CHIAMATE DI SISTEMA

Dott. Franco Liberati


Argomenti
01
Chiamate di sistema nel MIPS
02
Elenco delle chiamate di sistema

Chiamate di sistema nel MIPS

Chiamate di sistema

Generalità

Una **eccezione** MIPS consente al sistema di rispondere ad eventi che si verificano mentre un programma utente è in esecuzione

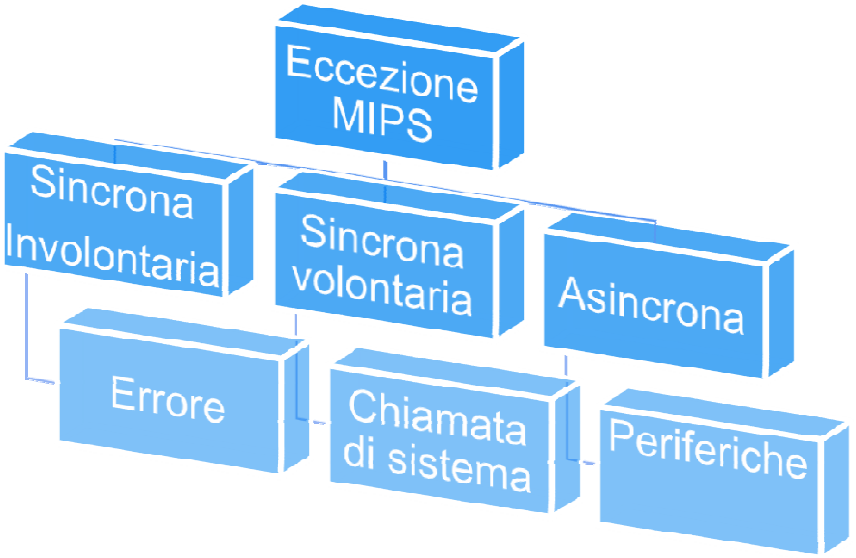
Quando si concretizza una eccezione, si verifica il cambio del flusso di controllo verso una subroutine che non è legata strettamente al codice utente, ma riguarda per lo più funzioni realizzate da terzi e conservate nella parte essenziale (*kernel*) del Sistema Operativo per operare con le parti fisiche dell'elaboratore o ad esso collegate

Una eccezione può verificarsi a seguito di un errore del programma (una situazione indeterminata, cioè non voluta dal programmatore); a causa di una richiesta del programma (voluta esplicitamente dal programmatore) o dovuta ad un evento esterno (indipendente dal programma in esecuzione).

Chiamate di sistema

Generalità

Nel MIPS la classificazione di una eccezione, formalmente, fa riferimento al rapporto con il temporizzatore della macchina:
Sincrona involontaria. È causata quando, dopo un calcolo, si produce un valore non-ammissibile come l'*overflow* aritmetico o la CPU richiede un indirizzo di memoria non accessibile, non allineato, o al di fuori della dimensione del processo. Questa tipologia di eccezione è anche detta **Errore**. Un Errore può essere fisiologico o patologico. Nel primo caso l'istruzione 'colpevole' di aver determinato questa condizione di anormalità può essere riprovata (indirizzo errato per la memoria fisica ma non per la Memoria Virtuale) o simulata (ampliamento dell'area riservata alla pila per evitare uno stack overflow) e il programma utente può continuare; nel secondo caso si ha una interruzione di quanto è in corso di elaborazione (*abort*).

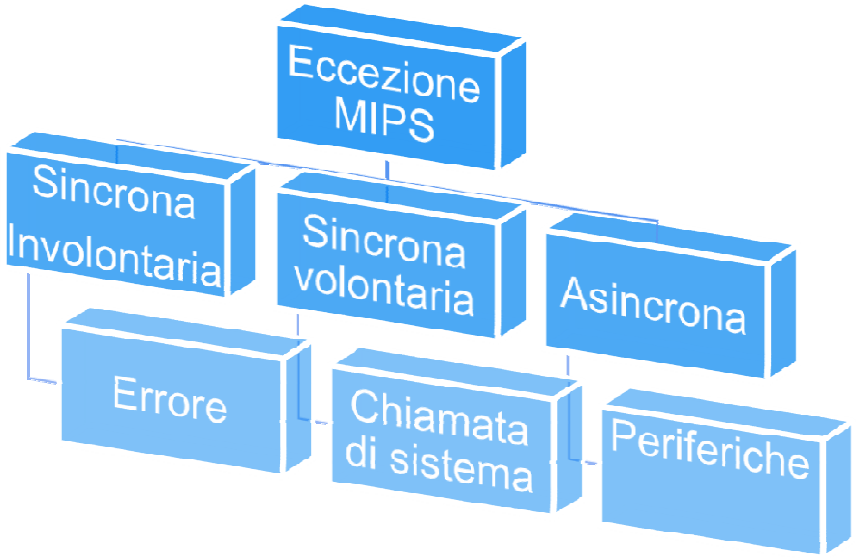


Chiamate di sistema

Generalità

Sincrona volontaria. Si verifica in un tempo stabilito dal programmatore, come l'interruzione (*break*) subito dopo l'elaborazione di una istruzione, che è un'eccezione evocata per motivi diagnostici, o la richiesta di una istruzione di **Chiamata di Sistema** (*syscall*), cioè un'eccezione causata dalla richiesta di una procedura insita nel Sistema Operativo utile, ad esempio, per interagire con le periferiche della macchina (es.: una stampa di un intero sul videoterminale o una lettura di una stringa da tastiera).

Asincrona che è indipendente dall'esecuzione del programma in corso ed è provocata dai Dispositivi di Ingresso e di Uscita. L'eccezione causata da eventi esterni è anche definita, nel contesto MIPS, come **Interrupt**.



Elenco delle Chiamate di sistema

Elenco di chiamate di sistema

Generalità

La comunicazione tra un utente e il sistema di elaborazione MIPS avviene grazie alla **Chiamata di Sistema**, cioè una interruzione sincrona mediante la quale è possibile richiamare una routine, **servizio**, appartenente al Sistema Operativo che consente l'interazione con i dispositivi di Ingresso (es.: tastiera) e di Uscita (es.: videoterminale).

In generale quando si richiede un servizio bisogna specificare il suo identificativo (cosa si richiede) nel registro $v0 e, se necessari, i parametri (quali valori gestire) sfruttando i registri $a0,$a1,$a2,$a3

ESEMPIO

```
li $v0,1          #richiesta del servizio di stampa di un intero su videoterminale
move $a0,$t0      #in $t0 è presente il valore da stampare
syscall           #Attivazione del servizio

li $v0,5          #richiesta del servizio di lettura da tastiera
syscall           #Attivazione del servizio
move $t0,$a0      #recupero del valore immesso da tastiera e spostamento in $t0

li $v0,10         #richiesta del servizio di terminazione del programma
syscall           #Attivazione del servizio
```

Elenco di chiamate di sistema

Principali chiamate

Funzione	Nome	Codice	Argomento	Valore di ritorno
Stampa di un intero	**PRINT_INT**	**1**	$a0=operando intero da stampare	
Stampa di un valore reale in singola precisione	**PRINT_FLOAT**	**2**	$f12=operando reale in singola precisione da stampare	
Stampa di un valore reale in doppia precisione	**PRINT_DOUBLE**	**3**	$f12=operando reale in doppia precisione da stampare	
Stampa di una stringa	**PRINT_STRING**	**4**	$a0=indirizzo iniziale della stringa	
Letturadi un intero	**READ_INT**	**5**		$v0← operando intero immesso da tastiera
Lettura di un valore reale in singola precisione	**READ_FLOAT**	**6**		$f0← operando reale in singola precisione immesso da tastiera
Lettura di un valore reale in doppia precisione	**READ_DOUBLE**	**7**		$f0← operando reale in doppia precisione immesso da tastiera
Lettura di una stringa	**READ_STRING**	**8**		$a0←indirizzo iniziale della stringa $a0←lunghezza della stringa da leggere
Allocazione di memoria	**SBRK**	**9**	$a0=numero di byte da riservare in memoria	$v0 contiene l'indirizzo iniziale della zona di memoria allocata
Terminazione del programma	**EXIT**	**10**		

Elenco di chiamate di sistema

Esempio (stampa su videoterminale intero/stringa e acquisizione di intero da tastiera)

```
                .text
                .globl main
main:
                la $a0,mex1             #inserimento locazione dell'inizio del messaggio
                li $v0,4                #servizio di stampa di una stringa
                syscall                  #chiamata di sistema

                li $v0,5                #servizio di lettura di un intero da tastiera
                syscall                  #chiamata di sistema
                move $t0,$v0            #spostamento del valore letto da tastiera
                mul $t1,$t0,$t0         #operazione (a^2)
                la $a0,mex2             #inserimento locazione dell'inizio del messaggio
                li $v0,4                 #servizio di stampa di una stringa
                syscall                  #chiamata di sistema

                move $a0,$t1            #spostamento del valore intero da stampare
                li $v0,1                #servizio di stampa di un intero
                syscall                 #chiamata di sistema
                li $v0,10
                syscall

                .data
mex1:.asciiz "Inserire il numero: "
mex2:.asciiz "\n Il quadrato del numero e': "
```

Elenco di chiamate di sistema

Principali chiamate MARS

Funzione	Nome	Codice	Argomento	Valore di ritorno
Numero casuale intero in un intervallo	**RANDOM_INT_RANGE**	**42**	$a0=identificatore di numeri aleatori (un intero) $a1=limite superiore dell'intervallo dei numeri casuali.	$a0=numero casuale appartenente all'intervallo [0,…,n] con n numero preimpostato in $a1
Numero causale reale (singola precisione)	**RANDOM_FLOAT**	**43**	$a0=identificatore di numeri aleatori (un intero)	f0=numero casuale reale in singola precisione appartenente all'intervallo [0.0,…,1.0]
Stampa di un intero in rappresentazione esadecimale	**PRINT_INT_TO_EX**	**34**	$a0=operando intero da riprodurre in esadecimale	Mostra il valore intero espresso in esadecimale con una dimensione di 8 cifre
Stampa di un intero in rappresentazione binaria	**PRINT_INT_TO_BIN**	**35**	$a0=operando intero da riprodurre in esadecimale	Mostra il valore intero espresso in binario con una dimensione di 32 cifre
Stampa di un intero senza segno	**PRINT ABS_INT**	**36**	$a0=operando intero	
GESTIONE SINTETIZZATORE MUSICALE		**31-33**		
GESTIONE CASELLE DI DIALOGO		**50-59**		

Elenco di chiamate di sistema

Esempio (generazione numero casuale)

Generare due numeri casuali da 0 a 10 e analizzare se è uscita una coppia

```
        .txt
        .globl main
main:
        li $a0,0
        li $a0,10
        li $v0,42
        syscall
        move $t0,$a0
        li $a0,1
        li $a0,10
        li $v0,42
        syscall
        move $t1,$a0
        beq $t0,$t1, uguali
        la $a0,nocoppia
        li $v0,4
        syscall
        j fine
uguali:
        la $a0, coppia
        li $v0,4
        syscall
fine:
        li $v0,10
        syscall
```

```
.data
coppia: .asciiz “COPPIA”
nocoppia:  .asciiz “NON COPPIA”
```

Elenco di chiamate di sistema

Principali chiamate

Funzione	Nome	Codice	Argomento	Valore di ritorno
Stampa di un carattere	**PRINT_CHAR**	**11**	$a0=carattere (si considerano solo gli 8bit meno significativi di $a0)	
Lettura di un carattere	**READ_CHAR**	**12**		$a0←carattere (il carattere è codificato negli 8bit meno significativi di $a0)
Apertura di un file	**OPEN_FILE**	**13**	$a0=indirizzo della stringa che identifica il nome del file $a1 = flag di apertura $a2 = modo di apertura (*vedere capitolo successivo*)	$v0←Descrittore del file (se c'è un errore il valore è negativo)
Lettura di un file	**READ_FILE**	**14**	$a0 = Descrittore del file $a1 = indirizzo del buffer in cui risiedono i dati $a2 = lunghezza del buffer (in byte)	$v0← Numero di caratteri letti
Scrittura in un file	**WRITE_FILE**	**15**	$a0 = Descrittore del file $a1 = indirizzo del buffer in cui risiedono i dati da scrivere $a2 = numero di dati da scrivere (in byte)	$v0← Numero di caratteri scritti
Chiusura del file	**CLOSE_FILE**	**16**	$a0 = Descrittore del file	
Terminazione del programma	**EXIT2**	**17**		

Elenco di chiamate di sistema

Esempio (creazione file di testo e chiusura)

```
        .txt
        .globl main
main:
        la $a0,nome_file        #apertura file (creazione)
        li $a1,1                #file aperto per la scrittura
        li $a2,0                #
        li $v0,13               #
        syscall                 #
        move $t0,$v0            #descrittore del file (indirizzo del file in memoria)

        move $a0,$t0            #chiusura file
        li $v0,16               #
        syscall                 #

        li $v0,10
        syscall

        .data
nome_file: .asciiz "C:\\Pippo.txt"
```

Elenco di chiamate di sistema

Esempio (scrittura nel file di testo e chiusura)

```
        .txt
        .globl main
main:
        la $a0,nome_file     # apertura file per scrivere
        li $a1,1             #
        li $a2,0             #
        li $v0,13   #
        syscall              #
        move $t0,$v0         # memorizzazione descrittore file
        move $a0,$t0        #scrittura file
        la $a1,messaggio     #locazione dove risiede il dato da scrivere
        li $a2,5            #numeri di caratteri da scrivere
        li $v0,15           #
        syscall             #
        move $a0,$t0         #chiusura file
        li $v0,16           #
        syscall              #
        li $v0,10           #chiusura programma
        syscall              #

        .data
nome_file: .asciiz "C:\\Pippo.txt"
messaggio: .asciiz “Nel mezzo del cammin di nostra vita"
```

Elenco di chiamate di sistema

Esempio (lettura del contenuto di un file di testo e chiusura)

```
.text
.globl main
main:
        la $a0,nome_file
        li $a1,0
        li $a2,0
        li $v0,13
        syscall                 #apertura file per lettura
        move $a0,$v0
        la $a1,messaggio_letto
        li $a2,100
        li $v0,14
        syscall                  #lettura messaggi su file
        la $a0,messaggio_letto
        li $v0,4
        syscall                 #stampa a video quando letto nel file
        li $v0,16
        syscall                  #chiusura file
        li $v0,10
        syscall                  #chiusura programma
        .data
nome_file: .asciiz "C:\\Pippo.txt"
messaggio_letto: .space 255
```

FINE