

# GENERADOR DE SEQÜÈNCIES DE TEST PER CIRCUITS INTEGRATS NMOS

CARLES FERRER I RAMIS, JEAN PIERRE DESCHAMPS,  
JOAN OLIVER I MALAGELADA, JORDI CARRABINA I BORDOLL,  
ELENA VALDERRAMA I VALLÉS

Universitat Autònoma de Barcelona

*El generador de seqüències de test que es presenta en aquest article utilitza una descripció del circuit a nivell de transistor representant les xarxes de transistors d'enriquiment de les funcions lògiques NMOS mitjançant grafs no orientats. Per la generació de vectors de test s'empren l'algorisme-D, havent desenvolupat un mètode enumeratiu de recerca de camins al damunt del graf, a partir del camí mínim, que passa pel flanc que representa el transistor afectat per la falta.*

**Test pattern generator for NMOS integrated circuits**

**Keywords:** ATPG, D-Algorithm, Fault detection, Switch-level, non-oriented graphs.

## 1. INTRODUCCIÓ

Durant les etapes de fabricació d'un circuit integrat, s'introdueixen un cert nombre de faltes, defectes que poden provocar un mal funcionament; per tal d'augmentar el rendiment en la fabricació i la qualitat dels circuits, aquests s'han de sotmetre a una fase de comprovació que permeti discernir quins poden produir errors i per tal de rebutjar-les.

Per comprovar el bon funcionament d'un circuit integrat es requereix verificar les sortides que dona el circuit en presència d'una seqüència determinada a les seves entrades. Si el circuit no és massa complex es poden comprovar exhaustivament totes les combinacions possibles de les seves entrades. En circuits VLSI, degut a l'elevada densitat d'integració, s'introdueixen nous problemes:

---

- Carles Ferrer i Ramis, Jean Pierre Deschamps, Joan Oliver i Malagelada, Jordi Carrabina i Bordoll, Elena Valderrama i Vallés - Universitat Autònoma de Barcelona - Dep. d'Informàtica - 08193 Bellaterra - Barcelona.

1. D'una banda, el nombre de dispositius dins d'un únic CI es tant elevat, que una comprovació exhaustiva de tots els possibles estats del circuit és impensable.
2. El nombre d'entrades i de sortides en un CI és molt baix en comparació amb el nombre de nodes i dispositius interns, la qual cosa porta a un baix nivell de controlabilitat i observabilitat; a diferència de les plaques de circuit imprès, els nodes interns són inaccessibles directament.
3. D'altra banda, els models clàssics de faltes ("stuck-at") emprats a nivell lògic no poden reproduir els defectes que aparèixen a nivell elèctric.

En aquest sentit, darrerament es tendeix al desenvolupament de generadors de test que treballen a nivell elèctric. Dues formes d'atacar el problema aparèixen clarament:

- Els que actuen a nivell elèctric directament, tractant les xarxes de transistors com a grafs, utilitzant algorismes per a grafs, en les diferents etapes de la generació [1].
- Els que representen els circuits MOS per circuits equivalents a nivell de portes lògiques, utilitzant llavors, algorismes generadors de seqüències de test a aquest nivell [8].

En aquest article presentem un generador automàtic de seqüències de test per circuits integrats NMOS, desenvolupat en el Departament D'Informàtica de la U.A.B./ Centre Nacional de Microelectrònica. C.S.I.C.[5]. Aquest generador segueix la primera línia, de les explicades anteriorment, donant, a partir d'una descripció del circuit a nivell de transistors, la seqüència de vectors de test necessària per detectar cadascuna de les faltes considerades pel model. Donada la complexitat dels circuits VLSI, es bàsic trobar una estructura de dades que s'adapti bé a la descripció del circuit i a la búsqueda de seqüències de test. A continuació descriurem el generador, fent especial referència a la seva estructura informàtica.

## 2. MODEL DE REPRESENTACIÓ I DE FALTES

La representació d'un circuit integrat MOS es realitza a nivell de transistor, ja que és la que millor reflexa les particularitats d'aquests tipus de circuits.

En aquesta representació agafa tal qual el circuit i se li associa un graf no orientat a cadascuna de les funcions del circuit [1].

La xarxa de transistors de senyal és representada mitjançant un graf on cada flanc representa un transistor, i la variable associada a ell és la línia que comanda la porta del transistor. Cada vèrtex representa el node d'interconnexió entre transistors.

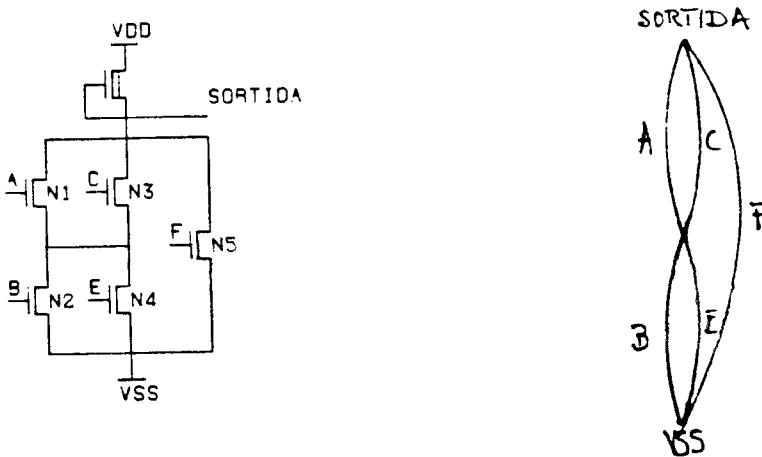


FIGURA 1. Representació amb un graf d'una funció lògica NMOS

Els tipus de faltes que es modelen sobre aquest tipus de representació són:

- i Fixació a 0 i a 1 de la sortida de la porta MOS.
- ii Fixació a 0 i a 1 de l'entrada de la porta MOS.
- iii Transistor MOS permanentment obert o tancat.

L'afecte dels dos primers tipus de faltes consisteix en deixar a un valor fixe la línia afectada, mentre que el tercer tipus deixa obert o tancat el transistor afectat, sigui quin sigui l'estat al qual es trobi la seva porta.

### 3. ALGORISMES GENERADORS DE VECTORS DE TEST

La majoria d'algorismes generadors treballen sobre circuits descrits a nivell de portes lògiques, per la qual cosa s'ha cregut més convenient d'efectuar una adaptació de l'algorisme-D [2], per tal de fer-lo treballar sobre una descripció a nivell de transistor.

Nogensmenys s'han introduït certs criteris heurístics que faciliten la tasca i l'eficiència de l'algorisme.

- Generació del senyal d'error a la sortida de la funció afectada per la falta.
- Propagació d'aquest senyal d'error fins a una sortida primària del circuit.

- Justificació de totes les entrades primàries.
- Recerca d'inconsistències.

D'altres algorismes generadors de vectors de test, molt eficients sobre circuits lògics descrits amb portes lògiques, com poden ser els algorismes PODEM [7] i FAN [6] no s'adapten bé a la generació sobre circuits formats per funcions lògiques NMOS.

Així doncs, l'algorisme utilitzat per la generació de vectors de test funciona a dos nivells. La generació del senyal d'error s'efectua a nivell de la funció lògica basant-se en un mètode enumeratiu de recerca de camins actius sobre el graf de la marxa de transistors; mentre que les altres dues etapes de propagació i de justificació actuen a nivell del circuit.

#### 4. EXTRACCIÓ LÒGICA I ESTRUCTURA DE DADES.

Dels circuits NMOS pels quals es volen generar els vectors cal conèixer-ne la distribució dels transistors que componen llur interconnexió, així com quines són les entrades i les sortides primàries del circuit i les restriccions que s'han de prendre en algunes de les entrades.

Els transistors del circuit vindran descrits d'una forma semblant a com es faria si es volés simular amb el simulador SPICE [9], a partir de la sortida d'un extractor elèctric. Nogensmenys si no es disposa d'aquesta informació, llavors cal fer una numeració de tots els nodes d'interconnexió, d'acord amb la nomenclatura abans exposada.

En tot cas caldrà afegir quines són les entrades i sortides primàries i a quin estat prefixat es troben les entrades primàries, si no és així resten en un estat indeterminat fins que el generador les hi assigni l'estat.

D'aquesta descripció del circuit se n'obtindrà una estructura de dades que reflexarà la distribució de transistors i el funcionament del circuit, i que representarà el circuit a nivell de commutació.

Un cop descrit el circuit es fa una extracció lògica. Això vol dir obtenir totes les funcions lògiques i transistors de pas de que es compon el circuit i a més, per cadascun d'aquests elements, buscar quines funcions o transistors de pas estan connectats a les seves entrades i a les seves sortides ja que això és molt útil durant la generació de vectors de test.

Per realitzar aquesta estructura de dades s'han utilitzat les estructures dinàmiques que permet el llenguatge PASCAL, a base de les variables tipus apuntador [3], [10].

#### 4.1. DESCRIPCIÓ DEL CIRCUIT

Els tres tipus d'informació que necessita el generador són la descripció de cada transistor de senyal o de càrrega quines són les entrades i les sortides primàries del circuit i el valor prefixat, si existeix, al qual es troben les entrades primàries.

L'única diferència respecte d'un fitxer d'entrada SPICE rau en l'especial forma en que es declaren les entrades i sortides primàries. En tot cas, les línies d'un fitxer d'entrada que no comencin per E, S o M, entrades i sortides primàries i descripció dels transistors respectivament, no tenen cap efecte pel generador, així com la informació adicional que normalment s'afegeix en la descripció d'un transistor com poden ser àrees, capacitats, etc.

Això permet facilitar a l'usuari la descripció del circuit, en ésser l'SPICE un simulador força utilitzat, no suposant un esforç adicional la descripció del circuit, quan prèviament s'hagi fet alguna simulació amb SPICE, només cal afegir la descripció d'entrades i sortides primàries.

#### 4.2. EXTRACCIÓ LÒGICA

La següent etapa, prèvia a la generació de l'estructura de dades que ha de descriure el circuit, consisteix en obtenir, a partir d'aquest fitxer on es descriuen tots els transistors del circuit, totes les funcions lògiques, transistors de pas i llurs interconnexions. Això és totalment imprescindible ja que el generador de vectors de test treballa sobre aquests elements constituents i no sobre tot el circuit alhora.

Per tal de trobar totes les funcions lògiques del circuit es parteix del transistor de càrrega que és fàcilment localitzable, tant perquè és d'un tipus diferent, com perquè té la porta connectada a la font. Aquest node és molt important ja que a ell va connectada la xarxa de transistors de senyal i és, alhora, la sortida de la funció lògica.

Inicialment s'extreuen les funcions lògiques, buscant quina xarxa de transistors està connectada entre la sortida de la funció i terra. Per trobar cada xarxa es busquen inicialment els transistors que tinguin la font o el drenador en comú amb la sortida de la funció. Posteriorment es buscarien tots els transistors que tinguin un node en comú amb l'altre extrem dels transistors que s'han trobat en el pas anterior, sempre que aquest extrem no estigui connectat a terra. Procedint recurrentment fins obtenir la xarxa de transistors.

La resta de transistors que no formin part de les funcions lògiques seran els transistors de pas dels quals s'haurà de buscar la seva entrada i la seva sortida. Això dependrà d'on estiguin connectats el drenador i la font, ja que la línia de control (la porta del transistor) sempre serà una entrada, primària o no.

### 4.3. ESTRUCTURA DE DADES

Degut al fet que un transistor MOS és un dispositiu no direccional, en el que a priori no es pot distingir el drenador de la font, excepte quan es coneix molt bé la situació del transistor en el circuit, no es pot determinar correctament la seva direccionalitat. Així doncs, s'ha intentat que l'estructura que representa el circuit mantingui el màxim grau de llibertat, no quedant en cap moment limitat el funcionament del transistor.

La millor forma de representar a un circuit NMOS, és mitjançant una llista de totes les funcions lògiques i de tots els transistors de pas que el componen. Els transistors de pas estan perfectament orientats i no representa cap limitació el fet de tenir-los en una llista. En canvi, cadascuna de les xarxes de transistors de senyal de cada funció es representa per un graf no direccional, com els explicats en l'apartat 2, el qual conserva tota la llibertat de la xarxa.

Cada graf ve representat per una llista de tots els vèrtex d'on cadascun d'ells es pot accedir a una llista de tots els flancs que parteixen d'ell.

Per tal de facilitar la generació dels vectors de test, es creen dues llistes relacionades amb les etapes de la propagació del senyal d'error i la justificació de les entrades.

Per facilitar la propagació es crea, per cada funció lògica i cada transistor de pas, una llista de les funcions que tenen almenys una entrada igual a la sortida de la funció o transistor de pas. D'aquesta manera, es pot saber a priori per quines altres funcions es podrà propagar la senyal d'error.

Aquesta llista s'ordena seguint un criteri que facilita la tasca de la propagació del senyal d'error, adjudicant a cada funció un pes igual al nombre de flancs del graf de la pròpia funció més el pes mínim d'entre totes les funcions per les quals es pugui propagar des de la seva sortida, excepte en el cas de que la sortida d'aquesta funció sigui primària, en que es suma zero. Un cop calculat els pes de cada funció, aquestes s'ordenen pel seu pes, des del més petit fins al més gran.

L'utilització d'aquestes llistes per la propagació del senyal d'error d'una funció a una altra, correspon a un dels criteris heurístics emparats en l'aplicació de l'algorisme generador de vectors de test. El criteri consisteix en provar de propagar sempre primer pels camins de cost mínim, menor nombre de flancs i menor distància a una sortida primària.

D'altre banda per a cada funció es realitza una llista de totes les funcions que tenen connectada la sortida a alguna de les entrades de la funció cara a la posterior fase de justificació. Aquesta llista no caldrà ordenar-la pel fet de que s'han de justificar totes les entrades no primàries de cada funció.

Per a la construcció de cada vector de test s'utilitzen dos altres tipus de llistes. La primera és una llista ordenada en forma d'arbre per a cada funció, on es

posen tots els flancs de cada funció ordenats pel número de node corresponent a la variable de comandaments associada, que és alhora entrada de la funció. La segona és una llista ordenada de totes les entrades i sortides de les funcions lògiques, on s'assignaran els estats tal i com es vagin trobant. Inicialment tots els estats estan desassignats, excepte en el cas de tenir estats preassignats de les entrades primàries.

El muntatge de les llistes y llurs connexions, com es pot veure en la figura 2, s'ha disposat de forma que la llista bàsica a partir de la qual s'accedeix a les altres sigui la llista de les funcions del circuit. Des de cada node d'aquesta llista es té accés al graf de la xarxa de transistors de senyal, a la llista ordenada de flancs, a la llista ordenada per a la propagació, i a la llista de funcions a justificar,  $A_i$ ,  $B_i$  y  $C_i$  respectivament en la figura 2. També, tant directament com a través de les altres llistes, es pot accedir a la llista ordenada de les entrades i sortides de totes les funcions del circuit.

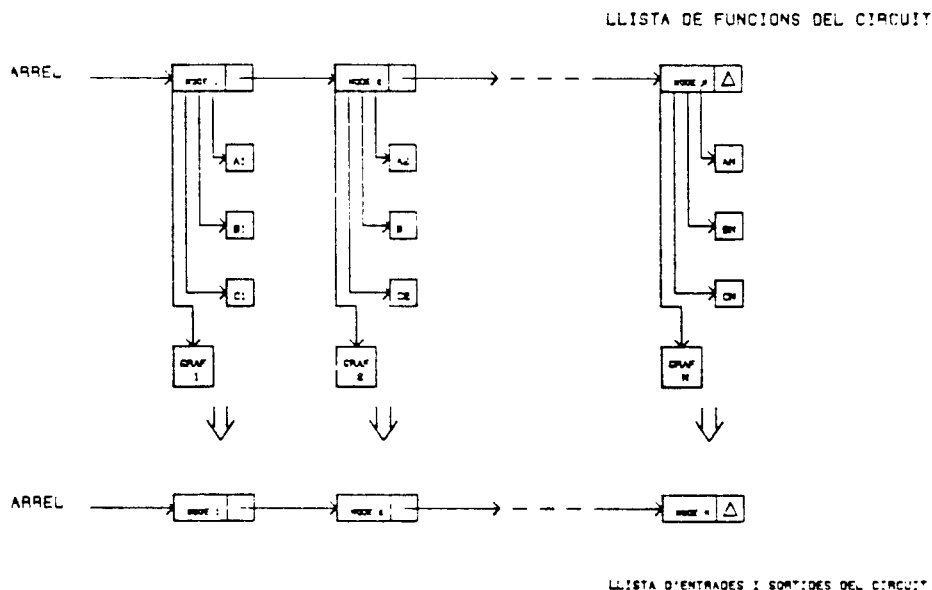


FIGURA 2. Esquema de l'estructura de dades

## 5. GENERADOR DE VECTORS DE TEST.

El generador desenvolupat busca un vector de test que sigui capaç de detectar la presència d'una falta. Això és possible si almenys alguna sortida primària canvia d'estat segons si la falta és o no present.

En un principi, totes les faltes són detectables, excepte aquelles que deixen els transistors de càrrega permanentment tancats ja que la diferència de potencial entre la porta i la font es de zero volts (perquè es troben connectades) essent sempre superior a la tensió llindar ( $V_T < 0$ ) y per tant sempre conduirà.

El generador, per cada falta, executa consecutivament les següents tres etapes: generació del senyal d'error, propagació del senyal d'error i justificació de les entrades primàries. La recerca d'inconsistències es realitza cada cop que es mira una funció per poder prosseguir amb l'algorisme, per tal d'estar segur, de que les assignacions introduïdes en noves línies no afecten per res els estats de sortida de les funcions seleccionades en passos anteriors.

### 5.1. GENERACIÓ DEL SENYAL D'ERROR

La generació del senyal d'error consisteix en cercar una combinació de les entrades de la funció, capaç de posar en evidència la presència de la falta en la funció amb un canvi d'estat a la sortida de la funció.

Fer funcions amb xarxa de transistors de senyal, s'utilitza un mètode enumeratiu de recerca de camins al damunt d'un graf. Aquest mètode troba un camí que passa pel transistor afectat per la falta i garanteix que en cap moment apareixerà un altre camí paral·lel directe entre la sortida i terra que emascararà l'estat de sortida de la funció lògica.

La búsqueda d'aquest camí es realitza adaptant l'algorisme de Moore de recerca de camins mínims sobre un graf no orientat [4], aplicant una enumeració de tots aquells possibles que passen pel flanc del transistor afectat.

Pels transistors de pas es consideren les faltes que els deixen permanentment oberts o tancats, per això són necessaris dos vectors de test degut a la memorització de l'estat emmagatzemat en la porta de transistor, a la qual es troba connectada la sortida del transistor de pas. Per les faltes de fixació a zero o a  $u$  en la seva entrada o sortida, només cal que propagui el senyal d'error generat, posant-lo a conduir.

### 5.2. PROPAGACIÓ DEL SENYAL D'ERROR

L'etapa de propagació del senyal d'error pretén fer observable el senyal d'error, generat en l'etapa anterior, utilitzant per aquesta tasca les llistes de funcions a propagar creades, per cada funció, en la fase de construcció de l'estructura de dades pel generador.

El procés comença intentant propagar el senyal d'error per alguna de les funcions de la lista de funcions a propagar de la funció a la qual s'ha generat el senyal d'error, excepte si la sortida d'aquesta funció és primària. Si és possible s'ha de continuar aquest pas iterativament fins arribar a una sortida primària.



Per propagar el senyal d'error per una funció s'ha d'escollir una combinació de les seves entrades que dongui un senyal d'error a la sortida. El mètode utilitzat és altre cop la recerca de camins aplicant l'algorisme de Moore modificat.

### 5.3. JUSTIFICACIÓ DE LES ENTRADES PRIMÀRIES

Un cop s'ha aconseguit propagar, comença l'etapa de justificació de totes les entrades de les funcions, de les quals s'ha estimat l'estat necessari, i que corresponen a sortides d'altres funcions per les que no s'ha determinat si l'actual estat de les seves entrades permet o no obtenir l'estat de sortida desitjat. En aquesta etapa es fan servir les llistes de funcions a justificar creades anteriorment que ens permeten saber quines de les entrades d'una funció depenen de sortides d'altres funcions i no directament d'entrades primàries, les quals es poden forçar a l'estat que determini el generador, mentre que les altres dependran del valor donat per la funció connectada a la seva entrada.

El que cal fer és escollir les combinacions de les entrades de les funcions a justificar que permetin donar el valor de la sortida desitjat i que alhora no apareguin diferències amb altres valors assignats, tant per línies internes com per entrades primàries.

### 5.4. RECERCA D'INCONSISTÈNCIES.

En qualsevol de les etapes anteriors poden aparèixer inconsistències, en el sentit que, es volen assignar valors diferents als assignats per funcions anteriors. Llavors, si els valors determinats no són assignables o provoquen canvis en les sortides d'altres funcions prèviament utilitzades, es detecta una inconsistència.

En aparèixer una inconsistència s'ha d'escollir una altra combinació de les entrades de la funció, que igualment detecti la falta.

## 6. CONCLUSIONS

El generador desenvolupat s'ha aplicat a diferents tipus de circuits combinatorials com ara sumadors i multiplicadors, entre d'altres [5]. Els principals resultats assolits consisteixen en obtenir una bona cobertura de faltes, com pels exemples de la taula 1, en el càlcul d'aquests cobriments no s'han comptat com a faltes no detectades aquelles que deixen conduint els transistors de càrrega.

TAULA 1: COBRIMENT DE FALTES I TEMPS DE CÀLCUL SOBRE VAX/750

Circuit	Cobriment de faltes	temps de CPU
sumador1(4 bits)	99.75%	19.6s
sumador2(2x2bits)	94.12%	29.0s
multiplicador(2x2 bits)	96.97%	2m35s

Ara bé, en el cas d'augment del tamany o de disminució de la controlabilitat i observabilitat del circuit llavors la cobertura de faltes tendeix a disminuir i el temps de càlcul a augmentar.

D'altre banda si es tenen entrades prefixades la cobertura de faltes es redueix gairebé a la meitat. Per exemple, en el cas del multiplicador (2x2 bits), la cobertura es redueix fins arribar al 47,5%.

Degut a la manca de resultats de cobertures de faltes per generadors de seqüències de test a nivell de transistor, no es pot donar una comparació del mètode que aquí presentem amb d'altres de similars. Les úniques explicacions existents són manuals sobre portes NMOS o petits circuits.

Pel que respecte a generadors de seqüències de test que treballin a nivell de portes lògiques, no és suficient comparar només el generador, ja que també cal considerar la fase de traducció del circuit representat amb transistors a un circuit equivalent amb portes lògiques. Aquesta no introdueix desavantatges en quan la cobertura de faltes, però no es pot evaluar el cost de realització del generador ja que depèn d'aquesta fase i del cost introduït pel propi generador, per a comparar-ho amb el genrador que aquí presentem.

Properament pensem adaptar aquest mètode, a la generació de seqüències de test en circuits integrats CMOS.

## 7. REFERENCIAS BIBLIOGRAFICAS

- [1] **Agrawal, P.** "Test generation at switch level". IEEE International Conference on Computer Aided Design, Santa Clara, CA, November 12-15, 1984.
- [2] **Breuer, M.A.; Friedman, A.D.** "Diagnosis and reliable design of digital systems". Computer Science Press, INC 1976.
- [3] Vax/11 Pascal Installation Guide/Realease Notes and Language Reference Manual. Digital Equipment Corporation 1979.
- [4] **Even, S.** "Graph Algorithms". Pitmann 1979.
- [5] **Ferrer, C.** "Generació de vectors de test per circuits NMOS". Tesi de Llicenciatura, Universitat Autònoma de Barcelona, Facultat de Ciències (Secció de Físiques), Octubre 1986.
- [6] **Fujiwara, H., Shimono, T.** "On the acceleration of test generation algorithms". IEEE Transactions on Computers, vol C-32, No. 12, December 1983, pp 1137-1144.
- [7] **Goel, P.** "An implicit enumeration algorithm to generate tests for combinational logic circuits". IEEE Transactions on Computers, vol C-30, No. 3, March 1981, pp 215-222.
- [8] **Jain, S.K.; Agrawal V.D.** "Modeling and test generation algorithms for MOS circuits" IEEE Transactions on Computers, vol C-34, No. 5, May 1985, pp 426-433.
- [9] **Newton, A.R.** "The simulation of large-scale integrated circuits". Memorandum No. UCB/ERC M78/52. University of California 1978.
- [10] **Wirth, N.** "Estructuras de datos + algoritmos = programas". Ediciones del Castillo 1980.

