

INTRODUCCIÓN AL DISEÑO DE CIRCUITOS INTEGRADOS CON *TEDMOS*

M. Perez-Castellanos, V. Rodellar, C. Gonzalo, V. Nieto y P. Gómez-Vilda.

Universidad Politécnica de Madrid
Facultad de Informática
Departamento de Arquitectura y Tecnología de Sistemas Informáticos
Campus de Montegancedo, Boadilla del Monte, 28660 Madrid

Tfno: (91) 3367382
Fax: (91) 3367412
e-mail: marga@fi.upm.es

RESUMEN

Este trabajo presenta un conjunto de prácticas de Introducción al Diseño de Circuitos Integrados mediante herramientas CAD de tipo educacional. La propuesta va dirigida a alumnos de segundo curso de Licenciatura en Informática, con el fin de realizar una primera aproximación a los aspectos tecnológicos y de implementación de la Arquitectura de Computadores.

1. INTRODUCCIÓN

El diseño y realización física de Arquitecturas de Computadores es un tema multidisciplinar que abarca tres dominios muy bien identificados y estructurados según Gajski [1]. El primero de los dominios llamado de *especificación*, trata de definir a distinto nivel de detalle aspectos generales del problema tales como algoritmos a realizar, funciones, instrucciones, etc., el segundo conocido como *estructural*, aborda también de forma jerárquica qué sistemas, procesadores, unidades funcionales, puertas, etc., podrían realizar los distintos niveles del dominio de especificación. Y el último de los dominios, conocido como de *implementación*, concreta cual sería la realización física de cada uno de los niveles del dominio estructural, contemplando aspectos tales como, particiones, módulos, macrocélulas, geometría de dispositivos, etc. La metodología de diseño de Circuitos Integrados (C. I.) utiliza la aproximación *top-down* en el dominio *estructural* y *bottom-up* en el de *implementación*.

Los dominios de *especificación* y *estructural*, han sido abordados con cierto nivel de profundidad en los currícula de la mayoría de los planes de estudio de las Licenciaturas y Diplomaturas en Informática, mientras que el dominio tecnológico o de implementación tradicionalmente no ha sido abordado. Las razones de ello, quizá hayan residido en la no

disponibilidad de productos educativos de bajo coste y fácil manejo unido a la escasa presencia de conceptos formativos relacionados con aspectos más tecnológicos. No obstante, la formación de un alumno presenta en estos casos importantes carencias que en este momento deberían subsanarse dada la presencia de medios disponibles y las perspectivas hacia donde parece apuntar el futuro en este área concreta. Citando textualmente a Hennessy, *"la interacción entre la arquitectura y la tecnología de circuitos integrados es compleja y bidireccional, ya que las características de las diferentes tecnologías afectan a las decisiones de los arquitectos de computadores influyendo en las prestaciones, coste y otros atributos del sistema. Asimismo los nuevos avances en el planteamiento de nuevas arquitecturas implican un nuevo desafío en la búsqueda de nuevas tecnologías. En el diseño de un computador hoy, las consideraciones más importantes a tener en cuenta en términos generales, son prestaciones y coste, y como medidas de segundo orden, las cuales pueden variar en importancia, pueden considerarse, tolerancia a fallos, potencia, tamaño, ruido, etc..."* [2]. Por tanto, parece necesario incluir de manera paulatina y estructurada aspectos tecnológicos relacionados con el diseño de computadores dentro de las materias formativas del primer ciclo. Ello implica un número elevado de alumnos y por tanto la disponibilidad de un amplio conjunto de recursos materiales. Por otro lado, hay que tener en cuenta los conocimientos que los alumnos poseen en este ciclo de su formación acerca del manejo de equipos informáticos, que en la mayoría de los casos se limita al entorno de Computadores Personales, recurso que muchos de ellos incluso disponen a nivel particular.

Existen varias herramientas de corte educacional que sirven para la realización de C. I. No obstante, la mayoría de las conocidas y accesibles están sujetas a licencia de uso, lo cual imposibilita proporcionar una copia individual a aquellos alumnos que la demanden por disponer de recursos propios, en este sentido, podemos citar el MSK de Etienne Sicard [3,4] de fácil uso y muy pedagógico. Una herramienta no sujeta al uso de licencia es el Turbo Editor de Circuitos Integrados MOS (TEDMOS) [5], realizado por el Núcleo de Computación de la Universidad Federal de Rio de Janeiro y subvencionado por la Unesco. La versión original del programa tiene las pantallas de dialogo escritas en Portugués, y con objeto de facilitar el manejo del mismo, se ha realizado la correspondiente traducción al castellano [6]. Su ejecución se realiza desde DOS, y su mayor inconveniente reside en la gran pobreza que presentan sus interfaces de usuario, lo cual hace su utilización un tanto incomoda y farragosa. No obstante, es previsible que en breve esté disponible una nueva versión en entorno Windows.

La introducción al diseño de circuitos integrados con TEDMOS se realiza con la siguiente estructura, en primer lugar, se propone a los alumnos una práctica sobre el diseño de un inversor CMOS, con objeto de mostrar todas y cada una de las posibilidades de la herramienta CAD, para a continuación pasar al diseño de una puerta NOR. En segundo lugar, se propone realizar una Unidad Aritmético Lógica tipo Mead y Conway [7], lo cual requiere de la planificación cuidadosa del *lay-out*, teniendo en cuenta aspectos del proceso de fabricación a utilizar, pozo p o pozo n, diseño estructurado, modularidad, regularidad, etc.

2. TOMA DE CONTACTO CON LA HERRAMIENTA CAD.

La toma de contacto con la herramienta de diseño se realiza a través de la construcción de uno de los circuitos más simples que se pueden construir en CMOS, como es el caso del inversor.

Las únicas restricciones del diseño es que ambos transistores deben tener inicialmente la misma relación de aspecto y deben ser de dimensiones mínimas de acuerdo con las reglas de diseño que utiliza el TEDMOS.

Se parte de una especificación del circuito en diagrama de barras y a partir de él, se solicita que los alumnos realicen sobre el papel un esquema borrador del mismo, bien mediante código de patrones o bien de colores, con las dimensiones mínimas que ha de tener cada una de las capas que componen el *lay-out*, por ejemplo tamaño de los metales, policristalinos, zonas activas, separación entre elementos de una misma capa, pozo, etc.. Esta tarea previa al diseño sobre la máquina, es de gran ayuda para hacerse una primera idea acerca de como ubicar los distintos elementos del *lay-out* y no tener que realizar demasiadas modificaciones sobre un posible diseño inicial realizado sin método con el editor de células, ya que el manejo de las facilidades de edición no es demasiado flexible.

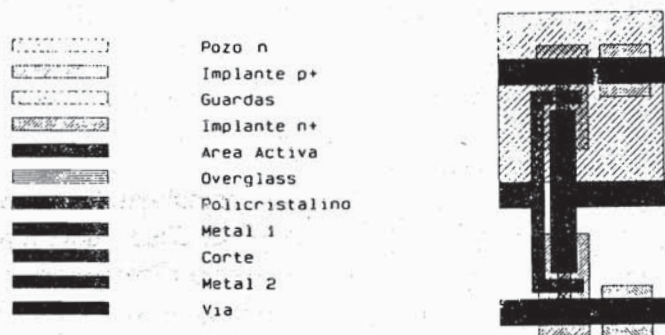
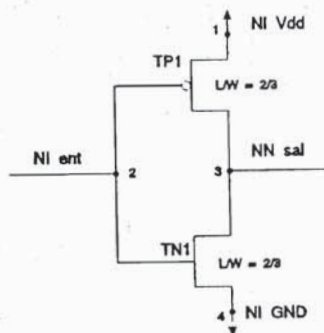


Figura 1. *Lay-out* del circuito inversor realizado con el editor de células y código de patrones asociado.

Una vez realizado el *lay-out* (Fig. 1) y verificadas las reglas de diseño, se ha de proceder a la extracción del circuito. A partir del fichero que contiene esta última información (Fig.2a), la cual básicamente consiste en tipo de transistor, número asignado al transistor, número de identificación de los terminales de puerta, fuente y drenador de los transistores, longitud y anchura de los mismos y nudos de entrada, salida e internos, debe reconstruirse el esquema circuital del *lay-out*.

TP	1	2	1	3	2	3		
TN	2	2	4	3	2	3		
NIvdd		1	19	0	92	18	0	0
NIent		2	0	100	0	0	0	0
NNsal		3	38	0	100	36	0	0
NIgnd		4	19	0	92	18	0	0

(a)



(b)

Figura 2. Contenido del fichero resultante de la extracción del circuito (a). Reconstrucción circuital del mismo (b).

El siguiente paso a dar, consiste en comprobar a nivel lógico la funcionalidad del diseño. El simulador lógico de la herramienta, considera a los transistores como interruptores ideales que admiten tres estados posibles, abierto, cerrado e indeterminado. La preparación de los valores de las variables de entrada al simulador puede realizarse *off line*, mediante la confección de un fichero de estímulos de entrada, o bien en modo *on line* aplicando una serie de comandos de forma interactiva cuando el simulador se encuentra activado. La evolución de las distintas señales puede visualizarse de forma gráfica a través de la invocación de un comando especial que convierte la pantalla del sistema informático en una pantalla de un modesto analizador lógico sobre el cual es posible visualizar simultáneamente hasta 10 canales. La Fig. 3 muestra la simulación lógica de un inversor.

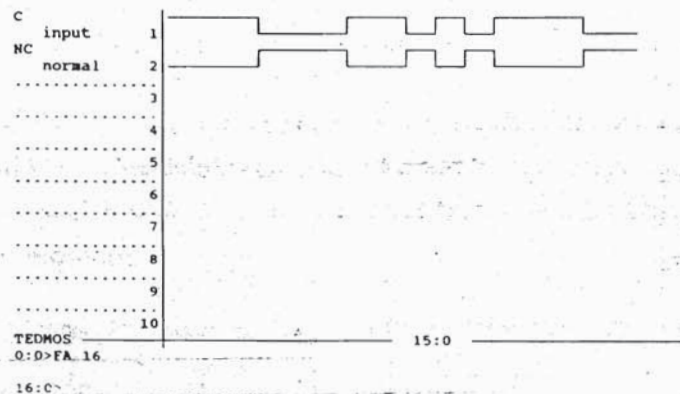


Figura 3. Visualización en modo gráfico de la entrada (C) y la salida (NC) de un inversor a nivel lógico.

Los apartados descritos anteriormente resultan bastante atractivos y no ofrecen mayores dificultades a los alumnos, las dificultades surgen al abordar la simulación eléctrica. La herramienta dispone de un preprocesador eléctrico que genera como salida sentencias directamente procesables por un generador eléctrico tipo SPICE; dicha facilidad por un lado obvia conocer esta última herramienta u otra de tipo similar, y por otro no exige poseer unos conocimientos profundos sobre electrónica y dispositivos. La entrada a dicho preprocesador se realiza a través de un menú relativamente simple que permite seleccionar cuatro modelos distintos de transistor, las capacidades de carga, las fuentes de tensión, los nudos a visualizar, el tiempo y paso de simulación, las curvas características de los transistores y curvas de transferencia. La principal limitación del conjunto preprocesador-procesador eléctrico radica en la capacidad de representación gráfica, ya que aunque es posible seleccionar en el preprocesador simultáneamente más de una de las opciones disponibles, por ejemplo, curvas características de los transistores p y n y relación de transferencia entre la entrada y la salida, desde el procesador eléctrico solamente es posible representar una de las gráficas cada vez, y en concreto a la que se tiene acceso es a la primera que figura en el fichero de salida del procesador eléctrico. Esta limitación de la interfaz gráfica obliga a seguir la secuencia preprocesador-procesador eléctrico tantas veces como información de tipo gráfico queramos obtener.

La salida del procesador eléctrico es un fichero que incluye información en forma de comentarios acerca de cuál es el significado y a qué está referido el conjunto de números

organizado en forma de matriz que sigue a continuación. La Fig. 4 muestra un fragmento del fichero de salida correspondiente a las curvas características de un transistor tipo n para el modelo lento-lento y su correspondiente representación gráfica asociada. La V_{DS} está comprendida entre 0 y 5V. y los valores de la V_{GS} entre 0 y 3V. ambas representadas en incrementos de 0,5V. Esta curva, así como la del transistor tipo p, deben ser obtenidas por los alumnos y como ya se ha mencionado anteriormente, han de seleccionarse por separado en el preprocesador eléctrico.

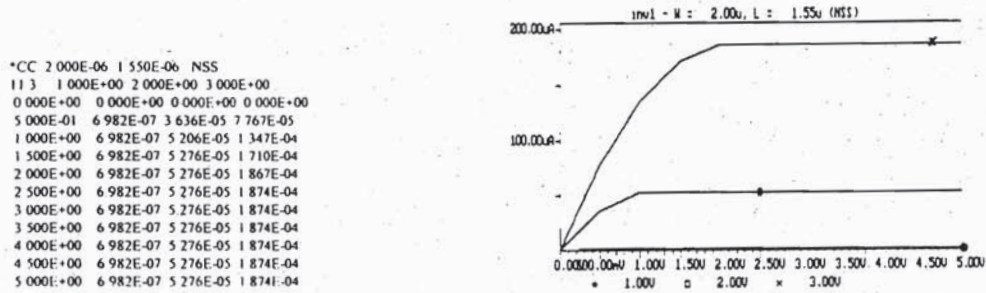


Figura 4. Valores de las distintas coordenadas y curvas características de un transistor tipo n

A partir de estas curvas características, y conocida la tensión umbral del transistor que puede obtenerse directamente del fichero de salida que genera el preprocesador eléctrico, se pide que los alumnos calculen de forma gráfica la zona de separación entre el funcionamiento en zona lineal y de saturación del transistor.

Una vez caracterizados a nivel eléctrico los transistores p y n, se pasa a obtener la relación de transferencia en continua entre la entrada y la salida (Fig. 5).

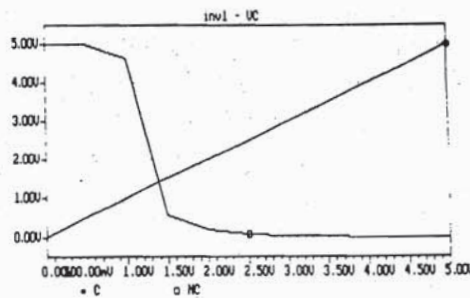


Figura 5. Curva de transferencia en continua de un inversor

Sobre la gráfica anterior, los alumnos deben extraer los valores de las tensiones umbrales de los transistores p y n y observar si los resultados obtenidos son coherentes con los anteriormente identificados. Por otro lado, deben asimismo identificar el estado de

polarización (corte, óhmica o saturación) de cada transistor para distintos valores de la entrada entre 0 y 5V. Esta última exigencia les obliga a interpretar en detalle el comportamiento eléctrico del circuito inversor, a distinguir claramente entre los modelos de transistor utilizados en la simulación lógica y eléctrica y a reflexionar sobre el tipo de información que proporcionan y cuál es la utilidad de cada una de las simulaciones.

El último de los puntos a tratar en el desarrollo de esta parte introductoria, es el estudio del comportamiento en régimen transitorio del circuito. El preprocesador eléctrico contiene la facilidad de generar ondas cuadradas de amplitud, periodo, ciclo de trabajo y tiempos de subida y bajada variables, seleccionables a través de un menú. Los alumnos deben modificar las características de estas señales cuadradas así como las capacidades de carga y observar la influencia de las características temporales de las mismas en la respuesta del circuito, a través de la medida de tiempos de subida, bajada y retardo. Asimismo deben interpretar cuál de los dos transistores es más rápido en realizar la conmutación entre los estados, y relacionar este hecho con la relación de aspecto L/W de los mismos debiendo modificar el *lay-out* resultante del inversor para que los dos respondan con la misma rapidez. Este último aspecto pone en evidencia la influencia decisiva y determinante de las dimensiones longitud/anchura adoptadas para el canal sobre las características eléctricas del transistor resultante (resistencia de canal, capacidad asociada).

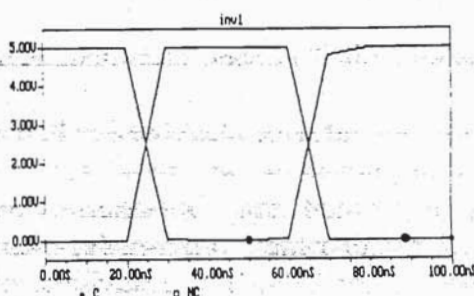


Figura 6. Respuesta transitoria del inversor CMOS

Con objeto de poner en práctica los conocimientos que se han ido adquiriendo de forma paulatina y tutorada, esta parte introductoria concluye con la realización de forma más independiente de una puerta NOR de dos entradas, en la que se completan aspectos del diseño no estudiados en la parte anterior, tales como la interpretación del fichero CIF, y la elección de las señales de test para las simulaciones lógica y eléctrica.

3. DISEÑO DE UNA UNIDAD ARITMÉTICO-LÓGICA

La segunda práctica plantea un ejercicio más complejo que los propuestos en la primera, con el fin de que el alumno se introduzca en los problemas de diseño de Circuitos Integrados, realizando un sistema regular, modular y estructurado, partiendo de células elementales. Concretamente se propone sintetizar una de las unidades funcionales más típicas de un computador, la Unidad Aritmético Lógica (UAL). Con esta segunda propuesta los alumnos se

centran en aspectos propios del diseño VLSI, dado que ya cuentan con una cierta destreza y habilidad en el manejo de la herramienta, objetivo de la primera práctica.

Existen muchas realizaciones de la UAL de un computador, pero debido a su simplicidad, regularidad y modularidad, se ha elegido el modelo de Mead y Conway [7] en el que hemos sustituido la etapa precargada de generación del acarreo, por una etapa estática de multiplexores [8]. En esta unidad se puede realizar cualquier operación sencilla, tanto aritmética como lógica, entre dos operandos codificados en formato binario natural o en complemento a dos, utilizando tantos módulos elementales como componentes tengan los vectores de datos. Un esquema de un módulo de bit de la UAL se incluye en la Fig. 7; en él pueden verse cuatro unidades funcionales construidas con multiplexores de cuatro entradas de datos, que para el tercer multiplexor están previamente fijadas, y corresponden a las señales P_j , K_j y R_j ($j=0,1,2,3$), para el primero segundo y cuarto, respectivamente. Estas últimas señales se emplearán para programar la función que van a realizar los multiplexores de cada columna, sobre los bits a_i b_i que se introducen en cada fila de la UAL a través de las entradas de control de cada unidad funcional. Las entradas a cada módulo serán los correspondientes bits a_i y b_i de los datos de entrada y el bit de acarreo c_{i-1} generado en el módulo anterior, teniendo dos salidas, el resultado s_i de operar los bits de entrada y el bit de acarreo c_i que se transmite al módulo siguiente, siempre que la operación así lo requiera.

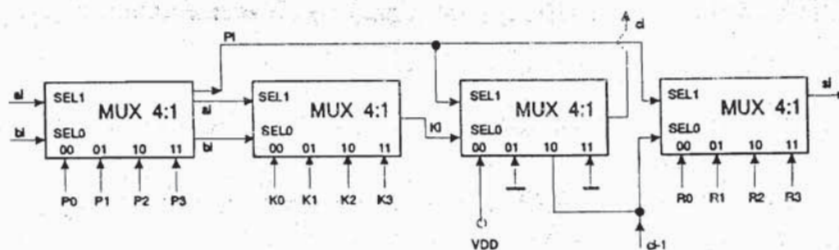


Figura 7. Etapa de un bit de la Unidad Aritmético-Lógica

La primera de las cuatro columnas de multiplexores de cada módulo de bit genera la función P_i de propagación del acarreo, la segunda columna genera la función K_i destrucción de acarreo, la tercera columna transmite el acarreo de salida y la cuarta columna genera el resultado.

Una vez planteado el problema, es decir definida la estructura y funcionalidad de la Unidad Aritmético-Lógica, el paso siguiente es abordar el diseño de la estructura. Se plantea el trabajo a los alumnos siguiendo una estructura *bottom-up*; en primer lugar se ha de diseñar la célula básica, ensamblando estas células se construye una unidad funcional y conectando adecuadamente varias unidades funcionales se construye un módulo de bit de la unidad. Para construir una UAL con un tamaño predeterminado, se procede a ensamblar adecuadamente tantos módulos de bit, como número de bits tengan los vectores de datos a procesar.

En nuestro caso particular al alumno se le sugiere la utilización de multiplexores 2:1 como células elementales. El diseño de estas células debe realizarse mediante el uso intensivo de puertas de transmisión CMOS, partiendo de la especificación mediante diagrama de barras de las mismas. En la edición de la célula el alumno ha de prestar especial atención a la

localización del pozo, contactos de pozo y sustrato y en general tener presente las diferentes características impuestas por el proceso de fabricación CMOS; el siguiente paso será, una vez realizada la extracción de la célula, efectuar las simulaciones lógica y eléctrica de la misma. La conexión de los tres multiplexores 2:1 conforma la unidad funcional (Fig. 8), en nuestro caso un multiplexor 4:1 y la especificación en diagrama de barras será el primer paso en el diseño de esta nueva estructura. En ella deben destacarse las conexiones entre las tres células básicas, así como las señales de masa, alimentación, selección y salidas que han de distribuirse por las diferentes rutas para conformar la unidad completa. La Fig. 9 recoge el citado diagrama de barras.

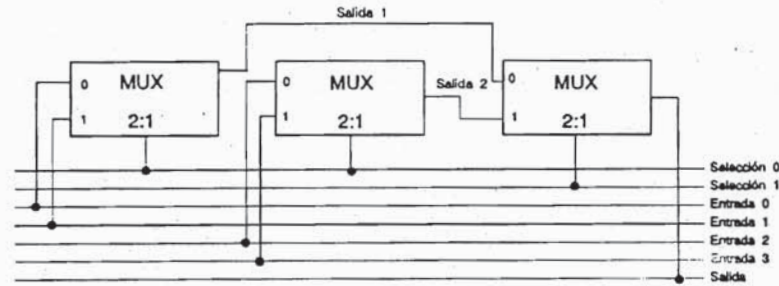


Figura 8. Construcción de un multiplexor 4:1 partiendo de células elementales

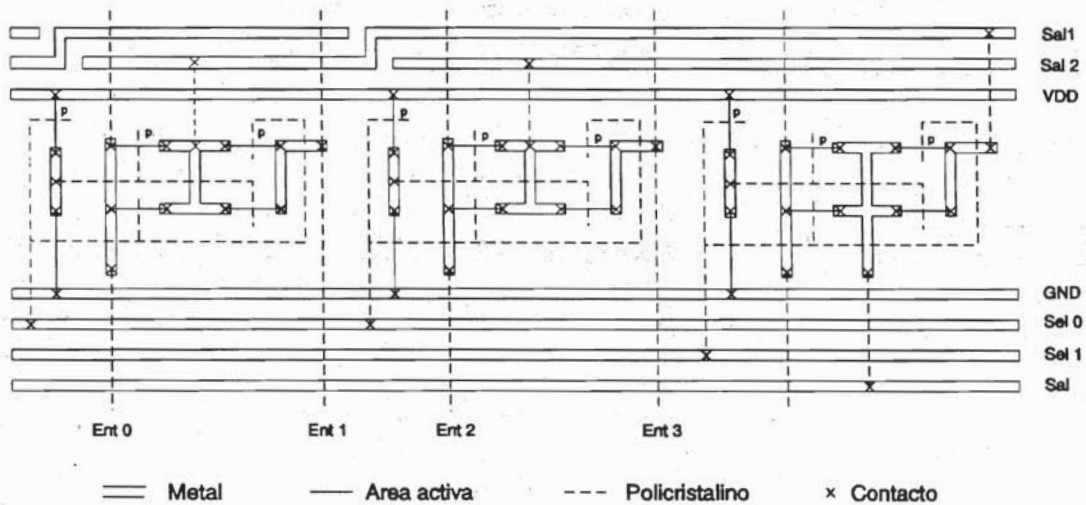


Figura 9. Diagrama de barras de una unidad funcional

La conexión adecuada de cuatro unidades funcionales constituirá un módulo de bit de la Unidad Aritmético-Lógica. Las entradas E_j ($j=0,1,2,3$) de cada una de las unidades funcionales de la Fig. 9, tendrán que ser personalizadas para cada uno de los multiplexores 2:1 reflejados en la Fig. 7; así, la Entrada E_0 de cada unidad funcional corresponderá con las entradas de menor peso P_0 , K_0 , I y R_0 de los citados multiplexores. De forma similar se establecerán las adecuadas correspondencias con el resto de las señales implicadas.

La construcción de una UAL de una dimensión predeterminada precisa de la conexión de un número de módulos de bit prefijado por la dimensión de los vectores de datos. Para abordar el diseño de esta estructura, se indica a los alumnos unas ciertas pautas sobre la distribución de los diferentes módulos, de los canales de ruteo de las señales de alimentación, masa, entrada, selección y acarrees, etc., dejando los detalles de *grano fino* del diseño a ser determinados por ellos. Un diagrama de bloques para una estructura de n bits está recogido en la Fig. 10.

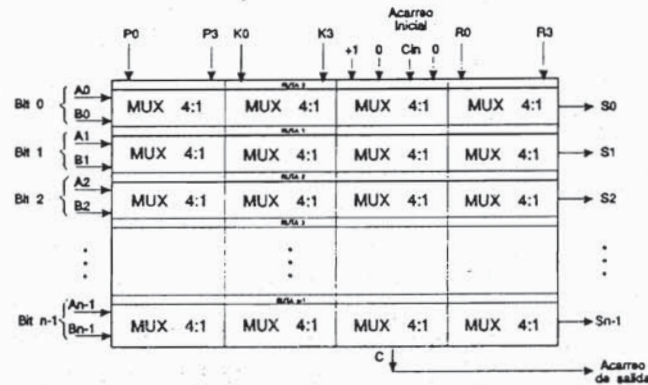


Figura 10. Estructura de una UAL de n bits

4. METODOLOGÍA DE IMPARTICIÓN

La práctica introductoria se desarrolla inmediatamente despues de haber tratado desde el punto de vista teórico tanto los aspectos funcionales como los tecnológicos de los dispositivos básicos. A los alumnos se les imparte un cursillo introductorio sobre las prestaciones y comandos básicos del editor TEDMOS, de dos horas de duración en el aula con transparencias de apoyo y una sesión de demostración de una hora sobre la máquina. Asimismo, se les proporciona una copia del manual de usuario y una copia de la herramienta CAD a todos aquellos que así lo requieran. Los alumnos deberán dedicar una media de unas ocho horas de trabajo personal, las cuales incluyen tanto el tiempo dedicado a la realización del trabajo sobre máquina, como a la redacción de una memoria que debe contener la respuesta a una serie de pequeños ejercicios que se han prefijado en un guión de la práctica.

Para abordar la segunda práctica, los alumnos cuentan con una cierta destreza en el manejo de la herramienta y conocen a nivel muy elemental los aspectos básicos del diseño de Circuitos Integrados. Con objeto de asegurarse de que todos los alumnos han estudiado los mismos contenidos, la primera práctica se desarrolla de forma guiada por el tutor, sin embargo para abordar la segunda, aunque se dan unas ciertas pautas para su desarrollo, se deja al alumno la libertad de buscar distintas soluciones de la propuesta con respecto a la filosofía de diseño, y tambien es trabajo totalmente personal la organización y estructuración de la memoria final. La carga promedio a trabajo, se estima en unas seis horas aproximadamente, incluyendo la redacción de la memoria final.

5. CONCLUSIONES

El diseño de circuitos integrados es una disciplina que parece razonable se encuentre contenida en el perfil de un Licenciado en Informática dada su estrecha relación e

implicaciones en la Arquitectura de Computadores. El trabajo aquí presentado supone una primera toma de contacto con el manejo de herramientas CAD para diseño de circuitos integrados, así como con la filosofía de diseño VLSI. No obstante, dado el carácter introductorio de la misma no ha sido posible abordar desde un punto de vista metodológico aspectos más complejos tales como estructuración de planos de planta, estrategias de ruteado de señales, rutas de datos, pines, etc., que se abordarán y completarán en asignaturas de cursos posteriores.

En general, podemos concluir que a los alumnos les motiva la realización de este tipo de prácticas, porque por un lado contrastan con los aspectos teóricos de la asignatura y por otro se desarrollan en un entorno muy atractivo para ellos. En el curso actual, han realizado estas prácticas aproximadamente unos cuatrocientos alumnos y prácticamente la totalidad de este elevado número ha demostrado alcanzar el nivel de suficiencia exigido en las mismas. La experiencia ha resultado tan motivante para algunos de ellos, que incluso han mostrado interés en colaborar en un futuro próximo en algunos trabajos relacionados con el diseño VLSI que actualmente se están llevando a cabo en el Departamento.

6. BIBLIOGRAFÍA

- [1] D. D. Gajski and R. H. Kuhn. "New VLSI Tools". *Computer*. Vol 16, nº 12, pp 11-14. 1983.
- [2] J. L. Hennessy. "Computer Technology and Architecture: An Evolving Interaction". *Computer*. pp 18-29. September, 1991.
- [3] E. Sicard, A. Rubio and K. Kozo Kinashita. "A VLSI Design System for Teaching Introduction to Microelectronics". *IEEE Transactions on Education*. Vol 35, nº 4, pp 311-320. November, 1992.
- [4] E. Sicard. "Introduction to Microelectronics". *User's Manual for IBM PC. Version 4.0.*, Software for Engineering Students, A project of the American Society for Engineering Education, 1992.
- [5] M. Anido and C. E. T. Oliveira. "A CAD System for Teaching the Design of VLSI Circuits - Status and Evolution". *Actas de PANEL '92, XVIII Conferencia Latinoamericana de Informática*, pp 70-78. Septiembre, 1992.
- [6] "TEDMOS IV. Turbo editor para Circuitos Integrados. Manual de Operación". Traducción al español por Pedro Gómez Vilda, *Publicación interna del Departamento de Arquitectura y Tecnología de Sistemas Informáticos, U. P. M.* 1992.
- [7] C. Mead and L. Conway. "Introduction to VLSI Systems". Addison-Wesley Publishing Company, 1980.
- [8] V. Rodellar, M. Hermida, M. M. Pérez-Castellanos, E. A. Castillo y P. Gómez-Vilda. "Fundamentos del Material Informático. Prácticas de laboratorio". Paraninfo, 1991.