



Ordenadores - I. T. Telecomunicación

Examen ordinario - 1 de febrero de 2001

Notas importantes:

- No se considerarán válidas las soluciones entregadas a lápiz.
- En todas las hojas debe figurar el nombre del alumno y el grupo al que pertenece.
- Se recomienda leer *detenidamente* los enunciados.

Problema 1 (4 puntos)

En un ordenador Intel 8086 se ejecuta el programa que aparece a continuación. Las cuatro columnas de las que consta el listado indican lo siguiente:

- Columna 1 (la de la izquierda): Número de la instrucción, en decimal.
- Columna 2: Dirección del primer byte de la instrucción, en hexadecimal. Ese será el valor que tendrá el contador de programa (el registro IP) al inicio de la ejecución de esa instrucción.
- Columna 3: Codificación de la instrucción de la columna 4, una vez traducida a ensamblador, en hexadecimal.
- Columna 4: La instrucción correspondiente.

1	0000	B8 00 00	mov ax,datos
2	0003	8E D8	mov ds,ax
3	0005	B9 09 00	mov cx,0x0009
4	0008	51	push cx
5	0009	E8 06 00	call rutina
6	000C	59	pop cx
7	000D	B8 00 4C	mov ax,0x4c00
8	0010	CD 21	int 0x21
9	0012	5B	rutina: pop bx
10	0013	59	pop cx
11	0014	41	inc cx
12	0015	51	push cx
13	0016	53	push bx
14	0017	C3	ret

Se pide lo siguiente:

1. Indicar lo que hace este programa.
2. Supongamos que este sistema posee una cache asociativa por conjuntos de cuatro vías, de escritura diferida (*write back*), con una capacidad total de 2 Kbytes y un tamaño de bloque de 32 bytes. Indicar el número de bits que tiene cada uno de los campos de la dirección desde el punto de vista de la cache. El número de bits del bus de direcciones es el típico del 8086 (20).

3. Supongamos que la dirección `datos` representa el valor `0x4c58`. Justo antes de ejecutarse la primera instrucción el contenido de los registros de la CPU es el siguiente:

CS = 0x4c68
DS = 0x0000
SS = 0x4c58
IP = 0x0000
SP = 0x0100

Se pide lo siguiente: construir una tabla en la que, para cada acceso de memoria que se produzca, indique la siguiente información:

- Dirección (en hexadecimal).
- Campo de etiqueta (en hexadecimal).
- Campo de conjunto (en hexadecimal).
- Campo de palabra (en hexadecimal).
- ¿Lectura o escritura?
- ¿Se produce fallo?
- Etiqueta del nuevo bloque introducido, en su caso.

Rellenar esta tabla con los accesos de memoria que se produzcan al ejecutar secuencialmente el programa desde la instrucción número 1 hasta la instrucción número 7, ambas inclusive. Se supone que la cache está inicialmente vacía.

Nota: Tener en cuenta que el 8086 tiene un bus de datos de 16 bits, y puede realizar lecturas o escrituras de uno o dos bytes en una única operación, según convenga. Por otra parte, la pila del 8086 crece hacia direcciones decrecientes, y el puntero de pila apunta al primer byte del elemento que ocupa la tapa de la pila. Realizar las suposiciones adicionales que se consideren necesarias, justificándolas debidamente.

Problema 2 (2 puntos)

1. Aplicar el algoritmo de suma en punto flotante para sumar los siguientes números en formato IEEE de punto flotante de simple precisión, expresados en hexadecimal: A = 0x5f2a00be, B = 0xbfe00000. Expresar el resultado en notación IEEE de punto flotante. Comentar el resultado.
2. Dividir A = 0x7f800000 entre B = 0x7f800000. Expresar el resultado en notación IEEE de punto flotante. Comentar el resultado.

Cuestión 1 (1 punto)

Supongamos que al encender un ordenador no se inicializaran los bits de validez presentes en sus dos niveles de cache y, por lo tanto, contuvieran valores aleatorios. ¿Qué sucedería *exactamente* en ese caso? ¿Y si sólo se limpiaran los bits correspondientes a la cache de nivel 2?



Cuestión 2 (1 punto)

1. Describir brevemente los fundamentos teóricos del algoritmo de Booth. Desarrollar dicho algoritmo.
2. Aplicarlo a la multiplicación de los números 00001010_b por 00001011_b . ¿Bajo qué circunstancias este algoritmo se comporta peor que el algoritmo de suma y desplazamiento? Dar un ejemplo.

Cuestión 3 (1 punto)

1. Explicar la división en campos que se utiliza en una palabra de microinstrucción de la unidad de control.
2. Indicar para qué sirve cada campo, y como afectan a la ejecución del microprograma.

Cuestión 4 (1 punto)

Indicar por qué es importante una correcta elección del quantum de tiempo en el algoritmo de planificación round-robin. Dar ejemplos de lo que sucede si se elige un quantum demasiado pequeño o demasiado grande. ¿Se te ocurre alguna forma para calcular analíticamente la duración adecuada de un quantum si se dispusiera de cien duraciones de ráfaga típicas?