un generator de vectori de test pentru defectele de blocare ce se aplică pe un circuit modificat față de cel inițial, în care se introduce câte o poartă XOR ce are ca intrări cele două noduri ce formează un defect, iar ieșirea porții se consideră ca ieșire observabilă. Această metodă este limitată la defectele de scurt-circuit la nivel logic din circuitele combinaționale.

Algoritmul de generare a vectorilor de test pentru defectele de scurt-circuit propus de [Bollinger1994] folosește o descriere la nivel de tranzistor pentru circuitele combinaționale, ceea ce pune probleme circuitelor de dimensiune foarte mare (peste 100000 de tranzistoare). Un defect de scurt-circuit  $\langle i, j \rangle$  este activat dacă există o cale de conducție de la  $VDD$  la  $GND$  prin cele două noduri  $i$  și  $j$ . Astfel că acest articol parcurge toate căile de la  $VDD$  la  $GND$  pentru a găsi condițiile de activare pentru fiecare defect.

În [Lee1996] este prezentat un algoritm genetic de generare a vectorilor de test pentru circuitele secvențiale. Secvențele de vectori de test formează cromozomii algoritmului iar

funcția fitness asociată fiecărui cromozom ce trebuie maximizată este numărul de defecte detectate de o astfel de secvență. Pentru fiecare cromozom (secvență de vectori) se simulează circuitul la nivel de tranzistor pentru a obține valoarea funcției de fitness. Prin mecanismele specifice algoritmilor genetici populația de cromozomi evoluează până se atinge o anumită condiție de terminare. Și această metodă este lentă deoarece s-a ales modelul de defecte de scurt-circuit exhaustive iar simularea se face la nivel de tranzistor.

O altă variantă îmbunătățită a utilizării unui algoritm genetic pentru generarea vectorilor de test pentru circuitele secvențiale este prezentată în [Reddy1995]. În acest algoritm s-a introdus și o fază de reducere a setului de defecte de scurt-circuit prin eliminarea redundanțelor. În plus, după faza euristică se mai folosește un pas deterministic în care se încearcă și detecția defectelor ce au rămas neacoperite. O variantă asemănătoare a algoritmului pentru circuitele combinaționale este prezentată în [Thadikaran1996].

Structura generală a unui algoritm de generare a vectorilor de test pentru testarea termică este prezentată în figura 2.68. Un astfel de algoritm poate implementa toate fazele din figura 2.68 sau doar o parte din ele, pentru fiecare fază existând diferite variante de implementare. Un exemplu de aplicație folosită la generarea vectorilor de test pentru testarea  $I_{DDQ}$  a circuitelor combinaționale și secvențiale, ce implementează toate fazele din figura 2.68 este prezentat în [Mahlstedt1995].

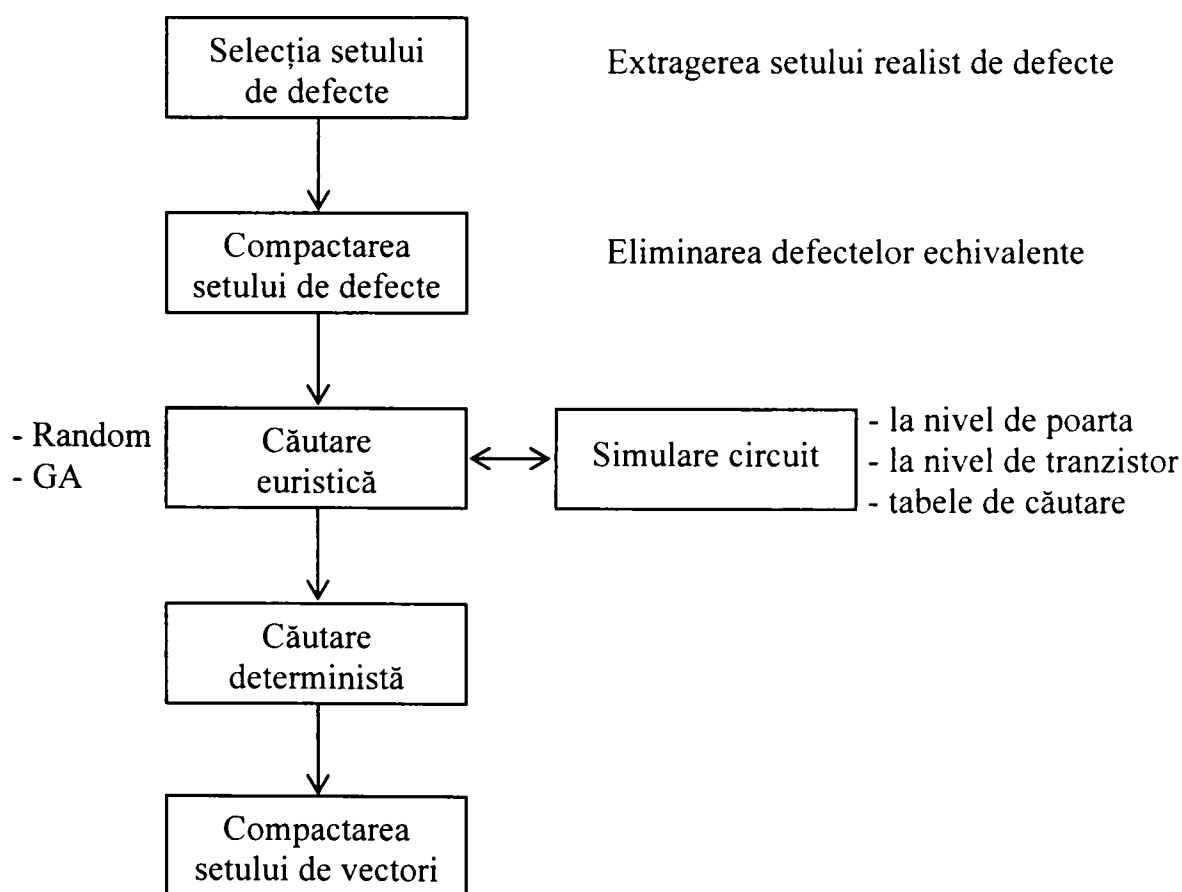


Figura 2.68 – Schema generală a unui algoritm de generare a vectorilor de test pentru testarea termică

În [Chen1997] este prezentat un alt generator pentru vectorii de test pentru testarea  $I_{DDQ}$  bazat pe o strategie hibridă de testare: logică +  $I_{DDQ}$ , cu scopul de a obține o secvență de vectori de test  $I_{DDQ}$  cât mai redusă ca dimensiune. În faza euristică a generatorului este folosit un algoritm genetic iar modulul de simulare al circuitului folosește simularea la nivel de poartă logică și pe tabele de căutare pentru nodurile interne ale unei porți.

Defectele reale de scurt-circuit din circuitele VLSI au și o anumită rezistență a defectului, astfel că nu pot fi toate modelate ca defecte ideale de scurt-circuit. Din păcate, defectele rezistive (din care fac parte și defectele de scurt-circuit poartă-oxid) au o pondere destul de mare din totalul defectelor din circuitele VLSI [Hawkins1994], iar pe măsură ce dimensiunea tranzistoarelor scade, aceste defecte influențează circuitele într-o măsură și mai mare [Attarha2000]. Doar un număr mic de articole tratează defectele de scurt-circuit rezistive și există la fel de puține aplicații pentru acest tip de defecte [Engelke2003].

Datorită neglijării rezistenței defectelor de scurt-circuit, datele prezentate în [Chen1997] nu sunt reale. În acest articol se prezintă o strategie hibridă de generare a vectorilor de test atât pentru testarea logică cât și pentru testarea  $I_{DDQ}$ . Ideea este foarte bună în sine doar că datele prezentate nu sunt reale pentru cazul defectelor de scurt-circuit rezistiv, așa cum s-a considerat în articol. Rezultatele articolului sunt valabile doar pentru defectele de scurt-circuit de rezistență zero. În algoritmul propus în articol se folosește un generator de teste existent pentru testarea logică (HITEC). Cu setul de teste obținut se simulează circuitul pentru a detecta toate defectele de scurt-circuit activate de acesta. După care se generează setul de vectori de test pentru testarea  $I_{DDQ}$  doar pentru defectele de scurt-circuit neacoperite de testarea logică. Prin această metodă se obține într-adevăr un set de vectori de test pentru testarea  $I_{DDQ}$  mult mai scurt decât dacă s-ar fi generat pentru toate defectele de scurt-circuit extrase, însă nu toate defectele rezistive sunt detectate de testarea logică. Astfel că valorile calculate în articol nu sunt reale, deoarece există o submulțime de defecte de scurt-circuit rezistiv ce nu sunt detectate de testarea logică însă sunt detectate de testarea  $I_{DDQ}$  sau testarea termică. O soluție pentru determinarea corectă a raportului de acoperire a defectelor rezistive de scurt-circuit de către setul de vectori de test propus.

## 2.8 Simularea defectelor de scurt-circuit rezistiv

În cazul defectelor de scurt-circuit rezistiv apare o necunoscută în plus: valoarea rezistenței defectului. Rezistența defectului depinde de tipul defectului și de tipul particulei ce produce defectul (dimensiune, formă, conductivitate, locație, etc.) [Engelke2003]. Pentru un același vector de test un defect poate fi detectat dacă valoarea rezistenței are anumite valori sau nu va fi detectat în caz contrar. Pentru modelarea defectelor rezistive Renovell a introdus conceptul de interval analog de detectabilitate (ADI – Analogue Detectability Interval) pentru un defect [Renovell1995]. Un interval  $[R_{\min}, R_{\max}]$  este definit pentru fiecare defect și pentru fiecare vector de test, ca fiind intervalul în care trebuie să se încadreze defectul pentru a fi detectat de respectivul vector de test.

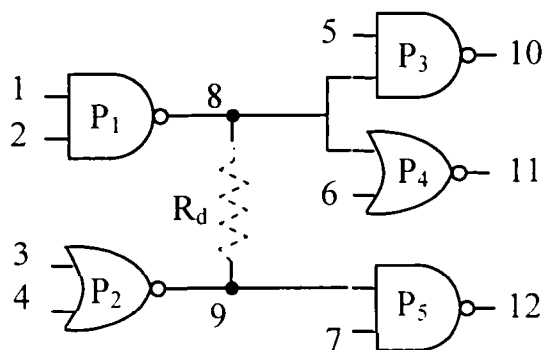


Figura 2.69 – Exemplu circuit cu defect de scurt-circuit rezistiv

Intervalele de detectabilitate pentru un defect de scurt-circuit rezistiv sunt diferite pentru testarea logică, testarea termică sau testarea  $I_{DDQ}$ . Procentajul de acoperire a defectelor de către fiecare tip de testare este dependent de aceste intervale de detectabilitate. Testarea logică a defectelor de scurt-circuit presupune aducerea nodurilor cuplate de defect la valori logice diferite și propagarea acestor valori la ieșirile observabile ale circuitului. În cazul defectelor de scurt-circuit rezistiv apare o cădere de tensiune pe rezistența defectului, astfel că porțile care au ca și intrări nodurile conectate de defect se poate să nu sesizeze defectul.

Detecția prin testarea logică a defectelor de scurt-circuit rezistiv depinde de mai mulți factori:

- valorile logice prezente la intrările porților ce au ca ieșiri nodurile defectului;
- structura tranzistorilor ce formează cele două porți ce au ca ieșiri nodurile defectului;
- valoarea de tensiunilor de prag ale porților ce au ca intrări nodurile defectului;
- valorile logice prezente la celelalte intrări ale porților ce au ca intrări nodurile defectului.

Pentru defectul din figura 2.69, la aplicarea vectorului  $T_1 = \langle 1, 1, 0, 0, 1, 0, 1 \rangle$  la intrările 1, 2, 3, 4, 5, 6 și 7, posibilitatea de detecție a defectului prin testarea logică funcție de valoarea rezistenței defectului este prezentată în figura 2.70. În graficul din figura 2.70 este prezentată variația ieșirilor observabile ale circuitului în funcție de rezistența defectului. Din acest grafic se poate observa că testarea logică este limitată doar la anumite valori ale rezistenței defectului:  $[0, 1.25 \text{ k}\Omega]$  la ieșirea 11 și  $[0, 2.3 \text{ k}\Omega]$  la ieșirea 10.

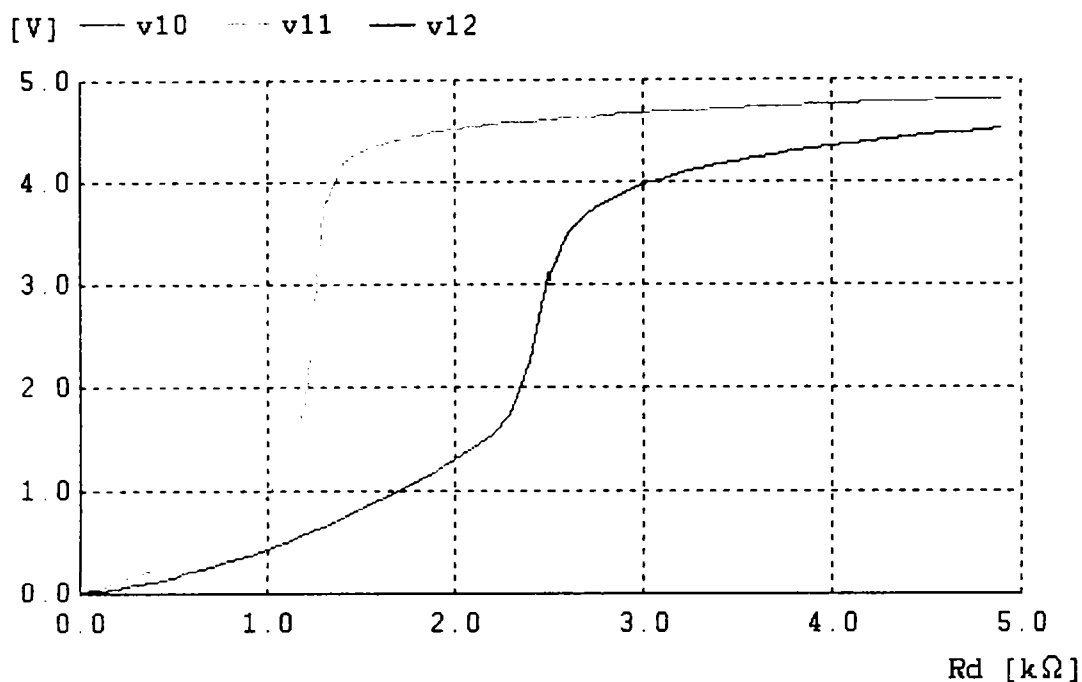


Figura 2.70 – Variația ieșirilor observabile în funcție de rezistența defectului pentru  $T_1$

Pentru un al doilea vector de test  $T_2 = \langle 0,0,1,1,1,0,1 \rangle$ , pentru defectul din figura 2.69, variația valorilor de tensiune la ieșirile observabile ale circuitului este prezentată în figura 2.71. În acest caz valoarea maximă a rezistenței defectului ce poate fi detectată de testarea logică este sub 1 kΩ.

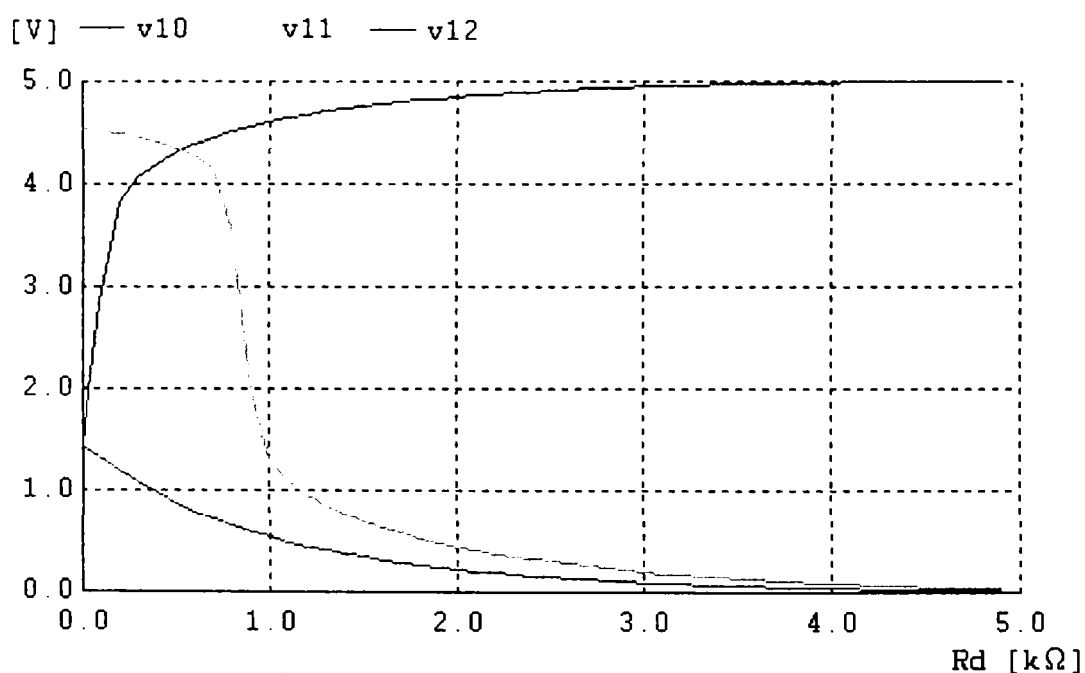


Figura 2.71 - Variația ieșirilor observabile în funcție de rezistența defectului pentru  $T_2$

Al treilea vector de test ( $T_3 = \langle 1,0,1,1,1,0,1 \rangle$ ) aplicat la intrările circuitului produce la ieșirile observabile ale circuitului variația de tensiune din figura 2.72. În acest caz, la ieșirea 11 se pot detecta defectele de scurt-circuit cu o valoare a rezistenței de până la 2.6 kΩ.

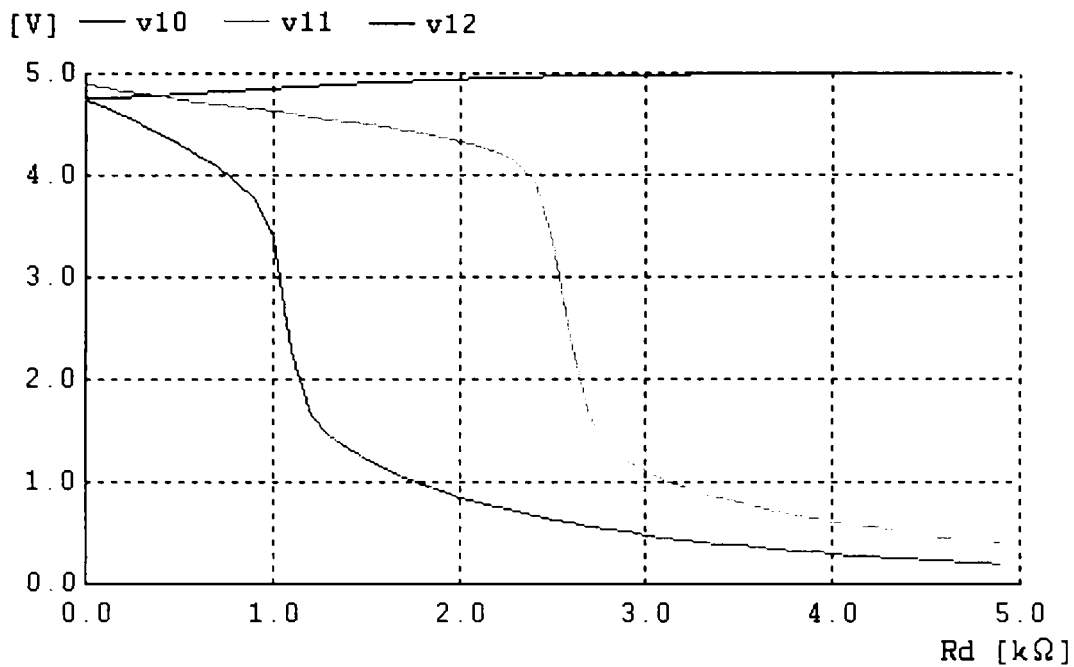


Figura 2.72 – Variația ieșirilor observabile în funcție de rezistența defectului pentru  $T_3$

Puterea disipată de circuitul fără defecte la aplicarea vectorului de intrare  $T_1$  este de 1.8nW iar pentru  $T_5$  este 1.97nW. Pentru ceilalți vectori de test (vezi tabelul 2.8) valoarea puterii disipate este mai mică decât aceste valori. Considerând un nivel de prag pentru puterea disipată cu șase ordine de mărime mai mare decât valoarea maximă obținută pentru circuitul fără defecte se obține 2mW. Toate circuitele care disipă o putere mai mare de valoarea stabilită pentru nivelul de prag se consideră defecte. Astfel că la aplicarea vectorului  $T_1$ , se pot detecta scurt-circuitele de tipul celui din figura 2.69, cu o valoare a rezistenței de până la 13kΩ.

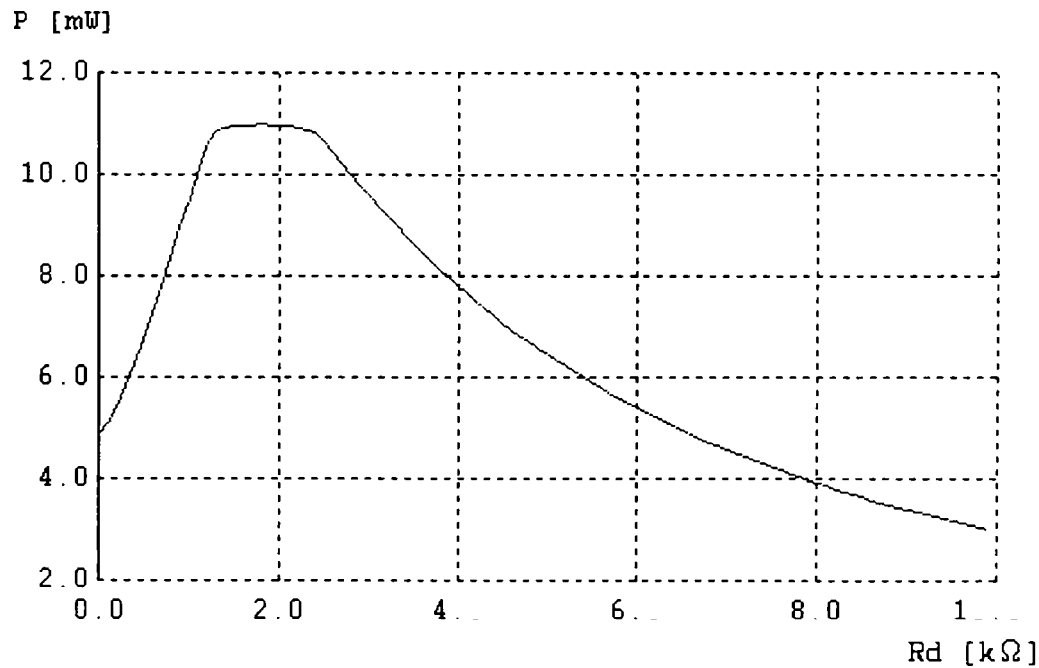


Figura 2.73 – Variația puterii disipate în funcție de rezistența defectului pentru  $T_1$

La aplicarea vectorului de test  $T_3$ , variația puterii disipate funcție de rezistența defectului este prezentată în figura 2.74. Rezistența maximă detectată este de aproximativ 11kΩ.

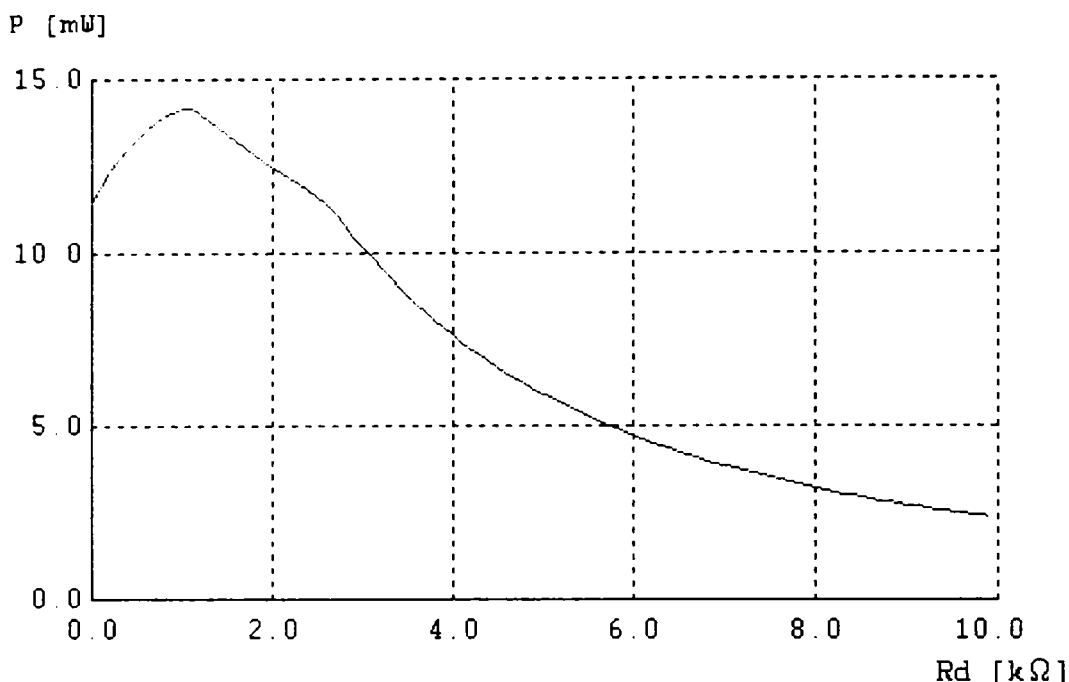


Figura 2.74 – Variația puterii disipate în funcție de rezistența defectului pentru  $T_3$

În tabelul 2.8 s-au prezentat valorile obținute pentru intervalele de detectabilitate ale defectului din figura 2.69, pentru testarea logică, termică și  $I_{DDQ}$ .

Vector de test	Testare logică	Testare termică	Testare $I_{DDQ}$
$T_1 = \langle 1, 1, 0, 0, 1, 0, 1 \rangle$	[0-2.3 kΩ] (v10)	[0-13.1 kΩ]	[0-47.3 kΩ]
$T_2 = \langle 0, 0, 1, 1, 1, 0, 1 \rangle$	[0-0.8 kΩ] (v11)	[0-11.2 kΩ]	[0-48.8 kΩ]
$T_3 = \langle 1, 0, 1, 1, 1, 0, 1 \rangle$	[0-2.5 kΩ] (v11)	[0-11.3 kΩ]	[0-48.1 kΩ]
$T_4 = \langle 0, 0, 1, 0, 1, 0, 1 \rangle$	[0-1.2 kΩ] (v12)	[0-10.7 kΩ]	[0-48.2 kΩ]
$T_5 = \langle 1, 0, 1, 0, 1, 0, 1 \rangle$	[0-1.7 kΩ] (v11)	[0-10.7 kΩ]	[0-47.5 kΩ]

Tabelul 2.8 – Intervalele de detectabilitate pentru circuitul din figura 2.69

Pentru determinarea valorilor din tabelul 2.8 s-a folosit metoda nivelului de prag atât pentru testarea termică cât și pentru testarea  $I_{DDQ}$ . Nivelul de prag stabilit pentru testarea termică a fost ales 2mW iar pentru testarea  $I_{DDQ}$ , 100μA. Valorile au fost obținute simulând în SPICE circuitul din figura 2.69.

Din exemplul prezentat se poate observa faptul că intervalele de detectabilitate pentru defectele de scurt-circuit rezistiv depind în primul rând de vectorul de test aplicat. În al doilea rând, detecția unui defect de scurt-circuit rezistiv depinde și de modul în care porțile logice ce au ca intrări nodurile defectului interpretează valorile de tensiune stabilite datorită rezistenței defectului. Pentru valori mai mari ale rezistenței defectului porțile ce au ca intrări nodurile defectului vor interpreta corect din punct de vedere logic valorile tensiunilor nodurilor defectului, astfel că defectul nu va putea fi detectat prin testarea logică. Testarea termică și testarea  $I_{DDQ}$  au o acoperire mult mai bună a defectelor de scurt-circuit rezistiv.

În continuare voi prezenta o modalitate de simulare pentru defectele de scurt-circuit rezistiv și modul în care rezistența defectelor de scurt-circuit influențează factorul de acoperire a defectelor pentru testarea logică și testarea termică.

Simularea defectelor de scurt-circuit rezistiv este mai dificilă decât simularea defectelor de blocare și a celor de scurt-circuit de rezistență 0. Dificultatea simulării se datorează parametrului suplimentar ce trebuie luat în calcul: rezistența defectului. Problema constă în faptul că nu se mai lucrează cu niveluri logice bine determinate. Prezența unei rezistențe diferite de 0 pentru un defect de scurt-circuit aduce nodurile conectate de defect la



niște valori de tensiune diferite de cele pentru valorile logice “0” și “1”. Simularea unui defect de scurt-circuit rezistiv trebuie să stabilească modul în care aceste valori de tensiune sunt interpretate de porțile ce au ca intrări aceste noduri. În funcție de nivelul tensiunii de prag pentru fiecare poartă logică, valorile de tensiune de la intrările porții sunt considerate “1” dacă depășesc nivelul tensiunii de prag și “0” dacă sunt sub această valoare. Nivelurile tensiunilor de prag sunt de obicei diferite nu numai de la un tip de poartă logică la alta ci chiar porțile de același tip pot avea implementări diferite. Chiar mai mult, fiecare intrare a unei porți logice poate avea un nivel de prag diferit în funcție de structura tranzistoarelor ce formează poarta.

Din acest motiv simularea defectelor de scurt-circuit rezistiv necesită o simulare a circuitului la nivel electric. Pentru simularea defectelor de scurt-circuit rezistiv există mai multe variante de soluții:

- implementarea simulatorului pe baza unor modele electrice simplificate pentru porțile logice și defectele de scurt-circuit. Variante ale acestei soluții au fost propuse în [Li2003] și [Engelke2003];
- implementarea simulatorului pe baza unor tabele de căutare generate de un program de tip SPICE. Câte un exemplu pentru acest tip de soluție este prezentat în [Dessai1998] și [Lee2000];
- implementarea simulatorului prin utilizarea unui simulator extern de tip SPICE.

În [Engelke2003] se folosește un set de ecuații electrice pentru determinarea rezistenței critice. Rezistența critică este valoarea maximă a rezistenței unui defect pentru care acesta mai poate fi detectat de către porțile ce succed nodurile defectului. Există o funcție de acest tip pentru porțile ce au ca și intrare nodul defectului setat la “1” și una pentru porțile ce au ca intrare nodul defectului setat la “0”. Dacă rezistența defectului este mai mică decât valoarea calculată pentru rezistența critică corespunzătoare nodului setat pe “1” atunci defectul va fi detectat de poarta ce ar ca intrare acest nod și la fel pentru celălalt caz. Cele două funcții de descriu din punct de vedere electric comportamentul defectelor de scurt-circuit rezistiv depind de tensiunea de alimentare, capacitatea stratului de oxid  $C_{ox}$ , dimensiunea tranzistoarelor  $W_p/L_p$  și  $W_n/L_n$ , mobilitatea purtătorilor de sarcină  $\mu_p$  și  $\mu_n$ , tensiunile de prag  $V_{tp0}$  și  $V_{tn0}$ , potențialul electrostatic al substratului și tensiunea de prag pentru intrările porților comandate de nodurile defectului ( $V_{ti1}$  și  $V_{ti2}$ ):

$$R_{sh-p} = K_1 \cdot \left( V_{tn} - V_{DD} + V_{tn0} + \sqrt{(V_{DD} - V_{tn0})^2 - \frac{2}{K_1 C_{ox} \mu_n W_n / L_n}} \right)$$

$$R_{sh-n} = K_2 \cdot \left( V_{tp0} - V_{ti2} + \sqrt{(V_{DD} - V_{tp0})^2 - \frac{2}{K_2 C_{ox} \mu_p W_p / L_p}} \right)$$

unde  $K_1$  și  $K_2$  sunt niște coeficienți ce depind și ei la rândul lor de mai mulți parametrii.

Prima soluție este aplicată și în [Li2003], prin aproximarea electrică a circuitului printr-un model bazat pe rețele RC. Acest model presupune înlocuirea porților ce au ca noduri de ieșire nodurile defectului cu comutatoare și rezistențe iar porțile ce au ca intrări nodurile defectului sunt înlocuite de buffer-e. Pentru circuitul din figura 2.69, modelul simplificat propus în [Li2003], este prezentat în figura 2.75. Parametrii RC ai circuitului din figura 2.75 conțin rezistențele rețelelor de tranzistoarea PMOS și NMOS din porțile  $P_1$  și  $P_2$  (vezi figura 2.69), rezistențele parazite ale interconexiunilor ( $R_1$ ,  $R_2$ ,  $R_3$  și  $R_4$ ), rezistența defectului  $R_d$ , capacitățile parazite de la intrările porților ce au ca intrări nodurile defectului și tensiunile de prag ale intrărilor acestor porți. În continuare, în funcție de valorile logice stabilite la nodurile defectului se determină rezistențele critice, pentru fiecare din porțile ce au ca intrări nodurile defectului. Dacă rezistența defectului este mai mică decât rezistența critică pentru o poartă ce succede nodurile defectului, atunci defectul poate fi detectat ca un defect logic de către poarta în cauză.

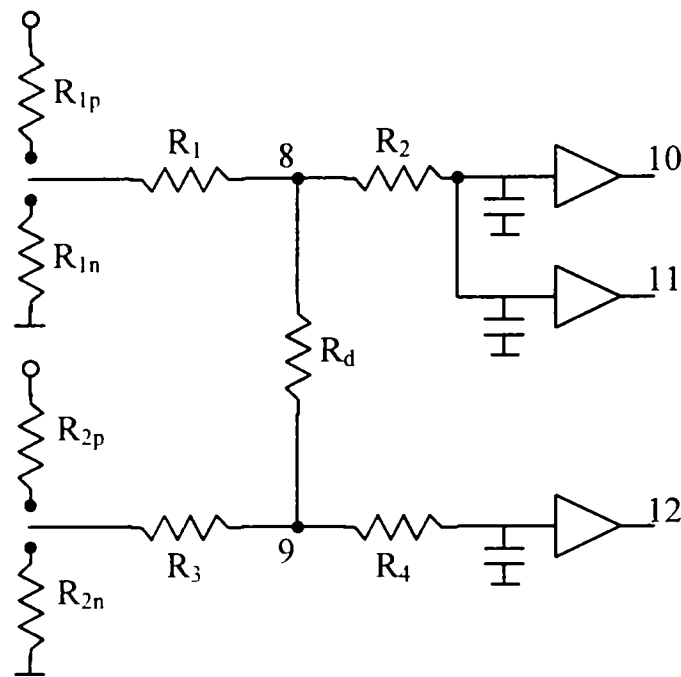


Figura 2.75 – Modelul RC pentru circuitul din figura 2.69

Fiecare din rezistențele din figura 2.75 depind de implementarea în siliciu a circuitului, astfel că pentru determinarea lor trebuie apelat tot la modelul și simularea în SPICE a circuitului.

În [Dessai1998] pentru simularea defectelor de scurt-circuit rezistiv se folosesc tabele de căutare. Aici defectele sunt clasificate în mai multe categorii:

- scurt-circuite între două intrări primare (1);
- scurt-circuite între o intrare primară și un oricare alt nod (2);
- scurt-circuite între două noduri interne ce sunt succedate de porți diferite (3);
- scurt-circuite între două noduri interne ce sunt succedate de intrările unei aceleiași porți (4);
- scurt-circuite între o ieșire observabilă și un oricare alt nod (5).

Tabelele de căutare sunt generate folosind SPICE și pe baza lor se estimează dacă un defect de scurt-circuit cu o anumită rezistență poate fi detectat sau nu ca un defect logic. Aceste tabele de căutare conțin combinații de vectori de intrare pentru fiecare defect din fiecare categorie împreună cu intervalele de detectabilitate pentru respectivele defecte. În procesul de simulare se aplică vectorii de test și se compară valorile logice obținute din simulare cu cazurile de detectabilitate din tabelele de căutare și sunt detectate acele defecte ce au rezistențele în intervalele de detectabilitate determinate din tabelele de căutare pentru o anumită combinație de intrare. Numărul tabelelor de căutare pentru un circuit VLSI sau pentru o clasă de circuite (cum ar fi circuitele ISCAS85) este foarte mare (de ordinul miilor). Numărul mare de tabele se datorează numărului mare de combinații între diferitele tipuri de porți logice ce sunt folosite într-un circuit VLSI. În [Dessai1998] s-a redus numărul de tabele (la aproximativ 200) făcând mai multe presupuneri: toate intrările porților și toate porțile de același tip au același nivel de prag, au fost luate în considerare doar porțile cu maxim 5 intrări și au fost luate în calcul doar defectele cu probabilitatea cea mai mare de apariție.

Fiecare din soluțiile prezentate are avantajele și dezavantajele ei. Primele două soluții au ca avantaj viteza, în timp ce a treia soluție este consumatoare de timp. Ca dezavantaje pentru primele soluții sunt aproximările și presupunerile făcute pentru reducerea timpului de calcul și a consumului de memorie. Un alt dezavantaj al primelor două soluții, important în opinia mea, constă în folosirea de modele și/sau implementări diferite pentru același tip de operație: simularea electrică a circuitului sau a unor părți din circuit. În procesul de proiectare

a unui circuit integrat o fază folosită de toți producătorii de circuite integrate constă în simularea și verificarea din punct de vedere electric a circuitului. În această fază se folosește o unealtă de simulare bazată pe modelul SPICE. Astfel că fiecare circuit integrat are o descriere în SPICE pe baza căreia se fac verificările și mai apoi se plasează poștile pe pastila de siliciu. Tot pe baza modelului SPICE au fost generate funcțiile de trecere sau tabelele de căutare utilizate de primele două soluții de simulare. Problema acestor abordări se datorează implementărilor diferite pentru simularea electrică: una pentru SPICE și celalaltă pentru simularea defectelor. Din acest motiv la fiecare modificare în modelul SPICE al circuitului, modificare a unei porți sau chiar al unui parametru, este necesară actualizarea și a modelului pentru simularea defectelor. Această actualizare presupune pentru soluția a doua, regenerarea tabelelor de căutare, care poate lua destul de mult timp.

Din acest motiv doresc să propun o soluție ce folosește direct modelul SPICE al circuitului și un simulator de tip SPICE, soluția ce se încadrează în clasa a treia de soluții. Din această clasă de soluții nu s-a găsit nici o implementare datorită timpului mare de simulare. Timpul mare de simulare se datorează faptului că simulatoarele SPICE simulează evoluția electrică a fiecărui nod al circuitului, în timp ce într-un simulator pentru defectele de scurt-circuit rezistiv este necesară o simulare electrică doar la nivelul celor două noduri ce formează defectul. Pentru reducerea timpului de simulare s-a propus și implementat o metodă de simulare logico-electrică. Această soluție se bazează pe simularea logică în cele mai multe cazuri și numai pentru cazurile care necesită simulare electrică se recurge la simularea în SPICE. Pentru simularea în SPICE s-a recurs la două variante: prima din ele constă în simularea întregului circuit iar în a doua variantă este simulat în SPICE doar a unui subcircuit ce implică doar nodurile defectului. Soluția este prezentată în continuare în timp ce implementarea ei și rezultatele sunt prezentate în secțiunea următoare.

Considerând un set de vectori de test  $T = \langle T_1, T_2, \dots, T_k \rangle$  se notează cu  $S_i(d)$  intervalul în care trebuie să se încadreze rezistența defectului de scurt-circuit  $d$  pentru a fi detectată de vectorul de test  $T_i$ .  $S(d)$  va fi domeniul de rezistențe pentru defectul  $d$  în care acesta este detectat de întreg setul de vectori  $T$ :

$$S(d) = \bigcup_{i=1}^k S_i(d)$$

Considerăm  $\sigma(r)$  ca fiind funcția densitate a probabilității rezistenței defectului  $r$ . În [Renovell1995] se consideră o distribuție normală pentru funcția  $\sigma(r)$ , în [Li2003] se consideră o funcție de distribuție uniformă iar în [Dessai1998]  $\sigma(r)$  este o distribuție geometrică:

$$\sigma(r) = 1 - (1 - p)^r$$

unde  $p$  este determinat empiric (în [Dessai1998] este 0.00258).

În [Engelke2003] sunt prezentate mai multe variante de estimare pentru factorul de acoperire a defectelor de scurt-circuit rezistiv de către un set de vectori de test. Valoarea pesimistă a factorului de acoperire pentru un defect este dată de raportul dintre intervalele de detectabilitate pentru defect și întregul interval în care se încadrează rezistența acestuia  $[0, \infty)$ .

$$PFC(d) = \frac{\int_{S(d)} \rho(r) dr}{\int_0^{\infty} \rho(r) dr} \cdot 100$$

Pentru tot setul de defecte ales acoperirea defectelor pesimistă este suma PFC pentru fiecare defect:

$$PFC = \frac{1}{N} \cdot \sum_{i=1}^N PFC(d_i)$$

Valoarea realistă a factorului de acoperire a defectelor este raportul dintre intervalul de detectabilitate al setului de vectori de test și intervalul de detectabilitate al setului exhaustiv de combinații de intrare. Această valoare este greu de calculat deoarece setul de combinații binare de la intrările circuitului este foarte mare, astfel că se pretează doar la circuitele cu un număr mic de intrări.

$$GFC(d) = \frac{\int_{i(d)} \rho(r) dr}{\int_{ii(d)} \rho(r) dr} \cdot 100$$

Dacă  $R_{max}$  este rezistența critică maximă la oricare din porțile ce succed defectul, atunci defectele ce au valoarea rezistenței între 0 și  $R_{max}$  sunt recunoscute de cel puțin una din porțile ce au ca intrări nodurile defectului. Această formulă de calcul pentru acoperirea defectelor se potrivește pentru testarea termică sau testarea  $I_{DDQ}$  deoarece ia în calcul doar activarea defectului nu și propagarea efectului defectului la ieșirile observabile ale circuitului. Această definiție este mai exactă decât PFC și presupune un timp de calcul mai redus decât GFC.

$$EFC(d) = \frac{\int_{i(d)} \rho(r) dr}{\int_0^{R_{max}} \rho(r) dr} \cdot 100$$

Relația între aceste definiții pentru factorul de acoperire a defectelor de scurt-circuit rezistiv este prezentată mai jos, unde OFC este considerat ca fiind factorul de acoperire a defectelor de scurt-circuit de rezistență 0 (valoarea optimistă a acoperirii a defectelor):

$$PFC \leq EFC \leq GFC \leq OFC$$

Calculul acestor valori pentru un circuit integrat presupune utilizarea unei metode de simulare logico-electrică. Prin această metodă se pot simula defectele de scurt-circuit rezistiv atât pentru testarea logică cât și pentru testarea termică și  $I_{DDQ}$ . Algoritmul de simulare logico-electrică este prezentat în figura 2.76. Rezistența  $R_{min}$  este rezistența minimă a oricărui defect de scurt-circuit pentru care trebuie efectuată o simulare electrică. Pentru defectele ce au rezistențe cu valori mai mici decât  $R_{min}$ , se poate face o simulare logică.  $R_{max}$  este rezistența maximă a oricărui defect pentru care trebuie făcută o simulare electrică. Defectele ce au o valoare a rezistenței peste  $R_{max}$  nu sunt detectabile.  $R_{min}$  și  $R_{max}$  au valori diferite pentru testarea logică, testarea  $I_{DDQ}$  și testarea termică.

Simularea electrică a defectului presupune extragerea subcircuitului ce conține porțile ce au ca noduri de intrare sau ieșire nodurile defectului și simularea acestuia în SPICE. Opțiunile de configurare a algoritmului sunt parametrizate în aplicație:  $R_{min}$  și  $R_{max}$  (se pot seta astfel încât să se facă o simulare electrică sau logică pentru întregul set de defecte), selecția vectorilor de test se poate face aleator sau secvențial, se poate renunța la testarea anumitor defecte dacă au un număr prea mare de simulări, ș.a.

Descrierea circuitului se citește fie dintr-un fișier specific aplicațiilor de tipul SPICE (\*.cir) fie dintr-un fișier specific setului de circuite benchmark ISCAS95 (\*.isc). Simularea în SPICE se face pe baza descrierii la nivel de tranzistor a circuitului sau a subcircuitului format din nodurile defectului. Pentru faza de simulare se generează din modelul de intrare un fișier SPICE nou ce conține vectorul sau secvența de intrare aplicată circuitului, descrierea defectului și parametrii circuitului ce trebuie simulați. Parametrii de ieșire ai unei simulări SPICE pot fi valorile logice ale anumitor noduri ale circuitului sau ale ieșirilor observabile ale circuitului, curentul  $I_{DD}$ , puterea disipată de circuit sau temperatura estimată a circuitului. Acești parametri sunt preluați mai apoi în aplicație, pe baza lor se realizează mai departe

simularea defectului și apoi se iau decizii în ceea ce privește posibilitatea de detecție a defectului de vectorul de test aplicat circuitului.

Simularea logică este implementată în cadrul aplicației prin implementarea funcțiilor logice corespunzătoare fiecărui tip de poartă logică. Pe baza rezultatului simulării se setează starea defectului pe “detectat”, “nedetectat” sau “nedetectabil”. Decizia de detectabilitate pentru un defect se face în funcție de valoarea curentului  $I_{DDQ}$ , pentru simularea  $I_{DDQ}$ , puterea disipată, pentru testarea termică și vectorul logic de ieșire, pentru testarea logică. În final, după aplicarea tuturor vectorilor de test, se calculează factorul de acoperire a defectelor.

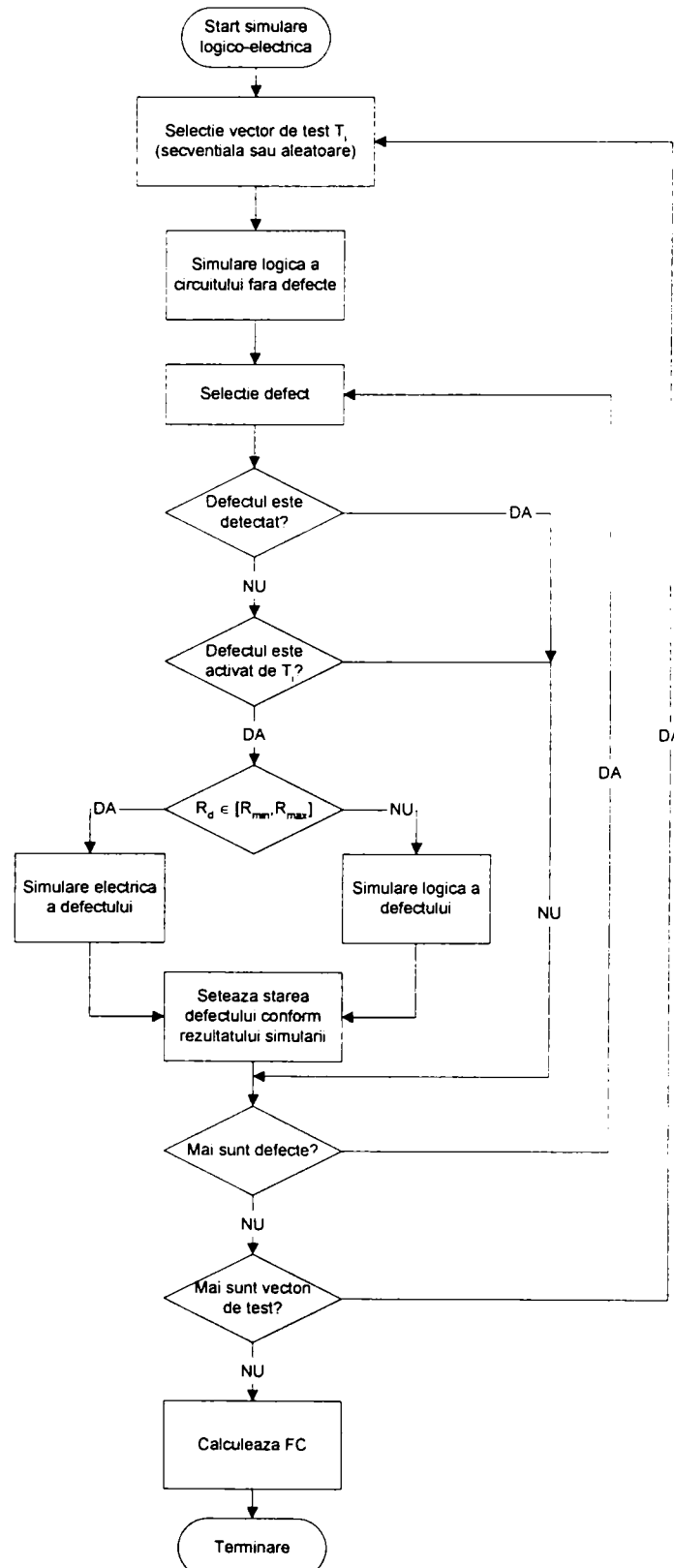


Figura 2.76 – Algoritm de simulare logico-electrică

## 2.9 Aplicația de studiu și simulare a defectelor de scurt-circuit rezistiv

Pentru efectuarea testelor din acest capitol a fost proiectată și implementată o aplicație de simulare a circuitelor logice combinaționale. Interfața principală a aplicației este prezentată în figura 2.77. Aplicația poate încărca descrierea circuitului din fișiere în formatul ISCAS95 sau SPICE. Setul de vectori de test și setul de defecte pot fi încărcate din fișiere sau se pot genera în cadrul aplicației.

### 2.9.1 Selecția defectelor și a vectorilor de test

Defectele generate de aplicație pot fi:

- toate defectele de blocare;
- toate defectele de scurt-circuit externe;
- toate defectele de scurt-circuit între nodurile învecinate;
- un număr selectabil de defecte generate aleator.

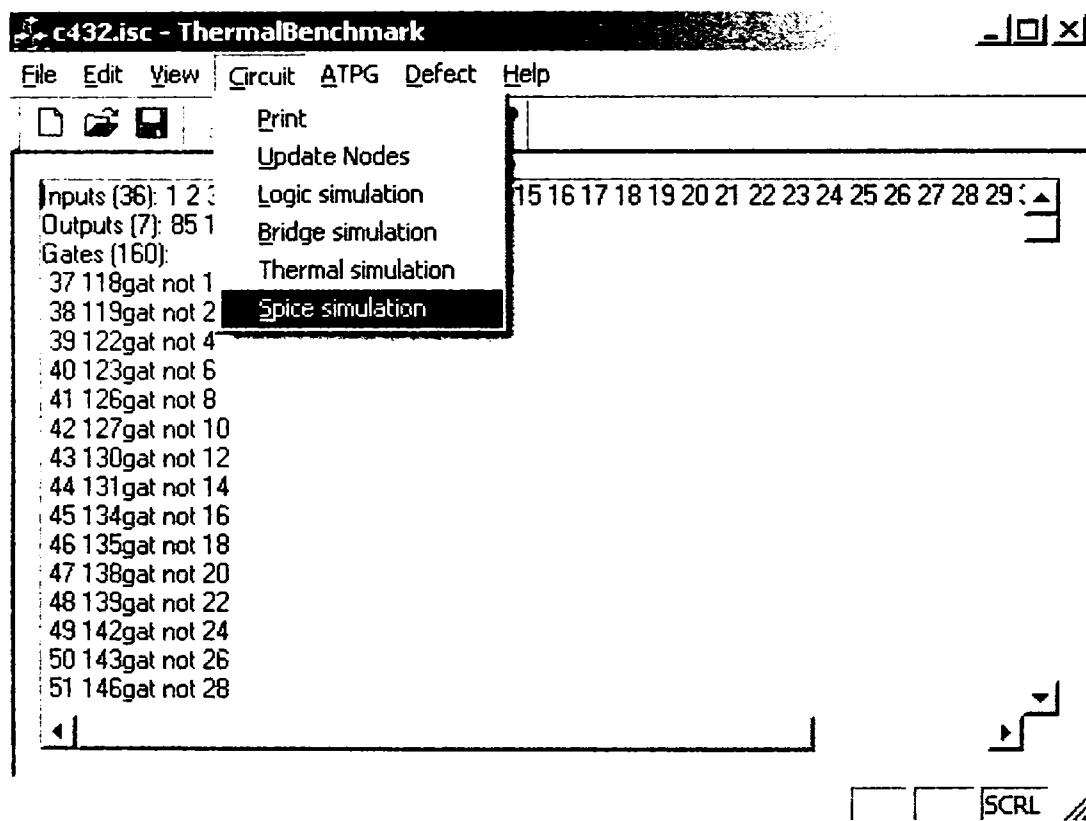


Figura 2.77

Numărul total de defecte de scurt-circuit ce pot apare într-un circuit integrat VLSI este foarte mare ( $N(N-1)/2$  unde  $N$  este numărul de noduri). Procesul de simulare a întregului set de defecte pentru calculul acoperirii defectelor de un set de vectori de test este consumatoare de timp datorită acestui număr mare de defecte posibile. Pentru estimarea într-un interval de timp rezonabil a factorului de acoperire a defectelor de către un set de vectori de test se recurge la selecția unui subset de defecte și acoperirea defectelor se estimează pe baza acestui subset. De obicei selecția subsetului se face aleator și duce la o valoare foarte apropiată de valoarea reală a acoperirii defectelor. Acest lucru se observă și în tabelul 2.9 unde este prezentat factorul de acoperire a defectelor pentru circuitul c432 pentru seturi diferite de defecte generate aleator. În acest tabel sunt prezentate valorile factorilor de acoperire a

defectelor pentru un set de 91 vectori de test generați folosind algoritmul PODEM. Acoperirea defectelor s-a calculat pentru testarea logică și testarea termică.

Defecte		Testare logică		Testare termică	
Subset	Număr	FC	Durata	FC	Durata
Aleator	300	60.00 %	0:20	100 %	0:00
Blocare	392	99.23 %	0:09	100 %	0:00
Aleator	1000	62.90 %	1:10	100 %	0:00
Extrase	1357	66.47 %	1:00	99.48 %	0:00
Aleator	3000	62.87 %	3:30	99.97 %	0:00
Aleator	10000	62.66 %	11:37	99.99 %	0:01
Exhaustiv	19503	62.52 %	25:57	99.96 %	0:04

Tabelul 2.9

În testele efectuate, s-au încercat mai multe variante de generare a unui subset de defecte pentru care să se poată estima factorul de acoperire a defectelor de către o metodă de testare. Tot pentru circuitul c432, s-au generat mai multe subseturi de 10000 defecte pentru care, mai apoi s-a calculat factorul de acoperire a defectelor de testarea logică și testarea termică. În tabelul 2.10 sunt prezentate rezultatele obținute. Prima din liniile tabelului conține valorile FC pentru un subset de 10000 defecte generate aleator. Pentru a minimiza sau maximiza FC s-a considerat că probabilitatea ca un defect de scurt-circuit să nu poată fi detectat de testarea logică este mai mare pentru defectele ce implică nodurile ce au o controlabilitate mică și observabilitate mică. În ceea ce privește testarea termică sau testarea pe bază de curent probabilitatea ca un defect de scurt-circuit să nu poată fi detectat crește cu scăderea controlabilității nodurilor ce formează defectul.

Defecte		Testare logică		Testare termică	
Subset	Număr	FC	Durata	FC	Durata
Aleator	10000	62.66 %	11:37	99.99 %	0:01
Minim	10000	61.68 %	11:54	97.95 %	0:01
Maxim	10000	87.88 %	5:56	100 %	0:01
Genetic	10000	61.85 %	12:02	97.99 %	0:01
Exhaustiv	19503	62.53 %	25:57	99.96 %	0:04

Tabelul 2.10 – Subseturi de defecte

Controlabilitatea și observabilitatea nodurilor unui circuit combinațional pot fi descrise prin valori numerice, existând diferite metode prin care acestea se pot calcula [Butler1992]. În aplicația implementată s-au utilizat valorile propuse în [Goldstein1979]. Pentru fiecare poartă logică se consideră CC0 ca fiind controlabilitatea lui "0" la ieșirea porții și CC1, controlabilitatea lui "1" la ieșirea porții. Formulele de calcul pentru aceste două valori depind de tipul porții logice.

De exemplu pentru o poartă AND cu N intrări ( $X_1, X_2, \dots, X_N$ ) avem:

$$CC0(Y) = \min_{i=1,N}(CC0(X_i)) + 1$$

$$CC1(Y) = \sum_{i=1,N} CC1(X_i) + 1$$

Pentru o poartă OR cu N intrări avem:

$$CC0(Y) = \sum_{i=1,N} CC0(X_i) + 1$$

$$CC1(Y) = \min_{i=1,N} CC1(X_i) + 1$$

Iar pentru o ieșire inversată a unei porți, sau pentru poarta NOT avem:

$$CC1(Y) = CC0(X) + 1$$

$$CC0(Y) = CC1(X) + 1$$

Folosind formulele de mai sus se pot determina măsurile de controlabilitate pentru fiecare nod dintr-un circuit logic combinațional.

Observabilitatea unei intrări a unei porți logice se exprimă în funcție de observabilitatea ieșirii porții și controlabilitatea celorlalte intrări ale porții. De exemplu, observabilitatea unei intrări  $X_i$  a unei porți logice AND cu  $N$  intrări este:

$$OC(X_i) = \sum_{j=1,N, j \neq i} CC1(X_j) + OC(Y) + 1$$

iar observabilitatea unei intrări a unei porți OR este:

$$OC(X_i) = \sum_{j=1,N, j \neq i} CC0(X_j) + OC(Y) + 1$$

Generarea aleatoare a subsetului de defecte se face considerând ca egale probabilitățile de apariție pentru fiecare defect. În tabelul 2.10 s-au prezentat câteva metode diferite de generare a subsetului de defecte ce stă la baza estimării FC. Fiecare metodă presupune diferite probabilități pentru fiecare tip de defect. Considerând un defect de scurt-circuit între două noduri  $X_i$  și  $X_j$ , notăm controlabilitatea defectului cu  $CC(X_i, X_j)$  și observabilitatea acestuia cu  $OC(X_i, X_j)$ . Controlabilitatea și observabilitatea unui defect se pot exprima în funcție de controlabilitatea și observabilitatea nodurilor ce formează defectul:

$$CC(X_i, X_j) = \min(CC0(X_i) + CC1(X_j), CC1(X_i) + CC0(X_j))$$

$$OC(X_i, X_j) = \min(OC(X_i), OC(X_j))$$

Probabilitatea de detectabilitate a unui defect folosind testarea logică este mai mică dacă controlabilitatea și observabilitatea defectului sunt mai mici. O valoare minimă pentru FC se va obține dacă alocăm ponderi mai mari defectelor ce au o controlabilitate și observabilitate mai mică (linia minim din tabelul 2.10). Dacă alocăm o probabilitate mai mare defectelor ce au o controlabilitate și observabilitate mai mare se va obține o valoare maximală pentru FC (linia maxim din tabelul 2.10).

O altă metodă pentru generarea subsetului de defecte pentru estimarea FC se bazează pe evoluția mai multor generații de defecte care în final vor constitui subsetul dorit. Evoluția urmărește stabilizarea FC și obținerea unei dispersii cât mai mici. Această metodă corespunde liniei "genetic" din tabelul 2.10.

În tabelul 2.11 sunt prezentate valorile FC pentru patru circuite benchmark ISCAS95 împreună cu erorile aferente. Numărul de defecte considerat a fost 10000. În cazul generării genetice acest număr a fost împărțit și a evoluat în 10 generații a câte 1000 de defecte.



Algoritmul genetic a fost orientat pe minimizarea și stabilizarea FC. Au fost făcute 20 măsurători pentru fiecare circuit.

Prin generarea aleatoare a unui subset de defecte se obține o aproximare foarte bună pentru FC. Dacă însă se dorește obținerea unor valori minimale sau maximale pentru FC sau dacă este necesară încadrarea acesteia într-o marjă de eroare bine definită atunci se poate opta pentru selecția genetică a subsetului de defecte.

Circuit	Valoarea medie		Eroarea medie		Eroarea maximă		Deviația standard		Valoarea reală
	Aleator	Genetic	Aleator	Genetic	Aleator	Genetic	Aleator	Genetic	
c432	62.741	61.773	0.486	0.261	1.079	0.597	0.565	0.308	62.52
c499	94.653	92.281	0.189	0.155	0.431	0.307	0.216	0.177	94.51
c880	97.292	93.277	0.122	0.107	0.283	0.242	0.154	0.135	97.22
c1355	95.740	90.817	0.234	0.131	0.373	0.280	0.275	0.155	95.56

Tabloul 2.11 – FC pentru defecte generate aleator și genetic împreună cu erorile aferente

Vectorii de test generați în cadrul aplicației folosesc:

- algoritmul D;
- algoritmul PODEM;
- aleator.

Datorită faptului că pentru detecția unui defect prin testarea termică este suficientă doar activarea defectului algoritmul de generare a vectorilor de test pentru testarea termică este relativ simplu și presupune doar activarea nodurilor defectelor pe valori logice complementare.

## 2.9.2 Simularea defectelor circuitelor combinaționale

Procesul de calcul a factorului de acoperire a defectelor (FC) este cunoscut sub numele de simularea defectelor. Pentru calculul factorului de acoperire a defectelor este necesară simularea atât a circuitului fără defecte cât și a circuitului defect.

Un circuit integrat combinațional este reprezentat printr-o rețea de noduri și porți logice. Fiecare poartă logică are un set de noduri de intrare, un nod de ieșire sau mai multe noduri de ieșire și codul pentru simularea logică a porții. Codul de simulare este distribuit în fiecare clasă ce implementează un tip de poartă logică (figura 2.78), clase derivate din `CLogicGate`. Nodurile din circuit sunt identificate printr-o adresă și au mai multe attribute utilizate în procesul de simulare, iar implementarea lor este conținută în clasa `CLogicNode`. O poartă logică poate conține și implementarea la nivel de tranzistori pentru cazul în care este necesară o simulare la acest nivel (switch level simulation). Pentru fiecare tip de poartă logică, indiferent de numărul de intrări, există câte o clasă ce îi implementează funcționalitatea.

Circuitul combinațional este implementat de o clasă numită `CLogicCircuit`. Această clasă conține o listă de obiecte derivate din `CLogicGate` precum și nodurile de intrare și ieșirile observabile ale circuitului. Pe lângă metodele folosite în simulare clasa `CLogicCircuit` mai oferă și niște funcții pentru încărcarea și salvarea circuitului din/în diferite formate de fișiere (ISCAS85, SPICE).

Defectele de scurt-circuit simulate în cadrul aplicației sunt implementate în clasa `CLogicDefect`. Defectele de blocare sunt o subclasă a defectelor de scurt-circuit și anume defectele ce implică nodurile  $V_{DD}$  și masa.

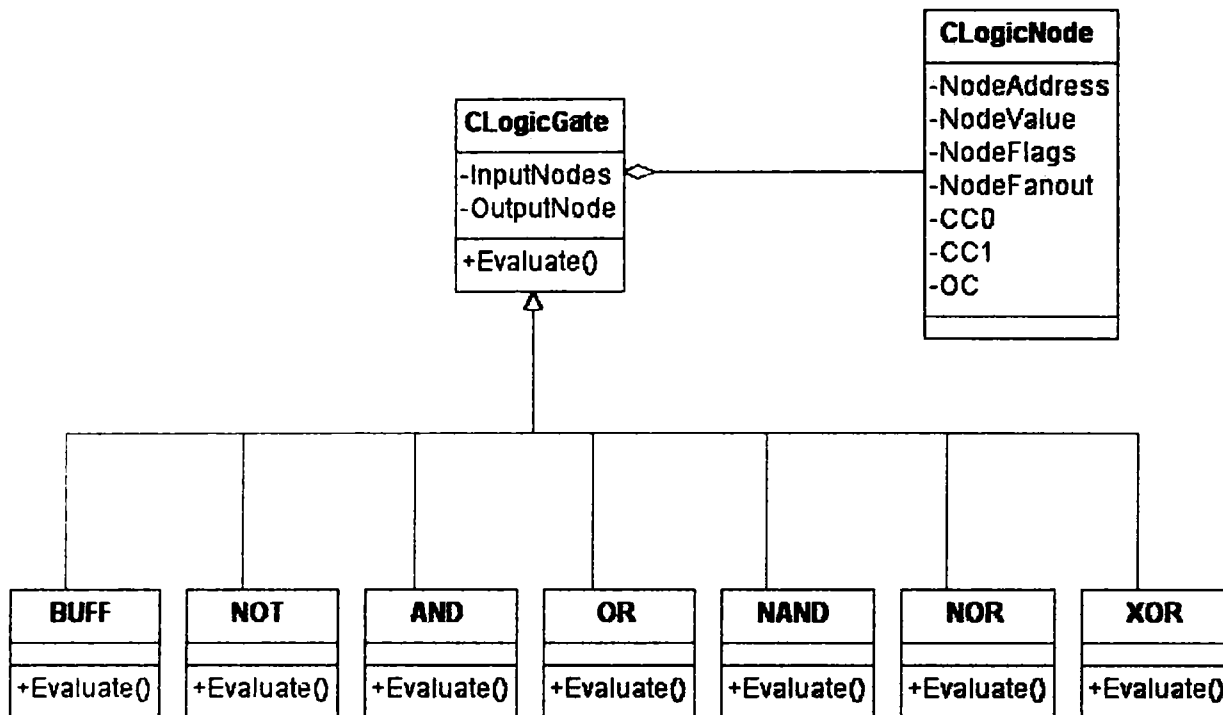


Figura 2.78 – Ierarhia de clase pentru porțile logice

Clasele de simulare sunt prezentate pe scurt în diagrama din figura 2.79. În cadrul aplicației s-a implementat câte o clasă pentru fiecare tip de simulare: simularea logică a defectelor de blocare, simularea logică și termică a defectelor de scurt-circuit și simularea electro-logică folosind SPICE. Aceste clase de simulare sunt derivate din CLogicSimulator.

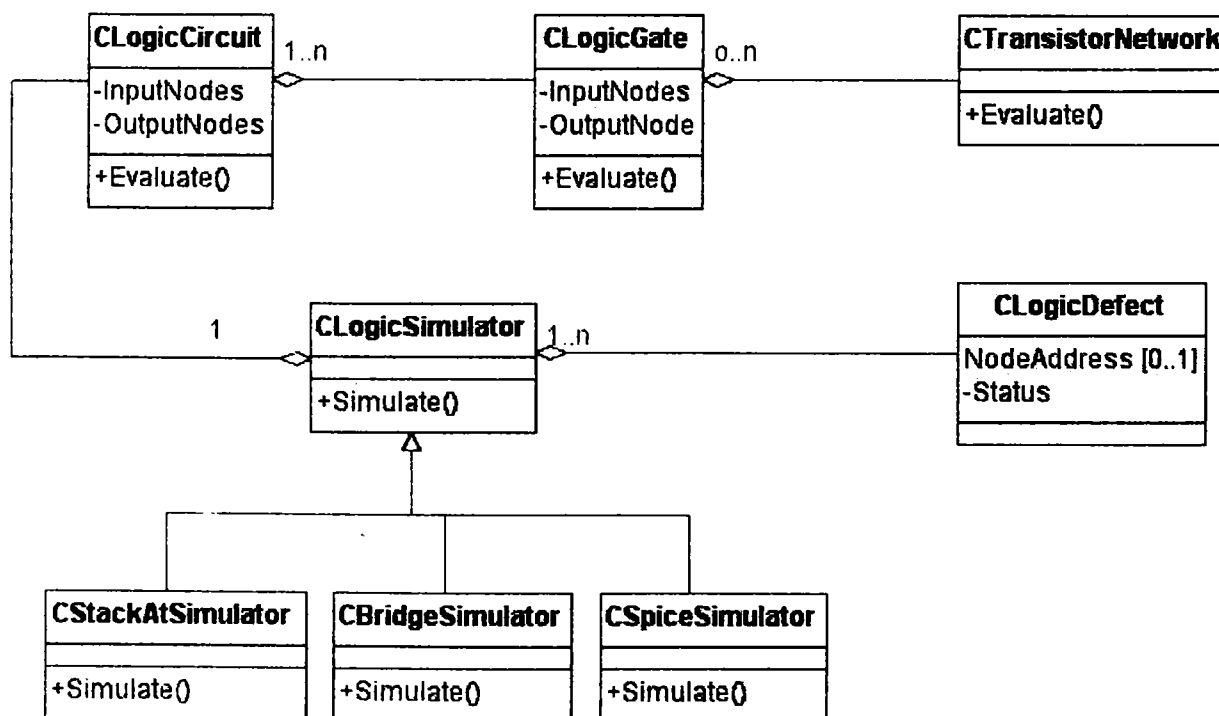


Figura 2.79 – Ierarhia de clase pentru simularea logică

Aplicația a fost dezvoltată în mediul de programare Microsoft Visual C++ 6.0/.NET, are în jur de 25 clase și 6000 linii de cod (figura 2.80).

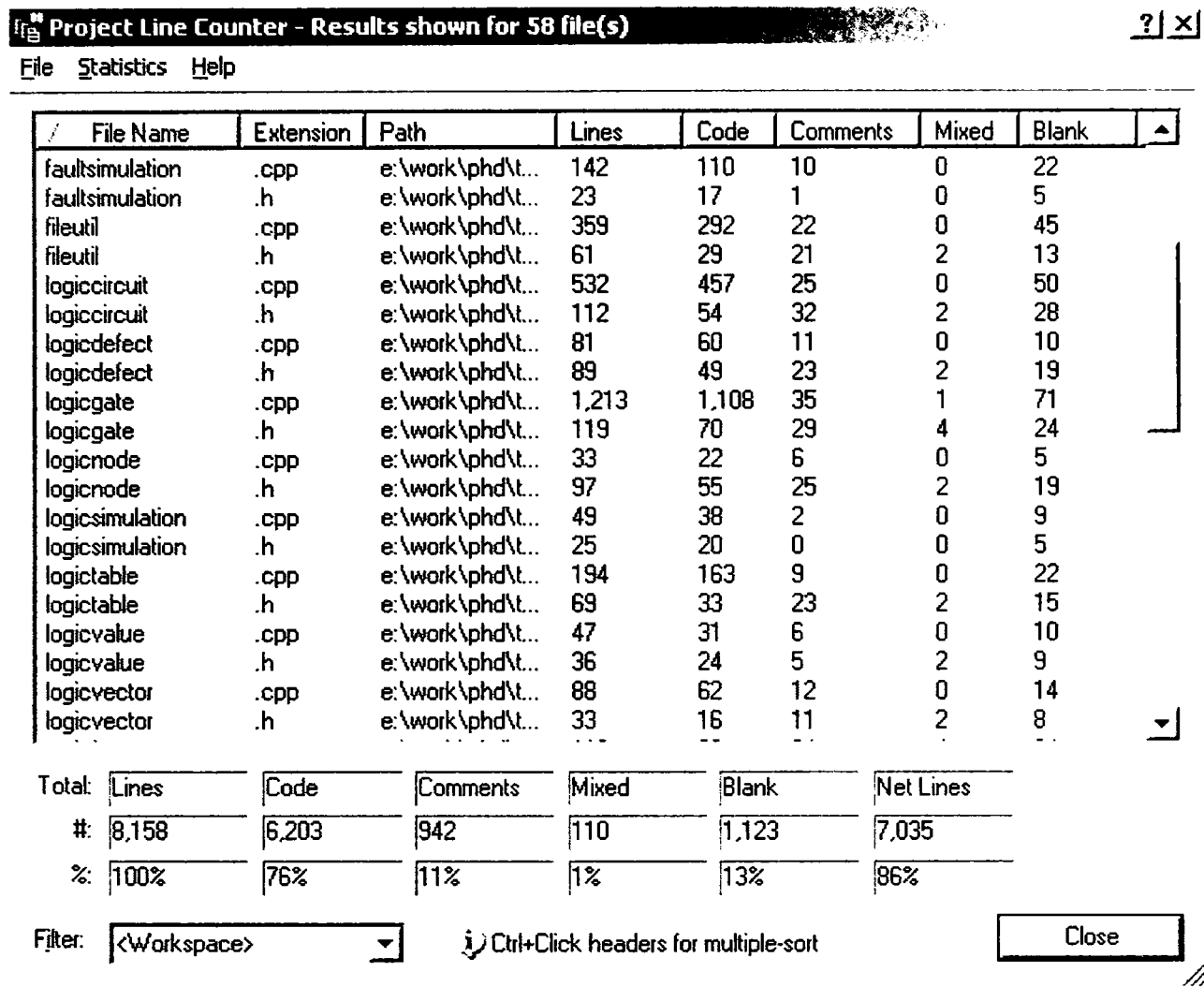


Figura 2.80 – Informații statistice despre aplicație

Folosind această aplicație au fost simulate circuitele ISCAS95 cu mai multe tipuri de defecte pentru testarea logică și testarea termică. Într-o primă fază s-au simulat defectele de blocare pentru a determina acoperirea defectelor de către testarea logică și testarea termică (Tabelul 2.12). Vectorii de test au fost generați folosind algoritmul PODEM. Defectele de blocare sunt în număr mult mai mic decât defectele descurt-circuit, astfel că s-a putut face o simulare exhaustivă.

Circuit			Testare logică		Testare termică	
Nume	Defecte	Vectori test	FC	Durata	FC	Durata
c432	392	91	99.23%	0:09	100 %	0:01
c499	486	123	100 %	0:10	100 %	0:01
c880	886	84	100 %	0:14	100 %	0:01
c1355	1174	123	100 %	0:33	100 %	0:02
c1908	1826	155	99.89 %	1:53	100 %	0:02
c2670	2852	166	97.51 %	6:37	99.99 %	0:03
c3540	3438	201	97.06 %	16:12	99.97 %	0:05
c5315	4970	171	99.96 %	39:17	99.98 %	0:06
c6288	4896	45	99.65 %	5:17	99.65 %	0:02
c7552	7438	289	98.87 %	52:38	99.95 %	0:15

Tabelul 2.12 – Rezultatele simulării defectelor de blocare

Din Tabelul 2.12 se poate observa că pentru defectele de blocare atât testarea logică cât și testarea termică au o acoperire a defectelor foarte bună, cu un plus pentru testarea termică. Din punctul de vedere al timpului de simulare, simularea defectelor pentru testarea termică este mult mai rapidă decât simularea defectelor pentru testarea logică. Aceasta se datorează faptului că detecția defectelor prin testarea termică necesită doar activarea defectului în timp ce la testarea logică efectul defectului trebuie propagat către ieșirile observabile ale circuitului.

Defectele de scurt-circuit externe pentru circuitele ISCAS95 sunt prezentate în tabelul 2.13. De aici se poate observa că vectorii de test generați pentru defectele de blocare au o acoperire foarte bună și pentru defectele de scurt-circuit externe. Tot din aceste rezultate se mai poate observa faptul că testarea termică și testarea pe bază de curent au o acoperire foarte bună pentru defectele de scurt-circuit, de peste 99%. În ceea ce privește testarea logică există o mai mare variație a factorului de acoperire a defectelor, între 60% și 97%. Timpii de simulare sunt mult mai mari în cazul testării logice față de cei pentru testarea termică, din aceeași cauză prezentată la simularea defectelor de blocare.

Circuit			Testare logică		Testare termică	
Nume	Defecte	Vectori test	FC	Durata	FC	Durata
c432	19503	91	62.52 %	58:41	99.96 %	0:04
c499	29890	123	94.51 %	1:09:20	99.73 %	0:05
c880	98790	84	97.22 %	2:43:59	99.93 %	0:10
c1355	173166	123	95.56 %	3:25:33	99.94 %	0:16
c1908	418155	155	94.03 %	23:19:30	99.77 %	0:20
c2670	1018878	166	86.47 %	112:08:13	99.91 %	0:30
c3540	1480060	201	69.15 %	360:06:12	99.85 %	0:35
c5315	3091341	171	-	-	99.95 %	1:19
c6288	3000025	45	-	-	99.99 %	0:29
c7552	6921060	289	-	-	99.96 %	20:18

Tabelul 2.13 – Rezultatele simulării defectelor de scurt-circuit externe

Rezultatele precedente prezentate în tabelele 2.12 și 2.13 au fost obținute pentru defecte de blocare și scurt-circuit ideale, de rezistență 0. În continuare voi prezenta rezultatele simulărilor efectuate pentru defectele de blocare și scurt-circuit rezistive. S-a folosit o funcție uniformă pentru distribuția valorilor rezistențelor defectelor conform [Li2003] și un interval de variație al rezistențelor de [0, 40kΩ]. Chiar dacă simulările defectelor de scurt-circuit rezistiv au fost efectuate pentru o funcție de distribuție uniformă, programul permite și alte tipuri de distribuții iar intervalul de variație pentru rezistența defectului este și el parametrizabil.

Circuit c432		Testare logică		Testare termică		Testare I <sub>DDQ</sub>	
Metoda simulare	Simulari	FC [%]	Durata [min]	FC [%]	Durata [min]	FC [%]	Durata [min]
logică	392	99.23	0	100	0	100	0
logico-electrică (1)	392	7.85	43	29.45	34	100	32
logico-electrică (2)	1960	7.83	1:27	25.51	1:07	100	0:51
logico-electrică (3)	3136	7:40	2:46	26.19	2:44	100	2:43
logico-electrică (4)	392	7.54	2:25	27.05	2:32	100	2:35
logico-electrică (5)	392	7.54	6	27.05	4	100	5

Tabelul 2.14 – Rezultatele simulării defectelor de blocare pentru c432

Pentru a face o comparație între acoperirea defectelor de scurt-circuit rezistiv de cele trei metode de testare s-a considerat același interval pentru rezistența defectului pentru toate metodele de testare. Simularea logică din tabelul 2.14 a vizat doar defectele de rezistență 0. Rezistența defectului este luată în calcul doar de testarea logico-electrică. În tabelul 2.14 sunt prezentate mai multe tipuri de testări logico-electrice folosind SPICE. Toate testele sunt pentru defectele de blocare rezistive pentru circuitul c432. Metoda de simulare electrică este foarte mare consumatoare de timp, astfel că s-au încercat mai multe metode de simulare logico-electrică pentru a obține un timp rezonabil de simulare și un număr cât mai mic de simulări.

În prima variantă de simulare logico-electrică, numărul (1) din tabel, s-a generat aleator (după o funcție de distribuție uniformă) câte o valoare a rezistenței (în intervalul [0,40kΩ]) pentru fiecare defect din setul de defecte. În a doua variantă s-au generat aleator mai multe rezistențe pentru fiecare defect din setul de defecte (în exemplu din tabelul 2.14, câte 5 valori pentru fiecare defect). A treia variantă de simulare (3) conține mai multe valori secvențiale pentru fiecare defect din setul de defecte. Varianta numărul (4) conține simularea în SPICE a rezistențelor pentru un defect. În această variantă s-a implementat în SPICE un scurt program ce calculează acoperirea rezistențelor defectului de un vector de test. În aceste prime patru variante de simulare s-a folosit simularea în SPICE pentru întreg circuitul și într-adevăr se observă un timp foarte mare de simulare, inacceptabil pentru circuite de mari dimensiuni și număr mare de defecte de simulat.

Ultima variantă de simulare folosește pentru simularea electrică în SPICE a unui subcircuit format doar din porțile conectate la nodurile ce formează defectul. Timpul de simulare este mult mai mic decât cel obținut în variantele de simulare electrică a întregului circuit. Un al doilea avantaj al acestei metode este acela că timpul de simulare electrică este nu depinde de complexitatea circuitului, ci doar de numărul de defecte și de numărul de vectori de test.

```

*#let res = 10
*#let fc = 100
*#while res < 40000
*#  destroy all
*#  alter Rd = res
*#  op
*#  if abs(i(vdd)) < 400u
*#    let fc = res * 100 / 40000
*#    let res = 50000
*#  else
*#    let res = res + 100
*#  end
*#end
*#print fc > c432.out

```

Din tabelul 2.14 se observă că valorile FC pentru testarea logică și testarea termică sunt mult mai mici decât cele ale testării  $I_{DDQ}$  datorită intervalelor diferite de detectabilitate. Pentru calculul factorului de acoperire a defectelor s-a folosit EFC, pentru un interval maxim de detectabilitate de [0,40kΩ]. S-a folosit acest interval și nu câte un interval pentru fiecare tip de testare deoarece valorile FC simulate trebuie să servească la compararea celor trei metode de testare.

Pe baza observației anterioare se poate spune că utilizarea unei singure valori pentru reprezentarea factorului de acoperire a defectelor de scurt-circuit rezistiv nu este cea mai potrivită, deoarece ascunde anumite detalii în ceea ce privește performanța setului de vectori folosit și a metodei de testare utilizate. De exemplu, valorile mici ale lui FC pentru testarea logică nu pot spune dacă setul de vectori folosit are o acoperire slabă a defectelor din setul de

defecte ales, dacă circuitul are o controlabilitate și observabilitate slabă sau dacă metoda de testare are un interval de detectabilitate mai mic.

Din acest motiv propun ca pentru compararea vectorilor de test pentru diferite metode de testare a defectelor de scurt-circuit rezistive să se utilizeze două valori (ZFC,RFC), prima din ele fiind factorul de acoperire pentru defectele de scurt-circuit de rezistență zero iar a doua valoare reprezentând o valoare procentuală pentru intervalul mediu de detectabilitate. Cu o astfel de pereche de valori se pot descrie defectele de scurt-circuit rezistiv în modul următor: (ZFC,RFC) reprezintă un vector de test pentru o anumită metodă de testare cu o acoperire a defectelor de ZFC și un interval de detectabilitate de RFC. Această descriere conține atât performanța vectorului de test cât și performanța metodei în a detecta un cât mai mare interval de rezistențe.

În tabelul 2.15 sunt prezentate rezultatele simulării defectelor de scurt-circuit rezistiv pentru câteva din circuitele ISCAS95. Pentru fiecare circuit au fost simulate 10000 de defecte de scurt-circuit selectate aleator.

Circuit		Testare logică		Testare termică		Testare I <sub>DDQ</sub>	
Nume	Simulari	FC [%]	Durata [min]	FC [%]	Durata [min]	FC [%]	Durata [min]
c432	13801	(63.08,7.54)	3:38	(99.90,26.79)	3:21	(99.90,100)	3:30
c499	17359	(94.51,8.11)	2:15	(99.75,28.09)	2:06	(99.75,99.90)	2:12
c880	13479	(97.67,7.83)	1:40	(99.92,27.30)	1:34	(99.92,100)	1:41
c1355	11988	(95.52,7.78)	1:11	(99.92,27.87)	0:51	(99.92,100)	1:02
c1908	10548	(94.24,6.93)	1:35	(99.75,26.87)	1:04	(99.75,100)	1:08
c2670	14878	(86.53,6.55)	3:55	(99.90,26.87)	2:51	(99.90,100)	3:02
c3540	10992	(68.41,6.31)	4:30	(99.86,26.35)	1:14	(99.86,100)	1:21
c5315	11974	(84.44,6.39)	5:23	(99.98,26.62)	2:31	(99.98,100)	2:42
c6288	10800	(99.89,7.56)	1:19	(99.98,27.73)	0:55	(99.98,100)	1:07
c7552	11183	(94.83,6.45)	6:53	(99.94,26.84)	2:45	(99.94,100)	3:06

Tabelul 2.15 – Rezultatele simulării logico-electrice a defectelor de scurt-circuit rezistiv

Urmărind rezultatele din tabelul 2.15 se pot observa intervalele relative de detectabilitate pentru fiecare metodă de testare. Astfel, testarea pe bază de curent are acoperirea cea mai mare pentru defectele de scurt-circuit rezistiv, după care urmează testarea termică cu aproximativ 25-30% din intervalul de detectabilitate al testării I<sub>DDQ</sub> și în final testarea logică cu un interval mic de detectabilitate a defectelor rezistive. Testarea logică poate detecta doar defectele de scurt-circuit ce au rezistențe mici, de până la aproximativ 2kΩ, în timp ce testarea termică și testarea I<sub>DDQ</sub> pot detecta defecte rezistive cu rezistențe de aproximativ 10kΩ și respectiv 40kΩ.

Timpii de execuție pentru metoda propusă de simulare electro-logică sunt mai mari decât cei de simulare logică, dar sunt și mult mai mici decât cei de simulare electro-logică exhaustivă. Rezultatele simulării logice pentru aceleași defecte pentru care s-a obținut tabelul 2.15, sunt prezente în tabelul 2.16. Comparativ cu rezultatele prezentate în [Engelke2003], [Dessai1998] și [Li2003], pentru care timpii de simulare sunt la nivelul simulării logice, metoda propusă are timpi de execuție mai mari pentru circuitele mai simple însă pentru circuitele de dimensiuni mari timpii de execuție sunt comparabili ca și ordine de mărime cu cei pentru simularea logică.

Timpul de simulare a ramas totuși destul de mare și datorită faptului că simularea SPICE se execută de un program extern aplicației, ce este lansat în execuție la fiecare simulare. Dacă nucleul de simulare SPICE ar putea fi compilat în aplicație sau s-ar putea

încărca sub forma unul DLL, atunci timpul de execuție s-ar micșora și mai mult, ajungând la ordinul de mărime al simulării logice.

Circuit		Testare logică		Testare termică		Testare I <sub>DDQ</sub>	
Nume	Vectori	FC [%]	Durata [min]	FC [%]	Durata [min]	FC [%]	Durata [min]
c432	91	63.08	0:12	99.90	0:00	99.90	0:00
c499	123	94.51	0:06	99.75	0:00	99.75	0:00
c880	84	97.67	0:06	99.92	0:00	99.92	0:00
c1355	123	95.52	0:14	99.92	0:00	99.92	0:00
c1908	155	94.24	0:32	99.75	0:00	99.75	0:00
c2670	166	86.53	1:21	99.90	0:00	99.90	0:00
c3540	201	68.41	3:26	99.86	0:00	99.86	0:00
c5315	171	84.44	3:04	99.98	0:00	99.98	0:00
c6288	45	99.89	0:23	99.98	0:00	99.98	0:00
c7552	289	94.83	4:04	99.94	0:00	99.94	0:00

Tabelul 2.16 – Rezultatele simulării logice a defectelor de scurt-circuit de rezistență 0

În plus, aplicația poate fi configurată să dea mai multe informații despre rezultatul simulării. De exemplu se pot obține date de simulare despre fiecare combinație defect-vector de test, câteva sunt prezentate în listingul de mai jos. Aceste date pot fi utilizate mai departe în procesul de selecție al vectorilor de test.

```

Vector 1 - Defect 1 - Detected 1 - FC = 26.02
Vector 1 - Defect 2 - Detected 1 - FC = 28.52
Vector 1 - Defect 3 - Detected 0 - FC = 00.00
Vector 1 - Defect 4 - Detected 1 - FC = 27.27
Vector 1 - Defect 5 - Detected 1 - FC = 28.27
Vector 1 - Defect 6 - Detected 1 - FC = 26.27
Vector 1 - Defect 7 - Detected 1 - FC = 27.77
Vector 1 - Defect 8 - Detected 1 - FC = 25.77
Vector 1 - Defect 9 - Detected 1 - FC = 28.02
Vector 1 - Defect 10 - Detected 1 - FC = 23.02
Vector 1 - Defect 11 - Detected 0 - FC = 00.00
Vector 1 - Defect 12 - Detected 1 - FC = 28.52
Vector 1 - Defect 13 - Detected 1 - FC = 24.52
Vector 1 - Defect 14 - Detected 1 - FC = 26.27
Vector 1 - Defect 15 - Detected 1 - FC = 22.77
Vector 1 - Defect 16 - Detected 1 - FC = 26.02
Vector 1 - Defect 17 - Detected 1 - FC = 29.77
Vector 1 - Defect 18 - Detected 1 - FC = 27.77
Vector 1 - Defect 19 - Detected 1 - FC = 27.02
Vector 1 - Defect 20 - Detected 1 - FC = 26.77
...

```

Testele au fost efectuate pe trei sisteme, rezultatele de timp fiind o mediere a valorilor obținute pe fiecare sistem:

- Sistem 1: procesor Intel P4 2.8 GHz, 512 MB memorie 400 MHz, 800 MHz FSB
- Sistem 2: procesor AMD Duron 1.2 MHz, 256 MB memorie 166 MHz
- Sistem 3: procesor Intel PIII 1.1 MHz, 256 MB memorie 166 MHz.

Rezultatele prezentate se bazează pe un număr mare de simulări efectuate (de ordinul sutelor), cumulate în timp la nivelul zilelor.

## 2.10 Concluzii

În prima parte a acestui capitol s-a evaluat aplicabilitatea metodei de testare termică pentru circuitele integrate CMOS. Pentru a înțelege necesitatea unei metode de testare ce vizează defectele de scurt-circuit pentru circuitele VLSI CMOS au fost prezentate:

- limitările testării logice aplicate circuitelor CMOS;
- modul în care testarea termică și testarea pe bază de curent rezolvă aceste limitări.

Încă de la început trebuie specificat că aceste metode de testare nu înlocuiesc în totalitate testarea logică, ci sunt folosite împreună cu aceasta.

Testarea logică presupune că orice defect fizic poate fi modelat ca un defect la nivelul porților logice ce compun circuitul. Cea mai utilizată formă a testării logice presupune că defectele blochează permanent o linie din circuit pe 0 logic sau pe 1 logic. Această metodă este cunoscută sub numele de modelul de defectare de blocare permanentă a unei linii (single stuck-at fault). Defectele de blocare pot fi detectate printr-un vector de intrare (sau o secvență de vectori de intrare în cazul circuitelor secvențiale). Testarea logică constă în calculul setului de vectori test, iar apoi acești vectori se aplică la intrările DUT în timp ce i se monitorizează ieșirile. Dacă există diferențe între valorile așteptate la liniile de ieșire și cele observate la aplicarea testului, circuitul este considerat defect.

Dacă un set de test este ales astfel încât să detecteze defectele de blocare vom numi această strategie de testare, testarea defectelor de blocare. Acesta este modul predominant de testare folosit astăzi. Însă, după cum s-a prezentat în acest capitol, defectele de blocare nu modelează cel mai bine defectele fizice ce apar cel mai frecvent în circuitele integrate CMOS. În consecință, testele logice determinate folosind modelul defectelor de blocare nu au garanția să acopere defectele apărute în circuitele CMOS.

Un exemplu de astfel de defecte fizice sunt defectele de tip scurt-circuit, prin care sunt conectate două sau mai multe noduri din circuit. Pentru a modela corect la nivel de poartă logică aceste defecte este necesar a se estima rezistența defectului. Altfel valorile de la ieșirea circuitului în prezența unui defect, nu pot fi precise și deci, nu pot fi detectate. Însă, aceste defecte, pot fi ușor detectate prin aplicarea unui test termic sau a unui test bazat pe curent.

În a doua parte a acestui capitol s-au prezentat principalele probleme care apar la testarea defectelor de scurt-circuit atât în cadrul metodei de testare logică cât și pentru metodele de testare termică și  $I_{DDQ}$ . Principalele probleme care au fost enunțate și investigate sunt:

- creșterea curentului consumat și a temperaturii disipate de circuitele integrate datorită procesului de scalare;
- lipsa unei metode și a unei unelte integrate de simulare electro-termică pentru circuitele integrate;
- lipsa unei metode unitare adecvate de simulare și reprezentare a defectelor de scurt-circuit rezistiv pentru testarea logică, testarea termică și testarea pe bază de curent.

În a treia parte a capitolului s-au prezentat soluțiile existente și cele propuse pentru aceste probleme. În continuare s-au prezentat aplicația de simulare implementată și rezultatele obținute în cadrul cercetării. Contribuțiile aduse și prezentate în acest capitol sunt următoarele:

- prezentarea și justificarea prin simulări a efectelor termice ale defectelor de scurt-circuit și posibilitatea de detecție a lor prin testarea termică;
- implementarea în SPICE a unei metode de simulare electro-termice pentru circuitele integrate VLSI;



- propunerea și implementarea unei metode genetice de selecție a setului de defecte pentru minimizarea sau maximizarea factorului de acoperire a defectelor de către metodele de testare prezentate;
- propunerea, implementarea și testarea unei metode de simulare logică-electrică pentru defectele de scurt-circuit pentru cele trei metode de testare studiate;
- propunerea unei notații pentru reprezentarea factorului de acoperire a defectelor pentru defectele de scurt-circuit rezistive;
- studiul comparativ pe baza rezultatelor simulărilor între testarea logică, testarea termică și testarea  $I_{DDQ}$ . Datele comparative pentru cele trei metode de testare sunt prezentate în tabelul 2.17.

	Testare logică	Testare termică	Testare $I_{DDQ}$
Număr vectori test	mare	mic	mic
Viteza testării	rapidă	lentă	lentă
Complexitate ATPG	mare	medie	medie
Complexitate simulare	mică	medie	medie
Acoperirea defectelor de scurt-circuit ideale	medie	mare	mare
Acoperirea defectelor de scurt-circuit rezistive	mică	medie	mare
Suprafața de integrare suplimentară	nu	senzor integrat	senzor integrat
Proiectare pentru testare	posibil	da	da
Posibilitate de autotestare	da	da	da

Tabelul 2.17 – Date comparative între cele trei metode de testare

#### Avantajele testării termice:

- nu este necesar un contact fizic cu unitatea ce se testează;
- este independentă de numărul de terminale;
- efectul defectului nu trebuie propagat la ieșirile observabile ale circuitului;
- număr mic de vectori de test;
- acoperire bună a defectelor de scurt-circuit, inclusiv cele rezistive;
- cost scăzut – în cazul utilizării senzorilor de temperatură;
- nu produce deteriorări produsului testat;
- localizarea exactă a erorii (metoda termo-optică).

#### Dezavantajele testării termice:

- lentă;
- depinde de tipul de împachetare folosit și de mediul de utilizare;
- cost mare pentru cazul testării termice în infraroșu (costul unei camere video în infraroșu aplicabilă la circuitele integrate costă între 10000 și 100000 USD);
- în cazul utilizării senzorilor de temperatură integrați aceștia ocupă spațiu din pastila de siliciu;
- datorită procesului de scalare a dimensiunilor și tensiunii de alimentare crește temperatura de operare a unui circuit CMOS.

#### Avantajele testării $I_{DDQ}$ :

- efectul defectului nu trebuie propagat la ieșirile observabile ale circuitului;
- număr mic de vectori de test;

- acoperire foarte bună a defectelor de scurt-circuit, inclusiv cele rezistive;
- cost scăzut.

Dezavantajele testării  $I_{DDQ}$  sunt:

- lentă;
- nu toate circuitele CMOS sunt testabile  $I_{DDQ}$ ;
- datorită procesului de scalare a dimensiunilor și tensiunii de alimentare cresc curenții de scurgere în structurile CMOS;
- introducerea senzorilor de curent în serie cu  $V_{DD}$  sau GND introduc distorsiuni ale semnalelor.

Pe lângă creșterea calității produselor, un alt beneficiu al testării termice este creșterea fiabilității circuitelor. Există unele defecte fizice (scurt-circuit poartă-oxid) care nu modifică comportamentul funcțional al circuitului, deci testele logice nu le pot detecta. Însă majoritatea acestor defecte duc la un consum sporit de curent în timpul funcționării și cauzează până la urmă și defectarea circuitului mai repede decât un circuit fără astfel de defecte. Și acest tip de defecte poate fi pus în evidență printr-un test termic sau printr-un test de curent.

Pe baza acestor observații și a datelor prezentate în acest capitol se poate spune că folosirea testării termice sau a testării  $I_{DDQ}$  împreună cu testarea logică îmbunătățește calitatea, fiabilitatea și procentul de produse fără defecte din numărul de produse oferite spre vânzare.

În ceea ce privește diferențierea între testarea termică și testarea  $I_{DDQ}$ , trebuie luate în calcul mai multe elemente: dimensiunea și complexitatea circuitului, tehnologia de fabricație utilizată, utilizarea sau nu a senzorilor integrați, nivelul de calitate cerut a se obține, costurile alocate pentru testare și autotestare, ș.a. Testarea termo-optică este recomandată în faza de proiectare și verificare termică a circuitului integrat, precum și în faza de analiză a defectelor pentru localizarea și identificarea cauzelor apariției acestora. Testarea termo-optică este mai puțin utilizată în faza de testare a circuitelor integrate deoarece este mai lentă decât metoda termo-electrică și implică costuri suplimentare destul de mari, atât pentru echipamentul de testare precum și datorită timpului mare de testare. Metoda de testare termo-electrică se poate utiliza în cazul circuitelor integrate VLSI ce au implementați senzori de temperatură integrați. În celelalte cazuri este recomandată metoda de testare  $I_{DDQ}$ .

# 3 Testarea termică a plăcilor cu circuite integrate

## 3.1 Modelarea disipării temperaturii în plăcile cu circuite integrate

Rezistența termică la nivel de placă cu circuite integrate (PCB) constă în rezistența întâmpinată de fluxul de căldură de la suprafața circuitului la un punct de referință pentru întregul sistem. Acest punct de referință poate fi temperatura aerului ce înconjoară componenta, marginea cablajului imprimat în care diferitele componente sunt fixate sau chiar un element de răcire care de obicei este prevăzut la unele circuite

În cazul circuitelor montate pe o placă (PCB) fenomenele care stau la baza răcirii lor sunt conducția (prin pinii componentei și placă), convecția (circuit-aer și placă-aer) și, în cazul în care se dorește o acuratețe mai mare, radiația (între componentele alăturate și între plăcile alăturate). O prezentare schematizată a acestor fenomene e prezentată în figura 3.1.

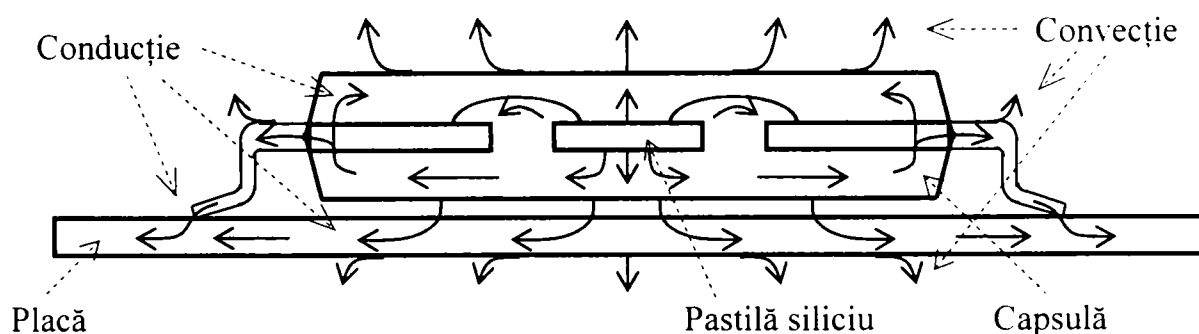


Figura 3.1 – Căile de disipare a căldurii într-un circuit integrat montat pe o placă

Mecanismul predominant de transfer al căldurii degajate de un circuit integrat montat pe o placă este dependent de mai multe elemente: prezența și tipul elementelor de răcire, dispunerea componentelor pe placă și a plăcilor în sistem, etc. În cazul în care un circuit este prevăzut cu elemente de răcire (statice sau dinamice) răcirea este predominant realizată prin convecție (figura 3.2). Răcirea satică sau pasivă este realizată prin răcitoare de diferite forme și dimensiuni, realizate din materiale cu rezistență termică foarte mică, ce sunt plasate pe capsula circuitului și lipite de acesta cu o pastă ce facilitează conducția între capsulă și răcitor. Răcirea dinamică sau activă presupune și existența unui ventilator al cărui rol este acela de a crește răcirea prin convecție. Datorită faptului că rezistența termică pe calea pastilă de siliciu - capsulă circuit - răcitor - mediu ambiant este mică, aceasta este principala cale de disipare a căldurii degajate de circuit. Fiecare din mediile străbătute de această cale se stabilizează la o anumită temperatură:

- $T_j$  – temperatura medie stabilită la nivelul pastilei de siliciu;
- $T_c$  – temperatura capsulei ce împachetează circuitul;
- $T_h$  – temperatura elementului de răcire (heatsink);
- $T_a$  – temperatura mediului ambiant.

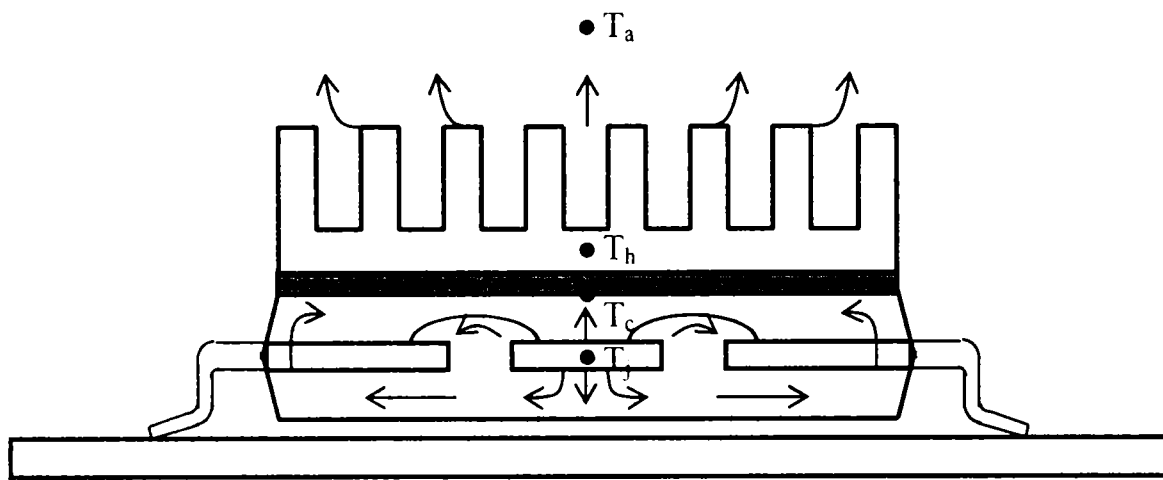


Figura 3.2 – Transferul de căldură prin convecție

Dacă elementele de răcire sunt prevăzute la nivel de placă și nu există convecție forțată, atunci mecanismul predominant de transfer al căldurii degajate de circuitele de pe placă este conducția, prin placă către elementele de răcire (figura 3.3). Călea principală de răcire a unei componente în acest caz este: pastilă siliciu - capsulă circuit - placă sau capsulă - pini - placă și în final elementele de răcire de la nivelul plăcii. Fiecare din mediile străbătute de această cale se stabilizează la o anumită temperatură:

- $T_j$  – temperatura medie stabilită la nivelul pastilei de siliciu;
- $T_c$  – temperatura capsulei ce împachetează circuitul;
- $T_b$  – temperatura plăcii;
- $T_h$  – temperatura elementului de răcire.

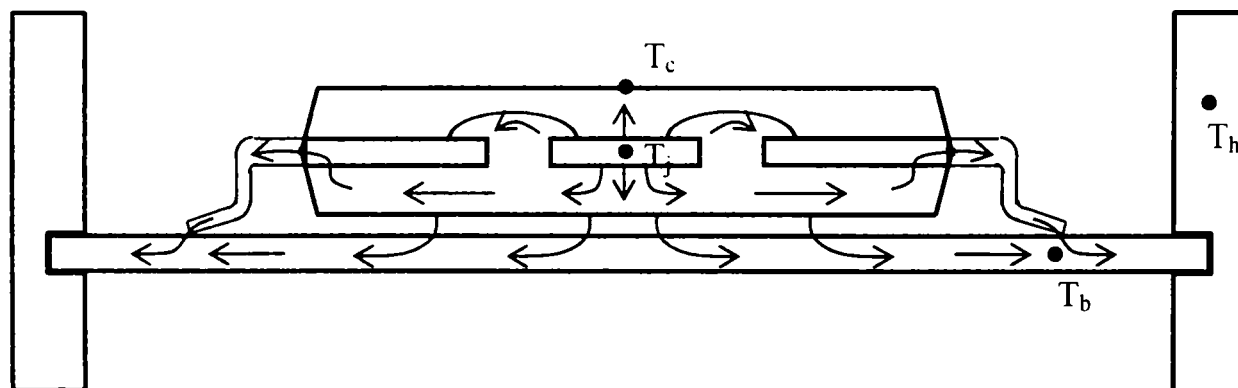


Figura 3.3 – Transferul de căldură prin conducție

În realitate însă, mecanismele de transfer de căldură sunt mult mai complexe și toate influențează, într-o pondere mai mică sau mai mare, răcirea circuitelor de pe placă. În timp ce controlul termic al sistemelor electronice crește în complexitate, provocarea de a integra constrângerile termice încă din procesul de proiectare a produsului rămân constante. Evoluția modelelor componentelor compacte subliniază nevoia de a prevedea performanța sistemului integrat pe baza caracteristicilor individuale ale componentelor. Maniera corectă de tratare a interacțiunilor, cum ar fi formularea condițiilor limita de interfațare, rămâne în continuare un lucru dificil și de multe ori este sursa controverselor [Manno1997]. Astfel că, analizele termice actuale iau în calcul atât conducția cât și convecția (uneori chiar și radiația), toate acestea fiind calculate diferit și putând fi estimate cu o acuratețe foarte bună doar cu ajutorul metodelor numerice.

### 3.1.1 Modelul rezistențelor termice

Figura 3.4 arată în mod simplificat diagrama bidimensională a rezistenței termice, pornind de la joncțiune până la suprafața dispozitivului și a plăcii cu circuite precum și efectele factorilor externi ai rezistenței termice a unui circuit integrat montat pe o placă.

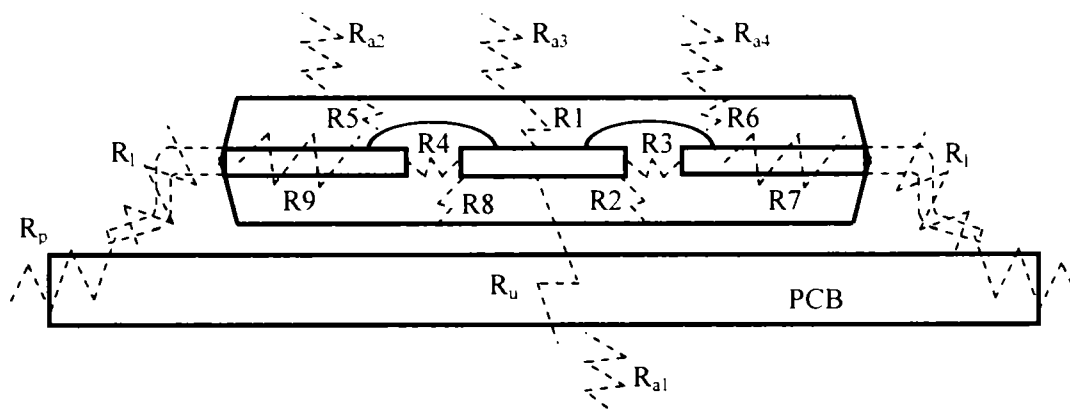


Figura 3.4 – Rezistențele termice ce apar la un circuit integrat montat pe o placă

unde:

$R_{a1}$ - $R_{an}$  – rezistența termică a aerului din jurul dispozitivului;

$R_n$  – rezistența termică a elementelor de submodul (grăsimi termice de lipire, goluri de aer, etc.);

$R_l$  – rezistența termică a conectorilor;

$R_b$  – rezistența termică a plăcii (PCB).

Dacă sunt prezente și elemente de răcire atunci se mai introduc și rezistențele termice pentru fiecare mediu suplimentar prin care se transferă căldura până la mediul ambiant.

Datorită faptului că rezistențele externe ale aerului înconjurător cablajului, grăsimii termice, elementelor de răcire și capsulei circuitului sunt în serie cu rezistența modului, factorii externi afectează semnificativ temperatura joncțiunii. Dacă modulul are o rezistență termică globală echivalentă cu cea din figura 2.49, în jur de  $14\text{ }^{\circ}\text{C}/\text{W}$ , după adăugarea factorilor externi (figura 3.4) rezistența termică poate atinge valori de  $40\text{ }^{\circ}\text{C}/\text{W}$  sau chiar mai mult [Colquitt2000].

Această lucră prezintă o problemă pentru producătorii care doresc să prezinte informații termice despre circuitele lor, deoarece factorii externi nu sunt cunoscuți la momentul caracterizării termice a modului. Ca o soluție, producătorii de semiconductori au furnizat două tipuri de valori ale rezistențelor:  $\theta_{j-c}$  (conductivitatea termică joncțiune - capsulă) și  $\theta_{j-a}$  (conductivitatea termică joncțiune - mediu ambiant) în încercarea de a permite luarea în calcul în cadrul mediului final al dispozitivului [TI1999, JESD51-4].

Pentru a descrie într-un mod simplificat transferul de căldură la nivelul componentelor plasate pe o placă este folosit modelul rețelelor termice integrate și legea conservării energiei termice pentru fiecare sursă de căldură.

Pentru cazul transferului de căldură prin convecție așa cum este prezentat în figura 3.2, aplicarea modelului rezistențelor termice duce la schema din figura 3.5, model utilizat și în [Colquitt2000] și [Mahajan2002]. În această figură sunt prezentate temperaturile din mai multe puncte aflate pe calea principală de răcire a componentei:  $T_j$ ,  $T_c$ ,  $T_h$  și  $T_a$ . Între aceste puncte se interpune câte o rezistență ce este dependentă de tipul materialului și aria de disipare a căldurii:

În schema din figura 3.7 există următoarele elemente:

- $T_j$  – temperatura medie la nivelul pastilei de siliciu;
- $T_s$  – temperatura medie a componentei de răcire internă circuitului (spreader);
- $T_c$  – temperatura medie la nivelul capsulei;
- $T_h$  – temperatura medie la nivelul componentei de răcire externă circuitului (heatsink);
- $T_b$  – temperatura medie la nivelul plăcii;
- $R_{js}$  – rezistența termică între joncțiune și componenta de răcire internă;
- $R_{sc}$  – rezistența termică între componenta de răcire internă și circuit;
- $R_{ch}$  – rezistența termică între capsulă și componenta de răcire externă;
- $R_{cb}$  – rezistența termică între capsulă și placă;
- $R_{ba}$  – rezistența termică între placă și mediul ambiant;
- $R_{ha}$  – rezistența termică între componenta de răcire externă și mediul ambiant.

```
.SUBCKT THERMSIM Tj Ts Tc Th Tb Ta VDD GND
* microarchitecture thermal simulation
* nodes Tj, Ts, Tc, Th, Tb, Ta
.param rthjs=0.5 rthsc=1.1 rthch=2.5 rthcb=5 rthha=5 rthba=10
.param cthj=100n cths=1u cthc=10u cthh=20u cthb=50u
F1 Tj GND VDD 5
VA Ta GND DC=TEMP
RJS Tj Ts rthjs
RSC Ts Tc rthsc
RCH Tc Th rthch
RCB Tc Tb rthcb
RHA Th Ta rthha
RBA Tb Ta rthba
CJ Tj GND cthj
CS Ts GND cths
CC Tc GND cthc
CH Th GND cthh
CB Tb GND cthb
.ENDS THERMSIM
```

O variantă simplificată a modelului propus este folosită în [Malhammar2002\_3] pentru implementarea conceptului de teritoriu sau suprafață termică. Suprafața termică se definește ca fiind cea mai mică suprafață rectangulară din PCB necesară pentru răcirea pasivă a unei componente (circuit).

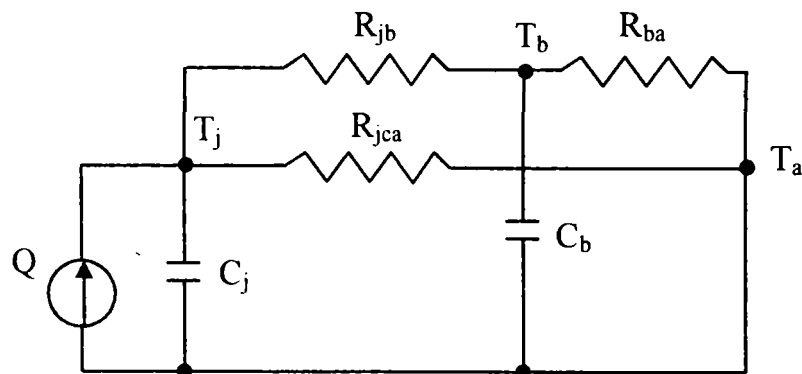


Figura 3.8 – Rețeaua termică ce modelează o suprafață termică

În cazul în care nu sunt componente de răcire active, o parte din căldura generată de circuit este disipată prin suprafața componentei iar o parte din căldură este disipată prin placă. Conceptul de suprafață termică se poate folosi la proiectarea termică a plăcilor cu circuite integrate fără componente de răcire active. În procesul de proiectare se pornește de la valorile

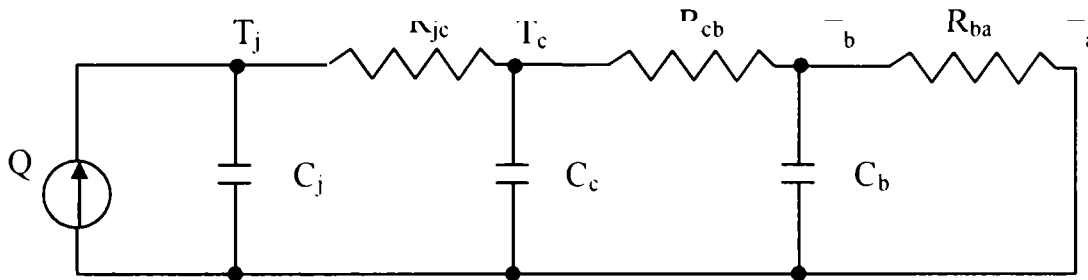


Figura 3.6 – Rețeaua termică ce descrie transferul de căldură de la jonțiune prin capsulă și placă la mediul ambiant pentru cazul prezentat în figura 3.3

Implementarea în SPICE a acestui circuit de simulare termică este prezentată mai jos:

```
.SUBCKT THERMSIM Tj Tc Tb Ta VDD GND
* microarchitecture thermal simulation
* nodes Tj, Tc, Tb, Ta
.param rthjc=2.5 rthcb=3 rthba=1
.param cthj=100n cthc=10u cthb=100u
F1 Tj GND VDD 5
VA Ta GND DC=TEMP
RJC Tj Tc rthjc
RCB Tc Tb rthcb
RBA Th Ta rthba
CJ Tj GND cthj
CC Tc GND cthc
CB Th GND cthb
.ENDS THERMSIM
```

În realitate căldura degajată de un circuit integrat se disipă atât prin răcitor cât și prin placă. Din acest motiv, sa propus o nouă rețea termică care combină primele două variante, a cărei schemă este reprezentată în figura 3.7 iar implementarea în SPICE este prezentată în continuarea schemei. În acest model sa mai introdus un element prezent mai nou la circuitele integrate VLSI, în special la procesoare: o componentă de răcire încapsulată în circuit (spreader) pe care este montată pastila de siliciu, pentru a favoriza disiparea de căldura de la jonțiune la capsulă (sau în unele cazuri direct către modulele din exteriorul circuitului).

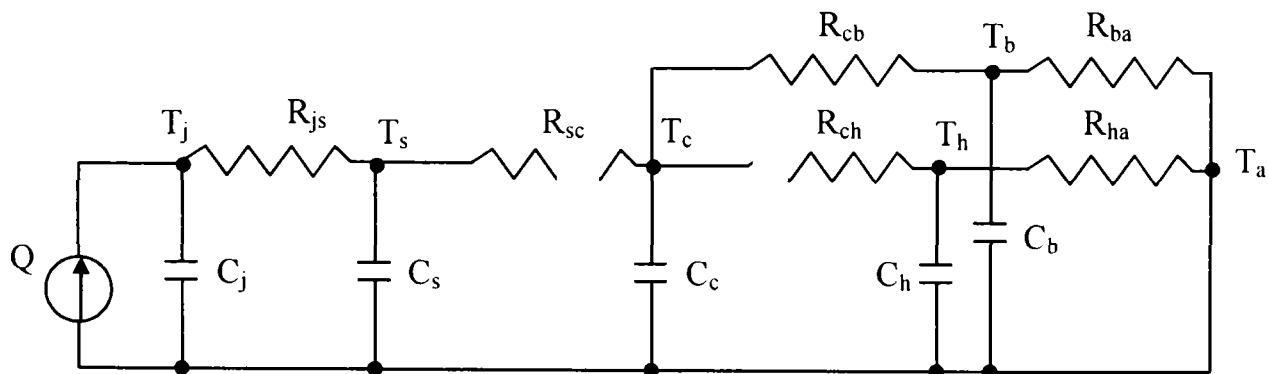


Figura 3.7 – Rețeaua termică ce descrie transferul de căldură pentru un circuit montat pe o placă PCB

În schema din figura 3.7 există următoarele elemente:

- $T_j$  – temperatura medie la nivelul pastilei de siliciu;
- $T_s$  – temperatura medie a componentei de răcire internă circuitului (spreader);
- $T_c$  – temperatura medie la nivelul capsulei;
- $T_h$  – temperatura medie la nivelul componentei de răcire externă circuitului (heatsink);
- $T_b$  – temperatura medie la nivelul plăcii;
- $R_{js}$  – rezistența termică între joncțiune și componenta de răcire internă;
- $R_{sc}$  – rezistența termică între componenta de răcire internă și circuit;
- $R_{ch}$  – rezistența termică între capsulă și componenta de răcire externă;
- $R_{cb}$  – rezistența termică între capsulă și placă;
- $R_{ba}$  – rezistența termică între placă și mediul ambiant;
- $R_{ha}$  – rezistența termică între componenta de răcire externă și mediul ambiant.

```
.SUBCKT THERMSIM Tj Ts Tc Th Tb Ta VDD GND
* microarchitecture thermal simulation
* nodes Tj, Ts, Tc, Th, Tb, Ta
.param rthjs=0.5 rthsc=1.1 rthch=2.5 rthcb=5 rthha=5 rthba=10
.param cthj=100n cths=1u cthc=10u cthh=20u cthb=50u
  F1  Tj GND VDD 5
  VA  Ta GND DC=TEMP
  RJS Tj Ts rthjs
  RSC Ts Tc rthsc
  RCH Tc Th rthch
  RCB Tc Tb rthcb
  RHA Th Ta rthha
  RBA Tb Ta rthba
  CJ  Tj GND cthj
  CS  Ts GND cths
  CC  Tc GND cthc
  CH  Th GND cthh
  CB  Tb GND cthb
.ENDS THERMSIM
```

O variantă simplificată a modelului propus este folosită în [Malhammar2002\_3] pentru implementarea conceptului de teritoriu sau suprafață termică. Suprafața termică se definește ca fiind cea mai mică suprafață rectangulară din PCB necesară pentru răcirea pasivă a unei componente (circuit).

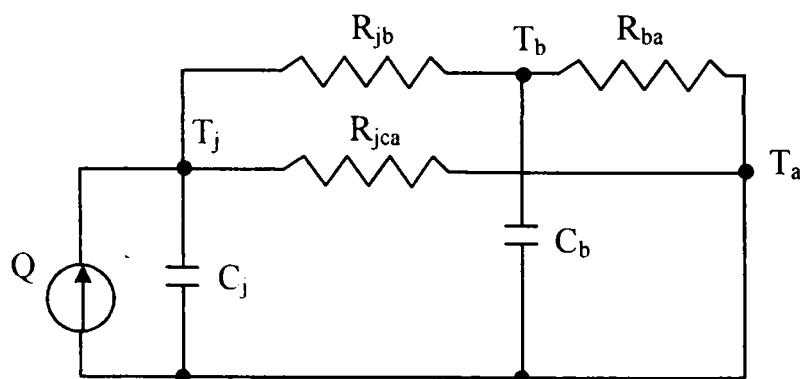


Figura 3.8 – Rețeaua termică ce modelează o suprafață termică

În cazul în care nu sunt componente de răcire active, o parte din căldura generată de circuit este disipată prin suprafața componentei iar o parte din căldură este disipată prin placă. Conceptul de suprafață termică se poate folosi la proiectarea termică a plăcilor cu circuite integrate fără componente de răcire active. În procesul de proiectare se pornește de la valorile



cunoscute și anume  $T_j$  pentru un circuit trebuie păstrat sub o anumită valoare (dată de producător) pentru ca această să nu se deterioreze și  $R_{jca}$  și  $R_{jcb}$  ce sunt determinate în funcție de proprietățile materialelor folosite. Iar scopul proiectării termice este determinarea ariei de răcire necesare pentru fiecare circuit de pe placă.

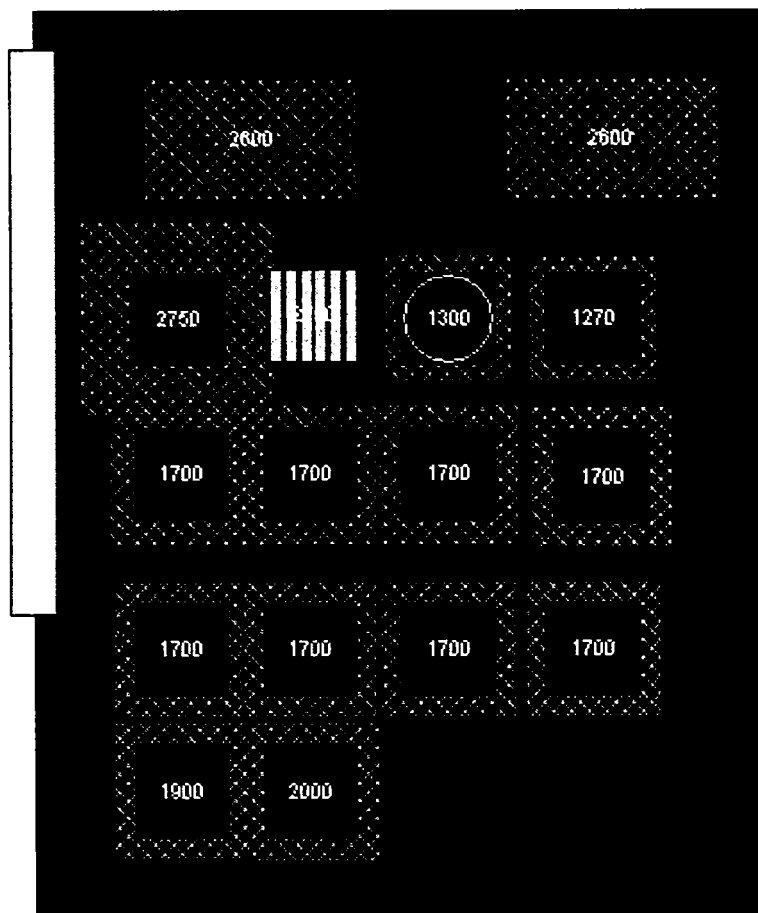


Figura 3.9 – Suprafețele termice pentru o placă cu circuite integrate

Un al doilea exemplu, pentru modul în care rezistențele termice sunt utilizate la caracterizarea termică a unei plăci cu circuite integrate sau a unui sistem de plăci în prezența răcirii active, este prezentat în figura 3.10 [Manno1997].

În figura 3.10, plăcile cu circuite integrate, identificate prin indexul  $k$  sunt montate în paralel. Aceste plăci pot fi divizate în un anumit număr de linii ( $i$ ) și coloane ( $j$ ), o secțiune poate corespunde regiunii din vecinătatea unei componente. Regiunile comunică termic prin conducție, convecție și radiație. Într-un model termic integrat sunt necesare trei părți structurale:

- o unitate de modelare termică a regiunilor (TMU – Thermal Modeling Unit);
- o metodologie pentru integrarea comunicației termice între TMU-uri;
- o strategie computațională pentru estimarea soluțiilor setului de ecuații combinate.

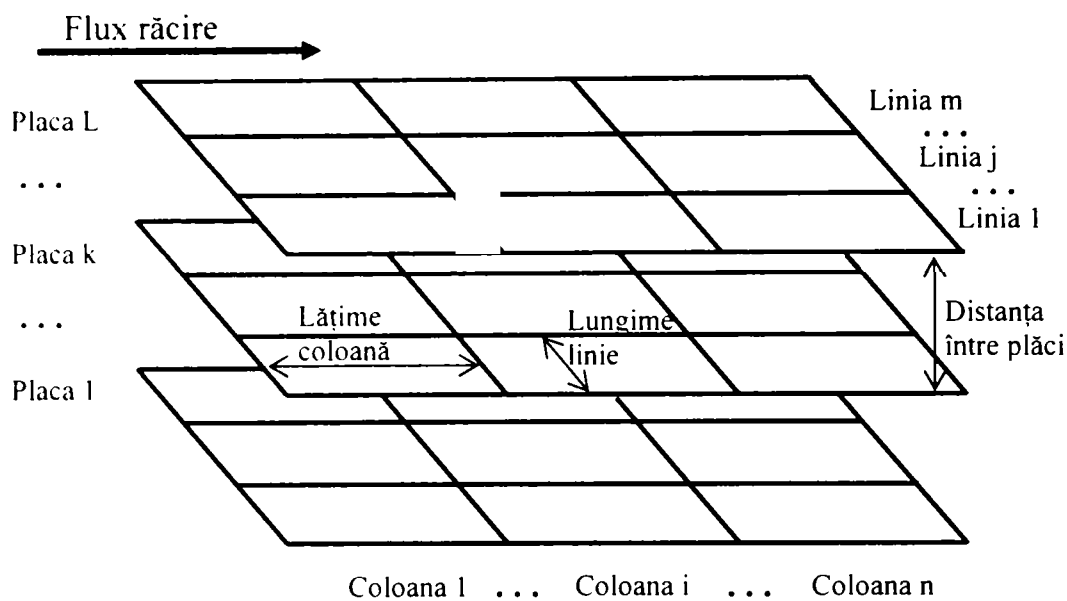


Figura 3.10 – Discretizarea unui set de plăci paralele [Manno1997]

La nivelul unei componente de pe o placă (k) descrierea modurilor de transfer de căldură este prezentată în figura 3.11. Fiecare TMU e compus din o componentă, placă și regiunea de răcire asociată. Mai multe ecuații descriu acest TMU, numărul acestora variază în funcție de complexitatea modelului. TMU include temperaturile caracteristice care-i definesc starea termică și care furnizează parametrii de interfațare între unități. Numărul temperaturilor caracteristice definește numărul ecuațiilor care trebuie rezolvate per TMU.

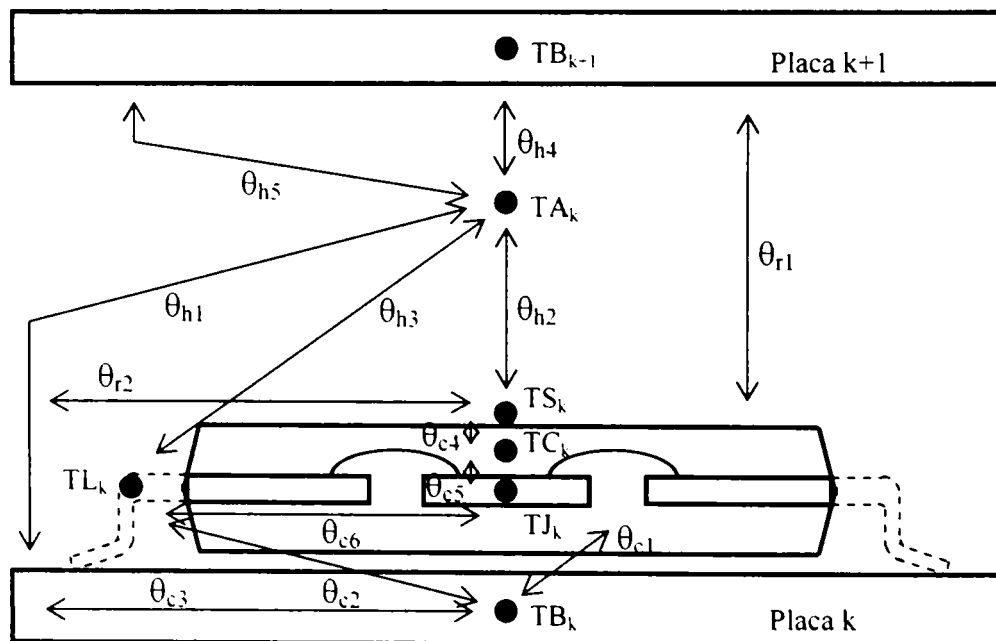


Figura 3.11 – Mecanismele locale de transfer de căldură [Manno1997]

Ecuațiile de echilibru pentru punctele din figura 3.11, considerate pentru un circuit i, j de pe placa k sunt:

- pentru  $TB_{ijk}$ :

$$\theta_{h1}(TA_k - TB_k) + \theta_{c1}(TC_k - TB_k) + \theta_{h4}(TA_{k-1} - TB_k) + \theta_{c2}(TL_k - TB_k) + \sum \theta_{c3}(TB_n - TB_k) = 0$$

n fiind vecinii elementului i,j,k.

- pentru  $TL_{ijk}$ :

$$\theta_{h3}(TA_k - TL_k) + \theta_{c2}(TB_k - TL_k) + \theta_{c6}(TJ_k - TL_k) + \theta_{h5}(TA_{k-1} - TL_k) = 0$$

- pentru  $TJ_{ijk}$ :

$$\theta_{c5}(TC_k - TJ_k) + \theta_{c6}(TL_k - TJ_k) = \dot{Q}$$

- pentru  $TC_{ijk}$ :

$$\theta_{c1}(TB_k - TC_k) + \theta_{c4}(TS_k - TC_k) + \theta_{c5}(TJ_k - TC_k) = 0$$

- pentru  $TS_{ijk}$ :

$$\theta_{h2}(TA_k - TS_k) + \theta_{c4}(TC_k - TS_k) + \theta_{r1}(TB_{k+1} - TS_k) + \sum \theta_{r2}(TS_n - TS_k) = 0$$

- pentru  $TA_{ijk}$ :

$$\theta_{h1}(TB_k - TA_k) + \theta_{h2}(TS_k - TA_k) + \theta_{h3}(TL_k - TA_k) + \theta_{h4}(TB_{k+1} - TA_k) + \theta_{h5}(TL_{k+1} - TA_k) =$$

$$\rho C_p \left[ \left( F TA_k - F_{t-1} TA_{t-1k} \right) + \chi \right]$$

unde:

TB ( $TB_{ijk}$ ) – temperatura stabilită pe placa k la nivelul circuitului i,j;

TA ( $TA_{ijk}$ ) – temperatura aerului la nivelul plăcii k și circuitului i,j;

TJ ( $TJ_{ijk}$ ) – temperatura joncțiunii circuitului i,j de pe placa k;

TC ( $TC_{ijk}$ ) – temperatura capsulei circuitului i,j de pe placa k;

TS ( $TS_{ijk}$ ) – temperatura suprafeței circuitului i,j de pe placa k;

TL ( $TL_{ijk}$ ) – temperatura pinilor circuitului i,j de pe placa k;

$\theta_{c1}$  ( $\theta_{c1ijk}$ ) – conductanța termică datorată conducției între placă și capsula circuitului;

$\theta_{c2}$  ( $\theta_{c2ijk}$ ) – conductanța termică datorată conducției între placă și pinii circuitului;

$\theta_{c3}$  ( $\theta_{c3ijk}$ ) – conductanța termică datorată conducției la nivelul unei plăci;

$\theta_{c4}$  ( $\theta_{c4ijk}$ ) – conductanța termică datorată conducției între capsula circuitului și suprafața circuitului;

$\theta_{c5}$  ( $\theta_{c5ijk}$ ) – conductanța termică datorată conducției între capsula circuitului și joncțiunea circuitului;

$\theta_{c6}$  ( $\theta_{c6ijk}$ ) – conductanța termică datorată conducției între joncțiunea circuitului și pinii circuitului;

$\theta_{h1}, \theta_{h4}$  ( $\theta_{h1ijk}, \theta_{h4ijk+1}$ ) – conductanța termică datorată convecției la nivelul plăcii;

$\theta_{h2}$  ( $\theta_{h2ijk}$ ) – conductanța termică datorată convecției la nivelul suprafeței circuitului;

$\theta_{h3}, \theta_{h5}$  ( $\theta_{h3ijk}, \theta_{h5ijk+1}$ ) – conductanța termică datorată convecției la nivelul pinilor circuitului;

$\theta_{r1}$  ( $\theta_{r1ijk}$ ) – conductanța termică datorată radiației între circuit și placă;

$\theta_{r2}$  ( $\theta_{r2ijk}$ ) – conductanța termică datorată radiației între circuitele vecine;

În cazul în care există o sursă de răcire forțată, trebuie aleasă o direcție dominantă, pe care se cunoaște sau se estimează fluxul volumetric (F). Conducția între unități (c) și schimburile radiative (r) se presupune că sunt dominate de interacțiuni cu unitățile din imediata vecinătate. Schimburile radiative trebuie să mai includă transferul placă la placă. Schimburile de tip convecție (h) descriu transferurile de căldură între suprafață și ambientul local.

### 3.1.2 Analiza numerică a câmpurilor staționare de temperaturi

Analiza numerică a câmpurilor staționare de temperaturi se face prin aplicarea diferențelor finite la ecuațiile diferențiale ale câmpului și prin rezolvarea sistemului de ecuații liniare care se obține pe această cale. În afara metodelor clasice, metoda plurilocală dă rezultate mai exacte și are ca bază o asociere între dezvoltarea în serie Taylor a unei funcții și ecuația ei diferențială.

Rezolvarea ecuațiilor de tip Poisson

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = -F(x, y)$$

se poate face și prin metoda diferențelor finite. Considerându-se o rețea pătratică  $\Delta x = \Delta z = d$  se dezvoltă în serie Taylor funcția  $T(x, y)$

$$T(x, y + d) = T_{m, n+1} = T_{m, n} + d \cdot T'_y + \frac{d^2}{2} \cdot T''_y + \frac{d^3}{6} \cdot T'''_y + \dots$$

$$T(x, y - d) = T_{m, n-1} = T_{m, n} - d \cdot T'_y + \frac{d^2}{2} \cdot T''_y - \frac{d^3}{6} \cdot T'''_y + \dots$$

$$T(x + d, y) = T_{m+1, n} = T_{m, n} + d \cdot T'_x + \frac{d^2}{2} \cdot T''_x + \frac{d^3}{6} \cdot T'''_x + \dots$$

$$T(x - d, y) = T_{m-1, n} = T_{m, n} - d \cdot T'_x + \frac{d^2}{2} \cdot T''_x - \frac{d^3}{6} \cdot T'''_x + \dots$$

Prin adunare se obține operatorul:

$$\begin{aligned} \Phi_1 &= T_{m, n+1} + T_{m, n-1} + T_{m+1, n} + T_{m-1, n} - 4T_{m, n} = \\ &= d^2 (T''_x + T''_y) + \frac{d^4}{12} (T^{(4)}_x + T^{(4)}_y) + \frac{d^6}{360} (T^{(6)}_x + T^{(6)}_y) + \dots \end{aligned}$$

Știind că pentru  $x = m$  și  $y = n$ :

$$T''_x + T''_y = -F_{m, n}$$

se obține:

$$\Phi_1 = T_{m, n+1} + T_{m, n-1} + T_{m+1, n} + T_{m-1, n} - 4T_{m, n} = -d^2 \cdot F_{m, n} - \varepsilon_3$$

în care restul  $\varepsilon_3$ , ce se poate neglija, este de forma:

$$\varepsilon_3 = \frac{d^4}{12} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) F_{m, n} + \frac{d^6}{360} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right)^{(2)} F_{m, n} + \dots$$

Sistemul de ecuații liniare poate fi construit pe baza relației:

$$T_{m, n+1} + T_{m, n-1} + T_{m+1, n} + T_{m-1, n} = 4T_{m, n} - d^2 \cdot F_{m, n}$$

care pentru ecuații de tip Laplace ( $F_{m, n} = 0$ ) conduce la:

$$4T_{m, n} = T_{m, n+1} + T_{m, n-1} + T_{m+1, n} + T_{m-1, n}$$

molecula de calcul fiind reprezentată în figura de mai jos.

În cazul în care se urmărește un grad de exactitate mai mare, se utilizează și dezvoltările în serie:

$$T_{m-1,n+1} = T_{m,n} - d(T'_x - T'_y) + \frac{d^2}{2}(T''_x + T''_y) - \frac{d^3}{6}(T'''_x - T'''_y) + \frac{d^4}{24}(T^{(4)}_x + T^{(4)}_y) - \dots$$

$$T_{m-1,n-1} = T_{m,n} - d(T'_x + T'_y) + \frac{d^2}{2}(T''_x + T''_y) - \frac{d^3}{6}(T'''_x + T'''_y) + \frac{d^4}{24}(T^{(4)}_x + T^{(4)}_y) - \dots$$

$$T_{m+1,n+1} = T_{m,n} + d(T'_x + T'_y) + \frac{d^2}{2}(T''_x + T''_y) + \frac{d^3}{6}(T'''_x + T'''_y) + \frac{d^4}{24}(T^{(4)}_x + T^{(4)}_y) + \dots$$

$$T_{m+1,n-1} = T_{m,n} + d(T'_x - T'_y) + \frac{d^2}{2}(T''_x + T''_y) + \frac{d^3}{6}(T'''_x - T'''_y) + \frac{d^4}{24}(T^{(4)}_x + T^{(4)}_y) + \dots$$

care prin însumare conduc la operatorul:

$$\begin{aligned} \Phi_2 &= T_{m+1,n+1} + T_{m-1,n+1} + T_{m+1,n-1} + T_{m-1,n-1} - 4T_{m,n} = \\ &= 2d^2(T''_x + T''_y) + \frac{d^4}{6}(T^{(4)}_x + T^{(4)}_y) + \frac{d^6}{180}(T^{(6)}_x + T^{(6)}_y) + \dots \end{aligned}$$

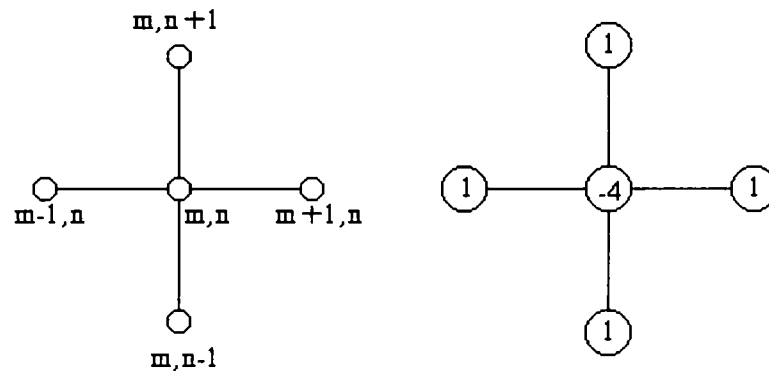


Fig 3.12

După gradul de precizie urmărit pot fi utilizate combinații liniare între operatorii  $\Phi_1$  și  $\Phi_2$

$$\Phi = a\Phi_1 + b\Phi_2$$

în care  $a$  și  $b$  sunt numere întregi și pozitive. Pentru cazul particular  $a = 4$ ;  $b = 1$  și utilizând ecuația Poisson se obține:

$$\begin{aligned} &4(T_{m+1,n} + T_{m-1,n} + T_{m,n+1} + T_{m,n-1}) + T_{m+1,n+1} + T_{m-1,n-1} + T_{m-1,n+1} + T_{m+1,n-1} = \\ &= 20T_{m,n} - 6d^2 F_{m,n} - \varepsilon_4 \end{aligned}$$

unde restul neglijabil  $\varepsilon_4$  este dat de expresia:

$$\varepsilon_4 = \frac{d^4}{2} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) F_{m,n} + \frac{d^6}{60} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right)^{(2)} F_{m,n} + \dots$$

Pentru ecuații de tip Laplace ( $F_{m,n}=0$ ) se obține:

$$\begin{aligned} 20T_{m,n} &= 4(T_{m+1,n} + T_{m-1,n} + T_{m,n+1} + T_{m,n-1}) + \\ &+ T_{m+1,n+1} + T_{m-1,n-1} + T_{m-1,n+1} + T_{m+1,n-1} \end{aligned}$$

cu ajutorul căreia se pot scrie sistemele de ecuații liniare.

Molecula de calcul a laplaceanului în metoda plurilocală pentru spațiul cu două dimensiuni este prezentată în figura 3.13.

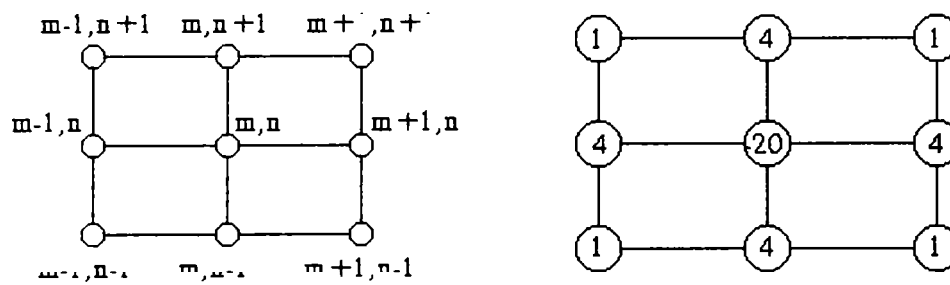


Fig 3.13

### 3.1.3 Estimarea puterii disipate de circuitele integrate pe baza termogramei

În acest paragraf voi descrie o modalitate, relativ simplă, de estimare a puterii disipate de un circuit pe baza imaginii în infraroșu a circuitului respectiv. Pentru aceasta se pleacă de la ecuația generală a conducției căldurii care se particularizează pentru cazul bidimensional asociat pentru o placă cu circuite integrate.

Ecuația generală a conducției căldurii este:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} + \frac{q_i}{k} = \frac{1}{\alpha} \cdot \frac{\partial T}{\partial t}$$

unde

$$\alpha = \frac{k}{c\rho}$$

k - conductivitatea termică a materialului;

c - căldura specifică;

$\rho$  - densitatea materialului;

$q_i$  - căldura incidentă pe unitatea de volum.

S-a considerat că k, c,  $\rho$  sunt independente de temperatură.

În cazul unei plăci plane, care este supusă unui flux de căldură incidentă pe unitatea de suprafață  $q_i$ , avem:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{q_i}{k\delta_0} = 0$$

unde s-au făcut următoarele reduceri:

$$\frac{\partial^2 T}{\partial z^2} = 0$$

ne referim la o placă plană;

$$\frac{\partial T}{\partial t} = 0$$

s-a considerat că temperatura nu variază cu timpul (echilibrul termic);

$\delta_0$  - grosimea plăcii.

În continuare se va prezenta o metodă numerică de rezolvare a acestei ecuații diferențiale, cunoscând temperaturile la marginea plăcii (considerăm o placă având o singură sursă de căldură).

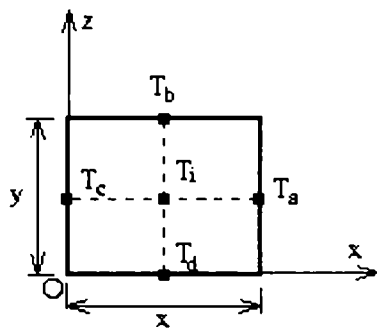


Fig 3.14

Cunoscând temperaturile la marginea plăcii se poate afla temperatura oricărui punct de pe placă. Ca exemplu calculăm temperatura în centrul plăcii  $T_i$ , în funcție de temperaturile de la mijloacele laturilor  $T_a$ ,  $T_b$ ,  $T_c$ ,  $T_d$ . Pentru aceasta folosim dezvoltarea în serie Taylor a funcției  $T(x, y)$ , funcție ce reprezintă distribuția temperaturilor pe placă.

$$T(x, y) = T_i + (x - x_i) \left( \frac{\partial T}{\partial x} \right)_i + (y - y_i) \left( \frac{\partial T}{\partial y} \right)_i + \frac{1}{2} (x - x_i)^2 \left( \frac{\partial^2 T}{\partial x^2} \right)_i + \\ + \frac{1}{2} (y - y_i)^2 \left( \frac{\partial^2 T}{\partial y^2} \right)_i + \frac{1}{2} (x - x_i)(y - y_i) \left( \frac{\partial^2 T}{\partial x \partial y} \right)_i + \dots$$

Din această dezvoltare vom lua doar primii șase termeni:

$$T(x, y) \cong T_i + (x - x_i) \left( \frac{\partial T}{\partial x} \right)_i + (y - y_i) \left( \frac{\partial T}{\partial y} \right)_i + \\ + \frac{1}{2} (x - x_i)^2 \left( \frac{\partial^2 T}{\partial x^2} \right)_i + \frac{1}{2} (y - y_i)^2 \left( \frac{\partial^2 T}{\partial y^2} \right)_i$$

$T_i$  - temperatura în punctul  $i$ ;

$x_i, y_i$  - coordonatele punctului  $i$ .

Înlocuim în dezvoltarea de mai sus valorile pentru punctele a, b, c și d.

$$\Rightarrow T_a \cong T_i + (x_a - x_i) \left( \frac{\partial T}{\partial x} \right)_i + (y_a - y_i) \left( \frac{\partial T}{\partial y} \right)_i + \\ + \frac{1}{2} (x_a - x_i)^2 \left( \frac{\partial^2 T}{\partial x^2} \right)_i + \frac{1}{2} (y_a - y_i)^2 \left( \frac{\partial^2 T}{\partial y^2} \right)_i$$

$$x_a - x_i = \frac{x}{2}$$

$$y_a - y_i = 0$$

$$\Rightarrow T_a = T_i + \frac{x}{2} \left( \frac{\partial T}{\partial x} \right)_i + \frac{x^2}{8} \left( \frac{\partial^2 T}{\partial x^2} \right)_i$$

Analog:

$$T_b = T_i + \frac{y}{2} \left( \frac{\partial T}{\partial y} \right)_i + \frac{y^2}{8} \left( \frac{\partial^2 T}{\partial y^2} \right)_i$$

$$T_c = T_i - \frac{x}{2} \left( \frac{\partial T}{\partial x} \right)_i + \frac{x^2}{8} \left( \frac{\partial^2 T}{\partial x^2} \right)_i$$

$$T_d = T_i - \frac{y}{2} \left( \frac{\partial T}{\partial y} \right)_i + \frac{y^2}{8} \left( \frac{\partial^2 T}{\partial y^2} \right)_i$$

Cu cele patru relații de mai sus se formează un sistem de patru ecuații cu patru necunoscute, necunoscutele fiind:

$$\left( \frac{\partial T}{\partial x} \right)_i, \left( \frac{\partial T}{\partial y} \right)_i, \left( \frac{\partial^2 T}{\partial x^2} \right)_i, \left( \frac{\partial^2 T}{\partial y^2} \right)_i.$$

Rezolvând sistemul obținem:

$$\left( \frac{\partial T}{\partial x} \right)_i = \frac{T_a - T_c}{x}$$

$$\left( \frac{\partial T}{\partial y} \right)_i = \frac{T_b - T_d}{y}$$

$$\left( \frac{\partial^2 T}{\partial x^2} \right)_i = \frac{4}{x^2} (T_a + T_c - 2T_i)$$

$$\left( \frac{\partial^2 T}{\partial y^2} \right)_i = \frac{4}{y^2} (T_b + T_d - 2T_i)$$

Aceste valori se vor introduce în ecuația diferențială a distribuției căldurii, pentru punctul  $i$ .

$$\left( \frac{\partial^2 T}{\partial x^2} \right)_i + \left( \frac{\partial^2 T}{\partial y^2} \right)_i + \frac{q_i}{k\delta_0} = 0$$

$$\Rightarrow \frac{4}{x^2} (T_a + T_c - 2T_i) + \frac{4}{y^2} (T_b + T_d - 2T_i) + \frac{q_i}{k\delta_0} = 0$$

iar

$$q_i = \frac{P}{xy}$$

$$\Rightarrow \frac{P}{xy\delta_0 k} = \frac{4}{x^2} (2T_i - T_a - T_c) + \frac{4}{y^2} (2T_i - T_b - T_d)$$

Odată ajunși la această formulă, putem trece la cazul nostru concret. Sunt cunoscute în cazul nostru, de pe termogramă, temperaturile în orice punct al plăcii.

Astfel putem folosi relația de mai sus în aflarea puterii disipare de un circuit integrat de pe placă.

$$\Rightarrow P = 4k\delta_0 \left[ \frac{y}{x} (2T_i - T_a - T_c) + \frac{x}{y} (2T_i - T_b - T_d) \right] \quad (3.1)$$



### 3.1.4 Generarea imaginilor termice ideale

Pentru modelarea combinată a conducției și convecției, simularea și estimarea transferului de căldură la nivel de componentă, placă și sistem există două mari clase de metode:

- metode numerice:
  - metoda diferențelor finite (Finite Differences Method – FDM);
  - metoda elementelor finite (Finite Elements Method – FEM);
  - metoda elementelor marginale (Boundary Elements Method – BEM);
- metode analitice:
  - metoda seriilor Fourier;
  - metoda funcțiilor Bessel;
  - metoda funcțiilor Gauss.

Toate metodele numerice pot rezolva cu o acuratețe foarte bună ecuațiile de tip Laplace/Poisson, care stau la baza disipării căldurii prin conducție și convecție. În fiecare din cele trei metode numerice, pentru a se obține soluțiile cu o anumită precizie sunt necesare resurse computaționale și de memorare foarte mari, astfel că de cele mai multe ori se folosesc doar în probleme locale. Metodele numerice oferă flexibilitate și acuratețe, însă sunt complexe și foarte lente.

Pe de altă parte metodele analitice sunt rapide și ușor de implementat însă au o flexibilitate redusă, în sensul că sunt specifice pentru un anumit tip de arhitectură fizică. Iar în ceea ce privește acuratețea aceasta depinde de numărul termenilor folosiți în estimare, adică trebuie recurs la un compromis între acuratețe și timpul de calcul.

În procesul de testare termo-optică parametrul cel mai important este viteza, astfel că s-a optat pentru metodele analitice de simulare, având în vedere și că arhitectura fizică a sistemelor este cunoscută: plăcile echipate cu circuite integrate. Un circuit plasat pe o placă este reprezentat schematic în figura 3.15.

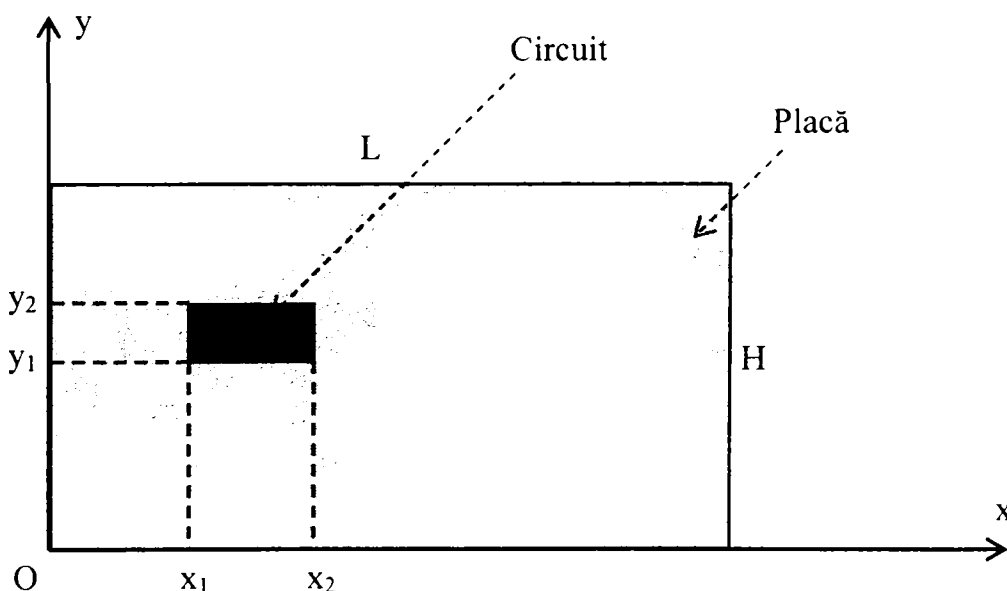


Figura 3.15 – O sursă de căldură plasată pe o placă

Fiecare circuit este identificat prin câțiva parametri: dimensiune, poziția pe placă, puterea disipată și coeficientul de conducție termică (figura 3.16). În graficele din figura 3.16 sunt secțiunile transversale pe axa Ox și respectiv Oy prin funcția bidimensională de distribuție termică. În fiecare din metodele analitice amintite funcția de distribuție a

temperaturilor pe placă este aproximată prin diferite mecanisme: serii Fourier, funcții Bessel sau funcții Gauss.

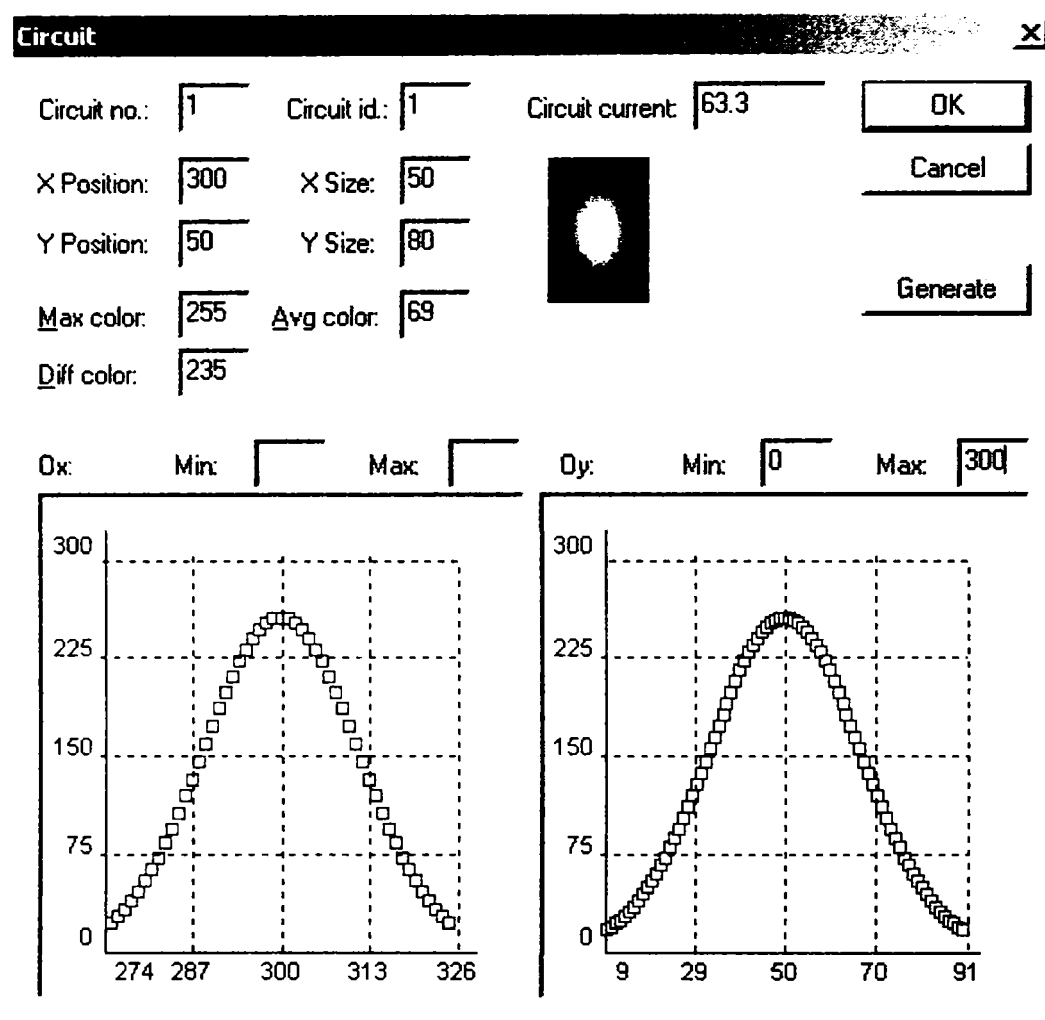


Figura 3.16 – Parametrii termo-optici ai unui circuit integrat

Estimarea distribuției de temperaturi și generarea unei imagini termice pe baza de serii Fourier ([Malhammar2002\_1]) are la bază următoarea funcție:

$$\tau(x, y) = \frac{\gamma}{\kappa^2} \cdot \sum_{n=0}^N f(n, y) \cdot A_n \cdot \cos(n\varpi_x x)$$

$$f(n, y) = \sum_{m=0}^M B_{mn} \cdot \cos(m\varpi_y y)$$

unde:

$$\gamma = \frac{q}{k\delta}; \kappa = \sqrt{\frac{2h}{k\delta}}; \varpi_x = \frac{\pi}{L}; \varpi_y = \frac{\pi}{H}$$

$$A_0 = \frac{\varpi_x \cdot (x_2 - x_1)}{\pi}; A_n = \frac{2 \cdot [\sin(n\varpi_x x_2) - \sin(n\varpi_x x_1)]}{n \cdot \pi}$$

$$B_{0n} = \frac{\varpi_x \cdot (y_2 - y_1)}{\pi \cdot \left[ \left( \frac{n\varpi_x}{\kappa} \right)^2 + 1 \right]}; B_{mn} = \frac{2 \cdot [\sin(m\varpi_y y_2) - \sin(m\varpi_y y_1)]}{m \cdot \pi \cdot \left[ \left( \frac{n\varpi_x}{\kappa} \right)^2 + \left( \frac{m\varpi_y}{\kappa} \right)^2 + 1 \right]}$$

$$N \geq \frac{3L}{x_2 - x_1}; M \geq \frac{3H}{y_2 - y_1}$$

q – fluxul de căldură degajat de sursă [W/m<sup>2</sup>];

k – conductivitatea termică [W/mK];

h – coeficientul de transfer termic [W/m<sup>2</sup>K];

δ – grosimea plăcii [m].

Esimarea distribuției de temperaturi și generarea unei imagini termice folosind funcții Gauss:

$$g(x, y) = a \cdot e^{-\left(\frac{x^2}{w^2} + \frac{y^2}{h^2}\right)} + g_0$$

Aceste funcții se aplică pentru fiecare sursă de căldură (circuit integrat) de pe placă, astfel că se obțin mai multe suprafețe termice, câte una pentru fiecare sursă. Combinarea acestora se face prin superpoziție liniară [Sofia1996, Malhammar2002\_1, Altet2001] sau însumare. Un exemplu schematic pentru superpoziție este prezentat în figura 3.17.

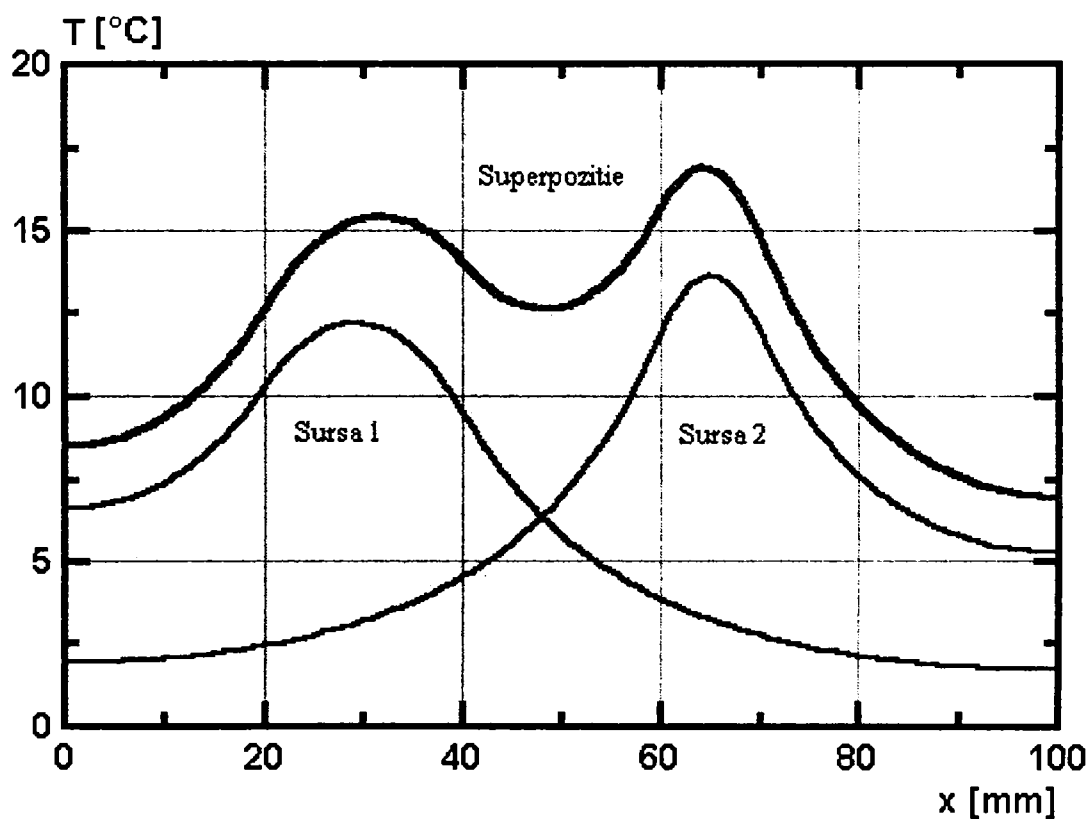


Figura 3.17 – Convoluția funcțiilor termice pentru două surse de căldură

Un exemplu de simulare termică cu serii Fourier pentru o placă cu 8 circuite integrate este prezentat în figura 3.18. Fiecare circuit are o anumită dimensiune și disipă o anumită putere.

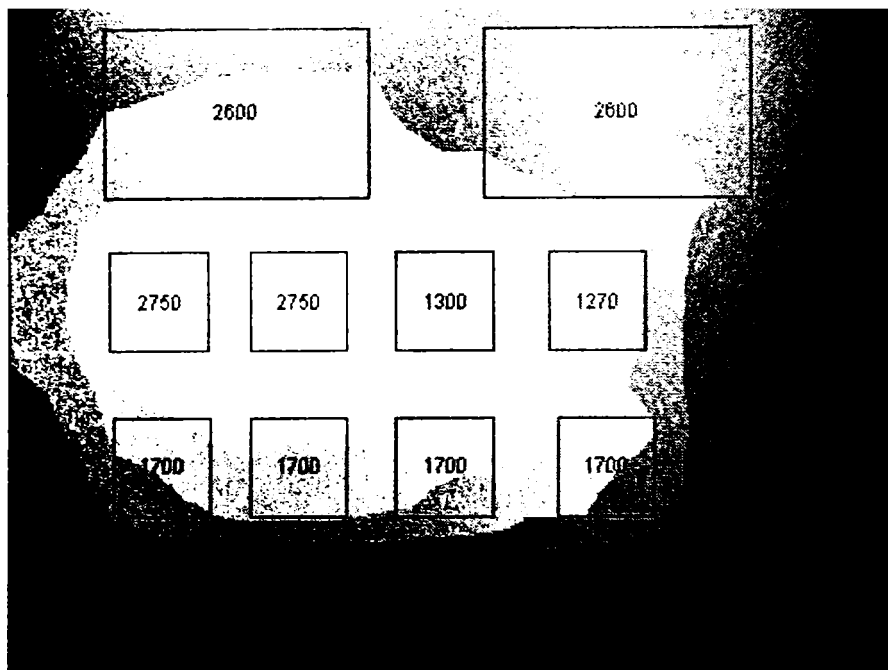


Figura 3.18 – Profilul termic obținut prin serii Fourier pentru un exemplu de PCB

O altă metodă de generare a imaginilor termice este cea propusă în [Vladutiul996]. Această metodă utilizează convoluția a două imagini: o primă imagine conține câte un pixel pe poziția fiecărui circuit de pe placă iar culoarea pixelului este proporțională cu puterea disipată de respectivul circuit și a doua imagine este dată de funcția termică de disipare a căldurii pentru un circuit. Dacă toate circuitele de pe placă au aceeași dimensiune și au același tip de capsulă atunci există o singură funcție de disipare a căldurii. Dacă nu, atunci se vor grupa circuitele echivalente și procesul de convoluție se va repeta pentru toate aceste clase, iar în final imaginile obținute se vor însuma pentru a obține imaginea termică finală.

Convoluția a două funcții  $f(x,y)$  și  $g(x,y)$ , notată cu  $f(x,y) \circ g(x,y)$ , este definită prin relația de mai jos (3.2):

$$f(x,y) \circ g(x,y) = \int_x \int_x f(\alpha,\beta) \cdot g(x-\alpha,y-\beta) d\alpha d\beta \quad (3.2)$$

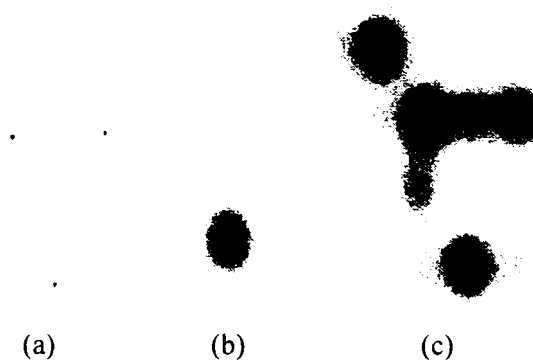


Figura 3.19 – Exemplu de convoluționare a doua imagini termice  
(a) – imaginea  $f$ , (b) – imaginea  $g$ , (c) – imafinea  $h = f \circ g$

Un exemplu de imagine termică generată prin metoda convoluției este prezentat în figura 3.20.



Figura 3.20 – Imaginea termică ideală pentru o placă cu circuite

### 3.2 Deconvoluția imaginilor termice

Convoluția a două imagini  $f(x,y)$  și  $g(x,y)$ , notată cu  $f(x,y) \circ g(x,y)$ , este definită prin relația:

$$h(x,y) = f(x,y) \circ g(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha,\beta) \cdot g(x-\alpha, y-\beta) d\alpha d\beta \quad (3.2)$$

Problema deconvoluției presupune găsirea imaginii  $f$  (sau  $g$ ) dacă se cunosc imaginile  $h$  și  $g$  (respectiv  $h$  și  $f$ ). Soluționarea acestei probleme este relativ simplă dacă se utilizează teorema convoluției. Teorema convoluției afirmă: Dacă  $f(x,y)$  are transformata Fourier  $F(u,v)$ , iar  $g(x,y)$  are transformata Fourier  $G(u,v)$  atunci  $f(x,y) \circ g(x,y)$  are transformata Fourier  $F(u,v)G(u,v)$ . Astfel că deconvoluția (inversa convoluției) a două imagini se poate realiza prin împărțirea transformatelor Fourier ale celor două imagini și apoi aplicând rezultatului transformata Fourier inversă.

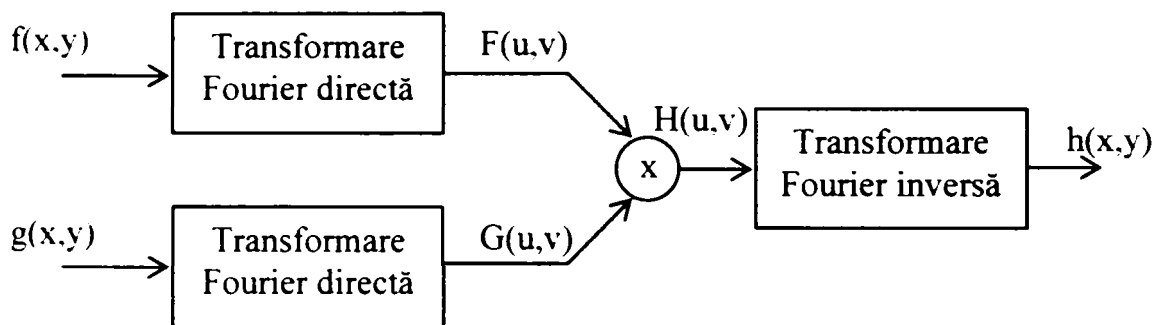


Figura 3.21 – Teorema convoluției

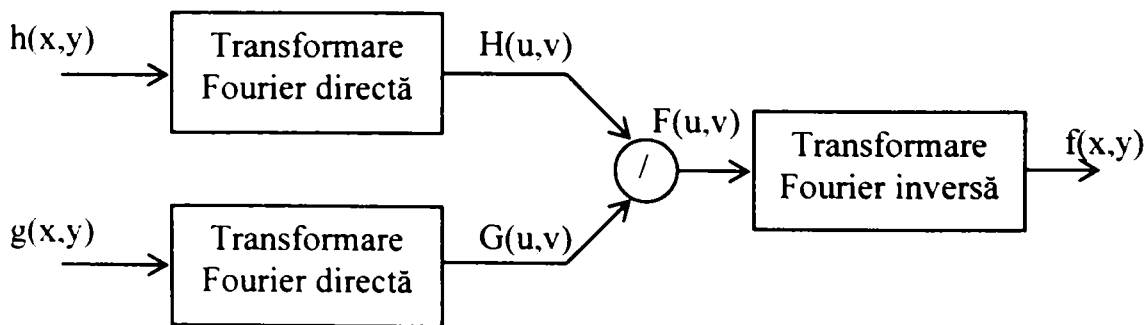


Figura 3.22 – Deconvoluția

Probleme la deconvoluție apar atunci când nu se cunoaște exact funcția  $g$  sau dacă funcția  $h$  este distorsionată sau conține zgomote. În aceste cazuri deconvoluția nu se poate realiza pe calea prezentată, însă se poate utiliza o metodă de deconvoluție oarbă (blind deconvolution). Aceste metode de deconvoluție au drept scop aflarea atât a funcției  $f$  cât și a funcției  $g$  plecând doar de la funcția  $h$ .

Imaginile termice sunt influențate de poziția și sensibilitatea camerei video în infraroșu cu care acestea sunt achiziționate. Se consideră imaginea  $h$  ca fiind imaginea termică ideală a plăcii monitorizate de o cameră video iar imaginea achiziționată de cameră o notăm cu  $h'$ . Relația între imaginile  $h$  și  $h'$  este dată de ecuația 3.3:

$$h'(x,y) = h(x,y) \circ b(x,y) + n(x,y) \quad (3.3)$$

unde  $b$  este o funcție de aplatizare a imaginii (blur) iar  $n$  este o funcție ce conține zgomotele introduse de procesul de achiziție a imaginii termice. Înlocuind funcția  $h$  cu convoluția dintre  $f$  și  $g$  se obține relația (3.4).

$$h'(x, y) = f(x, y) \circ g(x, y) \circ b(x, y) + n(x, y) \quad (3.4)$$

Având în vedere că în realitate imaginile folosite conțin zgomote, părerea mea e că ar fi foarte necesară căutarea unei metode de soluționare a problemei deconvoluției imaginilor termice ale plăcilor cu circuite integrate. Metoda propusă și detaliată în subcapitolul următor face parte din clasa tehnicilor de deconvoluție oarbă.

Deconvoluția oarbă a fost introdusă pentru prima dată în domeniul procesărilor de imagini de către Ayers și Dainty în [Ayers1988]. Deconvoluția oarbă a imaginilor este procesul de identificare a imaginilor  $f$  și  $g$  din imaginea degradată  $h'$ . Majoritatea metodelor de deconvoluție constau în aplicarea de filtre singulare sau multiple pe imaginea degradată [Holmes2002]. Deconvoluția oarbă este aplicată în cele mai multe cazuri la imaginile spațiale ale diferitelor puncte din univers obținute cu ajutorul telescoapelor [Ayers1988, Holmes2002]. Nu există la momentul actual nici o metodă de deconvoluție pentru imaginile termice ale plăcilor cu circuite integrate iar pentru imaginile termice în general s-a găsit doar un singur articol ce tratează și încearcă să reducă influența poziției camerei video asupra imaginii termice [Meinders1999]. În acest articol se folosește un filtru Wiener pentru reducerea influenței unghiului de poziționare a camerei asupra imaginii. Algoritmul genetic a fost și el aplicat în deconvoluția imaginilor în [Chen1997, Liang1997].

### 3.2.1 Algoritm pseudo-genetic pentru deconvoluția imaginilor termice

În continuare voi prezenta o soluție pentru problema deconvoluției a două imagini. Acest algoritm de deconvoluție a fost folosit pentru deconvoluția imaginilor termice prin aproximarea disipării de căldură printr-o distribuție gauss-iană.

Propunerea mea este un nou algoritm de căutare pe baze genetice a unei soluții optime pentru deconvoluția a două entități (imagini, dar se poate aplica și semnalelor). Exemplul de la care s-a pornit a fost deconvoluția unei imagini cu “clopotul” lui Gauss. Aplicabilitatea acestei operații vizează deconvoluția imaginilor termice, în special pentru plăcile cu circuite integrate.

Definirea problemei: considerăm o imagine ce conține mai multe puncte (funcția  $f$ ) – figura 3.23 – și o imagine ce conține o distribuție gauss-iană normală (funcția  $g$ ) – figura 3.24. Dacă se convoluționează cele două imagini se obține o imagine (figura 3.25) cu câte o distribuție gauss-iană de o anumită înălțime pe poziția fiecărui punct din figura 3.23, funcția  $h$ . Se cere să se găsească imaginea cu spike-uri ( $f$ ) dacă se cunosc  $h$  și  $g$ , însă imaginea  $h$  este distorsionată și conține zgomote. Algoritmul propus poate fi utilizat și dacă nu se cunoaște funcția  $g$ , dar se știe că este o distribuție gauss-iană.

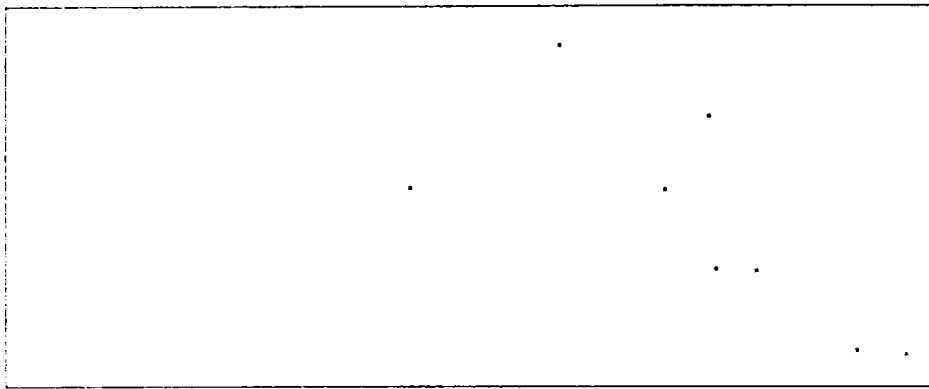


Figura 3.23 – Imaginea  $f$

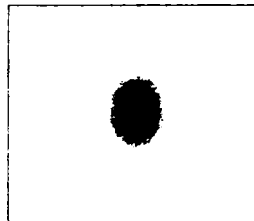


Figura 3.24 – Imaginea  $g$



Figura 3.25 – Imaginea  $h = fog$

Algoritmii genetici sunt algoritmi iterativi de căutare aleatoare pentru probleme neliniare, bazați pe mecanismele selecției naturale și geneticii naturale [Chen1997, Liang1997]. Ei folosesc reguli de tranziție probabiliste pentru găsirea soluției optime. Algoritmul general de căutare pe baze genetice este prezentat în figura 3.26 [Chen1997].

Căutarea și selecția soluțiilor se face pe baza unor reprezentări interne ale funcțiilor. Fiecare soluție intermediară este reprezentată și codificată prin două matrici (figura 3.27):

- $A[m \times n]$  ce codifică funcția  $f$ , unde  $m$  este numărul de circuite de pe o linie de pe PCB iar  $n$  este numărul de linii cu circuite de pe PCB. Pentru fiecare circuit de pe placă, este rezervat un element în matrice a cărei valoare reală reprezintă nivelul de gri al punctului corespunzător circuitului în imaginea  $f$ ;
- $G[2 \times 1]$  codifică funcția  $g$ , adică parametrii  $w$  și  $h$  din funcția Gauss.

În prima etapă a algoritmului se generează aleator populațiile inițiale pentru  $f$  și  $g$ . După care se intră într-un ciclu în care se calculează convoluția între toate perechile  $f_i$  și  $g_j$  din populațiile  $f$  și  $g$ . Funcția de fitness pentru o pereche cromozomială se calculează după relația:

$$E_{i,j} = |g - f_i \circ g_j|$$



Soluția optimă se atinge minimizând costurile pentru  $f_i$  și  $g_j$ , date de:

$$E_{f_i} = \min_{j=1,p} (E_{i,j})$$

$$E_{g_j} = \min_{i=1,p} (E_{i,j})$$

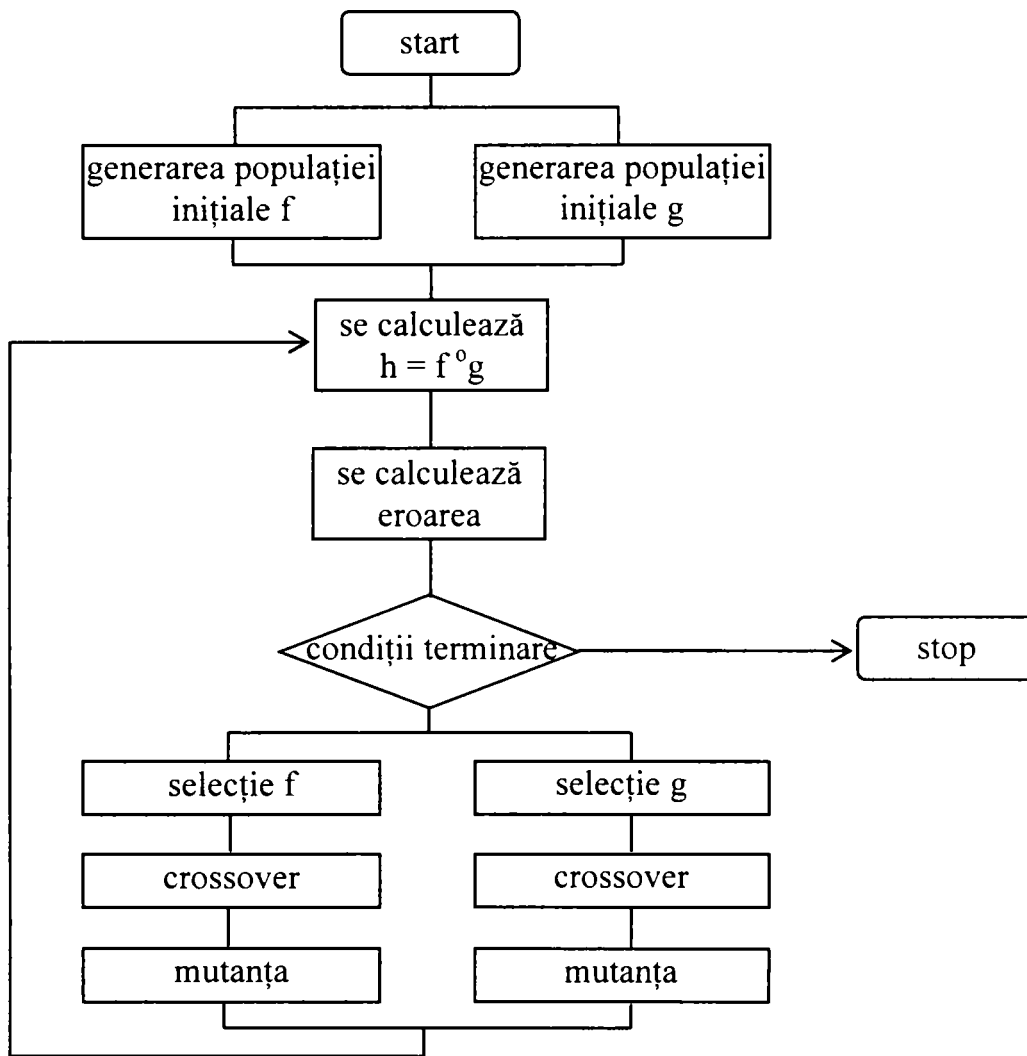


Figura 3.26 – Algoritmul genetic de deconvoluție

$$A = \begin{bmatrix} 34. & 11 & 56. \\ 0 & 47. & 35. \end{bmatrix} \quad G = \begin{bmatrix} 12 \\ 15 \end{bmatrix}$$

Figura 3.27 – Exemplu de codificare cromozomială

În procesul de selecție cei mai buni cromozomi sunt aleși aleator pentru noua generație a populației. Selecția se face în funcție de funcția de cost, cromozomii ce au o funcție cost mică vor fi aleși de mai multe ori, iar cei ce au o funcție cost mare nu vor mai fi selecționați. Cromozomii selecționați se combină pe baza regulilor de crossover prezentate în figura 3.28:

- interschimbarea a două linii (a);
- interschimbarea a două coloane (b);
- interschimbarea cu pivot (c).

Pentru a accelera procesul de căutare s-a modificat algoritmul genetic original cu următoarele elemente:

- în faza de crossover dintr-un algoritm genetic se face interschimbarea unor părți din cromozomii părinților. Aici s-a introdus o funcție ce poate efectua aleator următoarele operații asupra cromozomilor părinților: copiere, medie, minim și maxim;
- în faza de selecție s-a implementat și un mecanism adaptiv de căutare ce se aplică aleator cromozomilor cu costul cel mai mic.

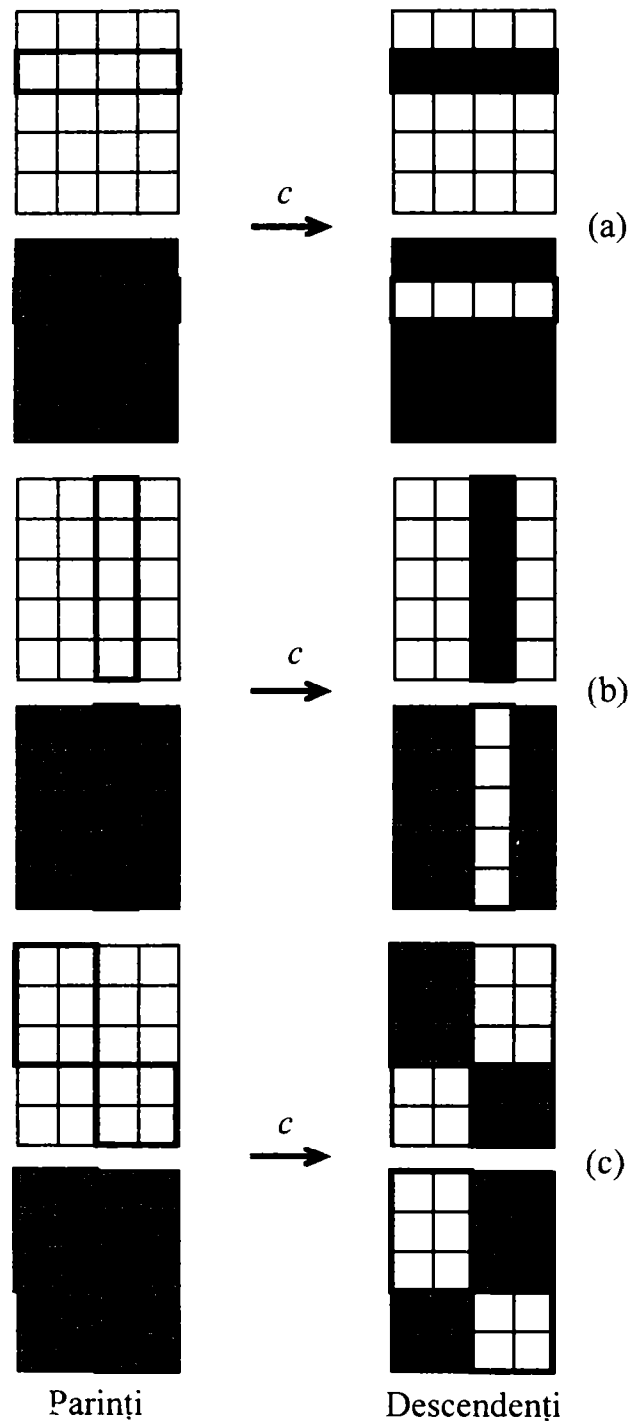


Figura 3.28 – Reguli de crossover

Pentru verificarea acestui algoritm s-au generat mai multe imagini termice ideale de diferite dimensiuni și număr diferit de circuite. Algoritmul de deconvoluție a fost testat pe:

- imaginile ideale obținute prin convoluție;
- imaginile ideale pe care s-au aplicat diferite niveluri de zgomote;
- imaginile ideale pe care s-au aplicat diferite filtre de aplatizare;
- imaginile ideale pe care s-au aplicat și zgomote și filtre;
- imaginile reale unor plăci cu circuite integrate.

## Test 1 – deconvolutia imaginilor ideale originale

În prima fază, algoritmul propus a fost încercat pe imaginea convoluționată originală, pentru un număr de 3 circuite.



Figura 3.29 – Imagine de test

Imaginea a fost creată cu următorii parametri pentru distribuția gauss-iană:  $a=1$ ,  $w=12$ ,  $h=15$ . În acest algoritm se poate face uz și de anumite informații a priori sau se pot lăsa toți parametri să varieze. Cu cât mai multe informații a priori se folosesc cu atât algoritmul este mai rapid. În cazul testelor s-au folosit ca informații a priori pozițiile și numărul gauss-ienelor, lăsând să varieze înălțimile spike-urilor ( $A$ ) și parametrii  $w$  și  $h$  ai gauss-inelor. În cazul în care se cunosc  $w$  și  $h$  sau se știe că aceștia rămân neschimbați pentru toate gauss-ienele timpul de găsimare a soluției scade semnificativ.

Figurile 3.30 și 3.31 prezintă valorile soluției obținute după 10 generații (cu eroare de 1.19%) folosind algoritmul genetic standard, respectiv 5 generații (cu o eroare de 0.11%) pentru cazul în care s-a utilizat algoritmul genetic modificat propus. Timpul de calcul în primul caz a fost de 11 min iar în al doilea caz de 3 min.

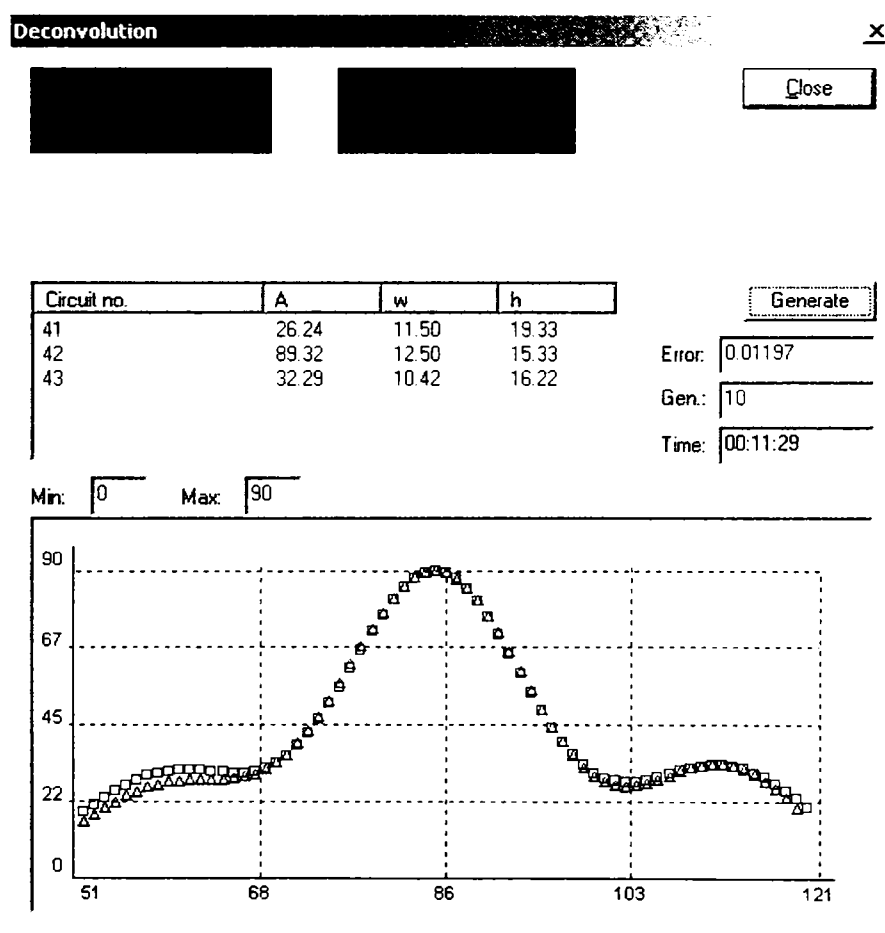


Figura 3.30 – Deconvoluția imaginii de test (după 10 generații)

Număr circuit	A original	A	w	h	Eroare
41	30	26.24	11.50	19.33	12.53%
42	89	89.32	12.50	15.33	0.36%
43	32	32.29	10.42	16.22	0.90%

Tabelul 3.1

Date statistice pentru testul din figura 3.30:

Număr de generații: 10

Eroare procentuală totală: 1.19%

Timp de execuție: 11:29 min.

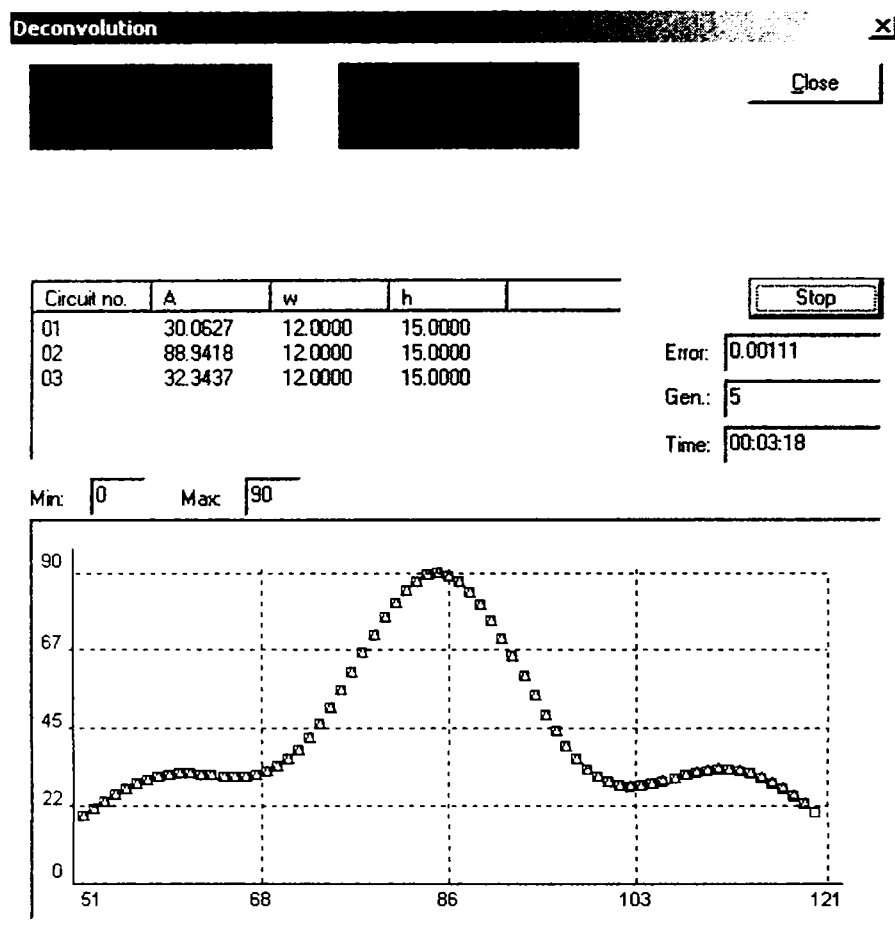


Figura 3.31 – Deconvoluția imaginii de test (după 5 generații)

Număr circuit	A original	A	w	h	Eroare
1	30	30.06	12.00	15.00	0.20%
2	89	88.94	12.00	15.00	0.06%
3	32	32.34	12.00	15.00	1.06%

Tabelul 3.2

Date statistice pentru testul din figura 3.31:

Număr de generații: 5

Eroare procentuală totală: 0.11%

Timp de execuție: 3:18 min.

Rezultatul deconvoluției imaginii de test după 20 de generații este prezentat în figura 3.32 și tabelul 3.3.

Număr circuit	A original	A	w	h	Eroare
1	30	30.00	12.00	15.00	0.00%
2	89	89.00	12.00	15.00	0.00%
3	32	32.00	12.00	15.00	0.00%

Tabelul 3.3

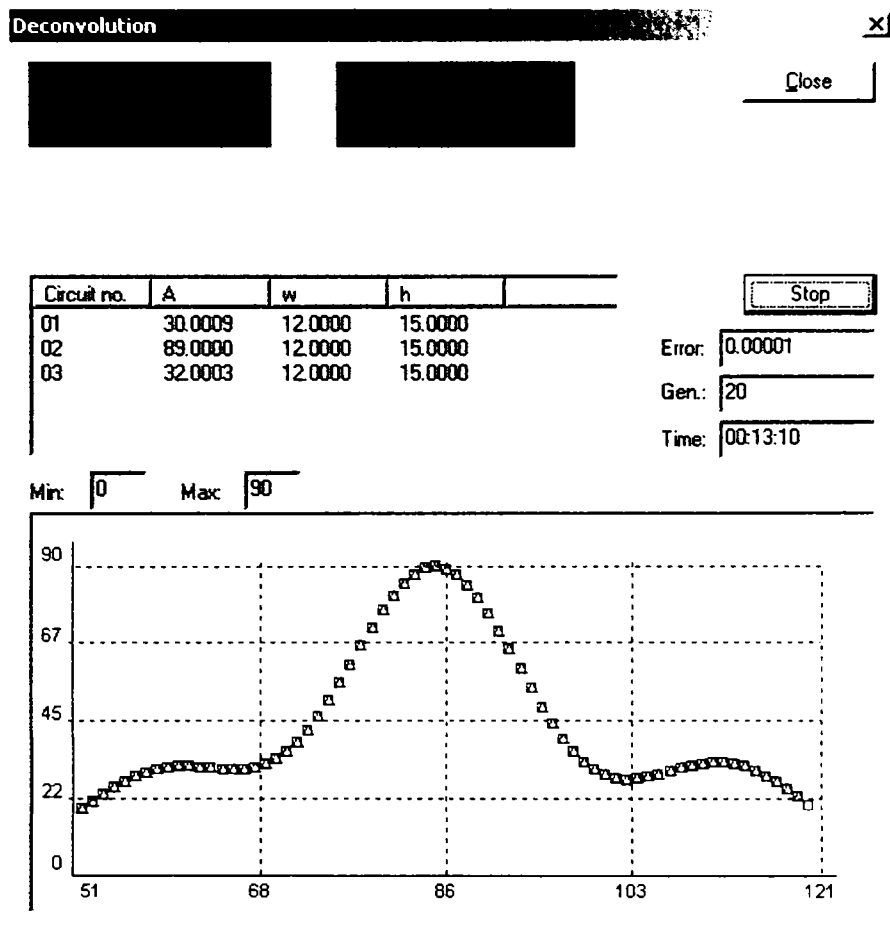


Figura 3.32 – Deconvoluția imaginii de test (după 20 de generații)

Date statistice pentru testul din figura 3.32:

Număr de generații: 5

Eroare procentuală totală: 0.19%

Timp de execuție: 3:18 min.

3.4. Evoluția algoritmului de deconvoluție în cele 20 de generații este prezentată în tabelul

Gen.	A	Eroare [%]	Timp [s]
1	30.97, 88.94, 32.34	3.7490	40
2	30.97, 88.94, 32.34	1.4591	80
3	31.07, 89.84, 32.56	1.3573	119
4	30.19, 88.94, 32.34	0.6317	159
5	30.19, 88.94, 32.34	0.1432	198
6	29.90, 88.94, 32.34	0.1243	238
7	30.18, 89.02, 32.09	0.0751	267
8	30.18, 89.02, 32.09	0.0621	307
9	30.13, 89.10, 31.98	0.0550	356
10	30.00, 88.94, 31.96	0.0241	397
20	30.00, 89.00, 32.00	0.0010	795

Tabelul 3.4

Eroarea pe parcursul celor 20 de generații este prezentată în graficul din figura 3.33. Se observă că eroarea scade foarte repede după doar câteva generații.

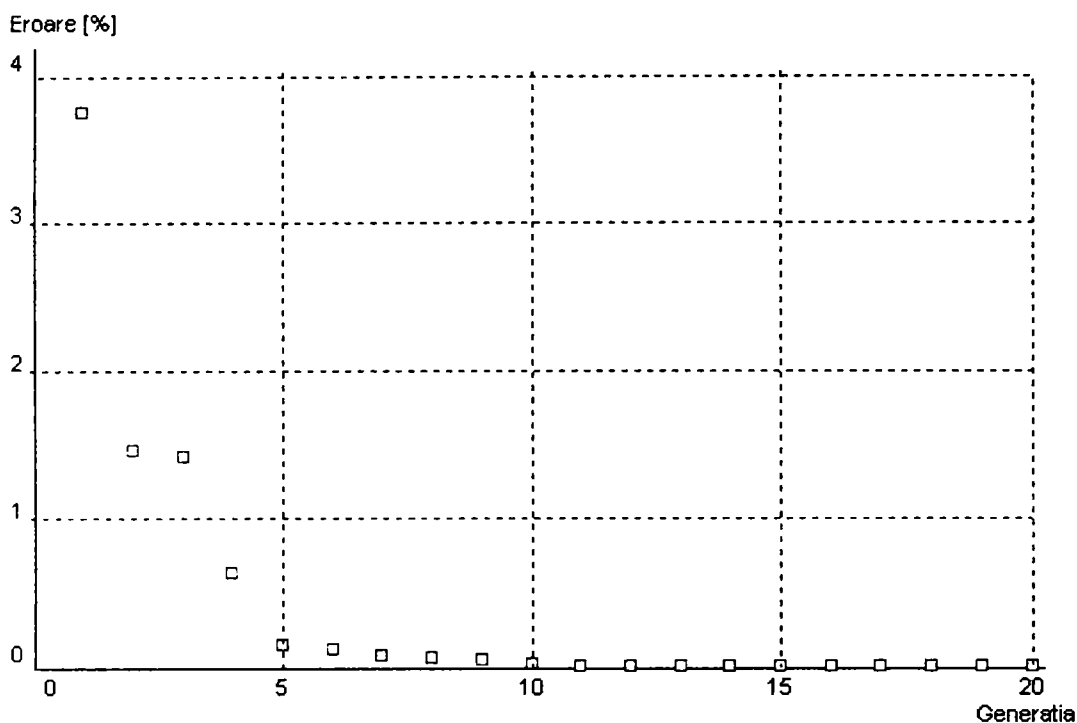


Figura 3.33 – Evoluția erorii procesului de deconvoluție

### Test 2 – Deconvoluția imaginilor ideale cu zgomote

În cel de-al doilea set de teste s-a încercat punerea în evidență a modului în care zgomotele influențează algoritmul de deconvoluție propus. Pentru aceasta s-a plecat tot de la o imagine ideală obținută prin convoluție pe care au fost aplicate diferite niveluri de zgomote liniare uniforme.

Imagine cu zgomote	Eroare imagine [%]	Eroare A [%]	Eroare G [%]	Generatii	Timp [min]
5%	1.08	0.11	0.40	12	8:01
10%	2.05	0.15	0.35	13	8:36
15%	2.96	0.76	1.45	16	10:35
20%	3.71	0.62	1.03	20	13:14
25%	4.64	1.41	1.36	20	13:12
30%	5.32	1.32	1.17	20	13:13
40%	6.25	0.82	0.68	20	13:29
50%	7.09	1.84	1.01	20	13:18

Tabelul 3.5 – Influența zgomotelor asupra algoritmului

În figura 3.34 este prezentat un exemplu de deconvoluție a unei imagini cu zgomote. După cum se poate observa din tabelul 3.5 algoritmul de deconvoluție se comportă foarte bine în prezența zgomotelor. Eroarea depinde foarte puțin de nivelul zgomotelor, la un raport mare între zgomote și imagine, eroarea crește doar puțin, în jurul lui 1%. Eroarea A și eroarea G au ca valori de referință parametrii funcțiilor  $f$  și  $g$  ai imaginii originale, fără zgomote.

### Test 3 – Deconvoluția imaginilor ideale filtrate

În cel de-al treilea set de teste s-a încercat obținerea modului în care filtrele de aplatizare influențează algoritmul de deconvoluție propus. Pentru aceasta s-a plecat tot de la o imagine ideală obținută prin convoluție pe care au fost aplicate diferite tipuri de filtre: filtru median, filtru gaussian, filtru de accentuare a detaliilor (enhance), ș.a. Rezultatele sunt prezentate în tabelul 3.6.

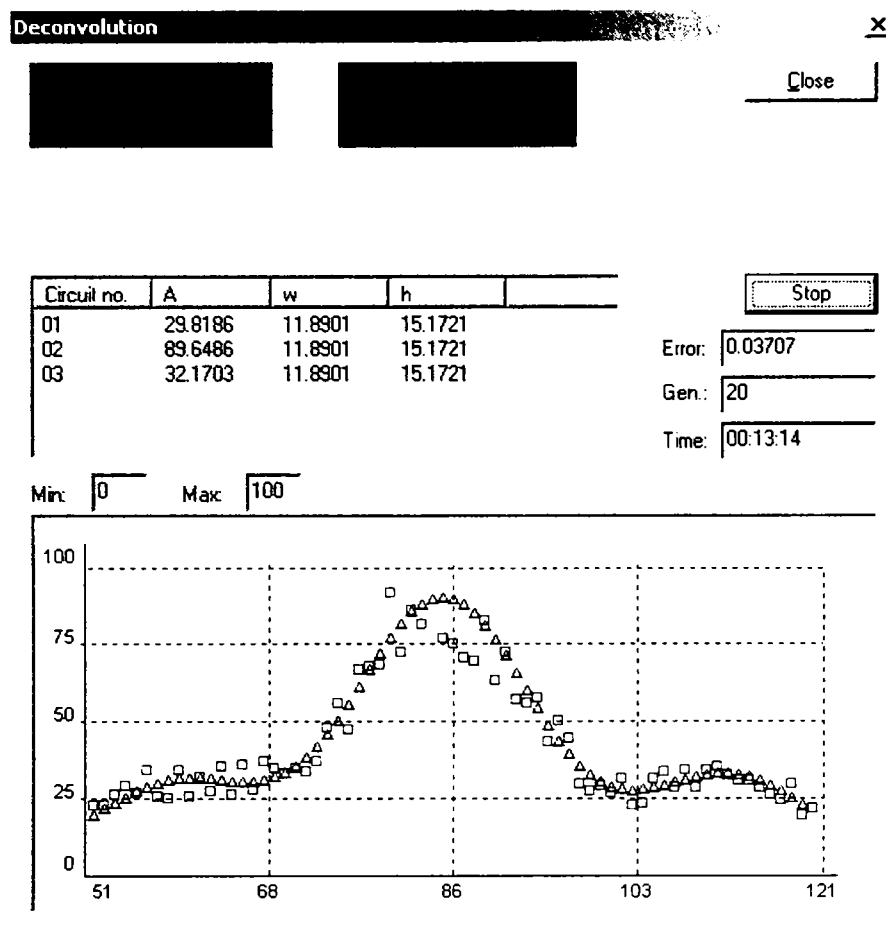


Figura 3.34 – Deconvoluția unei imagini cu zgomote

Filtru	Eroare imagine [%]	Eroare A [%]	Eroare G [%]	Generatii	Timp [min]
Enhance	0.00	2.27	1.62	20	13:11
Enhance (*2)	0.03	4.67	2.29	20	13:36
Enhance (*4)	0.02	9.79	4.57	20	13:21
Gauss	0.00	0.57	0.27	20	13:09
Gauss (*2)	0.00	1.11	0.57	20	13:18
Gauss (*4)	0.00	2.21	1.12	20	13:13
Median	0.05	0.01	0.01	20	13:15
Median (*2)	0.06	0.01	0.00	20	13:35
Median (*4)	0.07	0.03	0.02	20	13:11
Enhance+Gauss (*2)	0.04	1.08	0.55	20	13:27
Enhance+Gauss (*4)	0.00	0.05	0.03	20	13:18
Gauss (*2)+Enhance	0.00	1.08	0.53	20	13:11
Gauss (*4)+Enhance	0.01	0.04	0.02	20	13:20

Tabelul 3.6 – Influența filtrelor asupra algoritmului

Filtrul de evidențiere a detaliilor (enhance) influențează în mod direct eroarea cu care este obținută imaginea originală (eroare A și eroare G). La aplicarea unui astfel de filtru, valorile matricei A cresc în timp ce valorile matricei G se micșorează, în funcție de gradul de acoperire al filtrului. Filtrul gaussian aplatizează imaginea și are un efect invers asupra fidelității algoritmului relativ la filtrul enhance. Și în acest caz eroarea este direct proporțională cu gradul de acoperire a filtrului. Filtrul median este un filtru pentru eliminarea zgomotelor și are un comportament foarte bun pentru algoritmul de deconvoluție.

Din tabelul 3.6 se observă că algoritmul de deconvoluție este influențat direct proporțional atât de filtrele enhance cât și de cele gauss. Efectul acestor două filtre asupra algoritmului de deconvoluție este însă compensatoriu, în sensul că dacă distorsiunile sunt de tip gauss atunci înainte de aplicarea deconvoluției este recomandat a se aplica un filtru enhance și invers. dacă distorsiunea introdusă de procesul de achiziție este de tip enhance se recomandă a se utiliza un filtru gauss înainte de deconvoluție. Pe de altă parte deconvoluția nu este influențată de filtrul median astfel că acesta nu necesită nici un fel de corecție.

#### Test 4 – Deconvoluția imaginilor ideale distorsionate și cu zgomote

Cel de-al patrulea test a încercat aplicarea deconvoluției pe o imagine distorsionată și cu zgomote. Rezultatul deconvoluției imaginii de test după 20 de generații este prezentat în figura 3.35 și tabelul 3.7.

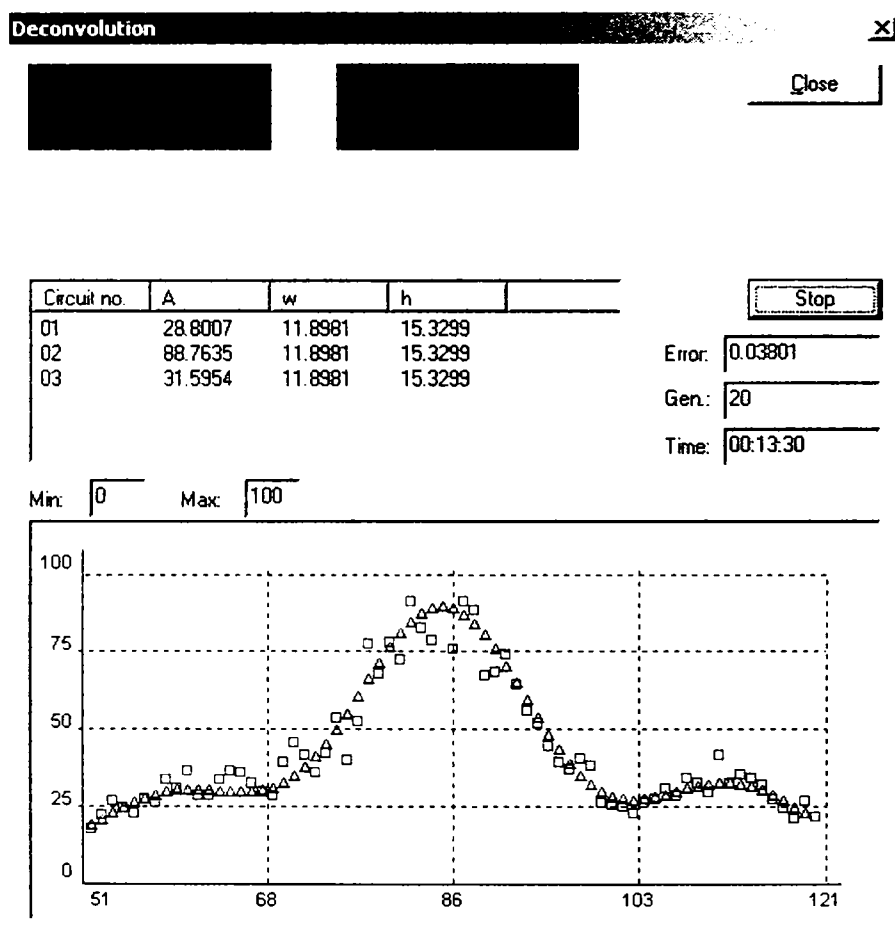


Figura 3.35 – Deconvoluția unei imagini distorsionate și cu zgomote

Număr circuit	A	w	h	Eroare A	Eroare G
1	28.80	11.90	15.33	4.00%	1.52%
2	88.76	11.90	15.33	0.27%	1.52%
3	31.60	11.90	15.33	1.25%	1.52%

Tabelul 3.7

După cum se poate observa în figura 3.35 și tabelul 3.7 deconvoluția dă rezultate bune și pentru imaginile distorsionate (blur) și cu zgomote.



### Test 5 – Deconvoluția imaginilor ideale în funcție de dimensiunea imaginii

Dimensiunea imaginii este un alt parametru care influențează în mod direct performanța algoritmului de deconvoluție. Pentru a ilustra această dependență s-a aplicat algoritmul pe mai multe imagini de dimensiuni diferite ce conțineau fiecare câte trei circuite plasate în poziții diferite.

Dimensiune imagine	Eroare imagine [%]	Eroare A [%]	Eroare G [%]	Generatii	Timp [min]
136x48	0.00	0.00	0.00	20	13:11
144x60	0.00	0.01	0.00	20	17:45
162x48	0.00	0.00	0.00	20	15:58
170x60	0.00	0.00	0.00	20	20:30
136x130	0.01	0.01	0.00	20	36:14
144x142	0.00	0.00	0.00	20	41:01

Tabelul 3.8 – Influența dimensiunii imaginii asupra performanței algoritmului

În graficul din figura 3.36 este prezentată relația între durata algoritmului și dimensiunea imaginii exprimată în pixeli.

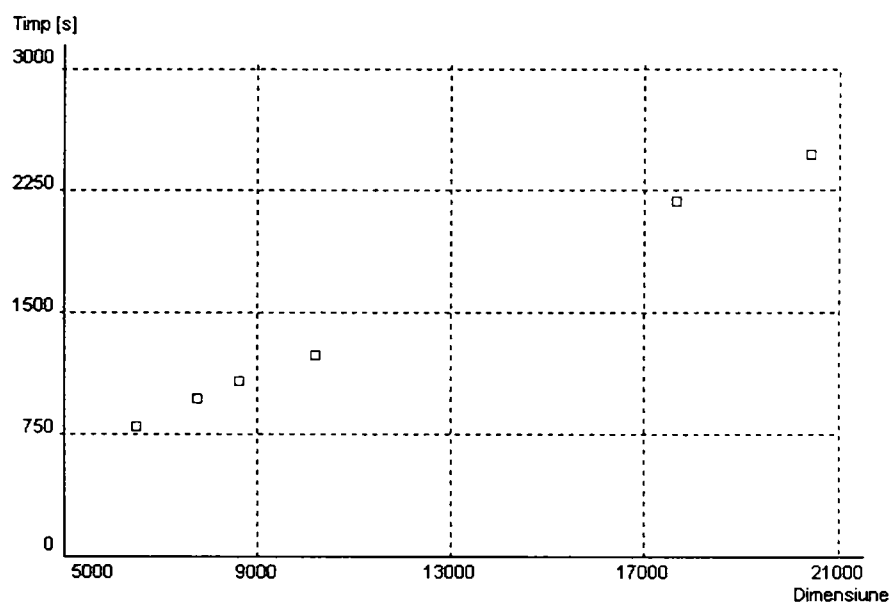


Figura 3.36 - Influența dimensiunii imaginii asupra performanței algoritmului

### Test 6 – Deconvoluția imaginilor ideale în funcție de numărul de circuite

În cadrul acestui set de test s-a încercat evidențierea modului în care performanța algoritmului este influențată de numărul de circuite de pe placă. Pentru aceasta s-au considerat mai multe imagini de aceeași dimensiune în care se găsesc diferite circuite integrate.

Număr circuite	Eroare imagine [%]	Eroare A [%]	Eroare G [%]	Generatii	Timp [min]
1	0.00	0.00	0.00	20	23:16
2	0.00	0.00	0.00	20	29:18
3	0.01	0.01	0.01	20	35:31
4	0.03	0.03	0.03	20	43:01
5	0.04	0.08	0.02	20	49:55
6	0.07	0.28	0.07	20	58:15

Tabelul 3.9 – Influența dimensiunii imaginii asupra performanței algoritmului

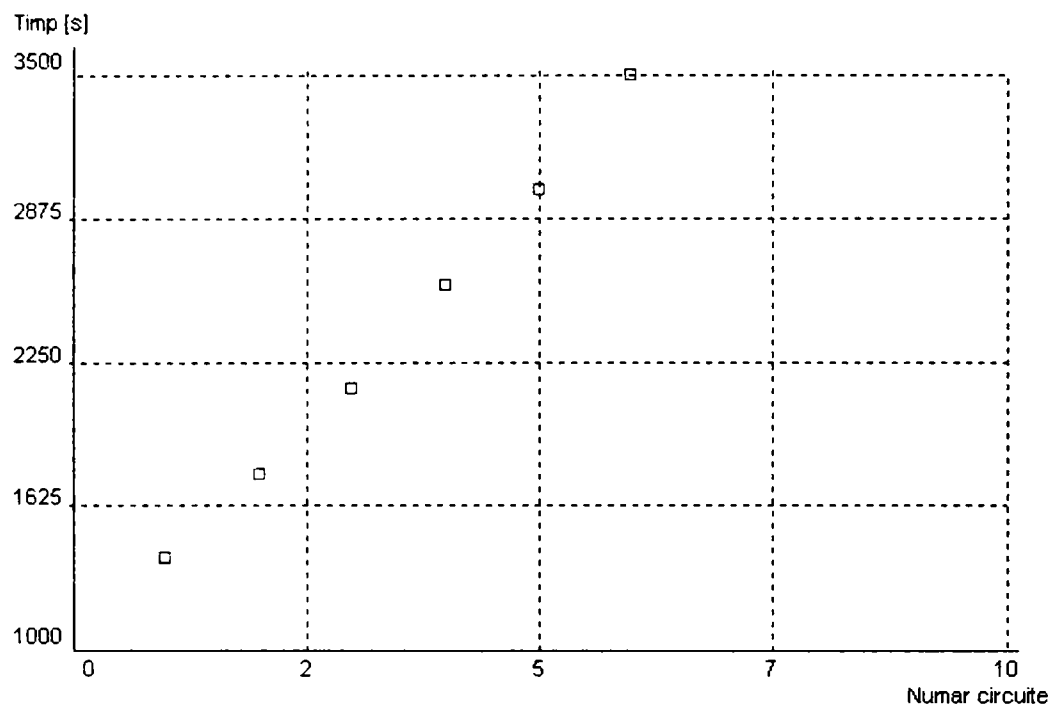


Figura 3.37 – Influența numărului de circuite asupra performanței deconvoluției

### Test 7 – Deconvoluția imaginilor termice reale

În continuare algoritmul de deconvoluție a fost aplicat pe un grup de 3 circuite din imaginea termică a plăcii cu circuite integrate. Rezultatele sunt prezentate în figura 3.38.

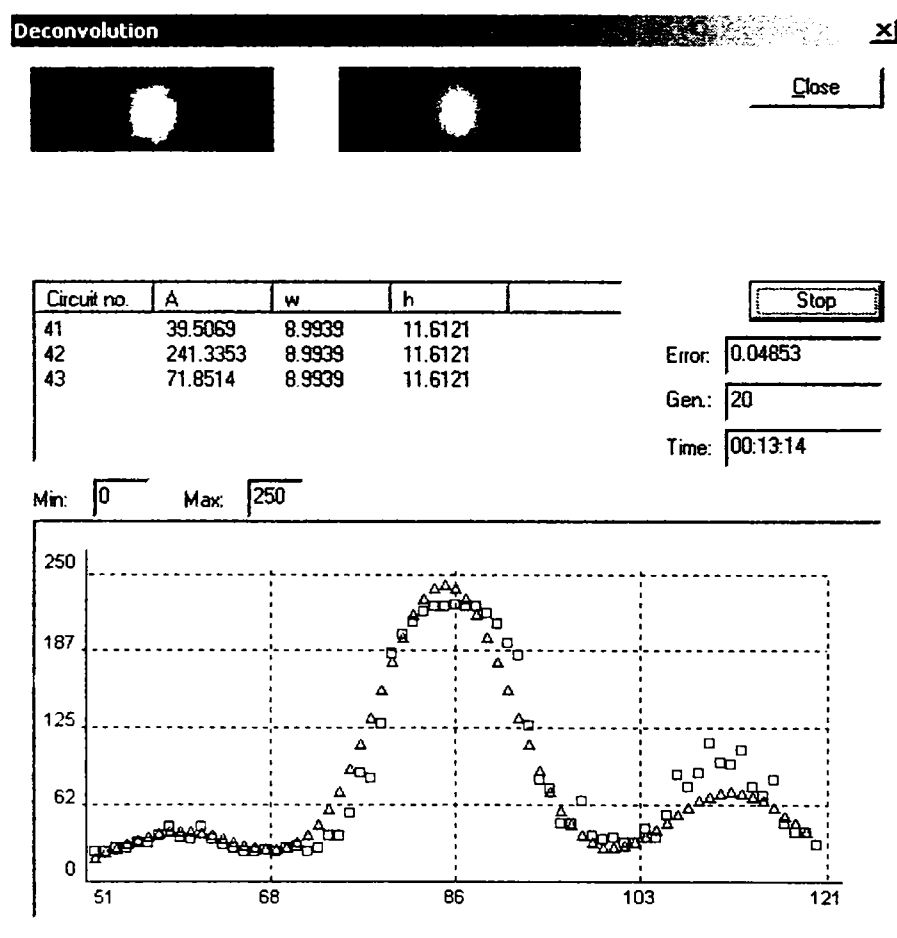


Figura 3.38 – Deconvoluția unei imagini termice reale

Aplicând algoritmul pentru întreaga placă (figura 3.39) s-a obținut imaginea ideală din figura 3.40. Relația între valorile funcției  $f$  (matricea  $A$ ) și curentul absorbit de circuitul corespunzător este prezentată în figura 3.40.

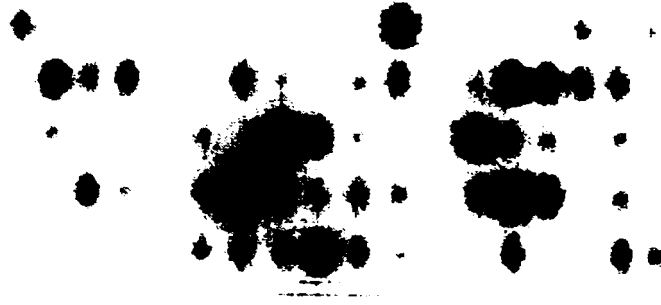


Figura 3.39 – Imaginea reală a plăcii (în invers video)

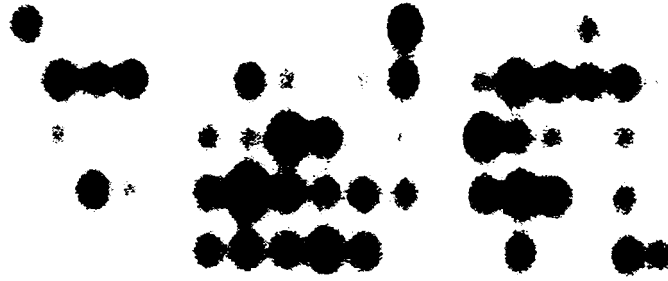


Figura 3.40 – Imaginea deconvoluționată (în invers video)

Algoritmul de deconvoluție propus poate fi utilizat în faza de analiză a imaginilor termice pentru estimarea puterii dissipate de fiecare circuit de pe placă. Deconvoluția folosește o metodă evolutivă bazată pe algoritmi genetici, care se aplică pe o imagine termică și generează o listă cu consumul estimativ al fiecărui circuit de pe placă. Relația între funcția  $f$  obținută prin deconvoluție și puterea disipată sau curentul consumat de circuit este prezentată în figura 3.41. În această figură este prezentată amplitudinea obținută prin deconvoluție pentru fiecare circuit și curentul absorbit de circuit.

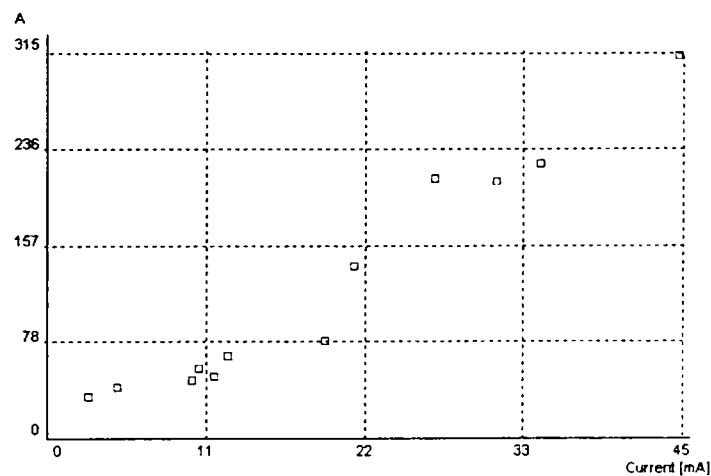


Figura 3.41 – Amplitudinea funcție de curent

Eroarea medie: 5.86 %  
 Eroarea maximă: 14.73 %  
 Eroarea pătratică medie: 23.28

### 3.2.2 Deconvoluția imaginilor termice cu surse de căldură neuniforme

Algoritmul de deconvoluție prezentat se poate folosi doar pentru surse de căldură uniforme și de aceeași dimensiune. În mod normal pe o placă cu circuite integrate există mai multe tipuri de circuite, de dimensiuni diferite și funcții de transfer termic diferite. Pentru acest caz s-a propus modificarea algoritmului de deconvoluție standard și înlocuirea lui cu cel prezentat în figura 3.42.

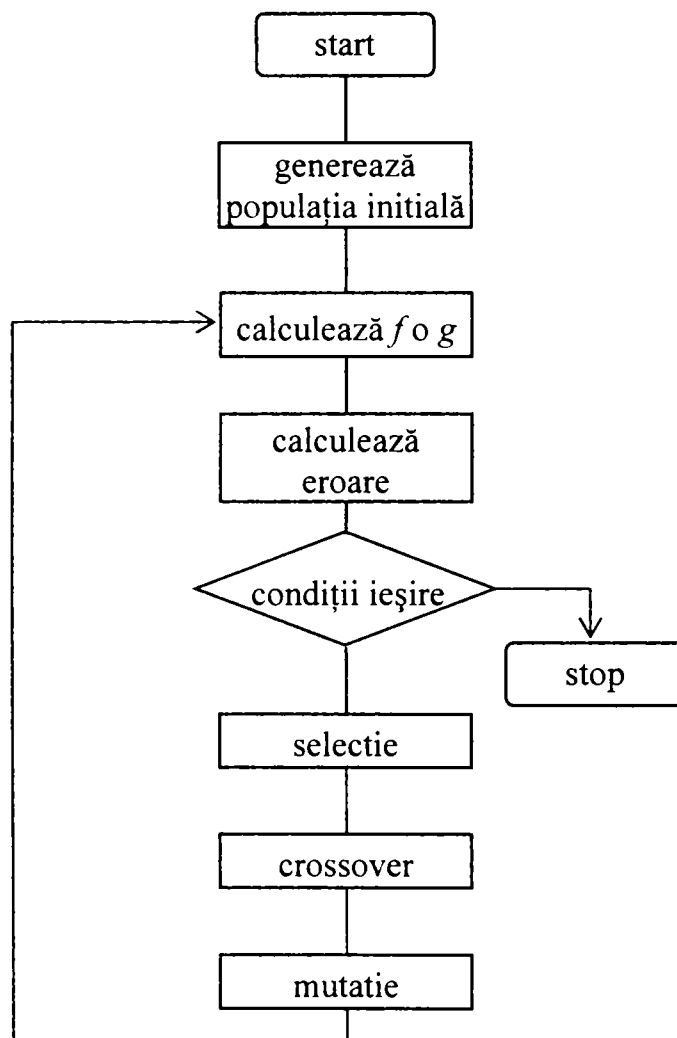


Figura 3.42 – Algoritmul genetic modificat

În acest algoritm informația este codificată pentru fiecare circuit în parte și datele pentru fiecare circuit evoluează separat. Informația cromozomială este o matrice  $H[n \times 3]$ , unde  $n$  este numărul de circuite. Fiecare linie a acestei matrici codifică un circuit cu următorii parametri:  $A$  – amplitudinea sursei de căldură,  $h$  și  $w$  – funcția gauss ce estimează disiparea căldurii.

Acest algoritm față de cel precedent converge mai lent pe un număr mai mare de generații însă pentru evoluția unei generații este necesar un timp mai mic decât la algoritmul precedent.

Date statistice pentru testul din figura 3.43:

Număr de generații: 126

Eroare procentuală totală: 0.04%

Timp de execuție: 16:26 min.

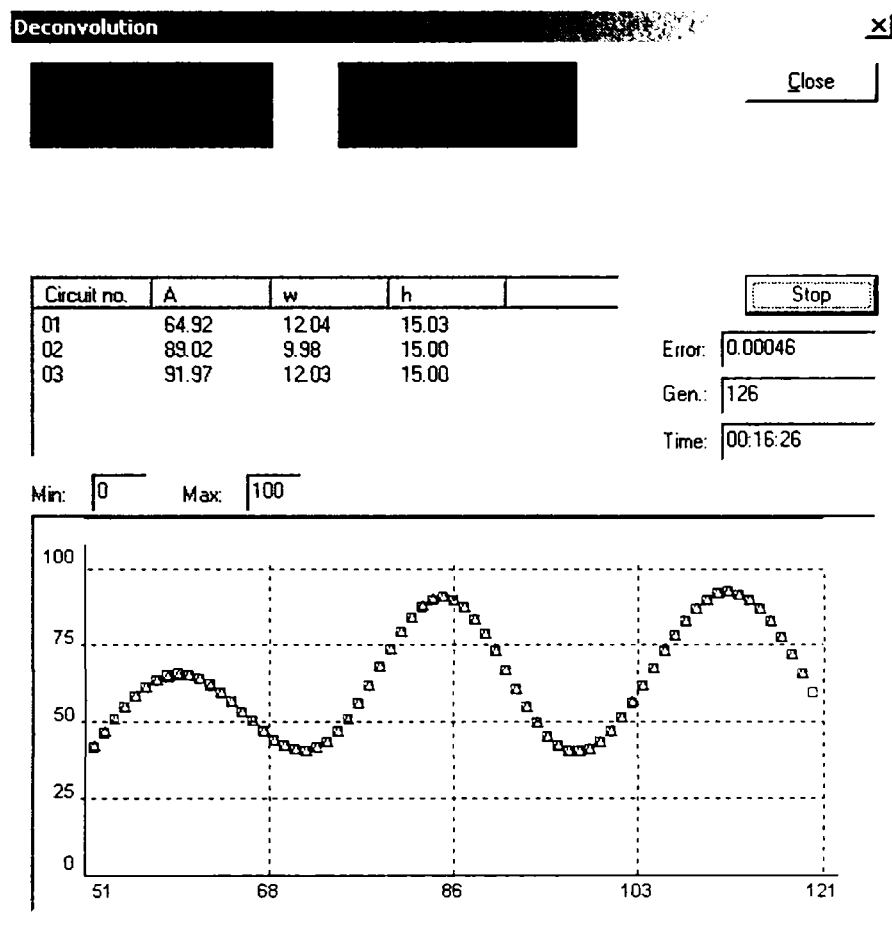


Figura 3.43

Imagina originală a fost creată prin convoluția a trei puncte de amplitudini: 65, 89 și 92 și fiecare având o funcție gauss distinctă: (12,15), (10,15) și (12,15).

Număr circuit	A	w	h	Eroare A	Eroare G
1	64.92	12.04	15.03	0.12%	0.27%
2	89.02	9.98	15.00	0.02%	0.10%
3	91.97	12.03	15.00	0.03%	0.12%

Tabelul 3.10

Algoritmul de deconvoluție propus poate fi utilizat și pentru alte modele de distribuție a temperaturilor pe placă cum ar fi seriile Fourier sau funcțiile Bessel. Ceea ce trebuie pentru modificarea algoritmului pentru un alt model este găsirea unei codificări cromozomiale corespunzătoare.

### 3.3 Inspecția vizuală a plăcilor cu circuite integrate

O etapă importantă în cadrul producției de masă industriale o constituie testarea plăcilor cu circuite integrate, etapă ce are ca scop asigurarea unei calități de 100% a produsului final (circuite, cablaj, lipituri, etc.). Cea mai dificilă sarcină din cadrul testării este inspecția vizuală a produselor. Inspecția vizuală încearcă să identifice atât defectele funcționale cât și defectele cosmetice. Chiar dacă domeniile procesărilor de imagini, recunoașterii tiparelor și inteligenței artificiale au evoluat suficient cât să poată automatiza procesul de inspecție vizuală, încă mai se mai folosesc operatori umani, a căror performanță este în general neadecvată și variabilă. Însă prin scăderea prețului echipamentelor de testare automată [Moganti1996, Tatibana1997]

Procedurile de detecție a defectelor la plăcile cu circuite integrate se împart în două mari clase [Goodall1991]:

- metode electrice (cu contact direct) – presupun măsurarea unor parametrii electrice în anumite puncte de control de pe placă;
- metode non-electrice (fără contact direct) – folosesc imagini dintr-un anumit domeniu de frecvențe pentru estimarea calității plăcilor cu circuite integrate.

Metodele de inspecție fără contact direct a plăcilor cu circuite integrate cele mai întâlnite sunt următoarele [Moganti1996, Jacob1992, Bond1991]:

- inspecția optică automată – detectează aceleași defecte ce sunt vizate de un operator uman: aspectul traseelor de cablaj și a găurilor, scurt-circuite ale punctelor de lipire, lipsa lipiturii, lipsa sau plasarea incorectă a unor componente, ș.a.
- inspecția cu raze X – se folosește pentru testarea rapidă și precisă a plăcilor multistrat. Această metodă poate detecta lipsa de material, alinierea greșită a componentelor, punți de material, lipitură insuficientă sau în excess, lipitură rece, pini îndoșiți, ș.a.
- laminografie – folosește tot razele X pentru a obține secțiuni transversale prin cablaj, astfel că se pot obține imagini separate pentru fiecare strat din placă.
- inspecția ultrasonică – este folosită în special pentru testarea lipiturilor. Sistemul de captare ultrasonică generează imagini precise prin scanarea cu un semnal piezoelectric direcționat a lipiturilor.
- inspecția termică – este folosită la verificare prin metode termice a funcționării plăcii cu circuite integrate.

Datorită tendinței crescătoare a puterii disipate de circuitele integrate se folosește tot mai mult faza de inspecție termică a plăcilor cu circuite integrate pentru testarea acestora din punct de vedere termic și poate fi aplicată chiar pentru testarea funcțională. Comparată cu inspecția optică și cu cea cu raze X, inspecția termică este mai puțin automatizabilă [Bond1991]. În cele ce urmează se va prezenta o soluție de automatizare a fazei de testare termică printr-un sistem de clasificare a semnăturilor fiecărui circuit de pe placă.

La acest moment metoda de testare termică folosind imaginile în infraroșu este implementată pe mașini speciale dotate cu una sau mai multe camere video în infraroșu ce scanează placa ce se testează, imaginile obținute sunt mai apoi prelucrate de un anumit program de verificare [Sexton2001]. Aplicația de verificare compară imaginea infraroșu achiziționată cu un model statistic obținut din imaginile termice ale unui set de plăci martor considerate fără defecte. Modelul un anumit nivel de variație, astfel încât micile diferențe de temperatură ce pot apare, să nu influențeze procesul de decizie.

Această soluție uzuală de implementare a testării termice are câteva neajunsuri [Moganti1996, Tatibana1997]:

- cantitatea de date de referință ce trebuie stocate este mare;

- este necesară o aliniere foarte precisă a celor două imagini, lucru ce poate genera probleme serioase în ceea ce privește clasificarea corectă a plăcilor;
- sensibilitate la condițiile de iluminare;
- stabilirea setului de plăci martor pentru construirea modelului statistic al plăcilor fără defecte sau actualizarea modelului pe durata de viață a produsului.

Din acest motiv s-a investigat posibilitatea de extragere a unor caracteristici din imaginea termică pentru fiecare circuit integrat și clasificarea plăcii testate pe baza acestor caracteristici. Noțiunea de semnătură termică va fi rafinată într-unul din capitolele următoare, obținându-se și o compactare a datelor memorate și chiar a timpului de verificare.

### 3.4 Studiul imaginilor termice ale plăcilor cu circuite integrate

Scopul acestei lucrări a fost de a găsi o nouă metodă de testare a plăcilor echipate cu componente electronice bazată pe imaginea termică a plăcii. Ca punct de plecare în acest studiu l-a constituit o placă cu 66 de circuite integrate TTL (deoarece aceste circuite consumă mai mult și se încălzesc mai tare, astfel că imaginea termică are o rezoluție mai bună) și mai multe imagini generate. Pe această placă au fost create artificial șapte defecte de scurt-circuit și un defect de întrerupere. Placa a fost alimentată în condiții statice și la echilibru termic a fost achiziționată imaginea termică a plăcii cu defecte. Din păcate nu a fost luată nici o imagine termică a plăcii inițiale fără defecte, doar, într-un al doilea pas, a fost întreruptă alimentarea pentru circuitele defecte și s-a obținut o imagine termică a plăcii cu defectele eliminate. Figura 3.44 conține imaginea termică a plăcii cu defecte.



Figura 3.44 – Imaginea termică a unei plăci cu circuite integrate cu defecte

Imaginile termice obținute cu acest sistem sunt foarte mult influențate de modul de calibrare al camerei de luat vederi (prin stabilirea nivelului de negru al imaginii, prin contrastul camerei, etc), intervenind subiectivismul operatorului în alegerea celei mai bune imagini. Sistemul de achiziție nefiind conectat la un computer, singura posibilitate de a capta aceste imagini termice a fost utilizarea unui aparat foto pentru a fotografia ecranul sistemului și apoi scanarea acestor fotografii. În momentul de față, la dispoziția cercetătorilor din domeniu se află aparatură cu calități mult îmbunătățite față de cele prezentate anterior, însă imaginile pe care s-a lucrat au avantajul că permit și evaluarea metodelor de procesare a imaginilor termice, iar rezultatele obținute pe aceste imagini vor fi valabile și mai bune pentru imaginile termice mai bune.

### 3.4.1 Metoda de abordare a lucrării

Testele și încercările efectuate asupra imaginilor termice până în acest moment s-au încadrat în următoarele faze (figura 3.45):

- faza de analiză preliminară;
- faza de prelucrări primare;
- faza de testare;
- faza de evaluare a rezultatelor.

Faza de testare propriu-zisă a fost abordată din mai multe perspective pentru găsirea unei metode de testare eficiente:

- analiza imaginilor – folosind doar informații din imaginea termică cum ar fi nivel de gri maxim, mediu, suprafață, contur, etc. și se bazează pe procesarea și analiza imaginilor;
- analiza termică – folosind fenomenele termice la nivelul plăcii cu circuite integrate s-a încercat echivalarea și potrivirea acestora cu imaginea termică;
- analiza caracteristicilor – o nouă metodă de testare pe baza semnăturii termice.

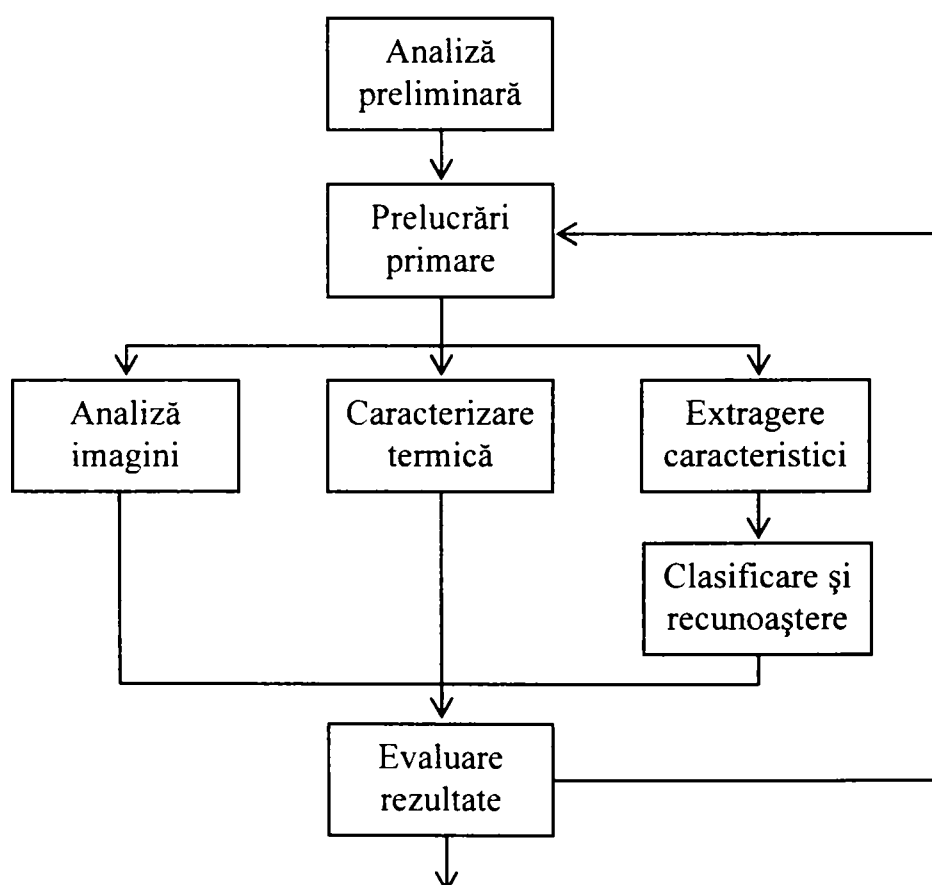


Figura 3.45 – Metoda de lucru

În prima fază s-a efectuat un studiu preliminar asupra plăcii cu circuite integrate și a imaginilor termice obținute. Au fost numerotate cele 66 de circuite conform imaginii din figura 3.46 și au fost alese imaginile cele mai bune ca și contrast, luminozitate și poziționare pentru a putea fi utilizate în fazele următoare.



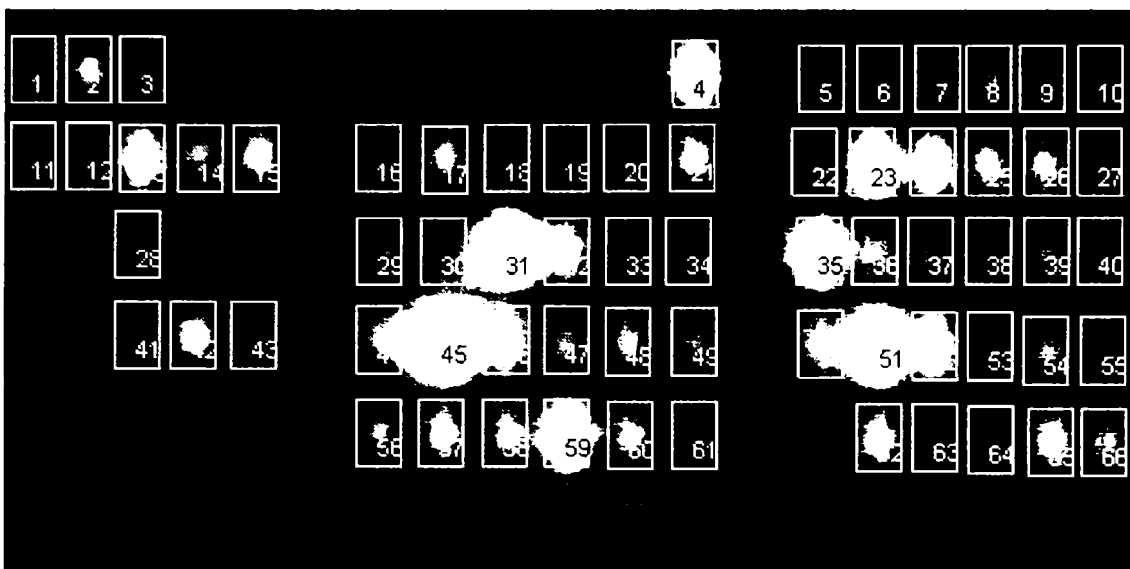


Figura 3.46 – Identificarea circuitelor de pe placă

Datorită mecanismului de achiziție a imaginilor termice menționat, imaginile sunt foarte perturbate, cu zgomote foarte mari. Din acest motiv în cea de-a doua fază au aceste imagini au fost supuse unui număr de procesări primare. Menționez că toate încercările ulterioare prelucrărilor primare au fost făcute atât pe imaginile prelucrate cât și pe cele reale (neprelucrate). Prelucrările primare cele mai utile au fost:

- corecția nivelurilor de gri;
- egalizarea histogramei;
- filtre median, gaussian, evidențierea detaliilor, etc.

Scopul cercetărilor efectuate a fost acela de a găsi o mărime numerică determinabilă din imaginea termică a plăcii cu circuite integrate, pe baza căreia pe de-o parte să se poată evalua corectitudinea circuitelor și pe de altă parte să se poată estima consumul de putere al respectivelor circuite. Această mărime numerică căutată pentru fiecare circuit se numește semnătură termică. Asemănător semnăturii de curent s-a încercat obținerea unei semnături termice ce caracterizează fiecare circuit de pe placă.

### 3.4.2 Analiza imaginilor termice

Primul pas a fost calculul valorilor maxime și medii ale nivelelor de gri pentru fiecare circuit și suprafața pe care este disipată căldura unui circuit. Folosind aceste valori s-a încercat obținerea unei dependențe între nivelul de gri din termogramă și curentul absorbit de fiecare circuit, acesta fiind știut prin măsurare directă.

**Nivelul de gri maxim** este valoarea de gri din imaginea termică din poziția corespunzătoare centrului unui circuit integrat. Pozițiile circuitelor integrate pe placă (și în imagine) sunt considerate cunoscute. Reprezentând grafic acest nivel de gri maxim în funcție de curentul real absorbit de circuit, se obține graficul din figura 3.47. În acest grafic se observă existența unei zone considerate liniare care este reprezentată în figura 3.48 și o zonă de saturație, în care pentru valori mari ale curentului nivelul de gri nu mai variază (datorită numărului limitat de nivele de gri).

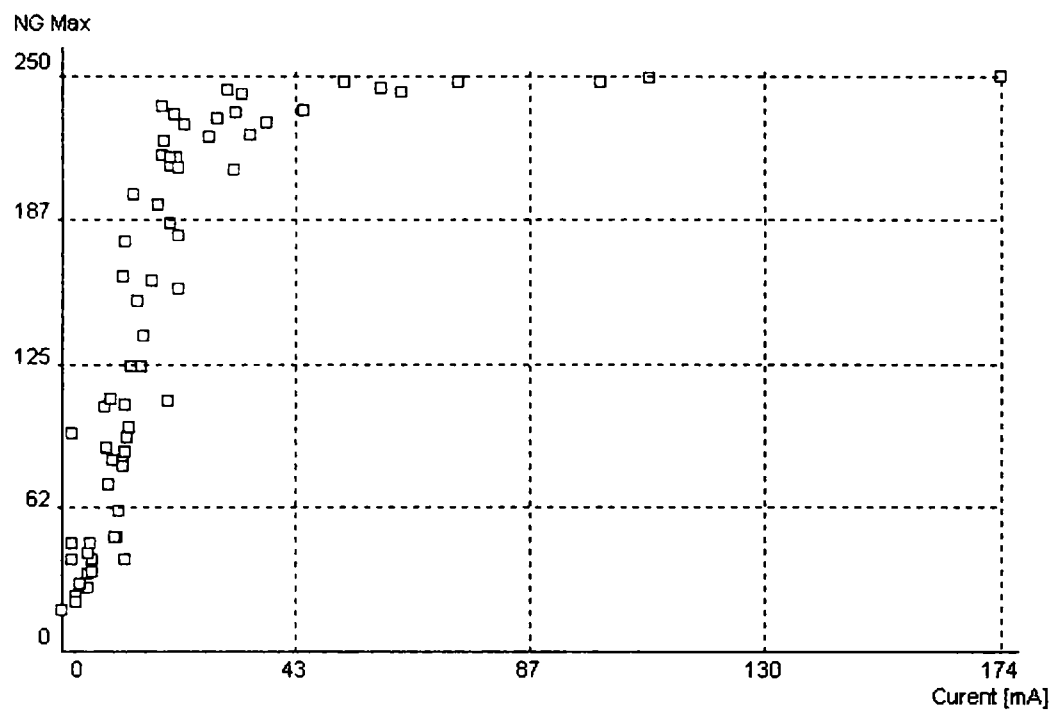


Figura 3.47 – Nivelul de gri maxim

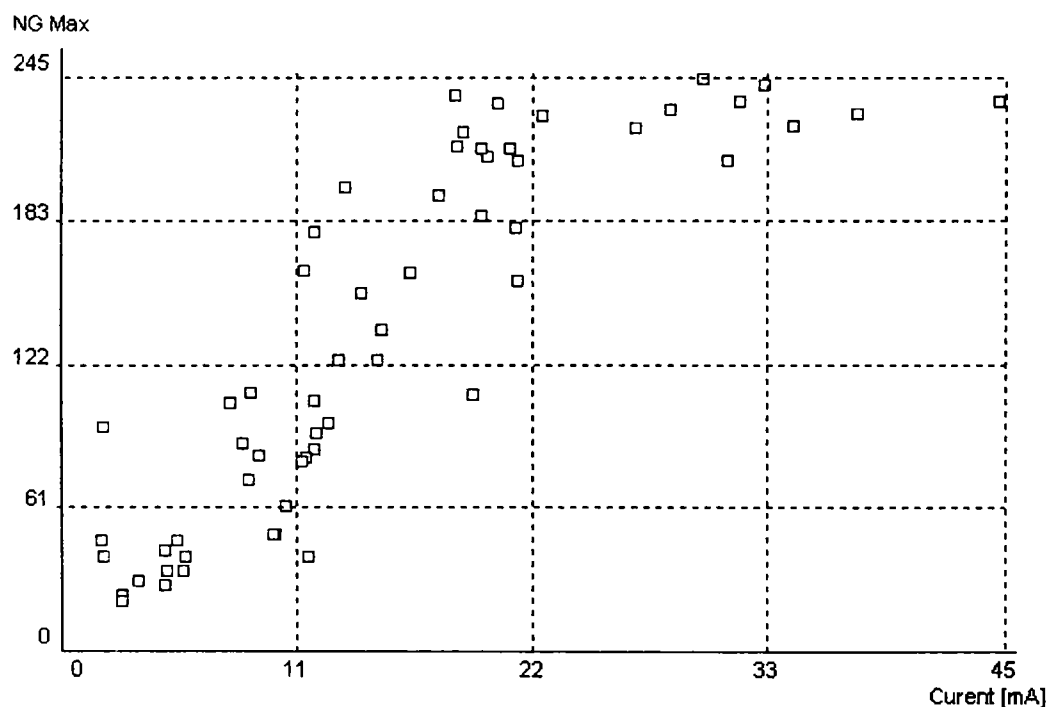


Figura 3.48 – Nivel de gri maxim

Eroarea patrată medie: 37.01  
 Eroarea procentuală medie: 12.82%  
 Eroarea procentuală maximă: 31.60%

**Nivelul de gri mediu** este o valoare medie a nivelului de gri calculată pentru un circuit. Această valoare este media aritmetică a nivelurilor de gri ale pixelilor într-o regiune de dimensiunea unui circuit, regiune (dreptunghi) plasată în imagine peste un circuit. Figura 3.49 conține reprezentarea grafică a nivelului de gri mediu pentru fiecare circuit în funcție de curentul absorbit de respectivul circuit.

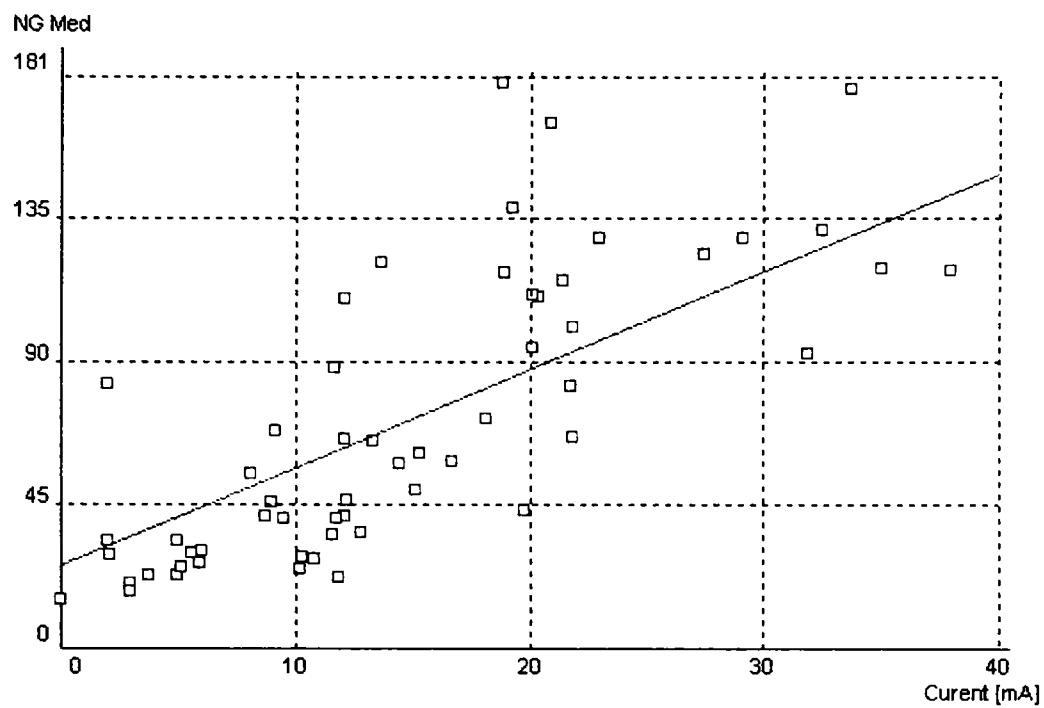


Figura 3.49 – Nivel de gri mediu

Eroarea patratică medie: 29.90

Eroarea procentuală medie: 13.09%

Eroarea procentuală maximă: 52.36%

Se poate observa că doar informația de nivel de gri mediu sau maxim pentru un circuit integrat este insuficientă pentru caracterizarea termică a aceluși circuit. O altă mărime utilizată a fost **nivelul de gri relativ (diferențial)** care este calculat ca și diferența dintre maximum și minimum valorilor de gri pentru fiecare circuit. Reprezentând grafic aceste valori pentru fiecare circuit în funcție de curent s-a obținut graficul din figura 3.50.

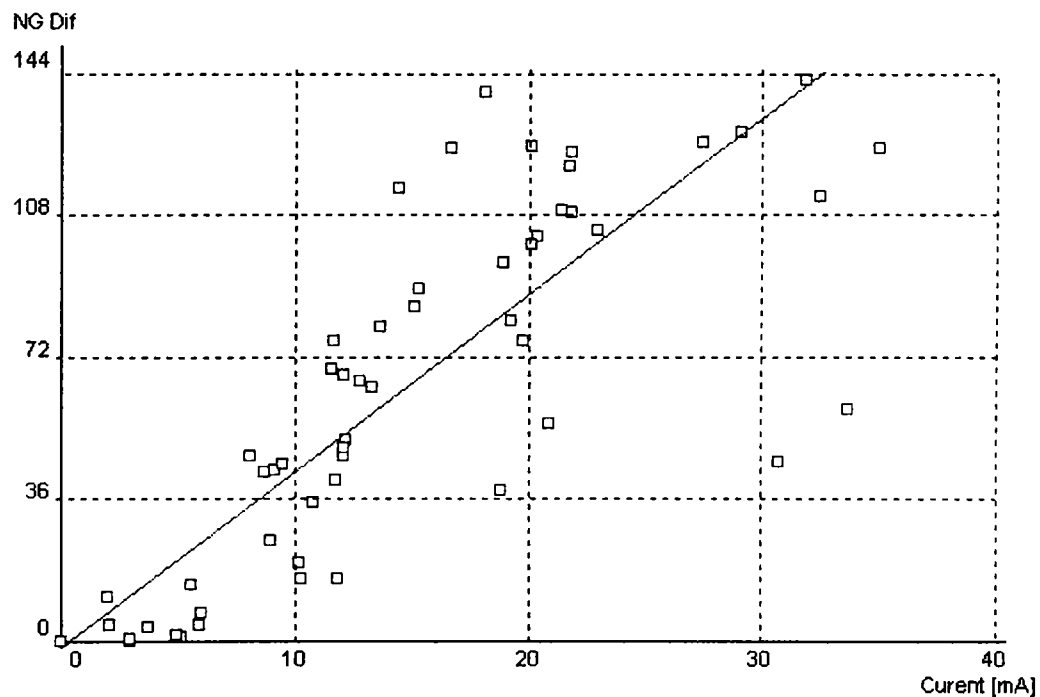


Figura 3.50 – Nivel de gri relativ

Eroarea patritică medie: 26.98  
Eroarea procentuală medie: 13.07%  
Eroarea procentuală maximă: 57.32%

### 3.4.3 Analiza fenomenelor termice

O a doua direcție de cercetare a fost studiul fenomenelor termice la nivelul plăcilor cu circuite integrate. În subcapitolul 3.1.3 s-a prezentat o metodă de rezolvare a ecuației de conducție termică pentru o placă plană. S-a obținut o formulă numerică de calcul a puterii consumate de un circuit integrat în funcție de distribuția de temperatură la nivelul aceluși circuit. Această formulă (3.1) a fost aplicată la imaginea termică a plăcii, obținându-se graficul din figura 3.51.

Chiar dacă această metodă este mai bună decât metodele precedente, nici aici nu se elimină influențele dintre circuite, astfel că și eroarea introdusă este destul de mare.

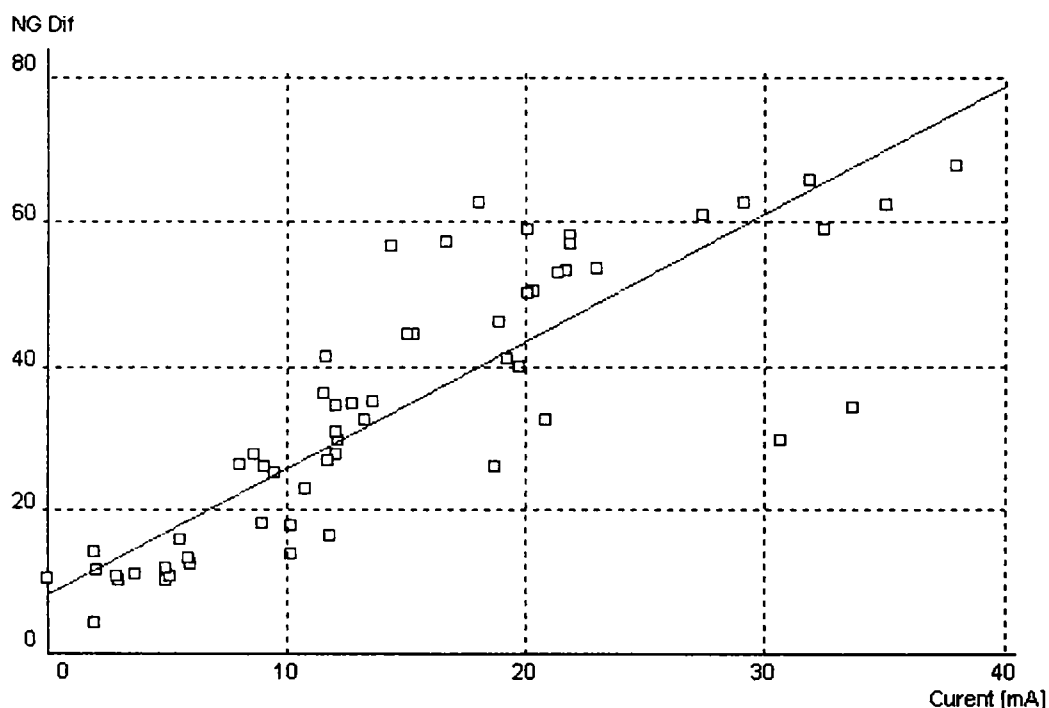


Figura 3.51

Eroarea patritică medie: 10.26  
Eroarea procentuală medie: 10.11%  
Eroarea procentuală maximă: 25.72%

### 3.5 Testarea pe bază de semnătură termică

Având la bază algoritmul propus în subcapitolul 3.2.1 și utilizând o aplicație creată (tot în contextul acestei lucrări) special pentru analiza imaginilor termice au fost realizate mai multe teste, rezultatele cele mai semnificative fiind prezentate în continuare.

#### Test 1

În acest test au fost luate câteva circuite răspândite pe întreaga placă, circuite care au un consum de curent foarte apropiat.

Circuit	Curent [mA]	Dimensiune	Culoare			Gauss			Eroare [%]
			Ng max	Ng med	Ng dif	A	w	h	
28	19.75	11x15	94	37	60.50	89.33	5.42	6.29	6.63
28	19.75	16x24	94	19	91.75	93.29	4.79	6.35	4.70
28	19.75	20x30	94	12	93.75	94.33	4.74	6.23	3.21
17	20.40	11x15	206	124	65.75	207.70	8.00	9.78	10.20
17	20.40	16x24	206	94	114.50	207.70	7.91	11.49	6.32
17	20.40	20x30	206	76	151.50	205.10	8.25	11.67	5.20
48	20.10	11x15	177	108	57.00	181.80	7.31	10.61	5.90
48	20.10	16x24	177	76	111.75	182.80	7.20	10.68	5.41
48	20.10	20x30	177	58	139.00	181.40	7.45	10.71	5.10
60	20.10	11x15	209	133	60.50	212.70	10.05	10.44	6.90
60	20.10	16x24	209	95	123.75	217.50	8.89	10.09	5.10
60	20.10	20x30	209	72	152.00	219.90	8.61	9.92	6.00
50	19.25	11x15	216	146	46.50	217.00	9.63	13.87	6.90
50	19.25	16x24	216	127	89.00	217.00	9.68	14.27	8.70

Tabelul 3.11 – Test 1

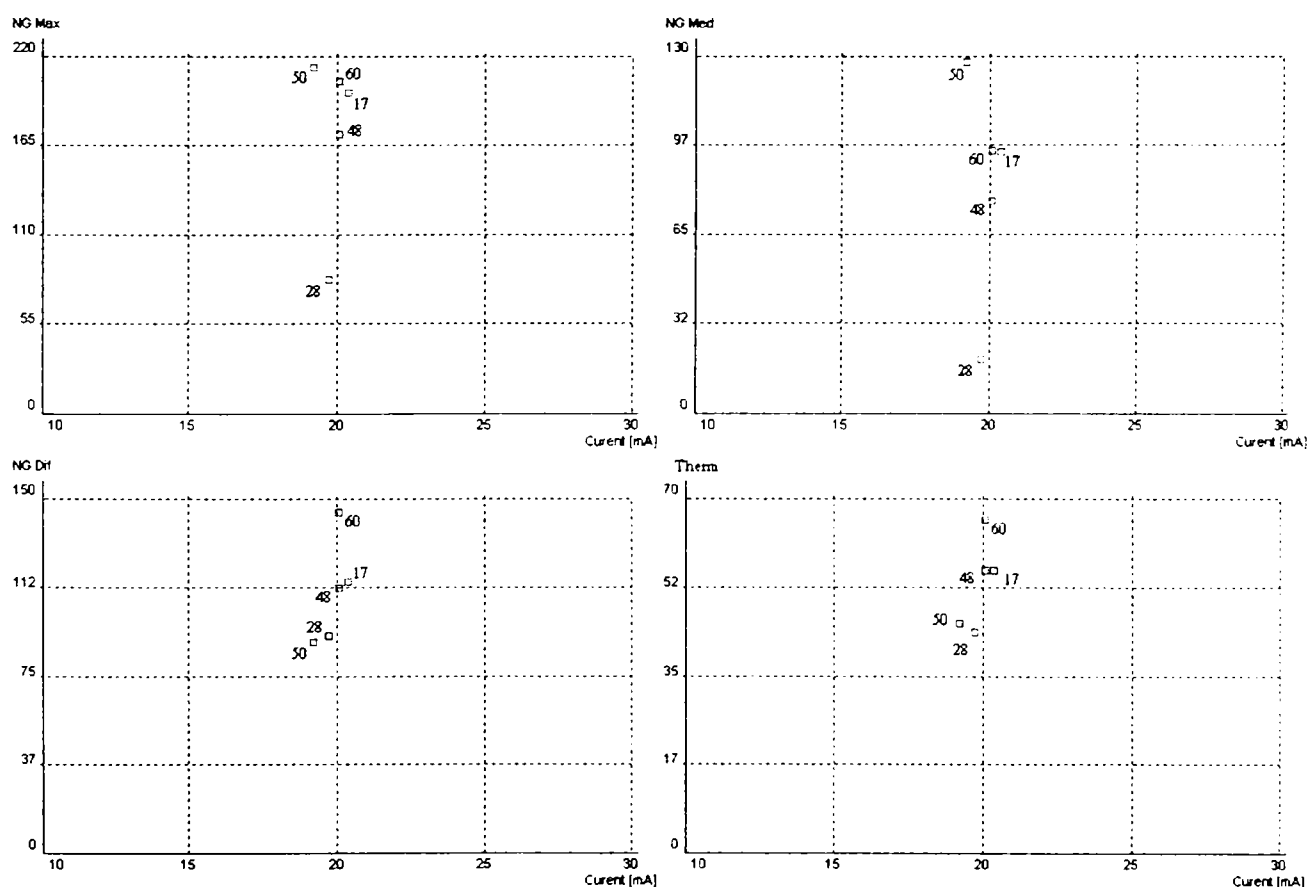


Figura 3.52 – Parametrii imaginii termice pentru circuitele selecționate în testul 1

Eroarea maximă pentru relația între nivelul de gri maxim și curenți: 32.82% (NG Max);  
 Eroarea maximă pentru relația între nivelul de gri mediu și curenți: 50.22% (NG Med);  
 Eroarea maximă pentru relația între nivelul de gri diferențial și curenți: 17.98% (NG Dif);  
 Eroarea maximă pentru relația între caracterizarea termică și curenți: 13.48% (Therm);  
 Eroarea maximă pentru relația între nivelul de gri obținut prin deconvoluție și curenți: 10.68% (figura 3.53).

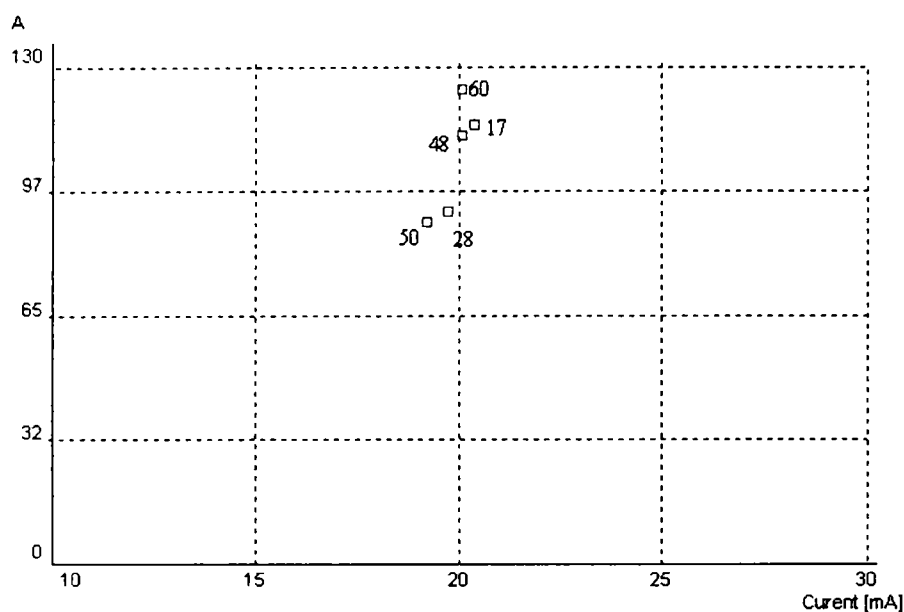


Figura 3.53 – A pentru circuitele selecționate în testul 1

Se poate observa din acest test că pentru circuite ce absorb același curenți imaginea termică a acestora este diferită și depinde de locația unde acestea sunt plasate pe placă. În plus se poate constata că cele mai bune distribuții pentru acest grup de circuite sunt obținute prin metodele: deconvoluției, termică și diferențelor.

### Test 2.

În acest test au fost alese acele circuite care au o distribuție termică asemănătoare, indiferent de cât consumă fiecare. Tabelul 3.12 conține rezultatele acestui test.

Circuit	Curenți [mA]	Dimensiune	Culoare			Gauss			Eroare [%]
			Ng max	Ng med	Ng dif	A	w	h	
28	19.75	11x15	94	37	60.50	89.33	5.42	6.29	6.63
28	19.75	16x24	94	19	91.75	93.29	4.79	6.35	4.70
28	19.75	20x30	94	12	93.75	94.33	4.74	6.23	3.21
43	12.80	11x15	75	26	54	75.00	6.19	4.63	7.84
43	12.80	16x24	75	12	73	77.69	5.03	4.53	4.52
43	12.80	20x30	75	8	74	78.90	4.87	4.53	3.08
33	8.70	11x15	73	26	48	63.00	5.06	6.41	6.89
33	8.70	16x24	73	16	61	56.90	5.51	7.50	6.89
33	8.70	20x30	73	13	65	53.25	6.74	7.76	7.77
61	11.55	11x15	75	26	54	79.29	5.85	4.50	7.25
61	11.55	16x24	75	12	71	81.13	5.21	4.46	5.26
61	11.55	20x30	75	8	74	82.80	4.88	4.46	3.79
10	12.10	11x15	80	28	54	78.22	4.24	6.40	7.06
10	12.10	16x24	80	16	69	71.41	4.68	7.10	5.55
10	12.10	20x30	80	11	58	58.10	5.93	7.11	4.94
27	11.75	11x15	63	26	37	68.44	4.66	6.50	6.91
27	11.75	16x24	63	15	49	61.05	5.32	7.11	5.64
27	11.75	20x30	63	11	58	58.11	5.93	7.11	4.94

Tabelul 3.12 – Test 2

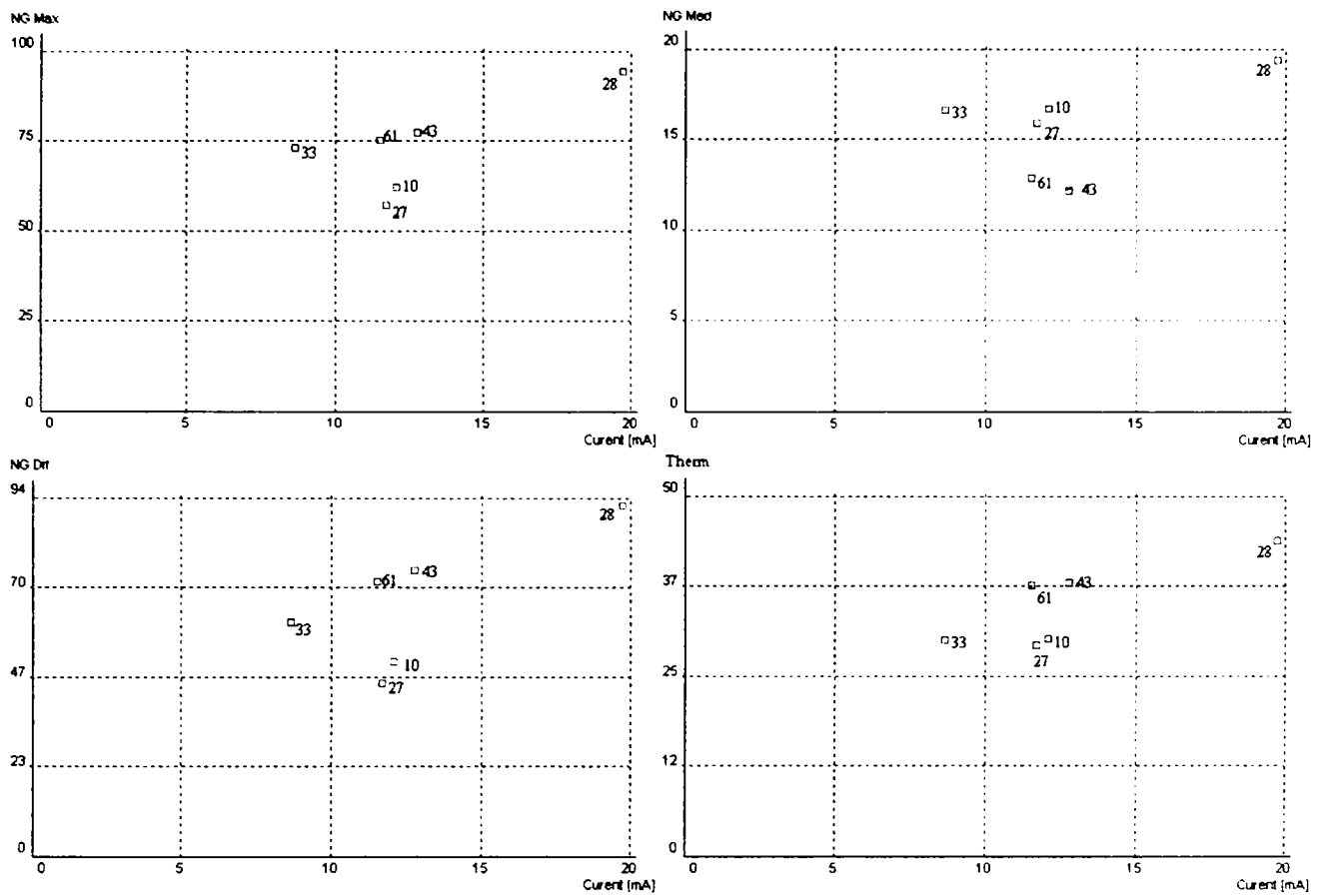


Figura 3.54 – Nivelul de gri maxim pentru circuitele selecționare în testul 2

Eroarea maximă pentru relația între nivelul de gri maxim și curent: 43.08% (NG Max);  
 Eroarea maximă pentru relația între nivelul de gri mediu și curent: 25.33% (NG Med);  
 Eroarea maximă pentru relația între nivelul de gri diferențial și curent: 23.58% (NG Dif);  
 Eroarea maximă pentru relația între caracterizarea termică și curent: 23.21% (Therm);  
 Eroarea maximă pentru relația între nivelul de gri obținut prin deconvoluție și curent: 21.83% (figura 3.55).

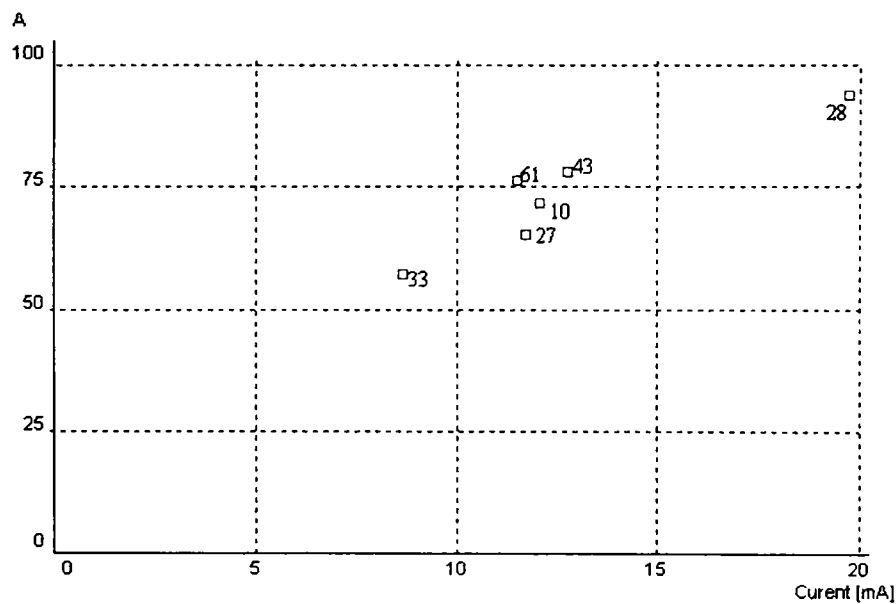


Figura 3.55 – A pentru circuitele selecționare în testul 2

Și din acest test se poate observa că pentru circuite ce absorb curent curenți semnificativ diferiți (8.70 mA – 19.75 mA) imaginile termice pentru aceste circuite este asemănătoare și depinde de locația unde acestea sunt plasate pe placă. În plus se poate

constata că cele mai bune distribuții pentru acest grup de circuite sunt obținute prin metodele: diferențelor, deconvoluției și termică.

Din cele prezentate până în acest moment, o imagine termică oferă o serie de informații (nivelele de gri maxim, mediu, relativ, suprafața de împrăștiere a căldurii degajate de un circuit, curentul calculat cu formula 3.1, raza conturului unui circuit, gauss-iana echivalentă etc.), dar toate aceste mărimi, luate fiecare individual nu oferă o acuratețe foarte mare pentru estimare puterii absorbite de un circuit pe baza termogramei (o eroare maximă între 10% și 50%).

### **Definiția 3.1**

Prin semnătură termică se înțelege răspunsul termic al dispozitivului testat (DUT) la aplicarea unui vector de test sau a unei secvențe de vectori de test.

### **Definiția 3.2**

Semnătura termică a unei plăci cu circuite integrate (PCB) este răspunsul termic al fiecărui circuit integrat de pe placă la aplicarea la intrările primare ale plăcii a unui vector de test sau a unei secvențe de vectori de test.

În cazul testării termice pe baza imaginilor în infraroșu, răspunsul termic al plăcii sau dispozitivului este considerat a fi imaginea acestuia în infraroșu. Informațiile oferite de o imagine termică se consideră parametrii ce pot fi utilizați pentru stabilirea semnăturii termice.

### **Definiția 3.3**

Semnătura termică uni-dimensională este definită ca fiind semnătura termică bazată pe un parametru al imaginii termice a plăcii.

### **Definiția 3.4**

Semnătura termică multi-dimensională este definită ca fiind semnătura termică bazată pe mai mulți parametrii din imaginea termică a plăcii.

Semnătura termică multi-dimensională poate fi formată din toți pixelii ce formează imaginea pentru un circuit sau poate conține doar parametrii de genul: maxim, mediu, etc.

Testarea pe baza semnăturii termice presupune determinarea semnăturii fiecărei plăci testate, iar dacă există diferențe semnificative între semnătura originală și cea a plăcii testate se consideră că placa conține defecte, iar pozițiile pentru care apar aceste diferențe indică circuitele defecte.

### **Semnătura termică uni-dimensională**

O primă propunere este aceea de a folosi semnătura termică a plăcii cu circuite integrate. Un exemplu de semnătură termică este cel în care se folosește nivelul de gri relativ (NG Dif). Pentru placa fără defecte, se ordonează și se indexează circuitele de pe placă în funcție de valoare nivelului de gri relativ determinat pentru respectivul circuit (figura 3.56).

În figura 3.57 este prezentată semnătura termică a unei plăci cu câteva circuite integrate defecte. Circuitele defecte sunt puse în evidență prin abaterea lor față de semnătura standard a circuitelor fără defecte.

Semnătura termică permite ca circuitele ce disipă aceeași putere să aibă parametrii termici diferiți în funcție de poziția acestora pe placă, vecini lor, coeficientul termic al materialului din care este făcut circuitul, plasarea elementelor de răcire, etc.



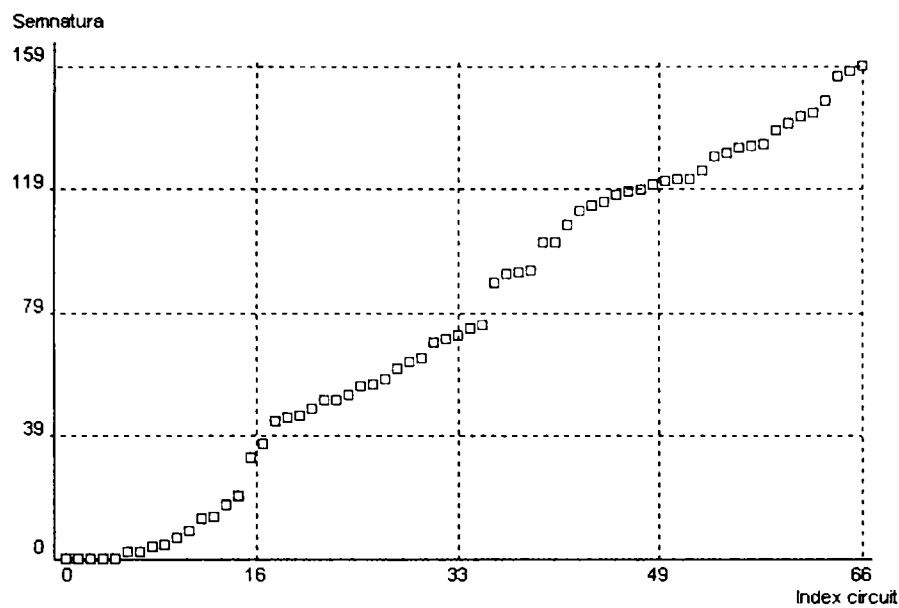


Figura 3.56 – Semnătura termică a plăcii fără defecte

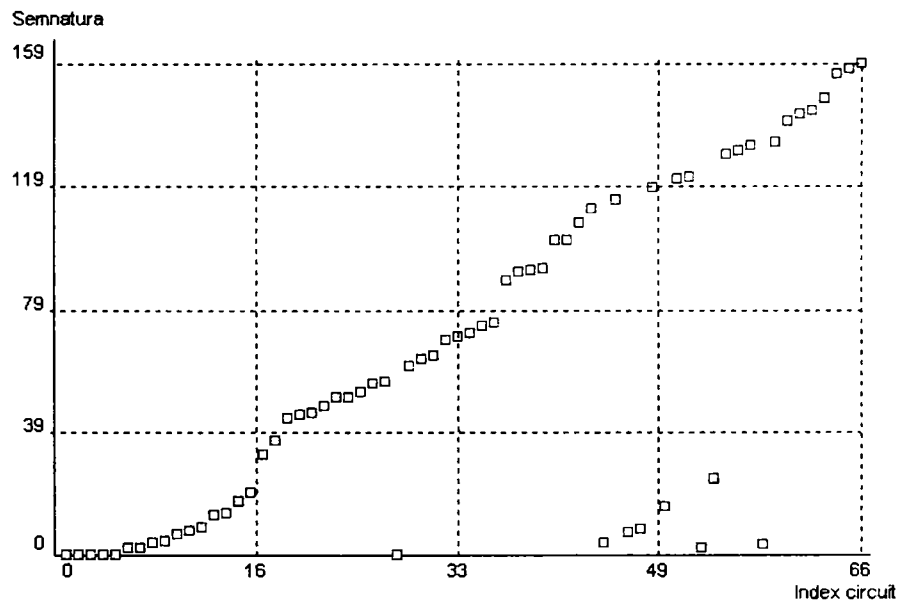


Figura 3.57 – Semnătura termică a plăcii cu defecte

### Semnătura termică multi-dimensională

A doua propunere de testare pe baza semnăturii termice vizează luarea în calcul a mai multor caracteristici determinate din imagine. Astfel se poate spune că problema noastră se poate reduce la o problemă de clasificare a circuitelor de pe placă (din termogramă) pe baza informațiilor determinate din imaginea termică. S-a considerat că cea mai simplă metodă de implementare a unui sistem clasificator ar fi o rețea neuronală feed-forward.

În continuare voi prezenta și discuta rezultatele aplicării rețelelor neuronale feed-forward în testarea termică. Și în acest caz s-au făcut mai multe teste, folosind rețele neuronale de diferite dimensiuni, antrenându-le pe imagini ideale sau pe imagini reale, direct pe imaginile circuitelor sau pe diferite caracteristici ale acestora. Aplicația realizată permite construirea de diferite rețele neuronale cu un număr configurabil de intrări și ieșiri și cu un număr oarecare de nivele ascunse, fiecare nivel putând fi configurat cu un anumit număr de neuroni (figura 3.58).

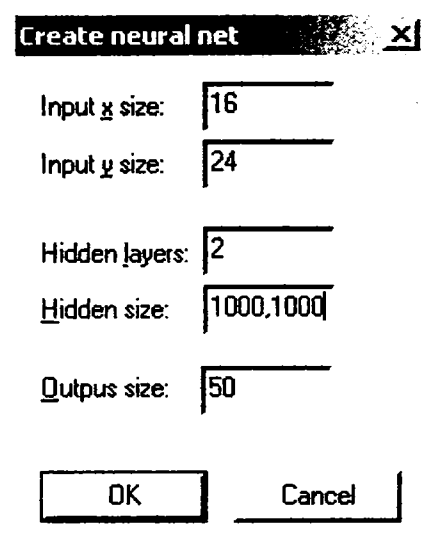


Figura 3.58

Rețelele neuronale utilizate au fost de tipul celei prezentate în figura 3.59, modificându-se dimensiunile acesteia:

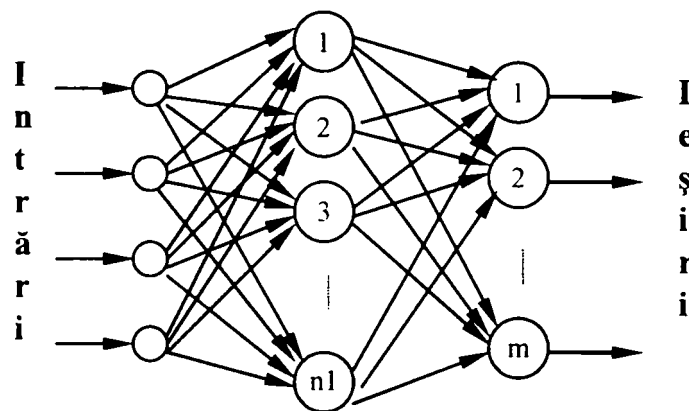


Figura 3.59 – Tipul de rețea neuronală utilizată

Și în acest caz au fost făcute mai multe teste pe imagini ideale și pe imagini reale, acestea fiind orientate în două direcții: una în care au fost folosite ca și tipare pentru rețeaua neuronală imaginea circuitelor, iar a doua în care au fost folosite caracteristicile imaginii circuitului (nivel de gri maxim, mediu, relativ, etc.).

### Test 1 – Semnătura termică pentru imagini ideale

Datorită faptului că nu s-au putut folosi foarte multe imagini reale și nici nu s-a găsit cameră video în infraroșu cu rezoluția necesară imaginilor termice pentru plăcile cu circuite integrate, un prim set de teste a fost aplicat pe imagini termice generate artificial. Antrenarea rețelei neuronale presupune existența unui set relativ mare de date. Din acest motiv au fost generate 50 de imagini termice cu 66 de circuite integrate pe imagine, care au fost folosite la antrenarea rețelei. Apoi rețeaua a fost folosită pe un alt set de imagini generate aleator, rezultatul fiind prezentat în figura 3.60.

Rețea neuronală feed-forward (2 straturi ascunse, 150 neuroni pe fiecare strat)

Clasa 1: 0 mA - 40 mA;

Clasa 2: 40 mA - 80 mA;

Clasa 3: >80 mA - faulty.

Număr tipare: 3500

Antrenarea rețelei: ~3 h, 1300 epoci

Rata de învățare: 99.4%

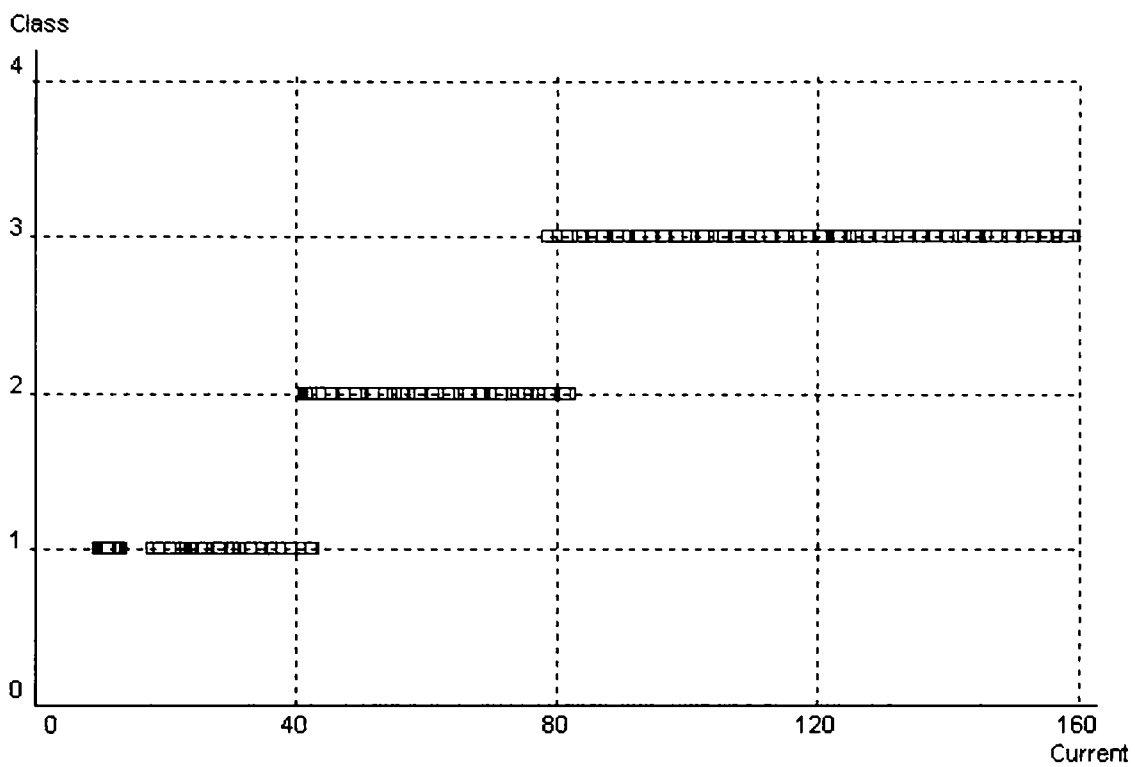


Figura 3.60 – Clasificare imaginilor termice ideale

**Test 2 – Semnătura termică pentru imagini reale**

Cel de-al doilea set de test a avut la bază imaginile reale ale unei plăci cu circuite integrate. Rețeaua neuronală folosită a fost organizată pe trei straturi: un strat de intrare ce conținea 5 unități, un strat de ieșire cu 3 unități și un strat ascuns ce conținea 100 de neuroni. Această rețea a fost antrenată cu o rată de 100% pe un set de 172 de tipare, a durat aproximativ 5 min și s-a întins pe 2133 de epoci (figura 3.60).

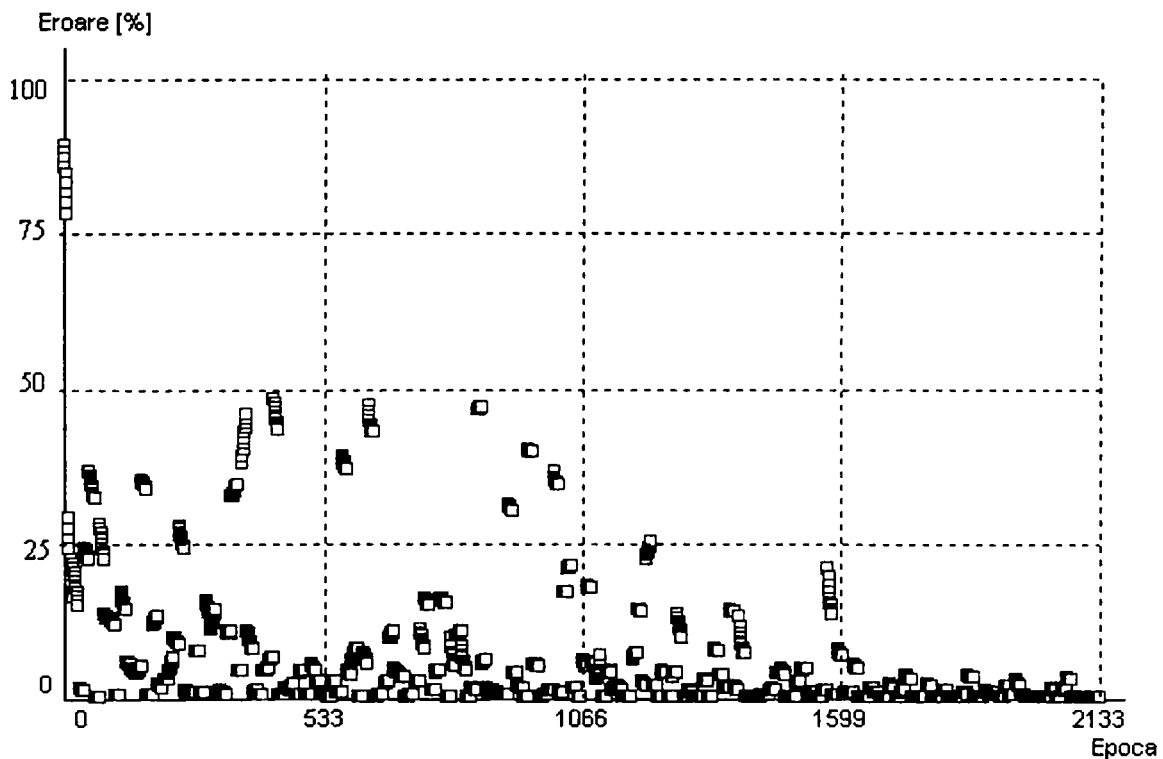


Figura 3.60 – Evoluția erorii în antrenarea rețelei neuronale

Rețea neuronală feed-forward (1 strat ascuns, 100 neuroni pe strat)

Clasa 1: circuitele ce absorb un curent între 0 mA și 26 mA;

Clasa 2: circuitele ce absorb un curent între 26 mA și 46 mA;

Clasa 3: circuitele ce absorb un curent de peste 46 mA – acestea fiind și circuitele defecte.

Număr tipare: 172

Antrenarea rețelei: ~5 min, 2100 epoci

Rata de învățare: 100%

Clasificarea a fost realizată în proporție de 100% (figura 3.61).

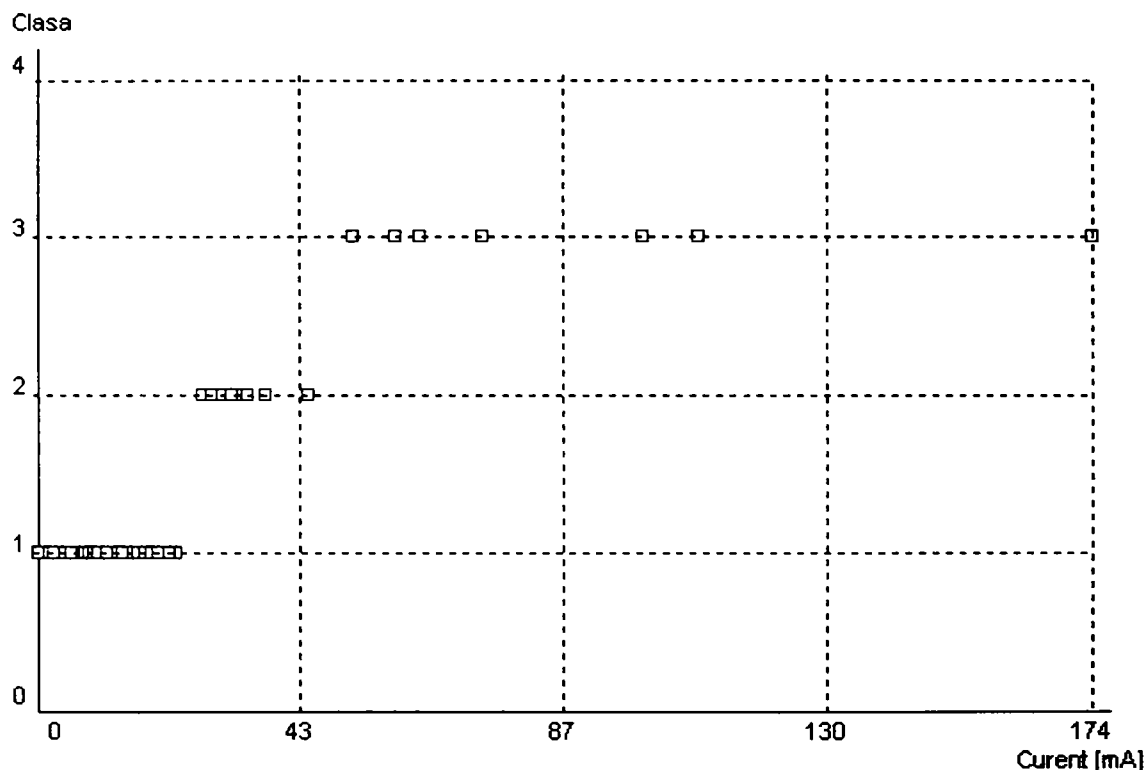


Figura 3.61 – Clasificare circuitelor de pe o imagine reală

Atât pe imaginile ideale cât și pe cele reale clasificarea s-a efectuat corect în proporție foarte mare (peste 99%). Pentru a elimina efectul nedorit al clasificărilor incorecte ale circuitelor ce se află la limita de trecere între două clase se pot introduce mai multe clase intermediare.

Comparativ cu metoda de testare termică utilizată în prezent, metodă ce folosește compararea imaginii termice a plăcii testate cu un model de imagine statistică obținută pentru plăcile fără defecte, metoda propusă prezintă următoarele avantaje:

- modelul statistic este înlocuit de rețeaua neuronală, rețea ce poate fi ușor actualizată prin (auto)antrenarea ei în mod continuu pe toată durata de viață a produsului;
- reducerea necesarului de stocare, deoarece nu se mai stochează imagini ci doar rețeaua neuronală;
- chiar dacă timpul de antrenare este relativ mare, faza de clasificare, cea care contează în procesul de testare, este foarte rapidă.

Independent de lucrarea de față, în faza de redactare a tezei s-a găsit o lucrare care confirmă propunerea și testele de față. În [Hsieh2004] este folosită o rețea neuronală aplicată imaginilor termice pentru stabilirea unui profil termic dependent de stresul la care sunt supuse componentele electronice.

## 3.6 Aplicația pentru studiul imaginilor în infraroșu

Pentru studiul imaginilor termice ale plăcilor cu circuite integrate s-a implementat o aplicație, dezvoltată în mediul Microsoft Visual C++ 6.0. Aplicația este structurată pe mai multe module:

- modulul de generare de imagini;
- modulul de prelucrări de imagini;
- modulul de deconvoluție;
- rețeaua neuronală;
- interfața cu utilizatorul.

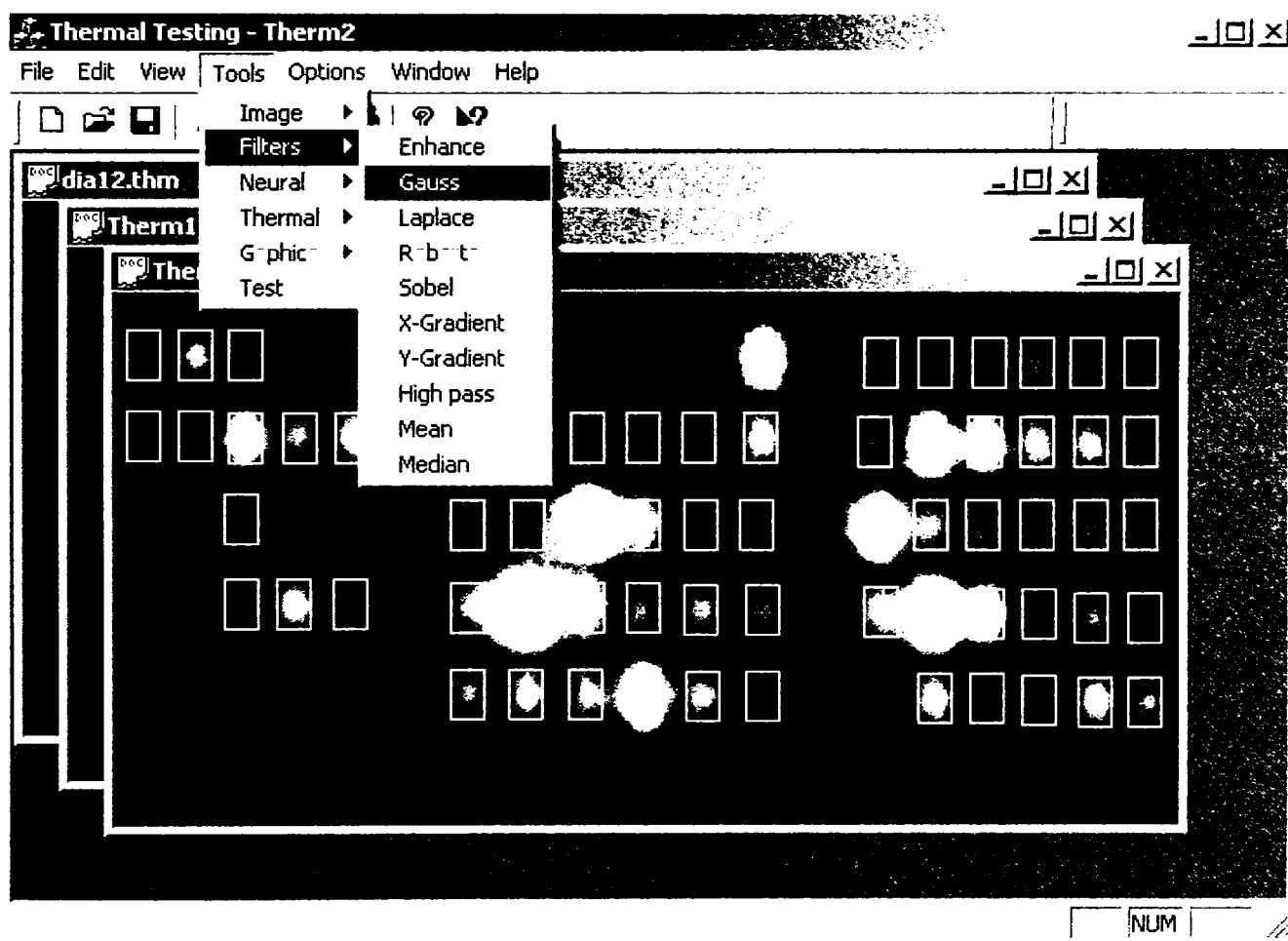


Figura 3.62 – Interfața aplicației de studiu a imaginilor termice

Modulele de generare și procesare de imagini sunt încapsulate în câte o clasă (CImageGenerator și CImageProcessor prezentate în figura 3.63) și implementează următoarele operații:

- aplică unei imagini zgomote de o anumită amplitudine;
- crează imagini cu distribuție gaussiană;
- convoluționează două imagini;
- aplică unei imagini un filtru de un anumit tip: Gauss, median, mediere, Laplace, Roberts, Sobel, gradient, accentuarea detaliilor, filtru trece-sus;
- egalizarea histogramei.

Aceste două clase operează asupra unei imagini ce este încapsulată într-o altă clasă (CImageContainer), ce conține un bitmap cu imaginea și un tampon cu valori reale, câte una

pentru fiecare pixel din imagine. CImageContainer oferă și câteva operații primare pentru imagini: scalare, copiere, zoom, comparare, inversare.

Modulul de deconvoluție se folosește de un procesor de imagini și de un generator de imagini pentru implementarea algoritmilor de deconvoluție a imaginilor termice. Codificarea cromozomială a imaginilor folosite în deconvoluție se face prin niște clase derivate din CPopulationData.

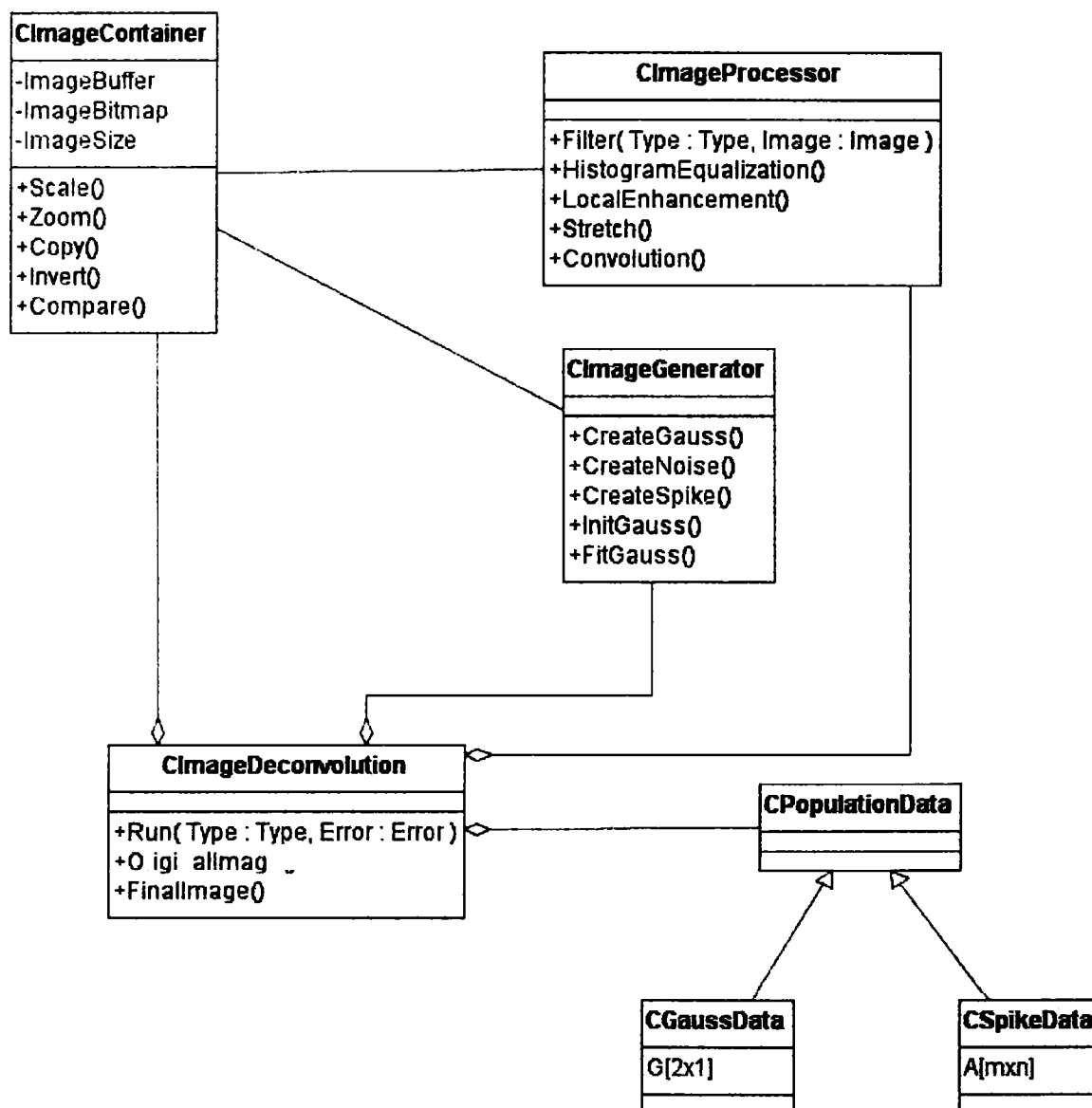


Figura 3.63 – Clasele ce implementează modulele de operare asupra imaginilor

Aplicația realizată permite construirea de diferite rețele neuronale cu un număr configurabil de intrări și ieșiri și cu un număr oarecare de nivele ascunse, fiecare nivel putând fi configurat cu un anumit număr de neuroni (figura 3.58, 3.59). Rețeaua neuronală este implementată și distribuită în trei clase: CNeuron, CNeuralLayer și CNeuralNetwork (figura 3.64). Un neuron are o ieșire și mai multe intrări, fiecare intrare având câte o pondere. Intrările unui neuron sunt niște pointeri către elementele unui tampon ce se găsește la nivelul stratului rețelei ce conține respectivul neuron. Ieșirea unui neuron este un pointer către un element al unui tampon din nivelul următor celui ce conține neuronul. Un nivel al rețelei (CNeuralLayer) conține o listă de neuroni și un vector de intrare la care sunt conectați neuronii. Rețeaua neuronală (CNeuralNetwork) conține unul sau mai multe niveluri, o listă de pointeri de intrare ce arată spre lista de intrări a primului nivel și o listă de ieșiri la care sunt legate ieșirile neuronilor de pe ultimul strat.

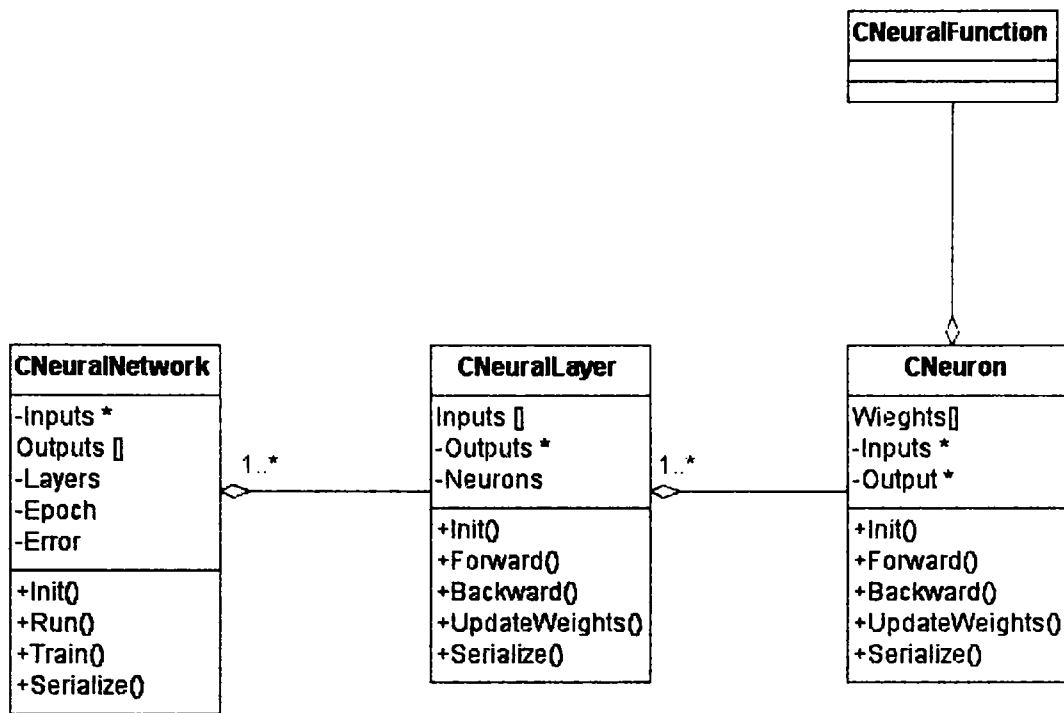


Figura 3.64 – Rețeaua neuronală

Rețeaua neuronală funcționează în două moduri: unul de antrenare pe baza unui set de tipare și unul de execuție. În faza de antrenare se aplică algoritmul de antrenare al unei rețele neuronale feed-forward, în cei doi pași Backward și Forward implementați la nivel de neuron și la nivelul fiecărui strat al rețelei. Rețeaua poate fi antrenată cu diferiți parametri ai imaginii termice (semnătura termică multi-dimensională) sau cu imaginea fiecărui circuit de pe placă. Rețeaua antrenată poate fi serializată într-un fișier pentru a putea fi utilizată oricând pe orice imagine.

În faza de execuție rețeaua antrenată funcționează ca un sistem de clasificare pentru circuitele prezente pe placă. La intrările rețelei se introduc parametrii imaginii termice a circuitului ce trebuie clasificat iar la ieșirile rețelei se va stabili rezultatul clasificării.

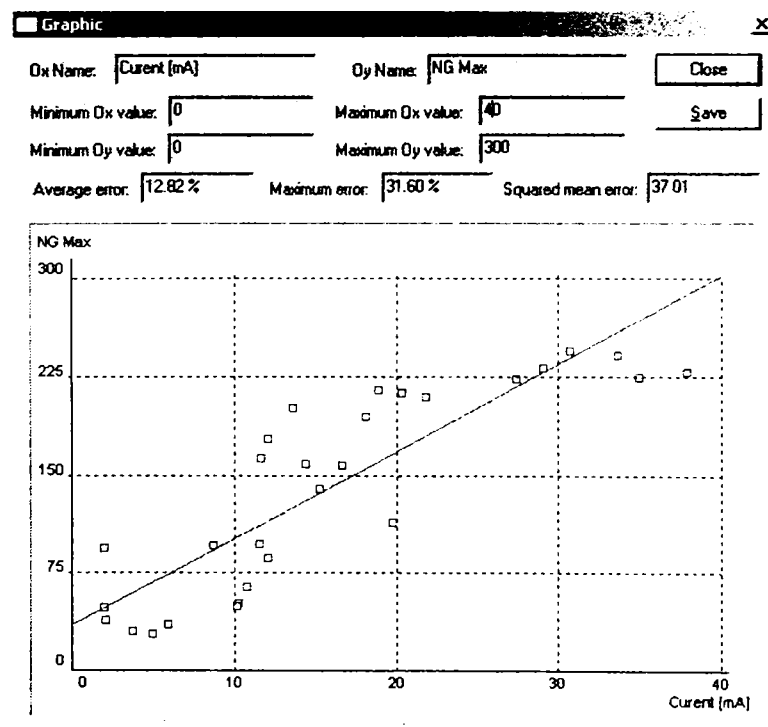


Figura 3.65 – Modulul de prezentare a graficelor

Modulul de interfață cu utilizatorul conține:

- o fereastră de dialog pentru descrierea parametrilor circuitului (figura 3.16);
- o fereastră de dialog pentru algoritmul de deconvoluție (figura 3.30);
- o fereastră de dialog generală pentru reprezentări grafice (figura 3.65).

Aplicația este scrisă în C++, are aproximativ 50 de clase și 11000 de linii de cod (figura 3.66). Testele din acest capitol au fost executate pe un laptop cu procesor Intel Pentium III la 1 GHz, 256 MB memorie, sistem de operare Windows XP.

**Project Line Counter - Results shown for 108 file(s)**

File Statistics Help

File Name	Extension	Path	Lines	Code	Comments	Mixed	Blank	
messageobject	.cpp	e:\work\phd...	37	14	18	0	5	
messageobject	.h	e:\work\phd...	63	30	26	2	9	
messagetype	.h	e:\work\phd...	42	22	17	2	5	
messagetype	.cpp	e:\work\phd...	36	13	18	0	5	
messageview	.h	e:\work\phd...	43	19	18	2	8	
messageview	.cpp	e:\work\phd...	89	58	18	0	13	
neuraldialog	.cpp	e:\work\phd...	81	56	13	1	13	
neuraldialog	.h	e:\work\phd...	52	24	22	4	10	
neuralfunction	.h	e:\work\phd...	41	30	5	2	8	
neuralfunction	.cpp	e:\work\phd...	73	57	7	1	10	
neurallayer	.h	e:\work\phd...	73	56	7	3	13	
neurallayer	.cpp	e:\work\phd...	166	133	17	1	17	
neuralnetwork	.h	e:\work\phd...	65	48	5	2	14	
neuralnetwork	.cpp	e:\work\phd...	324	264	32	11	39	
neuron	.cpp	e:\work\phd...	79	60	6	0	13	
neuron	.h	e:\work\phd...	60	43	11	8	14	
numberconvertor	.h	e:\work\phd...	42	23	13	2	8	
numberconvertor	.cpp	e:\work\phd...	170	133	19	0	18	
populationdata	.h	e:\work\phd...	36	25	5	2	8	
populationdata	.cpp	e:\work\phd...	150	125	8	0	17	
...	...	...	...	...	...	...	...	

Total:	Lines	Code	Comments	Mixed	Blank	Net Lines
#:	15,998	11,140	2,826	222	2,254	13,744
%:	100%	69%	17%	1%	14%	85%

Filter: <Workspace>    Ctrl+Click headers for multiple-sort    Close

Figura 3.66 – Sursele aplicației



## 3.7 Concluzii

Lucrarea de față s-a concretizat în propunerea unei noi metode de testare a plăcilor cu circuite integrate pe baza semnăturii termice. Semnătura termică a fost obținută în două moduri: prin utilizarea unei caracteristici a imaginii termice sau prin utilizarea unei rețele neuronale artificiale cu propagare înainte. Au fost făcute mai multe teste pentru fiecare din cele două metode, cele mai semnificative rezultate fiind prezentate în acest capitol.

Cea de-a doua contribuție prezentată în acest capitol, algoritmul de deconvoluție a imaginilor termice, își propune obținerea unei imagini termice ideale din imaginea achiziționată de camera video infraroșu, imagine pe baza căreia se poate estima puterea disipată de fiecare circuit de pe placă. Algoritmul de deconvoluție are la bază un algoritm genetic modificat cu o procedură de căutare adaptivă în faza de selecție pentru reducerea timpului de căutare. Sunt prezentate două variante ale acestui algoritm: una pentru surse de căldură uniforme și cealaltă pentru surse de căldură neuniforme. În cadrul capitolului au fost prezentate mai multe teste prin care a fost validat algoritmul: influența zgomotelor asupra deconvoluției, influența filtrelor de aplatizare, influența dimensiunii imaginii și a numărului de circuite.

Contribuțiile acestui capitol sunt:

- propunerea, implementarea și testarea unei metode de deconvoluție a imaginilor și semnalelor pe baza unui algoritm pseudo-genetic (IJECE'2004);
- modificarea algoritmului standard de deconvoluție pentru a implementa deconvoluția imaginilor termice cu surse de căldură neuniforme (HACIPPR'2005);
- deconvoluția imaginilor termice și aproximarea acestora prin curbe gauss;
- propunerea, implementarea și testarea unei metode de testare pe baza semnăturii termice a unei plăci cu circuite integrate (SSD'05);
- implementarea semnăturii termice utilizând rețelele neuronale și compararea ei cu metoda scăderii imaginilor (IMCT'2005);
- proiectare și implementarea unei aplicații destinată studiului și analizei imaginilor termice a plăcilor cu circuite integrate.

Apariția acestor metode de testare termică se datorează atât creșterii complexității circuitelor și scăderii numărului pinilor de test în PCB-uri (care au condus la scăderea eficienței testărilor circuitelor electronice prin metode convenționale), cât și datorită faptului că echipamentele de testare obișnuite (ATE sau "pat-de-cuie") necesită timpi foarte mari în testarea in-circuit. Câteva din avantajele folosirii tehnicilor de testare termică sunt prezentate în continuare.

Avantajele testării termice:

- nu este necesar un contact fizic cu unitatea ce se testează;
- este independentă de numărul de terminale;
- efectul defectului nu trebuie propagat la ieșirile observabile ale circuitului;
- număr mic de vectori de test;
- acoperire bună a defectelor de scurt-circuit, inclusiv cele rezistive;
- cost scăzut – în cazul utilizării senzorilor de temperatură;
- nu produce deteriorări produsului testat;
- localizarea exactă a erorii (metoda termo-optică).

Comparativ cu metoda de testare termică utilizată în prezent, metodă ce folosește compararea imaginii termice a plăcii testate cu un model de imagine statistică obținută pentru plăcile fără defecte, metoda propusă prezintă următoarele avantaje:

- modelul statistic este înlocuit de rețeaua neuronală, rețea ce poate fi ușor actualizată prin (auto)antrenarea ei în mod continuu pe toată durata de viață a produsului;
- reducerea necesarului de stocare, deoarece nu se mai stochează imagini ci doar rețeaua neuronală;
- chiar dacă timpul de antrenare este relativ mare, faza de clasificare, cea care contează în procesul de testare, este foarte rapidă.

Dezavantajele testării termice:

- lentă;
- depinde de tipul de împachetare folosit și de mediul de utilizare;
- cost mare pentru cazul testării termice în infraroșu (costul unei camere video în infraroșu aplicabilă la circuitele integrate costă între 10000 și 100000 USD);
- în cazul utilizării senzorilor de temperatură integrați aceștia ocupă spațiu din pastila de siliciu și o metodă de interconectare;
- datorită procesului de scalare a dimensiunilor crește temperatura de operare a circuitelor de pe placă și implicit a plăcii.

Pe de altă parte, deoarece căldura are efecte semnificative asupra performanței și fiabilității circuitelor, unul din factorii principali ce trebuie luați în considerație la proiectarea circuitelor integrate și a plăcilor cu circuite integrate, este controlul căldurii disipate. Independent de lucrarea de față, în faza de redactare a tezei s-a găsit o lucrare care confirmă propunerea și testele prezentate în acest capitol. În [Hsieh2004] este folosită o rețea neuronală aplicată imaginilor termice pentru stabilirea unui profil termic dependent de stresul la care sunt supuse componentele electronice.

# 4 Monitorizarea și controlul temperaturii în sistemele de calcul

## 4.1 Metode electrice de măsurare a temperaturii

### 4.1.1 Tipuri de senzori de temperatură

În Capitolul 2 s-au prezentat cele două mari metode de determinare a temperaturii utilizate în domeniul calculatoarelor: metoda electrică și metoda optică. Metoda optică de determinare a temperaturii împreună cu aplicabilitatea ei în testarea termică a fost prezentată în capitolele 2 și 3. Există mai multe metode electrice de măsurare a temperaturii, pentru fiecare existând mai multe tipuri de senzori de temperatură. Nu există însă, o metodă de măsurare a temperaturii care să fie universal valabilă pentru toate categoriile de aplicații; o metodă poate să dea rezultate bune pentru o categorie de aplicații dar pentru altă categorie de aplicații poate să eșueze. În continuare voi prezenta cei mai des folosiți senzori de temperatură: termocuplul, termorezistența (RTD), termistorul și senzorul de temperatură integrat în pastilă de siliciu.

Fiecare tip de senzor are o serie de parametrii care îl caracterizează: domeniul de măsurare a temperaturii, sensibilitatea, linearitatea, dimensiunea, prețul senzorului și multe altele. Caracteristicile celor patru tipuri de senzori de temperatură sunt prezentate în tabelul 4.1.

	Termocuplul	RTD	Termistor	Siliciu
Intervalul de temperatură [°C]	[-279,1800]	[-250,900]	[-100,450]	[-55,150]
Acuratețe [°C]	±0.5	±0.01	±0.1	±1
Mărime de ieșire	Tensiune	Rezistență	Rezistență	Tensiune, Curent, Digital
Sensibilitate	10 μV/°C	3 mΩ/°C	1 Ω/°C	2 mV/°C
Linearitate	polinom de grad 4	polinom de grad 2	polinom de grad 3	liniar
Ruggedness				
Timp de răspuns [s]	< 1	[1,10]	[1,5]	[4,60]
Mod de excitare	Nu	Sursă de curent	Sursă de tensiune	Sursă de tensiune
Preț [USD]	[1,50]	[25,1000]	[2,10]	[1,10]

Tabelul 4.1 – Caracteristicile principalilor tipuri de senzori [Baker1999]

Termocuplul este format din două conductoare de metal sau aliaje diferite care sunt sudate împreună [Baker1999]. Punctul de sudură al aliajelor se poziționează în locul în care se dorește măsurarea temperaturii. Funcționarea termocuplului se bazează pe efectul termoelectric direct, prin care se produce o diferență de potențial între cele două joncțiuni, această diferență de potențial fiind proporțională cu temperatura măsurată. Teoretic termocuplul se poate realiza din orice două metale diferite, dar practic există numai câteva tipuri de termocupluri, deoarece se dorește ca acestea să fie cât mai sensibile la variațiile de temperatură. În comparație cu celelalte tipuri de senzori termocuplul are cea mai mare

neliniaritate, astfel că necesită algoritmi de liniarizare sau tabele de conversie. Timpul de răspuns al termocuplului pentru temperaturi ridicate este mai mic raportat la timpul de răspuns al termistorului sau RTD-ului. Domeniul termocuplului ( $-270^{\circ}\text{C}$ ,  $+1800^{\circ}\text{C}$ ) este mai mare decât domeniul termistorului sau RTD-ului, așadar se poate folosi în condiții extreme de temperatură. Prețul termocuplurilor este scăzut iar ca și dimensiune termocuplurile sunt mici. Ele pot fi încapsulate într-un tub protector format din ceramică și metal. Dezavantajele termocuplurilor sunt: semnalul de ieșire (diferența de potențial) este slab, necesită calibrare termică, semnalul de ieșire nu este liniar, acuratețea este scăzută.

Termorezistența are la bază variația rezistenței electrice a materialelor conductoare cu variația temperaturii. Pentru a produce senzori de temperatură RTD de calitate și acuratețe mare trebuie alese cu atenție materialele componente ale acestuia. Aceste materiale sunt de obicei metalele rezistive: Platină, Nichel sau Cupru. Aceste metale sunt alese pentru că la o variație a temperaturii au o variație mare a rezistenței. Senzorii sunt caracterizați prin rezistența lor la  $0^{\circ}\text{C}$ . Relația dintre variația temperaturii și variația rezistenței RTD-ului este aproape liniară. Dezavantajele RTD-ului sunt: pot apărea erori datorită impurității materialelor din care este construit, vibrațiile și șocurile au efect negativ asupra repetabilității măsurării, apare efectul electrocaloric care induce erori suplimentare în procesul de măsurare. Prețul senzorilor RTD este mult mai mare decât a celorlalți senzori așa că nu este rentabil folosirea lor în toate tipurile de aplicații.

Termistorul este un element de circuit cu proprietăți semiconductoare a cărui rezistență variază cu temperatura după o lege exponențială. Există două tipuri de termistoare: cu coeficient negativ de temperatură (NTC) și cu coeficient pozitiv de temperatură (PTC). Termistorul NTC este construit din oxizi metalici, (de exemplu: oxid de mangan, oxid de nichel, oxid de cobalt, oxid de fier sau oxid de zinc) sinterizați. Acesta are acuratețea mult mai mare decât un senzor RTD, așadar este și mult mai sensibil la variațiile de temperatură. Termistorul PTC, are un coeficient de temperatură pozitiv și este construit din titanat de bariu. Sensibilitatea termistorului PTC este mult mai mare față de termistorul NTC. Domeniul de măsurare al termistorului PTC este ( $-25^{\circ}\text{C}$ ,  $+150^{\circ}\text{C}$ ).

Senzorul de temperatură integrat în pastilă de siliciu este o alternativă viabilă pentru a realiza monitorizarea temperaturii. Avantajele acestui tip de senzor sunt: semnalul de ieșire este ușor de prelucrat; se poate instala ușor pe o placă de test, preț scăzut. De obicei acești senzori au și un circuit de prelucrare a semnalului integrat în același pastilă de siliciu; așadar semnalul de ieșire fiind în format digital. Rezultă că nu mai este nevoie de circuite suplimentare de prelucrare a semnalului pentru a realiza interfațarea senzorului cu un microcontroler sau microprocesor. Dezavantajele senzorilor integrați sunt: acuratețe mai scăzută decât la celelalte tipuri de senzori; domeniul de măsurare scăzut.

În cazul sistemelor de calcul, cel mai potrivit loc pentru senzorii de temperatură este cât mai aproape de sursa de căldură, adică în interiorul capsulei, iar ideal ar fi pe suprafața pastilei de siliciu. Cerințele pentru senzorii de temperatură integrați sunt următoarele:

- compatibilitate cu procesul tehnologic;
- dimensiune redusă;
- consum redus;
- interfață în format digital.

Temperatura în orice punct de pe suprafața de siliciu poate fi măsurată folosind o diodă și o sursă stabilă de curent. Căderea de tensiune pe diodă este dependentă de curentul care trece prin ea și temperatura joncțiunii diodei. Prin menținerea unui curent constant și stabil prin diodă, tensiunea pe diodă poate fi utilizată ca o măsură exactă a temperaturii de operare a diodei [Gunther2001].

Senzorii de temperatură diferențiali produc o tensiune proporțională cu diferența de temperatură a celor două puncte de monitorizare plasate pe suprafața de siliciu. Avantajul

acestei strategii de măsurare este obținerea unui grad mai mare de imunitate la variațiile de deplasare ale hărții termice la suprafața siliciului [Altet1997]. Aceste variații ar putea proveni de la schimbări ale temperaturii ambientului sau de la componentele statice ale puterii disipate de circuitul integrat.

Acuratețea senzorilor prezentați este destul de slabă: între 1 °C și 3 °C. Aria ocupată de acești senzori e de aproximativ 0.5 mm<sup>2</sup> în procesul tehnologic de 0.7 microni. Puterea disipată este însă foarte mică, de aproximativ 7μW [Szekely1998]. Ieșirea acestor senzori constă într-o valoare analogică, care ar trebui digitizată într-o valoare numerică. Un alt exemplu de senzor, prezentat în [Szekely1998\_1], se bazează pe dependența de temperatură a parametrilor unui tranzistor MOS. Acest senzor are curentul ca și componentă analogică dependentă de temperatură, iar ieșirea digitală este dată de un convertor curent-frecvență. Ieșirea în frecvență poate fi introdusă într-un numărător, existând astfel posibilitatea de a oferi o valoare numerică digitală dependentă liniar de temperatură. Acuratețea măsurării e de 0.35 °C, aria ocupată de senzor este 0.005 mm<sup>2</sup> pentru procesul tehnologic de 0.8 microni. Puterea disipată este însă destul de mare: 200 μW.

După cum s-a putut observa, există numeroase tipuri și modele de senzori. În urma creșterii frecvenței de lucru a microprocesoarelor și a puterii consumate într-un sistem de calcul, s-a pus problema monitorizării și controlului anumitor parametri cum ar fi: temperatura, viteza de rotație a ventilatoarelor și valoarea efectivă a tensiunilor de lucru. Astfel au apărut senzorii integrați pe pastila de siliciu, cu informația convertită în format digital, care asigură o precizie și un timp de măsurare acceptabil, și totodată o integrare și interfațare facilă în majoritatea aplicațiilor.

În vederea asigurării unor soluții complete, producătorii plăcilor de bază au început integrarea acestor senzori direct pe plăcile de bază. La început numărul de senzori integrați pe pastila de siliciu a fost de unul singur. Cu timpul însă au apărut soluții integrate care includ senzori pentru măsurarea temperaturii, tensiunii și a vitezei de rotație a ventilatoarelor. Astfel o soluție integrată are 2 sau chiar mai mulți senzori de temperatură (1 măsurând temperatura CPU-ului, 1 măsurând temperatura globală a sistemului, iar restul depinzând de produs), 2 sau chiar mai mulți senzori pentru măsurarea vitezei de rotație a ventilatoarelor (1 pentru CPU, 1 pentru sistem, etc), 4 până la 6 senzori pentru măsurarea principalelor tensiuni prezente într-un sistem de calcul (V<sub>core</sub>, +3.3V, +5V, +12V, -5V, -12V). Alimentarea senzorilor se face de pe placa de bază. Interfața și protocolul de comunicație poate să difere de la un producător la altul. În general protocolul de comunicație este unul serial compatibil I<sup>2</sup>C, iar în ceea ce privește interfața ea poate fi o magistrală ISA sau SMBus. În sistemele mai vechi interfața se realizează pe magistrala ISA, iar în sistemele mai noi pe SMBus.

## 4.2 Metode de conectare a senzorilor

Senzorii de monitorizare prezenți într-un calculator (tensiuni de alimentare, temperaturi și ventilatoare) trebuie să poată fi accesați de către aplicațiile utilizator pentru a se putea lua anumite decizii pe baza informațiilor furnizate de ei. Pentru ca senzorii să poată fi accesați de către procesor trebuie să existe un modul de interfață cu rol de conversie numerică a informației (dacă această interfață nu este prevăzută în senzor) și fiecare senzor trebuie să aibă alocată o adresă unică în spațiul adreselor de porturi. Având în vedere că numărul de senzori (interni sau externi) poate fi oricât de mare, alocarea unei adrese din spațiul de adresare al procesorului pentru fiecare senzor este neeconomică deoarece spațiile de adrese al procesorului este limitat și există foarte multe zone de adrese rezervate pentru diferite tipuri de dispozitive. Pe de altă parte, pentru a implementa fiecare senzor cu o facilități de autodetecție (Plug-and-Play) ar fi necesară o creștere a complexității interfeței senzorilor, ceea ce din nou ar fi neeconomic din punctul de vedere al costului.

Soluția pentru problemele de conectare a senzorilor, care este și implementată în majoritatea sistemelor de calcul actuale, este utilizarea unei magistrale de tip serial cu una sau două linii de date. Pe placa de bază a PC-urilor este implementată o astfel de magistrală, numită SMBus (System Management Bus) la care sunt conectați senzorii de monitorizare prezenți în sistem.

### 4.2.1 Magistrala SMBus

Magistrala de management a sistemului (SMBus) este o interfață de două fire prin care pot comunica diversele circuite componente ale unui sistem. Această magistrală se bazează pe principiile de funcționare a magistralei I<sup>2</sup>C. SMBus este o magistrală de control pentru managementul sistemului și a puterii consumate. Aceasta se poate folosi pentru transmiterea de mesaje înspre și dinspre echipamente înlocuind liniile de control individuale, prin eliminarea liniilor de control reducându-se numărul de pini.

Prin magistrala de control a sistemului, un echipament poate pune la dispoziție informații de la producător, poate comunica sistemului propriul număr de model/componentă, să-și salveze starea pentru un eveniment special, să raporteze diverse tipuri de erori, să accepte parametrii de control, să returneze starea proprie, etc. La un segment SMBus pot fi conectate două tipuri de dispozitive: dispozitive master și dispozitive slave. În general un dispozitiv master inițiază un transfer între el și un slave și pune la dispoziție semnalul de sincronizare (clock). Singura excepție de la această regulă este în timpul instalării inițiale, când un singur master poate iniția tranzacții simultan cu mai multe dispozitive slave. Un dispozitiv slave poate primi date de la un master sau poate trimite date către un master. La un moment dat doar un singur dispozitiv SMBus poate fi master pe magistrală. Dat fiind faptul că mai multe dispozitive încearcă să obțină controlul magistralei ca și master, protocolul SMBus pune la dispoziție un mecanism de arbitraj care se bazează pe conexiunea ȘI-cablat a dispozitivelor interfațate la SMBus.

Datorită diversității dispozitivelor ce pot fi conectate la magistrala SMBus, standardul permite surse de alimentare și tensiuni de alimentare diferite pentru fiecare dispozitiv (figura 4.1).  $V_{DD}$  poate fi de la 3 la 5 volți +/- 10% și pot exista dispozitive alimentate direct de la magistrală  $V_{Bus}$ . Ambele linii SMBCLK și SMBDAT sunt bidirecționale, conectate la o sursă de tensiune pozitivă printr-o rezistență de limitare, de la o sursă de curent sau alt circuit similar. Când magistrala este liberă, ambele linii sunt pe nivelul superior. Etajele de ieșire ale dispozitivelor conectate la magistrală trebuie să fie cu colectorul în gol sau drenă deschisă pentru a se putea realiza conexiunea de ȘI cablat.

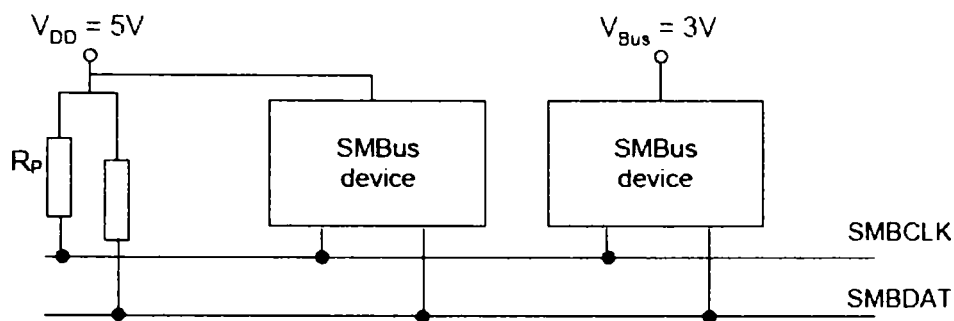


Figura 4.1 – Topologia magistralei SMBus [SMBus2000]

Comunicația pe magistrala SMBus este structurată pe trei nivele, peste care se pot implementa nivelurile specifice aplicației:

- nivelul fizic – stabilește caracteristicile electrice pentru dispozitivele SMBus, timpii protocolului, specificații pentru diferite niveluri de consum ale dispozitivelor, ș.a.;
- nivelul de date – stabilește nivelurile logice pentru “1” și “0”, protocolul de transfer de date, sincronizarea, arbitrarea, formatul datelor;

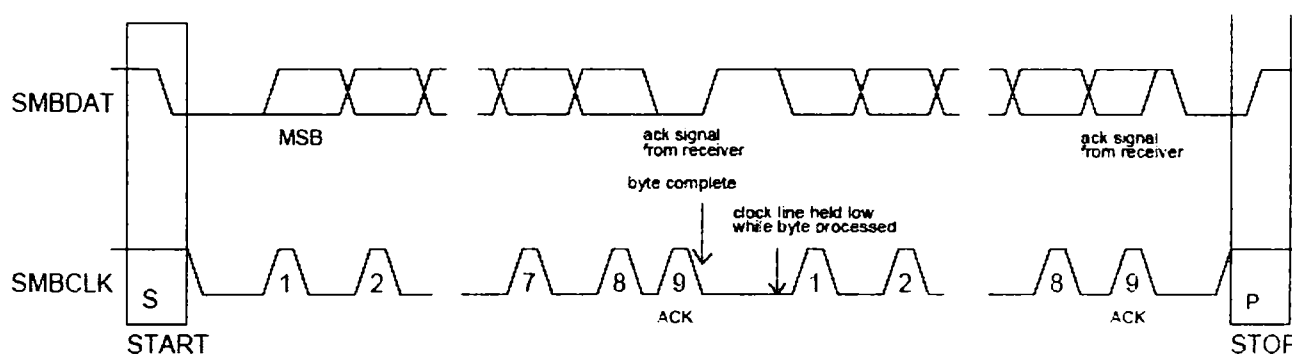


Figura 4.2 – Transferul unui octet pe SMBus [SMBus2000]

- nivelul rețea – stabilește modul de identificare și adresare a dispozitivelor, tipurile de adrese, detecție și corecție de erori prin coduri CRC, protocolul de rețea al magistralei, ș.a. Protocolul de rețea al magistralei SMBus este orientat pe mesaje, existând posibilitate de comunicare pe octet, pe cuvânt și pe bloc. În figura 4.3 sunt prezentate două mesaje de transfer pe un octet, primul de scriere și al doilea de citire. Mesajul este format dintr-un bit de start (S), adresa dispozitivului slave, un bit de descriere a operației (Wr – scriere sau Rd – citire), octetul de date și bitul de stop (P). După comandă și date este necesar un raspuns de ACK de la partener (A).

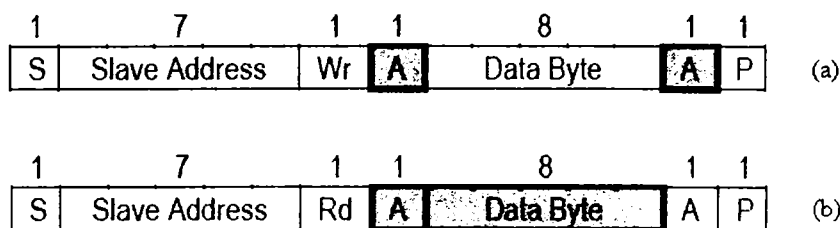


Figura 4.3 – Exemple de mesaje SMBus: (a) Scriere 1 octet; (b) Citire 1 octet [SMBus2000]

Dezvoltarea aplicațiilor orientate pe SMBus ar lua destul de multe resurse dacă nu ar exista suport specific implementat deja. La nivelul chipset-ului plăcii de bază există implementată interfața pentru magistrala SMBus, ce are alocate anumite adrese PCI, ce pot fi utilizate într-o aplicație utilizator pentru scanarea SMBus și implementarea dialogului cu un dispozitiv SMBus. Interfața SMBus este de obicei implementată în controller-ul integrat pentru periferie (PCI Super I/O Integrated Peripheral Controller sau South Bridge). Interfața SMBus are o adresă de bază în spațiul adreselor I/O și are câteva registre ce folosesc la comunicarea cu un dispozitiv SMBus.

Un transfer pe magistrala SMBus folosind interfața oferită de chipset-ul plăcii de bază presupune următorii pași, implementați în aplicația ce va fi descrisă ceva mai târziu:

- se așteaptă eliberarea magistralei (terminarea transferului curent, dacă există);
- se șterg registrele de date la o citire sau se scrie data de trimis în registrele de date pentru o scriere;
- se scrie adresa dispozitivului SMBus slave în registrul de adrese;
- se scrie comanda în registrul de comandă (comanda poate fi citire sau scriere, pe octet, cuvânt sau bloc);
- se lansează în execuție transferul, tot printr-o scriere în registrul de control al interfeței SMBus;
- se așteaptă terminarea transferului;
- în cazul unei citiri se preiau datele citite din registrele de date.

## 4.2.2 Magistrala 1-Wire Bus

O a doua magistrală des utilizată pentru conectarea senzorilor externi este magistrala 1-Wire Bus propusă de firma Dallas Semiconductor. Magistrala 1-Wire Bus folosește un dispozitiv master pentru a controla celalalte dispozitive de tip slave (unul sau mai multe). Fiecare dispozitiv (master sau slave) este conectat la linia de date printr-o poartă cu trei stări sau cu colector în gol. Acest lucru asigură posibilitatea “eliberării” liniei atunci cand dispozitivul nu comunică, lasând linia liberă pentru celalalte dispozitive conectate. Toate datele și comenzile sunt transmise începând cu cel mai puțin semnificativ bit. Dacă se dorește ca o secvență de comunicare să fie întreruptă pentru o perioadă de timp, atunci linia de date trebuie lăsată inactivă pentru a putea relua secvența la un moment dat. Dacă linia este lasată inactivă (1 logic), între biții tranzacționați pot apărea pauze de timp infinite. Iar, dacă linia stă pe 0 logic mai mult de 480  $\mu$ s, toate dispozitivele vor fi resetate.

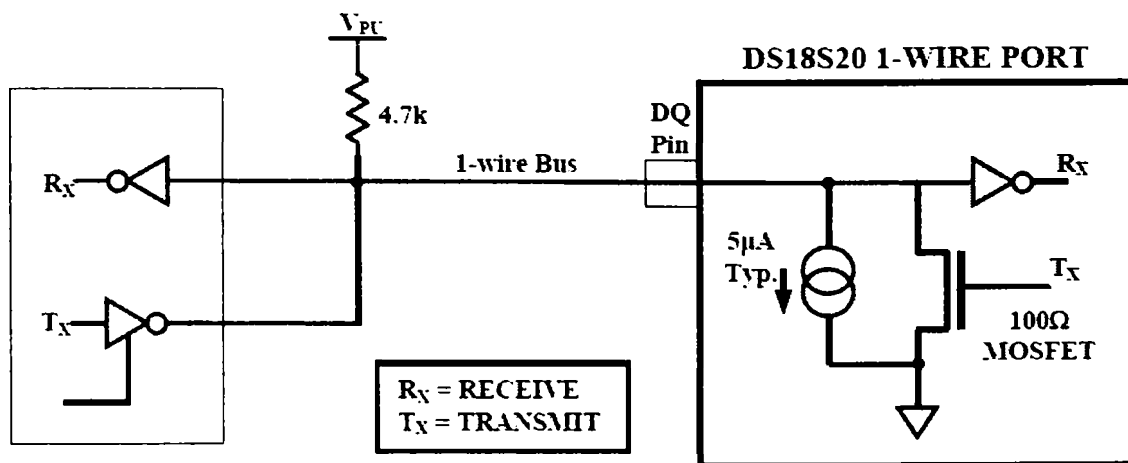


Figura 4.4 – Configurația unui dispozitiv 1-Wire Bus [1WireBus]



Secvența de comunicare cu un dispozitiv 1-Wire Bus se stabilește în câțiva pași:

- inițializarea comunicației - toate tranzacțiile de date încep cu secvența de inițializare. Aceasta constă dintr-un impuls de reset transmis de dispozitivul master, urmat apoi de un impuls de prezență transmis de dispozitivele slave. Dispozitivul master își poate da seama dacă există alte dispozitive slave conectate cu ajutorul acestui impuls de prezență;
- adresarea dispozitivului slave – după sesizarea de către master a impulsului de prezență, acesta poate selecta un dispozitiv pe baza unui cod unic pe 64 de biți;
- comanda dispozitivului – după selecția dispozitivului masterul poate trimite o comandă specifică dispozitivului: pornire conversie, citire/scriere date, setare alarme, etc.

Orice transfer între master și dispozitivele slave începe cu secvența de inițializare, care constă dintr-un impuls de inițializare trimis de master, urmat apoi de impulsul de prezență trimis de dispozitivele slave (figura 4.5). Când dispozitivul slave trimite răspunsul de prezență, înseamnă că este prezent și pregătit pentru operare. În timpul secvenței de inițializare dispozitivul master transmite impulsul de inițializare ținând linia pe "0" timp de cel puțin 480  $\mu$ s. Apoi master-ul intră în modul de recepție, lăsând linia liberă, adică pe "1". După ce dispozitivul slave detectează frontul crescător, așteaptă 15-60  $\mu$ s, după care transmite impulsul de prezență, ținând linia pe "0" timp de 60-240  $\mu$ s.

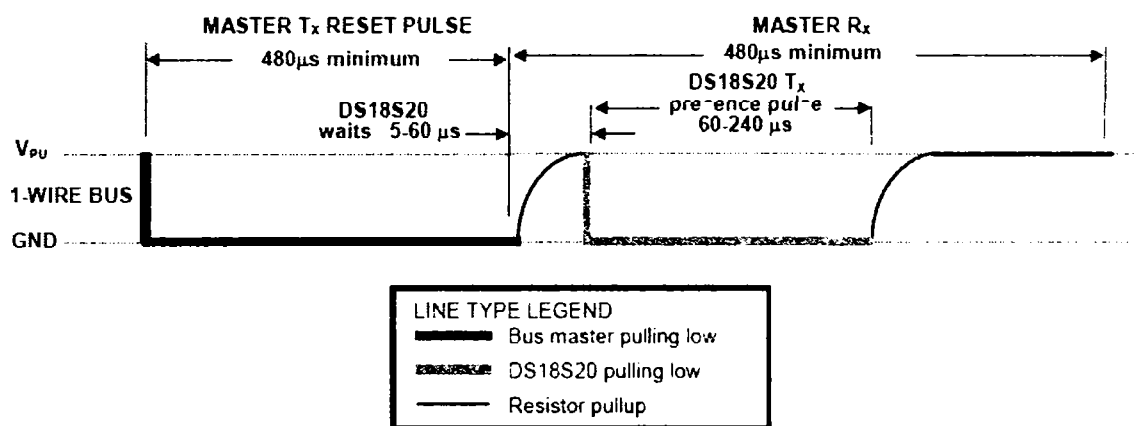


Figura 4.5 – Inițializarea comunicației [1WireBus]

Dispozitivul master poate scrie sau citi date pe baza unor operații specifice descrise la nivelul unui singur bit de informație (figura 4.6). Prin înlanțuirea lor se poate tranzacționa un octet.

Există două tipuri de semnale de scriere (inițiate de dispozitivul master): "write 1" (scrie 1 logic) și "write 0" (scrie 0 logic). Fiecare semnal de scriere trebuie să aiba o durată minimă de 60  $\mu$ s, iar între acestea trebuie să existe minimum 1  $\mu$ s pauză. Ambele tipuri de semnale de scriere sunt inițiate de master prin aducerea la 0 logic a liniei. Pentru a genera un semnal de tipul "write 1", după ce master-ul a adus linia pe 0 logic, trebuie ca după 15  $\mu$ s să o lase liberă. Apoi rezistența de 5 k $\Omega$  va aduce linia pe 1 logic. Pentru a genera un semnal de tipul "write 0", după ce master-ul a adus linia pe 0 logic, trebuie să țină în continuare linia pe 0 logic pe durata unui semnal de scriere (în total minimum 60  $\mu$ s). Dispozitivele slave monitorizează linia timp de 15-60  $\mu$ s, după ce master-ul a inițiat semnalul. Dacă linia este pe 1 logic în acea perioadă, se va scrie un 1 logic la slave; iar dacă linia va fi pe 0 logic, atunci se va scrie 0 logic.

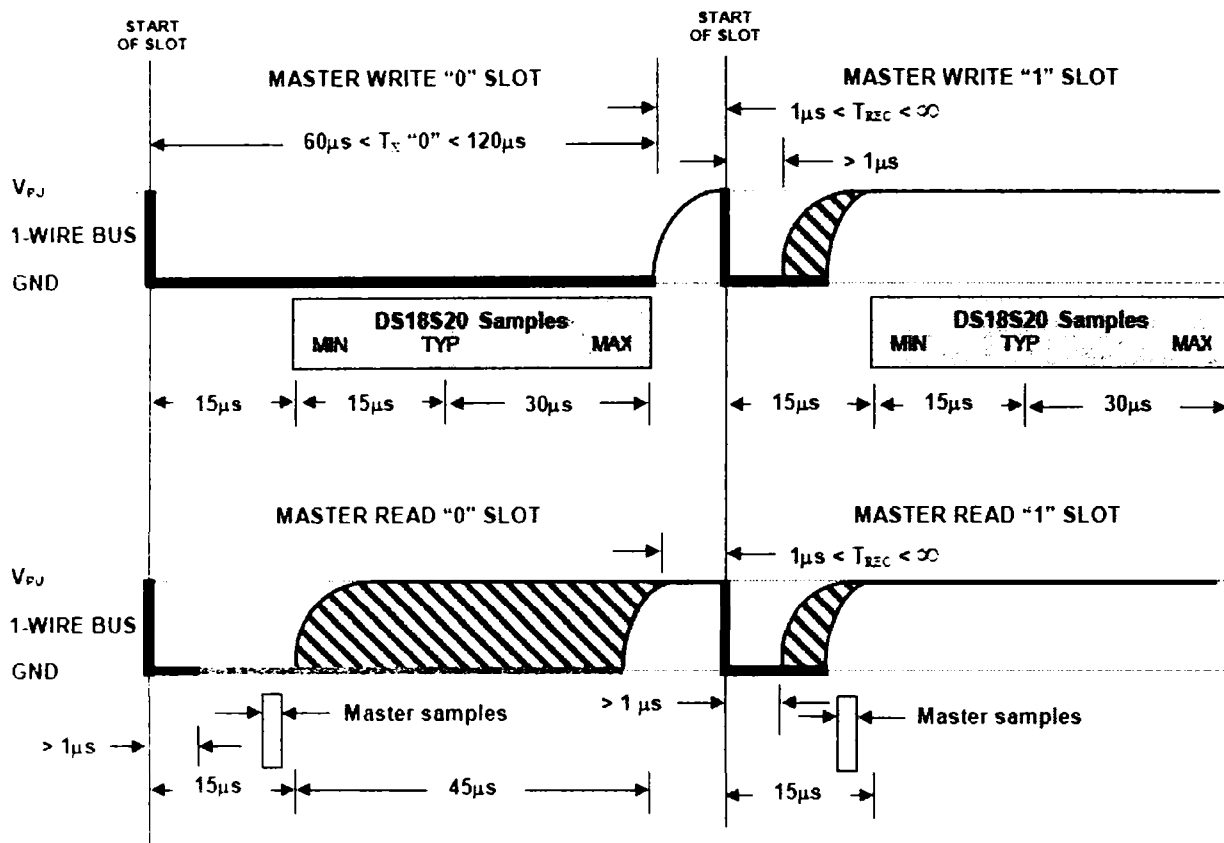


Figura 4.6 – Ciclurile de scriere și citire [1WireBus]

Dispozitivele slave pot trimite date spre master numai atunci când acesta inițiază semnale de citire. De aceea master-ul trebuie să genereze aceste semnale imediat după o comandă, ce are ca scop o citire de date de la un dispozitiv slave. În funcție de comanda transmisă inițial de către master, o citire ce o urmeza poate avea diferite semnificații. Fiecare semnal de citire trebuie să aibă minimum  $60\mu s$ , iar între acestea trebuie să existe minimum  $1\mu s$  pauză. Semnalul de citire este inițiat de master prin aducerea la "0" a liniei pentru o durată de  $1-5\mu s$ , după care acesta va elibera linia. După ce master-ul a inițiat un semnal de citire, dispozitivul slave va transmite un bit de informație (0 sau 1). Dispozitivul slave transmite un 1 logic lăsând linia să revină la 1 logic sau transmite un 0 logic dacă o mai ține pe 0 logic un timp. După ce a transmis un 0 logic, dispozitivul slave va lasă linia, iar rezistența o va aduce pe 1 logic. Bitul de informație trimis de către dispozitivul slave este valid după  $15\mu s$  după primul front căzător (generat de master). De aceea, dispozitivul master trebuie să lase linia și să o monitorizeze (pentru a citi bitul) într-un interval de  $15\mu s$  de la inițierea semnalului de citire. Detalii suplimentare cu privire la duratele acestor semnale se găsesc în documentațiile complete ale fiecarui tip de senzori sau dispozitive Dallas.

## 4.3 Semnătura termică a unui sistem de calcul

Temperatura a devenit unul din cei mai importanți parametri din sistemele electronice digitale. Pe baza acestui parametru se pot obține mai multe informații despre modul de funcționare al dispozitivului respectiv: dispozitivul funcționează în parametrii de temperatură specificați de producător, răcirea dispozitivului este eficientă sau ineficientă, dispozitivul este defect; în plus se pot obține și informații despre puterea disipată de dispozitiv în fiecare moment de timp [Szekely1998]. Din acest motiv este util și important a avea un mecanism simplu și eficient de monitorizare a temperaturii diferitelor părți componente ale unui sistem de calcul.

### Definiția 4.1

Prin semnătură termică a unui sistem de calcul se înțelege răspunsul termic al fiecărei componente a sistemului pentru un anumit mod de utilizare a acestuia.

În continuare în acest capitol voi prezenta două metode de monitorizare a temperaturii unui sistem, prima metodă folosește senzori externi de temperatură iar cea de-a doua metodă folosește senzorii integrați împreună cu anumite componente ale sistemului (procesor, harddisk). Implementarea prin program a unui protocol de comunicație pentru senzorii de temperatură presupune folosirea facilităților de timp real oferite de sistemul de operare în care se implementează aplicația. Din acest motiv s-a studiat posibilitatea de implementare a unui protocol de timp real în sistemul de operare Windows XP și nivelul la care ar trebui să se implementeze segmentele critice din protocol.

### 4.3.1 Implementarea aplicațiilor de timp real sub sistemul de operare Windows XP

Sistemele de operare bazate pe nucleu Windows NT (Windows NT, 2000, XP, 2003) nu au fost proiectate pentru aplicații de timp real. Însă, datorită cerințelor pieței și a largii răspândiri a sistemelor de operare Windows, majoritatea aplicațiilor de monitorizare și control trebuie să folosească aceste sisteme de operare, în ciuda lipsei lor de determinism. Procesele de monitorizare și control în mod normal necesită implementarea unor secvențe de cod critice din punctul de vedere al timpului, însă sistemele de operare Windows nu au fost proiectate pentru a suporta în totalitate acest lucru. Astfel că pentru a putea implementa totuși secvențele de cod critice trebuie să se găsească diferite artificii specifice cerințelor aplicației și bazate pe mecanismele interne ale sistemului de operare.

Sistemul de operare Windows NT a fost proiectat ca un sistem de operare de uz general, astfel că mecanismele lui interne sunt optimizate pentru a satisface majoritatea cerințelor, însă nu toate cazurile posibile, ceea ce intră în conflict cu necesitatea unei bune predictibilități specifice sistemelor de timp real [Ramamritham1998, Landis1996]. Din punct de vedere practic, aplicațiile de timp real se pot clasifica în două clase: stricte (hard real-time) și permissive (soft real-time) [Ragen1995]. Aplicațiile de timp real stricte trebuie să ofere un timp de răspuns prestabilit pentru un anumit eveniment independent de celelalte activități ce sunt în execuție la un moment dat. Pe de altă parte, aplicațiile de timp real permissive au un grad mai redus al constrângerilor pentru timpul de răspuns, adică există un anumit interval de valori în care se poate încadra. Protocele de comunicație pentru senzorii de temperatură ce au fost prezentate se încadrează în categoria aplicațiilor de timp real permissive, astfel că pot fi implementate sub sistemul de operare Windows.

Pentru a implementa în program un protocol de timp real, sistemul de operare trebuie să ofere un mecanism de temporizare cu o precizie înaltă și un număr de niveluri de priorități pentru execuția codului [Briskin1998]. Sistemul de operare Windows NT și cele bazate pe el, oferă două niveluri de privilegii hardware: modul utilizator – în care sunt executate aplicațiile utilizator și modul nucleu – în care sunt executate modulele sistemului de operare și driverele [Solomon2000, Oney1999, Baker1997]. În fiecare din cele două moduri de lucru (utilizator sau nucleu), aplicațiile au la dispoziție 32 de niveluri de priorități: nivelul de execuție al proceselor și firelor de execuție pentru aplicațiile din spațiul utilizator și IRQ (Interrupt Request Level) pentru drivere. În ceea ce privește funcțiile de contorizare a timpului, sistemul de operare Windows oferă mai multe opțiuni pentru fiecare din cele două niveluri de execuție. Aceste funcții vor fi prezentate și evaluate comparativ în continuare.

În literatură există destul de multe lucrări ce tratează mecanismele puse la dispoziție de sistemele de operare Windows pentru implementarea aplicațiilor de timp real. Majoritatea din ele tratează latența [Ramamritham1998, Landis1996], durata de execuție a apelurilor [INTEL2002, NVIDIA2001], utilizarea priorităților, metode de implementare a sincronizărilor, etc.; iar altele sunt axate pe anumite categorii de aplicații. În continuare voi prezenta problemele specifice unui protocol de comunicație și o comparație între implementarea unei secvențe de cod într-o aplicație din spațiul utilizator sau implementarea ei într-un driver din spațiul nucleu. Testele au fost realizate sub sistemele de operare Windows XP și 2000 însă ele sunt valabile și pentru celelalte sisteme de operare bazate pe nucleul Windows NT.

Una din problemele ce apar la un protocol de comunicație este introducerea de perioade de așteptare cu o rezoluție mare (de ordinul microsecundelor). Funcții de așteptare cu o rezoluție de peste 1 ms sunt oferite de Windows SDK (funcția Sleep ce primește ca parametru numărul de milisecunde cu care trebuie întârziată execuția unui fir de execuție). Pentru întârzieri de ordinul microsecundelor sistemul de operare nu oferă nici o funcție, astfel că ele trebuie implementate de către cel care dezvoltă aplicația.

Pentru implementarea unei funcții de întârziere de rezoluție înaltă este necesară utilizarea unui temporizator de înaltă rezoluție și trebuie avută în vedere obținerea unei încărcări cât mai reduse a procesorului, încărcare care are în vedere atât numărul de operații executate cât și reducerea temperaturii procesorului. Aceste funcții de întârziere sunt necesare și se pot implementa atât la nivelul aplicațiilor utilizator cât și în drivere, la nivelul nucleu.

### **Aplicații utilizator**

La nivelul programelor executate în spațiul utilizator există patru funcții ce pot fi utilizate la implementarea unei temporizări cu o înaltă rezoluție, de ordinul microsecundelor, pe lângă cele ce au o rezoluție mai mică, de ordinul milisecundelor. Trei din cele patru funcții sunt oferite de sistemul de operare (GetTickCount, GetSystemTimeAsFileTime și QueryPerformanceCounter) iar a patra variantă se utilizează instrucțiunea RDTSC implementată în în setul de instrucțiuni IA32 începând cu procesoarele Pentium. Începând cu procesoarele Pentium este implementat un contor pe 64 de biți (numit Time Stamp Counter - TSC) ce se incrementează la fiecare impuls al tactului procesorului. Instrucțiunea RDTSC încarcă în registrele EAX și EDX valoarea curentă a contorului, astfel încât poate fi utilizată în aplicații. Utilizând acest mecanism bazat pe RDTSC și cunoscând frecvența semnalului de tact se pot obține temporizări foarte exacte și de rezoluție foarte mare. Funcția QueryPerformanceCounter oferită de sistemul de operare are la bază tot instrucțiunea RDTSC.

Dacă se implementează câte o buclă de așteptare folosind fiecare din cele patru funcții și se măsoară timpul scurs pentru fiecare iterație se poate obține o estimare a rezoluției funcției prin niște valori maxime, minime și medii. În figura 4.7 sunt prezentate comparativ

rezultatele pentru trei din cele patru funcții, `GetSystemTimeAsFileTime` având același comportament cu `GetTickCount`. `GetSystemTimeAsFileTime` are o acuratețe de 100 ns însă se actualizează o dată cu întreruperea de timp real a sistemului (ce apare de 64 ori pe secundă). `GetTickCount` folosește tot numărul de impulsuri de tact de la pornirea sistemului, însă valoarea nu este citită direct din registrul TSC al procesorului ci dintr-o variabilă implementată în sistemul de operare, variabilă ce este actualizată tot de către ceasul sistemului. În figura 4.7 se poate observa acest comportament datorită faptului că reprezentarea nu este una liniară ci una treaptă, valoarea este actualizată tot la 15 ms, astfel că funcția aceasta nu poate fi utilizată pentru a obține temporizări mai mici de 15 ms. Aceste funcții au însă avantajul că sunt mai rapide decât celelalte două (deoarece ele trebuie să acceseze registrul TSC al procesorului), introducând o latență de aproximativ 0.01  $\mu$ s.

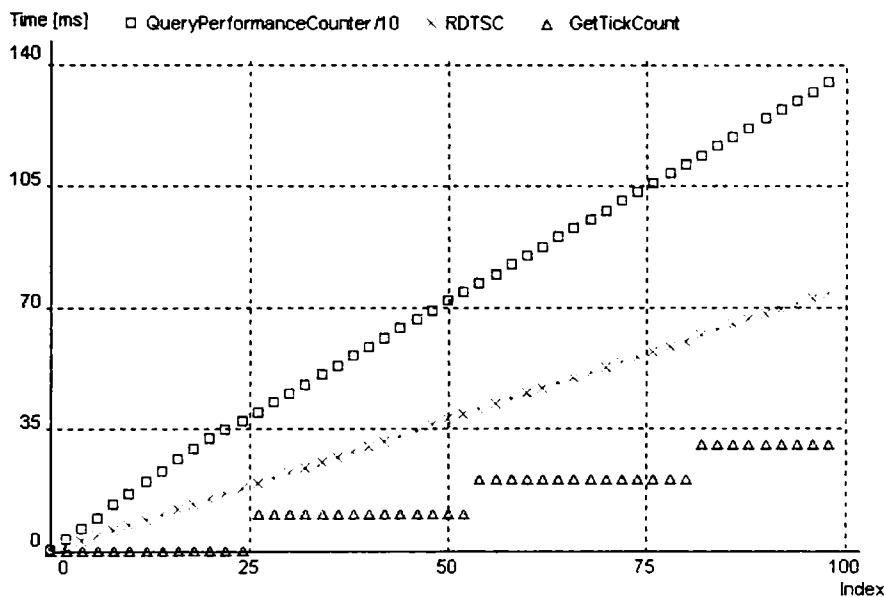


Figura 4.7 – Rezultate pentru funcțiile de temporizare din modul utilizator

Funcția `QueryPerformanceCounter` are o latență medie de aproximativ 1.3  $\mu$ s. Chiar dacă această funcție utilizează la bază tot RDTSC, ea este mai lentă pentru că este proiectată să poată fi utilizată și în sistemele multiprocesor. Rezoluția unei metode de întârziere bazate pe `QueryPerformanceCounter` poate fi de câteva  $\mu$ s. În cazul sistemelor uniprocessor se poate utiliza direct instrucțiunea RDTSC obținându-se o creștere a performanței de aproximativ 40 de ori, metoda bazată pe RDTSC introducând o latență de aproximativ 0.03  $\mu$ s.

```

__forceinline LARGE_INTEGER GetPentiumCounter(void)
{
    __asm
    {
        xor    eax,eax
        xor    edx,edx
        _emit 0x0F    // RDTSC instruction code
        _emit 0x31    // TSC returned to edx:eax
    }
}

```

Orice aplicație utilizator are alocat un nivel de prioritate din cele 32 de niveluri posibile. Pe baza acestor priorități sistemul de operare alocă timp procesor fiecărui fir de execuție. Procesele se încadrează într-o clasă de priorități între `IDLE_PRIORITY_CLASS` și `REALTIME_PRIORITY_CLASS`, procesele normale având prioritatea `NORMAL_PRIORITY_CLASS`. Firele de execuție au și ele alocată câte o prioritate între

THREAD\_PRIORITY\_IDLE și THREAD\_PRIORITY\_TIME\_CRITICAL. Pe baza calsei de proiritate a procesului și a priorității firului de execuție se stabilește prioritatea curentă a firului de execuție, cu o valoare între 0 și 31. În mod uzual aplicațiile au un nivel normal de prioritate și partajează același microprocesor, astfel că este posibil ca buclele de întârziere să fie întrerupte de sistemul de operare pentru a aloca timp procesor unui alt proces. Din acest motiv se va ajunge ca două apeluri succesive ale funcțiilor de timp de înaltă rezoluție să poată depăși cu mult valoarea latenței funcției, fapt ce poate afecta acuratețea metodei de măsurare a timpului. În figura 4.8 este prezentată grafic relația între modul de execuție al aceleași bucle de întârziere în contexte de încărcare diferită a procesorului. Este prezentată bucla de întârziere implementată cu funcția GetPentiumCounter, bazată pe RDTSC. În figura 4.8 se pot observa discontinuități în reprezentările grafice ale iterațiilor pentru încărcări mai mari ale procesorului. Aceste discontinuități pun în evidență perioadele de timp în care procesorul este alocat unui alt proces. Dacă aceste intervale de timp apar într-o funcție de întârziere ce trebuie să aibă o acuratețe foarte bună, se poate depăși intervalul de timp pentru care a fost prevăzută întârzierea și este afectată acuratețea funcției.

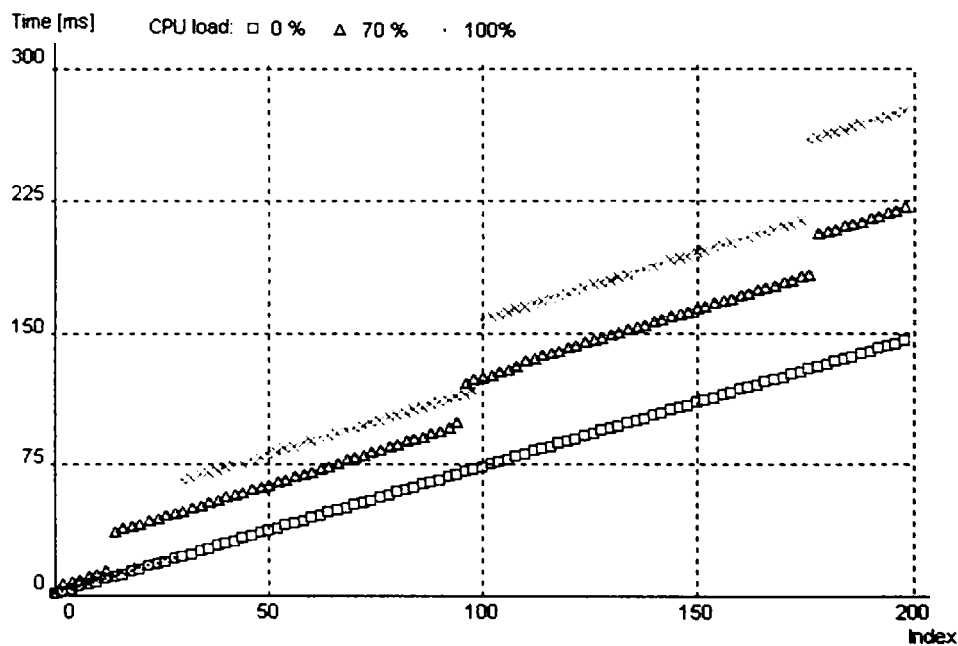


Figura 4.8 – Modul în care încărcarea procesorului influențează mecanismul de întârziere

Acest aspect al modului în care gradul de încărcare al procesorului influențează acuratețea funcțiilor de întârziere este accentuat și de prioritățile diferite ce le pot avea procesele. Graficul din figura 4.8 a fost obținut pentru procese ce aveau toate aceeași prioritate (NORMAL\_PRIORITY\_CLASS). Dacă există la un moment dat în execuție o aplicație ce are un nivel de prioritate mai mare decât aplicația în care este implementată o buclă de întârziere, se poate ca bucla de întârziere să nu se termine până când nu se execută aplicația cu nivelul mai mare de prioritate, afectând și mai grav acuratețea funcției de întârziere. Soluția ce poate fi aplicată pentru păstrarea acurateții funcțiilor de întârziere de mare precizie este ridicarea nivelului de prioritate pe durata funcției de întârziere. Se poate ridica nivelul de prioritate până la valoarea maximă pentru situațiile critice.

În figura 4.9 s-a prezentat aceeași funcție de întârziere rulată la o încărcare a procesorului de 20%, în care s-a modificat nivelul de prioritate la care este executat codul de întârziere, de la IDLE\_PRIORITY\_CLASS la NORMAL\_PRIORITY\_CLASS și până la REALTIME\_PRIORITY\_CLASS. Dacă funcția este executată la un nivel mic de privilegiu ea este mai des întreruptă de sistemul de operare în timp ce la nivelul maxim funcția de întârziere nu este întreruptă până nu își încheie activitatea. Din acest motiv funcțiile de

întârziere de rezoluție și acuratețe înaltă vor trebui să fie executate la un nivel cât mai mare de prioritate. Nivelul de execuție cel mai înalt trebuie utilizat cu atenție deoarece pe durata cât un proces este setat la acest nivel de prioritate (REALTIME\_PRIORITY\_CLASS) nici un alt proces nu va mai primi timp procesor până când procesul de nivel înalt nu își încheie activitatea. Din acest motiv întârzierile de o durată mai mare și apelate repetat pot duce la blocări ale sistemului observabile de utilizator, inclusiv mișcarea mouse-ului.

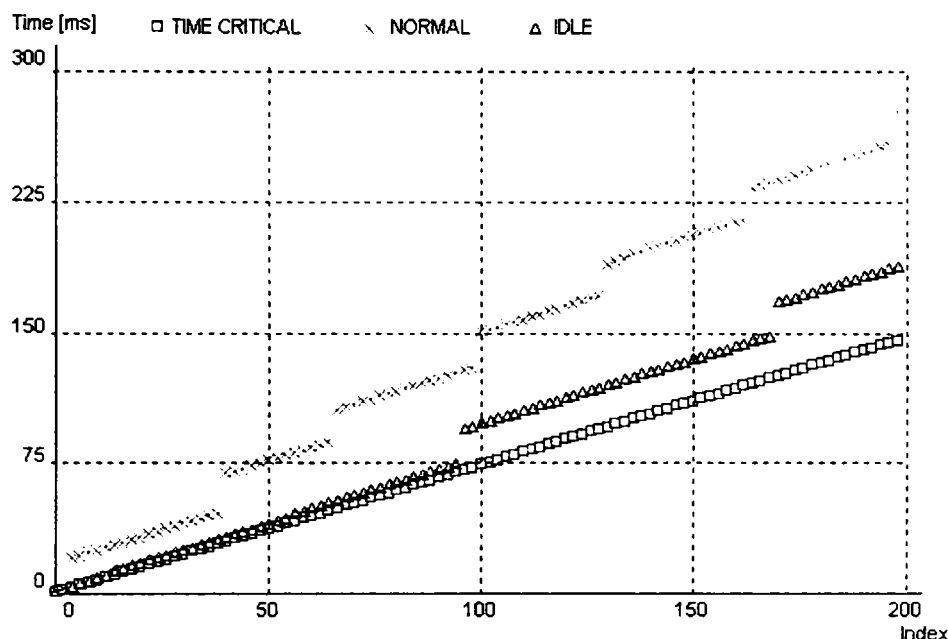


Figura 4.9 – Modul în care nivelul de prioritate afectează mecanismul de întârziere

Cel de-al doilea aspect ce trebuie avut în vedere la implementarea unei funcții de întârziere este încărcarea procesorului. Într-o funcție de întârziere nu se face nici o activitate utilă în afară de cea de așteptare, astfel că procesorul nu ar trebui să fie ocupat ci ar trebui să fie lăsat liber pentru execuția altor procese. Pentru a face acest lucru se poate reduce nivelul de prioritate al firului de execuție la IDLE\_PRIORITY\_CLASS sau se poate cere sistemului de operare (printr-un apel de funcție sistem) să aloce o cantă de timp procesor pentru alte procese ce există în coada de alocare în execuție. Acest lucru afectează însă, așa cum s-a văzut anterior, acuratețea funcției de întârziere. Varianta cea mai elegantă ar fi fost utilizarea de evenimente, care să apară automat și să deblocheze aplicația din așteptare (WaitForSingleObject) după intervalul de timp dorit. Acest lucru este posibil însă doar pentru rezoluții de timp mai mari de 1 ms.

Studiind diagramele de timp ale protocolului ce trebuie implementat se pot stabili câteva tipuri de întârzieri, pentru care se pot implementa mecanisme diferite de așteptare. În cazul protocolului 1-Wire Bus s-au identificat trei clase de întârziere:

- intervale de timp limitate și scurte ca durată – [15-60 $\mu$ s] din semnalul de inițializare;
- intervale de timp limitate și medii ca durată – [60-240 $\mu$ s] din semnalul de detecție a prezenței;
- intervale de timp nelimitate – [1 $\mu$ s- $\infty$ ) din operațiile de citire/scriere.

Intervalele de așteptare scurte ca durată (zeci de microsecunde) se pot implementa utilizând funcțiile GetPentiumCount (pe sisteme uniprocessor) și QueryPerformanceCounter (pe sisteme multiprocessor) la un nivel de prioritate de timp real, REALTIME\_PRIORITY\_CLASS. Intervalele de așteptare nelimitate ca durată se pot implementa în două moduri: dacă timpul minim de așteptare este de ordinul milisecundelor se va utiliza funcția sistem Sleep, dacă însă timpul minim de așteptare este sub 1 ms, pentru a crește performanța protocolului se poate implementa o funcție de așteptare utilizând aceleași

funcții `GetPentiumCount` și `QueryPerformanceCounter` la un nivel de prioritate redus. Implementarea intervalelor de așteptare medii ca durată sunt puțin mai dificil de implementat deoarece pot introduce încărcări mari ale procesorului și blocări observabile de către utilizator dacă sunt implementate la un nivel mare de prioritate, iar dacă se implementează la un nivel mic de prioritate atunci acuratețea este diminuată. La acest nivel se poate implementa o metodă dinamică de așteptare la un nivel de prioritate între normal și timp real (de exemplu `HIGH_PRIORITY_CLASS`). La începutul intervalului de așteptare se permite cedarea procesorului și altui proces, explicit prin apelul unei funcții sistem, însă pe măsură ce intervalul de așteptare se apropie de final nu se mai permite acest lucru.

### Driveri nucleu

Protocolul de comunicație cu senzorii de temperatura se poate implementa și ca un driver nucleu. Un driver dispozitiv este format dintr-o colecție de proceduri apelate de managerul de intrare/ieșire într-un mod dependent de evenimentele apărute. Managerul de I/O este un modul al executivului sistemului de operare al cărui rol este acela de a transforma cererile de intrare/ieșire din aplicațiile utilizator în secvențe corespunzătoare de apeluri ale procedurilor din drivere. Accesul la porturile de intrare/ieșire din aplicațiile utilizator se realizează prin intermediul subsistemului Win32 (figura 4.10). Apelul ajunge la nivelul managerului de I/O și mai apoi este trimis de acesta driver-ului. Driverul cunoaște particularitățile dispozitivului fizic, protocolul de comunicație cu acesta astfel că aici se poate implementa toată logica pentru un anumit dispozitiv.

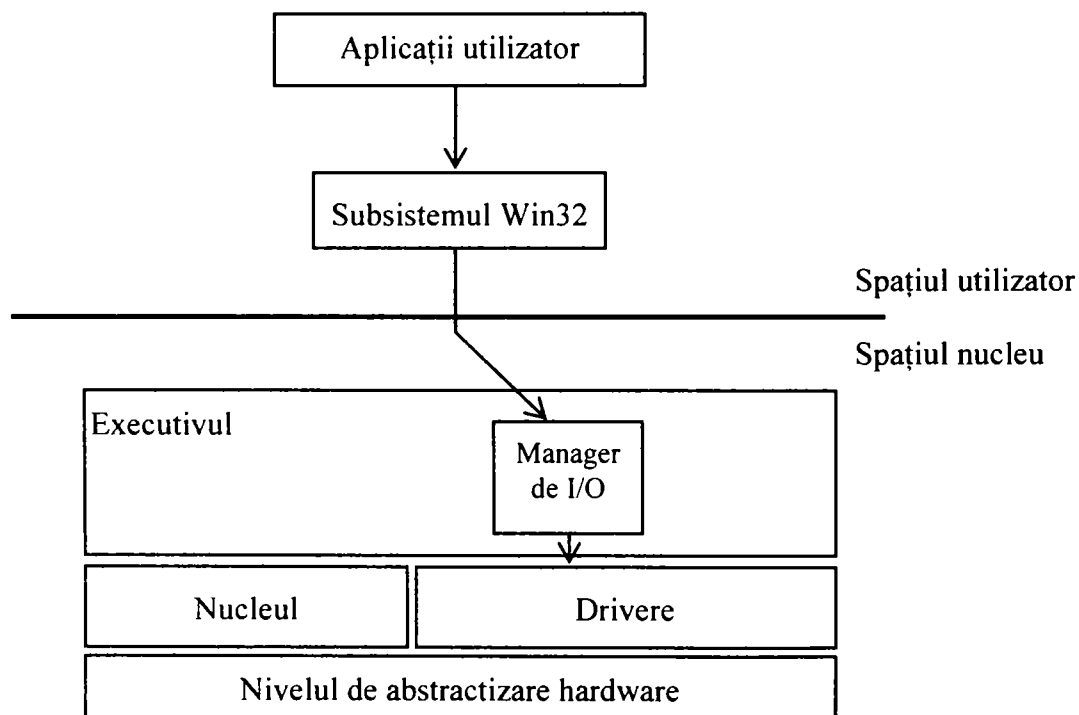


Figura 4.10 – Arhitectura sistemului de intrare/ieșire

La fel ca într-o aplicație utilizator și la nivelul driver-elor există mai multe modalități de temporizare. Scrierea codului într-un driver pe lângă posibilitatea accesului direct la dispozitivul fizic are și avantajul de a fi mai rapid. Echivalentul la nivelul nucleului pentru funcția de citire a contorului TSC din procesor este `KeQueryPerformanceCounter`. În figura 4.11 este prezentată diferența de performanță între același cod implementat într-o aplicație utilizator folosind `QueryPerformanceCounter` și un driver `KeQueryPerformanceCounter`. Implementarea codului într-un driver este de 4-5 ori mai rapidă decât același cod din spațiul utilizator. Latența funcției `KeQueryPerformanceCounter` este de aproximativ 0.25  $\mu$ s. La



încărcări diferite ale procesorului, comportamentul funcției de întârziere este același, fără întreruperi din partea sistemului de operare. Utilizarea repetată a acestei funcții duce la degradarea performanțelor sistemului.

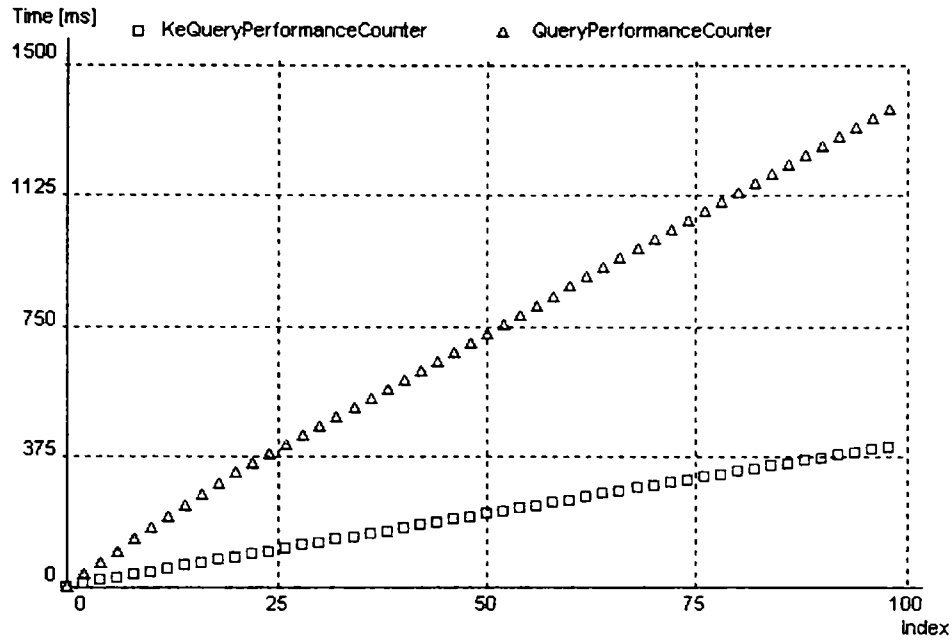


Figura 4.11 – Diferența între latența unei funcții din spațiul utilizator față de cea din spațiul nucleu

O a doua funcție ce poate fi utilizată pentru introducerea de întârzieri este KeQuerySystemTime. Chiar dacă funcția are o acuratețe de 100 ns ea folosește același mecanism ca al funcției GetSystemTimeAsFileTime, astfel că nu poate fi folosită pentru întârzieri de sub 10 ms. GetPentiumCounter, bazat pe utilizarea directă al instrucțiunii RDTSC are același comportament cu cel dintr-o aplicație utilizator. Pentru intervale de așteptare limitate și scurte ca durată se mai poate utiliza funcția DDK KeStallExecutionProcessor, ce poate fi utilizată la a introduce întârzieri de câteva zeci de  $\mu$ s (recomandat a fi utilizată până la 50  $\mu$ s).

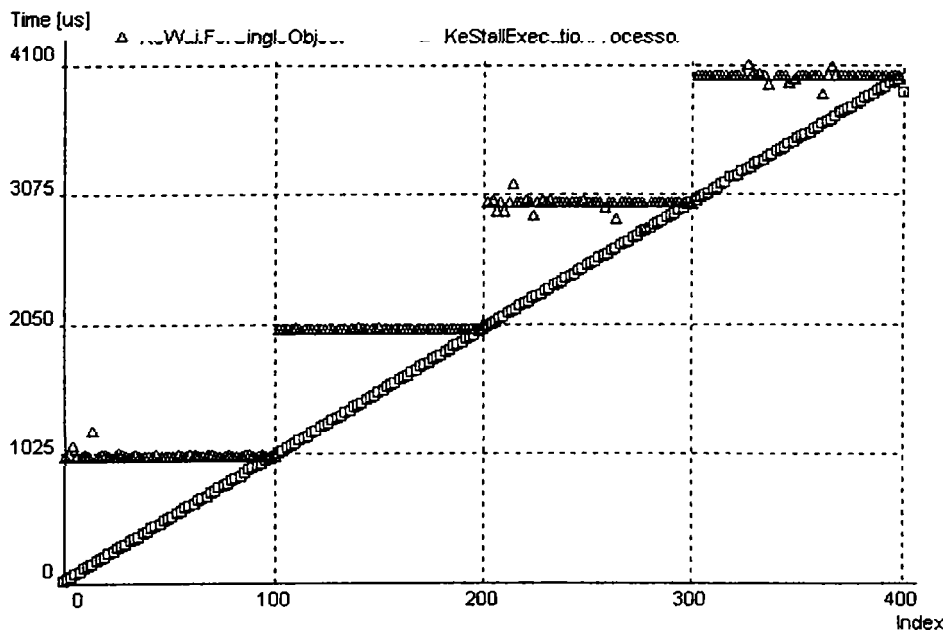


Figura 4.12

La nivelul unui driver există și metode directe de sincronizare. Obiectele de temporizare oferă o modalitate de planificare a execuției unei anumite secvențe de cod. Planificarea poate fi făcută la un anumit timp absolut sau relativ la timpul curent. Obiectele de temporizare sunt utile deoarece nu încarcă suplimentar procesorul. Aceste obiecte se folosesc în temporizate ca parametrii ai funcțiilor: KeSetTimer și KeWaitForSingleObject, însă apare aceeași problemă, aceea a intervalelor de timp de 10 ms. Acest lucru se observă și în graficul din figura 4.12 unde s-a prezentat comparativ comportamentul în timp al funcțiilor KeWaitForSingleObject și KeStallExecutionProcessor.

Un protocol de comunicație pentru un anumit dispozitiv fizic poate fi implementat atât la nivelul aplicațiilor utilizator cât și la nivelul driver-elor nucleu, fiecare din variante având avantajele și dezavantajele ei. În mod uzual codul specific pentru comunicația cu un anumit dispozitiv fizic este implementat într-un driver ce este atașat dispozitivului respectiv deoarece acesta modul în care codul executat are drepturi depline asupra sistemului. Codul executat la acest nivel are acces la instrucțiunile privilegiate, are drepturi de accesare a porturilor de intrare-ieșire, poate adresa orice adrese virtuale de memorie, latența anumitor apeluri de funcții este mai mică, etc. Dezavantajul dezvoltării de aplicații în spațiul nucleu constă în dificultatea cu care se implementează acest tip de aplicații. Setul de funcții oferit de modulele executivului sistemului de operare este complet diferit de cel cunoscut programatorului de aplicații Win32, bibliotecile de funcții sunt diferite de cele din modul utilizator, fluxul de control al unui driver este diferit de cel al unei aplicații utilizator, ș.a. În cazul aplicațiilor utilizator lucrurile stau exact invers, adică dezvoltarea aplicațiilor este relativ ușoară, există medii de dezvoltare bine puse la punct și cunoscute de majoritatea programatorilor; însă controlul asupra resurselor fizice este limitat și supervizat de sistemul de operare, latența funcțiilor este mai mare, apar probleme cu partajarea resurselor (timp procesor) între procese, etc. Soluția recomandată în urma experienței de programare Win32 și drivere presupune împărțirea aplicației în două module, unul executat în spațiul utilizator care va conține toate funcționalitățile ce nu necesită drepturi speciale de execuție și nu au nevoie de timpi critici de execuție, iar celălalt modul este implementat ca driver și va conține toate funcționalitățile ce nu pot fi implementate în spațiul utilizator.

### **4.3.2 Metode de măsurare a temperaturii folosind senzori de temperatură externi**

În multe din situațiile întâlnite este necesar a se utiliza sistemul de operare Windows pentru comunicarea cu diferite dispozitive fizice externe ce nu au o logică internă foarte evoluată și nu folosesc un protocol de comunicație standard (RS232, USB, EPP). Un astfel de exemplu sunt senzorii în general și cei de temperatură în special. Monitorizarea temperaturii (mediului, componentelor unui sistem) poate fi un proces destul de important în diferite tipuri de aplicații. În continuare voi prezenta trei soluții de conectare a senzorilor de temperatură atât în modul utilizator cât și în modul nucleu, utilizând diferitele porturi de comunicație standard ale unui calculator (portul serial și cel paralel).

#### **Protocolul 1-Wire Bus pe interfața serie**

În cazul folosirii interfeței serie pentru conectarea senzorilor de temperatură este necesară transformarea nivelelor de tensiune din TTL în EIA și invers. Pentru conectarea senzorilor la interfața serie există două variante:

- implementarea prin program a protocolului 1-Wire Bus pe liniile controlabile direct din interfața serie (CTS/RTS și DTS/DSR);
- implementarea unui convertor 1-Wire Bus – RS232.

Prima din soluții este prezentată în figura 4.13, în care se folosește linia RTS pentru alimentarea senzorilor și liniile DTR/DSR pentru transmisia și recepția de date. Logica de control a protocolului de comunicație a fost implementată într-un driver care se atașează ca un filtru la driver-ul standard pentru interfața serie, al sistemului de operare Windows.

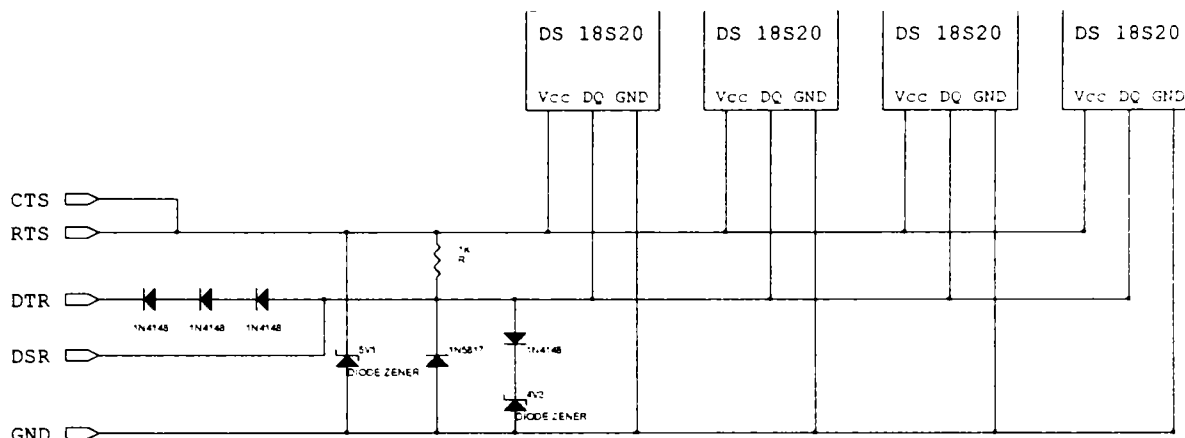


Figura 4.13 – 1-Wire Bus pe interfața serie

Pentru a compara soluțiile de monitorizare a temperaturii s-au făcut un număr de 100 citiri repetate ale senzorilor pentru a obține o valoare a timpului mediu de acces pe senzor. Rezultatele sunt prezentate în tabelul 4.2. Din teste nu se remarcă diferențe sistematice în funcție de numărul senzorilor prezenți și dimensiunea liniilor de comunicație.

Număr senzori	Timp de acces /senzor [ms]	
	25 cm	100 cm
1	611.12	619.25
2	612.63	612.23
3	621.35	622.31
4	618.37	625.76

Tabelu 4.2 – Timpii de monitorizare a senzorilor

Cea de-a doua variantă de implementare a comunicației pe interfața serie presupune conversia protocolului 1-Wire Bus la RS232 (Figura 4.14). Transmisia de date se face pe liniile TXD și RXD folosind protocolul RS232 cu diferiți parametrii de comunicație pentru fiecare din operațiile de pe 1-Wire Bus.

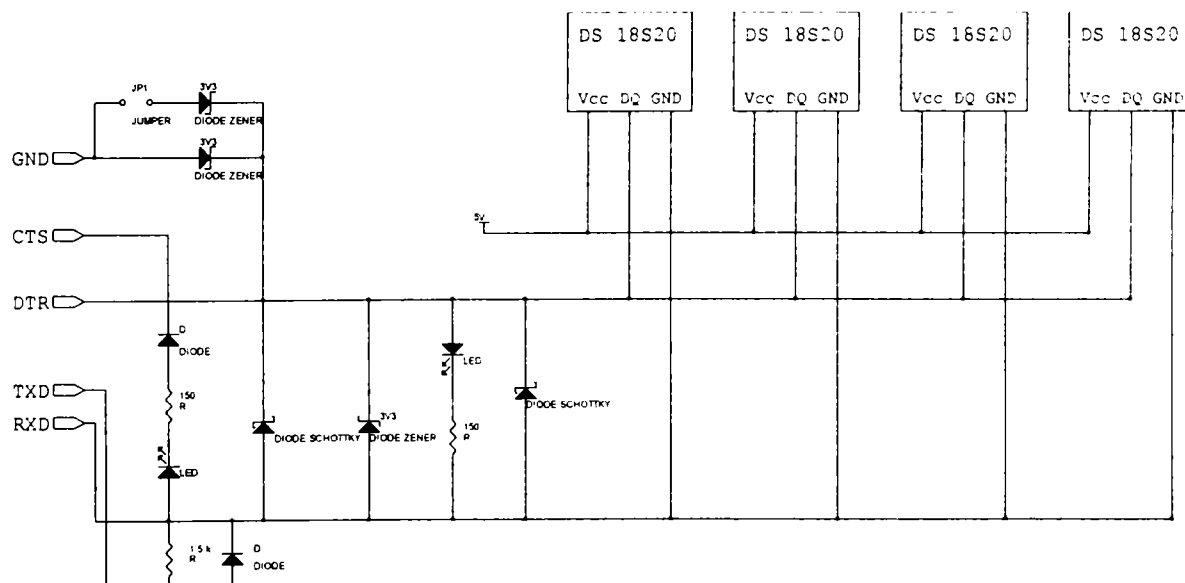


Figura 4.14 – 1-Wire Bus pe RS232

Soluția aceasta folosește driverul standard de Windows pentru interfața serie iar logica specifică de comunicație este implementată la nivelul aplicației utilizator. Se folosesc două seturi de parametrii de comunicație RS232: unul pentru secvența de inițializare de pe 1-Wire Bus (9600 bps, 8 biți de date, fără paritate și 1 bit de stop) și unul pentru secvența de transmisie/recepție a unui bit de date (115000 bps, 6 biți de date, fără paritate și 1 bit de stop).

Același set de teste a fost aplicat și pe schema din figura 4.14. Timpul mediu de citire al unui senzor a fost de 1960 ms, dacă s-a utilizat funcția de așteptare de rezoluție mică (Sleep). Timpul mediu de acces al unui senzor a ajuns la aproximativ 367 ms dacă s-a folosit metoda de întârziere propusă în paragraful precedent.

### Protocolul 1-Wire Bus pe interfața paralelă

Pe interfața paralelă schema de conectare a senzorilor se simplifică deoarece nu mai sunt necesare conversiile între nivelurile de tensiune TTL și EIA (figura 4.15). Alimentarea senzorilor se face de la unul din pinii de control ai interfeței paralele (nSTROBE) iar pentru date se folosește unul din biții de date (D0).

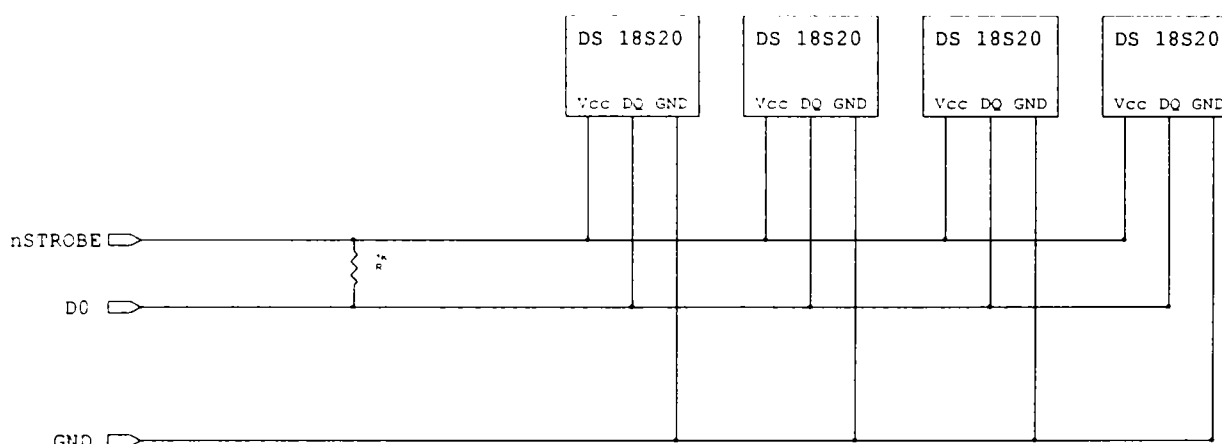


Figura 4.15 – 1-Wire Bus pe interfața paralelă

Protocolul de comunicație este și în acest caz implementat ca un driver, însă se folosește modelul de driver dispozitiv (nu filtru ca în cazul primei variante cu interfața serie). Rezultatele testelor sunt prezentate în tabelul 4.3 obținându-se un timp mediu de acces per senzor este de aproximativ 400 ms.

Numărul senzorilor	Timp de acces /senzor [ms]	
	25 cm	100 cm
1	400.52	409.29
2	392.17	412.68
3	411.33	410.92
4	408.17	410.36

Tabelul 4.3 – Timpii de monitorizare a senzorilor

### Protocolul 1-Wire Bus folosind un microcontroller

Cea de-a treia soluție testată a fost bazată pe un sistem cu microcontroller 80C552. Pentru a nu încărca suplimentar calculatorul se poate folosi o placă externă bazată pe un microcontroller ce va fi dedicat monitorizării senzorilor. Microcontroller-ul va rula codul de monitorizare a senzorilor și cel de comunicație cu calculatorul iar aplicația de monitorizare ce rulează pe calculator va prelua doar valorile măsurate și citite de placa de monitorizare atunci când acestea sunt disponibile.

Metodele precedente se pot folosi în cazul în care numărul de senzori este mic, deoarece monitorizarea fiecărui senzor presupune intervenția procesorului. Pentru un număr

mare de senzori se poate recurge la soluția prezentată în figura 4.16, astfel încât procesorul calculatorului va fi doar întrerupt din activitate doar atunci când există valori disponibile pe placa externă pentru a fi preluate și procesate. Schema completă a acestei soluții este prezentată în anexele 3 și 4.

### 4.3.3 Aplicația de monitorizare a temperaturii componentelor unui sistem

În timpul funcționării unui calculator, în interiorul carcasei, mai ales vara, temperatura aerului poate atinge ușor valori de peste 45°C, datorită tuturor componentelor care degajă căldură. Odată cu apariția de componente mari consumatoare de energie a crescut și căldura disipată de acestea și implicit și cea a sistemului ca întreg. Temperatura prea ridicată diminuează performanța componentelor și este dăunătoare acestora putând duce la deteriorarea ireversibilă a acestora. Din acest motiv componentele ce disipă o cantitate mare de căldură (uzual procesorul și harddisk-ul) sunt prevăzute cu senzori de temperatură ce pot fi monitorizați de către aplicațiile utilizator sau serviciile sistem.

Aplicația a fost concepută pentru monitorizarea proceselor ce rulează la un moment dat și a temperaturii componentelor unui sistem de calcul (PC), oferind informații detaliate despre procese, despre temperatura componentelor prevăzute cu senzori de temperatură și modul în care procesele își aduc aportul la consumul total și temperatura totală a sistemului. Aplicația suportă și un număr nelimitat de senzori externi de tipul 1-Wire Bus conectați la interfața serie.

Un avantaj al acestei aplicații este acela că poate citi informații de pe mai multe plăci de bază, iar montajul extern se poate extinde ușor la mai mulți senzori. Aplicația citește informația de pe magistrala SMBus, fapt care-i conferă avantajul de a fi ușor extinsă pentru mai multe plăci de bază cu un efort mic. Senzorii de pe montajul extern vor fi legați pe aceeași magistrală folosind protocolul 1-Wire Bus dezvoltat de firma Dallas Semiconductor. Numărul senzorilor care se pot lega pe această magistrală este teoretic nelimitat, acest număr depinzând de necesitățile utilizatorului.

De asemenea, aplicația conține și un modul de control ce poate fi programat a lua anumite decizii în funcție de anumite valori limită ale senzorilor: în cazul detectării unei temperaturi prea ridicate se poate atenționa utilizatorul, opri sistemul sau chiar efectua alte operații ce pot duce la scăderea temperaturii. Primele două sarcini sunt mai ușor de implementat dar, pentru conceperea unor operații mai complexe, sunt necesare anumite studii referitoare la diferitele categorii de componente ce pot duce la încălzirea excesivă a sistemului și studierea componentelor din aceeași categorie dar de la producători diferiți.

Aplicația constă dintr-un program ce a fost dezvoltat în Visual C++ 7.0 (.NET) și Visual C++ 6.0 și rulează pe un sistem de operare Windows (98, 2000, ME, NT și XP) și dintr-un montaj hardware experimental. Programul (denumit „Power Spy”) are scopul de a monitoriza procesele ce rulează la un moment dat pe un calculator și valorile (temperaturilor, tensiunilor de alimentare și a vitezei ventilatoarelor) citite de la senzorii integrați pe placa de bază. Aplicația are un mod de afișare text a datelor, un mod de afișare grafică a anumitor date culese, cât și posibilitatea de a salva aceste date în fișiere, pentru eventuale prelucrări ulterioare.

Având posibilitatea de a înregistra și salva datele despre procesele software și diferitele valori citite de la senzori (interni și externi), aplicația poate fi folosită în numeroase scopuri, cum ar fi: monitorizarea unei sesiuni de lucru a calculatorului, identificarea diferitelor procese nedorite ce rulează sau au rulat la un moment dat, monitorizarea încărcării procesorului unui sistem concomitent cu încărcarea termică și relația între ele.

Programul PowerSpy este structurat pe mai multe module, fiecare având propria interfață cu utilizatorul. Modulul de vizualizare a proceselor poate afișa date despre toate procesele ce rulează la un moment dat pe calculator (date de genul: PID (process id), nume, prioritate, gradul de utilizare al procesorului, memoria utilizată de program, etc). Aceste date sunt afișate în mod text, fiind grupate într-o listă (figura 4.16); separat afișându-se numărul total de procese ce rulează și gradul total de utilizare al procesorului.

### Modulul de monitorizare al proceselor

Modulul de monitorizare a proceselor poate fi configurat să salveze informații despre procesele aflate în execuție, detaliate pe fiecare moment de timp sau mediate pe un anumit interval de timp. În anexa 5 este prezentat un exemplu de raport cu procesele executate pe un server într-un interval de timp de 5 ore. Pe sistemul respectiv este instalat sistemul de operare Windows 2000 Server (SP 4) și este configurat ca server WEB (IIS4) și server de baze de date (Oracle și Microsoft SQL Server).

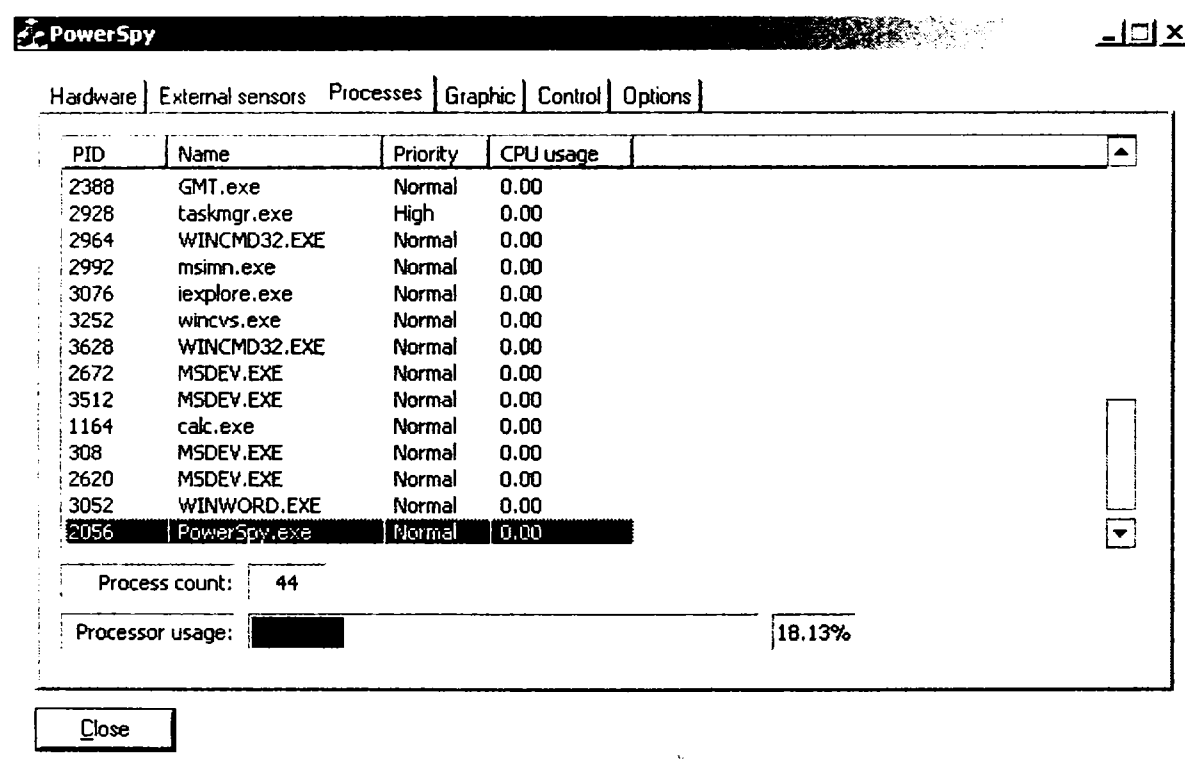


Figura 4.16 – Modulul de vizualizare a proceselor

Structura de clase ce implementează modulul de monitorizare a proceselor este prezentată în figura 4.18. Structura de date CProcessData conține toate informațiile specifice unui anumit proces: identificatorul unic al procesului, numele procesului, clasa de prioritate, timpul la care a fost lansat procesul, timpul când a fost încheiat, memoria ocupată, numărul de fire de execuție și timpul procesor alocat. Cunoscând tipul procesor alocat fiecărui proces se poate estima procentual gradul de încărcare a procesorului de către procesul respectiv, într-un anumit interval de timp. La un moment dat pe sistem se află în execuție mai multe procese ce sunt reținute în aplicație într-o listă – CProcessList. Însușind timpii procesor alocați pentru toate procesele se poate estima gradul total de încărcare al procesorului. Lista de procese se actualizează automat la anumite intervale de timp selectabile de către utilizator. Mai apoi, lista de procese se afișează într-o fereastră (figura 4.16) și se poate salva într-un fisier de log. Date statistice despre procesele executate pe un interval mai mare de timp se strâng la nivelul clasei CProcessHistory (Anexa 5).

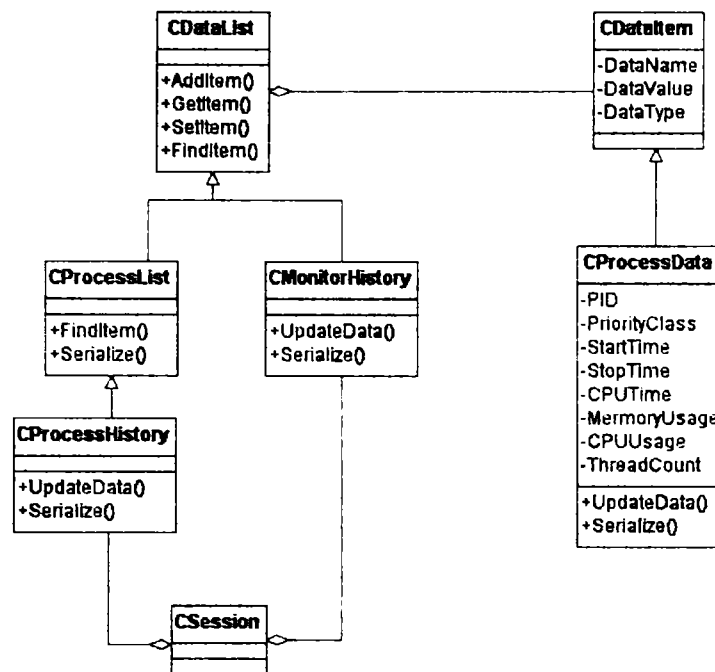


Figura 4.17 – Structurile de date de monitorizare (senzori și procese)

### Modulul de monitorizare al senzorilor interni

O a doua categorie de informații furnizate de către aplicație sunt valorile achiziționate de senzorii integrați de pe placa de bază și din procesor. Aceste informații sunt culese și prezentate de modulul de monitorizare hardware, a cărui interfață este prezentată în figura 4.18. Pe lângă informațiile citite de la senzori, aplicația identifică tipul și frecvența procesorului, tipul chipset-ului plăcii de bază (scanarea făcându-se pe magistrala PCI), tipul de acces la senzori (SMBus sau ISA), adresele și tipul senzorilor prezenți în sistem. Valorile citite de la senzorii integrați sunt grupate pe trei categorii: temperaturi, tensiuni de alimentare și viteze de rotație ale ventilatoarelor.

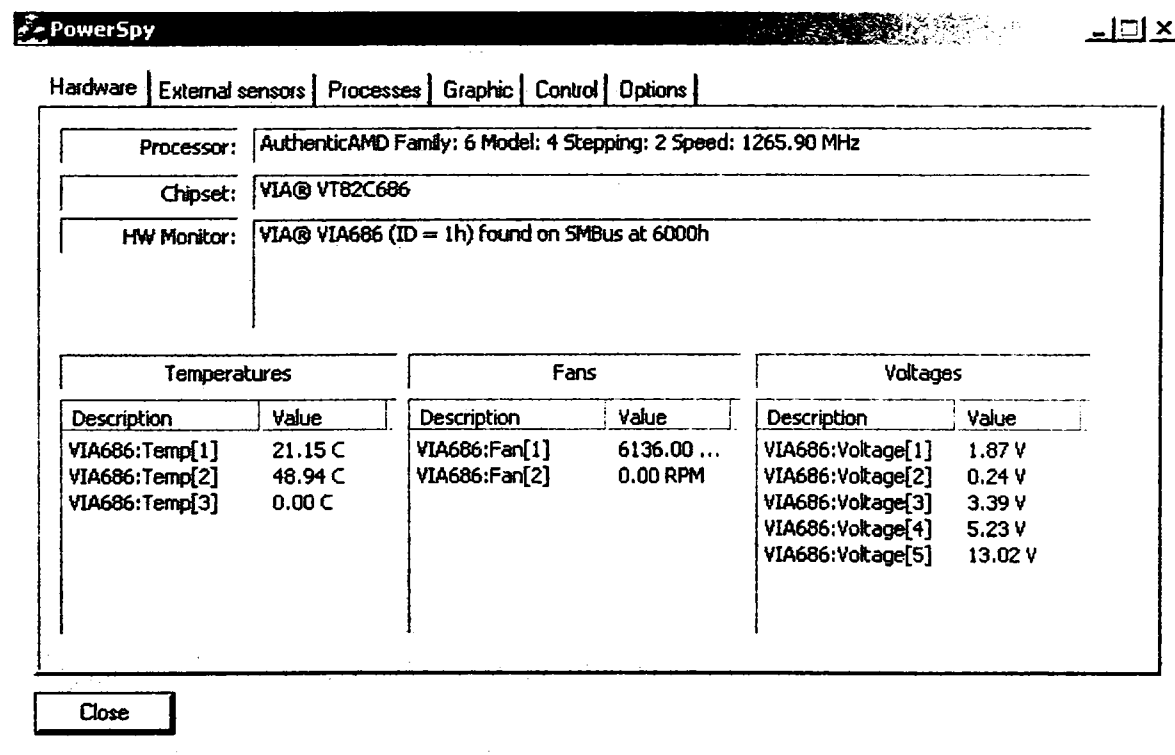


Figura 4.18 – Modulul de vizualizare a senzorilor de monitorizare

Datele achiziționate de la senzori sunt salvate într-o listă de obiecte de tipul `CDataItem`, ce conțin numele mărimii achiziționate, valoarea și tipul ei (figura 4.17). Și aceste date pot fi salvate într-un fișier de log și se pot acumula în clasa `CMonitorHistory`, pentru a oferi date statistice pe o perioadă mai lungă de timp.

Pentru a ajunge să comunice cu circuitul de monitorizare hardware și pentru a putea prelua date de la senzorii de pe placa de bază și procesor, aplicația trebuie să implementeze un întreg algoritm de identificare a fiecărui dispozitiv cu care trebuie să comunice. În primul pas al algoritmului de identificare aplicația trebuie să scaneze magistrala PCI pentru a identifica chipset-ul de intrare/ieșire (South Bridge Super I/O Chipset). Acest pas este relativ ușor de implementat deoarece toate dispozitivele PCI au același mod, standardizat, de identificare printr-un identificator unic al producătorului dispozitivului (VendorID) și un identificator unic al tipului dispozitivului (DeviceID). Aplicația poate detecta un număr mare de modele de chipset-uri produse de Intel, VIA, AMD, nVidia și ServerWorks.

Al doilea pas al algoritmului de identificare vizează detecția magistralei SMBus. Majoritatea (dacă nu chiar toate) chipset-urile I/O noi au implementată o interfață hardware de comunicație cu magistrala SMBus. La această magistrală sunt conectate circuitele de monitorizare hardware și prin acestea, senzorii prezenți în sistem. Problema implementării acestui al doilea pas a fost lipsa unei standardizări a interfeței de programare și detecție a modulului master SMBus. Fiecare producător a implementat în chipset-urile sale adrese diferite și moduri de identificare diferite pentru modulul de interfață cu magistrala SMBus, astfel că au fost studiate datele de catalog pentru chipset-urile suportate și s-a implementat cod distinct de identificare pentru fiecare chipset în parte.

În continuare, dacă a fost detectată cu succes prezența magistralei SMBus, trebuie scanată și această magistrală pentru a identifica circuitele de monitorizare hardware prezente în sistem. La fiecare circuit de monitorizare pot fi conectați unul sau mai mulți senzori, pentru diferite mărimi (temperatură, tensiuni și viteze de rotație). Probleme apar și aici deoarece nici interfața cu circuitele de monitorizare hardware nu este standardizată. Varietatea de circuite de monitorizare este și mai mare decât în cazul chipset-urilor, astfel că implementarea acestei părți a aplicației a fost și mai dificilă și a implicat un volum mare de muncă și căutări de documentații. Aplicația suportă următoarele circuite de monitorizare hardware: ADM (9240, 1025, 1027), LM (75,78), VIA (586, 686), WinBond (83781, 83627, 83782, 83783).

Dacă magistrala SMBus nu a fost detectată atunci trebuie scanate anumite adrese de pe magistrala ISA pentru a detecta posibilele circuite de monitorizare ce au și interfață ISA.

Fiecare tip de interfață la câte o magistrală este implementată într-o clasă derivată din `CBusInterface` (figura 4.19). Această clasă oferă un set de funcții de interfață pentru citirea și scrierea datelor la un anumit dispozitiv de pe magistrală. Implementarea funcțiilor de citire/scriere se face în clasele derivate în funcție de specificul comunicației pe tipul corespunzător de magistrală.

La scanarea magistrelor, pentru fiecare dispozitiv detectat și recunoscut de aplicație (și util aplicației) se crează câte un obiect dispozitiv (derivat din `CDevice`). Obiectele dispozitiv au o referință către magistrala la care sunt atașate (`CBusInterface`) pentru a avea acces la implementarea protocolului de comunicație specific magistralei respective (funcțiile de citire/scriere). Obiectele corespunzătoare dispozitivelor de monitorizare hardware au o aceeași interfață – `CHardwareMonitor` (figura 4.20). Pentru fiecare tip specific sau grup de dispozitive asemănătoare ale aceluiași producător se implementează câte o clasă cu operațiile specifice circuitului respectiv. În acest mod aplicația poate fi ușor extinsă pentru a putea fi interfațată și cu alte circuite de monitorizare hardware.



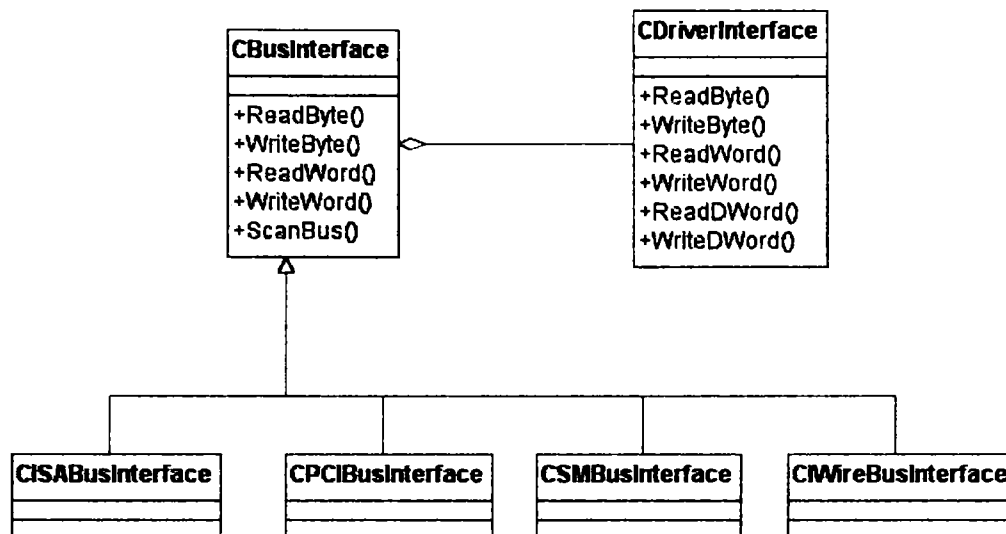


Figura 4.19 – Clasele de interfață pentru magistralele sistem

Obiectele dispozitiv create se introduc într-o listă ce este accesată la intervale de timp stabilite de către utilizator pentru a se obține date de la toți senzorii detectați de către aplicație. Aceste date se introduc niște structuri de date de tipul CDataItem ce sunt mai apoi afișate și salvate pentru operații și statistici ulterioare.

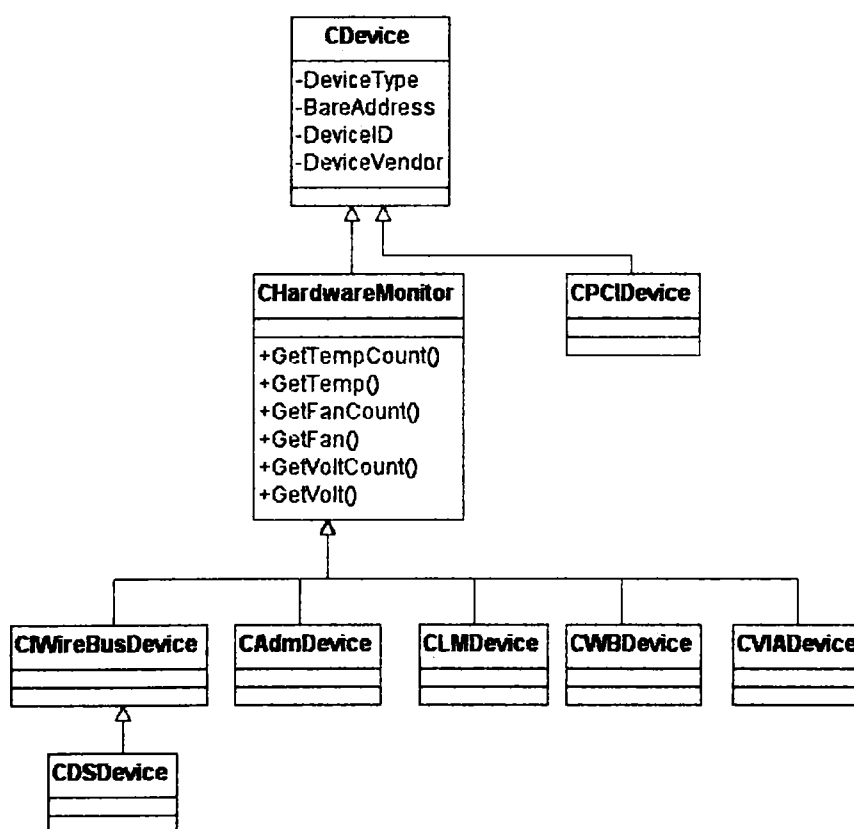


Figura 4.20 – Dispozitivele hardware suportate de aplicație

### Modulul de monitorizare al senzorilor externi

Acest modul este destinat monitorizării unor dispozitive externe conectate prin 1-Wire Bus la interfața serie sau paralelă a calculatorului. Schemele utilizate pentru 1-Wire Bus pot fi cele prezentate în subcapitolul anterior. Structurile de date utilizate de acest modul sunt aceleași ca cele folosite de modulul de monitorizare a senzorilor interni. Aici se mai definește o clasă pentru dispozitivele ce au o interfață 1-Wire Bus. La această magistrală se pot conecta

senzorii de temperatură Dallas (DS1820, DS1821 și DS1822) implementați în program prin obiecte de tipul CDSDevice (figura 4.20).

Acest modul scanează porturile serie și cele paralele pentru a identifica vreun senzor extern de tip 1-Wire Bus. Senzorii detectați și valorile returnate de ei sunt prezentate în interfața din figura 4.21.

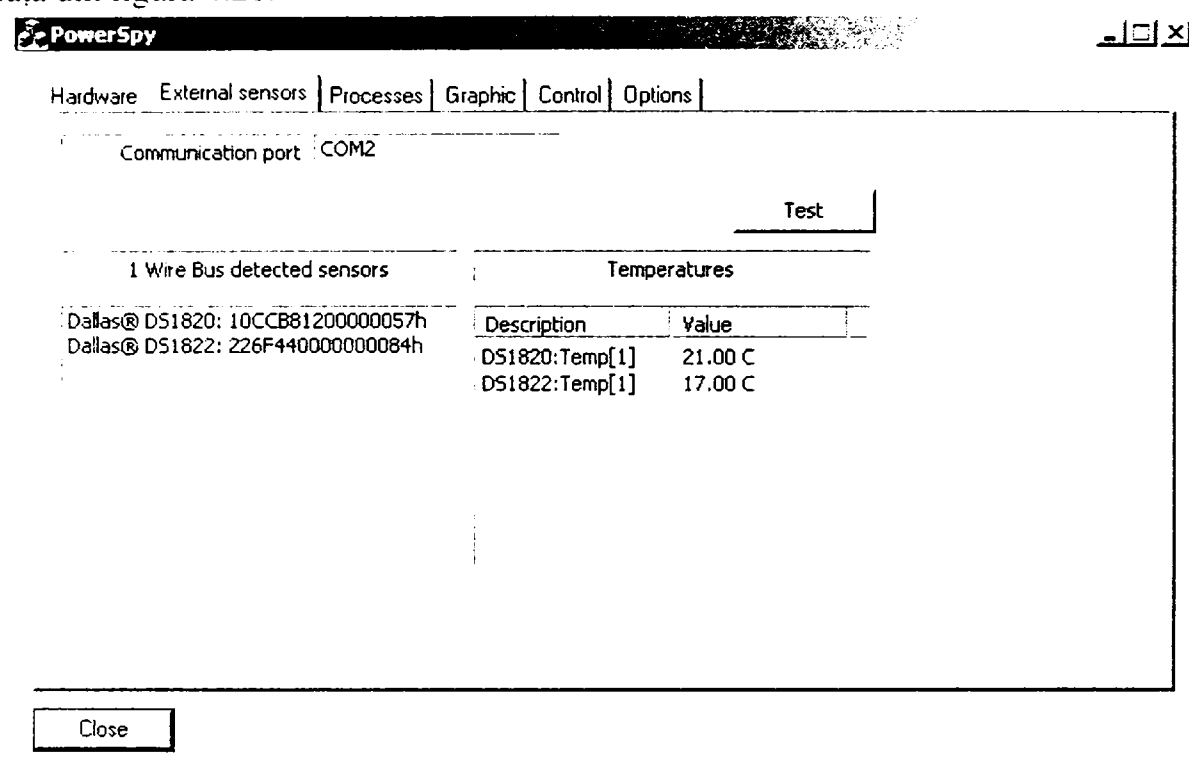


Figura 4.21 – Modulul de vizualizare al senzorilor externi

### Modulul de vizualizare grafică

Pe lângă afișarea datelor în mod text a fost prevăzut și un mod grafic mai intuitiv (figura 4.22). Acesta poate afișa doar anumite informații, cum ar fi: gradul global de utilizare a procesorului, temperaturile, tensiunile și vitezele de rotație a ventilatoarelor. Mărimile afișate în format grafic sunt selectate din interfața de opțiuni a aplicației.

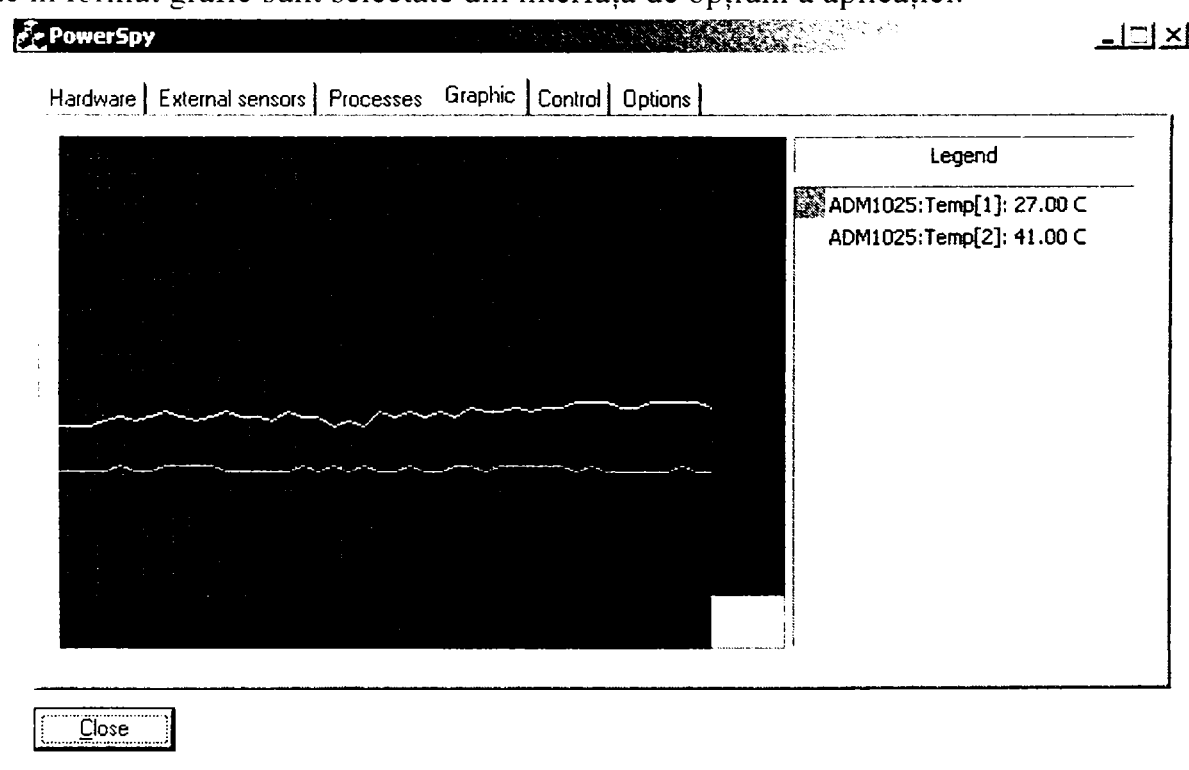


Figura 4.22 – Modulul de vizualizare în format grafic

## Modulul de control

Pe baza datelor de monitorizare se pot lua anumite decizii pentru buna funcționare în continuare a sistemului. Temperatura este unul din principalii parametri ce influențează fiabilitatea componentelor unui sistem de calcul, astfel că este foarte utilă existența pe sistemele de calcul a unei aplicații ce poate fi configurată să execute anumite operații la încălzirea componentelor peste anumite limite.

Acțiunile de control cu care poate fi configurată aplicația sunt: afișarea unor mesaje, oprirea sistemului, oprirea aplicațiilor ce încarcă procesorul pentru o perioadă îndelungată de timp, pornirea unor aplicații de răcire a sistemului, pornirea procesului de autocontrol pentru task-urile sistemului.

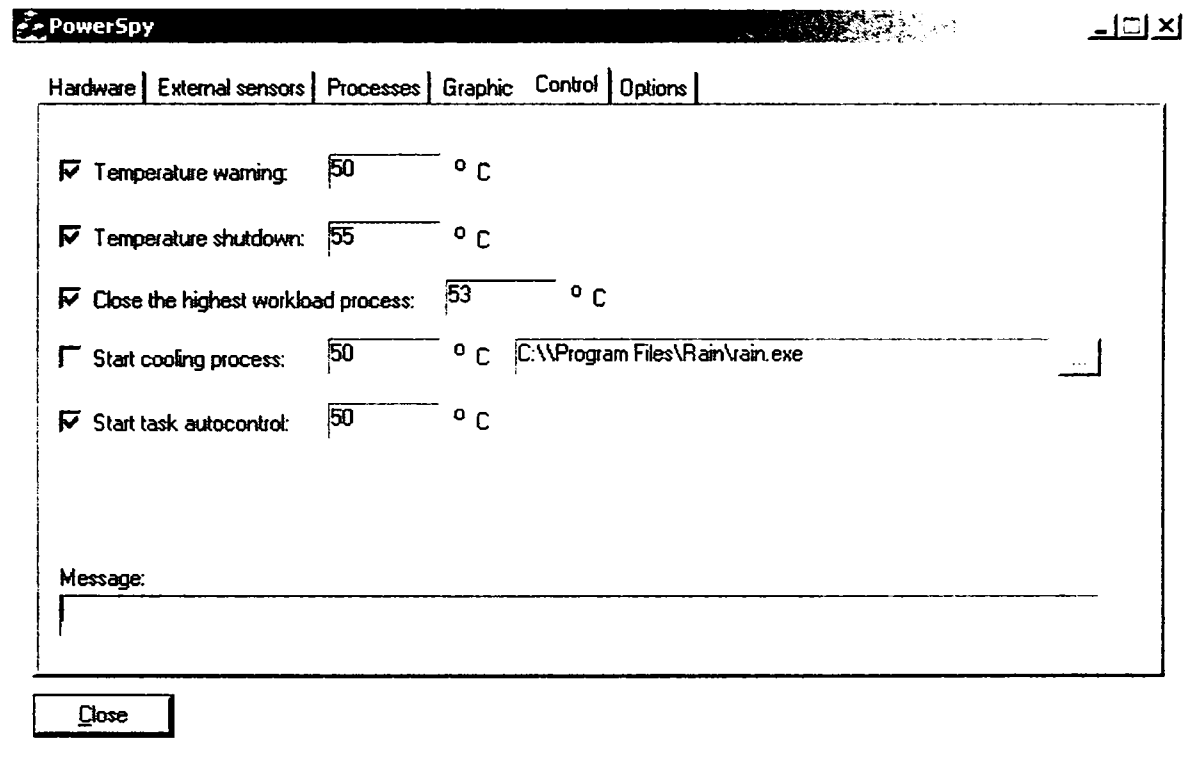


Figura 4.23 – Interfața modului de control

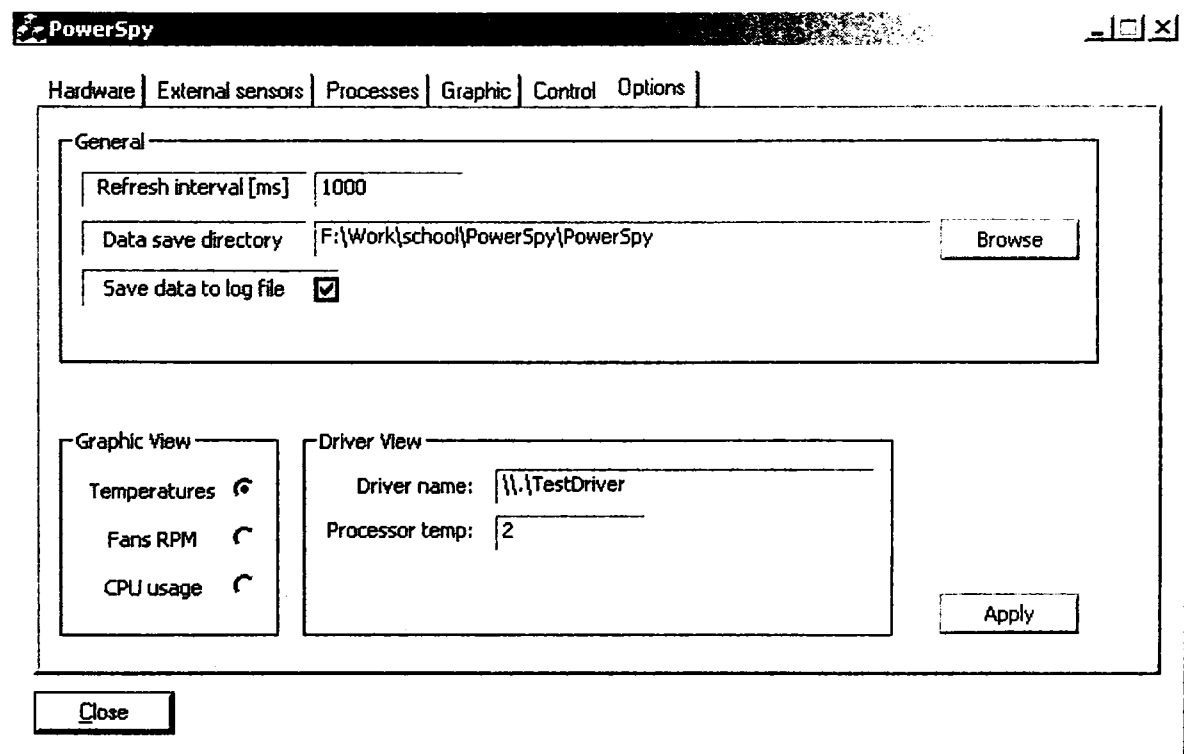
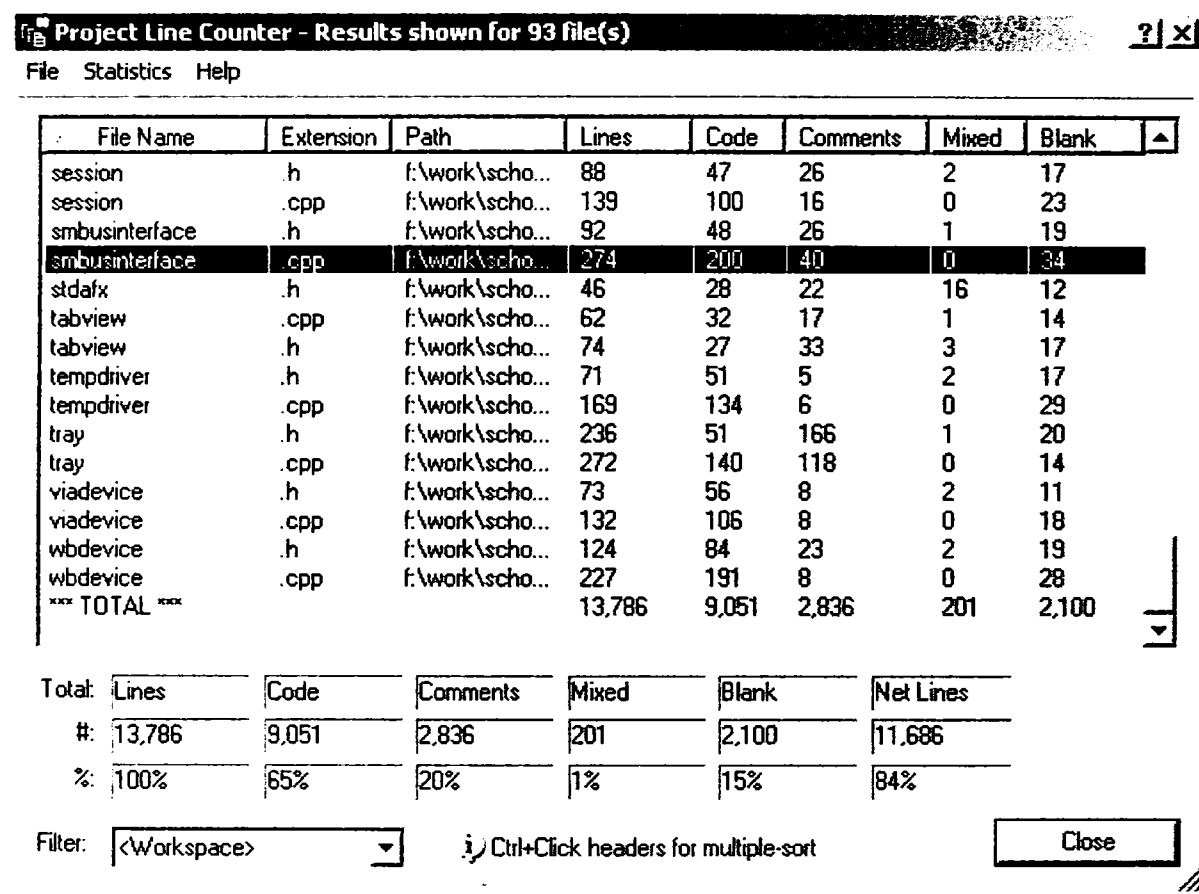


Figura 4.24 – Opțiunile aplicației

## Opțiunile aplicației

Aplicația are anumiți parametri ce pot fi configurați de către utilizator. Opțiunile globale permit modificarea intervalului de achiziție a datelor, selectarea modului de înregistrare precum și numele de bază a fișierelor în care vor fi stocate datele culese. Intervalul de reîmprospătare a datelor se referă atât la procese cât și la senzori.



File Name	Extension	Path	Lines	Code	Comments	Mixed	Blank
session	.h	f:\work\scho...	88	47	26	2	17
session	.cpp	f:\work\scho...	139	100	16	0	23
smbusinterface	.h	f:\work\scho...	92	48	26	1	19
smbusinterface	.cpp	f:\work\scho...	274	200	40	0	34
stdafx	.h	f:\work\scho...	46	28	22	16	12
tabview	.cpp	f:\work\scho...	62	32	17	1	14
tabview	.h	f:\work\scho...	74	27	33	3	17
tempdriver	.h	f:\work\scho...	71	51	5	2	17
tempdriver	.cpp	f:\work\scho...	169	134	6	0	29
tray	.h	f:\work\scho...	236	51	166	1	20
tray	.cpp	f:\work\scho...	272	140	118	0	14
viadevice	.h	f:\work\scho...	73	56	8	2	11
viadevice	.cpp	f:\work\scho...	132	106	8	0	18
wbdevice	.h	f:\work\scho...	124	84	23	2	19
wbdevice	.cpp	f:\work\scho...	227	191	8	0	28
*** TOTAL ***			13,786	9,051	2,836	201	2,100

Total:	Lines	Code	Comments	Mixed	Blank	Net Lines
#:	13,786	9,051	2,836	201	2,100	11,686
%:	100%	65%	20%	1%	15%	84%

Filter: <Workspace> Ctrl+Click headers for multiple-sort Close

Figura 4.25 – Statistici pentru aplicația PowerSpy

Aplicația PowerSpy este structurată pe aproximativ 40 de clase și are aproximativ 9000 linii de cod (figura 4.25).

### 4.3.4 Studiul semnăturii termice a sistemelor de calcul

Aplicația prezentată în subcapitolul precedent a fost utilizată pentru determinarea semnăturii termice a unui sistem. Semnătura termică a unui sistem este definită ca fiind răspunsul termic al fiecărei componente a sistemului pe un anumit interval de timp sau la o anumită încărcare a sistemului. Majoritatea sistemelor actuale, atât sistemele portabile cât și serverele și calculatoarele personale, sunt prevăzute cu senzori interni de monitorizare a temperaturii. Toate sistemele actuale au montat un senzor de temperatură pentru monitorizarea temperaturii în interiorul carcasei calculatorului. Componentele unui sistem care consumă cel mai mult și se încălzesc cel mai puternic sunt procesorul și harddisk-ul [Viswanath2000, Lorch98]. În majoritatea sistemelor noi și aceste două componente sunt prevăzute cu senzori de temperatură, însă în cadrul cercetărilor ce au stat la baza acestei teze s-a luat ca și componentă țintă microprocesorul sistemului.

Un exemplu de semnătură termică a microprocesorului unui sistem este prezentat în figura 4.26. Temperatura procesorului este dependentă de temperatura mediului ambiant, de mecanismul de răcire utilizat, de modelul procesorului (tipul de capsulă utilizat, numărul de tranzistoare integrate, ș.a.) și în final de aplicațiile ce rulează la un moment dat pe sistem.

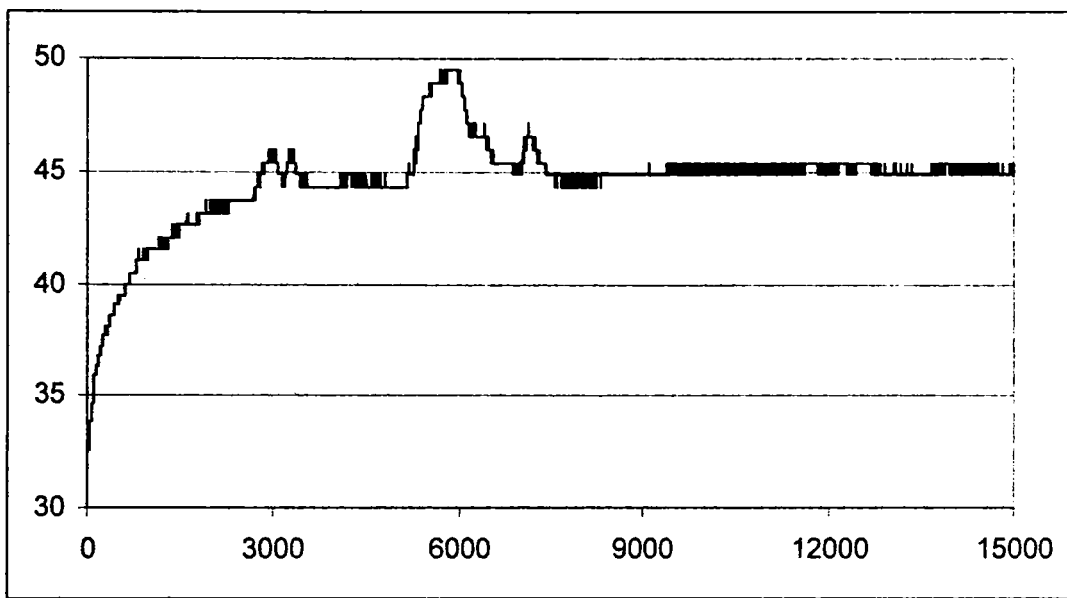


Figura 4.26 – Semnătura termică a procesorului unui sistem

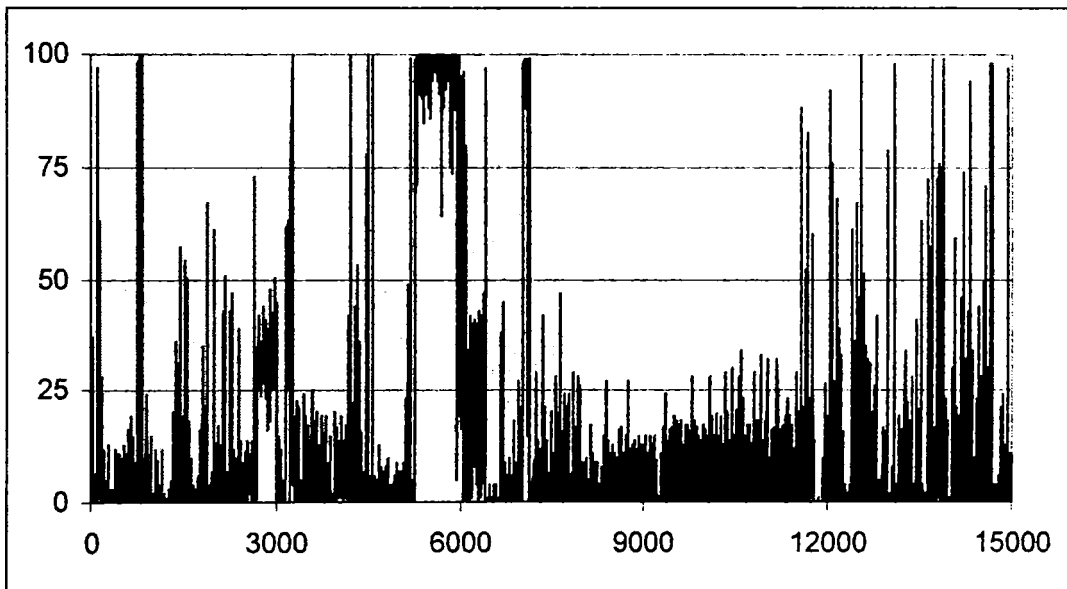


Figura 4.27 – Încărcarea procesorului pe perioada de achiziție a semnăturii

Semnătura termică pentru exemplul din figura 4.26 a fost achiziționată pe un sistem cu procesor AMD Athlon la 1200 MHz și 256 MB memorie pe durata a aproximativ 4 ore. Încărcarea procesorului pe perioada achiziției semnăturii este prezentată în figura 4.27. Perioadele de încălzire accentuată a procesorului apar în momentele în care procesorul a fost utilizat în procent de 100% pentru o perioadă mai îndelungată de timp. Aplicațiile utilizate în acest interval de timp sunt specifice pentru dezvoltarea de aplicații software: mediu de dezvoltare și aplicațiile de compilare, pe lângă acestea mai intră testarea aplicației și uneltele de documentare a ei. Distribuția procentuală a timpului procesor alocat fiecărei aplicații sau grup de aplicații este prezentată în figura 4.28. Încărcarea suplimentară introdusă în sistem de către mecanismul de monitorizare și achiziționare a semnăturii termice este de aproximativ 0.02%, echivalentul încărcării unei aplicații antivirus.

Încălzirea cea mai accentuată a procesorului a avut loc la compilarea aplicației ce a durat aproximativ 15 minute iar temperatura procesorul a ajuns la aproximativ 50 °C, cu 5 °C peste temperatura normală de lucru a procesorului. Celelalte puncte în care procesorul a fost solicitat și s-a încălzit corespund momentelor în care a fost testată aplicația. Aplicația testată este o aplicație de baze de date scrisă în Visual C++ pentru gestionarea costurilor de telefonie (Unitel).

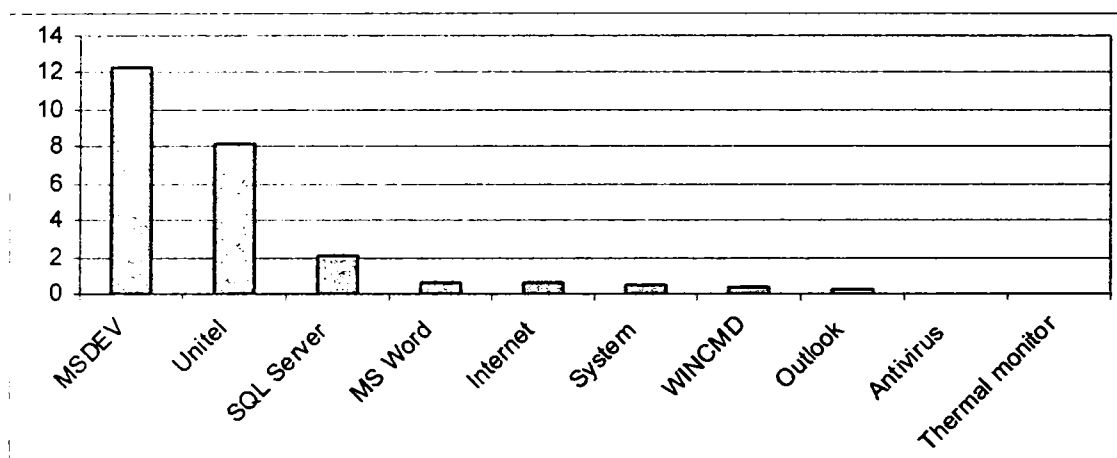
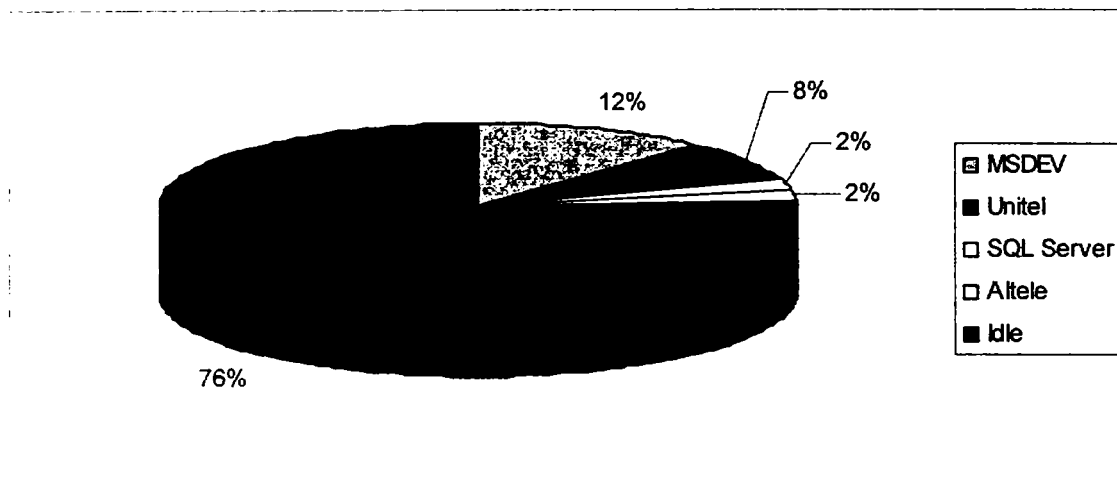


Figura 4.28 – Aplicațiile executate pe perioada de achiziție a semnăturii

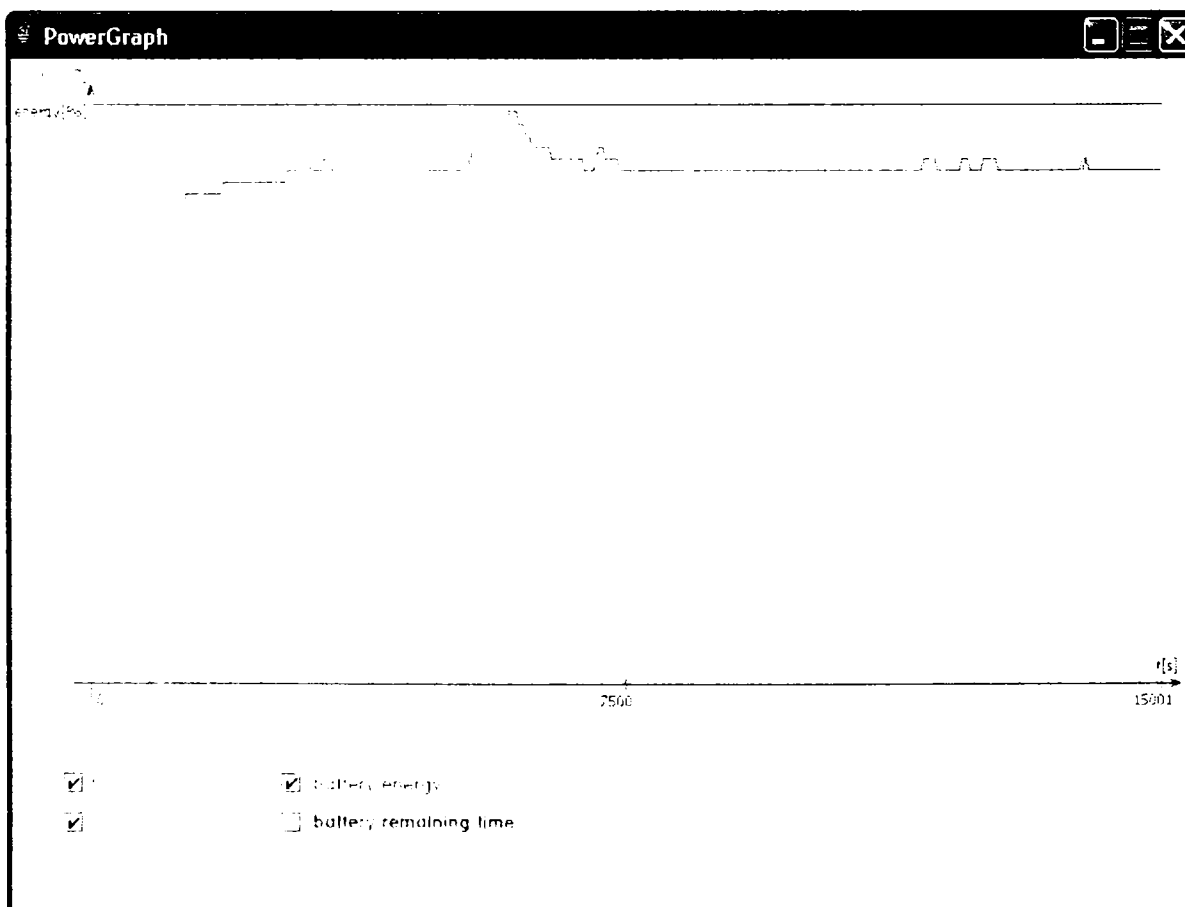


Figura 4.29 – Grafic comparativ între temperatură și încărcarea procesorului

Pe lângă aplicațiile executate pe sistem, temperatura procesorului este influențată în mod direct și de temperatura mediului ambiant. Un studiu al semnăturilor procesorului achiziționate la temperaturi diferite ale mediului, oferă informații despre gradul în care temperatura procesorului este influențată de temperatura mediului și modul în care mecanismele interne de răcire fac față modificărilor de temperatură. În figura 4.30 sunt prezentate semnăturile aceluiași sistem achiziționate la temperaturi diferite ale mediului.



Figura 4.30 – Semnătura termică a unui sistem achiziționate la temperaturi diferite

Semnătura termică poate fi utilizată și la compararea sistemelor, a mecanismelor de răcire, a procesoarelor și oferă date despre diferitele moduri de dispunere fizică a componentelor în sistem și pe placa de bază.

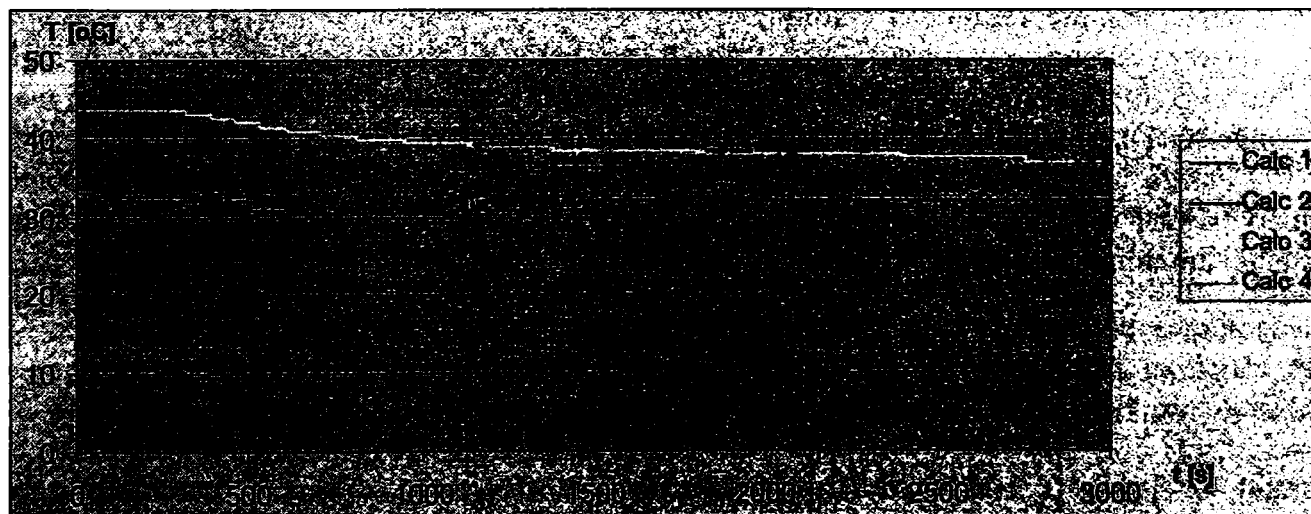


Figura 4.31 – Semnăturile termice pentru 4 sisteme diferite

În figura 4.31 sunt prezentate semnăturile termice a patru sisteme. Semnăturile au fost achiziționate timp de aproximativ o oră începând de la pornirea sistemului, iar după încărcarea sistemului de operare, calculatoarele nu au fost folosite la nimic (au fost în starea Idle). Semnăturile vor fi în continuare detaliate în subcapitolul de benchmark-uri termice.

## 4.4 Semnătura termică a programelor

### 4.4.1 Mecanisme de reducere a temperaturii și puterii disipate

În ultimii ani, progresele realizate în domeniul microelectronicii au fost foarte mari și foarte rapide, în special în ceea ce privește tehnologia de fabricație a circuitelor integrate VLSI (microprocesoare, microcontrollere, procesoare DSP, SoC). Pentru a oferi o putere de calcul tot mai mare (mai multe funcționalități, complexitate ridicată) și viteză de procesare tot mai mare, la baza evoluției circuitelor integrate a stat așa numitul proces de scalare (Tabelul 1.3). Scalarea este procesul prin care se poate crește densitatea de integrare și viteza de operare, prin micșorarea dimensiunilor tranzistorilor, scăderea tensiunii de prag și a tensiunii de alimentare. Pe un plan secund au fost lăsate alte caracteristici importante cum ar fi puterea consumată, temperatura disipată și în unele cazuri chiar calitatea și fiabilitatea circuitelor, care au fost în mod direct afectate de procesul de scalare. Însă, în ultimul timp, datorită problemelor care apar, tot mai mult se discută despre controlul puterii consumate și scăderea temperaturii disipate de circuitele integrate VLSI.

Principalele cauze ale creșterii puterii consumate sunt creșterea densității de integrare (în special datorită micșorării grosimii stratului de oxid) și a dimensiunilor pastilei de siliciu, numărul tot mai mare de tranzistoare integrate într-un circuit și frecvența de operare tot mai mare. Pentru a limita creșterea consumului de putere datorită procesului de scalare, o dată cu scăderea dimensiunilor tranzistoarelor s-a impus și micșorarea tensiunii de alimentare, aceasta fiind o primă metodă de control al consumului de putere – tensiunile de alimentare variabile.

Cele mai importante efecte ale creșterii temperaturii sunt următoarele:

- scăderea vitezei – viteza scade cu 0.15% pentru fiecare creștere a temperaturii cu 1 °C. Creșterea temperaturii afectează timpii de comutare la nivel de poartă logică;
- scăderea fiabilității – durata de viață se diminuează cu 50% la o creștere a temperaturii cu 10-15 °C. Creșterea temperaturii afectează și degradează fizic structura de siliciu ducând la defectarea prematură a circuitului;
- creșterea curenților reziduali –  $I_{OFF}$  crește cu temperatura. Poate apare fenomenul de fugă termică (thermal runaway) datorită creșterea curenților reziduali implică o încălzire și mai puternică a circuitului, care la rândul ei produce un curent mai mare, ș.a.m.d.

Pentru a evidenția distribuția consumului între componentele unui sistem se poate urmări tabelul 4.4. Se poate observa că o pondere foarte mare din consumul total al unui sistem o are unitatea centrală.

Componenta	Putere consumata [W]	Pondere [%]
Microprocesor	60-105	30
Monitor	55-110	30
HDD/CDROM	10-15	7
Boxe	10	3.5
Modem/Retea	10	3.5
Imprimanta	75	26

Tabelul 4.4 – Distribuția consumului de putere într-un sistem PC [Viswanath2000]

Un alt exemplu de distribuție a consumului de putere, între componentele unui sistem portabil, se poate vedea în tabelul 4.5. Componentele în acest al doilea exemplu sunt de consum redus.



Componenta	Putere consumata [W]	Pondere [%]
Microprocesor	2.24	28
Ecran	2.80	35
HDD	0.64	8
Video	0.80	10
Memorie	0.16	2
Altele	1.36	17

Tabelul 4.5 – Distribuția consumului de putere într-un sistem portabil [Lorch98]

Consumul de energie a devenit astfel o preocupare principală a proiectanților de circuite integrate și sisteme digitale datorită recentei popularități a sistemelor portabile și în al doilea rând, datorită cerințelor ecologice pentru sistemele desktop și servere. Capacitatea de stocare și durata de viață a bateriilor au crescut relativ lent (o creștere de 4 ori în ultimii 30 de ani [Simunic2001]), în timp ce nevoile computaționale au crescut mult mai mult în același interval de timp (Tabelul 1.4).

Metodele de control pentru puterea disipată de sistemele de calcul au la bază câteva mecanisme hardware implementate la nivelul fiecărei componente a acestuia. Controlul puterii disipate sau a energiei consumate de un sistem se realizează prin utilizarea independentă sau colaborată a mecanismelor de control de la nivelul fiecărei componente în parte în funcție de sarcinile pe care le are de îndeplinit sistemul. Pentru ca o componentă a sistemului să fie inclusă în mecanismul global de control al puterii, trebuie să fi fost proiectată pentru a suporta două sau mai multe stări sau moduri de lucru, fiecare având un anumit nivel de consum. În cazul cel mai simplu o componentă va suporta două stări: una activă și una pasivă. În starea activă componenta este pe deplin funcțională și disipă o cantitate de putere dependentă de activitatea la care este supusă. În cea de-a doua stare, cea pasivă, componenta este oprită (de exemplu nu este alimentată) astfel că atât timp cât ea nu este utilizată poate fi trecută în starea pasivă în care consumul este redus. În mod uzual, în funcție de complexitatea dispozitivului și de nivelul de consum al acestuia, se pot implementa și alte mecanisme prin care se poate efectua controlul puterii disipate de către dispozitiv.

Mecanismele ce stau la baza reducerii consumului de putere sunt aceleași indiferent de tipul sistemelor:

- majoritatea componentelor sistemului au implementate una sau mai multe stări cu consum redus. În stările cu consum redus circuitele efectuează un număr redus de activități sau chiar nu fac nimic. Revenirea din aceste stări într-o stare activă se face după un anumit timp de revenire. Principalele cercetări la acest nivel vizează algoritmi pentru trecerea componentelor sistemului dintr-o stare de consum în alta în funcție de evenimentele apărute în sistem. Datorită faptului că aceste evenimente nu pot fi nici modelate nici prevăzute, algoritmi DPM (Dynamic Power Management) se bazează pe euristici. [Ramanathan2002, Simunic2001, Lu2000];
- posibilitatea de control (prin software sau hardware) al frecvenței: frecvența de lucru a componentelor (microprocesor, microcontroler în special) se modifică în funcție de raportul dintre consum și performanță specificat;
- posibilitatea de control (prin software sau hardware) a tensiunilor de alimentare: tensiunile de alimentare ale componentelor se modifică în funcție de raportul dintre consum și performanță specificat. Și în acest caz accentul se pune pe implementarea unor algoritmi numiți DSV (Dynamic Voltage Scaling) [Rakhmatov2002, Zhang2002];
- microprocesoare/sisteme morfice: este un concept nou, apărut puțin înainte de anul 2000 și a fost implementat pentru prima dată la procesoarele Crusoe ale firmei Transmeta. Aceste procesoare au implementat un set de instrucțiuni simplu și cu consum redus. Pe baza

acestui set de instrucțiuni se emulează într-un nivel software setul de instrucțiuni al arhitecturii x86 [Zwicker2001].

Logica pe baza căreia se iau, în fiecare moment de timp, deciziile de utilizare a diferitelor mecanisme de control al puterii disipate este încapsulată într-un algoritm. Implementarea unui algoritm de control al puterii disipate se poate face la mai multe niveluri:

- la nivelul fizic (hardware) – algoritmul este implementat în logica de control a componentei vizate. În acest caz mecanismul de control este specializat strict pe specificul de funcționare al componentei respective, astfel că se poate obține un control maxim într-un timp scurt (fără o latență prea mare). Pe de altă parte, algoritmul este dificil de controlat de către nivelurile superioare. Acest nivel de implementare a mecanismului de control este utilizat la dispozitivele portabile, acolo unde acțiunile specifice modului de consum redus sunt decisive și trebuie efectuate fără o latență prea mare.
- la nivelul sistemului de operare – deciziile pentru controlul puterii disipate de sistem ca întreg revin sistemului de operare iar mecanismul de control al puterii, specific fiecărui dispozitiv, se implementează în driverul atașat dispozitivului. Între drivere și nucleu se stabilește o relație foarte strânsă existând un dialog permanent între ele. Comunicarea între nucleul sistemului de operare (mai concret între modulul de control al puterii – Power Manager) și drivere se realizează prin mesaje (POWER\_MESSAGES). Dacă în cazul precedent utilizatorul nu avea aproape nici o modalitate de a interveni în algoritmi de control al puterii, în acest caz sistemul de operare oferă posibilitatea de a configura acești algoritmi. Controlul de către utilizator este totuși destul de limitat, oferindu-i utilizatorului doar posibilitatea de a modifica anumiți parametri ai algoritmului sau de a trece sistemul într-una din cele trei stări generale de consum implementate în majoritatea sistemelor de operare: starea activă (pornit), starea oprită și starea de așteptare (sleep). În starea de așteptare majoritatea componentelor sistemului sunt oprite, existând posibilitatea de a fi repornite automat la apariția anumitor evenimente (trecerea unui anumit interval de timp, primirea unui apel pe modem sau a unui pachet pe una din interfețele de comunicație, ș.a.).
- la nivelul aplicației – deciziile pentru controlul puterii disipate sunt distribuite între aplicațiile utilizator. Aplicațiile utilizator trebuie să comunice cu nucleul sistemului de operare și cu driverele ce au implementate mecanisme de control al puterii. Implementarea algoritmului de control al puterii la nivelul aplicațiilor utilizator oferă utilizatorului gradul cel mai mare de controlabilitate și se poate obține reducerea cea mai semnificativă de consum de energie. Majoritatea algoritmilor de control al puterii sunt implementați la nivel fizic sau la nivel de sistem de operare datorită complexității implementării acestora la nivelul aplicațiilor utilizator. Datorită acestei complexități există mai multe grade de implicare a aplicațiilor utilizator în procesul de control al puterii. În cazul cel mai simplu o aplicație utilizator poate trata mesajele specifice trimise de nucleul sistemului de operare. Un grad mai mare de implicare al aplicațiilor utilizator în procesul de control al puterii, fiecare aplicație își estimează aportul adus consumului total, oferă informații despre posibilitatea de finalizare a operațiilor înainte de a se termina bateria, oferă posibilitatea utilizatorului de a controla execuția fiecărui proces sau fir de execuție pentru a permite finalizarea operațiilor ce sunt mai importante din punctul de vedere al utilizatorului.

Implementarea controlului puterii disipate la nivelurile cele mai înalte duce la o reducere considerabilă a consumului de energie relativ la implementarea mecanismelor de gestiune a puterii la nivelurile inferioare. Cele mai întâlnite articole însă, vizează primele din nivelurile prezentate mai sus: nivelul fizic și sistem de operare. În cadrul cercetărilor ce au stat la baza acestei teze s-a abordat nivelul aplicație și un nivel intermediar între aplicații și

sistemul de operare și anume driverele. S-a optat pentru această soluție deoarece aplicațiile știu sau pot estima cel mai bine operațiile necesare, durata și importanța lor, iar pe de altă parte, driverele sunt cele care pot obține informații de monitorizare de la senzorii de temperatură și pot controla anumiți parametri ai componentelor sistemului.

Algoritmii DPM (Dynamic Power Management) de control al puterii reduc consumul de energie prin plasarea selectivă a componentelor neutilizate din sistem în stările de consum redus specifice componentelor respective [Lorch1998, Simunic2001, Chedid2002]. De exemplu, un sistem poate avea o stare activă, o stare "idle" și o stare "sleep" în care consumă cel mai puțin. Tranzițiile între stările de consum sunt controlate de un modul de Power Management (PM) ce observă încărcarea sistemului și decide momentul de timp și în ce stare să se efectueze o tranziție.

Cei mai întâlniți algoritmi PM sunt cei bazați pe timpi de expirare. Dezavantajul acestora constă în consumul inutil din perioada de timp în care se așteaptă expirarea și trecerea într-o stare de consum redus. Un al doilea tip de algoritmi PM sunt cei bazați pe prevederea comportamentului sistemului sau a unei componente pentru următorul interval de timp. În acest caz, o componentă trece într-o stare de consum redus dacă ea nu este folosită și modulul de control prevede o perioadă mai lungă de inactivitate. O estimare incorectă duce atât la un consum mai mare cât și la degradarea performanțelor aplicației. Un al treilea tip de algoritmi sunt cei stocastici, ce se bazează pe distribuții ce descriu timpii de activare a cererilor pentru componente. Aceste distribuții se bazează pe simulări sau pe monitorizarea evenimentelor într-un sistem pentru un anumit interval de timp.

Cea de-a doua clasă de algoritmi de control al puterii sunt algoritmii DSV (Dynamic Voltage Scaling). Algoritmii din această clasă ajustează frecvența și tensiunea în acord cu încărcarea curentă a procesorului. Deoarece majoritatea sistemelor nu au nevoie de performanțe deosebite de fiecare dată, scăderea vitezei și tensiunii echipamentului în timpul perioadelor nu foarte solicitate, mărește eficiența energiei. Implementarea unui algoritm DVS pentru un procesor necesită atât suport hardware cât și software care însă nu este foarte răspândit, chiar dacă sunt câteva exemple de cum ar trebui implementat un algoritm DVS.

Sistemele de operare Windows (98, 2000, XP și 2003) oferă o interfață de programare (API) minimală pentru aplicațiile utilizator în vederea participării la procesul de control al puterii [OnNow2001, INTEL2002\_1]. Arhitectura OnNow propusă de Microsoft oferă un set de stări de consum de putere și politici de control al puterii printr-un set de funcții specifice ce pot fi utilizate de către aplicațiile utilizator pentru a crește eficiența și robustețea procesului de control al puterii disipate de sistem. Această arhitectură oferă proiectanților de aplicații software, posibilitatea de a lua parte activă sau pasivă în procesul de control al puterii și de a oferi utilizatorilor o interfață prin care să poată monitoriza și/sau acționa asupra acestuia.

Arhitectura OnNow oferă un set minimal de operații pe baza căruia se pot implementa sisteme de control mai complexe. Funcționalitățile implementate la momentul actual în aplicațiile Windows ce utilizează OnNow sunt prezentate pe scurt în continuare:

- planificarea pornirii automate a sistemului pentru anumite tipuri de aplicații ce nu necesită intervenție din partea utilizatorului și care se execută de obicei, în afara perioadei de lucru: operații de mentenanță, descărcarea de fișiere de pe Internet, actualizări ale sistemului, arhivări, etc. Când aceste operații s-au încheiat sistemul trebuie pus din nou în starea de consum redus.
- revenirea aplicațiilor din starea de așteptare (sleep). Dacă sistemul este trecut de utilizator în starea de așteptare (Sleep sau Hibernate în Windows), sistemul de operare salvează pe disc starea curentă a sistemului și aplicațiilor pentru ca la revenire să ofere utilizatorului exact aceleași aplicații ce rulau la momentul închiderii, într-un timp mult mai scurt decât dacă sistemul s-ar fi reîncărcat de la început. Acest mod de lucru prezintă câteva probleme care dacă nu sunt tratate în mod corespunzător de către aplicații va necesita

repornirea aplicației cu posibilitatea de pierdere a datelor. O primă problemă apare în cazul în care aplicația folosește fișiere. De exemplu, dacă se execută pe un laptop o aplicație ce deschide un fișier în rețea localizat pe un alt sistem (server), apoi se trece laptopul în starea de așteptare (sleep) înainte de a salva fișierul și închide aplicația. Dacă la repornirea sistemului (revenirea din starea de așteptare), laptopul nu mai este în rețea sau este în altă locație, atunci aplicația nu va mai putea lucra cu fișierul inițial și va trebui repornită iar datele se vor pierde. Dacă se folosesc mesajele trimise de sistemul de operare la intrarea și revenirea în/din starea de așteptare, atunci se pot trata cu succes aceste situații. De exemplu, la revenirea aplicației din starea de așteptare se poate cere noua locație a fișierului. Cea de-a doua problemă vizează păstrarea conexiunilor de rețea. La trecerea în starea de așteptare se pierd conexiunile de rețea existente cu diferitele servere cu care se lucrează. La repornire aceste conexiuni trebuie refăcute. În această categorie intră conexiunile client/server, conexiunile cu diferitele baze de date, etc.

- tratarea evenimentelor de la dispozitivele periferice. Anumite aplicații necesită o monitorizare continuă a anumitor evenimente (de exemplu, un fax sau un monitor de rețea). Pentru acest gen de aplicații sistemul ar trebui să fie pornit tot timpul, chiar și atunci când nu există evenimente de tratat pentru o perioadă mai îndelungată de timp. Acest tip de aplicații pot fi implementate folosind OnNow prin trecerea sistemului în starea de așteptare atunci când nu există evenimente iar sistemul va reporni automat la apariția unui eveniment extern.
- prevenirea activării stării de așteptare atunci când nu trebuie. În cazul aplicațiilor ce nu necesită intervenție din partea utilizatorului (prezentări, filme), activarea automată a unei stări de așteptare poate interveni în timpul execuției aplicației respective (de exemplu, se stinge monitorul în timpul unui film). Prin OnNow aplicațiile îi pot specifica sistemului de operare când poate opri automat anumite componente.
- creșterea timpului de utilizare a bateriei. În cazul sistemelor portabile, aplicațiile pot afla dacă sistemul este alimentat la rețea sau la baterie. În cazul în care sistemul funcționează pe baterie se pot reduce în mod automat operațiile care nu sunt importante, pentru a menține sistemul autonom mai mult timp.

#### **4.4.2 Proiectarea sistemelor cu consum optim de energie**

Proiectarea sistemelor cu consum optim de energie necesită reducerea energiei la nivelul tuturor componentelor sistemului. Arhitectura hardware la nivelul sistemului are ca scop selecția și organizarea componentelor. Arhitectura software are ca scop definirea și selectarea sistemului de operare, aplicațiile software și compilatoarele. Interacțiunea dintre partea software și cea hardware poate afecta în mod substanțial consumul de energie la nivelul sistemului. Astfel, este necesar de a avea un mod rapid și ușor de a evalua consumul de energie al întregului sistem în timpul etapelor de proiectare a părților hardware și software.

##### **Proiectarea hardware**

Când se încearcă proiectarea unui sistem electronic, proiectanții propun un număr limitat de posibile arhitecturi și apoi le testează funcționalitatea, consumul de energie și performanța. În mod tradițional, proiectanții au dorit să obțină performanțe maxime fără a fi preocupați prea mult de consumul de energie. Astfel, întreaga metodologie de proiectare oferea arhitecturi capabile să obțină performanțe maxime tot timpul. Optimizarea proiectării, atât pentru performanțe cât și pentru consumul de energie, a devenit o prioritate în ultimul timp. Ca rezultat al acestei necesități, metodologia proiectării trebuia să fie schimbată pentru a include criteriul consumului de energie.

Cel mai clar mod de a evalua arhitectura unui sistem este acela de a construi mai întâi un prototip, dar această orientare nu modelează clar performanța și consumul de energie, de asemenea este o metodă înceată și foarte costisitoare. Alternativ, performanța poate fi evaluată folosind un simulator pentru un set de instrucțiuni, dar este un model limitat sau nu are suport pentru evaluarea consumului de energie.

Consumul de energie la nivelul procesorului este, în general, estimat prin intermediul analizei puterii consumate la nivelul instrucțiunilor. Această tehnică estimează consumul total de energie al unui program prin însumarea consumului fiecărei instrucțiuni executate de acel program. Costurile "instrucțiune cu instrucțiune" sunt prestabilite odată pentru totdeauna pentru fiecare procesor studiat. Modelul puterii la nivelul instrucțiunii poate fi argumentat prin luarea în calcul a efectului primului nivel de cache și a efectului inter-instrucțiune. O orientare nouă încearcă să evalueze efectele diferitelor cache-uri și configurații de magistrale folosind ecuații interliniare pentru a stabili caracteristicile, performanțele și consumul de energie al întregului sistem. Această orientare nu ia în seamă comportările neliniare în accesarea cache-ului pentru diferite configurații ale acestuia care sunt dependente de arhitectură și date în același timp.

Există un număr relativ mic de unelte legate de prototipuri ce estimează consumul de energie la nivelul procesorului, la nivelul memoriei cache și a memoriei principale. Consumul de energie la nivelul memoriei este estimat pe baza modelelor de cost per acces. Energia totală a sistemului se obține prin însumarea contribuției fiecărei componente. Dezavantajul acestei metode constă în faptul că interacțiunile dintre memoria sistemului sau periferice, și procesor nu sunt modelate.

O abordare mai recentă combină mai mulți estimatori ai puterii consumate într-un singur motor de simulare, astfel se asigură o simulare mai amănunțită a unor componente în timp ce altele se modelează la un nivel mai înalt. Această abordare se poate folosi și pentru interacțiunile dintre procesor și memorii (cache, memoria principală) în timpul rulării, dar cu posibilitatea prelungirii execuției datorită acestui fapt. Aceste prelungiri sunt cauzate de diferitele nivele de abstractizare a unor simulări sau datorită comunicărilor dintre componente.

În final, majoritatea orientărilor mai vechi nu se concentrază pe optimizarea duratei de viață a bateriilor, care este de fapt scopul optimizării energiei în cazul sistemelor portabile. De fapt, când bateria nu este luată în calcul în estimarea consumului de energie, pot apare erori semnificative.

### **Proiectarea software**

În trecut, în procesul de proiectare al aplicațiilor software, au fost luate în considerare doar performanța și funcționalitatea. În ultimii ani însă, performanței în funcție de consumul de energie i s-a acordat o tot mai mare atenție. Din moment ce programele nu sunt o realitate fizică materială, este important de analizat impactul pe care aplicațiile software îl are asupra consumului de energie în hardware. În plus, este nevoie de a analiza care dintre implementările software, cum ar fi cel de la nivelul sistemului sau de la nivelul aplicației și compilarea lor în cod mașină, este cea mai potrivită. De asemenea, este vital de a optimiza software-ul atât pentru performanță cât și pentru consumul de energie. Aceasta se poate face prin schimbarea codului sursă, direct, și cu ajutorul compilatoarelor.

Ideal ar fi să se implementeze sisteme de operare și aplicații de genul "energy-aware", care să semnalizeze componentelor hardware când este nevoie de ele și astfel să se asigure performanța optimă a consumului de energie. Momentan, software-ul comercial suportă doar performanța software-ului. Deoarece efectul software-ului asupra consumului de energie este foarte important, una dintre cerințele de bază în cadrul proiectării sistemelor este acela de a suporta optimizarea consumului de energie la nivelul codului sursă.

O serie de tehnici pentru optimizarea codului în acest sens au fost prezentate în trecut. Una dintre ele folosește modele ale energiei la nivelul instrucțiunii pentru a optimiza consumul, prin rearanjarea instrucțiunilor, reducerea operanzilor din memorie, o folosire mai eficientă a bancurilor de memorie și o serie de optimizări la nivelul procesorului. O altă optimizare se referă, de exemplu, la o mai bună proiectare a memoriei pentru a se potrivi cu cerințele software.

Toate aceste tehnici descrise mai sus se referă la optimizări la nivel de instrucțiune. Optimizările aduse compilatoarelor în acest sens sunt mult inferioare celor câștigate prin optimizarea codului sursă al aplicațiilor în vederea reducerii consumului de energie.

### 4.4.3 Benchmark termic

Benchmark-urile reprezintă un standard al măsurării sau al evaluării. Prin intermediul lor se măsoară performanța unui sistem sau a unui subsistem. Un benchmark pentru un sistem de calcul este, tipic, un program care execută un set strict de operații (workload) și oferă un rezultat prin intermediul căruia se pot stabili concluzii asupra performanței calculatorului testat. De obicei prin intermediul benchmark-urilor se testează viteza (cât de repede s-a executat setul de operații) sau se mai poate măsura numărul de operații care s-au executat în unitatea de timp. Rulând același program benchmark pe mai multe calculatoare se poate stabili o comparație între calculatoarele testate.

Ideal ar fi ca aceste comparații să fie făcute prin rularea programelor fiecărui utilizator pentru a se cunoaște cu exactitate nevoile fiecărui utilizator în parte. Totuși se pot folosi benchmark-uri standardizate ca punct de referință. În mod ideal, un benchmark standardizat trebuie să fie portabil și să ruleze pe orice platformă de care un utilizator ar putea fi interesat. Înainte de a se lua în calcul rezultatele unui benchmark trebuie să se știe corelația dintre cerințele utilizatorului și ceea ce benchmark-ul oferă. Setul de operații ce caracterizează un benchmark trebuie să aibă aceleași caracteristici cu cerințele utilizatorului.

Pe lângă benchmark-urile pentru performanță se pot imagina și alte aplicații ce vizează alți parametri ai unui sistem de calcul. Un prim parametru ce poate fi vizat datorită importanței lui este temperatura.

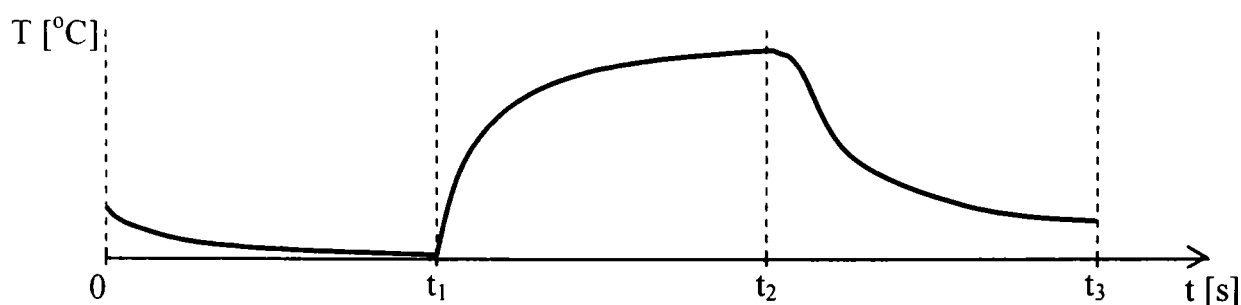


Figura 4.32 – Semnătura unui benchmark termic

#### Definiția 4.2

Un benchmark termic este definit ca fiind o aplicație de tip benchmark ce pune în evidență caracteristicile termice ale unui sistem sau componente a sistemului la aplicarea anumitor tipuri de stimuli.

Spre deosebire de un benchmark pentru estimarea performanței, în care se contorizează timpul de execuție al unui set de operații, benchmark-ul termic presupune aplicarea unui anumit tip de stimulului sistemului sau componentei verificate, un anumit

interval de timp. Pe perioada aplicării stimulilor se monitorizează temperatura sistemului sau componentei, iar rezultatul benchmark-ului este semnătura termică achiziționată la aplicarea lui sau o anumită valoare extrasă din această semnătură.

Benchmark-ul termic vizează modul în care sistemul sau componenta se încălzește la aplicarea stimulilor și modul în care se răcește după aplicarea lor. Din acest motiv benchmark-ul termic este compus din trei intervale (figura 4.32):

- un prim interval,  $(0, t_1]$ , în care componenta sau sistemul sunt lăsate neutilizate pentru a se atinge starea de echilibru termic. În această fază temperatura componentei ar trebui să ajungă la o valoare ce se stabilește de obicei când componenta nu este utilizată (este în starea idle). În acest interval se asigură ca nici o altă aplicație să nu utilizeze componenta respectivă;
- în cel de-al doilea interval  $(t_1, t_2]$  se aplică componentei repetat un anumit set de operații. În acest interval temperatura componentei crește ajungând la o anumită valoare maximă, valoare ce diferă în funcție de tipul de stimuli aplicat și diferă de la un sistem la altul. În acest interval este pusă în evidență influența pe care diferitele tipuri de instrucțiuni o au asupra componentei;
- al treilea interval  $(t_2, t_3]$ , este folosit pentru răcirea componentei. În acest interval temperatura componentei scade până la o anumită valoare, punându-se în evidență eficiența mecanismelor de răcire cu care este prevăzută componenta.

### Definiția 4.3

Un benchmark termic pentru procesor este o aplicație ce pune în evidență caracteristicile termice ale procesorului și ale mecanismului de răcire atașat lui la aplicarea anumitor tipuri de instrucțiuni.

În cele ce urmează voi prezenta câteva tipuri de benchmark-uri pentru procesor și în final utilitatea lor.

### Test 1

Un benchmark ar trebuie să aibă rezultate asemănătoare pe sistemele de același tip și la diferite execuții pe un același sistem să se obțină același rezultat. Din acest motiv un prim test a fost acela de a stabili repetabilitatea rezultatelor unui benchmark termic pe același sistem. Din testele efectuate, semnăturile termice pentru un benchmark termic executat repetat pe același sistem, au ieșit foarte asemănătoare (două dintre ele sunt prezentate în figura 4.33).

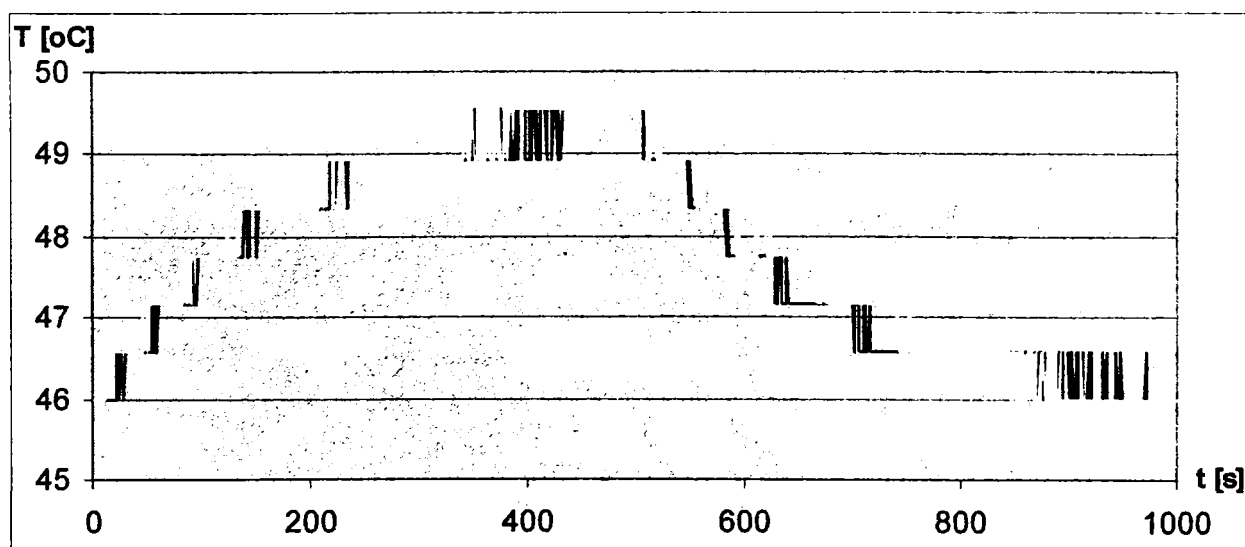


Figura 4.33 – Două semnături pentru un benchmark executat pe același sistem

S-au facut în continuare un număr de 10 teste pe același sistem, s-au mediat semnăturile termice și s-a obținut semnătura standard pentru benchmark-ul aplicat (figura 3.34).

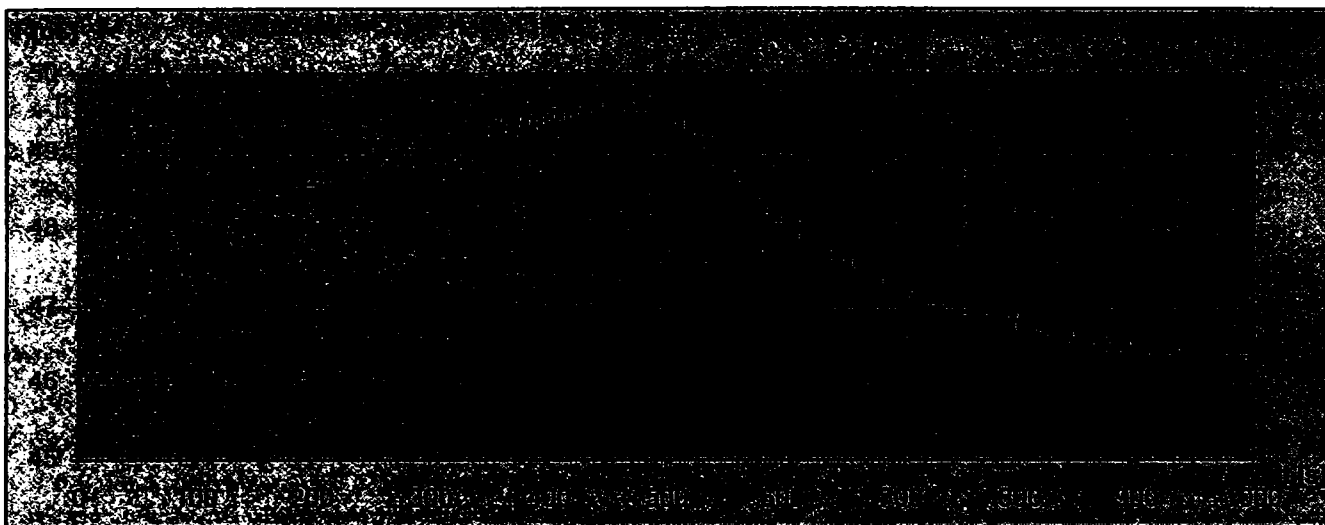


Figura 4.34 – Semnătura termică standard pentru un sistem



Figura 4.35 – Încărcarea procesorului la determinarea semnăturii unui benchmark

Testele au fost realizate pe un sistem cu procesor AMD Athlon la 1200 MHz, 256 MB memorie, chipset VIA686 și sistem de operare Windows XP.

## Test 2

În cel de-al doilea set de teste s-a executat aplicația benchmark termice pe sisteme identice sau asemănătoare. S-a rulat același benchmark pe două sisteme identice d.p.d.v. hardware și un al treilea sistem asemănător cu primele. Testele au fost efectuate la aceeași temperatură a camerei. În figura 4.36 se poate observa că cele trei semnături au forme asemănătoare însă diferă nivelul mediu de temperatură. Pe primul sistem din grafic sunt instalate multe aplicații, temperatura normală de lucru crescând destul de mult. Cel de-al doilea sistem, identic cu primul este nou instalat doar cu sistemul de operare. Cel de-al treilea sistem este configurat ca server de baze de date și are instalat sistemul de operare Windows 2000 Server, spre deosebire de primele, pe care este instalat Windows XP.

Se poate concluziona că și sistemul de operare și numărul serviciilor instalate au o influență asupra temperaturii medii de lucru a procesorului.



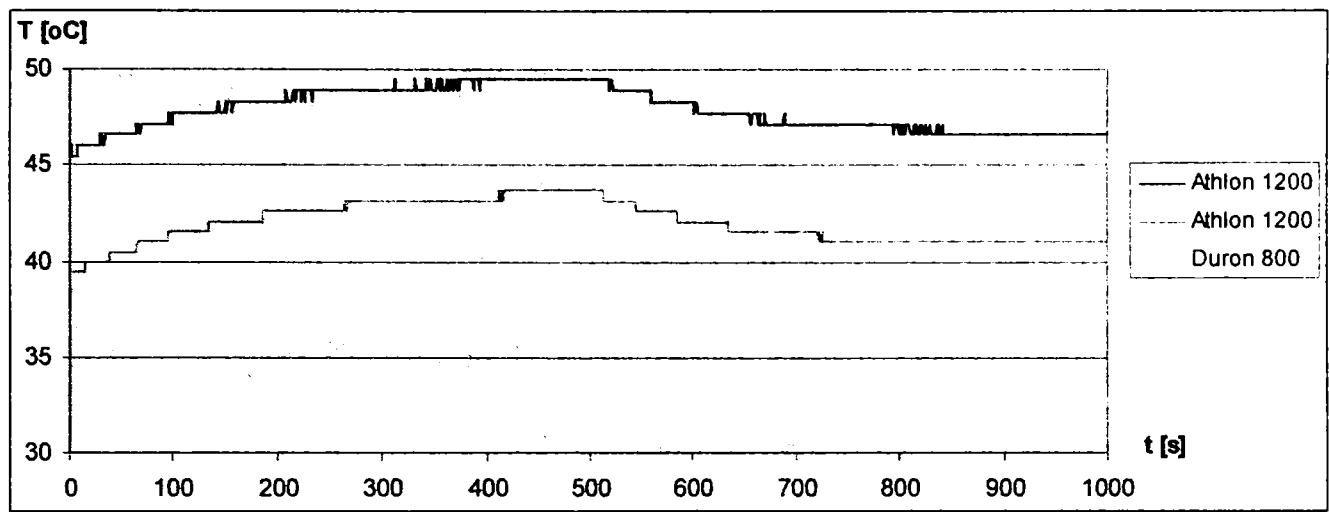


Figura 4.36 – Semnăturile pentru două sisteme identice și un al treilea asemănător

Sistemul 1 – procesor AMD Athlon 1200 MHz, memorie 256 MB, chipset VIA686

Sistemul 2 – procesor AMD Athlon 1200 MHz, memorie 256 MB, chipset VIA686

Sistemul 3 – procesor AMD Duron 800 MHz, memorie 256 MB, chipset VIA686

### Test 3

Testele au continuat cu aplicarea benchmark-urilor pe sisteme diferite (figura 4.37). La compararea semnăturilor termice achiziționate la aplicarea unui benchmark termic se urmărește valoarea maximă atinsă în timpul stimulării procesorului, valoarea minimă la care se răcește procesorul după aplicarea stimulilor și valoare medie pe fiecare din cele două intervale.

Din figura 4.37 se poate observa că la aplicarea aceluiași stimul pe sisteme diferite, răspunsul termic este diferit. Unele procesoare se încălzesc mai repede și mai mult iar altele se încălzesc mai lent, atingând o valoare mai mică de temperatură.

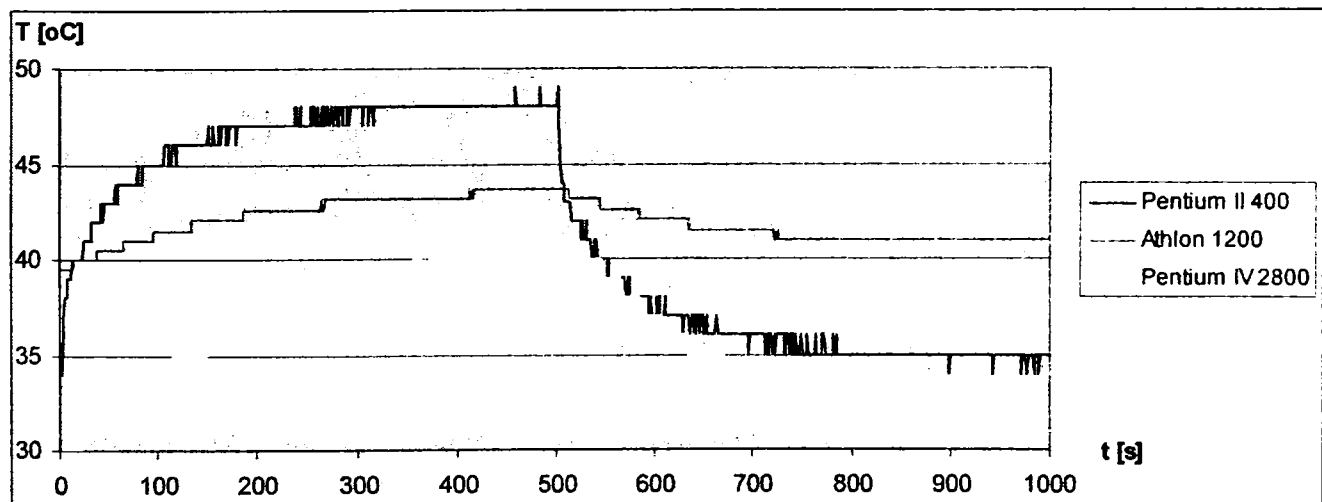


Figura 4.37 – Semnăturile pentru trei sisteme diferite

Sistemul 1 – procesor Intel Pentium II 400 MHz, memorie 128 MB, chipset Intel 82801AA/ICH, senzor ADM 1025;

Sistemul 2 – procesor AMD Athlon 1200 MHz, memorie 256 MB, chipset VIA VT82C686, senzor VIA 686;

Sistemul 3 – procesor Intel Pentium IV 2800 MHz, memorie 512 MB, chipset Intel 82801EB/ICH5, senzor ADM 1027.

#### Test 4

În cel de-al patrulea test s-au încercat diferite tipuri de stimuli pentru procesor pentru a evidenția efectul lor asupra temperaturii. S-au implementat trei tipuri de benchmark-uri termice: unul pentru operațiile cu numere întregi, unul pentru lucrul cu memoria și ultimul pentru operațiile în virgulă flotantă. În figura 4.38 este prezentat rezultatul celor trei tipuri de benchmark-uri.

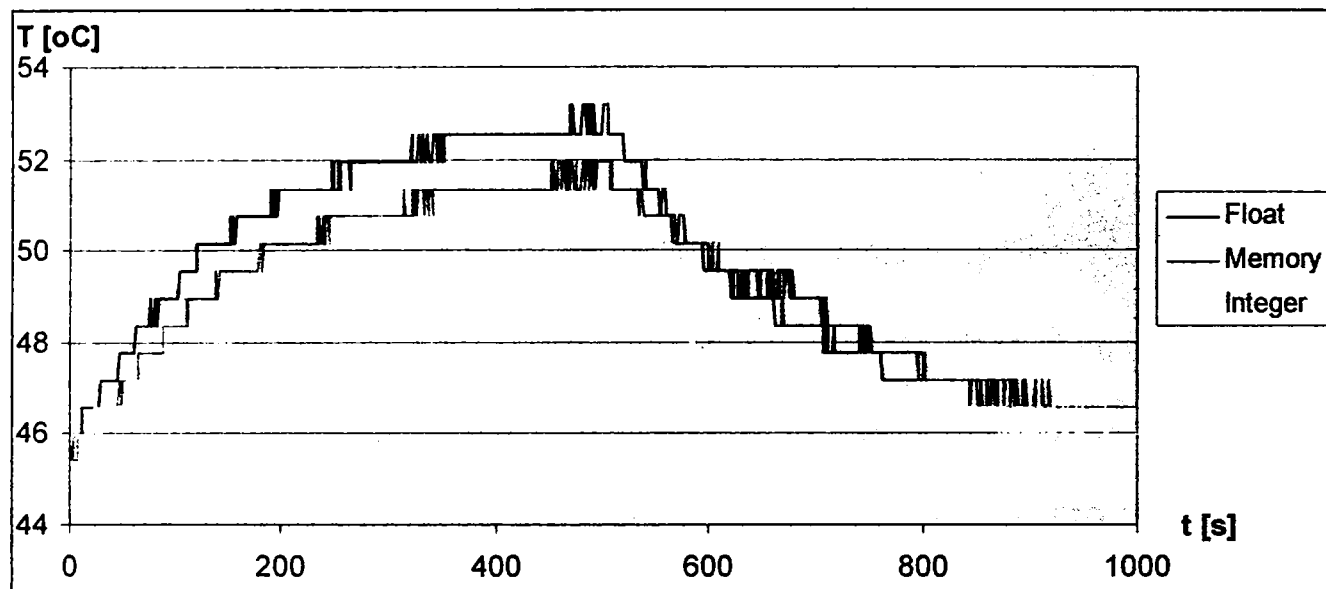


Figura 4.38 – Trei tipuri de benchmark-uri termice

Benchmark-urile termice pot fi utilizate atât pentru a pune în evidență defectele sau dezavantajele unui sistem, cât și pentru a testa din punct de vedere termic un sistem pentru diferite tipuri de aplicații. Defectele ce pot fi puse în evidență sunt: lipsa senzorilor de temperatură sau neconectarea lor la magistrala SMBus sau ISA, nefuncționarea ventilației și chiar se pot pune în evidență mecanismele de răcire inadecvate pentru procesorul curent.

#### 4.4.4 Controlul puterii și temperaturii la nivelul aplicațiilor utilizator

Consumul de putere a devenit una din preocupările principale în procesul de proiectare al sistemelor electronice digitale, datorită popularității din ce în ce mai mari a sistemelor portabile și a interesului manifestat față de acțiunea asupra mediului a sistemelor desktop și a serverelor. Controlul și reducerea consumului de energie este necesară în primul rând pentru a putea prelungi durata de viață a bateriilor ce însoțesc sistemele portabile și de asemenea de a reduce costurile operaționale și impactul asupra mediului înconjurător ale sistemelor de tip desktop. Căldura disipată este, pe de altă parte, o problemă majoră atât în cazul sistemelor portabile cât și a celor de tip desktop. Mecanismele de răcire necesare reducerii temperaturii componentelor unui sistem portabil sunt și ele consumatoare de energie, ducând la reducerea duratei de viață a bateriei. Proprietățile bateriilor s-au îmbunătățit foarte puțin, cu un factor de 2-4 în ultimii 30 de ani, în timp ce cererile de viteză de calcul tot mai mare a crescut drastic în aceeași perioadă de timp. Din acest motiv este binevenită implementarea unui mecanism unitar pentru reducerea puterii și a temperaturii unui sistem. Și în cele ce urmează s-a vizat tot procesorul, el fiind una din componentele cu consum sporit într-un sistem iar mecanismul se poate extinde mai apoi și la alte tipuri de componente.

Din cele prezentate în subcapitolul 4.4.2, se poate spune că majoritatea mecanismelor de control ale puterii și temperaturii sunt implementate la nivelul sistemului de operare și la

cel fizic al componentelor. La fel și articolele ce vin cu propuneri noi vizează tot aceste niveluri și sunt orientate ori pe reducerea puterii sau pe reducerea temperaturii. Singurele articole cu aplicabilitate practică imediată, găsite pe tema controlului puterii la nivelul aplicațiilor utilizator sunt propuse de firmele Microsoft și Intel în [OnNow2001, INTEL2002\_1].

Implementarea unui mecanism de reducere a puterii sau temperaturii la nivelul aplicațiilor utilizator este dificilă din mai multe motive: posibilitatea redusă de control a resurselor fizice de către aplicațiile utilizator, lipsa unei interfețe standard de comunicație cu dispozitivele fizice, mecanismele de reducere a puterii și temperaturii, precum și modul de apelare al lor, sunt specifice pentru fiecare dispozitiv în parte, comportamentul bateriei este dificil de simulat și de prevăzut, etc.

Avantajul implementării mecanismului de control la nivelul aplicațiilor utilizator ar fi creșterea semnificativă a eficienței și posibilitatea de intervenție a utilizatorului. În mod normal, utilizatorul este cel mai în măsură să stabilească modul de lucru dorit pentru aplicațiile de care are nevoie.

Propunerea pe care o voi detalia în continuare este proiectarea, implementarea și testarea unui mecanism unitar de reducere a temperaturii și puterii pe baza monitorizării parametrilor sistemului (frecvență, tensiuni de alimentare, temperatură, viteză ventilatoare, tip alimentare, procentajul și timpul de încărcare al bateriei) și luarea de decizii la nivelul aplicațiilor utilizator dependent de acești parametri și de configurația specificată de utilizator.

Scopul acestui mecanism este acela de a oferi aplicațiilor utilizator o modalitate cât mai simplă de a obține informații despre parametrii curenți ai sistemului, influența pe care aplicația o are asupra acestor parametri și adaptarea automată, pe baza specificațiilor utilizatorului, pentru a ajunge la parametrii doriți de acesta. Un alt scop ulterior ar fi acela de implementare a acestui mecanism într-un cadru (framework) pentru aplicațiile cu facilități de control a puterii și/sau temperaturii.

Arhitectura mecanismului de control propus este prezentată în figura 4.39. PowerSpy este aplicația de monitorizare prezentată în subcapitolul 4.3.3. Acest gen de aplicație poate fi integrată între modulele utilizator ale sistemului de operare (de exemplu poate apărea în Control Panel în Windows).

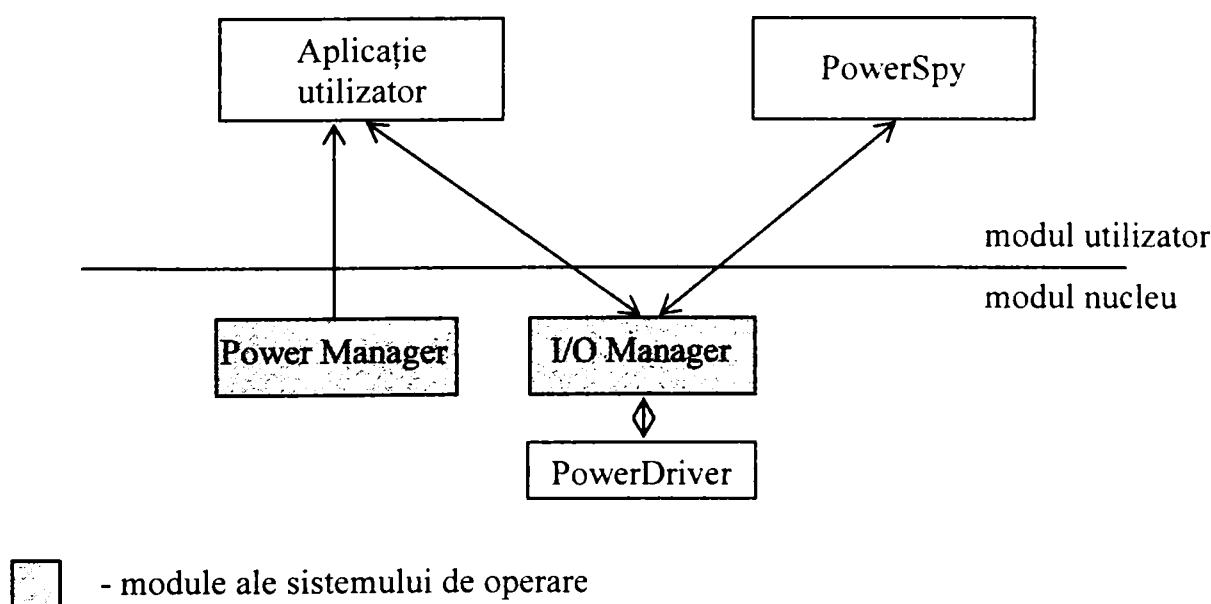


Figura 4.39 – Arhitectura mecanismului de control

PowerDriver este un driver Windows cu rol de accesare și control al componentelor hardware ale sistemului. Rolul acestui driver este acela de a scana și detecta senzorii prezenți în sistem, pornirea unui fir de execuție independent ce achiziționează periodic valorile dorite ale parametrilor sistemului: temperaturi, tensiuni de alimentare, viteze ventilatoare, frecvență procesor, etc. Acest driver oferă o interfață de programare (API) standard pentru toate aplicațiile utilizator prin care acestea pot afla valoarea curentă a parametrilor sistemului.

Power Manager este un modul al sistemului de operare Windows ce se ocupă de controlul puterii. Acest modul trimite mesaje specifice către drivere și către aplicațiile utilizator, conform arhitecturii OnNow. Pe baza acestor mesaje o aplicație poate fi doar reactivă și nu proactivă, așa cum este modelul propus de noi.

Monitorizând descărcarea bateriei unui sistem portabil la diferite tipuri de aplicații (figura 4.40) se poate observa o relație de dependență între durata de viață a bateriei și încărcarea procesorului.

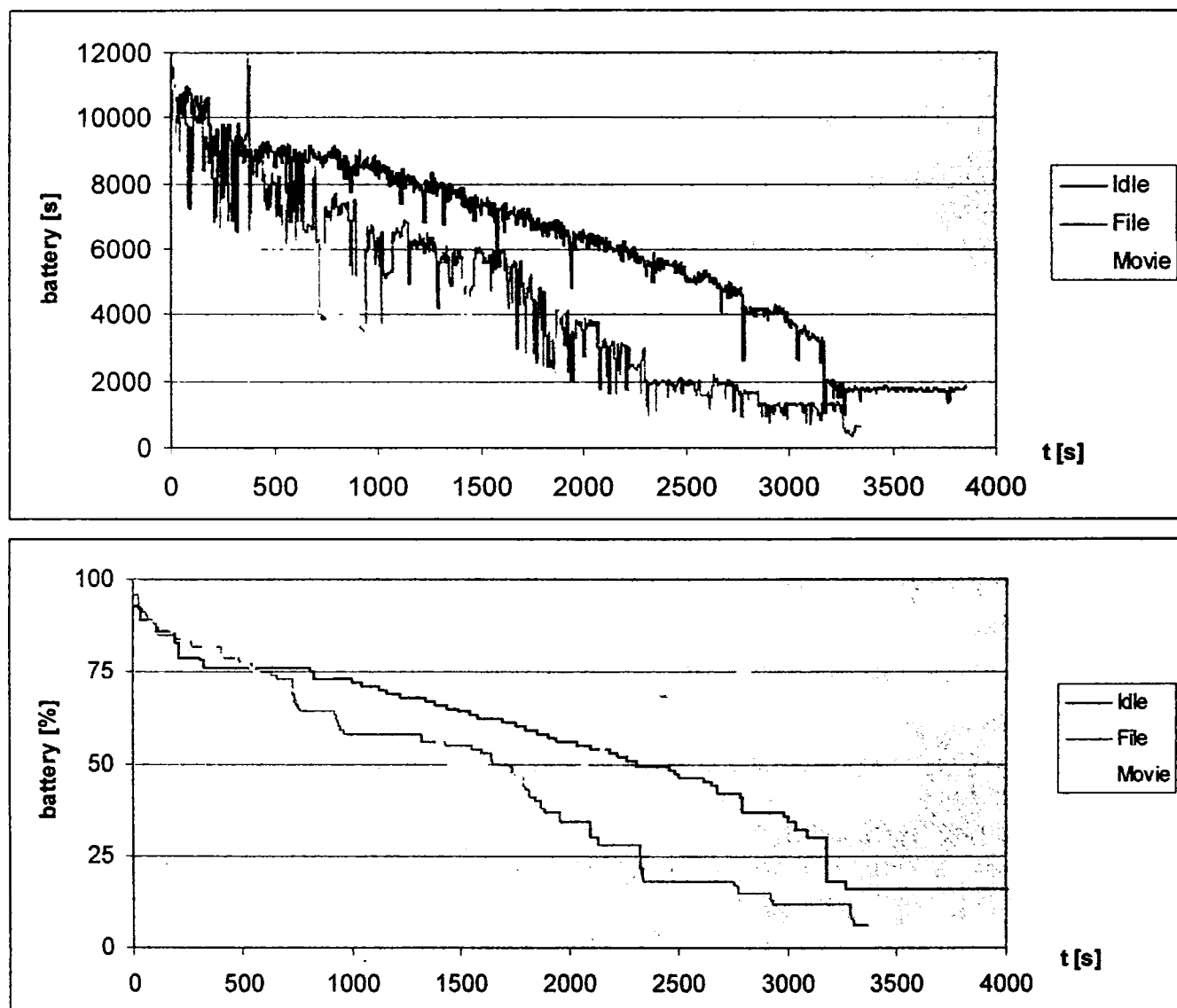


Figura 4.40 – Durata de viață a bateriei la diferite tipuri de utilizare a sistemului

Aplicația benchmark termic prezentată în subcapitolul precedent, poate fi utilizată și pentru studiul descărcării bateriilor sistemelor portabile la diferite tipuri de operații efectuate de către procesor. În graficele din figura 4.41 s-a prezentat timpul și procentajul de descărcare al bateriei la aplicarea a trei tipuri de operații de stimulare a procesorului (numere întregi, transferuri cu memoria și numere în virgulă flotantă).

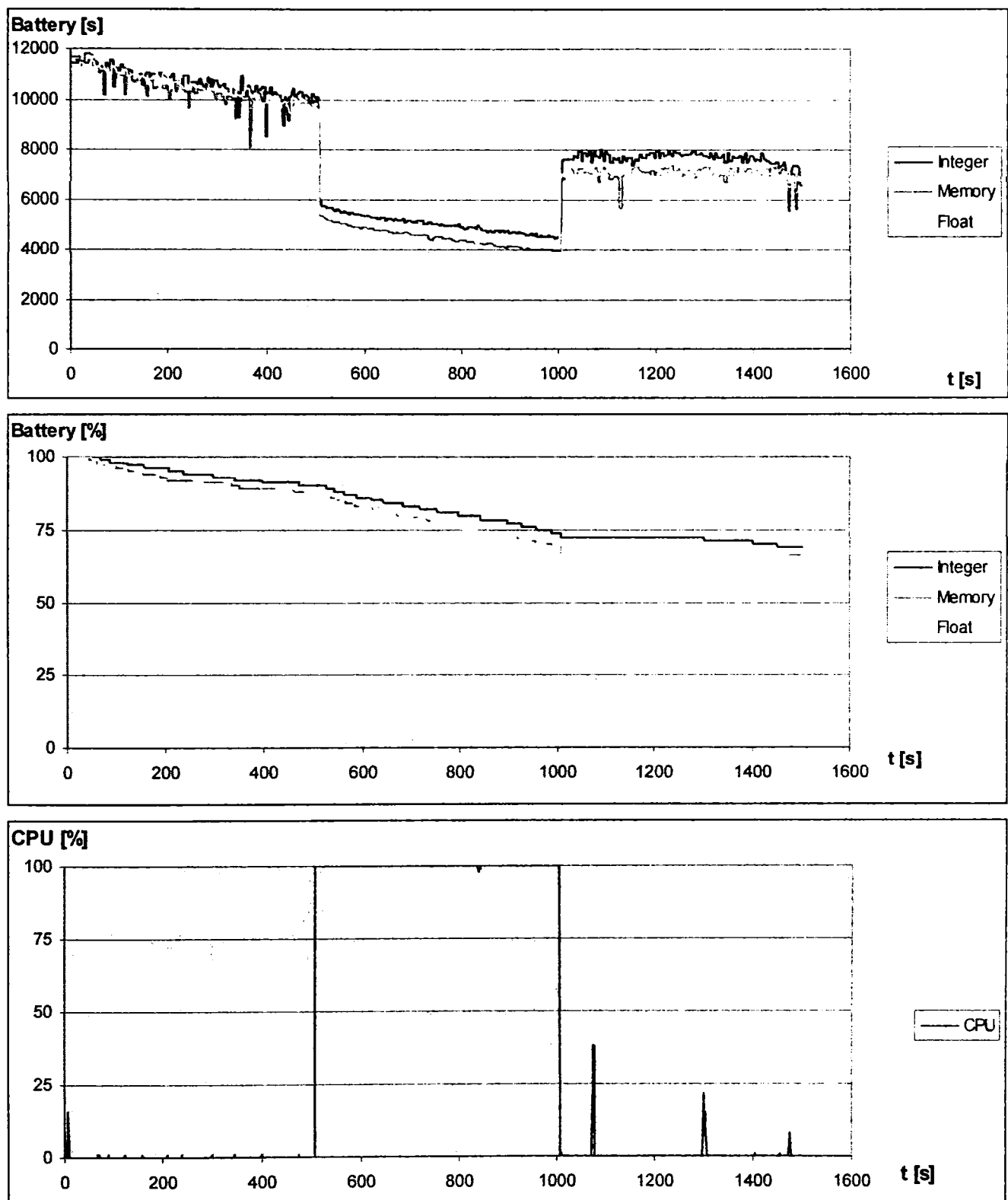


Figura 4.41 – Benchmark-urile termice aplicate pentru studiul descărcării bateriei unui sistem portabil

În continuare s-a încercat punerea în evidență a modului în care încărcarea procesorului influențează timpul de descărcare al bateriei. Pentru aceasta s-au folosit aceleași operații ca la benchmark-ul termic, doar că s-a urmărit să implementez același număr de operații cu încărcări diferite ale procesorului. În mod uzual, execuția unei anumite secvențe de program se realizează la capacitatea maximă a procesorului (procesorul este încărcat 100%). Pentru a obține o încărcare mai mică a procesorului la execuția unei anumite secvențe de cod se pot introduce stări de așteptare până se ajunge la un anumit nivel de utilizare a timpului procesorului.

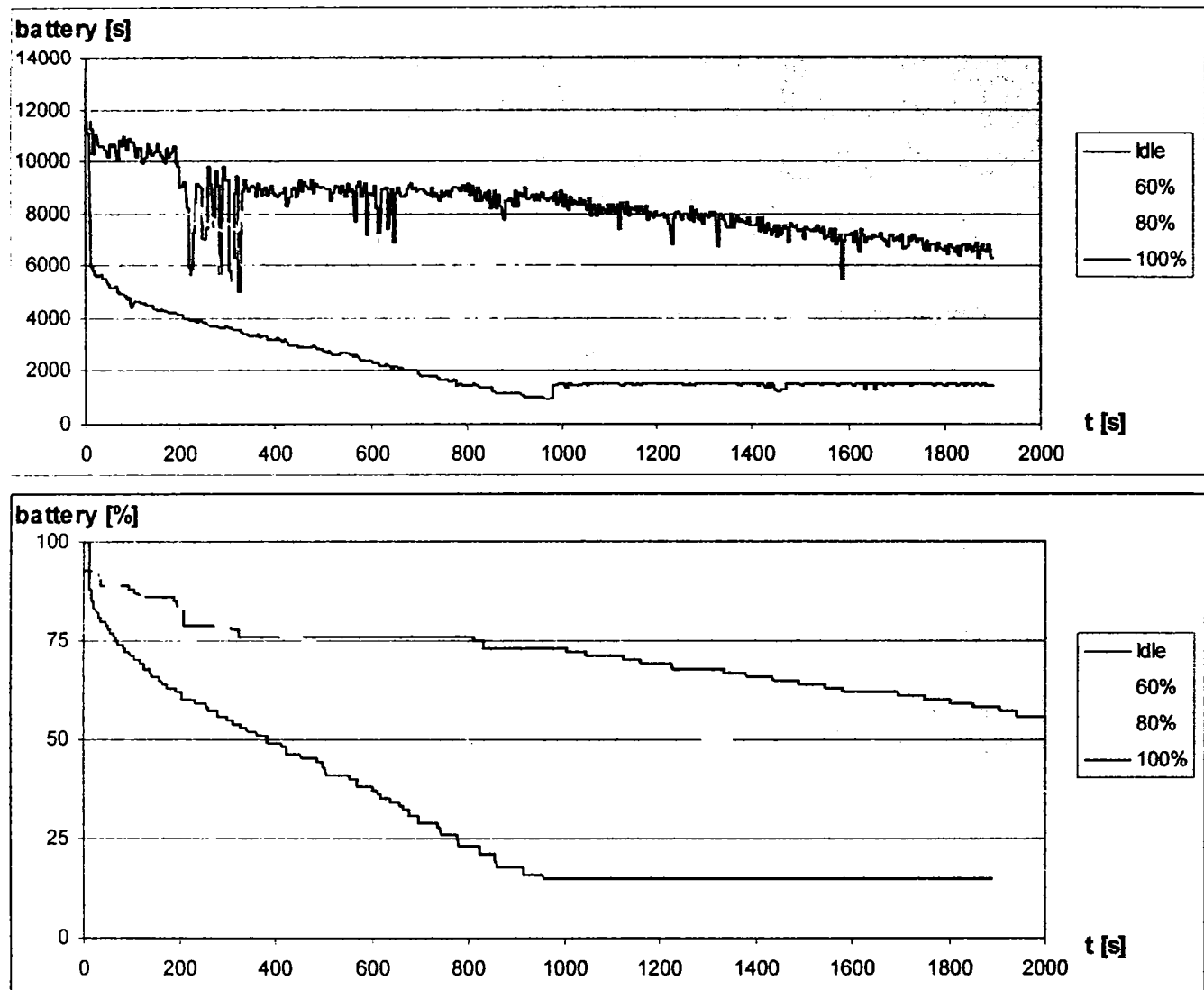


Figura 4.42 – Un task termic pentru operații cu numere întregi executat la diverse încărcări ale procesorului

Din figura 4.42 se poate observa că dacă o aceeași aplicație folosește în mod diferit procesorul pentru a realiza aceleași operații se poate obține o creștere a duratei de viață a bateriei. Toate cele patru teste au rulat aproximativ 30 minute. După trecerea celor 30 de minute, în cazul în care nu s-a rulat nici un task pe sistem, bateria a scăzut la 60% (au mai rămas aproximativ 1 oră și 40 de minute de funcționare). Dacă în intervalul de 30 de minute s-a rulat task-ul selectat la o încărcare de 60% din capacitatea procesorului, bateria s-a redus la 44% (au mai rămas aproximativ 1 oră și 20 minute de funcționare), adică cu 22% mai puțin timp decât în cazul precedent. Dacă task-ul a fost executat la o încărcare de 80% din capacitatea procesorului, bateria s-a descărcat la 37% (și a mai rămas aproximativ 1 oră de funcționare), cu 12% mai puțin decât în cazul precedent. Iar dacă task-ul s-a executat la capacitatea maximă a procesorului, bateria a scăzut la 15% (și au mai rămas aproximativ 25 minute de funcționare).

Operații	Baterie [%]	Pierdere [%]	Baterie [s]	Pierdere [%]
Idle	60	0	6327	0
Integer 60%	44	16	4929	22
Integer 80%	37	23	4137	34.5
Integer 100%	15	45	1487	76.5

Tabelul 4.6 – Compararea execuției task-urilor

Utilizarea procesorului sub capacitatea maximă poate duce la o creștere a duratei de viață de la câteva procente și până la aproximativ 50%, dar cu o reducere a timpului de execuție al aplicației. Teoretic ar fi trebuit ca energia totală consumată în fiecare din testele efectuate să fie aceeași, însă la o putere mai mică același task consumă mai puțină energie (acest lucru este prezentat schematic în figura 4.43).

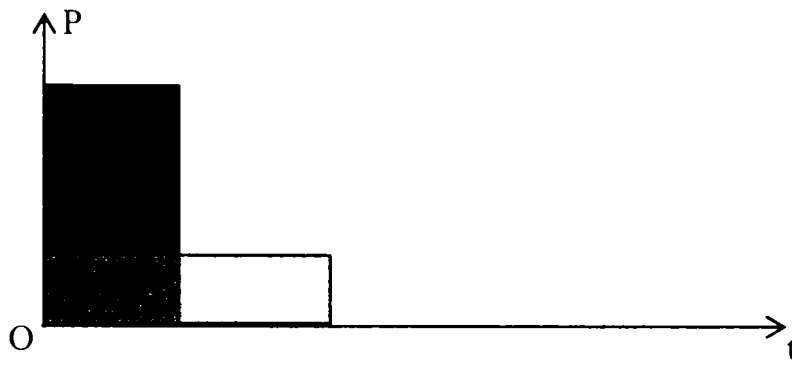


Figura 4.43 – Energia consumată de același task rulat la viteze diferite

Dacă se efectuează același test, însă în loc de durata de viață a bateriei se monitorizează temperatura, se observă că temperatura are aceeași creștere indiferent de procentajul din procesor utilizat (figura 4.44).

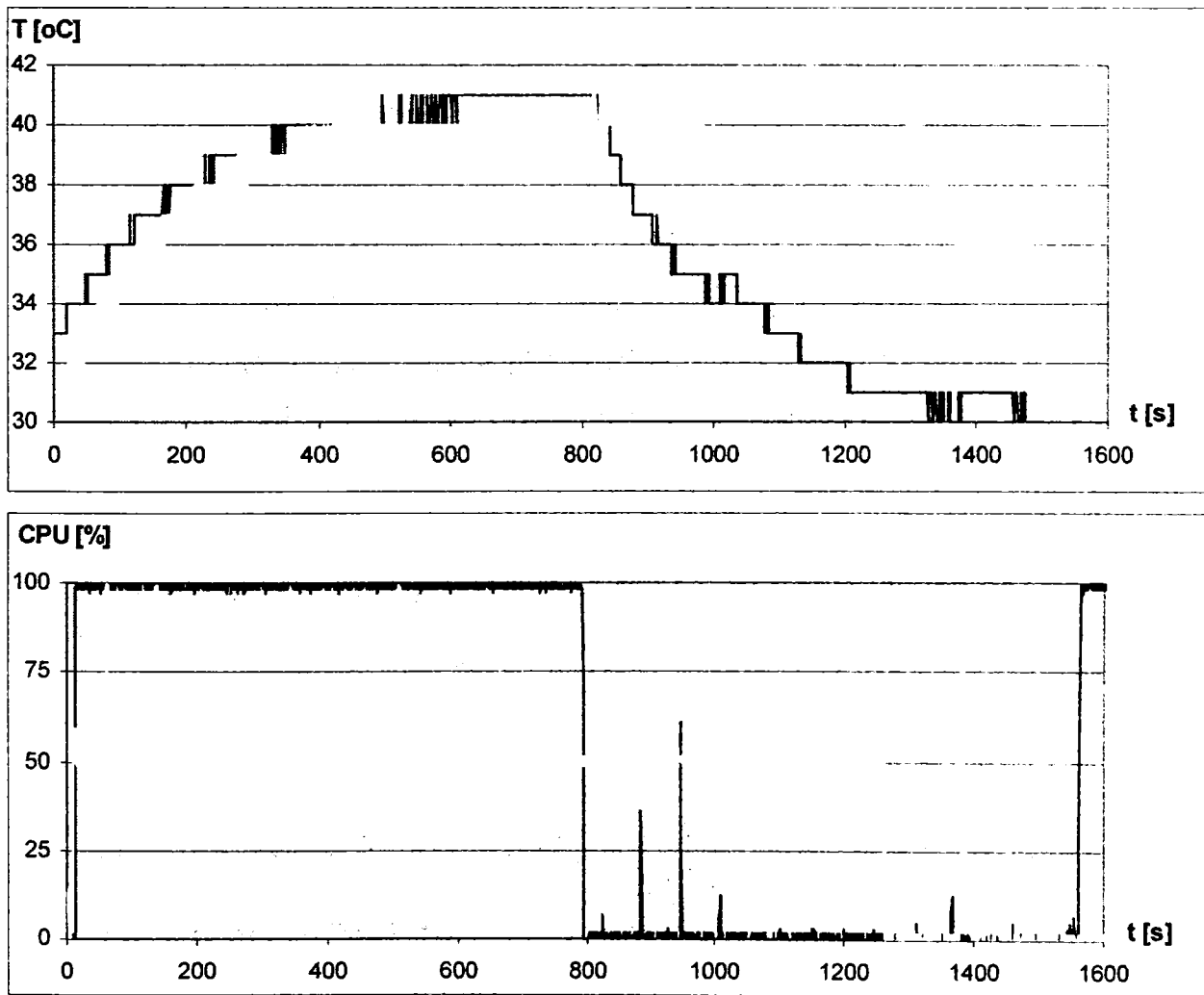


Figura 4.44 – Temperatura disipată de același task executat la viteze diferite

Temperatura disipată de procesor la execuția unei secvențe de cod nu se modifică dacă se introduc stări de așteptare de interval mic pentru controlul încălzirii procesorului. Această observație pune în evidență câteva lucruri:

- semnătura termică a unei secvențe de cod este validată și prin acest test datorită faptului că ea nu se modifică la încălzări diferite ale procesorului. Semnătura termică doar se scalează pe perioade diferite de timp;
- introducerea de stări de așteptare de dimensiune mică nu este suficientă pentru răcirea corespunzătoare a procesorului;
- în loc de introducerea de stări de așteptare sau împreună cu ele, pentru reducerea temperaturii datorate unui anumit task trebuie folosite mecanismele de răcire specifice componentei pentru care se dorește controlul temperaturii.

Pentru ca procesorul să nu se încălzească foarte mult trebuie introduse stări de întârziere mai mari, pentru ca procesorul să se poată răci (figura 4.45).

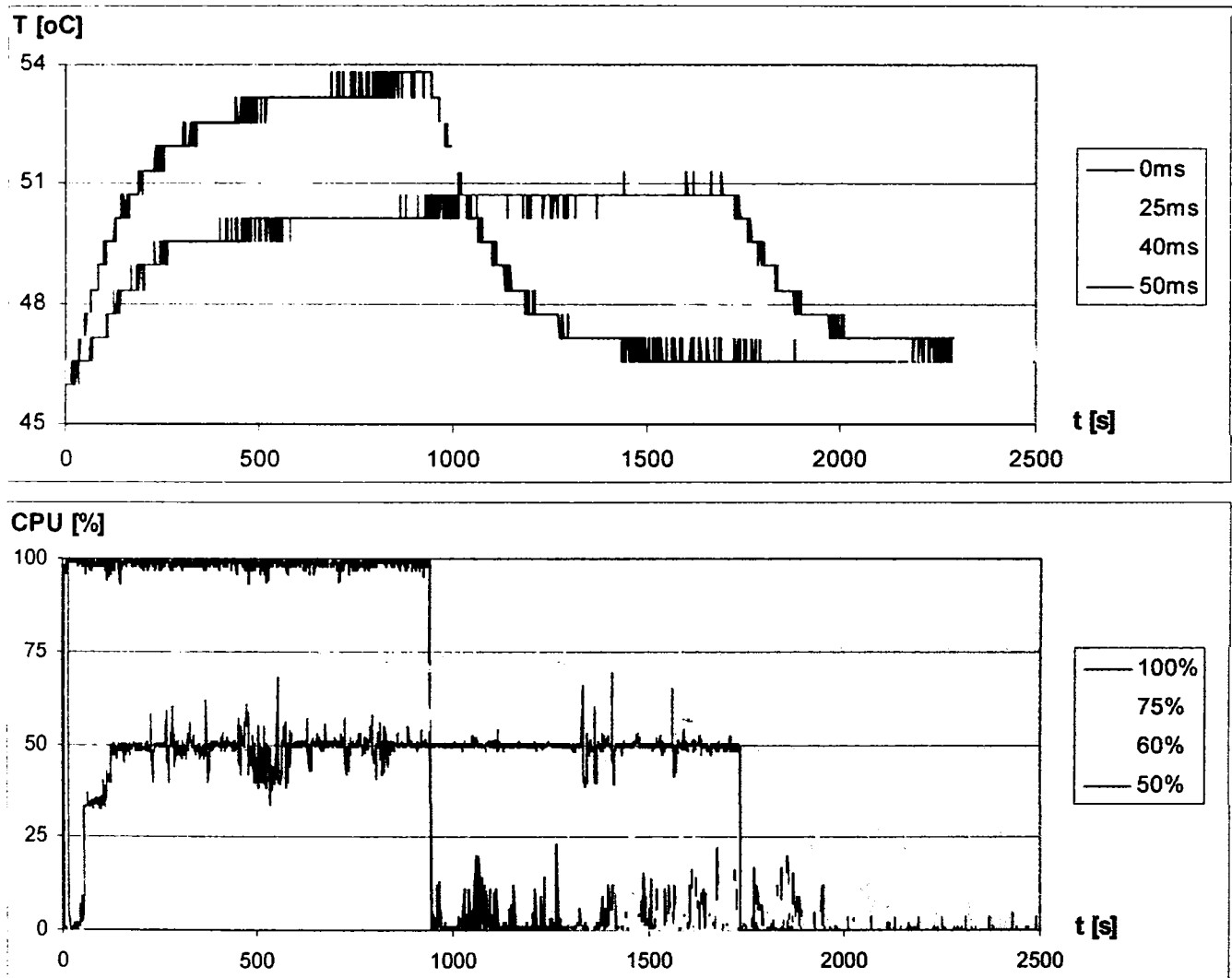


Figura 4.45 – Temperatura în funcție de timpii de întârziere

Operații	Întârziere [ms]	T maximă [ $^{\circ}\text{C}$ ]	Reducere T [%]
Float 100%	0	53.78	0
Float 75%	25	53.16	1.15
Float 60%	40	51.33	4.56
Float 50%	50	50.73	5.67

Tabelul 4.7 – Compararea execuției task-urilor



Un ultim test a fost efectuat pe o aplicație reală utilizată în practică. S-a implementat mecanismul de control al consumului pe o aplicație de baze de date pentru analiza costurilor de telefonie. Procesarea unui număr de 20000 de apeluri se face într-un interval de timp de aproximativ 20 minute. Descărcarea bateriei la execuția unei procesări la încărcări diferite ale procesorului este prezentată în figura 4.46. După procesarea la 60% din CPU s-a obținut un câștig la nivel de încărcare al bateriei de aproximativ 15% cu o dublare a timpului de execuție.

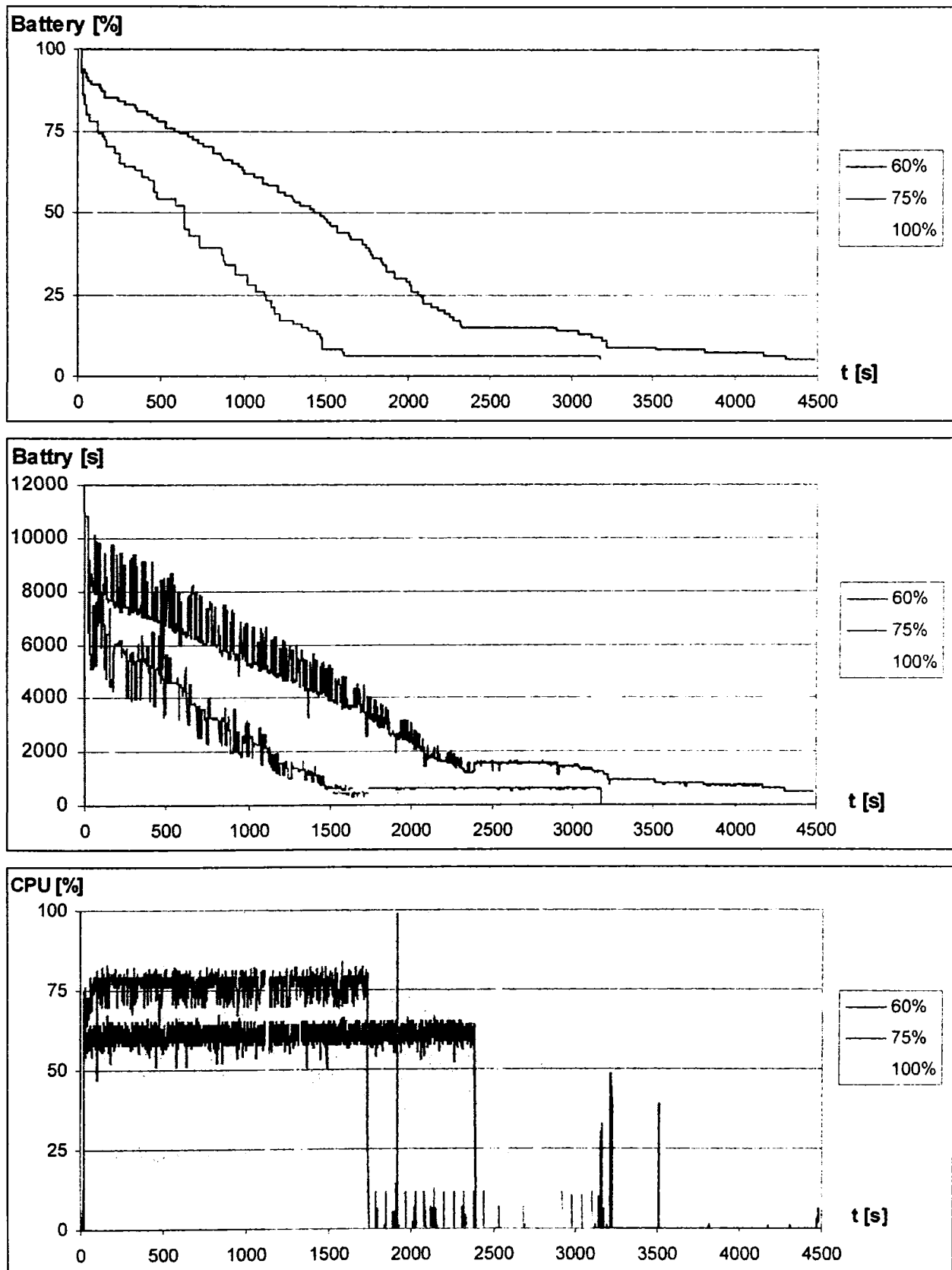


Figura 4.46 – Consumul unei aplicații de baze de date

Același test a fost aplicat în continuare și pentru studiul temperaturii, rezultatele obținute fiind prezentate în figura 4.47. În acest exemplu au fost executate aceleași operații pe același set de date la încărcări diferite ale procesorului: 100%, 75% și 60%. În acest caz au fost folosite și mecanismele hardware de răcire de la nivelul procesorului. Prin controlul temperaturii la nivelul aplicației prin autoadaptarea încărcării procesorului se obține o reducere cu 2.5% și 10% pentru o încărcare a procesorului de 75% și respectiv 60%. Prin activarea mecanismelor hardware de la nivelul procesorului la nivelul aplicației față de activarea lor de către SO se mai obține o reducere a temperaturii cu aproximativ 10%.

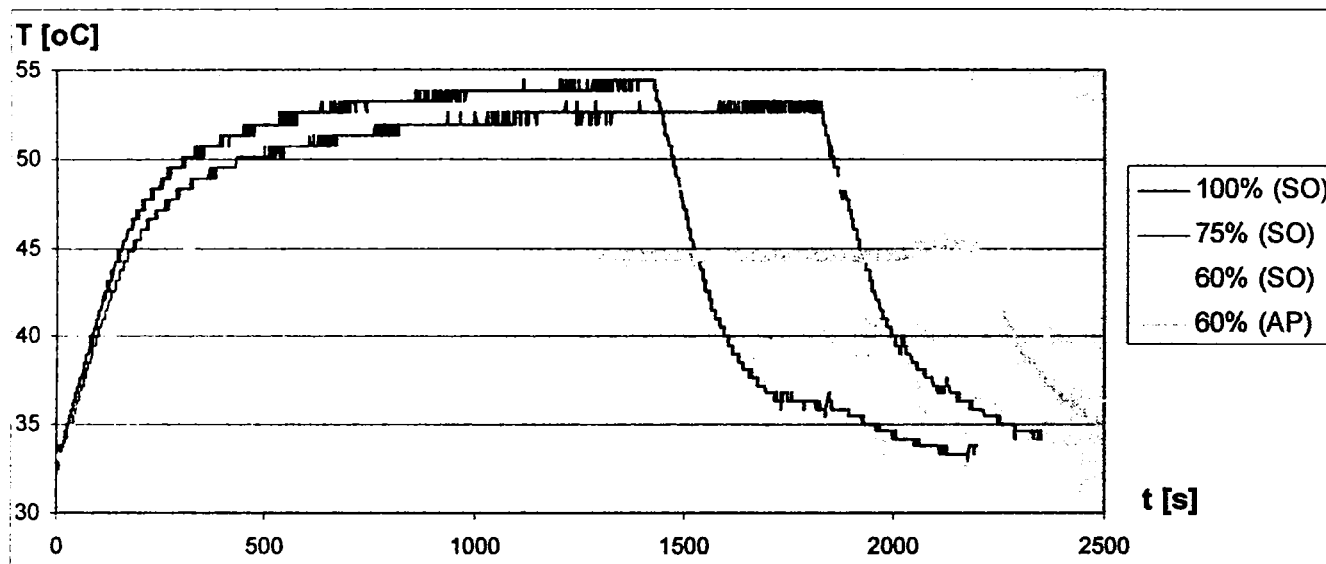


Figura 4.47 – Temperatura unei aplicații de baze de date

Pentru aplicarea și evaluarea conceptului de benchmark termic și cel de task termic a fost implementată o aplicație de test (figura 4.48).

Name	Type	Priority	Value	Status	Time	Temp
Task 1	Floating point	Normal		Finalized	00:20:37	
Task 2	Memory	Normal		Aborted	00:00:30	
Benchmark 3	Integer	Normal	0	Execution		
Task 4	Counter	Normal		Finalized	00:01:40	
Task 5	Random	Normal	Random 0	Execution		

Figura 4.48 – Aplicația de benchmark termic

#### Definiția 4.4

Un task termic (Thermal-Aware Task) se definește ca fiind o secvență de cod ce are posibilitatea să își autocontroleze nivelul de temperatură al procesorului.

Un task termic este format din secvența de cod utilă (implementarea funcționalității task-ului) și din modulul de monitorizare și control al temperaturii și eventual, al puterii disipate. Secvența de cod utilă, în aplicația de test, a fost aleasă ca fiind formată tot din aceleași operații ca la secvența de stimulare a procesorului dintr-un benchmark termic: operații cu numere întregi, operații cu memoria și operații în virgulă flotantă.

Modulul de control al temperaturii, pe lângă secvența de cod de control mai posedă și o interfață pentru utilizator, prezentată în figura 4.49.

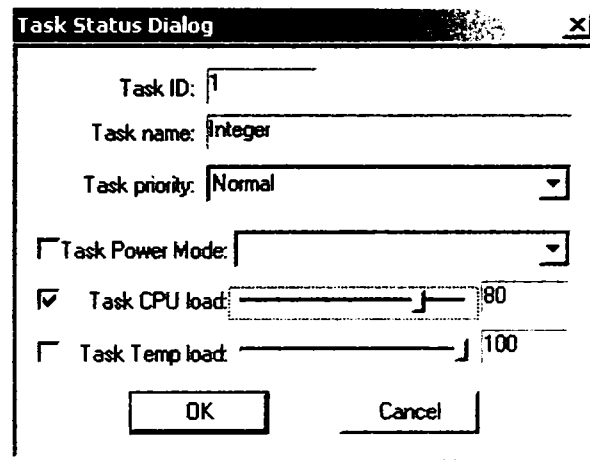


Figura 4.49 – Parametrii de control ai unui task termic

Diagrama de clase ce formează nucleul aplicației de test este prezentată în figura 4.50. Task-urile ce trebuie să ofere facilități de control al puterii sau temperaturii sunt derivate din clasa CThermalTask.

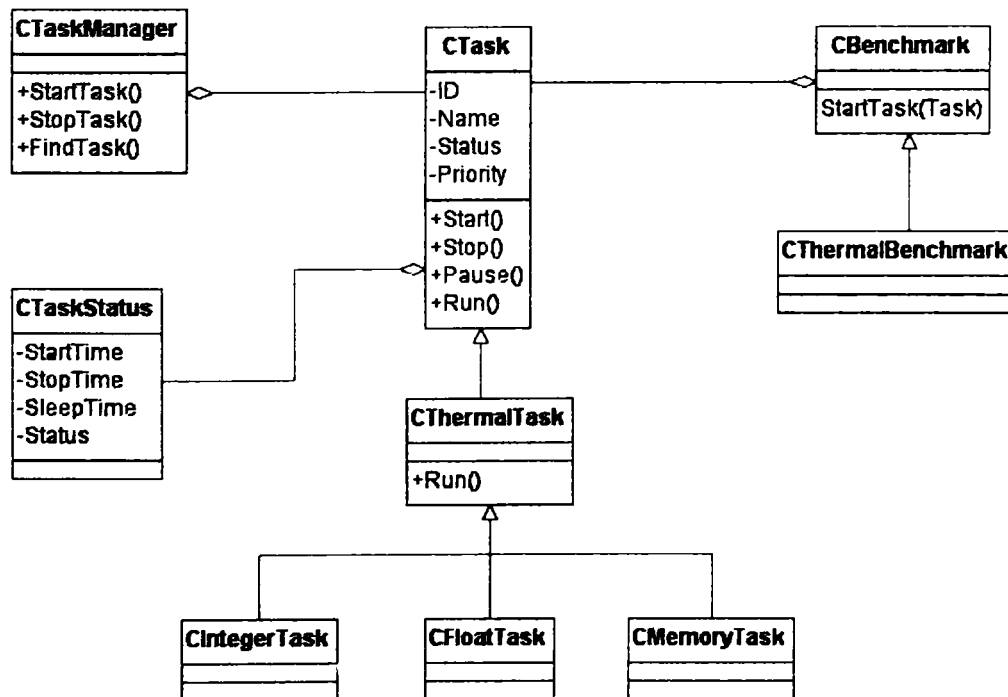


Figura 4.50 – Diagrama de clase a aplicației pentru evaluarea task-urilor termice

Câteva date statistice pentru aplicația de test pentru benchmark-urile termice și task-urile termice sunt prezentate în figura 4.51.

Project Line Counter - Results shown for 83 file(s)								
File Name	Extension	Path	Lines	Code	Comments	Mixed	Blank	
tempdriver	.cpp	f:\work\scho...	169	134	6	0	29	
tempdriver	.h	f:\work\scho...	71	51	5	2	17	
test	.cpp	f:\work\scho...	74	34	28	2	14	
test	.h	f:\work\scho...	49	18	21	3	13	
test	.rc	f:\work\scho...	306	228	50	9	37	
test	.rc2	f:\work\scho...	13	3	7	1	4	
testdlg	.cpp	f:\work\scho...	571	434	64	8	81	
testdlg	.h	f:\work\scho...	82	51	21	4	14	
thermalbenchmark	.cpp	f:\work\scho...	225	190	12	2	25	
thermalbenchmark	.h	f:\work\scho...	36	21	10	2	7	
thermaltask	.cpp	f:\work\scho...	124	81	25	0	18	
thermaltask	.h	f:\work\scho...	33	18	10	2	7	
threadcontext	.cpp	f:\work\scho...	28	15	8	0	5	
threadcontext	.h	f:\work\scho...	34	17	11	2	8	
threadmanager	.cpp	f:\work\scho...	343	214	68	0	61	
threadmanager	.h	f:\work\scho...	83	39	35	2	11	
threadobject	.cpp	f:\work\scho...	126	83	25	0	18	
Total:			Lines	Code	Comments	Mixed	Blank	Net Lines
#:	9,471	6,149	2,120	188	1,390	8,081		
%:	100%	64%	22%	1%	14%	85%		

Filter: <Workspace> Ctrl+Click headers for multiple-sort Close

Figura 4.51 – Date statistice pentru aplicația de benchmark termic

## 4.5 Concluzii

Consumul de energie a devenit una din preocupările principale în procesul de proiectare al sistemelor electronice digitale, datorită popularității din ce în ce mai mari a sistemelor portabile și a interesului manifestat față de acțiunea asupra mediului, a sistemelor desktop și a serverelor. Controlul și reducerea consumului de energie este necesară în primul rând pentru a putea prelungi durata de viață a bateriilor ce însoțesc sistemele portabile și de asemenea de a reduce costurile operaționale și impactul asupra mediului înconjurător ale sistemelor de tip desktop.

Din acest motiv, în acest capitol a fost prezentată problematica temperaturii și a consumului de energie la nivelul unui sistem de calcul. Aceste două domenii sunt în atenția cercetătorilor de câțiva ani, astfel că ceea ce s-a început ar putea constitui o întreagă lucrare de doctorat.

În acest capitol s-a încercat o abordare diferită față de celelalte lucrări în domeniu. Din cele prezentate în subcapitolul 4.4.2, se poate spune că majoritatea mecanismelor de control ale puterii și temperaturii sunt implementate la nivelul sistemului de operare și la cel fizic al componentelor. La fel și articolele ce vin cu propuneri noi vizează tot aceste niveluri și sunt orientate ori pe reducerea puterii sau pe reducerea temperaturii. Pe de altă parte, implementarea unui mecanism de reducere a puterii sau temperaturii la nivelul aplicațiilor utilizator este dificilă din mai multe motive: posibilitatea redusă de control a resurselor fizice de către aplicațiile utilizator, lipsa unei interfețe standard de comunicație cu dispozitivele fizice, mecanismele de reducere a puterii și temperaturii, precum și modul de apelare al lor, sunt specifice pentru fiecare dispozitiv în parte, comportamentul bateriei este dificil de simulat și de prevăzut, etc. Însă, avantajul implementării mecanismului de control la nivelul aplicațiilor utilizator ar fi creșterea semnificativă a eficienței și posibilitatea de intervenție a utilizatorului. În mod normal, utilizatorul este cel mai în măsură să stabilească modul de lucru dorit pentru aplicațiile de care are nevoie.

Contribuția principală ce a fost prezentată în acest capitol a fost implementarea unui mecanism unitar de reducere a temperaturii și puterii pe baza monitorizării parametrilor sistemului (frecvență, tensiuni de alimentare, temperatură, viteză ventilatoare, tip alimentare, procentajul și timpul de încărcare al bateriei) și luarea de decizii la nivelul aplicațiilor utilizator dependent de acești parametri și de configurația specificată de utilizator.

Scopul acestui mecanism este acela de a oferi aplicațiilor utilizator o modalitate cât mai simplă de a obține informații despre parametrii curenți ai sistemului, influența pe care aplicația o are asupra acestor parametri și adaptarea automată, pe baza specificațiilor utilizatorului, pentru a ajunge la parametrii doriți de acesta. Această propunere se poate continua prin implementare acestui mecanism într-un cadru (framework) pentru aplicațiile cu facilitate de control a puterii și/sau temperaturii.

Contribuțiile acestui capitol sunt următoarele:

- studiul facilităților de timp-real în sistemele de operare bazate pe nucleu Window NT (NT, 2000, XP) particularizate pe implementarea funcțiilor de întârziere de rezoluție înaltă. Acest studiu s-a finalizat prin furnizarea unei soluții în acest sens (IMCT'05);
- studiul comparativ al metodelor de măsurare a temperaturii unui sistem pe bază de senzori externi și furnizarea a patru soluții de conectare a senzorilor (SSD'05);
- proiectarea, implementarea și testarea unei aplicații de monitorizare și control al parametrilor sistemului pe baza senzorilor interni sistemului și a celor externi conectați la diferitele interfețe ale sistemului;
- introducerea și definirea conceptului de benchmark termic;

- implementarea, testarea și compararea mai multor tipuri de benchmark-uri termice;
- introducerea și definirea conceptului de task termic;
- propunerea și implementarea unei arhitecturi pe baza căreia aplicațiile utilizator pot afla valorile curente ale parametrilor sistemului în vederea autocontrolului puterii și temperaturii la nivelul lor;
- implementarea unei aplicații de test pentru demonstrarea eficienței controlului puterii și al temperaturii la nivelul aplicațiilor utilizator.

#### Avantajele unui benchmark termic:

- pune în evidență caracteristicile termice ale unui sistem la aplicarea unui anumit tip de stimuli;
- poate detecta câteva tipuri de defecte dintr-un sistem de calcul;
- pune în evidență caracteristicile bateriei unui sistem portabil la aplicarea unui anumit tip de stimuli.

#### Avantajele unui task termic:

- oferă utilizatorilor posibilitatea de a alege (în timpul funcționării) modul de execuție dorit (performanță maximă, temperatură minimă, consum redus);
- poate oferi informații despre posibilitatea de încheiere cu succes a operațiilor cu bateria disponibilă la momentul respectiv;
- se pot specifica la nivelul aplicației care sunt momentele de timp cele mai oportune pentru a fi pornite mecanismele de reducere ale puterii și temperaturii de pe nivelurile inferioare (fizic și sistem de operare).

#### Dezavantaje:

- mecanismul trebuie implementat la nivelul aplicației utilizator (ca soluție se poate furniza un framework pentru astfel de aplicații);
- este introdusă o degradare a performanțelor aplicației (din teste această degradare este de sub 1%).

# 5 Concluzii

Teza de doctorat se încadrează în domeniul de foarte mare actualitate al testării termice și al controlului termic așa cum sunt ele aplicate la diferitele niveluri de abstractizare ale unui sistem de calcul. Obiectivul principal al doctoratului a fost propunerea unei noi metode de testare folosind conceptul de semnătură termică și compararea acestei metode cu metodele de testare existente. Munca de cercetare a pornit cu studiul fenomenelor termice ce apar la diferitele niveluri de detaliu dintr-un sistem de calcul: circuit integrat, placă cu circuite integrate și sistem; iar mai apoi s-a încercat adaptarea metodei de testare termică la fiecare din aceste niveluri. Astfel că teza este structurată pe aceleași trei secțiuni: testarea termică a circuitelor integrate, testarea termică a plăcilor cu circuite integrate și testarea și controlul termic al sistemelor de calcul.

În cazul circuitelor integrate s-au studiat modurile de defectare fizică, efectul acestor defecte asupra temperaturii și posibilitatea de detecție a diferitelor metode de testare (testare logică, testare  $I_{DDQ}$  și testare termică). Efectele defectelor de scurt-circuit dintr-un circuit integrat au fost simulate și studiate într-un mediu de simulare electric (SPICE). Pe baza simulărilor și a testelor efectuate s-au prezentat și câteva rezultate comparative între testarea termică, testarea pe bază de curent și testarea logică.

În prima parte a capitolului 2 s-a evaluat aplicabilitatea metodei de testare termică pentru circuitele integrate CMOS. Pentru a înțelege necesitatea unei metode de testare ce vizează defectele de scurt-circuit pentru circuitele VLSI CMOS s-au prezentat limitările testării logice aplicate circuitelor CMOS și modul în care testarea termică și testarea pe bază de curent rezolvă aceste limitări. Din simulări și teste se poate concluziona că aceste noi metode de testare nu înlocuiesc în totalitate testarea logică, ci sunt folosite împreună cu aceasta pentru a ridica nivelul de detecție a circuitelor defecte.

În a doua parte a capitolului 2 au fost prezentate principalele probleme care apar la testarea defectelor de scurt-circuit atât în cadrul metodei de testare logică cât și pentru metodele de testare termică și  $I_{DDQ}$ . În a treia parte a capitolului 2 s-au prezentat soluțiile existente și cele propuse pentru aceste probleme. S-a prezentat mai apoi aplicația de simulare implementată și rezultatele obținute în cadrul cercetării.

Contribuțiile aduse și prezentate în capitolul 2 sunt următoarele:

- prezentarea și justificarea prin simulări a efectelor termice ale defectelor de scurt-circuit și posibilitatea de detecție a lor prin testarea termică;
- implementarea în SPICE a unei metode de simulare electro-termice pentru circuitele integrate VLSI;
- propunerea și implementarea unei metode genetice de selecție a setului de defecte pentru minimizarea sau maximizarea factorului de acoperire a defectelor de către metodele de testare prezentate;
- propunerea, implementarea și testarea unei metode de simulare logică-electrică pentru defectele de scurt-circuit pentru cele trei metode de testare studiate;
- propunerea unei notații pentru reprezentarea factorului de acoperire a defectelor pentru defectele de scurt-circuit rezistive;
- studiul comparativ pe baza rezultatelor simulărilor între testarea logică, testarea termică și testarea  $I_{DDQ}$ .

În ceea ce privește diferențierea între testarea termică și testarea  $I_{DDQ}$ , trebuie luate în calcul mai multe elemente: dimensiunea și complexitatea circuitului, tehnologia de fabricație utilizată, utilizarea sau nu a senzorilor integrați, nivelul de calitate cerut a se obține, costurile

alocate pentru testare și autotestare, ș.a. Testarea termo-optică este recomandată în faza de proiectare și verificare termică a circuitului integrat, precum și în faza de analiză a defectelor pentru localizarea și identificarea cauzelor apariției acestora. Testarea termo-optică este mai puțin utilizată în faza de testare a circuitelor integrate deoarece este mai lentă decât metoda termo-electrică și implică costuri suplimentare destul de mari, atât pentru echipamentul de testare precum și datorită timpului mare de testare. Metoda de testare termo-electrică se poate utiliza în cazul circuitelor integrate VLSI ce au implementați senzori de temperatură integrați. În celelalte cazuri este recomandată metoda de testare  $I_{DDQ}$ .

Studiul fenomenelor termice la nivelul plăcilor echipate cu circuite integrate a fost o a doua direcție de cercetare în cadrul tezei. Aici au fost folosite imaginile în infraroșu ale unei plăci cu circuite integrate și de niște programe de simulare termică. Rezultatele obținute sunt concretizate în implementarea a doi algoritmi genetici de deconvoluție a imaginilor termice și o metodă de testare și clasificare a plăcilor cu circuite integrate pe bază de semnătură termică.

Metoda de testare termică a plăcilor cu circuite integrate bazată pe semnătura termică este descrisă în Capitolul 3. Sunt introduse două tipuri de semnături termice: semnătura termică uni-dimensională – bazată pe una din caracteristicile extrase din imagine și semnătura termică multi-dimensională bazată pe mai multe caracteristici ale imaginii sau chiar imaginea circuitului. Pentru implementarea conceptului de semnătură termică s-a folosit o rețea neuronală ce funcționează ca un sistem de clasificare a circuitelor de pe placă. Rețeaua neuronală poate fi configurată pentru diferite tipuri de clasificări, cu diferite niveluri ascunse și un număr diferit de ieșiri (clase de clasificare). Rețeaua a fost supusă la mai multe teste efectuate atât pe imaginile ideale (generate) cât și pe imaginile reale.

Contribuțiile acestui capitol sunt:

- propunerea, implementarea și testarea unei metode de deconvoluție a imaginilor și semnalelor pe baza unui algoritm pseudo-genetic (IJECE'2004);
- modificarea algoritmului standard de deconvoluție pentru a implementa deconvoluția imaginilor termice cu surse de căldură neuniforme (HACIPPR'2005);
- deconvoluția imaginilor termice și aproximarea acestora prin curbe gauss;
- propunerea, implementarea și testarea unei metode de testare pe baza semnăturii termice a unei plăci cu circuite integrate (SSD'05);
- implementarea semnăturii termice utilizând rețelele neuronale și compararea ei cu metoda scăderii imaginilor (IMCT'2005);
- proiectare și implementarea unei aplicații destinată studiului și analizei imaginilor termice a plăcilor cu circuite integrate.

A treia direcție de cercetare a tezei a avut în vedere studiul temperaturii într-un sistem de calcul ca ansamblu. Într-un sistem, pe lângă utilizarea temperaturii ca metodă de testare, sunt foarte importante și mecanismele de monitorizare, control și reducere a temperaturii. În cadrul cercetărilor efectuate s-a introdus, prezentat și implementat noțiunea de benchmark termic al unui sistem, s-a implementat o aplicație de monitorizare și control d.p.d.v. termic pentru un sistem și s-a propus și implementat o interfață de programare (API) pentru managementul și reducerea temperaturii la nivelul aplicațiilor utilizator.

Consumul de energie a devenit una din preocupările principale în procesul de proiectare al sistemelor electronice digitale, datorită popularității din ce în ce mai mari a sistemelor portabile și a interesului manifestat față de acțiunea asupra mediului, a sistemelor desktop și a serverelor. Controlul și reducerea consumului de energie este necesară în primul rând pentru a putea prelungi durata de viață a bateriilor ce însoțesc sistemele portabile și de asemenea de a reduce costurile operaționale și impactul asupra mediului înconjurător ale sistemelor de tip desktop.



Din acest motiv, în acest capitol s-a încercat a se prezenta problematica temperaturii și a consumului de energie la nivelul unui sistem de calcul, încercând o abordare diferită față de celelalte lucrări în domeniu. Majoritatea mecanismelor de control ale puterii și temperaturii sunt implementate la nivelul sistemului de operare și la cel fizic al componentelor, la fel și articolele ce vin cu propuneri noi vizează tot aceste niveluri și sunt orientate ori pe reducerea puterii sau pe reducerea temperaturii. Pe de altă parte, implementarea unui mecanism de reducere a puterii sau temperaturii la nivelul aplicațiilor utilizator este dificilă din mai multe motive: posibilitatea redusă de control a resurselor fizice de către aplicațiile utilizator, lipsa unei interfețe standard de comunicație cu dispozitivele fizice, mecanismele de reducere a puterii și temperaturii, precum și modul de apelare al lor, sunt specifice pentru fiecare dispozitiv în parte, comportamentul bateriei este dificil de simulat și de prevăzut, etc. Însă, avantajul implementării mecanismului de control la nivelul aplicațiilor utilizator ar fi creșterea semnificativă a eficienței și posibilitatea de intervenție a utilizatorului. În mod normal, utilizatorul este cel mai în măsură să stabilească modul de lucru dorit pentru aplicațiile de care are nevoie.

Propunerea principală care a fost prezentată în acest capitol a fost implementarea unui mecanism unitar de reducere a temperaturii și puterii pe baza monitorizării parametrilor sistemului (frecvență, tensiuni de alimentare, temperatură, viteză ventilatoare, tip alimentare, procentajul și timpul de încărcare al bateriei) și luarea de decizii la nivelul aplicațiilor utilizator dependent de acești parametri și de configurația specificată de utilizator.

Scopul acestui mecanism este acela de a oferi aplicațiilor utilizator o modalitate cât mai simplă de a obține informații despre parametrii curenți ai sistemului, influența pe care aplicația o are asupra acestor parametri și adaptarea automată, pe baza specificațiilor utilizatorului, pentru a ajunge la parametrii doriți de acesta. Această propunere se poate continua prin implementare acestui mecanism într-un cadru (framework) pentru aplicațiile cu facilitate de control a puterii și/sau temperaturii.

Contribuțiile acestui capitol sunt următoarele:

- studiul facilităților de timp-real în sistemele de operare bazate pe nucleu Window NT (NT, 2000, XP) particularizate pe implementarea funcțiilor de întârziere cu rezoluție înaltă. Acest studiu s-a finalizat prin furnizarea unei soluții în acest sens (IMCT'05);
- studiul comparativ al metodelor de măsurare a temperaturii unui sistem pe bază de senzori externi și furnizarea a patru soluții de conectare a senzorilor (SSD'05);
- proiectarea, implementarea și testarea unei aplicații de monitorizare și control al parametrilor sistemului pe baza senzorilor interni sistemului și a celor externi conectați la diferitele interfețe ale sistemului;
- introducerea și definirea conceptului de benchmark termic;
- implementarea, testarea și compararea mai multor tipuri de benchmark-uri termice;
- introducerea și definirea conceptului de task termic;
- propunerea și implementarea unei arhitecturi pe baza căreia aplicațiile utilizator pot afla valorile curente ale parametrilor sistemului în vederea autocontrolului puterii și temperaturii la nivelul lor;
- implementarea unei aplicații de test pentru demonstrarea eficienței controlului puterii și al temperaturii la nivelul aplicațiilor utilizator.

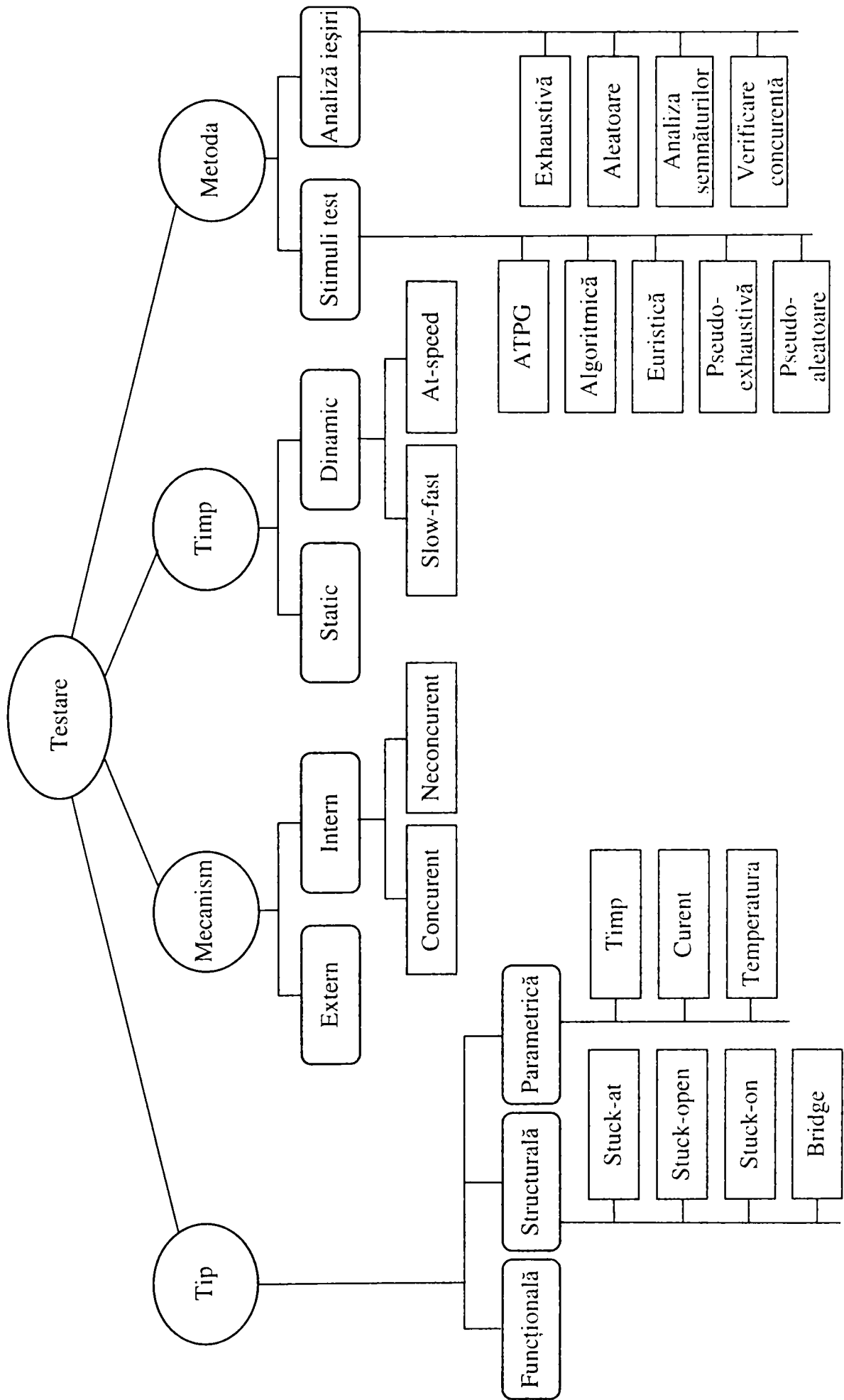
Ca și concluzii finale, la nivelul circuitelor integrate se recomandă utilizarea metodei de testare termo-electrică folosind senzori de temperatură integrați pe pastila de siliciu, dacă circuitul este suficient de complex să justifice dimensiunea de siliciu alocată senzorului. Din acest motiv microprocesoarele, procesoarele video, procesoarele DSP, microcontrolerele ar trebui să aibă cel puțin un senzor de temperatură integrat care să poată fi utilizat atât în faza de testare cât și în faza de exploatare. Plăcile cu circuite integrate se pretează la metoda de testare termo-optică atât în faza de proiectare cât și în faza de testare. Utilizarea metodei de testare bazată pe semnături termice ar putea aduce câteva avantaje semnificative față de metoda de testare utilizată în prezent. La nivel de sistem este foarte utilă implementarea de senzori de temperatură în componentele sistemului ce se încălzesc cel mai mult și conectarea lor la o magistrală comună pentru a putea fi accesați din aplicații. Acești senzori pot fi folosiți la monitorizarea și controlul temperaturii sistemului așa cum a fost prezentat în capitolul 4 al tezei.

Pentru a avea o privire de ansamblu asupra muncii depuse la conceperea și finalizarea acestei teze, voi prezenta câteva date statistice. S-au implementat patru aplicații software, una pentru simularea logico-electrică a defectelor de scurt-circuit rezistiv, una pentru studiul imaginilor termice, o aplicație pentru monitorizarea unui sistem de calcul și ultima pentru implementarea conceptelor de benchmark termic și task termic. Aceste aplicații conțin împreună aproximativ 40000 linii de cod în 150 clase. S-a implementat o schemă cu microcontroller 80C552 pentru monitorizarea temperaturii externe și încă trei modalități de conectare a senzorilor de temperatură direct la porturile unui sistem. S-au realizat câteva sute de teste, care cumulează împreună sute de ore de funcționare a programelor de test. Testele au fost efectuate pe un număr de 11 sisteme: 6 calculatoare PC, 3 laptop-uri și 2 servere.

Perspectivile de continuare a cercetărilor vizează în primul rând soluțiile de control și reducere a temperaturii la nivelul componentelor unui sistem de calcul:

- contribuțiile din Capitolul 4 pot fi implementate și testate pe diferite tipuri de sisteme portabile și pot fi incluse într-un framework pentru dezvoltarea aplicațiilor;
- posibilitatea de implementare a mecanismului de control al task-urilor la nivelul nucleului sistemului de operare;
- utilizarea benchmark-ului termic în faza de testare și dimensionare a PC-urilor.

# Anexa 1. O taxonomie a testării



# Anexa 2

## SPICE BSIM3 VERSION 3.1 PARAMETERS

\* SPICE 3f5 Level 8, Star-HSPICE Level 49,\* UTMOST Level 8

\* DATE: Feb 18/03

\* LOT: T2CU WAF: 0001

\* Temperature\_parameters=Default

```
.MODEL NMOS18 NMOS (
+VERSION = 3.1          TNOM   = 27          TOX    = 4.1E-9
+XJ    = 1E-7          NCH    = 2.3549E17      VTH0   = 0.3669193
+K1    = 0.592797     K2    = 2.518108E-3    K3     = 1E-3
+K3B   = 4.7942179    W0    = 1E-7          NLX    = 1.745125E-7
+DVTOW = 0            DVT1W = 0            DVT2W = 0
+DVT0  = 1.3683195    DVT1  = 0.4097438    DVT2  = 0.0552615
+U0    = 263.5112775  UA    = -1.363381E-9  UB     = 2.253823E-18
+UC    = 4.833037E-11 VSAT  = 1.017805E5    A0     = 1.9261289
+AGS   = 0.4192338   B0    = -1.069507E-8  B1     = -1E-7
+KETA  = -8.579587E-3 A1    = 2.789024E-4   A2     = 0.8916186
+RDSW  = 126.5291844 PRWG  = 0.4957859    PRWB   = -0.2
+WR    = 1            WINT  = 0            LINT   = 7.790316E-9
+XL    = -2E-8        XW    = -1E-8        DWG    = -1.224589E-8
+DWB   = 1.579145E-8 VOFF  = -0.0895222   NFACTOR = 2.5
+CIT   = 0            CDSC  = 2.4E-4        CDSCD  = 0
+CDSCB = 0            ETA0  = 2.95614E-3    ETAB   = 1.374596E-4
+DSUB  = 0.013974    PCLM  = 0.7291486    PDIBLC1 = 0.1332365
+PDIBLC2 = 2.151668E-3 PDIBLCB = -0.1        DROUT  = 0.6947618
+PSCBE1 = 7.412661E10 PSCBE2 = 1.812826E-7 PVAG   = 9.540595E-3
+DELTA = 0.01        RSH   = 5.9          MOBMOD  = 1
+PRT   = 0            UTE   = -1.5        KT1    = -0.11
+KT1L  = 0            KT2   = 0.022       UA1    = 4.31E-9
+UB1   = -7.61E-18   UC1   = -5.6E-11    AT     = 3.3E4
+WL    = 0            WLN   = 1           WW     = 0
+WWN   = 1            WWL   = 0           LL     = 0
+LLN   = 1            LW    = 0           LWN    = 1
+LWL   = 0            CAPMOD = 2          XPART  = 0.5
+CGDO  = 8.71E-10    CGSO  = 8.71E-10    CGBO   = 1E-12
+CJ    = 9.67972E-4  PB    = 0.6966474    MJ     = 0.3609772
+CJSW  = 2.443898E-10 PBSW  = 0.8082076    MJSW  = 0.1013742
+CJSWG = 3.3E-10    PBSWG = 0.8082076    MJSWG = 0.1013742
+CF    = 0            PVT0  = 7.226579E-4 PRDSW  = -4.5298309
+PK2   = -4.696208E-4 WKETA = 6.028223E-3   LKETA  = -8.791311E-3
+PU0   = 17.2549887 PUA   = 6.802365E-11 PUB    = 4.224871E-24
+PVSAT = 1.298468E3 PETA0 = 1.003159E-4  PKETA  = -3.864603E-4
TEMP = 25)
*
```

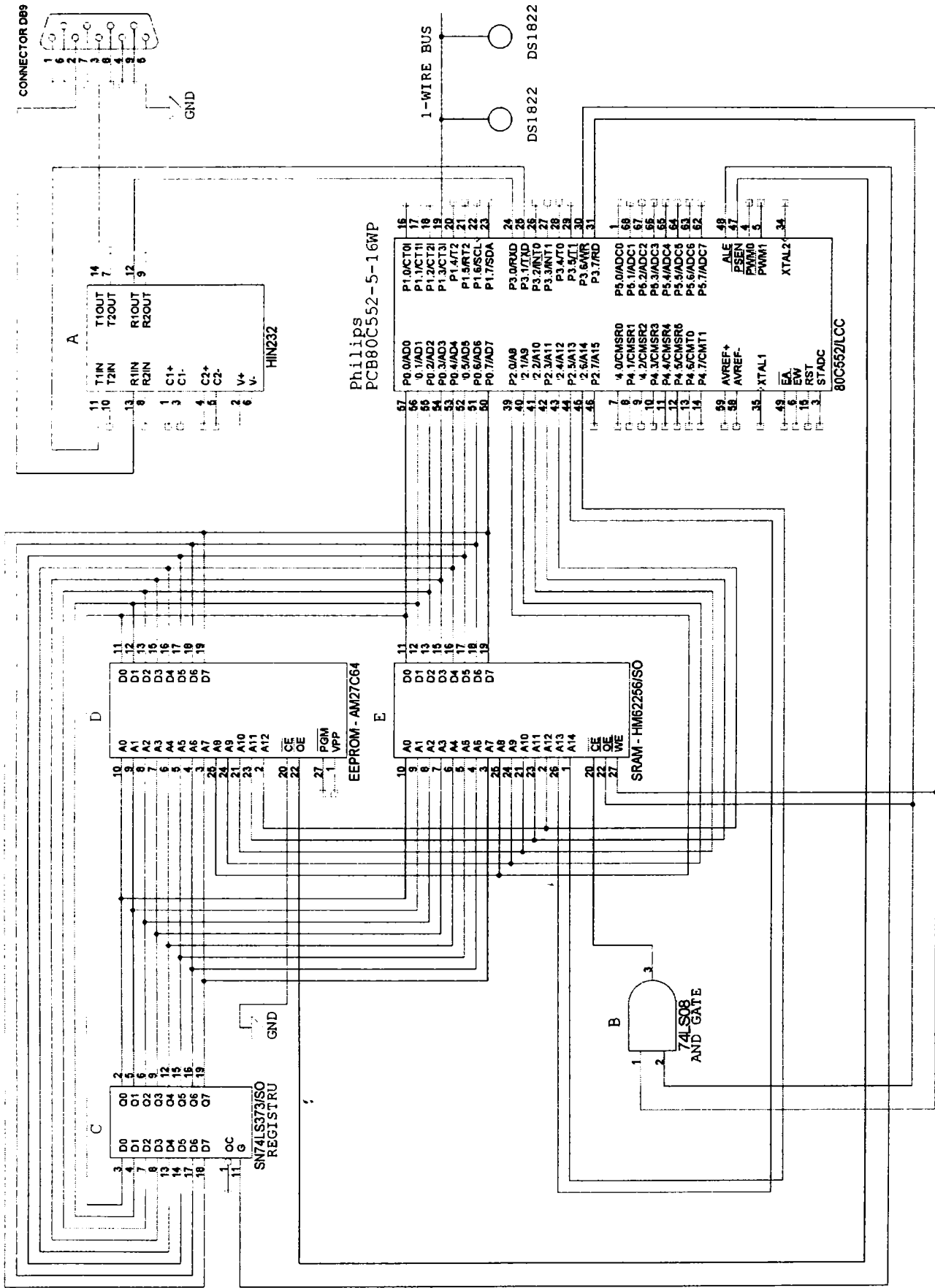
```

.MODEL PMOS18 PMOS (
+VERSION = 3.1
+XJ = 1E-7
+K1 = 0.5772615
+K3B = 14.2532769
+DVTOW = 0
+DVT0 = 0.6718731
+U0 = 118.0541064
+UC = -1E-10
+AGS = 0.4096261
+KETA = 0.0212376
+RDSW = 306.4304418
+WR = 1
+XL = -2E-8
+DWB = 8.005928E-9
+CIT = 0
+CDSCB = 0
+DSUB = 0.7172358
+PDIBLC2 = 0.0165863
+PSCBE1 = 7.71553E9
+DELTA = 0.01
+PRT = 0
+KT1L = 0
+UB1 = -7.61E-18
+WL = 0
+WWN = 1
+LLN = 1
+LWL = 0
+CGDO = 6.92E-10
+CJ = 1.173089E-3
+CJSW = 2.217367E-10
+CJSWG = 4.22E-10
+CF = 0
+PK2 = 1.495689E-3
+PU0 = -1.2891258
+PVSAT = -50
TEMP = 25)
*
TNOM = 27
NCH = 4.1589E17
K2 = 0.026742
W0 = 1E-6
DVT1W = 0
DVT1 = 0.3118588
UA = 1.626518E-9
VSAT = 2E5
B0 = 7.705744E-7
A1 = 0.5260122
PRWG = 0.5
WINT = 0
XW = -1E-8
VOFF = -0.0992452
CDSC = 2.4E-4
ETA0 = 0.0331989
PCLM = 1.5224082
PDIBLCB = -1E-3
PSCBE2 = 2.228426E-9
RSH = 6.7
UTE = -1.5
KT2 = 0.022
UC1 = -5.6E-11
WLN = 1
WWL = 0
LW = 0
CAPMOD = 2
CGSO = 6.92E-10
PB = 0.8524959
PBSW = 0.5936755
PBSWG = 0.5936755
PVTH0 = 1.425828E-3
WKETA = 0.0286138
PUA = -5.395E-11
PETA0 = 1.003159E-4
LEVEL = 8
TOX = 4.1E-9
VTH0 = -0.4002789
K3 = 0
NLX = 9.883899E-8
DVT2W = 0
DVT2 = 0.1
UB = 1.229265E-21
A0 = 1.8109799
B1 = 2.657048E-6
A2 = 0.3207082
PRWB = 0.0612789
LINT = 2.043723E-8
DWG = -4.602158E-8
NFACTOR = 2
CDSCD = 0
ETAB = -0.0375363
PDIBLC1 = 2.700462E-4
DROUT = 1.640424E-4
PVAG = 5.1166248
MOBMOD = 1
KT1 = -0.11
UA1 = 4.31E-9
AT = 3.3E4
WW = 0
LL = 0
LWN = 1
XPART = 0.5
CGBO = 1E-12
MJ = 0.415401
MJSW = 0.2603391
MJSWG = 0.2603391
PRDSW = 0.9887283
LKETA = -2.746502E-3
PUB = 1E-21
PKETA = -2.891811E-3

```

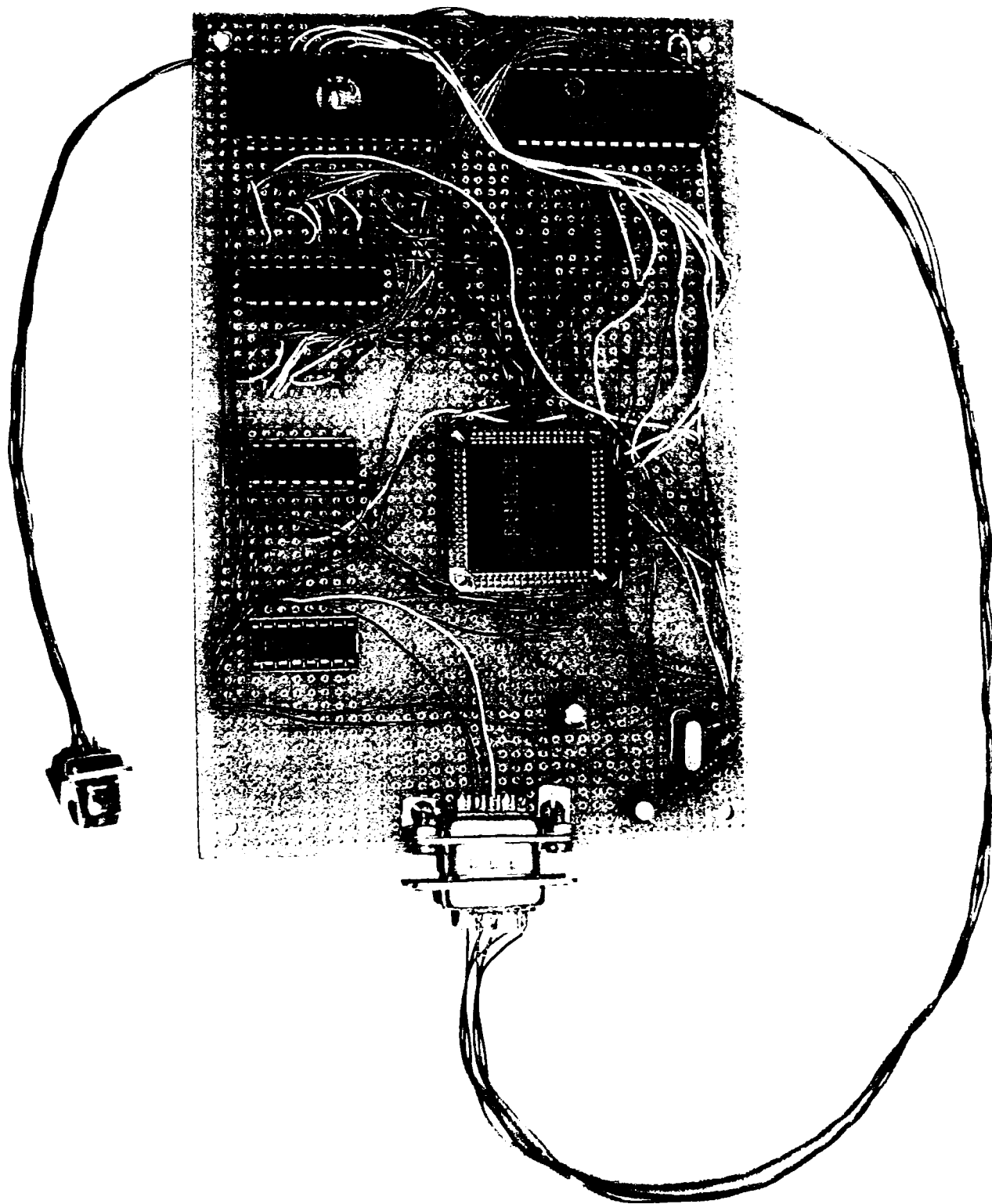
# Anexa 3

## Schema plăcii externe de monitorizare a senzorilor de temperatură



## Anexa 4

### Placa externă de monitorizare a senzorilor de temperatură



# Anexa 5

## Procesele executate pe un server WEB și DB

```
<25.06.04 10:51:35 (127326234951513280) - 25.06.04 15:53:44 (127326416233680000)>
[Processes]
 8 System                0.51 32 12865099104                0 127326416233680000    32768 0
196 SMSS.EXE             0.00 32   5608064 127320445850895408 127326416233680000    225280 0
244 WINLOGON.EXE        0.01 128  436627840 127320445973571808 127326416233680000    233757 0
272 SERVICES.EXE       0.07 32   992727472 127320445990696432 127326416233680000    173668 0
284 LSASS.EXE           0.10 32  4464119088 127320445990996864 127326416233680000    23009 0
398 termsrv.exe         0.00 32   80015056 127320446004416160 127326416233680000     5645 0
500 svchost.exe         0.00 32   15822752 127320446033858496 127326416233680000    104261 0
532 SPOOLSV.EXE         0.00 32   187669856 127320446040267712 127326416233680000    194739 0
824 msdtc.exe           0.00 32   7210368 127320446590458848 127326416233680000     90963 0
952 dfssvc.exe          0.00 32  133191520 127320446605580592 127326416233680000    119631 0
980 dptcom.exe          0.00 32   300432 127320446606882464 127326416233680000         0 0
996 dptserv.exe         0.00 32   300432 127320446607883904 127326416233680000         0 0
1004 dptscom.exe        0.00 32   600864 127320446607984048 127326416233680000    28672 0
1016 dptelog.exe        0.00 256   801152 127320446608484768 127326416233680000     1150 0
1028 svchost.exe        0.00 32   12417856 127320446608985488 127326416233680000    113804 0
1048 ismserv.exe        0.00 32   2703888 127320446610587792 127326416233680000    217202 0
1064 LLSSRV.EXE         0.00 32  108455952 127320446611088512 127326416233680000    19977 0
1112 sqlservr.exe       1.29 32 94699771584 127320446613291680 127326416233680000    162190 0
1216 ntfers.exe         0.01 32   187169136 127320446630015728 127326416233680000    105004 0
1384 agntsrv.exe        0.00 32   1902736 127320446749187088 127326416233680000    28672 0
1416 Apache.exe         0.00 32   29141904 127320446769316032 127326416233680000    28672 0
1424 CMD.EXE            0.00 32   300432 127320446780732448 127326416233680000         0 0
1440 dbnmp.exe          0.00 32   13118864 127320446798558080 127326416233680000    66343 0
1456 TNSLNSR.EXE        0.00 32   3304752 127320446803164704 127326416233680000    90903 0
1480 oracle.exe         0.14 32  7792404928 127320446832206464 127326416233680000    141505 0
1520 regsvc.exe         0.00 32   1902736 127320446871062336 127326416233680000    78904 0
1532 LOCATOR.EXE        0.00 32   400576 127320446880876448 127326416233680000    98304 0
1544 mstask.exe          0.00 32   2403456 127320446882979472 127326416233680000    222368 0
1620 lserver.exe        0.00 32   54778768 127320446898601936 127326416233680000    84664 0
 436 WinMgmt.exe         0.00 32   58984816 127320446924839664 127326416233680000         0 0
1756 svchost.exe        0.00 32   25636864 127320446942965728 127326416233680000    181813 0
1768 certsrv.exe        0.00 32   8912816 127320446944868464 127326416233680000    227189 0
1796 inetinfo.exe       0.01 32   156725360 127320446949274800 127326416233680000    147294 0
1836 mssearch.exe       0.00 32   3204608 127320446962994528 127326416233680000    100311 0
1900 sqlagent.exe       0.00 32   326970160 127320447041908000 127326416233680000    121407 0
1932 Apache.exe         0.03 32   846617376 127320447077759552 127326416233680000    218036 0
2172 java.exe           0.01 32   444939792 127320447101794112 127326416233680000    19742 0
2180 isqlplus           0.00 32   1602304 127320447102194688 127326416233680000    12135 0
2796 explorer.exe       0.00 32   141102896 127320447671713616 127326416233680000    75361 0
2988 DUMETER.EXE        0.04 32   221818960 127320447924176640 127326416233680000    168566 0
3092 qttask.exe         0.00 32   300432 127320447978254400 127326416233680000    118784 0
2740 svchost.exe        0.00 32   1201728 127320448043648432 127326416233680000     8121 0
 756 rdpclip.exe        0.00 32   600864 127321341984972928 127326416233680000    218090 0
3324 explorer.exe       0.00 32   113262864 127321341996689776 127326416233680000    92962 0
2756 mdm.exe            0.00 32   701008 127321342843507440 127326416233680000    167936 0
 544 wuaucit.exe         0.00 32   1101584 127324708800616064 127326416233680000     5272 0
3460 rdpclip.exe        0.00 32   300432 127325455257969024 127326416233680000    38056 0
3712 explorer.exe       0.01 32   48469696 127325617093777488 127326416233680000    70022 74
3792 qttask.exe         0.00 32   400576 127325617104492896 127326416233680000    230550 0
3660 WINCMD32.EXE       0.00 32   20729808 127325617511478112 127326416233680000    139291 295
3192 taskmgr.exe        0.00 128   3505040 127325626770692208 127326416233680000    233672 0
1484 mmc.exe            0.29 32   608575088 127326224892248768 127326416153560000     72526 264
3688 PowerSpy.exe      2.47 32  4477938960 127326234945805072 127326416233680000    126548 158
3832 php.exe            0.02 32   15221888 127326305780860880 127326305895020000    4373320 11
3352 php.exe           0.17 32   152018592 127326317518038112 127326321066840000    6601732 172
3040 rundll32.exe       0.00 32   1602304 127326319849390432 127326320455960000    4841454 16
3752 mmc.exe            0.01 32   10314832 127326319967960928 12732632043593000010740922 45
[Monitor]
VIA686:Temp[1] = 22.4700
VIA686:Temp[2] = 43.2300
VIA686:Temp[3] = 0.0000
VIA686:Fan[1] = 0.0000
VIA686:Fan[2] = 4911.7138
VIA686:Voltage[1] = 1.7713
VIA686:Voltage[2] = 2.0400
VIA686:Voltage[3] = 3.3646
VIA686:Voltage[4] = 5.4184
VIA686:Voltage[5] = 12.2212
Processor Usage = 5.0800 %
```



# Bibliografie

[1WireBus] – “High-Precision 1-Wire Digital Thermometer”. Dallas Semiconductor DS18S20 datasheet.

[Abramovici1990] – M. Abramovici, M. A. Breuer, A. D. Friedman, “Digital Systems Testing and Testable Design”, Computer Science Press, 1990.

[Acquaviva2001] – A. Acquaviva, L. Benini, B. Ricco, “Energy Characterization of Embedded Real-Time Operating Systems”. ACM Computer Architecture News, vol. 29, no. 5, pp. 13-18, December 2001.

[Altet1997] – J. Altet, A. Rubio, H. Tamamoto, “Analysis of the Feasibility of Dynamic Thermal Testing in Digital Circuits”, Proceedings of 5th Asian Test Symposium, p. 149-154. Nov. 1997.

[Altet2000] – J. Altet, A. Rubio, E. Schaub, S. Dilhaire, W. Claeys, “Thermal Testing: Fault Location Strategies”, 18<sup>th</sup> VLSI Test Symposium, p. 189-193, Apr. 2000.

[Altet2001] – J. Altet, A. Rubio, E. Schaub, S. Dilhaire, W. Claeys, “Thermal Coupling in Integrated Circuits: Application to Thermal Testing”. IEEE Journal of Solid-State Circuits, Jan. 2001.

[Attarha2000] – A. Attarha, M. Nourani, C. Lucas, “Modeling and Simulation of Real Defects Using Fuzzy Logic”, (DAC), 2000.

[Awtrey1997] – D. Awtrey, “Transmitting Data and Power over a One-Wire Bus”. Sensors the Journal of Applied Sensing Technology, Feb. 1997.

[Ayers1988] – G.R. Ayers, J.C. Dainty, “Iterative Blind Deconvolution Method and its Applications”, Optics Letter, 13, pp. 547-549, 1988.

[Baker1997] – A. Baker, “The Windows NT Device Driver Book: A Guide for Programmers”, Prentice Hall, 1997.

[Baker1999] – B. Baker, “Temperature Sensing Technologies”, Microchip Technology Inc., Application Note AN679, Aug. 1999.

[Biggs1997] – D. Biggs, M. Andrews, “Iterative Blind Deconvolution of Extended Objects”. International Conference on Image Processing (ICIP '97), 1997.

[Briskin1998] – G. Briskin, D. Cutts, G. Watts, R. Zeller, “The Use of NT in Quasi-Realtime Environments”, CHEP98, 1998.

[Brooks2001] – D. Brooks, M. Martonosi, “Dynamic Thermal Management for High-Performance Microprocessors”, Proceedings of the 7<sup>th</sup> International Symposium on High-Performance Computer Architecture, January 2001.

[Brown2004] – C.Brown, “Thermal Repeatability in Lead-Free Array Rework”, Circuits Assembly, April 2004.

[Bond1991] – J. Bond, “Inspection Systems Distribute Test Throughout Manufacturing”, Test & Measurement World, pp. 65-66, Dec. 1991.

[Butler1992] – K.M. Butler, R. Kapur, M.R. Mercer, D.E. Ross, “The Roles of Controllability and Observability in Design for Test”, 1992 IEEE VLSI Test Symp., pp. 211-216, Apr. 1992.

[Chakravarty1994] – S. Chakravarty, P. Thadikaran, “A Study of IDDQ Subset Selection Algorithms for Bridging Faults”, IEEE International Test Conference (ITC), pp. 403-412, 1994.

[Chakravarty1997] – S. Chakravarty, P. Thadikaran, “Introduction to IDDQ Testing”, Kluwer Academic Publishing, 1997.

[Chang1997] – J. Chang, E. McCluskey, “Detecting Bridging Faults in Dynamic CMOS Circuits”, Workshop on IDDQ Testing, Nov. 1997.

[Chang1998] – J. Chang, C.W. Tseng, Y.C. Chu, S. Wattal, M. Purtell, E. McCluskey, “Experimental Results for IDDQ and VLV Testing”, 16<sup>th</sup> IEEE VLSI Test Symposium, Apr. 1998.

[Chedid2002] – W. Chedid, C. Yu, “Survey on Power Management Techniques for Energy Efficient Computer Systems”, Mobile Computing Research Lab., Sep. 2002.

[Chen1997] – T. Chen, I.N. Hajj, “Extraction, Simulation and IDDQ Test Generation for Efficient Bridging Fault Detection in Digital VLSI Circuits”, European Conference on Circuit Theory and Design, 1997.

[Chen1997] – Y. Chen, Z. Nakao, K. Arakaki, S. Tamura, “Blind Deconvolution Based on Genetic Algorithms”, IEICE Transaction Fundamentals, vol. E80-A, pp. 2603 - 2607, Dec. 1997.

[Chiang2001] – T.Y. Chiang, K. Banerjee, K. Saraswat, “Compact Modeling and SPICE-Based Simulation for Electrothermal Analysis of Multilevel ULSI Interconnects”, IEEE International Conference on Computer Aided Design (ICCAD), San Jose, pp. 165-172, Nov. 2001.

[Chung2001] – W. Chung, “Challenges and Perspective of Integrated Circuits and Systems Research in the New Century”, <http://soc1.cs.ccu.edu.tw/vlsi2002/>, Mar. 2001.

[Colquitt2000] – M. Colquitt, “Thermal Characterization of Packages for ICs”, Intersil Technical brief, 2000.

[Corno2000] – F. Corno, M. Reorda, G. Squillero, M. Violante, “A Genetic Algorithm-based System for Generating Test Programs for Microprocessor IP Cores”, The 12<sup>th</sup> IEEE International Conference on Tools with Artificial Intelligence, ICTAI2000, pp. 195-198, Canada, Nov. 2000.

- [Crouch1999] – A. Crouch, “Semiconductor IC Test and Design-for-Test Fundamentals”, WhitePaper, Inovys Corporation, 1999. ([www.inovys.com/login/DFTwhitepaper.pdf](http://www.inovys.com/login/DFTwhitepaper.pdf))
- [Daasch2000] – R. Daasch, J. McNames, D. Bockelman, K. Cota, “Variance Reduction Using Wafer Patterns in IDDQ Data”, Proceedings of International Test Conference, p. 189-198, Oct. 2000.
- [Dashney1999] – G. Dashney, L. Walker, “SPICE Generates the Thermal Response Models of a Power Semiconductor”, Thermal Modeling Brochure. ON Semiconductor, Oct. 1999.
- [Dessai1998] – V. Sar-Dessai, D.M.H. Walker, “Accurate Fault Modeling and Fault Simulation of Resistive Bridges”, International Symposium on Defect and Fault Tolerance in VLSI Systems, pp. 102-107, Nov. 1998.
- [Dillner1998] – U. Dillner, “Thermal Simulation and Realization of Micromachined Thermal Sensor Array”, Proceedings of 5<sup>th</sup> NEXUSPAN Workshop on Thermal Aspects in Microsystem Technology, Budapest, 1998, pp. 133-141.
- [Electus2001] – “Heatsink Basics”, Electus Distribution Reference Data Sheet, 2001.
- [Engelke2003] – P. Engelke, I. Polian, M. Renovell, B. Becker, “Simulating Resistive Bridging and Stuck-At Faults”, International Test Conference (ITC), pp. 1051-1059, 2003.
- [Farina1995] – D. Farina, “Making Surface Temperature Measurements Using Liquid Crystal Thermography”, Electronics Cooling, pp. 10-14, Oct. 1995.
- [Ferguson1989] – F.J. Ferguson, M. Taylor, T. Larrabee, “Feasibility Study on the Cost of IDDQ Testing in CMOS Circuits”, Proceedings of the International Test Conference, 1989.
- [Ferguson1990] – F.J. Ferguson, M. Taylor, T. Larrabee, “Testing for Parametric Faults in Static CMOS Circuits”, Proceedings of the International Test Conference, p. 436-443, 1990.
- [Fradin1997] – J.P. Fradin, L. Molla, B. Desauettes, “Analysis of 3D Conjugate Heat Transfers in Electronics”, ED&TC 1997.
- [Franco1995] – P. Franco, E. McCluskey, “An Experimental Chip to Evaluate Test Techniques and Experiment Design”, Proceedings of International Test Conference, p. 653-662, Oct. 1995.
- [Gattiker1996] – A.E. Gattiker, W. Maly, “Current Signatures in VLSI Circuit Testing”, 14<sup>th</sup> IEEE VLSI Test Symposium, p. 112-117, Apr. 1996.
- [Glaser1994] – U. Glaser, H. T. Vierhaus, M. Kley, A. Wiederhold, “Test Generation for Bridging Faults in CMOS Ics Based on Current Monitoring Versus Signal Propagation”, IEEE International Conference on Computer-Aided Design (ICCAD), pp.36-39, 1994.
- [Goldstein1979] – L.H. Goldstein, “Controllability/Observability Analysis of Digital Circuits”, IEEE Trans. on Circuits and Systems, vol. 26, no. 9, pp. 685-693, Sept. 1979.

- [Gonzalez1987] – R.C. Gonzalez, P. Wintz, "Digital Image Processing", Addison Wesley, 1987.
- [Goodall1991] – A.J.E. Goodall, E.K. Lo, "A Review of Inspection Techniques Applicable to PCB Manufacturing and Assembly, particularly with respect to SMT", Advanced Manufacturing Engineering, Vol. 3, Jan. 1991.
- [Gunther2001] – S. Gunther, F. Binns, D. Carmean, J. Hall, "Managing the Impact of Increasing Microprocessor Power Consumption", Intel Technology Journal, 2001.
- [Hagan1996] – M.T. Hagan, H.B. Demuth, M. Beale, "Neural Networks Design", PWS Publishing Company. 1996.
- [Hawkins1994] – F. Hawkins, J. Soden, A. Righter, F. Ferguson, "Defect Classes - An Overdue Paradigm for CMOS IC Testing", International Test Conference (ITC), pp. 413-425, 1994.
- [Henderson2001] – C. Henderson, "Electrical Testing", Lecture Notes, 2001.
- [Holmes2002] – T. Holmes, "Background of Deconvolution", Technical Note, AutoQuant Imaging Inc., Feb. 2002.
- [Hora2002] – C. Hora, "On Diagnosing Faults in Digital Circuits", PhD Thesis, Technische Universiteit Eindhoven, 2002.
- [Hsieh2004] – S.J. Hsieh, "Artificial Neural Networks and Statistical Modeling for Electronic Stress Prediction Using Thermal Profiling", IEEE Transactions on Electronics Packaging Manufacturing, Vol. 27, Issue 1, pp. 49-58, Jan. 2004.
- [Huang2000] – M. Huang, J. Renau, S. Yoo, J. Torrellas, "A Framework for Dynamic Energy Efficiency and Temperature Management", ACM/IEEE International Symposium on Microarchitecture, 2000.
- [Huang2004] – W. Huang, M. Stan, K. Skadron, K. Sankaranarayanan, S. Ghosh, S. Velusamy, "Compact Thermal Modeling for Temperature-Aware Design", DAC 2004, Jun. 2004.
- [INTEL2002] – "Measuring Instruction Latency and Throughput", Intel developer services, <http://www.intel.com/cd/ids/developer/asmo-na/eng/microprocessors/ia32/20243.htm>, May 2002.
- [INTEL2002\_1] – "Application Power Management for Mobility", Intel White Paper, Mar. 2002.
- [Ind2002] – "Industry Outlook 2002", Printed Circuit Design, [www.pcdmag.com](http://www.pcdmag.com), 2002.
- [Isern1993] – E. Isern, J. Figueras, "Test Generation with High Coverages for Quiescent Test of Bridging Faults in Combinational Circuits", IEEE International Test Conference (ITC), pp. 21-23, 1993.

[Jacob1992] – G. Jacob, “Advances in Board Inspection”, Evaluation Engineering, pp. 126-133, Sep. 1992.

[JESD51-4] – “Thermal Test Chip Guideline”, EIA/JEDEC STANDARD, 1997.

[Johnson1989] – B.W. Johnson, “Design and Analysis of Fault-Tolerant Digital Systems”, Addison-Wesley Publishing Company, 1989.

[Joiner1999] – B. Joiner, “Use of Junction-to-Board Thermal Resistance in Predictive Engineering”, Cooling Zone, Jan. 1999.

[Kameyama1998] – K. Kameyama, Y. Kosugi, T. Okahashi, M. Izumita, “Automatic Defect Classification in Visual Inspection of Semiconductors Using Neural Networks”, IEICE Trans. Inf. & Syst., Vol. E81-D, No. 11, Nov. 1998.

[Keshavarzi1998] – A. Keshavarzi, K. Roy, C.F. Hawkins, “Intrinsic Leakage in Deep Submicron CMOS IC’s – Measurement Based Test and Power Solutions”, IEEE Transactions on VLSI Systems, 1998.

[Kilbride2000] – B. Kilbride, “Windows NT and Windows 2000 for Real-Time Applications”, International IC, pp. 70-72, 2000.

[Klaassen1997] – B. Klaassen, “A Method for Tightly-Coupled Thermal-Electrical Simulation”, Microelectronics Journal, Vol. 28, No. 3, pp. 239-245. <http://www.ais.fraunhofer.de/~klaassen/papers/Microel.J.pdf>, 1997.

[Korvink2003] – J. Korvink, E. Rudnyi, “Computer-Aided Engineering of Electro-Thermal MST Devices: Moving from Device to System Simulation”, 4<sup>th</sup> International Conference on Thermal and Mechanical Simulation and Experiments in Micro-Electronics and Micro-Systems, EUROSIME'03, France, 2003.

[Krauss1987] – A.D. Krauss, A. Bar-Cohen, “Thermal Analysis and Control of Electronic Equipment”, McGraw-Hill, 1987.

[Kundur1998] – D. Kundur, D. Hatzinakos, “A Novel Blind Deconvolution Scheme for Image Restoration Using Recursive Filtering”, IEEE Transactions on Signal Processing, Vol. 46, No. 2, Feb. 1998.

[Lai2000] – W. Lai, A. Krstic, K. Cheng, “On Testing the Path Delay Faults of a Microprocessor Using its Instruction Set”, Proc. of IEEE VLSI Test Symposium, May 2000.

[Lai2001] – W. Lai, K. Cheng, “Instruction-Level DFT for Testing Processor and IP Cores in Sistem-on-a-Chip”, DAC2001 - Design Automation Conference, 2001.

[Landis1996] – R. Landis, “Why NT and When CE”, Real-Time Magazine 98-3, 1996.

[Laquai1997] – B. Laquai, H. Richter, H. Werkmann, “A Production Oriented Measurement Method for Fast and Exhaustive IDDQ Tests”, ED&TC 1997.

- [Lee1996] – T. Lee, I.N. Hajj, E.M. Rudnick, J.H. Patel, “Genetic Algorithm Based Test Generation for Current Testing of Bridging Faults in CMOS VLSI Circuits”, IEEE VLSI Test Symposium (VTS). pp. 456-461, 1996.
- [Lee2000] – C. Lee, D.M.H. Walker, “PROBE: A PPSFP Simulator for Resistive Bridging Faults”. VLSI Test Symp., pp. 105-110, 2000.
- [Lee2003] – K.K. Lee, E.J. Paradise, S.K. Lim, “Thermal-Driven Circuit Partitioning and Floorplanning with Power Optimization”, CERCS Technical Reports, 2003.
- [Leonachescu1981] – N. Leonachescu, “Termotehnica”, Editura Didactica si Pedagogica, Bucuresti, 1981.
- [Li1998] – J. Li, E. McCluskey, “IDDQ Data Analysis Using Current Signature”, IEEE International Workshop on IDDQ Testing, November 1998.
- [Li2003] – Z. Li, X. Lu, W. Qiu, W. Shi, D.M.H. Walker, “A Circuit Level Fault Model for Resistive Bridges”, ACM Transactions on Design Automation of Electronic Systems, Vol. 8, No. 4. pp.546-559, Oct. 2003.
- [Li2003] – Z. Li, X. Lu, W. Qiu, W. Shi, D.M.H. Walker, “A Circuit Level Fault Model for Resistive Opens and Bridges”, Proceedings of 21st IEEE VLSI Testing Symposium (VTS), April 2003, pp. 379-384.
- [Liang1997] – B. Liang, S.U. Pillai, “Two-Dimensional Blind Deconvolution Using a Robust GCD Approach”, International Conference on Image Processing (ICIP’97), 1997.
- [Liao2003] – W. Liao, F. Li, L. He, “Microarchitecture Level Power and Thermal Simulation Considering Temperature Dependent Leakage Model”, ISLPED’03, Seoul, 2003.
- [Liao2004] – W. Liao, L. He, K. Lepak, “Temperature-Aware Performance and Power Modeling”, Technical Report UCLA 04-250, 2004.
- [Lienhard2004] – J.H. Lienhard, “A Heat Transfer Textbook”, 3<sup>rd</sup> Edition, Phlogiston Press, Cambridge, 2004.
- [Likas2003] – A. Likas, N.P. Galatsanos, “A Variational Method for Bayesian Blind Image Deconvolution”, Int. Conf. on Image Processing, Sep. 2003.
- [Lopez1998] – S. Lopez-Buedo, J. Garrido, E. Boemo, “Thermal Testing on Programmable Logic Devices”, IEEE ISCAS, Vol II, pp. 240-243, June 1998.
- [Lopez2000] – S. Lopez-Buedo, J. Garrido, E. Boemo, “Thermal Testing on Reconfigurable Computers”, IEEE Design&Test of Computers, pp. 84-91, Jan. 2000.
- [Lorch1998] – J. Lorch, A.J. Smith, “Software Strategies for Portable Computer Energy Management”, IEEE Personal Communications Magazine, 5(3):60-73, June 1998.
- [Lu2000] – Y. Lu, E. Chung, T. Simunic, L.Benini, G. De Micheli, “Quantitative Comparison of PM Algorithms”, Design, Automation and Test in Europe, pp. 20-26, 2000.

[Lubaszewski1998] – M. Lubaszewski, E. Cota, B. Courtois, “Microsystem Testing: an Approach and Open Problems”, Design Automation and Test in Europe, DATE98, pp. 524, France, Feb. 1998.

[Mahajan2002] – R. Mahajan, “Thermal Management of CPUs: A Perspective on Trends, Needs and Opportunities”, Thermic Workshop, Oct. 2002.

[Mahlstedt1992] – U. Mahlstedt, M. Heinitz, J. Alt, “Test Generation for IDDQ Testing and Leakage Fault Detection in CMOS Circuits”, European Design Automation Conference, pp. 486-491, Sept. 1992.

[Mahlstedt1995] – U. Mahlstedt, J. Alt, M. Heinitz. “CURRENT: A Test Generation System for IDDQ Testing”, 13<sup>th</sup> VLSI Test Symposium, pp. 317-323, April 1995.

[Makar1996] – S. Makar, E. McCluskey, “Some Faults Need an IDDQ Test”, IEEE International Workshop on IDDQ Testing, Apr. 1996.

[Maldague2000] – X. Maldague, “Applications of Infrared Thermography in NonDestructive Evaluation”, Trends in Optical Nondestructive Testing (invited chapter), Pramond Rastogi, pp. 591-609, 2000.

[Malhammar2002\_1] – A. Malhammar, “A Fourier Series Solution for the Temperature Distribution on Convection Cooled Plates with Discrete Heat Sources”, CoolingZone, [http://www.coolingzone.com/Guest/News/NL\\_MAY\\_2002/Ake/Ake\\_May2002.html](http://www.coolingzone.com/Guest/News/NL_MAY_2002/Ake/Ake_May2002.html), Mai 2002.

[Malhammar2002\_2] – A. Malhammar, “A Bessel function Solution for the Temperature Distribution on Convection Cooled Plates with Discrete Heat Sources”. CoolingZone, [http://www.coolingzone.com/Guest/News/NL\\_JUN\\_2002/Ake/Ake\\_June\\_2002.html](http://www.coolingzone.com/Guest/News/NL_JUN_2002/Ake/Ake_June_2002.html), Jun. 2002.

[Malhammar2002\_3] – A. Malhammar, “The Thermal Territory Concept”, CoolingZone, [http://www.coolingzone.com/Guest/News/NL\\_JULY\\_2002/Ake/Ake\\_July\\_2002.html](http://www.coolingzone.com/Guest/News/NL_JULY_2002/Ake/Ake_July_2002.html), Jul. 2002.

[Malhammar2002\_4] – A. Malhammar, “Thermal Component Models”, CoolingZone, [http://www.coolingzone.com/Guest/News/NL\\_DEC\\_2002/Ake/Ake\\_Dec2002.html](http://www.coolingzone.com/Guest/News/NL_DEC_2002/Ake/Ake_Dec2002.html), Dec. 2002.

[Manno1997] – V. P. Manno, “Integrated Thermal Network Models”, Electronics Cooling, Sep. 1997.

[Mao1990] – W. Mao, “QUIETEST: A Quiescent Current Testing Methodology for Detecting Leakage Faults”, IEEE International Conference on Computer-Aided Design (ICCCAD), pp. 280-283, 1990.

[Marculescu1999] – D. Marculescu, S. Haga, “Adaptive Program Execution for Low Power in Superscalar Processors”, Technical Report UMD-RT 99-10, Oct. 1999.

- [Maxwell1997] – P. Maxwell, P. O'Neill, R. Aitken, R. Dudley, N. Jaarsma, M. Quach, D. Wiseman. "Current Ratios: A Self-Scaling Technique for Production IDDQ Testing", IEEE International Workshop on IDDQ Testing, Nov. 1997.
- [Mazor1995] – S. Mazor, "The History of Microcomputer – Invention and Evolution", Proceedings of the IEEE vol. 83, no. 12, Decembrie 1995.
- [McCluskey1986] – E. McCluskey, "Logic Design Principles", Prentice Hall, 1986.
- [McCluskey2000] – E. McCluskey, C.W. Tseng, "Stuck-Fault Tests vs. Actual Defects", Proceedings of International Test Conference, 2000.
- [Meinders1999] – E.R. Meinders, G.M.P. van Kempen, L.J. van Vliet, T.H. van der Meer, "Measurement and Application of an Infrared Image Restoration Filter to Improve the Accuracy of Surface Temperature Measurements of Cubes", Experiments in Fluids, vol. 26, 1999.
- [Miao2000] – L. Miao, Z. Runde, G. Yuanqing, "A New Electro-Thermal Simulator Based on Relaxation Method for Integrated Circuits with Distributed Temperatures", ICDA 2000.
- [Michalski2001] – L. Michalski, K. Eckersdorf, J. Kucharski, J. McGhee, "Temperature Measurement", 2<sup>nd</sup> Edition, John Wiley & Sons Ltd., 2001.
- [Miczo2003] – A. Miczo, "Digital Logic Testing and Simulation", John Wiley & Sons, Inc., 2003.
- [Milanovic2000] – V. Milanovic, M. Hopcroft, C. Zincke, M. Zaghoul, K. Pister, "Modeling of Thermoelectric Effects in Planar Micromachined Structures Using SPICE", Int. Workshop on Thermal Investigations of IC's and Systems, THERMINIC'2000, Budapest, 2000.
- [Moayeri1996] – N. Moayeri, K. Konstantinides, "An Algorithm for Blind Restoration of Blurred and Noisy Images", HP Labs Technical Report, HPL-96-102, Jul. 1996.
- [Moganti1996] – M. Moganti, F. Ercal, C. Dagli, S. Tsunekawa, "Automatic PCI Inspection Algorithms: A Survey," Computer Vision and Image Understanding, Vol. 63, Nr. 2 , pp. 287-313, Mar. 1996.
- [Moldovan1999] – H. Moldovan, "Testarea termica a placilor cu circuite integrate", Teza Doctorat, 1999.
- [Mohapatra2002] – S. Mohapatra, N. Venkatasubramanian, "Optimizing Power Using Reconfigurable Middleware", Technical Report DSM\_TR03-04, Nov. 2002.
- [NVIDIA2001] – "Timer Function Performance", NVIDIA developer site, [http://www.nvidia.com/object/timer\\_function\\_performance.html](http://www.nvidia.com/object/timer_function_performance.html), Jul. 2001.
- [Negru1990] – L.D. Negru, "Transmiterea caldurii si dinamica gazelor", vol. 1, Litografiat IPTV, Timisoara 1990.



[Nigh1990] – P. Nigh, W. Maly, “Test Generation for Current Testing”, IEEE Design and Test of Computers, pp. 26-38, 1990.

[Nikolaidis2002] – S. Nikolaidis, N. Kavvadias, P. Neofotistos, K. Kosmatopoulos, T. Laopoulos, L. Bisdounis, “Instrumentation Set-up for Instruction Level Power Modeling”, Workshop on Power and Timing Modeling, Optimization and Simulation, Spain, 2002.

[OnNow2001] – “OnNow Power Management Architecture for Application”, Microsoft White Paper, Dec. 2001.

[Oney1999] – W. Oney, “Programming the Microsoft Windows Driver Model”, Microsoft Press, 1999.

[Qstar2002] – Q-Star Test, “IDDQ Efficiency”. [http://www.qstar.be/html/iddq\\_efficiency.html](http://www.qstar.be/html/iddq_efficiency.html), 2002.

[Ragen1995] – M. Ragen, “Real-Time Systems With Microsoft Windows NT”, MSDN Technical Articles, [http://msdn.microsoft.com/library/en-us/dndllpro/html/msdn\\_realtime.asp](http://msdn.microsoft.com/library/en-us/dndllpro/html/msdn_realtime.asp) Jun. 1995.

[Rajsuman2000] – R. Rajsuman, “IDDQ Testing for CMOS VLSI”. Proceedings of IEEE. Vol. 88, No. 4, April 2000.

[Rakhmatov2002] – D. Rakhmatov, S. Vrudhula, C. Chakrabarti, “Battery-Conscious Task Sequencing for Portable Devices Including Voltage/Clock Scaling”, Design Automation Conference, DAC2002, June 2002.

[Ramamritham1998] – K. Ramamritham, C. Shen, O. Gonzalez, S. Sen and S. Shirgurkar, “Using Windows NT for Real-Time Applications: Experimental Observations and Recommendations”, IEEE Real-Time Technology and Applications Symposium, 1998.

[Ramanathan2002] – D. Ramanathan, S. Irani, R. Gupta, “An Analysis of System Level Power Management Algorithms and Their Effects on Latency”, IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, March 2002.

[Reddy1995] – R.S. Reddy, I. Pomaranz, S.M. Reddy, S. Kajihara, “Compact Test Generation for Bridging Faults Under IDDQ Testing”, IEEE VLSI Test Symposium (VTS), pp. 310-316, 1995.

[Remsburg2001] – R. Remsburg, “Thermal Design of Electronic Equipment”. CRC Press LLC, 2001.

[Renovell1995] – M. Renovell, P. Huc, Y. Bertrand, “The Concept of Resistnace Interval: A New Parametric Model for Resistive Bridging Fault”, VLSI Test Symposium (VTS), pp. 184-189, 1995.

[Rosenfeld1982] – A. Rosenfeld, A.C. Kak, “Digital Image Processing”, Academic Press, 1982.

- [Rudnick1994] – E.M. Rudnick, J.H. Patel, G.S. Greenstein, T.M. Niermann, “Sequential Circuit Test Generation in a Genetic Algorithm Framework”, 31<sup>st</sup> Design Automation Conference, Jun. 1994.
- [Sexton2001] – J. Sexton, “IR Inspection Finds Elusive Faults”, Test & Measurement World, Aug. 2001.
- [SMBus2000] – “System Management Bus (SMBus) Specification Version 2.0”, Aug. 2000.
- [Sabade2002] – S. Sabade, H. Walker, “Evaluation of Outlier Rejection Methods for IDDQ Testing”, ASP-DAC/VLSI Design, Jan. 2002.
- [Sachdev1997] – M. Sachdev, “Deep Sub-micron IDDQ Testing: Issues and Solutions”, ED&TC 1997.
- [Sexton2001] – J. Sexton, “Infrared Inspection Finds Elusive Faults”, Test & Measurement World. 2001.
- [Simunic1999] – T. Simunic, L. Benini, G. De Micheli, “Energy-Efficient Design of Battery-Powered Embedded Systems”, Int. Symposium on Low Power Electronics and Design, pp. 212-217, 1999.
- [Simunic2001] – T. Simunic, L. Benini, P. Glynn, G. De Micheli, “Event-driven Power Management”, IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, Vol. 20, No. 7, Jul. 2001.
- [Skadron2003] – K. Skadron, M. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, D. Tarjan, “Temperature-Aware Microarchitecture: Extended Discussion and Results”, 30<sup>th</sup> International Symposium on Computer Architecture (ISCA-30), Canada, June 2003.
- [Sofia1995] – J.W. Sofia, “Analysis of Thermal Transient Data with Synthesized Dynamic Models for Semiconductor Devices”, IEEE Computers of Hybrids and Manufact. Tech., Mar. 1995.
- [Sofia1996] – J.W. Sofia, “Fundamentals of Thermal Resistance Measurement”, [http://www.analysisstech.com/PDF Files/fundamen.pdf](http://www.analysisstech.com/PDF%20Files/fundamen.pdf), 1996.
- [Sofia2000] – J.W. Sofia, “Trends in Electronics Reliability Testing”, <http://www.analysisstech.com/Trends.ppt>, 2000.
- [Solomon2000] – D. Solomon, M. Russinovich, “Inside Microsoft Windows 2000”, Third Edition, Microsoft Press, 2000.
- [Sony2001] – “Quality and Reliability HandBook”, HandBook, Sony Semiconductor, 2001. (<http://www.sony.net/Products/SC-HP/tec/catalog/qr.html>)
- [Szekely1997] – V. Szakely, “CMOS Sensors for On-Line Thermal Monitoring of VLSI Circuits”, IEEE Trans. VLSI Systems, Vol. 5, No. 3, pp.1-9, 1997.

[Szekely1998] – V. Szekely, M. Rencz, B. Courtois, “Tracing the Thermal Behavior of ICs”, IEEE Design & Test of Computers, April-June 1998.

[Szekely1998\_1] – V. Szekely, “Thermal Testing and Control by Means of Built-In Temperature Sensors”, CoolingZone, Sep. 1998.

[TI1999] – “Package Thermal Characterization Methodologies”, Texas Instruments – Application Report, 1999.

[Tadayon2000] – P. Tadayon, “Thermal Challenges During Microprocessor Testing”, Intel Technology Journal, 2000.

[Tatibana1997] – M.H. Tatibana, R.A. Lotufo, “Novel Automatic PCB Inspection Technique Based on Connectivity”, Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens, SIBGRAPI'97, Oct. 1997.

[Thadikaran1993] – P. Thadikaran, S. Chakravarty, “Simulation and Generation of IDDQ Tests for Bridging Faults in Combinational Circuits”, IEEE VLSI Test Symposium (VTS), pp. 25-32, 1993.

[Thadikaran1996] – P. Thadikaran, S. Chakravarty, “Fast Algorithms for Computing IDDQ Tests for Combinational Circuits”, IEEE International Conference on VLSI Design, pp. 103-106, 1996.

[Thibeault1999] – C. Thibeault, “On the Comparison of Delta IDDQ and IDDQ Testing”, IEEE VLSI Test Symposium, p. 143-150, Apr. 1999.

[Transmeta2002] – “Crusoe TM5500/TM5800 Thermal Design Guide”, Transmeta Processor Specification, June 2002.

[Veijola1995] – T. Veijola, M. Valtonen, “An Object-Oriented Approach to Combined Electrical and Thermal Circuit Analysis”, Proceedings of ECCTD'95, Istanbul, pp. 1055-1058, Aug. 1995.

[Viswanath2000] – R. Viswanath, V. Wakharkar, A. Watwe, V. Lebonheur, “Thermal Performance Challenges from Silicon to Systems”, Intel Technology Journal, 2000.

[Wang2004] – T.Y. Wang, C.C. Chen, “SPICE-Compatible Thermal Simulation with Lumped Circuit Modeling for Thermal Reliability Analysis Based on Modeling Order Reduction”, 5<sup>th</sup> International Symposium on Quality Electronic Design, ISQED'04, March 2004.

[Wiklund1998] – K. Wiklund, “Relating IC Power Dissipation to Circuit Reliability”, <http://www.dtek.chalmers.se/~kw/papers/thermal.pdf>, 1998.

[Xiaoqing1998] – W. Xiaoqing, H. Tamamoto, K. Saluja, K. Kinoshita, “Transistor Leakage Fault Diagnosis with IDDQ and Logic Information”, IEICE Trans. Inf. & Syst., Vol. E81-D, Apr. 1998.

[Yamazaki1996] – H. Yamazaki, Y. Miura, “IDDQ Testability of Flip-Flop Structures”, IEEE International Workshop on IDDQ Testing, IDDQ'96, October 1996.

[Zhang2002] – Y. Zhang, X. Hu, D. Chen, “Task Scheduling and Voltage Selection for Energy Minimization”, DAC2002, Jun. 2002.

[Zwicker2001] – K. Zwicker, “Hardware Code Morphing”, Transmeta Technical Documentation, Aug. 2001.

# Articole publicate de autor

- [1] M. Marcu, M. Vladutiu, H. Moldovan, "PCB Testing Using Infrared Thermal Signatures", Accepted to 22<sup>nd</sup> IEEE Instrumentation and Measurement Technology Conference IMTC2005, Ottawa, Canada, May. 2005.
- [2] M. Marcu, H. Moldovan, V. Tomescu, M. Zilahi, M. Ionas, "Windows XP Issues for Real-Time Temperature Monitoring", Accepted to 22<sup>nd</sup> IEEE Instrumentation and Measurement Technology Conference IMTC2005, Ottawa, Canada, May. 2005.
- [3] M. Marcu, M. Vladutiu, H. Moldovan, "Blind Deconvolution of PCB Thermal Images", Accepted to Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition, HACIPPR2005, Ungaria, May 2005.
- [4] M. Marcu, V. Tomescu, M. Zilahi, M. Ionas, "Using Windows NT Based Operating Systems for Real-Time Temperature Monitoring", Third IEEE International Conference on Systems, Signals & Devices SSD'05, Sousse, Tunisia, Mar. 2005.
- [5] M. Marcu, M. Vladutiu, H. Moldovan, "Thermal Testing of Printed Circuit Boards Using Thermal Signatures", Third IEEE International Conference on Systems, Signals & Devices SSD'05, Sousse, Tunisia, Mar. 2005.
- [6] M. Marcu, M. Vladutiu, "A Genetic Algorithm for Thermal Image Deconvolution", 2004, Iranian Journal of Electrical and Computer Engineering IJECE2004, 2004.
- [7] M. Popa, M. Marcu, A. Popa, "A Microcontroller Based Data Acquisition System With USB Interface", IEEE International Conference on Electrical, Electronic and Computer Engineering ICEEC'04, Egipt, Sep. 2004.
- [8] M. Vladutiu, M. Marcu, "Printed Circuit Boards Testing Using Thermal Signatures", 6<sup>th</sup> International Conference on Technical Informatics, CONTI2004, Timisoara, Romania, May 2004.
- [9] M. Popa, M. Marcu, S. Fuicu, A. Jucu, "Microcontroller Based Programmable Logic Controller", 6<sup>th</sup> International Conference on Technical Informatics, CONTI2004, Timisoara, Romania, May 2004.
- [10] M. Vladutiu, M. Marcu, "Pseudo-Genetic Algorithm for Thermal Image Deconvolution", Buletinul Stiintific UPT, Tom 48 (62), 2003.
- [11] M. Vladutiu, H. Moldovan, M. Marcu, "A New Method of Testing Using Infrared Images", International Conference on Technical Informatics, CONTI2000, Timisoara, Romania, 2000.
- [12] M. Marcu, S. Fuicu, "I<sub>DDQ</sub> Testing of Deep Sub-micron Devices", International Conference on Technical Informatics, CONTI2000, Timisoara, Romania, 2000.
- [13] M. Vladutiu, H. Moldovan, M. Marcu, "Ideal Thermal Image Generation", Buletinul Stiintific UPT, Tom 41 (55), 1996.
- [14] M. Vladutiu, H. Moldovan, M. Marcu, "Current Calculation from Thermogram Using a Bi-Directional Model for Heat Transfer", Buletinul Stiintific UPT, Tom 41 (55), 1996.
- [15] M. Vladutiu, H. Moldovan, M. Marcu, O. Bota, "Current Calculation from Thermogram Using a Unidirectional Model for Heat Transfer", Buletinul Stiintific UPT, Tom 41 (55), 1996.