



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

IIC2343 – Arquitectura de Computadores

Ayudantía 11

Profesores: Hans-Albert Löbel Díaz, Jurgén Dieter Heysen Palacios

Ayudante: Germán Leandro Contreras Sagredo (glcontreras@uc.cl)

Temas a tratar

Los temas a tratar dentro de esta ayudantía son: **todos**, dentro de lo posible.

Preguntas

1. (Examen - I/2018)

- a. A continuación, se presenta un diagrama de una memoria RAM para el computador básico:

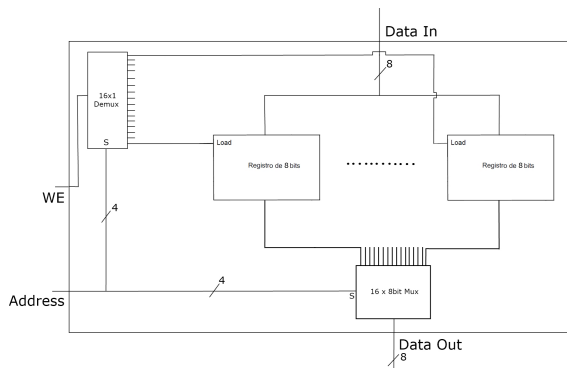


Figura 1: Representación de la memoria RAM.

A partir de este diagrama, ¿cómo modificaría esta componente para poder habilitar dos accesos (tanto para lectura como escritura) de forma simultánea? Puede hacer un diagrama o explicar en detalle su implementación.

b. Haciendo uso del componente anterior, modifique el computador básico para que pueda implementar las siguientes instrucciones:

- **DREAD** (*dir1*),(*dir2*): Se almacena la palabra ubicada en *dir1* dentro del registro A y la palabra ubicada en *dir2* dentro del registro B.
- **DWRITE** (*dir1*),(*dir2*): Se almacena en la dirección *dir1* el valor contenido en el registro A y en la dirección *dir2* el valor contenido en el registro B.
- **RW** (*dir1*),(*dir2*): Se almacena la palabra ubicada en *dir1* dentro del registro A y en la dirección *dir2* el valor contenido en el registro B.
- **WR** (*dir1*),(*dir2*): Se almacena en la dirección *dir1* el valor contenido en el registro A y se almacena la palabra ubicada en *dir2* dentro del registro B.

Si su implementación genera algún cambio con respecto al funcionamiento del computador básico, debe indicarlo y explicar qué se debe hacer para que no cambie la ejecución del resto de las instrucciones. Además, para cada instrucción, debe indicar las señales a activar (sean del diagrama original o las añadidas por usted) para su ejecución correcta. Puede realizar un diagrama o explicar en detalle su implementación.

Nota: Puede asumir que no se ejecutarán programas donde *dir1* = *dir2*.

c. Asumiendo que ya posee el computador básico de la implementación anterior, ¿qué otra instrucción podría incluir? No es necesario que mencione las señales involucradas para implementarla o el *opcode* asociado, pero sí que justifique por qué se podría implementar.

3. a. **(I3 - II/2011)** El siguiente programa se ejecutó en un computador con arquitectura x86 que tiene una *caché* de 4 bloques de 2 palabras cada uno:

Dirección	Label	
0		MOV [var2], 1
1	loop1:	MOV AL, [var2]
2		MUL [var2]
3		CMP [var1], AL
4		JL end
5		INC [var2]
6		JMP loop1
7	end:	DEC [var2]
8		RET
9	var1	db ?
10	var2	db 0

Cuadro 4: Programa con las direcciones y *labels*.

Al ejecutar el programa completo se obtuvo la siguiente secuencia de accesos a memoria:

0-10-1-10-2-10-3-9-4-5-10-6-1-10-2-10-3-9-4-5-10-6-1-10-2-10-3-9-4-7-10-8

Esta secuencia de accesos a memoria generó los siguiente estados en la *caché*:

Dir	B0	B1	B2	B3
0	0-1			
10	0-1	10-11		
1	0-1	10-11		
10	0-1	10-11		
2	0-1	10-11	2-3	
10	0-1	10-11	2-3	
3	0-1	10-11	2-3	
9	0-1	10-11	2-3	8-9
4	0-1	10-11	2-3	4-5
5	0-1	10-11	2-3	4-5
10	0-1	10-11	2-3	4-5
6	6-7	10-11	2-3	4-5
1	6-7	10-11	0-1	4-5
10	6-7	10-11	0-1	4-5
2	6-7	10-11	0-1	2-3
10	6-7	10-11	0-1	2-3

Dir	B0	B1	B2	B3
3	6-7	10-11	0-1	2-3
9	8-9	10-11	0-1	2-3
4	8-9	10-11	4-5	2-3
5	8-9	10-11	4-5	2-3
10	8-9	10-11	4-5	2-3
6	6-7	10-11	4-5	2-3
1	0-1	10-11	4-5	2-3
10	0-1	10-11	4-5	2-3
2	0-1	10-11	4-5	2-3
10	0-1	10-11	4-5	2-3
3	0-1	10-11	4-5	2-3
9	8-9	10-11	4-5	2-3
4	8-9	10-11	4-5	2-3
7	6-7	10-11	4-5	2-3
10	6-7	10-11	4-5	2-3
8	8-9	10-11	4-5	2-3

Cuadro 5: Estado de la *caché* por cada acceso.

En base a esta información, responda lo siguiente:

- I. ¿Qué valores puede tener la variable **var1** para que efectivamente se genere la secuencia de accesos detallada previamente?

- II. ¿Cuál es el *hit rate*?
 - III. ¿Qué tipo de *caché* es: *unified* o *split*?
 - IV. ¿Qué función de correspondencia y algoritmo de reemplazo (si corresponde) utiliza esta *caché*?
 - V. ¿Es posible mejorar el desempeño de esta *caché* durante la ejecución de este programa, sin modificar la cantidad y tamaño de los bloques? Si es posible, explique una posible mejora que se podría realizar para lograr un mejor *hit rate* que el actual y demuestre que efectivamente su modificación logra mejorarlo. Si no es posible, justifique por qué.
- b. **(Examen - II/2014)** Complete la siguiente tabla asumiendo que por cada entrada de la tabla de páginas, se utilizan 4 bits para *flags*:

Bits Dir. Virt.	Bits Dir. Fís.	Tam. Pág.	Bits Pág.	Bits Marco	Bits por entrada
32	32	16KB			
32	26			13	
	32		21		21
		32KB	25		25
64			48		28

- c. **(I3 - II/2011)** Asuma que una CPU tiene un espacio de direccionamiento virtual de 13 bits cuyas páginas son de 1KB. Esta máquina, sin embargo, cuenta tan solo con 4KB de memoria RAM disponibles para marcos. Asuma que los marcos están inicialmente vacíos. En un momento comienza a ejecutarse un proceso (P1) el cual, durante su ejecución, utiliza las direcciones de memoria desde la 0 hasta la 1500. Luego de esto el sistema operativo hace un cambio de contexto con lo que empieza a ejecutarse un segundo proceso (P2) el cual, durante su ejecución, utiliza las direcciones de memoria desde la 0 hasta la 500, y desde la 4500 a la 5000. Los datos que el proceso P2 almacena en estas últimas direcciones (de la 4500 hasta la 5000) son compartidos por el proceso P3 (tanto para lectura como para escritura), el que accede a ellos a través de las direcciones virtuales 2452 a la 2952. Se genera otro cambio de contexto y empieza a ejecutarse este tercer proceso (P3), el cual hace uso de las direcciones de memoria desde la 0 hasta la 1000, utilizando además datos desde la dirección 2500 a la 2600. Suponga que la política de reemplazo de páginas en los marcos es **FIFO**.
- I. Determine, para cada marco, de qué proceso o procesos es la información y/o datos que contiene. También indique qué páginas, de haber, se encuentran en disco.
 - II. Escriba las tablas de página asociadas a estos tres procesos.
- Luego de un cambio de contexto el proceso P1 lee la dirección de memoria 500. Posterior a esto, el mismo proceso requiere escribir en la dirección de memoria 600.
- I. Determine en qué dirección real se escribe en la memoria principal al escribir este proceso en la dirección 600.
 - II. Determine, para cada marco, de qué proceso o procesos es la información y/o datos que contiene. También indique qué páginas, de haber, se encuentran en disco.
 - III. Escriba las tablas de página asociadas a estos tres procesos.

- IV. Indique en qué direcciones físicas, de estar, se encuentran las direcciones virtuales:
- 2500 del proceso P3.
 - 2000 del proceso P1.
 - 4548 del proceso P2.
4. a. **(Examen - I/2015)** ¿Cómo podría solucionarse un *hazard* estructural que involucra la colisión de dos etapas que requieren la lectura de datos desde un registro?
- b. **(I3 - I/2017)** Indique por qué agregaría una complejidad adicional el tener soporte para llamado a subrutinas en el computador básico con *pipeline*.
- c. **(I3 - I/2017)** Considere un computador RISC-Harvard con un *pipeline* de 12 etapas, donde la unidad de salto se activa en la etapa 6 (40 % de las veces) o en la etapa 11 (60 % de las veces), dependiendo del origen de los parámetros necesarios para resolver el salto. Dado que la unidad de predicción de saltos de este computador acierta en el 75 % de las oportunidades, en promedio, ¿cuántos ciclos por salto pierde este computador?
- d. **(I3 - I/2017)** Indique cuándo y cómo podría implementarse la aceleración de un *pipeline*, si se sabe de antemano que una instrucción no utiliza una etapa.
- e. **(I3 - I/2016)** Diseñe un computador con *pipeline* de al menos 5 etapas, donde debido a restricciones del *hardware*, la etapa MEM debe ejecutarse antes que la etapa EX. El computador debe soportar las mismas funcionalidades que el computador básico con *pipeline*.