UNIVERSIDAD DE GRANADA

Facultad de Ciencias

Departamento de Electrónica

UNIVERSIDAD DE GRANADA

FACULTAD DE CIENCIAS

Núm. .....

Tengo el gusto de remitir 1 ejemplar - de la Tesis Doctoral de D. FRANCISCO JOSE PELAYO VALLE, para su archivo - en la Biblioteca de esta Facultad.

Dios guarde a Vd. muchos años.

Granada, 16 de Noviembre de 1.990

EL SECRETARIO,

UNIVERSIDAD DE GRANADA — FACULTAD DE CIENCIAS

Sr.Director de la Biblioteca de esta Facultad=-

T.R. 10/106
T
13
130

CONTRIBUCION AL DISEÑO DE CIRCUITOS INTEGRADOS MULTIUMBRALES Y MULTIVALUADOS

MEMORIA PARA OPTAR AL GRADO DE
DOCTOR EN CIENCIAS FÍSICAS

FRANCISCO JOSÉ PELAYO VALLE

BIBLIOTECA UNIVERSITARIA
GRANADA
Nº Documento 0196577778
Nº Copia 1212039 70

GRANADA, JUNIO DE 1989

DEPARTAMENTO DE ELECTRÓNICA Y
TECNOLOGÍA DE COMPUTADORES

UNIVERSIDAD DE GRANADA

Alberto Prieto Espinosa, y
Antonio Lloris Ruiz
Profesores Titulares del Dpto.
de Electrónica y Tecnología
de Computadores de la Univer
sidad de Granada

CERTIFICAN:

Que la memoria titulada: "CONTRIBUCION AL DISEÑO DE CIRCUITOS INTEGRADOS MULTIUMBRALES Y MULTIVALUADOS" ha sido realizada por D. Francisco José Pelayo Valle bajo nuestra dirección, en el Departamento de Electrónica y Tecnología de Computadores de la Universidad de Granada. Esta memoria constituye la tesis que D. Francisco José Pelayo Valle presenta para optar al gra do de Doctor en Ciencias Físicas.

Granada, 20 de Mayo de 1989.

Fdo. Alberto Prieto Espinosa
Codirector de la Tesis

Fdo. Antonio Lloris Ruiz
Codirector de la Tesis

A MI FAMILIA

Agradezco a Alberto Prieto y Antonio Lloris su orientación y ayuda constante en la preparación de este trabajo.

A Pedro Cartujo, Juan Enrique Carceller, y todos los compañeros del Departamento de Electrónica y Tecnología de Computadores agradezco su interés, ayuda, y estímulo para la consecución de mi doctorado. Especialmente a Pedro Martín Smith y Julio Ortega por su colaboración en las tareas docentes y en la preparación de este trabajo.

La parte experimental de esta tesis no habría sido posible sin los medios materiales ajenos de que he podido disponer. Por ello deseo expresar mi gratitud al Departamento de Diseño del Centro Nacional de Microelectrónica, al grupo de investigación µCEUS de la Universidad de Sevilla, al Vicerrectorado de Investigación de la Universidad de Granada,y a todo el personal del Instituto de Astrofísica de Andalucía. En particular, a Tere Osés, Lluis Terés, Miguel Angel Lopez Valverde, Antonio Ruíz Falcó, y Jose Antonio Lopez Jimenez, agradezdo la atención directa que me han prestado.

PROLOGO

Es bien conocido el continuo desarrollo de la tecnología de fabricación de circuitos integrados, el cual se manifiesta en la constante aparición de nuevos sistemas electrónicos cada vez con mejores prestaciones de velocidad, consumo de potencia, y capacidad de procesamiento.

Desde que surgieron los primeros circuitos integrados hasta nuestros días el número de transistores por chip ha seguido un crecimiento exponencial, el cual se espera continúe durante algunos años más. Para mantener este crecimiento, los fabricantes de circuitos integrados deben solucionar serios problemas ocasionados por las interconexiones internas y externas de los chips.

El trazado de interconexiones dentro del chip es una de las tareas más complejas del diseño de circuitos integrados VLSI. Además, la superficie de silicio ocupada por estas interconexiones puede incluso superar la utilizada por los componentes activos.

Por otra parte se plantea el problema de comunicar el chip con el exterior: la comunicación mediante señales binarias hace que el número de terminales de entrada/salida requeridos sea muy elevado. Esta dificultad ha sido en parte superada por los fabricantes de circuitos integrados mediante la utilización de encapsulados especiales, distintos de los "dual-in-line" tradicionales. Aun así, sigue existiendo el problema de conexionado entre chips, el cual resulta ahora más complejo, y sigue dedicándose una buena parte del chip para integrar los circuitos que activan los terminales de entrada/salida.

Una forma de paliar los problemas relacionados con las interconexiones, dentro y entre circuitos integrados, es utilizar más de dos niveles lógicos para codificar la información digital; es decir, utilizar señales multivaluadas. Si los circuitos que reciben o generan esta información tienen además la característica de funcionar internamente con señales multivaluadas, es posible lograr que la capacidad de procesamiento por unidad de superficie también se vea aumentada.

El trabajo que presentamos tiene precisamente por objeto contribuir al desarrollo de realizaciones físicas de circuitos que funcionan con más de dos niveles lógicos: circuitos multivaluados y multiumbrales.

PROPOSITO Y DESARROLLO DE LA MEMORIA

Esta memoria tiene por objeto presentar una parte del trabajo que hemos desarrollado sobre el análisis y diseño de circuitos multivaluados y multiumbrales. Los aspectos tratados en ella se refieren en particular a circuitos integrados CMOS en modo de corriente; es decir, circuitos en los que las señales utilizadas para representar los distintos niveles lógicos son corrientes eléctricas.

La memoria incluye una parte teórica en la cual se procede a la caracterización funcional y evaluación de prestaciones de los circuitos básicos CMOS en modo de corriente, la presentación de realizaciones físicas de nuevos circuitos multiumbrales y multivaluados, y el desarrollo de herramientas de ayuda a la síntesis y test de este tipo de circuitos.

La parte teórica se complementa con diseños detallados de las realizaciones integradas de los circuitos y resultados experimentales obtenidos mediante simulaciones eléctricas. Con algunos de los circuitos ideados se ha diseñado un prototipo de chip multivaluado CMOS, el cual se encuentra actualmente en

proceso de fabricación.

Esta memoria se ha estructurado en siete capítulos, seguidos de un índice con las referencias bibliográficas, y de dos apéndices. En estos últimos se incluyen listados de programas y de ficheros de descripción de circuitos.

En lugar de dedicar un capítulo a la introducción de las bases matemáticas de las lógicas multiumbral y multivaluada se ha optado por incluir, en los capítulos necesarios, sólo aquellos conceptos imprescindibles para el desarrollo de la memoria. Un tratamiento general de estos temas puede encontrarse en las referencias [114, 136, 139, 158].

El capítulo 1 consiste en una revisión crítica de las realizaciones físicas de circuitos multiumbrales y multivaluados existentes. De esta revisión se extraen consideraciones acerca de las limitaciones que han impedido una mayor profusión de este tipo de circuitos. Estas consideraciones han inspirado las directrices seguidas en el trabajo de investigación realizado, las cuales se detallan en este capítulo a modo de conclusión.

El capítulo 2 incluye una primera parte dedicada al análisis de los circuitos básicos CMOS utilizados en modo de corriente. Como consecuencia de este análisis se obtiene un conjunto de expresiones aplicables al diseño de realizaciones

integradas construidas con estos circuitos. Estos resultados han sido comprobados mediante simulación eléctrica con SPICE, haciendo uso de parámetros de modelo de un proceso CMOS real. La segunda parte del capítulo presenta realizaciones de puertas umbral y multiumbral CMOS en modo de corriente. La utilización de estos elementos umbral para la síntesis de sistemas binarios se ilustra con el diseño detallado de un multiplicador en modo de corriente, cuya realización integrada ocupa menos de la mitad de superficie que la de un multiplicador binario convencional.

Los capítulos 3 y 4 tratan de la síntesis de funciones multivaluadas mediante un nuevo tipo de matrices lógicas programables CMOS en modo de corriente. En el **capítulo 3**, tras una introducción referida a PLAs binarias, se hace una recopilación de definiciones que aparecen dispersas en distintas publicaciones, la cual es de interés para el desarrollo del resto de la memoria. La segunda parte del capítulo presenta un estudio comparativo de las estructuras de PLAs propuestas en la bibliografía para la síntesis de funciones multivaluadas. Asímismo, se plantea el problema de la simplificación de las funciones multivaluadas como medio de optimizar el tamaño de las PLAs, y se discute la elección de un algoritmo de simplificación adecuado. En el **capítulo 4** se presenta la nueva estructura de PLA multivaluada y se propone un procedimiento heurístico adecuado para la simplificación de las funciones que sintetiza. Para esta PLA se lleva a cabo una estimación de

tamaño en términos de los números equivalentes de entradas y salidas binarias, y se determina cual es la base lógica que implica una realización más favorable. El capítulo incluye un conjunto de realizaciones CMOS alternativas de los planos y los decodificadores de entrada de la PLA. Así mismo, se presentan elementos de memoria en modo de corriente mediante los cuales es posible ampliar la PLA básica para la síntesis completa de sistemas secuenciales multivaluados.

En el **capítulo 5** se trata el problema de la detección de fallos en la PLA multivaluada y se propone un procedimiento que permite comprobarla mediante una secuencia universal de patrones de test. El procedimiento exige que se añadan algunos circuitos a la PLA básica, los cuales se utilizan para la aplicación de patrones de test y la comprobación de respuestas. El capítulo concluye con el cálculo de la fracción de fallos detectados, la cual constituye una medida de la efectividad del procedimiento de test propuesto.

El capítulo 6 presenta resultados experimentales referentes a los circuitos propuestos en capítulos anteriores. En primer lugar se detallan realizaciones integradas y resultados de simulación eléctrica correspondientes a circuitos propuestos en el capítulo 4. Las realizaciones integradas se describen gráficamente, utilizando reglas lambda, en términos de las máscaras de un proceso CMOS binario. A continuación, en este capítulo, se presenta el diseño de un prototipo de PLA CMOS

tetravaluada integrada, que incluye gran parte de la circuitería necesaria para su test mediante el procedimiento propuesto en el capítulo 5.

Por último, el capítulo 7 resume las principales aportaciones del trabajo realizado y plantea posibles actividades a desarrollar en investigaciones futuras.

Para facilitar la lectura de la presente memoria conviene tener en cuenta las siguientes observaciones:

. Cada capítulo se ha dividido en secciones, las cuales a veces incluyen uno o más apartados. Cuando se hace alusión a cualquier parte del texto se indica el número de capítulo seguido de la sección y, en su caso, del número de apartado. Ejemplos: sección 2.1, apartado 4.2.3 .

. Las figuras y tablas utilizadas están numeradas por capítulos. Ejemplos: figura 2.3, tabla 2.3

. Las expresiones matemáticas tienen igualmente una numeración por capítulos. Las referencias a estas se indican mediante sus números entre paréntesis. Ejemplo: (4.15)

. Las referencias bibliográficas se indican en el texto mediante un número, o varios separados por comas, entre corchetes. Ejemplos: [10], [1, 13, 15-17]

INDICE:

Prólogo. 3

Propósito y desarrollo de la memoria 5

Indice 11

Capítulo 1: INTRODUCCION 15

1.1 Circuitos multiumbrales y multivaluados: evolución y perspectivas 16

1.1.1 Circuitos multiumbrales. 19
1.1.2 Circuitos multivaluados. 26

1.2 Objetivo y directrices del trabajo de investigación realizado. 31

Capítulo 2: CIRCUITOS BASICOS CMOS EN MODO DE CORRIENTE. . . 35

2.1 Introducción. 36

2.2 Comparación de los circuitos IIL y CMOS en modo de corriente. 39

2.3 Análisis de los circuitos básicos CMOS en modo de corriente.. 46

2.3.1 Espejos de corriente CMOS. 47
2.3.2 Detectores de umbral de corriente. 63
2.3.3 Comportamiento dinámico de los circuitos básicos CMOS en modo de corriente.. 70

2.4 Puertas umbral y multiumbral CMOS en modo de corriente.. 81

2.4.1 Diseño de un multiplicador binario CMOS en modo de corriente.. 86

2.5 Conclusión. 94

Capítulo 3: MATRICES LOGICAS PROGRAMABLES MULTIVALUADAS. . . 97

3.1 Introducción. PLAs binarias.. 98

3.1.1 Optimización del tamaño de una PLA.. 103

3.2 Síntesis de funciones lógicas multivaluadas mediante PLAs. 107

3.2.1 Definiciones.. 108
3.2.2 Estructuras de PLAs multivaluadas previamente propuestas. 113

3.3 Técnicas de minimización de funciones multivaluadas. 124

3.3.1 Métodos heurísticos. 125

3.4 Conclusión. 129

Capítulo 4: DISEÑO DE PLAs CMOS MULTIVALUADAS EN MODO DE CORRIENTE.. 131

4.1. Estructura de la PLA propuesta.. 132

4.1.1 Comparación con PLAs binarias. 137

4.2. Procedimiento de minimización. 143

4.2.1 Selección del minterm a cubrir.. 146
4.2.2 Cálculo de implicantes.. 148
4.2.3 Selección del implicante óptimo. 150
4.2.4 Algoritmo de minimización. 154

4.3. Realizaciones CMOS de la PLA propuesta.. 163

4.3.1 Celdas generadoras de literales. 164
4.3.2 Planos de la PLA.. 169
4.3.3 Celdas de memoria en modo de corriente.. . . .180

4.4. Conclusión.. 184

Capítulo 5: TEST DE PLAs MULTIVALUADAS.. 187

5.1 Introducción. 188

5.2 Técnicas de test de PLAs binarias.. 196

5.3 Procedimiento de test de PLAs multivaluadas.. . . .201

5.3.1 Modelos de fallos. 202
5.3.2 Test propuesto.. 214
5.3.3 Evaluación de la cobertura de fallos.. 222

5.4 Conclusión. 231

Capítulo 6: REALIZACIONES INTEGRADAS Y RESULTADOS DE SIMULACION. 235

6.1 Introducción. 236

6.2. Celdas generadoras de literales. 238

6.2.1 Descripción gráfica. 239
6.2.2 Resultados de simulación.. 241

6.3. Planos de la PLA.. 250

6.3.1 Descripción gráfica. 250
6.3.2 Resultados de simulación.. 254

6.4. Celdas de memoria en modo de corriente.. 268

6.5. Prototipo de PLA integrada facilmente comprobable. 270

6.6. Conclusión.. 275

Capítulo 7: CONSIDERACIONES FINALES Y CONCLUSIONES 277

BIBLIOGRAFIA 283

APENDICE A.- Listados de los programas de minimización. . .299

APENDICE B.- Listados de los ficheros de descripción SPICE .315

CAPITULO 1

INTRODUCCION

Con este capítulo se pretende dar una visión del estado actual del tema de diseño de circuitos integrados multiumbrales y multivaluados.

Para ello se ha creido conveniente hacer un resumen de las posibles realizaciones físicas en distintas tecnologías, destacando en ellas su adecuación para la construcción de este tipo de circuitos (sección 1.1).

De los aspectos tratados en la sección 1.1 pueden extraerse conclusiones acerca de las limitaciones que han impedido hasta ahora una mayor utilización práctica de los circuitos multiumbrales y multivaluados. Estas conclusiones han inspirado las directrices del trabajo realizado dadas en la sección 1.2.

1.1 Circuitos multiumbrales y multivaluados: evolución y perspectivas.

El interés por el desarrollo de circuitos integrados multiumbrales y multivaluados se remonta hasta hace dos décadas aproximadamente, coincidiendo con la aparición de los primeros circuitos integrados LSI binarios. Este interés ha motivado a cientos de investigadores de todo el mundo y ha dado lugar a un amplio espectro de publicaciones sobre el tema. Estas publicaciones se pueden categorizar en tres grupos:

1.- Lógicas multiumbral y multivaluada.

2.- Diseño de circuitos.

3.- Desarrollo de aplicaciones.

En el primer grupo se pueden incluir todos los trabajos teóricos que fundamentan matemáticamente los sistemas multiumbrales y multivaluados: formas de representación, diseño lógico, procedimientos de síntesis, etc [4, 36, 55, 61, 73, 76, 90, 118, 120, 145, 169, 202]

En el segundo grupo deben incluirse aquellos trabajos referentes a obtención de realizaciones físicas de los elementos de circuito necesarios para la construcción de sistemas multiumbrales y multivaluados integrados: diseño de familias lógicas de circuitos, caracterización y evaluación de prestaciones, desarrollo de dispositivos electrónicos con funcionamiento multinivel, viabilidad de los procesos de fabricación,

etc [6, 7, 11, 62, 80, 102, 110, 112, 138, 144, 180, 205].

En el grupo tercero se pueden clasificar los trabajos cuyo objetivo es presentar aplicaciones concretas de los circuitos multiumbrales y multivaluados [1, 19, 24, 26, 92, 93, 94, 100, 103, 131, 132, 135, 168]. En este grupo tienen cabida aplicaciones especiales tales como: codificación multivaluada y procesamiento de señales analógicas [39, 46, 48, 63, 167, 195], utilización de elementos umbrales en la construcción de redes neuronales [87, 111, 121, 178], aplicación de técnicas multivaluadas a la síntesis y test de circuitos binarios [40, 41, 70-72, 160, 163], etc.

Desde la creación, en el año 1971, del Comité Técnico en Lógica multivaluada del IEEE, viene celebrándose anualmente un Simposium internacional cuyas actas constituyen una buena fuente de información acerca de los trabajos de investigación que se llevan a cabo en lógicas multiumbral y multivaluda.

A pesar de los avances conseguidos en el diseño de circuitos y sistemas cuyo funcionamiento es total o parcialmente multivaluado, en el momento actual sería ilógico pensar que una lógica no binaria pueda reemplazar a esta en el ámbito de los circuitos integrados digitales o en el de sitemas de procesamiento de información digital en general. No obstante, para el desarrollo de determinadas aplicaciones puede ser ventajoso, con la tecnología actual, utilizar realizaciones

basadas en tales lógicas.

De hecho, algunos fabricantes de circuitos integrados han incorporado módulos multivaluados dentro de chips binarios, consiguiendo con ello optimizar el tamaño del circuito integrado resultante o reducir el número de sus conexiones externas. Ejemplos de estos circuitos son las memorias ROM tetravaluadas MOS incluidas en los procesadores 8087 y 432-03 de Intel [176]. Diseños alternativos de ROMs tetravaluadas han sido utilizadas por General Instruments en varios de sus chips [157]. Motorola presentó una versión de ROM tetravaluada CMOS en 1985 [31]. Este mismo fabricante tiene en producción la familia de circuitos MC-145026/7/8 [130], los cuales emplean una codificación ternaria de direcciónes (utilizan 9 términales para codificar 3^9=19638 direcciones, en lugar de las 2^9=512 que se tendrían con una codificación binaria). La empresa Signetics ha utilizado puertas umbral en modo de corriente en algunos de sus circuitos integrados IIL [38]. IBM y Mitsubishi [174] han aportado diseños de memorias serie tetravaluadas en tecnología CCD, e Hitachi ha presentado un estudio pre-comercial de un prototipo de RAM dinámica 16-valuada [5].

Presentar una revisión exhaustiva de las distintas realizaciones físicas de circuitos multivaluados y multiumbrales propuestas en la bibliografía extendería demasiado nuestra exposición. El lector interesado puede encontrar gran parte de esta información en trabajos de recopilación existentes [12,

43, 82, 83, 117, 172-174, 189]. En esta sección introductoria tan solo nos vamos a referir a las realizaciones más significativas en las distintas tecnologías, haciendo especial hincapié en las ventajas y limitaciones que cada una de ellas presenta. Estas realizaciones se tratan por separado para circuitos multiumbrales y multivaluados en los dos apartados siguientes.

1.1.1 Circuitos multiumbrales.

Los circuitos multiumbrales son una generalización de los circuitos umbrales. El elemento básico de construcción de un circuito umbral es la denominada *puerta umbral* [114]. La salida de una puerta umbral depende de la relación existente entre una combinación lineal de sus entradas y un cierto valor de umbral. Una puerta umbral de n entradas $(x_1, x_2, .., x_n)$ puede caracterizarse mediante un vector de n+1 componentes $\{p_1, p_2, .., p_n; U\}$, en donde p_i representa el peso asignado a la entrada x_i, y U es el valor del umbral. La salida z de la puerta es, al igual que las entradas, una variable binaria. El valor de z viene definido por:

$$z = 0 \text{ si } E<U$$

$$z = 1 \text{ si } E\geq U \qquad (1.1)$$

La magnitud E, la cual recibe el nombre de excitación, es la suma aritmética de las entradas x_i ponderadas por los pesos p_i:

$$E = \sum_{1}^{n} p_i \cdot x_i \qquad (1.2)$$

Una función booleana realizable mediante una puerta umbral recibe el nombre de función umbral o función linealmente separable. Para realizar una función booleana cualquiera interconectando puertas umbrales, dicha función debe expresarse en términos de funciones linealmente separables.

Sin perder generalidad, los pesos p_i utilizados pueden siempre considerarse valores enteros, en cuyo caso a los umbrales se les asigna valores en la forma $U_j = k_j + 0.5$, con k entero.

Los circuitos multiumbrales son una generalización de los circuitos umbrales, y sus elementos básicos de construcción son las puertas multiumbrales.

Una puerta multiumbral de n entradas y m umbrales está caracterizada por un vector de n+m componentes $\{p_1, p_2, .., p_n; U_1, .., U_m\}$, en donde $U_i > U_{i+1}$. El valor que toma la salida de la puerta multiumbral es [151]:

$$z = 1 \text{ si } E \geq U_1 \text{ ó } U_{2j} > E \geq U_{2j+1}$$

$$z = 0 \text{ en el resto de los casos.} \qquad (1.3)$$

De la definición de puerta multiumbral resulta claro que cualquier función booleana puede sintetizarse mediante una

única puerta multiumbral con un número suficientemente grande de umbrales. No obstante, a fin de posibilitar realizaciones físicas fiables, se han desarrollado métodos de síntesis de funciones mediante puertas multiumbrales de pocos umbrales. Como ejemplo, con una, o a lo sumo dos, puertas multiumbrales de tres umbrales es posible sintetizar cualquier función binaria de cuatro variables [115].

La adecuación de una determinada tecnología de circuitos integrados para la construcción de circuitos multiumbrales depende de la facilidad de esta para realizar las operaciones de suma ponderada y comparación requeridas.

Dada la dificultad para efectuar una suma ponderada de tensiones eléctricas, tradicionalmente esta operación se ha realizado en modo de corriente o de carga; es decir, sumando corrientes eléctricas o paquetes de carga multivaluados. Las primeras realizaciones significativas de circuitos basados en elementos umbrales en modo de corriente fueron ideadas para la tecnología ECL [79, 81, 190].

La figura 1.1 muestra un esquema simplificado de puerta umbral ECL. En ella la corriente que representa la suma ponderada E se obtiene por simple conexión directa de las líneas que suministran las corrientes $p_i . I_0 . x_i$ (entradas ponderadas). Una réplica de E es convertida a tensión y comparada con la tensión de umbral U. En los circuitos ECL las

comparaciones se efectúan realmente en modo de tensión, utilizándose elementos resistivos para efectuar la conversión previa de corriente a tensión, y circuitos auxiliares para generar las distintas tensiones de umbral necesarias.

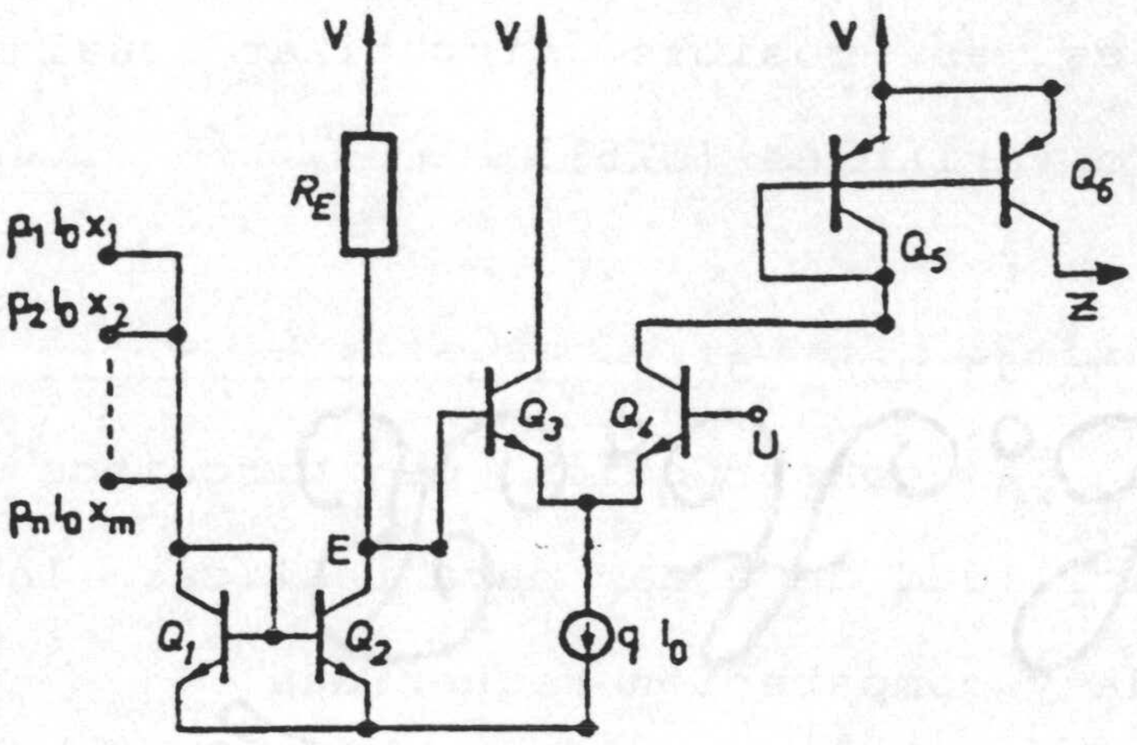


FIGURA 1.1: Puerta umbral ECL.

Con la aparición de la tecnología IIL surge la posibilidad de construir circuitos lógicos LSI y VLSI en modo de corriente puro; es decir, efectuando sumas ponderadas de corriente y comparaciones de corriente. Desde 1974 hasta la fecha han aparecido varias publicaciones relativas al diseño de circuitos multiumbrales en esta tecnología [25, 27, 88, 144, 152]. Los elementos básicos de construcción de estos circuitos se ilustran en la figura 1.2. La suma ponderada de corrientes es fácilmente realizable mediante conexión directa de salidas de espejos de corriente (ver figura 1.3).

En los últimos años, el interés por los circuitos en modo

de corriente se ha desplazado hacia la tecnología CMOS [95, 96, 140-142, 204].

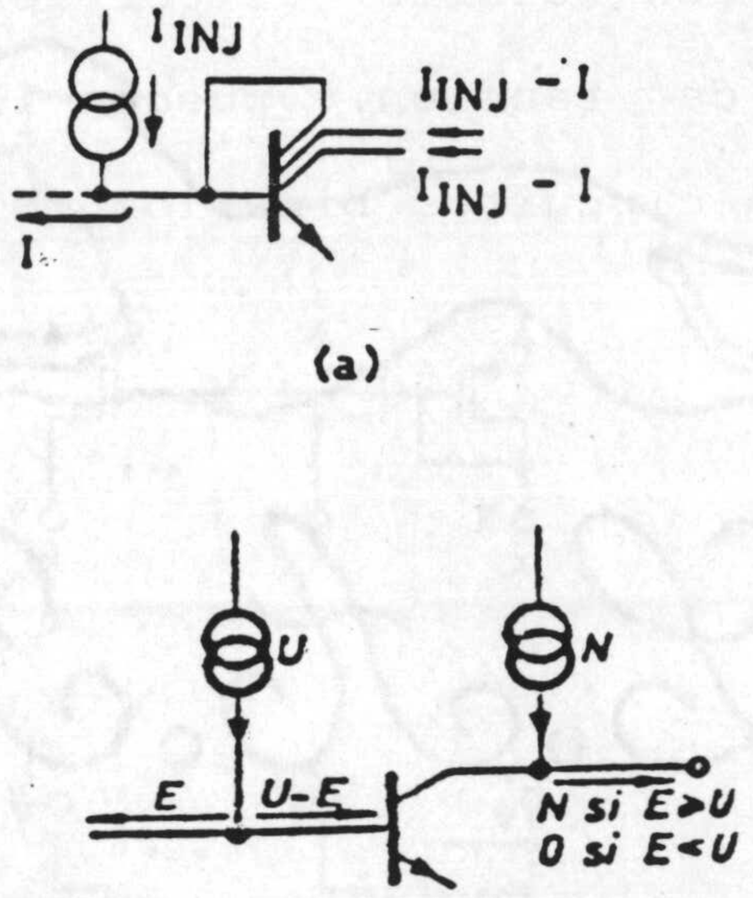


FIGURA 1.2: Circuitos básicos IIL en modo de corriente. (a) Espejo de corriente. (b) Comparador de umbral de corriente.

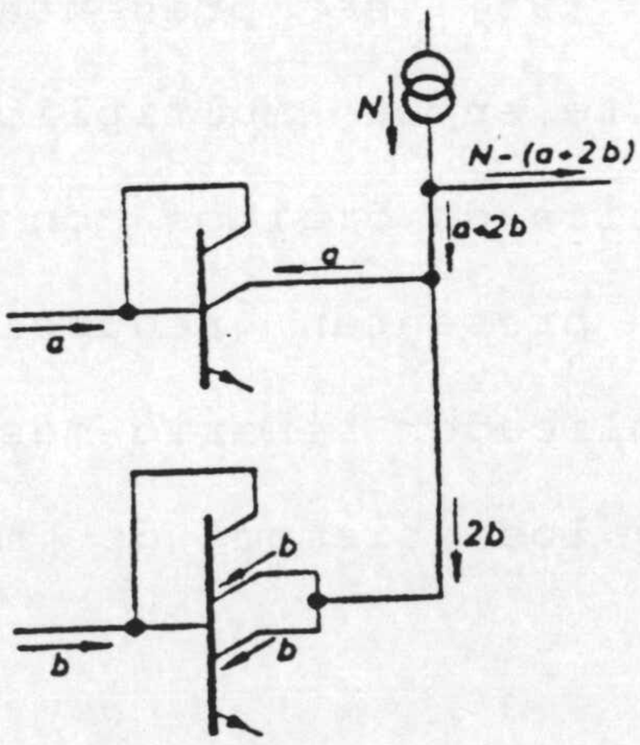


FIGURA 1.3: Suma ponderada de corrientes en circuitos IIL.

Los elementos básicos de construcción de estos circuitos

son los espejos de corriente PMOS y NMOS cuyas configuraciones más sencillas se indican en la figura 1.4. Estos elementos básicos, combinados con fuentes de corriente y circuitos CMOS binarios en modo de tensión, pueden llevar a realizaciones competitivas con los circuitos binarios CMOS convencionales.

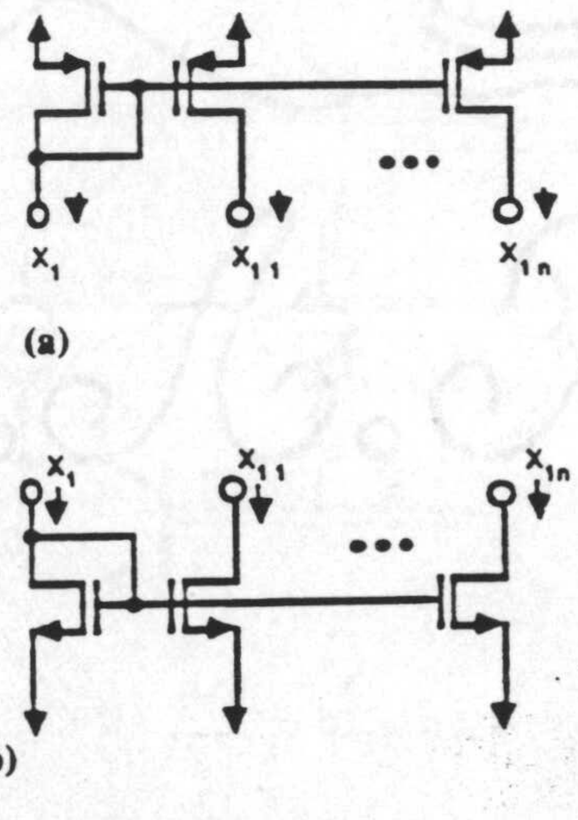


FIGURA 1.4: Configuraciones básicas de espejos de corriente CMOS [174]:(a) Espejo PMOS; (b) Espejo NMOS.

Como ejemplo, en [95] se presenta un chip CMOS en modo de corriente consistente en un multiplicador de 32x32 bits, el cual emplea aritmética de dígitos con signo. La superficie y el consumo del chip presentan reducciones del orden del 50% respecto del multiplicador binario más rápido construido con la misma tecnología, y los tiempos de multiplicación son similares.

Al igual que ocurre en los circuitos en modo de corriente, las operacioens básicas necesarias para realizar elementos multiumbrales se obtienen fácilmente si las señales utilizadas

son cargas eléctricas. En la tecnología CCD, el dispositivo elemental (un condensador MOS) puede almacenar varios niveles de carga dependiendo de la altura del pozo de potencial que se origine bajo su terminal de puerta. Este concepto aparece ilustrado en la figura 1.5.a, junto con un símbolo utilizado para representar tal dispositivo [12, 101, 203]. Controlando adecuadamente las transferencias de carga entre distintos dispositivos pueden efectuarse las operaciones de suma lineal y comparación con umbrales (ver figuras 1.5.b y 1.5.c).

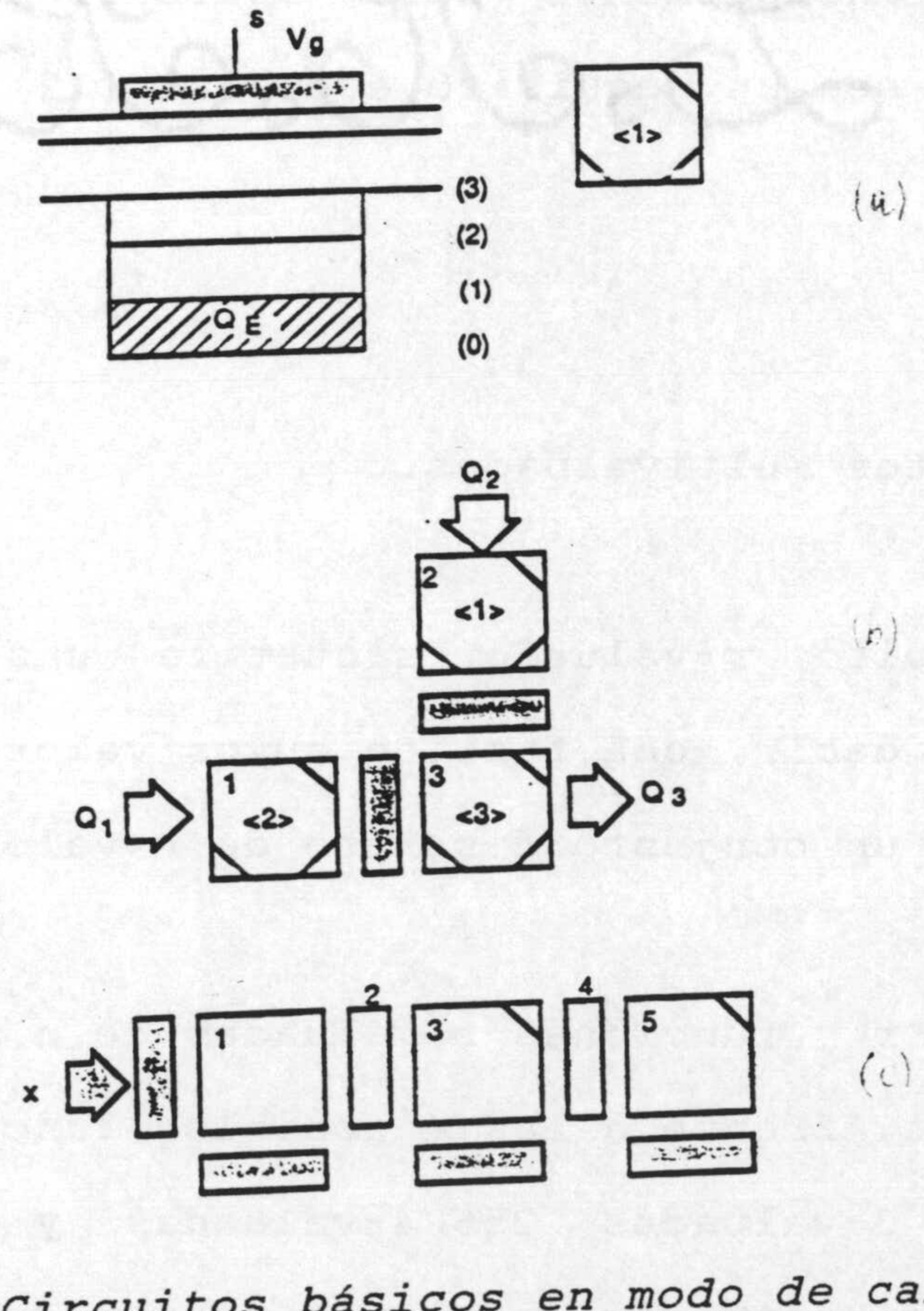


FIGURA 1.5: Circuitos básicos en modo de carga. (a) Dispositivo CCD con capacidad de almacenamiento de 4 niveles de carga, y símbolo utilizado para representarlo. (b) Suma linal de cargas. (c) Comparador de tres umbrales de carga [12].

En [146] se presentan posibles realizaciones de puertas umbral y multiumbral CCD. En ellas los pesos y los umbrales son programables mediante tensiones de polarización externas.

Las mayores limitaciones de los circuitos en modo de carga son las propias de los circuitos CCD binarios: son relativamente lentos y requieren utilizar múltiples señales de reloj para gobernar su funcionamiento. No obstante, en la actualidad se mantiene el interés por la realización de circuitos CCD no binarios, principalmente para aplicaciones en las que la velocidad no sea un requisito esencial [12].

1.1.2 Circuitos multivaluados.

Un circuito r-valuado sintetiza una función lógica r-valuada; es decir, una función cuyos valores y cuyas variables pertenecen a un conjunto discreto de r valores distintos.

Existen r^{r^n} funciones r-valuadas de n variables. De una sola variable, frente a las 4 posibles funciones binarias, hay 27 funciones 3-valuadas, 256 4-valuadas, y 16777216 8-valuadas. Esta sencilla comparación sirve para justificar la existencia de una gran diversidad de alternativas de síntesis de funciones multivaluadas, las cuales difieren en los conjuntos de operadores básicos de una y dos variables considerados. En consecuen-

cia, son posibles muy diversas realizaciones físicas basadas en los circuitos o puertas lógicas que sintetizan dichos operadores. La complejidad de estas puertas, y por tando de los sistemas multivaluados construidos con ellas, está muy influenciada por la tecnología de fabricación y la naturaleza física de las señales utilizadas.

Los circuitos multivaluados básicos en modo de tensión, cuyas realizaciones físicas más representativas corresponden a las tecnologías NMOS y CMOS, tienen la estructura general de la figura 1.6. Mediante comparadores de nivel se efectua una decodificación a binario de las señales de entrada. Las salidas del decodificador se procesan mediante circuitos binarios y posteriormente se lleva a cabo una codificación multivaluada. Evaluando la complejidad de los circuitos en términos de la superficie necesaria para su integración es claro que, a la vista del esquema de la figura 1.6, sólo es posible obtener circuitos integrados multivaluados en modo de tensión ventajosos respecto de otros binarios cuando los distintos módulos indicados en la figura admitan realizaciones muy simples.

FIGURA 1.6: Esquema general de un circuito multivaluado en modo de tensión.

Aunque para algunos circuitos concretos la separación en módulos indicada no sea tan explícita, de alguna manera siempre se efectua en ellos una cuantización de niveles lógicos de las entradas mediante comparadores con salida binaria.

En las tecnologías MOS, estos comparadores suelen construirse con inversores binarios cuya característica de transferencia DC presenta su transición desplazada positiva o negativamente. Formas alternativas de conseguir los desplazamientos relativos mencionados consisten en utilizar varias tensiones de alimentación [78, 150, 192, 201], o añadir transistores MOS conectados a modo de diodos [116, 150], o utilizar transistores de diversas geometrías [43], o incluso con inversores cuyos transistores tengan múltiples tensiones umbrales [77, 78, 91, 185, 205].

Las realizaciones de circuitos multivaluados en modo de tensión propuestas en la bibliografía se refieren principalmente a lógica trivaluada [65, 66, 74, 75, 78, 107, 119, 133, 134, 136, 138, 149, 201]. Son menos las realizaciones para lógicas de orden superior debido a la dificultad para mantener los suficientes márgenes de ruido entre niveles sin emplear tensiones de alimentación superiores a las utilizadas para circuitos binarios. Mención aparte merecen los circuitos de memoria ROM y RAM multivaluadas en modo de tensión [5, 156]. En estos circuitos, aunque se utilizan mecanismos de lectura y (para la RAM) escritura más complejos que en memorias MOS

binarias, la complejidad de los circuitos de lectura/escritura necesarios se ve compensada por el gran ahorro de superficie que se consigue al hacer funcionar cada celda de memoria con más de dos niveles. La RAM dinámica presentada en [5] es 16-valuada, lo que implica una reducción de 4 a 1 en el número de celdas necesarias.

Las puertas lógicas multivaluadas en modo de corriente o de carga pueden tener estructuras alternativas dependiendo del tipo de operadores básicos que sintetizan. Si se trata de operaciones tales como el máximo y mínimo multivaluados (generalizaciones de los operadores AND y OR binarios), la estructura de dichas puertas lógicas es análoga a la indicada en la figura 1.6 [122, 179]. Un esquema similar se utiliza en las puertas T multivaluadas [25], las cuales son una generalización de los multiplexiores binarios. En cambio, existe la posibilidad de construir puertas lógicas multivaluadas cuya estructura es más próxima a la de una puerta multiumbral, este es el caso de las denominadas puertas multiumbrales multivaluadas [32].

Una puerta multiumbral r-valuada de n entradas viene definida por un vector de pesos $\{p_1, p_2, .., p_n\}$ y una característica de transferencia H(e) que relaciona los valores de la salida r-valuada con la excitación "e". H(e) define implícitamente el conjunto de umbrales de la puerta. La función realizada por dicha puerta es $F(x_1, .., x_n)=H(e)$. Como ejemplo, la

figura 1.7 ilustra la síntesis de una función 4-valuada de dos variables mediante una puerta multiumbral 4-valuada con pesos de entrada unidad y tres umbrales.

La mayoría de las realizaciones de circuitos multivaluados en tecnología ECL, así como gran parte del resto de las realizaciones en modo de corriente y de carga, utilizan puertas lógicas multiumbrales multivaluadas [2, 16, 20, 21, 28, 32, 35, 49, 89, 194, 196], las cuales son especialmente adecuadas cuando se trata de sintetizar de funciones aritméticas.

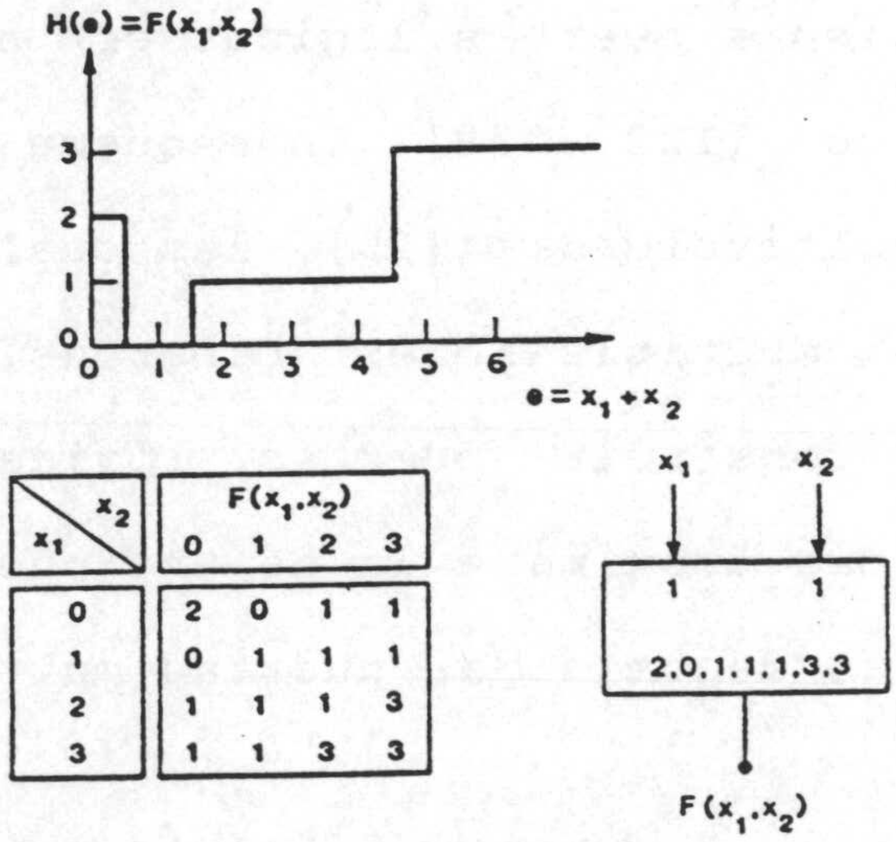


FIGURA 1.7: Ejemplo de síntesis de una función tetravaluada mediante una puerta multiumbral multivaluada [173].

Según lo expuesto en los párrafos anteriores, los circuitos en modo de corriente pueden considerarse más versátiles que los que funcionan en modo de tensión, sobre todo para lógicas

de orden cuatro o superiores. Así mismo, como indicamos en el apartado 1.1.1, el modo de corriente permite obtener realizaciones de puertas umbrales y multiumbrales relativamente sencillas. Como inconveniente, estos circuitos tienen por construcción un "fan-out" unidad, lo que impide utilizar el modo de corriente para conexiones entre chips, a menos que se trate de conexiones punto a punto.

1.2 Objetivo y directrices del trabajo de investigación realizado.

El trabajo desarrollado tiene como objetivo fundamental el diseño de circuitos integrados multiumbrales y multivaluados CMOS en modo de corriente.

Aunque muchas de las realizaciones de circuitos multiumbrales y multivaluados propuestos en la bibliografía sean viables, su impacto en la industria de los circuitos integrados digitales ha sido hasta ahora escaso: los circuitos de memoria multivaluados son básicamente los que han presentado atractivo para los fabricantes de circuitos integrados [5, 31, 157, 174, 176].

Esta falta de interés hacia los sistemas lógicos no binarios se debe principalmente a la inercia de los diseñadores

y de los propios fabricantes, la cual está justificada si tenemos en cuenta la cuantía de las inversiones realizadas y la disponibilidad de potentes herramientas de ayuda al diseño y comprobación de circuitos binarios. El diseño lógico de circuitos multiumbrales y multivaluados es en general más complejo que el de circuitos binarios, y no se dispone de herramientas CAD eficientes para su diseño.

Particularizando a la tecnología CMOS, y circuitos en modo de tensión, existen además inconvenientes de carácter práctico que limitan su competitividad con los circuitos binarios. Estos inconvenientes, en parte expresados en la sección 1.1, pueden resumirse en los siguientes puntos:

. En estos circuitos cada transistor funciona en realidad con señales binarias, motivo por el cual el número de dispositivos e interconexiones necesarios no es tan ventajoso como pudiera esperarse.

. En muchos casos pueden requerir procesos de fabricación especiales más complejos que los utilizados para circuitos CMOS binarios.

. La necesidad de mantener unos márgenes de ruido aceptables limita el número de posibles niveles lógicos y/o requiere utilizar tensiones de alimentación elevadas en el circuito.

Teniendo presentes los puntos anteriores, el trabajo desarrollado ha seguido unas líneas de actuación que podemos resumir en:

. Estudio de circuitos que funcionen en modo de corriente y que sean integrables mediante un proceso CMOS binario convencional. Esta labor es una continuación de la que iniciamos para circuitos en tecnología IIL [144], y está motivada por la ventaja del modo de corriente para hacer funcionar realmente cada dispositivo con varios niveles sin requerimientos especiales de alimentación.

. Caracterización del funcionamiento estático y dinámico de los circuitos básicos utilizados, mediante la cual sea posible evaluar las prestaciones de velocidad-consumo y márgenes de ruido de los diseños realizados. Además, dicha caracterización sienta las bases para la elaboración de modelos simplificados de los circuitos básicos mediante los cuales se puedan desarrollar herramientas eficientes de ayuda al diseño de circuitos más complejos.

. Diseño de nuevos circuitos integrados multivaluados y multiumbrales en los cuales se aprecie la ventaja de reducción de tamaño que teóricamente implica utilizar más de dos niveles lógicos. Con objeto de poner de manifiesto esta ventaja, para los circuitos presentados se detallan

realizaciones integradas concretas, las cuales se comparan con realizaciones de circuitos binarios construidas con las mismas reglas de diseño.

CAPITULO 2

CIRCUITOS BASICOS CMOS EN MODO DE CORRIENTE

En este capítulo presentamos un estudio del comportamiento de los circuitos básicos CMOS en modo de corriente como elementos para la construcción de circuitos multivaluados y multiumbrales.

Las operaciones básicas realizadas en circuitos lógicos en modo de corriente requieren el uso de dos elementos de circuito: los espejos de corriente y los detectores de umbral de corriente. Estos elementos se definen en la sección 2.1.

En la sección 2.2 se comentan las ventajas e inconvenientes que presentan las realizaciones CMOS de circuitos en modo de corriente respecto de sus precursoras en tecnología IIL.

La viabilidad de los circuitos CMOS en modo de corriente depende de la adecuación de sus realizaciones integradas para funcionar con varios niveles de corriente. En la sección 2.3, utilizando modelos simplificados de dispositivos, se analizan las características estáticas y dinámicas de los elementos básicos de construcción de dichos circuitos. Los resultados teóricos que se obtienen están contrastados con simulaciones eléctricas de dichos circuitos, utilizando parámetros de modelos de dispositivos correspondientes a un proceso CMOS real.

En la sección 2.4 presentamos realizaciones de puertas umbral y multiumbral basadas en los elementos básicos descritos en la sección anterior. La utilidad de dichas puertas se ilustra con el diseño de un multiplicador paralelo con entradas y salidas binarias, que funciona internamente con varios niveles de corriente. La síntesis de sistemas multivaluados CMOS en modo de corriente será motivo de estudio en capítulos posteriores.

La sección 2.5 es una conclusión.

2.1 Introducción.

Como indicamos en la sección 1.1, los circuitos multiumbrales y multivaluados que funcionan en modo de corriente se

basan en la utilización de dos configuraciones básicas: el comparador o detector de umbral de corriente y el espejo de corriente. En esta memoria utilizaremos los términos de detector de umbral de corriente y espejo de corriente con los siguientes significados:

Un *detector de umbral* de corriente es un circuito con una salida binaria y una entrada multivaluada en modo de corriente, presentando su salida un estado distinto dependiendo de si la corriente de entrada supera o no un cierto valor umbral (ver figura 2.1.a). No se exige condición alguna sobre la magnitud física que representa la salida.

Utilizaremos la denominación de *espejo de corriente* para referirnos a un circuito con una entrada y varias salidas y tal que la corriente en cada una de sus salidas es proporcional a la corriente de su entrada (ver figura 2.1.b).

Los espejos de corriente se emplean, cuando la constante de proporcionalidad entre las corrientes de entrada y salida es la unidad, para obtener réplicas de las corrientes de salida de aquellos circuitos que requieran fan-out mayor que la unidad. También se utilizan, produciendo múltiplos o fracciones de su corriente de entrada, para ponderar señales binarias o multivaluadas.

La conexión directa de las líneas de salida de espejos de corriente es la forma de realizar la suma ponderada de varia-

bles, siendo esta la operación básica utilizada en el diseño lógico de los circuitos en modo de corriente.

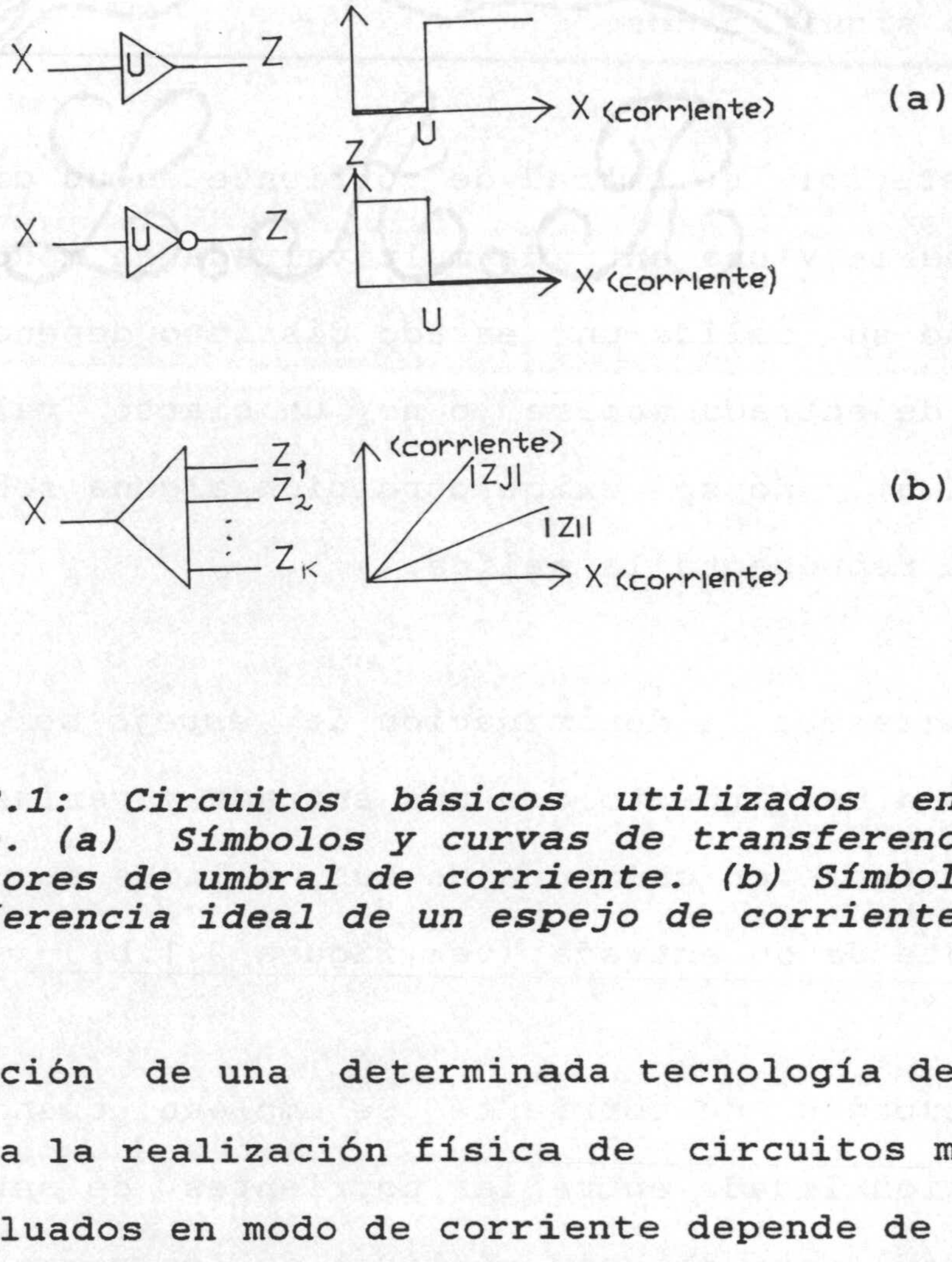


FIGURA 2.1: Circuitos básicos utilizados en modo de corriente. (a) Símbolos y curvas de transferencia ideales de detectores de umbral de corriente. (b) Símbolo y curva de transferencia ideal de un espejo de corriente.

La adecuación de una determinada tecnología de circuitos integrados para la realización física de circuitos multiumbrales y multivaluados en modo de corriente depende de la facilidad para construir los circuitos básicos mencionados anteriormente, así como de las características funcionales que estos presenten. Interesa que las realizaciones integradas de estos circuitos sean del menor tamaño posible y que su comportamiento sea lo más próximo al reflejado en la figura 2.1. Evidentemente, deben considerarse además las prestaciones en cuanto a velocidad y consumo de potencia que se obtienen con dichas realizaciones.

Cuando nos referimos a detectores de umbral de corriente, implícitamente estamos considerando la existencia de fuentes de referencia con las cuales se efectua la comparación de corriente. También se requieren fuentes de referencia que suministren los valores de corriente constantes correspondientes a los distintos niveles lógicos utilizados en los circuitos. La tecnología utilizada debe permitir la construcción de dichas fuentes de referencia mediante circuitos de poca complejidad.

En los últimos años (desde el 1985), la mayoría de las publicaciones referentes a circuitos en modo de corriente corresponden a realizaciones físicas en tecnología MOS [22, 95, 96, 98, 99, 140-142, 204]. Los diseños de circuitos en esta tecnología son, en la mayoría de los casos, adaptaciones de otros ideados para la tecnología IIL. Dada esta similitud es conveniente, antes de proceder a un analisis más detallado de los circuitos básicos CMOS en modo de corriente, hacer un breve resumen de las limitaciones que presentan las realizaciones de circuitos IIL en modo de corriente, y de las ventajas que aporta utilizar la tecnología CMOS.

2.2 Comparación de los circuitos IIL y CMOS en modo de corriente.

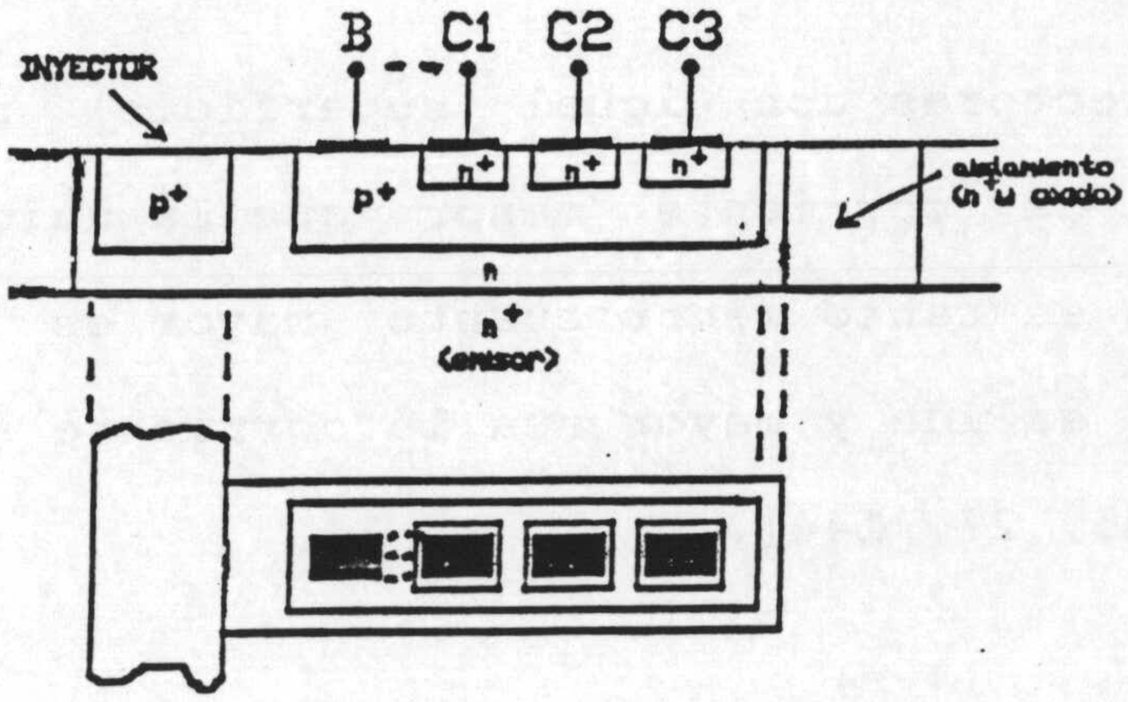
En la tecnología IIL, el componente utilizado para la construcción de espejos de corriente y detectores de umbral de

corriente es el dispositivo multicolector mostrado en la figura 2.2. Las fuentes de corriente de referencia utilizadas en los circuitos IIL las constituyen los transistores pnp laterales incorporados en los propios dispositivos multicolector. Esto contribuye a que los circuitos integrados que se obtienen presenten una elevada densidad de integración funcional. El valor de la corriente de referencia depende de la superficie efectiva de colector dada al transistor pnp y de la tensión aplicada a su terminal de emisor (INYECTOR).

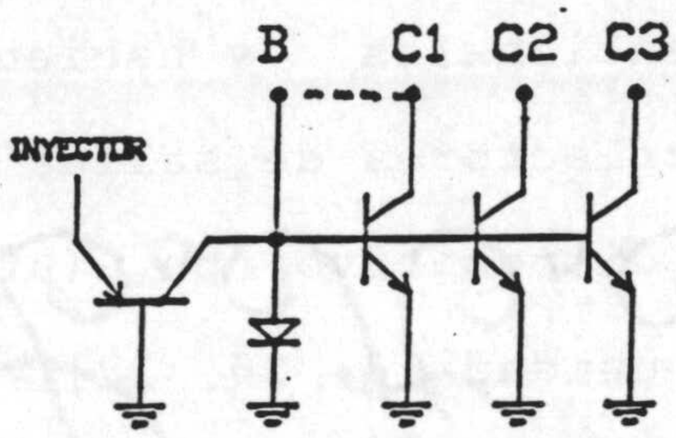
Para que el dispositivo multicolector IIL funcione como espejo de corriente se realimenta uno de los colectores de salida con el terminal de base B (conexión indicada con líneas discontínuas en la figura 2.2). Las relaciones de tamaño entre los distintos colectores de salida y el realimentado con la base determinan las relaciones existentes entre las corrientes de salida y de entrada del espejo.

La principal ventaja de la tecnología IIL para la construcción de circuitos multiumbrales y multivaluados en modo de corriente es la elevada densidad de integración funcional que se consigue gracias al reducido espacio que ocupan los circuitos básicos utilizados en el diseño.

Como inconveniente podemos resaltar la dificultad para obtener detectores de umbral y espejos de corriente cuyo comportamiento se aproxime al ideal indicado en la figura 2.1. Esta dificultad tiene su origen en dos aspectos:



(a)



(b)

FIGURA 2.2. Dispositivo multicolector IIL. (a) Estructura física. (b) Circuito equivalente.

(1) Los transistores laterales pnp que realizan las fuentes de corriente funcionan en saturación, motivo por el cual la corriente que inyectan en la base de los dispositivos npn presenta variaciones muy acusadas.

(2) La ganancia de corriente de los transistores verticales npn es muy pequeña debido a su funcionamiento en modo inverso (los terminales de colector de un dispositivo npn IIL hacen las veces de terminales de emisor en un dispositivo npn convencional).

Debido a estos inconvenientes, un espejo de corriente diseñado, por ejemplo, para obtener n réplicas de corriente

(todos sus colectores con igual superficie) presenta siempre una ganancia de corriente menor que la unidad. El valor de dicha ganancia es tanto menor cuanto mayor es el número n de colectores de salida y mayor sea la corriente suministrada por el inyector [25, 27, 144].

Las limitaciones (1) y (2) anteriores pueden soslayarse mediante el uso de configuraciones tipo Darlington, en lugar de los transistores pnp convencionales, y haciendo modificaciones de la geometría de los colectores de salida para aproximar la ganancia de corriente del dispositivo npn (utilizado este como espejo de corriente) a la unidad [8, 25, 27].

La no idealidad de los espejos de corriente IIL puede también corregirse calculando la superficie efectiva de los colectores de los dispositivos npn según un modelo propuesto por nosotros [144]. En dicho cálculo se tiene en cuenta tanto el error producido por la baja ganancia de corriente de los dispositivos npn como la no idealidad de los inyectores de corriente, evitandose la utilización de inyectores más complejos.

Utilizando un proceso IIL estándar no es posible construir espejos de corriente pnp, lo cual resta versatilidad en el diseño de circuitos en modo de corriente. Para su fabricación es necesario utilizar un proceso más complejo que permita integrar los transistores pnp como componentes aislados de los npn.

Los circuitos IIL presentan además un inconveniente que complica en general su diseño: el fan-out de toda puerta IIL coincide con el número de colectores del dispositivo multicolector npn (fan-out unidad para cada salida).

En tecnología CMOS, utilizando un proceso ordinario de fabricación de circuitos binarios pueden realizarse espejos de corriente complementarios (NMOS y PMOS). En la figura 2.3 se muestran posibles realizaciones físicas de dichos espejos. En este caso, a diferencia de lo que ocurre en la tecnología IIL, un aumento en el número de salidas del espejo no implica un deterioro de su funcionamiento, dado que el terminal utilizado para controlar la corriente de salida (terminal de puerta de los transistores MOS) no consume corriente de la entrada. Al aumentar el número de salidas, tan solo se verán afectadas las características dinámicas del espejo debido al aumento de la capacidad asociada al nudo de entrada.

Para un mismo número de salidas, un espejo de corriente PMOS o NMOS ocupa más superficie que uno IIL construido con las mismas dimensiones mínimas de diseño (ver figuras 2.2 y 2.3). Esto es debido a la estructura lateral de los dispositivos MOS frente a la vertical de los dispositivos npn IIL. Los valores de las relaciones de tamaño W/L (anchura de canal/longitud de canal) de los transistores de salida respecto del valor de la relación de tamaño del transistor de entrada determinan las relaciones existentes entre las corrientes de salida y de entrada en un espejo CMOS.

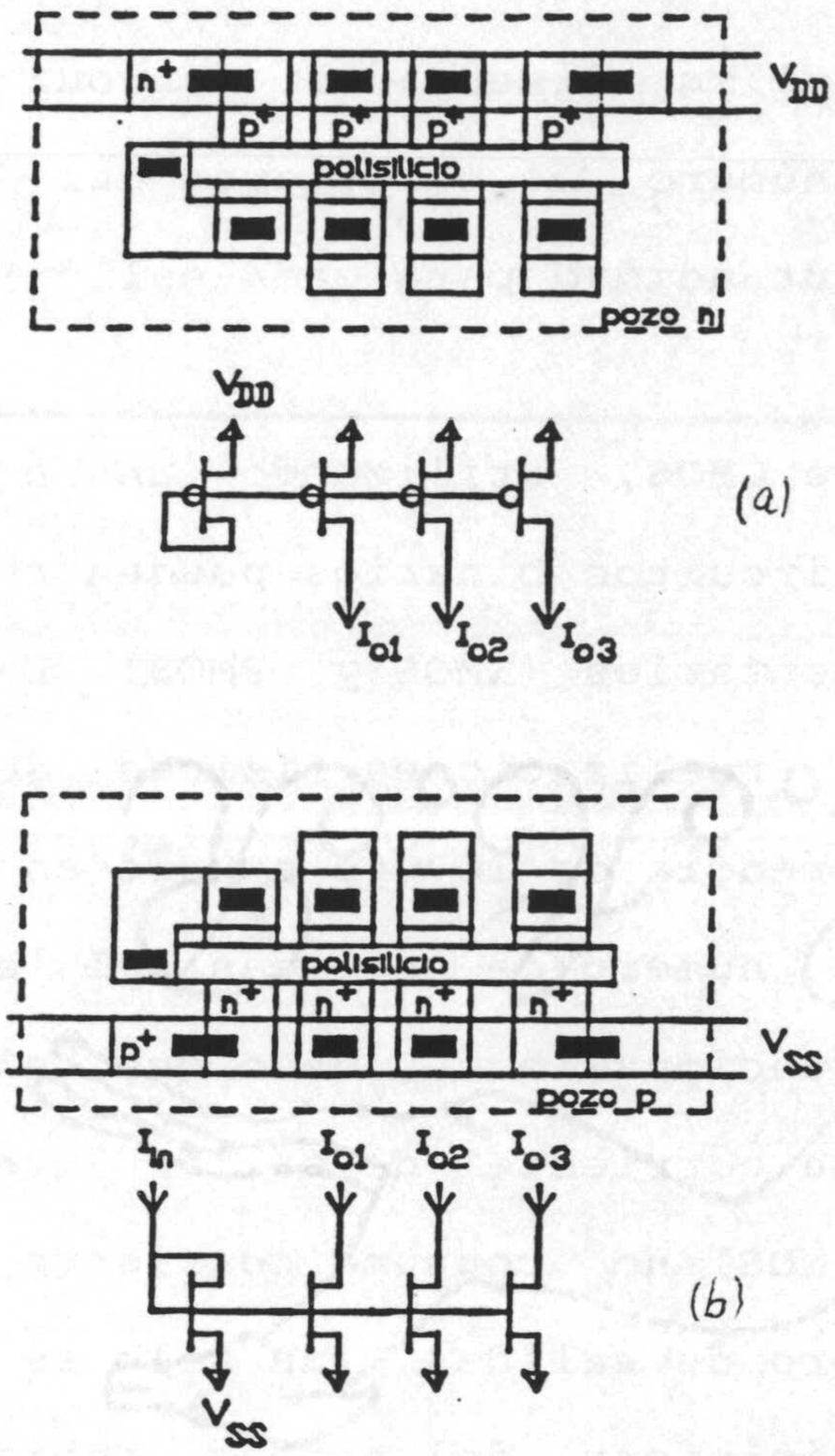


FIGURA 2.3: Estructura física y circuitos equivalentes de espejos de corriente CMOS. (a) Espejo PMOS. (b) Espejo NMOS.

Las fuentes de corriente de referencia necesarias para establecer umbrales y niveles lógicos pueden realizarse mediante transistores MOS con tensiones de puerta-fuente constantes. Utilizando una misma tensión de referencia, los valores de las corrientes suministradas vienen determinados por la magnitud de las relaciones de tamaño W/L de dichos transistores.

Mediante dos espejos de corriente complementarios, uno de

ellos con corriente de entrada constante, se construye un detector de umbral de corriente. En la figura 2.4 se muestran las dos posibles realizaciones de detectores de umbral. Si se dispone de las tensiones de referencia V_{rP} y V_{rN}, los espejos de corriente que fijan la corriente umbral pueden simplificarse a un único transistor (MPR2 y MNR2 en las figuras 2.4.a y 2.4.b respectivamente). Estos detectores de umbral producen una

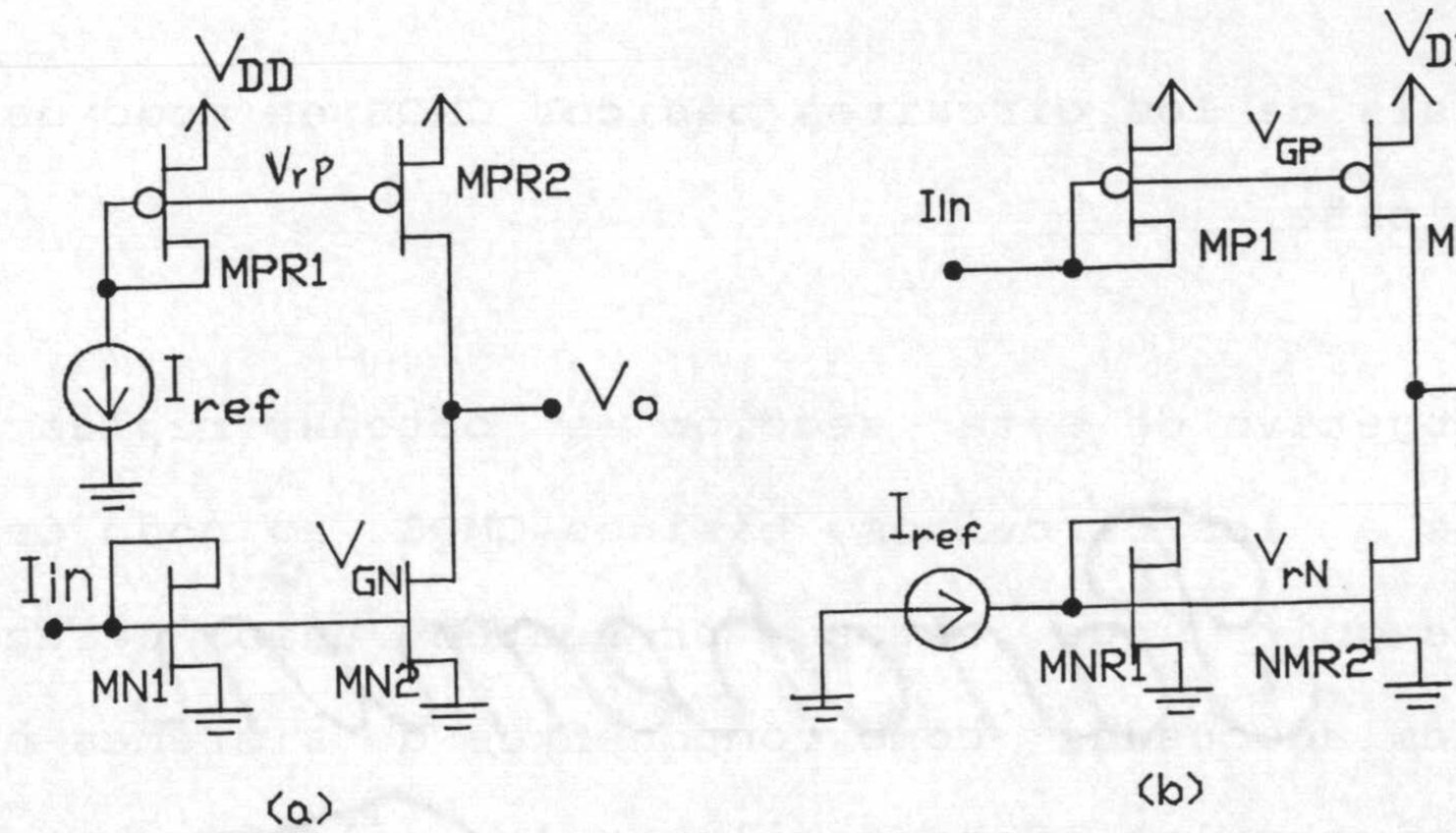


FIGURA 2.4: Detectores de umbral de corriente CMOS. (a) con fuente de referencia PMOS. (b) Con fuente de referencia NMOS.

salida en modo de tensión y no tienen la limitación de fan-out de los detectores de umbral IIL. De hecho, aunque a lo largo de esta memoria utilicemos la denominación de circuitos CMOS en modo de corriente, lo correcto sería hablar de circuitos en modo mixto. Las salidas en modo de tensión de los detectores de umbral posibilitan la combinación de funciones binarias con realizaciones CMOS convencionales junto con las multivaluadas

en modo de corriente.

En la siguiente sección realizaremos un análisis cuantitativo del funcionamiento de los circuitos básicos CMOS en modo de corriente mencionados anteriormente. Dicho análisis permitirá obtener reglas de diseño aplicables en la síntesis de circuitos multivaluados y multiumbrales.

2.3 Análisis de los circuitos básicos CMOS en modo de corriente.

El objetivo de esta sección es obtener reglas de diseño aplicables a los circuitos básicos CMOS en modo de corriente para conseguir que estos presenten unas características funcionales adecuadas como componentes de sistemas multivaluados y multiumbrales. Para facilitar los cálculos consideraremos modelos simplificados de dispositivos, procediendo posteriormente a la comprobación de resultados mediante simulación eléctrica con SPICE. En las simulaciones consideramos modelos de dispositivo LEVEL=3 [188] con los siguientes valores de parámetros de modelo:

```
.MODEL  NMOS   NMOS(LEVEL=3  VTO=0.947  GAMMA=0.2925  RSH=33.38
CJ=0.15E-3 TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
ETA=0.221 KAPPA=0.212)
```

```
.MODEL   PMOS   PMOS(LEVEL=3   VTO=-0.773   GAMMA=0.7164   RSH=49.40
CJ=0.25E-3 TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
LD=0.8E-6   UO=217   VMAX=2.09E5   DELTA=0.0 THETA=0.087 ETA=0.234
KAPPA=5.995)
```

Estos parámetros corresponden al proceso CMOS de 3µm, pozo N, y un solo nivel de metalización, de la Universidad Católica de Louvain (Bélgica).

2.3.1 Espejos de corriente CMOS.

Cuando se utilizan como componentes lógicos en circuitos multivaluados y multiumbrales, las salidas de los espejos de corriente deben conectar con entradas o con salidas de espejos de corriente complementarios. La segunda situación se presenta cuando el espejo de corriente forma parte de un detector de umbral de corriente. En este apartado nos referiremos a espejos de corriente cuyas salidas conectan con entradas de otros espejos, dejando para el siguiente apartado el estudio de los detectores de umbral.

Consideremos los espejos de corriente de una salida mostrados en la figura 2.5. Los transistores MPL y MNL representan los transistores de entrada de espejos complementarios a los indicados en la figura.

En primera aproximación, las corrientes de drenador de los transistores NMOS y PMOS podemos expresarlas mediante las ecuaciones [68, 193]:

Transistor NMOS:

Región lineal ($V_{DS} < V_{GS} - V_{TN}$; $V_{GS} > V_{TN}$)

$$I_{DS} = \beta_N . (2.(V_{GS} - V_{TN}).V_{DS} - V_{DS}^2) \qquad (2.1)$$

Región de saturación ($V_{DS} \geq V_{GS} - V_{TN}$; $V_{GS} > V_{TN}$)

$$I_{DS} = \beta_N . (V_{GS} - V_{TN})^2 \qquad (2.2)$$

Transistor PMOS:

Región lineal ($V_{SD} < V_{SG} - V_{TP}$; $V_{SG} > V_{TP}$)

$$I_{SD} = \beta_P . (2.(V_{SG} - V_{TP}).V_{SD} - V_{SD}^2) \qquad (2.3)$$

Región de saturación ($V_{SD} \geq V_{SG} - V_{TP}$; $V_{SG} > V_{TP}$)

$$I_{SD} = \beta_P . (V_{SG} - V_{TP})^2 \qquad (2.4)$$

Los subíndices S, G, y D, de las expresiones anteriores hacen referencia a los terminales de fuente, de puerta, y de drenador, respectivamente. V_{TN} y V_{TP} son los valores absolutos de las tensiones umbral de los transistores NMOS y PMOS respectivamente.

β_N y β_P son los factores de transconductancia de los transistores NMOS y PMOS respectivamente:

$$\beta_N = \tfrac{1}{2} . \varepsilon_{ox} . \mu_N . T_{ox}^{-1} . (W/L)_N$$

$$\beta_P = \tfrac{1}{2} . \varepsilon_{ox} . \mu_P . T_{ox}^{-1} . (W/L)_P \qquad (2.5)$$

en donde ε_{ox} y T_{ox} son la constante dieléctrica y el espesor del óxido de puerta de los transistores. μ_N y μ_P son las movilidades superficiales de los portadores n y p respectivamente. W/L es la relación de tamaño del transistor, definida como el cociente entre la anchura y la longitud efectivas de su canal.

Refiriéndonos al esquema de la figura 2.5.a, dado que el transistor MN1 del espejo NMOS funciona en saturación, la corriente de entrada al espejo podemos expresarla, utilizando la relación (2.2), como:

$$I_{in} = \beta_{N1} . (V_G - V_{TN})^2 \qquad (2.6)$$

en donde β_{N1} es el factor de transconductancia del transistor MN1.

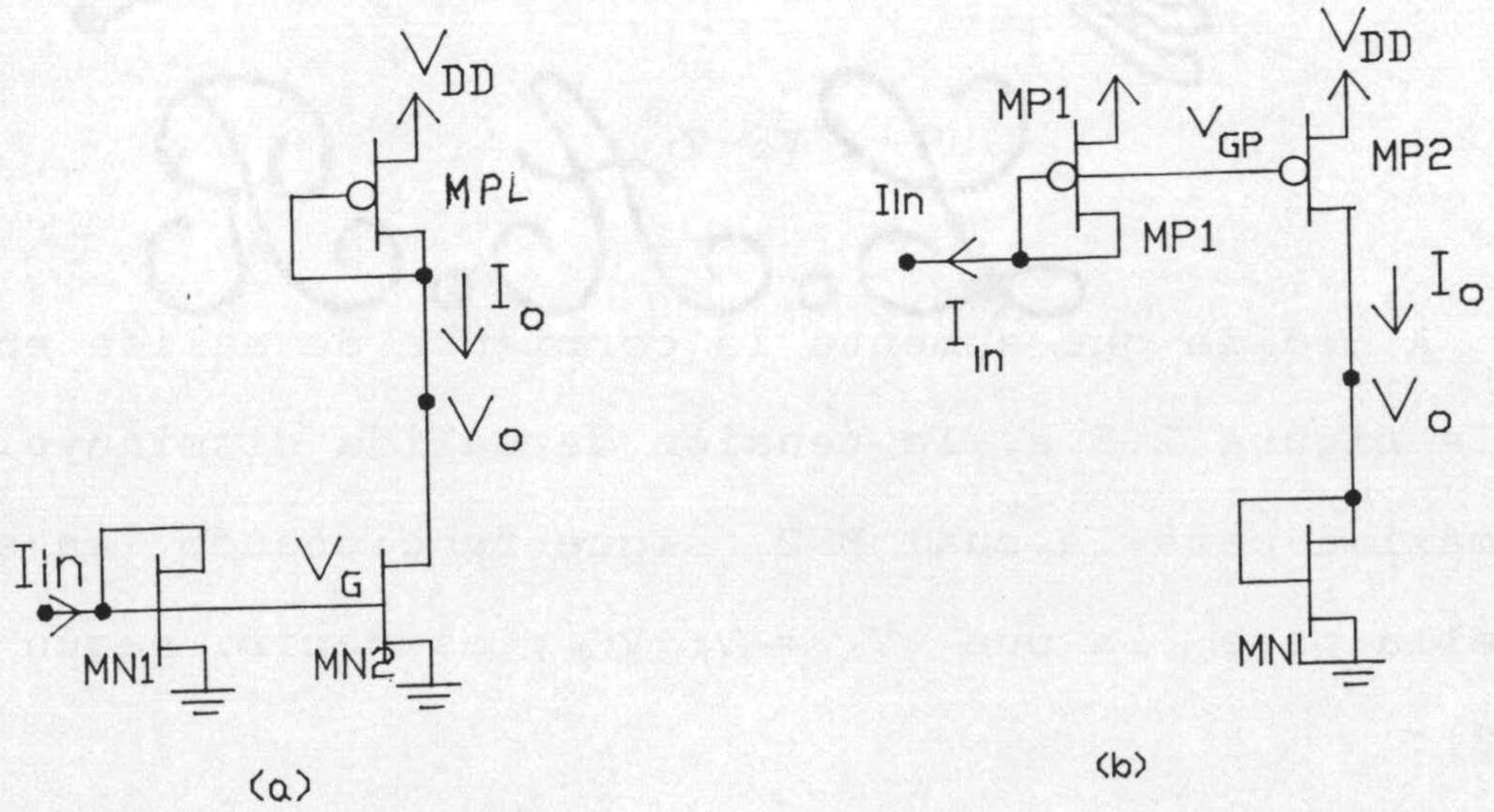


FIGURA 2.5: Espejos de corriente monosalida. (a) Espejo NMOS. (b) Espejo PMOS.

El transistor MPL también funciona siempre en saturación, por lo que la corriente de salida del espejo puede expresarse mediante la relación (2.4) como:

$$I_o = \beta_P . (V_{DD} - V_o - V_{TP})^2 \qquad (2.7)$$

El funcionamiento correcto del espejo de corriente NMOS exige que el transistor MN2 funcione también en saturación, estando su corriente de drenador (I_o) dada por:

$$I_o = \beta_{N2} . (V_G - V_{TN})^2 \qquad (2.8)$$

en donde β_{N2} es el factor de transconductancia de MN2 y V_{TN} su tensión umbral, la cual se supone igual a la del transistor MN1.

La condición de saturación del transistor MN2 exige que la tensión de salida cumpla la condición:

$$V_o \geq V_G - V_{TN}$$

A medida que aumenta la corriente de salida en el circuito de la figura 2.5.a, la tensión de salida disminuye. La corriente máxima para la cual MN2 sigue funcionando en saturación es aquella para la que $V_o = V_G - V_{TN}$; es decir, según la expresión (2.8):

$$I_{o,max} = \beta_{N2} . V_o^2 \qquad (2.9)$$

Relacionando las expresiones (2.7) y (2.9) se obtiene:

$$(I_{o,max}/\beta_P)^{1/2} + (I_{o,max}/\beta_{N2})^{1/2} = V_{DD} - V_{TP} \qquad (2.10)$$

Un planteamiento similar al anteriormente realizado, pero referido al espejo PMOS de la figura 2.5.b, da como resultado:

$$(I_{o,max}/\beta_N)^{1/2} + (I_{o,max}/\beta_{P2})^{1/2} = V_{DD} - V_{TN} \qquad (2.11)$$

β_N y β_{P2} son los factores de transconductancia de los transistores MNL y MP2 del esquema de la figura 2.5.b. V_{TN} es la tensión umbral del transistor MNL.

Para un determinado proceso CMOS, las ecuaciones (2.10) y (2.11) establecen condiciones sobre las relaciones de tamaño mínimas que deben tener los espejos de corriente para funcionar con los niveles de corriente máximos utilizados.

Si existe un camino resistivo entre la salida del espejo de corriente y la entrada del siguiente, que produzca una caida de tensión V*, las corrientes máximas utilizables deben verificar:

$$(I_{o,max}/\beta_P)^{1/2} + (I_{o,max}/\beta_{N2})^{1/2} = V_{DD} - V_{TP} - V^*$$

ó

$$(I_{o,max}/\beta_N)^{1/2} + (I_{o,max}/\beta_{P2})^{1/2} = V_{DD} - V_{TN} - V^* \qquad (2.12)$$

según se trate de un espejo NMOS o PMOS.

Cuando se intercalan transistores o puertas de paso entre la salida del espejo de corriente y la entrada del siguiente, V* coincide con la tensión drenador-fuente en conducción de dichos elementos de paso.

La validez de las expresiones anteriores está evidentemente limitada por el grado de aproximación de las ecuaciones de modelo utilizadas. Considerando que los transistores del mismo tipo (NMOS o PMOS) tienen idénticas tensiones umbral, que estas se mantienen constantes, y que las corrientes de drenador son independientes de las tensiones drenador-fuente, los espejos de corriente considerados tendrían una ganacia de corriente constante siempre que la corriente de salida sea inferior a la dada por las ecuaciones (2.10) y (2.11). Además, el valor de dicha ganancia vendría exclusivamente determinado por el cociente entre las relaciones de tamaño de los transistores de entrada y de salida del espejo.

En la práctica, debido a fluctuaciones propias del proceso de fabricación, así como a efectos de segundo orden no considerados, el comportamiento de los espejos de corriente puede alejarse del indicado anteriormente, produciendose cierto deterioro de los niveles de corriente utilizados.

Entre los referidos efectos de segundo orden podemos considerar: la degradación de la movilidad superficial, las

variaciones de las tensiones umbral, y la modulación de la longitud del canal. Ya que estos efectos son tanto más significativos cuanto menor es el tamaño de los transistores, para que los espejos de corriente funcionen con varios niveles de corriente garantizando ciertos márgenes de ruido entre ellos, es necesario que los transistores que los configuran tengan dimensiones superiores a unas mínimas.

No es posible establecer cuales son en general dichas dimensiones mínimas dado que la magnitud de los efectos relacionados depende en gran medida del proceso concreto de fabricación. Sin embargo, podemos evaluar qué efectos son más significativos sobre el funcionamiento de los espejos de corriente analizando el tipo de dependencia de las corrientes de salida respecto de las distintas magnitudes afectadas por la reducción de tamaño.

Degradación de la movilidad superficial:

La degradación debida al campo eléctrico transversal en el canal puede evaluarse expresando la movilidad efectiva mediante la siguiente relación empírica [184]:

$$\mu_{ef} = \mu.(1+\Theta.(V_{GS}-V_T))^{-1}$$

Ya que en un espejo de corriente la tensión de puerta-fuente es común para los transistores de entrada y de salida,

la variación de la movilidad indicada anteriormente en la práctica no tiene efecto sobre la ganancia de corriente. Cuando hay diferencia entre las tensiones umbrales de los transistores del espejo, se tendrán movilidades distintas. No obstante, su efecto contrarresta el producido por los términos cuadráticos en $(V_{GS}-V_T)$ sobre las corrientes de drenador, contribuyendo a que la ganancia del espejo no se aleje de su valor ideal.

Por otro lado, debido a la saturación de la velocidad de los portadores en el canal, existe una reducción de las corrientes drenador-fuente en saturación [184]. Como resultado, la dependencia de dicha corriente con $(V_{GS}-V_T)$ tiende a ser lineal en lugar de cuadrática, de forma más acusada cuanto más corto sea el canal.

En principio, este cambio de dependencia no provoca un deterioro significativo de la ganancia de corriente de los espejos. Su efecto es más acusado sobre las corrientes máximas de operación de los espejos, las cuales serán en la práctica sensiblemente inferiores a las dadas por (2.10) y (2.11).

En la figura 2.6 se muestran las corrientes de salida de un espejo de corriente NMOS simulado con SPICE. I(R) coincide con la corriente de entrada del espejo, e I(MNi) es la corriente de drenador del i-ésimo transistor de salida. Las relaciones de tamaño asignadas a los terminales de puerta de estos transistores son las siguientes:

MN1: 4.5µm/6µm; MN2: 6/6; MN3: 9/6; MN4: 12/6

El transistor de entrada del espejo es idéntico al MN2, y los terminales de puerta de los transistores de carga PMOS tienen asignada la relación de tamaño:

MPL: WG/LG=9µm/6µm

Como se observa en la figura 2.6, con la longitud de canal considerada para los transistores del espejo (4.5µm de longitud efectiva una vez descontadas las difusiones laterales), las corrientes de salida mantienen la proporcionalidad impuesta con la corriente de entrada hasta que se alcanzan ciertos valores máximos.

Las corrientes máximas según la expresión (2.10) debieran ser:

ID(MN1)≈54µA, ID(MN2)≈65µA, ID(MN3)≈80µA, e ID(MN4)≈90µA

Para la obtención de estos valores se ha evaluado la anchura efectiva de los transistores NMOS y PMOS a partir de los resultados de simulación de dispositivos de canal largo, igualando las corrientes de saturación que se obtienen mediante simulación con las dadas por las expresiones (2.2) y (2.4).

Si la relación de tamaño de las puertas de los transistores PMOS es de 12/6 en lugar de 9/6 se obtienen los resultados de la figura 2.7. Los valores de corriente máxima que se obtienen con (2.10) son en este caso:

ID(MN1)≈66µA, ID(MN2)≈81µA, ID(MN3)≈102µA, e ID(MN3)≈115µA

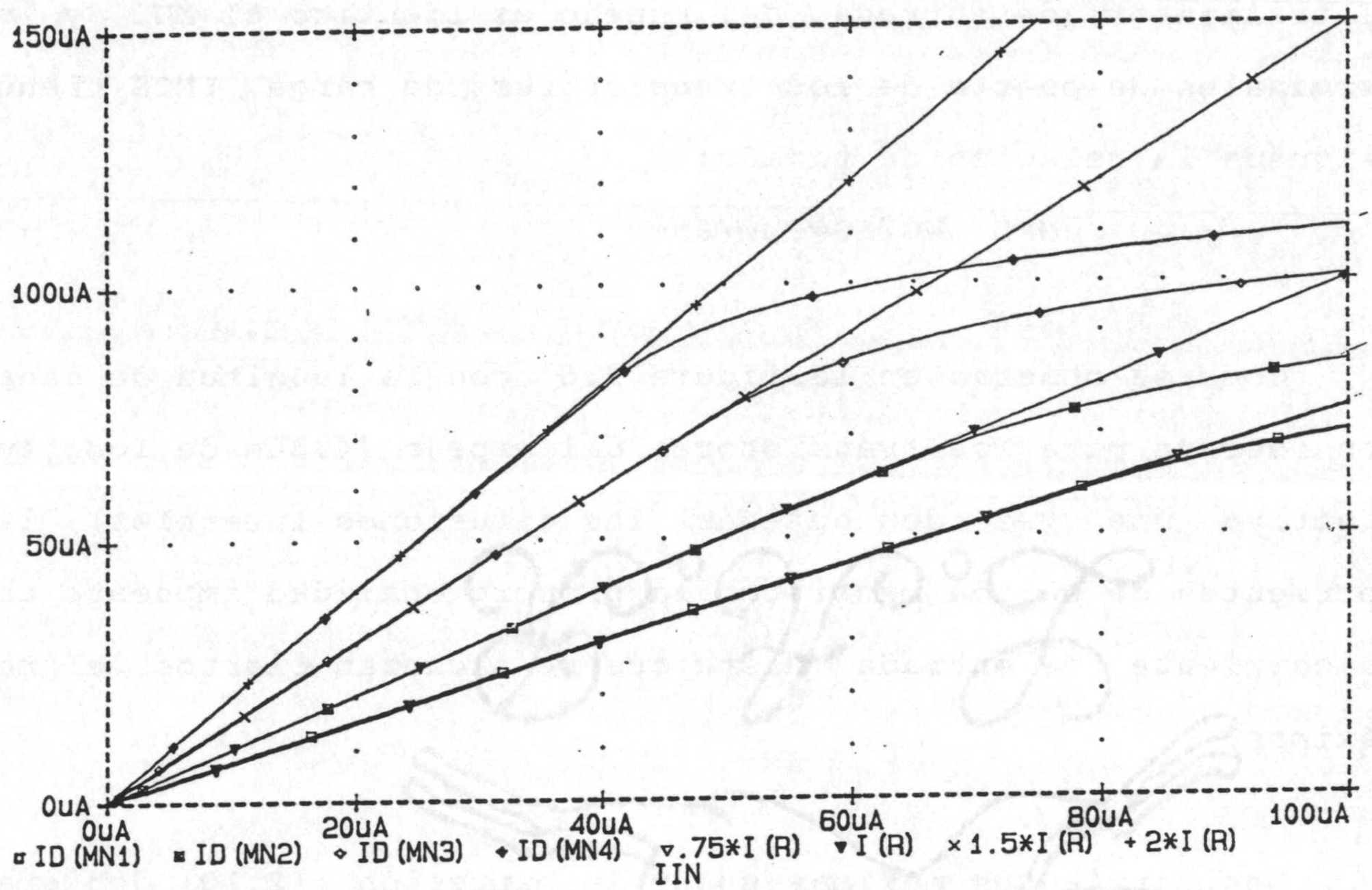


FIGURA 2.6: Corrientes de salida de un espejo NMOS con longitudes de canal L=4.5µm. Las cargas PMOS tienen tamaños de puerta WG/LG=9µm/6µm.

los cuales, al igual que los referidos a la figura 2.6, constituyen una buena estimación de los valores de corriente máximos observados en la simulación. En ambos casos no obstante, los valores obtenidos por simulación son ligeramente inferiores a los previstos.

Diferencias entre las tensiones umbral:

Refiriéndonos al espejo de corriente NMOS de la figura

2.5.a, supongamos que las tensiones umbral de los transistores MN1 y MN2 de dicho espejo son V_{TN} y $V_{TN} \pm \delta V_{TN}$ respectivamente. Sustituyendo estos valores en las expresiones (2.6) y (2.8), y despreciando el término $(\delta V_{TN})^2$, se obtiene:

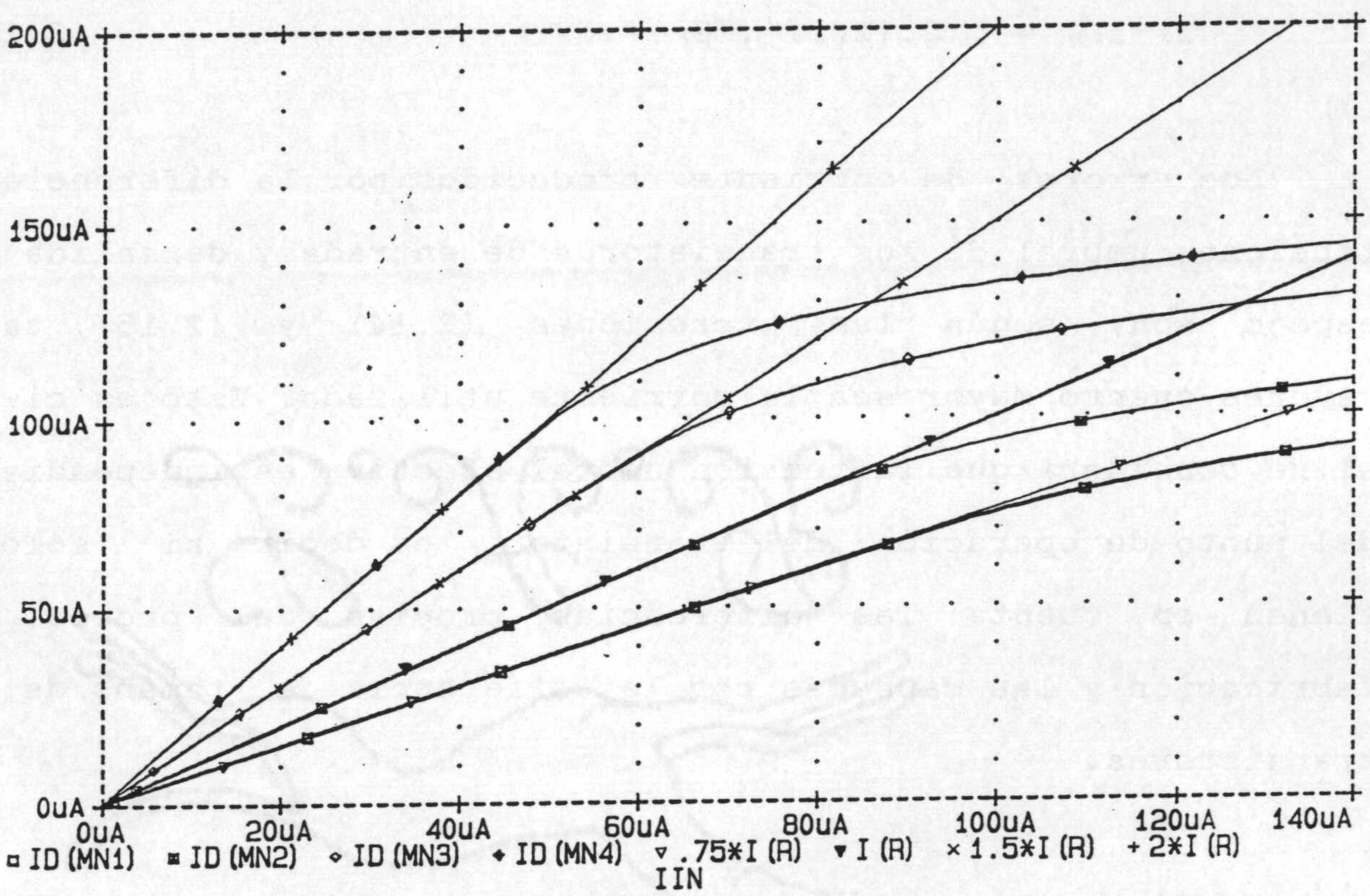


FIGURA 2.7: Corrientes de salida de un espejo NMOS con longitudes de canal L=4.5µm. Las cargas PMOS tienen tamaños de puerta WG/LG=12µm/6µm.

$$I_o = I_{in} . \beta_{N2} / \beta_{N1} \mp 2 . \beta_{N2} . (I_{in})^{½} . (\beta_{N1})^{-½} . \delta V_{TN} \qquad (2.13)$$

En el caso particular de que los transistores del espejo tengan igual transconductancia $\beta_{N1} = \beta_{N2} = \beta_N$, de la expresión (2.13) resulta:

$$I_o - I_{in} = \mp 2 . (I_{in})^{½} . (\beta_N)^{½} . \delta V_{TN} \qquad (2.14)$$

Siguiendo un desarrollo similar al anterior pero referido al espejo PMOS de la figura 2.5.b, cuando los dos transistores que lo forman tienen la misma transconductacia β_P, se obtiene:

$$I_o - I_{in} = \mp 2.(I_{in})^{1/2}.(\beta_P)^{1/2}.\delta V_{TP} \qquad (2.15)$$

Los errores de corriente producidos por la diferencia de tensiones umbral de los transistores de entrada y de salida del espejo son, según las expresiones (2.14) y (2.15), tanto mayores cuanto mayor sea la corriente utilizada. Esto es cierto si se considera que la tensión umbral efectiva es independiente del punto de operación del transistor, es decir si sólo se tienen en cuenta las diferencias propias del proceso de fabricación y las causadas por la diferencia de tamaño de los transistores.

En dispositivos de canal corto debe considerarse además la reducción de la tensión umbral efectiva provocada por el drenador.

Refiriéndonos al espejo NMOS de la figura 2.5.a, y suponiendo que los transistores de entrada y de salida del espejo son idénticos, la diferencia de tensiones umbral puede expresarse en términos de la diferencia de tensiones de drenador de ambos como [184]:

$$\delta V_T = -\sigma.(V_o - V_G) \qquad (2.16)$$

en donde el factor de proporcionalidad σ es tanto mayor cuanto menor sea la longitud de canal de los dispositivos.

El error de corriente que resulta de sustituir (2.16) en (2.14) es positivo o negativo dependiendo de la relación entre tensiones de drenador y de puerta del transistor de salida. Es pues positivo cuando la corriente de salida es pequeña y se hace negativo para niveles altos de corriente.

Modulación de la longitud del canal:

Otra causa de error en el funcionamiento de los espejos de corriente es la variación de la longitud efectiva del canal de los transistores en saturación. Este efecto podemos tenerlo en cuenta, en primera aproximación, expresando las corrientes de drenador de los transistores NMOS y PMOS en saturación como [68, 184]:

$$I_{DS} = \beta_N (V_{GS} - V_{TN})^2 . (1 + \lambda_N . V_{DS}) \qquad (2.17)$$

$$I_{SD} = \beta_P (V_{SG} - V_{TP})^2 . (1 + \lambda_P . V_{SD}) \qquad (2.18)$$

en donde λ_N y λ_P son los factores de modulación de la longitud efectiva del canal de los transistores NMOS y PMOS respectivamente.

Al utilizar la expresión (2.17) para las corrientes de drenador de los transistores MN1 y MN2 de la figura 2.5.a,

suponiendo que ambos tienen la misma transconductancia β_N, se obtiene:

$$I_o - I_{in} = I_{in} \cdot ((1+\lambda_N \cdot V_o) \cdot (1+\lambda_N \cdot V_G)^{-1} - 1) \qquad (2.19)$$

Para el espejo PMOS de la figura 2.5.b, utilizando la expresión (2.18) para evaluar las corrientes de drenador, se obtiene:

$$I_o - I_{in} = I_{in} \cdot ((1+\lambda_P \cdot (V_{DD} - V_o) \cdot (1+\lambda_P \cdot (V_{DD} - V_G))^{-1} - 1) \qquad (2.20)$$

En la figura 2.8 se muestran las curvas de transferencia DC obtenidas con SPICE para transistores NMOS del proceso CMOS que estamos considerando. Descontadas las difusiones laterales, las longitudes efectivas de canal correspondientes a las figuras 2.8.a, 2.8.b, y 2.8.c, son de 4.5µm, 3µm, y 1.5µm, respectivamente. El efecto de la modulación de la longitud del canal comienza a ser significativo ($\lambda_N \approx 0.01$) cuando la longitud de puerta es de 4.5µm (3µm de longitud de canal). Cuando la longitud de puerta es de 3µm, el factor de modulación que se obtiene (figura 2.8.c) es de $\lambda_N \approx 0.06$.

En las figuras 2.9 y 2.10 se representan las corrientes de entrada y de salida de un espejo de corriente NMOS, en el cual las longitudes de puerta de los transistores que lo configuran son 4.5µm y 3µm respectivamente. Las anchuras de canal, así como el tamaño de los transistores de carga PMOS, coinciden con las utilizadas para la obtención de la figura 2.6.

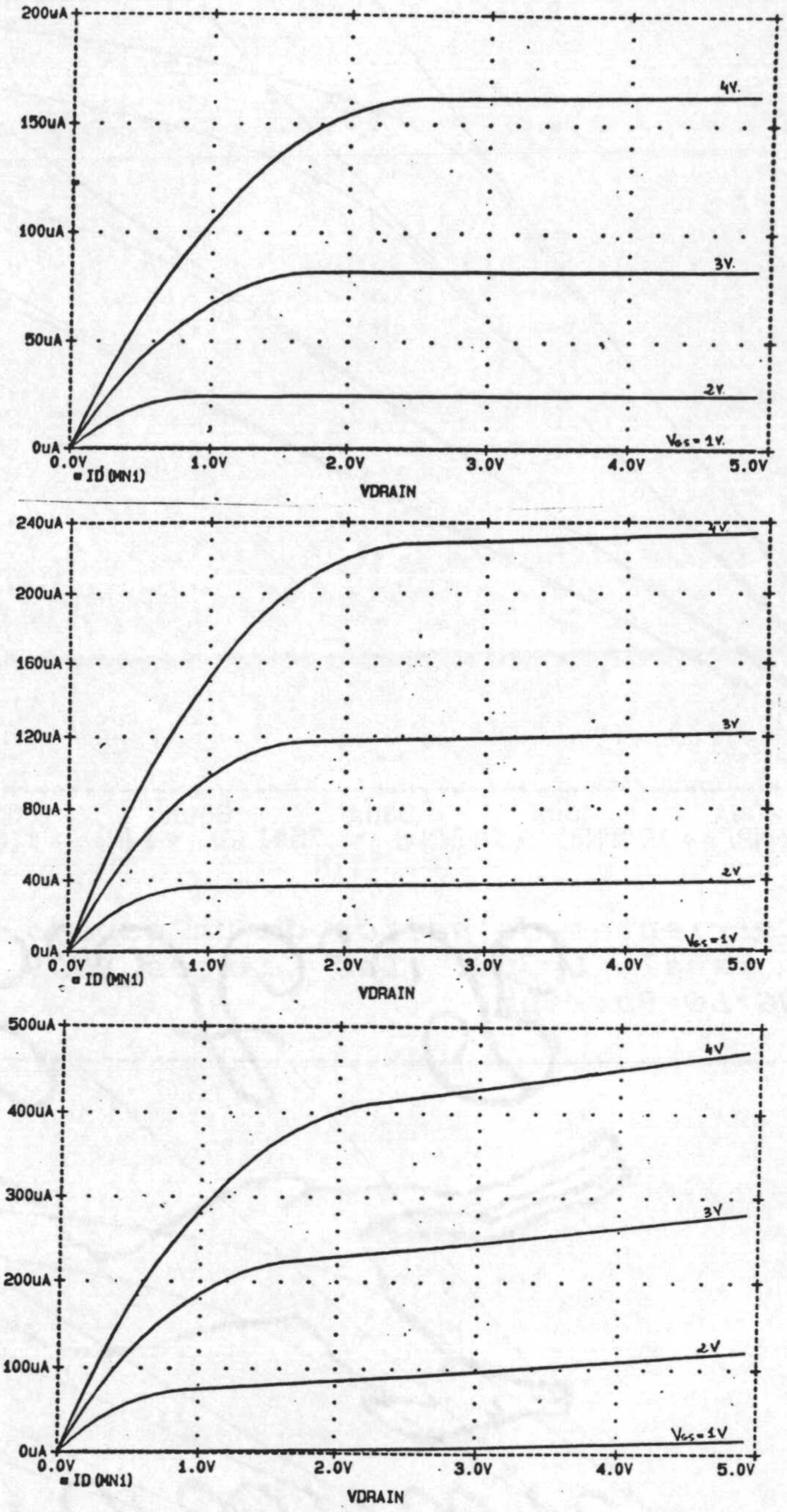


FIGURA 2.8: Curvas de transferencia DC de transistores NMOS del proceso CMOS de la U.C.L. (a) Con tamaño de puerta WG/LG=6µm/6µm. (b) WG/LG=6µm/4.5µm. (c) WG/LG=6µm/3µm.

Como puede observarse, para las nuevas longitudes de puerta consideradas se obtiene una falta de proporcionalidad entre las corrientes de entrada y de salida bastante acusada en comparación con la referida a una longitud de puerta de 6µm (figura 2.6).

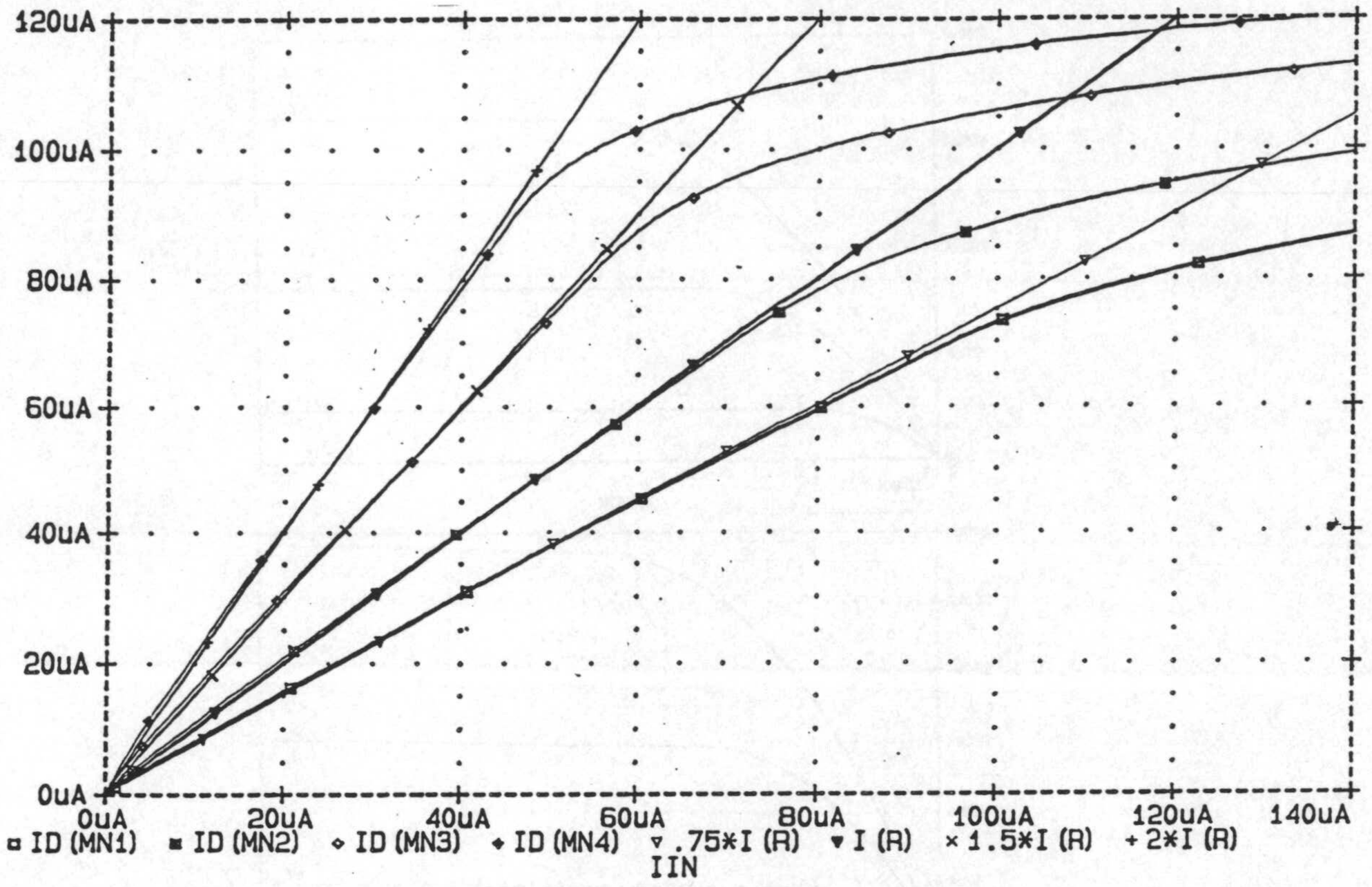


FIGURA 2.9: Corrientes de salida de un espejo NMOS con longitudes de canal L=3µm. Las cargas PMOS tienen tamaños de puerta WG/LG=9µm/6µm.

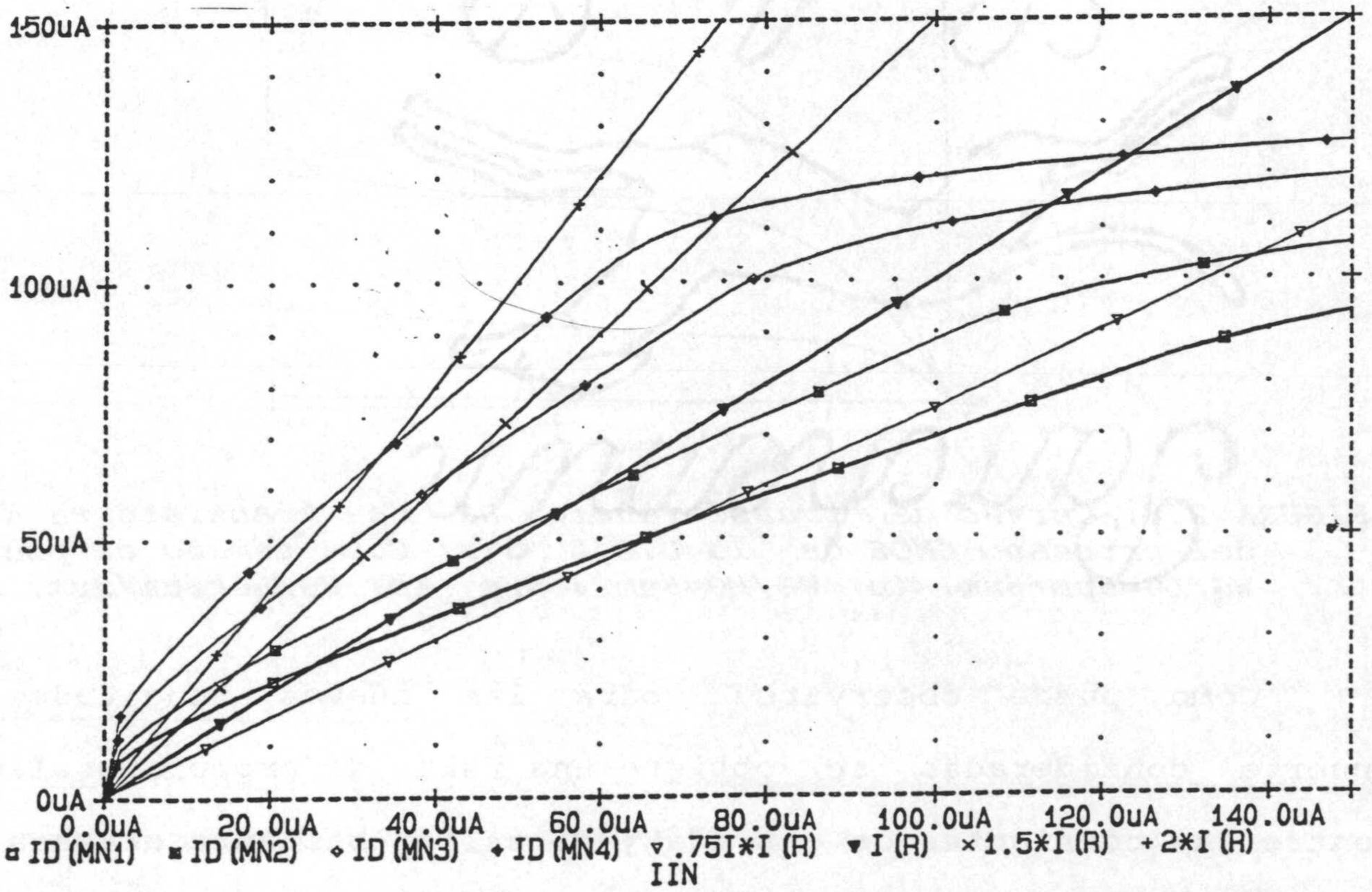


FIGURA 2.10: Corrientes de salida de un espejo NMOS con longitudes de canal L=1.5µm. Las cargas PMOS tienen tamaños de puerta WG/LG=9µm/6µm.

Los errores de corriente obtenidos en la simulación se ajustan bastante bién a los estimados por las expresiones (2.14) y (2.19). En efecto, en la figura 2.11 se compara la diferencia de corrientes observada para el transistor MN2 (transistor de salida del espejo cuyas dimensiones coinciden con el de entrada) con la que resulta de sumar las contribuciones de error dadas por (2.14) y (2.19). En dicha figura, el error debido a la diferencia de tensiones umbral se ha calculado suponiendo que σ (expresión (2.16)) depende de la longitud de canal del dispositivo como [188]:

$$\sigma = 8.15.10^{-22}.(\varepsilon_{ox})^{-1}.T_{ox}.L^{-3}$$

2.3.2 Detectores de umbral de corriente.

Utilizando los modelos simplificados de dispositivos MOS del apartado 2.3.1 (descritos por las expresiones 2.1 a 2.5) efectuaremos un análisis cuantitativo del funcionamiento de un detector de umbral de corriente a fin de determinar cómo afecta el tamaño de los transistores al comportamiento estático del mismo.

Consideremos el detector de umbral de corriente mostrado en la figura 2.12.a, en el cual se supone que los transistores MPR1 y MPR2 tienen la misma transconductancia β_P, y los transistores MN1 y MN2 la misma transconductancia β_N.

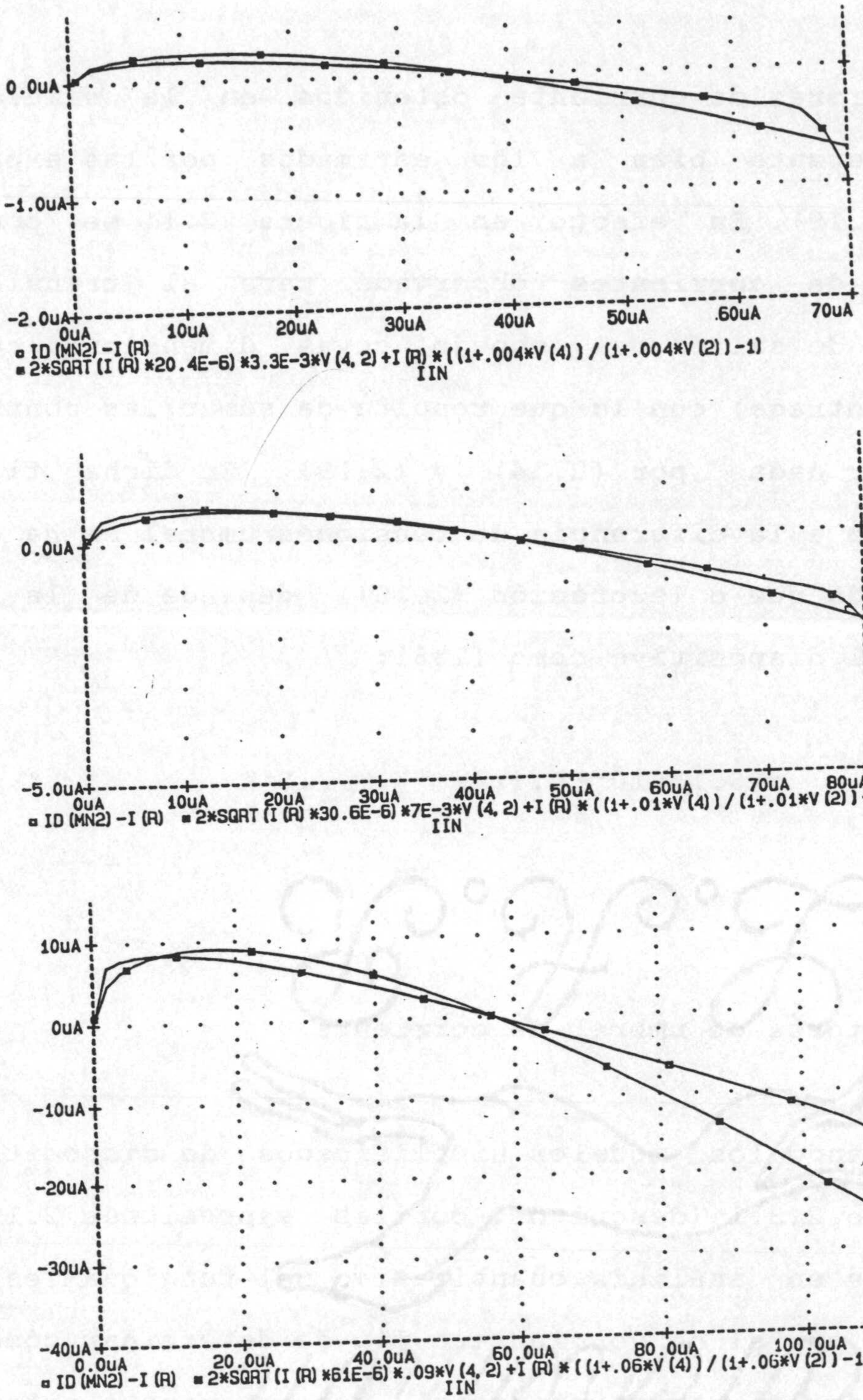


FIGURA 2.11: Diferencia entre corrientes de entrada y de salida del espejo NMOS para distintos valores de longitud de canal. (a) L=4.5µm. (b) L=3µm. (c) L=1.5µm.

La curva de transferencia del detector de umbral debe presentar una zona de transición (ver figura 2.12.b) correspondiente al funcionamiento simultáneo en saturación de los

transistores MN2 y MPR2. Denominemos I_{iL} e I_{iH} a los valores de corriente de entrada que delimitan dicha zona de transición, y sean V_{oH} y V_{oL} las correspondientes tensiones de salida.

Cuando $I_{in} < I_{iL}$ el transistor MPR2 funciona en regimen lineal y el MN2 en saturación. Si $I_{in} > I_{iL}$, MPR2 funciona en saturación y MN2 en régimen lineal.

Despreciando el efecto de la modulación de la longitud de canal, para valores de corriente $I_{in} \geq I_{iH}$, la corriente I_o en el circuito de la figura 2.12 puede evaluarse utilizando las

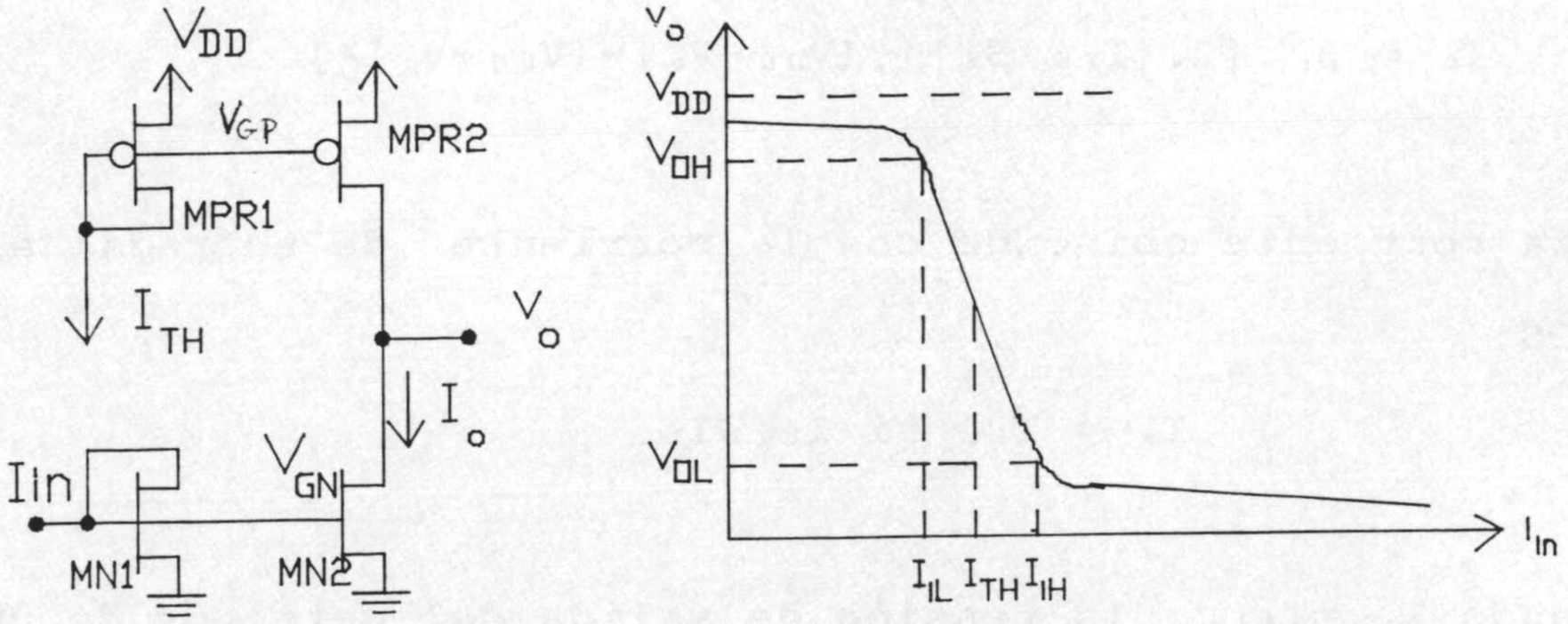


FIGURA 2.12: Detector de un umbral de corriente. (a) esquema de circuito. (b) Curva de transferencia DC.

expresiones (2.1) y (2.2) para los transistores MN2 y MN1 respectivamente, resultando:

$$I_o = \beta_N . (2.(I_{in}/\beta_N)^{1/2} . V_o - V_o^2) ; \quad I_{in} \geq I_{iH} \qquad (2.21)$$

dicha corriente coincide con la suministrada por MPR2, es decir:

$$I_o = I_{TH} \text{ si } I_{in} \geq I_{iH} \qquad (2.22)$$

Cuando $I_{in}=I_{iH}$, la tensión de salida del detector de umbral es $V_o=V_{oH}$. Sustituyendo estos valores en (2.21) resulta:

$$I_{iH} = [(I_{TH}+\beta_N . V_{oL}^2)/(2.\beta_N^{1/2} . V_{oL})]^2 \qquad (2.23)$$

Por otra parte, cuando $I_{in} \leq I_{iL}$, la corriente I_o podemos evaluarla utilizando las expresiones (2.3) y (2.4) para los transistores MPR2 y MPR1 respectivamente, resultando:

$$I_o = \beta_P . [2.(I_{TH}/\beta_N)^{1/2} . (V_{DD}-V_o)-(V_{DD}-V_o)^2] \qquad (2.24)$$

esta corriente coincide con la corriente de entrada al espejo NMOS:

$$I_o = I_{in} \text{ si } I_{in} \leq I_{iL} \qquad (2.25)$$

Cuando $I_{in}=I_{iL}$, la tensión de salida del detector de umbral es $V_o=V_{oH}$. La relación entre I_{iL} y V_{oH} es, según las expresiones (2.23) y (2.24):

$$I_{iL} = \beta_P . (2.(I_{TH}/\beta_P) . (V_{DD}-V_{oH})-(V_{DD}-V_{oH})^2) \qquad (2.26)$$

Imponiendo la condición de que las corrientes que limitan la zona de transición del detector de umbral vengan dadas por:

$$I_{iL} = I_{TH} - \varepsilon . I_0$$

$$I_{iH} = I_{TH} + \varepsilon . I_0 \; ; \; 0<\varepsilon<<1 \qquad (2.27)$$

en donde I_0 es la unidad básica de corriente utilizada en el circuito, las tensiones V_{oL} y V_{oH} que resultan de (2.23) y (2.26) son:

$$V_{oL} = \beta_N^{-½} . [(I_{TH}+\varepsilon . I_o)^{½} - (\varepsilon . I_0)^{½}]$$

$$V_{oH} = V_{DD} - \beta_P^{-½} . [I_{TH}^{½} - (\varepsilon . I_0)^{½}] \qquad (2.28)$$

Estas expresiones permiten estimar los factores de transconductancia β_N y β_P mínimos necesarios para que los niveles de tensión de salida del detector de umbral sean los adecuados.

Cuando la salida del detector de umbral se utilice para conmutar puertas lógicas CMOS binarias, lo ideal es exigir que los niveles de tensión referidos cumplan:

$$V_{oL} < V_{TN}$$

$$V_{oH} > V_{DD} - V_{TP} \qquad (2.29)$$

Con ello se garantizan unos márgenes de ruido aceptables y se evita que dichas puertas produzcan consumo estático de potencia cuando la corriente de entrada al detector de umbral se mantiene próxima a su zona de transición. No obstante, en ocasiones pueden interesar condiciones menos restrictivas que las (2.29) a fin de reducir el tamaño del detector de umbral. En la fase de diseño de dicho detector deberá en general

adoptarse una situación de compromiso tras valorar el aumento de consumo y deterioro de márgenes de ruido que ocasiona la reducción de tamaño.

El parámetro ε de las expresiones 2.27 y 2.28 deberá tener un valor lo suficientemente pequeño como para garantizar la discriminación correcta entre los valores de corriente correspondientes a los niveles lógicos anterior y posterior al umbral.

En un detector de multiumbral utilizado para discriminar r niveles lógicos, los valores de corriente umbral tienen la forma $I_{TH}(k)=(k+1/2).I_0$, $k=0,1,..,r-1$. Los valores de corriente correspondientes al nivel lógico k estarán comprendidos entre $I(k)=k.I_0 \pm \delta I$, en donde δI es una cota superior de los errores de corriente producidos por los espejos que conectan con el detector multiumbral (ver apartado 2.3.1). En δI pueden incluirse también los errores debidos a los espejos que constituyen el detector multiumbral.

Para que la discriminación de niveles sea correcta cada uno de los detectores umbral deberá tener asignado un parámetro ε tal que (ver figura 2.13):

$$0 < \varepsilon < 1/2 - \delta I/I_0 \qquad (2.30)$$

En la práctica, el valor de ε no puede hacerse tan pequeño

como se quiera debido al efecto de la modulación de la longitud del canal de los transistores que constituyen el detector de umbral. La transición de V_{oH} a V_{oL} de la salida es tanto más gradual cuanto más significativo sea este efecto.

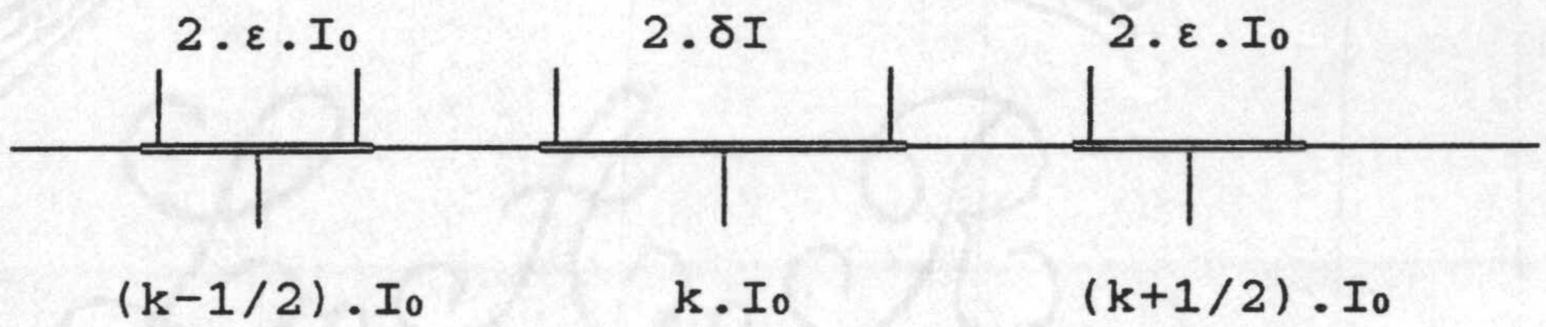


FIGURA 2.13: Diagrama de corrientes entre dos umbrales consecutivos.

En las figuras 2.14 y 2.15 se muestran las curvas de transferencia DC obtenidas con SPICE para dos detectores de umbral y distintos valores de corriente umbral. La figura 2.14 corresponde a un detector en el cual la relación de tamaño de los transistores PMOS y NMOS es de WG/LG=6µm/6µm, y la figura 2.15 a otro con relaciones de tamaño WG/LG=6µm/4.5µm.

Las líneas rectas de ambas figuras representan los valores de tensión V_{oL} y V_{oH} previstos por las ecuaciones (2.28) suponiendo que $\varepsilon.I_o = 2\mu A$.

Con las longitudes de canal consideradas en la obtención de la figura 2.15, debido a que la modulación de la longitud del canal tiene ya un efecto apreciable (ver figura 2.8), las transiciones de la tensión de salida del detector son más

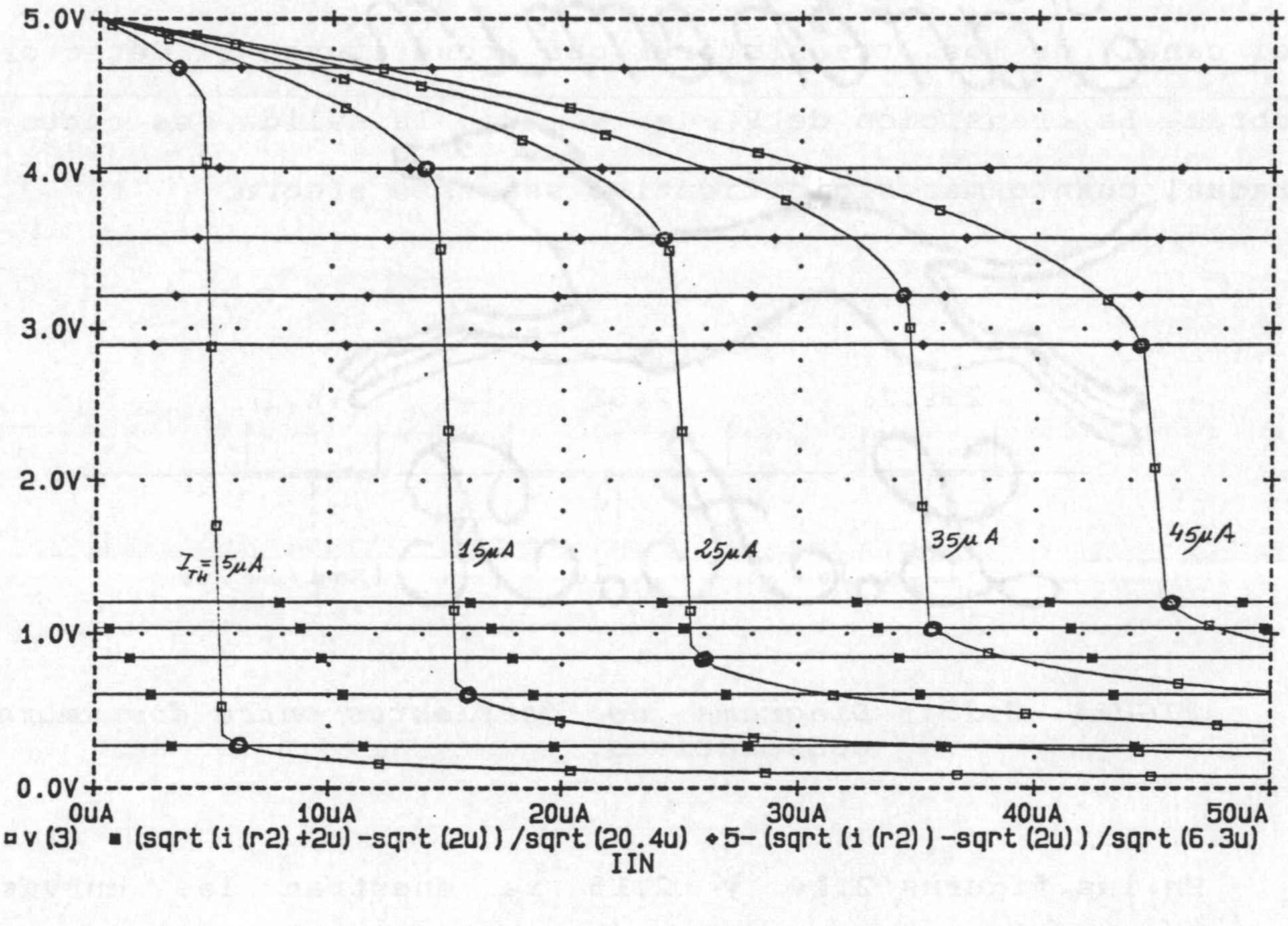


FIGURA 2.14: Curvas de transferencia DC de un detector de umbral con WG/LG=6μm/6μm.

graduales. Como resultado, los valores de V_{oL} y V_{oH} estimados por las expresiones (2.28) se consiguen realmente para valores de corriente que difieren de I_{TH} más de la cantidad prevista $\varepsilon . I_0 = 2\mu A$.

2.3.3 Comportamiento dinámico de los circuitos básicos CMOS en modo de corriente.

En los circuitos CMOS en modo de corriente, las capacidades parásitas asociadas a los transistores e interconexiones se

someten a cambios de carga normalmente pequeños en comparación

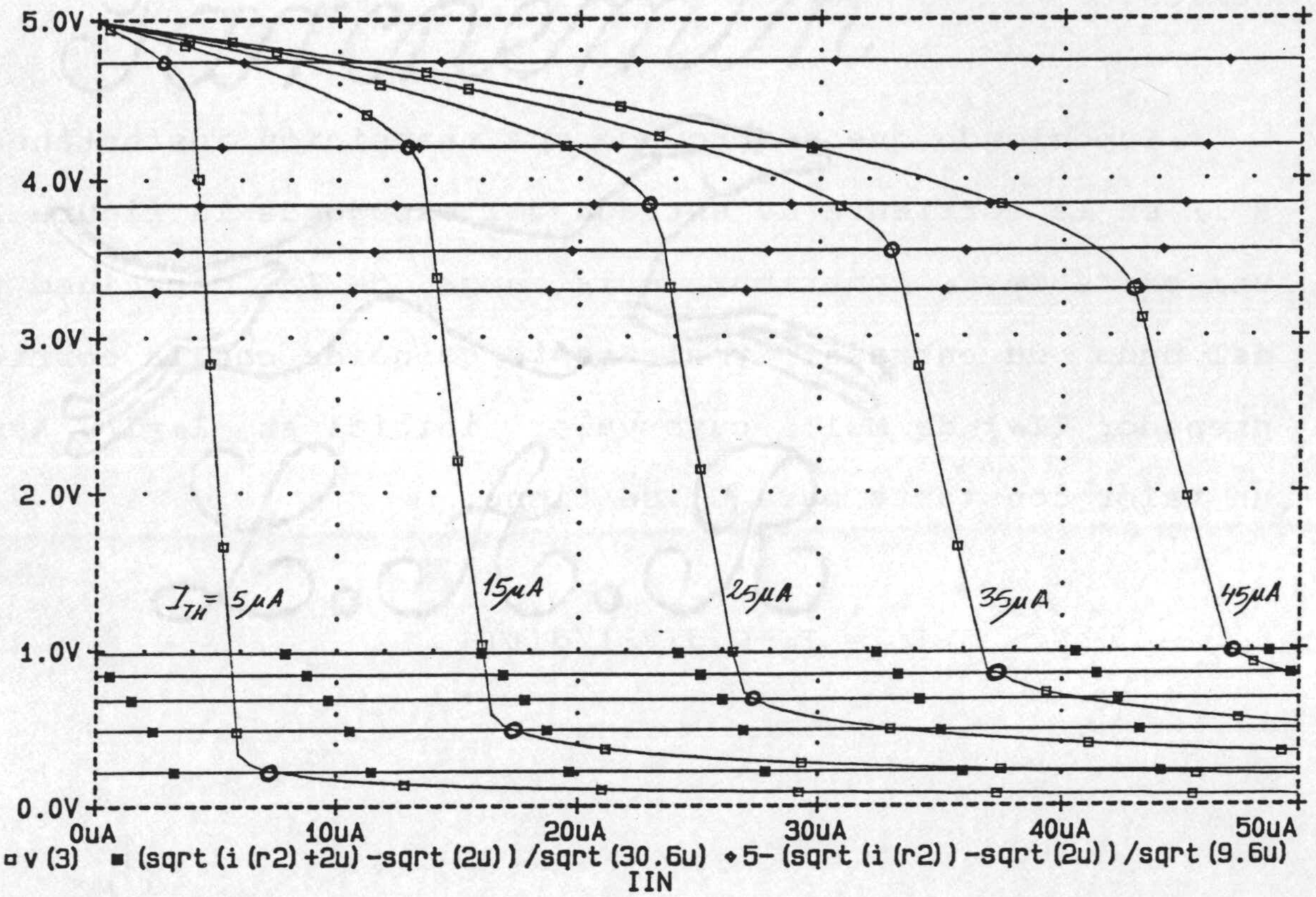


FIGURA 2.15: Curvas de transferencia DC de un detector de umbral con WG/LG=6μm/4.5μm.

con los que tienen lugar en circuitos binarios en modo de tensión. No obstante, los tiempos necesarios para efectuar dichos cambios de carga pueden en ocasiones ser mayores debido a que se realizan con corrientes más pequeñas.

Para obtener una medida de la velocidad de respuesta de los espejos de corriente y detectores de umbral procederemos al cálculo de los tiempos máximos de transición de las salidas de dichos circuitos al aplicarle una transición entre niveles de corriente de entrada. En la figura 2.16 se indican los circuitos equivalentes utilizados. Las capacidades indicadas en la figura se supone que incluyen tanto las asociadas a los

transistores como las contribuciones de las líneas de interconexión.

Suponiendo que se produce una transición instantánea de I_1 a I_2 en la corriente de entrada del espejo de la figura 2.16.a, una parte de I_2 contribuye a la carga de la capacidad total C del nudo de entrada, y el resto coincide con la corriente de drenador (I_D) de MN1, cuyo valor inicial es $I_D=I_1$. Asumiendo un valor constante para C, se tiene:

$$I_2 = I_D + C.d(V_C)/d(t) \qquad (2.31)$$

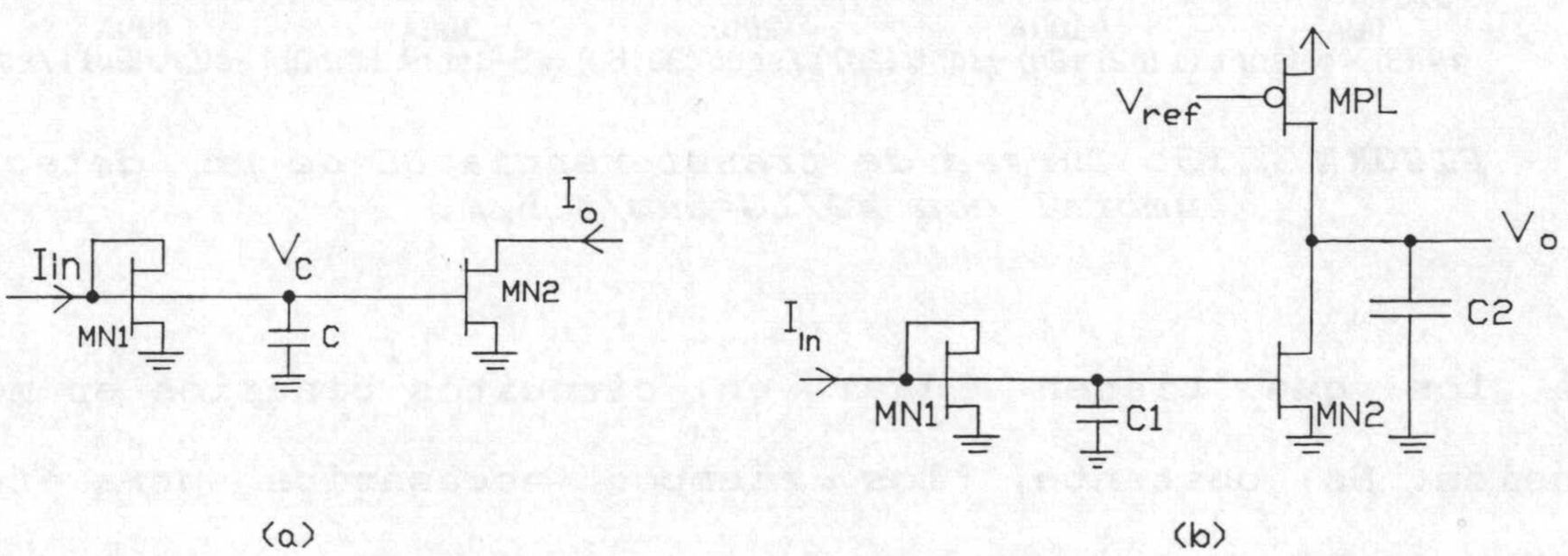


FIGURA 2.16: Circuitos equivalentes para el cálculo de los tiempos de transición. (a) Espejo de corriente. (b)Detector de umbral de corriente.

V_C podemos relacionarla con la corriente de drenador de MN1 mediante:

$$V_C = (I_D/\beta_N)^{1/2} + V_{TN} \qquad (2.32)$$

cuya sustitución en (2.31) da:

$$I_2 = I_D + \tfrac{1}{2}.C.(\beta_N . I_D)^{-\frac{1}{2}} . d(I_D)/d(t) \qquad (2.33)$$

Suponiendo que la transición de la corriente de entrada tiene lugar en t=0, la ecuación anterior tiene como soluciones:

Si $I_2 > I_1$ (transición ascendente):

$$t = \tfrac{1}{2}.C.(\beta_N . I_2)^{-\frac{1}{2}} . Ln\{\frac{(1+a_1(t)).(1-a_2)}{(1-a_1(t)).(1+a_2)}\} \qquad (2.34)$$

Si $I_1 > I_2 > 0$ (transición descendente):

$$t = \tfrac{1}{2}.C.(\beta_N . I_2)^{-\frac{1}{2}} . Ln\{\frac{(a_1(t)+1).(a_2-1)}{(a_1(t)-1).(a_2+1)}\} \qquad (2.35)$$

en donde:

$$a_1(t) = (I_D(t)/I_2)^{\frac{1}{2}}$$

$$a_2 = (I_1/I_2)^{\frac{1}{2}}$$

Si $I_2 = 0$ (en una transición descendente):

$$t = C.\beta_N^{-\frac{1}{2}} . [(I_D(t))^{-\frac{1}{2}} - I_1^{-\frac{1}{2}}] \qquad (2.36)$$

Para obtener una medida de los tiempos de transición del espejo de corriente, las expresiones (2.34) a (2.36) deben evaluarse con valores de I_D muy próximos a I_2.

El tiempo de subida (t_r) lo definimos como el tiempo transcurrido entre una transición ascendente (de I_1 a I_2 en t=0) de la corriente de entrada y el instante en que se cumple:

$$I_D(t_r) = I_2 - \varepsilon.I_0 ;\ 0<\varepsilon<<1 \qquad (2.37)$$

Denominaremos tiempo de bajada (t_f) al transcurrido desde que se produce una transición descendente de la corriente de entrada hasta conseguir:

$$I_D(t_f) = I_2 + \varepsilon.I_0 ;\ 0<\varepsilon<<1 \qquad (2.38)$$

Los tiempos de transición que se obtienen resultan ser máximos cuando se trata de transiciones ascendentes de 0 a I_0, o de transiciones descendentes de (r-1).I_0 a 0, siendo r-1 el mayor nivel lógico utilizado en el circuito. Estos tiempos de transición máximos vienen dados por (expresiones (2.34) y (2.36) con los valores de I_D indicados en (2.37) y (2.38) respectivamente):

$$t_{r,max.} = C.(\beta_N.I_0)^{-1/2}.Ln\left\{\frac{1+(1-\varepsilon)^{1/2}}{\varepsilon^{1/2}}\right\} \qquad (2.39)$$

$$t_{f,max.} = C.(\beta_N.I_0)^{-1/2}.(\varepsilon^{-1/2}-(r-1)^{-1/2}) \qquad (2.40)$$

En la figura 2.17 se muestra la respuesta transitoria obtenida con SPICE para un espejo de corriente como el de la figura 2.16.a, en el cual las dimensiones de puerta de los transistores MN1 y MN2 son WG/LG=6µm/6µm. En la simulación se han considerado transiciones descendentes de corriente de entrada I(R) desde 3.I_0 a 0, y ascendentes desde 0 a I_0. La figura 2.17.a corresponde a un valor de corriente unidad I_0=10µA, y la 2.17.b a I_0=20µA. ID(MN2) representa la corriente de salida cuando en C (ver figura 2.16.a) sólo se incluye la

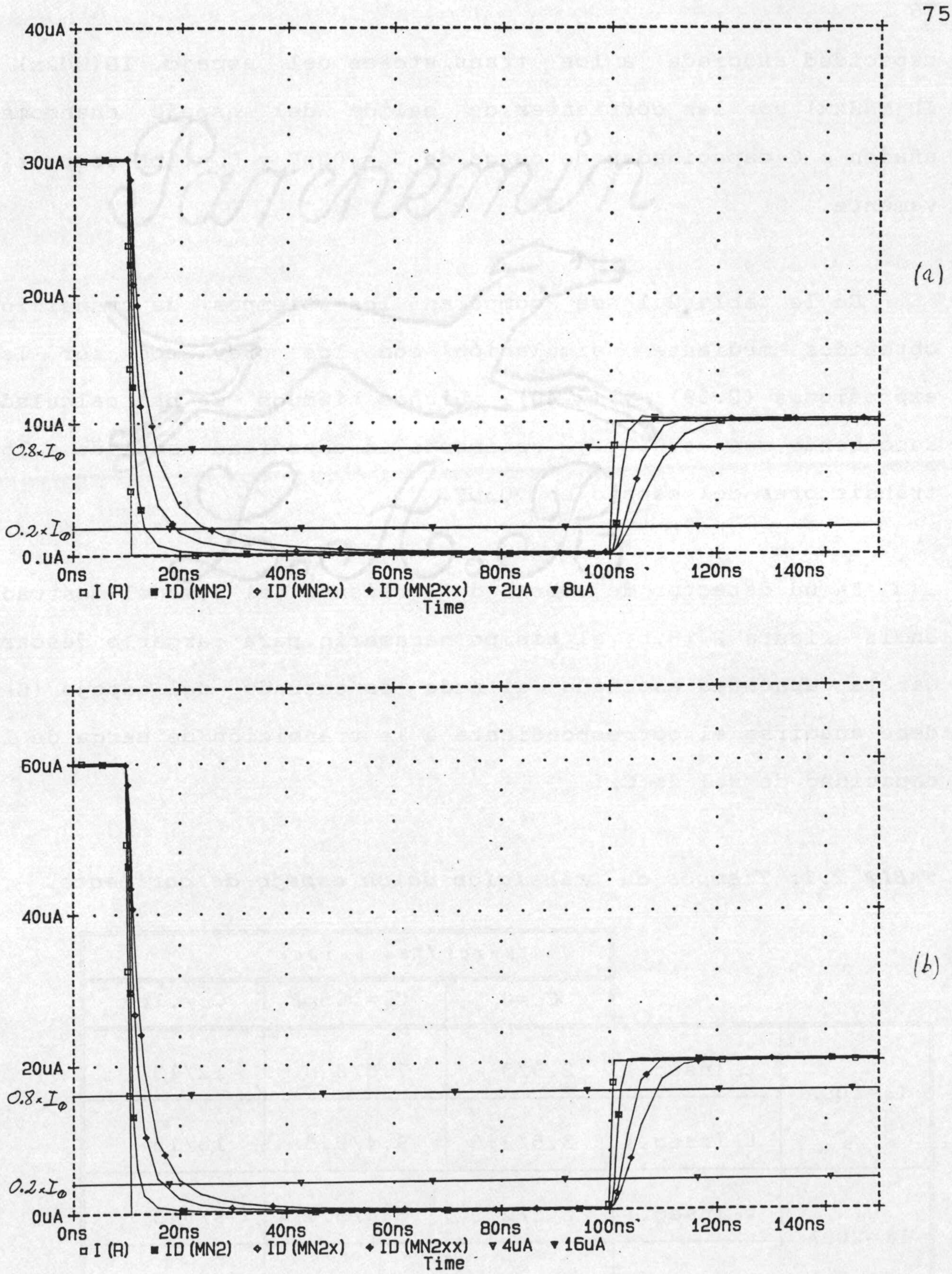


FIGURA 2.17: Respuesta transitoria de un espejo de corriente NMOS. (a) I_0 =10µA. (b) I_0 =20µA.

capacidad asociada a los transistores del espejo. ID(MN2x) e ID(MN2xx) son las corrientes de salida del espejo cuando se añaden a C capacidades de carga de C_L=.05pF y C_L=.1pF respectivamente.

En la tabla 2.1 se comparan los tiempos de transición obtenidos mediante simulación con los previstos por las expresiones (2.39) y (2.40). Dichos tiempos se han calculado suponiendo que ε=0.2, y estimando la capacidad asociada a los transistores del espejo en .03pF.

En un detector de umbral de corriente tal como el mostrado en la figura 2.16.b, al tiempo necesario para cargar o descargar la capacidad asociada al nudo de entrada del espejo (C_1) debe añadirse el correspondiente a la transición de carga de la capacidad de salida C_2.

TABLA 2.1: Tiempos de transición de un espejo de corriente.

		$t_{SPICE}/t_{calculado}$		
		C_L=0	C_L=.05pF	C_L=.1pF
I_0=10µA	t_r (nseg.)	2.9/3	7.6/8	12/13
	t_f (nseg.)	3.5/3.5	9.4/9.3	15/15
I_0=20µA	t_r (nseg.)	2.2/2.1	5.6/5.7	9/9.3
	t_f (nseg.)	2.5/2.5	6.7/6.6	10.5/10.7

Los cambios de tensión a los que se somete C_2 son en magnitud próximos a la tensión de alimentación, y necesitarán en general más tiempo para producirse que los cambios de corriente en el espejo.

Para evaluar los tiempos de transición de la salida del detector de umbral supondremos que se producen transiciones instantáneas de la corriente de salida del espejo NMOS entre valores inferiores y superiores a la corriente umbral. La transición de la tensión de salida es ascendente (de V_{oL} a V_{oH}) si la corriente de drenador de MN2 efectua una transición descendente de $I_1 > I_{TH}$ a $I_2 < I_{TH}$ (I_{TH} es la corriente umbral). La corriente de carga de C_2 es aproximadamente constante y viene dada por la diferencia $I_{TH} - I_2$. Esta corriente es mínima cuando I_2 corresponde al nivel lógico inmediatamente inferior al umbral. En consecuencia, una medida del tiempo de subida máximo, considerando $I_2 = I_{TH} - I_0/2$, es la siguiente:

$$t_{r,max.} \approx 2.C_2.(V_{oH} - V_{oL})/I_0 \qquad (2.41)$$

De igual manera, en una transición de salida descendente el tiempo invertido es máximo cuando el nivel final de corriente es el inmediatamente superior a I_{TH} (es decir $I_2 = I_{TH} + I_0/2$), obteniéndose un tiempo de bajada máximo que coincide con el de subida dado por (2.41), es decir:

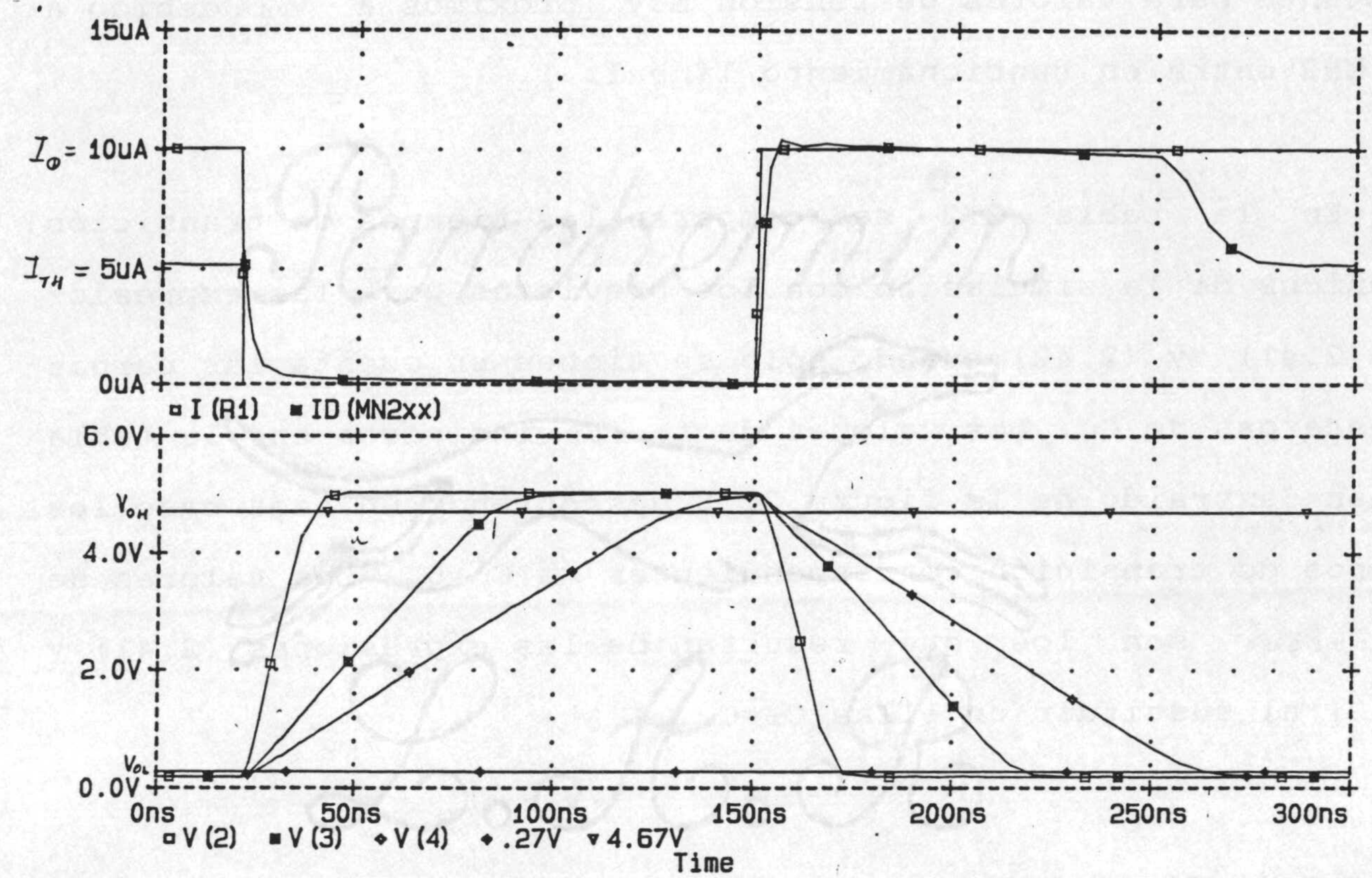
$$t_{f,max.} \approx 2.C_2.(V_{oH} - V_{oL})/I_0 \qquad (2.42)$$

Un análisis transitorio con SPICE del detector de umbral de la figura 2.16.b produce los resultados mostrados en la figura 2.18. La corriente de entrada al detector se ha sometido a transiciones de I_0 a 0 y de 0 a I_0. I_0=10μA para la figura 2.18.a e I_0=20μA para la 2.18.b. En cada caso, la corriente umbral considerada es $I_0/2$.

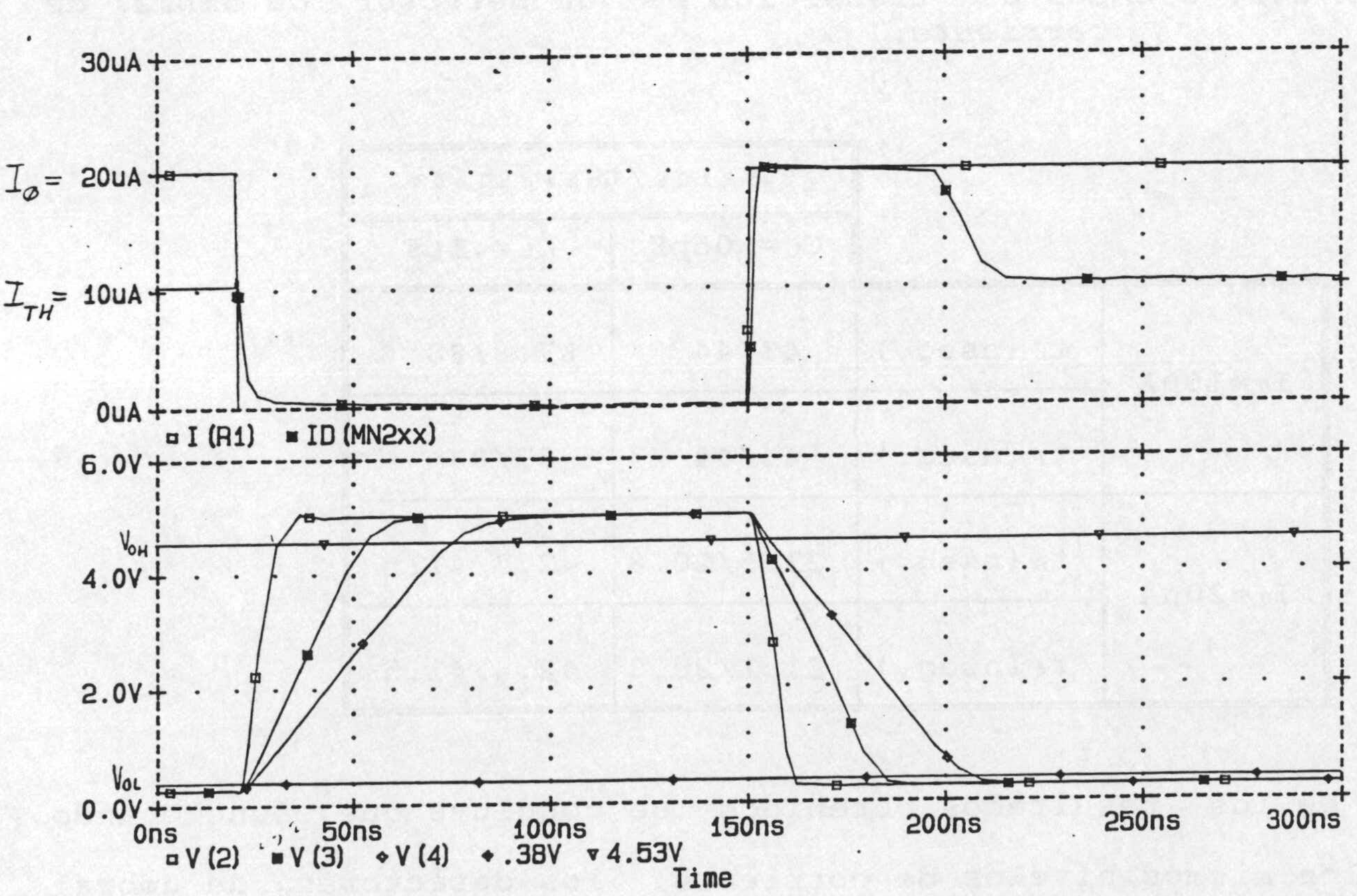
V(2) en la figura 2.18 es la tensión de salida del detector de umbral cuando este no tiene conectada ninguna capacidad de carga; es decir, cuando C_2 coincide con la suma de las capacidades de drenador de los transistores de salida del detector (ver figura 2.16.b). V(3) y V(4) son las tensiones de salida cuando a C_2 se añaden capacidades de carga C_L=.05pF y C_L=.1pF respectivamente.

Los valores constantes añadidos a las gráficas de las tensiones de salida representan las tensiones V_{oL} y V_{oH} calculadas mediante las expresiones (2.28).

En la parte superior de las figuras 2.18.a y 2.18.b se representan las transiciones de la corriente de entrada I(R1), así como la corriente de drenador ID(MN2xx) del transistor de salida MN2 (ver figura 2.16.b) en el caso en que la capacidad de carga añadida es C_L=.1pF. Como puede observarse en estas gráficas, tal y como habíamos supuesto, dicha corriente de drenador es aproximadamente constante durante el tiempo de transición de la salida. En una transición descendente de la salida, la corriente que descarga la capacidad C_2 deja de ser



(a)



(b)

FIGURA 2.18: Respuesta transitoria de un detector de umbral de corriente. (a) I_0 =10μA. (b) I_0 = 20 μA.

constante para valores de tensión muy próximos a V_{oL} debido a que MN2 entra en funcionamiento lineal.

En la tabla 2.2 se comparan los tiempos de transición obtenidos de la simulación con los previstos por las expresiones (2.41) y (2.42) cuando sólo se tienen en cuenta las cargas y descargas de C_L. Los valores de t_{SPICE} indicados en la tabla se han extraido de la figura 2.18 descontando en cada caso los tiempos de transición correspondientes a $C_L=0$. Los valores de $t_{calculado}$ son los que resultan de las expresiones (2.41) y (2.42) al sustituir en ellas $C_2=C_L$.

TABLA 2.2: Tiempos de transición de un detector de umbral de corriente.

		$t_{SPICE}/t_{calculado}$	
		C_L=.05pF	C_L=.1pF
I_o=10μA	t_r (nseg.)	43/44	87.8/88
	t_f (nseg.)	44/44	87/88
I_o=20μA	t_r (nseg.)	21.5/20.8	42.7/41.5
	t_f (nseg.)	21.1/20.8	42.4/41.5

De los resultados obtenidos se concluye que, funcionando con los mismos niveles de corriente, los detectores de umbral introducirán en general bastante mayor retardo que los espejos de corriente. Para un valor dado de corriente unidad, la forma

de consegir tiempos de transición pequeños es haciendo que los detectores de umbral tengan capacidades de carga mínimas. Cuando la velocidad sea un requisito esencial puede ser conveniente incluir en el esquema del detector de umbral un inversor CMOS, el cual sea el único circuito que conecte con la salida del detector original y esté situado físicamente lo más próximo de esta.

2.4 Puertas umbral y multiumbral CMOS en modo de corriente.

En la sección anterior hemos obtenido expresiones que caracterizan el comportamiento estático y dinámico de los circuitos básicos CMOS en modo de corriente. En esta sección presentamos posibles realizaciones CMOS de elementos umbral y multiumbral (ver apartado 1.1.1) basadas en los referidos circuitos básicos.

Una puerta umbral de n entradas puede construirse mediante dos espejos de corriente y un inversor binario CMOS según se muestra en el esquema presentado en la figura 2.19. Esta realización utiliza el modo de corriente para generar la suma ponderada y el modo de tensión para representar las entradas y la salida.

El transistor MPi de la figura 2.19 debe tener un tamaño proporcional al peso p_i de la correspondiente entrada. Las

corrientes suministradas por estos transistores, conmutadas selectivamente por las entradas, se suman en la entrada del espejo de corriente ME0-ME1. El transistor de salida de este espejo y el transistor MU constituyen el detector de umbral en el que se lleva a cabo la comparación de la suma ponderada de corrientes.

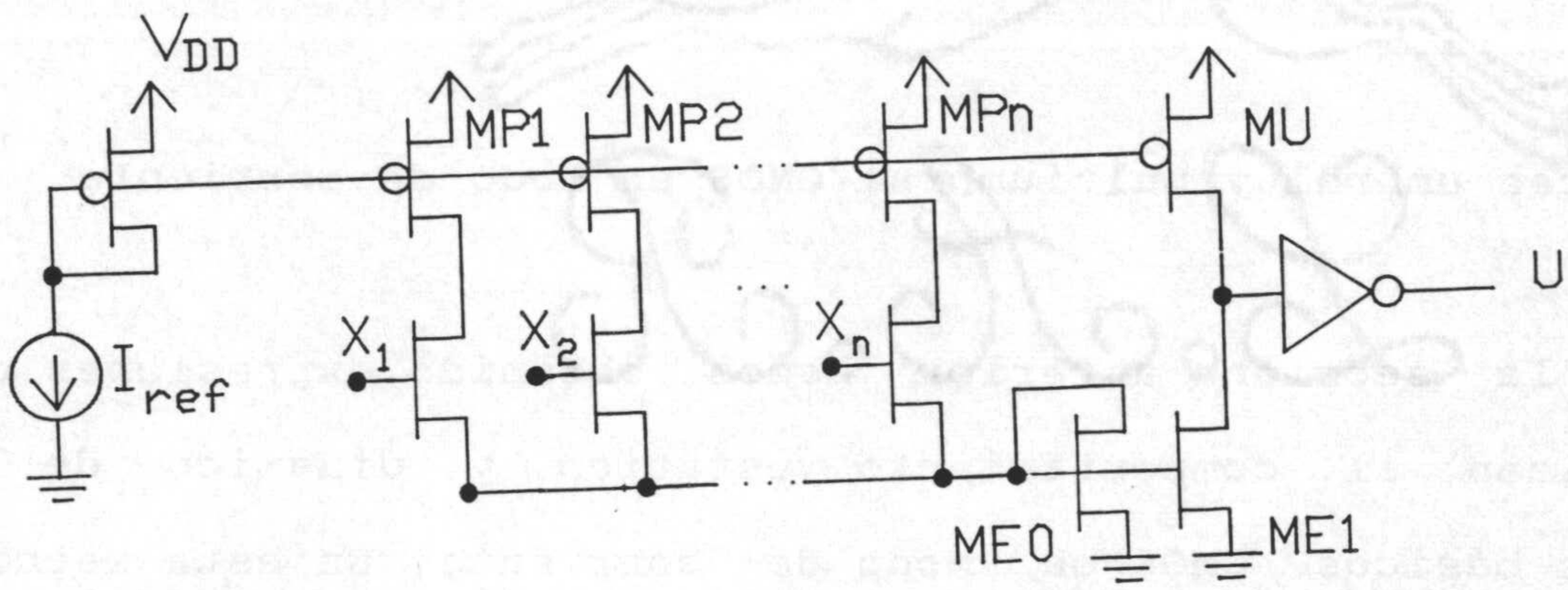


FIGURA 2.19: Puerta umbral CMOS en modo de corriente.

Asignando los tamaños apropiados de los transistores MPi y MU se establece el vector que caractiza la puerta umbral.

En la figura 2.20 presentamos una posible realización CMOS de una puerta multiumbral de n entradas y m umbrales. La suma ponderada de corrientes es en este caso entrada a un detector multiumbral cuyas salidas se combinan mediante una puerta XNOR binaria. Dado que las salidas del detector multiumbral sólo pueden tomar un conjunto reducido de combinaciones de valores, la síntesis de la operación XNOR de estas salidas admite

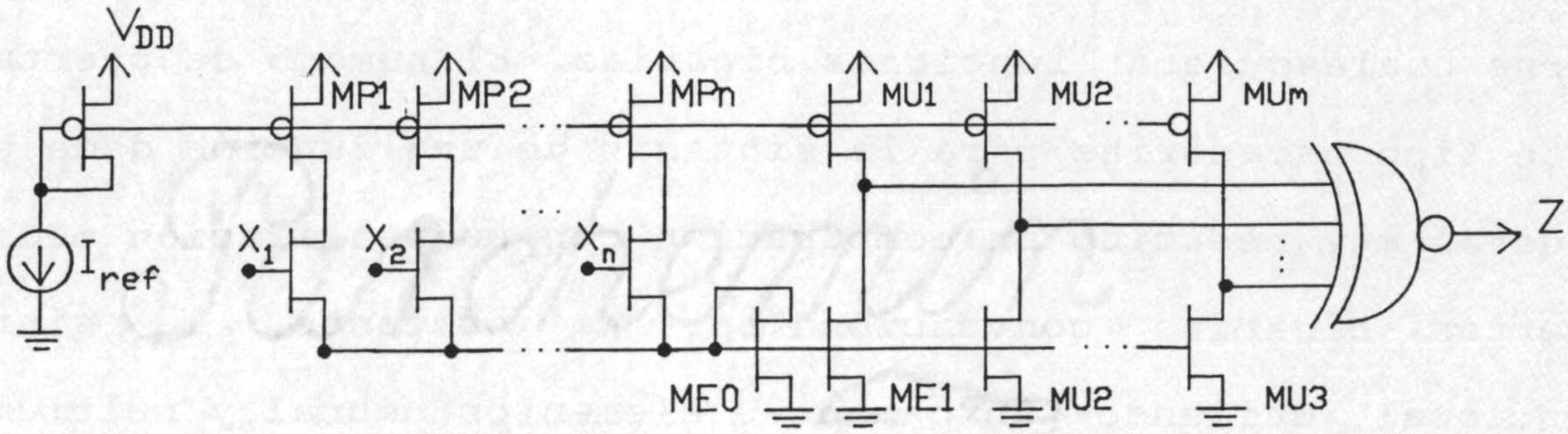
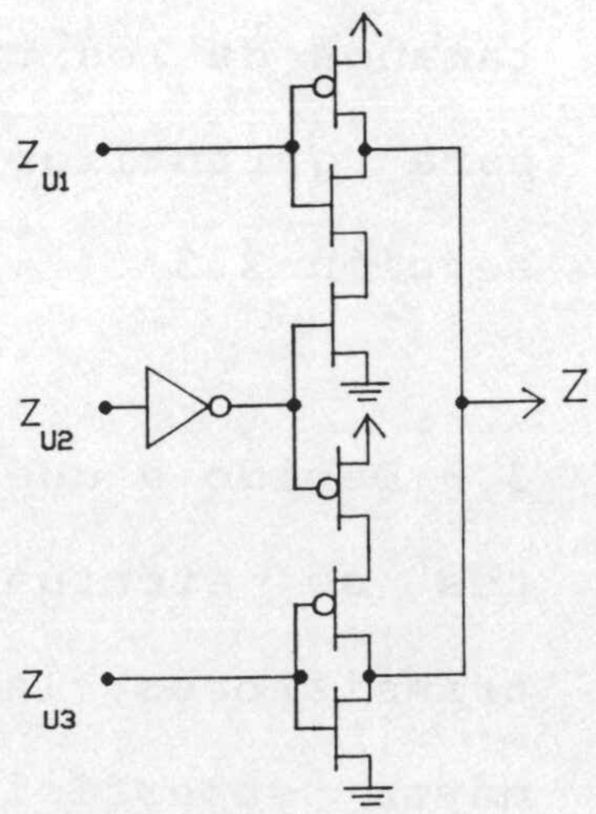


FIGURA 2.20: Puerta multiumbral CMOS de n entradas y m umbrales.

simplificaciones. En la figura 2.21 se muestra una posible realización CMOS estática de esta operación para una puerta de tres umbrales. Z_{U1}, Z_{U2}, y Z_{U3}, representan las salidas de los detectores de umbral U_1, U_2, y U_3, respectivamente. En la misma figura se indican las posibles combinaciones de valores de las Z_{U1} en función de la excitación.

E	Z_{U1}	Z_{U2}	Z_{U3}	Z
$E<U_3$	1	1	1	0
$U_3<E<U_2$	1	1	0	1
$U_2<E<U_1$	1	0	0	0
$U_1<E$	0	0	0	1

(a)



(b)

FIGURA 2.21: Etapa de salida de una puerta multiumbral de tres umbrales. (a) Codificación de la salida en función de la excitación. (b) Realización CMOS estática.

Como indicamos en el apartado 1.1.1, mediante puertas umbral y multiumbral tales como las descritas pueden sintetizarse cualesquiera funciones binarias. El número de puertas de este tipo necesarias para la síntesis de una función dada es en general muy reducido en comparación con su realización mediante puertas binarias convencionales. No obstante, la síntesis funcional utilando únicamente elementos umbral y multiumbral como los presentados tiene algunas limitaciones que deben tenerse en cuenta:

1.- El proceso de diseño lógico es más complejo que el basado en elementos binarios convencionales.

2.- Dado que el número de niveles lógicos utilizados puede ser elevado, el diseño eléctrico de las puertas requiere una buena caracterización previa de los dispositivos MOS que las configuran. De esta manera es posible adecuar los tamaños de los transistores y los niveles de corriente para garantizar que su funcionamiento sea fiable (ver sección 2.3).

3.- Debido a que la ponderación y comparación de corrientes se efectúa mediante el ajuste de tamaño de los transistores, las realizaciones integradas pueden requerir mayor superficie que las correspondientes binarias aun cuando el número total de transistores sea menor en el primer caso. La superficie media por dispositivo es tanto mayor cuanto más dispares sean los pesos y umbrales

utilizados.

Partiendo de un conjunto arbitrario de funciones lógicas, no se puede asegurar que la síntesis basada en sólo elementos umbral y multiumbral lleve a la realización más eficiente en cuanto a superficie ocupada. No obstante, debido a la forma de operación interna, estos circuitos son especialmente adecuados para la síntesis de funciones aritméticas binarias [152]. Este aspecto queda ilustrado en el siguiente apartado con una aplicación concreta: el diseño de un multiplicador binario CMOS en modo de corriente, que hemos construido combinando elementos umbrales con circuitos binarios convencionales.

En la actualidad, además de su potencial utilidad para la construcción de determinados circuitos de procesamiento digitales, los elementos umbral han recobrado un grán interés como celdas básicas de circuitos o redes neuronales [53, 57, 58, 85, 108, 137, 197].

Para implantar mecanismos de aprendizaje en circuitos neuronales, los elementos umbral que lo configuran deben tener la posibilidad de modificar dinámicamente sus pesos y/o umbral. Un modelo apropiado de neurona se consigue con una puerta umbral tal como la mostrada en la figura 2.19, en la cual las corrientes suministradas por los transistores MPi y MU vengan determinadas por entradas de control adicionales. Basándonos en este modelo, en la actualidad estamos desarrollando un prototipo de red neuronal CMOS [121], la cual

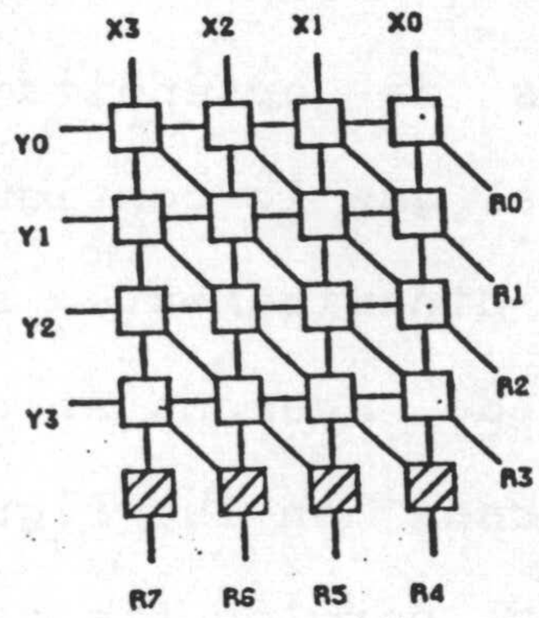
pretendemos integrar en un próximo MPC.

2.4.1 Diseño de un multiplicador binario CMOS en modo de corriente.

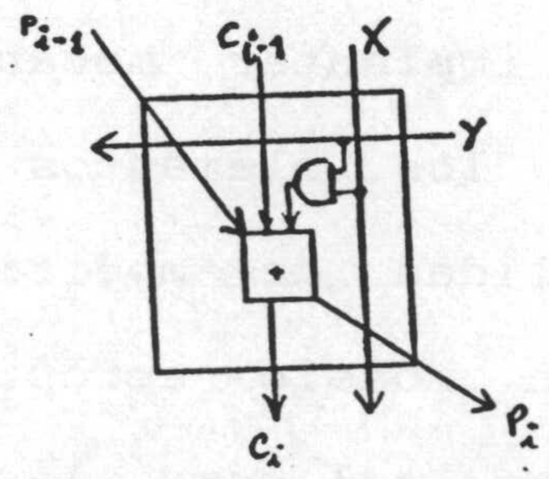
La realización física de un multiplicador binario combinacional requiere la utilización de celdas sumadoras completas de un bit. En la figura 2.22 se ilustra un esquema de multiplicador binario 4x4 en términos de dichas celdas [187]. Multipli

```
Multiplicando:                X3   X2   X1   X0
Multiplicador:                Y3   Y2   Y1   Y0
                              ------------------
                              X3Y0 X2Y0 X1Y0 X0Y0
                         X3Y1 X2Y1 X1Y1 X0Y1
                    X3Y2 X2Y2 X1Y2 X0Y2
               X3Y3 X2Y3 X1Y3 X0Y3
          ------------------------------------
Producto: R7   R6   R5   R4   R3   R2   R1   R0
```

(a)



(b)



(c)

FIGURA 2.22: Mutiliplicador paralelo de 4x4 bits.
(a) Definición de la operación realizada.
(b) Estructura del multiplicador.
(c) Celda básica.

cadores paralelo de mayor número de bits pueden construirse formando árboles de Wallace con varios de 4x4 bits [14].

Refiriéndonos a un multiplicador 4x4 como el indicado en la figura 2.22, varias de sus celdas básicas podemos sustituir-las por una sola celda como la indicada en la figura 2.23, que realiza la suma aritmética módulo dos, de cinco datos de un bit, proporcionando el resultado codificado en tres bits.

Aunque las entradas y salidas de la celda sean binarias, internamente realiza una suma ponderada en modo de corriente (con pesos unidad) de las entradas; es decir:

$$E = \sum_{1}^{4} a_i \cap b_i + C_{j-1} \quad ; \quad a_i, b_i, C_{j-1} \in \{0,1\}$$

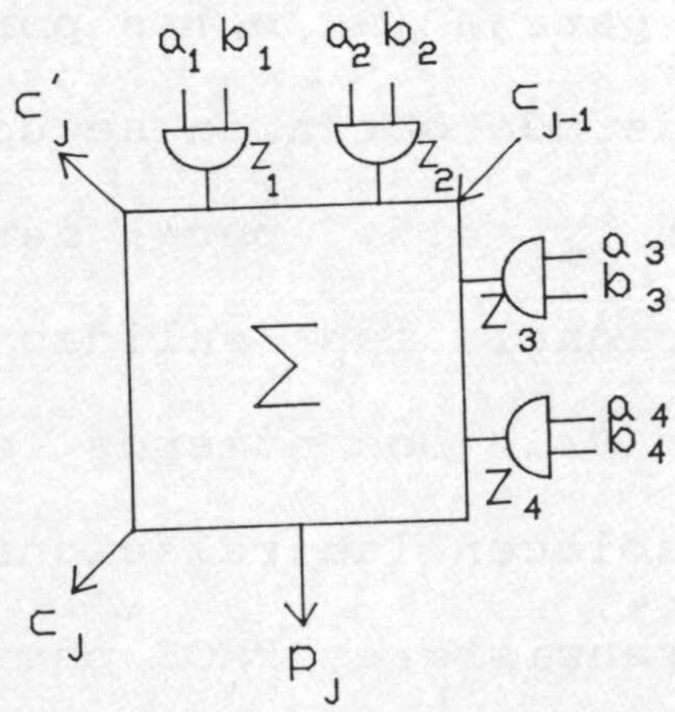


FIGURA 2.23: Celda básica propuesta para la construcción del multiplicador 4x4 en modo de corriente.

La corriente que representa la suma ponderada se compara entonces con distintos umbrales para determinar las salidas. La codificación de las salidas en función de E es la que se indica

en la tabla 2.3.

En la figura 2.24 presentamos una posible realización CMOS estática de la celda de la figura 2.23. En este esquema,

TABLA 2.3: Codificación de las salidas de la celda multiplicadora de la figura 2.23.

E	P_j	C_j	C_j '
0	0	0	0
1	1	0	0
2	0	1	0
3	1	1	0
4	0	0	1
5	1	0	1

la suma ponderada de corrientes se obtiene conmutando selectivamente fuentes de corriente de magnitud unidad. Cada fuente es conmutada por una pareja (a_i ,b_i) o por una entrada de acarreo C_{j-1}. La suma ponderada obtenida se compara con los umbrales 0.5, 1.5, 2.5, 3.5, y 4.5, para determinar, mediante lógica CMOS binaria adicional, las salidas de acarreo C_j y C_j ', y la salida de producto P_j. Los números entre paréntesis indicados en la figura establecen las relaciones relativas de tamaño que deben tener los transistores PMOS para efectuar la ponderación y comparación correcta de corrientes.

Para el diseño de la parte de la celda que funciona en modo de corriente deben tenerse en cuenta los resultados obtenidos en la sección anterior. El valor de la corriente unidad (I_0) utilizada es el principal factor determinante de

las características de velocidad y consumo de potencia de la celda. Una vez elegido el valor de I_0, las dimensiones de los transistores del espejo que efectua la suma ponderada se calculan de manera que su funcionamiento sea correcto cuando dicha suma sea máxima (en nuestro caso la suma máxima corresponde a un valor de corriente $5.I_0$). La longitud mínima de canal de los transistores está limitada por el error máximo de corriente permisible.

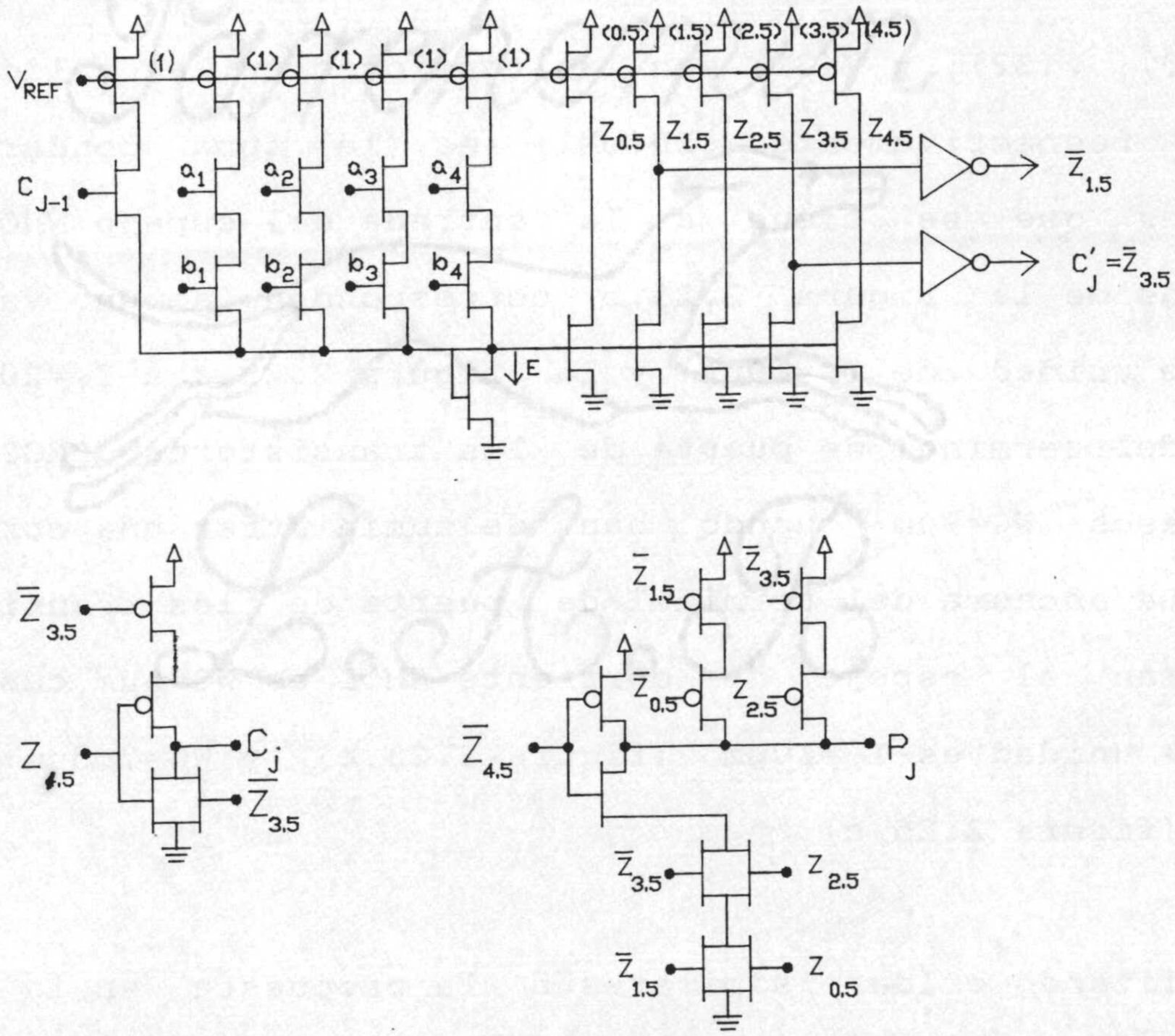


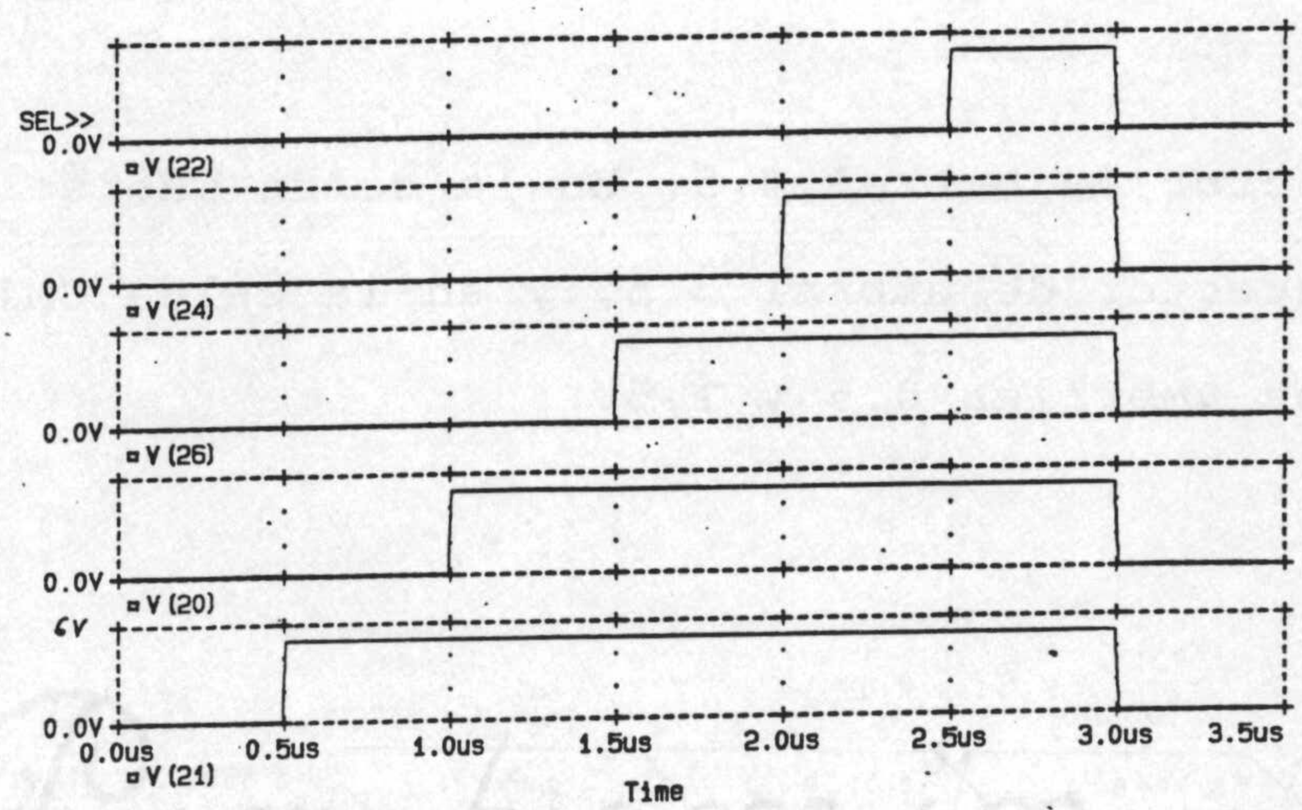
FIGURA 2.24: Realización CMOS estática de la celda de la figura 2.23.

En la figura 2.25 se muestran los resultados de un análisis transitorio con SPICE del circuito de la figura 2.24

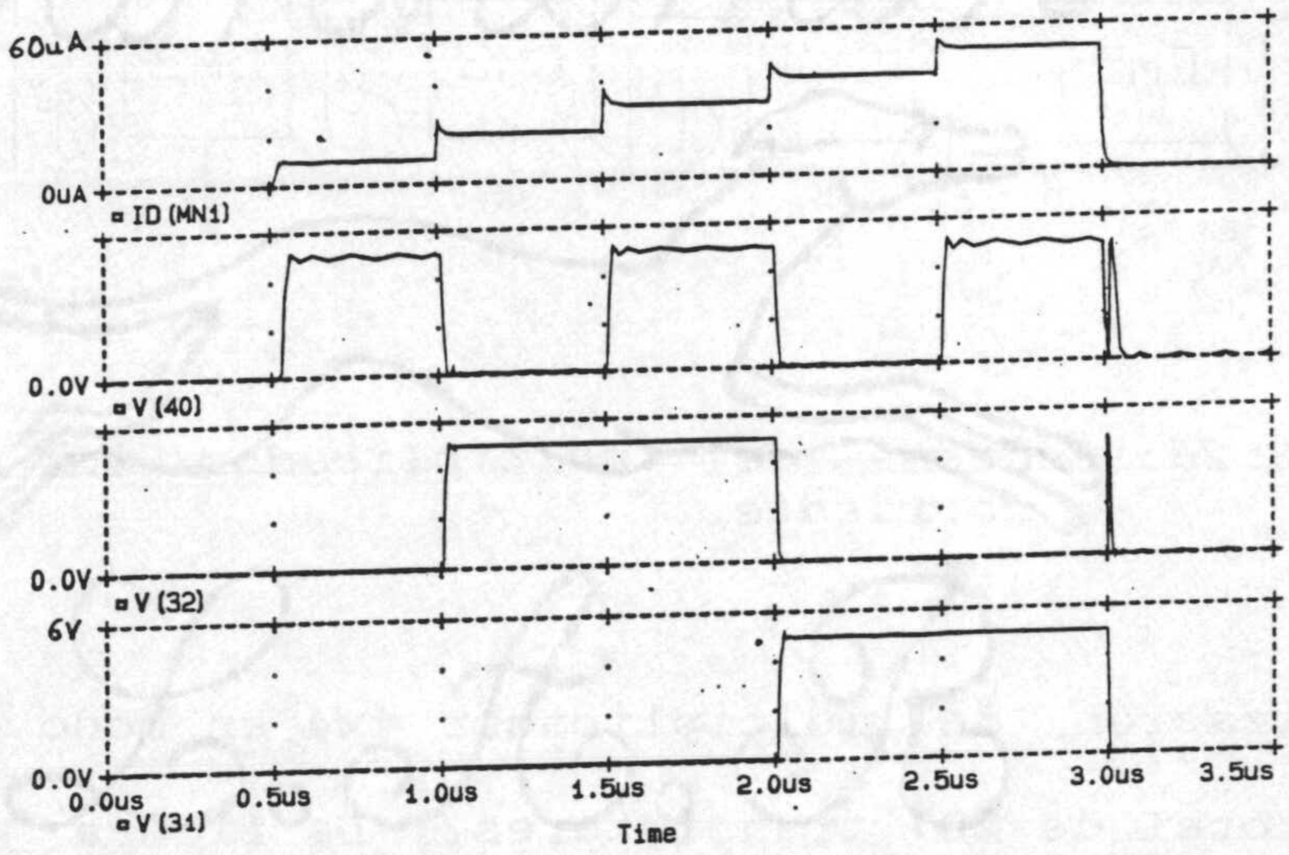
suponiendo que la longitud de puerta de los transistores NMOS y PMOS que funcionan en modo de corriente es de LG=6µm (es decir, longitudes de canal de 4.5µm y 4.4µm respectivamente). En la simulación, las entradas b_i siempre están activas y en el resto de las entradas se aplican las señales de la figura 2.25.a.

V(40), V(32), y V(31) son las tensiones en las salidas P_j, C_j y C_j' respectivamente. ID(MN1) es la suma ponderada de corrientes que se tiene a la entrada del espejo NMOS. Los resultados de la figura 2.25.b corresponden a un valor de corriente unidad de I_0=10µA y la figura 2.25.c a I_0=20µA. La anchura del terminal de puerta de los transistores PMOS es en ambos casos WG=9µm cuando han de suministrar una corriente unidad. La anchura del terminal de puerta de los transistores que forman el espejo de corriente NMOS es WG=6µm cuando la corriente unidad es I_0=10µm (figura 2.25.b) y WG=7.5µm cuando I_0=20µm (figura 2.25.c).

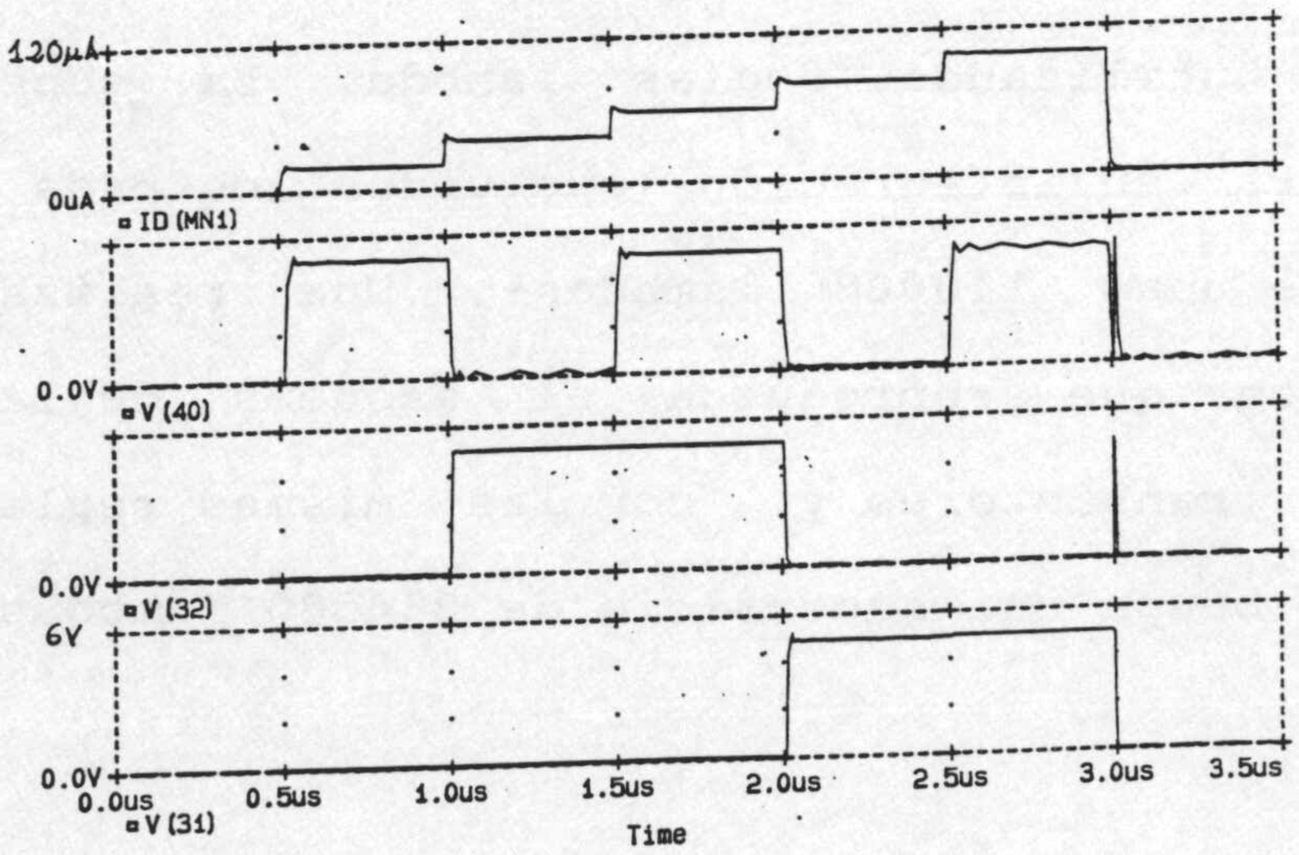
Utilizando celdas similares a la propuesta en la figura 2.23 puede construirse un multiplicador binario 4x4 en la forma indicada en la figura 2.26. De las celdas CMi indicadas en este esquema, la CM3 tiene la realización mostrada en la figura 2.24, y el resto de ellas admiten realizaciones más simples. La celda CM4 se diferencia de la CM3 en una de sus entradas: una pareja de entradas (a_i,b_i) se sustituye por una entrada adicional de acarreo, lo cual implica una realización con un transistor menos. En las celdas CM2 y CM5 la suma ponderada de corrientes toma como máximo el valor 4.I_0; en ellas prescinde



(a)



(b)



(c)

FIGURA 2.25: Respuesta transitoria del circuito de la figura 2.24. (a) Entradas. (b) Excitación y salidas para I_0 =10μA. (c) Excitación y salidas para I_0 =20μA.

pues del detector de umbral 4.5. En la celda CM6 se prescinde además del detector de umbral 3.5, y en la celda CM1 sólo deben detectarse los umbrales 0.5 y 1.5.

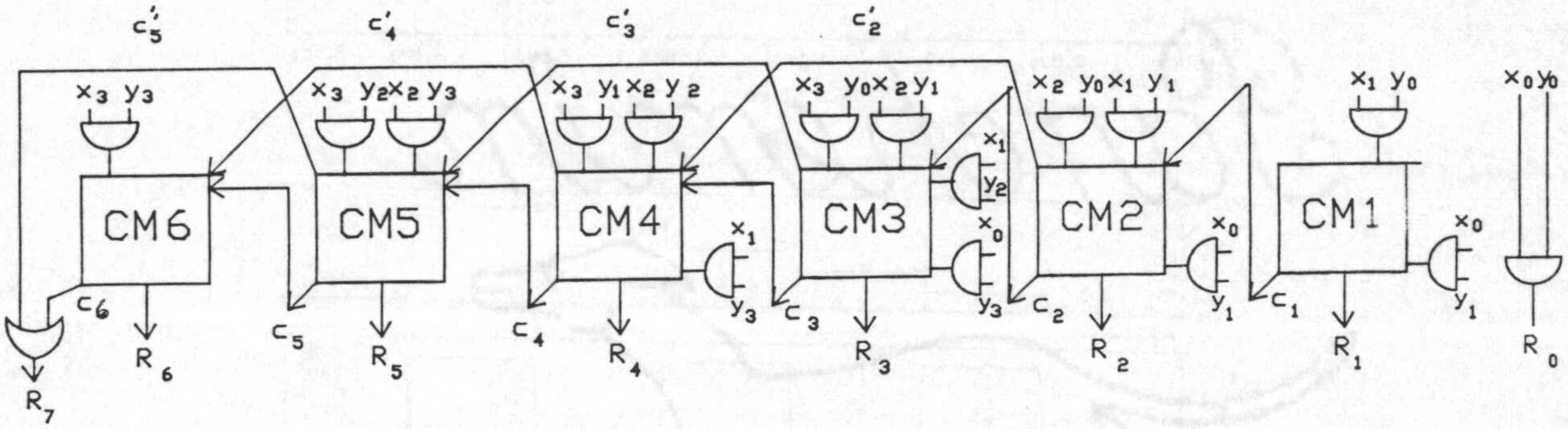


FIGURA 2.26: Esquema del multiplicador 4x4 en modo de corriente.

La realización del multiplicador 4x4 en modo de corriente requiere un total de 207 transistores. La figura 2.27 muestra una posible descripción gráfica de las máscaras del multiplicador diseñada utilizando reglas lambda. La superficie total necesaria para su integración (excluidos los pads de entrada/-salida) es de unas 110000 lambdas^2. Una realización binaria CMOS estática que reproduzca el esquema de la figura 2.22 requiere 496 transistores y, con las mismas reglas de diseño geométricas, ocupa una superficie de 264500 lambdas^2 [187].

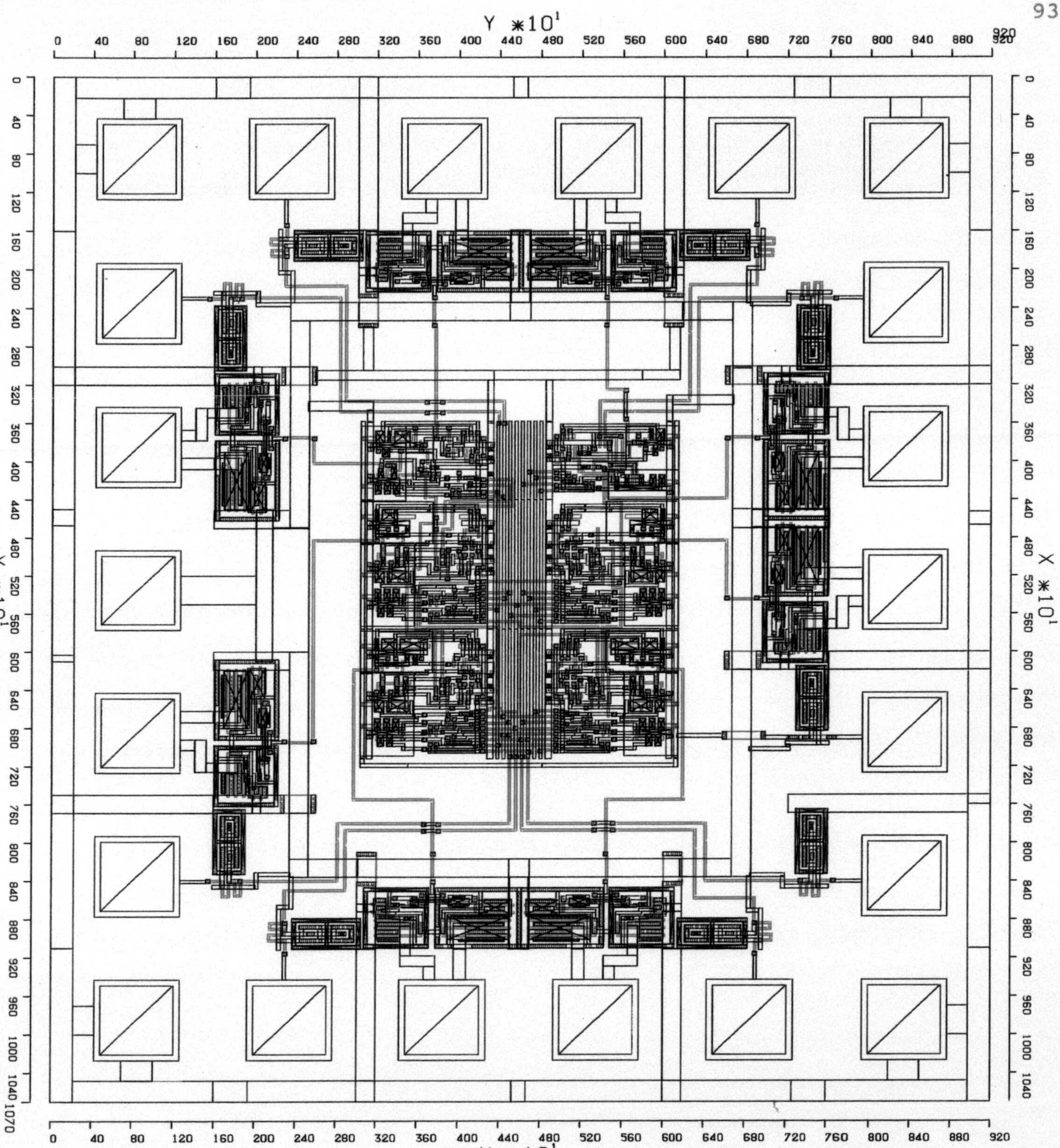


FIGURA 2.27: Descripción gráfica de las máscaras del multiplicador 4x4 en modo de corriente, diseñada con las reglas lambda del proceso CMOS de la U.C.L.

2.5 Conclusión.

En este capítulo hemos realizado un análisis de los circuitos básicos CMOS en modo de corriente, particularizando en aquellas características relacionadas con su uso en la construcción de sistemas multivaluados y multiumbrales.

Partiendo de modelos simplificados de dispositivos hemos obtenido expresiones que caracterizan el comportamiento estático y dinámico de este tipo de circuitos. Para un determinado proceso CMOS, la corriente máxima de funcionamiento y el error debido a los espejos de corriente vienen determinados por las relaciones de tamaño de los transistores utilizados. Los niveles de tensión de salida de un detector de umbral han sido igualmente caracterizados en términos del tamaño de los transistores que lo forman y de la corriente unidad utilizada.

Los resultados obtenidos han sido comprobados mediante simulación con SPICE, utilizando para ello parámetros de modelo de dispositivos de un proceso CMOS concreto.

Como consecuencia del análisis realizado podemos concretar las siguientes reglas, aplicables al diseño de circuitos integrados CMOS en modo de corriente:

- Fijado el valor de la corriente unidad y el número máximo de niveles lógicos, la longitud de canal mínima de los dispositivos MOS debe elegirse de manera que los

errores producidos por los espejos de corriente no deterioren el funcionamiento del circuito; es decir, que quede garantizada la discriminación correcta de niveles de corriente por parte de los detectores de umbrales. Para ello pueden utilizarse las expresiones (2.14), (2.15), (2.19), (2.20), (2.27), y (2.30).

-La anchura de canal de los transistores conectados como espejos de corriente debe calcularse de manera que su funcionamiento sea correcto cuando la corriente correspon-da al nivel lógico máximo (expresiones (2.10) a (2.12)). Para calcular la anchura de canal de los transistores que forman detectores de umbral, deben además tenerse en cuenta los niveles de tensión de salida requeridos (expresiones (2.28)).

-El valor de corriente que representa el nivel lógico unidad es el principal factor determinante de las caracte-rísticas dinámicas y de consumo de potencia del circuito. Los tiempos de transición máximos de los espejos de corriente y de los detectores de umbrales pueden calcular-se en función de dicha corriente unidad mediante las expresiones (2.39) a (2.42). Aunque el fan-out de los detectores de umbral no está limitado a la unidad, para optimizar la velocidad del circuito interesa que cada detector de umbral tenga una capacidad de carga mínima; por lo tanto, en general conviene utilizar inversores CMOS para complementar sus salidas cuando estas deban

actuar sobre varias puertas.

En la sección 2.4 hemos presentado esquemas de puertas umbral y multiumbral CMOS en modo de corriente construidas con los elementos de circuito tratados en las secciones anteriores. Mediante puertas de este tipo, combinadas con circuitos lógicos CMOS binarios hemos diseñado un multiplicador binario en modo de corriente, que ha sido incluido en el MPC-88. La superficie necesaria para integrar este multiplicador es inferior al 50% de la ocupada por un multiplicador CMOS binario diseñado con las mismas reglas geométricas.

Los circuitos básicos analizados se utilizarán en los siguientes capítulos como elementos de síntesis de sistemas multivaluados en modo de corriente.

CAPITULO 3

MATRICES LOGICAS PROGRAMABLES MULTIVALUADAS

En este capítulo se presenta una introducción a la síntesis de sistemas multivaluados mediante matrices lógicas programables.

En la sección 3.1 se hace una revisión de las estructuras básicas y técnicas de optimización de PLAs binarias.

La sección 3.2 resume los distintos esquemas previamente propuestos para la síntesis de funciones multivaluadas mediante PLAs. Algunos de estos esquemas se basan en la utilización de PLAs binarias, mientras que otros se basan en estructuras de PLA propiamente multivaluadas.

En la sección 3.3 se comentan algunos métodos de minimización de funciones multivaluadas apropiados para la síntesis

de estas mediante PLAs.

El capítulo finaliza con una conclusión (sección 3.4).

3.1 Introducción. PLAs binarias.

En el diseño de sistemas digitales en general, y en el de circuitos integrados digitales en particular, se tiende a utilizar módulos de circuito con estructura regular, los cuales posibilitan efectuar el diseño de forma automática, así como prevenir y detectar fallos en la realización física final.

Para la síntesis de funciones lógicas en general, se adopta como solución el uso de módulos lógicos universales. El tipo de módulos lógicos utilizados depende principalmente del número de funciones lógicas a sintetizar y del número de variables de que dependan.

Para realizar funciones binarias aisladas de hasta 6 variables es adecuado utilizar multiplexores. Si el número de variables es mayor y/o se trata de una síntesis multifuncional, puede resultar más económico utilizar como módulo lógico una ROM. El tamaño de la ROM practicamente se duplica con cada variable lógica adicional; motivo por el cual, para un elevado número de variables es en general preferible el uso de un módulo lógico similar a una ROM, pero sin el problema de creci-

miento de tamaño anteriormente referido. Este módulo se denomina *matriz lógica programable* (PLA) [45].

Una PLA tiene una estructura regular adecuada para la síntesis de un conjunto de funciones lógicas que dependan de un gran número de variables. Normalmente, con una PLA se realiza una síntesis en dos niveles. La estructura básica de una PLA binaria se muestra en la Figura 3.1. Está constituida por dos submatrices o planos: producto y suma. En el plano producto se obtienen productos lógicos de las variables de entrada (complementadas o sin complementar). Estos términos producto se combinan en el plano suma para realizar las expresiones booleanas que definen el conjunto de funciones de salida.

Aunque se utilice la denominación de planos producto y suma, las operaciones lógicas realizadas en estos planos no están limitadas al producto lógico (AND) y suma lógica (OR), sino que pueden adoptar, dependiendo de la tecnología, formas equivalentes más adecuadas para su integración (NAND/NAND o NOR/NOR por ejemplo).

El calificativo *programable* en el término PLA hace referencia al hecho de que las funciones lógicas concretas que sintetiza la PLA dependen de modificaciones realizadas, sobre una misma estructura de base, en alguna de las etapas del proceso de su fabricación o con posterioridad al mismo. Las modificaciones que definen la programación de la PLA pueden ser el establecimiento o eliminación de conexiones o la definición

de transistores en puntos concretos de los planos producto y suma.

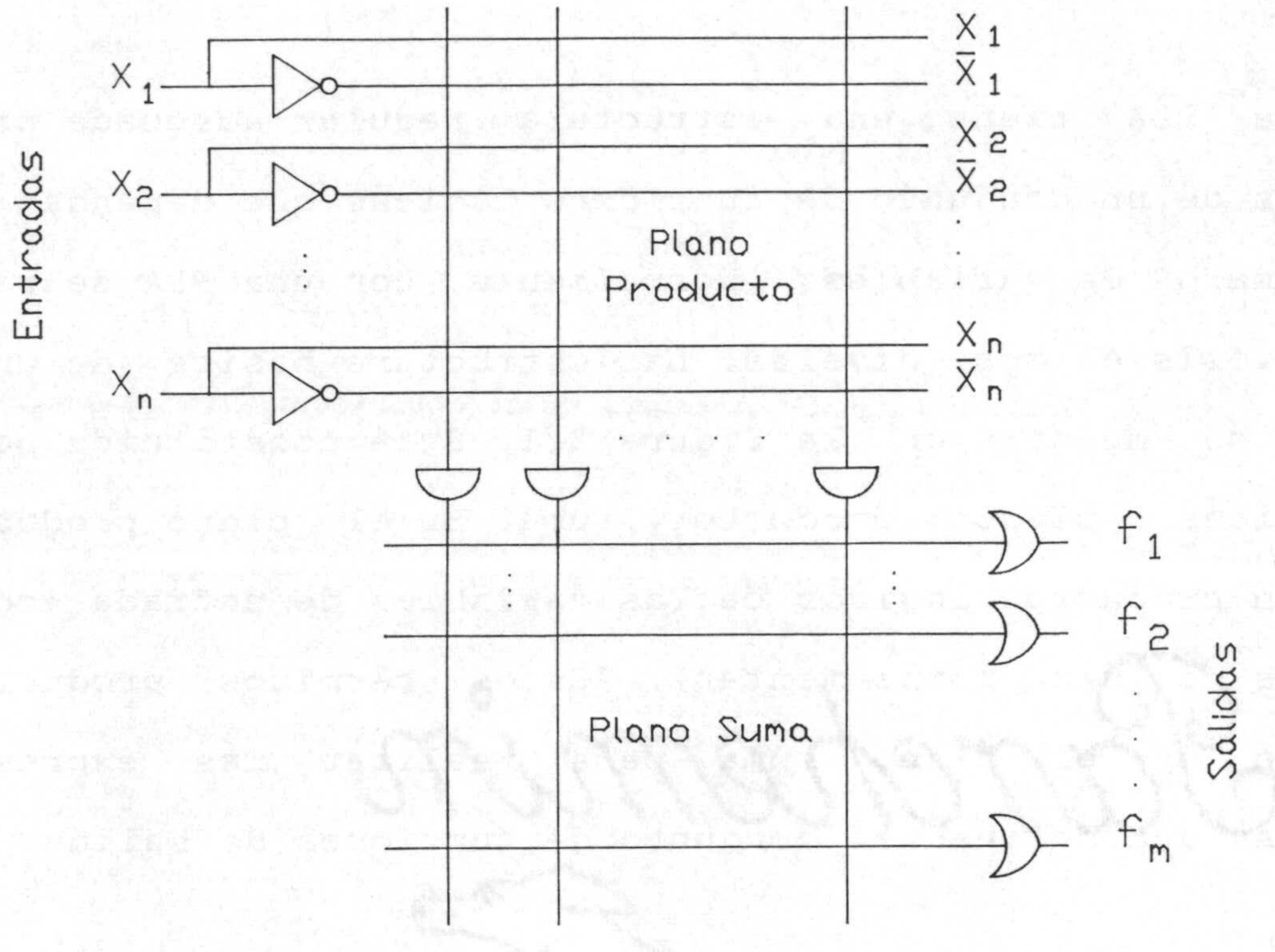


FIGURA 3.1 Estructura básica de una PLA binaria

La programación del plano producto define el conjunto concreto de términos producto necesarios, y la programación del plano suma establece qué términos producto contribuyen a cada una de las funciones de salida; existiendo, por regla general, la posibilidad de términos producto compartidos por varias de las funciones.

Una PLA binaria puede incorporar la posibilidad de realimentar algunas de sus salidas a través de celdas de registro añadidas a la estructura básica. La estructura resultante, conocida con el nombre de *secuenciador lógico programable*, se emplea como elemento programable para la

síntesis completa de sistemas secuenciales.

Existen diversas variantes de matrices lógicas programables disponibles como circuitos integrados LSI, las cuales se emplean para sustituir conjuntos de circuitos SSI/MSI en la síntesis de sistemas combinacionales y secuenciales. En este tipo de PLAs la programación puede efectuarla el fabricante (mediante máscara), o bién, utilizando dispositivos de programación especiales, puede llevarla a cabo el usuario (programación mediante campo). La programación mediante campo consiste en fundir o eliminar selectivamente conexiones preestablecidas en los planos producto y suma y, en su caso, en los lazos de realimentación existentes.

El mayor interés de las PLAs radica en su uso como módulos de circuitos integrados VLSI más complejos, tales como controladores digitales y microprocesadores. Por ejemplo, microprocesadores de 32 bits aparecidos estos últimos años, como el AT&T-WE32100, MOTOROLA-MC68020, el INTEL-80386, o el IBM-Micro/370, incluyen varias PLAs en sus secciones de control [29, 54, 113].

Una PLA que sintetice un conjunto dado de funciones lógicas puede ocupar mayor superficie de silicio que una realización utilizando puertas independientes ("random logic"), diseñada y optimizada para tal conjunto de funciones. Sin embargo, suelen preferirse soluciones basadas en PLAs fundamentalmente por dos motivos:

1.- Su diseño es realizable de forma automática mediante computadora.

2.- Son fácilmente comprobables debido a su estructura regular.

Estas características son de especial importancia en circuitos integrados VLSI, dado que los costes asociados con su diseño y desarrollo superan a menudo los correspondientes a la fase de producción del circuito durante todo el tiempo en que éste va a ser comercializado [68].

Por otra parte, cada vez hay una mayor tendencia a incorporar en el propio circuito integrado una parte destinada a facilitar su comprobación (test) o incluso para la autocomprobación (self-test) [123]. Los circuitos de autocomprobación ocupan un cierto porcentaje de la superficie total del chip, pero permiten la detección de fallos de operación, al final de su proceso de fabricación, y durante el tiempo de funcionamiento del mismo.

Las PLAs, al igual que otras estructuras regulares, como las ROMs, son muy apropiadas para la incorporación de circuitería de autocomprobación. Este aspecto será tratado con más detalle en el capítulo 5.

3.1.1 Optimización del tamaño de una PLA.

Una de las etapas más importantes en el diseño de una PLA integrada es la optimización de la superficie que ocupa. Dado un conjunto de m funciones lógicas que dependan de n variables, la superficie ocupada por el plano suma de la PLA que las sintetice es proporcional a k.m, siendo k el número de términos producto necesarios para la síntesis. La superficie ocupada por el plano producto es proporcional a 2.n.k. Las técnicas de optimización utilizadas para reducir la superficie total de la PLA tienen como principal objetivo minimizar el número k de términos producto; con ello, además de una optimización del tamaño de la PLA (el cual es proporcional a k), se obtiene en general una reducción del tiempo de retardo de la misma.

Partiendo de las expresiones canónicas de las funciones a realizar, la primera optimización posible es la obtención del conjunto mínimo de implicantes primos que constituyan una cobertura de las funciones. Las técnicas clásicas de minimización booleana [124, 153] sólo son aplicables en la práctica cuando se trata de un número reducido de variables de entrada, debido a que el problema de obtención de la cobertura mínima es NP-completo. La complejidad de cálculo asociada a la minimización absoluta de funciones lógicas ha motivado la investigación de algoritmos heurísticos alternativos, los cuales llevan en general a soluciones próximas a la mínima, y permiten, dada la reducción drástica en el tiempo de cómputo para obtener la solución, tratar simultáneamente un gran número de funciones

lógicas que dependan de muchas variables [160].

Otras técnicas utilizadas para la optimización del tamaño de PLAs integradas consisten en el empleo de decodificadores de dos bits en las entradas y en la técnica de plegado ("folding"), las cuales se describen brevemente a continuación.

En la figura 3.1, cada entrada externa se suministra al plano producto complementada y sin complementar. Si en lugar de esto las variables de entrada se agrupan por parejas y se decodifican, la estructura de PLA resultante [45] (figura 3.2), conocida con el nombre de PLA con decodificadores de dos bits, requiere en general entre un 10 y un 20 por ciento menos lineas producto [162] que la PLA tradicional, manteniendose constante el número total de lineas asociadas con las variables de entrada (lineas de bit).

La obtención de la PLA óptima es una tarea más compleja cuando se utilizan en las entradas decodificadores de dos bits en lugar de la solución tradicional (decodificadores de un bit). La forma óptima de abordar el problema es, por el momento, la minimización multivaluada. La cual consiste en considerar cada pareja de variables binarias como una variable 4-valuada, y minimizar la función de entradas multivaluadas y salida binaria resultante [160, 165].

La técnica de plegado consiste en efectuar modificaciones estructurales sobre los planos de la PLA para permitir que dos

líneas de bits o de salida compartan una misma fila, o bién que dos líneas producto compartan la misma columna.

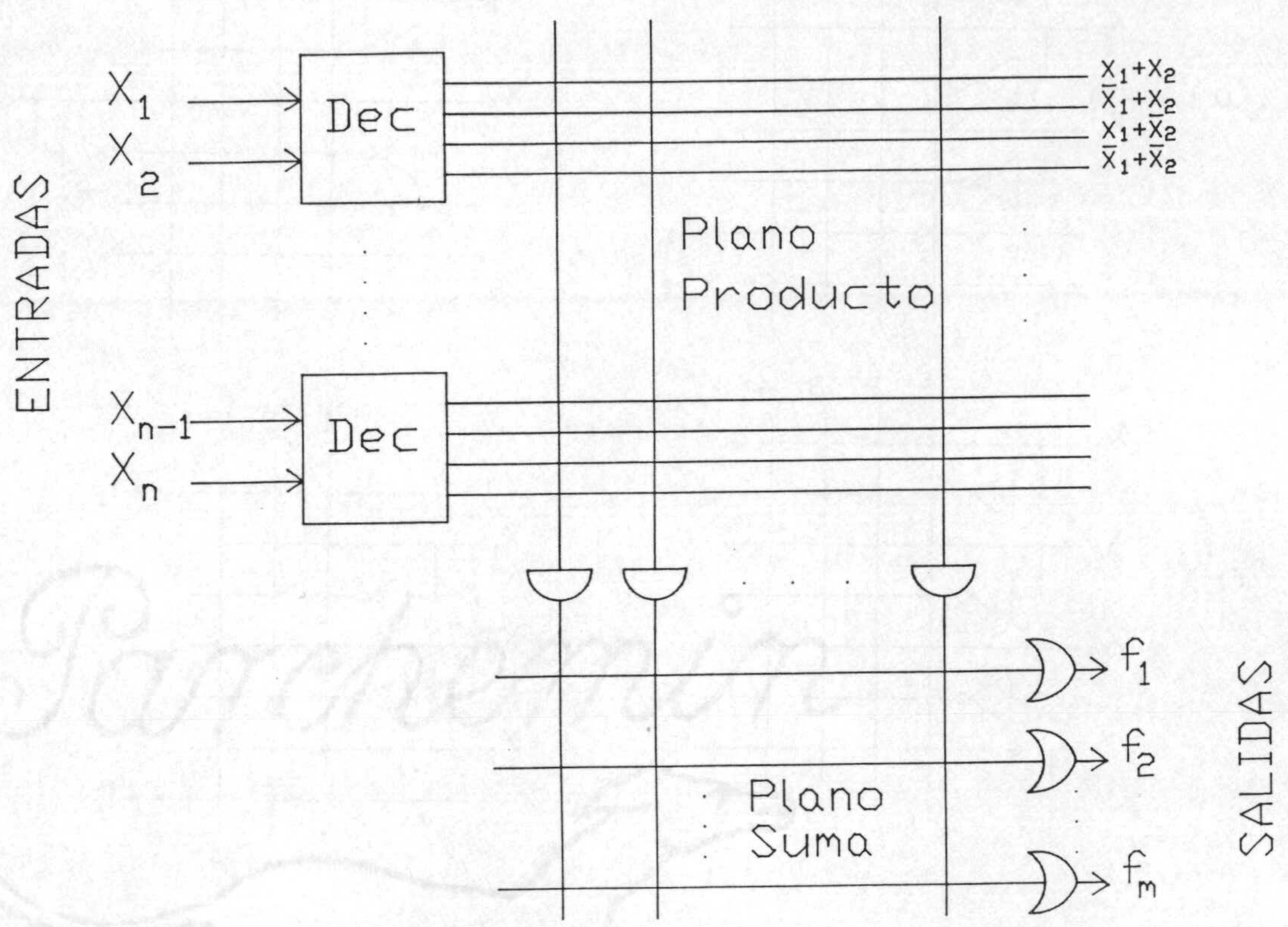


FIGURA 3.2 Estructura de una PLA binaria con decodificadores de dos bits

En la figura 3.3.a se muestra el esquema de una PLA que sintetiza las funciones:

$$f_1 = \bar{x}_1 \bar{x}_3 + x_1 x_2$$

$$f_2 = \bar{x}_1 \bar{x}_2 + x_3$$

Dado que los términos producto a los que contribuyen las líneas de bits de x_2 y x_3 son distintos, puede realizarse la partición

de filas indicada en la figura 3.3.b.

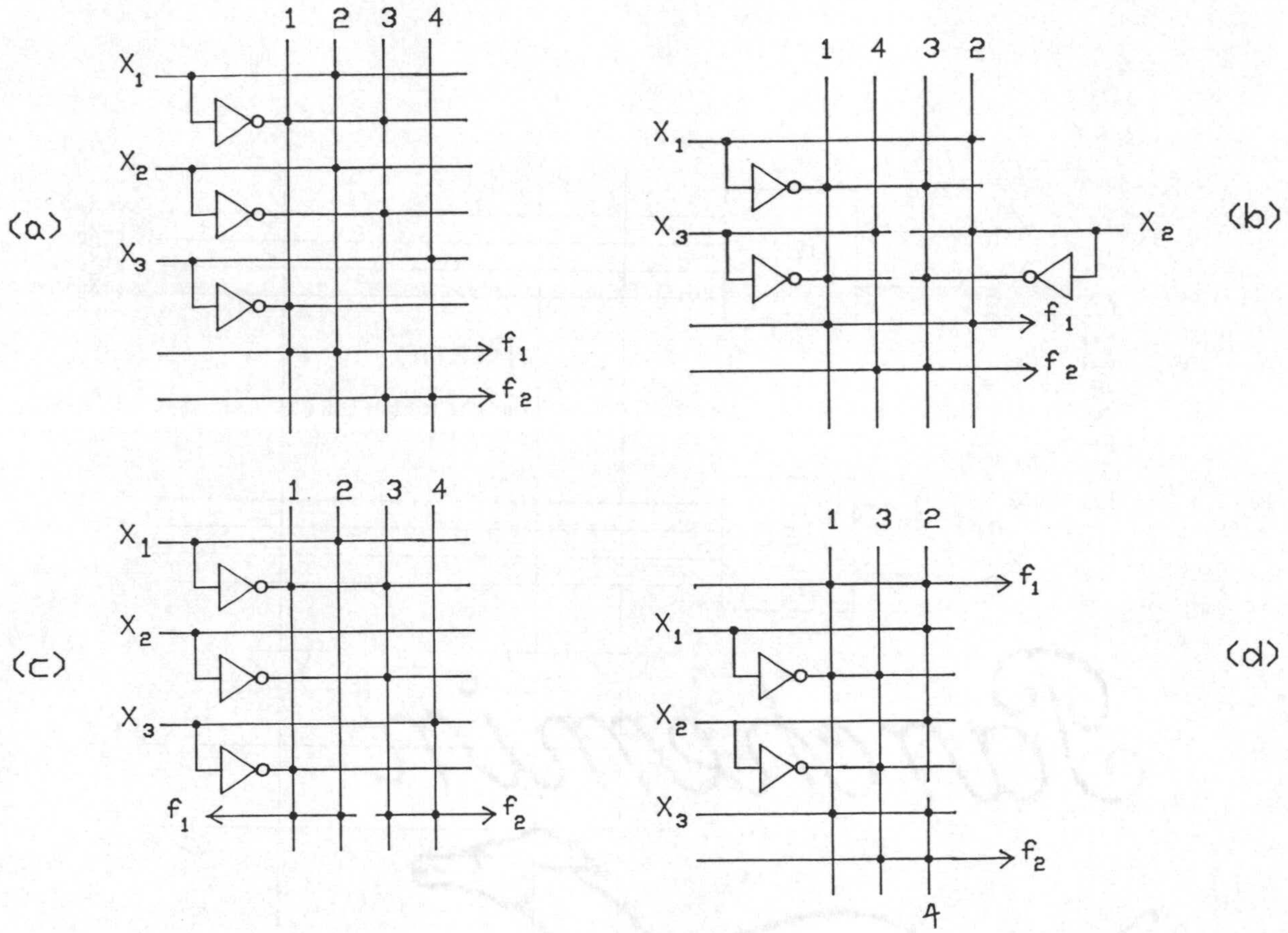


FIGURA 3.3. Ejemplo de plegado de una PLA binaria. (a):PLA original. (b) y (c): Plegado por partición de filas. (d): Plegado por partición de columnas.

Otra opción es efectuar la partición de filas sobre las líneas de salida. Sobre la PLA de la figura 3.3.a puede realizarse la partición indicada en la figura 3.3.c, ya que las dos funciones que se sintetizan no comparten términos producto.

En lugar de una partición de filas, puede llevarse a cabo una partición de columnas como la mostrada en la figura 3.3.d; en este caso dos líneas producto comparten la misma columna de

la PLA.

En general pueden realizarse tanto particiones de filas como de columnas sobre una misma PLA, para lo cual se han desarrollado procedimientos que efectuan el plegado de forma automática (ver [60], por ejemplo).

3.2 Síntesis de funciones multivaluadas mediante PLAs

Al igual que en el diseño de circuitos VLSI binarios, la complejidad de diseño de sistemas multivaluados crece exponencialmente con el número de componentes activos. Además, el diseño de sistemas multivaluados es, por regla general, más complicado que el de sistemas binarios. Dado que el diseño de una PLA integrada puede llegar a realizarse de forma totalmente automática, existe un gran interés por la síntesis de funciones multivaluadas basada en este tipo de circuitos programables.

En esta sección se hace una revisión breve de las posibles alternativas a la síntesis de funciones multivaluadas mediante PLAs. Previamente, con objeto de clarificar la exposición, vamos a recordar una serie de conceptos relacionados con la lógica multivaluada. Las definiciones que se dan a continuación servirán de base para el desarrollo de secciones posteriores.

3.2.1 Definiciones.

Sea R el conjunto de valores lógicos {0,1,..,r-1}, siendo r una constante entera positiva que, salvo que se especifique r=2, consideraremos mayor que 2.

Definición 3.1 (función multivaluada): *Una función r-valuada de n variables r-valuadas* $f(X_1, X_2, .., X_n)$ *es una aplicación del conjunto producto cartesiano* R^n *en R.*

$$f(X_1, X_2, .., X_n): R^n \dashrightarrow R$$

Definición 3.2 (función binaria de variables multivaluadas): *Una función binaria de n variables r-valuadas es una aplicación de* R^n *en el conjunto {0,a}, siendo "a" un elemento de R que cumple:* $0<a\leq(r-1)$.

Definición 3.3 (función literal): *Un literal de una variable r-valuada X es una función binaria de dicha variable definida por:*

$$X^S = \begin{cases} r-1 & si\ X \varepsilon S \\ 0 & si\ X \notin S \end{cases}$$

en donde S es un subconjunto de valores de R.

Definición 3.4 (función literal intervalo): *Un literal intervalo de la variable r-valuada X es una función binaria de S definida por:*

$$^a X^b = \begin{cases} r-1 & si\ a\leq X\leq b \\ 0 \text{ para el resto de los valores de } X. \end{cases}$$

en donde a y b son elementos de R tal que a≤b.

Definición 3.5 (función mínimo): La función mínimo (MIN) de n variables r-valuadas es una función r-valuada que aplica a cada combinación de valores de las n variables el mínimo de dichos valores, es decir:

$$MIN(X_1, X_2, \ldots, X_n) = min.\{X_1, X_2, \ldots, X_n\}$$

$$X_1, X_2, \ldots, X_n \varepsilon R$$

Para representar el operador MIN utilizaremos el símbolo "∩", es decir:

$$MIN(X_1, X_2, \ldots, X_n) = X_1 \cap X_2 \cap \ldots \cap X_n$$

En el caso de variables binarias (r=2) la función mínimo coincide con el producto lógico (AND).

Definición 3.6 (término producto multivaluado): Se denomina término producto de n variables r-valuadas a una expresión que relaciona, mediante el operador MIN, una constante no nula cε{1,2,..,r-1} y literales de las n variables. Para cada variable puede aparecer, en dicha expresión, como máximo un literal:

$$c \cap X_1^{s_1} \cap X_2^{s_2} \cap \ldots \cap X_n^{s_n}$$

$$c, X_i \varepsilon R \; ; \; c \neq 0; \; S_i \subseteq R; \; i \varepsilon \{1,2,..,n\}$$

La constante c recibe el nombre de valor del término producto.

Según las definiciones anteriores, un término producto de n variables r-valuadas es una función binaria de las n variables multivaluadas que tiene como conjunto imagen {0,c}.

Cualquiera de los literales X^{S_i} del término producto que verifique $S_i = R$, puede eliminarse de dicho término.

Atendiendo a la definición del operador MIN, si c=r-1, ésta constante puede eliminarse del término producto.

Definición 3.7 (minterm multivaluado): Denominamos minterm de n variables multivaluadas a cualquier término producto de la forma $c \cap X_1^{s_1} \cap X_2^{s_2} \cap \ldots \cap X_n^{s_n}$ *en el que aparecezcan literales de todas las n variables y cada* S_i *incluya un sólo elemento de R. Es decir, una expresión en la forma:*

$$c \cap X_1^{a_1} \cap X_2^{a_2} \cap \ldots \cap X_n^{a_n}$$

$$a_i \varepsilon R, \quad i=1,2,..,n$$

La constante c recibe el nombre de valor del minterm.
Las constantes a_i se denominan coordenadas del minterm, y el conjunto de coordenadas $\{a_1, a_2, .., a_n\}$ recibe el nombre de posición del minterm.

Definición 3.8 (relación de inclusión entre términos producto): Dados dos términos producto de n variables multivaluadas :

$$P_1 = c_1 \cap X_1^{s_{11}} \cap .. \cap X_n^{s_{1n}}$$

$$P_2 = c_2 \cap X_1^{s_{21}} \cap .. \cap X_n^{s_{2n}}$$

se dice que P_1 *está incluido en* P_2 $(P_1 \subseteq P_2)$ *si se cumplen las dos condiciones:*

$$S_{1\,i} \subseteq S_{2\,i} \; ; \; i=1,2,..n$$

$$c_1 \leq c_2$$

Definición 3.9 (implicante de una función multivaluada): Dada una función multivaluada de n variables $f(X_1, ..X_n)$, *un término producto* $I=c \cap X_1^{s1} \cap .. \cap X_n^{sn}$ *se dice que es un implicante de f si para cada combinación de valores* $(a_1, a_2, .., a_n)$ *de las n variables se cumple:* $c \leq f(a_1, a_2, .., a_n)$.

Definición 3.10 (implicante primo): Un implicante I_1 *de la función f se dice que es primo si no existe otro implicante* I_2 *de f tal que* $I_1 \leq I_2$.

La síntesis de funciones multivaluadas con PLAs se basa en las expresiones de dichas funciones mediante ecuaciones que relacionan términos producto (implicantes) por medio de algún otro operador. Puede demostrarse que cada uno de los operadores (a), (b), y (c) que se indican a continuación, junto con el operador MIN y las funciones literal, constituyen conjuntos funcionalmente completos.

(a).- Operador máximo (MAX):

$$X \cup Y = \max\{X, Y\} \; ; \; X, Y \varepsilon R$$

(b).- Operador suma módulo r (SUM):

$$X \oplus Y = (X+Y)_{\text{módulo } r} \; ; \; X, Y \varepsilon R$$

(c).- Operador suma truncada (TSUM):

$$X \boxplus Y = \min\{X+Y, r-1\} \; ; \; X,Y \varepsilon R$$

en donde el signo "+" representa la suma aritmética.

Si X e Y son, por ejemplo, variables 4-valuadas, las operaciones anteriores dan como resultado los valores indicados en las siguientes tablas:

MAX:

X \ Y	0	1	2	3
0	0	1	2	3
1	1	1	2	3
2	2	2	2	3
3	3	3	3	3

SUM:

X \ Y	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

TSUM:

X \ Y	0	1	2	3
0	0	1	2	3
1	1	2	3	3
2	2	3	3	3
3	3	3	3	3

En general, dependiendo de cual sea la función multivaluada, el número mínimo de términos producto necesarios para expresarla es diferente según se utilice el operador máximo, suma, o el suma truncada [9]. Por otra parte, siempre se consiguen expresiones más simples si los literales utilizados no son sólo literales intervalo.

Como ejemplo, la función 4-valuada de dos variables dada por la tabla 3.1 puede expresarse, utilizando los operadores mencionados, como:

(a).-Con el operador MAX:

$$f(X,Y)=(2 \cap X^{0,1} \cap Y^{0,1}) \cup (2 \cap X^{0,2} \cap Y^{1,3}) \cup (3 \cap X^{1} \cap Y^{1}) \cup (1 \cap X^{1} \cap Y^{3})$$

(b).-Con el operador SUM:

$$f(X,Y)=(2 \cap X^{0,1} \cap Y^{0,1,3}) \oplus (2 \cap X^{2} \cap Y^{1,3}) \oplus (1 \cap X^{1} \cap Y^{1}) \oplus (3 \cap X^{1} \cap Y^{3})$$

(c).-Con el operador TSUM:

$$f(X,Y)=(2 \cap X^{0,1} \cap Y^{0,1}) \boxplus (2 \cap X^{1,2} \cap Y^{1}) \boxplus (1 \cap X^{0,1,2} \cap Y^{3}) \boxplus (1 \cap X^{0,2} \cap Y^{3})$$

TABLA 3.1: Ejemplo de función 4-valuada de 2 variables

		Y			
		0	1	2	3
X	0	2	2	0	2
	1	2	3	0	1
	2	0	2	0	2
	3	0	0	0	0

Una vez dadas las definiciones anteriores procedemos a la revisión de las estructuras y técnicas de optimización de PLAs multivaluadas que se encuentran en la bibliografía.

3.2.2 Estructuras de PLAs multivaluadas previamente propuestas.

Desde hace unos años, existe un considerable interés por la utilización de PLAs para la síntesis de funciones multivaluadas [9, 10, 33, 34, 104, 109, 147, 148, 164, 181, 182]. Se

han propuesto algunas posibles estructuras de PLAs [104, 109, 164, 182], de las cuales una [12, 104] ha sido realizada físicamente en tecnología CCD.

Tirumalai y Butler [182] proponen un esquema de PLA 4-valuada realizable en tecnología CCD y basada en los operadores MIN y MAX. La figura 3.4 representa la estructura de la PLA propuesta para el caso de una única función 4-valuada. En el plano producto se generan por separado los términos correspondientes a los distintos valores no nulos de la función; existiendo pues tres subplanos, cada uno de los cuales opera en forma binaria. En realidad se trata de una realización con tres niveles: AND/OR/MAX. Es decir, mínimo y máximo binarios en cada uno de los subplanos, cuyas salidas ponderadas se combinan mediante la operación MAX tetravaluada.

Las entradas al plano producto proceden de unos módulos (UICPC) que generan todos los literales de cada variable (2^r literales por variable r-valuada). El plano producto está diseñado de manera que para transportar las señales de los 2^r literales sólo se requieren $2^{r-1}-1$ lineas.

Si k es el número de terminos producto necesarios, el tamaño del plano producto de la PLA mencionada puede expresarse como:

$$n.k.(2^{r-1}-1)$$

siendo n el número de variables de la función.

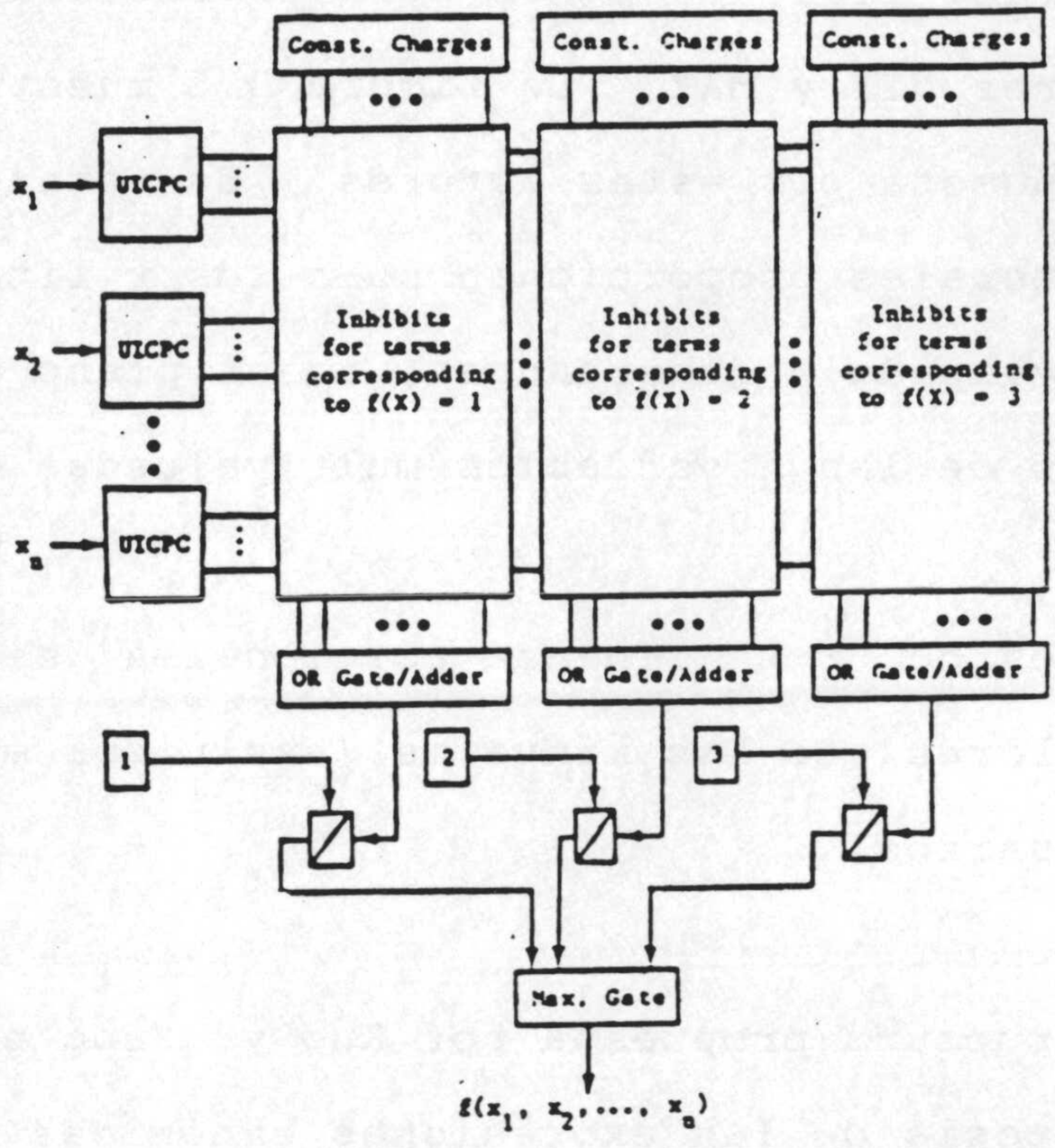


FIGURA 3.4. Estructura de PLA propuesta por Tirumalai y Butler [182].

Manteniendo la misma estructura básica, el plano MAX de salida puede sustituirse por planos SUM o TSUM. Dueck y Miller [33] presentan posibles realizaciones de estos planos en tecnología CCD.

Analizando el esquema de PLA propuesta por Tirumalai y Butler encontramos una gran limitación; las señales procedentes de los generadores de literales son cargas electricas, motivo por el cual cada literal solo puede contribuir a un término producto. Para contribuir a varios términos deberían generarse las correspondientes réplicas de carga en los puntos de cruce del plano producto.

Kuo y Fang [109] proponen otra estructura de PLA basada en los operadores MIN y MAX. La figura 3.5 muestra un esquema de la PLA propuesta por estos autores. En este caso, los generadores de literales proporcionan sólo los r literales X^i de cada variable r-valuada; generandose, en el plano MIN, términos que son minterms de las n variables multivaluadas de entrada.

A través de r-2 líneas adicionales se suministran los posibles valores de los minterms (excluidos el 0 y el r-1, que no son necesarios).

La estructura propuesta por Kuo y Fang es pues apropiada para la síntesis de las expresiones canónicas de las funciones, es decir, las expresiones de estas como máximo de minterms multivaluados. Para funciones de n variables, el tamaño del plano MIN de esta PLA es:

$$k.(n.r+r-2)$$

siendo k en este caso el número de minterms distintos que aparecen en las expresiones canónicas de las funciones.

La principal ventaja de la PLA de Kuo y Fang, frente a la ideada por Tirumalai y Butler, es la reducción del número de lineas lireral del plano MIN. El mayor inconveniente es la imposibilidad de reducir el número k de términos producto mediante alguna técnica de simplificación. Este inconveniente puede superarse eligiendo otro conjunto de literales para cada una de las variables, en lugar de los literales X^1, $i \varepsilon R$.

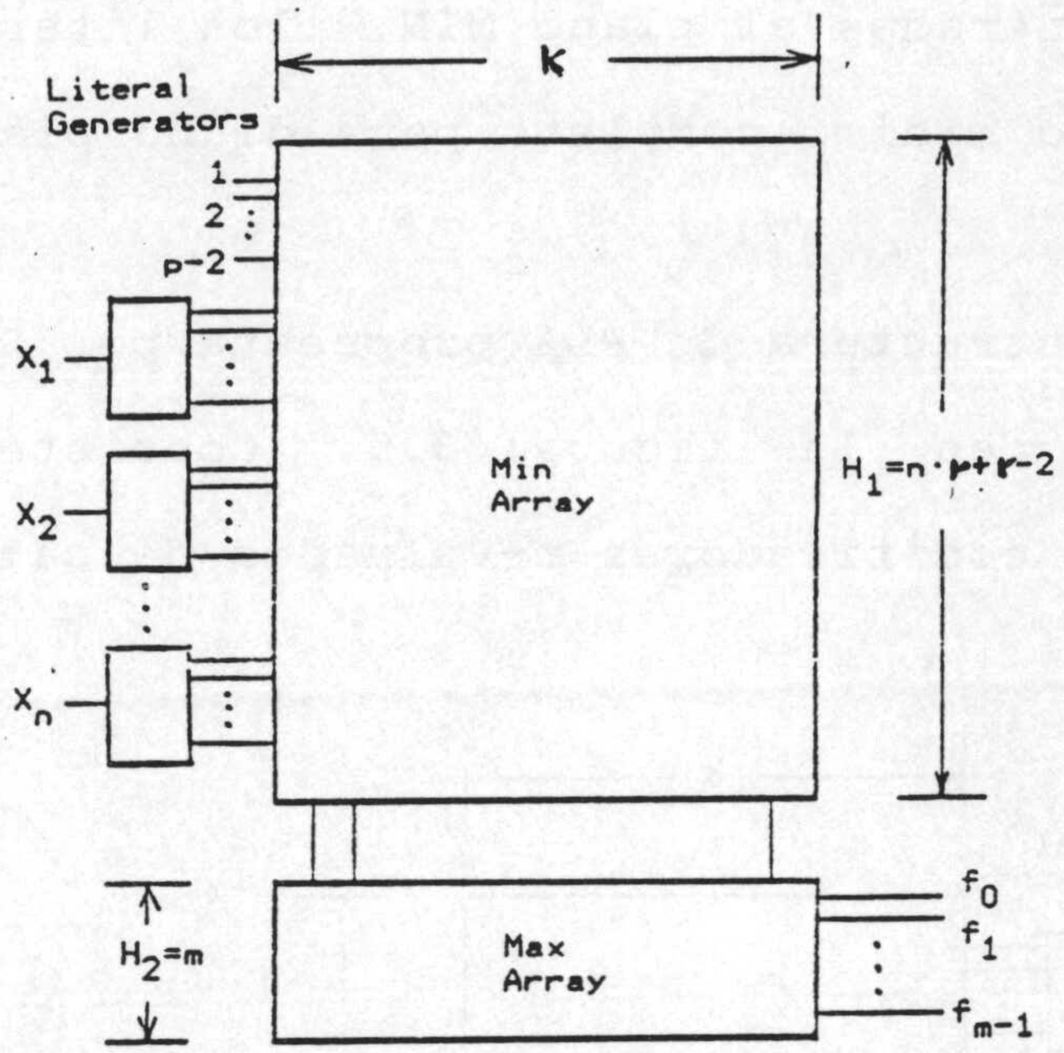


FIGURA 3.5 Estructura de PLA propuesta por Kuo y Fang [109].

Está demostrado [164] que el conjunto de r literales de la variable r-valuada X:

$$\{X^{S_i}\} \; ; \; S_i = R-\{i\}, \; i=0,1,..,r-1 \qquad (3.1)$$

es un conjunto mínimo completo de literales para un plano MIN; es decir, cualquier otro literal de X puede obtenerse relacionando varios del conjunto anterior mediante el operador MIN, y no existe un conjunto con menor número de literales que tenga esta propiedad.

Sasao [164] propone dos estructuras de PLA para la síntе-

sis de funciones multivaluadas. La primera de ellas (PLA del tipo 1) coincide con la de Kuo y Fang (figura 3.5) pero utilitando, como entradas al plano MIN, los literales pertenecientes al conjunto mínimo completo para dicho plano.

La segunda estructura de PLA propuesta por Sasao (PLA del tipo 2) se muestra en la figura 3.6. Consiste en una PLA binaria seguida de codificadores r-valuados de salida. Las

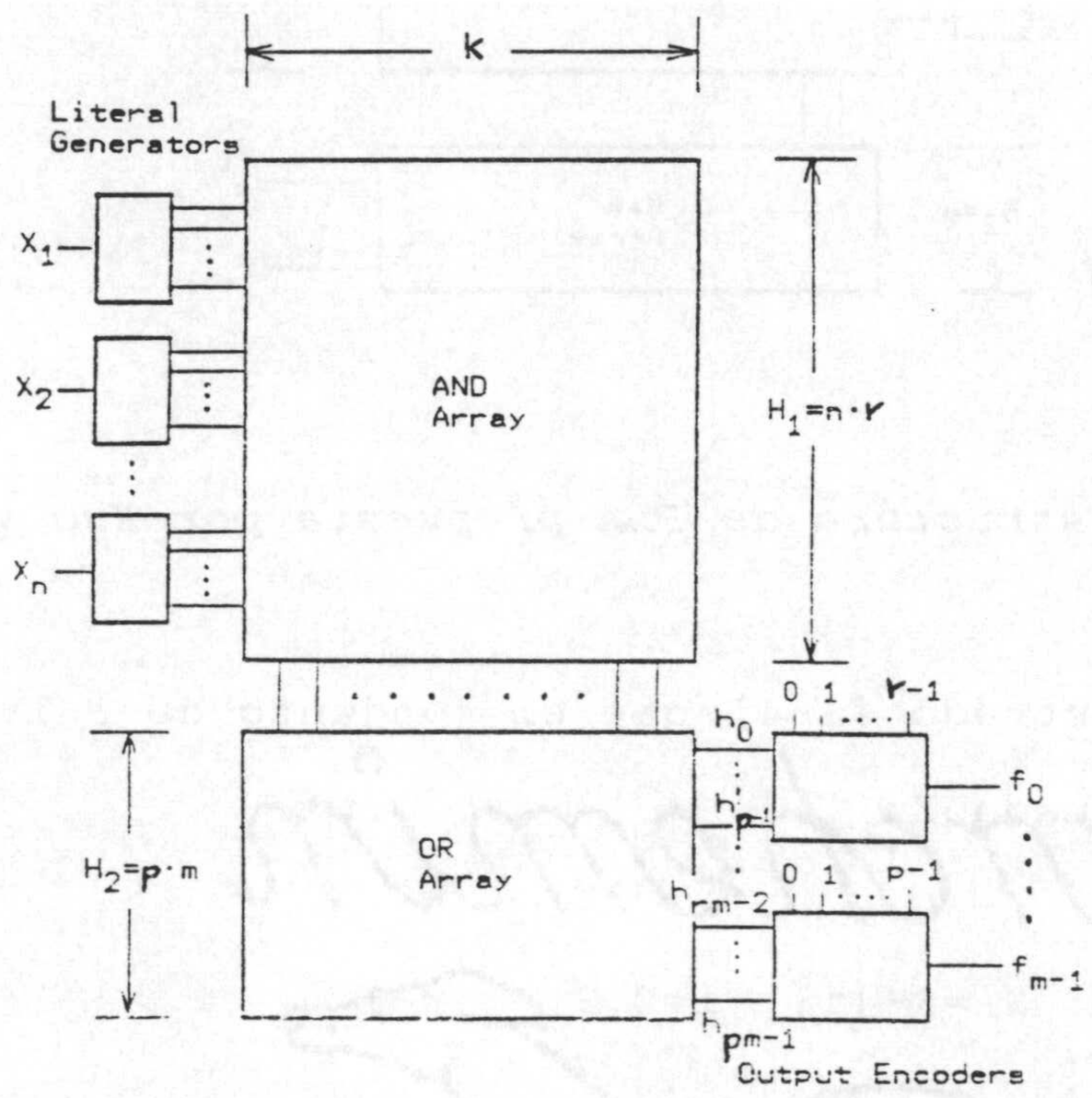


FIGURA 3.6 Estructura de la PLA del tipo 2 propuesta por Sasao [164].

entradas al plano producto binario son, si se trata de una PLA binaria AND/OR o NAND/NAND, los conjuntos mínimos completos de literales definidos por las expresiones (3.1). Si la PLA binaria está formada por dos planos NOR, el conjunto mínimo completo de literales de cada variable r-valuada X es:

$$\{X^i\} \; ; \; i \varepsilon R \qquad (3.2)$$

Cada columna de la PLA binaria de la figura 3.6 realiza productos lógicos en la forma:

$$X_1^{s_1} \cap X_2^{s_2} \cap \ldots \cap X_n^{s_n} \; ; \; S_i \leq R \; ; i:1,2,\ldots,n$$

En el plano suma, estos productos se combinan para sintetizar las siguientes funciones binarias:

$$h_{l,j} = \bigcup_q (X_1^{s_{1q}} \cap X_2^{s_{2q}} \cap \ldots \cap X_n^{s_{nq}}) \; ; \quad l=1,2,\ldots,m-1 \quad (3.3)$$

$$j=0,1,\ldots,p-1$$

$$S_{iq} \leq R$$

en donde p es el entero más pequeño mayor o igual que $\lg_2 r$:

$$p=[\lg_2 r]$$

La función multivaluada f_l se obtiene codificando las funciones binarias $h_{l,j}$ (j=0,1,..,p-1) en la forma:

$$f_l = (0 \cap g_{l,0}) \cup (1 \cap g_{l,1}) \cup \ldots \cup ((r-1) \cap g_{l,r-1}) \qquad (3.4)$$

en donde $g_{l,i}$, i=0,1,..,r-1, son funciones binarias definidas por:

$$\begin{aligned}
g_{l,0} &= \bar{h}_{l,0} \cap \bar{h}_{l,1} \cap \ldots \cap \bar{h}_{l,p-2} \cap \bar{h}_{l,p-1} \\
g_{l,1} &= \bar{h}_{l,0} \cap \bar{h}_{l,1} \cap \ldots \cap \bar{h}_{l,p-2} \cap h_{l,p-1} \\
g_{l,2} &= \bar{h}_{l,0} \cap \bar{h}_{l,1} \cap \ldots \cap h_{l,p-2} \cap \bar{h}_{l,p-1} \qquad (3.5) \\
&\ldots \qquad\qquad\qquad \ldots \\
g_{l,r-1} &= h_{l,0} \cap h_{l,1} \cap \ldots \cap h_{l,p-2} \cap h_{l,p-1}
\end{aligned}$$

con $\bar{h}_{l,j} = (r-1) - h_{l,j}$.

El tamaño del plano producto de la PLA de la figura 3.6 es:

$$k.n.r$$

siendo k el número de términos producto necesarios para la síntesis de las funciones $h_{i,j}$.

La principal ventaja de esta PLA es la posibilidad de aplicarle las mismas técnicas de simplificación existentes para PLAs binarias. Como inconvenientes pueden citarse, la necesidad de utilizar codificadores multivaluados en las salidas, y el aumento del número de líneas del plano suma en un factor $[\lg_2 r]$ respecto a una PLA del tipo 1.

En la sección 3.1 indicamos que una de las técnicas utilizadas para reducir el número de términos producto de una PLA binaria es el empleo de decodificadores binarios en las entradas. La reducción en el número de términos producto depende de la forma en que se asignen las entradas a los decodificadores [163,165]. Este tipo de simplificación puede trasladarse a una PLA multivaluada con la estructura de la figura 3.6. Para ello, es necesario introducir en las entradas decodificadores de 2 bits y una red de permutación programable entre estos y los generadores de literales (ver figura 3.7). En este caso, los generadores de literales no tienen por qué ser los correspondientes al conjunto mínimo completo.

También pueden conseguirse reducciones en el número de términos producto modificando adecuadamente la codificación de

las salidas, en términos de las funciones $h_{i,j}$, respecto de la codificación original dada por las ecuaciones (3.4) y (3.5). Considerando el ejemplo sencillo de un sumador tetravaluado de dos entradas, pueden obtenerse los tamaños mínimos de PLA indicados en la tabla 3.2 [164]. En dicha tabla se consideran

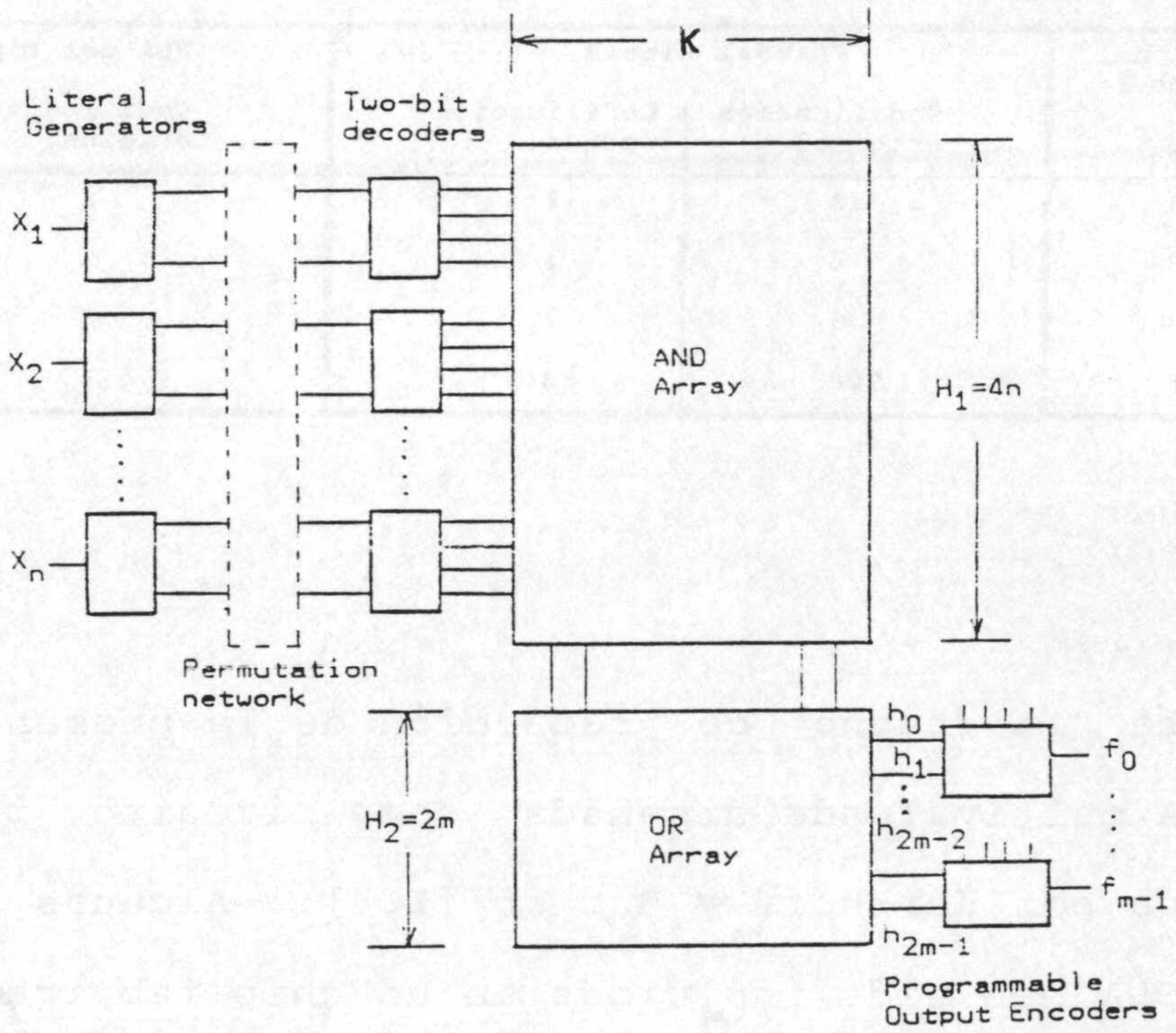


FIGURA 3.7 PLA del tipo 2 tetravaluada con decodificadores de dos bits y codificadores de salida programables [164].

sólo los planos producto y suma de las PLAs. La reducciones totales de tamaño no son tan acusadas como indica la tabla si tenemos en cuenta el aumento de la complejidad de los circuitos adicionales:

- Codificadores r-valuados de salida (PLA del tipo 2).

- Posibilidad de programación de los codificadores de salida (para seleccionar la codificación optima de las salidas).
- Decodificadores de dos bits y red de permutación programable (para seleccionar la asignación optima de las entradas).

TABLA 3.2: Comparación de las PLAs del tipo 1 y 2 propuestas por Sasao [164].

	PLA del tipo 1	PLA del tipo 2		PLA del tipo 2 con decodificadores de dos bits	
		Codificación original	Codificación óptima	Codificación original	Codificación óptima
H_1	0	8	8	8	8
H_2	2	4	4	2	2
k	13	9	7	5	4
Tamaño	156	108	82	60	48

Hasta la fecha de redacción de la presente memoria, la única PLA multivaluada diseñada como circuito integrado es la presentada por Kerkhoff y Butler [104]. Algunas celdas básicas de esta PLA han sido incluidas en un chip fabricado en tecnología P2CCD (Profiled Peristaltic CCD) [12]. Esta PLA está ideada para una realización MIN/TSUM, pero difiere de otras estructuras propuestas para tecnología CCD y basadas en los mismos operadores [33, 182]. Como se muestra en la figura 3.8, en lugar de utilizar generadores de literales de entrada al plano MIN, las señales de entrada multivaluadas (en modo de tensión) recorren todo este plano, utilizándose celdas programables generadoras de literales intervalo en cada punto de cruce fila-columna. Esta solución reduce a n (número de variables multivaluadas de entrada) el número de filas del plano MIN,

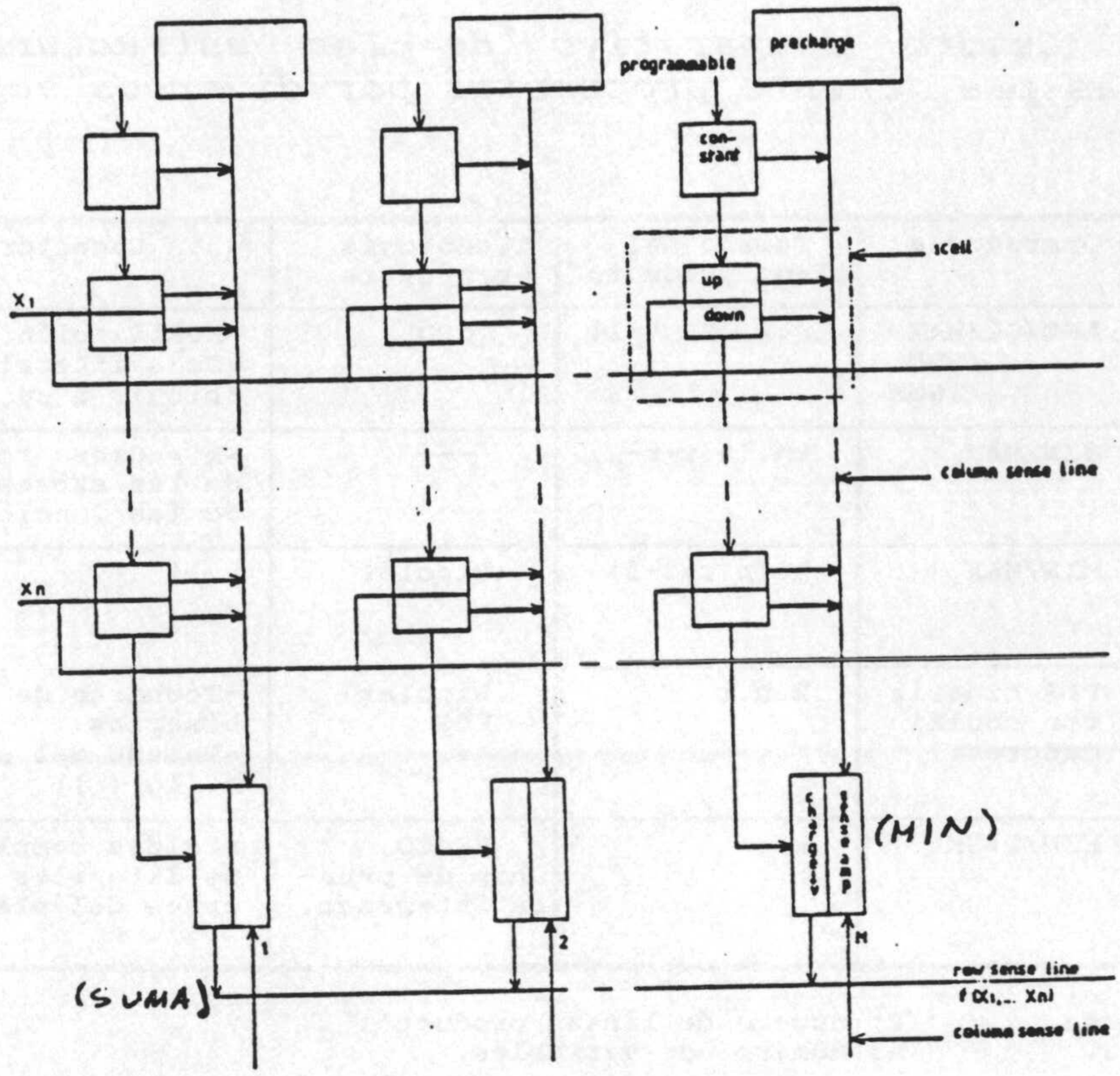


FIGURA 3.8 Esquema de la PLA propuesta por Kerkhoff y Butler [104]

pero aumenta la anchura de cada fila. Por otra parte, cada columna debe transportar, además de la carga resultante de la operación MIN y las señales de temporización que controlan las transferencias de carga, líneas de tensión de referencia para programar los generadores de literales de ducha columna. Utilizando reglas de diseño de 5 µm y dos niveles de polisilicio, cada celda básica del plano MIN, para lógica 4-valuada, ocupa una superficie de 100x100 μm^2 [104].

La tabla 3.3 resume las características más relevantes de las estructuras de PLAs propuestas para la síntesis de funciones multivaluadas y comentadas en esta sección.

TABLA 3.3: Cuadro comparativo de las estructuras de PLAs multivaluadas previamente propuestas por diversos autores.

Autores	Literales/ variable	Operadores	Tamaño del plano producto	Tecnología propuesta	Comentarios
TIRUMALAI y BUTLER [182]	2^r	AND/OR/MAX /SUM /TSUM	$n.k.(2^{r-1}-1)$	CCD	-Realización con tres niveles. -Cada literal sólo puede contribuir a un término producto.
KUO y FANG [109]	r	MIN/MAX	$k^*.(n.r+r-2)$	---	-k^*=número total de minterms de las expresiones canónicas de las funciones.
SASAO [164] (tipo 1)	r	MIN/MAX	$k.(n.r+2-2)$	Bipolar	
SASAO [164] (tipo 2)	r	PLA binaria con codificadores.	k.n.r	Bipolar MOS	-Técnicas de minimización binarias. -Tamaño del plano suma: $m.[\lg_2(r)]$
KERKHOFF y BUTLER [104]	Programables.	MIN/TSUM	k.n	P2CCD Chip de prueba integrado.	-Celdas complejas generadoras de literales en los puntos de cruce del plano MIN.

m: número de funciones. k: número de líneas producto.
r: base de la lógica. n: número de variables.

3.3 Técnicas de minimización de funciones multivaluadas.

Para optimizar el tamaño de una PLA multivaluada, al igual que se hace para PLAs binarias (sección 3.1), debe aplicarse alguna técnica de simplificación que minimice el número de términos producto necesarios para las funciones a sintetizar.

En un principio los métodos de simplificación de funciones multivaluadas propuestos consistían en adaptaciones a la

lógica multivaluada de los métodos clásicos de minimización booleana [158], basados en la generación del conjunto de todos los implicantes primos y la selección posterior de una cobertura mínima a partir de dicho conjunto. Sin embargo, en lógica multivaluada no es cierto que la cobertura mínima de una función conste necesariamente de implicantes primos, lo que obliga a tener que considerar, en la selección del conjunto mínimo, todos los implicantes de la función. Esto hace que al aumentar el número de variables (o el valor r de la lógica) el problema de la minimización se haga practicamente intratable [128].

3.3.1 Métodos heurísticos.

Se han desarrollado métodos de simplificación de funciones multivaluadas que, si bien no dan siempre como resultado la solución mínima, se aproximan a ella en tiempos de cálculo y cantidad de memoria ocupada razonables. Los más referenciados son los propuestos por Pomper y Armstrong [147, 148], Besslich [10], y Dueck y Miller [34]. Estos tres métodos heurísticos se basan en la aproximación de cobertura directa de la función a simplificar y, con las diferencias que indicaremos más adelante, parten de una réplica de la función como colección de minterms cuyos valores se corresponden con los de la función, y obtienen la cobertura de esta siguiendo los pasos:

1.- Seleccionar uno de los minterms mediante alguna regla heurística.

2.- Elegir uno de los implicantes de la función que cubra el minterm seleccionado, es decir, que incluya y tenga el mismo valor que el minterm en cuestión.

3.- Eliminar o sustraer, de los minterms que queden por cubrir, todos los cubiertos por el implicante elegido.

El proceso anterior se repite hasta que no quede ningún minterm por cubrir.

La eficiencia del método de cobertura directa en la obtención de soluciones mínimas depende de cuales sean las reglas heurísticas aplicadas en la selección de cada mimterm y del implicante que lo incluya.

Pomper y Armstrong [148] presentan dos algoritmos alternativos aplicables a la simplificación de expresiones MIN/MAX o MIN/TSUM. La selección del minterm a cubrir se lleva a cabo de forma aleatoria. Cuando se consideran realizaciones MIN/TSUM la elección se hace entre los minterms de valor mínimo. Una vez seleccionado un minterm, el implicante que se elige es el de mayor tamaño que lo cubra. Cuando se trata de una realización MIN/MAX, para cada implicante elegido se actualiza la réplica de la función eliminando los minterms que cubre. Si la realización es MIN/TSUM, el valor del implicante se resta de los

valores actuales de la función correspondientes a minterms incluidos en el implicante.

La simplicidad de este algoritmo hace que su mayor ventaja sea la rapidez de ejecución, pero es el que se aleja más de la solución mínima.

El algoritmo propuesto por Besslich [10] introduce reglas de selección más complejas que las mencionadas para el algoritmo anterior. En primer lugar, en la elección del minterm, se considera el más aislado de entre los que quedan por cubrir en la réplica actual de la función. Con la elección de minterms aislados para cubrir se intenta encontrar posibles implicantes esenciales (cubren un minterm no cubierto por ningún otro implicante) y evitar la existencia de implicantes redundantes en la solución final.

Para cada minterm elegido se generan todos los implicantes que lo cubren y se selecciona el más eficiente. El factor de eficiencia considerado es el cociente entre el número de minterms que cubre el implicante y el coste asociado con la realización física de dicho implicante. Aplicado a una PLA en la que todos los literales tengan asociado el mismo coste, la regla anterior lleva a la misma selección de implicantes que el algoritmo de Pomper y Armstrong.

El algoritmo de Besslich puede aplicarse a realizaciones MIN/MAX o MIN/TSUM; sin embargo, en el segundo caso, la

selección de los implicantes de mayor tamaño puede hacer más compleja la cobertura del resto de la función [33].

Dueck y Miller [34] proponen un algoritmo similar al de Besslich pero especialmente ideado para realizaciones MIN/TSUM. La elección del minterm a cubrir se efectua de manera similar a la del algoritmo de Besslich: siempre se considera el más aislado de entre los de valor mínimo. Sin embargo, de entre todos los implicantes que cubren el minterm elegido se selecciona aquel que, al restarse de los valores de la función, haga que los valores que quedan por cubrir sean lo más similares posible, de tal manera que, para el resto de la función, puedan construirse implicantes de mayor tamaño y la cobertura exija menor número de ellos.

Tirumalai y Butler [181] efectúan una comparación de los tres algoritmos de cobertura directa mencionados, aplicándolos a una realización MIN/TSUM y adaptados para utilizar sólo literales intervalo. La aplicación de los algoritmos a un total de 7000 funciones de dos variables 4-valuadas con un número de valores no nulos entre 3 y 16 (500 funciones de cada tipo) prueba que, si bién no hay diferencias muy acusadas entre los resultados producidos por los tres algoritmos, el de Dueck y Miller es más eficiente que el de Besslich, y éste más que el de Pomper y Armstrong. Analizando los datos presentados en [181] resulta que, bajo las condiciones de prueba indicadas con anterioridad, las soluciones producidas por el algoritmo de Dueck y Miller contienen aproximadamente un 4% más implicantes

que la cobertura mínima, un 4.6% el algoritmo de Besslich y, algo más distanciado, un 6.5%, el de Pomper y Armstrong. La solución mínima se obtiene para un 79%, 76%, y 69%, respectivamente, de las funciones generadas. La aplicación para cada función del algoritmo más eficiente de los tres (para dicha función) hace que el número de implicantes requerido difiera sólo en un 1.5% de la cobertura mínima, y se obtenga esta última para un 91% aproximadamente de las funciones generadas.

3.4 Conclusión.

En este capítulo hemos realizado una introducción a la síntesis de sistemas digitales multivaluados mediante PLAs. Se han descrito en primer lugar las estructuras básicas de PLAs binarias y las técnicas usualmente empleadas para su optimización.

Hemos definido algunos términos de utilidad en este y posteriores capítulos, y se ha hecho una revisión de las estructuras de PLAs multivaluadas hasta el momento propuestas.

En cuanto a las técnicas de optimización, hemos centrado la atención en los métodos de minimización de funciones multivaluadas que consideramos apropiadas para la síntesis mediante PLAs; en particular, métodos heurísticos aplicables a realizaciones de las funciones en términos de los operadores

mínimo y suma truncada. De los métodos heurísticos considerados para este tipo de realizaciones, el propuesto por Dueck y Miller [34] es el más eficiente, al menos cuando se utilizan literales intervalo. En el capítulo 4 consideraremos este método, así como una versión modificada del mismo, para la obtención de las expresiones simplificadas de las funciones a sintetizar mediante una nueva PLA multivaluada; expresiones en las que se incluyen cualquier tipo de literales.

CAPITULO 4

DISEÑO DE PLAs CMOS MULTIVALUADAS EN MODO DE CORRIENTE

En este capítulo presentamos una solución a la síntesis de funciones multivaluadas consistente en una matriz lógica programable integrable en tecnología CMOS.

La estructura de PLA propuesta lleva a cabo una síntesis MIN/TSUM adaptada para su realización en la mencionada tecnología (sección 4.1). La particularización de la PLA puede efectuarse de forma automática, partiendo de la especificación de las funciones a sintetizar, mediante el uso de los programas de minimización que hemos desarrollado y que presentamos en la sección 4.2.

En la sección 4.3 se incluyen distintas realizaciones CMOS de la PLA multivaluada propuesta. En esta misma sección se

proponen celdas de memoria multivaluadas dinámicas; las cuales, añadidas a la PLA básica, permiten la construcción de sistemas secuenciales multivaluados.

4.1 Estructura de la PLA propuesta.

Para llevar a cabo una síntesis MIN/MAX de las funciones multivaluadas, otras tecnologías tales como la bipolar son más adecuadas que la MOS, debido a que en esta última la realización del mínimo y máximo multivaluados requiere un elevado número de componentes. En tecnología MOS es preferible, frente a la síntesis MIN/MAX, utilizar PLAs binarias NAND/NAND o NOR/NOR que tengan como entradas las funciones literal y cuyas salidas sean codificadas posteriormente a multivaluado [164].

En el caso de una síntesis MIN/TSUM o MIN/SUM, pueden en principio conseguirse realizaciones físicas con reducido número de componentes si la suma (truncada o no) se efectua en modo de corriente o de carga eléctrica. No obstante, al intentar diseñar un plano MIN en el que las señales de entrada sean corrientes o cargas eléctricas se plantea la necesidad de disponer de tantas réplicas de cada señal como términos producto a los que contribuya (vease por ejemplo el esquema propuesto por Tirumalai y Butler, comentado en la sección 3.2).

Nuestro propósito es aprovechar la ventaja de utilizar el modo de corriente para efectuar la suma de señales multivaluadas, pero sustituir la operación MIN sobre señales de corriente por otra operación más facilmente realizable en tecnología CMOS.

Dada una función r-valuada de n variables expresada como suma truncada de términos producto:

$$f(X_1, X_2, \ldots, X_n) = \sum_j c_j \cap X_1^{S_{1j}} \cap \ldots \cap X_n^{S_{nj}} \qquad (4.1)$$

$$0 \neq c_j \varepsilon R;\ X_j \varepsilon R;\ S_{ij} \subseteq R = \{0, 1, \ldots r-1\}$$

cada término producto, excluida la constante c_j, es una función binaria que toma valores en el conjunto $\{0, r-1\}$ y definida por el producto booleano de literales. Aplicando las leyes de De Morgan del algebra de Boole, cada una de estas funciones puede transformarse de la siguiente manera:

$$X_1^{S_{1j}} \cap X_2^{S_{2j}} \cap \ldots \cap X_n^{S_{nj}} = \overline{\overline{X_1^{S_{1j}}} \cup \overline{X_2^{S_{2j}}} \cup \ldots \cup \overline{X_n^{S_{nj}}}} \qquad (4.2)$$

Según la definición de función literal de la variable r-valuada X_i, se cumple lo siguiente:

$$\overline{X_i^{S_{ij}}} = X_i^{\overline{S}_{ij}} \qquad (4.3)$$

en donde $\overline{S}_{ij} = R - S_{ij}$.

Por otra parte, cada literal $X_i^{\overline{S}_{ij}}$ puede expresarse como:

$$X_i^{\overline{s_{ij}}} = \overline{\bigcup_{l \varepsilon \overline{S}_{ij}}} X_i^l \qquad (4.4)$$

Sustituyendo (4.3) y (4.4) en (4.2) resulta:

$$X_1^{s_{1j}} \cap X_2^{s_{2j}} \cap \ldots \cap X_n^{s_{nj}} = \overline{\bigcup_{i=1,..,n}} \left(\overline{\bigcup_{l \varepsilon \overline{S}_{ij}}} X_i^l \right) \qquad (4.5)$$

en donde el símbolo $\overline{\cup}$ representa el operador NOR binario.

Aplicando la propiedad asociativa de la operación OR a la expresión (4.5) y sustituyendo en (4.1) resulta:

$$f(X_1, X_2, \ldots, X_n) = \overline{\sum_j} c_j \cap \left(\overline{\bigcup_{\substack{i=1,..,n \\ l \varepsilon \overline{S}_{ij}}}} X_i^l \right) \qquad (4.6)$$

La expresión (4.6) indica que una función r-valuada puede obtenerse como suma truncada de términos NOR binarios ponderados con las constantes r-valuadas c_j. Los literales que intervienen en la expresión son los X^l, $l \varepsilon R$; es decir, sólo se necesita un conjunto mínimo de r literales por variable r-valuada.

En la figura 4.1 presentamos un esquema adecuado para la síntesis de m funciones de n variables r-valuadas en la forma indicada por (4.6). La PLA de la figura está constituida por n celdas generadoras de los literales X_i^l, $l=0,1,..,r-1$; un plano NOR binario que tiene como entradas los n.r literales, y un plano TSUM en el que se efectua la suma ponderada. El plano NOR opera con señales binarias, de valores 0 y r-1, en modo de

tensión. Cada línea producto de este plano conmuta una fuente de corriente de magnitud $c_j . I_0$, con $c_j \varepsilon \{1,2,..,r-1\}$.

En el plano TSUM se efectuan sumas de réplicas de las corrientes conmutadas por las líneas producto.

La programación o particularización de esta PLA afecta a los puntos de cruce de los planos NOR y TSUM y a las magnitudes c_j de las fuentes de corriente.

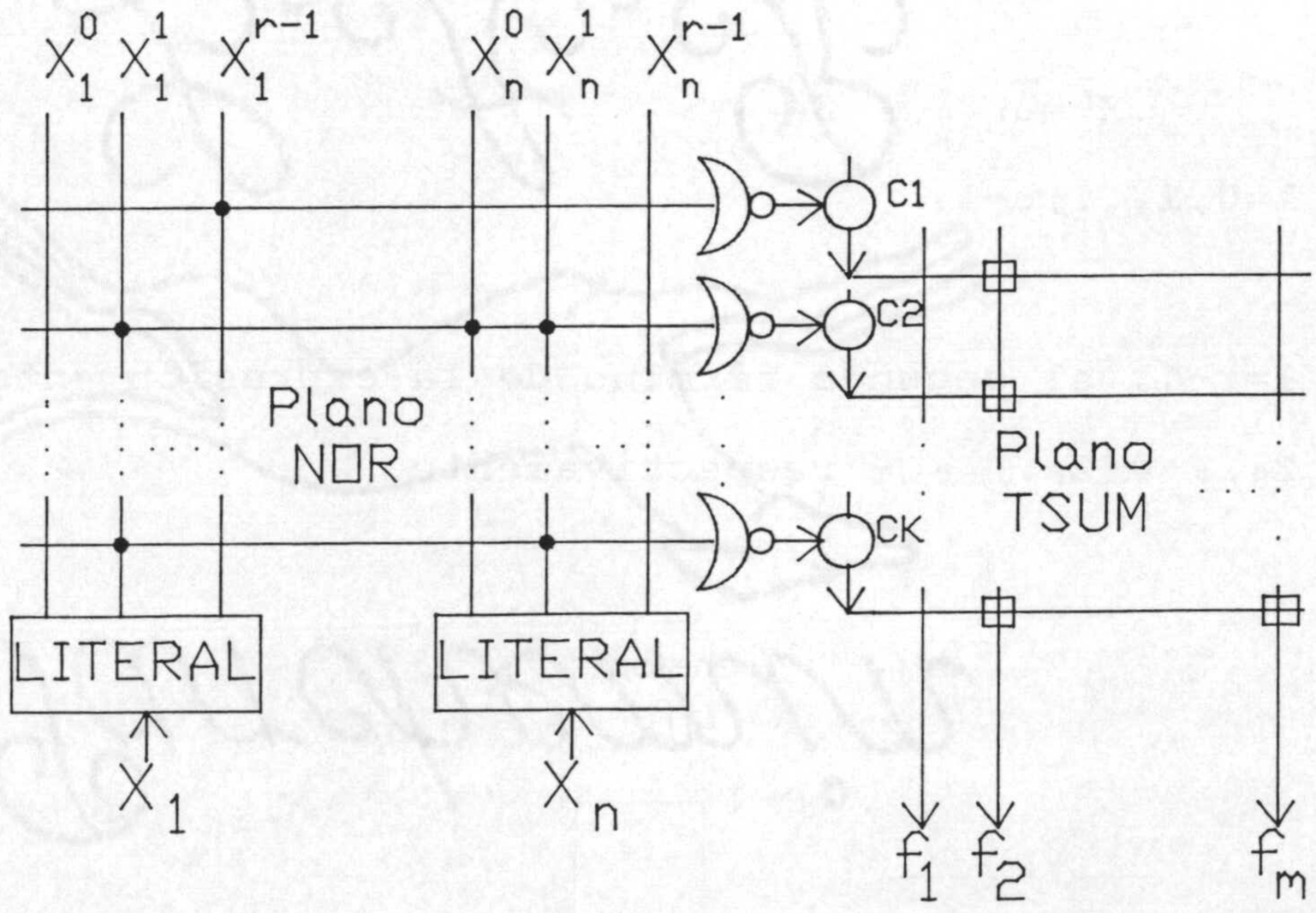


FIGURA 4.1. Estructura de PLA multivaluada propuesta.

Si no se añade al esquema de la figura 4.1 algún mecanismo de limitación de las corrientes en las salidas, estas pueden tomar valores superiores a $(r-1)I_0$; es decir, se obtiene la suma aritmética sin truncar. No obstante, en general no es

necesario limitar dichas corrientes dado que cualquier elemento conectado a las salidas de la PLA, al realizar una detección de umbrales de corriente, interpretará como nivel lógico r-1 cualquier valor de corriente mayor que $(r-1)I_0$.

La construcción de las celdas generadoras de literales requiere el empleo de detectores de umbral. Si se utilizan detectores de umbral cuya curva de transferencia sea la indicada en la figura 4.2, las funciones literal X^l de la variable r-valuada X podemos expresarlas en términos de las funciones de salida de los detectores de umbral (Z_U) como:

$$X^l = \overline{Z}_{l-0.5}\, Z_{l+0.5} \qquad (4.7)$$

en donde l=0,1,..,r-1.

Si l=0 ó l=r-1, el segundo término de la expresión anterior se reduce a $Z_{0.5}$ y $\overline{Z}_{r-1-0.5}$ respectivamente.

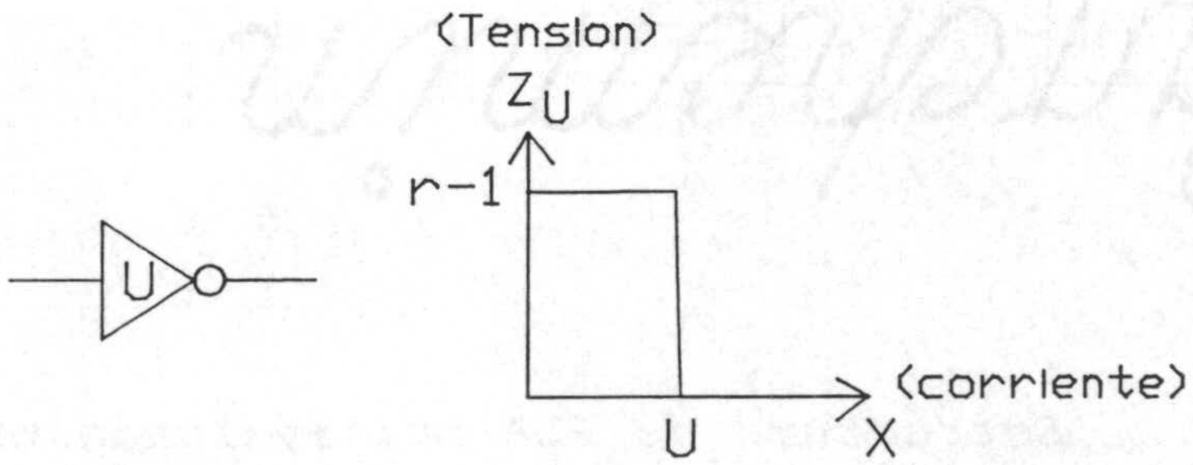


FIGURA 4.2. Símbolo y curva de transferencia de los detectores de umbral utilizados en las celdas generadoras de literales.

En la figura 4.3 se muestra el esquema lógico correspon-

diente a la celda generadora de literales para lógica 4-valuada.

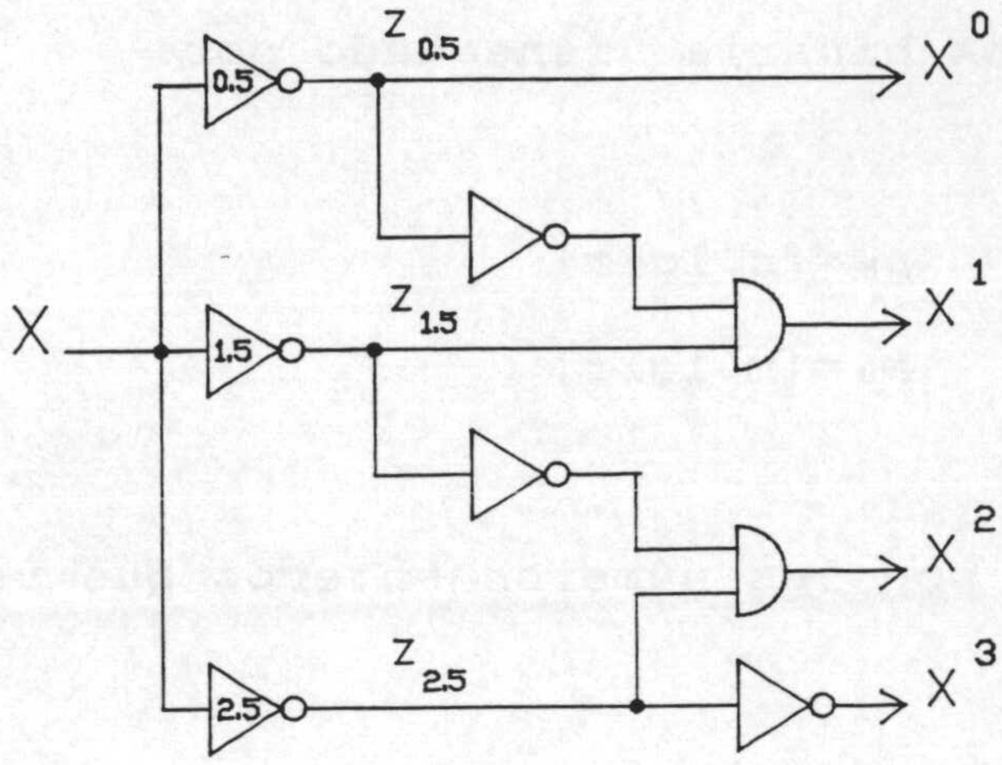


FIGURA 4.3. Generador de literales X^I 4-valuado.

4.1.1 Comparación con PLAs binarias

No es posible establecer expresiones generales que relacionen el tamaño de una PLA con el número de variables de entrada y el número de salidas. Fijado el número de entradas y de salidas, el número de términos producto necesarios depende de la naturaleza de las funciones a sintetizar (ver por ejemplo los resultados presentados en [163] para PLAs binarias con decodificadores de uno o dos bits en las entradas).

A toda PLA r-valuada con m líneas de salida y n de entrada se le puede hacer corresponder una PLA binaria lógicamente equivalente. La PLA binaria tendrá n_B entradas binarias, con

las que se puedan codificar las n entradas r-valuadas, y m_B salidas binarias que proporcionarán una codificación de las m funciones r-valuadas de salida. El número de entradas y de salidas de esta PLA binaria viene dado por:

$$n_B = [n.\lg_2 r]$$

$$m_B = [m.\lg_2 r] \qquad (4.8)$$

Es decir, n_B y m_B son los números enteros que verifican:

$$1+n.\lg_2 r > n_B \geq n.\lg_2 r$$

$$1+m.\lg_2 r > m_B \geq m.\lg_2 r \qquad (4.9)$$

La superficie ocupada por la realización integrada de una PLA r-valuada de n entradas, m salidas, y k términos producto, y con la estructura propuesta en esta sección, viene dada por:

$$S_r = n.S_{rL} + r.n.k.S_{rP} + m.k.S_{rS} \qquad (4.10)$$

en donde S_{rL} es la superficie ocupada por una celda generadora de literales r-valuada, S_{rP} es la superficie correspondiente a un punto de cruce en el plano NOR, y S_{rS} la de un punto de cruce en el plano TSUM.

La superficie ocupada por la correspondiente PLA binaria, suponiendo que utiliza decodificadores de un bit en las entradas, es:

$$S_B=[n.lg_2(r)].S_{BL} + 2.[n.lg_2(r)].k_B.S_{BP} + [m.lg_2(r)].k_B.S_{BS} \quad (4.11)$$

en donde S_{BL} es la superficie ocupada por un decodificador de un bit, y S_{BP} (S_{BS}) la superficie correspondiente a un punto de cruce en el plano producto (suma) de la PLA binaria.

Si la PLA binaria utiliza decodificadores de dos bits en las entradas (ocupando cada uno de ellos una superficie S_{2BL}), su tamaño podemos expresarlo como:

$$S_{2B}=[(n/2).lg_2(r)].S_{2BL} + 2.[n.lg_2(r)].k_{2B}.S_{BP} + \\ +[m.lg_2(r)].k_{2B}.S_{BS} \quad (4.12)$$

En las expresiones (4.11) y (4.12), k_B y k_{2B} son los números de términos producto de las respectivas PLAs binarias. Siempre se cumple que $k_{2B} \leq k_B$, pero no existe una relación similar entre k_{2B} ó k_B y el número k de términos NOR de la PLA r-valuada. Tirumalai y Butler [182], particularizando a lógica tetravaluada y realizaciones MIN/MAX, prueban la existencia de funciones para las cuales $k<k_B$ ó $k<k_{2B}$, y otras para las cuales se verifican las relaciones inversas. Este mismo resultado puede aplicarse para realizaciones MIN/TSUM. Dado que, fijados n y m (y suponiendo que las superficies correspondientes a los puntos de cruce son similares), es el número de términos producto el factor determinante de las superficies de PLA dadas por las expresiones (4.10) a (4.12), podemos concluir que, dependiendo del conjunto concreto de funciones a sintetizar,

puede resultar ventajosa o no (en cuanto a superficie ocupada) la realización multivaluada frente a la equivalente binaria. Otro aspecto a tener en cuenta es el ahorro de superficie que se consigue al aumentar r, debido a la reducción (en un factor $1/\lg_2 r$) del número de líneas de entrada y salida que conectan la PLA con otros circuitos.

Suponiendo que se mantiene constante la relación entre el número de términos producto de una PLA r-valuada y su equivalente mínima binaria cuando aumenta r, la relación entre la superficie ocupada por el plano producto de la realización r-valuada (segundo sumando de la expresión (4.10)) y la ocupada por el correspondiente binario (segundo término de la expresión (4.12)), tiene una dependencia con r:

$$(S_r / S_{2B})_{prod} \propto n.r/(2.[n.\lg_2 r]) \leq r/(2.\lg_2 r) \qquad (4.13)$$

Por otra parte, la relación entre las superficies ocupadas por los planos suma (últimos sumandos de las expresiones (4.10) y (4.12)) depende de r en la forma:

$$(S_r / S_{2B})_{suma} \propto m/[m.\lg_2 r] \leq 1/\lg_2 r \qquad (4.14)$$

Es decir, la superficie relativa del plano suma r-valuado (expresión (4.14)) disminuye al aumentar r, pero esta disminución se contrarresta por un aumento en la superficie relativa del plano producto (expresión (4.13)). No obstante, el último término de (4.13) presenta un mínimo cuando r coincide con el número *e*. La evaluación de dicho término para valores r=2 y r=3

(enteros más próximos al número *e*) indica que la lógica que optimiza la superficie del plano producto es la trivaluada.

Relacionando la superficie de la PLA r-valuada (expresión (4.10)) con las expresiones (4.9), se obtiene la siguiente desigualdad:

$$S_r \leq n_B . S_{rL} / lg_2 (r) + r . n_B . k . S_0 / lg_2 (r) + m_B . k . S_0 / lg_2 (r) \quad (4.15)$$

en donde se ha supuesto que la superficie ocupada por un punto de cruce en ambos planos de la PLA es la misma (S_0).

La superficie ocupada por una celda generadora de literales (S_{rL}) aumenta con r debido al aumento del número de puertas lógicas y de los detectores de umbrales necesarios para su realización. Un aumento unidad en r implica añadir, para obtener la nueva celda generadora de literales, un detector de umbral, un inversor binario, y una puerta AND binaria (ver figura 4.3). Suponiendo un crecimiento de S_{rL} lineal con r, la desigualdad (4.15) puede reescribirse en la forma:

$$S_r \leq (n_B . b . r + r . n_B . k + m_B . k) . S_0 / lg_2 (r) \quad (4.16)$$

en donde b es una constante de proporcionalidad.

Estableciendo alguna hipótesis sobre el valor del número de términos producto, podemos obtener una estimación del valor óptimo de r para el cual es mínima la superficie total de la

PLA r-valuada propuesta. Considerando como hipótesis que el valor del número k de términos producto es sólo función del número equivalente de entradas y salidas binarias (n_B y m_B), manteniendose constante al variar r, podemos determinar el valor de r que minimiza la superficie dada por el segundo término de (4.16). El valor de r óptimo es aquel para el que se verifica:

$$\ln(r) = m_B .k.(r.n_B .(b+k))^{-1} + 1 \qquad (4.17)$$

La solución de la ecuación (4.17) siempre da valores de r superiores al número *e*. Para un amplio rango de valores de la relación $m_B .k.(n_B .(b+k))^{-1}$, comprendidos entre .1 y 2, se obtienen los valores de r indicados en la figura 4.4, los cuales estan comprendidos entre 2.82 y 4.32. Esto indica la lógica más apropiada sería la trivaluada o la tetravaluada.

Fijados n_B y m_B, distintos conjuntos de funciones r-valuadas, codificables mediante m_B funciones binarias y dependientes de un conjunto equivalente de n_B variables binarias, requerirán en general distintos números de términos producto, y estos números presentarán distintas formas de variación con r. La solución de (4.17) proporciona el valor de r óptimo para todo conjunto de funciones tal que, las distintas codificaciones de estas funciones y de las variables de entrada correspondientes a distintos valores de r, exijan un mismo número de términos producto. Si no se cumple esta condición, una vez conocida la dependencia de k con r, la base óptima debe estimarse mediante la ecuación (4.16).

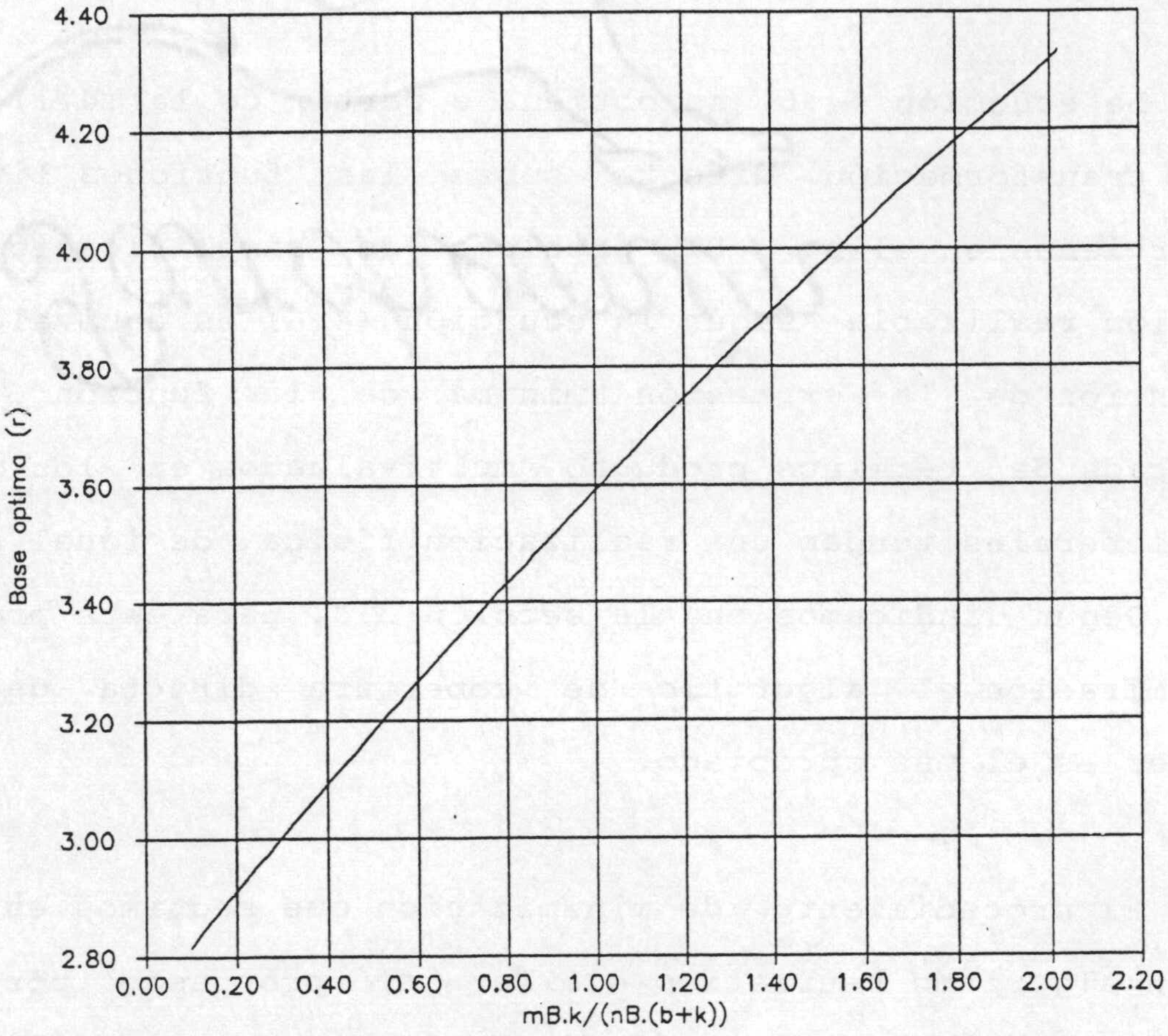


FIGURA 4.4: Valores de r que minimizan la superficie de la PLA, bajo la hipótesis de k=constante.

4.2 Procedimiento de minimización.

La síntesis de funciones multivaluadas mediante la PLA propuesta en la sección 4.1 requiere la simplificación previa de estas, es decir la obtención de expresiones de las funciones en la forma dada por la ecuación (4.6), en las cuales el número de términos en j sea mínimo. En esta sección se describe un

procedimiento de minimización adecuado para realizar esta simplificación.

La ecuación (4.6) se obtiene a partir de la (4.1) mediante una transformación directa sobre las funciones literal que intervienen en ella. El problema de simplificación de una función realizable según la ecuación (4.6) es equivalente a la obtención de la expresión mínima de la función como suma truncada de términos producto multivaluados en los que todos los literales tengan una realización física de igual complejidad. Según indicamos en la sección 3.3, para este problema de minimización el algoritmo de cobertura directa de Dueck y Miller es el más apropiado.

El procedimiento de minimización que seguimos está basado en un algoritmo heurístico similar al propuesto por Dueck y Miller. Dicho procedimiento queda resumido en los siguientes párrafos:

Dada una función r-valuada de n variables $f(X_1, X_2, ..., X_n)$, cada valor no nulo de f puede asociarse con un minterm del mismo valor cuya posición viene dada por la correspondiente combinación de las n variables. Las indiferencias se consideran asociadas con términos de valor r, para los cuales utilizaremos también la denominación de minterms.

Para obtener la expresión simplificada de la función f, se parte de una réplica de ésta, $f^*(X_1, X_2, .., X_n)$, y de su conjunto

de minterms asociados, seleccionando de entre estos uno con valor mínimo. A continuación se calculan los implicantes de f* que cubren el minterm seleccionado, determinándose cual de ellos es el más apropiado para ser incluido en la expresión de la función. El valor de este implicante debe restarse de los valores de aquellos minterms con posiciones comunes a los incluidos en el implicante. Dado que la solución final consiste en la suma truncada de términos producto, la operación referida es la resta aritmética. Como resultado se genera un nuevo conjunto de minterms, los cuales pueden considerarse como los valores no nulos de una nueva función f* .

El proceso anterior debe repetirse hasta conseguir que el conjunto de minterms resultante sea vacío o esté formado sólo por minterms de valor r; es decir, hasta obtener una función f* que sólo toma valores nulos e indiferencias.

En los apartados que siguen detallamos la forma en la que se efectua la selección del minterm a cubrir (apartado 4.2.1), la obtención de los implicantes que lo cubren (apartado 4.2.2), y el cálculo del implicante óptimo (apartado 4.2.3).

En el apartado 4.2.4 se describen brevemente los programas que hemos realizado para automatizar el proceso de minimización. Concretamente, un programa que implementa el algoritmo propuesto por Dueck y Miller, así como dos programas basados en versiones modificadas de dicho algoritmo.

4.2.1 Selección del minterm a cubrir.

Según indicamos en el apartado 3.3.1, los algoritmos de minimización propuestos por Besslich [10], y Dueck y Miller [34], seleccionan como minterm a cubrir aquel más aislado de entre los de valor mínimo.

El procedimiento de minimización que seguimos efectúa la selección del minterm como se propone en [34], y está basado en los siguientes conceptos:

Definición 4.1 (adyacencia). Dos minterms se dice que son adyacentes si sus posiciones difieren en una única coordenada. Si difieren en la coordenada a_i *se dice que son adyacentes respecto de la variable* X_i *o en la dirección i.*

A las posiciones de dos minterms adyacentes también le daremos el calificativo de adyacentes.

Definición 4.2 (adyacencia expandible): Dados dos minterms m_1 *y* m_2 *adyacentes, se dice que el segundo es una adyacencia expandible del primero si:*

$$\langle m_1 \rangle \leq \langle m_2 \rangle$$

en donde $\langle m_i \rangle$ *representa el valor del minterm* m_i *.*

Definición 4.3 (factor de aislamiento). Dado un conjunto M de minterms r-valuados, se define el factor de aislamiento del minterm m (XAIS(m)) como:

$$XAIS(m) = NAE_m + NDAE_m . (r-1) \qquad (4.18)$$

en donde NAE_m es el número de adyacencias expandibles del minterm m en el conjunto M, y $NDAE_m$ es el número de direcciones respecto de las cuales presenta adyacencias expandibles.

Según la definición anterior estamos considerando que el minterm más aislado tiene un factor de aislamiento mínimo. Dueck y Miller definen el factor de aislamiento como:

$$(NAE_m + NDAE_m . (r-1) + 1)^{-1}$$

Es decir, consideran que el minterm más aislado es el de factor de aislamiento máximo. No obstante, ambas formas de calcular el minterm más aislado dan el mismo resultado, siendo la primera más rápida de evaluar.

Ejemplo 4.1:

Consideremos el caso sencillo de dos variables 4-valuadas y el conjunto M especificado en la tabla 4.1.

De los minterms de valor mínimo de la tabla, los de coordenadas {0,0} y {0,3} son los más aislados (factor de aislamiento mínimo = 6). Según la ecuación (4.18) el minterm {2.1} resulta ser el menos aislado (factor de aislamiento 8) pues, aunque tiene menos adyacencias expandibles, estas corresponden a direcciones distintas.

TABLA 4.1: Conjunto M correspondiente al ejemplo 4.1

		X_1			
		0	1	2	3
X_2	0	1			
	1	1			1
	2	2		2	2
	3	1			

Una vez seleccionado el minterm a cubrir, debe procederse al cálculo de los implicantes que lo cubren. A continuación se propone una forma de realizar dicho cálculo.

4.2.2 Cálculo de implicantes.

Sea m el minterm seleccionado, definido por las coordenadas $a=\{a_1, a_2, .., a_n\}$, y M el conjunto de minterms cuyos valores se identifican con los valores no nulos de la réplica actual de la función a minimizar ($f^*(X_1, .., X_n)$).

Denominemos U_i al conjunto de valores b tal que, el minterm de coordenadas $\{a_1, .., a_{i-1}, b, a_{i+1}, .., a_n\}$ pertenezca a M y además sea una adyacencia expandible de m en la dirección i.

Por definición de adyacencia expandible, para cada dirección i=1,2,..,n, los términos producto en la forma:

$$\langle m \rangle \; {}_{n}X_i \, {}^{U\,i} {}_{n} \Big(\bigcap_{j \neq i} X_j \, {}^{a\,j} \Big) \qquad (4.19)$$

son implicantes de la función f* que cubren el minterm m.

Todos los posibles implicantes de f* que cubran el minterm m tienen necesariamente la forma:

$$\langle m \rangle \; {}_{n} \Big(\bigcap_{i=1}^{n} X_i \, {}^{v\,i} \Big) \qquad (4.20)$$

en donde los V_i son subconjuntos de los Ui tal que $a_i \varepsilon V_i$.

Como condición suficiente, para que un término en la forma (4.20) sea implicante de f*, cada elemento "e" del producto cartesiano $(V_i)^n$ debe verificar $f^*(e) \neq 0$ ó, lo que es equivalente, todos los minterms de posiciones $e \varepsilon (V_i)^n$ deben pertenecer al conjunto M.

Atendiendo a lo expuesto en los párrafos anteriores, mediante programa procederemos al cálculo de implicantes de la siguiente manera:

1.- Obtención de las adyacencias expandibles del minterm en las distintas direcciones y construcción de los conjuntos U_i.

2.- Generación de todos los subconjuntos V_i de cada U_i.

3.- Construcción de productos cartesianos $(V_i)^n$, los cuales definen términos producto en la forma (4.20).

4.- Comprobación de la condición suficiente de implicante para cada término producto generado.

Para cada conjunto U_i existen $2^{c(U_i)-1}$ subconjuntos V_i, en donde $C(U_i)$ es la cardinalidad de U_i. El número total de términos producto en la forma (4.20) que se generan es:

$$\prod_{i=1}^{n} 2^{c(U_i)-1} \qquad (4.21)$$

Si la generación de términos producto se realizase de forma directa, habría que considerar un total de $2^{(r-1)n}$ términos producto como posibles implicantes que cubren a un minterm dado. Esta última cantidad es, en general, bastante superior al número de términos generados mediante el procedimiento propuesto. Sólo en el caso de que todos los valores de la función sean no nulos, la cantidad (4.21) toma el valor máximo $2^{(r-1)n}$.

4.2.3 Selección del implicante óptimo.

De entre todos los implicantes de la función f* que cubran el minterm m (minterm más aislado de valor mínimo) se selecciona aquel que al sustraerse de los valores de f* haga que la función resultante presente un número mínimo de rupturas o valores diferentes en posiciones adyacentes. Con esta forma de selección se pretende, según indicamos en la sección 3.3, que con los valores de la función que quedan por cubrir se puedan formar implicantes de tamaño máximo, lo cual en principio facilita su cobertura mediante un número mínimo de términos.

En lugar de proceder al cálculo del número total de rupturas que resultan al seleccionar los distintos implicantes, interesa evaluar el cambio relativo en el número de rupturas que supone la selección de cada implicante, eligiéndose aquel para el cual el cambio obtenido sea mínimo.

Definición 4.4 (número de rupturas relativo): Dado un implicante I de la función $f^*(X_1, X_2, .., X_n)$, *se define el número de rupturas relativo de I* (NRR_I) *como el número total de rupturas que se provocan en f* al restarle el valor de I en todas las posiciones de minterms incluidos en dicho implicante.*

NRR_I es positivo, nulo, o negativo, dependiendo de que la función resultante al sustraer el valor del implicante presente mayor, igual, o menor número de rupturas que f*. Por regla general, el número de rupturas de la función correspondientes al conjunto de posiciones de minterms incluidos en el implicante no varía al sustraer el valor de éste. Para evaluar el número de rúpturas relativo del implicante I comparamos los valores de f* en posiciones de minterms incluidos en I, con los valores que toma dicha función en las posiciones adyacentes con cada uno de dichos minterms.

Sea <I> el valor del implicante I de la función f*, y A_I el conjunto de posiciones definido por los minterns incluidos en I. El cálculo del número de rupturas relativo de I lo efectuaremos de la siguiente manera:

1.-Hacer $NRR_I = 0$.

2.-Para cada minterm m_I *tal que* $\langle m_I \rangle \neq r$ *cuya posición* $a(m_I) \varepsilon A_I$, *y para cada dirección* $i=1,2,..,n$:

2.1.- NRR_I *se decrementa en 1 para cada posición* $b \notin A_I$, *adyacente a la posición* $a(m_I)$ *en la dirección i, que verifique:*

$$f^*(b) = \langle m_I \rangle - \langle I \rangle$$

2.2.- NRR_I *se incrementa en 1 para cada posición* $b \in A_I$, *adyacente a la posición* $a(m_I)$ *en la dirección i, que verifique:*

$$f^*(b) = \langle m_I \rangle$$

En el cálculo de NRR_I no se tienen en cuenta las indiferencias ya que no debe considerarse como ruptura de la función la existencia de una pareja de valores distintos (en posiciones adyacentes) en la que uno de ellos corresponda a una indiferencia.

Claramente, para cada posición b de minterm que cumpla la condición 2.1 anterior, al restar el valor del implicante se tendrá una ruptura menos de las que presenta f^*. Por el contrario, para cada posición que verifique la condición 2.2, al sustraer de f^* el valor del implicante, se está generando una nueva ruptura.

Dueck y Miller emplean [34] la denominación de "número de rupturas relativo" para referirse a una magnitud (RBC: "relative break count") con el mismo significado de la definición 4.4, pero proceden a una estimación de su valor como sigue:

1.- Hacer $RBC_I = 0$.

2.- Para cada minterm $m_I \varepsilon I$ tal que $\langle m_I \rangle \neq r$ cuya posición $a(m_I) \varepsilon A_I$, y para cada dirección i=1,2,..,n:

2.1.- Si hay una posición $b \notin A_I$, adyacente a la posición $a(m_I)$ en la dirección i, y tal que:

$$f^*(b) = \langle m_I \rangle - \langle I \rangle$$

entonces se resta 1 al valor de RBC_I.

2.2.- Si hay una posición $b \notin A_I$, adyacente a la posición $a(m_I)$ en la dirección i, y tal que:

$$f^*(b) = \langle m_I \rangle \neq r-1$$

entonces se suma 1 al valor de RBC_I.

En este caso, para una posición dada de un minterm de A_I y una dirección dada, se contabiliza como una sola ruptura menos la generación de dos o más valores iguales en posiciones adyacentes a la del minterm (condición 2.1). Y se contabiliza como una sola ruptura más la generación de dos o más valores distintos en posiciones adyacentes a la del minterm (condición 2.2). Además, en el cálculo propuesto por Dueck y Miller no se consideran las rupturas introducidas cuando el valor del minterm m_I es r-1, argumentando que, al tratarse de sumas

truncadas, en la posición de dicho minterm la contribución de los implicantes de la solución puede superar el valor r-1.

4.2.4 Algoritmo de minimización.

Dada la función r-valuada de n variables $f(X_1, X_2, .., X_n)$, obtendremos su expresión "mínima" como suma truncada de términos producto (equivalente a la suma truncada de términos NOR) según se indica en el siguiente algoritmo:

Paso 1: f* <- f (obtención de una réplica de f);

Paso 2: **Si** $f^*(X_1, X_2, .., X_n) \notin \{1, 2, .. r-1\}$ para cualesquiera $X_i \varepsilon \{0, 1, .., r-1\}$ **entonces FIN;**

Paso 3: XMAIS<- ∞, M_{min}=conjunto de minterms de f* con valor mínimo;
Para cada minterm $m_i \varepsilon M_{min}$:
Si XAIS(m_i)<XMAIS **entonces** m_{ais} <- m_i,
XMAIS<- XAIS(m_i);

Paso 4: NRR<- ∞, I_{opt} <- m_{ais};
Para cada implicante I de f* tal que $m_{ais} \varepsilon I$ y <I>=<m_{ais}>:
Calcular NRR_I,
Si $NRR_I \leq NRR$ **entonces** NRR<- NRR_I, I_{opt} <- I;

Paso 5: Incluir I_{min} en la expresión de f;
Para todas las posiciones *a* de minterms incluidos en I_{opt} tal que $f^*(a) \neq r$:
$f^*(a)$ <- $f^*(a)$-<I>,
Si $f^*(a)=0$ y $f(a)=r-1$ **entonces** $f^*(a)$ <- r;
Volver al *Paso 2*;

El cálculo del factor de aislamiento de cada minterm de valor mínimo (paso 3) se realiza en la forma indicada en el apartado 4.2.1.

El cálculo de los implicantes que cubren el minterm seleccionado (paso 4) se hace de la forma indicada en el apartado 4.2.2, considerando como primer implicante el propio minterm a cubrir, y procediendo posteriormente con implicantes de tamaño cada vez mayor; de tal manera que al final se seleccione el implicante de mayor tamaño de entre los que tienen un número de rupturas relativo mínimo.

Al restar el valor del implicante seleccionado de los valores de f* (paso 5), puede que resulte algún valor nulo como consecuencia de haber completado la cobertura de un valor r-1 de la función de partida (f). Este valor debe pasar a considerarse como indiferencia de f*, ya que la suma de términos producto es truncada al valor r-1.

Ejemplo 4.2:

Consideremos como ejemplo la función de dos variables tetravaluadas de la tabla 4.2. La aplicación de la ecuación (4.18) indica que el minterm de valor mínimo más aislado es el de coordenadas {2,1}.

Los implicantes de valor 1 que cubren el minterm {2,1} son:

$$I_1 = 1 \cap X_1^{2} \cap X_2^{1} \qquad I_2 = 1 \cap X_1^{1,2} \cap X_2^{1}$$

$$I_3 = 1 \cap X_1^{2,3} \cap X_2^{1} \qquad I_4 = 1 \cap X_1^{1,2,3} \cap X_2^{1}$$

Los números de rupturas relativos de estos implicantes, obtenidos siguiendo el procedimiento del apartado 4.2.3, son:

$NRR_{I\,1}=-3 \quad NRR_{I\,2}=-7 \quad NRR_{I\,3}=-1 \quad NRR_{I\,4}=-5$

TABLA 4.2: Función del ejemplo 4.2.

X_2 \ X_1	0	1	2	3
0	0	0	0	3
1	0	1	1	3
2	2	2	0	3
3	0	0	0	0

La tabla 4.3 ilustra el cálculo del número de rupturas relativo del implicante I_4. El implicante optimo es I_2, cuyo número de rupturas relativo es mínimo.

La sustracción del valor de I_2 en las posiciones de los minterms que incluye da como resultado los valores indicados en la tabla 4.4. Los dos minterms de valor mínimo de esta tabla tienen el mismo factor de aislamiento. La elección de cualquiera de ellos lleva a la obtención del implicante $I_5=2\,n\,X_1^{0,1}\,n\,X_2^{2}$ como implicante óptimo para ser incluido en la expresión de la función.

TABLA 4.3: Cálculo del número de rupturas relativo del implicante $I_4 = 1 \cap X_1^{1,2,3} \cap X_2^{1}$.

Minterm de I_4	Dirección i=1	Dirección i=2	NRR
$1\ X_1^{1}\ X_2^{1}$	-1	-2	-3
$1\ X_1^{2}\ X_2^{1}$	-1	-3	-4
$1\ X_1^{3}\ X_2^{1}$	0	+2	+2
		NRR resultante:	-5

TABLA 4.4: Tabla de la función resultante al sustraer I_2 .

X_2 \ X_1	0	1	2	3
0	0	0	0	3
1	0	0	0	3
2	2	2	0	3
3	0	0	0	0

Para la cobertura óptima de los valores no nulos que quedan al sustraer el valor de I_5, se obtiene el implicante $I_6 = 3 \cap X_1^{3} \cap X_2^{0,1,2}$.

La expresión simplificada de la función definida por la tabla 4.2 es pues:

$$f(X_1, X_2) = 1 \cap X_1^{1,2} \cap X_2^{1} \boxplus 2 \cap X_1^{0,1} \cap X_2^{2} \boxplus 3 \cap X_1^{3} \cap X_2^{0,1,2}$$

o bién, transformada a suma truncada de términos NOR (ecuación (4.6)):

$$f(X_1, X_2) = 1 \cap (\overline{X_1^0 \cup X_1^3 \cup X_2^0 \cup X_2^2 \cup X_2^3}) \boxplus 2 \cap (\overline{X_1^2 \cup X_1^3 \cup X_2^0 \cup X_1^1 \cup X_2^3}) \boxplus 3 \cap (\overline{X_1^0 \cup X_1^1 \cup X_1^2 \cup X_2^3})$$

La expresión obtenida aplicando el algoritmo propuesto coincide en este caso con la expresión mínima.

La aplicación del algoritmo de Dueck y Miller a la misma función da una solución que incluye un implicante más:

$$f(X_1, X_2) = 1 \cap X_1^{1,2,3} \cap X_2^1 \boxplus 2 \cap X_1^{0,1,3} \cap X_2^2 \boxplus 1 \cap X_1^3 \cap X_2^{0,2} \boxplus 2 \cap X_1^3 \cap X_2^{0,2}$$

Esto se debe a la forma de calcular el número de rupturas relativo de los implicantes (I_1 a I_4) que cubren el primer minterm seleccionado (minterm de coordenadas {2,1}). Dicho cálculo da como resultado:

$RBC_{I1} = -1 \quad RBC_{I2} = -4 \quad RBC_{I3} = -1 \quad RBC_{I4} = -4$

Lo cual implica considerar como implicante óptimo el I_4 pues, aunque su RBC coincide con el de I_2, su tamaño es mayor. La tabla 4.5 ilustra el cálculo de RBC_{I4}.

Hemos realizado tres programas para la minimización mediante cobertura directa de funciones multivaluadas, los cuales están redactados en FORTRAN-77 para una computadora

MV-10000 de Data General. Cualquiera de ellos es utilizable para obtener espresiones simplificadas de la función como suma

TABLA 4.5: Cálculo del número de rupturas relativo del implicante $I_4 = 1 \cap X_1^{1,2,3} \cap X_2^{1}$ *como se propone en [34].*

Minterm de I_4	Dirección i=1	Dirección i=2	NRR
$1\ X_1^{1}\ X_2^{1}$	-1	-1	-2
$1\ X_1^{2}\ X_2^{1}$	-1	-1	-2
$1\ X_1^{3}\ X_2^{1}$	0	0	0
		NRR resultante:	-4

truncada de términos producto. El número de términos producto obtenido por cada programa es en general distinto, no pudiéndose asegurar que uno de ellos proporcione siempre (para cualquier función) la solución óptima.

Los programas realizados, cuyos listados se incluyen en el apéndice 1, son:

MINTSUM1: implementa el algoritmo de Dueck y Miller [34].

MINTSUM2: implementa el algoritmo anteriormente propuesto en esta sección.

MINTSUM3: difiere de MINTSUM2 en la forma de elegir el minterm a cubrir. En lugar de calcular el factor de aislamiento de cada minterm de valor mínimo, para seleccionar el que se considera más aislado, se efectúa una elección al azar entre los minterms de valor mínimo de la función.

Cada uno de los tres programas se compone de un módulo principal, en el cual se incluye la selección de minterms a cubrir, y cuatro subrutinas: PRODUCTO, COBERTURA, IMPLICANTE, y RESTA. De ellas, las dos primeras y la última son idénticas para los tres programas.

Mediante las subrutinas PRODUCTO, COBERTURA, e IMPLICANTE se efectua el cálculo y selección de implicantes. En lugar de construirse una tabla con todos los implicantes que cubren el minterm seleccionado por el programa principal, y posteriormente analizar la tabla costruida para determinar el implicante óptimo, lo que se hace es calcular de forma recursiva cada implicante y determinar su correspondiente número de rupturas relativo; memorizándose sólo el implicante con un valor mínimo de este número.

La rutina PRODUCTO efectua el cálculo recursivo de términos producto que cubren el mínterm seleccionado, considerandose sólo términos producto en la forma (4.20). PRODUCTO utiliza la rutina COBERTURA para determinar si cada término producto generado es o no implicante. Si se trata de un implicante, se procede al cálculo de su número de rupturas relativo mediante la rutina IMPLICANTE.

IMPLICANTE siempre devuelve como resultado el implicante con número de rupturas relativo mínimo. Cuando se han considerado todos los implicantes, el último devuelto por IMPLICANTE se incluye en la solución.

El programa principal utiliza la subrutina RESTA para sustraer, de los correspondientes valores no nulos de la función, el valor del implicante incluido en la solución; dándose un tratamiento especial a las indiferencias y a los valores r-1 ya cubiertos.

Al objeto de comparar la eficiencia de los tres programas de minimización, hemos obtenido los resultados mostrados en la tabla 4.6, los cuales corresponden a la simplificación de un total de 1600 funciones tetravaluadas. Los valores de cada una de estas funciones han sido generados aleatoriamente, prefijando el porcentaje de ellos que deben ser no nulos.

Las dos primeras columnas de la tabla 4.6 especifican el número de variables de las funciones y el correspondiente número de funciones generadas. La tercera columna indica los porcentajes de valores no nulos considerados. En las tres columnas siguientes se especifica el valor medio, sobre el número de funciones generadas, del número de términos producto de las soluciones obtenidas por cada uno de los programas. La última columna indica el valor medio del número de términos producto cuando, para cada función, se considera la solución mínima de entre las proporcionadas por los tres programas.

De los datos de la tabla 4.6 pueden extraerse las siguientes conclusiones:

- El programa MINTSUM2 proporciona en la mayoría de los casos la solución óptima.

TABLA 4.6: Comparación de los programas de minimización.

Número de variables	Número de funciones	% valores no nulos	MINTSUM1	MINTSUM2	MINTSUM3	MINIMO
2	100	25	2.73	2.62	2.62	2.62
2	100	50	4.71	4.41	4.42	4.40
2	100	75	5.34	5.36	5.36	5.19
2	100	100	5.00	5.02	5.04	4.74
3	100	25	10.35	9.86	9.89	9.84
3	100	50	16.06	15.51	15.53	15.17
3	100	75	18.16	18.01	17.78	17.12
3	100	100	16.66	17.42	17.06	15.74
4	100	10	18.6	18.21	18.22	18.21
4	100	25	38.79	37.01	37.07	36.76
4	100	50	58.11	55.44	55.34	54.47
4	100	75	64.82	62.98	63.23	61.73
4	100	100	56.64	61.52	62.94	56.14
5	100	10	71.25	69.28	69.31	69.22
5	100	25	145.67	139.65	139.74	138.82
5	100	50	214.81	206.94	208.03	205.77

- La eficiencia de los programas MINTSUM2 y MINTSUM3 es muy similar, lo cual indica que no es de especial importancia elegir como minterms a cubrir los más aislados. De hecho, según puede verse en la tabla, hay casos en los que es preferible realizar una elección aleatoria de minterms.
- El programa MINTSUM1 sólo resulta más eficiente, en media, cuando el porcentaje de valores no nulos de la función es próximo al 100%.
- La elección de la solución óptima entre las proporcionadas por los tres programas, para un conjunto concreto de funciones, puede suponer una disminución considerable en el número total de términos producto requeridos.

4.3 Realizaciones CMOS de la PLA propuesta

En esta sección se proponen distintas realizaciones CMOS de la estructura de PLA presentada en la sección 4.1. La elección de la realización más apropiada para la síntesis de un conjunto dado de funciones multivaluadas depende de factores tales como el tamaño de la PLA, el consumo de potencia, y la velocidad de operación.

En primer lugar se presentan esquemas de las celdas generadoras de literales (apartado 4.3.1). En el apartado 4.3.2 se muestran esquemas alternativos de los planos NOR y TSUM correspondientes a realizaciones seudo-NMOS y CMOS dinámica. En el apartado 4.3.3 se propone un esquema de celda de memoria multivaluada dinámica, la cual es utilizable como celda de registro añadido a la PLA básica.

Todos los esquemas de circuito presentados en esta sección son integrables siguiendo un proceso CMOS estándar de fabricación de circuitos binarios.

Los resultados obtenidos mediante simulación con SPICE de los circuitos propuestos se presentan en el capítulo 6.

4.3.1 Celdas generadoras de literales

Los esquemas de celdas generadoras de literales que presentamos están ideados para señales de entrada multivaluadas en modo de corriente. No obstante, las entradas a la PLA propuesta en la sección 4.1 no han de ser necesariamente corrientes multivaluadas. La elección del modo de tensión en las entradas afecta sólo al diseño de los detectores de umbral incluidos en las celdas generadoras de literales.

En lo que sigue nos referiremos sólo a entradas multivaluadas en modo de corriente. Como detector de umbrales de corriente puede utilizarse un esquema similar al propuesto por Current y Freitas [47], consistente en dos espejos de corriente complementarios con tantas salidas como umbrales sean necesarios. La figura 4.5 muestra un esquema eléctrico del detector de umbrales de corriente para lógica 4-valuada. El circuito utiliza una sola referencia (de voltaje o corriente) para establecer la unidad de corriente básica I_0. Los múltiplos o fracciones de esta corriente (corrientes umbrales) se obtienen por modificaciones en la geometría de los transistores PMOS. En el capítulo 2 se trató en detalle el funcionamiento de este circuito.

El detector de umbrales de corriente podría simplificarse, eliminando la necesidad de la fuente de referencia, si en el proceso de fabricación del circuito se añade un implante iónico a los transistores PMOS que fijan las corrientes umbral para

hacerlos de empobrecimiento. La dosis de implante determinaría en este caso la magnitud de la corriente unidad utilizada.

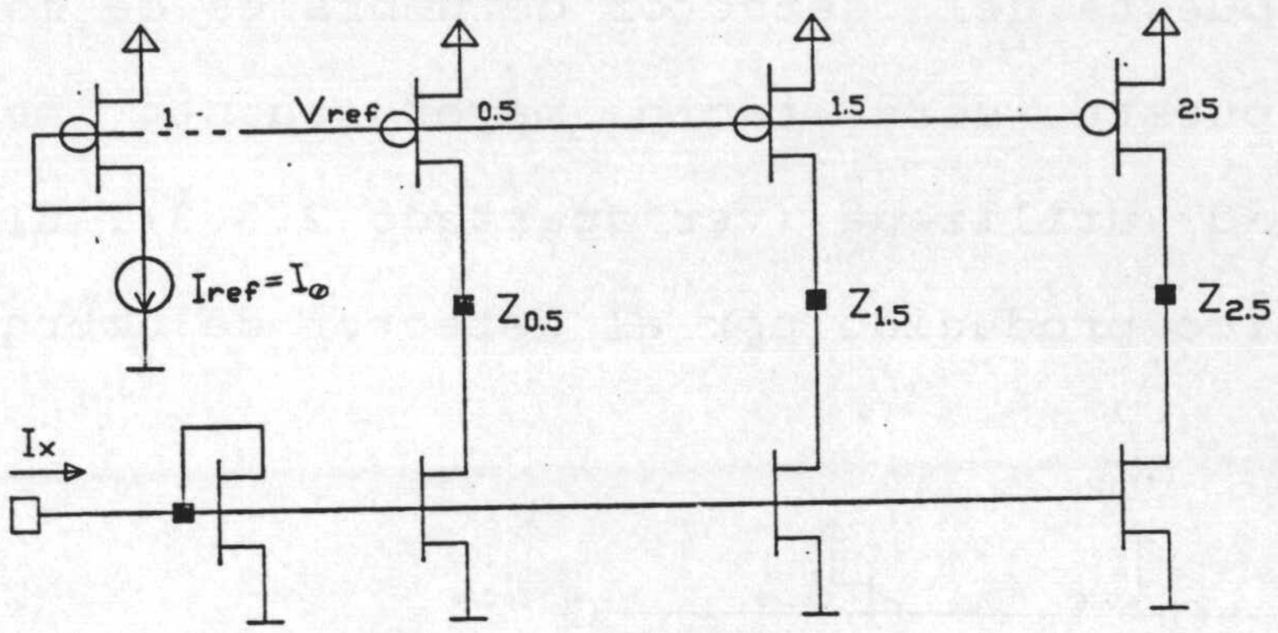


FIGURA 4.5. Detector de umbrales de corriente tetravaluado.

La figura 4.6 muestra el esquema lógico de una celda generadora de literales 4-valuada. Este esquema es una adaptación, para su realización en CMOS, del indicado en la sección 4.1.

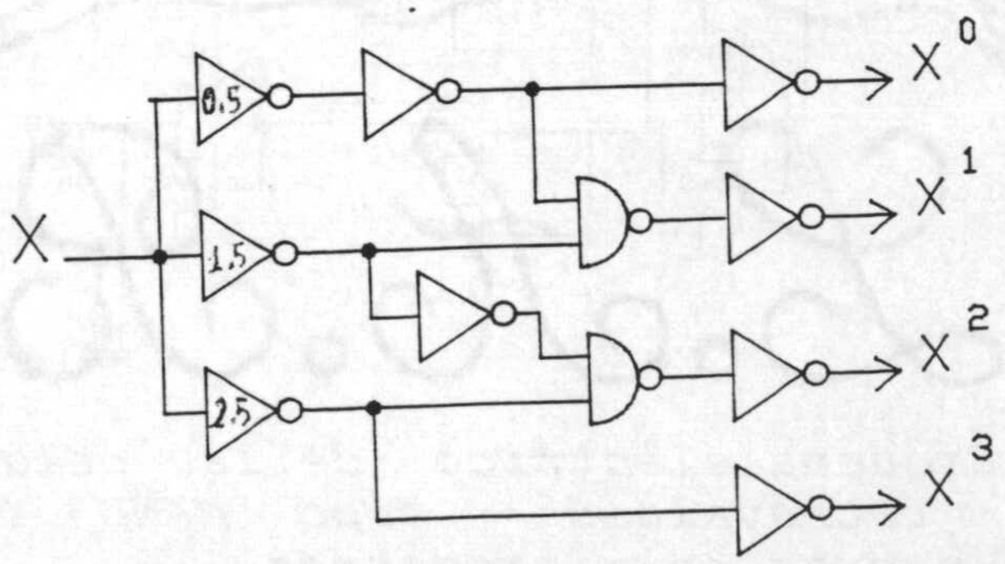


FIGURA 4.6: Esquema lógico de la celda generadora de literales tetravaluada tipo NAND

La figura 4.7 muestra un esquema de circuito correspondiente al esquema lógico anterior. Las relaciones de tamaño W/L que se indican en esta figura para los transistores de la última etapa

deben adaptarse, en cada caso, para producir la carga y descarga de las salidas (líneas de literal) en los tiempos requeridos. Con esta salvedad, la velocidad de operación del circuito de la figura 4.7 está limitada fundamentalmente por el tiempo de respuesta del detector de umbrales de corriente. El tiempo de respuesta será tanto mayor cuanto menor sea la corriente unidad utilizada (ver apartado 2.3.3). El consumo de potencia estático producido por el detector de umbrales de

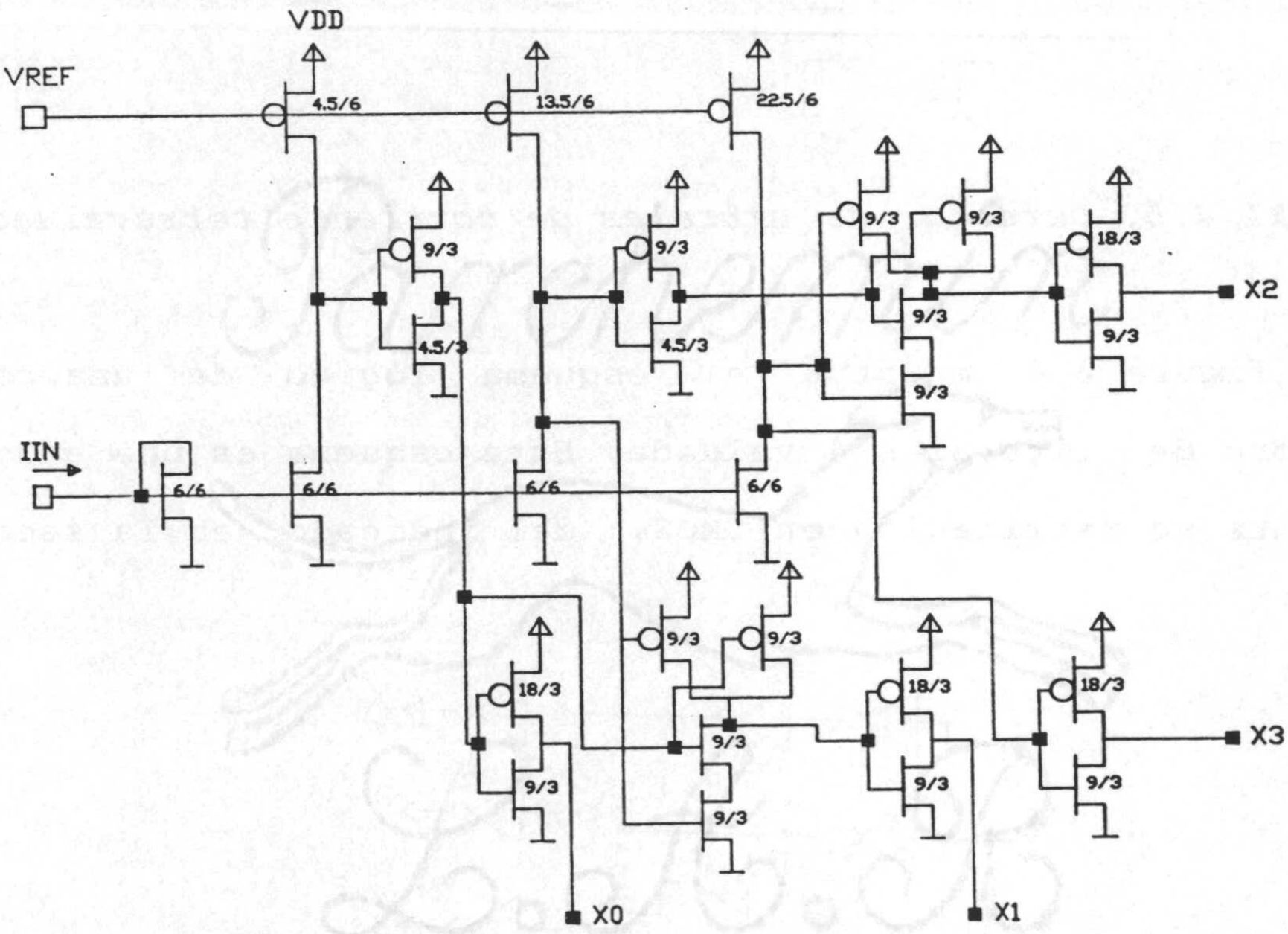


FIGURA 4.7: Esquema eléctrico de la celda generadora de literales tetravaluada tipo NAND (dimensiones de terminales puerta en micrómetros).

corriente podemos reducirlo utilizando algún esquema de conmutación de la fuente de referencia (por ejemplo el propuesto en la figura 4.8), de tal manera que se consuma corriente sólo en los tiempos de evaluación de las entradas. Cuando C=1,

el circuito de la figura 4.8 genera una tensión de referencia V_{REF} que es función de las dimensiones relativas de los transistores MN1, MN2, y MP1. Si C=0, MN2 permanece en corte y MP2 hace que la tensión de salida sea próxima a V_{DD}, anulando la corriente a través de todos los transistores PMOS conectados en paralelo con MP1.

Un esquema lógico alternativo de celda generadora de literales tetravaluada, basado en puertas NOR, se muestra en la figura 4.8. A la celda básica se han añadido entradas de control para anular todas sus salidas (mediante la línea de control X_C) o sólo algunas de ellas (mediante las líneas de control X_{Ci} independientes). Este tipo de celda será la utilizada, con la línea de control común, en las realizaciones de PLAs dinámicas (apartado 4.3.2). La activación selectiva de

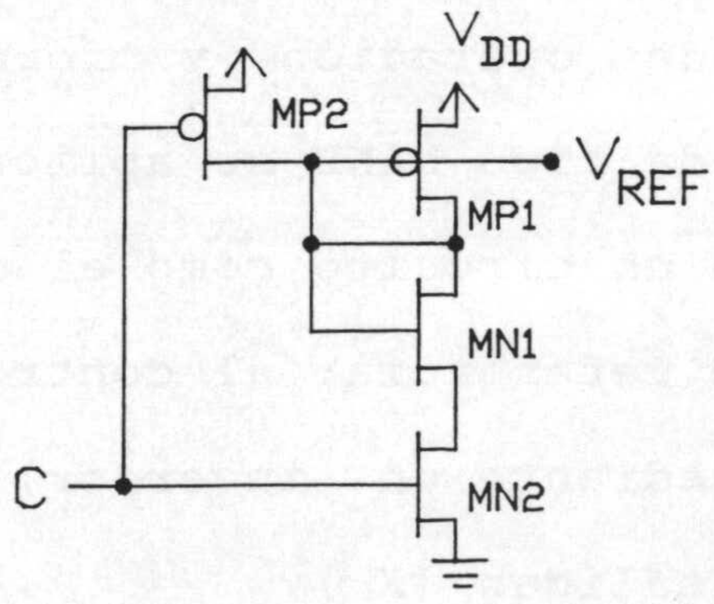


FIGURA 4.8: Circuito generador de V_{REF} conmutable

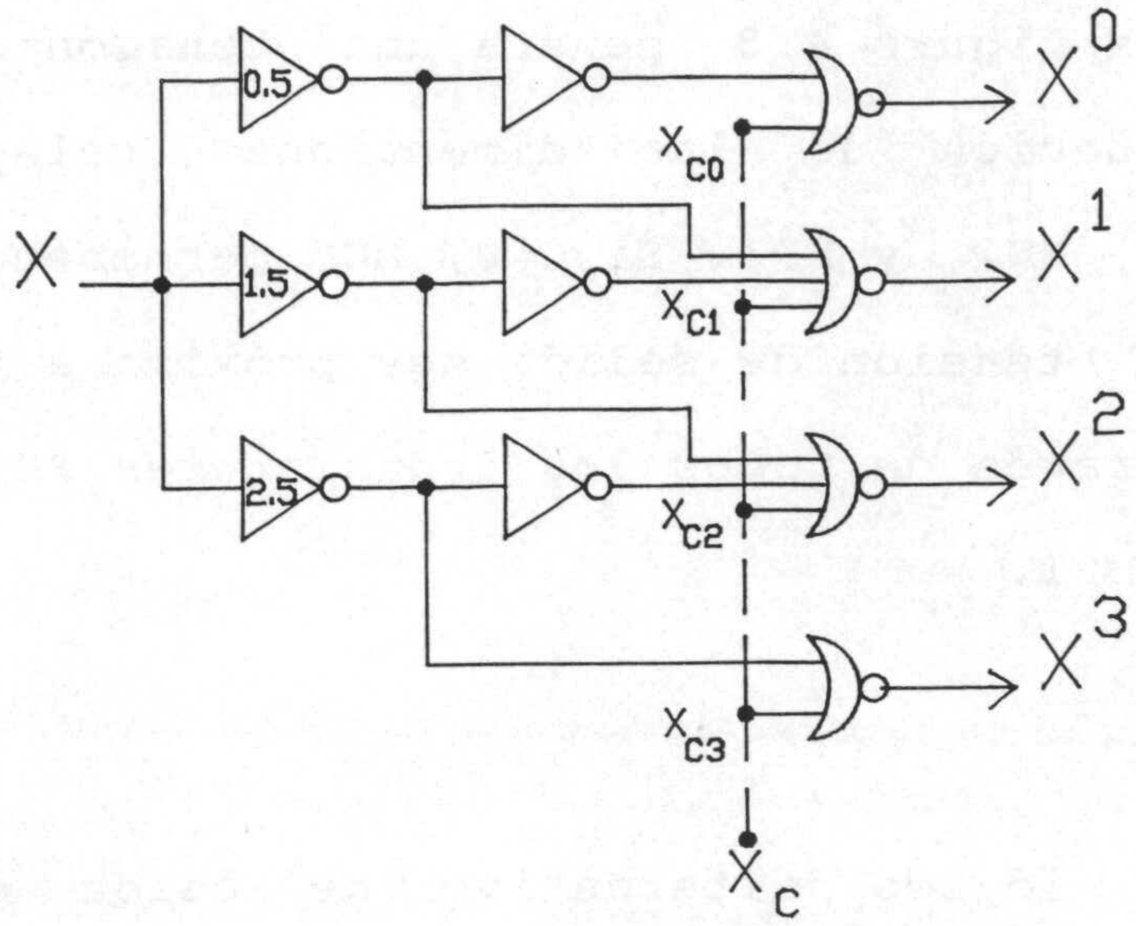


FIGURA 4.9: Esquema lógico de la celda generadora de literales tetravaluada tipo NOR.

las líneas de control es necesaria para posibilitar el procedimiento de test propuesto en el capítulo 5.

En la figura 4.10 se muestra un esquema eléctrico de la celda generadora de literales tipo NOR. Los mismos comentarios referentes a velocidad de operación y forma de reducir el consumo hechos para la celda tipo NAND se aplican a esta celda. En este caso, si se emplea un circuito como el de la figura 4.8 para conmutar la fuente de referencia, el control de este puede obtenerse directamente (mediante un inversor) a partir de la señal de anulación de las salidas (X_C).

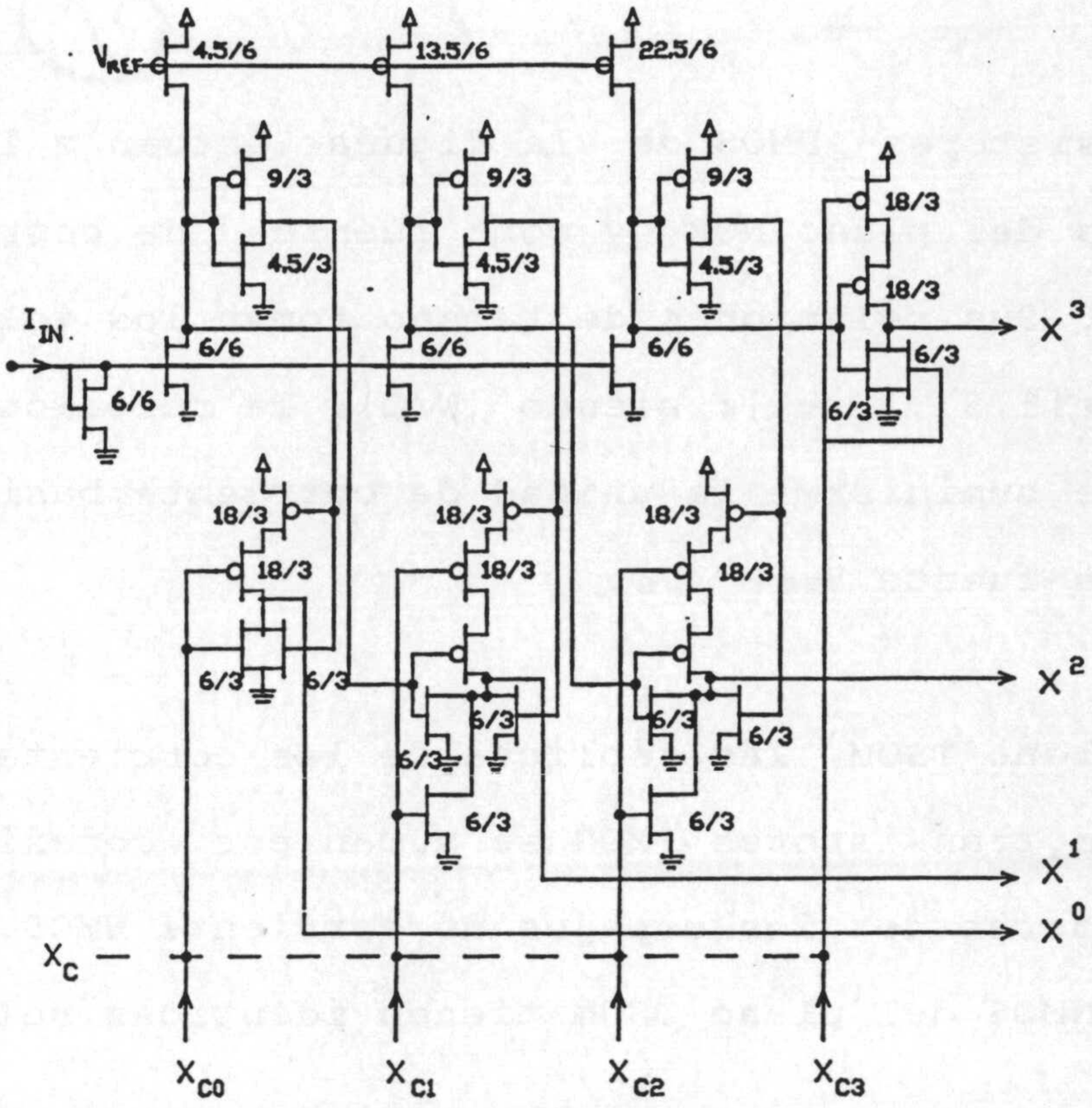


FIGURA 4.10: Esquema eléctrico de la celda generadora de literales tetravaluada tipo NOR (dimensiones de terminales de puerta en micrómetros).

4.3.2 Planos de la PLA.

Consideraremos realizaciones seudo-NMOS y CMOS dinámica. En ambos casos, los planos NOR tienen la forma de los correspondientes binarios [127, 193].

La figura 4.11 muestra una posible realización seudo-NMOS de los planos de la PLA multivaluada propuesta en la sección 4.1. Utilizaremos la denominación PLA del tipo 1 para referirnos a esta realización.

Los transistores PMOS de la figura actúan a la vez como cargas activas del plano NOR, y como fuentes de corriente para el plano TSUM. Sus relaciones de tamaño toman los valores $c_j.(W/L)_0$, $c_j \varepsilon \{1,2,..,r-1\}$; siendo $(W/L)_0$ la relación de tamaño de un PMOS que suministre la unidad de corriente básica con una tensión puerta-fuente $V_{REF}-V_{DD}$.

En el plano TSUM, las réplicas de las corrientes suministradas por los transistores PMOS se suman por conexión directa de los drenadores de los espejos de corriente NMOS. Todos los transistores NMOS del plano TSUM tienen idénticas relaciones de tamaño.

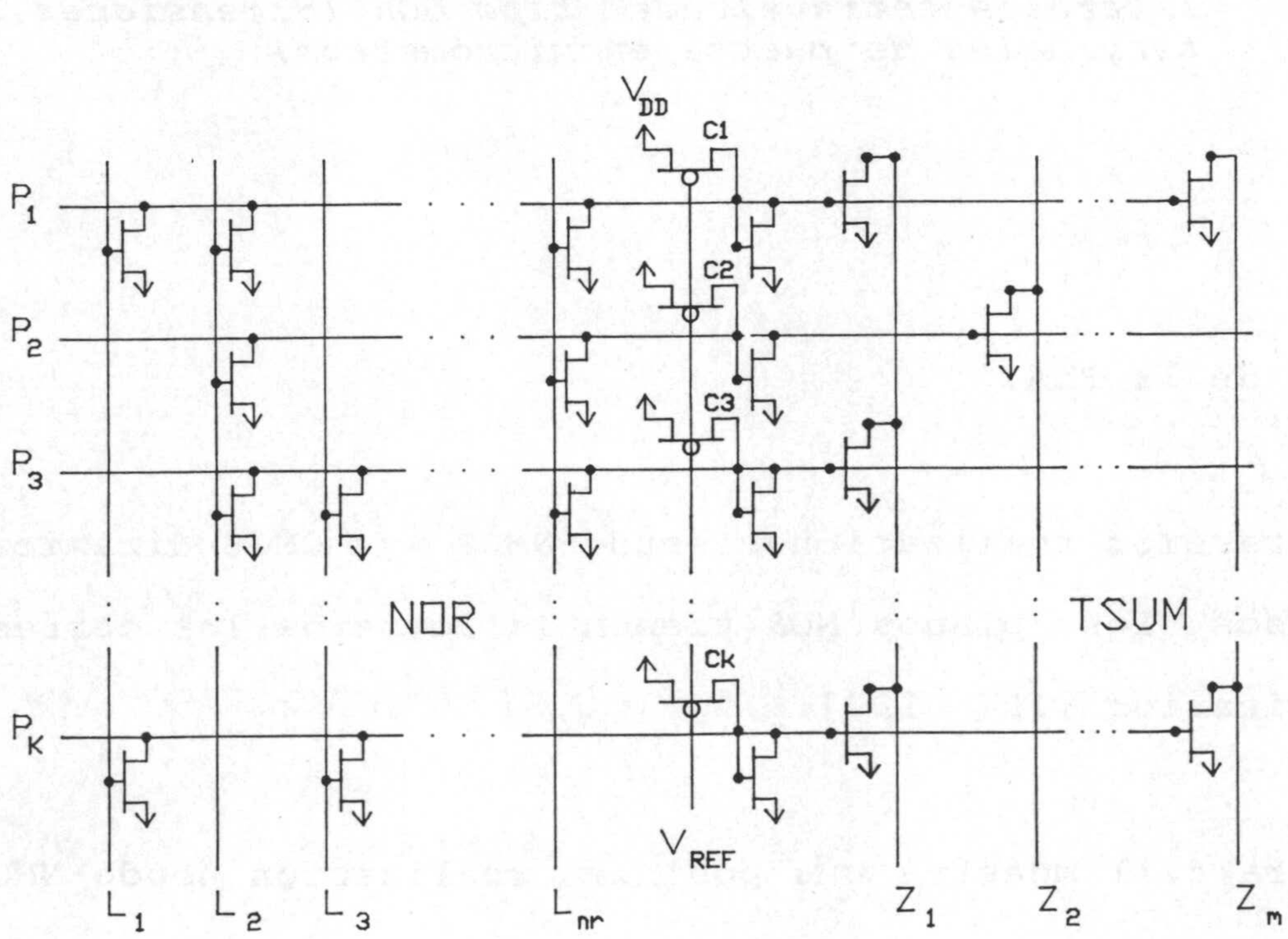


FIGURA 4.11: Planos de la PLA del tipo 1.

La programación de los planos de esta PLA puede realizarse

sobre la máscara de difusiones n^+ para definir los transistores NMOS de los puntos de cruce de ambos planos, y sobre la máscara de difusiones p^+ para fijar los distintos tamaños de transistores PMOS (la programación de la PLA afecta a dos máscaras). Otra posibilidad es efectuar la programación sobre una sola máscara, la de agujeros de contacto, definiendo en este caso las fuentes de valor c_j como c_j transistores independientes en paralelo.

Una PLA del tipo 1 tiene la ventaja de ser la que menos superficie ocupa y no requerir ningún mecanismo adicional de temporización. Como celdas generadoras de literales se pueden utilizar celdas de tipo NAND (ver apartado 4.3.1). El principal inconveniente que presenta es su relativamente baja velocidad de funcionamiento cuando la PLA es de gran tamaño.

Dado que no existe una etapa de separación entre los planos NOR y TSUM de la PLA, la capacidad asociada a cada línea producto (C_{Pj}) es la suma de las capacidades de la línea en ambos planos.

En las transiciones, cada transistor PMOS debe cargar la capacidad C_{Pj} con una corriente $c_j I_0$. El tiempo de transición es directamente proporcional a C_{Pj}, y es máximo cuando $c_j=1$. Este tiempo puede calcularse mediante la expresión (2.39) (ver apartado 2.3.3), añadiéndo a este valor el tiempo necesario para cargar C_{Pj} desde cero hasta V_{TN}.

La corriente suministrada por los transistores PMOS del esquema de la figura 4.11, o es conmutada a tierra en el plano NOR, o es introducida en los espejos NMOS del plano TSUM. Por tanto, el consumo de potencia de los planos de la PLA tipo 1 puede estimarse como:

$$P_1 = V_{DD} \cdot I_0 \cdot \sum_{1}^{K} c_j \qquad (4.22)$$

En la expresión anterior no se incluye el consumo debido a las corrientes extraidas (por los espejos NMOS) de las líneas de salida. Este consumo depende del número de salidas y del valor lógico que tomen estas para cada combinación de entradas.

Considerando por ejemplo una PLA que realice 100 términos producto de valor unidad (cj=1), el consumo de potencia dado por (4.22), con V_{DD}=5V, está comprendido entre 5mwat y 25mwat cuando se utilizan niveles de corriente unidad I_0 entre 10µA y 50µA.

Podemos concluir que, si se quiere mantener limitado el consumo de potencia, el esquema de la figura 4.11 es válido para PLAs de tamaño reducido. De no ser así, la PLA resultante tendrá un tiempo de respuesta excesivo. Debe tenerse en cuenta que, aparte del problema del consumo, la magnitud de los niveles de corriente utilizados no puede aumentarse por encima de cierto valor sin provocar un mal funcionamiento de los transistores del plano TSUM. Para un valor dado de la tensión de alimentación, y unas dimensiones concretas de los transisto-

res que forman el plano TSUM, la magnitud de la corriente debe ser lo suficientemente pequeña como para garantizar el funcionamiento de dichos transistores en la región de saturación (ver apartado 2.3.1).

Un esquema alternativo a la PLA del tipo 1 se muestra en la figura 4.12. Esta realización de PLA, que denominaremos PLA del tipo 2, incorpora una etapa de transistores que separa los planos, y cargas activas en las líneas producto dentro del plano NOR. A costa de un aumento del número de dispositivos, esta realización tiene como ventaja una mayor velocidad de funcionamiento que la PLA del tipo 1 (para un mismo valor de I_0).

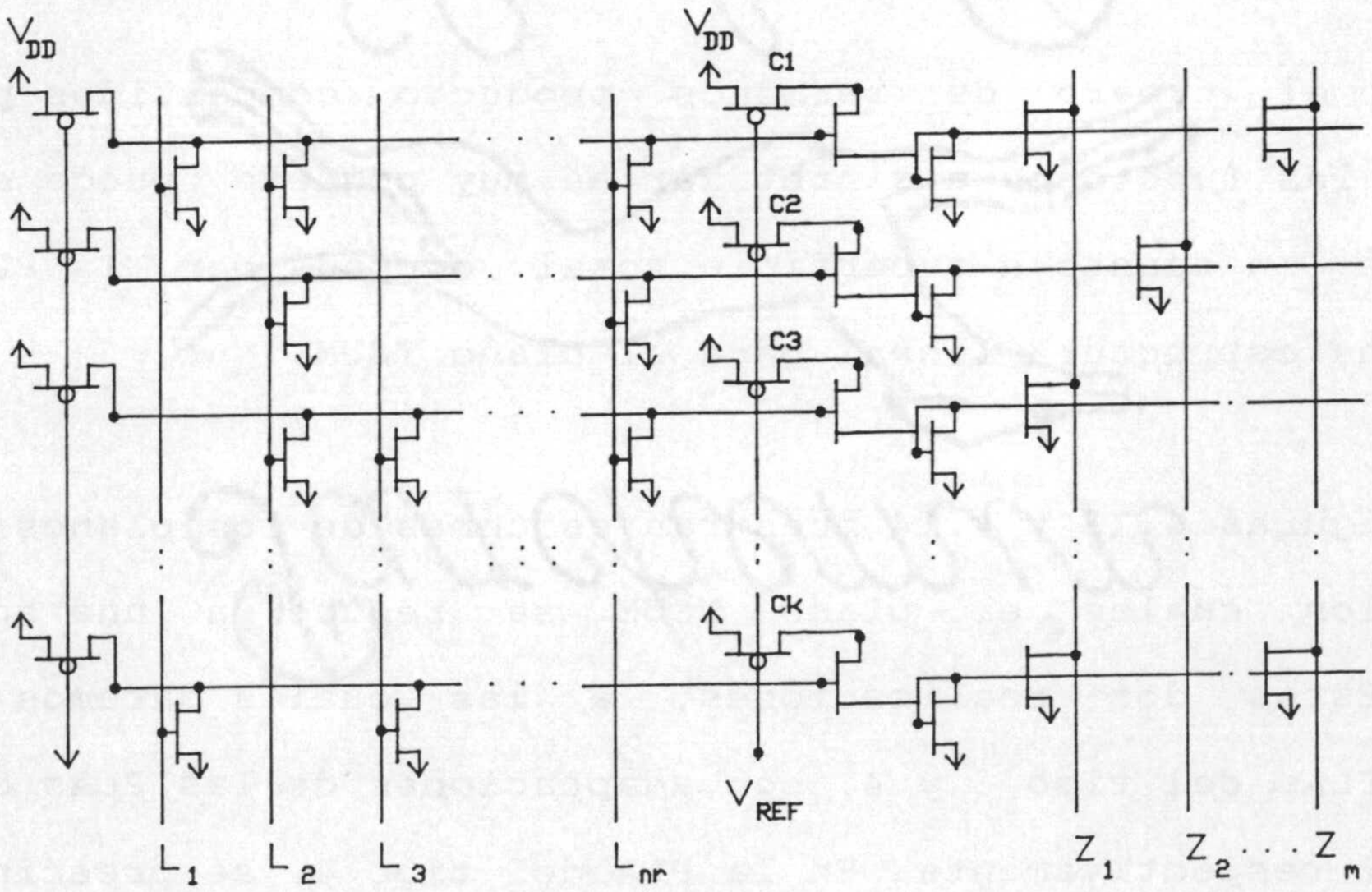


FIGURA 4.12: Planos de la PLA del tipo 2.

Las fuentes de corriente PMOS deben en este caso cargar

sólo las capacidades asociadas a la prolongación de las líneas producto en el plano TSUM, mientras que la carga de las líneas producto en el plano NOR es realizada por los transistores PMOS añadidos a dicho plano.

Como puede observarse en el esquema de la figura 4.12, de todos los transistores PMOS que actuan como fuentes de corriente, sólo producirán consumo de potencia aquellos que, para cada combinación de entradas, deban contribuir a las salidas. El consumo asociado al conjunto de las fuentes de corriente será en general bastante inferior al dado por (4.22). No obstante, el cosumo producido por los elementos añadidos (cargas PMOS del plano NOR) hace que el consumo total de esta PLA sea superior al de una realización del tipo 1.

Cuando el número de términos producto compartidos por varias de las funciones a sintetizar es muy pequeño, puede ser conveniente, en cuanto a superficie total ocupada por la PLA, utilizar una estructura lineal para el plano TSUM.

Las figuras 4.13 y 4.14 muestran esquemas de los planos de la PLA en los cuales el plano TSUM se reduce a una sola columna. Estas dos realizaciones, a las cuales daremos el nombre de PLAs del tipo 3 y 4, son adaptaciones de las PLAs del tipo 1 y 2 respectivamente. En la PLA del tipo 4, se prescinde de los espejos de corriente NMOS. La corriente en las líneas de salida es de sentido contrario al resto de las realizaciones consideradas.

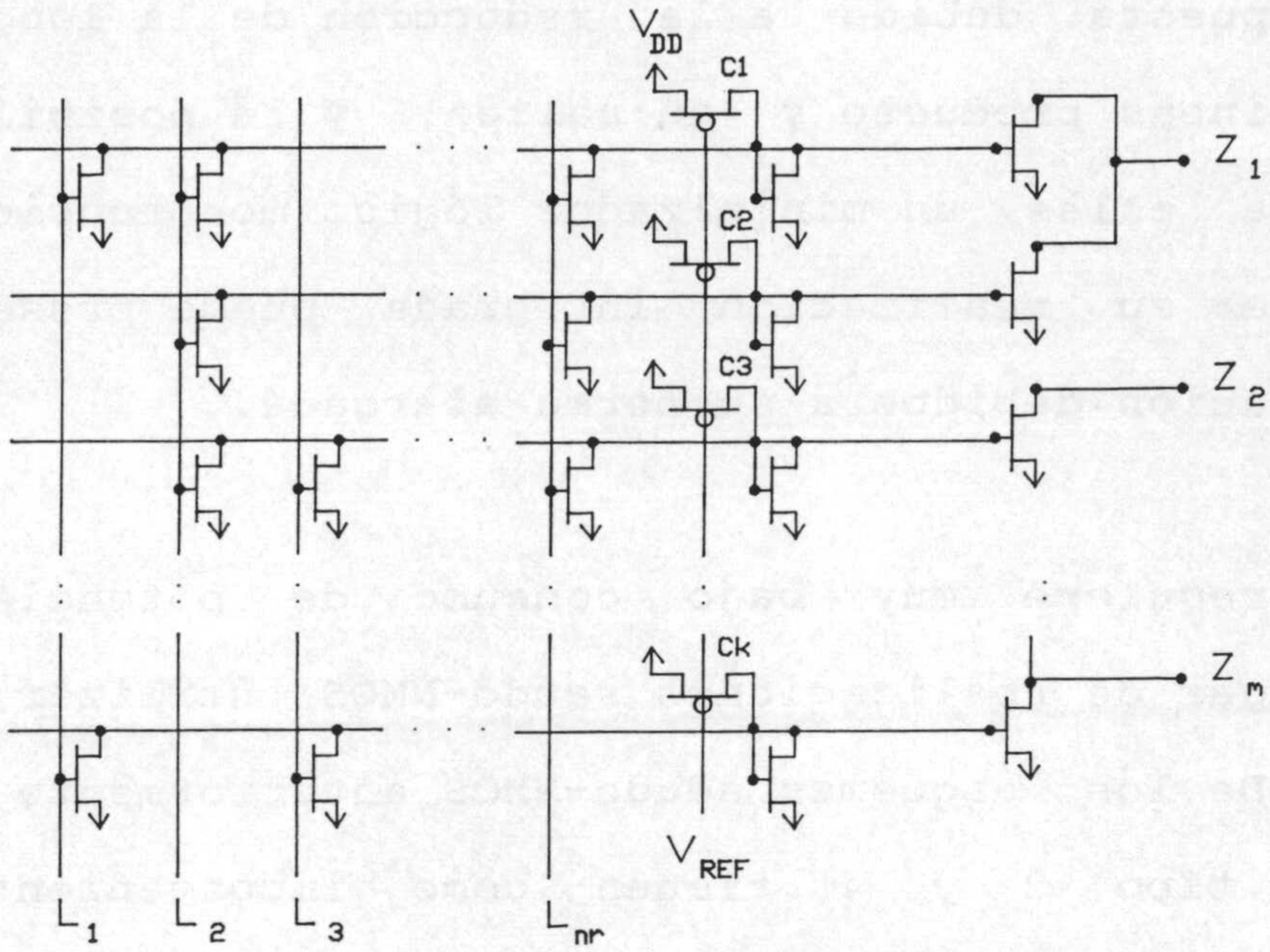


FIGURA 4.13: Planos de la PLA del tipo 3.

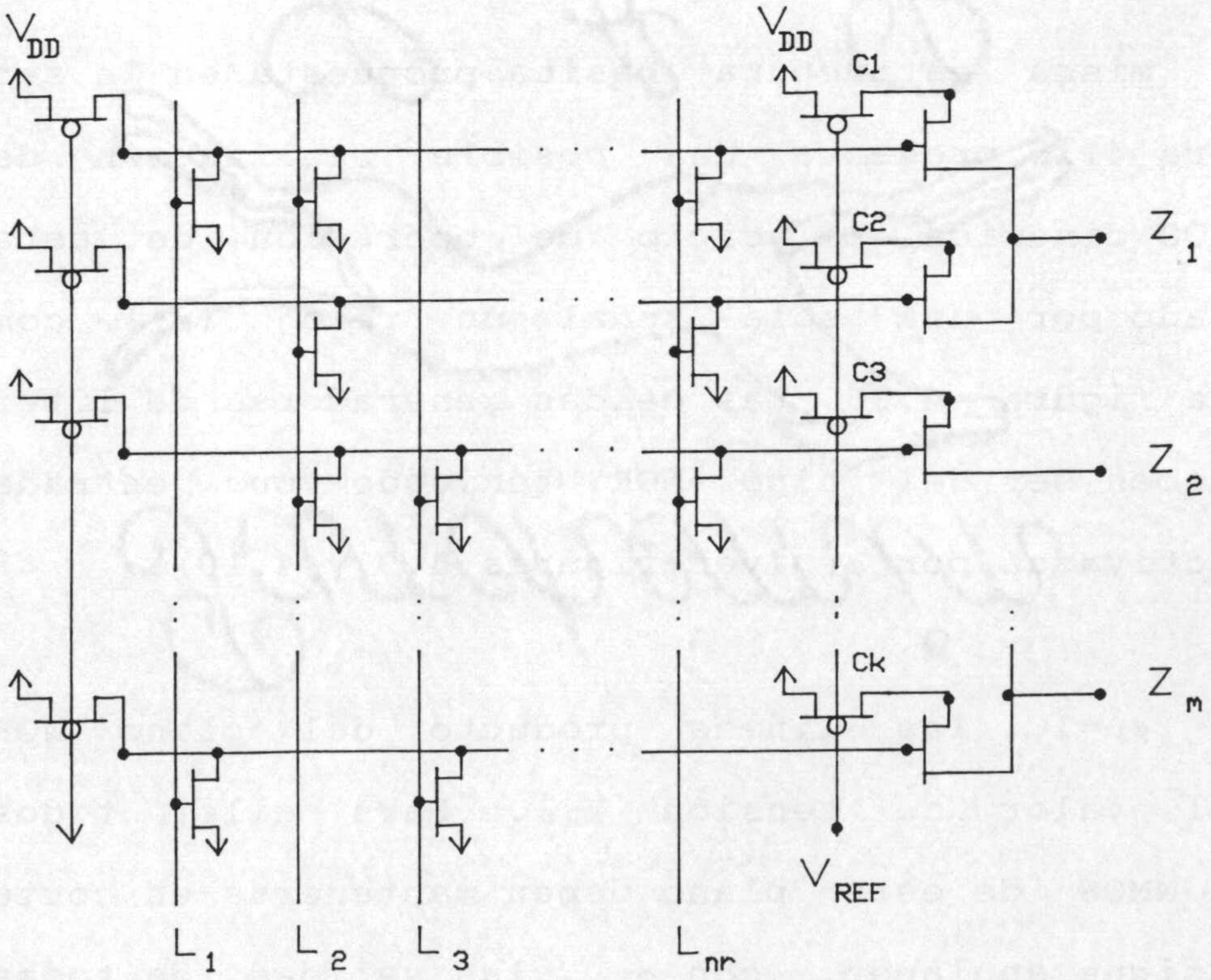


FIGURA 4.14: Planos de la PLA del tipo 4.

Las PLAs del tipo 3 y 4 tienen como ventajas una mayor velocidad de respuesta, debido a la reducción de la longitud efectiva de las líneas producto y de salida, y la posibilidad de utilizar para ellas un minimizador lógico monofuncional. Como inconveniente, su realización integrada puede presentar problemas de ubicación debidos a su forma alargada.

Cuando se requiere muy bajo consumo de potencia es preferible, en lugar de realizaciones seudo-NMOS, utilizar PLAs CMOS dinámicas. De los esquemas seudo-NMOS anteriormente propuestos, los del tipo 2 y 4 tienen como inconveniente el elevado consumo de sus planos NOR, el cual coincide con el producido por el plano NOR de una PLA seudo-NMOS binaria con el mismo número de líneas de literal y de producto.

Con la misma estructura básica propuesta en la sección 4.1, la figura 4.15 presenta una posible realización de PLA r-valuada CMOS dinámica. El ciclo de operación de esta PLA está controlado por una sola señal de reloj (ϕ_1) como se indica en la figura 4.16. Las celdas generadoras de literales utilizadas deben ser del tipo NOR teniendo sus entradas de control X_C activadas por ϕ_1 (ver figuras 4.9 y 4.10).

Durante $\phi_1=1$, las líneas producto del plano NOR se precargan al valor de tensión V_{DD}. Para ello, todos los transistores NMOS de este plano deben mantenerse en corte, lo cual se consigue anulando, con ϕ_1, las salidas de todas las celdas generadoras de literales (se ponen todas las líneas de

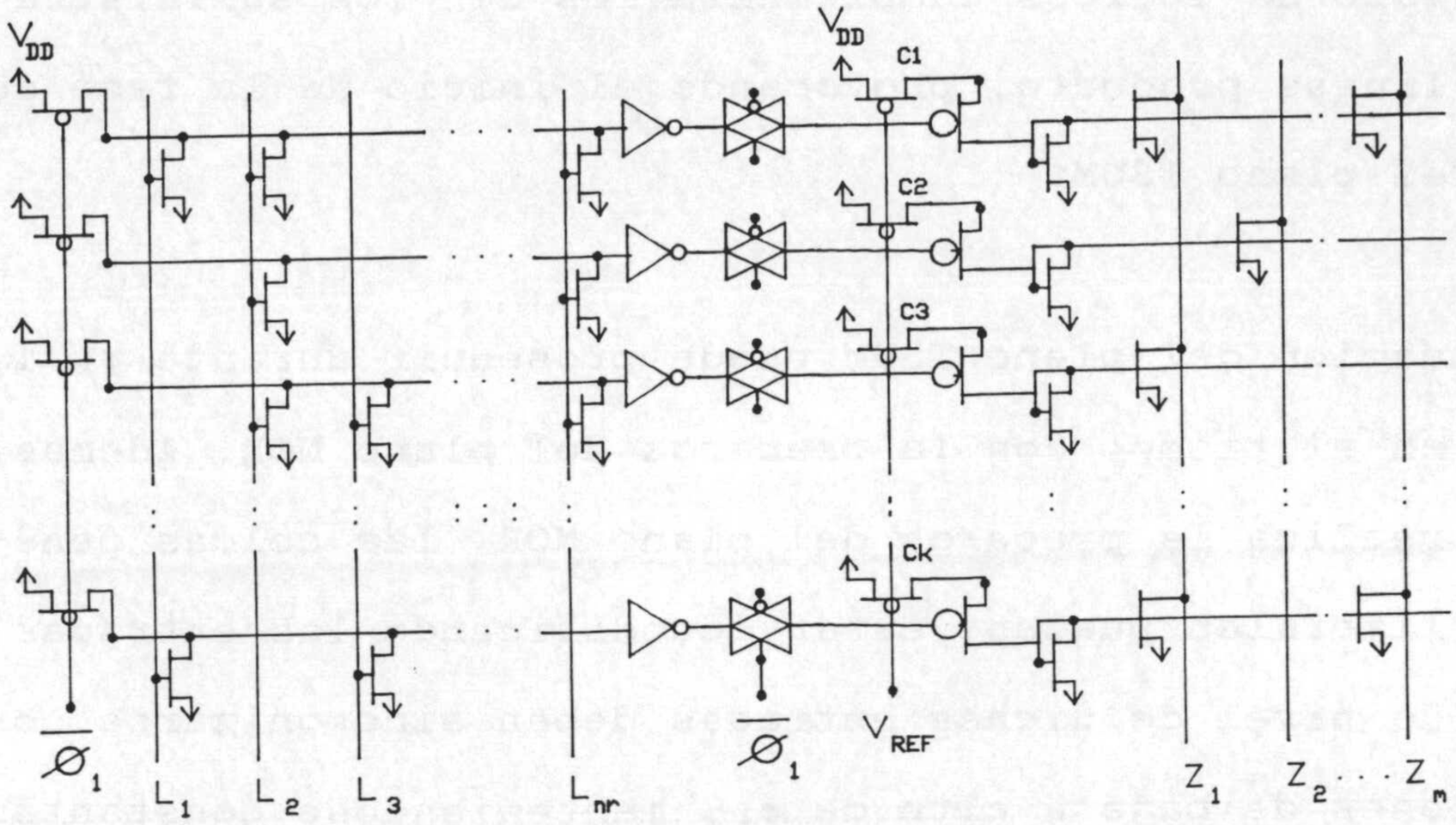


FIGURA 4.15: Planos de la PLA del tipo 5.

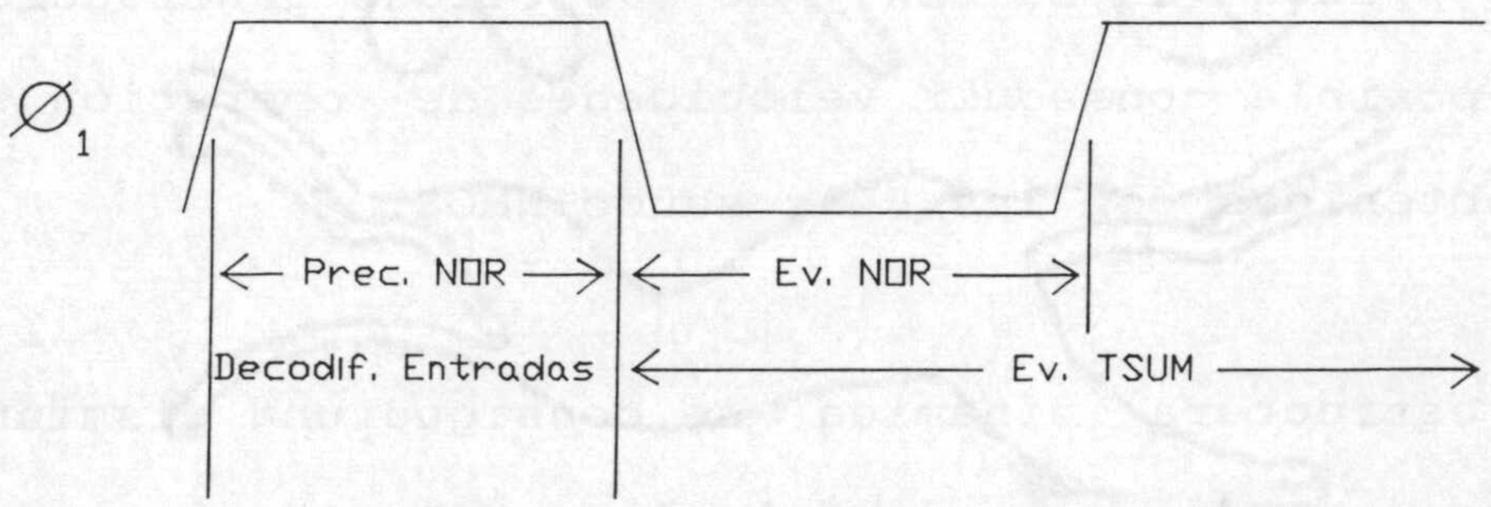


FIGURA 4.16: Señal de temporización de la PLA del tipo 5.

literal a 0).

Cuando ϕ_1 efectúa la transición a baja se inicia la evaluación de los planos NOR y TSUM. El plano TSUM permanece aislado del plano NOR durante el tiempo en que ϕ_1=1. Cuando ϕ_1

efectúa la transición de alta a baja, las puertas de los transistores PMOS que conmutan las fuentes de corriente PMOS se precargan a valores lógicos complementarios de los suministrados por las líneas producto, provocando el inicio de la fase de evaluación del plano TSUM.

La evaluación del plano TSUM puede proseguir durante $\phi_1 = 1$, solapandose en el tiempo con la precarga del plano NOR. Además, mientras se realiza la precarga del plano NOR, las celdas generadoras de literales pueden estar decodificando las entradas. Los cambios de nivel de dichas entradas deben sincronizarse con las transiciones de baja a alta de ϕ_1, manteniendose constantes el resto del tiempo.

Dado que se consigue cierto paralelismo en el funcionamiento de los planos de la PLA y de las celdas generadoras de líterales, es posible conseguir velocidades de operación superiores a las obtenidas con las PLAs seudo-NMOS.

Con la estructura dinámica se consigue una disminución drástica del consumo de potencia del plano NOR respecto de las realizaciones del tipo 2 y 4. Las fuentes de corriente que conectan con los espejos NMOS del plano TSUM, producen un consumo similar al obtenido para una PLA del tipo 2 dado que sólo se conmutan (para cada combinación de entradas) aquellas fuentes de corriente que han de contribuir a las salidas.

Como inconveniente, la PLA del tipo 5 es la que necesita

mayor superficie para su integración, no sólo debido a los inversores y puertas de paso introducidas entre sus planos, sino también porque requiere el uso de celdas generadoras de literales del tipo NOR (de mayor tamaño que las del tipo NAND), así como de la lógica necesaria para su temporización.

Si se trata de sintetizar funciones que no comparten términos producto es posible simplificar el plano TSUM de la PLA del tipo 5, adoptando la forma indicada en la figura 4.17. Utilizaremos la denominación de PLA del tipo 6 para esta realización de la PLA multivaluada. Las puertas de paso

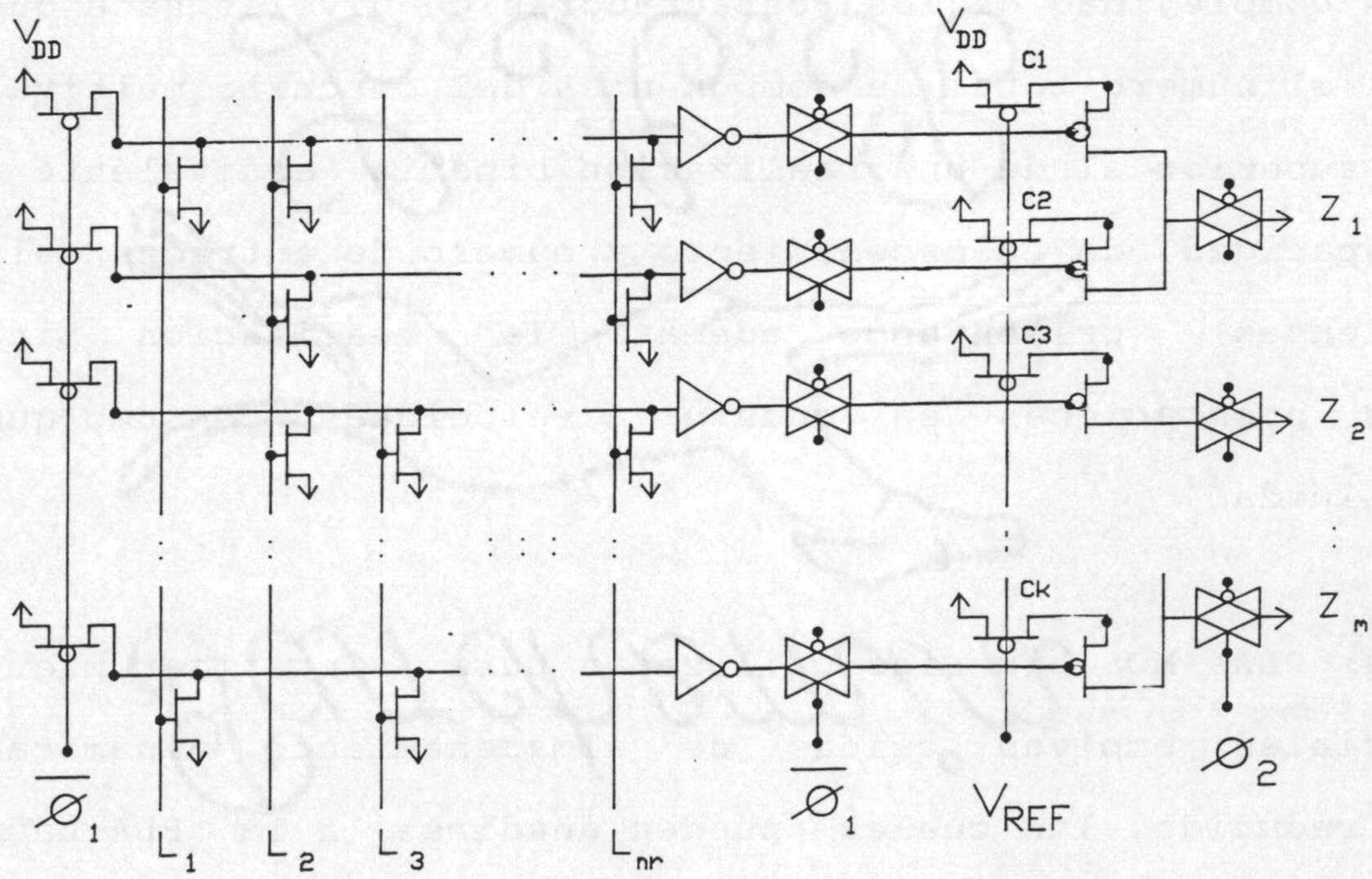


FIGURA 4.17: Planos de la PLA del tipo 6.

añadidas a las salidas en el esquema de la figura 4.17 están controladas por una fase ϕ_2 (la cual puede coincidir con ϕ_1) que determina el tiempo en que las fuentes de corriente PMOS deben estar conmutadas a las salidas.

4.3.3 Celdas de memoria en modo de corriente.

La mayoría de las realizaciones de elementos de memoria multivaluados propuestas por diversos autores se basan en la utilización de restauradores de niveles realimentados [27, 42, 66, 67, 74, 86].

La complejidad de los restauradores de niveles hace que en general el número total de componentes del circuito multivaluado sea superior al de una realización binaria equivalente (con una capacidad de almacenamiento y número de entradas/salidas equivalentes), presentando además la realización binaria mejores prestaciones en cuanto a velocidad y consumo que la multivaluada.

Las PLAs MOS binarias utilizadas para sintetizar máquinas secuenciales emplean celdas de almacenamiento dinámicas de tamaño reducido, las cuales pueden añadirse a la PLA básica, como celdas de salida, sin problemas de ubicación ni conexionado [127, 193].

Podemos considerar dos formas de ampliar la PLA multivaluada que se propone, para permitir con ella la síntesis completa de sistemas secuenciales multivaluados. Una posibilidad es decodificar las variables de estado siguiente (generadas por la PLA como funciones multivaluadas de las entradas externas y de las variables de estado presente) mediante celdas generadoras de literales, e intercalar celdas de registro binarias entre las salidas de estas celdas y las correspondientes líneas de literal. Esta solución exige r celdas de registro por cada variable de estado r-valuada.

La otra posibilidad es utilizar celdas de memoria r-valuadas (una por cada variable de estado). Esta solución supone un ahorro considerable en el número de elementos de memoria necesarios, pero sólo es preferible a la solución binaria si cada celda de memoria multivaluada tiene un tamaño muy reducido.

En la figura 4.18 presentamos dos esquemas de celdas de memoria multivaluadas CMOS dinámicas, las cuales presentan un tamaño muy reducido que es independiente del de la base r de la lógica. Cada una de ellas está constituida por una pareja de transistores (M1 y M2), conectados a modo de espejo de corriente, y separados por una puerta de transmisión.

La corriente en la entrada D de la celda, con el sentido indicado en la figura, produce una tensión en la puerta del transistor M1:

$$V_G \approx V_{DD} - (I_D/\beta_P)^{1/2} - |V_{TP}| \qquad (4.23)$$

ó

$$V_G \approx (I_D/\beta_N)^{1/2} - V_{TN} \qquad (4.24)$$

según se trate de una celda PMOS (figura 4.18.a) o NMOS (figura 4.18.b).

En la fase de carga (ø=1), la capacidad de puerta del transistor de salida M2 también se carga al mismo valor de tensión, cambiando su corriente de drenador al valor I_D si los transistores M1 y M2 son idénticos.

En la fase de almacenamiento (ø=0), la puerta de M2 permanece aislada, manteniendose constante la corriente de salida I_Q.

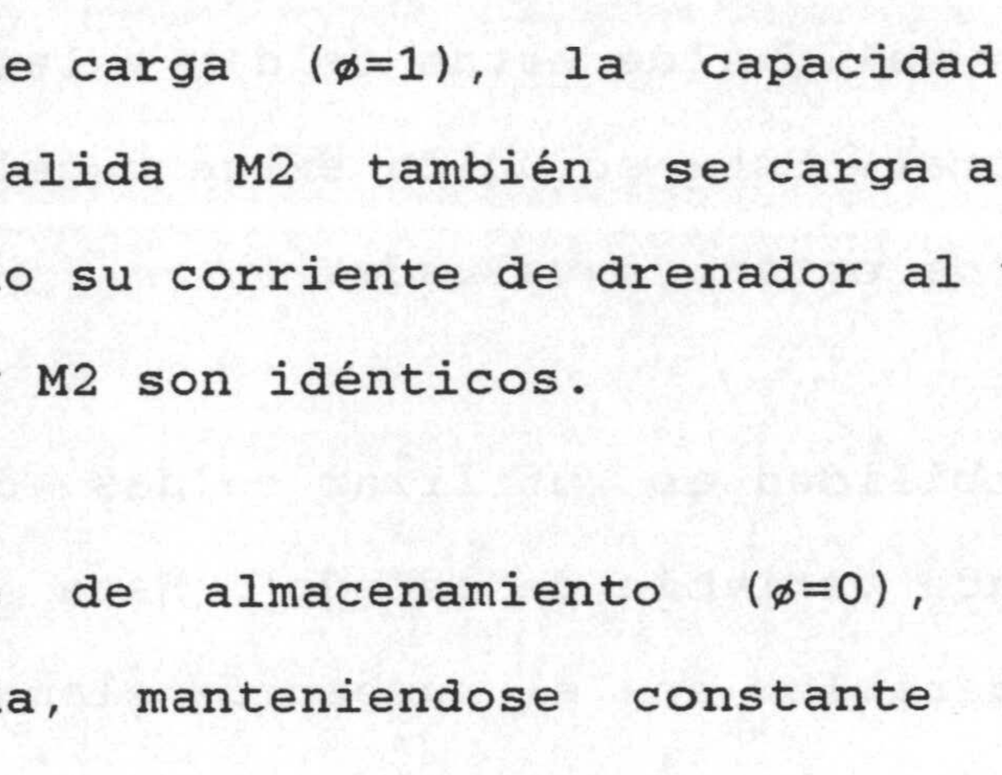


FIGURA 4.18: Celdas de memoria en modo de corriente. (a) Celda PMOS. (b) Celda NMOS.

Los elementos de memoria presentados funcionan en realidad como celdas "analógicas", ya que, salvo errores debidos a

asimetrías entre los transistores del espejo o a pérdidas de carga a través de la puerta de paso, la corriente de salida coincide con el valor analógico de la corriente de entrada previamente introducida, sin que se lleve a cabo ninguna restauración intermedia de niveles lógicos.

Al utilizarlos como celdas de registro en una PLA multivaluada, la discriminación de niveles se realiza en la celdas generadoras de literales a las que conecten sus salidas Q.

Si los planos de la PLA multivaluada tienen una realización del tipo 4 o del tipo 6, los elementos de memoria utilizados pueden reducirse a simples puertas de paso CMOS. El esquema de la figura 4.17 es por sí solo suficiente para la realización de máquinas secuenciales multivaluadas. Para ello debe hacerse coincidir la fase ϕ_2 de dicha figura con la fase que temporiza la precarga del plano NOR (ϕ_1), estando los cambios de las entradas multivaluadas sincronizados con las transiciones de baja a alta de ϕ_1.

La utilización de los elementos de memoria propuestos en la síntesis de circuitos secuenciales no basados en PLAs, tales como registros y contadores multivaluados, está limitada por el posible deterioro de niveles de corriente cuando se conectan varias celdas en cascada sin que tenga lugar alguna restauración local de niveles logicos.

En el capítulo 6 (sección 6.4) incluimos los resultados de

simulación con SPICE de las celdas de memoria propuestas. Los resultados concuerdan plenamente con los planteamientos efectuados en este capítulo.

4.4 Conclusión.

En el presente capítulo se ha propuesto una nueva estructura de PLA multivaluada, la cual es realizable en tectonogía CMOS mediante un proceso estándar de fabricación de circuitos binarios.

La superficie ocupada por la realización física de la PLA puede ser inferior a la de una realización binaria equivalente, dependiendo de cual sea el número total de términos producto requeridos en uno y otro caso. Bajo la condición de igual número de términos producto al codificar las entradas y salidas en distintas bases, se obtiene que las lógicas trivaluada o tetravaluada son las que llevan a realizaciones integradas de menor tamaño.

Hemos realizado programas de minimización para la obtención automática de la particularización óptima de la PLA propuesta. La evaluación de las soluciones obtenidas por estos programas sobre un total de 1600 funciones generadas aleatoriamente indica que el procedimiento de minimización presentado es en media más eficiente que el procedimiento propuesto por Dueck

y Miller [34]. La elección, para cada función, entre las soluciones proporcionadas por los tres programas implementados, puede suponer reducciones considerables en el número de términos producto necesarios.

Se han propuesto distintas realizaciones CMOS de la PLA multivaluada, así como de los elementos de memoria necesarios para la síntesis completa de sistemas secuenciales multivaluados. En el capítulo 6 se aportan resultados de la simulación eléctrica de estos circuitos, comprobándose la validez de los mismos.

CAPITULO 5

TEST DE PLAs MULTIVALUADAS

En este capítulo se presenta un procedimiento de test para la estructura de PLA multivaluada propuesta en el capítulo anterior.

En la sección 5.1 se ponen de manifiesto los problemas relacionados con el test funcional de circuitos integrados y la importancia que tiene idear mecanismos para simplificarlo.

La sección 5.2 se dedica al test funcional de las PLAs, resumiéndose las características más relevantes de los esquemas existentes para PLAs binarias.

El procedimiento de test ideado para PLAs multivaluadas se presenta en la sección 5.3, procediéndose a un análisis previo de los modelos apropiados de fallos, y concluyendo con

la evaluación de la cobertura de fallos obtenida.

La sección 5.4 es una conclusión.

5.1 Introducción

A lo largo del proceso de diseño y fabricación de un circuito integrado se ha de llevar a cabo un gran número de comprobaciones, las cuales tienen por objeto determinar si el producto final funciona de acuerdo con las especificaciones inicialmente fijadas.

Refiriéndonos a la fase de diseño, las comprobaciones son necesarias para verificar que, en cada uno de los niveles de descripción del circuito (estructural, lógico, eléctrico, y de descripción gráfica de máscaras) no existen fallos de concepción que puedan ocasionar un mal funcionamiento del prototipo final. Se trata con ello de evitar repetir el proceso de integración de prototipos, cuyo coste es muy elevado.

Los programas incluidos en los paquetes de software de ayuda al diseño de circuitos integrados (tales como simuladores funcionales, lógicos, temporales, y eléctricos, verificadores de correspondencia, verificadores de reglas geométricas de diseño, etc.) hacen abordable el problema de comprobación de los diseños de circuitos integrados LSI/VLSI, en cada una de

sus etapas.

La fabricación del circuito integrado introduce una serie de defectos que pueden ocasionar el mal funcionamiento del mismo. La miniaturización de los componentes y la elevada densidad de integración en circuitos VLSI lleva a fallos inevitables que, en algunas etapas del proceso de fabricación, afectan a un gran porcentaje del total de los circuitos procesados (entre un 20% y un 30% o más [29, 97]).

Es bastante dificil, dada la complejidad de los circuitos VLSI, comprobar de manera efectiva y económica si estos circuitos integrados han sido fabricados correctamente, y realizan sin errores las funciones para las que fueron diseñados (test funcional). La causa fundamental de este problema es la baja *controlabilidad* y *observabilidad* de estos sistemas, debidas a la muy alta relación (número de componentes lógicos internos)/-(número de términales de entrada-salida) que presentan.

La *controlabilidad* mide la facilidad para establecer un valor dado de señal en una determinada entrada de un componente del circuito actuando sobre las entradas primarias o externas del mísmo, y la *observabilidad* mide la facilidad para determinar, a partir de las salidas externas del circuito, qué ocurre en las salidas de un determinado componente interno del mismo [177].

Sólo debido al elevado número de componentes lógicos

internos, intentar llevar a cabo un test completo o exhaustivo del circuito integrado fabricado es ya inabordable para circuitos LSI. Esta dificultad puede ilustrarse con el ejemplo citado en [199]: "teóricamente, el test funcional completo de un circuito formado por una red combinacional con 25 entradas y 50 biestables, suponiendo que la aplicación de patrones de test se efectue a razón de 1 microsegundo por patrón, llevaría al menos 1000 años".

Como primera medida para posibilitar el test se recurre a establecer un conjunto de *modelos de fallos*, y a desarrollar métodos que, en tiempos razonables, sean capaces de detectar con una buena probabilidad de acierto uno o varios de los tipos de fallos. Tradicionalmente se han considerado fallos simples del tipo anclaje ("stuck-at") [3]. Los referidos métodos proceden al test desarrollando dos tareas:

1.- Generación automática de secuencias de test: obtención de un conjunto de vectores de test, lo más reducido posible, óptimo para la detección del mayor número de fallos del tipo o tipos elegidos.

2.- Verificación del test: probar que la secuencia de test generada es realmente efectiva en la detección de fallos. En general es dificil obtener pruebas formales, por lo que la alternativa para obtener una medida cuantitativa de la efectividad del test ha sido tradicionalmente la simulación de fallos mediante computadora.

El tiempo de ejecución necesario para la generación de secuencias de test y la simulación de fallos es en general proporcional a una potencia comprendida entre 2 y 3 del número de puertas lógicas del circuito sometido a test [56, 199], lo que hace este método muy costoso para circuitos de complejidad media o alta.

Más recientemente, se han investigado técnicas encaminadas a simplificar el test funcional de circuitos complejos, las cuales se basan en introducir modificaciones en los propios diseños de los circuitos integrados a fin de aumentar su controlabilidad y observabilidad, facilitando de esta manera la tarea de generación y aplicación de secuencias de test. Estas técnicas, agrupadas bajo el nombre genérico de *técnicas de diseño para la "testabilidad"* [30], pueden clasificarse en dos categorias: métodos de diseño "ad-hoc" [200], y métodos estructurados. Un resumen de las distintas técnicas de diseño para la testabilidad puede encontrarse en [199].

En cualquier caso, se supone que los vectores de test se aplican al circuito mediante una máquina o equipo de test, la cual se encarga también de observar las respuestas producidas y compararlas con las correctas que tiene almacenadas. Además, los vectores de test necesitan ser generados y almacenados para su posterior aplicación (generación "off-line"), o bien, ser calculados a medida que se aplican (generación concurrente).

Para poder prescindir de los costosos equipos de test funcional, la tendencia más actual es incorporar mecanismos de autocomprobación en los propios chips, lo cual facilita además comprobaciones posteriores de los sistemas en los que se incorporan estos circuitos. Hay que tener en cuenta que, aparte de los defectos introducidos durante la fase de diseño y fabricación del circuito, exiten otros, dependientes del tiempo, relacionados con el deterioro de las características de sus componentes internos y con el entorno en el que se encuentra funcionando [3, 44]. La inclusión en el propio chip de circuitería que proporcione los vectores de test y/o analice las respuestas de salida recibe las denominaciones alternativas de *auto-test incorporado* (BIST:"built-in self test), *test incorporado* (BIT), *auto-test*, o *test autónomo* [123]. Todas estas denominaciones se refieren a comprobaciones funcionales realizables de forma explícita (el circuito se hace funcionar en un modo específico de test), a diferencia del *auto-chequeo o auto-verificación* ("self-checking"), término este utilizado para referirse a comprobaciones implícitas o concurrentes, las cuales se efectuan sin que el circuito abandone su modo de funcionamiento normal.

La figura 5.1 muestra el esquema de auto-test usualmente utilizado. Durante el tiempo de funcionamiento en modo de test, las entradas del circuito a comprobar proceden de un generador de patrones incluido en el mismo circuito integrado. Mediante un circuito denominado compresor o compactador de datos, se extrae una característica de las respuestas del

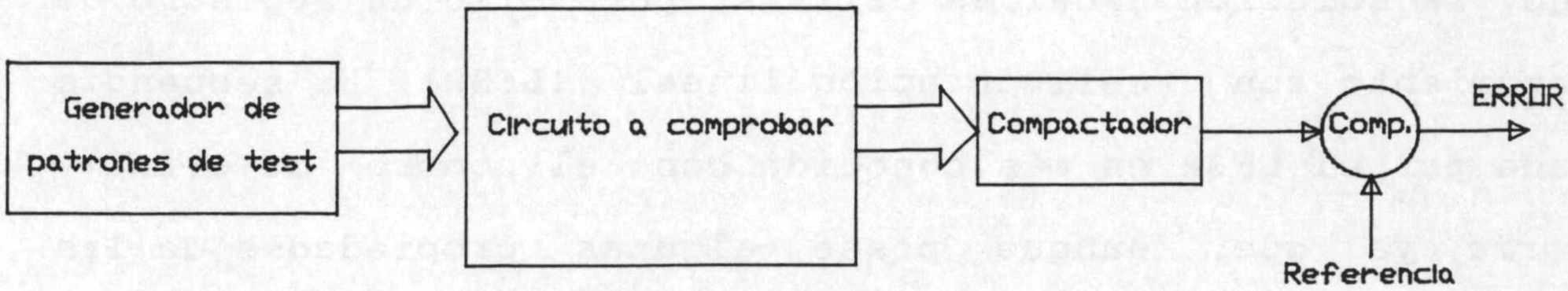


FIGURA 5.1 Esquema típico de auto-test

circuito sometido a la secuencia de patrones de test. Esta característica se compara con otra de referencia para determinar la posible existencia de fallos.

Las distintas técnicas de auto-test difieren básicamente en los esquemas utilizados para la generación de patrones de test y para la compresión de respuestas [97, 125, 159].

La generación de patrones es *exhaustiva* cuando se generan todas las posibles combinaciones de entrada al circuito a comprobar. Si el numero n de entradas al circuito es elevado, el tiempo necesario para la generación de todos los patrones (2^n) puede ser prohibitivo. Este inconveniente puede superarse si el circuito admite una partición en módulos [125], aplicándose entonces todas las posibles entradas de cada módulo (generación de patrones *seudo-exhaustiva*) [126].

Otra aproximación a la generación de patrones de test es la generación *aleatoria*. Si el chip en el que se encuentra el circuito a comprobar contiene una unidad de procesamiento, la

generación aleatoria puede realizarse algorítmicamente. Sin embargo, la solución usual es utilizar para ello un registro de desplazamiento con realimentación lineal (LFSR). La secuencia generada por el LFSR es más conocida con el nombre de *seudo--aleatoria* ya que, aunque posee algunas propiedades de las secuencias aleatorias, es determinista. El interés práctico de las técnicas de test basadas en la generación de patrones aleatoria o seudo-aleatoria depende del grado de reducción de la longitud del test (respecto de las secuencias exhaustivas) que se puede conseguir, sin excesiva perdida de cobertura de fallos [15, 166, 191].

La compresión de los datos de respuesta a la secuencia de test (ver figura 5.1) evita tener que determinar, así como almacenar en el propio chip, todas las respuestas correctas del circuito a los patrones de test para su posterior comparación. Evidentemente, toda técnica de compresión supone una reducción en la cobertura de fallos del esquema de test.

Las funciones de compresión más conocidas son la cuenta del número de unos a la salida y la división polinómica, las cuales se realizan respectivamente mediante un contador y un LFSR. El resultado de la compresión se denomina signatura. Para la primera de las funciones de compresión mencionadas, la signatura es más conocida con el nombre de "síndrome". Para la división polinómica, la signatura es el resto que se obtiene de dicha división. Las secuencias de respuestas del circuito a comprobar que producen la misma signatura (fenomeno denominado

"aliasing") se pueden caracterizar en términos de la división polinómina que realiza el LFSR [171, 198].

El estudio de la efectividad en la detección de fallos y las correspondientes necesidades hardware de las mencionadas técnicas, y de otras técnicas tales como las espectrales [84, 97], constituye en la actualidad un tema abierto e interesante de investigación.

Con lo expuesto en los párrafos anteriores hemos querido plantear los problemas que presenta el test de circuitos integrados, particularmente las dificultades relacionadas con el test funcional, indicando resumidamente algunas de las aproximaciones adoptadas para simplificarlo.

La utilidad de los circuitos que incorporan mecanismos para el test es una realidad asumida por los fabricantes de circuitos integrados, los cuales, dada la complejidad de los nuevos circuitos desarrollados, se han visto obligados a modificar sus estrategias de diseño e incorporar en ellos circuitos que faciliten el test al menos parcialmente [29,54].

La estructura regular de las PLAs ha servido para promover el estudio de un gran número de mecanismos para facilitar su test [23, 50-52, 64, 69, 105, 106, 143, 154, 155, 161, 170, 175, 183, 186]. En la siguiente sección resumimos los aspectos más relevantes de estos mecanismos.

5.2 Técnicas de test de PLAs binarias.

El primer problema que se plantea al desarrollar técnicas de test para un tipo de circuitos integrados es encontrar unos modelos de fallos adecuados para representar los posibles defectos físicos que pueden provocar su mal funcionamiento.

En lugar del modelo clásico de fallos de anclaje, el modelo más apropiado para la descripción de fallos en PLAs es el de *punto de cruce* o *contacto* ("crosspoint") [175].

Un fallo de contacto se produce por la aparición o desaparición no intencionada de un dispositivo en un punto de cruce de los planos producto o suma de la PLA.

Sobre una función binaria, expresada como suma lógica de implicantes, un fallo simple de contacto en la PLA que la sintetiza puede provocar los siguientes efectos lógicos [170]:

- *crecimiento* de un implicante por desaparición de un literal,
- *reducción* de un implicante por aparición de un nuevo literal en el mismo,
- *desaparición* de un implicante cuando el dispositivo desaparece en un punto de cruce del plano suma, o
- *aparición* de un nuevo implicante como consecuencia de un

dispositivo extra en el plano suma.

Un fallo simple de anclaje en una de las lineas de bit de la misma PLA produce la desaparición de un literal de los implicantes en que interviene (anclaje a 0), o la desaparición de los propios implicantes (anclaje a 1). Si el anclaje se considera en las líneas producto o en las líneas de salida, produce la desaparición de implicantes de la función o la transformación de esta en una función constante (1 ó 0).

La mayor variedad de modificaciones que sobre las funciones pueden inducir los fallos de contacto, en comparación con los de anclaje, hace que los mecanismos de test ideados para la detección de fallos de contacto proporcionen una mayor cobertura de otros fallos no considerados, tales como los de cortocircuito entre líneas [143], así como de gran parte de los fallos de anclaje.

La no detectabilidad de fallos de contacto en una PLA al aplicar todas las combinaciones de entrada y comprobar las salidas resultantes con la respuesta de la PLA libre de fallos está relacionada con el "solapamiento" entre implicantes. Existen aproximaciones de diseño de las PLAs, basadas en ampliaciones de sus planos y modificaciones sobre la particularización de estas para hacer posible que todos los fallos de contacto sean detectables [129, 154].

Respecto a la generación de secuencias de test para PLAs,

si su número de entradas es moderado, es viable la aplicación de una secuencia exhaustiva de vectores de test. Para una PLA binaria de 20 entradas se generarían algo más de un millón de patrones de test. Para una PLA de 50 entradas (ejemplo típico de PLA binaria de gran tamaño [113]), el número de patrones es de orden de 10^{15}, y el test exhaustivo es impracticable. Un test seudo-exhaustivo puede ser útil sólo cuando cada salida de la PLA dependa de un número limitado de entradas.

Por otra parte, las PLAs son un ejemplo típico de circuitos "resistentes" al test aleatorio o seudo-aleatorio [199]. Debido al elevado número de variables que contribuyen a cada producto interno, la probabilidad de detección de fallos al aplicar vectores de test aleatorios es muy pequeña y por lo tanto, para garantizar una cobertura de fallos aceptable se requieren longitudes de test excesivas [37, 59]. La longitud de la secuencia de test puede reducirse añadiendo a la PLA básica algún esquema de enmascaramiento selectivo de las entradas [51].

Si pensamos en una PLA como módulo de un circuito integrado más complejo, el acceso a la misma para la aplicación de patrones y análisis de las respuestas es muy limitado. Además, dependiendo del tamaño de la PLA, puede necesitarse demasiado tiempo de test. Este tiempo habría que sumarlo al necesario para comprobar el resto de los módulos (entre ellos posiblemente otras PLAs) del circuito integrado.

Estos problemas han encontrado soluciones adecuadas consistentes en modificaciones y ampliaciones de la PLA básica, las cuales pueden clasificarse en dos grupos:

- PLAs fácilmente comprobables (ET-PLA), y
- PLAs autocomprobables (BIST-PLA),

Una ET-PLA es una PLA con circuitos añadidos para facilitar su test con un conjunto reducido de patrones, los cuales deben ser preferiblemente independientes de la particularización de la PLA.

En una BIST-PLA, los circuitos añadidos realizan además la generación de patrones de test y la comprobación de las respuestas.

Los distintos esquemas de ET-PLAs y BIST-PLAs binarias ideados [23, 52, 64, 69, 105, 155, 161, 183, 186] incluyen mecanismos de activación selectiva de líneas de bit y de producto. Estudios comparativos de la mayoría de estos esquemas pueden encontrarse en [175, 183].

Para conseguir activar selectivamente las líneas de bit, lo usual es ampliar los decodificadores de las entradas de la PLA de manera que, mediante entradas de control adicionales, se pueda activar individualmente cada línea.

Los dispositivos de activación selectiva de líneas producto más utilizados son registros de desplazamiento con

tantas celdas como líneas producto. A pesar de que esta solución pueda ocasionar un incremento de superficie considerable, tiene la ventaja de que sólo se requiere una entrada adicional (entrada de datos serie del registro) para suministrar la parte correspondiente de los patrones de test.

Para implementar un esquema de compresión y comparación de respuestas, el cual sea independiente de la particularización de la PLA, suele recurrirse a mecanismos de generación/comprobación de paridad. Los planos de la PLA se amplían con una o varias líneas producto y líneas de salida, las cuales se programan para el establecimiento de una determinada paridad de las respuestas a los patrones de test.

Los aspectos más relevantes a tener en cuenta cuando se diseña un procedimiento de test de PLAs pueden concretarse en los siguientes puntos:

- Cobertura de fallos resultante, la cual debe estar referida a aquellos modelos de fallos más adecuados para representar los defectos físicos de la PLA y de la circuitería añadida para el test.

- Universalidad de la secuencia de test y de las respuestas de la PLA, lo cual simplifica el diseño del circuito generador de la secuencia y del evaluador de respuestas.

- Superficie ocupada por la circuitería añadida para el

test. Esta depende en gran medida de la facilidad de conexión entre los elementos añadidos y los planos de la PLA.

- Número de terminales añadidos para controlar los circuitos de test.

- Degradación de la velocidad de funcionamiento de la PLA. Deben evitarse circuitos conectados en serie con el camino de las señales en la PLA. Es conveniente además que la frecuencia de funcionamiento de la PLA en modo de test pueda hacerse coincidir con la de su funcionamiento normal.

5.3 Procedimiento de test de PLAs multivaluadas.

En esta sección presentamos un procedimiento de test para las PLAs multivaluadas propuestas en el capítulo anterior. En primer lugar (apartado 5.3.1) introducimos el conjunto de modelos de fallos a nivel lógico que se van a considerar, discutiendo sus limitaciones en la representación de defectos físicos en el circuito.

En el apartado 5.3.2 concretamos el procedimiento de test propuesto. La evaluación de su efectividad en la detección de fallos será tratada en el apartado 5.3.3.

5.3.1 Modelos de fallos

Al igual que para PLAs binarias, los modelos de fallos a considerar deben adaptarse a las características estructurales propias de este tipo de circuitos, así como estar íntimamente relacionados con posibles defectos físicos en su realización integrada.

Las distintas aproximaciones de diseño de PLAs binarias fácilmente comprobables y autocomprobables mencionadas en la sección anterior, utilizan básicamente los modelos de fallos de contacto, anclaje, y cortocircuito entre lineas.

Los esquemas de test propuestos buscan fundamentalmente la detección de fallos de contacto. El resto de los fallos quedan cubiertos por equivalencia con los primeros, o por la aplicación de algunos vectores de test adicionales.

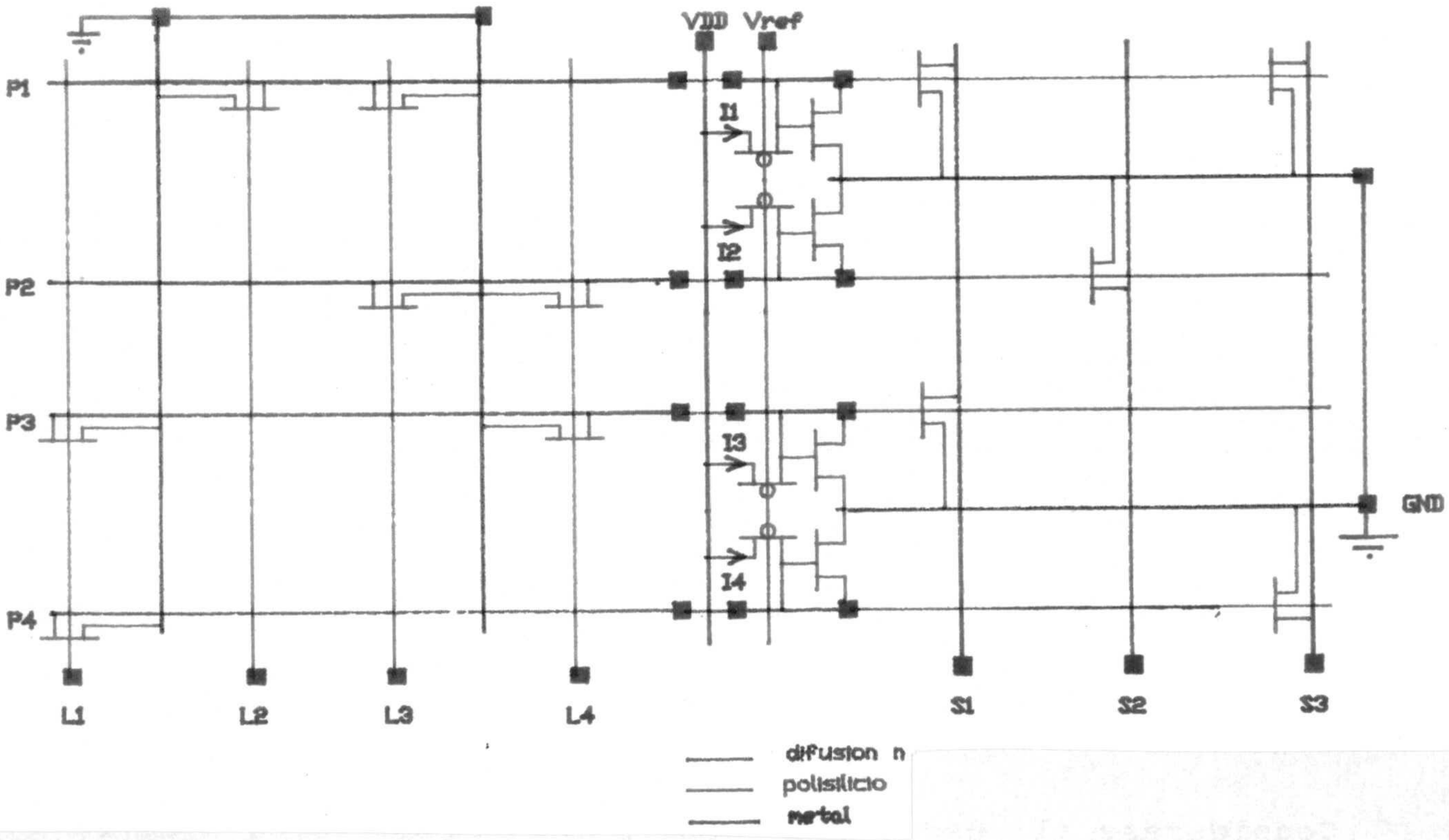
En PLAs multivaluadas como las propuestas en el capítulo 4, también podemos considerar modelos de fallos similares a los normalmente utilizados en PLAs binarias. Sin embargo, según puntualizaremos en los párrafos que siguen, hay determinados defectos en la realización física de la PLA cuyo efecto no se puede representar mediante tales modelos. Esta inadecuación de los modelos de fallos es en algunos casos debida a la naturale-

za multivaluada de la PLA, pero en otros casos se debe a detalles de la realización como circuito integrado, los cuales deben tenerse en cuenta también en PLAs binarias.

A continuación pasaremos a considerar un conjunto amplio de posibles defectos físicos en la realización integrada de la PLA multivaluada y analizaremos su correspondencia con modelos de fallos. Para no extendernos demasiado en la redacción sólo trataremos con detalle una estructura de PLA del tipo 1, para la cual concretamos un esquema de test en el apartado 5.3.2. Un análisis de los posibles defectos físicos en el resto de las estructuras de PLA propuestas en el capítulo 4 lleva a resultados similares a los que se obtienen para la PLA del tipo 1.

Considérese el esquema de PLA r-valuada del tipo 1 de la figura 5.2. En ella se han omitido las celdas generadoras de literales cuyo análisis se hará posteriormente. La figura 5.2.a representa el esquema eléctrico de los planos NOR y TSUM de forma muy aproximada a la realización integrada seudo-NMOS. La figura 5.2.b muestra el correspondiente esquema lógico.

Con referencia a un esquema lógico como el de la figura 5.2.b, se dice que una de sus líneas (*L*iteral, *P*roducto, o de *S*alida) presenta un anclaje a α cuando se mantiene constantemente a un nivel lógico $\alpha \varepsilon \{0,1,..,r-1\}$. Se indica con "*contacto*$^{+}$" la existencia de una conexión extra en un punto de cruce de alguno de los planos de la PLA, y mediante "*contacto*$^{-}$" la ausencia erronea de tal conexión.



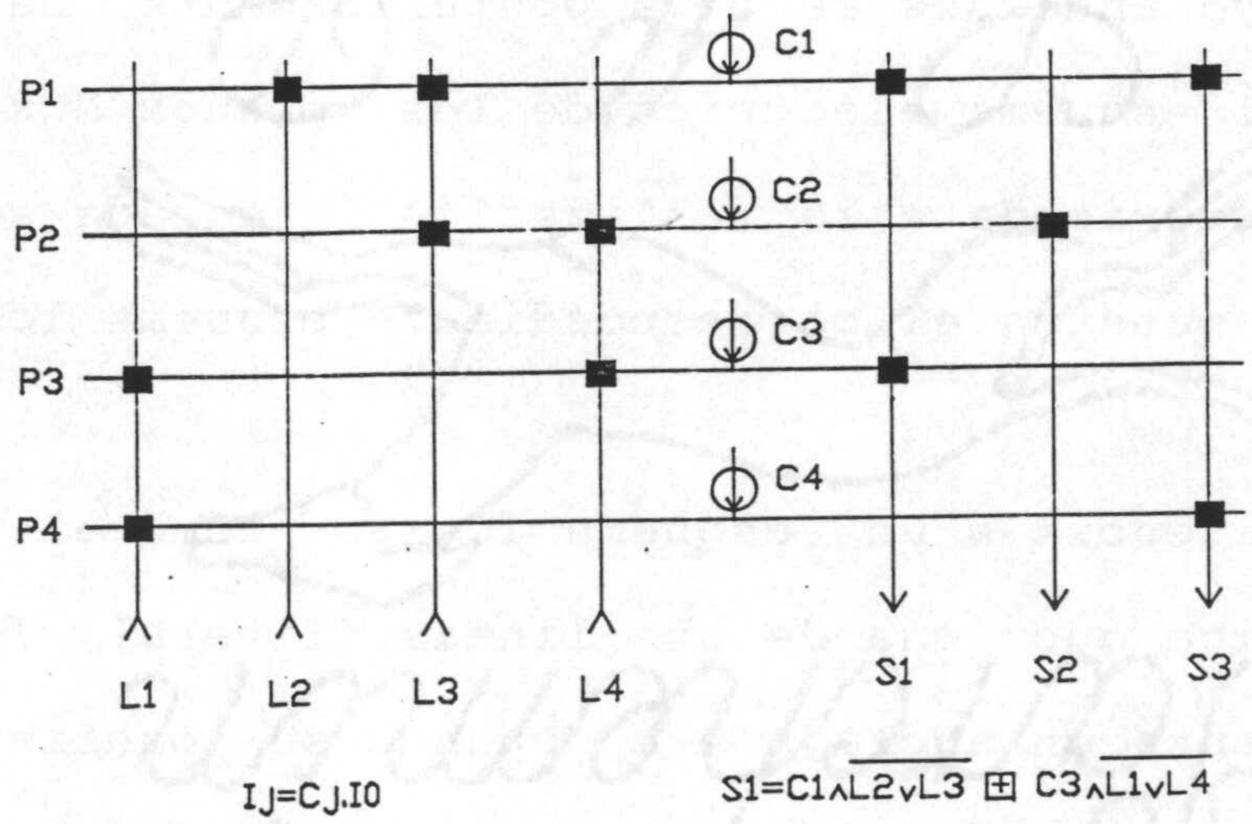


FIGURA 5.2: PLA multivaluada del tipo 1 con 4 líneas de literal, 4 líneas producto, y 3 líneas de salida. (a) Esquema aproximado de la realización integrada de los planos. (b) Esquema lógico correspondiente

Los fallos de contacto provocan, sobre las funciones sintetizadas por la PLA, efectos lógicos similares a los que se tienen en PLAs binarias; concretamente, crecimiento (reducción) de implicantes asociado con fallos de contacto$^-$ (contacto$^+$) en el plano NOR, y desaparición (aparición) de implicantes debida a fallos de contacto$^-$ (contacto$^+$) en el plano TSUM.

Si analizamos el esquema eléctrico de la figura 5.2.a, en la realización integrada de los planos de la PLA pueden darse los siguentes defectos:

1. Plano NOR:

1.1.- Pérdida de la conexión de drenador o de fuente de un transistor NMOS, o ausencia erronea del transistor.

1.2.- Existencia de un transistor NMOS extra, debido por ejemplo a una programación erronea de la PLA.

1.3.- Rotura de una línea de literal (L_i).

1.4.- Rotura de una línea producto (P_j).

1.5.- Rotura de la línea de tierra o falta de alguna conexión a ella.

1.6.- Cortocircuito entre dos líneas adyacentes.

1.7.- Cortocircuito entre dos líneas que se cruzan.

2. Plano TSUM:

2.1.- Ausencia de un transistor PMOS, o fallo de alguna de sus conexiones.

2.2.- Ausencia del transistor NMOS que genera las tensiones de puerta de los transistores de los puntos de cruce.

2.3.- Ausencia de un transistor NMOS en un punto de cruce.

2.4.- Existencia erronea de algún transistor en un punto de cruce.

2.5.- Rotura de una línea producto.

2.6.- Rotura de una línea de salida.

2.7.- Rotura de la línea de alimentación V_{DD}.

2.8.- Rotura de la línea de tierra o fallo de alguna conexión a ella.

2.9.- Cortocircuito entre dos líneas adyacentes.

2.10.- Cortocircuito entre dos líneas que se cruzan.

Los defectos 1.1, 1.2, 2.3, y 2.4 pueden representarse mediante fallos de contacto simples. El 1.1 y el 2.3 mediante fallos de $contacto^-$, y los defectos 1.2 y 2.4 mediante $contacto^+$.

La rotura de una línea de literal en el plano NOR (defecto 1.3) o de una línea producto en el plano TSUM (defecto 2.5), equivale a la desaparición de varios transistores NMOS en puntos de cruce de los respectivos planos. Por tanto, estos defectos son representables mediante fallos de $contacto^-$ múltiples, cuya multiplicidad depende de la posición de la rotura.

El defecto 1.4 también tiene un efecto dependiente de la posición de la rotura. Si ésta se produce a mitad del plano NOR se tendrá un fallo de $contacto^-$ múltiple. Si la rotura se debe a una falta de conexión con el plano TSUM, el correspon-

diente fallo es un anclaje a r-1 de la línea producto.

Un defecto del tipo 1.5 es equivalente a la falta de conexiones de drenador de al menos una pareja de columnas de transistores en el plano NOR, correspondiendo pues a un fallo de contacto- múltiple cuya multiplicidad coincide con el número total de transistores en dichas columnas.

Existen varios posibles cortocircuitos entre líneas adyacentes del plano NOR (defecto 1.6):

1.6.1.- Cortocircuito entre dos líneas producto: este defecto fuerza a ambas al mismo valor de tensión, la cual será nula si conduce alguno de los transistores conectados a ellas. Si todos estos transistores están en corte, los niveles de corriente en todas las salidas a las que contribuyen los términos producto se verán en general deteriorados. Consideremos como ejemplo un cortocircuito entre las líneas P_1 y P_2 del esquema de la figura 5.2.a. Si los cuatro transistores de los puntos de cruce con líneas de literal están en corte, la pareja de líneas producto recibe una corriente I_1+I_2. Una parte (I_1^*) de esta corriente se suma a las líneas S_1 y S_3, y otra parte (I_2^*) a la línea S_2. Normalmente se tendrá $I_1^* \neq I_1$ e $I_2^* \neq I_2$. El efecto sobre las funciones que sintetiza la PLA es la transformación de los términos $c_1 \cap \overline{L_2 \cup L_3}$ y $c_2 \cap \overline{L_3 \cup L_4}$, en $c_1^* \cap \overline{L_2 \cup L_3 \cup L_4}$ y $c_2^* \cap \overline{L_2 \cup L_3 \cup L_4}$ respectivamente; siendo $c_1 \neq c_1^*$ y $c_2 \neq c_2^*$. Si en las salidas de la PLA se lleva a

cabo una detección binaria (nivel 0 o distinto de 0), este defecto puede modelarse como un fallo de contacto⁺ en el plano NOR. En general, la multiplicidad del fallo de contacto⁺ asociado coincidirá con el número de literales en los que difieren ambos términos producto.

1.6.2.- Cortocircuito entre líneas de literal: su efecto depende de la posición del cortocircuito. Debido a la elevada resistividad de estas líneas, cuando ambas se activan por niveles lógicos diferentes se produce una distribución de voltaje que origina la conmutación de una parte, no predecible a priori, de los transistores que conectan a ellas. Este efecto también se presentaría si se tratase de PLAs MOS binarias, motivo por el cual consideramos que el modelo de cortocircuito tipo AND normalmente utilizado en estas [50, 106, 155] no tiene validez para cortocircuitos entre líneas de esta naturaleza.

1.6.3.- Cortocircuito entre una línea de literal o una de producto y tierra: el primero de ellos provoca efectos similares al 1.6.2. En este caso, cuando se activa la línea de literal, la distribución de tensión que se produce será tal que los dispositivos próximos al cortocircuito no pueden entrar en conducción, lo cual puede modelarse mediante un fallo de contacto⁻ simple o múltiple.

Un cortocircuito entre una línea producto y tierra puede deberse a la existencia erronea de un agujero de contacto

o a un defecto en el oxido grueso, o bien a la existencia de una unión entre esta línea y alguna línea paralela de polarización del sustrato (no dibujadas en el esquema de la figura 5.2.a). En cualquier caso, se tendrá un anclaje a 0 de dicha línea producto.

Como cortocircuitos entre líneas que se cruzan (1.7) podemos considerar los siguientes:

1.7.1.- Cortocircuito entre una línea de literal y una línea producto: produce un deterioro de niveles de corriente en varias líneas de salida. No sólo se ven afectadas las salidas a las que contribuye el término producto en cuestión, sino todas las salidas correspondientes a funciones que dependen del literal.

La variación de los niveles de corriente resultante depende de la relación de transconductancias entre los transistores NMOS del plano NOR y los del plano TSUM, y de la posición del cortocircuito.

1.7.2.- Cortocircuitos entre líneas de literal o líneas producto y tierra: coincide con 1.6.3.

En el plano TSUM, la ausencia de un transistor PMOS (defecto 2.1) puede modelarse mediante un anclaje a 0 de la correspondiente línea producto.

Un defecto del tipo 2.2 hace que, para todas las combina-

ciones de entradas que no anulan el correspondiente término producto, se detecte un nivel de corriente máximo en todas las salidas a las que contribuye dicho término. En general, este defecto puede representarse mediante un fallo múltiple de contacto$^+$ en las líneas de salida indicadas, junto con uno de contacto múltiple en el plano NOR.

Un defecto del tipo 2.5 equivale a varios del tipo 2.3, estando representado pues mediante un fallo múltiple de contacto$^-$.

La rotura de una línea de salida (2.6) anula la contribución de corriente de una o varias líneas producto a la salida. El modelo de fallo asociado es un contacto$^-$ múltiple.

La rotura de la línea de alimentación (2.7) anula la contribución de varias líneas producto a las salidas, lo que equivale a la desaparición de todos los transistores de puntos de cruce de dichas líneas (fallo múltiple de contacto$^-$).

Un defecto del tipo 2.8 provoca una contribución de corriente nula a las salidas por parte de al menos una pareja de filas de transistores NMOS. Se trata pues de un fallo múltiple de contacto$^-$, cuya multiplicidad coincide con el número total de transistores en puntos de cruce de las mencionadas filas.

Posibles defectos de cortocircuito entre líneas adyacentes

(2.9) son los siguientes:

2.9.1.- Cortocircuito entre dos líneas producto (sus prolongaciones en el plano TSUM): al activar una u otra línea producto se produce una contribución de corriente a las líneas de salida que presentan puntos de cruce con ambas líneas. Si se analizan las salidas para detectar en ellas la ausencia o no de corriente (es decir sólo se hace distinción entre niveles lógicos cero ó distinto de cero), el efecto del cortocircuito corresponde al de un fallo múltiple de contacto^{+}.

2.9.2.- Cortocircuito entre dos líneas de salida: su efecto es un deterioro de los niveles de corriente en ambas líneas, el cual en general no puede representanse mediante alguno de los modelos de fallos considerados.

2.9.3.- Cortocircuito entre una línea producto y tierra: corresponde a un fallo de contacto^{-} múltiple en dicha línea.

Cortocircuitos posibles entre líneas que se cruzan en el plano TSUM (defecto 2.10) son:

2.10.1.- Cortocircuito entre una línea de producto y una línea de salida: su efecto sobre el funcionamiento de la PLA no se evidencia fácilmente como ocurre con los defectos tratados anteriormente. Para observar las consecuencias del cortocircuito hemos realizado simulaciones eléctricas

de una PLA defectuosa, habiéndose comprobado que puede modelarse mediante un anclaje a r-1 de la línea de salida afectada. En los resultados de simulación se observa que el cortocircuito produce además un deterioro de los niveles de corriente en el resto de las líneas de salida que presentan transistores en los puntos de cruce con la línea producto.

2.10.2.- Cortocircuito entre una línea de salida y tierra: si tenemos en cuenta que las líneas de salida normalmente activarán espejos de corriente PMOS, debido al corto-circuito la corriente de salida será interpretada como un nivel de corriente máximo, teniendose un anclaje a r-1 en la correspondiente línea.

En la tabla 5.1 aparecen relacionados aquellos defectos en los planos NOR y TSUM que pueden representarse mediante modelos de fallos de contacto y/o anclaje.

En la tabla puede verse que los fallos de anclaje posibles son: líneas producto a 0 ó a r-1 en el plano NOR, línea producto a 0 en el plano TSUM, y línea de salida a r-1.

Un anclaje a 0 (r-1) de una línea producto en el plano NOR es equivalente a un fallo múltiple de contacto^{+} (contacto^{-}) en la misma línea. Visto sobre el esquema de la figura 5.2.a, la línea P_i se mantendría siempre conmutada a tierra si existieran transistores en los puntos de cruce con todas las líneas de

literal (contacto^{+}), o no conmutaría nunca en ausencia de todos los transistores de los puntos de cruce con líneas de literal (contacto^{-}).

TABLA 5.1: Defectos en la PLA multivaluada y modelos de fallos asociados.

defecto	modelo de fallos	
		PLANO NOR
1.1	contacto^{-}	
1.2	contacto^{+}	
1.3	contacto^{-} múltiple	
1.4	contacto^{-} múltiple	/ anclaje de P_j a r-1
1.5	contacto^{-} múltiple	
1.6.1	contacto^{+} múltiple	
1.6.3	contacto^{-} múltiple	/anclaje de P_j a 0.
1.7.2	contacto^{-} múltiple	/anclaje de P_j a 0
		PLANO TSUM
2.1	anclaje de P_j a 0	
2.2	contacto^{+} múltiple y contacto en el plano NOR	
2.3	contacto^{-}	
2.4	contacto^{+}	
2.5	contacto^{-} múltiple	
2.6	contacto^{-} múltiple	
2.7	contacto^{-} múltiple	
2.8	contacto^{-} múltiple	
2.9.1	contacto^{+} múltiple	
2.9.3	contacto^{-} múltiple	
2.10.1	anclaje de S_k a r-1	
2.10.2	anclaje de S_k a r-1	

El anclaje a 0 de la línea producto en el plano TSUM equivale a un fallo múltiple de contacto^{-} en dicha línea (ausencia de todos los transistores en los puntos de cruce con líneas de salida).

El anclaje de una línea de salida a 0 equivale a un fallo múltiple de contacto^{-} en dicha línea. La multiplicidad de este

fallo coincide con el número de transistores en los puntos de cruce de la línea de salida.

Un anclaje a r-1 de la línea de salida puede hacerse equivalente a un conjunto de fallos de contacto en ambos planos de la PLA; concretamente, a fallos de contacto$^+$ en el plano TSUM y de contacto$^-$ en el plano NOR.

En resumen, todos los defectos físicos recogidos en la tabla 5.1 pueden modelarse mediante fallos de contacto simples o múltiples. Para evaluar la capacidad de detección de fallos del procedimiento de test de la PLA multivaluada que estamos considerando basta pues con determinar la cobertura de fallos de contacto que proporciona dicho procedimiento.

El modelo de fallos de contacto no permite, sin embargo, representar algunos de los posibles defectos físicos enumerados. Concretamente, la detección de los defectos del tipo 1.6.2, 1.7.1, y 2.9.2, no quedaría garantizada por el test aun cuando presentara una cobertura del 100% de todos los fallos de contacto.

5.3.2 Test propuesto.

En este apartado se presenta el procedimiento de test para la PLA multivaluada. Este procedimiento se basa en la amplia-

ción de la PLA básica para facilitar su comprobación mediante un conjunto reducido de patrones de test independientes de la particularización de la misma. El exceso de superficie producido por la circuitería añadida para facilitar el test es similar al de esquemas existentes para PLAs binarias [183].

La circuitería añadida cumple dos objetivos: en primer lugar aumentar la controlabilidad en los planos de la PLA al permitir que se puedan activar puntos de cruce individuales. En segundo lugar, en los planos de la PLA se establece un mecanismo de paridad, el cual permite verificar sus respuestas a los patrones de test comprobando la paridad de las salidas. Por la forma en que se amplía la PLA y se lleva a cabo el test, dicha comprobación es independiente de la particularización de la PLA y se puede realizar mediante un circuito muy sencillo.

Dada una PLA **r-valuada**, la cual proporcione una realización NOR/TSUM de **m funciones de n variables** con un total de **k términos producto**, para realizar su test mediante un conjunto universal de patrones proponemos su ampliación con la siguiente circuitería (ver figura 5.3):

- **s** líneas producto configuradas de manera que cada línea literal tenga en total un número impar de puntos de cruce con transistores y un número impar de ellos sin transistor. Salvo el caso en que todas las líneas de literal cumplan inicialmente la condición anterior, el número de

líneas producto añadidas será s=1 si k es impar, y s=2 si k es par. El número total de líneas producto (k+s) será en consecuencia siempre par.

- Una línea de salida: Z_T, cuyos puntos de cruce deben programarse de manera que la prolongación de cada línea producto en este plano tenga un número impar de dispositivos. Las líneas de producto añadidas, dentro del plano TSUM sólo tendrán un transistor en sus puntos de cruce con la línea Z_T.

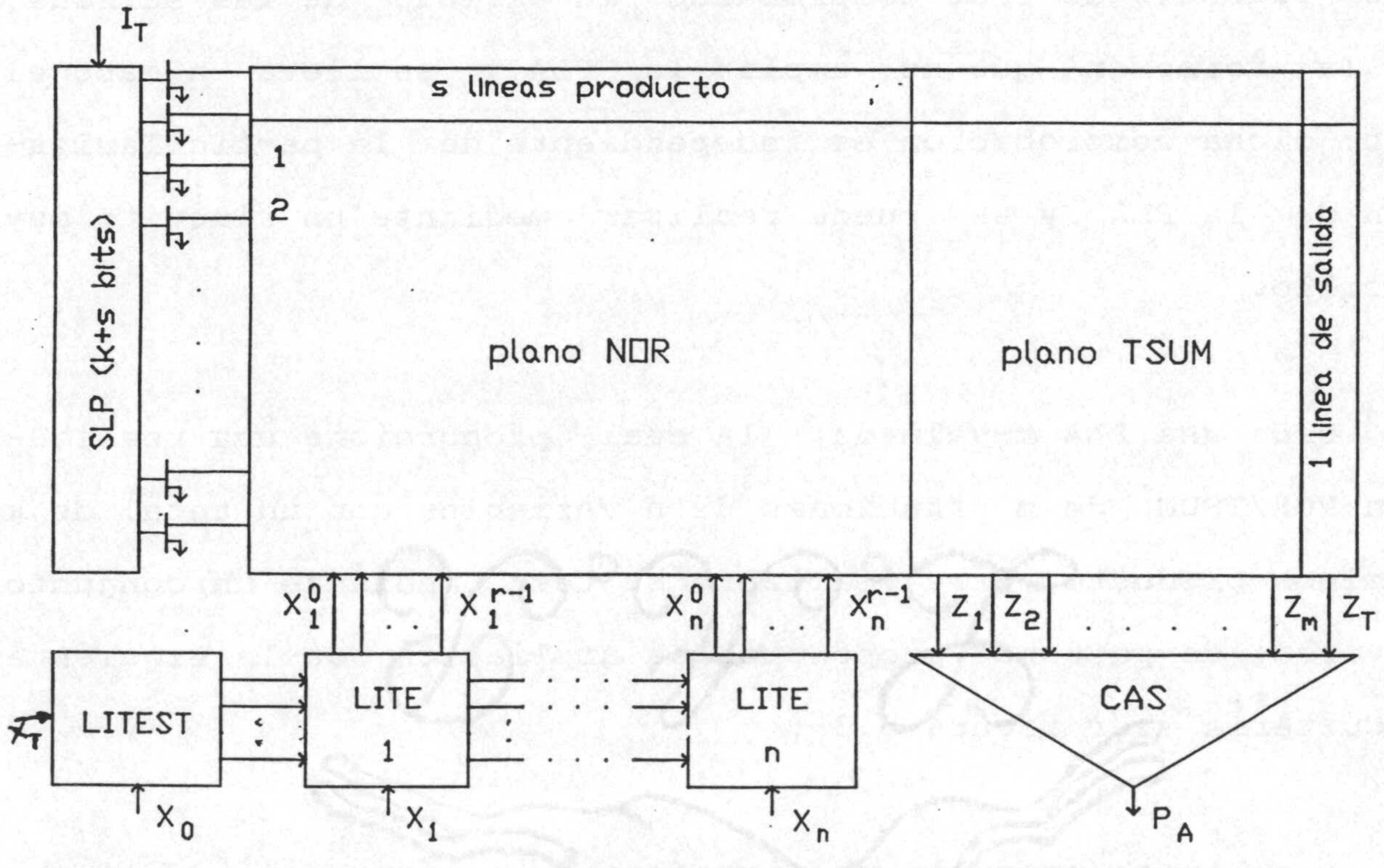


FIGURA 5.3: Esquema de la PLA aumentada con la circuitería de test.

- Un circuito secuencial para la selección individual de líneas producto (**SLP**), el cual puede consistir en un

registro de desplazamiento binario cuyas k+s salidas actuen sobre las líneas producto de la misma manera que lo hace una línea de literal. I_T es la entrada de datos serie mediante la cual se actualizarán las salidas (SLP_1, SLP_2,.., SLP_{k+s}) del SLP.

- Un circuito compactador y analizador de las salidas (**CAS**), el cual consiste en un generador y comparador de paridad acumulada.

- Una celda generadora de literales del tipo NOR (**LITEST**) cuyas salidas actuen sobre las entradas de control de las n celdas generadoras de literales que decodifican las entradas de la PLA. Estas celdas deberán ser también del tipo NOR. LITEST tiene dos entradas, una entrada r-valuada (X_0) en modo de corriente, y una entrada binaria (X_T) en modo de tensión. X_T actúa como entrada de control común para las r salidas de LITEST. Cuando $X_T=1$ se anulan las salidas de LITEST (modo de funcionamiento normal de la PLA). Si $X_T=0$ y $X_0=1$, $l\varepsilon\{0,1,..,r-1\}$, se anulan las líneas de literal X_i^l, i=1,2,..,n.

Los patrones de test utilizados en el esquema de la figura 5.3 deben definir tanto las entradas de las celdas generadoras de literales de la PLA como el estado del circuito SLP. Cada patrón de test lo identificaremos mediante el siguiente vector de k+s+n+1 componentes:

$$T \equiv \{SLP_1, SLP_2, .., SLP_{k+s}; X_0; X_1, X_2, .., X_n\} \qquad (5.1)$$

Las primeras k+s componentes de T identifican las salidas binarias del SLP. X_0 es la entrada r-valuada de la celda LITEST; y X_i, i=1,2,..n, son las entradas de las celdas LITE. En (5.1) no se indica el valor de la entrada X_T, la cual se supone a 0 mientras se efectúa el test.

La respuesta de la PLA al patrón T, PLA(T), la identificaremos mediante el siguiente vector de m+1 componentes:

$$PLA(T) \equiv \{Z_1, Z_2, .. Z_m; Z_T\} \qquad (5.2)$$

en donde las Z_i, i=1,2,..,m, son las salidas de función de la PLA. Z_T es la salida de la línea añadida en el plano TSUM.

Para realizar el test de la PLA se aplicará un subconjunto de los posibles vectores en la forma (5.1); concretamente, sólo se consideran patrones en la forma:

$$T^0 \equiv \{1,1,..,1;\alpha;\alpha,\alpha,..\alpha\} \qquad (5.3)$$

$$\alpha \varepsilon \{0,1,..,r-1\}$$

$$T^1{}_{i,\alpha} \equiv \{1,..,1,SLP_i=0,1,..,1;\alpha;\alpha,\alpha,..,\alpha\}$$

$$i=1,2,..,k+s;\ \alpha \varepsilon \{0,1,..,r-1\} \qquad (5.4)$$

$$T^2{}_{i,j,l} \equiv \{1,..,1,SLP_i=0,1,..,1;\alpha;\alpha,..,\alpha,X_j=1,\alpha,..,\alpha\}$$

$$i=1,2,..,k+s;\ j=1,2,..,n;\ \alpha,l \varepsilon \{0,1,..,r-1\},\ \alpha \neq l \qquad (5.5)$$

El patrón T^0 hace que todas las líneas de literal se pongan a 0. $T^1{}_{i,\alpha}$ activa la i-ésima línea producto y mantiene a 0 todas las líneas de literal. Un patron $T^2{}_{i,j,l}$ hace que se seleccione la línea de literal $X_j{}^l$ y la i-ésima línea producto. El resto de las líneas de literal y de producto quedan a 0.

Si la PLA funciona correctamente, la respuesta a cada uno de los patrones anteriores debe ser:

$PLA(T^0) \equiv \{0,0,..,0;0\}$

$PLA(T^1{}_{i,\alpha}) \equiv \{Z_1{}^*, Z_2{}^*, .., Z_m{}^*; Z_T\}$ Los valores Z^* que se obtienen en las salidas de función dependerán de la existencia o no de transistor en el punto de cruce de la i-ésima línea producto y la correspondiente línea de salida (dentro del plano TSUM). No obstante, dado que el número total de dispositivos de cada línea producto en el plano TSUM es impar, el número total de valores no nulos del vector de salida anterior también será impar.

$PLA(T^2{}_{i,j,l}) \equiv \{0,0,..,0;0\}$ si hay transistor en el punto de cruce de la i-ésima línea producto y la línea de bit $X_j{}^l$.

$PLA(T^2_{i,j,l}) \equiv \{Z_1^*, Z_2^*, .., Z_m^*; Z_T\}$ si no existe transistor en el punto de cruce mencionado. Los valores de Z^* y Z_T coinciden con los que se obtienen al aplicar $T^1_{i,\alpha}$, existiendo un número impar de ellos no nulos.

Al aplicar cada patrón de test (T), el circuito CAS conectado a la PLA determina la paridad del vector de salidas (PLA(T)). Sea P' la paridad de dicho vector, definida en la forma:

P' = 1 si PLA(T) incluye un número impar de valores no nulos.
P' = 0 si el número de valores no nulos es par o cero.

El circuito CAS genera además una paridad acumulada (P_A), la cual consiste en la suma aritmética módulo dos (operación lógica XOR) de las paridades P' de cada dos vectores de salida consecutivos.

Para realizar el test de la PLA r-valuada proponemos la la siguiente secuencia de aplicación y comparación de respuestas de patrones de test:

(1) P_A <-0 (iniciar la paridad acumulada), P_C <-0 (iniciar la paridad acumulada correcta);

(2) T^0 (aplicar T^0 para cualquier valor de $\alpha \varepsilon \{0,1,..,r-1\}$), P_A <-P_A +P', comparar P_A y P_C ;

(3) Para cada α (α=0 hasta α=r-1):
Para cada i (i=1 hasta i=k+s): $T^1_{i,\alpha}$ (aplicar $T^1_{i,\alpha}$),

P_C <-$(P_C+1)_{mod\ 2}$, P_A <-P_A+P', comparar P_A y P_C ;

(4) Para cada j (j=1 hasta j=n) y para cada l (l=0 hasta l=r-1):
Para cada i (i=1 hasta i=k+s): $T^2_{i,j,l}$ (aplicar $T^2_{i,j,l}$), P_A <-P_A+P';
P_C <-$(P_C+1)_{mod\ 2}$, comparar P_A y P_C ;

En la secuencia anterior se considera un total de 1+r.(k+s).(n+1) patrones de test. Al aplicar T^0, y cada vez que se aplica un patrón T^1, se compara la paridad acumulada con su valor correcto.

Cuando se aplican patrones del tipo T^2, la comparación de paridad acumulada sólo se efectúa al final de la activación de todos los puntos de cruce de cada línea de literal (cada k+s patrones). Esta forma de proceder implica evidentemente una pérdida de información acerca del estado de los puntos de cruce individuales de las líneas de literal, pero tiene la ventaja de que la secuencia de valores de P_A que se obtiene es independiente de las funciones realizadas por la PLA, y además dicha secuencia puede comprobarse mediante un circuito muy sencillo. En efecto, en ausencia de defectos en la PLA, la paridad P' del vector de salida es 0 al aplicar T^0 y P'=1 para cada patrón T^1; por lo tanto la paridad acumulada debe seguir una secuencia de k+s+1 ceros y unos en la forma:

$$P_C: 0;1;0;1;0;..;1;0 \qquad (5.6)$$

Cuando se aplican vectores del tipo T^2, los valores de la paridad acumulada no siguen la secuencia anterior para todos

los patrones, pero sí para aquellos cuya comparación se indica en el punto (4) de la secuencia de test. Dado que existe un número impar de puntos de cruce sin transistor en cada línea de bit, al activar mediante patrones T^2 todos los puntos de cruce de dicha línea debe obtenerse un número impar de respuestas con paridad P'=1; es decir, la contribución neta de cada una de estas subsecuencias a la paridad acumulada es $P_A=1$ y, por lo tanto, se obtendrán valores alternantes de P_A para cada dos líneas de literal consecutivas; es decir, una secuencia en la forma (5.6).

5.3.3 Evaluación de la cobertura de fallos.

Para obtener una medida de la eficacia del esquema de test propuesto en el apartado anterior evaluaremos la probabilidad de detección de un fallo en la PLA al aplicar la secuencia completa de test. Para obtener esta medida consideraremos la presencia en los planos de la PLA de fallos de contacto simples o múltiples y calcularemos el porcentaje de ellos que, respecto de todos los posibles, son detectables.

Tras el análisis realizado en el apartado 5.3.1 queda claro que un esquema de test con una buena cobertura de fallos de contacto en los planos de la PLA multivaluada también proporciona una buena cobertura de la mayoría de los posibles defectos físicos de su realización integrada. Resulta pues

adecuado evaluar la bondad del esquema de test propuesto considerando sólo el modelo de fallos de contacto.

En realidad, lo que interesaría conocer es la probabilidad de que se detecte un defecto físico de cualquier índole en la realización integrada de la PLA, pero esto obligaría a analizar cada defecto por separado, así como todas las posibles combinaciones de ellos, o bien realizar un estudio estadístico en base a simulaciones de PLAs con defectos físicos inyectados. Aún así, la medida que se obtendría no sería especialmente significativa ya que no se tiene información acerca de qué defectos físicos son más o menos probables. En el mejor de los casos, los defectos inyectados en las PLAs a simular habrían de generarse aleatoriamente.

Refiriéndonos a fallos de contacto, la cobertura que se obtiene con el esquema de test propuesto se concreta a continuación mediante el teorema 5.1, para cuya demostración se hace uso de los siguientes lemas:

Lema 5.1: *Mediante la secuencia de patrones de test* $\{T^2_{i,j,1}\}$ *se detectan todos los posibles fallos de contacto simples en el plano NOR de la PLA r-valuada.*

Demostración:

Al tratarse de un fallo simple, el número resultante de transistores en la correspondiente línea de literal será entonces par. El número de puntos de cruce sin transistor

también será par, motivo por el cual la contribución a la paridad acumulada que se obtiene al activar todos los puntos de cruce de la línea de literal mediante patrones $T^2_{i,j,l}$ será nula, detectándose pues el fallo.

Lema 5.2: Mediante la secuencia de patrones de test $\{T^1_{i,\alpha}\}$ se detectan todos los fallos de contacto simples en el plano TSUM de la PLA r-valuada.

Demostración:

Si existe un fallo de contacto simple en una de las líneas producto dentro del plano TSUM, el número total de dispositivos en dicha línea será par; por lo tanto, cada uno de los patrones $T^1_{i,\alpha}$ que activan la línea tendrá una respuesta de paridad distinta a la esperada y el fallo será detectado.

Lema 5.3: Existen $Nnd_{NOR} = 2^{(k+s-1).r.n} - 1$ posibles fallos de contacto múltiples no detectables en el plano NOR mediante la secuencia de patrones $\{T^2_{i,j,l}\}$.

Demostración:

La detección de fallos de contacto múltiples en el plano NOR se realiza mediante los patrones $T^2_{i,j,l}$. Las subsecuencias de patrones $T^2_{i,j,l}$, i=1,2,..k+s, activan los puntos de cruce de las distintas líneas de literal. Si la multiplicidad de un fallo de contacto en una línea de literal es par, la subsecuencia que activa esta línea produce una contribución a la paridad acumulada igual que si no existiese fallo. Para que un fallo de

contacto múltiple en el plano NOR no sea detectado deberá presentar multiplicidad par para todas y cada una de las líneas de literal afectadas.

El número total de posibles fallos de contacto de multiplicidad (μ) par en los k+s puntos de cruce de una línea de literal, incluyendo multiplicidad μ=0 (ausencia de fallo), viene dado por la siguiente suma de combinaciones:

$$Nnd_{j,l,NOR} = \sum_{\mu=0}^{\mu=k+s} \binom{k+s}{\mu} = 2^{k+s-1} \quad , \mu \text{ par} \qquad (5.7)$$

Por tanto,el número total de fallos de contacto múltiples no detectables en las r.n líneas de literal es, como queríamos demostrar:

$$Nnd_{NOR} = (Nnd_{j,l,NOR})^{rn} - 1 = 2^{(k+s-1)rn} - 1 \qquad (5.8)$$

en donde el término -1 corresponde a la situación de no fallo (multiplicidad μ=0 para todas las líneas de literal).

Lema 5.4: *Existen* $Nnd_{TSUM}=2^{(k+s)m} - 1$ *posibles fallos de contacto múltiples no detectables en el plano TSUM mediante la secuencia de patrones* $\{T^1_{i,\alpha}\}$.

Demostración:

La detección de fallos de contacto múltiples en el plano TSUM se realiza mediante los patrones de test $T^1_{i,\alpha}$. Un fallo

de contacto múltiple en este plano es no detectable siempre que su multiplicidad referida a cada una de las líneas de producto afectadas sea par, ya que en tal caso las paridades de las respuestas que se obtienen coinciden con las de la PLA libre de fallos.

El número de posibles fallos de contacto de multiplicidad par (incluida multiplicidad μ=0) en los m+1 puntos de cruce de una línea producto dentro del plano TSUM es:

$$Nsd_{i,TSUM} = \sum_{\mu=0}^{\mu=2\cdot[(m+1)/2]} \binom{m+1}{\mu} = 2^m, \text{ con } \mu \text{ par} \qquad (5.9)$$

en donde los corchetes se emplean para representar la parte entera del número que encierran, de tal manera que 2.[(m+1)/2] es el mayor número par menor o igual que m+1.

En las k+s líneas producto, el número de fallos múltiples no detectables es pues:

$$Nnd_{TSUM} = (Nnd_{i,TSUM})^{k+s} - 1 = 2^{(k+s)m} - 1 \qquad (5.10)$$

Lema 5.5: La detectabilidad de un fallo de contacto en el plano NOR no se ve afectada por la existencia de un fallo no detectable en el plano TSUM.

Demostración:

Si el fallo del plano TSUM es no detectable (mediante los patrones $T^1_{i,\alpha}$) es porque no cambia la paridad de ninguno de

los vectores de salida; por lo tanto, los valores de paridad que se obtienen al aplicar los vectores $T^2_{i,j,l}$ coinciden con los que se tendrían si el plano TSUM estuviese libre de fallos.

Lema 5.6: La detectabilidad de un fallo de contacto en el plano TSUM no se ve afectada por la existencia de un fallo no detectable en el plano NOR.

Demostración:

Cuando se aplican patrones $T^1_{i,\alpha}$ todas las líneas de literal se mantienen a 0; por lo tanto, la paridad de los vectores de salida que resultan es totalmente independiente de la distribución de dispositivos en puntos de cruce del plano NOR, manteniéndose pues la detectabilidad de fallos que se tenía para el plano TSUM.

Teorema 5.1:

Con la secuencia de patrones de test $\{T^1_{i,\alpha}\}-\{T^2_{i,j,l}\}$ se detectan todos los fallos de contacto simples y del orden del $100.(1-2^{-(r.n+k+s)})$% de todos los múltiples en los planos de la PLA r-valuada.

Demostración:

La detección de todos los fallos de contacto simples queda probada con los lemas 5.1 y 5.2.

Los lemas 5.3 y 5.4 expresan los números de fallos múlti-

ples no detectables en cada uno de los planos de la PLA. Con los lemas 5.5 y 5.6 queda probado que la detectabilidad de un fallo múltiple en uno de los planos no se ve afectada por la existencia de un fallo en el otro plano. Por tanto, el número total de posibles fallos múltiples no detectables en ambos planos de la PLA puede obtenerse como el producto de los números de fallos múltiples en los planos NOR y TSUM, incluidos los casos de no fallo en uno y otro plano, menos el caso en que ambos planos estén libres de fallos:

$$Nnd_{PLA} = (Nnd_{NOR}+1)\cdot(Nnd_{TSUM}+1) - 1$$
$$= 2^{(k+s)m+(k+s-1)rn} - 1 \qquad (5.11)$$

Por otra parte, el número total de posibles fallos múltiples en los (k+s).(r.n+m+1) puntos de cruce de ambos planos de la PLA es:

$$N_{PLA} = \sum_{\mu=2}^{\mu=(k+s)(rn+m+1)} \binom{(k+s)(rn+m+1)}{\mu} =$$

$$= 2^{(k+s)(rn+m+1)} - (k+s)(rn+m+1) \qquad (5.12)$$

De las expresiones (5.11) y (5.12) resulta que la fracción de fallos múltiples no detectables es:

$$Rnd_{PLA} = \frac{Nnd_{PLA}}{N_{PLA}} = \frac{2^{(k+s)m+(k+s-1)rn} - 1}{2^{(k+s)(rn+m+1)} - (k+s)(rn+m+1) - 1} \qquad (5.13)$$

Aunque se trate de una PLA de tamaño reducido, el término -(k+s).(r.n+m+1)-1 de la expresión (5.13) es despreciable frente a la potencia de 2 del denominador, y el -1 lo es frente

a la potencia de 2 del numerador, obteniéndose:

$$Rnd_{PLA} \approx 2^{-(r.m+k+s)} \qquad (5.14)$$

El porcentaje de fallos múltiples detectables puede obtenerse como:

$$Pd_{PLA} = 100.(1-Rnd_{PLA})\% \qquad (5.15)$$

el cual, con la aproximación indicada en el párrafo anterior, coincide con la cantidad que se quería demostrar.

La fracción de fallos de contacto detectados mediante el procedimiento de test que estamos considerando es tanto más pequeña cuanto menor sea el tamaño de la PLA. Aun así, considerando como ejemplo una PLA tetravaluada de tamaño muy reducido, con n=2 entradas y k+s=10 líneas producto, se obtiene un porcentaje de fallos múltiples detectados:

$$P_{PLA} \approx 100.(1 - 2^{-18})\% = 99.9996\%.$$

El porcentaje de detección de fallos indicados en el teorema 5.1 se refiere sólo a los planos de la PLA. El patrón de test T^0 que se aplica al principio de la secuencia de test tiene por objeto detectar la ausencia de cualquier transistor de conexión del circuito SLP con el plano NOR (ver figura 5.3). Asimismo, este patrón permite detectar cualquier anclaje a 0 en las líneas de salida del SLP. Los anclajes a 1 de estas líneas

son también detectables ya que producen vectores de salida de la PLA nulos durante la aplicación de patrones del tipo T^1.

Por otra parte, en las celdas LITE y LITEST, el test permite detectar defectos que producen una codificación incorrecta de las entradas r-valuadas o un fallo en el mecanismo de anulación de las salidas (líneas de literal). Esto último puede deberse a un mal funcionamiento de LITEST, a un defecto en la parte binaria de las celdas LITE donde actuan las señales de anulación (ver figura 4.9), o a una falta de conexión entre las celdas.

Aunque no efectuemos un análisis exhaustivo del grado de cobertura que se obtiene para los fallos en estas celdas, pondremos de manifiesto cómo se detectan algunos de ellos durante el test.

Cuando se aplican patrones de test $T^1_{i,\alpha}$ las entradas r-valuadas toman los valores:

$$\{X_T ; X_1 , X_2 , \ldots , X_n \} \equiv \{\alpha ; \alpha , \alpha , \ldots , \alpha\}$$

En ausencia de fallos, todas las líneas de salida de las celdas LITE deben mantenerse a 0. La activación erronea de una o varias de estas líneas es detectable. En efecto, dado que se aplican patrones $T^1_{i,\alpha}$ para activar todos los puntos de cruce de líneas producto con la línea de salida erronea, basta con que exita un transistor en alguno de dichos puntos de cruce

para que se produzca un vector de salida de la PLA nulo (P'=0) y se altere la secuencia correcta de la paridad acumulada.

Cuando se aplican patrones de test $T^2_{i,j,1}$ las entradas r-valuadas toman los valores:

$$\{X_T ; X_1, .., X_j, .., X_n\} \equiv \{\alpha; \alpha, .., \alpha, X_j=1, \alpha, .., \alpha\}, \quad \alpha \neq 1$$

Todas las líneas de salida de las celdas LITE, salvo la X_j^1 (perteneciente a la j-ésima celda LITE), deben ser 0. Si por un fallo la línea X_j^1 también se mantiene a 0, todos los patrones de test de la subsecuencia $T^2_{i,j,1}$, i=1,2,..,k+s, producen vectores de salida con paridad P'=1. En consecuencia, la contribución neta de esta subsecuencia a la paridad acumulada es $P_A=0$, y el fallo se detecta.

5.4 Conclusión.

Tras un planteamiento de los problemas que conlleva el test de circuitos integrados en general (sección 5.1) y de las PLAs en particular (sección 5.2), hemos presentado (sección 5.3) un esquema de test funcional de PLAs multivaluadas, el cual implica un incremento de superficie similar al de esquemas existentes para PLAs binarias del mismo número de líneas de literal, salidas, y líneas producto.

Dada una PLA r-valuada con n entradas, m salidas, y k términos producto, los planos de esta se amplían con 1 línea de salida y un máximo de s=2 líneas producto. Para activar selectivamente las líneas producto se añade un registro de desplazamiento de k+s bit. A fin de facilitar el acceso individual a los puntos de cruce del plano NOR se utiliza un mecanismo de anulación selectiva de líneas de literal: las n entradas r-valuadas se decodifican mediante celdas generadoras de literales del tipo NOR, cuyas salidas se controlan mediante una celda adicional del mismo tipo.

El test de la PLA se realiza aplicando una secuencia de 1+r.(k+s).(n+1) patrones (independientes de la particulariza-ción de la PLA) y comprobando la paridad acumulada de las salidas. Se efectúan 1+k+s+n.r comparaciones de la paridad acumulada, cuya secuencia de valores correcta es comprobable mediante un circuito muy sencillo (un contador módulo 2).

La circuitería añadida para el test no deteriora la velocidad de funcionamiento de la PLA. Además, la frecuencia de funcionamiento de la PLA en modo de test puede coincidir con la de su funcionamiento normal.

Con el procedimiento de test propuesto se consigue detectar cualquier fallo de contacto simple y del orden del $100.(1-2^{-(r.n+k+s)})\%$ de todos los fallos de contacto múltiples en los planos de la PLA, detectándose además fallos en los circuitos añadidos y en las celdas generadoras de literales.

Evidentemente, la cobertura de fallos puede aumentarse haciendo más comparaciones de la paridad acumulada, pero estas comparaciones serían dependientes de la particularización de la PLA.

En comparación con un procedimiento de test de PLAs NMOS binarias análogo [183], el procedimiento propuesto proporciona la misma cobertura de fallos de contacto simples y múltiples. La secuencia de test de la PLA r-valuada incluiría 2.(k+s) patrones más que la aplicada a la binaria (suponiendo que ambas tienen el mismo número de líneas producto), pero el tamaño del circuito comprobador de paridad de las salida se reduciría en un factor $\log_2(r)$.

En el capítulo 6 se incluye a modo de ejemplo un diseño de PLA 4-valuada integrada, la cual contiene gran parte de la circuitería necesaria para su comprobación mediante el procedimiento propuesto.

CAPITULO 6

REALIZACIONES INTEGRADAS Y RESULTADOS DE SIMULACION

En este capítulo presentamos posibles realizaciones integradas de circuitos propuestos en capítulos anteriores, así como resultados correspondientes a su simulación eléctrica mediante el programa SPICE.

Las secciones 6.2, 6.3, y 6.4, se refieren, respectivamente, a las celdas generadoras de literales, a las distintas realizaciones de los planos de la PLA multivaluada, y a las celdas de memoria en modo de corriente, propuestas en el capítulo 4.

La sección 6.5 describe el diseño de un prototipo de chip consistente en una PLA CMOS tetravaluada, la cual incluye la mayor parte de la circuitería necesaria para su comprobación mediante el procedimiento de test propuesto en el capítulo 5.

6.1 Introducción.

Para evaluar las posibles ventajas que tiene la utilización de una lógica no binaria en la síntesis de circuitos digitales deben considerarse sus realizaciones físicas a nivel de circuito integrado.

Es claro que cuanto mayor sea la base de la lógica utilizada, más se reduce la superficie ocupada por las interconexiones que transportan la información codificada en dicha base. Una reducción en el número de dispositivos activos también implica en general una reducción de superficie, tanto de la asociada con los propios dispositivos como la correspondiente a las interconexiones de estos.

Sin embargo, para evaluar el posible ahorro de tamaño del circuito integrado no basta con considerar el número de dispositivos y líneas de comunicación. La necesidad de introducir líneas adicionales para establecer niveles de referencia y/o el mayor tamaño requerido para los dispositivos activos cuando estos han de operar con más de dos niveles lógicos, pueden llevar a una realización integrada del circuito multivaluado poco favorable.

Por este motivo, para cada uno de los circuitos propuestos presentamos en primer lugar una posible descripción gráfica de las máscaras para su integración. Estas descripciones gráficas no sólo permiten evaluar el tamaño real de los circuitos, sino

además, poder efectuar simulaciones más exactas de su funcionamiento, introduciendo en ellas los efectos producidos por las interconexiones.

A continuación de la descripcion gráfica de cada circuito presentamos los resultados de su simulación eléctrica. En el apéndice B de esta memoria se incluyen listados de los ficheros de descripción SPICE de los circuitos simulados.

Los datos de proceso, reglas geométricas de diseño, y parámetros eléctricos de transistores que utilizamos para la obtención de los resultados expuestos en las secciones que siguen, son los proporcionados por el Laboratorio de Microelectrónica de la Universidad Católica de Louvain (Bélgica), los cuales corresponden a una tecnología CMOS de 3 micrómetros, puerta de polisilicio, pozo n, y un solo nivel de metalización (ver sección 2.3).

Como editor gráfico hemos utilizado el programa LUCIE, desarrollado por el grupo de arquitectura de ordenadores del instituto IMAG (Grenoble).

Las simulaciones eléctricas se han realizado con el programa PSPICE (adaptación de SPICE para ordenadores personales comercializada por MicroSim Corp.).

Para algunas de las descripciones gráficas presentadas ha sido posible, a través del Departamento de Diseño del Centro

Nacional de Microelectrónica, efectuar una extracción automática de parámetros eléctricos mediante el extractor incorporado en el paquete de diseño VTI. Para el resto de las descripciones gráficas hemos estimado las capacidades asociadas a las interconexiones, utilizando para ello los datos del proceso CMOS mencionado anteriormente.

La descripción gráfica del chip multivaluado presentada en la sección 6.4 ha sido comprobada mediante el verificador de reglas de diseño incorporado en el mismo paquete VTI. En la actualidad, este circuito integrado se encuentra en proceso de fabricación, formando parte del proyecto MPC-88 organizado por el Centro Nacional de Microelectrónica.

6.2 Celdas generadoras de literales.

En la sección 4.3 presentamos dos posibles esquemas de celdas generadoras de literales tetravaluadas. En las realizaciones CMOS de estas celdas (figuras 4.7 y 4.10), las dimensiones de los transistores PMOS del detector de umbrales de corriente son tales que, el transistor que fija el umbral mínimo (U=0.5), tiene el ancho de puerta mínimo permitido (WG_{min}=3 lambdas, siendo lambda=1.5µm).

De las dos celdas generadoras de literales, la realización integrada que ocupa más superficie es la correspondiente a la

celda NOR. Esto no sólo es debido a su mayor número de transistores, sino al mayor tamaño de los transistores PMOS de la parte binaria, y a las líneas necesarias para las entradas de control.

6.2.1 Descripción gráfica.

La figura 6.1 muestra una descripción gráfica, expresada en lambdas, de las máscaras de una celda tipo NOR tetravaluada (celda LITE). Ocupa una superficie de 25200 lambdas^2 (S_{4L} = 567x10^{-4} mm^2), valor que puede reducirse ya que el diseño de la figura 6.1 es susceptible de optimización.

El trazado dado a las líneas de alimentación, tensión de referencia, y entradas de control (X_{cl}, l=0,1,2,3), permite el enlace directo de celdas contiguas gobernadas por las mismas entradas de control.

Si sólo se requiere una línea de entrada de control para anular simultáneamente todas las salidas de literal, pueden eliminarse las otras tres, reduciendose la superficie de la celda a 24400 lambas^2 (0.05mm^2).

En una celda generadora de literales tipo NAND no están presentes las líneas de entrada de control y emplea 6 transistores menos que una celda tipo NOR, motivo por el cual es

posible realizarla en una superficie inferior a 155x115 lambdas² (0.04mm²).

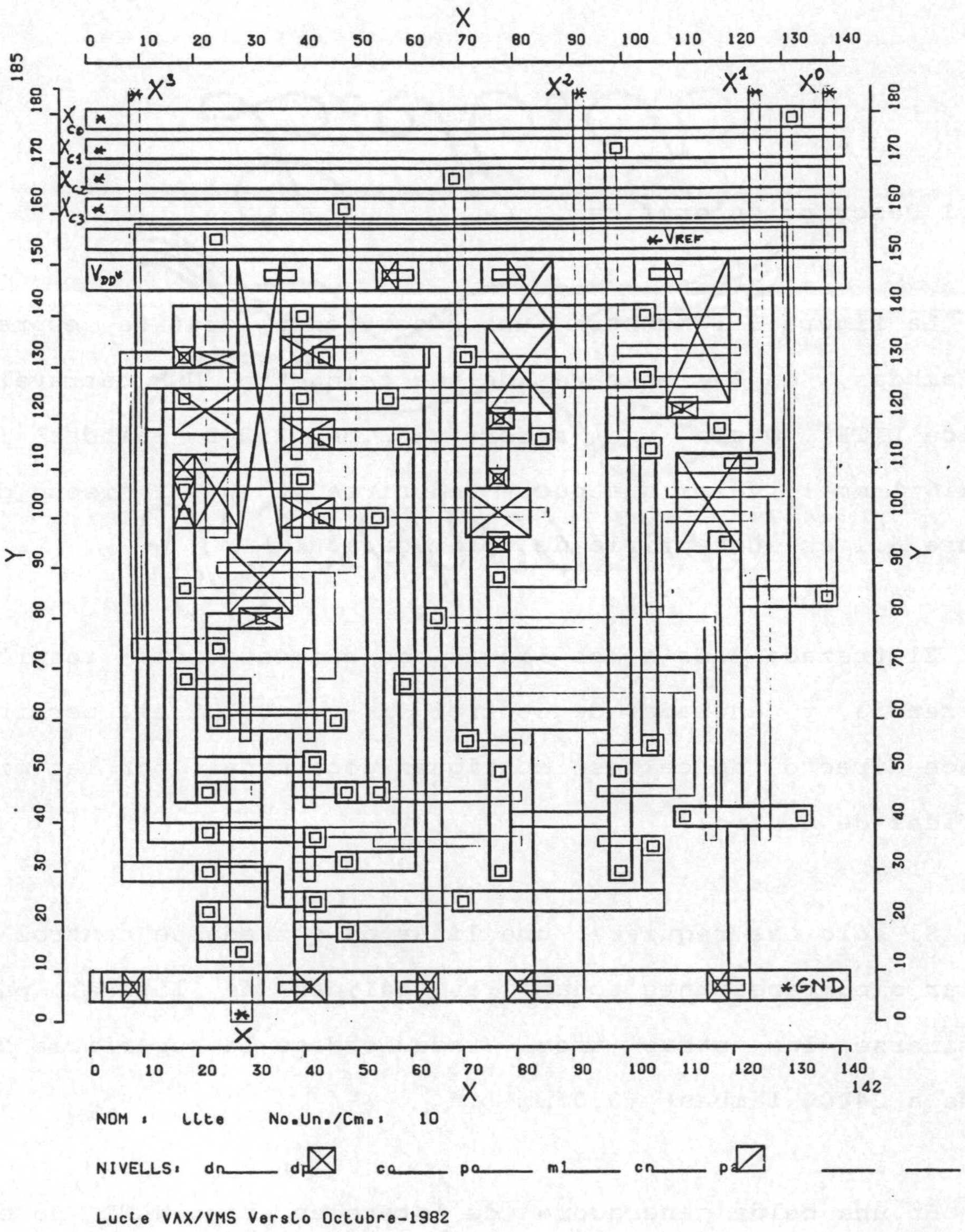


FIGURA 6.1: Descripción gráfica de las máscaras de una celda generadora de literales tetravaluada de tipo NOR.

6.2.2 Resultados de simulación.

Hemos realizado simulaciones del comportamiento estático y dinámico de las dos celdas generadoras de literales tetravaluadas, considerando distintos valores del nivel de corriente unidad (I_0). La tensión de puerta de los transistores PMOS que fijan los umbrales de corriente se genera a partir de una fuente de corriente I_0, utilizando un transistor PMOS adicional incluido en la descripción del circuito (ver listado en el apéndice B).

CELDA GENERADORA DE LITERALES TIPO NOR:

Respuesta DC:

La figura 6.2 muestra la respuesta DC del circuito cuando se aplica una rampa de corriente de 0 a $3I_0$ en la entrada, para algunos valores de I_0 comprendidos entre 10µA y 50µA. Los nudos 3, 6, y 9, son las salidas del detector de umbrales de corriente, y los nudos 10, 11, 12, y 13, las salidas de literal X^0, X^1, X^2, y X^3, respectivamente.

Como se observa en la figura, las transiciones de tensión en las salidas tienen lugar, con bastante precisión, cuando la corriente de entrada toma los valores $0.5I_0$, $1.5I_0$, y $2.5I_0$.

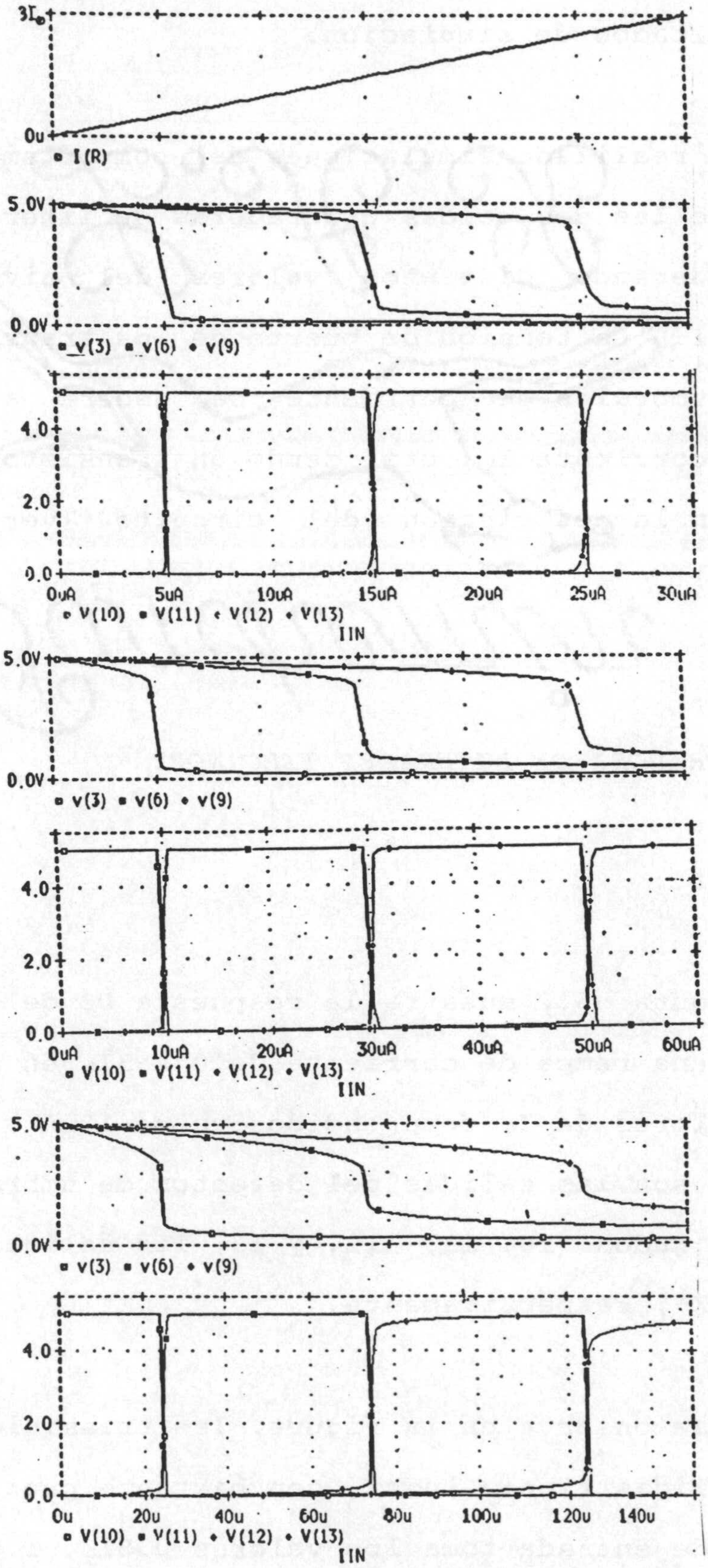


FIGURA 6.2: Respuesta DC de la celda generadora de literales tipo NOR. (a) Corriente de entrada. (b) Salidas del detector de umbrales de corriente y salidas de literal con I_0 =10μA. (c) Con I_0 =20μA. (d) Con I_0 =50μA.

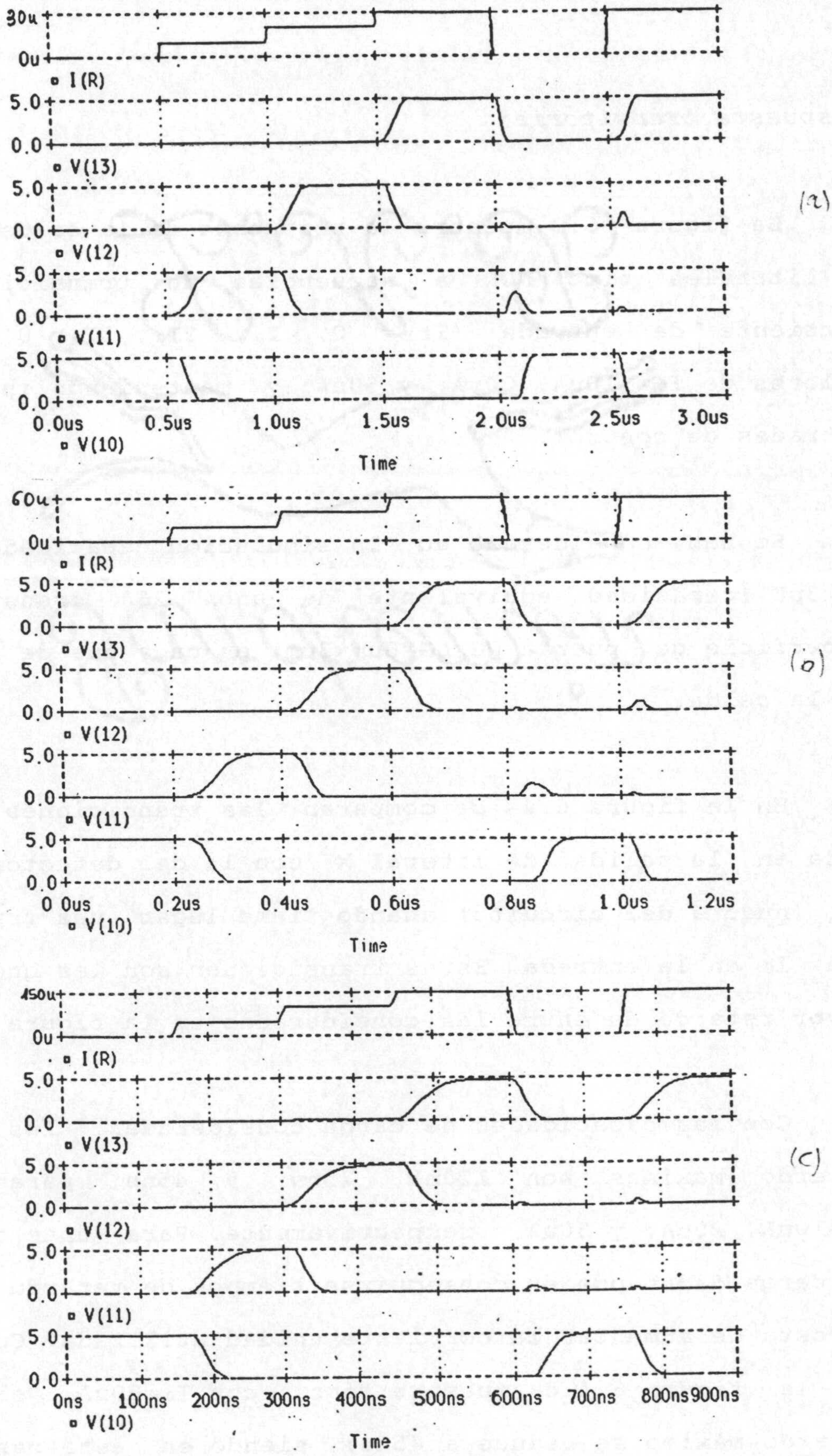
Respuesta transitoria:

La figura 6.3 muestra la respuesta de la celda generadora de literales tipo NOR a secuencias de transiciones de la corriente de entrada (I_{IN}= 0, I_0, $2I_0$, $3I_0$, 0, $3I_0$), para valores de I_0= 10μA, 20μA, y 50μA; y manteniendo inactivas las entradas de control.

Se han considerado en la simulación capacidades de carga de 3pF (capacidad equivalente de unos 260 transistores con superficie de puerta WGxLG=6μmx3μm) en cada una de las salidas de la celda.

En la figura 6.4 se comparan las transiciones de baja a alta en la salida de literal X^1 con la del detector de umbral 0.5 (nudo 3 del circuito) cuando tiene lugar una transición de 0 a I_0 en la entrada. Estas transiciones son las que presentan mayor retardo de entre las consideradas en la figura 6.4.

Con las capacidades de carga consideradas, los tiempos de retardo máximos son 120ns, 75ns, y 45ns, para valores de I_0=10μA, 20μA, y 50μA, respectivamente. Para unas capacidades de carga dadas pueden conseguirse tiempos de retardo inferiores a costa de aumentar la corriente unidad utilizada. Concretamente, la figura 6.4.c muestra cómo, con I_0=50μA, el tiempo de retardo máximo se reduce a 45ns; siendo en este caso superior el retardo producido por la parte binaria de la celda que el debido al detector de umbrales de corriente.

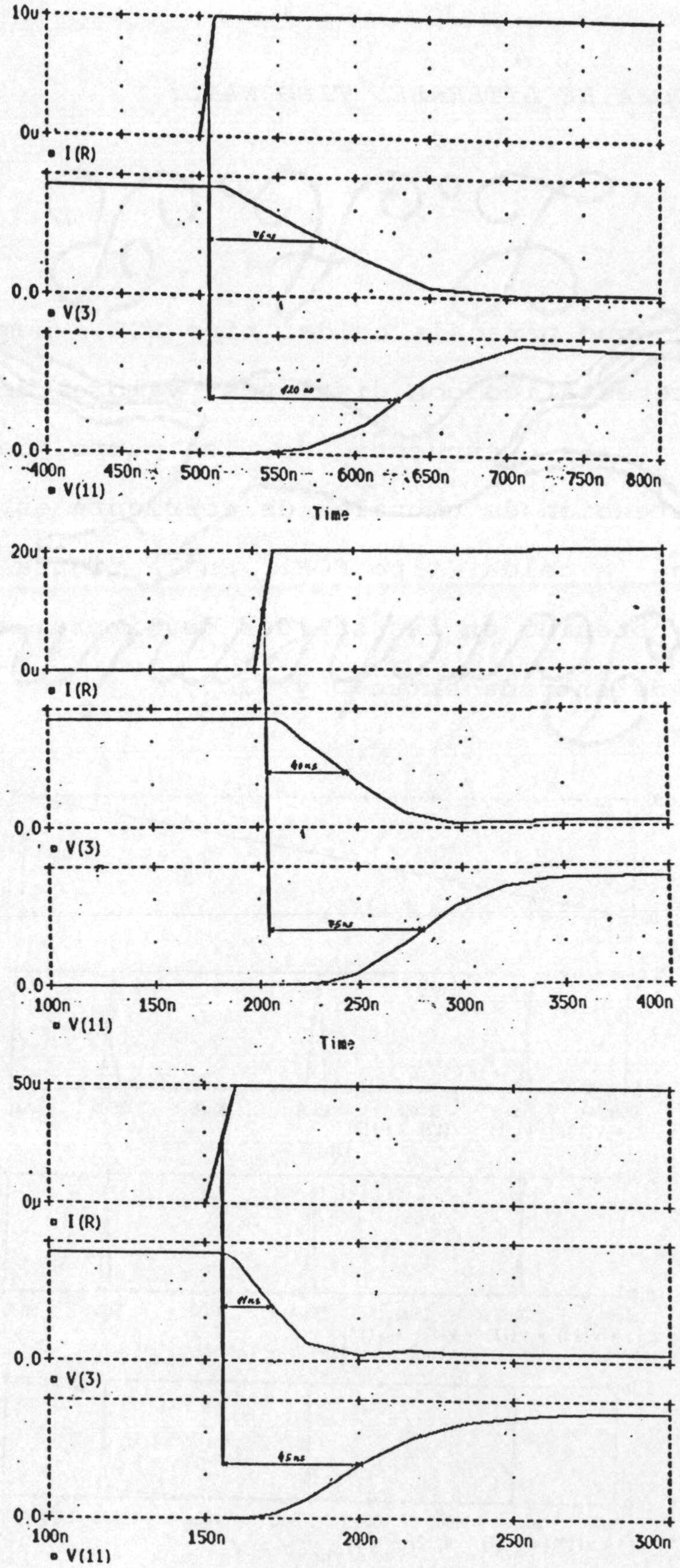


(a)

(b)

(c)

FIGURA 6.3: Respuesta transitoria de la celda generadora de literales tipo NOR. (a) Entrada y salidas con I_0 =10µA. (b) Con I_0 =20µA. (c) Con I_0 =50µA.



(a)

(b)

(c)

FIGURA 6.4: Transiciones en la salida del detector de umbral 0.5 y en la salida de literal X^1 de la celda tipo NOR. (a) Con I_0 =10µA. (b) Con I_0 =20µA. (c) Con I_0 =50µA.

CELDA GENERADORA DE LITERALES TIPO NAND:

Respuesta DC:

Al igual que para la celda tipo NOR, hemos simulado el comportamiento estático con distintos valores de la corriente de referencia. La respuesta de la parte de la celda que realiza la detección de umbrales de corriente es idéntica a la obtenida para la celda tipo NOR. En la figura 6.5 se muestra la respuesta obtenida en las salidas de literal al hacer variar la corriente de entrada entre 0 y $3I_0$.

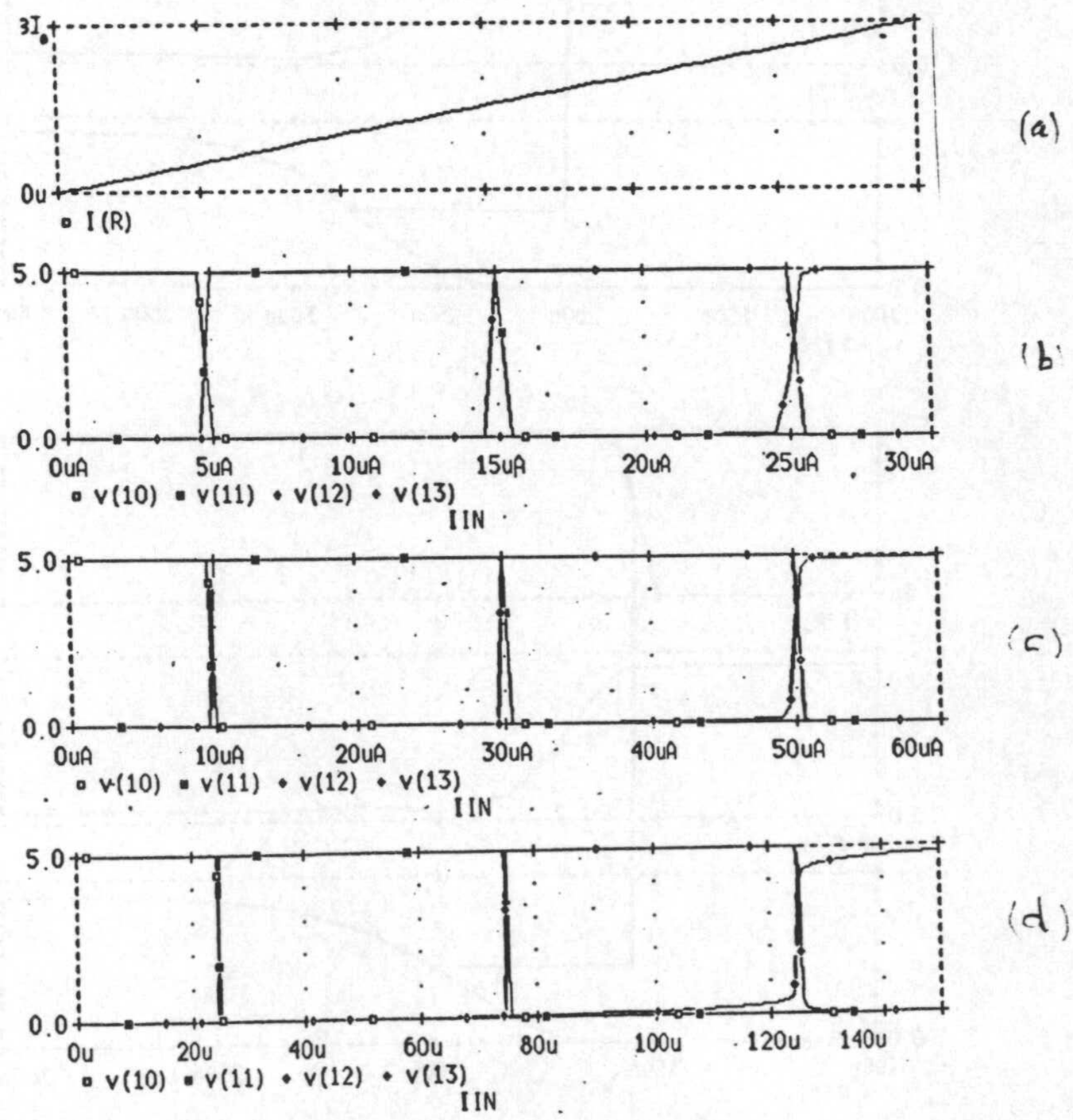


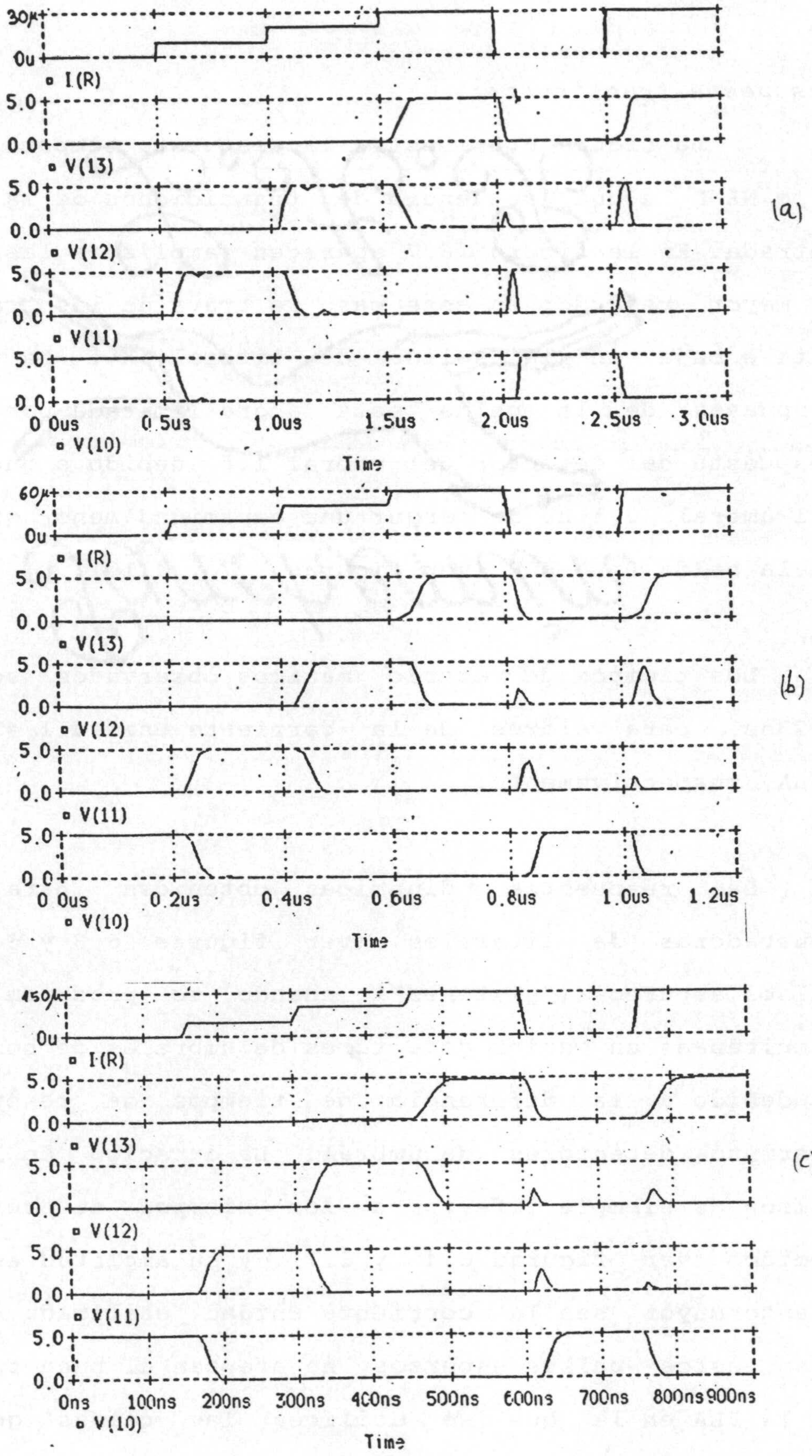
FIGURA 6.5: Respuesta DC de la celda generadora de literales tetravaluada tipo NAND. (a) Corriente de entrada. (b) Salidas con I_0 =10µA. (c) Con I_0 =20µA. (d) Con I_0 =50µA.

Respuesta transitoria:

La figura 6.6 muestra la respuesta temporal de la celda tipo NAND a una secuencia de transiciones de la corriente de entrada. En la figura 6.7 aparecen ampliadas las transiciones de mayor retardo; en este caso se trata de las transiciones de alta a baja en la salida de literal X^1. La velocidad de respuesta de la celda está ahora limitada por el tiempo de respuesta del detector del umbral 1.5, debido a que el detector del umbral 0.5 ha de cargar una capacidad menor que en el caso de la celda tipo NOR (ver figuras 4.7 y 4.10).

Los tiempos de retardo máximos observados son 82ns, 44ns y 23ns, para valores de la corriente unidad I_0=10µA, 20µA, y 50µA, respectivamente.

Las respuestas dinámicas obtenidas para las celdas generadoras de literales (ver figuras 6.3 y 6.6) presentan pulsos espúreos ("glitches") cuando se producen transiciones simultáneas en varios detectores de umbrales de corriente. Esto es debido a la diferencia de tiempos de respuesta de los distintos detectores de umbral. La duración de los referidos pulsos es siempre inferior a los tiempos de retardo máximos medidos (ver figuras 6.4 y 6.7), y su amplitud es tanto menor cuanto mayor sea la corriente unidad utilizada. En cualquier caso, estos pulsos espúreos no afectan al buen funcionamiento de la PLA en la que se utilicen las celdas generadoras de literales si se adopta la precaución de evaluar las salidas de la misma en el momento apropiado.



(a)

(b)

(c)

FIGURA 6.6: Respuesta transitoria de la celda generadora de literales tipo NAND. (a) Entrada y salidas con I_0 =10µA. (b) Con I_0 =20µA. (c) Con I_0 =50µA.

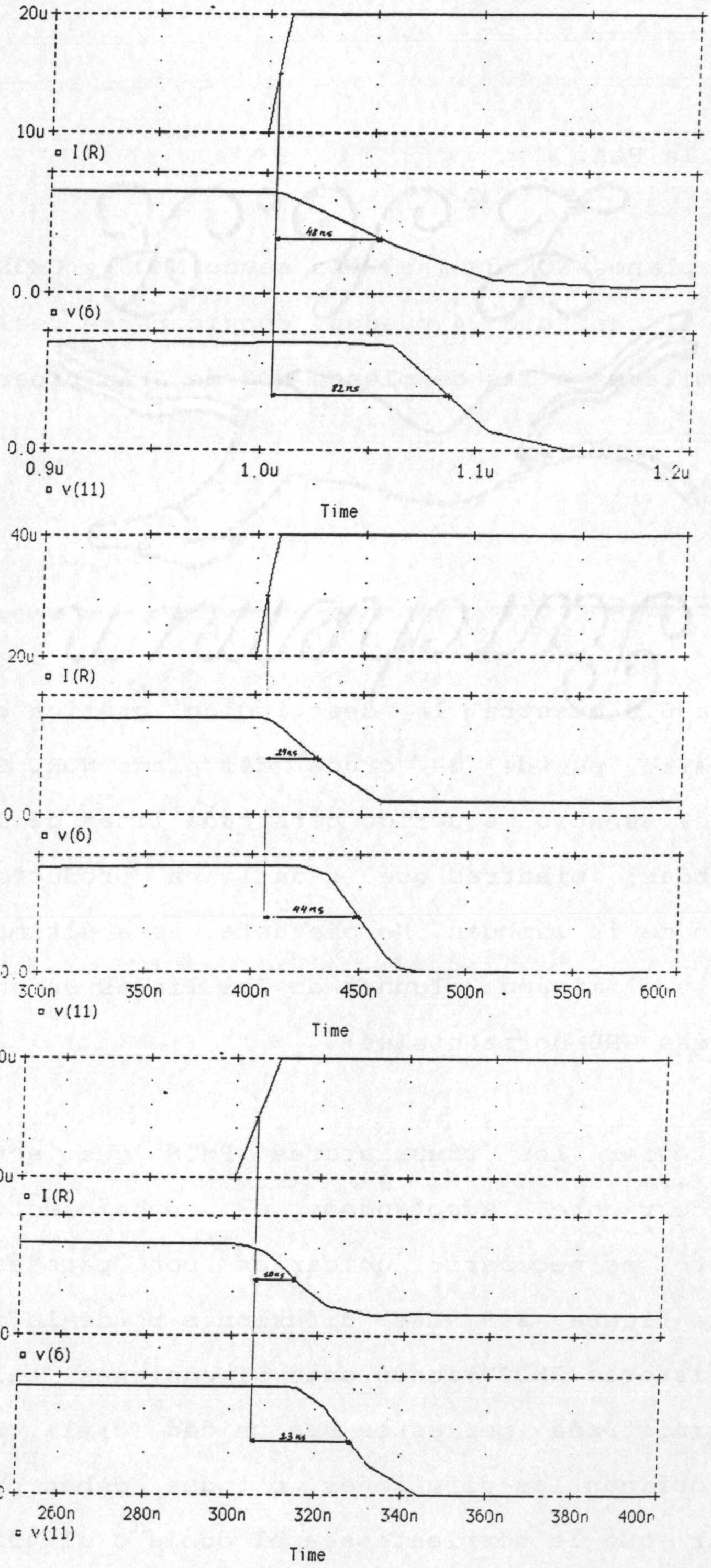


FIGURA 6.7: Transiciones en la salida del detector de umbral 1.5 y en la salida de literal X^1 de la celda tipo NAND. (a) Con I_0 =10μA. (b) Con I_0 =20μA. (c) Con I_0 =50μA.

6.3 Planos de la PLA.

Para los planos NOR de las PLAs seudo-NMOS y CMOS dinámica propuestas en el capítulo 4 pueden considerarse realizaciones integradas similares a las de planos NOR de PLAs binarias [127, 193].

6.3.1 Descripción gráfica.

La figura 6.8 muestra la descripción gráfica correspondiente a cuatro puntos de cruce del plano NOR. Según esta descripción, el espacio requerido para cada línea de literal L_i es de 10 lambdas; mientras que cada línea producto exige un ancho efectivo de 14 lambdas. No obstante, esta última cantidad puede reducirse eliminando algunas de las líneas de anclaje del sustrato (líneas GND horizontales).

Para integrar los transistores PMOS que actúan como fuentes de corriente, adaptándose al ancho efectivo de las líneas producto, es necesario ubicarlos por parejas como se muestra en la figura 6.9. Las difusiones p^+ de la figura que definen transistores PMOS tienen unas dimensiones tales que la corriente suministrada por estos sea unidad ($c_j=1$). Las líneas discontínuas definen las difusiones p^+ que deben programarse para conseguir que la corriente sea el doble o el triple de la unidad ($c_j=2$ ó $c_j=3$).

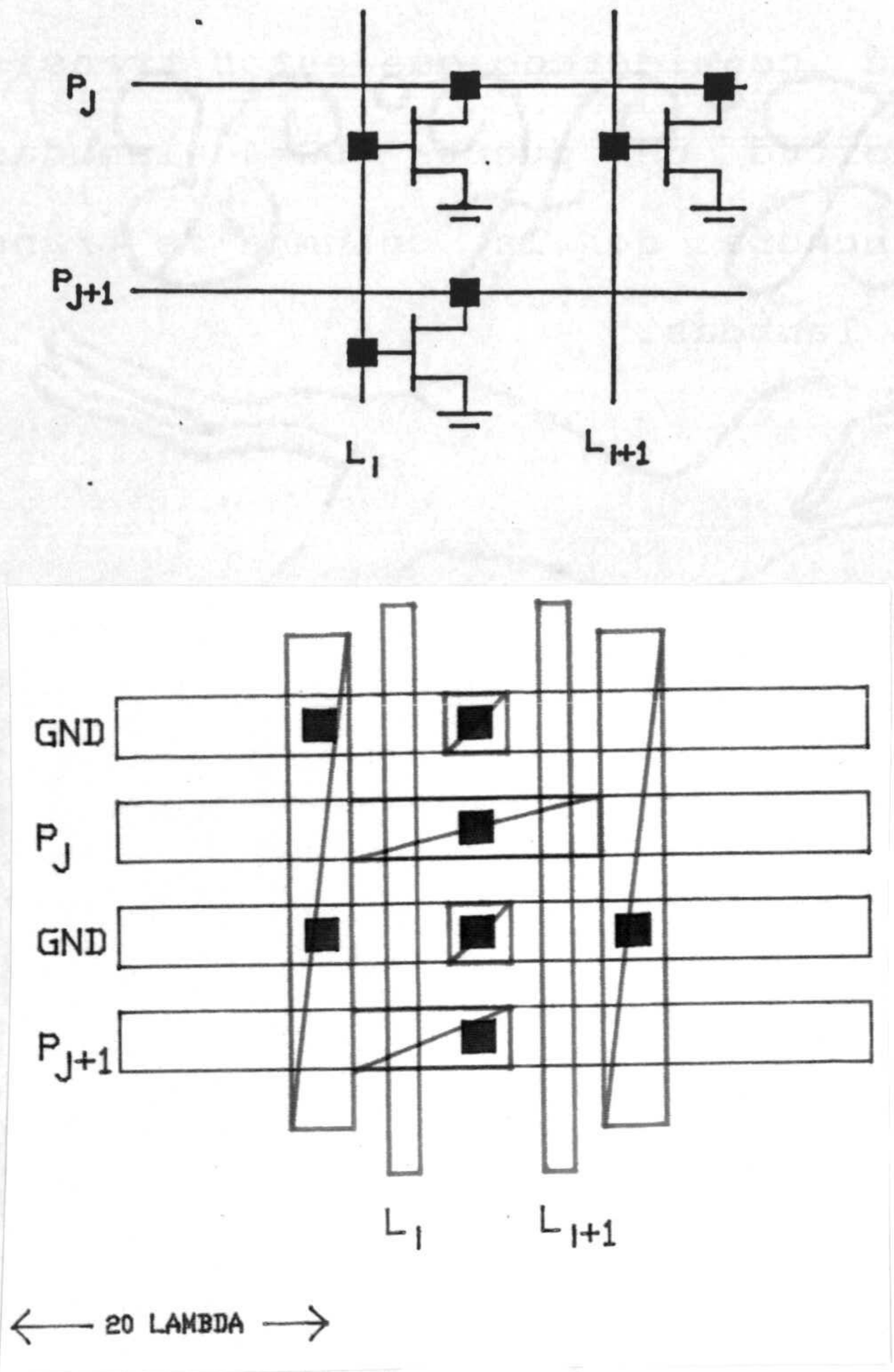


FIGURA 6.8: Descripción gráfica de las máscaras de 4 puntos de cruce del plano NOR.

Las dimensiones mínimas de transistor consideradas son tales que, para una misma tensión de referencia, los niveles de corriente suministrados coincidan con los niveles de entrada a las celdas generadoras de literales. Si $(WG/LG)_{0.5}$ es la relación de tamaño de un transistor PMOS de un detector de umbral 0.5, las relaciones de tamaño de los transistores PMOS que suministren niveles de corriente I_0, $2.I_0$, y $3.I_0$, deben ser: $2.(WG/GL)_{0.5}$, $4.(WG/LG)_{0.5}$, y $6.(WG/LG)_{0.5}$, respectivamente. Si $(WG/LG)_{0.5}$= 3 lambdas/4 lambdas, las relaciones de tamaño que se obtienen son precisamente las indicadas en la

figura 6.9 (se ha considerado que estos transistores tienen todos la misma longitud de puerta LG=4 lambdas. Con estas dimensiones, la anchura de la columna de transistores PMOS resultante es de 55 lambdas.

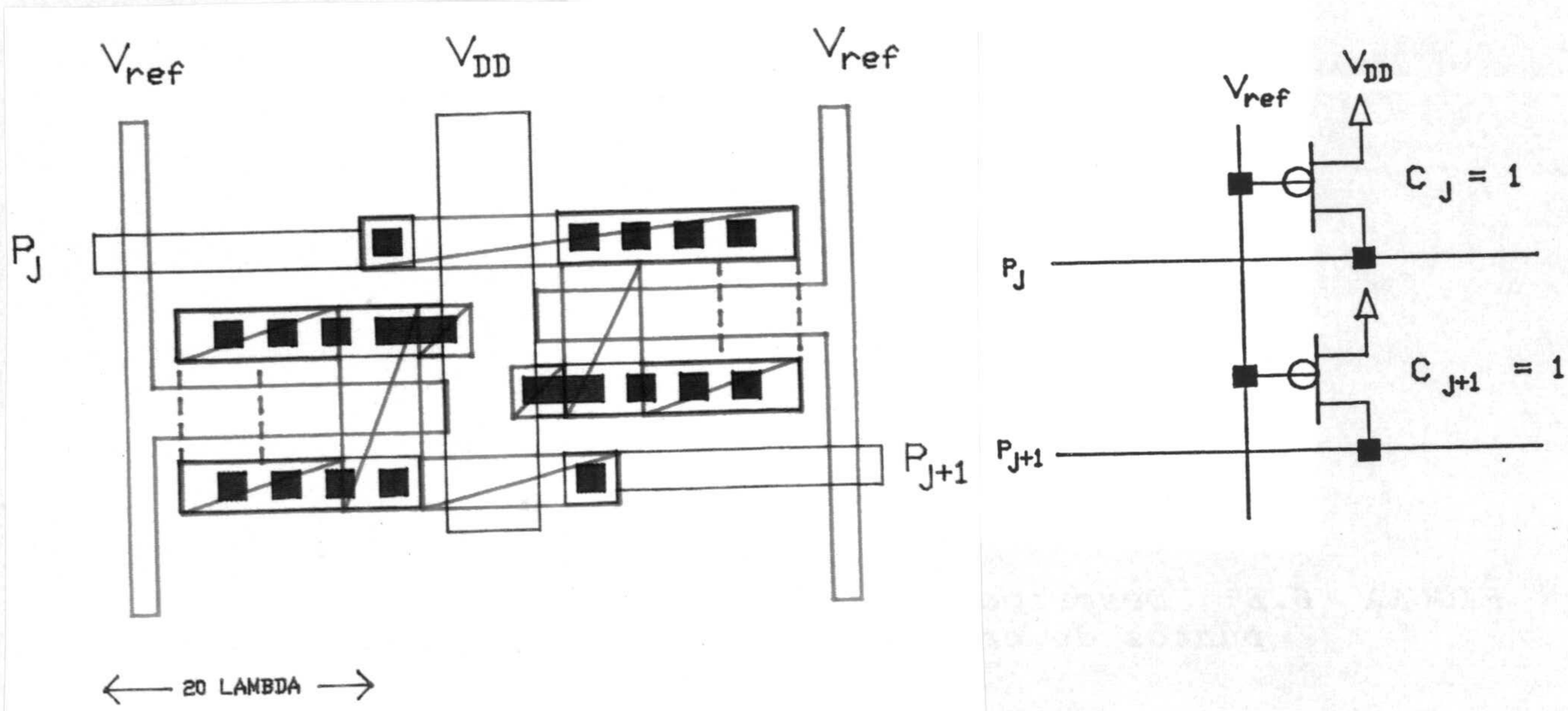


FIGURA 6.9: Descripción gráfica de las máscaras de 2 fuentes de corriente PMOS.

El tamaño de los transistores referidos puede en realidad reducirse para conseguir una columna de 43 lambdas de anchura, de tal manera que un transistor que suministre una corriente unidad tenga ahora una relación de tamaño 4 lambdas /4 lambdas. Para mantener la igualdad de niveles de corriente en las entradas y salidas de la PLA pueden en tal caso adoptarse distintas soluciones: una posibilidad es utilizar tensiones de

referencia distintas para los planos de la PLA y las celdas generadoras de literales. Otra forma alternativa consiste en dimensionar adecuadamente o bien los espejos de corriente de las celdas generadoras de literales, o bien los espejos de entrada de los circuitos que conecten con las salidas de la PLA, para efectuar la transformación de niveles necesaria.

La figura 6.10 muestra una descripción gráfica de las máscaras de dos espejos NMOS del plano TSUM con dos líneas de salida (cuatro puntos de cruce). El ancho efectivo dado a cada línea producto en este plano se ha hecho coincidir con el que tiene en el plano NOR, pero podría disminuirse de 14 lambdas a 12 lambdas. La anchura efectiva de cada línea de salida es 14 lambdas, cantidad también reducible si se disminuye el número de anclajes del sustrato a tierra (líneas GND verticales).

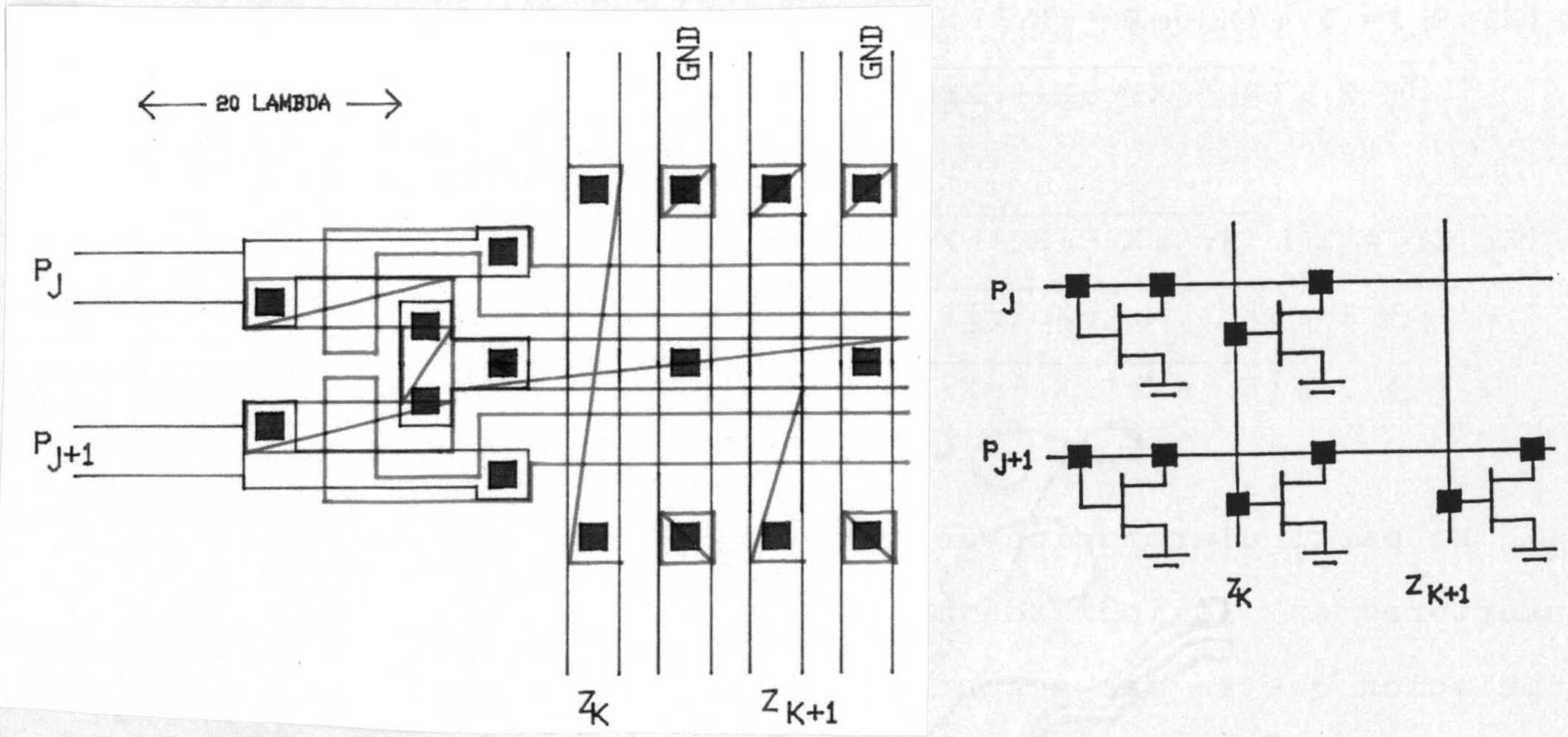


FIGURA 6.10: Descripción gráfica de las máscaras de 4 puntos de cruce del plano TSUM.

6.3.2 Resultados de simulación.

Consideremos como ejemplo los planos de una PLA que sintetice la pareja de funciones tetravaluadas de dos variables definidas por las siguientes tablas:

$f_1(X_1, X_2)$	X_1 = 0	1	2	3
X_2 = 0	1	0	0	0
1	1	0	1	0
2	2	2	2	0
3	1	0	0	0

$f_2(X_1, X_2)$	X_1 = 0	1	2	3
X_2 = 0	1	0	3	0
1	1	0	0	2
2	0	2	3	3
3	1	2	2	0

Las expresiones simplificadas NOR/TSUM de estas funciones son las siguientes (ver sección 4.2):

$$f_1(X_1, X_2) = 1 \cap \overline{(X_1^1 \cup X_1^2 \cup X_1^3 \cup X_2^2)} \boxplus 1 \cap \overline{(X_1^0 \cup X_1^1 \cup X_1^3 \cup X_2^0 \cup X_2^2 \cup X_2^3)} \boxplus 2 \cap \overline{(X_1^3 \cup X_2^0 \cup X_2^1 \cup X_2^3)}$$

$$f_2(X_1, X_2) = 1 \cap \overline{(X_1^1 \cup X_1^2 \cup X_1^3 \cup X_2^2)} \boxplus 2 \cap \overline{(X_1^0 \cup X_1^3 \cup X_2^0 \cup X_2^1)} \boxplus 1 \cap \overline{(X_1^0 \cup X_1^1 \cup X_2^0 \cup X_2^1 \cup X_2^3)} \boxplus 2 \cap \overline{(X_1^0 \cup X_1^1 \cup X_1^2 \cup X_2^0 \cup X_2^3)} \boxplus 3 \cap \overline{(X_1^0 \cup X_1^1 \cup X_1^3 \cup X_2^1 \cup X_2^3)}$$

La particularización de la PLA que realiza las expresiones anteriores es la indicada en la figura 6.11. Los resultados de simulación que se dan a continuación corresponden a las distin

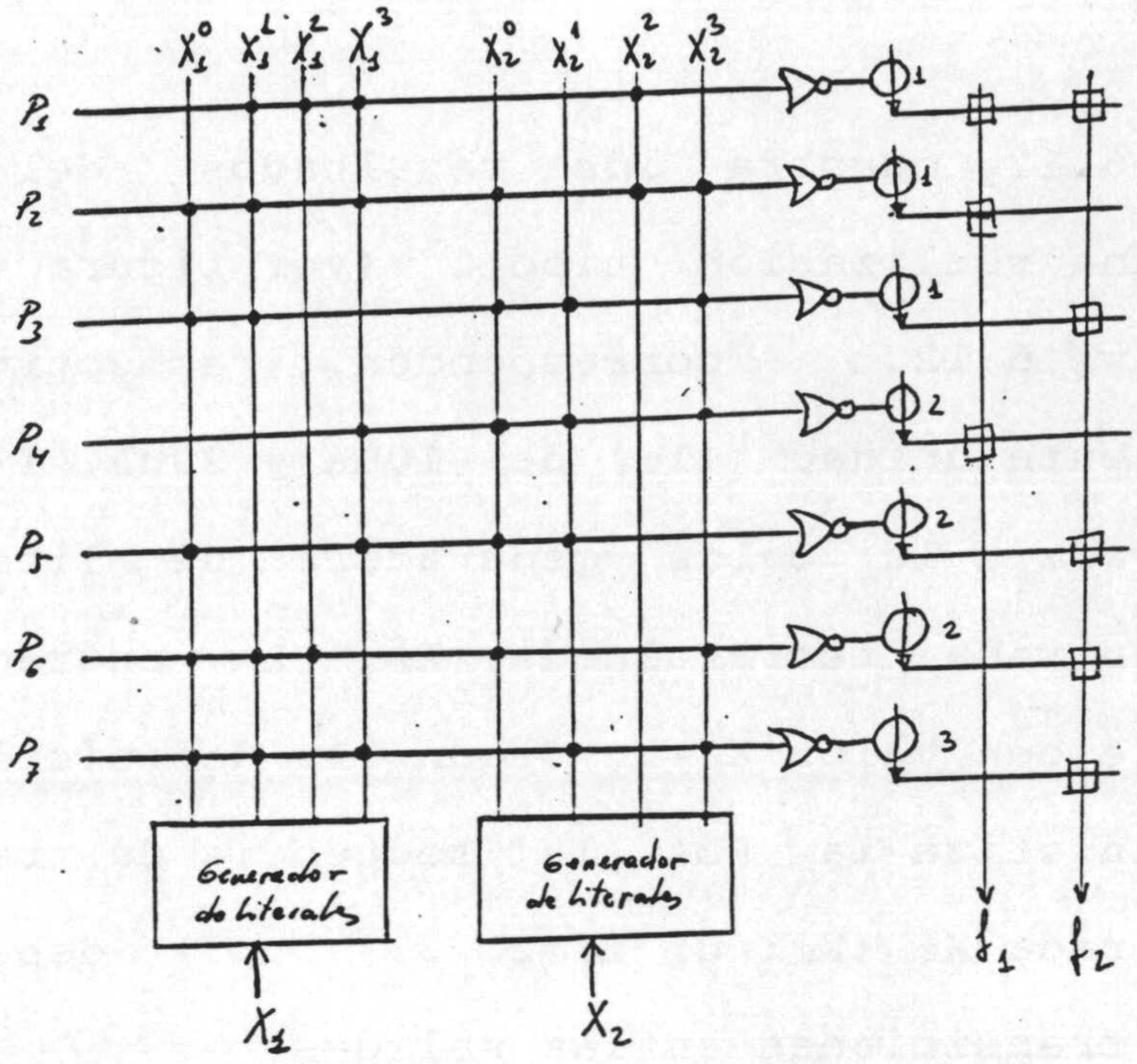


FIGURA 6.11: Particularización de la PLA utilizada en las simulaciones.

tas realizaciones integradas de los planos de la PLA mostrada en dicha figura. Hemos omitido los resultados de simulación de las PLAs del tipo 3, 4, y 6, las cuales presentan comportamientos similares a las realizaciones del tipo 1, 2, y 5, respectivamente.

En las simulaciones hemos considerado que las entradas de literal X_1^l, l=0,1,2,3, permanecen constantes y que se aplica, por medio de una celda generadora de literales del tipo NOR, una secuencia de transiciones en las líneas X_2^l.

PLANOS DE LA PLA DEL TIPO 1:

La figura 6.12 muestra los resultados del análisis transitorio de una realización tipo 1 (ver figura 4.11). Las figuras 6.12.a y 6.12.b corresponden, respectivamente, a niveles de corriente unidad (I_0) de 10μA y 20μA. I(R1) es la corriente de entrada a la celda generadora de literales que decodifica la entrada tetravaluada X_2. La entrada X_1 se mantiene constante con valor $X_1=2$. Según la definición de las funciones que sintetiza la PLA, la secuencia de transiciones aplicada a la entrada X_2 (X_2: 0, 1, 2, 3, 0, 3) debe provocar una secuencia de transiciones en las salidas:

$f_1(2,X_2)$: 0, 1, 2, 0, 0, 0

$f_2(2,X_2)$: 3, 0, 3, 2, 3, 2

Como puede observarse en la figura 6.12, las secuencias de salida que se obtienen son correctas para los dos niveles de corriente unidad utilizados. ID(MPS1) e ID(MPS2) son las corrientes de drenador de dos transistores PMOS utilizados en la simulación como cargas de las líneas de salida (f_1 y f_2) de la PLA. A estos transistores se les ha asignado una relación de tamaño de WG/LG=1.5 (ver apéndice B). Teniendo en cuenta que la relación de tamaño de los transistores NMOS del plano TSUM es WG/LG=1 (ver figura 6.10), la corriente máxima utilizable es, según la ecuación (2.10), $I_{o,max} \approx 63\mu A$. Al tratarse de lógica tetravaluada, la corriente unidad máxima es pues $I_o \approx 21\mu A$. Si se utiliza una corriente superior se aprecia un deterioro de los niveles altos de corriente.

La combinación de entradas $X_1=X_2=2$ hace que la corriente de salida ID(MPS2) sea superior a $3.I_0$, debido a que la suma de corrientes no es truncada en los planos de la PLA.

Las transiciones de salida de mayor retardo en la figura 6.12 son las que corresponden a la conmutación de una fuente de corriente PMOS de magnitud unidad. Este es el caso de la transición de 0 a I_0 en la corriente ID(MPS1), la cual es producida por la fuente de corriente unidad conmutada por la línea producto P_2 del esquema de la figura 6.11.

En la figura 6.13 se muestran ampliadas las transiciones de 0 a I_0 en ID(MPS1) junto con las transiciones de tensión que tienen lugar en la línea de literal X_2^1 (V(22)), en la línea producto P_2 (V(12)), y en la línea de salida f_1 (V(30)).

Respecto de la transición de tensión en la referida línea de literal se observan tiempos de retardo inferiores a 20ns y 15ns para los valores de corriente unidad de 10μA y 20μA respectivamente. Los tiempos de retardo referidos a las transiciones de las entradas, es decir incluyendo el tiempo de retardo de la celda generadora de literales, son aproximadamente 110ns y 65ns.

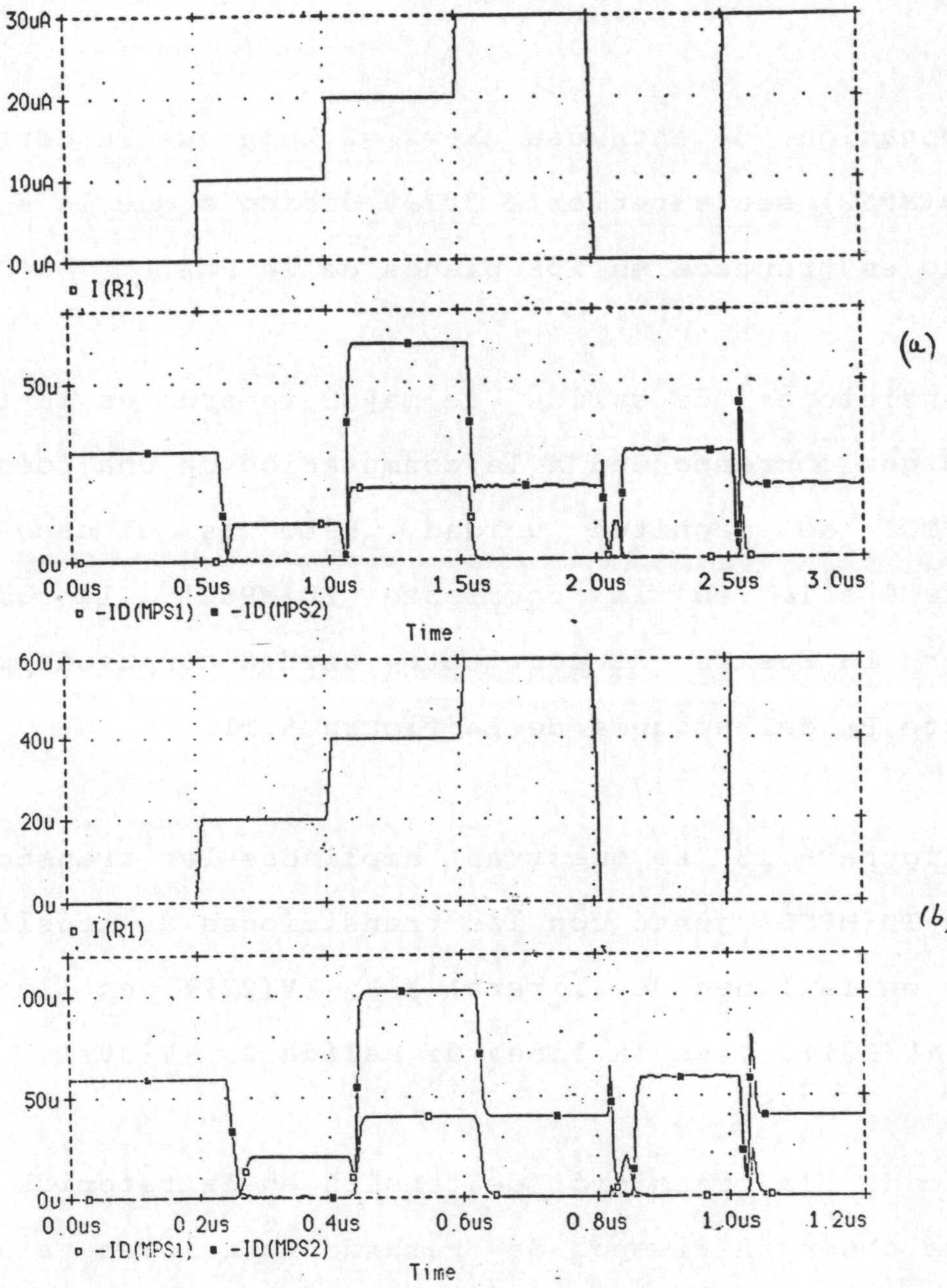


FIGURA 6.12: Respuesta transitoria de la PLA del tipo 1. (a) Con I_0 =10µA. (b) Con I_0 =20µA.

En la figura 6.13, ID(M1212) es la corriente de drenador del transistor NMOS que recibe la corriente de la fuente conmutada por la línea producto P_2. La diferencia de tiempo entre las transiciones de ID(M1212) e ID(MPS1) es una medida del retardo asociado a la carga de las líneas de salida, el cual es, para esta PLA, muy pequeño en comparación con el tiempo de carga de las líneas producto.

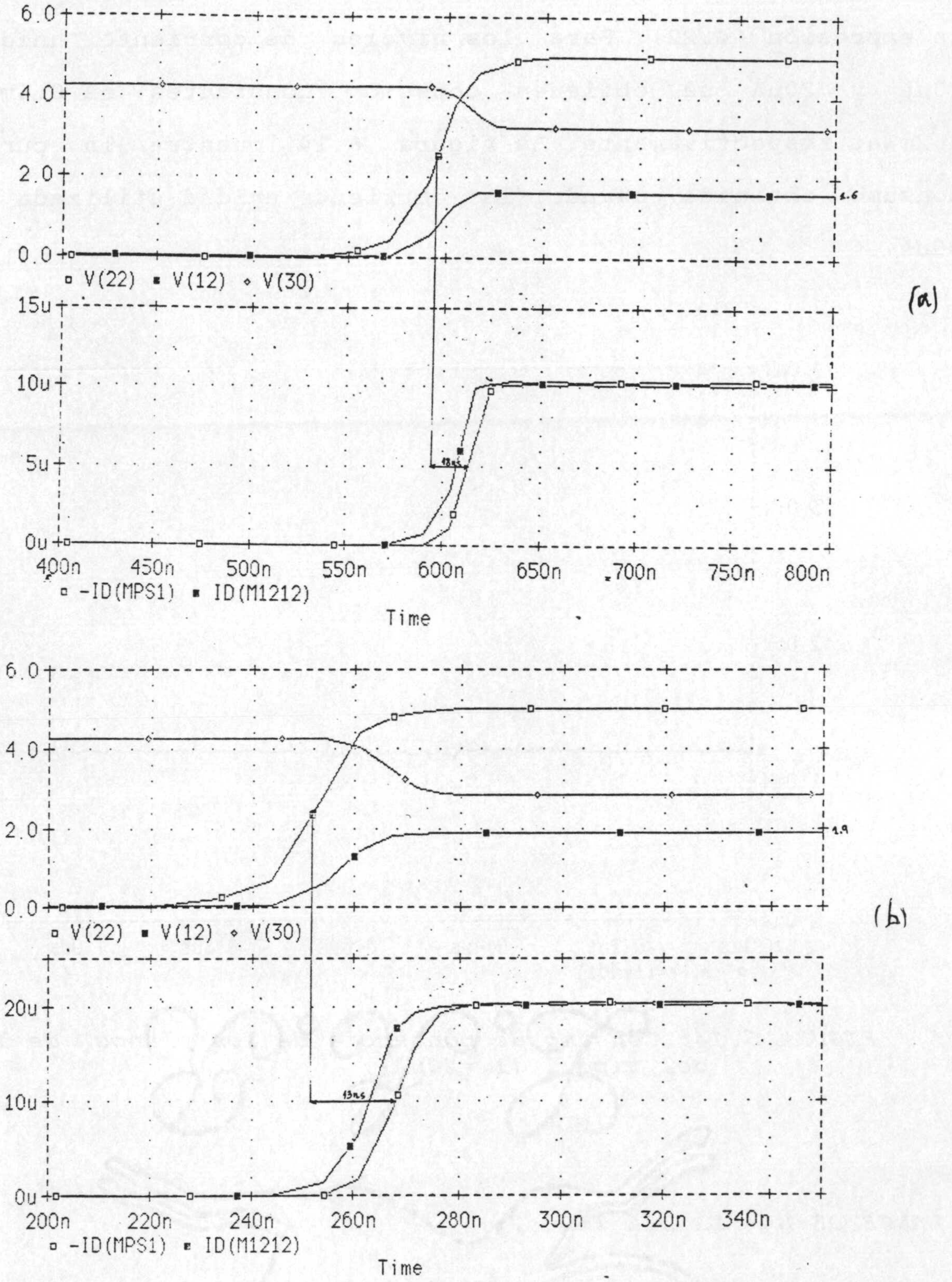


FIGURA 6.13: Transiciones de máximo tiempo de retardo en la PLA del tipo 1. (a) Con I_0 =10µA. (b) Con I_0 =20µA.

El consumo de potencia de los planos de la PLA, excluyendo

el debido a las corrientes de salida, coincide con el dado por la expresión (4.22). Para los niveles de corriente unidad de 10µA y 20µA se obtienen consumos constantes de 0.6mWat y 1.2mWat respectivamente. La figura 6.14 muestra la curva de consumo obtenida cuando la corriente unidad utilizada es de 20µA.

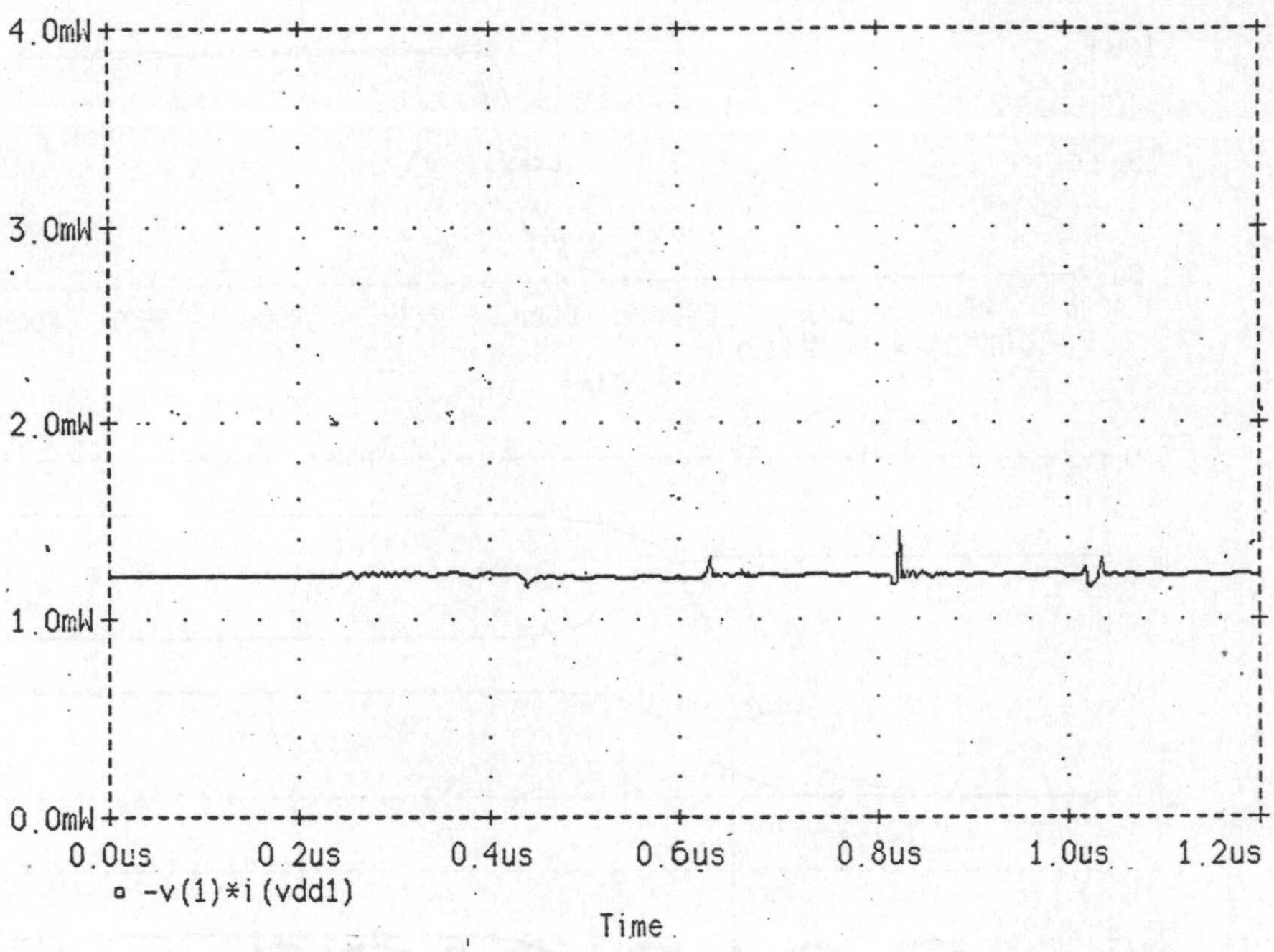


FIGURA 6.14: Consumo de pontencia de los planos de la PLA del tipo 1 (I_0 =20µA).

PLANOS DE LA PLA DEL TIPO 2:

En las simulaciones efectuadas de la PLA del tipo 1 hemos considerado que los niveles de corriente aplicados a la entrada de la celda generadora de literales coinciden en magnitud con

los suministrados por las fuentes de corriente PMOS. Según comentamos al principio de esta sección, refiriéndonos a la realización física de las fuentes de corriente PMOS, no es obligado utilizar los mismos niveles de corriente. Por el contrario, la utilización de niveles distintos puede mejorar la velocidad de operación de la PLA manteniendo limitado el consumo. Si la PLA es de tamaño reducido, como ocurre con la PLA ejemplo que estamos considerando, el retardo asociado a los planos de la misma puede ser inferior al correspondiente a las celdas generadoras de literales. Para disminuir el tiempo de retardo de la PLA (manteniendo constante el consumo) conviene en este caso aumentar los niveles de corriente en las celdas generadoras de literales respecto de los utilizados en el plano TSUM. La transformación de niveles necesaria es fácilmente realizable fuera de los planos de la PLA.

En la figura 6.15 se muestra la respuesta transitoria obtenida con SPICE de la PLA del tipo 2 (ver figuras 4.12 y 6.11). La secuencia de entradas aplicada coincide con la indicada anteriormente para la realización del tipo 1. Sin embargo, hemos considerado que en dichas entradas y en el plano TSUM se utilizan niveles de corriente distintos: I_0=50μA en la entrada de la celda generadora de literales que decodifica la variable X_2, e I_0=20μA en el plano TSUM.

La figura 6.16 muestra ampliada la transición de la salida f_1 entre los niveles lógicos 0 y 1 (transición de 0 a 20μA en ID(MPS1)). En la parte superior de la figura aparecen ampliadas

las transiciones de tensión en la línea de literal X_2^1 (V(22)), en la línea de salida f_1 (V(30)), en la línea producto P_2

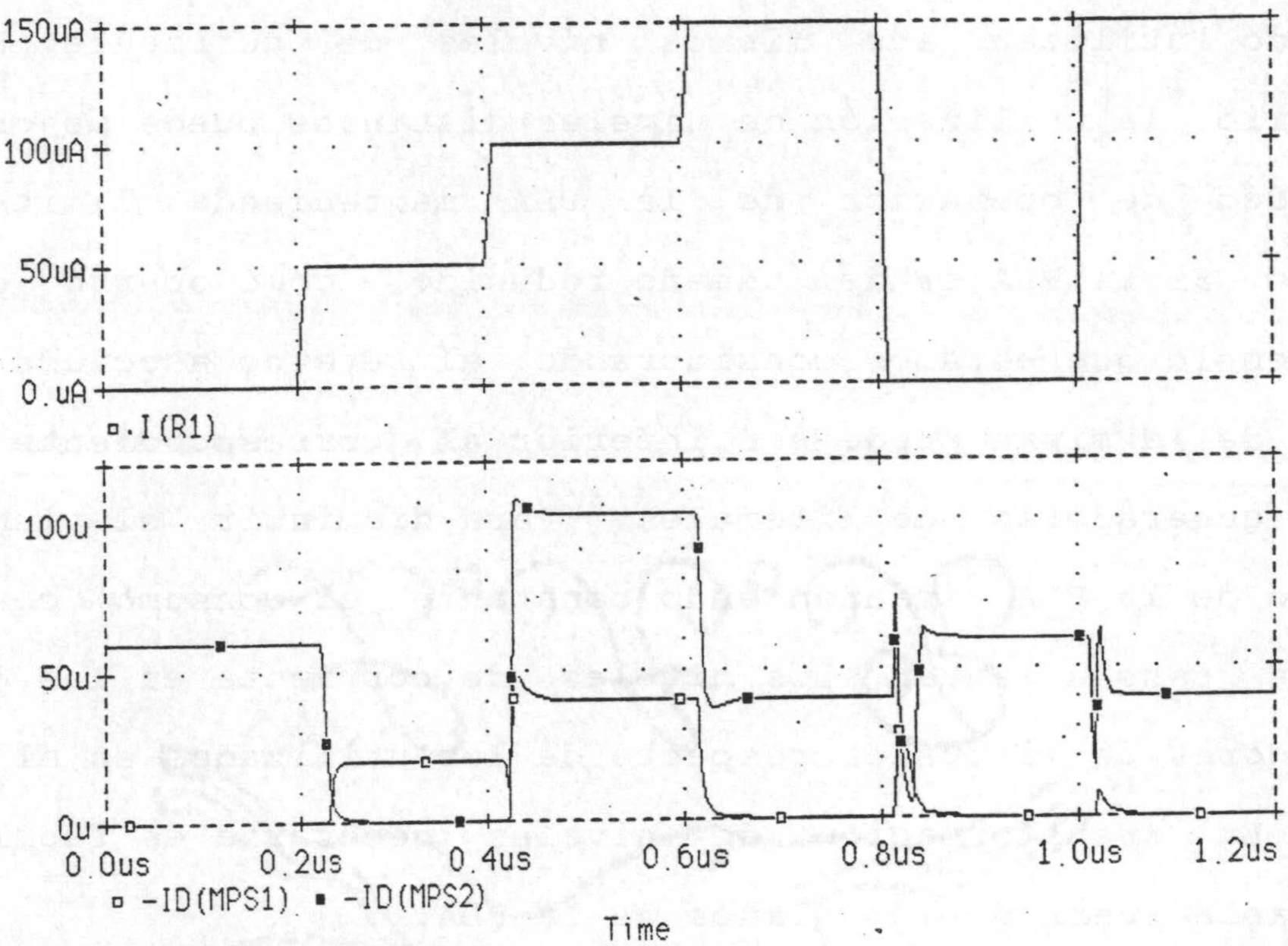


FIGURA 6.15. Respuesta transitoria de la PLA del tipo 2.

(V(12)), y en la puerta del espejo de corriente NMOS controlado por P_2 (V(312)). El tiempo de retardo de la salida de corriente, respecto de la transición de tensión de la correspondiente línea de literal, es del orden de 7ns (casi la mitad del obtenido para la realización del tipo 1 operando con el mismo valor de corriente unidad). El tiempo de retardo total, incluyendo el debido a la celda generadora de literales, es algo inferior a 30ns.

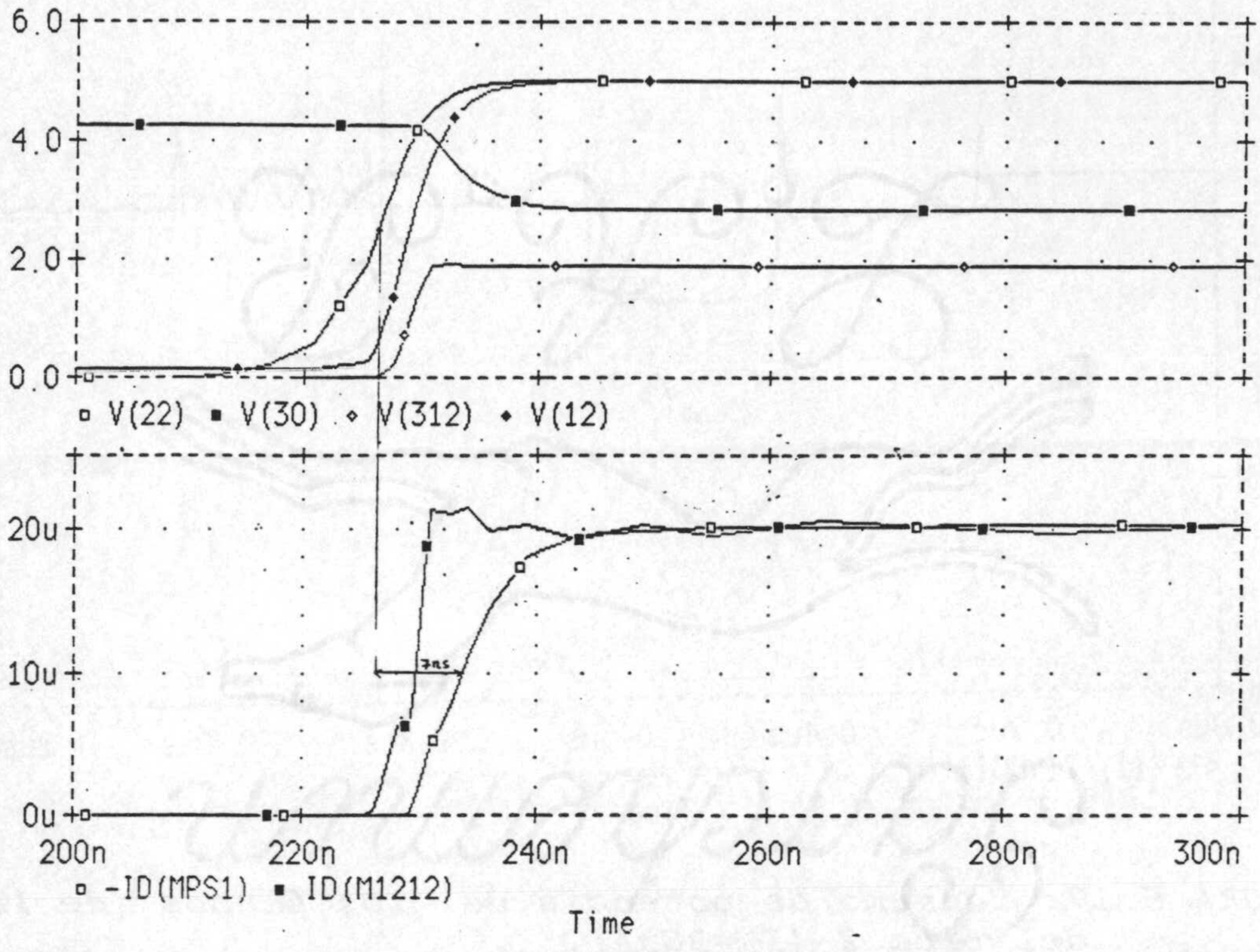


FIGURA 6.16: Transiciones de máximo tiempo de retardo de la PLA del tipo 2.

El consumo de potencia, cuya dependencia con el tiempo se muestra en la figura 6.17, es superior al de los planos de la PLA del tipo 1 (funcionando con el mismo valor de corriente unidad). La causa es, según indicamos en el capítulo 4, el consumo dominante de los transistores PMOS añadidos al plano NOR (ver figura 4.12).

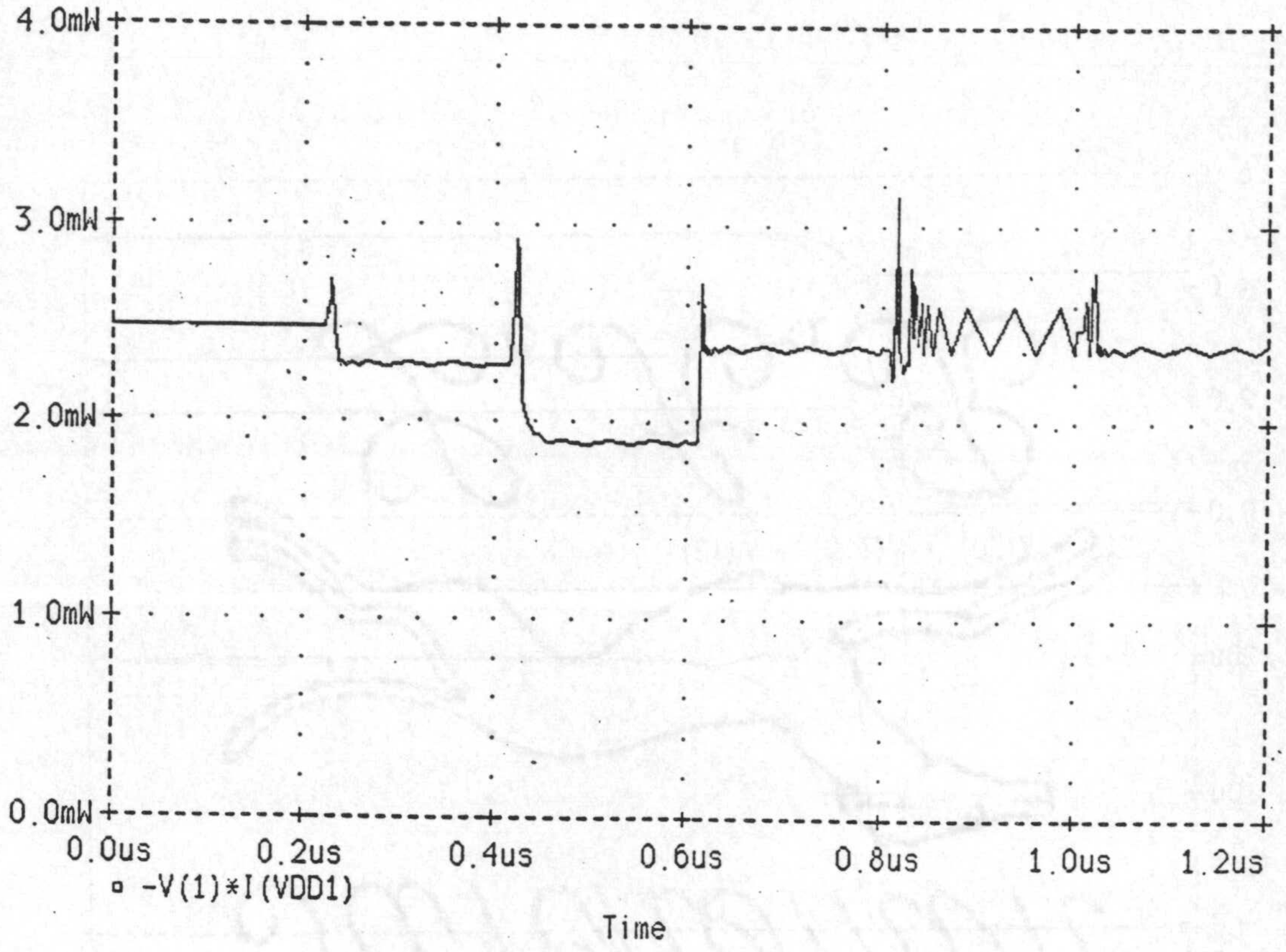


FIGURA 6.17: Consumo de potencia de los planos de la PLA del tipo 2 (I_0 =20µA).

PLANOS DE LA PLA DEL TIPO 5:

Como indicamos en la sección 4.3, una realización del tipo 5 (ver figuras 4.15 y 4.16) utiliza una señal de reloj para temporizar su funcionamiento. Para optimizar la velocidad de funcionamiento interesa que los cambios de las entradas estén sincronizados con el inicio de la fase de precarga del plano NOR, de tal manera que durante esta fase se lleve a cabo la decodificación de las entradas en las celdas generadoras de literales. Las salidas de dichas celdas se mantienen a 0 debido a la actuación de la misma señal de reloj, pero mientras tanto pueden actuar los detectores de umbrales de corriente (elemen-

tos más lentos de la celda generadora de literales).

El tiempo en alta de la señal de reloj debe ser suficiente para efectuar la precarga correcta de las líneas producto del plano NOR, y superior al tiempo de respuesta de los referidos detectores de umbral (ver figuras 6.3 y 6.4).

El tiempo en baja debe ser mayor que el tiempo de evaluación del plano NOR. La evaluación del plano TSUM puede proseguir durante la fase de precarga del plano NOR; por lo tanto, el periodo de la señal de reloj debe ser superior al tiempo de respuesta de dicho plano.

En la figura 6.18 se muestra la respuesta transitoria de los planos de la PLA del tipo 5 a la misma secuencia de transiciones de la corriente de entrada considerada para las PLAs anteriores. I(R1) es la corriente de entrada correspondiente a la variable X_2, ID(MPS1) e ID(MPS2) son las corrientes en las salidas f_1 y f_2, y V(100) es la señal de reloj de temporización (ϕ_1). La transición de salida con mayor tiempo de retardo corresponde al cambio de la salida f_2 entre los niveles lógicos 3 y 0. Esta transición aparece ampliada en la figura 6.19, presentando un tiempo de retardo de 10ns respecto de la transición de ϕ_1, que indica el inicio de la evaluación de los planos. En la parte superior de la figura se muestran además las transiciones que tienen lugar en la línea de literal y la línea producto implicadas en el cambio de la corriente de salida; concretamente, las líneas $X_2{}^1$ y P_7 (tensiones V(22) y

V(17)) del esquema de la figura 6.11. La variable V(317) es la tensión de puerta del espejo de corriente NMOS conmutado por la línea P_7.

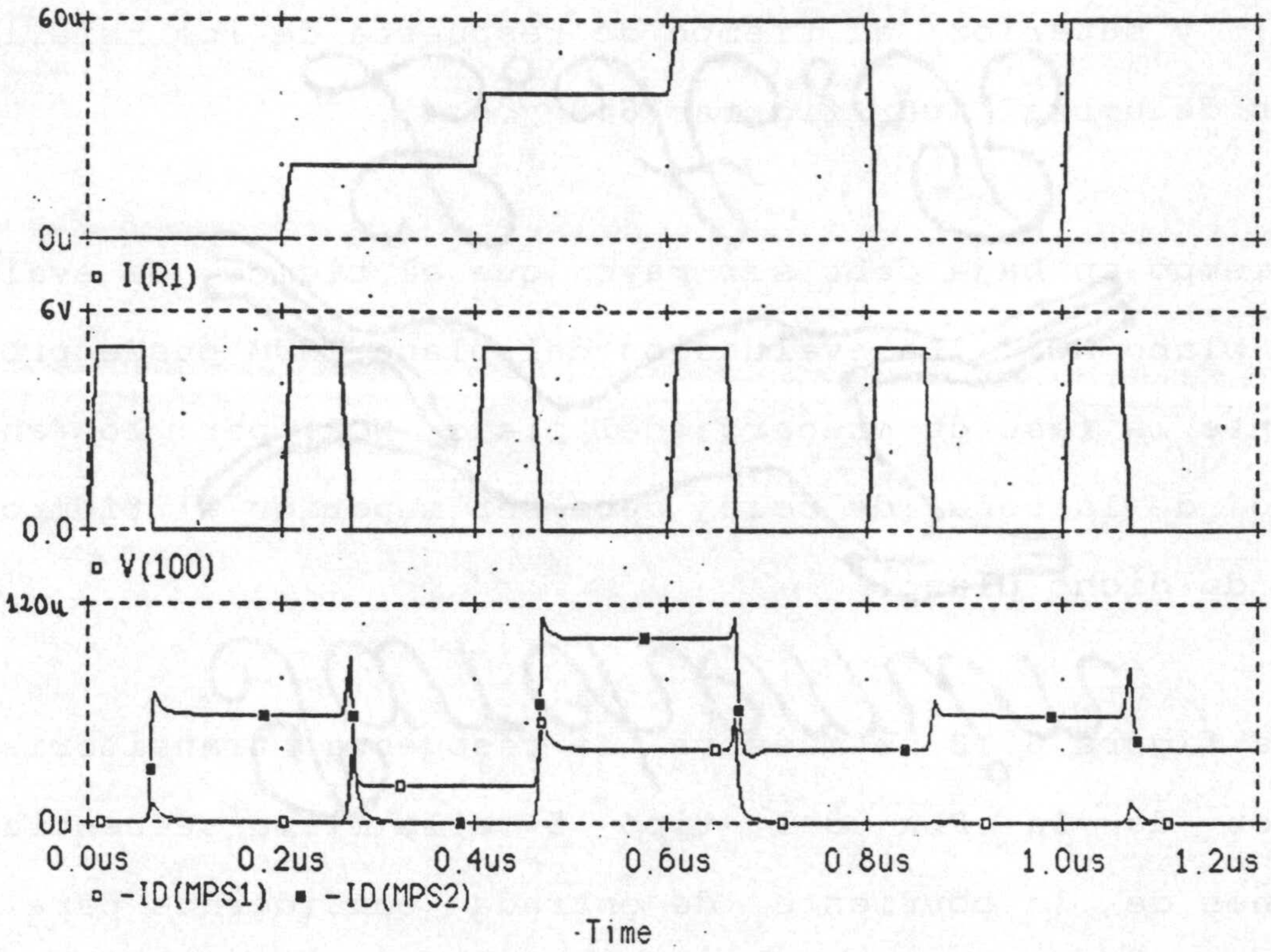


FIGURA 6.18: Respuesta transitoria de la PLA del tipo 5.

El consumo de potencia obtenido de la simulación de los planos de la PLA del tipo 5, cuya dependencia con el tiempo se muestra en la figura 6.20 es, en media, bastante menor del producido por la realización del tipo 2 (ver figura 6.17), presentando los picos acentuados propios de las transiciones de los inversores CMOS. No existe una diferencia tan acusada respecto del consumo producido por la realización del tipo 1 (ver figura 6.14) debido a que se trata de una PLA con sólo 7 términos producto.

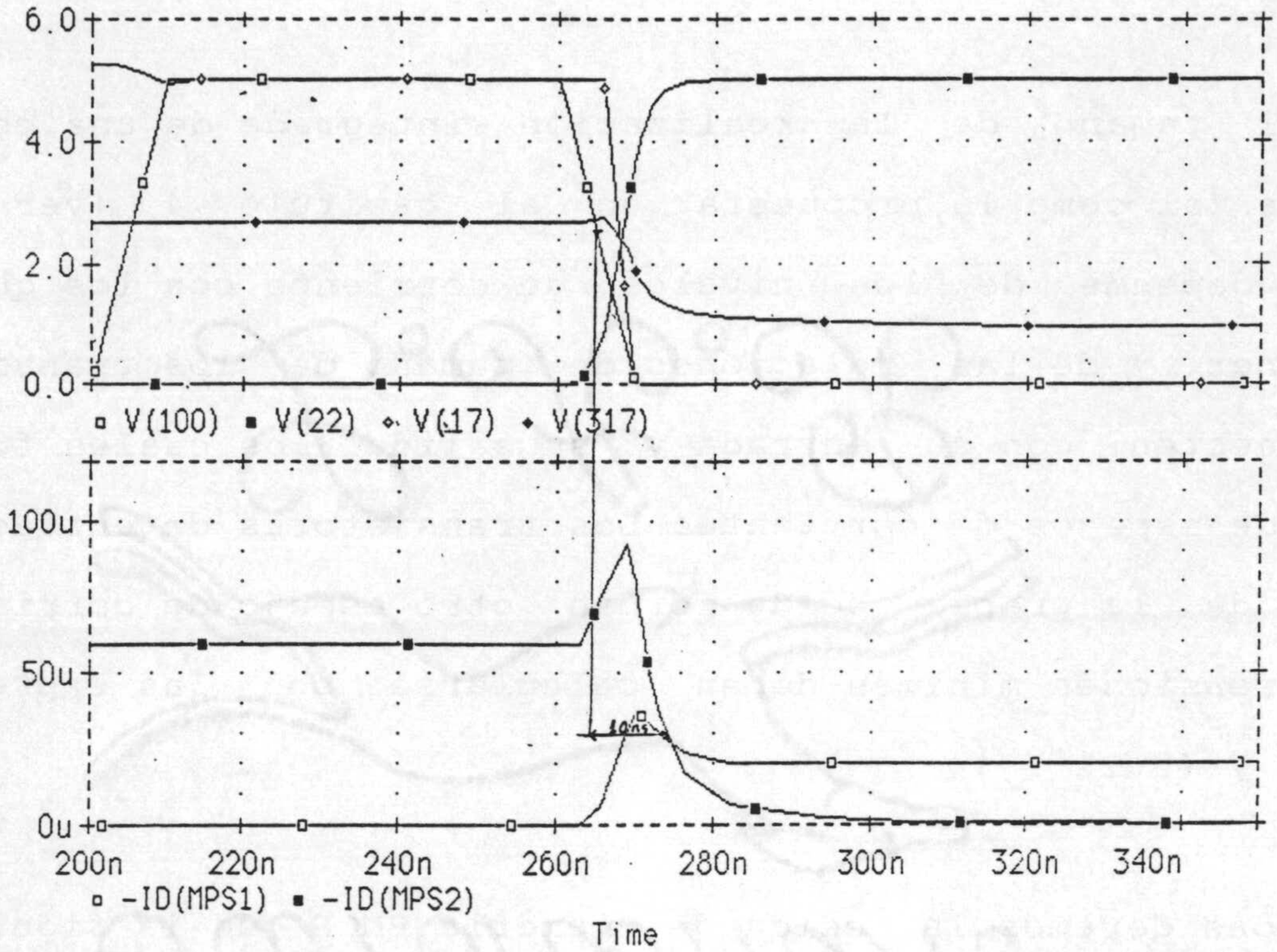


FIGURA 6.19: Transiciones de máximo tiempo de retardo de la PLA del tipo 5.

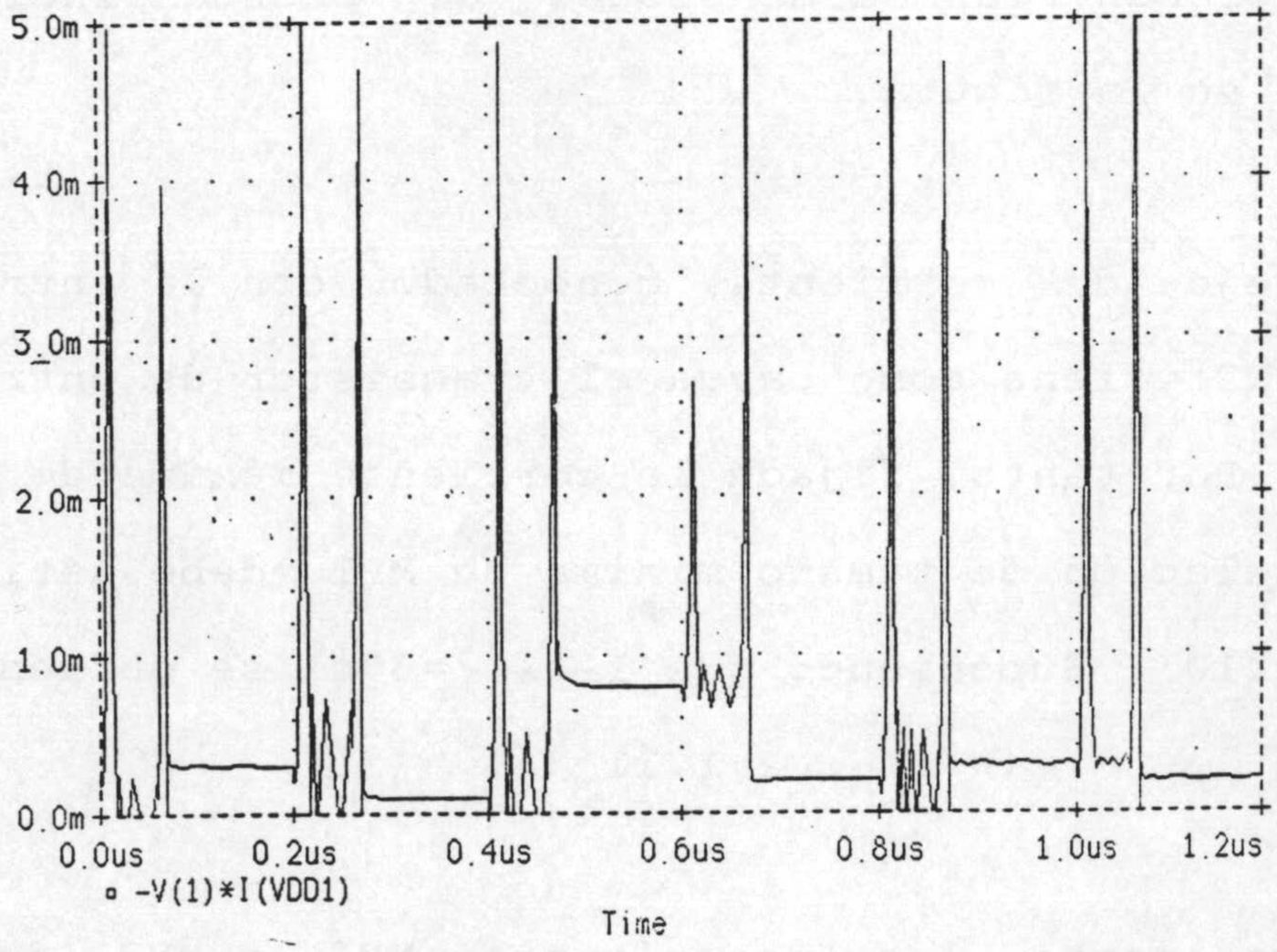


FIGURA 6.21: Consumo de potencia de los planos de la PLA del tipo 5 (I_0 =20μA).

6.4 Celdas de memoria en modo de corriente.

El tamaño de la realización integrada de una celda de memoria tal como la propuesta en el capítulo 4 (ver figura 4.18) depende de los niveles de corriente con los que deba funcionar, y de las relaciones de tamaño de los transistores que conecten con su entrada y su salida, los cuales formarán parte de espejos de corriente. Los transistores de entrada y de salida de la propia celda forman otro espejo de corriente, y sus dimensiones mínimas deben calcularse con las expresiones (2.10) y (2.11).

Consideremos la celda de memoria PMOS de la figura 6.21 (circuito enmarcado por líneas discontínuas), cuya entrada y salida conectan con transistores de espejos de corriente NMOS, los cuales tienen las dimensiones de puerta indicadas (en micrómetros) en la figura.

El espejo de corriente conectado con la entrada de la celda (MN1-MN2) tiene como carga el transistor de entrada de la misma (MP1). Por tanto, fijada la corriente máxima de funcionamiento, la relación de tamaño mínima de MP1 debe satisfacer la expresión (2.10). Suponiendo que $I_{o,max}=60\mu A$ se obtiene:

$$(W/L)_{MP1} \geq 1.11$$

Por otra parte, los transistores MP1 y MP2 de la celda constituyen un espejo de corriente PMOS, que tiene como carga el transistor MN4. Por tanto, la relación de tamaño mínima de

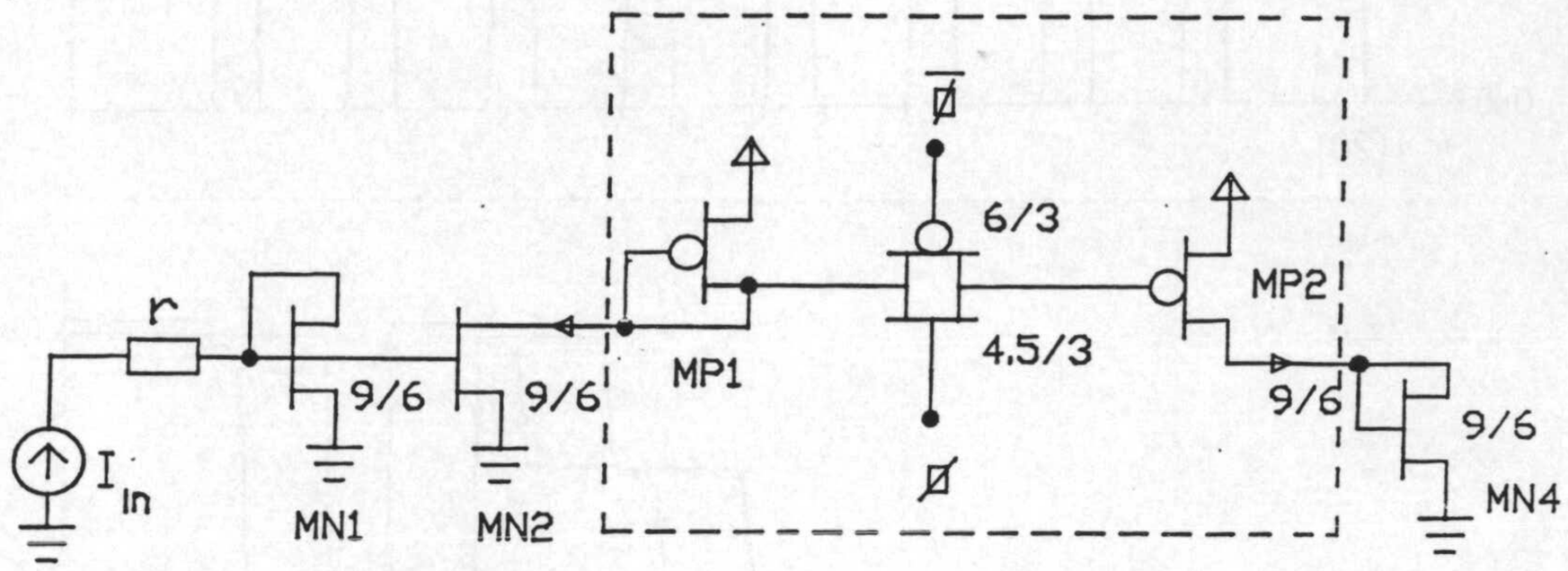


FIGURA 6.21: Circuito utilizado en la simulación de la celda de memoria PMOS.

MP2 se determina mediante la expresión (2.11), obteniéndose:

$$(W/L)_{MP2} \geq 1.25$$

Utilizando en la simulación las dimensiones de puerta:

$$(W/L)_{MP1} = (W/L)_{MP2} = 7.5\mu m/6\mu m = 1.25$$

se obtienen los resultados de la figura 6.22, los cuales prueban el correcto funcionamiento de la celda en el rango de corrientes fijado. Tan sólo se obtiene un deterioro apreciable de la corriente de salida cuando la corriente de entrada coincide con la máxima impuesta en el diseño. La simulación de una celda de memoria NMOS produce resultados similares a los obtenidos para esta celda.

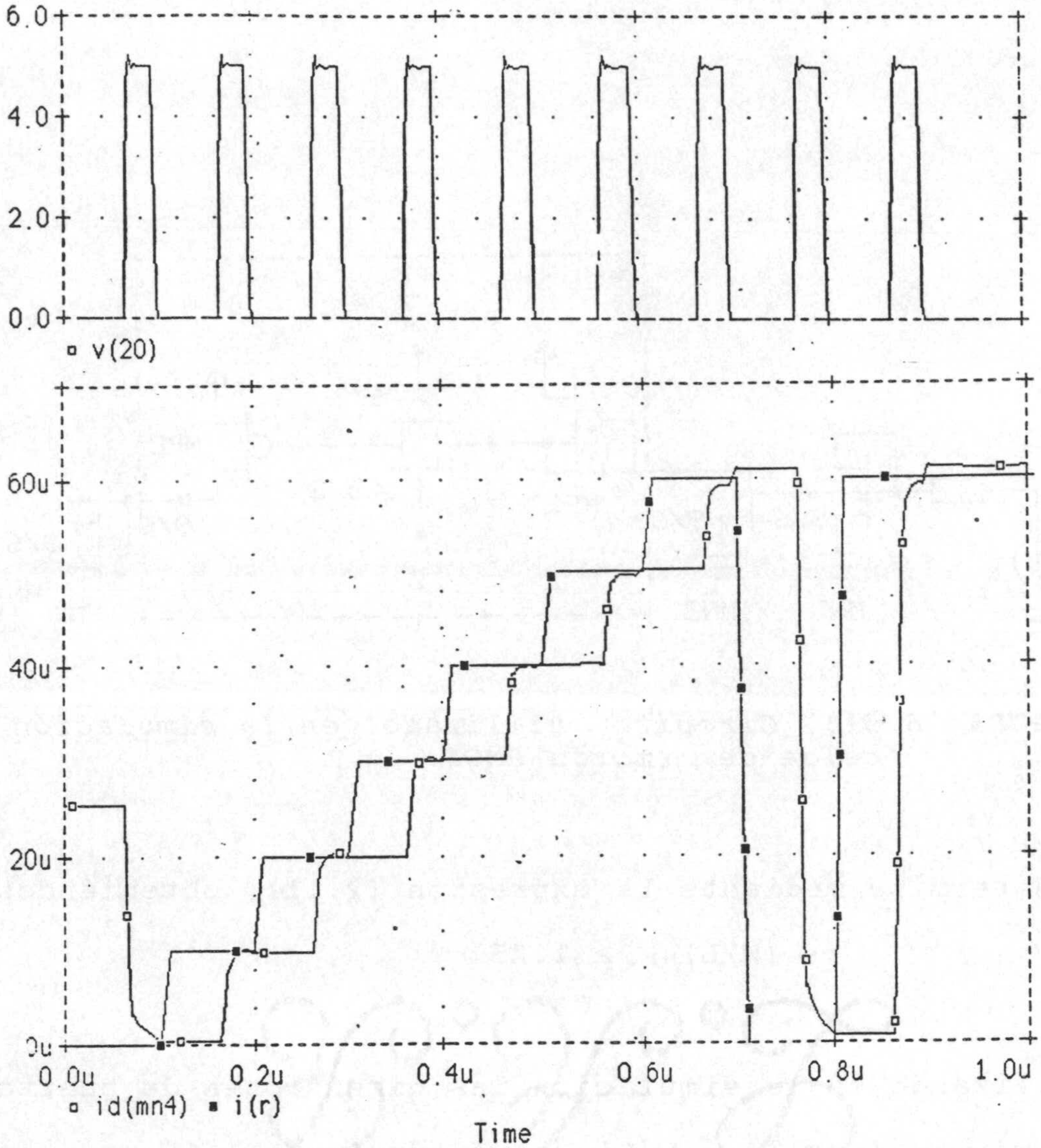


FIGURA 6.23: Respuesta transitoria del circuito de la figura 6.22. (a) Señal de reloj ø. (b) Corrientes de salida y de entrada.

6.5 Prototipo de PLA integrada fácilmente comprobable.

En esta sección se muestra el diseño de un prototipo de PLA tetravaluada integrada, a la cual se ha añadido circuitería para comprobarla mediante el procedimiento de test propuesto en el capítulo 5.

La PLA tiene tres entradas (X_1, X_2, X_3) y tres salidas (Z_1, Z_2, Z_3) tetravaluadas, y 30 líneas producto. En la tabla

6.1 se detallan las funciones sintetizadas por esta PLA, para la que hemos considerado una realización integrada del tipo 1 (ver figura 4.11).

TABLA 6.1: Funciones sintetizadas por la PLA integrada.

Entrada X_3	Salidas Z_2 y Z_1	Salida Z_3
0	$Z_2 Z_1 = X_1 * X_2$ (módulo 4)	$Z_3 = 0$ si $X_1 = X_2$ $Z_3 = 1$ si $X_1 > X_2$ $Z_3 = 2$ si $X_1 < X_2$
1	$Z_2 Z_1 = X_1 + X_2$ (módulo 4)	
2	Z_2 = mín$\{X_1, X_2\}$; Z_1 = máx$\{X_1, X_2\}$	
3	$Z_1 = 4 - X_1$; $Z_2 = 4 - X_2$	

Para facilitar el test de la PLA se le han añadido (ver figura 5.2) dos líneas producto, una línea de salida, una celda generadora de literales de tipo NOR modificada (LITEST), y un registro de desplazamiento de 32 bits (circuito SLP de la figura 5.2). Además, las celdas generadoras de literales utilizadas en las entradas son todas del tipo NOR como exige el procedimiento de test.

En la figura 6.23 se muestra el trazado de la celda integrada LITEST. Las líneas $X_{C\,i}$, i=0,1,2,3, son las salidas binarias que controlan las celdas LITE (ver figura 6.1). Mediante la entrada binaria X_T se selecciona el modo de funcionamiento de LITEST (ver apartado 5.3.2).

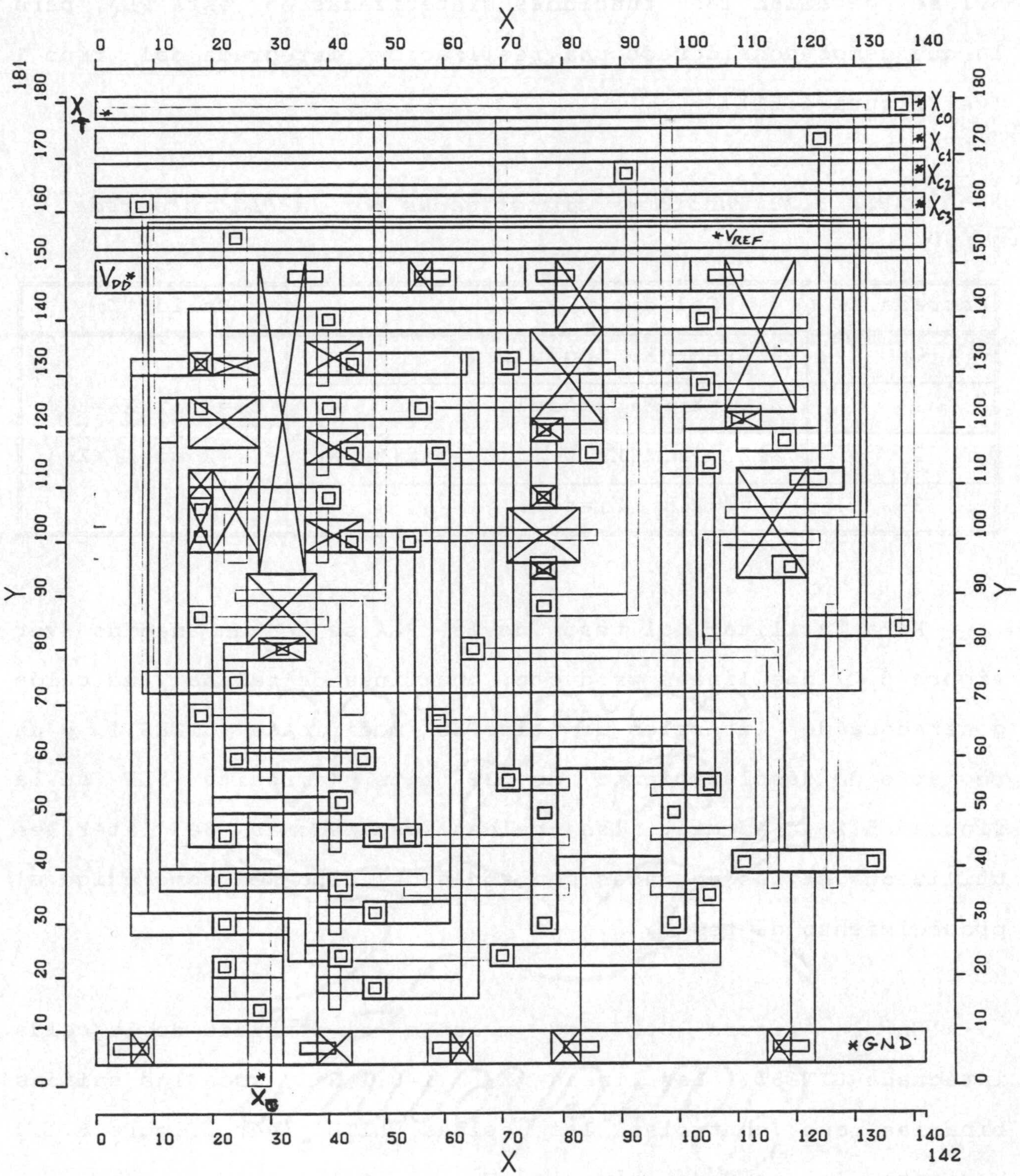
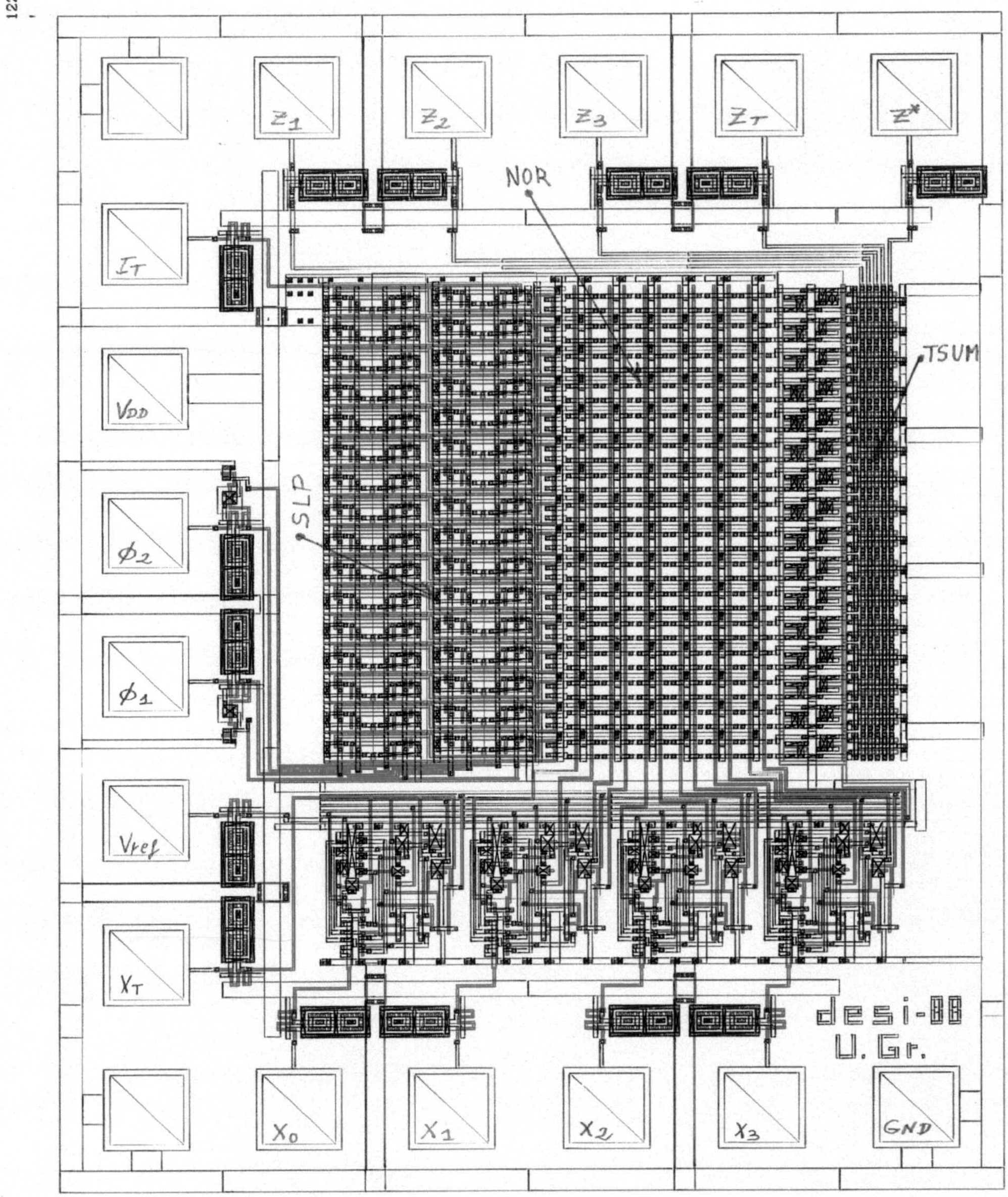


FIGURA 6.23: Descripción gráfica de las máscaras de la celda LITEST.

El registro de desplazamiento diseñado utiliza celdas dinámicas de dos fases (figura 5.40.a de la referencia [193]), las cuales se han ubicado por parejas para permitir su conexión con las líneas producto de la PLA.

La figura 6.24 muestra el trazado de la PLA integrada completa, incluidos los "pads" de entrada y de salida. El terminal I_T es la entrada de datos del registro de desplazamiento, ϕ_1 y ϕ_2 son las entradas de sincronización de dicho registro, y Z_T es el terminal correspondiente a la línea de salida añadida al plano TSUM (ver figura 5.2). En este plano se ha incluido otra línea (conectada con el terminal Z* de la figura 6.24) que presenta conexiones con todas las líneas producto, mediante la cual se hará un test eléctrico de las fuentes de corriente PMOS. Activando líneas producto aisladas con el SLP, y manteniendo a 0 todas las líneas de literal, la salida Z* permite medir las corrientes suministradas por dichas fuentes. Mediante la aplicación de patrones de test específicos, y medidas de las corrientes en esta salida, es posible identificar los defectos físicos no detectables mediante el procedimiento de test propuesto (ver apartado 5.3.1).

Como se observa en la figura 6.24, el registro de desplazamiento ocupa un porcentaje elevado de la superficie total del circuito integrado. Eligiendo un diseño alternativo del SLP, que utilice la mitad de celdas de registro y demultiplexores la superficie ocupada por este circuito puede reducirse considerablemente.



920

NOM : chLp1 No.Un./Cm.: 45

NIVELLS: dn____ dp☒ co____ po____ mi____ cn____ pa☐ ____ ____

Lucie VAX/VMS Versio Octubre-1982

Figura 6.24: Descripción gráfica del prototipo de PLA integrada

6.6 Conclusión.

En este capítulo hemos presentado descripciones a nivel de circuito integrado y resultados de simulación eléctrica de los circuitos propuestos en el capítulo 4, así como el diseño de una PLA integrada fácilmente comprobable mediante el procedimiento de test indicado en el capítulo 5.

Los parámetros eléctricos y las reglas geométricas de diseño utilizados corresponden al proceso CMOS de la U.C.L. (Bélgica). Los resultados de simulación obtenidos concuerdan con los planteamientos realizados sobre el funcionamiento de estos circuitos, y demuestran la viabilidad de integración de los mismos utilizando un proceso CMOS ordinario.

El prototipo de PLA integrada se ha incluido en el MPC-88 para su fabricación, junto con el multiplicador en modo de corriente presentado en el capítulo 2.

CAPITULO 7

CONSIDERACIONES FINALES Y CONCLUSIONES

El trabajo presentado en esta memoria ha consistido básicamente en la caracterización del funcionamiento de los circuitos básicos CMOS en modo de corriente, y en el diseño de estructuras de circuito adecuadas para la síntesis de sistemas multiumbrales y multivaluados integrados CMOS en modo de corriente.

La investigación realizada es susceptible de continuidad. Así pues, en trabajos futuros tenemos previsto acometer las siguientes tareas:

1.- Construcción de un simulador lógico-temporal para circuitos CMOS en modo de corriente basado en las expresiones simplificadas obtenidas en el capítulo 2. Este

simulador constituirá una herramienta adecuada para el diseño de circuitos integrados en modo de corriente con gran número de componentes.

2.- Realización física de redes neuronales CMOS en modo de corriente que utilicen como componentes básicos puertas umbrales en modo de corriente y celdas de memoria multivaluadas (ver capítulos 2 y 4). En la comunicación [121] hemos presentado una posible estructura de red neuronal de estas características y nuestro objetivo actual es la integración de un prototipo de esta red en un próximo MPC.

3.- En lo que respecta al diseño de sistemas multivaluados CMOS mediante PLAs en modo de corriente, son posibles las siguientes mejoras:

3.1.- Obtención de un algoritmo de simplificación multifuncional para realizaciones mínimo/suma-truncada que sea una extensión del propuesto en el capítulo 4.

3.2.- Elaboración de un paquete de programas para el diseño automático de máquinas secuenciales multivaluadas mediante PLAs que, partiendo de la especificación de la máquina, dé como resultado la descripción óptima del circuito integrado que la sintetice.

3.3.- Estudio de esquemas alternativos de test de PLAs

multivaluadas. En particular, la obtención de LFSRs multivaluados con realizaciones físicas sencillas facilitaría el test de PLAs de tamaño reducido con menos exceso de superficie que el producido por el esquema de test propuesto (ver capítulo 5).

Como punto final de esta memoria conviene comentar la contribución del trabajo de investigación realizado. Siguiendo las directrices marcadas en la sección 1.2, las principales aportaciones presentadas en esta Memoria son las siguientes:

1.- Se ha caracterizado el funcionamiento estático y dinámico de las configuraciones básicas utilizadas en la construcción de circuitos multiumbrales y multivaluados en modo de corriente. Esta caracterización ha servidou para demostrar la viabilidad de estos circuitos utilizando un proceso de fabricación CMOS ordinario: siempre es posible, asignando los niveles de corriente y las dimensiones de los dispositivos, reducir los errores de corriente para conseguir márgenes de ruido adecuados. Las expresiones obtenidas, cuya validez hemos comprobado mediante simulación eléctrica con SPICE, son aplicables al diseño y evaluación de prestaciones de circuitos CMOS en modo de corriente.

2.- Se han presentado estructuras de puertas umbrales y multiumbrales CMOS en modo de corriente, las cuales tienen como principales características su sencillez, en términos del número de transistores utilizados para su construcción, y la facilidad para programar sus vectores de pesos y umbrales. La adecuación de este tipo de circuitos para la síntesis de funciones aritméticas binarias se ha puesto de manifiesto con el diseño de un multiplicador paralelo en modo de corriente para datos de 4 bits, cuya realización física supone un ahorro considerable de superficie respecto de una solución CMOS estática convencional construida con las mismas reglas geométricas de diseño.

3.- Se ha propuesto una solución a la síntesis de funciones multivaluadas consistente en una estructura de PLA integrable mediante un proceso CMOS ordinario. La síntesis se basa en la expresión de las funciones como suma truncada de términos NOR, en los cuales sólo intervienen r literales distintos de cada variable. El resto de los literales se utilizan implícitamente en la síntesis, sin que ello implique aumento alguno del tamaño de la PLA. En comparación con las PLAs binarias, considerando conjuntos equivalentes de entradas y de salidas, y el mismo número de términos producto, los cálculos realizados demuestran que el tamaño óptimo se consigue cuando la lógica utilizada por la PLA es la trivaluada o la tetravaluada.

4.- Para la simplificación de las funciones multivaluadas sintetizadas por la PLA se ha implementado un programa en FORTRAN-77, cuyo algoritmo es una optimización del propuesto en 1987 por Dueck y Miller [34].

5.- Se han detallado posibles realizaciones CMOS estáticas y dinámicas de la estructura de PLA propuesta, las cuales proporcionan distintas alternativas de diseño según las necesidades de tamaño, velocidad, y consumo, requeridas. Asimismo, se ha contemplado la posibilidad de sintetizar máquinas secuenciales multivaluadas mediante estas PLAs ampliándolas con celdas de memoria dinámicas en modo de corriente.

6.- Se ha elaborado un procedimiento de test para las PLAs multivaluadas, el cual permite, mediante una secuencia universal de patrones de test, detectar un alto porcentaje de todos los posibles fallos de contacto en los planos de la PLA, así como fallos en los decodificadores de entrada y en la circuitería añadida para el test.

7.- Se ha diseñado un circuito integrado CMOS el cual incluye el multiplicador binario en modo de corriente aludido en el punto 2, así como un prototipo de PLA tetravaluada comprobable mediante el procedimiento de test propuesto.

Con la enumeración de las principales aportaciones concluye esta Memoria, que consideramos contribuye de manera significativa al diseño de circuitos integrados multiumbrales y multivaluados en particualar, y al de sistemas electrónicos digitales en general.

BIBLIOGRAFIA

1. ABD-EL BARR, M.H.; VRANESIC, Z.G.; ZAKY, S.G.: "Design and applications of binary and MVL CCD circuits". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 137-144; 1986.

2. ABD-EL BARR, M.H.; VRANESIC, Z.G.; ZAKY, S.G.: "Synthesis of MVL functions for CCD implementation". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 116-127; 1986.

3. ABRAHAM, J.A.; FUCHS, W.K.: "Fault and error models for VLSI". Proceedings of the IEEE; vol. 74, nº 5, pp. 639-654; May 1986.

4. ALLEN, C.M.; GIVONE, D.D.: "A minimization technique for multiple-valued logic systems". IEEE Transactions on Computers; pp. 182-184; 1968.

5. AOKI, M.; NAKAGOME, Y.; HORIGUCHI, M.; IKENAGA, S.; SHIMOHIGASHI, K.: "A 16-level/Cell dynamic memory". IEEE Journal of Solid-State Circuits; vol. SC-22, nº 2, pp. 297-299; 1987.

6. BALLA, P.C.; ANTONIOU, A.: "Low-power-dissipation MOS ternary-logic family". Proceeding of the 14th International Symposium on Multiple-Valued Logic. IEEE Computer Society;pp. 133-139; 1984.

7. BAUGHT, C.R.; WOOLEY, B.A.: "Statistical analysis of a differential threshold logic circuit configuration". IEEE Trans. on Comp. C-25, no 7, pp.745-754.; 1976.

8. BEACH, A.; ARMSTRONG, J.R.: "Results of a chip layout study for multivalued I^2L". Proceeding of the 12th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 55-68; 1982.

9. BENDER, E.A.; BUTLER, J.T.; KERKHOFF, H.G.: "Comparing the sum with the max for use in four-valued PLA'S".Proceeding of the 15th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 30-35; 1985.

10. BESSLICH, P.W.: "Heuristic minimization of MVL functions: A direct cover approach". IEEE Transactions on Computers; vol. c-35, nº 2, pp. 134-144; 1986.

11. BLEIKER, Ch.; MELCHIOR, H.: "A four-state EEPROM using floating-gate memory cells". IEEE Journal of Solid-State Circuits; vol. sc-22, nº 3, pp. 460-463; 1987.

12. BUTLER, J.T.; KERKHOFF, H.G.: "Multiple-valued CCD circuits". Computer; pp. 58-69; April 1988.

13. CARMONA, J.M.; HUERTAS, J.L.; ACHA, J.I.: "Realisation of three-valued C.M.O.S. cycling gates". Electronics Letters; vol. 14, nº 9, pp. 288-290; 1978.

14. CAVANAGH, J.J.F.: "Digital computer arithmetic. Design and implementation". McGraw-Hill, 1985.

15. CHIN, C.K.; McCLUSKEY, E.J.: "Test length for pseudorandom testing". IEEE Transactions on Computers; vol. c-36, nº 2, pp. 252-256; 1987.

16. CURRENT, K.W.; MOW, D.A.: "Four-valued threshold logic full adder circuit implementations". Pro. 8th Intern. Symp. Multi-valued Logic.; pp. 95-100, IEEE.; 1978.

17. CURRENT, K.W.: "A simultaneous analog-to-quaternary converter". IEEE Transactions on Circuits and Systems; vol. CAS-26, nº 11, pp. 967-970; 1979.

18. CURRENT, K.W.: "Circuits for quaternary-to-analogye conversion". International J. Electronics; vol. 53, nº 2, pp. 149-155; 1982.

19. CURRENT, K.W.: "Pipelined binary parallel counters employing latched quaternary logic full adders". IEEE Trassactions on Computers; vol. c-29, nº 5, pp. 400-403; 1980.

20. CURRENT, K.W.; MOW, D.A.: "Implementing parallel counters with four-valued threshold logic". IEEE Transactions on Computers; vol. c-28, nº 3, pp. 200-204; 1979.

21. CURRENT, K.W.; MOW. D.A.: "Proposed digital-correlator design with four-valued threshold logic". IEE J. Electron. Circ.& Syst.; vol. 2, no. 4, pp. 115-120.; 1978.

22. CURRENT, W.; EDWARDS, F.; FREITAS, D.: "A CMOS multiple valued logic test chip". Proceeding of the 17th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 16-19; 1987.

23. DAEHN, W.; MUCHA, J.: "A hardware approach to self-testing of large programmable logic arrays". IEEE Trans. Comput.; vol. c-30, pp. 829-833; Nov. 1981.

24. DAO, T.T.: "Cellular array complex number multiplier". Proceeding of the 13th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp.255-262; 1983.

25. DAO, T.T.: "Threshold IIL and its applications to binary symetric functions and multivalued logic". IEEE J. Solid-State Circ.; vol. SC-12, no. 5, pp. 463-472. ; 1977.

26. DAO, T.T.; DAVIO, M.; GOSSART, C.: "Complex number arithmetic with odd-valued logic". IEEE Transactions on Computers; vol. c-29, nº 7, pp. 604-610; 1980.

27. DAO, T.T.; McCLUSKEY, E.J.; RUSSELL, L.K.: "Multivalued integrated injection logic". IEEE Transactions on Computers; vol. c-26, nº 12, pp. 1233-1241; 1977.

28. DAO, T.T.; RUSSELL, L.K.; PREEDY, D.R.; McCLUSKEY, E.J.: "Multilevel IIL with threshold gates". ISSCC Dig. Tech. Papel rs.; 1977.

29. DANIELS, R.G.; BRUCE, W.C.: "Built-in self-test trends in Motorola microprocessors". IEEE Design and Test; pp. 64-71; april 1985.

30. DASGUPTA, S.; EICHELBERGER, E.B.; WILLIANS, T.W.: "LSI chip design for testability". Dig. 1978 Int. Solid-State Circuits Conf.; pp. 216-217; Feb. 1978.

31. DONOGHUE, B.; HOLLY, P; ILGENSTEIN, K.: "A 256k HCMOS ROM using four state cell aproach". IEEE Journal of Solid-State Circuits. Vol SC-20, no. 2; april 1985.

32. DRUZETA, A.; VRANESIC, Z.G.; SEDRA, A.S.: "Application of multithreshold elements in the realization of many-valued logic networks". IEEE Trans. Comp. C-23.; pp. 1194-8.; 1974.

33. DUECK, G.W.; MILLER, D.M.: "A 4-valued PLA using the modsum". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 232-240; 1986.

34. DUECK, G.W.; MILLER, D.M.: "A direct cover MVL minimization using the truncated sum". Proceeding of the 17th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 221-226; 1987.

35. EDWARDS, C.R.: "IIL threshold circuits for binary-quaternaty encoding and decoding". Int. J. Electron.; vol. 44, no. 4, pp. 445-448.; 1978.

36. EDWARDS, C.R.: "The application of the Rademacher-Walsh transform to boolean function classification and threshold logic synthesis". IEEE Trans. Comp.; Vol. C-24, no. 1, pp. 48-62.; 1975.

37. EICHELBERGER, M.; LINDBLOOM, E.: "Random pattern coverage and diagnosis for LSSD logic self-test". IBM J. Res. Develop.; vol. 27, pp. 265-272; May 1983.

38. ELECTRONICS REVIEW : "Four-level logic coming next year from signetics"; pp. 31-32; 1976.

39. ETIEMBLE, D.: "Multi-valued integrated circuits for signal transmission".Proceeding of the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 205-208; 1981.

40. ETIEMBLE, D.: "Multivalued I^2L circuits for TSC checkers". IEEE Transactions on Computers; Vol. c-29, nº 6, pp. 537-540; 1980.

41. ETIEMBLE, D.: "Multivalued integrated circuits for microcomputers systems". Proceeding of the 12th International Sympo-

sium on Multiple-Valued Logic. IEEE Computer Society; pp. 135-144; 1982.

42. ETIEMBLE, D.; ISRAEL, M.: "On the realization of multiple-valued flip-flops". Proceeding of the 10th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 16-23; 1980.

43. ETIEMBLE, D.; ISRAEL, M.: "Comparison of binary and multivalued ICs according to VLSI criteria". Computer; pp. 28-42; april 1988.

44. FANTINI, F.; MORANDI, C.: "Failure modes and mechanisms for VLSI ICs - a review". IEEE Proceedings; vol. 132, Pt. G, nº 3, pp. 74-81; 1985.

45. FLEISHER, H.; MAISSEL, L.I.: "An introduction to array logic". IBM J. Res. Dev.; pp. 98-109; March 1975.

46. FREITAS, D.C.; CURRENT, K.W.: "A quaternary logic encoder-decoder circuit design using CMOS". Proceeding of the 13th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 190-195; 1983.

47. FREITAS, D.C.; CURRENT, K.W.: "CMOS current comparator circuit". Electronics Letters; vol.19; no. 7; pp. 696-697; 18 August 1983.

48. FREITAS, D.C.; CURRENT, K.W.: "CMOS circuits for quaternary encoding and decoding". Proceeding of the 14th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 164-168; 1984.

49. FRIEDMAN, N.; SALAMA, C.A.T.; HOLMES, F.E.; THOMPSON, P.M.: "Realization of a multivalued integrated injection logic (MIIL) full adder". IEEE J. of Solid-state circuits.; vol. SC-12.; no. 5, pp. 432-4.; 1977.

50. FUCHS, W.K.; CHEN, Ch.R.; ABRAHAM, J.A.: "Concurrent error detection in highly structured logic arrays". IEEE Journal of Solid-State Circuits; vol. sc-22, nº 4, pp. 583-593.

51. FUJIWARA, H.: "A design of programmable logic arrays with random pattern-testability". IEEE Transactions on Computers Aided Design; vol. 7, nº 1, pp. 5-10; 1988.

52. FUJIWARA, H.; KINOSHITA, K.: "A design of programmable logic arrays with universal tests". IEEE Transactions on Computers; vol. c-30, nº 11, pp. 823-828; 1981.

53. FUKUSHIMA, K.: "A neural network for visual pattern recognition". Computer. pp. 65-75. March 1988.

54. GELSINGER, P.P.: "Built-in self test of the 80386". Int. Conf. Comput.; pp. 169-173; Oct. 1986.

55. GHOSH, S.; CHONDHURY, A.K.: "Partition of Boolean functions for realization with multithreshold threshold logic elements". IEEE Trans. Comp., C-22.; pp. 204-214.; 1973.

56. GOEL. P.: "Test generation costs analysis and projections". 17th Des. Automat. Conf.; 1980.

57. GRAF H.P.; JACKEL L.D.; HUBBARD W.E. "VLSI Implementation of a Neural Network Model" Computer; Vol. 21, Nº 3, pp.41-49; 1988.

58. GROSSBERG S.: Neural Networks and Natural Intelligence MIT Press; 1988.

59. HA, D.S.; REDDY, S.M.: "On the design of random pattern testable PLAs". Proc. 1986 Int. Test Conf.; pp. 688-695; 1986.

60. HACHTEL, G.D.: "An algorithm for optimal PLA folding". IEEE Trans. CAD, CAD-1, nº 2, pp. 63-77; 1982.

61. HAMPEL, D.; WINDER, R.O.: "Threshold logic". IEEE Spectrum.; vol. 8, pp. 32-29.; 1971.

62. HAMPEL, D.: "Multifunction threshold gates". IEEE Trans, Comp.; C-22, pp. 197-203.; 1973.

63. HASSAN, F.J.; ABDUL-KARIM, M.A.H.: "A six trit successive approximation analogue-to-ternary converter design". Int. J. Electronics; vol. 53, nº 3, pp. 253-260; 1982.

64. HASSAN, S.Z.; McCLUSKEY, E.J.: "Testing PLA's using multiple parallel signature analyzers". Proc. 13th Int. Symp. on Fault-Tolerant Computing; pp. 422-425; June 1983.

65. HEUNG, A.: "DECMOS - a low power family of three-valued logic circuits for VLSI implementation". Proceeding of the 14th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 120-124; 1984.

66. HEUNG, A.; MOUFTAH, H.T.: "An all-CMOS ternary identity cell for VLSI implementation". Electronics Letters; vol. 20, nº 5, pp. 221-222; 1984.

67. HIGUCHI, T.; KAMEYAMA, M.: "Static-hazard-free T-gate for ternary memory element and its application to ternary counters". IEEE Transactions on Computers; vol. c-26, nº 12, pp. 1212-1221; 1977.

68. HODGES, D.H.; JACKSON, H. G.: "Analysis and design of digital integrated circuits". McGraw-Hill, 1983.

69. HONG, S.J.; OSTAPKO, D.L.: "FITPLA: A programmable logic array for function-independent testing". Proc. 10th Int. Symp. on Fault-Tolerant Computing; pp. 131-136; Oct. 1980.

70. HU, M.; SMITH, K.C.: "On the use of CMOS ternary gates to realize a self-checking binary logic systems". Proceeding of

the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp.212-215; 1981.

71. HU, M.; SMITH, K.C.: "Ternary scan design for VLSI testability". IEEE Transactions on Computers; vol. c-35; nº 2, pp. 167-170; 1986.

72. HU, M.; SMITH, K.C.; MOUFTAB, H.T.: "Low power 2-of-3 valued CMOS self-checking circuits". Proceeding of the 13th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 64-69; 1983.

73. HUERTAS, J.L.; ACHA, J.I.: "J_K multistable: a generalization of the binary J-K flip-flop".Proceeding of the 7th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 138-142; 1977.

74. HUERTAS, J.L.; ACHA, J.I.; CARMONA, J.M.: "Design and implementation of tristables using c.m.o.s. integrated circuits". Electronics Circuits and Systems; vol. 1, nº 3, pp. 88-94; 1977.

75. HUERTAS, J.L.; ACHA, J.I.; CARMONA, J.M.: "Implementation of some ternary operators with C.M.O.S. integrated circuits". Electronics Letters; vol. 12, nº 15, pp. 385-386; 1976.

76. HUERTAS, J.L.; ACHA, J.I.; SANCHEZ GOMEZ, G.: "Theory and design of multivalued memory elements".Proceeding of the 8th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 213-220; 1978.

77. HUERTAS, J.L.; SANCHEZ GOMEZ, G.: "Low-power CMOS implementation of some operators". Proceeding of the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp.196-199; 1981.

78. HUERTAS, J.L.; SANCHEZ-GOMEZ, G.: "Synthesis of 3-valued unary operators in CMOS integrated technology". International Journal of Electronics; vol. 56, no. 5, pp. 713-726; 1984.

79. HURST, S.L.: "Digital-summation threshold-logic gates: a new circuit element". Proc. IEEE; vol. 120, nº 11, pp. 1301-1307; 1973.

80. HURST, S.L.: "A survey: developments in optoelectronics and its applicability to multiple-valued logic".Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 179-188; 1986.

81. HURST, S.L.: "Logic networks synthesis usign digital-summation threshold-logic gates". Proc. 12th Ann. Allerton. Circ. & Systems Confer. Univ. Illinois.; pp. 525-534.; 1974.

82. HURST, S.L.: "Multiple-valued logic -Its status and its future". IEEE Transactions on Computers; vol. c-33, nº 12, pp. 1160-1178; 1984.

83. HURST, S.L.: "Two decades of multiple-valued logic- an invited tutorial". Proceeding of the 18th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 164-175; 1988.

84. HURST, S.L.; MILLER, D.M.; MUZIO, J.C.: "Spectral techniques in digital logic". New York: Academic. 1985.

85. HUTCHINSON J.; KOCH C.; LUO J.; MEAD C. "Computing Motion Using Analog and Binary Resistive Networks" Computer; Vol.21, Nº 3, pp.52-63; 1988

86. IRVING, T.A.; SHIVA, S.G.; NAGLE, H.T. Jr.: "Flip-flops for multiple-valued logic". IEEE Transactions on Computers; vol. c-25, nº 3, pp. 237-246; 1976.

87. ISHIDA, H.; STEWART, R.M.: "A learning networks using adaptive threshold elements". IEEE Trans. Electr. Comps.; vol. EC-14, pp. 481-485.; 1965.

88. ISHIZUKA, O.: "IIL multi-threshold logic circuits". Trans. Inst. Electron. & Comm. Eng Japan. Sect.E.; vol. E63, no. 2, p. 173.; 1980.

89. ISHIZUKA, O.: "Multi-valued multi-threshold networks". Proceeding of the 6th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 44-47; 1976.

90. ISHIZUKA, O.: "On multivalued multithreshold networks composed of conventional threshold elements". IEEE. Transaction on Computers; C-26, no. 12, pp. 1251-1257.Dec.; 1977.

91. ISHIZUKA, O.: "Synthesis of a pass transistor network aplied to multi-valued logic". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 51-57; 1986.

92. KAMEYAMA, M.; HANYU, T.; HIGUCHI, T.: "Design and implementation of quaternary NMOS integrated circuits for pipelined image processing". IEEE Journal of Solid-State Circuits; vol. sc-22, nº 1, pp. 20-27; 1987.

93. KAMEYAMA, M.; HIGUCHI, T.: "Multiple-valued logic and special-purpose processors: overview and future".Proceeding of the 12th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 289-292; 1982.

94. KAMEYAMA, M.; HIGUCHI, T.: "Signed-digit arithmetic circuits based on multiple-valued logic and its applications". Proceeding of the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 41-47; 1981.

95. KAMEYAMA, M.; KAWAHITO, S.; HIGUCHI, T.: "A multiplier chip with multiple-valued bidirectional current-mode logic circuits". Computer; pp. 43-56; April 1988.

96. KAMEYAMA, M.; SEKIBE, T.; HIGUCHI, T.: "Design of highly parallel residue arithmetic circuits based on multiple-valued

bidirectional current-mode MOS technology". Proceeding of the 18th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 6-13; 1988.

97. KARPOVSKY, M.G.: "Spectral techniques and fault detection". Academic Press; 1985.

98. KAWAHITO, S.; KAMEYAMA, M.; HIGUCHI, T.: "VLSI-oriented bi-directional current-mode arithmetic circuits based on the radix-4 signed-digit number system".Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 70-77; 1986.

99. KAWAHITO, S.; KAMEYAMA, M.; HIGUCHI, T.; YAMADA, H.: "A high-speed compact multiplier based on multiple-valued bi-directional current-mode circuits". Proceeding of the 17th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 172-180; 1987.

100. KENT, S.A.: "A high-speed threshold gate multiplier". IEEE Trans. Comp., C-26, no. 12, pp. 1279-1283; 1977.

101. KERKHOFF, H.G.: TERVOERT, M.L.: "Multiple-valued logic charge-coupled devices". IEEE Transactions on Computers; vol. c-30, nº 9, pp. 644-652; september 1981.

102. KERKHOFF, H.G.; TERVOERT, M.L.; TILMANS, H.A.C.: "Design considerations and measurement results of multiple-valued logic CCD's". Proceeding of the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 205-211; 1981.

103. KERKHOFF, H.G.; TERVOERT, M.L.; STEMERDINK, J.A.: "The design and application of a CCD four-valued adder circuit". Proceeding of the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 96-99; 1981.

104. KERKHOFF, H.G.; BUTLER, J.T.: "Design of a high-radix programmable logic array using profiled peristaltic charge-coupled devices". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp.128-136; 1986.

105. KHAKBAZ, J.: "A testable PLA design with low overhead and high fault coverage". IEEE Transactions on Computers; vol. c-33, nº 8, pp. 743-750; 1984.

106. KHARBAZ, J.; McCLUSKEY, E.J.: "Concurrent error detection and testing for large PLA's". IEEE Journal of Solid-State Circuits; vol. sc-17, nº 2, pp. 386-394; 1982.

107. KOANANTAKOOLL, H.T.: "Implementation of ternary identity cells using C.M.O.S. integrated circuits". Electronics Letters; vol. 14, nº 15, pp. 462-464; 1978.

108. KOHONEN T. "The 'Neural' Phonetic Typewriter" Computer; Vol.21, Nº 3, pp.11-22; 1988.

109. KUO, H.; FANG, K.: "The multiple-valued programmable logic array and its application in modular design". Proceeding of the 15th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 10-18; 1985.

110. LAKHANI, A.A.; POTTER, R.C.: "Combining resonant tunnelling diodes for signal processing and multilevel logic". Appl. Phys. Letters; Vol. 52, nº 20, pp. 1684-1685; 1988.

111. LANGHELD, E.: "Threshold logic in the nervous system". Elektronik.; vol. 25, no. 3, pp. 99-104.; 1976.

112. LAUNNE, R.: "Higth-speed current-mode threshold gate with clamps and current comparison". Electr. Lett; vol. 13, no. 4, pp. 99-100.; 1977.

113. LAW, H.F.S.; SHOJI, M.: "PLA design for the BELLMAC-32A microprocessor". Proc. Int'l Conf. Circ. and Comp.; pp. 161-164; 1982.

114. LEWIS II P. M.; COATES C. L. "Threshold Logic" John Wiley and Sons, New York; 1967

115. LLORIS A.: "Una realización multiumbral para funciones booleanas de cuatro variables". Tesis Doctoral. Universidad de Madrid.; 1972

116. LLORIS A.; COBO A.; PRIETO A.: "Implementation of the unary operators in ternary logic: a universal CMOS circuit". Int. J. of Electronic.; Vol.52, No.4, pp.307-311; 1982

117. LLORIS A.; PRIETO A.: "Desarrollo y aplicaciones de la lógica multiumbral". Revista de Informática y Automática.; Vol.XV, No.51, pp.53-74; 1982

118. LLORIS A.; PRIETO A.; IGLESIAS J.M.: "Computer aided synthesis procedure for the unary functions of two variables". First International Fuzzy Systems Association (IFSA) Congress.; Vol.II, pp. S-7.5.1, S-7.5.12; 1985

119. LLORIS A.; PRIETO A.; VELASCO J.: "CMOS circuit for implementation of unary operators in ternary logic". Electronics Letters.; Vol.16, No.5, pp.161-2; 1980

120. LLORIS A.; VELASCO J.: Classification under permutations of the ternary function of two variables." IEEE proc.; Vol.128, No.4, pp.143-148; 1981

121. MARTIN-SMITH,P.; PRIETO,A.;PELAYO,F.J;ORTEGA,J.;LLORIS,A.: "CMOS Design of a Neural Network Model". EUROCAST'89 International Workshop on Computer Aided Systems Theory. Canary Island; pp.119-124; March 1989.

122. McCLUSKEY, E.J.: "Logic design of multivalued I^2L logic circuits". IEEE Transactions on Computers; vol. c-28, nº 8, pp. 546-559; 1979.

123. McCLUSKEY, E.J.: "Logic design principles: with emphasis on testable semicustom circuits". Prentice-Hall; 1986.

124. McCLUSKEY, E.J.: "Minimization of boolean functions". Bell Syst. Tech. J.; vol. 35, nº 5, pp. 1417-1144; 1956.

125. McCLUSKEY, E.J.; BOZORGUI-NESBAT, S.: "Design for autonomous test". IEEE Transactions on Computers; vol. c-30, nº 11, pp. 866-875; 1981.

126. McCLUSKEY, E.J.: "Verification testing-a pseudoexhaustive test technique". IEEE Trans. Comput.; vol. c-33, nº 6, pp. 541-546; June 1984.

127. MEAD, C.; CONWAY, L.: "Introduction to VLSI systems". Addison-Wesley; 1980.

128. MILLER, D.M.; MUZIO, J.C.: "On the minimization of many-valued functions". Proceeding of the 9th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 294-299; 1979.

129. MIN, Y.: "A PLA design for ease of test generation". Proc. 14th Fault Tolerant Computing Symp.; pp. 436-442; 1984.

130. MOTOROLA: "CMOS/NMOS special functions. Vol. 2"; Motorola Inc.; 1984.

131. MOUFTAH, H.T.; GARBA, A.I.: "VLSI implementation of a 5-trit full adder". IEEE Proceedings; vol. 131, pt. G, nº 5, pp. 214-220; 1984.

132. MOUFTAH, H.T.; HEUNG, A.N.C.; WONG, L.M.C.: "QTC-1: a CMOS ternary computer".Proceeding of the 14th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 125-132; 1984.

133. MOUFTAH, H.T.; SMITH, K.C.: "Injected voltage low-power CMOS for 3-valued logic". IEE. Proc.; vol. 129, pt. G, nº 6, pp. 270-272; 1982.

134. MOUFTAH, H.T.; SMITH, K.C.: "Three-valued C.M.O.S. cicling gates". Electronics Letters; vol. 14, nº 2, pp. 36-37; 1978.

135. MOUFTAH, H.T.; SMITH, K.C.; VRANESIC, Z.G.: "Ternary rate-multipliers". IEEE Transactions on Computers; vol. c-29, nº 10, pp. 929-931; 1980.

136. MUROGA, S.: "Threshold logic and its applications". Wiley.; 1971.

137. MURRAY, A.F.; SMITH, A.V.W.: "Asynchronous VLSI neural networks using pulse-stream arithmetic". IEEE Jour. of Solid-State Circuits; vol. 23, no. 3; pp. 688-697. June 1988.

138. MUTA, S.: "Micropower CMOS implementation of three-valued logic functions".Proceeding of the 13th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 61-63; 1983.

139. MUZIO, J.C.; WESSELKAMPER, T.C.: "Multiple-valued switching theory". Adam Hilger Ltd.; 1986.

140. ONNEWEER, S.; KERKHOFF, H.; BUTLER, J.: "Structural computer-aided design of current-mode CMOS logic circuits".Proceeding of the 18th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 21-30; 1988.
141. ONNEWEER, S.P.; KERKHOFF, H.G.: "Current-mode CMOS high-radix circuits". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 60-69; 1986.

142. ONNEWEER, S.P.; KERKHOFF, H.G.: "High-radix current-mode CMOS circuits based on the truncated difference operator". Proceeding of the 17th International Symposium on Multiple-valued logic. IEEE Computer Society; pp. 188-195; 1987.

143. OSTAPKO, D.L.; HONG, S.J.: "Fault analysis and test generation for programmable logic arrays (PLA's)". IEEE Transactions on Computers; vol. c-28, nº 9, pp.617-626; 1979.

144. PELAYO F.; GARCIA C.; PRIETO A.; LLORIS A.: "Some improvements in the implementation of multithreshold and multivalued I^2L circuits". Int. J. of Electronics.; Vol. 66, Nº1. pp.19-34. 1989

145. PICTON, P.D.: "Realization of multithreshold threshold logic networks using the Rademacher-Wals transform". IEE Proc.; vol. 128, part. E, no. 3, pp. 107-113.; 1981.

146. PICTON, P.D.: "Realisation of multithreshold threshold-logic gates using charge-coupled devices". IEE Proc.; vol. 129, pte. E, nº 3, pp. 107-110; 1982.

147. POMPER, G.: "An efficient multi-valued minimization algorithm". Proceeding of the 9th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 300-304; 1979.

148. POMPER, G.; ARMSTRONG, J.R.: "Representation of multivalued functions using the direct cover method". IEEE Transactions on Computers; vol. c-30, nº 9, pp. 674-679; 1981.

149. PRIETO, A.; COBO, A.; LLORIS, A.: "Circuitos en tecnología CMOS para la síntesis de las funciones ternarias". 5º Congreso de Informática y Automática (AEIA). Vol. 2, pp. 631-636; 1982.

150. PRIETO A.; MARTIN-SMITH P.; PELAYO F.; LLORIS A.: "The design of decoders for q-valued logic circuits". Proc. of the 18th International Symposium on Multiple-Valued Logic. Baleares. May 24-26. Computer Society of IEEE. pp.32-39; 1988

151. PRIETO A.; PELAYO F.; LLORIS A.: "Multithreshold logic circuits implemented with operational amplifiers". Int. J. Electronics.; Vol.58, No.3, pp.395-406; 1985

152. PUNTONET C.G.; PRIETO A.; LLORIS A.; PELAYO F.: "Módulos funcionales integrables multiumbrales y multivaluados en

tecnología IIL". Monografías del Departamento de Electrónica, No.9. Univ. de Granada.; 1986.

153. QUINE, W.V.: "The problem of simplifying truth functions". Am. Math. Monthly. Vol. 57, nº 8, pp. 521-531; 1952.

154. RAMANATHA, K.S.; BISWAS, N.N.: "A design for testability of undetectable crosspoint faults in programmable logic arrays". IEEE Trans. Comput.; vol. c-32, nº 6, pp. 551-557; June 1983.

155. REDDY, S.M.; HA, D.S.: "A new approach to the design of testable PLA's". IEEE Trans. Comput.; vol. c-36, pp. 201-211; Feb. 1987.

156. RICH, D.A.: "Survey of multivalued memories". IEEE Trans. Computer; pp. 99-106; Feb. 1986.

157. RICH, D.A.; NAIFF, K.C.; SMALLEY, K.G.: "A four-state ROM using multilevel process technology". IEEE J. Solid-State Circuits; vol. sc-19, pp. 174-179; 1984.

158. RINE, D.C.: "Computer Science and multiple-valued logic". North-Holland Publishing Company; 1977.

159. ROBINSON, J.P.; SAXENA, N.R.: "A unified view of test compression methods". IEEE Transactions on Computers; vol. c-36, nº 1, pp. 94-99; 1987.

160. RUDELL, R.; SANGIOVANNI-VICENTELLI, A.: "Multiple-valued minimization for PLA optimization". Proceeding of the 17th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 198-208; 1987.

161. SALUJA, K.K.; KINOSHITA, K.; FUJIWARA, H.: "An easily testable design of programmable logic arrays for multiple faults". IEEE Transactions on Computers; vol. c-32, nº 11, pp. 1038-1046; 1983.

162. SASAO, T.: "Multiple-valued logic and optimization of programmable logic arrays". Computer; pp. 71-80; April 1988.

163. SASAO, T.: "Mutiple-valued decomposition of generalized boolean functions and the complexity of programmable logic arrays". IEEE Transactions on Computers; vol. c-30, nº 9, pp. 635-643; 1981.

164. SASAO, T.: "On the optimal design of multiple-valued PLA's". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 214-222; 1986.

165. SASAO, T.: "Input variable assignement and output phase optimization of PLAs". IEEE Trans. on Computers; pp. 879-894; october 1984.

166. SAVIR, J.; DITLOW, G.; BARDELL, P.H.: "Random pattern Testability". Dig. 13 th Annu. Int. Symp. Fault-Tolerant Comput.; pp. 80-89; 1983.

167. SHIVASHANKAR, H.N.; SHIVAPRASAD, A.P.: "Pipeline analogue to ternary converter". Int. J. Electronics; vol. 53, nº 4, pp. 357-361; 1982.

168. SILIO, Ch.B.; PUGSLEY, J.H.; JENG, B.A.: "Control memory reduction using multivalued ROM'S". Proceeding of the 9th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 19-26; 1979.

169. SINGH, A.D.; ARMSTRONG, J.R.; GRAY, F.G.: "Combinational and sequential multivalued logic design using universal iterative tree structures". Proceeding of the 9th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 182-189; 1979.

170. SMITH, J.E.: "Detection of faults in programmable logic arrays". IEEE Transactions on Computers; vol. c-28, nº 11, pp. 845-853; 1979.

171. SMITH, J.E.: "Measures of the effectiveness of fault signature analysis". IEEE Transactions on Computers; vol. c-29, nº 6 , pp. 510-514; 1980.

172. SMITH, K.C.: "Circuits for multiple valued logic-A tutorial and appreciation".Proceeding of the 6th International Symposium on Multiple-Valued Logic. IEEE Computer Society. pp. 30-43; 1976.

173. SMITH, K.C.: "The prospects for multivalued logic: a technology and applications view". IEEE Transactions on Computers; vol. 30, nº 9, pp. 619-634; 1981.

174. SMITH, K.C.: "Multiple-valued logic: a tutorial and appreciation". Computer; pp. 17-27; April 1988.

175. SOMENZI, F.; GAI, S.: "Fault detection in programmable logic arrays". Proceedings of the IEEE; vol. 74, nº 5, pp. 655-668; May, 1986.

176. STARK, M.: "Two bits per cell ROM". Proc. COMPCON; pp. 209-212; 1981.

177. STEPHENSON, J.E.; GRASON, J.: "A Testability measure for register transfer level digital circuits". Dig. 6th International-nal Symposium on Fault-Tolerant Computing (FTCS-6). pp. 101-107. June 1976.

178. TAKIYAMA, R.: "Multiple threshold perceptron". Pattern Recognition.; vol. 10, no. 1, pp. 27-30.; 1979.

179. TANIGUCHI, K.; INOUE, T.; UENO, F.: "Realization and analysis of a mask-programmable I^2L multivalued logic circuit". Proceeding of the 13th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 196-200; 1983.

180. TANOUE, T.; MIZUTA, H.; TAKAHASHI,S.: "A triple-well resonant-tunneling diode for multiple-valued logic application". IEEE Electron Device Letters; vol. 9, nº 8, pp. 365-367; 1988.

181. TIRUMALAI, P.; BUTLER, J.T.: "Analysis of minimization algorithms for multiple-valued programmable logic arrays". pp. 226-236; 1988.

182. TIRUMALAI, P.; BUTLER, J.T.: "On the realization of multiple-valued logic functions using CCD PLA'S". Proceeding of the 14th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 33-42; 1984.

183. TREUER, R.; FUJIWARA, H.; AGARWAL, V.K.: "Implementing a built-in self-test PLA design". IEEE Design & Test; pp. 37-48; 1985.

184. TSIVIDIS, Y.P.: "Operation and modeling of the MOS transistor". McGraw-Hill; 1987.

185. UENO, F.; INOUE, T.; TANIGUCHI, K.; SAKAMOTO, N.: "Design method of CMOS quaternary threshold logic circuits realized by multiple ion implants". Proceeding of the 17th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 268-274; 1987.

186. UPADHYAYA, S.J.; SALUJA, K.K.: "A new approach to the design of built-in self-testing PLA's for high fault coverage". IEEE Transacionts on Computer-Aided Design; vol. 7, nº 1, pp. 60-67; 1988.

187. VALDERRAMA, E.: "Diseño asistido por computadora de circuitos integrados". VIII Escuela de Verano de Informática (AEIA). Monte Real; Portugal. Agosto 1986.

188. VLADIMIRESCU, A.; LIU, S.: "The simulation of MOS integrated circuits using SPICE2". Memorandum No, UCB/ERL M80/7; October 1980.

189. VRANESIC, Z.G.: "Applications and scope of multiple-valued LSI technology". Proceeding of the 11th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 213-216; 1981.

190. VRANESIC, Z.G.; SMITH, K.C.: "Engineering aspects of multi-valued logic systems". Computer; pp. 34-41; 1974.

191. WAGNER, K.D.; CHIN, C.K.; McCLUSKEY, E.J.: "Pseudorandom testing". IEEE Transactions on Computers; vol. c-36, nº 3, pp. 332-343; 1987.

192. WATANABE, T.; MATSUMOTO, M.: "Design of quaternary switching circuits using individual MOS pass-transistors and capacitor memory". Proceeding of the 16th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 26-32. 1986.

193. WESTE, N.; ESHRAGHIAN, K.: "Principles of CMOS VLSI design"; Addison-Wesley; 1985.

194. WHEATON, L.B.; CURRENT, K.W.: "A complementary-input quaternary threshold logic full adder". Proceeding of the 12th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 17-20; 1982.

195. WHEATON, L.B.; CURRENT, K.W.: "A modulo-four multiplier circuit for use in residue number system non-recursive digital filters". Int. J. Electronics; vol. 53, nº 5, pp. 473-483; 1982.

196. WHEATON, L.B.; CURRENT, K.W.: "Quaternary threshold logic full-adder circuit with complementary inputs". Int. J. Electronics; Vol. 56, nº 4, pp. 539-545; 1984.

197. WIDROW B.; WINTER R. "Neural Nets for Adaptative Filtering and Adaptive Pattern Recognition" Computer; Vol. 21, Nº 3, pp.25-39; 1988

198. WILLIAMS, T.; DAEHN, W.; GRUETZNER, M.; STARKE, C.W.: "Bounds and analysis of aliasing errors in linear feedback shift registers". IEEE Transactions on Computer Aided Design; vol. 7, nº 1, pp. 75-83; 1988.

199. WILLIAMS, T.W.; PARKER, K.P.: "Design for testability a survey". IEEE Transactions on Computers; vol. c-31, nº 1, pp. 2-15; 1982.

200. WILLIAMS, T.W.; PARKER, K.P.: "Testing logic networks and design for testability". Computer; pp. 9-21; Oct. 1979.

201. WU, X.; PROSSER, F.: "Ternary CMOS sequential circuits". Proceeding of the 18th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 307-313; 1988.

202. XUNWEI, W.; XIEXIONG, C.: "The synthesis of ternary functions under fixed polarities and ternary I^2L circuits". Scientia Sinica (series A); vol. XXVI, nº 12, pp. 1326-1337; 1983.

203. YAMADA, M.; FUJISHIMA, K.; NAGASAWA, K.; GAMOU, Y.: "A new multilevel storage structure for high density CCD memory". IEEE Journal of Solid-State Circuits; vol. SC-13, nº 5, pp. 688-692; 1978.

204. YAMAKAWA, T.: "CMOS multivalued circuits in hybrid mode". Proceeding of the 15th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 144-151; 1985.

205. ZUKERAN, C.; AFUSO, C.; KAMEYAMA, M.; HIGUCHI, T.: "Design of new low power quaternary CMOS logic circuits based on multiple ion implants". Proceeding of the 15th International Symposium on Multiple-Valued Logic. IEEE Computer Society; pp. 84-90. 1985.

APENDICE A: Listados de los programas de minimización.

```
C-----------------------------------------------------------------
C
C                  PROGRAMA PRINCIPAL MINTSUM1
C
C     Simplificacion de funciones multivaluadas para realizaciones
C    minimo/(suma truncada) segun el algoritmo propuesto
C    por Dueck y Miller (1987).
C
C-----------------------------------------------------------------

      IMPLICIT INTEGER (A-T)
      PARAMETER (R=4,N=8)
      CHARACTER*4 ADJ(N),V(N,2**(R-1))
      INTEGER*2 TFUN(0:R**N-1),FUN(0:R**N-1)
      DIMENSION COORD(0:(R**N)/2)
      COMMON // NR,TFUN,MINVAL,FUN,/MIN/
     +    MINTERM(0:N-1),/NRR/ NRR, /ADJ/ ADJ, /INDICE/ INDICE(N),
     +    /INDPROD/ INDPROD(N), /V/ V

      OPEN(1,FILE="@DATA")
      OPEN(2,FILE="@LIST")


C          Lectura de NR,FUN, y TFUN:

      READ(1,*) NR
      IF(NR.GT.N.OR.NR.LT.1) STOP
      READ(1,*) (FUN(J),J=0,R**NR-1)
      DO K=0,R**NR-1
      TFUN(K)=FUN(K)
      ENDDO

     NTERMS=0


C          Determinacion del conjunto de minterms de valor minimo:

1         MINVAL=R-1 !MINVAL=valor minimo de minterm
          NMINVAL=0   !NMINVAL=numero de minterms de valor minimo

          DO I=0,R**NR-1
               IF(TFUN(I).NE.0.AND.TFUN(I).NE.R.AND.FUN(I).NE.R) THEN
                    IF(TFUN(I).EQ.MINVAL) THEN
                    IF(NMINVAL.GT.((R**N)/2)) PRINT*,"ERROR EN COORD
```

```
                    COORD(NMINVAL)=I
                    NMINVAL=NMINVAL+1
               ELSE IF(TFUN(I).LT.MINVAL) THEN
                    NMINVAL=1
                    COORD(0)=I
                    MINVAL=TFUN(I)
                    ENDIF
               ENDIF
          ENDDO

C         Los minterms minimos tienen sus posiciones almacenadas en
C         COORD(0:NMINVAL-1) y su valor es MINVAL.

            IF(NMINVAL.LT.1) THEN

          WRITE(2,*) "Numero de terminos producto:.....", NTERMS
          WRITE(2,*) "Fin de ejecucion de MINTSUM1"
          PRINT*,"MINTSUM1.....", NTERMS," implicantes"
          CLOSE (2)
          CLOSE (1)

          STOP
            ENDIF

C
C         Determinacion del factor de aislamiento de cada uno de los
C         minterms de valor MINVAL y seleccion de uno de ellos para
C         el cual este factor sea maximo:
C         XAIS:factor de aislamiento
C         XMAIS:factor de aislamiento maximo
C         En COORAIS queda la posicion del minterm elegido

            XMAIS=0
            DO K1=0,NMINVAL-1
            NDAE=0 !numero de direcciones de adyacencias expandibles
            NAE=0  !numero de adyacencias expandibles en una direccion

            CALL COORDENADAS(COORD(K1))

                 DO K3=0,NR-1
                 POSICION=COORD(K1)-MINTERM(K3)*R**K3
C         POSICION: posicion base respecto de la variable K3-th
                 DO K4=0,R-1
                 GAMMA=0
                 IF(POSICION.NE.COORD(K1).AND.TFUN(POSICION).GE.MINVAL)
THEN
                 GAMMA=1
                 NAE=NAE+1
                 ENDIF
                 POSICION=POSICION+R**K3
                 ENDDO
               NDAE=NDAE+GAMMA
               ENDDO
          XAIS=1/(NDAE*(R-1)+NAE+1)
          IF(XAIS.GE.XMAIS) THEN
```

```
XMAIS=XAIS
COORAIS=COORD(K1)
ENDIF
ENDDO
```

```
        CALL COORDENADAS(COORAIS)

C       Construccion de los vectores de adyacencias expandibles en las
C       distintas direcciones (para el minterm seleccionado):
C       Se hace uso de la matriz character*4: ADJ(N)
C       ADJ(I)(1:1)="MINTERM(I-1)", I=1,2,..,NR
C       A cada ADJ(I) se añaden por orden los valores (caracter) de
C       coordenadas de adyacencias expandibles del minterm en la
C       direccion I.

        DO K5=1,NR
        ADJ(K5)=CHAR(MINTERM(K5-1)+48)//"    "
        ENDDO

        DO K6=1,NR
              DO K7=0,R-1
              IF(K7.NE.MINTERM(K6-1)) THEN
              POSICION=COORAIS-(MINTERM(K6-1)-K7)*R**(K6-1)
              IF(TFUN(POSICION).GE.MINVAL) THEN
              A=LENG(ADJ(K6))+1
              ADJ(K6)(A:A)=CHAR(K7+48)
              ENDIF
              ENDIF
              ENDDO
     ENDDO


C       Construccion de los conjuntos V(I,J): conjuntos de adyacencias
C       expandibles en la direccion I que incluyan el minterm.
C       Para logica 4-valuada existe un maximo de 2**(4-1)=8 conjuntos
C       para cada I; es decir Jmax=8

C       Si ADJ(I) tiene una longitud de L caracteres, se pueden formar
C       2**(L-1) conjuntos correspondientes a la direccion I.

        DO K8=1,NR
        V(K8,1)=ADJ(K8)(1:1)//"    "
        IF(LENG(ADJ(K8)).GE.2) V(K8,2)=ADJ(K8)(1:2)//"   "
        IF(LENG(ADJ(K8)).GE.3) THEN
        V(K8,3)=ADJ(K8)(1:1)//ADJ(K8)(3:3)//"   "
        V(K8,4)=ADJ(K8)(1:3)//" "
        ENDIF
        IF(LENG(ADJ(K8)).EQ.4) THEN
        V(K8,5)=ADJ(K8)(1:1)//ADJ(K8)(4:4)//"   "
        V(K8,6)=ADJ(K8)(1:2)//ADJ(K8)(4:4)//" "
```

```
          V(K8,7)=ADJ(K8)(1:1)//ADJ(K8)(3:4)//" "
          V(K8,8)=ADJ(K8)(1:4)
          ENDIF
          ENDDO


C         A continuacion, utilizando las rutinas PRODUCTO, COBERTURA, e
C         IMPLICANTE, se procede a la eleccion del implicante optimo
C         (de valor MINVAL) que sera incluido en la expresion de la
C         funcion:

            DO K9=1,NR
            INDPROD(K9)=1
            ENDDO

C         INDPROD contiene los segundos indices de los conjuntos V(I,J)
C         que definen el implicante optimo.  Se comienza haciendolos 1
C         (es decir, considerando el propio minterm)

            NRR=32000
            DIR=0
            CALL PRODUCTO(DIR)

C         ya se tiene definido el implicante optimo.

                  NTERMS=NTERMS+1


      WRITE(2,*) MINVAL,"*",( "X",I,"[",V(I,INDPROD(I)),"] " , I=1,NR),"+"
C               Se resta de TFUN el implicante seleccionado.
C               Si en alguna celda se tenia inicialmente el valor R-1 y tras
C               actualizar TFUN queda 0, esta celda se  pondra como indiferencia


              DIR3=0
              CALL RESTA(DIR3)

C         Se inicia otra iteracion:

          GO TO 1
            END
C*************************************************************************

              SUBROUTINE RESTA(DIR3)

C         Resta MINVAL a todas las celdas del implicante seleccionado.
C         Si alguna de estas celdas queda con valor 0, y en la definicion
C         de la funcion tenia asignado el valor R-1, la deja con valor R.
```

```
      IMPLICIT INTEGER (A-T)
      PARAMETER (N=8,R=4)
      INTEGER*2 TFUN(0:R**N-1),FUN(0:R**N-1)
      CHARACTER*8 CELDA
      CHARACTER*4 V(N,2**(R-1))
      COMMON // NR,TFUN,MINVAL,FUN,
     +     /V/ V, /INDPROD/ INDPROD(N), /CELDA/ CELDA

      DIRAUX3=DIR3+1
      DO K=1,LENG(V(DIRAUX3,INDPROD(DIRAUX3)))
      CELDA(DIRAUX3:DIRAUX3)=V(DIRAUX3,INDPROD(DIRAUX3))(K:K)
      IF(DIRAUX3.EQ.NR) THEN
      NCELDA=0
            DO M=1,NR
            NCELDA=NCELDA+(ICHAR(CELDA(M:M))-48)*R**(M-1)
            ENDDO

      IF(TFUN(NCELDA).NE.R) TFUN(NCELDA)=TFUN(NCELDA)-MINVAL
      IF(TFUN(NCELDA).EQ.0.AND.FUN(NCELDA).EQ.(R-1)) TFUN(NCELDA)=R
      ELSE
      CALL RESTA(DIRAUX3)
      ENDIF
      ENDDO
      RETURN
      END
C*************************************************************************

        SUBROUTINE COORDENADAS(POSICION)

C       Calcula las coordenadas en el mapa NR-dimensional de la funcion
C       de la celda definida por POSICION.
C       MINTERM(0:NR-1):coordenadas resultantes.
C       MINTERM(0):coordenada menos significativa

        IMPLICIT INTEGER (A-T)
        PARAMETER (R=4,N=8)
        COMMON // NR, /MIN/ MINTERM(0:N-1)
        DIV=POSICION
        DO K=0,NR-2
        COC=DIV/R +.01
        MINTERM(K)=DIV-COC*R
        DIV=COC
        ENDDO

        MINTERM(NR-1)=DIV
        RETURN
        END
C*************************************************************************

        SUBROUTINE PRODUCTO(DIR)
```

```
C       Calculo recursivo de cubos.  Llama a COBERTURA para determinar
C       si todas las celdas del cubo estan ocupadas. COBERTURA llama a
C       su vez a IMPLICANTE para calcular el numero de rupturas relati-
C       vo del termino producto considerado.
C       Como resultado final, en INDPROD quedan los segundos indices
C       de los conjuntos V(I,J) que definen el termino producto optimo.

          IMPLICIT INTEGER (A-T)
          PARAMETER (N=8,R=4)
          CHARACTER*4 ADJ(N)
          DIMENSION INDICE(N)
          COMMON // NR, /ADJ/ ADJ, /NRR/ NRR, /INDICE/ INDICE

          DIRAUX=DIR+1
          DO J=1,2**(LENG(ADJ(DIRAUX))-1)
          INDICE(DIRAUX)=J
          IF(DIRAUX.EQ.NR) THEN

          DIR1=0
          CALL COBERTURA(DIR1)
          ELSE
          CALL PRODUCTO(DIRAUX)
          ENDIF
          ENDDO
          RETURN
          END
C*************************************************************************

            SUBROUTINE COBERTURA(DIR1)

C       Comprueba si todas las celdas del cubo:
C       V(1,INDICE(1))xV(2,INDICE(2))x...xV(NR,INDICE(NR))
C       son no nulas.  Si encuentra alguna nula retorna; si no,
C       se trata de un implicante el cual pasa a ser analizado
C       por la rutina IMPLICANTE.

            IMPLICIT INTEGER (A-T)
            PARAMETER (N=8,R=4)
            INTEGER*2 TFUN(0:R**N-1)
            CHARACTER*8 CELDA  !N=8
            CHARACTER*4 ADJ(N), V(N,2**(R-1))
            COMMON // NR,TFUN, /INDICE/ INDICE(N),
     +     /V/ V, /CEL/ CELDA,/CUBRE/ CUBRE

            DIRAUX1=DIR1+1

          IF(DIR1.EQ.0) CUBRE=1



            DO K=1,LENG(V(DIRAUX1,INDICE(DIRAUX1)))
            CELDA(DIRAUX1:DIRAUX1)=V(DIRAUX1,INDICE(DIRAUX1))(K:K)
```

```
IF(DIRAUX1.EQ.NR)THEN
NCELDA=0

      DO M=1,NR
      NCELDA=NCELDA+(ICHAR(CELDA(M:M))-48)*R**(M-1)

      ENDDO
        IF (TFUN(NCELDA).LT.1) CUBRE=0

      ELSE
      CALL COBERTURA(DIRAUX1)
      ENDIF
      ENDDO

      IF(DIR1.EQ.0) THEN
            IF(CUBRE.NE.0)THEN

      DIR2=0
      CALL IMPLICANTE(DIR2)
            ENDIF
      ENDIF

      RETURN
      END

C*********************************************************************

        SUBROUTINE IMPLICANTE(DIR2)

C       Calcula el numero de rupturas relativo (NRRP) del implicante
C       definido por:
C       V(1,INDICE(1))xV(2,INDICE(2))x...xV(NR,INDICE(NR)).
C        Si NRRP resulta menor o igual que NRR, lo sustituye, y actualiz



C       INDPROD(I) con INDICE(I); I=1,.,NR
        IMPLICIT INTEGER (A-T)
        PARAMETER (N=8,R=4)
        INTEGER*2 TFUN(0:R**N-1)
        CHARACTER*1 COOR
        CHARACTER*8 CELDA !N=8
        CHARACTER*4 V(N,R**(N-1))
        COMMON // NR,TFUN,MINVAL, /NRR/ NRR, /INDICE/
     +  INDICE(N), /INDPROD/ INDPROD(N), /V/ V, /MASMENOS/ NMAS(N),
     +  NMENOS(N), /CELD/ CELDA,/NRRP/ NRRP

        IF(DIR2.EQ.0) NRRP=0
        DIRAUX2=DIR2+1
        DO K=1,LENG(V(DIRAUX2,INDICE(DIRAUX2)))
        CELDA(DIRAUX2:DIRAUX2)=V(DIRAUX2,INDICE(DIRAUX2))(K:K)

        IF(DIRAUX2.EQ.NR) THEN !se tiene def. una celda del impl.
        NCELDA=0
              DO K1=1,NR
              NCELDA=NCELDA+(ICHAR(CELDA(K1:K1))-48)*R**(K1-1)
```

```
            NMAS(K1)=0
            NMENOS(K1)=0
            ENDDO

          IF(TFUN(NCELDA).NE.R) THEN
                DO IDIR=1,NR
                  DO ICOOR=0,R-1
                  COOR=CHAR(48+ICOOR)
                  IF(INDEX(V(IDIR,INDICE(IDIR)),COOR).EQ.0) THEN
                   NCELCOOR=NCELDA-((ICHAR(CELDA(IDIR:IDIR))-48)-
     +     ICOOR)*R**(IDIR-1)
                   IF(TFUN(NCELCOOR).EQ.(TFUN(NCELDA)-MINVAL))
     +            NMENOS(IDIR)=-1
                   IF(TFUN(NCELCOOR).EQ.TFUN(NCELDA).AND.TFUN(NCELDA)
     +            .NE.(R-1)) NMAS(IDIR)=1
                   ENDIF
                   ENDDO
                ENDDO
                  DO K2=1,NR
                  NRRP=NRRP+NMAS(K2)+NMENOS(K2)
                  ENDDO
          ENDIF

          ELSE
          CALL IMPLICANTE(DIRAUX2)
          ENDIF
          ENDDO

          IF(DIR2.EQ.0.AND.NRRP.LE.NRR) THEN
          NRR=NRRP
          DO J=1,NR
          INDPROD(J)=INDICE(J)
          ENDDO
          ENDIF

          RETURN
          END
C*************************************************************************

          INTEGER FUNCTION LENG(CADENA)
C         Devuelve la longitud de la cadena indicada como argumento.

          IMPLICIT INTEGER (A-T)
            CHARACTER*(*) CADENA

            DO I=LEN(CADENA),1,-1
            IF(CADENA(I:I).NE." ") THEN
            LENG=I
            RETURN
            ENDIF
            ENDDO
            LENG=0
            END
```

```
C-----------------------------------------------------------------------
C
C                    PROGRAMA PRINCIPAL MINTSUM2
C
C       Simplificacion de funciones multivaluadas para realizaciones
C       minimo/(suma truncada). Salvo la rutina IMPLICANTE, todas las
C       subrutinas utilizadas por este programa coinciden con las de
C       MINTSUM1.
C
C-----------------------------------------------------------------------


            IMPLICIT INTEGER (A-T)
            PARAMETER (R=4,N=8)
            CHARACTER*4 ADJ(N),V(N,2**(R-1))
            INTEGER*2 TFUN(0:R**N-1),FUN(0:R**N-1)
            DIMENSION COORD(0:(R**N)/2)
            COMMON // NR,TFUN,MINVAL,FUN,/MIN/
     +      MINTERM(0:N-1),/NRR/ NRR, /ADJ/ ADJ, /INDICE/ INDICE(N),
     +      /INDPROD/ INDPROD(N), /V/ V

            OPEN(1,FILE="@DATA")
            OPEN(2,FILE="@LIST")


C           Lectura de NR,FUN, y TFUN:

            READ(1,*) NR
            IF(NR.GT.N.OR.NR.LT.1) STOP
            READ(1,*) (FUN(J),J=0,R**NR-1)
            DO K=0,R**NR-1
            TFUN(K)=FUN(K)
            ENDDO

                 NTERMS=0



C           Determinacion del conjunto de minterms de valor minimo:
1           MINVAL=R-1 !MINVAL=valor minimo de minterm
            NMINVAL=0   !NMINVAL=numero de minterms de valor m!nimo

            DO I=0,R**NR-1
                 IF(TFUN(I).NE.0.AND.TFUN(I).NE.R.AND.FUN(I).NE.R) THEN
                         IF(TFUN(I).EQ.MINVAL) THEN
                         IF(NMINVAL.GT.((R**N)/2)) PRINT*,"ERROR EN COORD



                         COORD(NMINVAL)=I
                         NMINVAL=NMINVAL+1
                    ELSE IF(TFUN(I).LT.MINVAL) THEN
                         NMINVAL=1
```

```
                    COORD(0)=I
                    MINVAL=TFUN(I)
                    ENDIF
                ENDIF
            ENDDO

C       Los minterms minimos tienen sus posiciones almacenadas en
C       COORD(0:NMINVAL-1) y su valor es MINVAL.

          IF(NMINVAL.LT.1) THEN
        WRITE(2,*) "Numero de terminos producto.....",NTERMS
        WRITE(2,*) "Fin de ejecucion de MINTSUM2"
        PRINT*,"MINTSUM2......",NTERMS," implicantes"
        CLOSE (2)
        CLOSE (1)

        STOP
          ENDIF

C       Determinacion del factor de aislamiento de cada uno de los
C       minterms de valor MINVAL y seleccion de uno de ellos para
C       el cual este factor sea minimo:

          XMAIS=32000
          DO K1=0,NMINVAL-1
          NDAE=0 !numero de direcciones de adyacencias expandibles
          NAE=0  !numero de adyacencias expandibles en una direccion

          CALL COORDENADAS(COORD(K1))

                DO K3=0,NR-1
                POSICION=COORD(K1)-MINTERM(K3)*R**K3
C           POSICION: posicion base respecto de la variable K3-th
                  DO K4=0,R-1
                  GAMMA=0
              IF(POSICION.NE.COORD(K1).AND.TFUN(POSICION).GE.MINVAL) THEN


                  GAMMA=1
                  NAE=NAE+1
                  ENDIF
                  POSICION=POSICION+R**K3
                    ENDDO
                NDAE=NDAE+GAMMA
                ENDDO
        XAIS=NDAE*(R-1)+NAE
        IF(XAIS.LT.XMAIS) THEN
        XMAIS=XAIS
        COORAIS=COORD(K1)
        ENDIF
        ENDDO
```

```
      CALL COORDENADAS(COORAIS)

C     Construccion de los vectores de adyacencias expandibles:

      DO K5=1,NR
      ADJ(K5)=CHAR(MINTERM(K5-1)+48)//"    "
      ENDDO

      DO K6=1,NR
            DO K7=0,R-1
            IF(K7.NE.MINTERM(K6-1)) THEN
            POSICION=COORAIS-(MINTERM(K6-1)-K7)*R**(K6-1)
            IF(TFUN(POSICION).GE.MINVAL) THEN
            A=LENG(ADJ(K6))+1
            ADJ(K6)(A:A)=CHAR(K7+48)
            ENDIF
            ENDIF
            ENDDO
      ENDDO


C     Construccion de los conjuntos V(I,J):

      DO K8=1,NR
      V(K8,1)=ADJ(K8)(1:1)//"    "
      IF(LENG(ADJ(K8)).GE.2) V(K8,2)=ADJ(K8)(1:2)//"   "
      IF(LENG(ADJ(K8)).GE.3) THEN
      V(K8,3)=ADJ(K8)(1:1)//ADJ(K8)(3:3)//"   "
      V(K8,4)=ADJ(K8)(1:3)//" "
      ENDIF
      IF(LENG(ADJ(K8)).EQ.4) THEN
      V(K8,5)=ADJ(K8)(1:1)//ADJ(K8)(4:4)//"   "
      V(K8,6)=ADJ(K8)(1:2)//ADJ(K8)(4:4)//" "
      V(K8,7)=ADJ(K8)(1:1)//ADJ(K8)(3:4)//" "
      V(K8,8)=ADJ(K8)(1:4)
      ENDIF
      ENDDO


C     A continuacion, utilizando las rutinas PRODUCTO, COBERTURA, e
C     IMPLICANTE, se procede a la eleccion del implicante optimo
C     (de valor MINVAL) que sera incluido en la expresion de la
C     funcion:

      DO K9=1,NR
      INDPROD(K9)=1
      ENDDO

C     INDPROD contiene los segundos indices de los conjuntos V(I,J)
C     que definen el implicante optimo.  Se comienza haciendolos 1
C     (es decir, considerando el propio minterm)

      NRR=32000
      DIR=0
      CALL PRODUCTO(DIR)
```

```
  310

C       ya se tiene definido el implicante optimo.

                NTERMS=NTERMS+1

      WRITE(2,*) MINVAL,"*",( "X",I,"[",V(I,INDPROD(I)),"] " , I=1,NR),"+

          DIR3=0
          CALL RESTA(DIR3)

C         Se inicia otra iteracion:

          GO TO 1
            END
C*************************************************************************

            SUBROUTINE IMPLICANTE(DIR2)
C         Calcula el numero de rupturas relativo (NRRP) del implicante
C         definido por:
C         V(1,INDICE(1))xV(2,INDICE(2))x...xV(NR,INDICE(NR)).
C         Si NRRP resulta menor o igual que NRR, lo  sustituye, y actualiz



C         INDPROD(I) con INDICE(I); I=1,.,NR
            IMPLICIT INTEGER (A-T)
            PARAMETER (N=8,R=4)
            INTEGER*2 TFUN(0:R**N-1)
            CHARACTER*1 COOR
            CHARACTER*8 CELDA !N=8
            CHARACTER*4 V(N,R**(N-1))
            COMMON // NR,TFUN,MINVAL, /NRR/ NRR, /INDICE/
     +      INDICE(N), /INDPROD/ INDPROD(N), /V/ V, /MASMENOS/ NMAS(N),
     +      NMENOS(N), /CELD/ CELDA,/NRRP/ NRRP

            IF(DIR2.EQ.0) NRRP=0
            DIRAUX2=DIR2+1
            DO K=1,LENG(V(DIRAUX2,INDICE(DIRAUX2)))
            CELDA(DIRAUX2:DIRAUX2)=V(DIRAUX2,INDICE(DIRAUX2))(K:K)

            IF(DIRAUX2.EQ.NR) THEN !se tiene def. una celda del impl.
            NCELDA=0
                  DO K1=1,NR
                  NCELDA=NCELDA+(ICHAR(CELDA(K1:K1))-48)*R**(K1-1)
                  NMAS(K1)=0
                  NMENOS(K1)=0
                  ENDDO
```

```
      IF(TFUN(NCELDA).NE.R) THEN
            DO IDIR=1,NR
              DO ICOOR=0,R-1
              COOR=CHAR(48+ICOOR)
              IF(INDEX(V(IDIR,INDICE(IDIR)),COOR).EQ.0) THEN
              NCELCOOR=NCELDA-((ICHAR(CELDA(IDIR:IDIR))-48)-
     +     ICOOR)*R**(IDIR-1)
              IF(TFUN(NCELCOOR).EQ.(TFUN(NCELDA)-MINVAL))
     +             NMENOS(IDIR)=NMENOS(IDIR)-1
               IF(TFUN(NCELCOOR).EQ.TFUN(NCELDA))
     +             NMAS(IDIR)=NMAS(IDIR)+1
              ENDIF
              ENDDO
            ENDDO
              DO K2=1,NR
              NRRP=NRRP+NMAS(K2)+NMENOS(K2)
              ENDDO
      ENDIF

      ELSE
      CALL IMPLICANTE(DIRAUX2)
      ENDIF
      ENDDO

      IF(DIR2.EQ.0.AND.NRRP.LE.NRR) THEN
      NRR=NRRP
      DO J=1,NR
      INDPROD(J)=INDICE(J)
      ENDDO
      ENDIF

      RETURN
      END
```

```
C-----------------------------------------------------------------------
C
C                  PROGRAMA PRINCIPAL MINTSUM3
C
C        Simplificacion de funciones multivaluadas para realizaciones
C        minimo/(suma truncada). Difiere de MINTSUM2 en la seleccion
C        del minterm a cubrir: no tiene en cuenta el aislamiento.
C
C-----------------------------------------------------------------------

      IMPLICIT INTEGER (A-T)
      PARAMETER (R=4,N=8)
      CHARACTER*4 ADJ(N),V(N,2**(R-1))
      INTEGER*2 TFUN(0:R**N-1),FUN(0:R**N-1)
      COMMON // NR,TFUN,MINVAL,FUN,/MIN/
     +     MINTERM(0:N-1),/NRR/ NRR, /ADJ/ ADJ, /INDICE/ INDICE(N),
     +     /INDPROD/ INDPROD(N), /V/ V

      OPEN(1,FILE="@DATA")
      OPEN(2,FILE="@LIST")



C     Lectura de NR,FUN, y TFUN:

      READ(1,*) NR
      IF(NR.GT.N.OR.NR.LT.1) STOP
      READ(1,*) (FUN(J),J=0,R**NR-1)
      DO K=0,R**NR-1
      TFUN(K)=FUN(K)
      ENDDO


            NTERMS=0



C     Eleccion de un minterm de valor minimo:

1     MINVAL=R !MINVAL=valor minimo de minterm

      DO I=0,R**NR-1
            IF(TFUN(I).NE.0.AND.TFUN(I).NE.R.AND.FUN(I).NE.R) THEN
                  IF(TFUN(I).EQ.1) THEN
                  MINVAL=1
                  COORAIS=I
                  GO TO 2
                  ENDIF
                  IF(TFUN(I).LT.MINVAL) THEN
                  MINVAL=TFUN(I)
                  COORAIS=I
                  ENDIF
            ENDIF
      ENDDO
```

```
      IF(MINVAL.EQ.R) THEN
      WRITE(2,*) "Numero de terminos producto.....",NTERMS
      WRITE(2,*) "Fin de ejecucion de MINTSUM3"
      PRINT*, "MINTSUM3.......",NTERMS," implicantes"
      CLOSE (2)
      CLOSE (1)
      STOP
      ENDIF

2           CALL COORDENADAS(COORAIS)

C         Construccion de los vectores de adyacencias expandibles:

            DO K5=1,NR
            ADJ(K5)=CHAR(MINTERM(K5-1)+48)//"    "
            ENDDO

            DO K6=1,NR
                  DO K7=0,R-1
                  IF(K7.NE.MINTERM(K6-1)) THEN
                  POSICION=COORAIS-(MINTERM(K6-1)-K7)*R**(K6-1)
                  IF(TFUN(POSICION).GE.MINVAL) THEN
                  A=LENG(ADJ(K6))+1
                  ADJ(K6)(A:A)=CHAR(K7+48)
                  ENDIF
                  ENDIF
                  ENDDO
            ENDDO


C         Construccion de los conjuntos V(I,J):

            DO K8=1,NR
            V(K8,1)=ADJ(K8)(1:1)//"    "
            IF(LENG(ADJ(K8)).GE.2) V(K8,2)=ADJ(K8)(1:2)//"   "
            IF(LENG(ADJ(K8)).GE.3) THEN
            V(K8,3)=ADJ(K8)(1:1)//ADJ(K8)(3:3)//"   "
            V(K8,4)=ADJ(K8)(1:3)//" "
            ENDIF
            IF(LENG(ADJ(K8)).EQ.4) THEN
            V(K8,5)=ADJ(K8)(1:1)//ADJ(K8)(4:4)//"   "
            V(K8,6)=ADJ(K8)(1:2)//ADJ(K8)(4:4)//" "
            V(K8,7)=ADJ(K8)(1:1)//ADJ(K8)(3:4)//" "
            V(K8,8)=ADJ(K8)(1:4)
            ENDIF
            ENDDO


C         A continuacion, utilizando las rutinas PRODUCTO, COBERTURA, e
C         IMPLICANTE, se procede a la eleccion del implicante optimo
C         (de valor MINVAL) que sera incluido en la expresion de la
C         funcion:
```

```
          DO K9=1,NR
          INDPROD(K9)=1
          ENDDO

C         INDPROD contiene los segundos indices de los conjuntos V(I,J)
C         que definen el implicante optimo.  Se comienza haciendolos 1
C         (es decir, considerando el propio minterm)

          NRR=32000
          DIR=0
          CALL PRODUCTO(DIR)

C         ya se tiene definido el implicante optimo.


                NTERMS=NTERMS+1

      WRITE(2,*) MINVAL,"*",( "X",I,"[",V(I,INDPROD(I)),"] " , I=1,NR),"+

          DIR3=0
          CALL RESTA(DIR3)


C         Se inicia otra iteracion:

          GO TO 1
            END
```

APENDICE B: Listados de los ficheros de descripción SPICE.

```
CELDA GENERADORA DE LITERALES TETRAVALUADA TIPO NOR
*
*Tecnología CMOS (3 micrómetros)- Louvain
*
.OPTIONS NOPAGE LIMPTS=2000
.WIDTH OUT=80
.PROBE
*
*Modelo MOS3 semiempírico de Vladimirescu & al.
*
.MODEL NMOS NMOS(LEVEL=3 VTO=0.947 GAMMA=0.2925 RSH=33.38 CJ=0.15E-3
+               TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
+               LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
+          ETA=0.221 KAPPA=0.212)
*
.MODEL PMOS PMOS(LEVEL=3 VTO=-0.773 GAMMA=0.7164 RSH=49.40 CJ=0.25E-3
+          TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
+          LD=0.8E-6 UO=217 VMAX=2.09E5 DELTA=0.0 THETA=0.087
+          ETA=0.234 KAPPA=5.995)
*
*Fuentes, señales de entrada, y tipos de analisis:
*
VDD 1 0 5V
IREF 2 0 10U
IIN 0 50 PWL(0 0 0.5U 0 .51U 10U 1U 10U 1.01U 20U 1.5U 20U
+1.51U 30U 2U 30U 2.01U 0 2.5U 0 2.51U 30U 3U 30U)
*.DC IIN 0 30U 0.1U
R 50 4 1K
V0 20 0 0
V1 21 0 0
V2 22 0 0
V3 23 0 0
.TRAN 0.03U 3U
*
*Descripcion del circuito:
*
MP0 2 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP1 3 2 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MP2 6 2 1 1 PMOS L=6U W=13.5U PS=40U PD=40U AS=100P AD=100P
MP3 9 2 1 1 PMOS L=6U W=22.5U PS=40U PD=40U AS=100P AD=100P
MP4 5 3 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP5 7 6 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP6 8 9 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP7 14 20 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP8 10 5 14 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP9 15 7 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP10 16 21 15 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP11 11 3 16 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP12 19 23 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
```

```
MP13 13 9 19 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP14 17 8  1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP15 12 6 18 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP16 18 22 17 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
*
MN0 4 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN1 3 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN2 6 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN3 9 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN4 5 3 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN5 7 6 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN6 8 9 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN7 10 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN8 10 5 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN9 11 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN10 11 7 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN11 11 3 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN12 13 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN13 13 9 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN14 12 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN15 12  6 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN16 12  8 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*Capacidades de las interconexiones:
*
C20 20 0 .116P
C21 21 0 .086P
C22 22 0 .073P
C23 23 0 .079P
C2 2 0 .063P
C10 10 0 .083P
C11 11 0 .122P
C12 12 0 .138P
C13 13 0 .150P
C3 3 0 .097P
C5 5 0 .084P
C7 7 0 .090P
C6 6 0 .063P
C9 9 0 .040P
C8 8 0 .028P
C4 4 0 .024P
*
*Capacidades de carga en las salidas:
*
CL0 10 0 3P
CL1 11 0 3P
CL2 12 0 3P
CL3 13 0 3P
*
.END
```

```
CELDA GENERADORA DE LITERALES TETRAVALUADA TIPO NAND
*
*Tecnología CMOS (3 micrómetros)- Louvain
*
.OPTIONS NOPAGE LIMPTS=450
.WIDTH OUT=80
.PROBE
*
*Modelo MOS3 semiempírico de Vladimirescu & al.
*
.MODEL NMOS NMOS(LEVEL=3 VTO=0.947 GAMMA=0.2925 RSH=33.38 CJ=0.15E-3
+                TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
+                LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
+          ETA=0.221 KAPPA=0.212)
*
.MODEL PMOS PMOS(LEVEL=3 VTO=-0.773 GAMMA=0.7164 RSH=49.40 CJ=0.25E-3
+          TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
+          LD=0.8E-6 UO=217 VMAX=2.09E5 DELTA=0.0 THETA=0.087
+          ETA=0.234 KAPPA=5.995)
*
*Fuentes, señales de entrada y tipo de analisis:
*
VDD 1 0 5V
IREF 2 0 20U
IIN 0 50 PWL(0 0 0.2U 0 0.21U 20U 0.4U 20U 0.41U 40U 0.60U 40U
+0.61U 60U 0.8U 60U 0.81U 0 1U 0 1.01U 60U 1.2U 60U)
R 50 4 1K
MP0 2 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
.TRAN .01U 1.2U
*.DC IIN 0 60U .5U
*
*Descripcion del circuito:
*
MP1 3 2 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MP2 6 2 1 1 PMOS L=6U W=13.5U PS=40U PD=40U AS=100P AD=100P
MP3 9 2 1 1 PMOS L=6U W=22.5U PS=40U PD=40U AS=100P AD=100P
MP4 5 3 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP5 7 6 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP6 10 5 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP7 11 14 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP8 13 9 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP9 12 8 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP10 8 9 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP11 8 7 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP12 14 5 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP13 14 6 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
*
MN0 4 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN1 3 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN2 6 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN3 9 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN4 5 3 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN5 7 6 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN6 10 5 0 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
```

```
MN7 11 14 0 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN8 13 9 0 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN9 12 8 0 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN10 16 9 0 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN11 8 7 16 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN12 14 5 15 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN13 15 6 0 0  NMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
*
*Capacidades de las interconexiones:
*
C2 2 0 .063P
C3 3 0 .040P
C5 5 0 .084P
C6 6 0 .063P
C9 9 0 .063P
C7 7 0 .084P
C8 8 0 .084P
C4 4 0 .024P
*
* Capacidades de carga en las salidas:
*
CL0 10 0 3P
CL1 11 0 3P
CL2 12 0 3P
CL3 13 0 3P
.END
```

```
PLA DEL TIPO 1
*
*Tecnologia CMOS (3 micrometros)- Louvain
*
.OPTIONS ITL5=0 ITL4=20
.OPTIONS NOPAGE LIMPTS=2000 ABSTOL=1PA VNTOL=10UV RELTOL=.001
.WIDTH OUT=80
.PROBE V(21),V(22),V(23),V(24),V(11),V(12),V(13),V(14),V(15),
+V(16),V(17),I(R1),ID(MPS1),ID(MPS2),V(30),V(31),ID(M1111),
+ID(M1212),ID(M1313),ID(M1414),ID(M1515),ID(M1616),ID(M1717),
+ID(M3011),ID(M3012),ID(M3014),ID(M3111),ID(M3113),ID(M3115),
+ID(M3116),ID(M3117)
*
*Modelo MOS3 semiempirico de Vladimirescu & al.
*
.MODEL NMOS NMOS(LEVEL=3 VTO=0.947 GAMMA=0.2925 RSH=33.38 CJ=0.15E-3
+               TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
+               LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
+          ETA=0.221 KAPPA=0.212)
*
.MODEL PMOS PMOS(LEVEL=3 VTO=-0.773 GAMMA=0.7164 RSH=49.40 CJ=0.25E-3
+          TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
+          LD=0.8E-6 UO=217 VMAX=2.09E5 DELTA=0.0 THETA=0.087
+          ETA=0.234 KAPPA=5.995)
*
VDD1 1 0 5V
VDD2 300 0 5V
IREF1 2 0 10U
IREF2 200 0 10U
IIN1 0 51 PWL(0 0 .5U 0 .51U 10U 1.0U 10U 1.01U 20U 1.5U 20U
+1.51U 30U 2.0U 30U 2.01U 0 2.5U 0 2.51U 30U 3U 30U)
.TRAN 10N 3000N
*
R1 51 61 1K
XLIT1 300 200 61 21 22 23 24 100 LITNOR
VC 100 0 0
V25 25 0 0
V26 26 0 0
V27 27 0 5
V28 28 0 0
*
* Transistores de salida y de generaci n de VREF:
*
MP0 2 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP00 200 200 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MPS1 30 30 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MPS2 31 31 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
*
*Fuentes de corriente PMOS del plano TSUM:
*
MP1 11 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP2 12 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP3 13 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP4 14 2 1 1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP5 15 2 1 1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP6 16 2 1 1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
```

```
MP7 17 2 1 1 PMOS L=6U W=27U PS=40U PD=40U AS=100P AD=100P
*
*
*Transistores nmos del plano NOR:
*
M1123 11 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1126 11 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1127 11 27 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1128 11 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1221 12 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1223 12 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1224 12 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1225 12 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1226 12 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1228 12 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1321 13 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1322 13 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1324 13 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1325 13 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1326 13 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1421 14 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1422 14 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1424 14 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1428 14 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1521 15 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1522 15 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1525 15 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1528 15 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1621 16 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1624 16 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1625 16 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1626 16 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1627 16 27 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1722 17 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1724 17 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1725 17 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1726 17 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1728 17 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*
*Transistores nmos del plano TSUM:
*
M1111 11 11 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1212 12 12 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1313 13 13 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1414 14 14 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1515 15 15 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1616 16 16 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1717 17 17 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3011 30 11 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3012 30 12 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3014 30 14 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3111 31 11 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3113 31 13 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3115 31 15 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3116 31 16 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
```

```
M3117 31 17 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
*
*
*Capacidades parasitas de las lineas de literal, de producto
*y de salida (C(poli/subs)=.6e-4pf/µm²  C(metal)/subs)=.3pfe-4pf/µm²):
C21 21 0 .03P
C22 22 0 .03P
C23 23 0 .03P
C24 24 0 .03P
C25 25 0 .03P
C26 26 0 .03P
C27 27 0 .03P
C28 28 0 .03P
C11 11 0 .05P
C12 12 0 .05P
C13 13 0 .05P
C14 14 0 .05P
C15 15 0 .05P
C16 16 0 .05P
C17 17 0 .05P
C30 30 0 .03P
C31 31 0 .03P
*
**********************************************************************
.SUBCKT LITNOR  1    2    4    10    11    12    13    20
*               VDD VREF IIN V0    V1    V2    V3    VC
*Descripcion del circuito:
*
MP1 3 2 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MP2 6 2 1 1 PMOS L=6U W=13.5U PS=40U PD=40U AS=100P AD=100P
MP3 9 2 1 1 PMOS L=6U W=22.5U PS=40U PD=40U AS=100P AD=100P
MP4 5 3 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP5 7 6 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP6 8 9 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP7 14 20 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP8 10 5 14 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP9 15 7 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP10 16 20 15 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP11 11 3 16 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP12 19 20 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP13 13 9 19 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP14 17 8  1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP15 12 6 18 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP16 18 20 17 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
*
MN0 4 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN1 3 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN2 6 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN3 9 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN4 5 3 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN5 7 6 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN6 8 9 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN7 10 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN8 10 5 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN9 11 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN10 11 7 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
```

```
MN11 11 3 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN12 13 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN13 13 9 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN14 12 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN15 12  6 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN16 12  8 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*Capacidades de las interconexiones:
*
C20 20 0 .116P
C21 20 0 .086P
C22 20 0 .073P
C23 20 0 .079P
C2 2 0 .063P
C10 10 0 .083P
C11 11 0 .122P
C12 12 0 .138P
C13 13 0 .150P
C3 3 0 .097P
C5 5 0 .084P
C7 7 0 .090P
C6 6 0 .063P
C9 9 0 .040P
C8 8 0 .028P
C4 4 0 .024P
*
.ENDS
********************************************************************************
.END
```

```
PLA DEL TIPO 2
*
*Tecnologia CMOS (3 micrometros)- Louvain
*
.OPTIONS NOPAGE LIMPTS=2000 ABSTOL=10PA VNTOL=10UV RELTOL=.001
.OPTIONS ITL5=0 ITL4=20
.WIDTH OUT=80
.PROBE V(21),V(22),V(23),V(24),V(112),V(30),ID(MPS1),ID(MPS2),
+I(R1),I(VDD1),V(1),ID(M1212),ID(MPL1),V(12),V(312)
*
*Modelo MOS3 semiempirico de Vladimirescu & al.
*
.MODEL NMOS NMOS(LEVEL=3 VTO=0.947 GAMMA=0.2925 RSH=33.38 CJ=0.15E-3
+                TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
+                LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
+          ETA=0.221 KAPPA=0.212)
*
.MODEL PMOS PMOS(LEVEL=3 VTO=-0.773 GAMMA=0.7164 RSH=49.40 CJ=0.25E-3
+          TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
+          LD=0.8E-6 UO=217 VMAX=2.09E5 DELTA=0.0 THETA=0.087
+          ETA=0.234 KAPPA=5.995)
*
VDD1 1 0 5V
VDD2 300 0 5V
IREF1 2 0 20U
IREF2 200 0 20U
IIN1 0 51 PWL(0 0 200N 0 210N 20U 400N 20U 410N 40U 600N 40U
+610N 60U 800N 60U 810N 0 1000N 0 1010N 60U 1200N 60U)
.TRAN 3N 1200N
*
R1 51 61 1K
XLIT1 300 200 61 21 22 23 24 100 LITNOR
VC 100 0 0
V25 25 0 0
V26 26 0 0
V27 27 0 5
V28 28 0 0
*
* Transistores de salida y de generaci n de VREF:
*
MP0 2 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP00 200 200 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MPS1 30 30 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MPS2 31 31 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
*
*Cargas PMOS de las lineas producto:
MPL1  11 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL2  12 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL3  13 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL4  14 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL5  15 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL6  16 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL7  17 0 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
*
```

```
*Fuentes de corriente PMOS del plano TSUM:
*
MP1 311 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP2 312 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP3 313 2 1 1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP4 314 2 1 1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP5 315 2 1 1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP6 316 2 1 1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP7 317 2 1 1 PMOS L=6U W=27U PS=40U PD=40U AS=100P AD=100P
*
*Transistores NMOS entre los planos:
*
MPA1 311 111 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA2 312 112 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA3 313 113 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA4 314 114 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA5 315 115 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA6 316 116 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA7 317 117 0 0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*
*Transistores nmos del plano NOR:
*
M1123 11 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1126 11 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1127 11 27 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1128 11 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1221 12 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1223 12 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1224 12 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1225 12 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1226 12 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1228 12 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1321 13 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1322 13 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1324 13 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1325 13 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1326 13 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1421 14 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1422 14 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1424 14 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1428 14 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1521 15 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1522 15 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1525 15 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1528 15 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1621 16 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1624 16 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1625 16 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1626 16 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1627 16 27 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1722 17 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1724 17 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1725 17 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1726 17 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1728 17 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
```

```
*
*Inversores de separacion entre planos:
XINV1 1 11 111 INV
XINV2 1 12 112 INV
XINV3 1 13 113 INV
XINV4 1 14 114 INV
XINV5 1 15 115 INV
XINV6 1 16 116 INV
XINV7 1 17 117 INV
*
*
*Transistores nmos del plano TSUM:
*
M1111 311 311 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1212 312 312 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1313 313 313 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1414 314 314 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1515 315 315 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1616 316 316 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1717 317 317 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3011 30 311 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3012 30 312 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3014 30 314 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3111 31 311 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3113 31 313 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3115 31 315 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3116 31 316 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3117 31 317 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
*
*
*Capacidades parasitas de las lineas de literal, de producto
* y de salida (C(poli/subs)=.6E-4pf/µm², C(metal/subs)=.3E-4pf/µm²)
C21 21 0 .03P
C22 22 0 .03P
C23 23 0 .03P
C24 24 0 .03P
C25 25 0 .03P
C26 26 0 .03P
C27 27 0 .03P
C28 28 0 .03P
C11 11 0 .03P
C12 12 0 .03P
C13 13 0 .03P
C14 14 0 .03P
C15 15 0 .03P
C16 16 0 .03P
C17 17 0 .03P
C311 311 0 .02P
C312 312 0 .02P
C313 313 0 .02P
C314 314 0 .02P
C315 315 0 .02P
C316 316 0 .02P
C317 317 0 .02P
C30 30 0 .03P
C31 31 0 .03P
```

```
*
*****************************************************************
.SUBCKT INV   1    2     3
*             VDD  IN    OUT
MP1  3  2  1  1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN1  3  2  0  0 NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
.ENDS
*****************************************************************
.SUBCKT LITNOR  1    2    4    10    11    12    13    20
*               VDD VREF IIN V0    V1    V2    V3    VC
*Descripcion del circuito:
*
MP1  3  2  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MP2  6  2  1  1 PMOS L=6U W=13.5U PS=40U PD=40U AS=100P AD=100P
MP3  9  2  1  1 PMOS L=6U W=22.5U PS=40U PD=40U AS=100P AD=100P
MP4  5  3  1  1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP5  7  6  1  1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP6  8  9  1  1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP7  14 20 1  1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP8  10 5  14 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP9  15 7  1  1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP10 16 20 15 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP11 11 3  16 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP12 19 20 1  1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP13 13 9  19 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP14 17 8   1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP15 12 6  18 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP16 18 20 17 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
*
MN0  4  4  0  0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN1  3  4  0  0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN2  6  4  0  0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN3  9  4  0  0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN4  5  3  0  0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN5  7  6  0  0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN6  8  9  0  0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN7  10 20 0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN8  10 5  0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN9  11 20 0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN10 11 7  0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN11 11 3  0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN12 13 20 0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN13 13 9  0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN14 12 20 0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN15 12  6 0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN16 12  8 0  0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*Capacidades de las interconexiones:
*
C20 20 0 .116P
C21 20 0 .086P
C22 20 0 .073P
C23 20 0 .079P
C2  2  0 .063P
C10 10 0 .083P
C11 11 0 .122P
```

```
C12 12 0 .138P
C13 13 0 .150P
C3 3 0 .097P
C5 5 0 .084P
C7 7 0 .090P
C6 6 0 .063P
C9 9 0 .040P
C8 8 0 .028P
C4 4 0 .024P
*
.ENDS
********************************************************************************
.END
```

```
PLA DEL TIPO 5
*
*Tecnologia CMOS (3 micrometros)- Louvain
*
.OPTIONS NOPAGE LIMPTS=2000 ABSTOL=10PA VNTOL=10UV RELTOL=.005
.OPTIONS ITL5=0 ITL4=20
.WIDTH OUT=80
.PROBE V(21),V(22),V(23),V(24),V(112),V(30),ID(MPS1),ID(MPS2),
+I(R1),I(VDD1),V(1),ID(M1212),ID(MPL2),V(12),V(212),V(312),
.PROBE V(100),V(3),V(412),V(13),V(14),V(15),V(16)
.PROBE V(17),V(113),V(114),V(115),V(116),V(117),V(313),V(314)
.PROBE V(315),V(316),V(317)
*
*Modelo MOS3 semiemp rico de Vladimirescu & al.
*
.MODEL NMOS NMOS(LEVEL=3 VTO=0.947 GAMMA=0.2925 RSH=33.38 CJ=0.15E-3
+                TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
+                LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
+          ETA=0.221 KAPPA=0.212)
*
.MODEL PMOS PMOS(LEVEL=3 VTO=-0.773 GAMMA=0.7164 RSH=49.40 CJ=0.25E-3
+          TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
+          LD=0.8E-6 UO=217 VMAX=2.09E5 DELTA=0.0 THETA=0.087
+          ETA=0.234 KAPPA=5.995)
*
VDD1 1 0 5V
VDD2 300 0 5V
IREF1 2 0 20U
IREF2 200 0 20U
IIN1 0 51 PWL(0 0 200N 0 210N 20U 400N 20U 410N 40U 600N 40U
+610N 60U 800N 60U 810N 0 1000N 0 1010N 60U 1200N 60U)
.TRAN 2N 1200N
*
R1 51 61 1K
XLIT1 300 200 61 21 22 23 24 100 LITNOR
XINV0 1 100 3 INV
CINV 3 0 .03P
VC1 100 0 PWL(0 0 10N 5 60N 5 70N 0 200N 0 210N 5 260N 5
+270N 0 400N 0 410N 5 460N 5 470N 0 600N 0 610N 5 660N 5 670N 0
+800N 0 810N 5 860N 5 870N 0 1000N 0 1010N 5 1060N 5 1070N 0
+1200N 0)
V25 25 0 0
V26 26 0 0
R27 3 27 0.1
V28 28 0 0
*
* Transistores de salida y de generaci n de VREF:
*
MP0 2 2 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP00 200 200 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MPS1 30 30 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MPS2 31 31 300 300 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
*
*Cargas PMOS de las lineas producto:
```

```
MPL1   11  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL2   12  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL3   13  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL4   14  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL5   15  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL6   16  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MPL7   17  3  1  1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
*
*Fuentes de corriente PMOS del plano TSUM:
*
MP1 211  2  1  1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP2 212  2  1  1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP3 213  2  1  1 PMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MP4 214  2  1  1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP5 215  2  1  1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP6 216  2  1  1 PMOS L=6U W=18U PS=40U PD=40U AS=100P AD=100P
MP7 217  2  1  1 PMOS L=6U W=27U PS=40U PD=40U AS=100P AD=100P
*
*Transistores de paso controlados por phi:
*
MPA1 411  3  111  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA2 412  3  112  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA3 413  3  113  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA4 414  3  114  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA5 415  3  115  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA6 416  3  116  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA8 411  100  111  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA9 412  100  112  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA10 413  100  113  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA11 414  100  114  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA12 415  100  115  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA13 416  100  116  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA14 417  100  117  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MPA7 417  3  117  0 NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*Inversores de las lineas producto:
*
XINV1  1  11  411 INV
XINV2  1  12  412 INV
XINV3  1  13  413 INV
XINV4  1  14  414 INV
XINV5  1  15  415 INV
XINV6  1  16  416 INV
XINV7  1  17  417 INV
*
*
*Transistores PMOS conmutadores de las fuentes de corriente:
*
MGND1 311  111  211  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MGND2 312  112  212  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MGND3 313  113  213  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MGND4 314  114  214  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MGND5 315  115  215  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MGND6 316  116  216  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MGND7 317  117  217  1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
```

```
*
*
*Transistores nmos del plano NOR:
*
M1123 11 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1126 11 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1127 11 27 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1128 11 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1221 12 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1223 12 23 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1224 12 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1225 12 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1226 12 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1228 12 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1321 13 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1322 13 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1324 13 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1325 13 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1326 13 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1421 14 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1422 14 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1424 14 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1428 14 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1521 15 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1522 15 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1525 15 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1528 15 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1621 16 21 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1624 16 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1625 16 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1626 16 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1627 16 27 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1722 17 22 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1724 17 24 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1725 17 25 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1726 17 26 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
M1728 17 28 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*
*Transistores nmos del plano TSUM:
*
M1111 311 311 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1212 312 312 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1313 313 313 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1414 314 314 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1515 315 315 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1616 316 316 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M1717 317 317 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3011 30 311 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3012 30 312 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3014 30 314 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3111 31 311 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3113 31 313 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3115 31 315 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3116 31 316 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
M3117 31 317 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
```

```
*
*
*Capacidades parasitas de las lineas de literal, de producto
* y de salida (C(poli/subs)=.6E-4pf/µm² ; C(metal/subs)=.3E-4pf/µm² ):
*
C21 21 0 .03P
C22 22 0 .03P
C23 23 0 .03P
C24 24 0 .03P
C25 25 0 .03P
C26 26 0 .03P
C27 27 0 .03P
C28 28 0 .03P
C11 11 0 .03P
C12 12 0 .03P
C13 13 0 .03P
C14 14 0 .03P
C15 15 0 .03P
C16 16 0 .03P
C17 17 0 .03P
C311 311 0 .02P
C312 312 0 .02P
C313 313 0 .02P
C314 314 0 .02P
C315 315 0 .02P
C316 316 0 .02P
C317 317 0 .02P
C30 30 0 .03P
C31 31 0 .03P
*
**********************************************************************
.SUBCKT INV  1   2    3
*           VDD IN   OUT
MP1 3 2 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN1 3 2 0 0 NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
.ENDS
**********************************************************************
.SUBCKT LITNOR  1   2   4   10   11   12   13   20
*              VDD VREF IIN V0   V1   V2   V3   VC
*Descripcion del circuito:
*
MP1 3 2 1 1 PMOS L=6U W=4.5U PS=40U PD=40U AS=100P AD=100P
MP2 6 2 1 1 PMOS L=6U W=13.5U PS=40U PD=40U AS=100P AD=100P
MP3 9 2 1 1 PMOS L=6U W=22.5U PS=40U PD=40U AS=100P AD=100P
MP4 5 3 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP5 7 6 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP6 8 9 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MP7 14 20 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP8 10 5 14 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP9 15 7 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP10 16 20 15 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP11 11 3 16 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP12 19 20 1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP13 13 9 19 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP14 17 8  1 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
MP15 12 6 18 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
```

```
MP16 18 20 17 1 PMOS L=3U W=18U PS=40U PD=40U AS=100P AD=100P
*
MN0 4 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN1 3 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN2 6 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN3 9 4 0 0  NMOS L=6U W=6U PS=40U PD=40U AS=100P AD=100P
MN4 5 3 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN5 7 6 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN6 8 9 0 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN7 10 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN8 10 5 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN9 11 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN10 11 7 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN11 11 3 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN12 13 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN13 13 9 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN14 12 20 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN15 12  6 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MN16 12  8 0 0  NMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
*
*Capacidades de las interconexiones:
*
C20 20 0 .116P
C21 20 0 .086P
C22 20 0 .073P
C23 20 0 .079P
C2 2 0 .063P
C10 10 0 .083P
C11 11 0 .122P
C12 12 0 .138P
C13 13 0 .150P
C3 3 0 .097P
C5 5 0 .084P
C7 7 0 .090P
C6 6 0 .063P
C9 9 0 .040P
C8 8 0 .028P
C4 4 0 .024P
*
.ENDS
********************************************************************************
.END
```

```
MEMOCEL: Celda de memoria PMOS mv-corriente
*
*Tecnolog a CMOS (3 micrometros)- Louvain
*
.OPTIONS NOPAGE LIMPTS=2000
.WIDTH OUT=80
.PROBE
*
*Modelo MOS3 semiempirico de Vladimirescu & al.
*
.MODEL NMOS NMOS(LEVEL=3 VTO=0.947 GAMMA=0.2925 RSH=33.38 CJ=0.15E-3
+                TOX=550E-10 NSUB=1E15 NFS=0.0 TPG=1 XJ=0.25E-6
+                LD=0.75E-6 UO=651 VMAX=1.69E5 DELTA=0.0 THETA=0.070
+          ETA=0.221 KAPPA=0.212)
*
.MODEL PMOS PMOS(LEVEL=3 VTO=-0.773 GAMMA=0.7164 RSH=49.40 CJ=0.25E-3
+          TOX=550E-10 NSUB=6E15 NFS=0.0 TPG=-1 XJ=0.47E-6
+          LD=0.8E-6 UO=217 VMAX=2.09E5 DELTA=0.0 THETA=0.087
+          ETA=0.234 KAPPA=5.995)
*
VDD 1 0 5V
MP1 2 2 1 1 PMOS L=6U W=7.5U PS=40U PD=40U AS=100P AD=100P
MP3 2 10 4 1 PMOS L=3U W=6U PS=40U PD=40U AS=100P AD=100P
MP2 5  4 1 1 PMOS L=6U W=7.5U PS=40U PD=40U AS=100P AD=100P
MN1 3 3 0 0  NMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MN2 2 3 0 0  NMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
MN3 2 20 4 0  NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
MN4 5 5 0 0  NMOS L=6U W=9U PS=40U PD=40U AS=100P AD=100P
XINV1 1 40 41 INV
XINV2 1 41 20 INV
XINV3 1 40 10 INV
C2 2 0 .01P
C4 4 0 .01P
R 30 3 1K
IIN 0 30 PWL(0 0 100N 0 110N 10U 200N 10U 210N 20U 300N 20U
+310N 30U 400N 30U 410N 40U 500N 40U 510N 50U 600N 50U 610N
+60U 700N 60U 710N 0 800N 0 810N 60U 1000N 60U)
*
VPHI 40 0 PWL(0 0 60N 0 70N 5 90N 5 100N 0 160N 0 170N 5 190N 5
+200N 0 260N 0 270N 5 290N 5 300N 0 360N 0 370N 5 390N 5 400N 0
+460N 0 470N 5 490N 5 500N 0 560N 0 570N 5 590N 5 600N 0 660N 0
+670N 5 690N 5 700N 0 760N 0 770N 5 790N 5 800N 0 860N 0 870N 5
+890N 5 900N 0 1000N 0)
*
**********************************************************************
.SUBCKT INV  1   2   3
MP1 3 2 1 1 PMOS L=3U W=9U PS=40U PD=40U AS=100P AD=100P
MN1 3 2 0 0 NMOS L=3U W=4.5U PS=40U PD=40U AS=100P AD=100P
.ENDS
**********************************************************************
*
.TRAN 1N 1000N
*.DC IIN 0 40U .5U
.END
```