- Arquitectura de un computador es el comportamiento funcional de un sistema computador, desde el punto de vista de lenguaje ensamblador.

Arquitectura es:
conjunto de utensilios que el Hadware da para que se puedan utilizar. Lo que se ofrece.

Organización es:
¿Cómo ofrezco esta información?

- ARQUITECTURAS DEL CONJUNTO DE INSTRUCCIONES (ISA):

Es la especificación del conjunto de instrucciones máquina que ofrece un computador.

- Formato de instrucciones (codificación y decodificación)
- Conjunto de instrucciones
- Almacenamiento de datos
- Interrupciones y opciones.
- Mecanismos E/S

- ORGANIZACIÓN DE UN COMPUTADOR:

Son unidades funcionales que realizan una determinada arquitectura.

(ESQUEMAS)

- ESTRUCTURA DE UN COMPUTADOR:

Es especificación de los componentes de los que está compuesto un computador y forman en la que se conectan entre sí.

- familia: conjunto de computadores que tienen misma arquitectura pero usan diferentes estructuras.

El diseñador debe considerar:
- Su estructura
- Su funcionamiento

TECNOLOGÍA ⟷ ORGANIZACIÓN ⟷ ARQUITECTURA

- COMPUTADOR MULTINIVEL:

5. Lenguaje orientado a problemas (alto nivel)
4. Lenguaje ensamblador
3. Sistema operativo
2. Arquitectura conjunto instrucciones
1. Microarquitectura
0. Lógica digital.

Modelos de computación tradicionales:

- Modelo de Von Neuman. (ver)
- Modelo de Bus. (ver)
- Modelo de Harvard. (ver)

Taxonomía de Flynn: en cuanto al tipo de procedimiento:

Instrucciones de datos	Una interrupción a la vez	Muchas instrucciones a la vez
Un dato a la vez	SISD Argoitec Von Neuman	MISD Máquinas sistólicos
Muchos datos a la vez	SIMD Procesadores escalares / vectoriales	MIMD Multi-procesadores / computadores, redes

Influencias que afectan a la arquitectura de un computador:

(ver)

- Software (ver)
- Tecnología (ver)

VER: TOP 500 en computación	: NOTA

MÁQUINA DE 2 DIRECCIONES:

```
X LOAD B,C
  ADD B,C
  STORE B,C
  MUL B,D
  STORE D,B
  LOAD E,F
  SUB E,F
  STORE E,F
  LOAD G,H
  MUL E,G
  STORE E,G
  SUB H,E
  STORE H,E
  MUL H,I
  STORE I,H
  DIV I,D
  STORE A,I
```

```
✓ LOAD R1,B
  LOAD R2,C
  LOAD R3,D
  ADD R1,R2
  MUL ,R1,R3
  LOAD R2,E
  LOAD R3,F
  LOAD R4,G
  MUL R3,R4
  SUB R2,R3
  LOAD R3,H
  LOAD R4,I
  MUL R3,R4
  SUB R2,R3
  DIV R1,R2
  STORE A,R1
```

Ese está bien

MÁQUINA DE 3 DIRECCIONES (MEM/MEM):

```
ADD B,C,aux1
MUL aux1, D, aux1
MUL F,G, aux2
MUL H,I, aux3
SUB E, aux2, aux2
SUB aux2, aux3, aux2
DIV aux1, aux2.
```

MÁQUINA DE 3 DIRECCIONES (REG/REG)

```
LOAD R1,B
LOAD R2,C
ADD R3,R1,R2
LOAD R1,D
MUL R2,R1,R3
LOAD R1,F
LOAD, R3,G
MUL R1,R1,R3
LOAD R3,H
LOAD R4,I
MUL R4,R4,R3
```

```
ADD R3,R1,R4
LOAD R1,E
SUB R1,R1,R3
DIV R3,R2,R1
STORE A,R3
```

CÓDIGO	DIRECCIONAMIENTO	OPERANDO	REG		TAMAÑO	ANCHO BANDA
1 BYTE	2 BYTE	2 BYTE	4 BYTES			
16 BYTES	18 BYTES	18 BYTES	9 bits	PILA	TOTAL: 52 BYTES	
17 BYTES	17 BYTES	17 BYTES	0 bits	ACUMULADOR	TOTAL: 51 BYTES	83 BYTES
16 BYTES	18 BYTES	18 BYTES	23 x 4 bits	R-R	TOTAL: 46 BYTES	64 BYTES
7 BYTES	21 x 2 BYTES	21 x 2 BYTES	0 bits	MEM-MEM	TOTAL: 49 BYTES	91 BYTES
16 BYTES	18 BYTES	18 BYTES	30 x 4 bits	REG-REG	TOTAL: 49 BYTES	67 BYTES

TEMA 2: ARQUITECTURA DEL REPERTORIO DE INSTRUCCIONES.

Especificación que detalla el conjunto o repertorio de instrucciones que puede ejecutar una CPU.

- Punto de encuentro entre el diseñador de la CPU y el programador del lenguaje ensamblador.
- Depende del computador.
- Afecta de forma muy específica a la implementación de la CPU.

- **Aspectos a considerar durante el diseño de una ISA.**
  - Conjunto de operaciones
  - Tipos de datos que debe soportar

  (Faltan 3 más)

- **Clasificación ISA:**
  - Según el tipo de almacenamiento interno de la CPU:
    - Operandos implícitos:
      - Arquitectura tipo Pila
      - " " Acumulador

    (Faltan los operadores explícitos)
    (Estudiar: diferencias entre, tipo PILA, tipo ACUMULADOR y tipo GRR)

- **Arquitecturas de GPR:**
  - Número de operandos explícitos por instrucción típica de la ALU.
  - Número...

    (Faltan los 3 tipos) (Con sus ventajas e inconvenientes, con ejemplos)

Ejercicio:

$A = ((B+C) \cdot D) / (E - F \cdot G - H \cdot I)$

MÁQUINA DE 0 DIRECCIONES:

```
PUSH B
PUSH C   } Queda en la
ADD      }   cima
PUSH D
MUL
PUSH E
PUSH F
PUSH G
MUL
SUB
PUSH H
PUSH I
MUL
SUB
DIV
POP A
```

MÁQUINA DE 1 DIRECCIÓN:

```
LOAD B
ADD C
MUL D
STORE R1
LOAD F
MUL G
STORE R2   ---> Lo que hace es guardarlo en auxiliar.
LOAD H
MUL I
STORE R3
LOAD E
SUB R2
SUB R3
STORE R4
LOAD R1
DIV R4
STORE A
```

TEMA 3: RENDIMIENTO Y COSTE

Ejemplo: Supongamos que tenemos dos implementaciones distintas A y B de una misma arquitectura del repertorio de instrucciones.
La máquina correspondiente a la implementación A tiene un tiempo de ciclo de reloj de 1ns y un CPI igual a 2 para un programa concreto.
La máquina B tiene un tiempo de ciclo de 3ns y un CPI de 1,2 para el mismo programa.
¿
¿

$\left.\begin{matrix} A \\ B \end{matrix}\right\}$ Tiene el mismo repertorio de instrucciones (ISA + COMPILADOR + PROGRAMA) $\Longrightarrow$ TRADUCCIÓN NI

PROBLEMA Nº 4: Se desea mejorar el rendimiento de un computador introducir una tarjeta aceleradora de video que realice las operaciones en la mitad de tiempo.
Calcular la ganancia en velocidad del sistema para la ejecución de un programa si el 87% de las operaciones del mismo son operaciones gráficas.
Si el programa tarda 32 segundos en ejecutarse sin la mejora, ¿Cuánto tardará con la mejora?

$$tpo_ejec_mejo = tpo_ejec_cofig \cdot \left(1 - fracción_mejorada + \frac{frac.mejorada}{acele.\ mejora}\right)$$

$$tpo_mej = 32 + \left(1 - 0'87 + \frac{0'87}{2}\right)$$

Nota:
En el ejemplo pág 38 donde pone dados reales son dados defectuosos, que en este caso son:
Dados defectuosos = 58

$\Longrightarrow$ De potencia no suelen entrar problemas.

PROBLEMAS

1) Considere dos implementaciones diferentes, M1 y M2, del mismo conjunto de instrucciones. Existen 3 clases de instrucciones (A, B, C) en dicho conjunto de instrucciones. M1 tiene una frecuencia de reloj de 80MHz y M2 de 100MHz. El número medio de ciclos de reloj por cada clase de instrucción y sus frecuencias son las siguientes:

Clase de Instrucción	Ciclos/Instrucción M1	Ciclo/Instrucción M2	Frecuencia
A	1	2	60%
B	2	3	30%
C	4	4	10%

a) Calcular los CPI medios para cada máquina, M1 y M2.

Sol: M1: Ciclos/Instrucción $= \left(\frac{60}{100}\right)\cdot 1 + \left(\frac{30}{100}\right)\cdot 2 + \left(\frac{10}{100}\right)\cdot 4 = 1'6.$

M2: Ciclos/Instrucción $= \left(\frac{60}{100}\right)\cdot 2 + \left(\frac{30}{100}\right)\cdot 3 + \left(\frac{10}{100}\right)\cdot 4 = 2'5.$

Calcular las tasas de MIPS medias para cada máquina M1 y M2.

Sol: M1: Tasa promedio de MIPS = Frecuencia de reloj/(CPI·10^6) $= \frac{80\cdot 10^6}{1'6\cdot 10^6} = 50.$

M2: Tasa promedio de MIPS = Frecuencia de reloj/(CPI·10^6) $= \frac{100\cdot 10^6}{2'5\cdot 10^6} = 40.$

b) ¿Qué máquina tiene una tasa de MIPS más baja? ¿Qué clase de instrucción necesita variar su CPI, y en cuanto, para...

Sol: La máquina M2 tiene una tasa de MIPS más baja. Si cambiamos el CPI de la clase A para la máquina M2 a 1, tendremos una mejor tasa de MIPS para M2:

Ciclos/instrucción $= \left(\frac{60}{100}\right)\cdot 1 + \left(\frac{30}{100}\right)\cdot 3 + \left(\frac{10}{100}\right)\cdot 4 = 1'9.$

Tasa promedio de MIPS = frecuencia de reloj/(CPI·10^6) $= \frac{100\cdot 10^6}{1'9\cdot 10^6} = 52'6.$

2) Supongamos que tenemos una máquina...

a) Los gestores quieren que la máquina sea 4 veces más rápida...

Sol: Ley de Amdhal: Tiempo de ejecución después de mejora = (Tiempo de ejecución afectado por la mejora)/(Cantidad de mejora) + Tiempo de ejecución no afectado.

Asumiendo que inicialmente las multiplicaciones en punto flotante, las divisiones en punto flotante y el resto de instrucciones...

Aceler div $= \frac{1}{(1-0'2)\cdot\left(\frac{0'2}{3}\right)} = \cancel{1'87}\ 18'75$ ~~1'45~~ Acller mult $= \frac{1}{(1-0'5)\cdot\left(\frac{0'5}{8}\right)} = \cancel{33'3}\ 32.$

... (El ejercicio 3 le sigue a este) $\Longrightarrow$ Acel ambas $= \frac{1}{(1-0'7)\cdot\left(\frac{0'5}{8} + \frac{0'2}{3}\right)} = 25'8$

Ejercicio de la foto.

Se dispone de un computador con un solo núcleo...

- Alternativa A: ...
- Alternativa B: ...

Se pide responder de forma justificada a las siguientes cuestiones...

1.
2. Hacer la fórmula de la Ley de Amdhl.

$$\boxed{T_{CPU} = NI \cdot CPI \cdot TC}$$

$$T_{CPU\ máq\text{-}origen} = NI \cdot ((0'75 \cdot 12) + (0'25 \cdot 4)) \cdot TC = NI \cdot (9+1) \cdot TC = \boxed{NI \cdot 10 \cdot TC_E}$$

$$T_{CPU}\ A = NI \cdot CPI \cdot TC_A = NI \cdot ((0'75 \cdot 1'1 \cdot 12) + (0'25 \cdot 1'25 \cdot 4)) \cdot TC_A = NI \cdot 11'15 \cdot \frac{TC}{1'5} = \boxed{NI \cdot 7'43 \cdot TC}$$

(~~$T_{CPU}\ B = NI \cdot CPI \cdot TC_B = NI \cdot (\text{C})$~~

$$Am\ A = \frac{CPU \cdot origen}{CPU \cdot A} = 1'34$$

Nota: Serían iguales si no tuvieran relación entrada/salida.

$$\text{Aceleración global} = \frac{1}{(1-0'9) + \left(\frac{0'9}{1'34}\right)} = \frac{1}{0'1 + \frac{0'9}{1'34}} = \boxed{1'29}$$

Sol: con alterativa A:

LO MISMO PARA LA ALTERNATIVA DE B.

$$T_{CPU}\ B = NI \cdot CPI \cdot TC_B = \frac{NI}{4} \cdot ((0'75 \cdot 0'8 \cdot 12) + (0'25 \cdot 4)) \cdot \frac{TC}{0'5} =$$

$$Am\ B = \frac{CPU \cdot origen}{CPU \cdot B} =$$

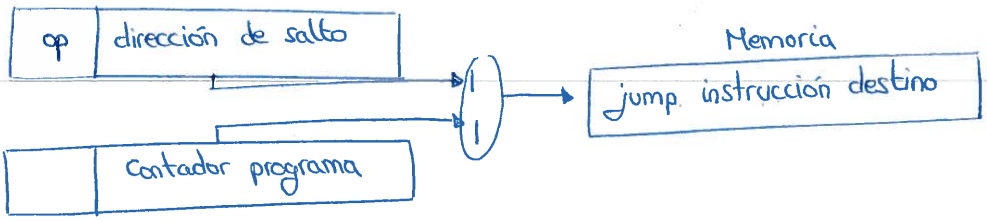
$$\text{Aceleración global} = \frac{1}{\quad} = \boxed{\quad}$$

Sol. con alternativa B

TEMA 5 : INSTRUCCIÓN MONOCICLOS.

(~~Ejercicio de examen~~)

¿Cómo se realiza el salto pseudo-directo?



(Sigue en diapositivas)

EJERCICIO 1:

Deseamos añadir al repertorio inicial de instrucciones de la CPU, modelo monociclo, la instrucción addi. Realizar las modificaciones necesarias en el datapath y en la unidad de control.

"No es necesario cambiar el hardware y lo que está en rojo" → Sol.

EJERCICIO 2:

Igual que la anterior pero con la instrucción bne que salta si no Z
beq salta si Z

Ejercicio 5:

Igual pero con la instrucción jm que funciona: PC ← Mem[R[rs]+ SignExt [inm16]].

(Esta dificultad es del tipo de ejercicios de examen)

⑥ a) Cuando R2 no es 0:

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
eti1:	LW R2, v(R1)	IF	ID	EX	ME	WB											
	BEQ R2, R0, eti2		IF	ID	ID	ID	EX	ME	WB								
	LW R3 w(R1)			–	–	~~IF~~	IF	ID	EX	ME	WB						
	SW v(R1), R3							IF	ID	EX	ME	WB					
	SW w(R1), R2								IF	ID	EX	ME	WB				
eti2:	SUBI R1, R1, #4									IF	ID	EX	ME	WB			
	BNE R1, R0, eti1										IF	ID	ID	EX	~~WB~~ ME	WB	
												–	–	IF	ID	EX	

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \cancel{\frac{15}{7} = 2'1}. \quad \frac{7 \text{ instrucciones} + 3 \text{ por datos} + 2 \text{ ciclos de saltos}}{7} = 1'71$$

Cuando R2 = 0.

		1	2	3	4	5	6	7	8	9	10	11	12	13
eti1:	LW R2 v(R1)	IF	ID	EX	ME	WB								
	BEQ R2, R0, eti2		IF	ID	ID	ID	EX	ME	WB					
Se quita con R2=0	~~LW R3, w(R1)~~													
	~~SW v(R1), R3~~													
	~~SW w(R1), R2~~													
eti2:	SUBI R1, R1, #4			–	–	–	IF	ID	EX	ME	WB			
	BNE R1, R0, eti1							IF	ID	ID	EX	ME	WB	
									–	–	IF	ID	EX	ME

$$CPI = \frac{\text{cidos}}{\text{instrucciones}} = \cancel{\frac{13}{4} = 2'25} = \frac{4 \text{ instrucciones} + 3 \text{ por datos} + 2 \text{ ciclos de saltos}}{4} = 2'25$$

Resultado: $\boxed{CPI = 0'8 \cdot 2'25 + 0'2 \cdot 1'71 = 2'14}$

b) Cuando $R2 \neq 0$.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LW R2 v(R1)	IF	ID	EX	ME	WB										
eti1: BEQ R2, R0, eti2		IF	ID	ID	ID	EX	ME	WB							
[SUBI R1, R1, #4]			IF	IF	IF	ID	EX	ME	WB						
LW R3, W(R1)						IF	ID	EX	ME	WB					
SW v(R1), R3							IF	ID	EX	ME	WB				
SW w(R1), R2								IF	ID	EX	ME	WB			
eti2: BNE R1, R0, eti1									IF	ID	EX	ME	WB		
[LW R2, v(R1)]										IF	ID	EX	ME	WB	
											IF	ID	EX	ME	WB

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \frac{7 \text{ instrucciones} + 2 \text{ por datos} + 0 \text{ ciclos por saltos}}{7} = 1'28$$

Cuando $R2 = 0$.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
LW R2 v(R1)	IF	ID	EX	ME	WB											
eti1: BEQ R2, R0, eti2		IF	ID	ID	ID	EX	ME	WB								
[SUBI R1, R1, #4]			IF	IF	IF	ID	EX	ME	WB							
eti2: BNE, R1, R0, eti1						IF	ID	ID	EX	ME	WB					

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \frac{4 \text{ instrucciones} + 3 \text{ por datos}}{4} = 2'33$$

Solución: $CPI = 0'8 \cdot 2'33 + 0'2 \cdot 1'128 = \boxed{1'66}$

2

LEY DE AMHBAR

$$\boxed{A = \frac{1}{(1-Fm)+\left(\frac{Fm}{Am}\right)}}$$

¿Cuál será el incremento en la velocidad si solo se mejora la multiplicación?

$Fm = 20\%$

$Am = 4$

$$A = \frac{1}{(1-4)+\left(\frac{0'2}{4}\right)} = \boxed{0'3279.}$$

¿Y si solo se mejoran los accesos a memoria?

$Fm = 50\% = 0'5$

$Am = 2$

$$A = \frac{1}{(1-2)+\left(\frac{0'5}{2}\right)} = \boxed{0'8}$$

¿Y si se realizan ambas mejoras?

$$A = \frac{1}{(1-6)+\left(\frac{0'7}{6}\right)} = \boxed{0'1954}$$

EXAMEN

56 | 8 → 0, 7 43 | 8

(5) 1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17

8 bloque LRU

1 | 8 → 1, 0 4 | 8 → 4, 0 8 | 8 → 0, 1 20 | 8 → 4, 2 17 | 8 → 1, 2 19 | 8 → 3, 2

Directa:

	1	4	8	5	20
0			M(8)	M(8)	M(8)
1	M(1)	M(1)	M(1)	M(1)	M(1)
2					
3					
4		M(4)	M(4)	M(4)	~~M(4)~~ M(20)
5				M(5)	M(5)
6					
7					

Fallo forzoso Fallo forzoso Fallo for Fallo for Fallo conflicto

	17	19	56
0	M(8)	M(8)	~~M(8)~~ M(56) conf
1	~~M(1)~~ M(17)	M(17)	~~M(17)~~ M(9) ~~conf~~ ~~Acierto~~ M(17) conflic
2			
3		M(19)	~~M(19)~~ ~~M(11)~~ conf M(43) conf
4	M(20)	M(20)	~~M(20)~~ M(4) capacidad
5	M(5)	M(5)	M(5) Acierto
6			M(6)
7			

Fallo conf Fallo forzoso Fallo confl

0	M(56)
1	M(17)
2	1
3	M(43)
4	M(4)
5	M(5)
6	M(6)

4-asociativa:

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

1) Ancho de palabra = 32 bits $\Rightarrow 2^5$

Registros = 32 bits $\Rightarrow 2^5$

Instrucciones = 64 $\Rightarrow 2^6 \rightarrow$ Cod. operando

TEMA 7

③ 16 bloques de 1 palabra cada uno

22, 26, 22, 26, 16, 3, 16, 18

a) Correspondencia directa:

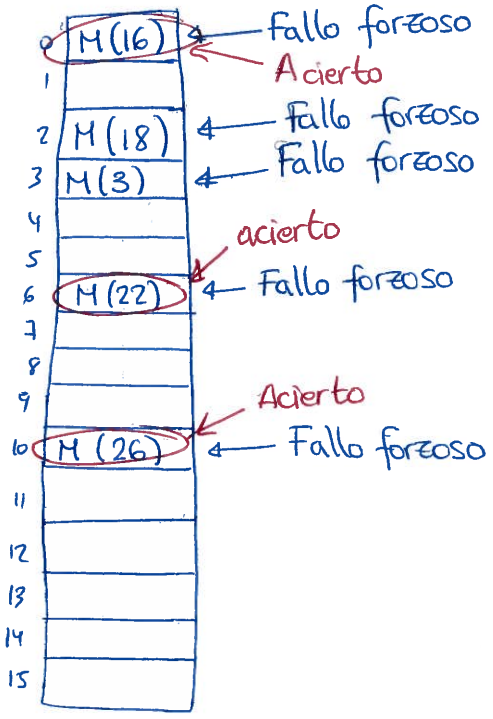
$22 \,|\, 16$ → 6 / 1

$26 \,|\, 16$ → 10 / 1

$16 \,|\, 16$ → 0 / 1

$3 \,|\, 16$ → 3 / 0

$18 \,|\, 16$ → 2 / 1



Correspondencia directa: 16 accesos a caché y 5 fallos ~~y 3 aciertos~~.

b) Directa con tamaño de bloque de 4 palabras.

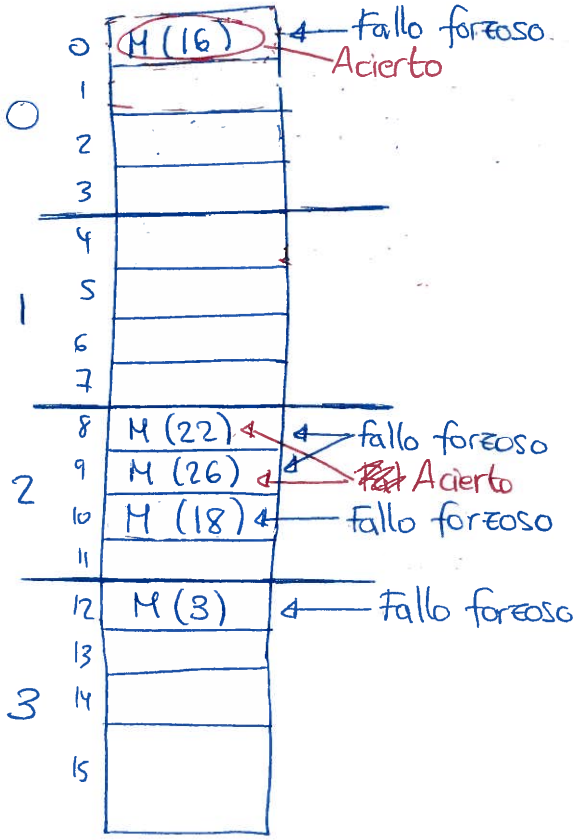
$22 \,|\, 4$ → 2 / 5

$26 \,|\, 4$ → 2 / 6

$16 \,|\, 4$ → 0 / 4

$3 \,|\, 4$ → 3 / 0

$18 \,|\, 4$ → 2 / 4



② 4 bloques LRU

0, 8, 0, 6, 8

a) Directa:

0 | 4 → 0, 0 8 | 4 → 0, 2 6 | 4 → 2, 1

0	~~M(0)~~
1	
2	M(6)
3	

← Fallo forzoso

~~M(8)~~ ← fallo conflicto

~~M(0)~~ ← Fallo conflicto

M(8) ← Fallo conflicto.

← fallo forzoso

b) Asociativo por conjunto de 2 vías:

0 | 2 → 0, 0 8 | 2 → 0, 4 6 | 2 → 0, 3

0	M(0)	
1	M(8)	
2	M(6)	
3		

Acierto

Fallo forzoso

Acierto

Fallo forzoso

Acierto

~~M(0)~~

~~M(8)~~

Fallo forzoso

M(8) ← Fallo conflicto

M(6) ← Fallo conflicto

c) Totalmente asociativo:

0	M(0)
1	M(8)
2	M(6)
3	

Acierto

Fallo Forzoso

Fallo forzoso

NOTA: Fallo forzoso: cuando el hueco estaba vacío.

Fallo conflicto: cuando el hueco ya estaba ocupado y se sustituye.

Fallo capacidad: cuando el hueco ya vuelve a ser sustituido por el inicial

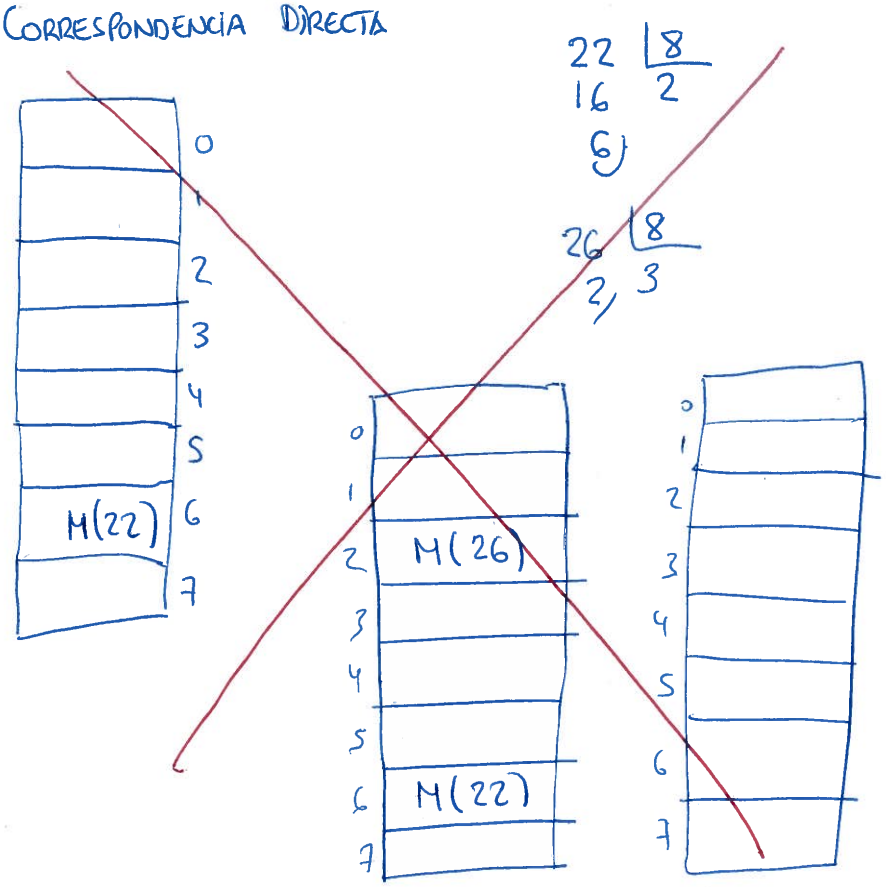
M(1)	M(2)	M(1)

LRU: reemplazo por el anterior del anterior.

FIFO: Cola

LIFO: Pila.

CORRESPONDENCIA DIRECTA

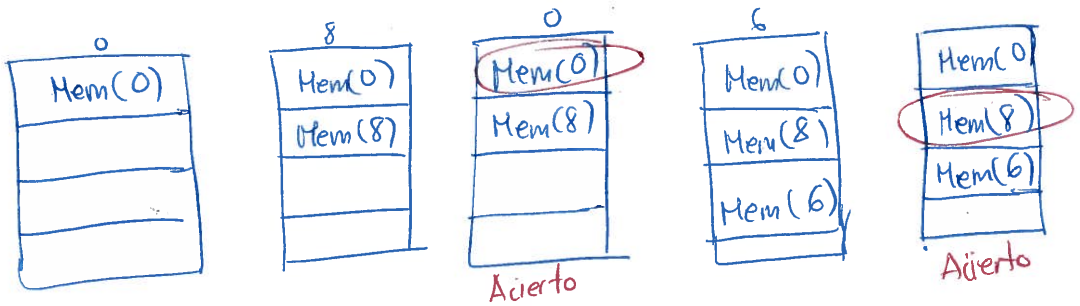


EJERCICIO 2

TOTALMENTE ASOCIATIVA con LRU → (anterior del anterior)

FIFO → (cola) LIFO → (Pila)

0, 8, 0, 6, 8



Ejercicio 1

CORRESPONDENCIA DIRECTA:

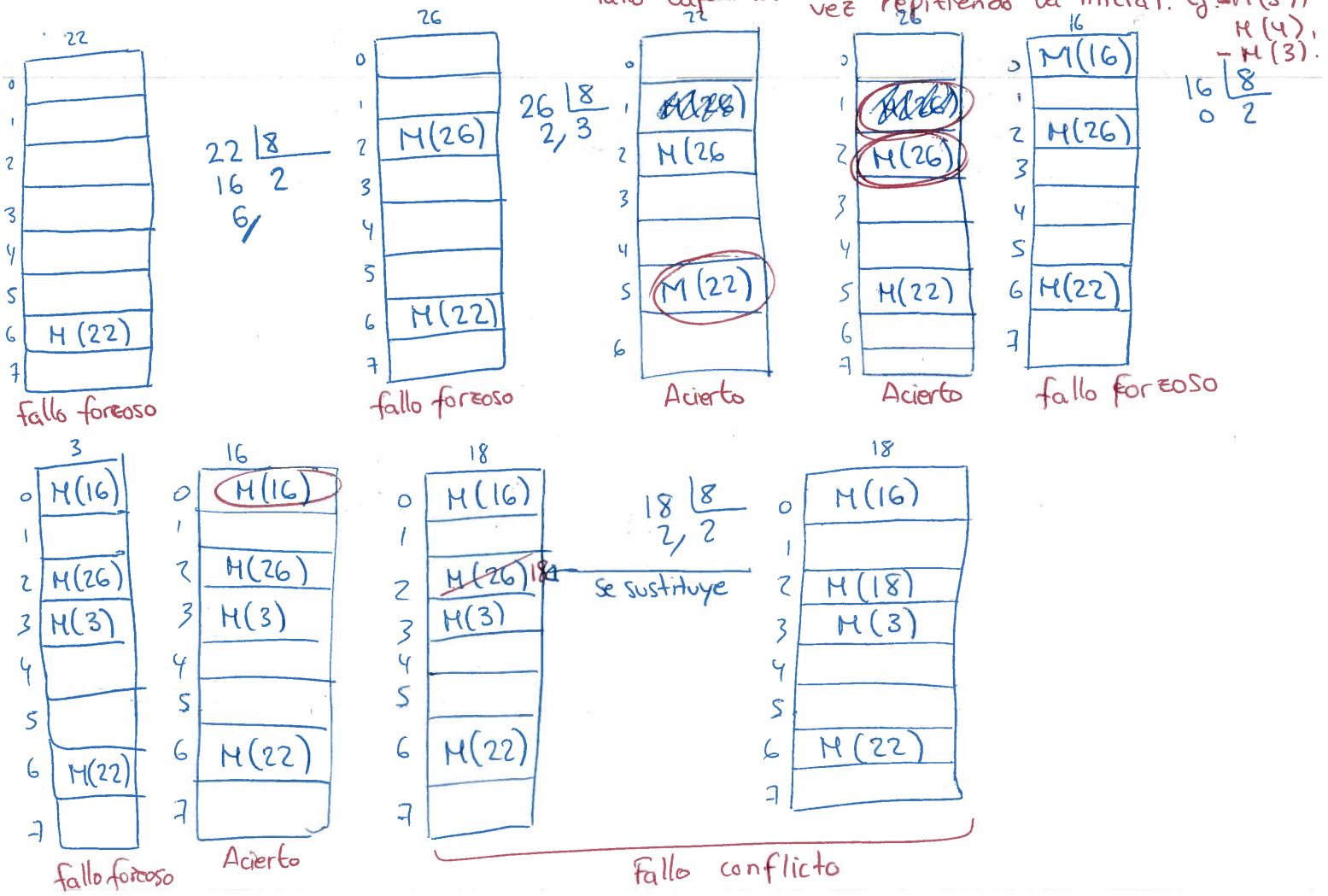
8 bloques.

22, 26, 22, 26, 16, 3, 16, 18

Fallo forzoso: cuando no hay nada en la celda y añadimos algo.

Fallo conflicto: cuando hay que sustituir porq ya está usado

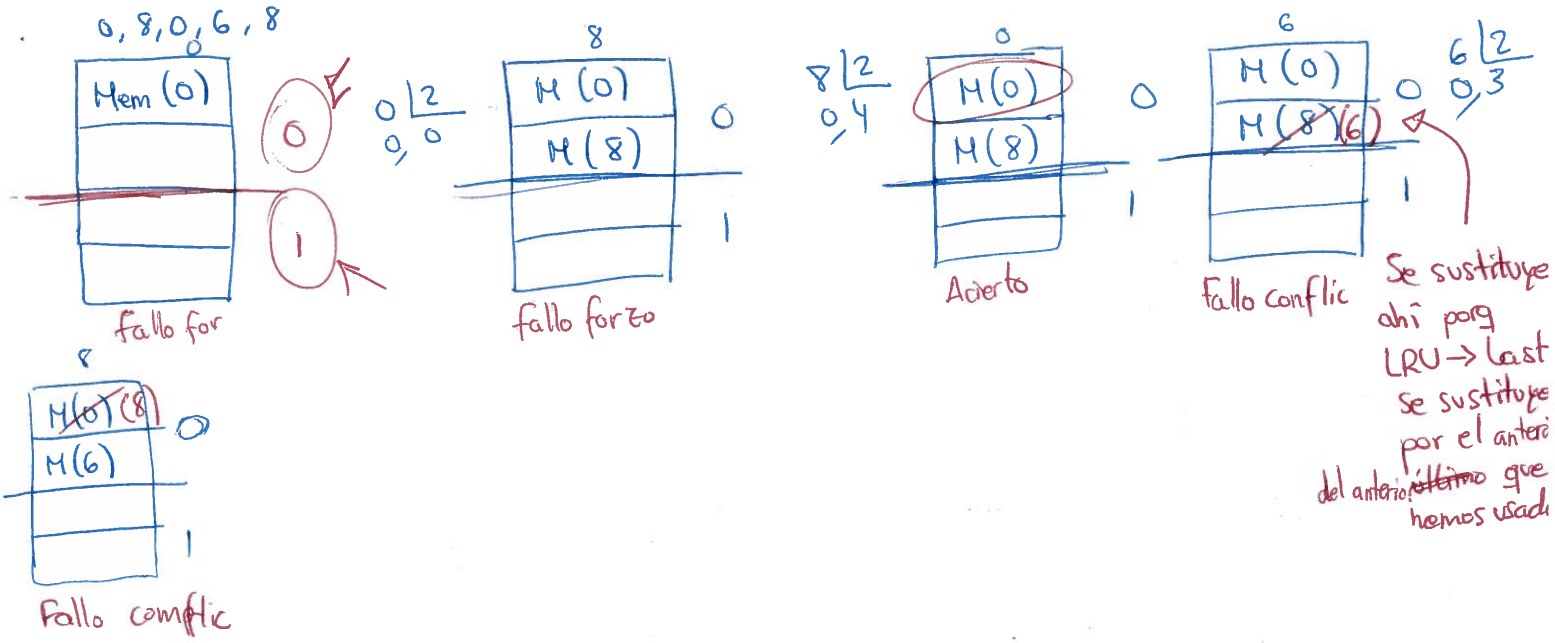
fallo capacidad: cuando sustituimos más de una vez repitiendo la inicial. Ej: M(3), M(4), M(3).



EJERCICIO 2:

ASOCIATIVO POR CONJUNTO DE 2 VÍAS (Dividir la caché en 2 partes)

0, 8, 0, 6, 8



TEMA 6.

1) a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
SUB R1, R2, R3	IF	ID	EX	ME	WB													
ADD R4, R5, R6		IF	ID	EX	ME	WB												
SUB R5, R4, R8			IF	ID	ID	ID	EX	ME	WB									
ADD R7, R2, R3				IF	IF	IF	ID	EX	ME	WB								
ADD R9, R7, R3							IF	ID	ID	ID	EX	ME	WB					
LW R1, 10(R6)								IF	IF	IF	ID	EX	ME	WB				
ADD R3, R1, R4											IF	ID	ID	ID	EX	ME	WB	
SUB R6, R7, R8												IF	IF	IF	ID	EX	ME	WB

a) 18 ciclos.

b)

	1	2	3	4	5	6	7	8	9	10	11	12	13
SUB R1, R2, R3,	IF	ID	EX	ME	WB								
ADD $\boxed{R4}$, R5, R6.		IF	ID	$\boxed{EX}$	ME	WB							
SUB R5, $\boxed{R4}$, R8			IF	ID	$\boxed{EX}$	ME	WB						
ADD $\boxed{R7}$, R2, R3				IF	ID	$\boxed{EX}$	ME	WB					
ADD R9, $\boxed{R7}$ R3					IF	ID	$\boxed{EX}$	ME	WB				
LW $\boxed{R1}$, 10 (R6)						IF	ID	EX	$\boxed{ME}$	WB			
ADD R3, $\boxed{R1}$, R4							IF	ID	ID	$\boxed{EX}$	ME	WB	
SUB R6, R7, R8								IF	IF	ID	EX	ME	WB

b) 13 ciclos.

c) Intercambiando las dos últimas instrucciones, así la dependencia desaparece y el ciclo tarda 12 ciclos.

2) a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
SUB R1, R2, R3	IF	ID	EX	ME	WB												
ADD R4, R5, R6		IF	ID	EX	ME	WB											
OR R5, R1, R4			IF	ID	ID	ID	EX	ME	WB								
SW R5, 10(R6)				IF	IF	IF	ID	ID	ID	EX	ME	WB					
LW R1, 20(R8)							IF	IF	IF	ID	EX	ME	WB				
AND R3, R8, R5										IF	ID	EX	ME	WB			
ADD R3, R1, R7											IF	ID	ID	ID	EX	ME	WB

a) 17 ciclos.

(duda)

b)

	1	2	3	4	5	6	7	8	9	10	11	12
SUB R1, R2, R3	IF	ID	EX	ME	WB							
ADD $\boxed{R4}$, R5, R6		IF	ID	$\boxed{EX}$	ME	WB						
OR $\boxed{R5}$, R1, $\boxed{R4}$			IF	ID	$\boxed{EX}$	$\boxed{ME}$	WB					
SW $\boxed{R5}$, 10 (R6)				IF	ID	EX	$\boxed{ME}$	WB				
LW R1, 20 (R8)					IF	ID	EX	$\boxed{ME}$	WB			
AND $\boxed{R3}$, R8, R5						IF	ID	EX	ME	WB		
ADD $\boxed{R3}$, R1, R7							IF	ID	$\boxed{EX}$	ME	WB	

b) 11 ciclos.

Tema 3

PROBLEMA Nº 1 (Ejercicio 3)

Si una máquina X ejecuta un programa en 10 segundos y una máquina Y ejecuta el mismo programa en 15 segundos. ¿Cuánto más rápida es X respecto a Y?

RESPUESTA:

Se sabe que una máquina X es n veces más rápida que otra Y si:

$$\frac{\text{Rendimiento } X}{\text{Rendimiento } Y} = \frac{\text{Tiempo de ejecución } Y}{\text{Tiempo de ejecución } X} = n.$$

Con lo que:

$$\frac{\text{Tiempo de ejecución } Y}{\text{Tiempo de ejecución } X} = \frac{15}{10} = 1'5 \text{ seg.}$$

Sol: La máquina X es 1'5 veces más rápida que Y.

(Ejercicio 4)

Igual pero con $X=20$ y $Y=15$

$$\frac{15}{20} = 0'75 \text{ Seg}$$

PROBLEMA Nº 2 (Ejercicio 5)

Se tiene un programa en un computador X que tarda 35 segundos en ejecutarse mientras que en el computador Y emplea 21 segundos. Se sabe que el programa está formado por 522 millones de instrucciones. ¿Cuánto más rápido es el computador Y que el X? ¿Qué cantidad de instrucciones por segundo (MIPS) ejecuta cada computador?

RESPUESTA:

a) Como en el anterior: $\frac{35}{21} = 1'667s$ donde el Y es 1'667 más rápido que el X.

b) Se ejecutan 522×10^6 instrucciones ——— n segundos
X instrucciones ——— 1 segundo

Donde el nº de instrucciones por segundos es:

$$X \text{ instrucciones} = \frac{522 \cdot 10^6 \cdot 1s}{ns}$$

Para el computador X:

$$\frac{522 \cdot 10^6 \cdot 1s}{35\,s} = 14'914 \cdot 10^6 \text{ instrucciones/seg}$$

Para Y:

$$\frac{522 \cdot 10^6 \cdot 1s}{21\,seg} = 24'857 \cdot 10^6 \text{ instrucciones/s}$$

Como se trata de calcular cuantos millones de instrucciones por segundo ejecuta cada uno hay que dividir por 10^6.

Con el computador $\boxed{\text{MIPS } X = 14'914 \;;\; \text{MIPS } Y = 24'857}$

PROBLEMA Nº 3 (Ejercicio 7)

	CPI
Tipo 1	1
Tipo 2	2
Tipo 3	3

	Total Instruc		
	Tipo 1	Tipo 2	Tipo 3
Prog 1	2	1	2
Prog 2	4	1	1

a) Programa que ejecuta mayor número de instrucciones.

b) Nº de ciclos que tarda en ejecutarse cada programa.

c) CPI para cada programa.

RESPUESTA:

a) Programa 1: $2+1+2=5$ instrucciones

Programa 2: $4+1+1=6$ instrucciones

b) Los ciclos de reloj de CPU es la suma del CPI de cada tipo de instrucción:

Ciclos de reloj CPU1 $= 2 \cdot 1 + 1 \cdot 2 + 2 \cdot 3 = 10$ ciclos.

Ciclos de ~~reloj~~ CPU2 $= 4 \cdot 1 + 1 \cdot 2 + 1 \cdot 3 = 9$ ciclos.

c) $$CPI1 = \frac{\text{Ciclos de reloj CPU1}}{\text{nº instruc 1}} = \frac{10}{5} = 2$$

$$CPI2 = \frac{9}{6} = 1'5$$

Gana el Programa 2.

(Ejercicio 8)

Sea una arquitectura cuyo repertorio de instrucciones está formado por cuatro tipos de instrucciones cuyas medidas de CPI se muestran en la siguiente tabla:

	CPI para cada tipo instruc
Tipo 1	1
Tipo 2	2
Tipo 3	3
Tipo 4	4

Se nos pide que evaluemos dos compiladores diferentes con vista a adquirir uno de ellos para ~~cada~~ nuestro departamento de desarrollo. Compilando un mismo programa en cada compilador se obtienen los siguientes valores (en miles de millones para cada tipo de instrucción).

Tabla:

	en miles de millones			
	Tipo 1	Tipo 2	Tipo 3	Tipo 4
Comp 1	5	1	1	2
Comp 2	10	1	1	4

Si la frecuencia de reloj de los computadores que tenemos es de 1'2 GHz:

a) Ejecutable más rápido en cuanto a tiempo de ejecución.

b) Ejecutable más rápido tomando como medida los MIPS.

Respuesta:

Comp 1 = 5+1+1+2 = 9 instrucciones

Comp 2 = 10+1+1+4 = 16 instrucciones

Ciclos de CPU1 = 5·1 + 1·2 + 1·3 + 2·4 = 18 ciclos

Ciclos de CPU2 = 10·1 + 1·2 + 1·3 + 4·4 = 31 ciclos

$1 \cdot CPI = \frac{18}{9} = 2$ ciclos/instr ; $2CPI = \frac{31}{16} = 1'93$ ciclos/inst

b) $MIPS = \frac{Fc}{CPI \cdot 10^6} \Rightarrow$ $MIPS\ Comp1 = \frac{1'2}{2 \cdot 10^6} = 6 \cdot 10^{-7}$; $MIPS\ Comp2 = \frac{1'2}{1'93 \cdot 10^6} = 6'21 \cdot 10^{-7}$

El más rápido 1

a) Tiempo de ejec $= \frac{n^o\ inst}{MIPS \cdot 10^6} \Rightarrow$ $Te1 = \frac{9}{6 \cdot 10^{-7} \cdot 10^6} = 15$

$Te2 = \frac{16}{6'21 \cdot 10^{-7} \cdot 10^6} = 25'76$

El más rápido 1

(Ejercicio 10)

	CPI
Tipo 1	2
Tipo 2	3
Tipo 3	1
Tipo 4	6
Tipo 5	3

Se desean evaluar tres traductores de COBOL 400 a RPG-II con vistas a adquirir uno de ellos para la empresa en la que trabajamos. Para ello, se ha desarrollado un programa en COBOL 400 que ha sido traducido por cada uno de los tres traductores a RPG-II: Tabla:

	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Tipo 5
Traductor 1	6	3	2	2	3
Traductor 2	8	2	1	3	1
Traductor 3	6	3	1	1	9

Si la fc = 1'8GHz:

a) Ejecutable más rápido en cuanto a tiempo de ejecución.

b) Ejecutable más rápido tomando como medida los MIPS.

RESPUESTA:

Nº instr(Tra1) = 6+3+2+2+3 = 16 instr

Nº int (Tra2) = 8+2+1+3+1 = 15 instr

Nº ins(Tra3) = 6+3+1+1+9 = 20 instr

Ciclos CPU(Trad1) = 6·1+3·2+2·3+2·4+3·5 = 44 ciclos

Ciclos CPU(2) = 8·1+2·2+1·3+3·4+1·5 = 44 ciclos

Ciclos CPU(3) = 6·1+3·2+1·3+1·4+9·5 = 55 ciclos

$CPI = \frac{\text{Ciclo reloj}}{\text{Nº inst}}$

$CPI(1) = \frac{44}{16} = 2'75$

$CPI(2) = \frac{44}{15} = 2'93$

$CPI(3) = \frac{55}{20} = 2'75$

b) $MIPS(1) = \frac{1'8}{2'75 \cdot 10^6} = 6'54 \cdot 10^{-7}$

$MIPS(2) = \frac{1,8}{2'93 \cdot 10^6} = 6'14 \cdot 10^{-7}$ ✓

$MIPS(3) = \frac{1'8}{2'75 \cdot 10^6} = 6'54 \cdot 10^{-7}$

a) $Te = \frac{\text{Nº inst}}{MIPS \cdot 10^6} \Rightarrow Te(1) = \frac{16}{6'54 \cdot 10^{-7} \cdot 10^6} = 24'46$

$Te(2) = \frac{15}{6'14 \cdot 10^{-7} \cdot 10^6} = 24'42$ ✓

$Te(3) = \frac{20}{6'54 \cdot 10^{-7} \cdot 10^6} = 30'58$

PROBLEMA Nº 4 (Ejercicio 9)

Se desea mejorar el rendimiento de un computador introduciendo una tarjeta aceleradora de video que realice las operaciones en la mitad de tiempo.

a) Calcular la ganancia en velocidad del sistema para la ejecución de un programa si el 87% del mismo se dedica a operaciones gráficas.

b) Si el programa tarda 32 segundos en ejecutarse sin la mejora. ¿Cuánto tardará con la mejora?

RESPUESTA:

a) Debemos aplicar la ley de Amdahl.

$$\boxed{A = \frac{1}{(1-Fm) + \frac{Fm}{Am}}} \qquad A = \frac{1}{(1-0'87) + \frac{0'87}{2}} = 1'7699$$

$1'7699 \cdot 100 = 76'99\%$

$Fm = 0'87$

$Am = 2$

Por lo que el nuevo sistema será un 76'99% más rápido

b) $A = \frac{\text{Tiempo ejecucion sin mejora}}{\text{" " con "}} \Rightarrow 1'7699 = \frac{32}{Te \text{ con mej}}$

$\text{Tiemp ejec con mejora} = \frac{32}{1'7699} = 18'08 \text{ seg}$

Con la mejora tardará 18'08 segundos en ejecutarse.

(Ejercicio 12) Se desea mejorar el rendimiento de un computador introduciendo un coprocesador matemático que realice las operaciones en mitad de tiempo.

a) Calcular ganancia en velocidad del sistema para la ejecución de un programa si el 96% del mismo se dedica a operaciones aritméticas.

b) Si el programa tarda 15 segundos en ejecutarse sin la mejora ¿Con ella?

RESPUESTA:

a) $A = \frac{1}{(1-Fm) + \frac{Fm}{Am}} \Rightarrow A = \frac{1}{(1-0'96) + \left(\frac{0'96}{2}\right)} = 1'92$ El nuevo será un 92% más rápido

b) $A = \frac{\text{sin}}{\text{con}} \Rightarrow 1'92 = \frac{15 \text{ seg}}{\text{con}} \Rightarrow \text{con} = \frac{15 \text{ seg}}{1'92} = 7'81 \text{ seg}$

Con la mejora tarda 7'81 seg en ejecutarse.

PROBLEMA Nº5 (Ejercicio 11)*

(Ejercicio 13)

Tipo instrucción	Porcentaje de empleo	Factor de mejora
Suma	30% 30%	5 10
Salto cond	34% 55%	4 2
Carga-almac	32% 12%	2 8
Resto de instruc	4% 3%	7 10

a) ¿Qué mejora recomiendas?

b) Si un programa tardaba antes de la mejora 28'3 seg 37'02 seg en ejecutarse calcule cuánto tardará con la mejora que hemos elegido antes.

RESPUESTA:

a) Suma $Fm = 30\%$ $Am = 5$ $A = \frac{1}{(1-Fm)+\left(\frac{Fm}{Am}\right)} = 1'31$

Salto Cond $Fm = 34\%$ $Am = 4$ $A = 1'34$

Carga-alma $Fm = 32\%$ $Am = 2$ $A = 1'19$

Resto de ins $Fm = 4\%$ $Am = 7$ $A = 1'03$

La mayor ganancia se da en Salto condicional.

b) $A = \frac{Sin}{Con} \Rightarrow 1'34 = \frac{28'3}{Te\ con} \Rightarrow Te\ Con = \frac{28'3}{1'34} = 22'119$ seg

El tiempo modificado con las inst de salto condicional será 22'119 seg.

PROBLEMA 4:

Saltos hacia atrás: $\frac{17+31+26}{3} = 25\%$

Saltos efectivos: $\frac{54+51+54}{3} = 53\%$

Total bifurcaciones: $\frac{70+63+63}{3} = 65\%$

$$53\% = \left(\left(\cancel{25\% + 65\%}\ 90\%\right) \times 25\%\right) + \left(\left(\text{delante}\right) \times 75\%\right)$$

Delante

$53 = 22'5 + (x \cdot 75)$

$53 - 22'5 = x \cdot 75 \quad ; \quad x = \frac{-22'5 + 53}{75} = 40'7\%$

PROBLEMA 1:

Palabra de 32 bits

16 Reg de 32 bits

Memoria 256 MPalabras

2 Instrucciones

$2^0 = 1 \quad 2^1 = 2 \quad 2^2 = 4 \quad 2^3 = 8 \quad 2^4 = 16$

MOVE:

C.Op	M. Direc	Dato	N usa	M.Di	Dato
(1)	(3)	(28)	(1)	(3)	(28)

ADD

C.OP	M.Dir1	Dato	N	MD2	Dato	N	MD3	Dato
(1)	(3)	(28)	(1)	(3)	(28)	(1)	(3)	(28)

MOD	Inmediato o a memoria
(3)	(28)

MD	Reg	No usado
(3)	(4)	(24)

MD	Reg	Desplazamiento
(3)	(4)	(24)

MD	pre	pos	reg	No usar
(3)	(1)	(1)	(4)	(22)

TEMA 2

Objetivo del diseñador:

(−) Coste

(+) Rendimiento

Clasificación ISA:

Según el tipo de almacenamiento interno de la CPU:

- Arquitectura tipo PILA.
- Arquitectura tipo ACUMULADOR.
- Arquitectura tipo GRP
  - Registro-Memoria
  - Registro-Registro
  - Memoria-Memoria

• PROBLEMAS

1. Palabras de 32 bits.

   16 registros de palabras de 32 bits.

   Memoria de 256 Mpalabras.

   2 Instrucciones.

---

③ TOTAL = 36 bits

- 7 instrucciones con dos direcciones de 15 bits y una de 3 bits
- 500 " de una " de 15 bits y 1 de 3 bits
- 50 " de 0 direcciones.

$2^0 = 1$

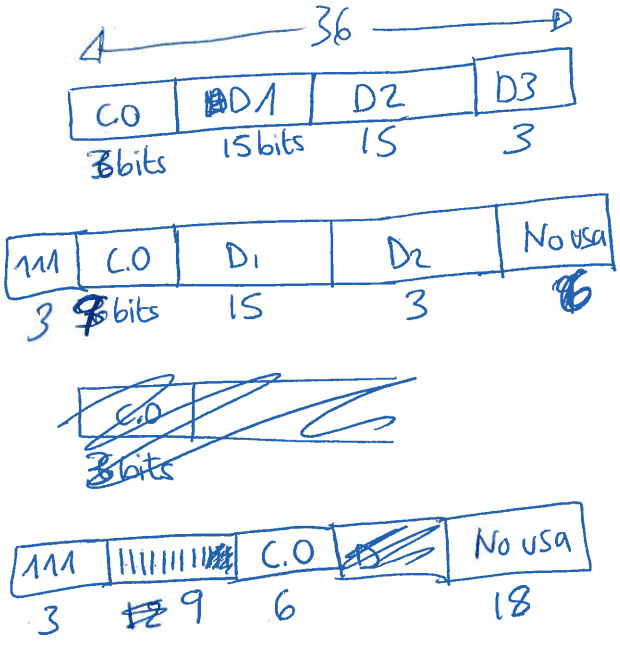
$2^1 = 2$

$2^2 = 4$

$2^3 = 8$

$2^4 = 16$

16 8 4 2 1



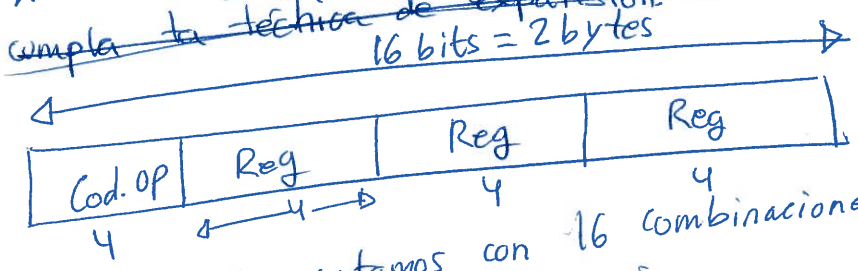
2. 16 Registros $\lambda = 2$ bytes Registro – Registro

- 15 inst de 3 operandos
- 14 " de 2 "
- 31 " " 1 "
- 16 " " 0 "

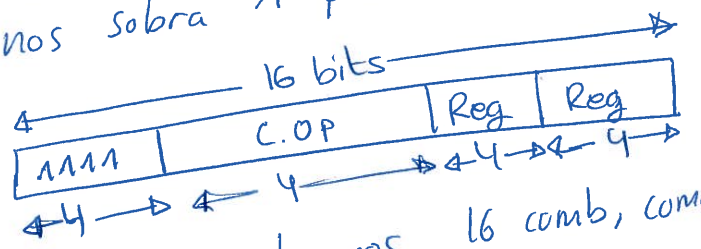
$\boxed{1 \text{ byte} = 8 \text{ bits}}$

$\lambda = 16$ bits y para diferenciar 16 registros necesitamos 4 bits ~~para que se cumpla la técnica de expansión~~

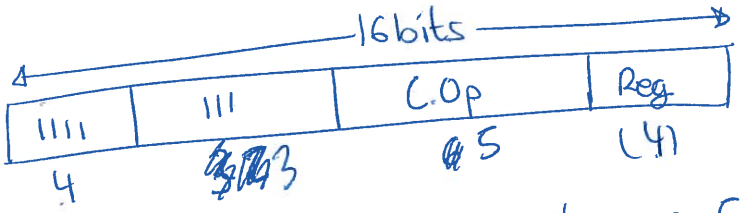
0000



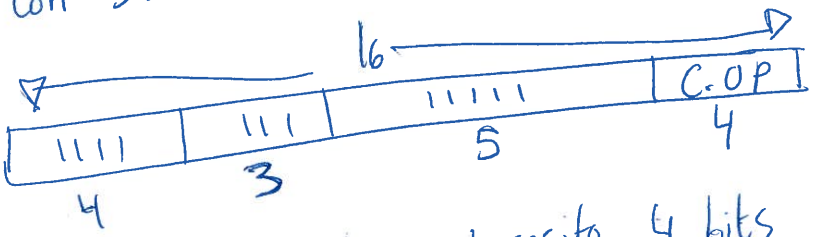
Con 4 bits contamos con 16 combinaciones, como empleamos 15 instruci nos sobra 1 para distinguir.



Con 4 bits tenemos 16 comb, como cogemos 14 instruc sobran 2: 1110 y 1111;



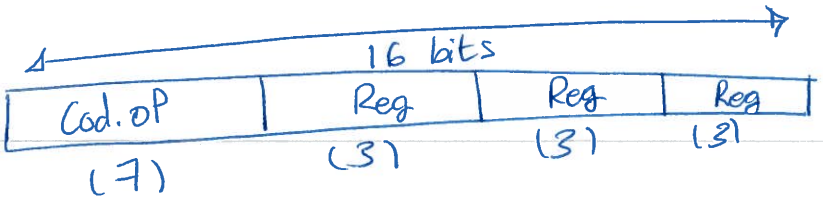
Con 31 combinaciones necesitamos 5 bits



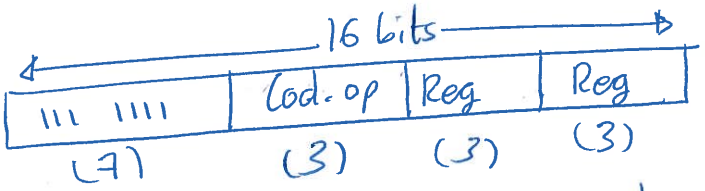
Con 16 combinaca necesito 4 bits

PROBLEMA 5:

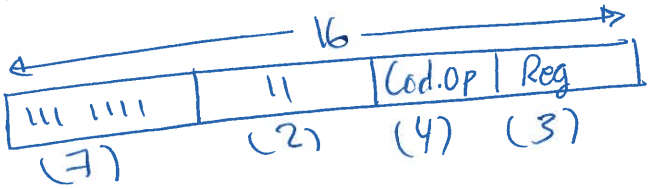
8 registros ⇒ 3 bits ; 2bytes = λ = 16 bits.



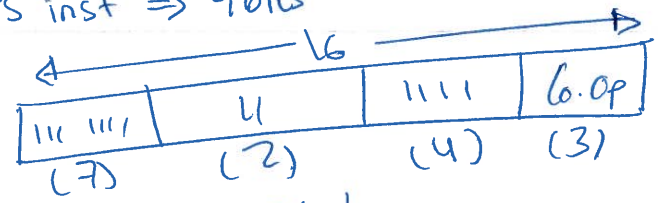
para 127 intruc necesi 7 bits nos sobra 1.



para 6 inst necesito 3 bits sobran 2: 110 111



15 inst ⇒ 4bits



8 inst ⇒ 3bits

① ESTADO 0:

ESTADO 0:

PCWrite	1
IorD	0
MemRead	1
IRWrite	1
ALUSelA	0
ALUSelB	01
ALUOp	00
* PC Source	00

→

ESTADO 1:

ALUSelA	0
ALUSelB	11
ALUOp	00

*

suma →	00
resta →	01
instrucción →	00

② BLEZ, BGTZ, BNE:

ESTADO 0:

PCWrite	1
IorD	0
MemRead	1
IRWrite	1
ALUSelA	0
ALUSELB	01
ALUOP	00
PCSource	00

→

ETAPA 1:

ALUSel_A	0
ALUSel_B	11
ALUOp	00

BNE →

ETAPA 2:

ALUSel A	1
ALUSel B	00
ALUOp	01
PCSource	01
Sel	01
PCWriteCond	1

BLEZ →

ETAPA 2:

ALUSel A	1
ALUSel B	00
ALUOp	01
Sel	10
PCSource	01
PCWriteCond	1

BGTZ ↓

ETAPA 2:

ALUSel A	1
ALUSel B	00
ALUOp	01
Sel	11
PCSource	01
PCWrite	1

1

$$\frac{\text{Rendimiento } x}{\text{Rendimiento } y} = \frac{\text{Tiempo de ejecución } y}{\text{Tiempo de ejecución } x} = n$$

$$\frac{\text{Tiempo ejecucion } y}{\text{Tiempo ejecucion } x} = \frac{15}{10} = 1'5$$

X es 1'5 veces más rápida que Y.

2

~~X = 35~~

$$\left.\begin{array}{l} \text{Tej } X = 35 \text{ seg} \\ \text{Tej } Y = 21 \text{ seg} \end{array}\right\} \text{Programa}$$

nº instrucciones = 522 millones

a) $$\frac{\text{Tej } X}{\text{Tej } Y} = \frac{35}{21} = 1'667$$ veces es más rápido Y que X.

b)

$522 \cdot 10^6$ instrucciones ——— n seg

X instrucciones ——— 1 seg

$$x \text{ instrucciones} = \frac{522 \cdot 10^6 \cdot 1}{n}$$

Con el X:

$$X = \frac{522 \cdot 10^6 \cdot 1}{35} = 14'914 \cdot 10^6 \text{ instrucciones/sg}$$

Con el Y:

$$X = \frac{522 \cdot 10^6 \cdot 1}{21} = 24'857 \cdot 10^6 \text{ int/sg}$$

(5)

$Fm = 30\%$
$Am = 10$

$$A = \frac{1}{(1-0'3) + \left(\frac{0'3}{10}\right)} = 1'37$$

$Fm = 55\%$
$Am = 2$

$$A = \frac{1}{(1-0'55) + \left(\frac{0'55}{2}\right)} = 1'38$$

$Fm = 12\%$
$Am = 8$

$$A = \frac{1}{(1-0'12) + \left(\frac{0'12}{8}\right)} = 1'12$$

$Fm = 3\%$
$Am = 10$

$$A = \frac{1}{(1-0'03) + \left(\frac{0'03}{10}\right)} = 1'03$$

a) ~~Re~~ La mejor ganancia se da en las instrucciones de salto condicional

b)

$$A = \frac{\text{tiempo Ejecucion Sin Mejora}}{\text{Con Mejora}} \Rightarrow 1'38 = \frac{37'02}{\text{Con mejora}}$$

$$\text{Con mejora} = \frac{37'02}{1'38} = 26'826 \text{ sg}$$

(12)

$$A = \frac{1}{(1-0'96) + \left(\frac{0'96}{2}\right)} = \boxed{1'92}$$

$Fm = 96\%$
$Am = 2$

a) La ganancia es de un 92% más rápido

b)

$$A = \frac{\text{Sin mejora}}{\text{con mejora}} \Rightarrow 1'92 = \frac{15s}{x}$$

$$x = \frac{15}{1'92} = 7'81s$$

Con la mejora el sistema tardará en ejecutarse 7'82s

③

a) Programa 1: $2+1+2=5$ instrucciones

Programa 2: $4+1+1=6$ instrucciones

b)

Ciclos de reloj CPU programa 1: $(2\cdot1)+(1\cdot2)+(2\cdot3)=10$ ciclos

Ciclos de reloj CPU programa 2: $(4\cdot1)+(1\cdot2)+(1\cdot3)=9$ ciclos

c)

$$CPI_{programa_1} = \frac{\text{Ciclos de reloj CPU Programa 1}}{\text{Número de instrucciones}} = \frac{10}{5} = 2$$

$$CPI\ programa2 = \frac{9}{6} = 1'5$$

Con lo que el programa 2 es el ganador.

④

a) $$A = \frac{1}{(1-Fm)+\left(\frac{Fm}{Am}\right)} = \frac{1}{(1-0'87)+\left(\frac{0'87}{2}\right)} = 1'7699$$

El nuevo sistema será un 76'99 % más rápido

b) $$A = \frac{\text{Tiempo de ejecución Sin Mejora}}{\text{Con mejora}} \Rightarrow 1'7699 = \frac{32}{\text{Con mejora}}$$

$$\text{Con mejora} = \frac{32}{1'7699} = 18'08\ sg$$

Con mejora el programa tardará 18'08 sg en ejecutarse.

2, 3, 11, 16, 21, 13, 64, 48, 19, 11, 3, 22, 4, 27, 6, 11

0	~~M(16)~~ ~~M(64)~~ M(48)
1	
2	M(2)
3	~~M(3)~~ ~~M(19)~~ M(3)
4	M(4)
5	M(21)
6	~~M(22)~~ M(6)
7	
8	
9	
6	
11	~~M(11)~~ ~~M(27)~~ M(11)
12	
13	M(13)
14	
15	

2 → fallo forzoso
3 → fallo forzoso
11 → fallo forzoso
16 → fallo forzoso
21 → fallo forzoso
13 → fallo forzoso
64 → fallo conflicto
48 → fallo conflicto
19 → fallo conflicto
11 → Acierto
3 → Fallo capacidad
22 → fallo forzoso
4 → fallo forzoso
27 → fallo conflicto
6 → fallo conflicto
11 → fallo capacidad

16 Accesos a memoria y 15 fallos

2, 3, 11, 16, 21, 13, 64, 48, 19, 11, 3, 22, 4, 27, 6, 11

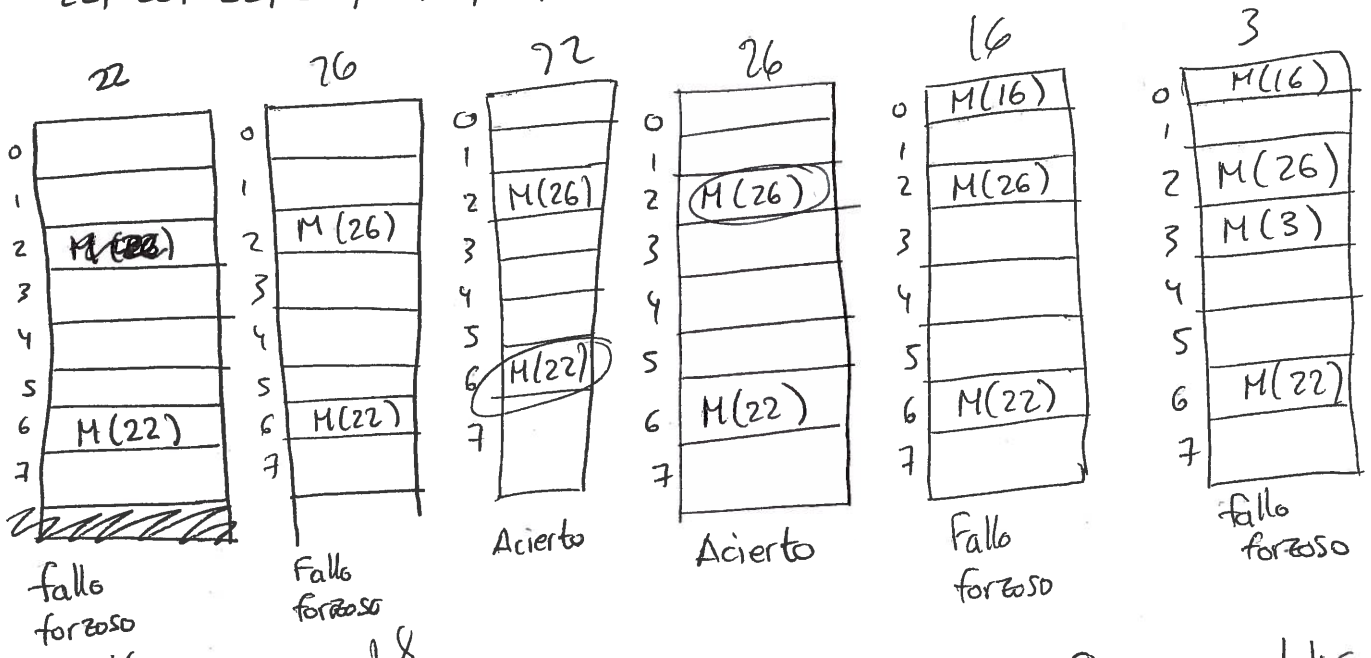
0	0	M(16)
	1	M(64)
	2	M(48)
	3	M(4)
1	4	M(21)
	5	M(13)
	6	
	7	
2	8	M(2)
	9	M(22)
	10	M(6)
3	12	M(3)
	13	M(11)
	14	M(19)
	15	M(27)

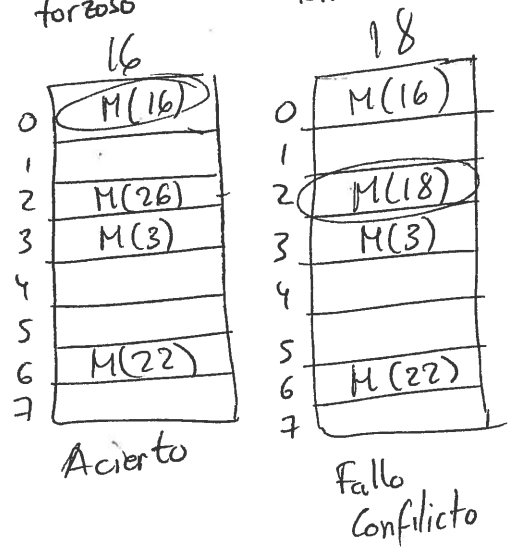
16 accesos a cadné y 13 fallos

2 → fallo forzoso
3 → fallo forzoso
11 → fallo forzoso
16 → fallo forzoso
21 → fallo forzoso
13 → " "
64 → " "
48 → " "
19 → " "
11 → Acierto
3 → Acierto
22 → fallo forzoso
4 → " "
27 → " "
6 → " "
11 → Acierto

① 8 bloques de 1 palabra cada uno

22, 26, 22, 26, 16, 3, 16, 18



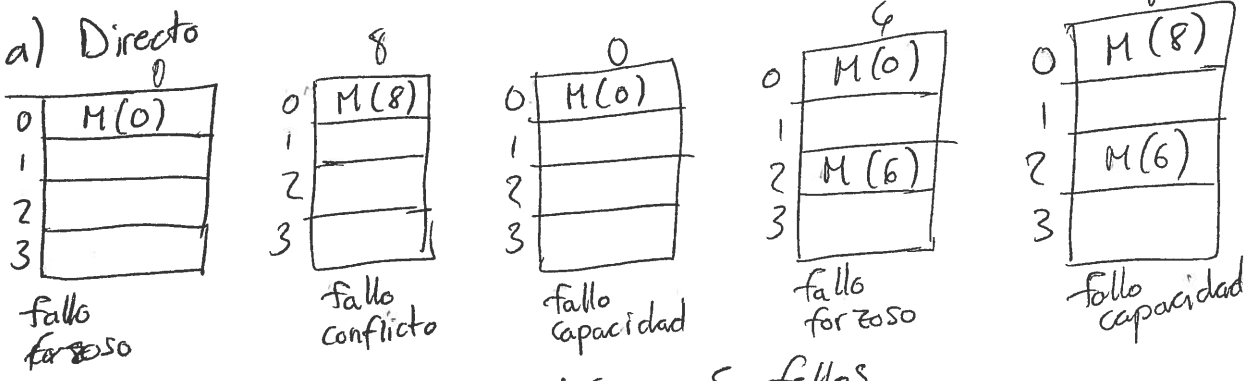


Correspondencia directa: 8 accesos a caché y 5 fallos.

② 4 bloques de 1 palabra cada uno LRU

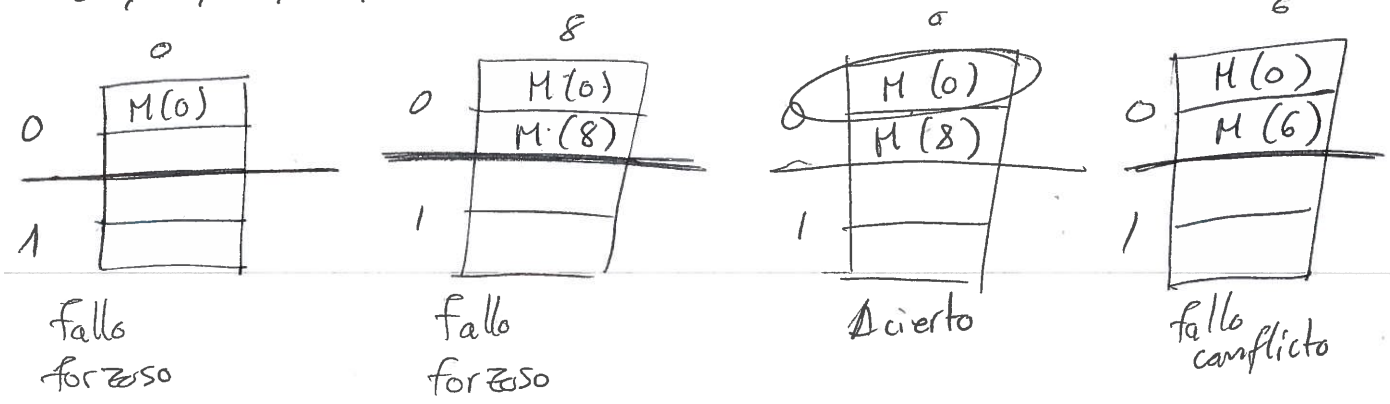
0, 8, 0, 6, 8

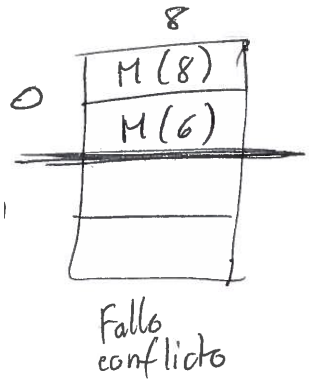
a) Directo



5 acceso a caché y 5 fallos.

0, 8, 0, 6, 8 CONJUNTO DE dos vías





5 accesos a caché y 4 fallos

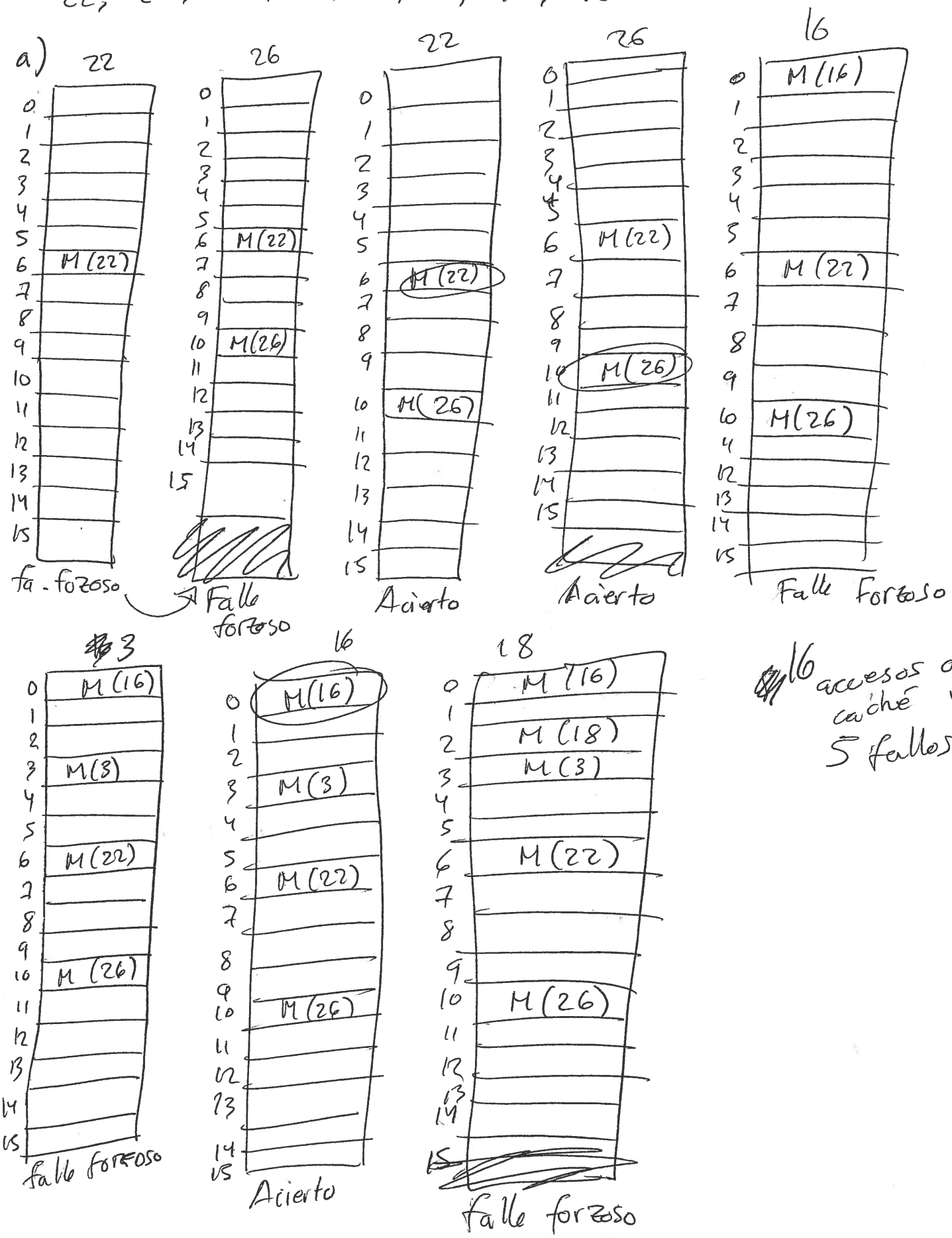
Totalmente asociativa:

5 acceso a cache y 3 fallos.

3) 16 bloges con 1 p en cada uno.

22, 26, 22, 26, 16, 3, 16, 18

a)

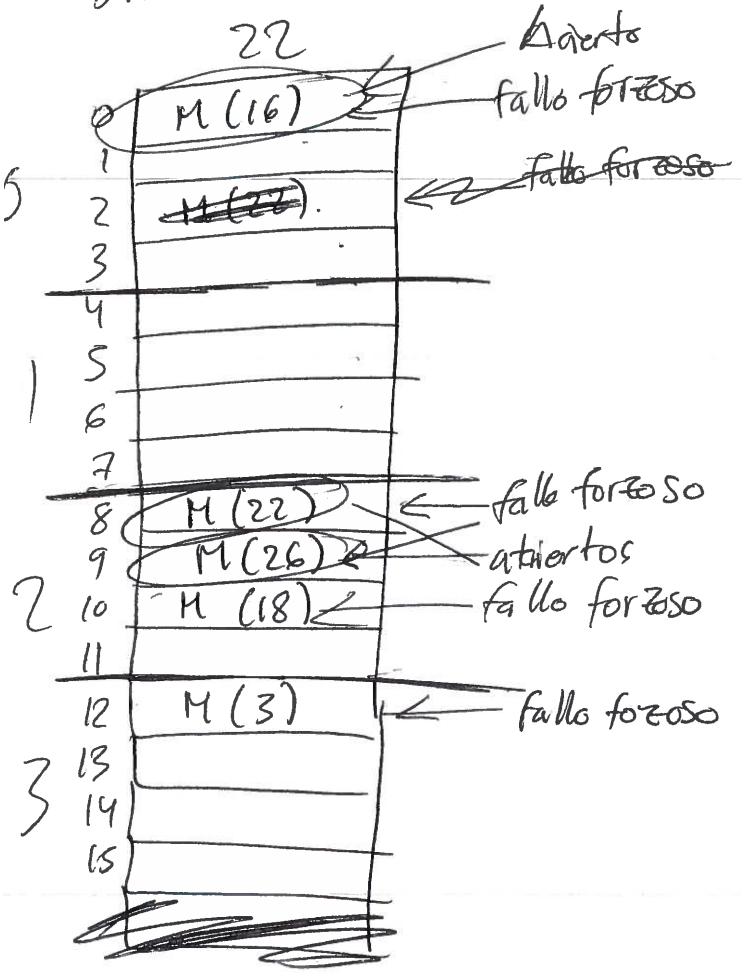


16 accesos a caché y 5 fallos

22, 26, 22, 26, 16, 3, 16, 18

Directa con tamaño de bloque de 4 palabras

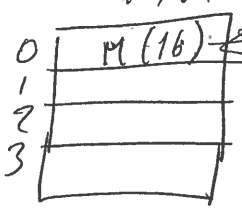
16 accesos a caché y 4 fallos



CORRESPONDENCIA DIRECTA

4 bloques de 1 p cada uno que tenga fallo 100%

16, 0, 16, 0



Fallo

Fallos

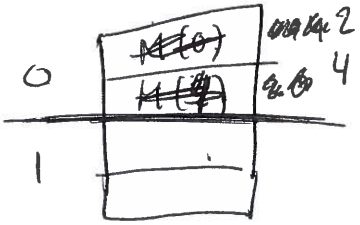
1	~~||||~~ |||
2	~~||~~ |||||

Aciertos

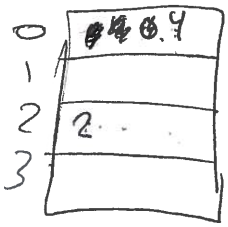
1	~~||||~~ |||
2	~~||~~	

~~0, 2, 2, 0, 0~~

0, 2, 4, 0, 2, 4



Fallos	Aciertos
						1



① Palabras = 32 bits

16 Registros = 2^4 → **Registros = 4 bits**

Memoria = 255 Mpalabras = $2^8 \cdot 2^{20} = 2^{28}$ → **Dato = 28 bits**

Move origen, destino

add destito, operando 1, operando 2.

Modos direccionamiento = 6 → 2^3 ⇒ **MD = 3 bits**

Instrusciones 2 → 2^1 ⇒ **COP = 1 bits**

MOVE ORIGEN, DESTINO

←1→	←3→	←28→	←1→	←3→	←28→
COP	MD.O	ORIGEN		MD.D	DESTINO

← 64 →

ADD DESTINO, OP1, OP2

←1→	3	28		3	28		3	28
COP	MD.D	DESTINO		MD.OP1	OP1		MD.OP2	OP2

⑤ ~~1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17~~ 8 palabras

0	~~M(8)~~ M(56)
1	~~M(1)~~ ~~M(17)~~ ~~M(9)~~ M(17)
2	
3	~~M(19)~~ ~~M(11)~~ M(43)
4	~~M(4)~~ ~~M(20)~~ M(4)
5	M(5)
6	M(6)
7	

0	~~M(4)~~ ~~M(20)~~ M(4))
	~~M(8)~~ M(56)
1	~~M(1)~~ ~~M(17)~~ ~~M(5)~~ M($7)
	~~M(5)~~ M(9)
2	~~M(no)~~ M(6)
	1
3	~~M(19)~~ M(43)
	M(11)

1 → fallo forzoso
4 → fallo forzoso
8 → fallo forzoso
5 → fallo forzoso
20 → fallo conflicto
17 → fallo conflicto
19 → fallo forzoso
56 → fallo conflicto
9 → fallo conflicto
11 → fallo conflicto

43 → Fallo conflicto
5 → acierto
6 → fallo forzoso
9 → Acierto
17 → fallo conflicto

O	·	·	·
X	O	·	·
O	X	O	X
·	O	X	O

① a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
SUB R1, R2, R3	IF	ID	EX	ME	WB													
ADD R4, R5, R6		IF	ID	EX	ME	WB												
SUB R5, R4, R8			IF	ID	ID	ID	EX	ME	WB									
ADD R7, R2, R3				IF	IF	IF	ID	EX	ME	WB								
ADD R9, R7, R3							IF	ID	ID	ID	EX	ME	WB					
LW R1, 10(R6)								IF	~~ME~~ IF	IF	ID	EX	ME	WB				
ADD R3, R1, R4											IF	ID	~~EX~~ ID	~~ME~~ ID	~~WB~~ EX	ME	WB	
SUB R6, R7, R8												IF	IF	IF	ID	EX	ME	WB

18 ciclos

b)

	1	2	3	4	5	6	7	8	9	10	11	12	13
SUB R1, R2, R3	IF	ID	EX	ME	WB								
ADD R4, R5, R6		IF	ID	EX	ME	WB							
SUB R5, R4, R8			IF	ID	EX	ME	WB						
ADD R7, R2, R3				IF	ID	EX	ME	WB					
ADD R9, R7, R3					IF	ID	EX	ME	WB				
LW R1, 10(R6)						IF	ID	EX	ME	WB			
ADD R3, R1, R4							IF	ID	ID	EX	ME	WB	
SUB R6, R7, R8								IF	IF	ID	EX	ME	WB

13 ciclos.

R1, R2, R3 — IF ID EX ME WB
R4, R5, R6 — IF ID EX ME WB
R7, R2, R3 — IF ID EX ME WB
R5, R4, R8 — IF ID ID
R9, R7, R3 — IF ID EX

LW R1, 10(R6) IF ID EX ME WB
R6, R7, R8 IF ID EX ME WB
R3, R1, R4 IF ID EX ME WB

① PALABRA = 32 bits

32 registros de 32 bits cada uno → Registros = 2^5 → $\boxed{\text{Registros} = 5 \text{ bits}}$

64 instrucciones → instrucciones = 2^6 → $\boxed{\text{COP} = 6 \text{ bits}}$

4 modos direccionamiento → 4 → $MD = 2^2$ → $\boxed{MD = 2 \text{ bits}}$

a) 2 operandos Registro-Memoria

COP	MD	REG	DIRECTO/ INDIRECTO

← 6 → ← 2 → ← 5 → ← 19 →

← 32 →

← 6 → ← 5 → ← 2 → ← 5 → ← 14 →

COP	REG	MD	REG	DESPLAZ

← 32 →

1 → fallo forzoso
4 → fallo forzoso
8 → " "
5 → " "
20 → " conflicto
17 → " "
19 → " forzoso
56 → " conflicto
9 → " "
11 → " forzoso
4 → " capacidad

43 → fallo conflicto
5 → " "
6 → " forzoso
9 → Acierto
17 → fallo conflicto

0	M(4)
	M(8)
1	M(1)
	M(5)

b) ← 2 → ← 5 →

MD	REGISTRO

← 32 →

② Multiplicación → 4 veces menos

Por acceso a memoria → 2 veces menos

$$\boxed{A = \frac{1}{(1 - Fm) + \frac{Fm}{Am}}}$$

Multiplicación → $Fm = 0'2$, $Am = 4$ ⇒ $A = \dfrac{1}{(1-0'2) + \left(\frac{0'2}{4}\right)} = \boxed{1'17}$

Memoria → $Fm = 0'5$, $Am = 2$ ⇒ $A = \dfrac{1}{(1-0'5) + \left(\frac{0'5}{2}\right)} = \boxed{1'33}$

Ambas ⇒ $A = \dfrac{1}{(1-(0'2+0'5)) + \left(\left(\frac{0'2}{4}\right) + \left(\frac{0'5}{4}\right)\right)} = \boxed{1'67}$

⇓

$Fm = 0'7$

2a)

	1	2	3	4	5	6	7	8	9	10	11	12	13
SUB R1, R2, R3	IF	ID	EX	ME	WB								
ADD R4, R5, R6		IF	ID	EX	ME	WB							
OR R5, R1, R4			IF	ID	ID	ID	EX	ME	WB				
SW R5, 10(R6)				IF	IF	IF	ID	EX	ME	WB			
LW R1, 20(R8)							IF	ID	EX	ME	WB		
AND R3, R8, R5								IF	ID	EX	ME	WB	
ADD R3, R1, R7									IF	ID	EX	ME	WB

LAVAR, SECAR, DOBLAR, GUARDAR

~~DIRECTO A MEMORIA~~

② Multiplicación 4 veces menos ⇒ 20%

Acceso a memoria 2 veces menos ⇒ 50%

a)

$F_m = 20\% = 0'2$

$A_m = 4$

$$A = \frac{1}{(1-0'2) + \left(\frac{0'2}{4}\right)} = 1'18.$$

Si solo se mejora la multiplicación, el incremento será de 1'18.

b) $F_m = 50\% = 0'5$

$A_m = 2$

$$A = \frac{1}{(1-0'5) + \left(\frac{0'5}{2}\right)} = 1'3$$

c)

$F_m = 70\% = 0'7$

$$A = \frac{1}{(1-0'7) + \left(\left(\frac{0'2}{4}\right) + \left(\frac{0'5}{2}\right)\right)} = 1'6$$

④ a)

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
SUB	R1, R2, R3	IF	ID	EX	ME	WB													
ADD	R4, R5, R6		IF	ID	EX	ME	WB												
SUB	R5, R4, R8			IF	ID	ID	ID	EX	ME	WB									
ADD	R7, R2, R3				IF	IF	IF	ID	EX	ME	WB								
ADD	R9, R7, R3							IF	ID	ID	ID	EX	ME	WB					
LW	R1, 10(R6)								IF	IF	IF	ID	EX	ME	WB				
ADD	R3, R1, R4											IF	ID	ID	ID	EX	ME	WB	
SUB	R6, R7, R8												IF	IF	IF	ID	EX	ME	WB