

Curso de Laboratorio Junior para el estudio del procesador

Julio Sahuquillo Borrás, Marina Alonso Díaz, Ana Pont Sanjuan, María E. Gómez Requena

Departamento de Informática de Sistemas y Computadores
Escuela U. de Informática –Universidad Politécnica de Valencia–

E_mail: {jsahuqui, malonso, apont, megomez} @ disca.upv.es

Resumen

El estudio del procesador en los actuales planes de estudio de las carreras universitarias de informática, varía ligeramente de unas universidades a otras dentro del territorio nacional. El presente trabajo se enmarca dentro de la Universidad Politécnica de Valencia por estar los autores adscritos a ella.

Este trabajo presenta un curso de laboratorio para la asignatura de Estructura de Computadores I (ECI) enfocado al estudio del procesador. La herramienta de diseño digital que se utiliza para la implementación de los circuitos propuestos es el CASCAD [1] que corre sobre un entorno PC. La herramienta, aunque no analiza con detalle los retardos de tiempo de los componentes, es adecuada para un estudiante de primer curso ya que ofrece entre sus ventajas, la incorporación de muchos objetos como unidades aritmético-lógicas y registros, que se utilizan en la implementación de la ruta de datos. Además, al ser el entorno gráfico, se puede apreciar con claridad, qué líneas de control se encuentran activas en cada momento.

1 Introducción

La primera toma de contacto que tiene el alumno que ingresa en la universidad con el procesador, es durante el primer cuatrimestre del primer curso de carrera, en la asignatura de Fundamentos de Computadores (FCO) donde estudia los conceptos básicos necesarios para comprender el funcionamiento del procesador. A continuación, durante el segundo cuatrimestre, en la asignatura de ECI el alumno profundiza en distintas técnicas de implementación de la unidad de control, además de estudiar los circuitos básicos para implementar una sencilla unidad aritmético-lógica. Ambas asignaturas son troncales y sus contenidos se imparten tanto en la Escuela como en la Facultad de Informática en todas sus titulaciones. En la

asignatura de Estructuras de Computadores II (ECII) el alumno estudia el resto de componentes de la máquina de Von Neumann: el subsistema de memoria y la entrada/salida.

En la Facultad, durante el séptimo y octavo cuatrimestre, en las asignaturas de Arquitectura de Computadores I y Arquitectura de Computadores II, se profundiza en el estudio de técnicas avanzadas de implementación del procesador, como la segmentación y se finaliza el curso con el estudio de los sistemas multiprocesadores de memoria distribuida.

Los autores presentan a lo largo de este trabajo un curso de laboratorio para un estudiante de primero. El orden de las prácticas es el mismo que se sigue en el modelo teórico que se estudia en clases magistrales, y que presentan Patterson y Hennessy en [2]. La descripción de algunas de estas prácticas, así como los pasos a seguir para su elaboración se encuentran en [3].

En primer lugar debemos tener en cuenta que el alumno ha estudiado en clases teóricas, antes de llegar a las prácticas, el procesador MIPS R2000 de 32 bits. Para las sesiones prácticas se utiliza una ruta de datos de 16 bits diseñada por los autores y detallada en [3]. Para esta modificación ha sido necesario un cambio en las instrucciones que se utilizan adaptándolas a 16 bits, de modo que el subconjunto de instrucciones que el alumno utilizará en la práctica con sus correspondientes formatos son las siguientes:

	C.O.	Rs	Rt	Rd	Func					
add	0000	rs	rt	rd	001					
sub	0000	rs	rt	rd	010					
or	0000	rs	rt	rd	100					
and	0000	rs	rt	rd	101					
sw	1011	rs	rt		Despl.					
lw	1110	rs	rt		Despl.					
beq	1001	rs	rt		Despl.					
	15	12	11	9	8	6	5	3	2	0

Tabla 1: Formato de las instrucciones.

El resto del artículo se organiza tal y como se detalla a continuación. En el apartado dos se describen cada una de las prácticas propuestas: banco de registros, monociclo, microprogramada monociclo, multiciclo de dos fases y circuitos sumadores. Respecto a este apartado, cabe destacar que la asignatura objeto de estudio, ECI, tiene un total de 1.5 créditos de laboratorio. Esto supone al alumno 15 horas de prácticas, con lo que resulta evidente la imposibilidad de llevar a cabo todas las prácticas aquí propuestas. La idea que se sugiere es proponer al alumno, como actividades complementarias, las prácticas correspondientes a la unidad de control microprogramada monociclo y la unidad de control multiciclo de dos fases (dejándolas enunciadas en el servidor web de la asignatura) con el objetivo de reforzar los conocimientos adquiridos en teoría. Es condición necesaria para que el alumno apruebe la asignatura, realizar el resto de prácticas que aquí se exponen. Por último, en el apartado tres se muestran las conclusiones.

2 Descripción de las prácticas propuestas

En los siguientes puntos pasamos a definir cada una de las prácticas con sus objetivos, la base teórica que los alumnos ya han recibido en clases magistrales cuando acuden a realizar la práctica y una pequeña explicación de la realización de cada práctica.

Práctica 1: Banco de registros

En esta práctica los alumnos implementan un banco de ocho registros que se va a utilizar en el resto de las prácticas, con dos puertos de lectura y uno de escritura, como se muestra en la Figura 1. Con este circuito se persiguen dos objetivos: por una parte que el alumno tenga la primera toma de contacto con la herramienta y adquiera soltura, y por otra que la implementación realizada sea de utilidad para prácticas posteriores. Este hecho motiva al alumno, ya que construye sus propios *custom-chips*.

En la implementación elegida (Figura 2) se envía la dirección o número de registro donde se desea escribir por las líneas de selección del demultiplexor y se indica qué registro se desea leer por las líneas de selección del multiplexor; además la lectura se encuentra siempre habilitada, de modo

que sólo será necesaria la señal que indica la operación de escritura sobre el banco de registros.

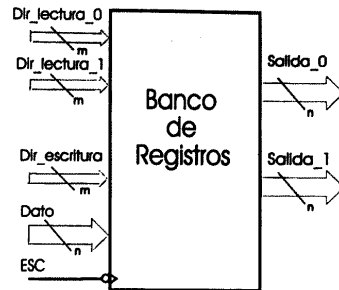


Figura 1: Banco de registros.

El alumno ha estudiado el banco de registros en la asignatura Fundamentos de Computadores en el cuatrimestre anterior. La relación entre esta implementación y los campos de las instrucciones que se han adoptado se muestra en la Tabla 2, donde también aparecen las relaciones existentes entre el banco de registros y la UAL de modo que queda definido para prácticas posteriores, así como la relación del banco de registros con la señal de reloj que indicará el momento de escritura (por flanco de bajada) en el registro destino.

Entradas y salidas del banco de registros	Patilla custom-chip	Procedencia (Entradas)/ Destino (Salidas)	Bits
Dir_lectura_puerto_0	I0	Campo Rs	11-10-9
Dir_lectura_puerto_1	I1	Campo Rt	8-7-6
Dir_escritura	I2	Campo Rd	5-4-3
Señal de escritura	I7	Reloj	—
Dato_escritura	I6	UAL	—
Dato_lectura_0	O0	Entrada I0 de la UAL	—
Dato_lectura_1	O1	Entrada I1 de la UAL	—

Tabla 2: Relación de entradas y salidas del banco con su procedencia o destino.

El trabajo del alumno consiste en el uso de la herramienta CASCAD para implementar el banco de registros.

Una vez concluido, el alumno tiene la posibilidad de guardar el banco de registros y de este modo utilizarlo en posteriores prácticas.

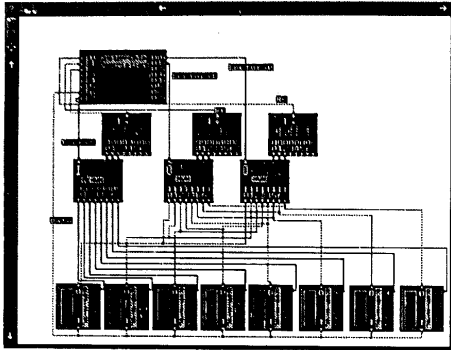


Figura 2: Implementación del banco de registros en CASCAD.

Práctica 2: Monociclo.

La ruta de datos que el alumno ha estudiado en clases magistrales se muestra en la Figura 3, puesto que es muy similar a la ruta práctica se utilizará como referencia a lo largo del presente trabajo.

En esta práctica el alumno debe implementar el control para algunos circuitos de la ruta de datos monociclo. Para ello se ofrece al alumno un fichero (Figura 4) con la ruta de datos para facilitarle la labor de cableado (donde se emplea gran cantidad de tiempo y no se adquieren nuevos conocimientos).

Primero el alumno debe relacionar los elementos de la ruta implementada en el CASCAD con sus correspondientes de la ruta teórica (Figura 3).

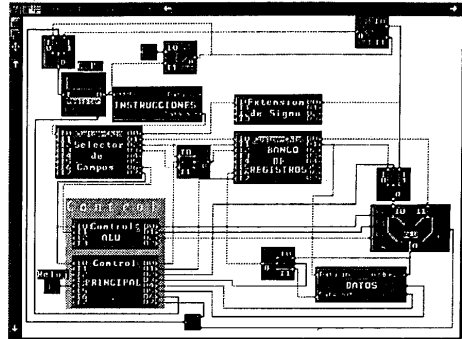


Figura 4: Ruta de datos monociclo.

Aparecen cuatro *custom-chips* vacíos: la unidad de extensión de signo, el selector de campos, el control de la UAL y el control principal, que el alumno deberá completar.

Para simplificar el diseño de la unidad de control, se realiza de forma independiente el control de la UAL y el control principal, tal y como los alumnos han aprendido a hacerlo en clases teóricas.

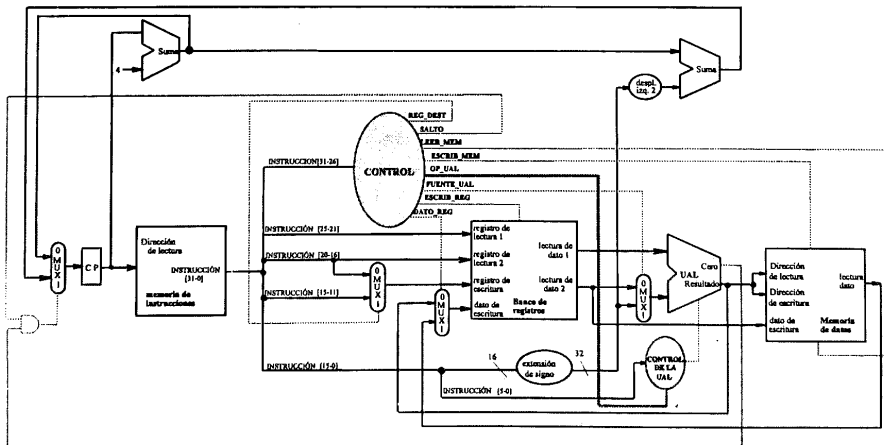


Figura 3: Ruta de datos para la arquitectura MIPS junto con la unidad de control.

	CO3	CO2	CO1	CO0	ESCRIB REG	FUENTE UAL	DATO REG	REG DEST	LEER MEM	ESCRIB MEM	SALTO
add	0	0	0	0	1	0	0	1	0	0	0
sub	0	0	0	0	1	0	0	1	0	0	0
and	0	0	0	0	1	0	0	1	0	0	0
or	0	0	0	0	1	0	0	1	0	0	0
beq	1	0	0	1	0	0	X	X	0	0	1
sw	1	0	1	1	0	1	X	X	0	1	0
lw	1	1	1	0	1	1	1	0	1	0	0

Tabla 3: Tabla de verdad asociada al circuito de control principal.

La tabla de verdad asociada con el circuito de control principal, aparece en la Tabla 3, teniendo como entrada el código de operación y como salidas de control: las de los cuatro multiplexores, las de escritura y lectura en la memoria de datos, la de salto y la señal de escritura en el contador de programa. En la Tabla 4 aparece la simplificación de las señales de control y la patilla del *custom-chip* principal por la que salen, el *custom-chip* correspondiente aparece en la Figura 5.

SIMPLIFICACIÓN	PATILLA DE SALIDA
$ESCRIB_REG = \overline{CO0}$	O1
$FUENTE_UAL = CO1$	O2
$DATO_REG = CO2$	O3
$REG_DEST = \overline{CO3}$	O0
$LEER_MEM = CO2$	O5
$ESCRIB_MEM = CO0 \cdot CO1$	O4
$SALTO = CO3 \cdot \overline{CO1}$	O7
"Escritura en CP"	O6

Tabla 4: Simplificación de las señales de control principal y patilla de salida del *custom-chip*.

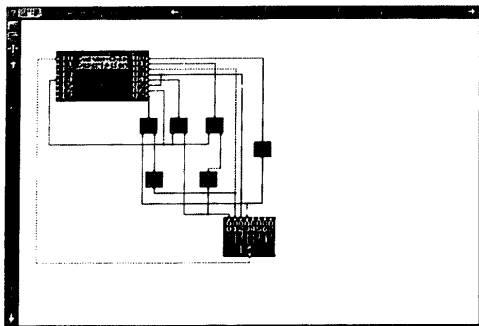


Figura 5: *Custom-chip* correspondiente al control principal.

La tabla de verdad asociada al circuito de control de la UAL se presenta en la Tabla 5, donde las entradas son el código de operación de la instrucción y el campo función, y las salidas son a su vez entradas de control de la UAL. La simplificación de las señales asociadas a dicha tabla aparecen en la Tabla 6, y el *custom-chip* correspondiente en la Figura 6.

CO3	CO2	CO1	CO0	F2	F1	F0	C2	C1	C0
0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	1	0	0	1	0
0	0	0	0	1	0	0	1	0	0
0	0	0	0	1	0	1	1	0	1
1	0	0	1	X	X	X	0	1	0
1	0	1	1	X	X	X	0	0	1
1	1	1	0	X	X	X	0	0	1

Tabla 5: Tabla de verdad asociada al circuito de control de la UAL.

SIMPLIFICACIÓN
$C2 = \overline{CO3} \cdot F2$
$C1 = \overline{CO3} \cdot F1 + CO3 \cdot \overline{CO1}$
$C0 = \overline{CO3} \cdot F0 + CO1$

Tabla 6: Simplificación de las señales de control asociadas al circuito de control de la UAL.

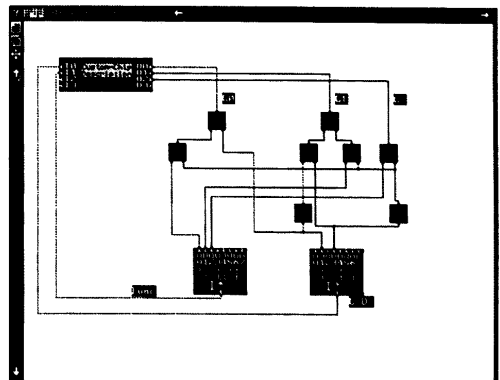


Figura 6: *Custom-chip* correspondiente al control de la UAL.

Una vez se ha implementado el circuito, el alumno debe comprobar su funcionamiento mediante la carga y ejecución de un sencillo programa en la memoria de instrucciones. Obviamente, si el alumno no implementa estos componentes de manera correcta, el procesador no tendrá el funcionamiento deseado.

CONTENIDO DE MEMORIA							
DIR	ESCRIB REG	FUENTE UAL	DATO REG	REG DEST	LEER MEM	ESCRIB MEM	SAL TO
0	1	0	0	1	0	0	0
1	0	0	0	0	0	0	1
3	0	1	0	0	0	1	0
2	1	1	1	0	1	0	0

Tabla 7: Memoria de microprograma.

Práctica 3: Microprogramada monociclo.

En esta práctica, el alumno simplemente tiene que variar el contenido de los circuitos de control que pasa de contener puertas lógicas a tener una memoria de microprograma, con una microinstrucción para cada instrucción máquina, debido a que al ser monociclo, todas las señales de control que activa una instrucción, las activa en el mismo ciclo.

A partir de la tabla de verdad realizada para el diseño de la unidad de control principal en la práctica anterior, el alumno puede apreciar que los bits CO0 y CO1 son suficientes para determinar el valor que deben tomar las señales de control, y por tanto el contenido de la memoria de microprograma sería el reflejado en la Tabla 7. Además, el alumno deberá relacionar las patillas de salida de la memoria y las patillas de salida del custom-chip (Tabla 8). El contenido del custom-chip es el representado en la Figura 7.

SEÑAL CONTROL	PATILLA SALIDA: CUSTOM-CHIP / MEMORIA CONTROL
ESCRIB REG	01
FUENTE UAL	02
DATO REG	03
REG DEST	00
LEER MEM	05
ESCRIB MEM	04
SALTO	07

Tabla 8: Relación entre las señales de control y patilla de salida del custom-chip.

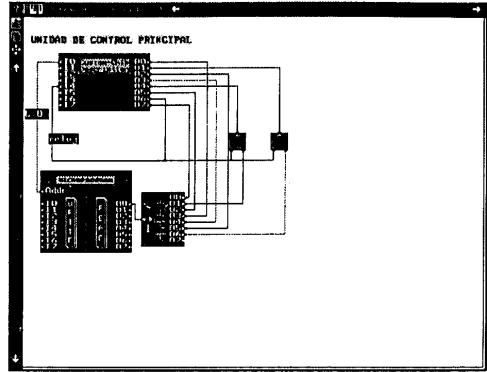


Figura 7: Custom-chip correspondiente al control principal.

Una vez concluido el diseño de la unidad de control principal, el diseño de la unidad de control de la UAL siguiendo asimismo dos etapas: la primera donde se realiza la tabla de verdad correspondiente (Tabla 9) con las simplificaciones que sean necesarias y la segunda que consiste en el diseño mediante la herramienta CASCAD de los componentes (custom-chips) necesarios (Figura 8).

	DIRECCIÓN		CONTENIDO					
	CO3	CO1	F2	F1	F0	C2	C1	C0
add	0	0	0	0	1	0	0	1
sub	0	0	0	1	0	0	1	0
or	0	0	1	0	0	1	0	0
and	0	0	1	0	1	1	0	1
beq	0	1	X	X	X	0	1	0
sw	1	0	X	X	X	0	0	1
lw	1	1	X	X	X	0	0	1

Tabla 9: Tabla de verdad asociada al circuito de control de la UAL.

Una vez se ha implementado el circuito, el alumno debe comprobar su funcionamiento. Para ello se le proporciona un programa que debe codificar y ejecutar en la máquina para comprobar su correcto funcionamiento.

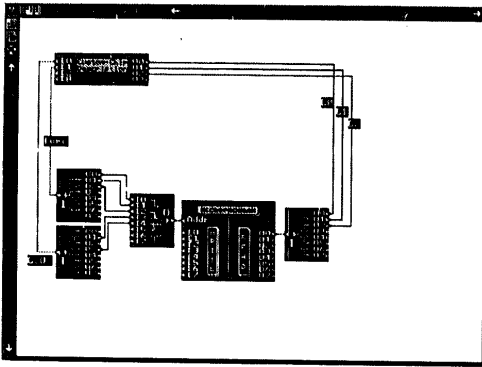


Figura 8: Custom-chip correspondiente al control de la UAL.

Práctica 4: Multiciclo de dos fases.

Esta práctica, al igual que la anterior no es obligatoria para el alumno, la intención de los autores es dejarlas propuestas y motivarles para que intenten resolverlas debido a su interés para reforzar los conocimientos adquiridos en clases teóricas. La práctica anterior se ha detallado para hacer ver que su realización no es demasiado compleja a partir de las nociones que el alumno tiene cuando la aborda.

La correspondiente a la unidad de control multiciclo de dos fases, no se va a resolver completamente sino únicamente a comentar cuales serían los pasos para su realización.

En clases teóricas se ha estudiado el diseño de una unidad de control clásica de cinco fases; para la realización de esta práctica, los autores ofrecen al alumno una ruta de datos de sólo dos fases (Figura 9). A partir de ella el alumno debe estudiarla para comprender las diferencias básicas entre ambas rutas de datos.

Una vez establecidas, la metodología a seguir es similar al resto de prácticas. Se aborda el diseño de la unidad de control por separado para la unidad de control principal y para la unidad de control de la UAL en las siguientes fases: en primer lugar se deben identificar cuales son las señales de control, a continuación realizar la máquina de estados para la unidad de control multiciclo de dos fases (en clase se ha visto para 5 fases), antes de llevar a cabo la implementación se realiza una simplificación de las señales que intervienen en la unidad de control y una vez realizada dicha simplificación se procede a la implementación del custom-chip correspondiente. Por último es interesante realizar una simulación para comprobar el correcto funcionamiento de la unidad de control implementada.

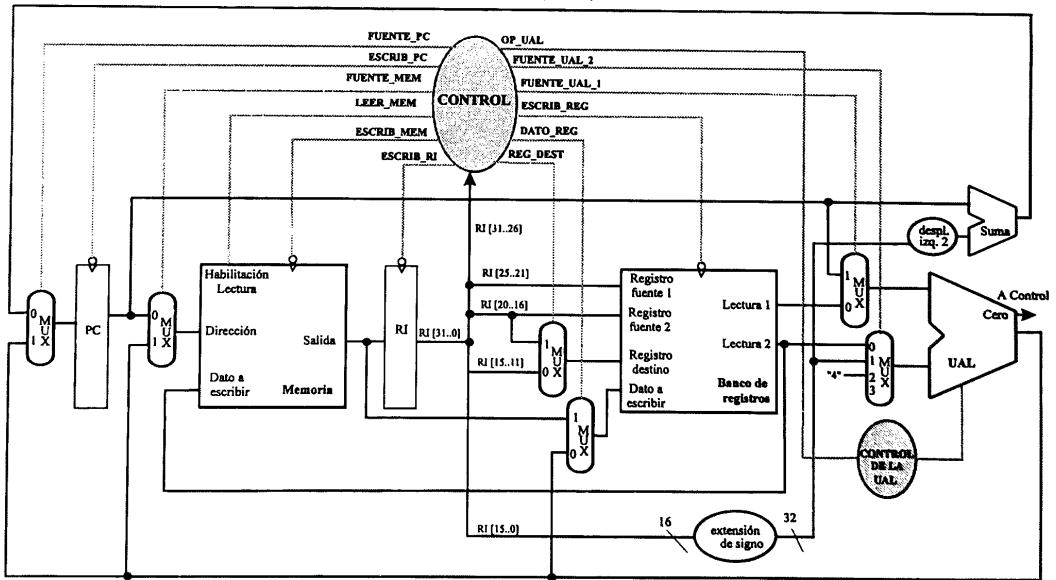


Figura 9: Ruta de datos y unidad de control de dos fases.

Práctica 5: Sumador-restador de 8 bits.

Para que el estudio del procesador sea completo es necesario incorporar alguna práctica sobre los circuitos aritmético-lógicos.

Se le propone al alumno la realización de dos prácticas relacionadas con la unidad aritmético-lógica. La primera de ellas es la realización de un sumador/restador de 8 bits. Los pasos a seguir para la realización de la práctica serían los que se detallan a continuación; en primer lugar el alumno deberá realizar un semisumador, cuyas características ya conoce de las clases teóricas, una vez realizado lo utilizará para implementar un sumador completo; el siguiente paso será construir un CPA (sumador con propagación de acarreo) a partir de los sumadores ya implementados para números de 8 bits, para terminar con la realización de un sumador/restador de 8 bits. De igual modo que en el resto de las prácticas, el alumno debe implementar un circuito de comprobación para verificar su correcto funcionamiento.

Práctica 6: Árbol de Wallace.

La segunda práctica relacionada con circuitos aritmético-lógicos que debe realizar el alumno es la implementación de un árbol de Wallace para 4 operandos de 4 bits. Para su realización el alumno dispone de los sumadores completos que ha diseñado en la práctica anterior. A partir de ellos deberá construir el árbol y comprobar su correcto funcionamiento.

3 Conclusiones

La implementación mediante herramientas de simulación de circuitos de control del procesador, ayuda a una mejor comprensión del funcionamiento del procesador. La ayuda de herramientas de simulación permite entrar en mucho más detalle (ciclo de simulación) que el que se imparte en lecciones magistrales.

La implementación de diversos circuitos de control con distintas técnicas ayuda no sólo a la comprensión sino que habilita al alumno para poder diseñar sus propios procesadores educativos.

La implementación de sumadores mediante árboles de Wallace, facilita su comprensión, ya que el alumno debe realizar correctamente sus conexiones para que funcione.

Referencias

- [1] *Computer Assisted Simulation for Circuit Analysis & Design (CASCAD). Reference Manual.* Edusoft.
- [2] Patterson, D.A., Hennessy, J.L. *Organización y diseño de computadores. La interfaz hardware/ software.* Editorial McGraw-Hill. Segunda edición en español, 1995.
- [3] Grup Tera. *Experiencias en Estructura de Computadores.* Servicio de publicaciones de la U.P.V. Ref.005.