

# Tema 1

$$\underline{\text{rendimiento}} = \frac{1}{\text{ejecución}}$$

$$\frac{\text{rendimiento} X}{\text{rendimiento} Y} = \frac{\text{ejecución} Y}{\text{ejecución} X} = R \text{ (ratio)}$$

$\uparrow$  rendimiento  $\Rightarrow \downarrow$  ejecución

$$\begin{aligned} 1\mu\text{s} &= 10^{-6}\text{s} \\ 1\text{n}s &= 10^{-9}\text{s} \\ 1\text{ps} &= 10^{-12}\text{s} \end{aligned}$$

$$t_{clk} = \frac{1}{f}$$

$\downarrow$  rendimiento  $\Rightarrow \uparrow$  ejecución

$$\begin{aligned} 1\text{MHz} &= 10^6 \\ 1\text{GHz} &= 10^9 \\ 1\text{THz} &= 10^{12} \end{aligned}$$

$$T_{cpu} = T_{cpu\text{hardware}} + T_{cpu\text{sistema}}$$

$$\frac{T_{cpu}}{\text{elapsed time}} = \frac{T_{cpu}U + T_{cpu}S}{\text{elapsed time}} = k \quad k \cdot 100 = j\% \text{ (porcentaje)} \\ \text{ocupación}$$

$$\text{nº ciclos} = NI \cdot CPI \quad T_{cpu} = \text{nº ciclos} \cdot t_{clk} = NI \cdot CPI \cdot t_{clk}$$

$$\underline{\text{CPI promedio}} = \sum_{i=1}^n \frac{NI_i}{NI} \cdot CPI_i$$

$$\begin{aligned} \uparrow T_{cpu} &\Rightarrow \uparrow NI \vee \uparrow CPI \vee \downarrow f \quad f = \frac{1}{t_{clk}} \\ \downarrow T_{cpu} &\Rightarrow \downarrow NI \vee \downarrow CPI \vee \uparrow f \end{aligned}$$

$$\underline{\text{MIPS}} = \frac{NI \text{ (millones)}}{T_{cpu} \text{ (segundos)}}$$

$$\begin{aligned} \uparrow \text{MIPS} &\Rightarrow \downarrow T_{cpu} \vee \uparrow NI \\ \downarrow \text{MIPS} &\Rightarrow \uparrow T_{cpu} \vee \downarrow NI \end{aligned}$$

$$\underline{\text{MFLOPS}} = \frac{\text{FLOP (OP. Punto + Blk, en millones)}}{T_{cpu} \text{ (segundos)}} \quad \begin{aligned} \uparrow \text{MFLOPS} &\Rightarrow \downarrow T_{cpu} \vee \uparrow \text{FLOP} \\ \downarrow \text{MFLOPS} &\Rightarrow \uparrow T_{cpu} \vee \downarrow \text{FLOP} \end{aligned}$$

Cuidado con los MIPS y MFLOPS, puede darse el caso de que ambos aumenten y que también aumente el  $T_{cpu}$ . Es decir, si aumentamos  $NI/FLOP$  podemos entascarar un aumento en  $T_{cpu}$  ya que MIPS/MFLOPS aumentan también.

$$\underline{\text{SPECspeed}_{2017}} = \sqrt[n]{\prod_{i=1}^n R_i} = \sqrt[n]{R_1 \cdot R_2 \cdot \dots \cdot R_n} \quad R_i = \frac{\text{ejecución Referencia}}{\text{ejecución } i}$$

NI: Número de instrucciones máquina

CPI: Ciclos por instrucción

t<sub>clk</sub>: Tiempo de reloj.

elapsed time: tiempo de transcurrido en completar una tarea.

## Ley de Amdahl

$$S = \text{Speed-up (velocidad)} = \frac{T_{CPU,SM}}{T_{CPU,CM}}$$

$$S_m = \text{Factor de mejora subsistema} = \frac{T_{CPU,SM}}{T_{CPU,CM}}$$

$$F_m = \text{Fracción de uso subsistema} = \frac{T_{CPU,SM}}{T_{CPU,SM}}$$

$$S = \frac{1}{1-F_m + \frac{F_m}{S_m}}$$

### Casos Límites

$$0 \leq F_m \leq 1$$

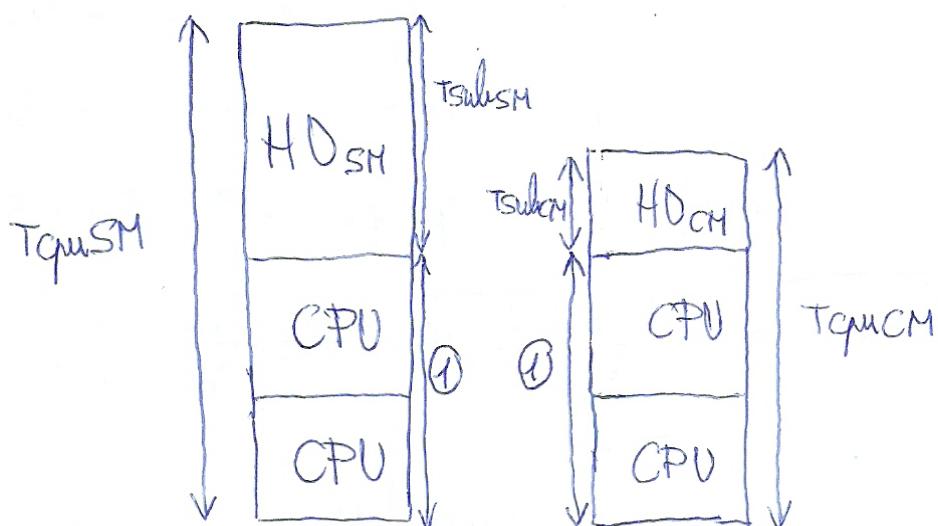
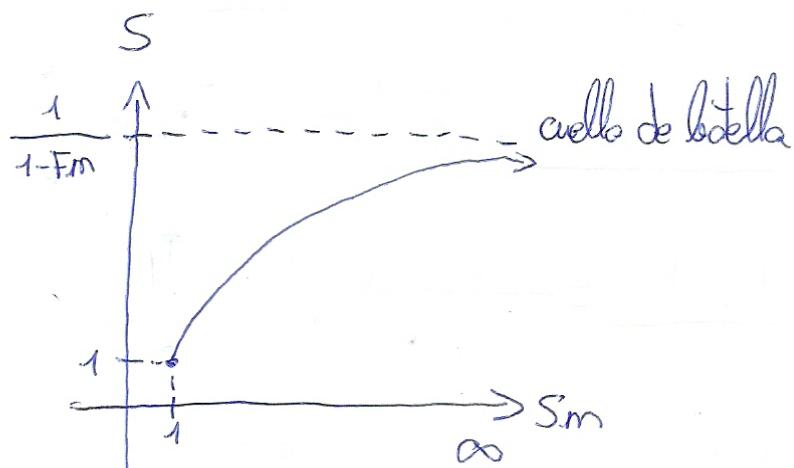
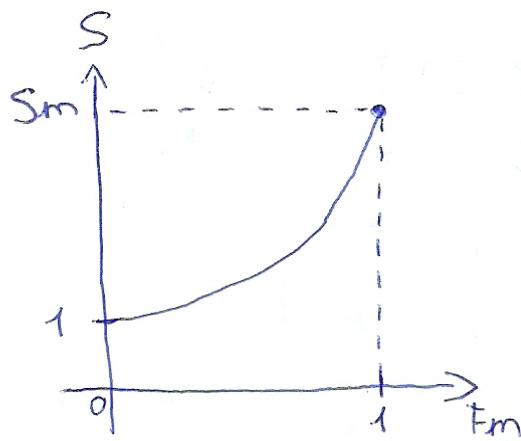
$$S_m \geq 1 (\infty)$$

$$F_m = 0 \Rightarrow S_m = 1$$

$$S_m = 1 \Rightarrow S = 1$$

$$F_m = 1 \Rightarrow S = S_m$$

$$S_m \rightarrow \infty \Rightarrow S = \frac{1}{1-F_m}$$



$$\textcircled{1} \quad T_{CPU,SM} - T_{SUB,SM} = T_{CPU,CM} - T_{SUB,CM}$$

# Ley de Amdahl

## Notación

S:	Speed-up.
TcpuCM:	Tiempo cpu con mejora.
TcpuSM:	Tiempo cpu sin mejora.
TsubCM:	Tiempo subsistema con mejora.
TsubSM:	Tiempo subsistema sin mejora.
FM:	Fracción de uso subsistema.
SM:	Factor de mejora subsistema.

*Ley de Amdahl para expresar la ganancia S de dos subsistemas conocido FM y SM de ambos subsistemas.*

Expresamos S como

$$S = \frac{TcpuSM}{TcpuCM}$$

Donde TcpuCM es

$$TcpuCM = (TcpuSM - TsubSM_1 - TsubSM_2) + TsubCM_1 + TsubCM_2$$

Sustituimos TsubSM y TsubCM por sus expresiones sacando TcpuSM factor común

$$TsubSM = FM \cdot TcpuSM \quad TsubCM = \frac{TsubSM}{SM} = FM \cdot TcpuSM \cdot \frac{1}{SM}$$

$$TcpuCM = TcpuSM(1 - FM_1 - FM_2) + TcpuSM \frac{FM_1}{SM_1} + TcpuSM \frac{FM_2}{SM_2}$$

Substituimos TcpuCM en la expresión de S

$$S = \frac{TcpuSM}{TcpuSM(1 - FM_1 - FM_2) + TcpuSM \frac{FM_1}{SM_1} + TcpuSM \frac{FM_2}{SM_2}}$$

Y simplificamos la expresión. Hallando la **expresión final** buscada.

$$S = \frac{1}{(1 - FM_1 - FM_2) + \frac{FM_1}{SM_1} + \frac{FM_2}{SM_2}}$$

**Ley de Amdahl para expresar la ganancia  $S$  de  $n$  subsistemas conocido  $FM$  y  $SM$  de los  $n$  subsistemas.**

Siguiendo los pasos del apartado anterior podemos hallar una **expresión general** para  $n$  subsistemas conociendo  $FM$  y  $SM$  de los  $n$  subsistemas.

$$S = \frac{1}{\left(1 - \sum_{i=1}^n FM_i\right) + \sum_{i=1}^n \frac{FM_i}{SM_i}}$$

• Dependencias  
A) Dependencias estructurales

$J_1$  e  $J_2$  que usan el mismo recurso (U.F)  $\rightarrow$  dep. estructural.

\*\* Riesgo estructural.

$J_1$  e  $J_2$  usan el mismo recurso en el mismo ciclo.

Solución HW:

\* Detener el cache  $\rightarrow$  Ejemplo:

\* Replicar el recurso

Código Univ IF+D

$J_W$  IF: JO EX: M.WB

$J_1$  ... IF: JO EX: M.WB

$J_2$  ... IF: JO EX: M.WB

$J_3$  ... IF: JO ...

↑  
nop { Detención  
→ Tick perdida (+ stall)

(Reading) (Waiting)

B. Reg. ordena el acceso a R y W.

$\Rightarrow$  W ocurre antes que las R  $\Rightarrow$  Anticipación B. Reg.

B) Dependencias de datos

$J_1$  e  $J_2$  tienen dependencia de datos si ambas acceden al mismo operando y uno de los accesos es de escritura, caso mínimo.

B-1) Dependencia verdadera.  $\rightarrow$  riesgos RAW (Read after write)

$J_1$  e  $J_2$  tienen dependencia verdadera, si las dos acceden al mismo operando.

$J_1(W)$  e  $J_2(R)$

Ej:  $J_1 \rightarrow add \$7, \$3, \$1$

$J_2 \rightarrow or \$4, \$7, \$0$

$J_3 \rightarrow sub \$2, \$3, \$7$

dep. verdadera a distancia 1  $\rightarrow J_1 - J_2 (\$7)$

dep. " " " 2  $\rightarrow J_1 - J_3 (\$7)$

B-2) Antidependencia  $\rightarrow$  riesgos WAR (Write after read)

$J_1$  e  $J_2$  antidependencia, si las dos acceden al mismo operando  $J_1(R)$  e  $J_2(W)$

Ej:

$J_1 \rightarrow or \$4, \$1, \$8$

$J_2 \rightarrow sub \$1, \$3, \$2$

B-3) Dependencia de salida  $\rightarrow$  riesgo WAW (write after write)  
I<sub>1</sub> e I<sub>2</sub> dep. salida, si las dos acceden al mismo operando.  
 $I_1(W) \in I_2(W)$

Ej: I<sub>1</sub>  $\Rightarrow$  sub  $\$2, \$4, \$3$   
I<sub>2</sub>  $\Rightarrow$  or  $\$2, \$1, \$1$

### c) Dependencias de control

I<sub>1</sub> e I<sub>2</sub> dep. de control, si la ejecución de I<sub>2</sub> depende de la ejecución de I<sub>1</sub>.  
I<sub>1</sub> es una instrucción de salto.

Ej: I<sub>1</sub>  $\Rightarrow$  beg \$5, \$4, Fin      I<sub>1</sub> e I<sub>2</sub> dep. control  $\xrightarrow{\text{MIPS}}$  riesgo control.  
I<sub>2</sub>  $\Rightarrow$  or \$3, \$2, \$1      I<sub>1</sub> e I<sub>3</sub> Fin

Fin: I<sub>3</sub>  $\Rightarrow$  sub \$7, \$0, \$0

## Riesgos datos MIPS

### • Riesgos RAW (dep. verobadera, RAW)

Si Anticipación B, Beg  $\rightarrow$  Dep. Verobaderas d<sub>1</sub>, d<sub>2</sub>  $\rightarrow$  RAW

No Anticipación B, Beg  $\rightarrow$  Dep. verobaderas d<sub>1</sub>, d<sub>2</sub>, d<sub>3</sub>  $\rightarrow$  RAW

### • Riesgos WAR (antidependencia, WAR) $\rightarrow$ No WAR

I<sub>1</sub>  $\rightarrow$  or \$3, \$1, \$8 ... , IF ID EX M WB

I<sub>2</sub>  $\rightarrow$  sub  $\$1, \$3, \$0, \dots, \text{IF ID EX M WB}$

### • Riesgo WAW (dep. salida, WAW) $\rightarrow$ No WAW

I<sub>1</sub>  $\rightarrow$  sub \$7, ... , \text{If ID EX M WB}

I<sub>2</sub>  $\rightarrow$  or \$7, ... , \text{If ID EX N WB}

Ej: MIPS modificadas  $\Rightarrow$  add  $\rightarrow$  EX(3ck)       $\Rightarrow$  Si se desordenan W  
or  $\rightarrow$  EX(1ck)       $\hookrightarrow$  Riesgo WAW

I<sub>1</sub>  $\rightarrow$  add \$1 ... , If ID EX1 EX2 EX3 M WB  
I<sub>2</sub>  $\rightarrow$  or \$1 ... , If ID EX1 M WB

## Tratamiento riesgo RAW en MIPs

A) Aprox. SW  $\rightarrow$  compilador

A-1) Introducir nops

A-2) Reordenar instrucciones.

Ej: I<sub>1</sub>: add \$1, \$1  
I<sub>2</sub>: sub \$4, \$1

compilar

I<sub>1</sub>: add \$1  
nop  $\rightarrow$  1ck perdida  
nop  $\rightarrow$  1ck perdida  
I<sub>2</sub>: sub \$4, \$1

Ej: I<sub>1</sub>: add \$1, \$1  
I<sub>2</sub>: sub \$4, \$1  
I<sub>3</sub>: or \$7, \$2, \$4

compilar  
No se puede reordenar

B) Aprox. HW  $\rightarrow$  detención riesgos  $\rightarrow$  Etapa ID

B-1) Detención (detention)

B-2) Anticipación (forwarding, cortocircuito)

B-1) Detención

$\rightarrow$  Suspend (stall) + Para el cause (para IF/ID, para PC)

I<sub>1</sub>: add \$1, \$1... IF ID EX M WB

I<sub>2</sub>: sub \$4, \$1... IF ID EX M WB

I<sub>3</sub>: and \$6, \$1... IF ID EX M WB

La memoria

2 stalls = 2 ck perdida

B-2) Anticipación  $\rightarrow$  Añadir caminos adicionales al procesador (cortocircuitos)

I<sub>1</sub>: add \$1, \$1... IF ID EX M WB

I<sub>2</sub>: sub \$4, \$1... IF ID EX M WB

I<sub>3</sub>: and \$6, \$5, \$1... IF ID EX M WB

ostall = 0 ck perdida

EX-EX M-EX

## Ejemplos cortocircuitos MIPs

• A/L - A/L d1

EX-EX

I<sub>1</sub>: add \$1, \$1... IF ID EX M WB

I<sub>2</sub>: sub \$3, \$1... IF ID EX

• A/L - A/L d2

M-EX

I<sub>1</sub>: add \$1, \$1... IF ID EX M WB

I<sub>2</sub>: sub \$4, \$1, \$1... IF ID EX M WB

I<sub>3</sub>: sub \$4, \$1, \$1... IF ID EX M WB

• A/L - Salto Condicional J1

I<sub>1</sub>: add \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: beq \$3, \$1, J1, IF ID, EX

EX-EX

• A/L - LW J1

I<sub>1</sub>: add \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: lw \$3, \$1, J1, IF ID EX M WB

EX-EX

• A/L - LW J2

I<sub>1</sub>: add \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: ..., IF ID EX M WB  
 I<sub>3</sub>: lw \$3, \$1, J2, IF ID EX M WB

M-EX

• A/L - SW J1

I<sub>1</sub>: add \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: sw \$3, \$1, J1, IF ID EX M WB

EX-EX

I<sub>1</sub>: add \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: sw \$1, \$3, J1, IF ID EX M WB

M-M

• A/L - SW J2

I<sub>1</sub>: add \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: ..., IF ID EX M WB  
 I<sub>3</sub>: sw \$1, \$3, J2, IF ID EX M WB

M-M

• LW - A/L J1

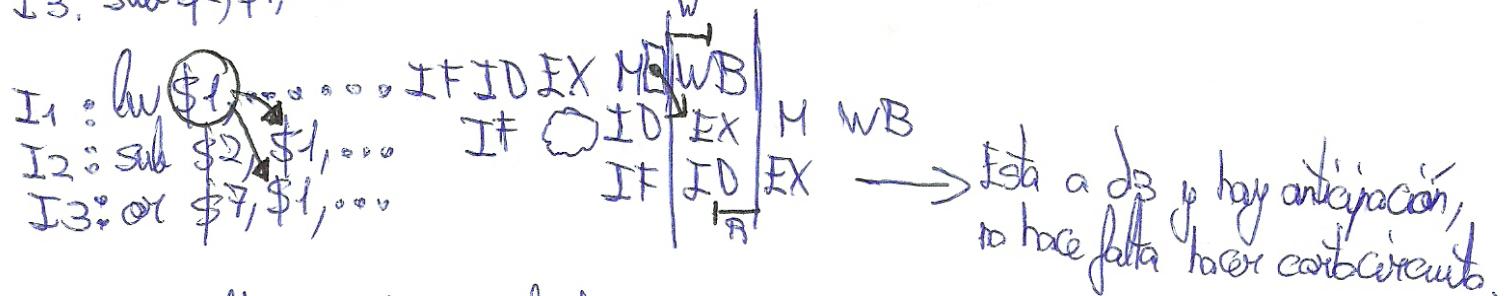
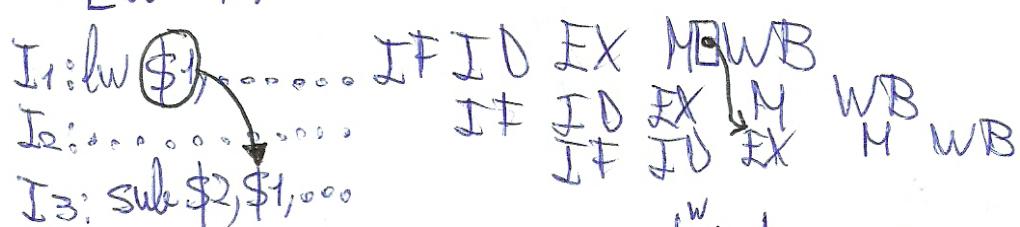
I<sub>1</sub>: lw \$1, ..., IF ID EX M WB  
 I<sub>2</sub>: sub \$7, \$1, ..., IF ID EX M WB  
 I<sub>3</sub>: add \$1, ..., IF ID EX M WB

M-EX + 1 bucleja detención

→ hace falta un stall.

load retardado

- LW - A/L  $\delta_2 \Rightarrow$  (caso \*)

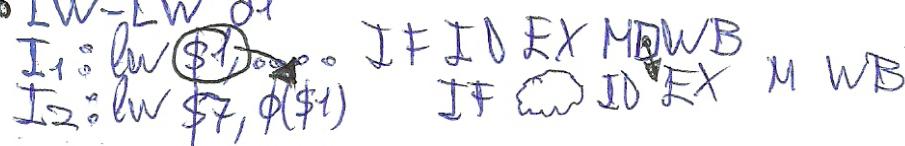


- LW - Salta condicional  $\delta_1$



- LW - Salta condicional  $\delta_2 \Rightarrow$  ver (caso \*)

- LW - LW  $\delta_1$

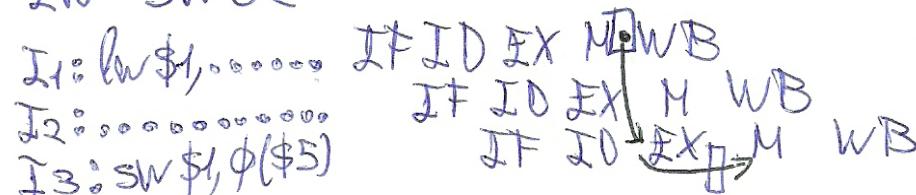


- LW - LW  $\delta_2 \Rightarrow$  ver (caso \*)

- LW - SW  $\delta_1$



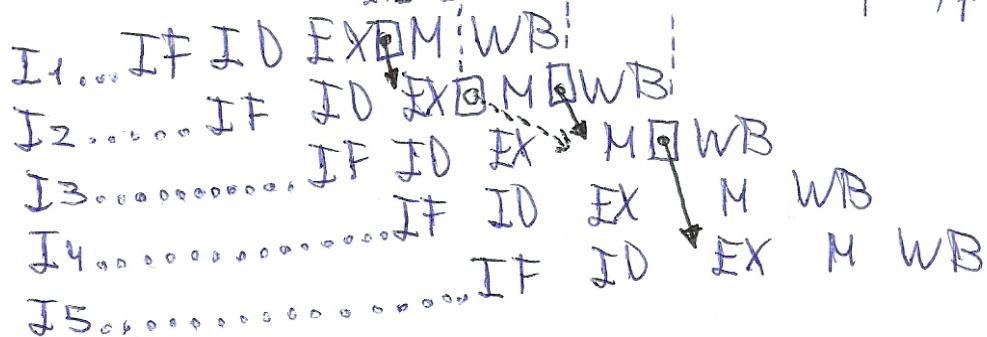
- LW - SW  $\delta_2$



Inst. Productoras: A/L, LW

Inst. Consumidoras: A/L, B/Q, LW/SW, SW(rt)

(EX) (EX) (EX) (M) → No se puede, por eso está en discontinua.



EX-EX

EX - M → No es factible temporal

M - EX

M - M

## Tratamiento riesgos de control

AdmPC

- J1: beg ... IF ID EX M; WB  
 J2: sub ... IF ID EX M; WB → Error  
 J3: ... IF ID EX M; WB → Error  
 J4: ... IF ID EX M; WB → Error  
 Instino.

### A) Tratamiento SW

- A-1) Inserción mips tras el salto  
 A-2) Planificación inst. tras el salto

### B) Tratamiento HW

- B-1) Detención  
 B-2) Predicción

B-1) Detención

beg IF ID EX M; WB

3dk

+ 2dk

AdmPC

beg IF ID EX M; WB

? idk

beg IF ID EX M; WB

skip  
salto;  
cond;  
salto;  
act.  
PC

→ beg IF ID EX M; WB

AdmPC

Isigdest. IF IF IF IF IF

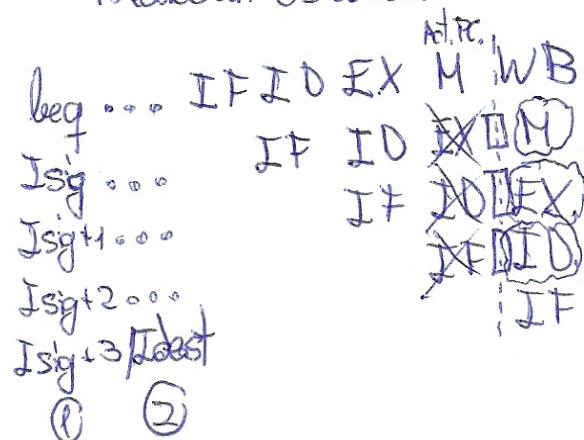
San flush

flush ≠ stall

## B-2) Predicción (estática, dinámica)

E.C.

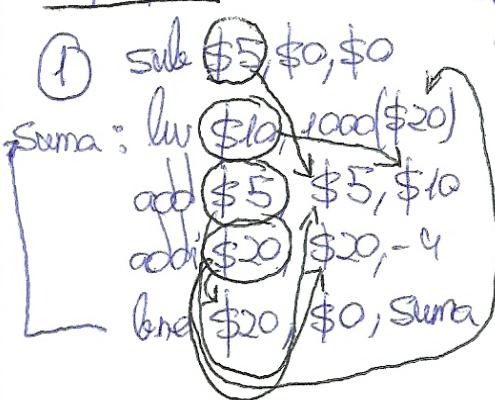
### Predicción estática



① Acertó en Predicción  $\rightarrow$  ok período  
No salto

② Falló en Predicción  $\rightarrow$  Zack período.  
Salto

5/11/18



a) hipótesis  $N^{\circ}$  veces ejecuta bucle =  $\frac{n}{4} = N$   
 $\$0 \leftarrow n$

b) Dependencias de datos

Ventaduras RAW

- J<sub>1</sub>-J<sub>3</sub> (\$5)
- J<sub>2</sub>-J<sub>3</sub> (\$10)
- J<sub>3</sub>-J<sub>3</sub> (\$5)
- J<sub>4</sub>-J<sub>4</sub> (\$20)
- J<sub>4</sub>-J<sub>5</sub> (\$20)
- J<sub>4</sub>-J<sub>2</sub> (\$20)

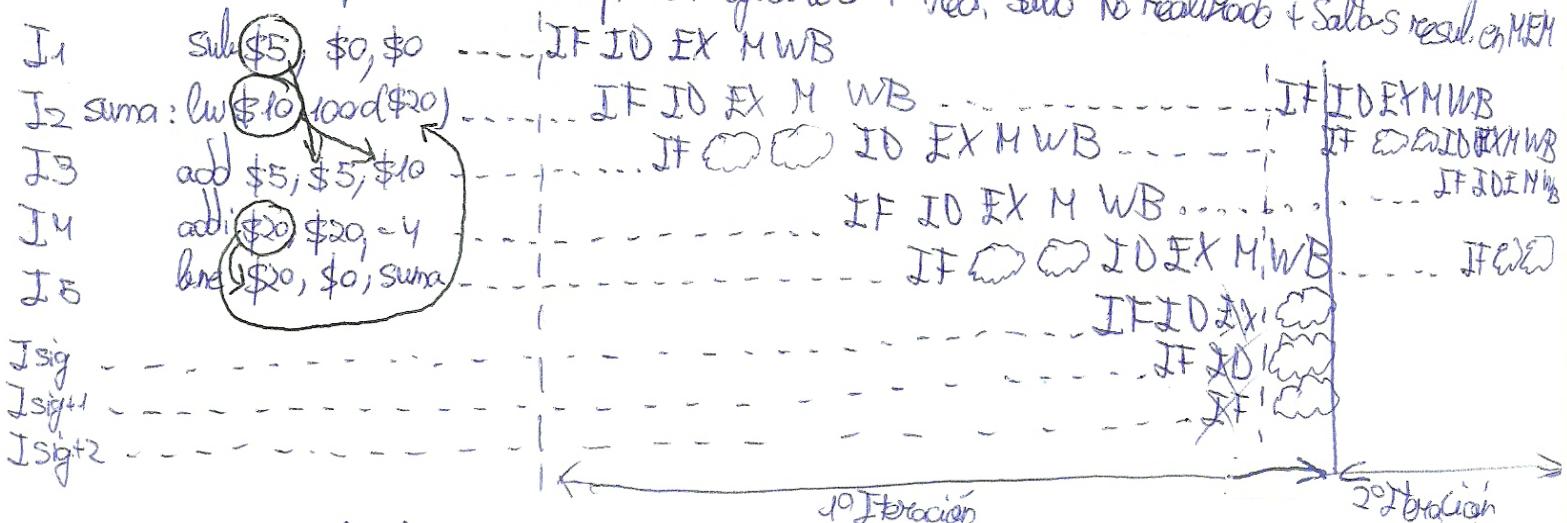
Antidependencias WAR

- J<sub>2</sub>-J<sub>4</sub> (\$20)
- J<sub>3</sub>-J<sub>3</sub> (\$5)
- J<sub>3</sub>-J<sub>2</sub> (\$10)
- J<sub>4</sub>-J<sub>4</sub> (\$20)
- J<sub>4</sub>-J<sub>5</sub> (\$20)
- J<sub>5</sub>-J<sub>4</sub> (\$20)

Salida WAR

- J<sub>1</sub>-J<sub>3</sub> (\$5)
- J<sub>3</sub>-J<sub>2</sub> (\$5)
- J<sub>4</sub>-J<sub>4</sub> (\$20)
- J<sub>2</sub>-J<sub>2</sub> (\$10)

d) HW detención dep. datos + Anticip. B. registros + Pred. salto No realizados + saltos result. en MEF

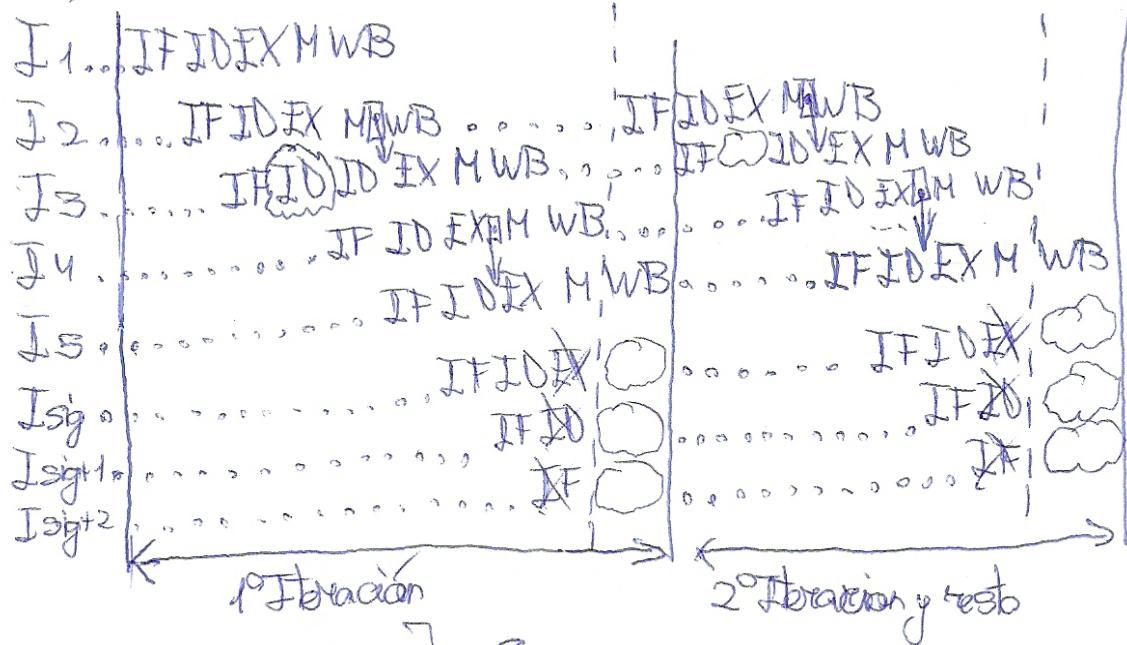


$$CP_{th} = \frac{13 + 11 \cdot (N-1)}{1 + 4 \cdot N} \Bigg|_{N=20} = \frac{11}{4}$$

2<sup>da</sup> iteración  
y resto iteraciones

Con shortcuts

d2)



$$CP_{Jd2} = \frac{10 + 8 \cdot N - 1}{1 + 4 \cdot N} \Big|_{N \rightarrow \infty} = \frac{8}{4}$$

d3)

$$S_2 = \frac{\text{tqm monacido}}{\text{tqm segm2}}$$

$$CP_{\text{monacido}} = 1 \\ n^{\circ} \text{acidos monacido} = 1 + 4 \cdot N$$

$$S_1 = \frac{\text{tqm monacido}}{\text{tqm segm2}}$$

$$S_2 = \frac{N! \cdot CP_{\text{mc}} \cdot t_{\text{kmc}}}{N! \cdot CP_{Jd2} \cdot t_{\text{ckd2}}} = \frac{1}{\frac{10 + 8(N-1)}{1 + 4N}} \cdot \frac{5!}{5!} = \frac{(1+4N)! \cdot 5}{10 + 8(N-1)} \Big|_{N \rightarrow \infty} = \frac{20}{8}$$

$$S_1 = \frac{N! \cdot CP_{\text{mc}} \cdot t_{\text{kmc}}}{N! \cdot CP_{Jd2} \cdot t_{\text{ckd2}}} = \frac{(1+4N)! \cdot 5}{10 + 11(N-1)} \Big|_{N \rightarrow \infty} = \frac{20}{11}$$

Salt se resuelve NEM

Beg IF INDEX M, WB

salto se resuelve en Ex, salto se resuelve en ID

beg IF ID EX; M WB      beg IF ID; EX M WB

Ad. PC  
IF ID; EX N WB  
EW;  
+ Adverb c.c EX-JD

Salto se resuelve en JD + Activar cc. (Anticip. c.c.)  
+ Riesgo control detención + Activar baraja  
- Si no hay anticipación B, Reg.

Ej: Sub \$3,\$4... IF ID EX MEWB  
or \$1,... IF ID EXP M WB  
leg \$1,\$3,etc IF ID ( )

Isig

## Tratamiento HW riesgo control

A- Detención (Insertas flush hasta que se resuelva el salto).

## B- Predicción

## B-1) Estática

## B-2) Dinámica.

## B-1) Estatística

B-1-1) Pedición estatal "Sale No Tomad"

④ Salto no salto = Acérto en Predicación

beg INDEX M.WB

ISg IF ID EX. M  
TT EA, EX

10

$\text{sig}^+$

Isigt2

J  
Isig + 3

② Salto Salta = No acierto en predicción  $\Rightarrow$  3 ciclos perdidos

beg IF ID EX M:WB

Jsig If I do this

Sight

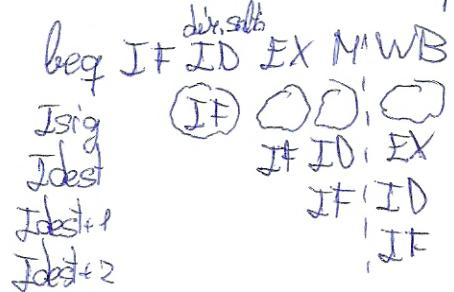
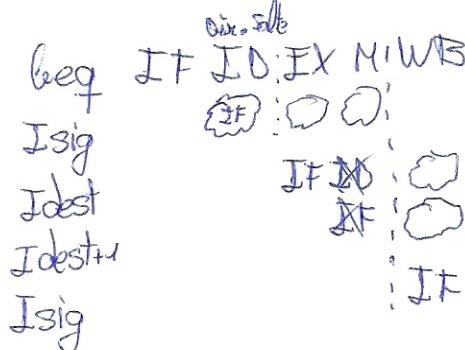
$$\sum_{i=1}^{n-1} \frac{1}{i+1} < \ln n$$

desl

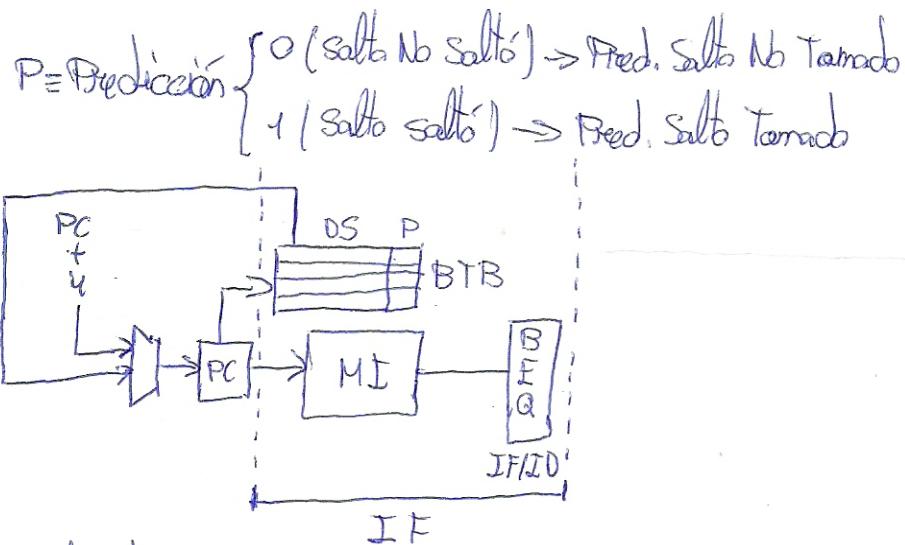
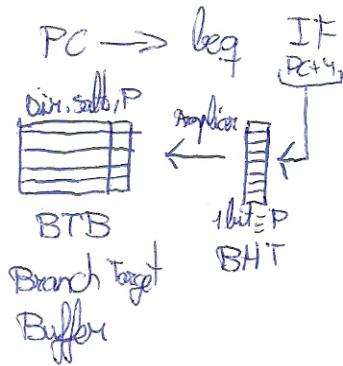
## B-1-2) Predicción estática "Salto tomado"

Se resuelve en MEM

- ① Salto No Saltó = Falla predicción  $\Rightarrow$  3 ciclos perdidos    ② Salto Saltó = Acierto en predicción

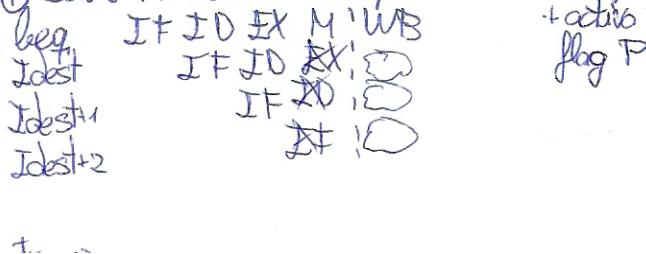


## B-2) Dinámica

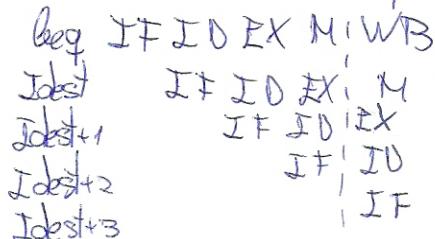


Ej: BTB en IF, salto registrado BTB, P = 1

- ① Salto no Saltó = Falla predicción  $\Rightarrow$  3 cl. perd.



- ② Salto saltó = Acierto predicción



③ MIPS clase + Anticip. B. Reg. Salto se resuelve en M + Reg. Salto No Tomado.

E.C.

Caso 1 → Riesgos dep. datos → Detención (Stall, buclejitos)

I1	or \$3, \$0, \$2
I2	add \$4, \$2, \$3
I3	and \$7, \$3, \$4
I4	beq \$3, \$2, fin
I5	lwne \$4, \$7, ite
I6 fin:	lw \$3, loc(\$7)

$$CPI = \frac{n^{\circ} ciclos}{n^{\circ} instrucciones} \leftarrow n^{\circ} instrucciones que ocasionan (Post-escritura)$$

Sin anticipación

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
J1	IF ID EX M WB															
J2	IF D ⊂ ID EX M WB															
J3		IF D ⊂ ID EX M WB														
J4			IF D ⊂ ID EX M WB													
J5				IF D ⊂ ID EX M WB												
J6					IF D ⊂ ID EX M WB											

Con anticipación

	0	1	2	3	4	5	6	7	8	9	10	11
J1	IF ID EX M WB											
J2	IF ID EX M WB											
J3	IF ID EX M WB											
J4	IF ID EX M WB											
J5	IF ID ⊂											
J6	IF ID, IF ID EX M WB											

(11)

Ad. PC

j	IF ID EX M WB
	○ ○
beq	IF ID EX M WB
	○ ○

20% bw/sw

40% A/L

25% beq { 80% saltos saltan  
20% saltos no saltan

5% j, jal

10% jal

} Saltos incondicionales

aj) Predicción Saltos no tomados

Importante

N = n° total instrucciones

$$CPI_{MIPSclase} = \frac{N \cdot 1 + 0'25N \cdot (0'8 \cdot 3 + 0'2 \cdot 0) + 0'15N \cdot 3}{N} = 2.05$$

$$CPI_{MIPSnuevo} = 1 + 0'25 \cdot 0'8 \cdot 2 + 0'15 \cdot 1 = 1.55$$

$$S = \frac{205}{1.55} = 132$$

## a) Predicción Salto tomado

	Salto no Salta	Salta Salta
MIPSbase	3 / -	1 / 1
MIPSmovo	2 / -	1 / 1

$$CPI_{MIPSbase} = 1.45$$

$$CPI_{MIPSmovo} = 1.1$$

$$S = \frac{1.45}{1.1} = 1.64$$

c) Predicción Salto tomado. Su CPI<sub>MIPSmovo</sub> es menor en b) que en a)

(10)

- I<sub>1</sub> add \$9, \$0, 0  
I<sub>2</sub> add \$10, \$0, 58(\$9)  
I<sub>3</sub> multi \$10, \$10, 4  
I<sub>4</sub> lw \$3, 0(\$11)  
I<sub>5</sub> sw \$3, 0(\$10)  
I<sub>6</sub> sw \$3, 4(\$10)  
I<sub>7</sub> lw \$3, 4(\$11)  
I<sub>8</sub> sub \$9, \$9, 0(\$4)

a) Tipos de saltos

A<sub>2</sub>-A<sub>1</sub>A<sub>2</sub>-A<sub>2</sub>M-A<sub>1</sub>

M-M

A<sub>2</sub>-MProductivas | Consumidoras

LW(M)

A<sub>1</sub>L (A<sub>2</sub>)A<sub>1</sub>L A<sub>2</sub>A<sub>1</sub>A<sub>1</sub>SW A<sub>1</sub> Mbeg A<sub>2</sub>

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
I <sub>1</sub>	IF	JG	A <sub>1</sub>	M	A <sub>2</sub>	WB												
I <sub>2</sub>	IF	○	○	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB										
I <sub>3</sub>				IF	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB									
I <sub>4</sub>				IF	○	○	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB							
I <sub>5</sub>					IF	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB								
I <sub>6</sub>						IF	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB							
I <sub>7</sub>							IF	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB						
I <sub>8</sub>								IF	○	ID	A <sub>1</sub>	M	A <sub>2</sub>	WB				

$$CPI = \frac{18}{6}$$

## Tratamiento SW riesgos control

A) Inserción hop

B) Salto retardado (Delayed Branch)

MIPS  
CS  
AT PC  
beq, j, If ID  
→ Salto = latencia f.c.k  
EX M WB (1 delay slot, + hueco retardo)

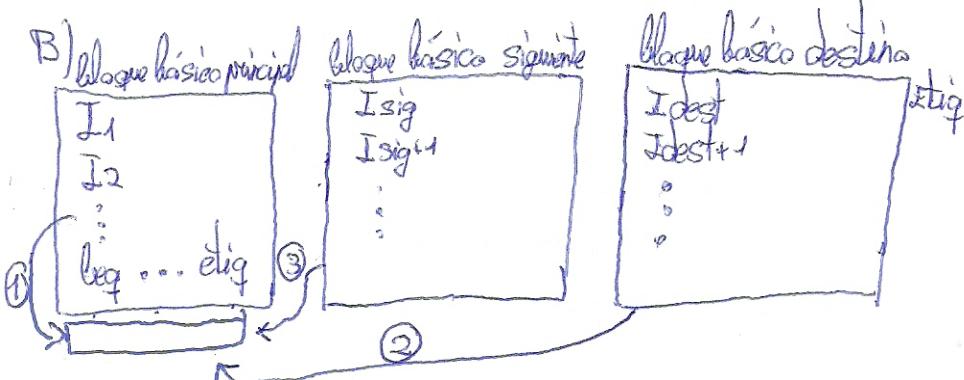
A) Inserción hop

beq .....  
top

B) Salto retardado

beq .....

← mover inst. programa.



③ mover Inst. de B.B.P

BB  
add \$1,\$2,\$3  
beq \$2,\$0,etiq

beq \$2,\$0,etiq.. IF ID EX M WB  
add \$1,\$2,\$3 .... IF ID EX M WB  
or \$7,\$3,\$1 .... IF  
etiq: sub \$2,\$0,\$0. .... IF

Isig: or \$7,\$3,\$1

Idest: sub \$2,\$0,\$0

④ mover Inst. de B.B.D

BB  
add \$2,\$0,\$3  
beq \$2,\$0,etiq

> X. Hay dependencia, no puedo  
Isig: or \$4,\$3,\$1

Idest: sub \$4,\$4,\$0

Idest+1

beq \$2,\$0,etiq.. IF ID EX M WB

sub \$4,\$0,\$0.... IF ID EX M WB

or \$4,\$3,\$1.... IF

solo si hay escritura-escritura.

Idest.... IF

③ Haber, Inst de B.I.B.S

BBP [odd \$2, \$0, \$3  
      [ even \$2, \$0, \$3

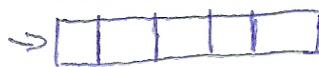
Isig: or \$7, \$3, \$1

dig. Idej. sub \$7, \$3, \$0

Ejemplos ② y ③ son prácticamente iguales; el concepto es el mismo, cambia el lugar de donde cogemos la instrucción que vamos a mover.

## Superscalar - 2

JF JDE EX N WB



$$CPI = \frac{1}{2} \approx 0.5$$

(13) BTB  
Cortacircuitos, Riesgos D. Objs

## Anticipación B. Registros

Saltos se reservan en M

$$\text{Nº especies} = \frac{1000}{8} = N = 125$$

I1 adding \$2, \$0,1000

J2 looks like  $\$3$  or  $\$2$

J3 add (\$4), \$3, \$3

In SW \$4,1000(\$2)

In Silki (\$2, \$2, 8)

T5 Sun (\$2, +2, 3  
T6 home (\$2, \$2, 4, 6)

## Dependencias RAW

$$J_1 - J_2 (\$2)$$

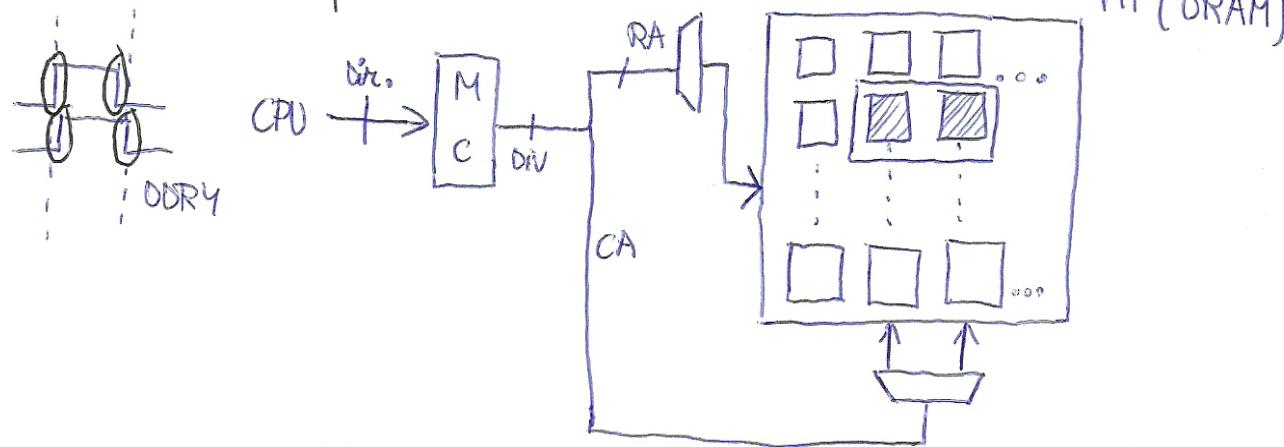
I<sub>2</sub>-I<sub>3</sub>(\$3)

I3-I4/\$4

J5-16 / \$2

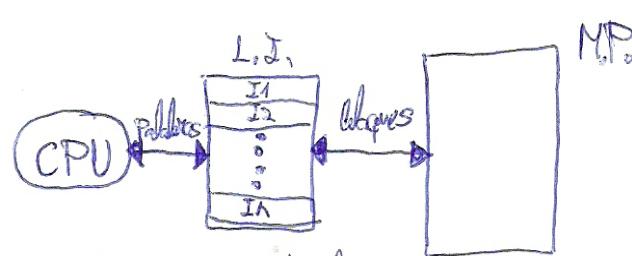
J5-J2 (\$

### Tema 3. Jerarquía de memoria

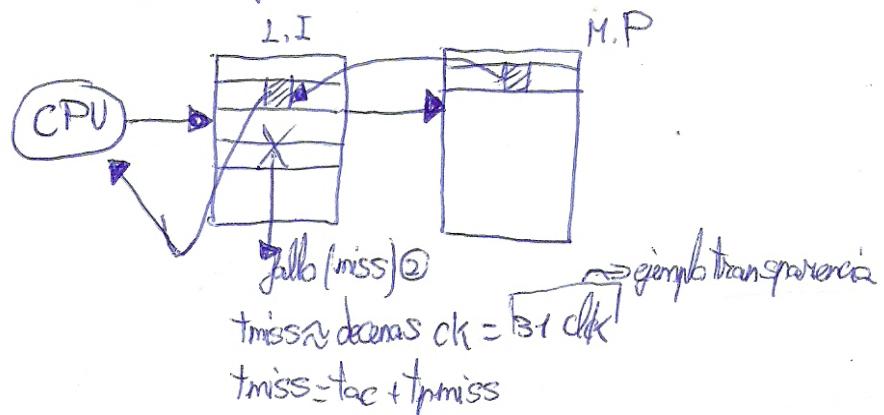
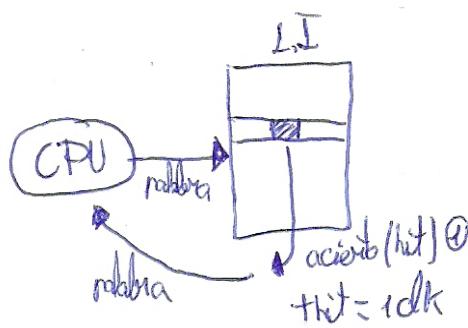


#### Principio de localidad

- Localidad temporal +  $\Delta t$ 
  - $t \rightarrow m$   $\rightarrow [I_1 : m]$
  - $\Delta t \rightarrow m$   $\rightarrow [I_1 : m]$   
  |  
  |  
  |  
  |  
  |  
  |  
 $I_2 : m$   
 $I_3 : m$   
 $I_n : m$
- Localidad espacial. +  $\Delta t$ 
  - $t \rightarrow m$   $\rightarrow [I_1 : m]$
  - $\Delta t \rightarrow m+Dm$   $\rightarrow [I_1 : m]$   
  |  
  |  
  |  
  |  
  |  
  |  
 $I_2 : m+1$   
 $I_3 : m+1$   
 $I_n : m+1$



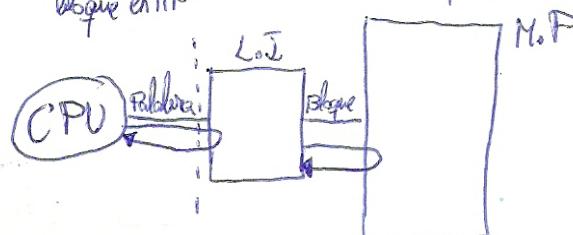
Promedio 96'8% de los accesos se encuentran en M.C.  
Bloque = conjunto de palabras.



$$t_{ac} : t_{hit}$$

$t_{pmiss}$ : tiempo de penalización por fallo =  $t_{acMP} + t_{tr}$

$t_{acMP}$ : acceso al bloque en MP  
 $t_{tr}$ : transporte del bloque por el bus de memoria



$$t_{miss} = t_{hit} + t_{pmiss}$$

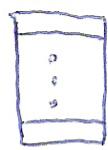
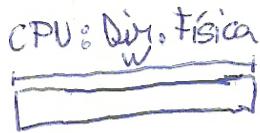
$$P_{hit} = \frac{n_{\text{aciertos}}}{n_{\text{referencias}}}$$

$$P_{miss} = \frac{n_{\text{fallo}}}{n_{\text{referencias}}}$$

$$P_{hit} + P_{miss} = 1$$

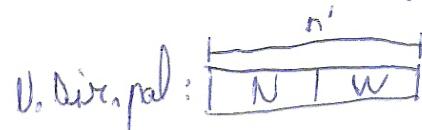
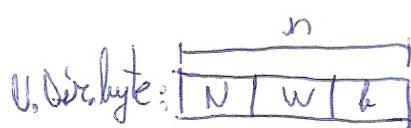
$$\begin{aligned} t_{hit} &= 1 \text{ ck} \\ t_{pmiss} &= 20 \text{ ck} \\ t_{miss} &= 1 + 20 = 21 \text{ ck} \\ P_{miss} &= 0'65 \end{aligned}$$

$$T_{acst} = t_{hit} + P_{miss} * t_{pmiss} = 1 + 0'65 * 20 = 21 \text{ ck}$$



U. Dir: byte  $\Rightarrow$  UF: Tamaño MP (bytes)

$$\left. \begin{array}{l} \text{Ej: } MP = 8M \text{ bytes} \\ U. Dir = \text{byte} \end{array} \right\} B. Dir \Rightarrow n = 23$$



$$\left. \begin{array}{l} \text{Ej. 2: } MP = 8M \text{ bytes} \\ U. dir = 2 \text{ bytes} \end{array} \right\} n = 22$$

b: byte dentro de una palabra  $\Rightarrow 1$  palabra = 2 bytes

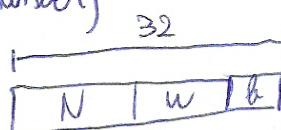
w: palabra dentro de un bloque  $\Rightarrow 1$  bloque = 2<sup>w</sup> pal

N: dir. logica (block number)

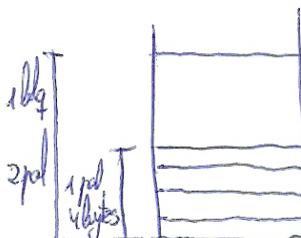
Tamaño NP =  $2^N$  bloques

$$\text{Ej: } MP = 2^{32} \text{ bytes}$$

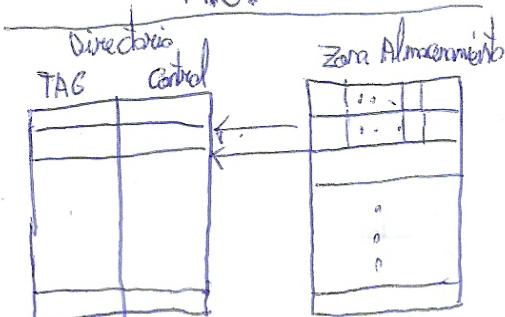
v. dir es el byte



$$\left. \begin{array}{l} 1 \text{ pal} = 4 \text{ bytes} = 2^2 \\ 1 \text{ bloq} = 8 \text{ pal} = 2^3 \\ b = 2 \\ w = 3 \end{array} \right\} N = 32 - w - b = 27$$



M.C.

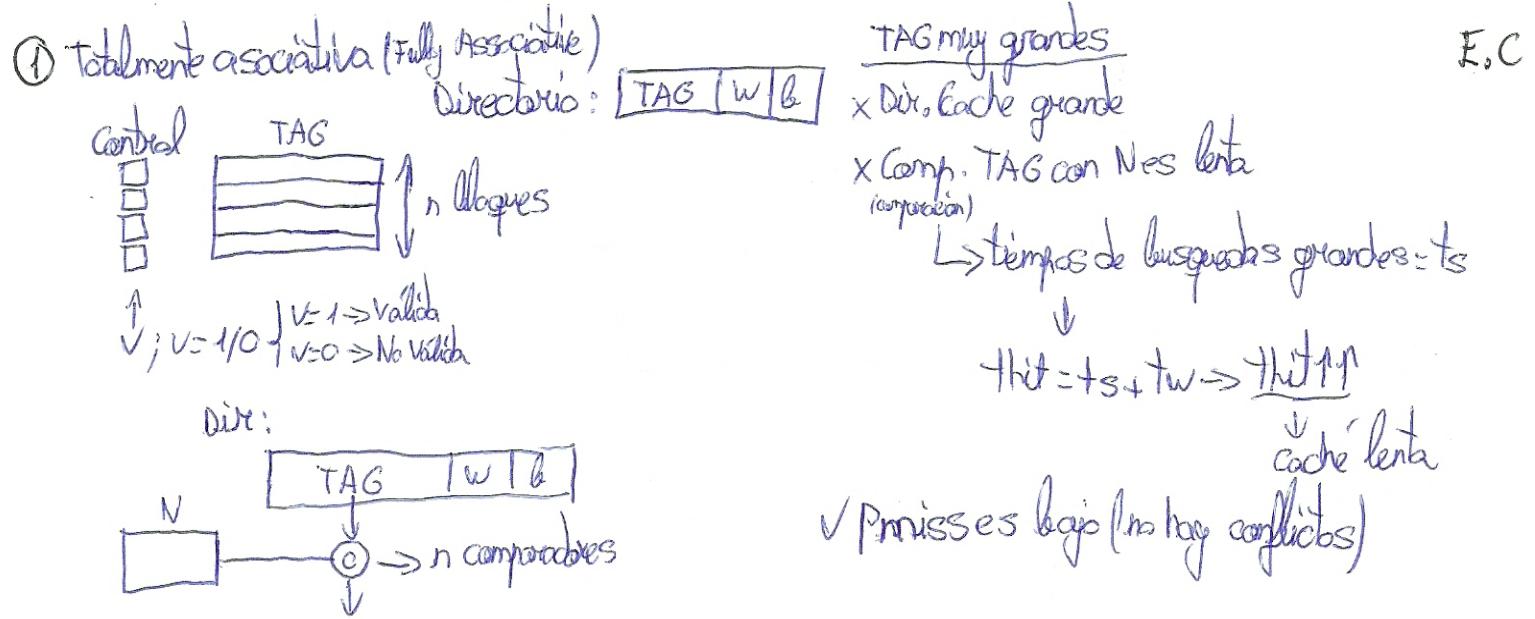


23/11/18

- ① Cómo busco un bloque en Directorio Cache?  $\Rightarrow$  Org. Cache (política de ubicaciones)
- ② Dónde coloco el bloque que me trae desde MP?
- ③ Que pasa si la caché se llena y necesito un nuevo bloque, dónde lo coloco? Política de reemplazo
- ④ Cómo gestiona las escrituras? Política de escritura

### Políticas de ubicación (organización)

- 1- Totalmente Asociativa (Fully Associative)
- 2- Directa (Direct)
- 3- Asociativa por Conjuntos (Set-Associative)

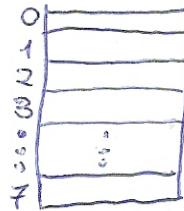


## ② Directa (Direct mapped)

MM Block	Index
0	000000 $\rightarrow$ pos 0
1	000001 $\rightarrow$ pos 1
2	000100 $\rightarrow$ pos 2
3	000101 $\rightarrow$ pos 3
.	.
7	000111 $\rightarrow$ pos 7
8	001100 $\rightarrow$ pos 0
9	001101 $\rightarrow$ pos 1
10	001110 $\rightarrow$ pos 2
.	.
15	001111 $\rightarrow$ pos 7
16	1

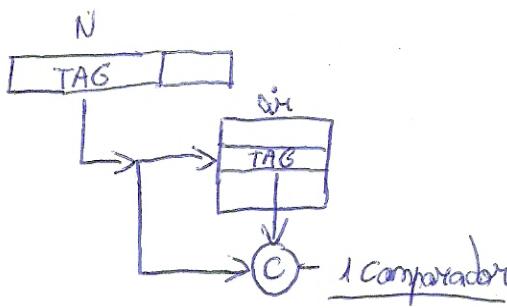
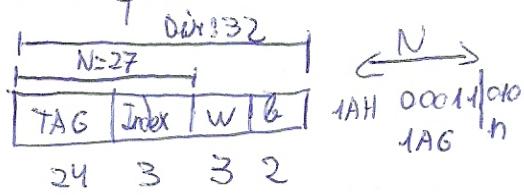
$$N \bmod 8 = N \bmod 2^3$$

CM Block



MIPS 1 bloq = 8 palabras

NC = 8 bloques



TAG más pequeña

$\checkmark$  Comparaciones más rápidas

$t_s \downarrow \Rightarrow t_{hit} \downarrow \Rightarrow MC \text{ rápida}$

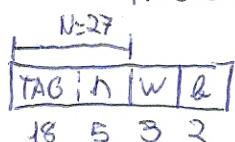
$\checkmark$  1 comparador  $\rightarrow$  Parc. HW control

$\checkmark$  MC Barata

$\times$  Pmiss  $\uparrow$  (Aparecen conflictos)

Ejemplo: MIPS

$$1 \text{ bloq} = 8 \text{ pal} \\ NC = 32 \text{ bloques} = 2^5$$



$$N=2^3$$

$$\underline{101} \mid 00011 \\ \text{TAG}$$

$$N=2^0$$

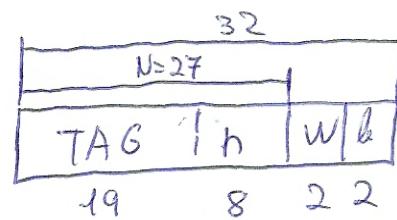
$$\underline{101} \mid 00000 \\ \text{TAG}$$

coinciden los 3 TAG pero son bloques distintos.

Ejemplo: MIPS

$$1 \text{ bloq} = 4 \text{ pal} = 2^2$$

$$MC = 1 \text{ K pal} = \frac{1 \text{ K pal}}{4 \text{ pal/bloq}} = 256 \text{ bloques} = 2^{n-2} = 2^8$$



E.C



## Organización Asociativa por Conjuntos

TAG	Ind	w	b
-----	-----	---	---

$$C \Rightarrow 2^C = n^{\circ} \text{ conjuntos}$$

$$n^{\circ} \text{ conjuntos} * n^{\circ} \text{ vías por conjuntos} = n^{\circ} \text{ bloque}$$

$$2^C * \text{asociatividad} = 2^n$$

- Asociatividad = 1 (1-way) (1 vía por conjunto)

$$2^C = 2^n \Rightarrow \text{Directa} (C=n)$$

- Asociatividad =  $2^n$  ( $2^n$ -way) ( $2^n$  vías por conjunto)

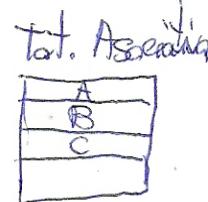
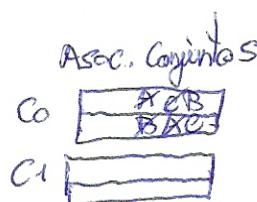
$$2^C = 1 \Rightarrow C = 0 \Rightarrow \text{Tot. Asociativa}$$

$\hookrightarrow$  1 conjunto.

30/11/18

### Tipos de Fallos

- Compulsory  $\rightarrow$  Obligatorio
- Capacity  $\rightarrow$  Capacidad
- Conflict  $\rightarrow$  Conflicto { Directa y Asoc. por conjuntos }  
No aparece en Tot. Asociativa.



### Políticas de reemplazo

- ① First In First Out (FIFO)

A, B, C, D, A, E	1 2 3 4 5 6	A	$\rightarrow$ E
F F F F A F	1 2 3 4 5 6	B	
3		C	
0		D	

- ② Last Recently Used (LRU)

A, B, C, D, A, E	1 2 3 4 5 6	A	
F F F F A F	1 2 3 4 5 6	B	$\rightarrow$ E
3		C	
0		D	

- ③ RANDOM

## Problemas

①

$$B_{Dir} = 32 \text{ bits}$$

v. Dir bytes

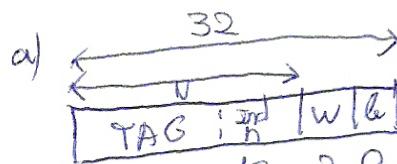
$$MC = 256 \text{ Kbytes}$$

$$1 \text{ pal} = 1 \text{ byte} = 2^0 \text{ bytes} = 2^0$$

$$1 \text{ bloque} = 32 \text{ bytes} = 2^5 \text{ bytes} \Rightarrow 4 \text{ pal} = 2^2 = 2^w$$

$$1 \text{ bloque} = 32 \text{ bytes} = 2^5 \text{ bytes} \Rightarrow 4 \text{ pal} = 2^2 = 2^w$$

$$b) \text{ Tamaño Directorio} = 2^{10} * (t + 1 + 1 + 14)$$



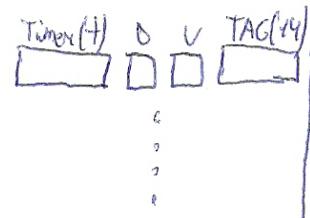
Org. Directa

$$\text{Nº bloques} = \frac{256 \text{ Kbytes}}{4 \text{ bytes}} = \frac{2^8 \cdot 2^{10}}{2^2} = 2^{16} = 2^n$$

\* Reemplazo L R U  $\rightarrow$  Timer (t bits)

\* Post Escritura  $\rightarrow$  D (1 bit)

\* V (Validación) (1 bit)



②

Asociativa por conjuntos - 2 bloques por conjuntos  
(2-ways / 2 vías por conjuntos)

$$B_{Dir} = 32 \text{ bits}$$

v. Dir bytes

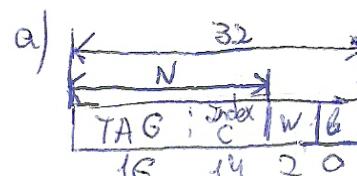
$$1 \text{ pal} = 1 \text{ byte} = 2^0 = 2^0$$

$$\text{Nº bloques} = \frac{2^7 \cdot 2^{10}}{2^2} = 2^{15}$$

$$1 \text{ bloq} = 4 \text{ bytes} = 4 \text{ pal} = 2^2 = 2^w$$

$$\text{Nº conjuntos} = \frac{2^{15}}{2^2} = 2^{14} = 2^c$$

$$MC = 128 \text{ Kbytes}$$



$\phi X 0284A282 \Rightarrow 0000\ 0010\ 1000\ 0000\ |\underbrace{1010}_{2}\ \underbrace{0010}_{8}\ \underbrace{1000}_{A}\ \underbrace{0010}_{0} \Rightarrow C = \phi X 28A0 F$

$\phi X 01148C89 \Rightarrow 0000\ 0001\ 0001\ 0000\ |\underbrace{1000}_{2}\ \underbrace{1100}_{3}\ \underbrace{1000}_{2}\ \underbrace{1001}_{2} \Rightarrow C = \phi X 2822 F$

$\phi X 0038CF00 \Rightarrow 0000\ 0000\ 0011\ 1000\ |\underbrace{1100}_{3}\ \underbrace{1110}_{3}\ \underbrace{0000}_{C}\ \underbrace{0000}_{0} \Rightarrow C = \phi X 33C0 F$

$\phi X 0038CF01 \Rightarrow 0000\ 0000\ 0011\ 1000\ |\underbrace{1100}_{3}\ \underbrace{1110}_{3}\ \underbrace{0000}_{C}\ \underbrace{0001}_{1} \Rightarrow C = \phi X 33C0 A$



## Ejemplo Análisis Primitivo

よ。C.

Analisis miss  
a) Directa, b) Totalmente asocial; c) Asocial por conjuntos

$$MP = 256 \text{ pal} = 2^8$$

V. D. Dr. ps.

$$\overbrace{1 \text{ kg} = 1 \text{ mol}}^{\text{w}}$$

$$MC = \frac{4}{4} \text{ kbytes} = 1$$

a) Directions

4      2      2

, TAG : n, w,

- 1)  $0x\ DC \rightarrow 1101\ 1100$   $\rightarrow pos=3, TAG=1101 \rightarrow F$
- 2)  $0x\ D8 \rightarrow 1101\ 1000$   $\rightarrow pos=2, TAG=1101 \rightarrow F$
- 3)  $0x\ D2 \rightarrow 1101\ 0010$   $\rightarrow pos=0, TAG=1101 \rightarrow F$
- 4)  $0x\ D9 \rightarrow 1101\ 1001$   $\rightarrow pos=2, TAG=1101 \rightarrow V$
- 5)  $0x\ 52 \rightarrow 0101\ 0040$   $\rightarrow pos=0, TAG=0001 \rightarrow F/R$
- 6)  $0x\ A1 \rightarrow 1010\ 0001$   $\rightarrow pos=0, TAG=1010 \rightarrow F/R$
- 7)  $0x\ A2 \rightarrow 1010\ 0010$   $\rightarrow pos=0, TAG=1010 \rightarrow V$

$$P_{miss} = \frac{5}{7}$$

00	1010	1
01		0
10	1101	1
11	1101	1

b) Tot, Asociativa (Reemplazamiento)

b)	lot, 1150ml/min	s
	N	W

	TAB	W	
1)	1101 1100		→ F
2)	1101 10 <sup>1</sup> 00		→ F
3)	1101 0010		→ A
4)	1101 10 <sup>1</sup> 01		→ F
5)	0101 00 <sup>1</sup> 0		→ F/R
6)	1010 0010 <sup>1</sup>		→ F/R
7)	1010 0010 <sup>1</sup> 0		→ A

Timer (LRU)	
	101000
6	110111
4	110110
3	110100
5	010100

$$P_{miss} = \frac{5}{7}$$

6	101000	1
4	110110	1
3	110100	1
5	1010100	1

c) Asociativa por conjuntos  $\rightarrow$  2 conjuntos, 2 bloques por conjunto  $\rightarrow$  2-ways

- 1) C=1, TAG = 11011  $\rightarrow$  F
- 2) C=0, TAG = 11011  $\rightarrow$  F
- 3) C=0, TAG = 11010  $\rightarrow$  F
- 4) C=0, TAG = 11011  $\rightarrow$  A
- 5) C=0, TAG = 01010  $\rightarrow$  F/R
- 6) C=0, TAG = 10100  $\rightarrow$  F/R
- 7) C=0, TAG = 10100  $\rightarrow$  A

TAG | index C | W

$$2\cos j = 2$$

5 (LRU) 1  
Timon 10100  
 7, GYR 1  
Co 58 1  
Holt  
Holo  
 10100

$$P_{miss} = \frac{5}{7}$$

C <sub>0</sub>	7	<table border="1"> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> </table>	1	0	1	0	0	1	0	1
1	0	1	0							
0	1	0	1							
C <sub>1</sub>	1	<table border="1"> <tr><td>1</td><td>0</td><td>1</td><td>1</td></tr> <tr><td></td><td></td><td></td><td>0</td></tr> </table>	1	0	1	1				0
1	0	1	1							
			0							

Ej: 5a, 6, 10

⑤

a) 2 bloques por conjunto

$$16 \text{ Mbytes de mem} = 2^2 \cdot 2^{10} \text{ bytes} = 2^6 \text{ bytes}$$

$$\text{Nº bloques} = \frac{2^6 \cdot 2^3}{2} = 2^8$$

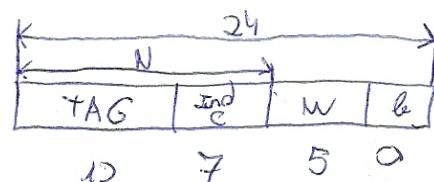
$$2^x = \frac{2^{13}}{2^8} = 2^5 = 2^w$$

U. Dér = byte

$$M.C = 8 \text{ bytes} = 2^3 \cdot 2^3 \text{ bytes}$$

$$1 \text{ pal} = 1 \text{ byte} = 2^0 = 2^6$$

$$\text{Nº conjuntos} = \frac{8}{2} = 2^7 = 2^c \text{ y } 2^8$$



Instrucciones tipo: 000A4yah, Hexadecimal 6 dígitos = 24 bits

$$\text{Nº conjuntos} = 128 = 2^7$$

$$\text{Tamaño de la etiqueta} = 12 \text{ bits}$$

$$\text{Tamaño de bloque} = 5 \text{ bytes}$$

Reemplazo: En la 3º referencia reemplazo en conjunto 82  
vía 0 y en la 4º referencia reemplazo conjunto 82  
vía 1. (Usando las políticas FIFO y LRU es igual)

⑥

$$1 \text{ pal} = 32 \text{ bits} = 4 \text{ bytes} = 2^2 = 2^6$$

$$\text{Nº bloques} = \frac{2^4}{2} = 2^3$$

$$M.C = 32 \text{ bytes} = 2^4 = 2^8$$

$$\text{líneas MC} = 8 \text{ bytes} = 2^3 \cdot 2 \text{ pal} = 2^1 = 2^w$$

a) o lw \$1, 10(\$2)  $\Rightarrow F \rightarrow F/R$   
 4 add \$1, \$1, \$3  $\Rightarrow F/R \rightarrow F/R$   
 8 sub \$4, \$5, \$1  $\Rightarrow F \rightarrow A$   
 $12 \downarrow 32 \quad \rightarrow F \rightarrow F/R$

a) Directa

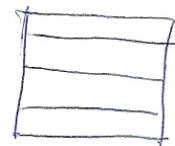


$$P_{MISS} = \frac{10}{12}$$

24 mul \$2, \$2, \$8  
 28 sub \$2, \$2, \$9  
 32 div \$8, \$1, \$4  $\Rightarrow F \rightarrow A$   
 $36 \downarrow 0 \quad \rightarrow F/R \rightarrow F/R$

b) Tot. Asoc.

0  $\rightarrow F \rightarrow F/R$   
 4  $\rightarrow F \rightarrow F/R$   
 8  $\rightarrow F \rightarrow F/R$   
 12  $\rightarrow F \rightarrow F/R$



$$P_{MISS} = \frac{42}{12}$$

24  
 28  
 $32 \rightarrow F/R \rightarrow F/R$   
 $36 \rightarrow F/R \rightarrow F/R$

⑤ L10

$$MP = 16 \text{ Mbytes} = 2^{4+20}$$

U. Dir byte

$$MC = 8 \text{ Kbytes}$$

$$1 \text{ dato} = 1 \text{ byte} = 1 \text{ palabra} = 2^0 = 2^6$$

Dir

$$0x000A40 \rightarrow C = 82, TAG = 0x000$$

$$0x004A40 \rightarrow C = 82, TAG = 0x004$$

$$0x008A40 \rightarrow C = 82, TAG = 0x008 \Rightarrow F/R \rightarrow FIFO/LRU$$

$$0x00BA40 \rightarrow C = 82, TAG = 0x00B \Rightarrow F/R \rightarrow FIFO/LRU$$

Nº conjuntos =  $128 = 2^7$

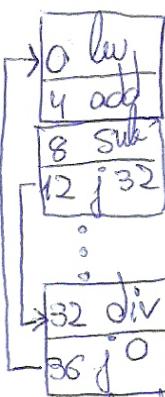
Nº bloques =  $256$

$$\frac{8 \text{ Kbytes}}{256} = \frac{2^{3+10}}{2^8} = 2^5 \text{ bytes/bloque} = 32 \text{ bytes} = 2^5 \text{ palabras/bloque}$$

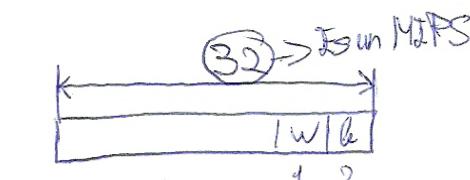
⑥ LJ = 32 bytes

$$1) 1 \text{ bloque}/1 \text{ linea} = 8 \text{ bytes} = 2 \text{ palabras} = 2^1 \text{ bytes}$$

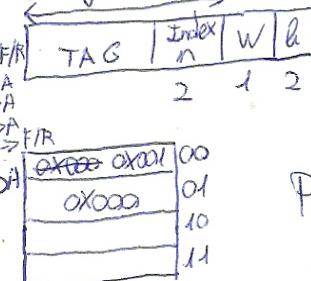
$$MIPS \rightarrow 1 \text{ palabra} = 4 \text{ bytes} = 2^2 \text{ bytes}$$



Dir	TAG N
0x00...00	0000 0000 $\rightarrow p=0 \Rightarrow F \Rightarrow A$
0x00...04	0000 0100 $\rightarrow p=0 \Rightarrow A \Rightarrow A$
0x00...08	0000 1000 $\rightarrow p=1 \Rightarrow F \Rightarrow A$
0x00...0C	0000 1100 $\rightarrow p=1 \Rightarrow A \Rightarrow A$
0x00...20	0010 0000 $\rightarrow p=0 \Rightarrow F \Rightarrow A$
0x00...24	0010 0100 $\rightarrow p=0 \Rightarrow A \Rightarrow A$



a) Org. Directa

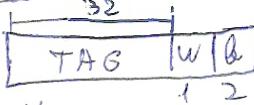


$$\frac{32 \text{ bytes}}{8 \text{ bytes/bloque}} = 4 \text{ bloques} = 2^2 = 2^1$$

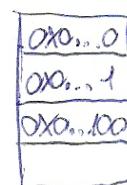
$$P_{miss} = \frac{3 + 2(N-1)}{6N}$$

$N = n^{\circ}$  iteraciones bucle.

b) Totalmente asociativa

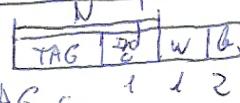


N	
0000 0000	$N = TAG = 000000 \Rightarrow F \Rightarrow A$
0000 0100	$N = TAG = 000000 \Rightarrow A \Rightarrow A$
0000 1000	$N = TAG = 000001 \Rightarrow F \Rightarrow A$
0000 1100	$N = TAG = 000001 \Rightarrow A \Rightarrow A$
0010 0000	$N = TAG = 000010 \Rightarrow F \Rightarrow A$
0010 0100	$N = TAG = 000010 \Rightarrow A \Rightarrow A$



$$P_{miss} = \frac{3 + 0(N-1)}{6N}$$

c) Asociación por conjuntos



C	TAG	2^2	W   b
0	0x00..0	0	
0	0x00..1	1	
0	0x00..100	100	
1	0x00..0	0	
1	0x00..1	1	
1	0x00..100	100	
2	0x00..0	0	
2	0x00..1	1	
2	0x00..100	100	



$$P_{miss} = \frac{3 + 0(N-1)}{6N}$$



Me quedo con c) ya que da el mismo Pmiss que b) pero es más barata y sencilla

2)

- Directa  $\Rightarrow P_{miss} = \frac{4 + \alpha(N-1)}{\sigma + 4N} \rightarrow$  te quedas con este.
- Tot. Asoc.  $\Rightarrow P_{miss} = \frac{4 + \alpha(Nn)}{\sigma + 4N}$
- Asoc. conj.  $\Rightarrow P_{miss} = \frac{4 + \alpha(N-1)}{\sigma + 4N}$   
(2 ways)

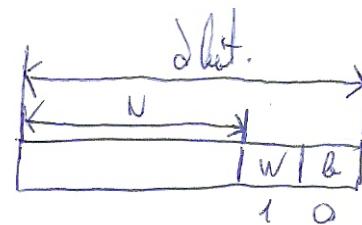
⑩ Do  $i=0, 9$

$$A(i) = 0$$

END DO

$$\begin{aligned} L &= 8 \text{ bytes} \\ 1 \text{ bloque} &= 2 \text{ bytes} \\ 1 \text{ elemento} &= 1 \text{ dato} = 1 \text{ byte} \\ 1 \text{ palabra} &= 1 \text{ byte} \end{aligned}$$

A(29)
:
:
A(1)
A(0)



Dir

N

A(0)	0x0..10	0x0..0
A(1)	0x0..1	0x0..0
A(2)	0x0..10	0x0..1
:	:	10
:	:	00
A(9)	0x0..1001	11

04/12/18

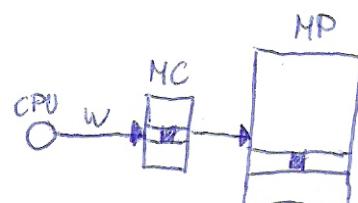
Políticas aciertos escritura

① write-through (Escritura directa)

② write-back (Post-escritura)

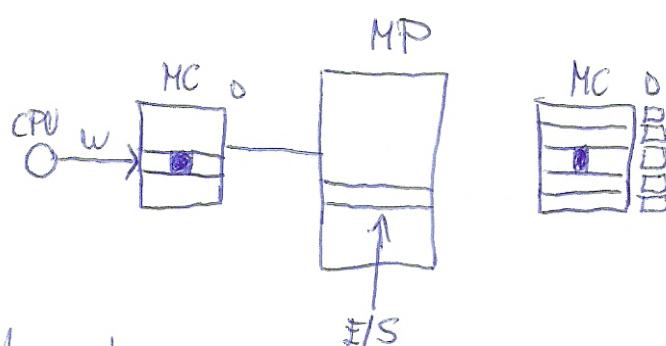
① write-through

- Act. pol. en M. C.
- Act. pol. en M. P.



② write-back

- Act. pol. en M. C
- \* Añadir un bit (dirty) al directorio; indica si un bloque ha sido modificado

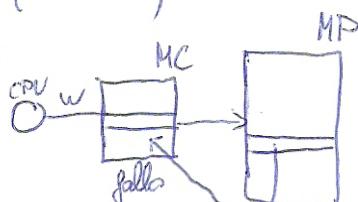


\*  $\hookrightarrow$  Algoritmo de reemplazo, antes de machacar un bloque con D=1, debe de copiar ese en MP.

# Políticas de fallo en escritura

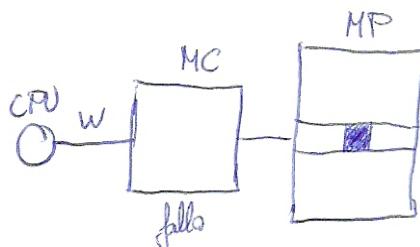
## ① Write allocate

- \* Me traejo a MC el bloque que contiene la palabra que me interesa (= lectura)
- \* A continuación un acierto en escritura (write-through / write-back)

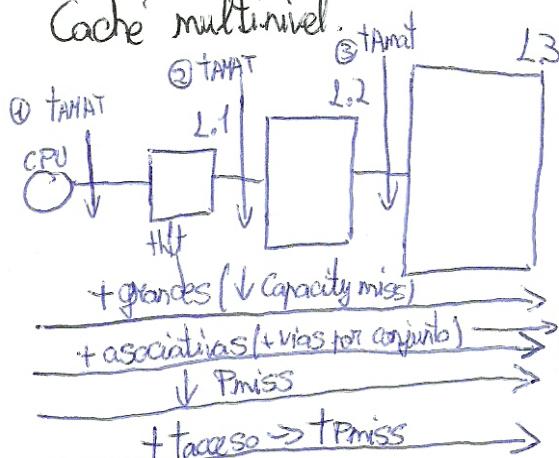


## ② No write allocate

- \* Solo act. pal en MP



## Caché multilvel



Una caché más pequeña permite disminuir los tiempos de búsqueda de un dato, pero al ser más pequeña tiene poca capacidad por lo que se implementan distintos niveles de M.C.

$$NR = N_{hitL1} + N_{missL1} = N_{hitL1} + (N_{hitL2} + N_{missL2})$$

$$R_{missL1} = \frac{N_{missL1}}{NR}$$

$$R_{missL2} = \frac{N_{missL2}}{N_{missL1}}$$

$$R_{miss} = \frac{N_{missL2}}{NR} = \frac{N_{missL1}}{NR} \times \frac{N_{missL2}}{N_{missL1}} = R_{missL1} \times R_{missL2} = \text{global miss rate.}$$

$$\textcircled{1} \quad TAHAT = t_{hitL1} + P_{missL1} \cdot t_{PmissL1}$$

$$\textcircled{2} \quad t_{PmissL1} = t_{hitL2} + P_{missL2} \cdot t_{PmissL2}$$

$$\textcircled{3} \quad t_{PmissL2} = t_{hitL3} + P_{missL3} \cdot t_{PmissL3}.$$

Ej:

$$\begin{aligned} NR &= 2500 \\ N_{missL1} &= 50 \\ N_{missL2} &= 5 \end{aligned}$$

$P_{missL1} = \frac{50}{2500}$

$P_{missL2} = \frac{5}{50}$

$$t_{missL2} = 100 \text{ ck}$$

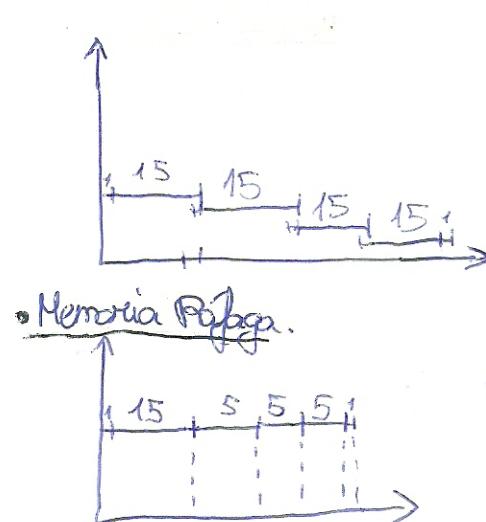
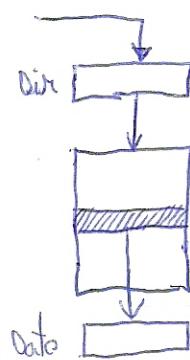
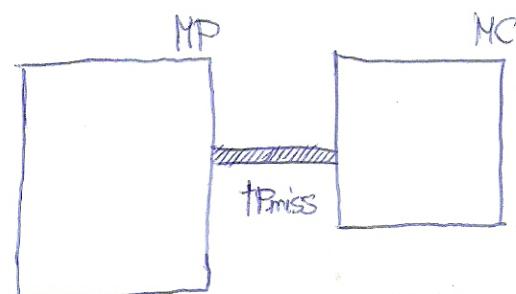
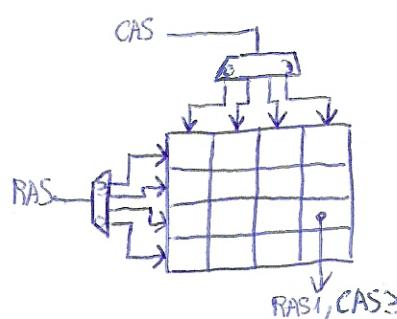
$$t_{accessL2} = t_{hitL2} = 12 \text{ ck}$$

$$t_{accessL1} = t_{hitL1} = 1 \text{ ck}$$

$$\begin{aligned} t_{missL1} &= t_{hitL1} + P_{missL1} \cdot t_{missL1} = 144 \\ t_{missL2} &= t_{hitL2} + P_{missL2} \cdot t_{missL2} = 22 \\ t_{missL1} &= 12 + \frac{50}{2500} \cdot 100 = 144 \\ t_{missL2} &= 12 + \frac{5}{50} \cdot 100 = 22 \end{aligned}$$

$$\begin{aligned} \text{sin } L_2 \\ t_{mat} &= t_{hitL1} + P_{missL1} \cdot t_{missL2} = 3 \\ t_{hitL1} &= 1 \\ P_{missL1} &= \frac{50}{2500} \\ t_{missL2} &= 100 \end{aligned}$$

## Memoria Principal



$$T_{miss} = 1 + 4 \cdot 15 + 1 = 62 \text{ ck}$$

$$BW = \frac{(4 \cdot 4) \text{ bytes}}{62 \text{ ck}} = 0.258 \text{ byte/ck}$$

→ Transferencia de Datos

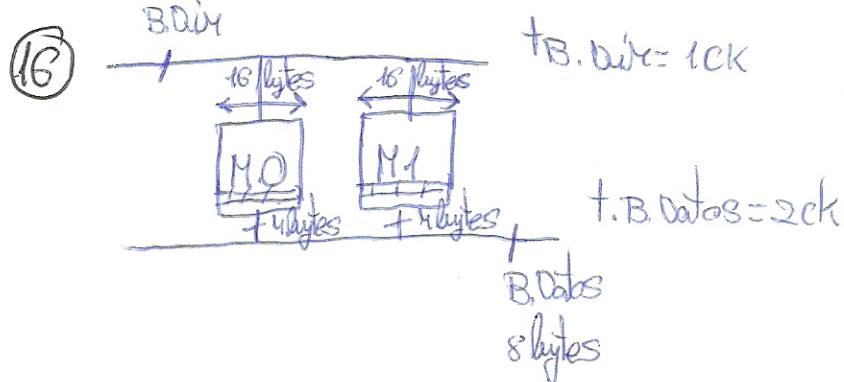
$$T_{miss} = 1 + 15 + 3 \cdot 5 + 1 = 32 \text{ ck}$$

$$BW = \frac{4 \cdot 4}{32} = 0.5 \text{ byte/ck}$$

## • Memoria Interlazada

$$\begin{array}{c} 1, 1, 1, 1, 1, 1, 1, 1 \\ | \quad | \end{array} \quad T_{miss} = 1 + 15 + 1 \cdot 4 = 20 \text{ ck} \quad BW = \frac{4 \cdot 4}{20} = 0.8 \text{ bytes/ck}$$





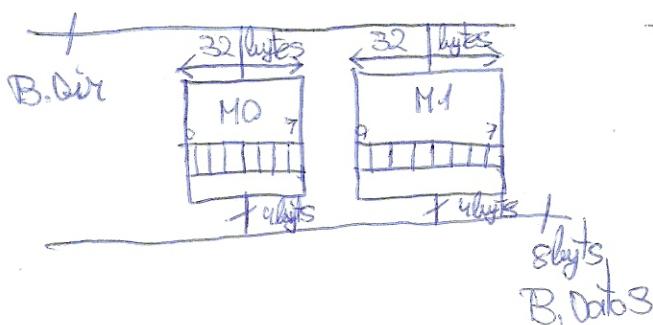
Tacc Mem (1º acceso) = 20 ck Lc  
Tacc Mem (2º y demás accesos) = 8 ck  
Tecnología Noda-Burst

- a) Tamaño B. Datos = 8 bytes, t bloq = 32 bytes 1 pal = 4 bytes  
 $t_{penalización\ falso} = 1 + 20 + 3 * 8 + 2 = 47 \text{ ck}$  BW = 32 bytes  
47 ck

Modulos

$S$	$M_1$	$M_2$	$M_3$
20			
8			
6			

- b) Tamaño B. Dato = 8 bytes Tamaño Bloque = 64 bytes



Tperiodización falla:  $1 + 20 + 7 * 8 + 2 = 79 \text{ ck}$

$BW = \frac{64 \text{ bytes}}{79 \text{ ck}}$

- 17) TAHAT?  $f_{\text{tend}} = 20 \text{ ck}$   $\text{TAHAT} = \text{thit} + \text{tpmiss}$ ,  $\text{Pmiss} = 2 \text{ ck}$   
 $f = 500 \text{ MHz}$   $\text{Phit} = 95\%$

$$2 \cdot \frac{1}{f} = 4 \cdot 10^{-9} \text{ seg} = 4 \text{ ns}$$

- $$\textcircled{18} \quad N_f = 1000 \quad f = 1.2 \text{ GHz} \quad h\nu / SW = 330 \quad \text{CPD}_{\text{ideal}} = 1/8 \quad P_{\text{miss},1} = 0.62 \quad P_{\text{miss},2} = 0.68$$

$$a) \text{Ciclos fallidos? Ciclos-fallados} = NJ * \left( P_{missLI} + P_{missLI} * \frac{330}{1000} + P_{missLO} + P_{missLO} * \frac{330}{1000} \right) = 59.6 \text{ ch}$$

$$b) \text{CPT efectivo} = \text{CPT ideal} + \underbrace{0.596}_{1/8} = 2.396$$

c) Tqm?  $T_{CPU} = NT \cdot CPU_{fakt} \cdot \frac{1}{f} = 11996 \cdot 10^{-6} \text{ seg.}$

$$d) T_{\text{AEST}} = \text{thit} + 0'596 = 1'596 \text{ ck}$$

## Tema 4.

### Tipos de puertos

$I \rightarrow CPU \leftarrow I/O \rightarrow$  Puertos de Estado

$O \rightarrow CPU \rightarrow I/O \rightarrow$  Puertos de Control

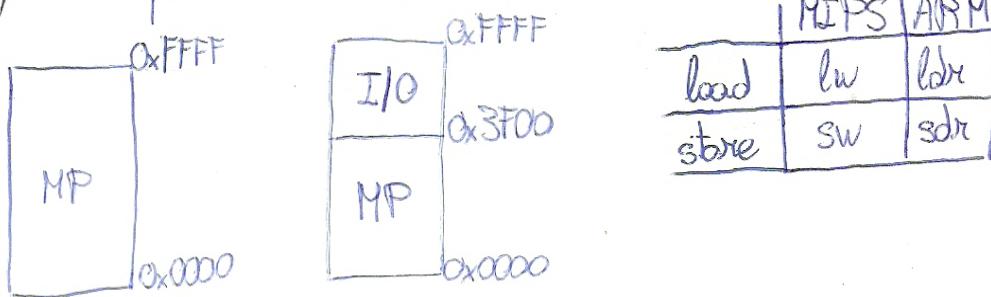
$I/O \rightarrow CPU \leftarrow I/O \rightarrow$  Puertos de Datos

### Modos E/S en un procesador

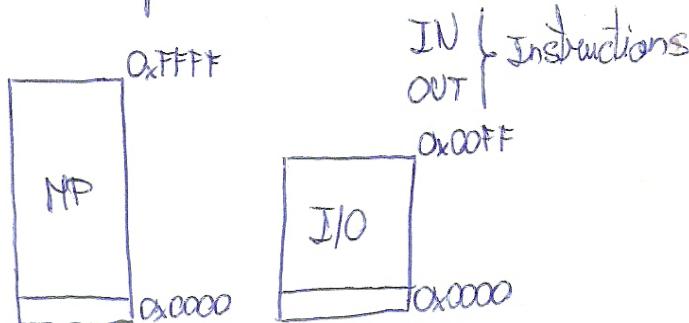
A. E/S mapeada en memoria

B. E/S no mapeada en memoria, o E/S específica

A) E/S mapeada en MEM (MIPS, ARM)



B) E/S no mapeado en MEM (80x86)



### Raspberry

GPIO memory mapping

GPFSEL0\* → GPIO 9: GPIO 00

GPFSEL1\* → GPIO 19: GPIO 10

GPFSEL2\* → GPIO 29: GPIO 20

GPFSEL3\* → GPIO 39: GPIO 30

GPFSEL4\* → GPIO 49: GPIO 40

GPFSEL5\* → GPIO 53: GPIO 50

## Métodos de programación E/S

A) Basado en Polling

B) Basado en Interpretaciones

A) Polling



- x Bucle de espera  $\Rightarrow$  ocupa ciclos de procesador  
gasto/capacitación procesador
  - + Responde inmediatamente al dispositivo

## B) Interrupciones

- HW separa interrupciones
  - Dispositivo cuando quiere ser atendido  $\Rightarrow$  genera INT al procesador



- + No hay ciclos de petición esperando E/S
  - X Tiempo de retraso en atender al dispositivo

### Ejemplo Polling:

No Pulsado

$$\begin{array}{r}
 \text{M1} \leftarrow \overline{\text{xxxxxx}} \\
 \downarrow \quad \downarrow \\
 \text{000100}
 \end{array}
 \quad \text{AND}$$

$\text{eq} \leftarrow \overline{\text{zzzz}} \leftarrow \text{000100}$

(ne)

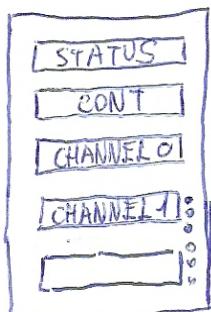
Pulsado

$$\begin{array}{r}
 M_1 \leftarrow \underline{\text{xxxx} \quad \text{xx}} \\
 \text{..} \quad \underline{00100} \\
 \text{AND} \\
 \text{(eq)} \leftarrow Z \text{..} 1 \leftarrow \underline{00000} \\
 \text{ne}
 \end{array}$$

## Timer ARM

$$f_{\text{timer}} = 1 \text{ MHz}$$

1 tick counter timer = 1 μseg

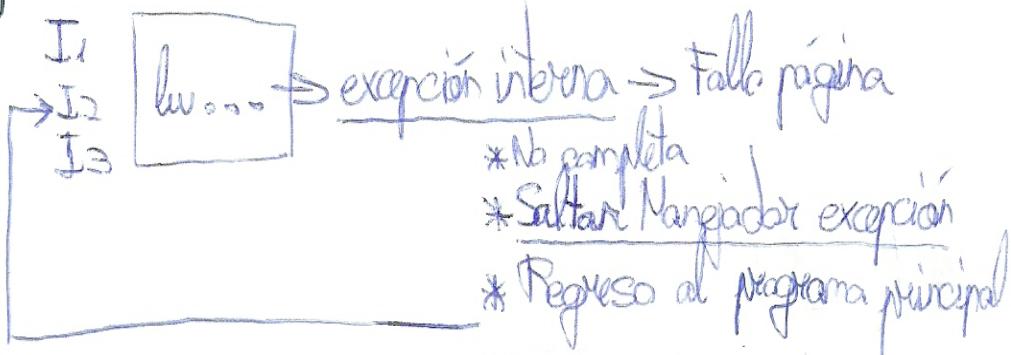


# 1) Excepción Interna

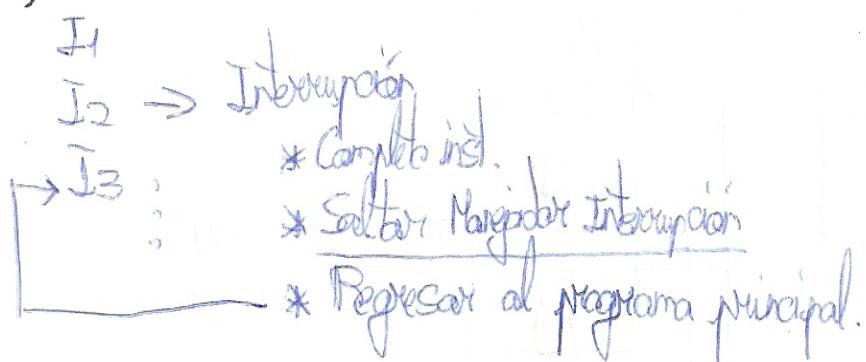
## 2) Interrupciones

- A) Int. no vectoriales
- B) Int. vectorizadas
- C) Int. autovectorizadas

1)

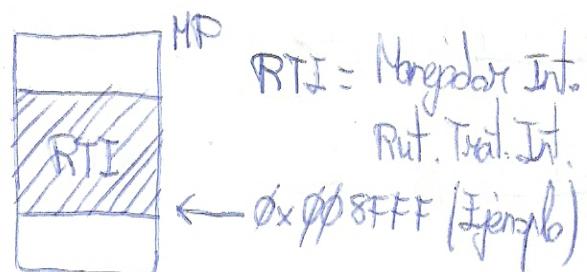
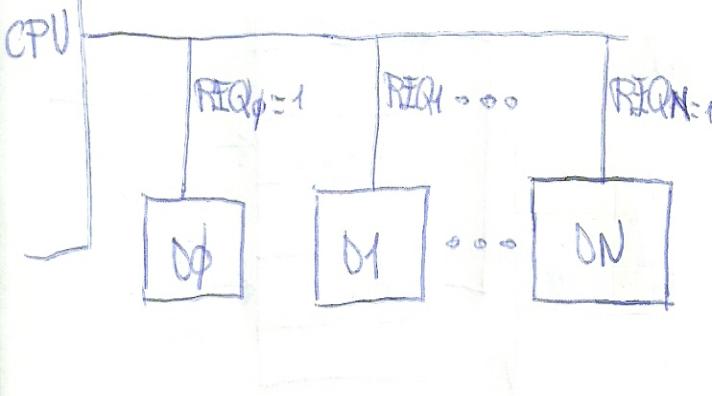
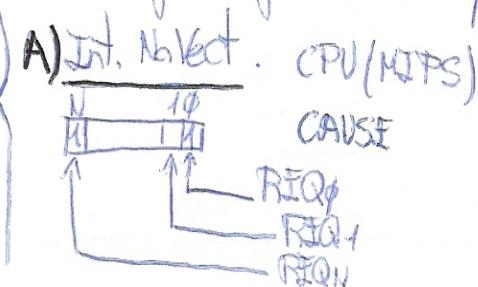


2)



Mecanismo Sistemas con Interrupciones, para identificar fuente de Interrupción

Interrupciones  $\rightarrow$  Mecanismos



INT=1  $\Rightarrow$  PC  $\leftarrow$  0x008FFF  $\Rightarrow$  Salir RTI

RTI

- \* Leer Cause (Saber quién me interrumpe) → Determina la prioridad con la que se tratan las disposiciones I/S → Ref. prioridad SW

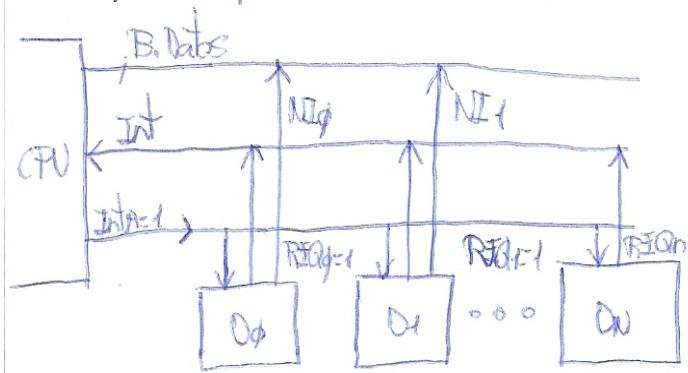
- \* Sultan Etig. proceder dispositivo que indique cause

Do : ;  
'ere

D1 : ; ;  
cret

On : :  
see

### B) Interrupciones vectorizadas (x86)

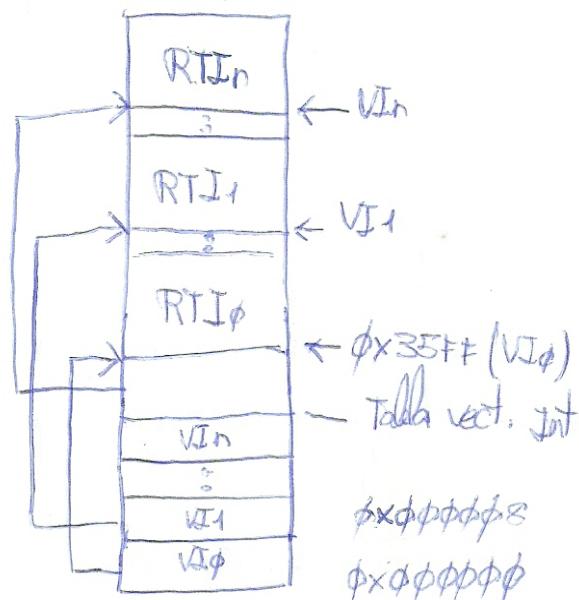


REQ $\phi = 1 \Rightarrow B.\text{Data} \leftarrow N\text{J}\phi$

$$PC \leftarrow MEM[\phi x \phi \phi \phi \phi \phi]$$

REQ<sub>1</sub> = ( $\rightarrow$  B, labs  $\leftarrow$  NI<sub>1</sub>)

$$PC \leftarrow N \in \mathbb{N} [0\phi\phi\phi\phi\phi\phi\phi]$$



c) Interrupciones ausécherixadas

