

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 1 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

Ejercicio 1

Una computadora tiene una memoria principal de $64K \times 16$ y una memoria caché, que utiliza la técnica de mapeo directo, capaz de almacenar 1K palabras de memoria principal. El tamaño de bloque de la caché es de 4 palabras. Determinar:

1. El número de bloques que puede alojar la caché.
2. El número de bits en los campos de etiqueta, línea y palabra del formato de dirección.
3. La cantidad de bits en cada línea de la caché y sus funciones, incluyendo un bit de validación.

Ejercicio 2

Se dispone de un sistema de memoria en el que interviene una memoria caché de mapeo directo organizada en bloques de 16 palabras. Un ejemplo de dirección de memoria principal en este sistema es FF1DE (en hexadecimal), mientras que un ejemplo de dirección de memoria caché es 732 (en octal). Determinar:

1. Longitud de una dirección de memoria física. Longitud de una dirección de memoria caché. Cuántos bits de la dirección de caché se destinan a indicar la línea y cuántos a la palabra. Longitud de la etiqueta.
2. Tamaño de la memoria principal si el ancho de palabra es de 16 bits. Tamaño de la memoria caché.

Ejercicio 3

Una memoria caché asociativa por conjuntos de 2 vías utiliza bloques de 4 palabras, pudiendo almacenar 2048 palabras de la memoria principal. El tamaño de la memoria principal es de $128K \times 32$. Determinar:

1. Organización de la caché.
2. Tamaño de la memoria caché.

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 2 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

Ejercicio 4

Se dispone de un sistema de memoria que utiliza direcciones de 32 bits. La unidad mínima direccionable es el byte. En memoria se encuentran tres arrays de elementos de 16 bits, almacenados en las direcciones representadas por el esquema siguiente:

	Array A	Array B	Array C
00101AF0	A[0]	3C509AF0	B[0]
00101AF2	A[1]	3C509AF2	B[1]
00101AF4	A[2]	3C509AF4	B[2]
00101AF6	A[3]	3C509AF6	B[3]
00101AF8	A[4]	3C509AF8	B[4]
	⋮		⋮

Se dispone también en memoria de dos programas que manipulan estos arrays y que se diferencian únicamente en la línea 5, siendo su código el siguiente:

Programa 1	Programa 2
1: int sp; 2: 3: sp = 0; 4: for (int i=0; i<n; ++i) 5: sp += A[i]*B[i];	1: int sp; 2: 3: sp = 0; 4: for (int i=0; i<n; ++i) 5: sp += A[i]*C[i];

1. Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo directo, que permite almacenar 32K palabras de memoria principal, y con un tamaño de bloque de 16 palabras, determinar qué programa se ejecutará más rápido.
2. Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo asociativo por conjunto de dos vías, que permite almacenar 32K palabras de memoria principal, y con un tamaño de bloque de 8 palabras, determinar qué programa se ejecutará más rápido.

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 3 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

Ejercicio 5

Un sistema de memoria de 256×8 incorpora una memoria caché de mapeo directo que puede almacenar 16 palabras de la memoria principal. El tamaño de bloque es de 4 palabras. En la memoria principal se encuentra una tabla A de 24 elementos, almacenada como indica la siguiente figura:

08	A[0,0]	A[0,1]	A[0,2]	A[0,3]
0C	A[1,0]	A[1,1]	A[1,2]	A[1,3]
• • • • • • • • • • • • • • • •				
24	A[7,0]	A[7,1]	A[7,2]	A[7,3]

Se diseñan dos programas para sumar todos los elementos de la tabla, cuyo código es el siguiente:

Programa 1	Programa 2
1: int s; 2: 3: s = 0; 4: for (int i=0; i<8; ++i) 5: for (int j=0; j<4; ++j) 6: s += A[i][j];	1: int s; 2: 3: s = 0; 4: for (int j=0; j<4; ++j) 5: for (int i=0; i<8; ++i) 6: s += A[i][j];

Determinar qué programa se ejecutará más rápido.

Ejercicio 6

Se dispone de un sistema de memoria que utiliza direcciones de 32 bits. La unidad mínima direccionable es el byte. En memoria se encuentran dos arrays de elementos de 8 bits, almacenados en las direcciones representadas por el esquema siguiente:

Array A. Dirección de comienzo: 00101AF0												
Contenido	12	4	67	81	231	89	106	198	206	11	39	162

Array B. Dirección de comienzo: 3C50BAF0												
Contenido	102	44	54	241	163	25	175	102	39	89	156	250

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 4 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo directo que permite almacenar 32 K palabras de memoria principal y que el tamaño de bloque es de 8 palabras, representar gráficamente el contenido de la memoria caché después de 3 iteraciones del bucle de código que se presenta a continuación:

```
1: int sp;
2:
3: sp = 0;
4: for (int i=0; i<n; ++i)
5:     sp += A[i]*B[i];
```

Nota: Representar únicamente las líneas de caché que han sido modificadas por el programa en ejecución.

Ejercicio 7

Se dispone de un sistema de memoria de $64K \times 8$. La unidad mínima direccionable es el byte. En memoria se encuentran dos arrays de elementos de 8 bits, almacenados en las direcciones representadas por el esquema siguiente:

Array A. Dirección de comienzo: 18F0												
Contenido	12	4	67	81	231	89	106	198	206	11	39	162
Array B. Dirección de comienzo: B2D0												
Contenido	102	44	54	241	163	25	175	102	39	89	156	250

Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo directo que permite almacenar 1K palabras de memoria principal y que el tamaño de bloque es de 4 palabras, representar gráficamente el contenido de la memoria caché después de 3 iteraciones del bucle de código que se presenta a continuación:

```
1: int sp;
2:
3: sp = 0;
4: for (int i=0; i<n; i+=2)
5:     sp += A[i]*B[i];
```

Nota: Representar únicamente las líneas de caché que han sido modificadas por el programa en ejecución.

Ejercicio 8 **NO ENTRA**

Se tiene el siguiente programa implementado en instrucciones MIPS. El array “data” se encuentra ubicado en memoria a partir de la posición 10010000 (en hexadecimal) y contiene 64 elementos de tipo palabra con el valor 1.

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 5 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

```
1:      move $s0, $zero # Índice X
2:      move $s1, $zero # Índice Y
3:      move $s2, $zero # Acumulador
4:      la $s3, data
5:      li $s4, 8 # Número de filas y columnas del array "data"
6:
7: loop:   sll $t0, $s1, 3
8:      add $t0, $s0, $t0
9:      sll $t0, $t0, 2
10:     add $t0, $s3, $t0
11:    lw $t0, 0($t0)
12:    add $s2, $s2, $t0
13:    addi $s1, $s1, 1
14:    bne $s1, $s4, loop
15:    move $s1, $zero
16:    addi $s0, $s0, 1
17:    bne $s0, $s4, loop
```

Determinar el porcentaje de aciertos de caché del programa según las siguientes configuraciones de la memoria caché:

1. Mapeo directo con 16 bloques de 4 palabras.
2. Mapeo directo con 8 bloques de 8 palabras.

Ejercicio 9

Dado un espacio de direccionamiento virtual, especificado mediante una dirección de 24 bits, y un espacio de memoria física, especificado mediante una dirección de 16 bits, determinar:

1. El número de palabras que existen tanto en el espacio de direccionamiento virtual como en el de memoria física.
2. Si se utiliza la técnica de paginación con un tamaño de página de 2K palabras, indicar cuántas páginas y bloques hay en total.

Ejercicio 10

Un sistema de memoria virtual paginado tiene un tamaño de página de 1K palabras. Pueden direccionarse 8 páginas. La memoria tiene capacidad para alojar 4 bloques. En un momento determinado, la tabla de páginas de un proceso contiene las siguientes entradas:

Página	Bloque
0	3
1	1
4	2
6	0

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 6 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

Elabórese una lista de los rangos de direcciones virtuales que producirán un fallo de página en caso de ser accedidas por la CPU.

Ejercicio 11

Un sistema de memoria virtual paginado tiene un tamaño de página de 1K palabras. El espacio de direcciones es de 8K palabras y el de memoria de 4K palabras. Durante un cierto intervalo de tiempo se referencian las siguientes páginas en el orden indicado:

4, 0, 2, 0, 1, 2, 6, 1, 4, 0, 1, 1, 0, 2, 6, 3, 3, 5, 7

Determinar las 4 páginas que permanecen en memoria principal después de acceder a las páginas indicadas, según el algoritmo de sustitución utilizado (FIFO, LRU, LFU).

Ejercicio 12

En un sistema de memoria virtual paginado el espacio de direcciones es de 4 MB, mientras que el espacio de memoria es de 32 KB. El tamaño de página es de 512 bytes. Determinar:

1. Longitud en bits de una dirección virtual y de una dirección física.
2. Número de páginas y bloques del sistema de memoria.
3. Dimensiones de las tablas de páginas, según se implementen mediante una memoria de acceso aleatorio o a través de una memoria asociativa.
4. Un proceso tiene una extensión de 32 páginas consecutivas en memoria virtual. El contenido de la tabla de páginas del proceso es el siguiente:

Página	Bloque
0	10
4	1
9	5

Página	Bloque
11	14
12	31
16	7

Página	Bloque
20	20
24	39
30	9

Realizar la traducción de las siguientes direcciones virtuales a direcciones físicas:

- 4.1. 000020
- 4.2. 0E8000
- 4.3. 0011A4
- 4.4. 0020A3

Ejercicio 13

En un sistema de memoria virtual de segmentación paginado, el espacio de direccionamiento lógico consta de 128 segmentos. Cada segmento puede tener hasta 32 páginas de 4K palabras cada una. La memoria física consta de 4K bloques de 4K palabras cada uno. Obtener los formatos de dirección lógica y física.

ARQUITECTURA DE COMPUTADORES

2º Grado en Ingeniería Informática

Curso 2011 – 2012

Página 7 de 7

Relación de ejercicios

Tema 5: Unidad de memoria. Jerarquía de la memoria de un computador

Ejercicio 14

En un sistema de memoria virtual de segmentación paginado, con un tamaño de página de 256 palabras, las tablas de segmento y página se encuentran en un instante determinado como indica la siguiente figura (todos los números están en hexadecimal):

Tabla de segmentos	
Segmento	Dir. Tabla de páginas
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

Tabla de páginas	
Página	Bloque
0	012
1	000
2	019
64	A61
9A	8C7

Tabla de páginas	
Página	Bloque
0	BA1
1F	6C2
A5	120
A6	FF0

Determinar:

1. Formato de dirección lógica y física.
2. Tamaño de la memoria lógica y física.
3. Rango de direcciones de memoria virtual que se encuentran en memoria física y rango de direcciones físicas que ocupan.
4. Contenido de la TLB suponiendo que todas las páginas indicadas en la figura han sido accedidas al menos una vez.

INICIALES

LDC = longitud de cache

LDMP = longitud de memoria principal

NLDMP = número LDMP

Bl = bloque

NBl = nº bloques

TMp = tamaño memoria principal

TMC = tamaño memoria cache

TVMC = Tamaño Vía memoria Cache

TBl = Tamaño Bloque

DHC = Direc memoria Cache

NLC = Número líneas cache

W = palabra

EDMP = Ejemplo Dir memoria Principal

EDMC = Ejemplo dir memoria cache

NV = número de vías

LDML = longitud de memoria lógica

TMF = tamaño memoria física

TTP = tamaño tabla paginación

NP = num página

NB = num bloque

TP = tam página

TB = tam bloque

LDMF = longitud de memoria físicas

LDMV = longitud de memoria virtual

TMV = tamaño memoria virtual

Ns = número segmento

NP/S = número de páginas/segmento

CACHÉ

Ejercicio 1

Una computadora tiene una memoria principal de $64K \times 16$ y una memoria caché, que utiliza la técnica de mapeo directo, capaz de almacenar 1K palabras de memoria principal. El tamaño de bloque de la caché es de 4 palabras. Determinar:

1. El número de bloques que puede alojar la caché.
2. El número de bits en los campos de etiqueta, línea y palabra del formato de dirección.
3. La cantidad de bits en cada línea de la caché y sus funciones, incluyendo un bit de validación.

$$T_{MP} = 64K \times 16$$

$$W = 16 \text{ b}$$

$$T_{MC} = 1K \times 16$$

$$Bl = 4W / \log_2 4 = 2 \text{ bits}$$

$$1) NBl = \frac{1K}{4} = 256 / \log_2 256 = 8 \text{ bits}$$

$$2) LDC = \frac{\text{LINEA}}{8 \text{ bits} + 2 \text{ bits}} = 10 \text{ bits} \quad 6 + 10$$

$$LDMp = \log_2 64K = \log_2 64 + \log_2 1K = \frac{\text{ETIQUETA}}{6 \text{ bits}} + \frac{\text{LINEA}}{8 \text{ bits}} + \frac{\text{DirPALABRA}}{2 \text{ bits}} = 16 \text{ bits}$$

$$3) \begin{array}{|c|c|c|c|c|c|} \hline & \text{bit Validación} & \text{Etiqueta} & \text{Palabra} & \text{Palabra} & \text{Palabra} \\ \hline & 1 \text{ b} & 6 \text{ b} & 16 \text{ b} \times 4 & & \\ \hline \end{array} = 1 + 6 + 4 \cdot 16 = 71 \text{ bits}$$

Ejercicio 2

Se dispone de un sistema de memoria en el que interviene una memoria caché de mapeo directo organizada en bloques de 16 palabras. Un ejemplo de dirección de memoria principal en este sistema es FF1DE (en hexadecimal), mientras que un ejemplo de dirección de memoria caché es 732 (en octal). Determinar:

1. Longitud de una dirección de memoria física. Longitud de una dirección de memoria caché. Cuántos bits de la dirección de caché se destinan a indicar la línea y cuántos a la palabra. Longitud de la etiqueta.
2. Tamaño de la memoria principal si el ancho de palabra es de 16 bits. Tamaño de la memoria caché.

$$TBL = 16W$$

$$1) \text{ DMP} = \text{FF1DE} \rightarrow \text{DMP} = 20 \text{ bits}$$

$$\text{DMC}_0 = 732 \rightarrow \text{DMC} = 9 \text{ bits}$$

Para direccionar 16 palabras $\rightarrow 4$ bits

$$\left. \begin{array}{l} \text{DMC} = 9 \text{ bits} \\ \text{DPal} = 4 \text{ bits} \end{array} \right\} \text{D Lines} = \text{DMP} - \text{DPal} = 9 - 4 = 5$$

$$\left. \begin{array}{l} \text{DMP} = 20 \text{ bits} \\ \text{DMC} = 9 \text{ bits} \end{array} \right\} \text{Etiqueta} = \text{DMP} - \text{DMC} = 20 - 9 = 11$$

$$2) \text{ TMP} = 2^{20} \times 16^6 = 2^{24} \text{ bits}$$

$$\begin{aligned} \text{DMC} &= \text{Codif Líneas} + V + \text{Etiquet} + 16 \text{ Palab} \\ &= 2^5 \times (1 + 11 + 16 \times 16) = 85766 \end{aligned}$$

Ejercicio 3

Una memoria caché asociativa por conjuntos de 2 vías utiliza bloques de 4 palabras, pudiendo almacenar 2048 palabras de la memoria principal. El tamaño de la memoria principal es de $128K \times 32$. Determinar:

1. Organización de la caché.
2. Tamaño de la memoria caché.

$$NV = 2$$

$$Bl = 4W$$

$$TMC = 2048 W$$

$$TMP = 128K \times 32$$

$$W = 32$$

$$2048b$$

1)	V1	V2
	$v Cfg w_0 w_1 w_2 w_3$	
	1024b	1024b

$$TVAC = \frac{TMC}{NV} = \frac{2048}{2} = 1024b$$

$$NLC = \frac{NBl}{V} = \frac{TVAC}{Bl} = \frac{1024}{4} = 256$$

$$DMC = \frac{\text{Líneas}}{8} + \frac{\text{Pal. Bloq}}{2} = 10 + 2 = 12$$

$$2) LDMP = \log_2 NLMF = \log_2 128K = \log_2 128 + \log_2 1K = 7 + 10 = 17b$$

$$Cfg = 17 - DMC = 17 - 10 = 7b$$

$$TMC = NV (\text{No Lin} (V + Cfg + 4W)) = \\ 2 \times 2^3 \times (1 + 7 + 4 \times 32)$$

Ejercicio 4

Se dispone de un sistema de memoria que utiliza direcciones de 32 bits. La unidad mínima direccionable es el byte. En memoria se encuentran tres arrays de elementos de 16 bits, almacenados en las direcciones representadas por el esquema siguiente:

	Array A	Array B	Array C
00101AF0	A[0]	3C509AF0	27B10330
00101AF2	A[1]	3C509AF2	27B10332
00101AF4	A[2]	3C509AF4	27B10334
00101AF6	A[3]	3C509AF6	27B10336
00101AF8	A[4]	3C509AF8	27B10338
	⋮	⋮	⋮

Se dispone también en memoria de dos programas que manipulan estos arrays y que se diferencian únicamente en la línea 5, siendo su código el siguiente:

Programa 1	Programa 2
1: int sp;	1: int sp;
2:	2:
3: sp = 0;	3: sp = 0;
4: for (int i=0; i<n; ++i)	4: for (int i=0; i<n; ++i)
5: sp += A[i]*B[i];	5: sp += A[i]*C[i];

- Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo directo, que permite almacenar 32K palabras de memoria principal, y con un tamaño de bloque de 16 palabras, determinar qué programa se ejecutará más rápido.
- Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo asociativo por conjunto de dos vías, que permite almacenar 32K palabras de memoria principal, y con un tamaño de bloque de 8 palabras, determinar qué programa se ejecutará más rápido.

$$LDMP = 32b$$

$$NLDMMP = 2^{32}$$

$$W = 8$$

$$TMp = 32K W$$

$$TMC = 16W$$

$$NBl = \frac{32K}{16} = 2K Bl$$

$$LDMC = \log_2 2K = \log_2 2 + \log_2 1K = 1 + 10 = 11b$$

$$DMC = \frac{\text{Lines}}{4} + \frac{\text{Registers}}{4} = 15b$$

$$Eff = 32 - 15 = 17$$

1) Programa 1

Elemento	Dir(MP)(H)	Etiqueta(17)	Línea(11)	Palabra	En cache	F/A	A cache
A[0]	00101AF0	00000000 0001 0000	Bit CORTADO 0001	1010 1111	0000	— A[0]-A[7]	F A[0]-A[7]
B[0]	3C501AF0	0011 1100 0101 0000	↑ Coincidencia 1001	1010 1111	0000 11	A[0]-A[7]	F B[0]-B[7]
A[1]	00101AF2	00000000 0001 0000	0001	1010 1111	0010	B[0]-B[7]	F A[0]-A[7]
B[1]	3C501AF0	0011 1100 0101 0000	1001	1010 1111	0010 10	A[0]-A[7]	F B[0]-B[7]
:							

Sigue dando fallas hasta el infinito, después van A[8]-A[15] y B[8]-B[15]...

Programa 2

Elemento	Dir(MP)(H)	Etiqueta(17)	Línea(11)	Palabra	En cache	F/A	A cache
A[0]	00101AF0	00000000 0001 0000	0001 1010 1111	0000	—	F	A[0]-A[7]
C[0]	27810330	0010 0111 1011 0001	0000 0011 0011	0000	—	F	C[0]-C[7]
A[1]	00101AF2	00000000 0001 0000	0001 1010 1111	0010	A[0]-A[7]	A	—
B[1]	27810332	0010 0111 1011 0001	0000 0011 0011	0010	C[0]-C[7]	A	—
:							

Hasta que llegue a A[8]-y C[8] no hará falta donde guardaban [8-15] y no dará falla hasta [16]

2) Al igual que en este caso el programa 2 será más rápido

Ejercicio 5

Un sistema de memoria de 256×8 incorpora una memoria caché de mapeo directo que puede almacenar 16 palabras de la memoria principal. El tamaño de bloque es de 4 palabras. En la memoria principal se encuentra una tabla A de 24 elementos, almacenada como indica la siguiente figura:

08	A[0,0]	A[0,1]	A[0,2]	A[0,3]
0C	A[1,0]	A[1,1]	A[1,2]	A[1,3]
• • • • • • • • • • • • • • • •				
24	A[7,0]	A[7,1]	A[7,2]	A[7,3]

Se diseñan dos programas para sumar todos los elementos de la tabla, cuyo código es el siguiente:

Programa 1	Programa 2
1: int s; 2: 3: s = 0; 4: for (int i=0; i<8; ++i) 5: for (int j=0; j<4; ++j) 6: s += A[i][j];	1: int s; 2: 3: s = 0; 4: for (int j=0; j<4; ++j) 5: for (int i=0; i<8; ++i) 6: s += A[i][j];

Determinar qué programa se ejecutará más rápido.

256×8

$8 \text{ bits} \times 1 \text{ byte}$

por filas (programa 1) 1F \rightarrow 3A cierto

por columnas (programa 2) todo falso

por lo que por filas sera mas rapido

Ejercicio 6

Se dispone de un sistema de memoria que utiliza direcciones de 32 bits. La unidad mínima direccionable es el byte. En memoria se encuentran dos arrays de elementos de 8 bits, almacenados en las direcciones representadas por el esquema siguiente:

Array A. Dirección de comienzo: 00101AF0												
Contenido	12	4	67	81	231	89	106	198	206	11	39	162

Array B. Dirección de comienzo: 3C50BAF0												
Contenido	102	44	54	241	163	25	175	102	39	89	156	250

Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo directo que permite almacenar 32 K palabras de memoria principal y que el tamaño de bloque es de 8 palabras, representar gráficamente el contenido de la memoria caché después de 3 iteraciones del bucle de código que se presenta a continuación:

```

1: int sp;
2:
3: sp = 0;
4: for (int i=0; i<n; ++i)
5:     sp += A[i]*B[i];

```

Nota: Representar únicamente las líneas de caché que han sido modificadas por el programa en ejecución.

$$THC = 32K W$$

$$TB = 8W$$

$$NBL = \frac{32K}{8} = 4K$$

$$LDAC = \log_2 4K = 12$$

$$DMC = \frac{\text{Lines}}{12} + \frac{\text{Palabra}}{3} = 15 \text{ b}$$

$$Efig = 32 - 15 = 17$$

Elemento	Dir (Mp)(H)	Etiqueta (17)	Lines (11)	Palabra	En cache	F/A	A cache
A[0]	00101AF0	0000 0000.0001.0000.0	001.1010.1111.0	000	_____	F	A[0]-A[7]
B[0]	3C50BAF0	0011 1100.0101.0000.1	011.1010.1111.0	000	_____	F	B[0]-B[7]
A[1]	00101AF1	0000 0000.0001.0000.0	001.1010.1111.0	001	A[0]-A[7]	A	_____
B[1]	3C50BAF1	0011 1100.0101.0000.1	011.1010.1111.0	001	B[0]-B[7]	A	_____
A[2]	00101AF2	0000 0000.0001.0000.0	001.1010.1111.0	010	A[0]-A[7]	A	_____
B[2]	3C50BAF2	0011 1100.0101.0000.1	011.1010.1111.0	010	B[0]-B[7]	A	_____

Ejercicio 7

Se dispone de un sistema de memoria de $64K \times 8$. La unidad mínima direccionable es el byte. En memoria se encuentran dos arrays de elementos de 8 bits, almacenados en las direcciones representadas por el esquema siguiente:

Array A. Dirección de comienzo: 18F0												
Contenido	12	4	67	81	231	89	106	198	206	11	39	162

Array B. Dirección de comienzo: B2D0												
Contenido	102	44	54	241	163	25	175	102	39	89	156	250

Suponiendo que el sistema de memoria incorpora una memoria caché con mapeo directo que permite almacenar 1K palabras de memoria principal y que el tamaño de bloque es de 4 palabras, representar gráficamente el contenido de la memoria caché después de 3 iteraciones del bucle de código que se presenta a continuación:

```

1: int sp;
2:
3: sp = 0;
4: for (int i=0; i<n; i+=2)
5:     sp += A[i]*B[i];

```

Nota: Representar únicamente las líneas de caché que han sido modificadas por el programa en ejecución.

$$T_{MP} = 64K \times 8$$

$$T_{MP} = 1K \text{ W}$$

$$T_{HC} = 4 \text{ W}$$

$$NBL = \frac{1K}{4} = 256 \text{ Bl}$$

$$LDMC = \log_2 256 = 8$$

$$DMC = \frac{\text{Lines}}{8} + \frac{\text{Palabras}}{4} = 12 \text{ b}$$

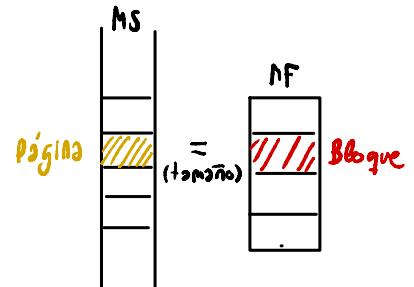
$$Efig = 16 - 12 = 4$$

Elemento	Dir(MP)(H)	Etiquetas(4)	Lines(8)	Palabras	En cache	F/A	A cache
A[0]	18F0	0001	1000.1111	0000	_____	F	A[0]-A[7]
B[0]	B2D0	1011	0010.1101	0000	_____	F	B[0]-B[7]
A[1]	18F1	0001	1000.1111	0001	A[0]-A[7]	A	_____
B[1]	B2D1	1011	0010.1101	0001	B[0]-B[7]	A	_____
A[2]	18F2	0001	1000.1111	0010	A[0]-A[7]	A	_____
B[2]	B2D2	1011	0010.1101	0010	B[0]-B[7]	A	_____

VIRTUAL

memoria principal = memoria física

Dir es más grande en DVV porque es mayor y hacen falta más bits



Ejercicio 9

Dado un espacio de direccionamiento virtual, especificado mediante una dirección de 24 bits, y un espacio de memoria física, especificado mediante una dirección de 16 bits, determinar:

1. El número de palabras que existen tanto en el espacio de direccionamiento virtual como en el de memoria física.
2. Si se utiliza la técnica de paginación con un tamaño de página de 2K palabras, indicar cuántas páginas y bloques hay en total.

$$LD_{hV} = 24$$

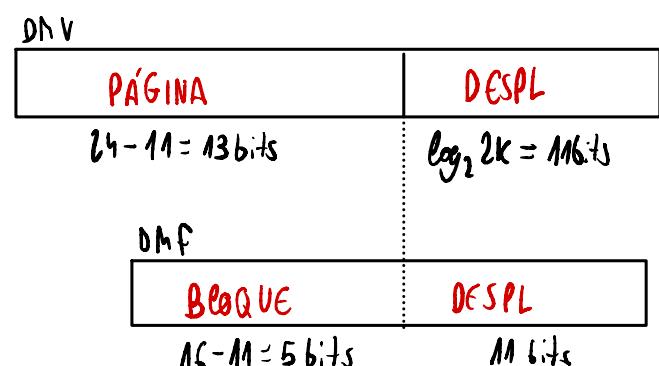
$$LD_{hf} = 16$$

$$ThV = 2^{24} W$$

$$Thf = 2^{16} W$$

$$TP = TB = 2K W$$

$$ThV = \frac{2^{24}}{2^{11}} = 2^{11-11} = 2^{13}$$



Ejercicio 10

Un sistema de memoria virtual paginado tiene un tamaño de página de 1K palabras. Pueden direccionarse 8 páginas. La memoria tiene capacidad para alojar 4 bloques. En un momento determinado, la tabla de páginas de un proceso contiene las siguientes entradas:

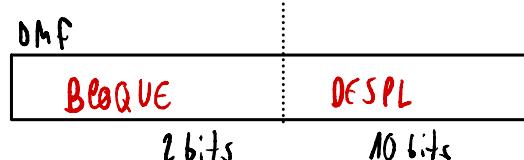
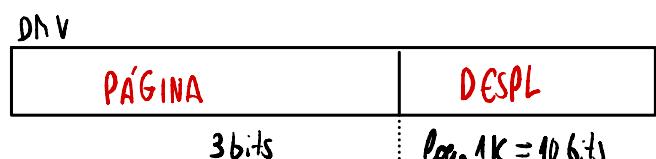
Página	Bloque
0	3
1	1
4	2
6	0

Elabórese una lista de los rangos de direcciones virtuales que producirán un fallo de página en caso de ser accedidas por la CPU.

$$TB = TP = 1K$$

$$NP = 8$$

$$NB = 4$$



Pág	Rango despl	Direcciones
010	00 0000 0000 11 1111 1111	0080 - 0BFF
011	00 0000 0000 11 1111 1111	0C00 - 0FFF
101	00 0000 0000 11 1111 1111	1400 - 17FF
111	00 0000 0000 11 1111 1111	1C00 - 1FFF

Ejercicio 11

Un sistema de memoria virtual paginado tiene un tamaño de página de 1K palabras. El espacio de direcciones es de 8K palabras y el de memoria de 4K palabras. Durante un cierto intervalo de tiempo se referencian las siguientes páginas en el orden indicado:

4, 0, 2, 0, 1, 2, 6, 1, 4, 0, 1, 1, 0, 2, 6, 3, 3, 5, 7

Determinar las 4 páginas que permanecen en memoria principal después de acceder a las páginas indicadas, según el algoritmo de sustitución utilizado (FIFO, LRU, LFU).

$$TP = 1K$$

$$NP = 8$$

$$NB = 4$$

FIFO

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

LRU

4 ⁰¹²³⁴⁵⁶	0 ¹²³⁴⁵⁶⁷	7
0 ⁰¹²³⁴⁵⁶	4 ⁰¹²³⁴⁵⁶	6
2 ⁰¹²³⁴⁵⁶	0 ¹²³⁴⁵⁶⁷	5
1 ⁰¹²³⁴⁵⁶	3 ⁰⁰⁰	3

LFU

4 ¹	6 ²	3 ³	3
0 ⁴²³⁴		0	
2 ¹²³		2	
1 ¹²³⁴		1	

Ejercicio 12

En un sistema de memoria virtual paginado el espacio de direcciones es de 4 MB, mientras que el espacio de memoria es de 32 KB. El tamaño de página es de 512 bytes. Determinar:

1. Longitud en bits de una dirección virtual y de una dirección física.
2. Número de páginas y bloques del sistema de memoria.
3. Dimensiones de las tablas de páginas, según se implementen mediante una memoria de acceso aleatorio o a través de una memoria asociativa.
4. Un proceso tiene una extensión de 32 páginas consecutivas en memoria virtual. El contenido de la tabla de páginas del proceso es el siguiente:

Página	Bloque
0	10
4	1
9	5

Página	Bloque
11	14
12	31
16	7

Página	Bloque
20	20
24	39
30	9

Realizar la traducción de las siguientes direcciones virtuales a direcciones físicas:

- 4.1. 0000020
- 4.2. 0E8000
- 4.3. 0011A4
- 4.4. 0020A3

$$TMV = 4 \text{ MB}$$

$$TMF = 32 \text{ KB}$$

$$TB = TP = 512 \text{ B} \longrightarrow 9 \text{ bits}$$

$$LDHV = \log_2(4M) = \log_2 4 + \log_2 1K + \log_2 1K = 2 + 10 + 10 = 22$$

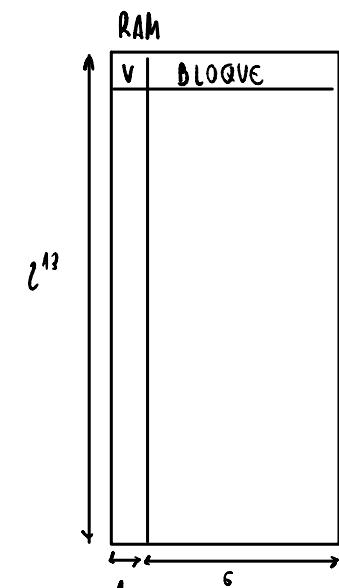
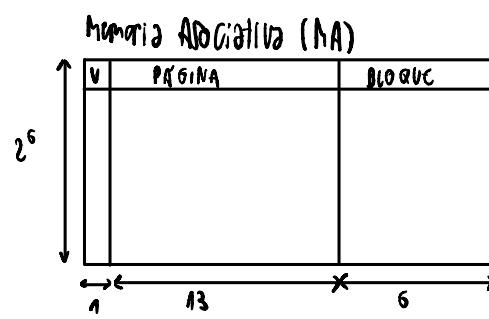
$$LDMF = \log_2(32K) = \log_2 32 + \log_2 1K = 15$$

$$NP = \frac{2^{22}}{2^9} = 2^{22-9} = 2^{13} \text{ p}$$

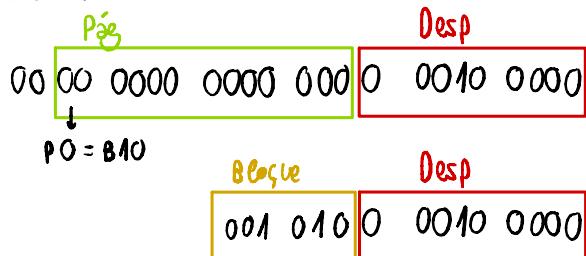
$$NB = \frac{2^{15}}{2^6} = 2^6 \text{ B}$$

$$TP_{RA} = 2^6 \cdot 20 = 280$$

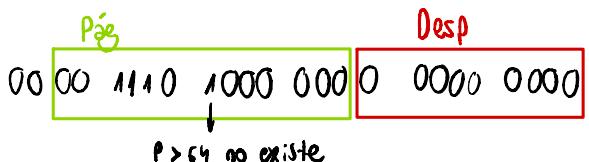
$$TP_{RM} = 2^{13} \cdot 7 = 51344$$



000020

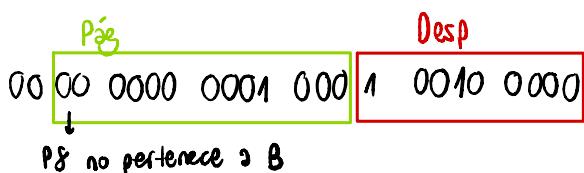


0E8000



Error de paginación, porque no está en memoria física, además esa página no aparece en proceso

0011A4



Error de paginación, porque no está en memoria física, pero pertenece, no ver desplazamiento se podría traducir.

0020A3 → 06A3

Ejercicio 13

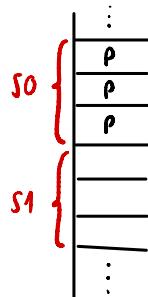
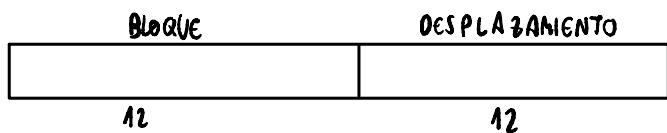
En un sistema de memoria virtual de segmentación paginado, el espacio de direccionamiento lógico consta de 128 segmentos. Cada segmento puede tener hasta 32 páginas de 4K palabras cada una. La memoria física consta de 4K bloques de 4K palabras cada una. Obtener los formatos de dirección lógica y física.

$$NS = 128$$

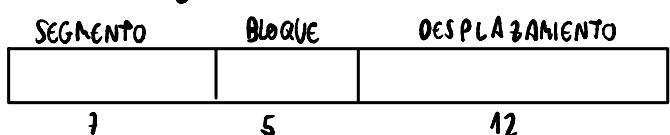
$$NP/S = 32$$

$$TP = TB = 4K$$

Dirección físicas (DF)



Dirección lógicas (DL)



Ejercicio 14

En un sistema de memoria virtual de segmentación paginado, con un tamaño de página de 256 palabras, las tablas de segmento y página se encuentran en un instante determinado como indica la siguiente figura (todos los números están en hexadecimal):

Tabla de segmentos	
Segmento	Dir. Tabla de páginas
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

Tabla de páginas	
Página	Bloque
0	012
1	000
2	019
64	A61
9A	8C7

Tabla de páginas	
Página	Bloque
0	BA1
1F	6C2
A5	120
A6	FF0

Tabla de páginas	
Página	Bloque
0	012
1	000
2	019
64	A61
9A	8C7

SEGMENTO 6
SEGMENTO F

Determinar:

4 bits 8 bits 12 bits

1. Formato de dirección lógica y física.
2. Tamaño de la memoria lógica y física.
3. Rango de direcciones de memoria virtual que se encuentran en memoria física y rango de direcciones físicas que ocupan. validación = 1
4. Contenido de la TLB suponiendo que todas las páginas indicadas en la figura han sido accedidas al menos una vez.

$$\begin{aligned} LDM_L &= 2^0 \\ TML &= 2^{20} W \\ LDmf &= 2^0 \\ fmf &= 2^{10} W \end{aligned}$$

DL

SEGMENTO	PÁGINA	DESPALZAMIENTO
4 bits	8 bits	8 bits

DF

BLOQUE	DESPALZAMIENTO
12 bits	8 bits

SEGMENTO	PÁGINA	DESPLAZAMIENTO	RANGO DL	RANGO DF
6	00	00 - FF	60000 - 600FF	61200 - 612FF
6	01	00 - FF	60100 - 601FF	60000 - 000FF
⋮	⋮	⋮	⋮	⋮
F	00	00 - FF	F0000 - F00FF	B1100 - B11FF
F	1F	00 - FF	F1F00 - F1FFF	6C200 - 6C2FF
⋮	⋮	⋮	⋮	⋮

Tabla TLB (segmento - página - bloque)

V	SEGMENTO	PÁGINA	BLOQUE
1	6	0	012
1	6	1	000
1	6	2	019
1	6	64	A61
1	6	9A	8C7
1	F	0	B11
1	F	1F	6C2
1	F	A5	120
1	F	A6	FF0