

ESTRUCTURA DE COMPUTADORES

Examen Parcial 3

Marzo 2013

Apellidos y Nombre

DNI

Grupo

--	--	--

1. **(2,5 puntos)** A un computador con procesador MIPS R2000 se le ha dotado de los siguientes módulos de memoria:

Módulo 1: BIOS en memoria EEPROM : 512MB a partir de la dirección 0x00000000

Módulo 2: SDRAM DDR: 1GB a partir de la dirección 0x40000000

Módulo 3: SDRAM DDR2 : 1GB a partir de la dirección 0xC0000000

Se pide:

- a) Complete el mapa de memoria de dicho computador, especificando claramente la dirección inicial y final de cada uno de los módulos y el tamaño de los huecos libres. **(1,5 puntos)**

Módulo	Dirección inicial/final	Tamaño
Modulo 1	0x00000000	512 MB
Espacio libre	0x1FFFFFFF	512 MB
	0x20000000	
Módulo 2	0x3FFFFFFF	1GB
	0x40000000	
Espacio libre	0x7FFFFFFF	1GB
	0x80000000	
Módulo 3	0xBFFFFFFF	1 GB
	0xC0000000	
	0xFFFFFFFF	

- b) Indique la función lógica que permite seleccionar el Módulo 3 (SDRAM DDR2), asumiendo lógica de selección a nivel bajo. **(0,5 puntos)**

$$\text{Sel_Modulo3}^* = a_{31}^* + a_{30}^*$$

- c) Indique qué líneas del bus de direcciones intervienen en el direccionamiento interno del Módulo 1. **(0,5 puntos)**

Desde a_2 hasta a_{27} (más BE_0^* , BE_1^* , BE_2^* , BE_3^*)

2. **(7,5 puntos)** Se dispone de un procesador MIPS R2000 con una cache L1 unificada para instrucciones y datos, la cual tiene una capacidad de 8 KB, tamaño de bloque de 16 bytes y correspondencia asociativa por conjuntos de 4 vías. La política de fallo en escritura es *write allocate*, la política de actualización *write back*, y el algoritmo de reemplazo LRU. En este contexto se desea acceder al vector A, de 16 elementos enteros (32 bits), con la finalidad de cambiar el signo de sus elementos. Para ello se dispone de la secuencia de código que se muestra a continuación:

```

A:          .data 0x10200300
           .word 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16

__start:    .text 0x00400000
           li $t1, 16
           lui $t0, 0x1020
           ori $t0, $t0, 0x0300
bucle:      lw $t2, 0($t0)
           sub $t2, $zero, $t2
           sw $t2, 0($t0)
           addiu $t0, $t0, 4
           addi $t1, $t1, -1
           bne $t1, $zero, bucle
           .end

```

- a) Indique el número de líneas que integran la cache y el número de conjuntos que comprende. **(0,5 puntos)**

Número de líneas	512 (8KB / 16B)
Número de conjuntos	128 (512 / 4 vías)

- b) Indique el nombre y el tamaño de los diferentes campos en que se interpreta la dirección de memoria para gestionar la cache L1. **(0,5 puntos)**

31	12	11	5	4	0
Etiqueta (21 b)			Conjunto (7 b)		Desplaz. (4 b)

- c) Indique los bits de la dirección que determinan el número de bloque de memoria al que pertenece. **(0,5 puntos)**

31

5 4

0

Número de Bloque_MP	Desplaz. (4 b)
---------------------	----------------

- d) Bajo los supuestos del enunciado del problema, calcule el tamaño en número de bits de la memoria de control (etiquetas y bits de control), indicando claramente el número de entradas (líneas) de la misma, así como la descomposición en campos y tamaño de cada una de las entradas. **(1 punto)**

512 líneas x [1(V) + 1(M) + 2 (LRU) + 21 (etiqueta)] = 512 x 25 = 12800 bits

- e) Indique los números de bloque de memoria que ocupan el vector A y el código, así como los conjuntos de cache a los que se mapearían. **(1 punto)**

Vector A		CÓDIGO	
Bloque de Memoria	Conjunto de Cache	Bloque de Memoria	Conjunto de Cache
0x1020030	0x30	0x0040000	0x00
0x1020031	0x31	0x0040001	0x01
0x1020032	0x32	0x0040002	0x02
0x1020033	0x33		

- f) Indíquese el valor de la etiqueta correspondiente a los bloques de memoria que ocupa el código. **(0,5 puntos)**

La etiqueta es la misma para los tres bloques de código

0x000800

- g) Contabilice el número total de fallos y de accesos tanto a los datos como al código. Justifique adecuadamente los valores reseñados. **(1 punto)**

DATOS		CÓDIGO	
Nº de Fallos	Nº de Accesos	Nº de Fallos	Nº de Accesos
4 (de inicialización) Uno por cada bloque	16x2=32 (lectura y escritura)	3 (de inicialización) Uno por cada bloque	3+6x16= 99
Tasa de Aciertos	$H = \frac{(32-4) + (99-3)}{32+99} = \frac{124}{131} = 0,9466$ <p>Al ser la cache unificada, H se calcula de manera conjunta</p>		

- h) En el contexto anterior, suponiendo que los tiempos de acceso a la L1 y a la memoria principal fueran de 1 ns y 200 ns, respectivamente, calcule cuál sería el tiempo medio de acceso a memoria. **(0,75 puntos)**

$$T_m = H \times T_{L1} + (1 - H) \times T_{MP} = 0,9466 \times 1 \text{ ns} + (1 - 0,9466) \times 200 \text{ ns} = 11,63 \text{ ns}$$

- i) ¿Qué influencia tendría sobre la tasa de aciertos ubicar el vector A en la dirección 0x10200000? Razone la respuesta. **(0,75 puntos)**

NINGUNA, pues aunque la nueva ubicación del vector A causaría que sus bloques se mapearan a los mismos conjuntos de cache que los bloques de código, no habría colisión porque el número de vías (4) es más que suficiente para albergar los dos bloques (uno de código y otro de datos)

- j) ¿Qué influencia tendría sobre la tasa de aciertos modificar la función de correspondencia? Razone la respuesta bajo los dos supuestos de aplicación de una correspondencia directa y de una totalmente asociativa. **(0,75 puntos)**

NINGUNA

En caso de correspondencia Directa, porque código y datos seguirían mapeándose a conjuntos (líneas en este caso) diferentes, luego no habría colisión.

En el caso de correspondencia totalmente Asociativa, porque la capacidad de la cache (128 líneas/vías) es suficiente para almacenar simultáneamente código y datos

- k) ¿Qué influencia tendría sobre la tasa de aciertos modificar la política de escritura a *write-no allocate*? Razone la respuesta. **(0,25 puntos)**

NINGUNA, porque cuando se va a escribir (`sw $t2, 0($t0)`) en el bloque, este ya se encuentra en la cache (acierto) como consecuencia del fallo producido anteriormente al leer (`lw $t2, 0($t0)`)