

Arquitectura e ingeniería de computadores II

1 bit \Rightarrow 0 ó 1

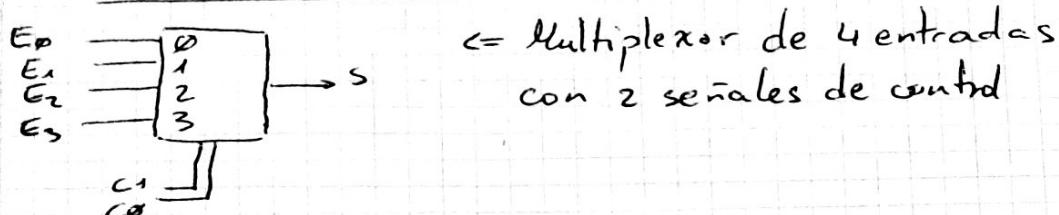
1 byte \Rightarrow 8 bits

\hookrightarrow Son 8 bits debido al código ASCII (256 caracteres)

1 palabra \Rightarrow N bytes (depende de la arquitectura del ordenador)

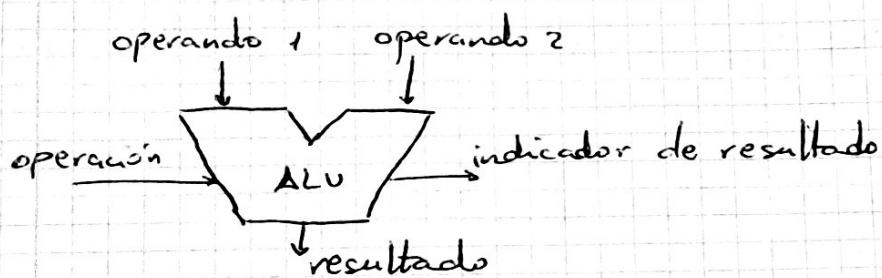
\hookrightarrow Si el ordenador es de 32 bits \Rightarrow 1 palabra = 4 bytes

Construcción de un ALU

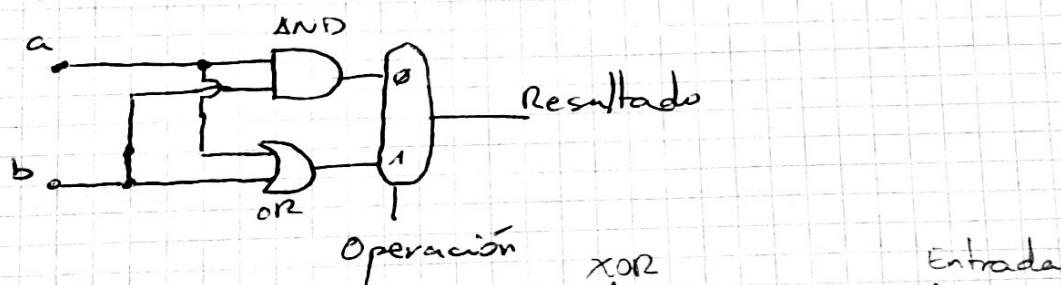


Nota: "exclusive or" se suele escribir como "a xor b" ó "a ⊕ b"

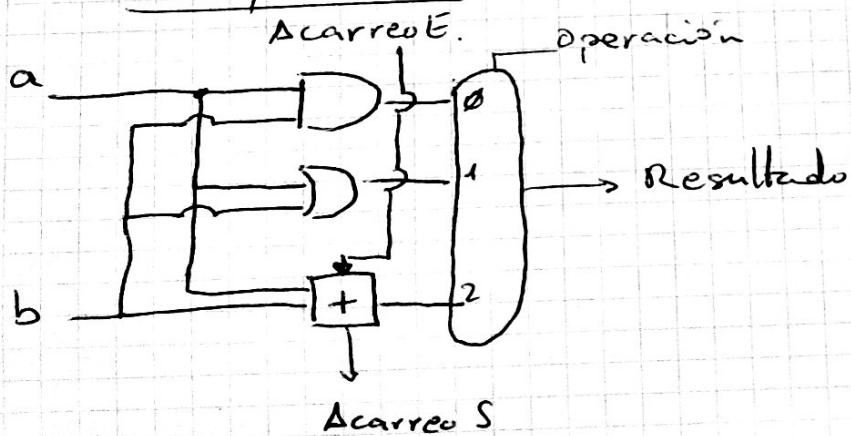
ALU



Ejemplo : unidad lógica de 1 bit con las funciones AND y OR



Fórmula para la suma: $a \oplus b \oplus \text{Acarreo E.}$



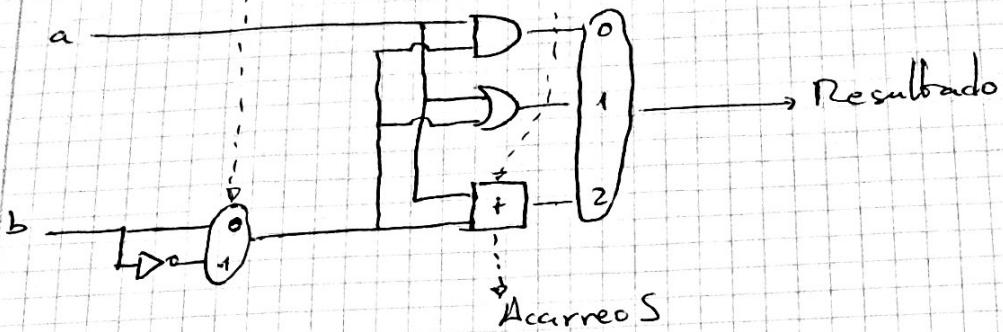
Camino de datos control

b" negado = b en la

ALU con resta

$$a - b = a + (-b) = a + (\bar{b} + 1) = a + \bar{b} + 1$$

[Recordatorio: la resta se hace con C_{a_2} , que es $C_{a_1} + 1$]
 , Invertir B Acarreo E



Función \sin

słt \$rd, \$rs, \$rt

Si $\$rs < \$rt \Rightarrow \$rd = 1 = \emptyset\emptyset\dots\emptyset 1$

$$\text{e.o.c.} \Rightarrow \$\text{rd} = \emptyset$$

↳ En otros casos

Nota: para saber cual es el signo de un número (en \mathbb{Q}_2) me fijare en el bit más significativo.

$$\begin{array}{l} a = \emptyset \emptyset \emptyset \emptyset \emptyset \\ b = \emptyset 1 1 1 \\ \hline s \Delta t = \emptyset \emptyset \emptyset \{ \square \end{array}$$

$a - b = \emptyset \dots$ si $a < b$
 ó
 $a - b = \emptyset \dots$ si e.o.c.

Nota: Como la resta es en C_2 , es decir, $C_1 + 1$, el 1 se coloca en el primer acarreo de entrada (cuando se suman los dos primeros bits de "a" y "b")

Nota: Si hago una multiplicación de n bits, el resultado se expresará en $2n$ bits (por ser potencia de 2)

Camino de datos y control

Memoria

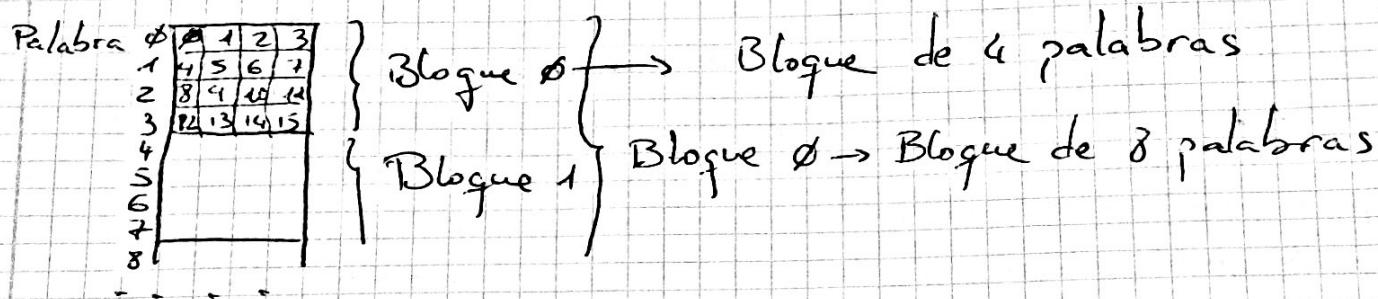
1	1b	1b	3
Byte			
1	2	3	4
2	3	4	5

Toda palabra válida empieza en una dirección de memoria que sea múltiplo de 4

[Pág 10] \Rightarrow Si tengo 001 y hago un desplazamiento de 2 bits obtengo 100, que es 4. Todo múltiplo de 4 tiene al menos 2 ceros.

II Tema 5: Cache

Nota: En ocasiones llamamos cache a aquellos elementos que cumplen el rol de una memoria cache: acelerar procesos. Este es el caso de los navegadores.



Ejercicio: Cuántos bits se necesitan para implementar una cache de correspondencia directa con 64 KB de datos y bloques de 8 palabras suponiendo direcciones de 32 bits?

Bloques de una palabra. Palabras de 4 bytes (32 bits). \rightarrow
 $\Rightarrow 16\text{K palabras} = 16\text{K bloques}$

2^{14}

Cada línea de cache tiene 1 bit de Val., 1 tag, 32 bits (datos)

Dir = 32 bits $\xrightarrow{2\text{ bits despl.}}$
 $\xrightarrow{14\text{ bits índice}}$
 $\xrightarrow{16\text{ bits etiqueta}}$

$$2^{14} \cdot (1 + 16 + 32) = 784 \text{ Kbits} = 98 \text{ KiB}$$

Tema 7: buses

* Bus síncrono: bus síncrono gobernado por un reloj

Funcionamiento de bus asíncrono

El dispositivo pone a 1 la señal ReadReg. y la dirección solicitada en el bus. Cuando la memoria esté disponible pone a 1 la señal Ack; el dispositivo ve esta señal Ack y retira el 1 de ReadReg y la dir. del bus.

La mem. verá que el dispositivo ya está esperando el dato, coloca Ack = 0, obtiene los datos, los coloca en el bus y pone la señal ReadReady a 1.

Rendimiento: bus síncrono vs. bus asíncrono

BUS SÍNCRONO (Ciclo=50 ns)

- Envío dir. memoria: 50 ns
 - Latencia de memoria: 200 ns
 - Envío de datos al disp.: 50 ns
- Total = 300 ns

$$\begin{aligned} 4 \text{ bytes} &\rightarrow 300 \text{ ns} \\ x \text{ bytes} &\rightarrow 1 \text{ s} \quad \left\{ \begin{array}{l} x = 13'3 \text{ MB/s} \\ 10^9 \text{ ns} \end{array} \right. \end{aligned}$$

BUS ASÍNCRONO

[Hay 7 etapas, pero algunas se solapan con otras]

Pasos 1 = 40 ns Pasos 2, 3, 4
 $\text{Max}(\frac{120 \text{ ns}}{\text{latencia}}, \frac{200 \text{ ns}}{\text{latencia}}) = 200 \text{ ns}$
 Pasos 5, 6, 7 = 120 ns

$$\text{TOTAL} = 360 \text{ ns}$$

$$\begin{aligned} 4 \text{ bytes} &\rightarrow 360 \text{ ns} \\ x &\rightarrow 1 \text{ s} \quad \left\{ \begin{array}{l} x = 1'1 \text{ MB/s} \\ 10^9 \text{ ns} \end{array} \right. \end{aligned}$$

* [PDF] Calcular ancho de banda ...

BLOQUES 4 PALS.

- Envío dir = 1 ciclo
- $200 \text{ ns}/5 \text{ ns/ciclo} = 40 \text{ ciclos}$
↳ latencia
- 2 ciclos \rightarrow Envío 128 bits de mem. a disp.
- 2 ciclos (tiempo entre operaciones)

$$\text{TOTAL: } 45 \text{ ciclos} \Rightarrow 45 * \frac{256}{4} = 2880 \text{ ciclos}$$

$$\Rightarrow 14400 \text{ ns} \Rightarrow 4'44 \text{ M.trans/s}$$

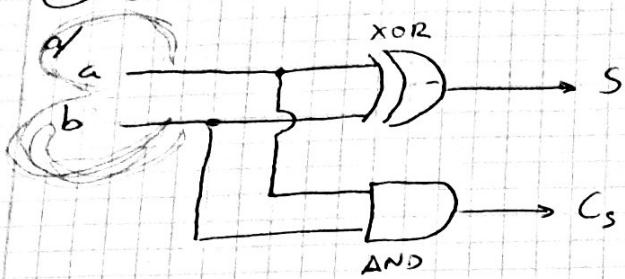
$$(\Rightarrow 5 \text{ ns/ciclo})$$

$$71'11 \text{ MB/s}$$

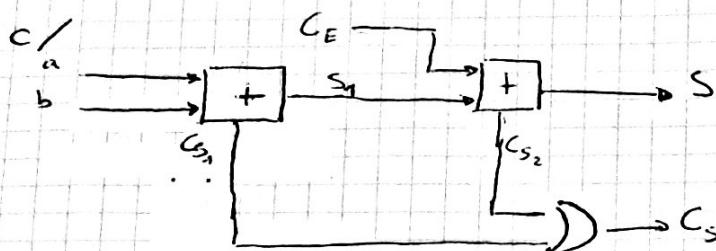
BLOQUES 16 PALS.

Hoja de problemas (tema 2)

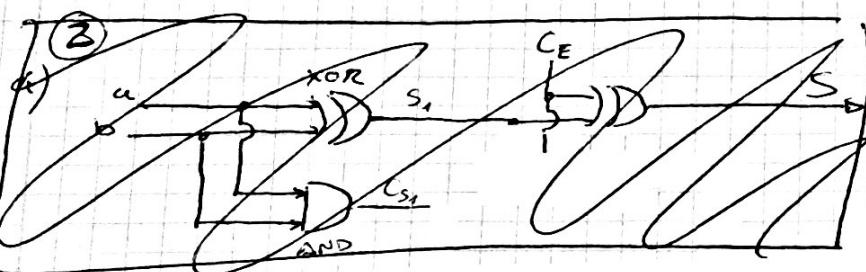
① [Sin tener en cuenta los retardos]



* El retardo será en función de lo que tarde más: el **XOR** ó el **AND**
 $\text{Retardo} = \text{MAX}(\text{XOR}, \text{AND})$



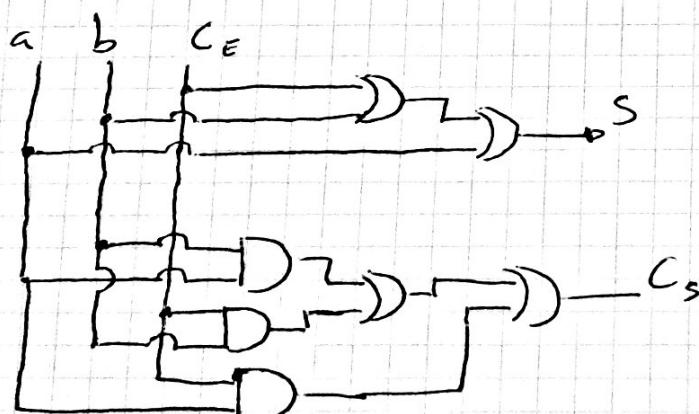
↳ Si hago la tabla de verdad veo que nunca cumple $C_{S_1} = 1$ y $C_S = 1$. Solo uno o ninguno es 1.



②

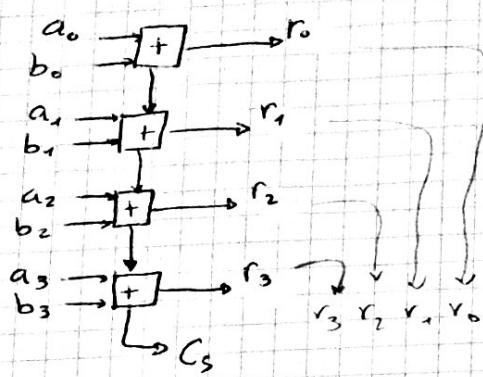
$$a) S = a \oplus b \oplus C_E$$

$$C_S = \bar{a} \cdot b \cdot C_E + a \cdot \bar{b} \cdot C_E + ab \cdot \bar{C}_E + ab \cdot C_E = \dots = ab + bC_E + aC_E$$



[Aquí se podría hacer como el 1.c]

② c)

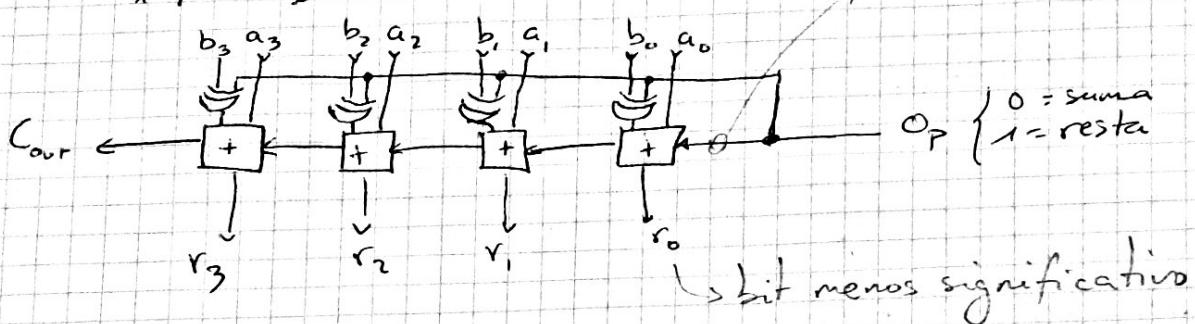


④ a)

a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

$$a \oplus 1 = \bar{a}$$

$$\text{ya que } C_{in} \Rightarrow a + b + 1$$



↳ bit menos significativo.

[Hacer ③]

*) ¿Qué implementación es la más adecuada?

Mem \rightarrow 2 ns
 ALU \rightarrow 2 ns
 Banco regis \rightarrow 1 ns

1) Impl. uniciclo (ciclo de tamaño fijo)

2) Ciclo long. variable

Ejecuto un programa con

24% \rightarrow lw 44% \rightarrow R 2% \rightarrow J

12% \rightarrow SW 18% \rightarrow BEQ

$$\begin{array}{cccc}
 \text{Instruc.} & \text{registro} & \text{ALU} & \text{Memory} \\
 \text{memoria} & \checkmark & \checkmark & \checkmark \\
 \text{lw} \rightarrow & 2 \text{ns} + 1 \text{ns} + 2 \text{ns} + 2 \text{ns} + 1 \text{ns} = 8 \text{ns}
 \end{array}$$

$$R \rightarrow 2 + 1 + 2 + 1 = 6$$

$$SW \rightarrow 2 + 1 + 2 + 2 = 7$$

$$BEQ \rightarrow 2 + 1 + 2 = 5 \text{ ns}$$

$$J \rightarrow 2 = 2 \text{ ns}$$

Tiempo ejecución = N° inst. \times tiempo de ciclo

$$1) T_{ciclo} = 8 \text{ ns}$$

$$2) T_a = 0'24 \cdot 8 + 0'12 \cdot 7 + 0'44 \cdot 6 + 0'18 \cdot 5 + 0'02 \cdot 2 = 6'34 \text{ ns}$$

La implementación 2 es 1'8 veces más rápida que la primera.

Hoja de problemas 03

① [Ver tabla de pg 22]

② No se podrían hacer operaciones de guardar y cargar correctamente

⑤ $j \text{ target } \quad PC \leftarrow PC[31..28] | (\text{target} \ll 2)$

- a) 1. Leo instrucción de memoria y aumento el contador del programa
2. Cojo el target (en binario), lo desplazo 2 posiciones a la izquierda, concateno con los bits 31-28 del contador del programa y el resultado lo almaceno en este último.

0x02	target
[31-26]	[15-0]

Luego no tengo que realizar ninguna modificación en el circuito.

⑥ a) addiu rt,rs,inm $rt \leftarrow rs + \text{ext_signo}(inm)$

0x09	rs	rt	inm
31-26	25-21	20-16	15-0

1. Leer instrucc. y $PC = PC + 4$

2. Leo registro \$rs y éste va al ALU

3. El inm. llega al ext-signo y después al ALU } Sumo ambas cosas

4. Indico que voy a escribir en el registro \$rt, y en éste guardo la suma anterior

Luego en el caminio de datos no tengo que hacer cambios. Sin embargo la unidad de control haré cambios.

Cod. op: 0x09 = 001001

ALUSrc = 1

MemRead = Ø (este me daría igual)

ALUOp = 01 (código de suma)

MemWrite = Ø

MemToReg = Ø

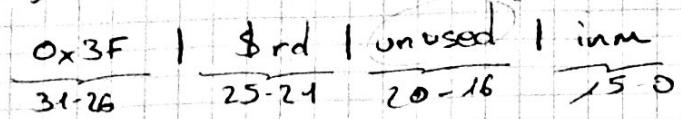
Branch = Ø

RegDst = Ø

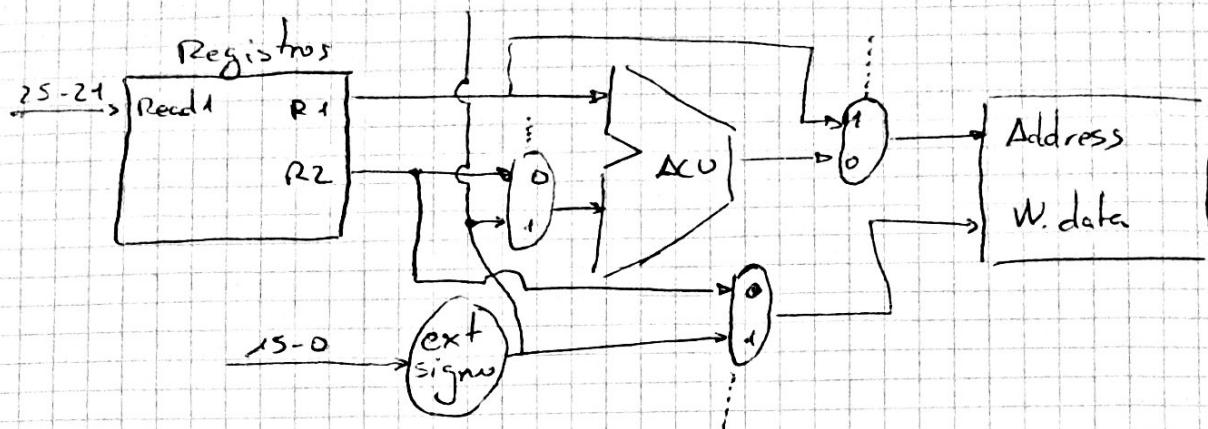
Jump = Ø

RegWrite = 1

O b) swi \$rd, imm \rightarrow Sin usar



Mem(rd) \leftarrow ext_signo(imm)



* Tener en cuenta que si añado mux. tendré que añadir señales de control para las anteriores instrucciones (p.ej.: lw). Por esto, se tiende a poner ϕ en el cable del Mux que esté por defecto.

swi \Rightarrow Leo la dirección de memoria que hay en el registro \$rd y guardo ahí el inmediato.

Tema 5: Ejercicios

DATOS	Poca localidad temporal	Mucha localidad temporal
Poca localidad espacial	Acceder a cada variab. 1 vez	Acceso repetitivo a las var.
Mucha loc. espacial	Acceder a cada var. 1 vez Uso de arrays	Acceso repetitivo a las var.

INSTRUCCIONES	Poca loc. temporal	Mucha loc. temporal
Poca loc. espacial	No reutilización de código Muchos saltos a zonas alejadas en memoria	Reutilización masiva de código
Mucha loc. espacial	No reutilización de código Sin saltos solo secuencial	Reutilización masiva de código

$$1) T_1 \cdot \text{ciclo} = 8 \text{ ns}$$

$$2) T_2 = 0'24 \cdot 8 + 0'12 \cdot 7 + 0'44 \cdot 6 + 0'18 \cdot 5 + 0'02 \cdot 2 = 6'34 \text{ ns}$$

La implementación 2 es 1'8 veces más rápida que la primera.

Hoja de problemas 03

① [Ver tabla de pg 22]

③ No se podrían hacer operaciones de guardar y cargar correctamente

⑤ j target $PC \leftarrow PC[31..28] | (\text{target} \ll 2)$

- a) 1. Leo instrucción de memoria y aumento el contador del programa
2. Cojo el target (en binario), lo desplazo 2 posiciones a la izquierda, concateno con los bits 31-28 del contador del programa y el resultado lo almaceno en este último.

0x02	target
[31-26]	[25-0]

Luego no tengo que realizar ninguna modificación en el circuito.

⑥ a) addiu rt,rs,inm $rt \leftarrow rs + \text{ext-signo}(inm)$

0x09	rs	rt	inm
31-26	25-21	20-16	15-0

1. Leer instrucc. y $PC = PC + 4$

2. Leo registro \$rs y éste va al ALU

3. El inm. llega al ext-signo y después al ALU } Sumo ambas cosas

4. Indico que voy a escribir en el registro \$rt, y en éste guardo la suma anterior

Luego en el caminho de datos no tengo que hacer cambios. Sin embargo en la unidad de control haré cambios.

Cod. op: 0x09 = 001001

ALUSrc = 1

ALUOp = 00 (código de suma)

MemToReg = 0

RegDst = 0

RegWrite = 1

MemRead = 0 (este me daría igual)

MemWrite = 0

Branch = 0

Jump = 0

- (7)
- 1 - F
 - 4 - F
 - 8 - F
 - 5 - F
 - 20 - F. Se quita la palabra 4 y se guarda la 20
 - 17 - F
 - 19 - F
 - 56 - F. Quito 8, guardo 56

- 9 - F
- 11 - F
- 4 - F. Se quita la 20 y guardo la 4
- 43 - F. Quito 11, guardo 43
- 5 - Acíerto
- 6 - F
- 9 - Acíerto
- 17 - Acíerto

(8)

L0	:	1	:	:	:
L1	:		:		
L2	:		:		
L3					

Si se produce un fallo en un bloque de varias palabras habría que traer el bloque entero de memoria principal.

(17)

$$10 \text{ ns} = T_A \cdot 8 \text{ ns} + T_F \cdot 60 \text{ ns} = \\ = T_A \cdot 8 \text{ ns} + (1 - T_A) \cdot 60 \text{ ns} \Rightarrow T_A = 96.2\%$$

Tema 6 - Ejercicios

(1) $1 \text{ KB} = 2^{10} \text{ bytes} \Rightarrow 2 \text{ KB} = \underbrace{2^{11} \text{ bytes}}_{\downarrow} = 2^{14} \text{ bits}$
 11 bits de desplaz. de página

$$\text{Nº pag. virtual} \Rightarrow 32 - 11 = 21 \Rightarrow 2^{21} \text{ páginas} \Rightarrow "2 \text{ Megapáginas"}$$

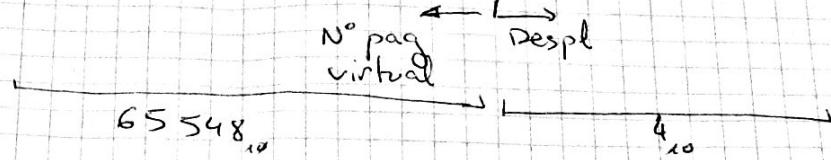
(2) Mem = 512 MB = 2^{29} bytes
 Páginas = 4 KB = 2^{12} bytes \rightarrow 12 bits de desplaz.

Formato \Rightarrow $\underbrace{20 \text{ bits}}_{\text{nº pag. virtual}} \mid \underbrace{12 \text{ bits}}_{\text{desplaz.}}$

$$\frac{2^{29}}{2^{12}} = 2^{17} \Rightarrow 131.072 \text{ páginas}$$

③ $8KB = 2^{13}$ bytes \rightarrow 13 bits desplz. $\rightarrow 32 - 13 = 19$ nº pag. virtual

$$0x20018004 = 0010\ 0000\ 0000\ 0001\ 1000\ 0000\ 0000\ 0100_2$$



⑥ Dir.: 32 bits

$8KB = 2^{13}$ bytes \Rightarrow 13 bits desplz. $\Rightarrow 19$ nº pag. virtual

a) Formato \Rightarrow $\begin{array}{|c|c|} \hline 19 & | & 13 \\ \hline \text{nº pag. virtual} & & \text{bits de desplz.} \\ \hline \end{array}$

b) Tamaño max.: 2^{19} páginas

Esto es porque tengo 19 bits para poner el número de página

c) 890.000 elementos $\cdot \frac{4 \text{ bytes}}{\text{elem.}} = 3.560.000$ bytes

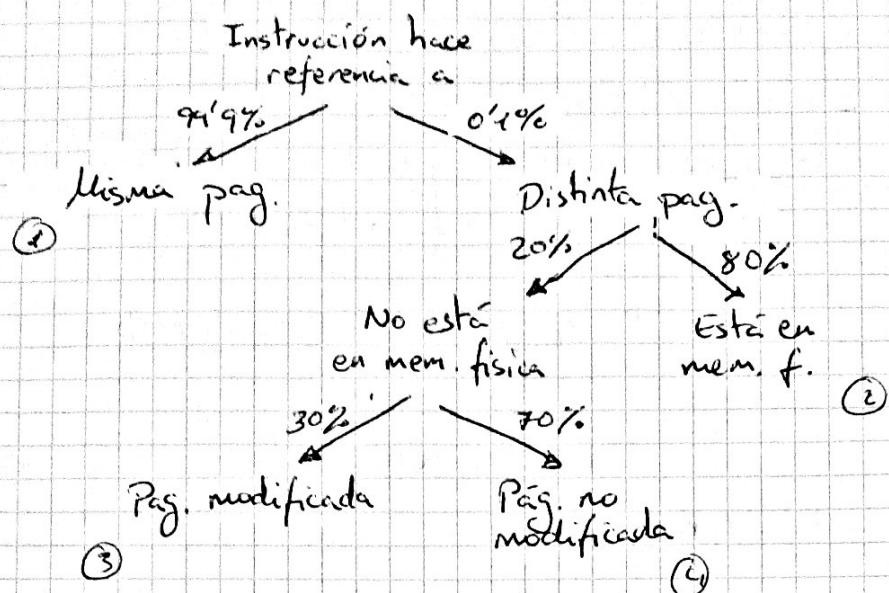
Si 1 pág. son $8KB \Rightarrow \frac{3560000}{2^{13}} = 434.57$ páginas ≈ 435 páginas

de los ✓

De aquí, 1 página será para instrucciones, ya que el código probablemente ocupe menos de 8 KB en ensamblador.

Luego tendré 436 fallos como mínimo.

(8)



Enunciado:
 Vel. de procesador
 10^6 instrucc. por seg.

Caso 1 \Rightarrow Prob.: 0'999

$$T_{\text{exec.}} = \frac{1}{10^6} = 1 \mu\text{s} \quad (\text{microseg.})$$

Caso 2 \Rightarrow Prob. = $0'001 \cdot 0'8 = 8 \cdot 10^{-4}$

$$T_{\text{exec.}} = 1 \mu\text{s}$$

Caso 3 \Rightarrow Prob. = $0'001 \cdot 0'2 \cdot 0'3 = 6 \cdot 10^{-5}$

$$T_{\text{exec.}} = 20 \text{ ms} + 1 \mu\text{s} = 20001 \mu\text{s}$$

Caso 4 \Rightarrow Prob. = $0'001 \cdot 0'2 \cdot 0'7 = 1'4 \cdot 10^{-4}$

$$T_{\text{exec.}} = 10 \text{ ms} + 1 \mu\text{s} = 10001 \mu\text{s}$$

$$\bar{T} = \sum_{i=1}^4 T_i \cdot P_i = 3'6 \mu\text{s}$$

(10) Mem. virtual = 16 MB = 10^{24} Bytes

$$\text{Mem. f.} = 512 \text{ KB} = 10^{19} \text{ Bytes}$$

Tamaño pag = $4KB = 10^{12}$ Bytes \Rightarrow 12 bits de desplz.
20 bits n° pag.

$$c) 16\text{ MB} / 4\text{ KB} = 2^{24} / 2^{12} = 2^{12} = 4096 \text{ páginas}$$

1bit de validez

1 marco de pag. \Rightarrow M.F 512KB / 4B/B = $2^7 = 128$ págs. \Rightarrow 7 bits cada marco
 $T_P = 4096 \cdot (1+7) = 4KB$

$$TP = 4096 \cdot (1+7) = 4KB$$

b) N° accesos a mem-p. \rightarrow 10 fallos

- En cada fallo:
[Ever enunciado: hay 10 pags diferentes]

- 1 por T.P.

- Llevar la pag. a m.p. $\Rightarrow 2^{12}$ (ya que cada pag. tendrá 2¹² bytes y en el enunciado no se menciona el tamaño de bloque o palabra)

- 1 por actualización de TP.

- 1 para acceder a la posición deseada

$$\text{Luego } (2^2 + 4) \cdot 10 = 4.100$$

- ④ N° acc. por acierto

- 1 por TP

- 1 para acceder a la pos. deseada

Luego 2

$$N^{\circ} \text{ aciertos} = N^{\circ} \text{ referencias} - N^{\circ} \text{ fallos} = 86006$$

21. 2² 10

El número total de accesos por acierto es $2 \cdot 86006 = 172012$

$$\text{--- --- --- --- --- a mem. p. } = 41000 + 172012 = 213012$$

f) Igual que en ⑥, ya que, el hecho de que haya un TEP no varía los accesos a memoria por falso

c) SIN TLB

1 a TP

2¹² copia de datos a MF

1 actualizar TP con nuevo marco de pg.

* 1 a TP

1 a dir. de mem. deseada

CON TLB

1 a TP

2¹² copia de datos a MF

1 actualizar TP con nuevo marco de pg.

* 1 para copiar marco de pg. a TLB

1 a dir de MF deseada

$$\text{Accesos a memoria} = 127 \text{ Kbytes}$$

Tema 8

④ El siguiente diagrama ...

El de la derecha, debido a que al terminar de leer el sector 7 tiene que cambiar a la pista continua (lo cual tiene un retardo). No es imposible que en el de la derecha se salte 3 sectores y, por tanto, este tendría menor rendimiento, pero esto es poco probable ya que son pistas continuas.

① Supóngase el protocolo ...

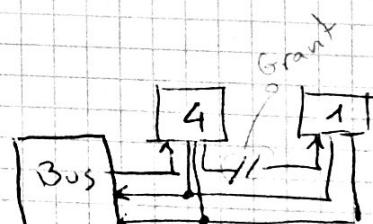
$$\left. \begin{array}{l} 10 \text{ ns petición} \\ 15 \text{ ns arbitraje} \\ 25 \text{ ns completar operación} \end{array} \right\} 50 \text{ ns} \xrightarrow{\text{1 oper.}} \left. \begin{array}{l} 1 \text{ s} \\ \xrightarrow{\text{x oper}} \end{array} \right\} \Rightarrow x = 20 \text{ millones}$$

② Un bus SCSI ...

4 dispositivos : 1 = menos prior. y 4 = más prior.

30 Mb/s = Petición de uso

80 Mb/s = Ancho total por dispositivo SCSI



$$\left. \begin{array}{l} 4 = 30 \text{ Mb/s} \\ 3 = 30 \\ 2 = 20 \\ 1 = 0 \end{array} \right\} \Rightarrow$$

El bus solo puede estar usado por un dispositivo en cada momento

$$\left. \begin{array}{l} 4 = 20 \\ 3 = 20 \\ 2 = 20 \\ 1 = 20 \end{array} \right\}$$

⑤ Leer de disco 4 KB (bloque) (Bloques contiguos)

Escribe el resultado en bloque de 4 KB

Disco rota 7200 rpm = 120 v/s

T. medir posicionamiento = 8 ms

Vel. transf. de 20 MB/s

Sobrecarga del controlador = 2 ms

Etapa de proceso de datos = 20 M ciclos

Frec. reloj = 400 MHz

$$T_p = 8 \text{ ms}$$

$$T_r = \frac{0.5}{7200/60} = 4.2 \text{ ms}$$

$$T_e = 4K \cdot \frac{1}{20000K} = 0.2 \text{ ms}$$

$$T_c = 2 \text{ ms}$$

$$400 \text{ MHz} \xrightarrow[1/400M]{} 2 \text{ ns/ciclo}$$



$$20 \cdot 10^6 \text{ ciclos} \Rightarrow 50 \text{ ms de procesado}$$

Tiempo por bloque

$$\text{Total} = 14.4 + 50 + 14.4 = 78.8 \text{ ms} \Rightarrow 12.7 \text{ bloques/seg.}$$

⑥ 300 M instr./seg.

Media = 50 000 instr. por operación E/S

Bus backplane vel. transf. = 100 MB/seg.

Control. bus vel. transf. = 20 MB/s (hasta 7 discos)

Ancho de banda lect./escr. = 5 MB/s

T_pos. + latencia rot. = 10 ms

Bloque = 64 KB

Un progr. necesita 100 000 instr. por op.

$$\text{CPU} \rightarrow \text{Max. vel} = \frac{300 \cdot 10^6}{50 \cdot 10^3 + 100 \cdot 10^3} = 2000 \text{ op. E/S / seg.}$$

$$\text{BUS} \rightarrow \text{Max. vel} = \frac{100 \cdot 10^6}{64 \cdot 10^3} = 1562 \text{ op. E/S / seg.}$$

resumir

(14)

16 sectores/pista de 1024 B cada uno

$$\text{Vel} = 3600 \text{ rpm} = 60 \text{ vueltas/seg}$$

$$T_{\text{búsqueda}} = 25 \text{ ms}$$

a) 1 vuelta — x
 $\frac{60 \text{ vueltas}}{1 \text{ seg}} \quad \left. \begin{array}{l} \\ x \end{array} \right\} x = 1/60 \text{ seg} = \frac{1}{60} \cdot 10^3 \text{ ms} = \frac{100}{6} \text{ ms} = 16'67 \text{ ms}$

$$\frac{3600 \text{ vueltas}}{\frac{1}{12} \text{ } \left. \begin{array}{c} \text{---} \\ x \end{array} \right\} x = 8'33 \text{ t. medio rotación}}$$

$$1^{\text{a}} \text{ pista} = 25 + 16'67 + 8'33 = 50 \text{ ms}$$

2^a pista

$$\frac{3600}{9/16} \text{ } \left. \begin{array}{c} \text{---} \\ x \end{array} \right\} x = 9'375 \quad \rightarrow \text{Supongo despreciable el tiempo de pos.}$$

$$t_{2^{\text{a}} \text{ pista}} = 9'375 + 16'67 = 26'05 \text{ ms}$$

$$T = 50 + 26'05 = 76'05 \text{ ms}$$

b) t. sector: $\frac{3600}{1/16} \text{ } \left. \begin{array}{c} \text{---} \\ x \end{array} \right\} x = 1'04 \text{ ms}$

$$T = 25 \cdot (1'04 + 8'33 + 25) = 859'25 \text{ ms}$$

(15) Sector = 4 KB

Sectores/pista = 250

Pistas = 15000

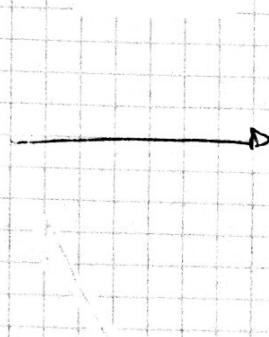
$$V. \text{ rotación} = 9600 \text{ rpm}$$

T_{transf} de sector = T en recorrido = T en mover cabezal

a) Capacidad = $15000 \cdot 250 \cdot 4 \text{ KB} \cdot \frac{1}{10^6} = 15 \text{ Gb}$

b) $\frac{9600}{1} \text{ } \left. \begin{array}{c} \text{---} \\ x \end{array} \right\} x = \frac{1}{9600/60} = 6'25 \cdot 10^{-3} \text{ seg} \quad \text{para 1 vuelta}$

$$T_{\text{sector}} = 6'25 \text{ ms} \cdot \frac{1}{250} = 0'025 \text{ ms}$$



→	POSICIONAMIENTO				ROTACIÓN				T_t $T_p + T_r + T_e$
	INICIO	Pista partida	Pistas (y sec.) recorridas	$T_{posic.}$ (seg.)	Sector partida	Sector scaneado	Sectores a Tr recorrer		
P0, S0									
P10, S50	10	10	0'00025	6	10	46	0'001	0'00025	0'001275
P12800, S200	10	12740	0'31975	51	91	149	0'002225	0'00025	0'3225
[- - -]									
$T_{total} = 0'94315 \text{ seg.}$									

$$\textcircled{(20)} \quad Frec. = 500 \text{ MHz}$$

$$\text{Vel. transf.} = 4 \text{ MB/bytes/seg}$$

$$\text{Tamaño transf.} = 8 \text{ KB/bytes}$$

$$\text{a) } \frac{8 \text{ KB/bytes}}{4 \text{ MB/bytes/seg}} = \frac{8 \text{ KB/bytes}}{4 \cdot 10^3 \text{ KB/seg}} = 2 \cdot 10^{-3} \text{ seg} = 2 \text{ ms}$$

$$\text{b) } \frac{2 \cdot 10^{-3} \text{ seg}}{1 \text{ seg}} = \frac{1 \text{ trans.}}{x} \quad \left. \begin{array}{l} x = 500 \text{ transf.} \end{array} \right\}$$

$$\text{c) Rotina} = 1000 \text{ ciclos}$$

$$\text{Interrupción} = 500 \text{ ciclos}$$

$$\text{c.1) } 500 \text{ transf./seg} \cdot 1500 \text{ ciclos/transf.} = 750.000 \text{ ciclos/seg}$$

$$\text{c.2) } \frac{750.000 \text{ ciclos/seg}}{500 \text{ M ciclos/s}} = 0'15\%$$

$$\textcircled{(27)} \quad Frec. = 10 \text{ MHz} = 10^7 \text{ ciclos/s} \quad \left. \begin{array}{l} \text{Lectura} = 1 \text{ ciclo} \\ \text{Escritura} = 1 \text{ ciclo} \end{array} \right.$$

$$CPI = 4 \text{ ciclos}$$

$$\text{a) Transf. 1 palabra} = 3 \text{ instr.}$$

$$\frac{10^7 \text{ ciclos/s}}{4 \text{ ciclos/instr.}} = 2.500.000 \text{ instr./s}$$

$$\frac{2.5 \cdot 10^6 \text{ instr./s}}{3 \text{ instr./palabra}} = 833.333'3 \text{ palabras/seg} = 8'3 \cdot 10^5 \text{ palab./s}$$

b) El DMA y el procesador se intercambian el bus sin mantener un orden. Para determinar la V_{max} supongo que el DMA tiene el bus.

$$V_{max} = \frac{10^7 \text{ ciclos/s}}{1 \text{ ciclo/pal}} = 10^7 \text{ pal/s}$$

$$\text{Y } V_{max} = \frac{10^7}{4+1} = 2 \cdot 10^6 \text{ pal/seg}$$

(26) Vel. lectura = 2^{21} pal/seg.

DMA por ráfagas // Vel. DMA = 4 ciclos/pal.

CP I = 8 ciclos

Frec. = $500 \cdot 10^6$ ciclos/s

$$\text{DMA} \cdot 2^{21} \text{ pal/seg} \cdot 4 \text{ ciclos/pal} = 2^{23} \text{ ciclos/seg}$$

$$500 \cdot 10^6 - 2^{23} = 4'92 \cdot 10^8 \text{ ciclos restantes}$$

$$4'92 \cdot 10^8 \text{ ciclos/s} / 8 \text{ ciclos/instr.} = 6'145 \cdot 10^2 \text{ instr./s con DMA}$$

Sin DMA: $500 \cdot 10^6 / 8 \text{ ciclos} = 6'25 \cdot 10^7$

Deja de ejecutar: $6'25 \cdot 10^7 - 6'145 \cdot 10^7 = 1'05 \cdot 10^6 \text{ inst./s}$

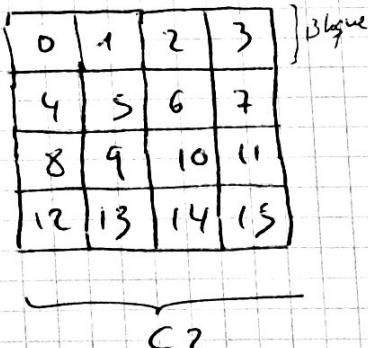
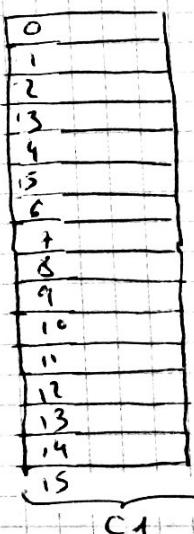
[Hecho por Oscar]

$$2^{21} \text{ pal/s} \cdot 4 \text{ ciclos/pal} = 2^{23} \text{ ciclos/s}$$

$$2^{23} \text{ ciclos/s} / 8 \text{ ciclos/s} = 2^{20} \text{ instr./s deja de ejecutar}$$

Caché

- (10)
- | |
|-------------------------------------|
| $C_1 = 16$ bloques de 1 palabra |
| $C_2 = 4$ bloques de 4 palabras |
| Penal por fallo de $C_1 = 8$ ciclos |
| " " " " $C_2 = 14$ ciclos |



Supongo que C_1 tendrá 4 fallos con penal. de 32 ciclos y C_2 con 3 fallos y penal. de 33 ciclos.

Lista de accesos: $\emptyset, 4, 8, 1\emptyset$

	0	4	8	1 \emptyset
C_1	F	F	F	Fallo
C_2	F	F	F	Acierto

$$\textcircled{20} \quad T. \text{ acierto cache} = 10 \text{ ns} \quad \begin{matrix} \text{(cambio de} \\ \text{cache)} \end{matrix} \quad T. \text{ fallos} = 3\% \\ \text{Tasa fallos} = 5\% \quad \quad \quad T. \text{ acierto} = 15 \text{ ns}$$

$$\bar{T} = t_a \cdot T_a + t_f \cdot T_f \quad \text{ist } \bar{T}_2 < \bar{T}_1?$$

$$\bar{T}_1 = 10 \text{ ns} - 0'95 + 0'05 \cdot T_{f_1}$$

$$\bar{T}_2 = 15 \text{ ns} \cdot 0'97 + 0'03 \cdot T_{f_2}$$

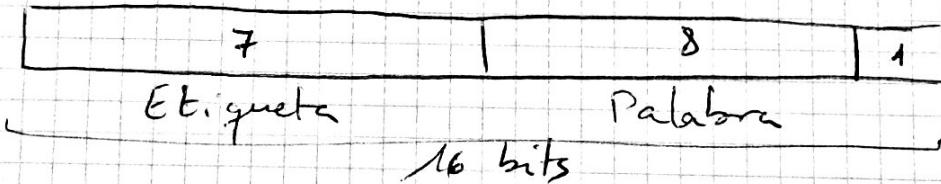
$$\leq 15.0'95 + 0'05 \cdot T_{f_1} < 15.0'97 + 0'03 \cdot T_{f_2} ? \quad [\text{Supongo } T_{f_1} = T_{f_2}]$$

$$\Rightarrow T_d > 252'5 \text{ ns}$$

* Mem. principal = 64 K palabras $T_{mp} = 700 \text{ ns}$
Palabra = 16 bits

Caché totalmente asociativa con tamaño de 2^k palabras y tamaño de bloque 256 palab./bloque. Algorit. de reemplazo: MRU. T_{acceso} = 100ms. Ejecuto bucle 5 veces que accede a 2301 pal.

a) ¿Cuáles son los campos de la dir. de MP?



$$MP = 64 \text{ kPa} = 2^6 \cdot 2^{10} = 2^{16} \text{ Pa}.$$

Tamano bloque = 256 pal = 2^8 pal.

b) ¿Tiempo accesos sin cache?

$$2301.5 \cdot 700 = 8053'5 \text{ seq}$$

c) Realizar una tabla con 5 columnas, 1 por cada vez que se ejecute el bucle y tantas filas como bloques tenga la cabé. En cada columna ponga el valor de la estig. existente en cada bloque al finalizar cada una de las 5 iteraciones

Algoritmo de reemplazo : el usado más reciente



Dir etiqueta

			It.1	It.2	It.3	It.4	It.5
0-255	00	B0	00	00	00	00	00
256-511	01	B1	01	01	01	01	01
512-767	02	B2	02	02	02	02	02
768-1023	03	B3	03	03	03	03	03 04
1024-1279	04	B4	04	04	04	04 05	05
1280-1535	05	B5	05	05	05 06	06	06
1536-1791	06	B6	06	06 07	07	07	07
1792-2047	07	B7	08 07	08	08	08	08
2048-2303	08						

9 fallos 1 fallo 1 fallo 1 fallo 1 fallo
13 fallos



Tengo el fallo con el Ø6 debido
a la política de reemplazo MRU
(reemplazo el que ha sido usado
más recientemente).