

Tema 8 Excepcions i Interrupcions

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



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
 - 3. 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 pàgina del disc a un marc de pàgina lliure en memòria
 - 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

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

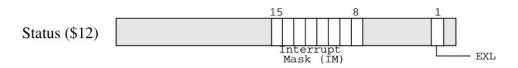
Registres del CP0

- 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

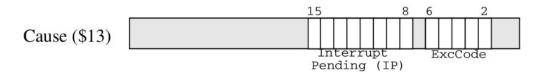
Registres del CP0

- 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	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

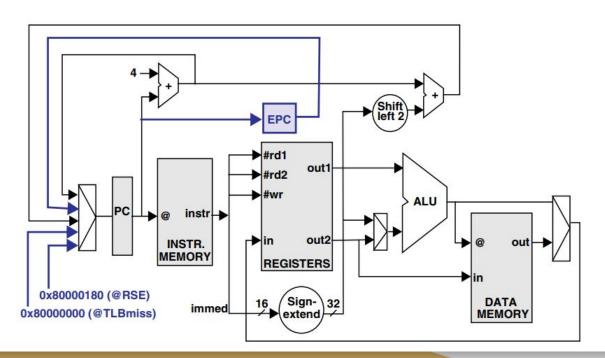
Accions hardware en cas d'excepció o interrupció

- 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



Segon pas: Accions a fer per el software (RSE)

Accions software (RSE)

- 1. La RSE ha de preservar l'estat del programa interromput
 - Es guarden tots els registres per enters a la pila
 - Excepcions: \$k0, \$k1, i els registres de coma flotant
- 2. S'identifica la causa de l'excepció consultant el camp *ExcCode* del registre *Cause*
 - Salta a la subrutina especifica que realitza el tractament
- 3. S'executa la subrutina específica, que pot:
 - Avortar el programa
 - Bloquejar-lo, donant pas a un altre programa
 - Solucionar el problema i retornar després al programa interromput

Accions software (RSE)

- 4. Només en el cas de syscall, ajustar EPC sumant-li 4 (següent instrucció)
- 5. Restaurar tots els registres salvats a la pila
- 6. Retornar al programa interromput usant instrucció eret (exception return)
 - Posa EXL = 0: pasa a mode usuari i activa les interrupcions
 - Copia EPC al PC: salta a l'adreça de retorn guardada en EPC

Exemple senzill de RSE

```
.ktext 0x80000180 # secció "kernel text"
RSE:
    # Salvar TOTS els registres incloent $hi, $lo però no $k0, $k1, $sp
    addiu $sp, $sp, -128
          $1, 0($sp)
    SW
    sw $2, 4($sp)
    sw $3, 8($sp)
    sw $4, 12($sp)
    . . .
          $31, 124($sp)
    SW
    # Passar paràmetres Cause i EPC a una rutina
    mfc0
          $a0, $13 # Passa Cause Register (amb ExcCode)
    mfc0 $a1, $14
                         # Passa EPC
    jal handler_dispatcher # invocarà el handler que correspongui
                             # i decidirà on ha de retornar la RSE
          $v0, $14
                             # copiar resultat a EPC
    mtc0
```

Exemple senzill de RSE

```
# Restaurar registres i pila
lw $1, 0($sp)
lw $2, 4($sp)
lw $3, 8($sp)
lw $4, 12($sp)
lw $31, 124($sp)
addiu $sp, $sp, 128
# Retornar al programa d'usuari, posant EVL=0 i PC=EPC
eret
```

Cert o fals

	Afirmació	V	F
	Si en una cache canviem la política d'escriptura <i>immediata amb assignació</i> a <i>retardada amb assignació</i> , sense cap més canvi, el nombre total de fallades no canvia.		
	En memòria virtual paginada, sempre que reemplacem de la memòria física una pàgina cal escriure-la en disc.		
	En un sistema amb memòria virtual, la mida total d'un programa i les seves dades no pot excedir la capacitat de la memòria física.		
	Si un cert número enter de 8 bits es representa en Ca2 per 0x8B, el mateix número es representa en Ca1 per 0x8C		
i	Si el resultat d'un càlcul en format normalitzat de coma flotant causa underflow, es pot reduir l'error absolut de precisió si el podem expressar en format denormal.		
i	A l'inici de la rutina genèrica de servei d'excepcions de MIPS (RSE) aquesta sols ha de salvar a la pila aquells registres segurs que es modifiquin durant l'execució de la RSE.		
'	Un programa en mode usuari pot copiar un registre qualsevol de la CPU al coprocessador CP0 per mitjà de la instrucció mtc0.		
	El tractament de les excepcions al MIPS es fa únicament en dues rutines de servei, una per a la fallada de TLB i una altra per a la resta d'excepcions.		
)	En el MIPS, el camp IM (Interrupt Mask) del registre Status usat en la gestió de les interrupcions serveix per indicar les peticions d'interrupció que no han de ser ateses quan s'acabi l'execució de la instrucció actual.		
0	La codificació <i>en excés</i> dels exponents dels nombres en coma flotant permet que la comparació de les seves magnituds es pugui fer amb un comparador de nombres naturals.		