

Tema 7 Memòria Virtual

Estructura de Computadors (EC) 2023 - 2024 Q2 Adrià Armejach (adria.armejach@upc.edu)



Problemes

Exercici 1

Considera un computador MIPS que gestiona memòria virtual paginada amb les pàgines de 4 KB i amb la restricció que el sistema operatiu admet fins a 6 pàgines com a màxim carregades a memòria física per programa. L'algorisme de reemplaçament de pàgines dins de memòria física és LRU.

La gestió de memòria virtual es realitza amb un TLB de 4 entrades, totalment associatiu i amb reemplaçament LRU.

El computador està executant el següent programa:

```
int V[102400];
main() {
    int i, sum=0;
    for (i=0; i<102400; i++) sum += V[i];
}</pre>
```

Considera que les variables globals s'ubiquen a partir de l'adreça 0x10010000 i que el codi s'ubica a partir de l'adreça 0x00400000.

El contingut del TLB i de memòria física (MF) en un moment donat de l'execució d'aquest programa, en què s'acaba de tractar una fallada de pàgina, és el següent.

	TLB					
entrada TLB	V	D	VPN	PPN		
0	1	0	0x00400	0		
1	1	0	0x10024	5		
2	1	0	0x10022	3		
3	1	0	0x10023	4		

PPN

Exercici 1

a)	(0,30p) Quin és el	VPN (en hexadecimal)	de la pàgina que s	'acaba de carregar a MF?
----	--------------------	----------------------	--------------------	--------------------------

b) (0,40p) Quantes pàgines ocupa el vector V?

c) (0,40p) Quantes fallades de pàgina es produiran en tota l'execució del programa?

d) (0,10p) Quantes fallades de TLB es produiran en tota l'execució del programa?

e) (0,40p) Quins 4 VPN (en hexadecimal) contindrà el TLB quan s'acabi d'executar tot el programa?

Problema 1

- **7.3.** Considerem un computador amb processador MIPS com l'estudiat a classe, que té una memòria cache de dades amb les següents característiques:
 - associativa per conjunts de 4 vies (és a dir, de 4 blocs per conjunt)
 - 64 conjunts
 - 32 bytes per bloc
 - algorisme de reemplaçament LRU

Considerem també que té un TLB de dades amb les següents característiques:

- completament associatiu
- 32 entrades
- mida de pàgina: 8 KB
- algorisme de reemplaçament LRU

Sobre aquest sistema s'executen 2 versions diferents d'una mateixa aplicació:

Suposant que les variables i, j, i sum A s'emmagatzemen en registres, indica quantes fallades hi ha a la cache i al TLB, per a cada versió.

Problema 2 - Part 1

Disposem d'un processador amb sistema de memòria virtual paginada que té les següents característiques:

- Pàgines de 256 bytes
- Mida total de la taula de pagines (TP) = 256 bytes
- Cada entrada de la TP consta dels bits P i D i del PPN.
- Espai de memòria física (MF) = 16 Kbytes
- Reemplaçament de pàgines físiques: LRU
- TLB totalment associatiu de 4 entrades

	0 1	•	••	1 1
a)	()mnle	les	següents	dades:
٠.,	Ompre	100	Section	autico.

Nombre de pàgines físiques =	Nombre de bits d'una entrada de la TP =	
Nombre de pàgines lògiques=	Nombre de bits de l'adreça lògica =	

Problema 2 - Part 2

b) Considerem els següents continguts inicials del TLB i d'una part de la taula de pàgines (TP). Suposarem que la resta d'entrades de la TP estan buides (P=0 i D=0) i que les pàgines físiques que no apareixen a la taula (PPN = 5, 6, 7, etc.) estan lliures. Per a noves pàgines a carregar a MF, s'assignaran números de pàgina física (PPN) correlatius, a partir del número de pàgina física lliure major (és a dir, PPN = 5, 6, 7, etc.). Per al funcionament de l'algorisme LRU del TLB suposarem que les entrades han estat accedides en l'ordre que apareixen a la taula.

	TP				
VPN	P	D	PPN		
0	1	0	4		
1	1	0	3		
2	1	0	2		
3	1	1	1		
4	1	0	0		
5	0	0	-		
6	0	0	-		
	0	0	-		

		TLB	
V	D	VPN	PPN
1	0	1	3
1	0	2	2
1	1	3	1
1	0	4	0

(entrada més antiga)

La següent taula mostra una seqüència de referències a memòria (E: escriptura/ L: lectura). Emplena la taula fent servir la informació del TLB i la taula de pàgines.

20 202 000			TLB		fallada	es llegirà	s'escriurà	PPN
ad	lr. lògica (hex)	VPN	fallada TLB? (SI/NO)	pàg. reemplaçada (VPN)	de pàg? (SI/NO)	del disc? (SI/NO)	al disc? (SI/NO)	resultant de la traducció
L	0x04F2							
E	0x00F1							
L	0x0132				8			
L	0x0541							
Е	0x065A							

Cert o Fals

- 1. Si l'accés a dades d'una instrucció produeix un encert al TLB, però el bit V val 0, llavors la instrucció causarà una excepció de fallada de pàgina
- 2. Una mateixa instrucció pot causar durant la seva execució 2 fallades de pàgina
- 3. Una fallada al TLB no implica que hi hagi una fallada de pàgina.
- 4. En memòria virtual paginada, sempre que reemplacem de la memòria física una pàgina cal escriure-la en disc

Tema 8 Excepcions i Interrupcions

Estructura de Computadors (EC)

Introducció

Excepcions i interrupcions

- Mecanisme del processador que altera el flux de control normal d'un programa
 - Sense la intervenció de cap instrucció de salt
- Excepcions
 - Events interns causats per l'execució d'una instrucció
 - Fallada de pàgina, accés a memòria no alineat, overflow, ...
- Interrupcions
 - Events externs al processador produïts per dispositius d'entrada/sortida
 - Teclat, ratolí, ...

Rutina de Servei d'Excepcions (RSE)

- RSE (exception handler)
 - Codi del sistema operatiu que s'encarrega de gestionar les excepcions i interrupcions
- Al detectar una excepció o interrupció:
 - 1. Es finalitza o cancel·la la instrucció en curs
 - 2. El processador escriu al registre PC l'adreça inicial de la RSE
 - S'executen les instruccions de la RSE
 - 4. S'aborta o es retorna el control al programa

Tipus d'excepcions i interrupcions (1)

- Excepcions causades al intentar executar una instrucció que viola alguna restricció
 - Exemples
 - Overflow aritmètic, accés a memòria no alineat, divisió per zero en coma flotant, instrucció reservada
 - La instrucció s'interromp sense escriure res en registres o en memòria
 - Normalment la RSE avorta el programa
 - En alguns casos es pot tractar:
 - La RSE resol el problema i escriu un resultat en el registre destinació de la instrucció causant de l'excepció
 - La RSE retorna al programa, continuant a partir de la instrucció següent

Tipus d'excepcions i interrupcions (2)

- Excepcions de fallada de pàgina i de fallada de TLB
 - Es produeixen durant la traducció de l'adreça virtual a física d'un accés a memòria
 - En cas de fallada de TLB:
 - La RSE copia la traducció de la taula de pàgines al TLB
 - En cas de fallada de pàgina:
 - La RSE copia la p\u00e0gina del disc a un marc de p\u00e0gina lliure en mem\u00f3ria
 - També s'actualitzen la taula de pàgines i el TLB
 - La RSE retorna al programa re-executant la instrucció que ha causat l'excepció

Tipus d'excepcions i interrupcions (3)

- Excepcions de trap i de crida al sistema
 - Excepció produïda per una instrucció especial de la ISA per accedir a un servei del sistema operatiu
 - Similar a una crida a subrutina
 - Pas de paràmetres i retorn de resultats
 - Instrucció syscall en MIPS
 - Un cop acabada la RSE es retorna al programa, continuant a partir de la instrucció següent

Tipus d'excepcions i interrupcions (4)

- Interrupcions
 - Causades per dispositius d'E/S (entrada/sortida)
 - Les peticions <u>no s'atenen immediatament</u> si hi ha una instrucció en curs d'execució
 - Queden registrades en un registre de peticions pendents
 - Per servir la interrupció, la RSE acostuma a transferir dades de/a algún dispositiu
 - Un cop servida, reprèn el programa interromput, continuant amb la següent instrucció

El Coprocessador del Sistema CP0

El Coprocessador del Sistema CP0

- El tractament d'una excepció el realitza el Sistema Operatiu
 - Però necessita suport de l'arquitectura del processador
- En MIPS aquesta funció la fa el Coprocessador del Sistema (CPO)
 - Controla les excepcions i la traducció ràpida d'adreces virtuals amb TLB
- Banc de registres especific
 - No poden ser usats com a operands de cap de les instruccions estudiades fins ara
- Instruccions per moure dades al CPO

 - o mtc0 rt, c0_rd # copia rt en c0_rd del CP0

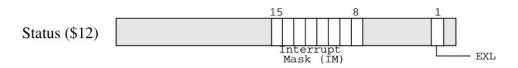
Registres del CP0

- Registre EPC (Exception Program Counter, \$14)
 - Registre de lectura/escriptura
 - Abans d'executar la RSE:
 - En cas d'excepció: es guarda el PC de la instrucció que la produeix
 - En cas d'interrupció: es guarda el PC de la següent instrucció a executar

EPC (\$14) Adreça (PC) de la instrucció interrompuda

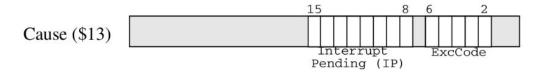
Registres del CP0

- Registre Status (\$12)
 - Bit EXL (Exception Level)
 - Si val 0 el processador està en mode usuari
 - Si val 1 el processador està en mode sistema
 - Es pot accedir a les adreces del SO i executar instruccions privilegiades
 - Totes les interrupcions estan inhibides (les peticions s'ignoren)
 - 8 bits de IM (Interrupt Mask)
 - 1 bit per cada tipus d'interrupció (lectura/escriptura)
 - Si el bit val 1, les interrupcions d'aquell tipus estan habilitades
 - Si EXL = 1 s'ignoren totes les interrupcions, independentment del IM



Registres del CP0

- Registre Cause (\$13)
 - 5 bits de ExcCode només lectura
 - El processador hi escriu un número que codifica la causa de l'excepció
 - 8 bits de Interrupt Pending (IP) només lectura
 - Un per cada tipus d'interrupció
 - A cada senyal de petició d'interrupció d'un dispositiu d'entrada/sortida se li assigna un d'aquests bits
 - Quan el bit corresponent val 1, li indica al processador que el dispositiu d'E/S ha sol·licitat una interrupció



Codis d'excepció

• Exemples d'alguns valors del camp ExcCode del registre Cause

Núm	Nom	Causa
0	Int	interrupció d'un dispositiu hardware d'E/S
1	Mod	fallada de TLB per pàgina modificada (primera escriptura a pàgina)
2	TLBL	fallada de TLB (o fallada de pàgina) per lectura
3	TLBS	fallada de TLB (o fallada de pàgina) per escriptura
4	AdEL	error d'adreça per lectura (accés mal alineat, o a espai de sistema no permès)
5	AdES	error d'adreça per escriptura (accés mal alineat, o a espai de sistema no permès)
6	IBE	error de bus (instruction fetch), p.ex. adreça física inexistent
7	DBE	error de bus (dades de load/store), p.ex. adreça física inexistent
8	Sys	crida al sistema (causada per la instrucció syscall)
9	Bp	breakpoint (causada per la instrucció break)
10	RI	instrucció reservada. p.ex. codis d'operació inexistents
11	CpU	coprocessador no implementat, p.ex. accés a CP0 en mode usuari
12	Ov	overflow aritmètic d'enters (instruccions add, sub, addi)
13	Tr	trap (causada per una instrucció de trap)
15	FPE	excepció de coma flotant: consultar detalls als registres del coprocessador CP1

Primer pas: Accions hardware en cas d'excepció o interrupció

Accions hardware en cas d'excepció o interrupció

- 1. En cas d'excepció:
 - La instrucció en curs s'avorta
 - No s'escriu en registres o memòria, no s'incrementa el PC
- 1. En cas d'interrupció:
 - Es finalitza l'execució de la instrucció en curs
 - Es comprova que el dispositiu i ha fet una petició (IP_i =1) i que està habilitat (EXL=0 i $IM_i=1$)
- 2. Es guarda el PC al registre EPC
- 3. S'escriu la causa de l'excepció al camp *ExcCode* del registre *Cause*
 - En cas de varies excepcions simultànies, s'estableix un sistema de prioritats

Accions hardware en cas d'excepció o interrupció

- 4. Es posa a 1 el bit EXL del registre Status (Mode Sistema)
- 5. El processador escriu en el PC l'adreça inicial predeterminada de la RSE
 - RSE (rutina genérica): PC = 0x80000180
 - TLBmiss (per fallades de TLB): PC = 0x80000000
 - Aquestes adreces pertanyen al espai d'adreces del SO
 - Només es poden executar en mode sistema

Accions hardware en cas d'excepció o interrupció

Possible disseny d'un processador MIPS que inclou suport per

excepcions

