

L'Interrupt nell'8086

M. Rebaudengo - M. Sonza Reorda

Politecnico di Torino
Dip. di Automatica e Informatica

1

Gestione dell'I/O

La funzione del sistema di Input/Output di un calcolatore è il trasferimento di dati tra la memoria principale ed il mondo esterno.

I metodi utilizzati dalla CPU per gestire le richieste di servizio da parte dei dispositivi di I/O possono essere di 2 tipi diversi:

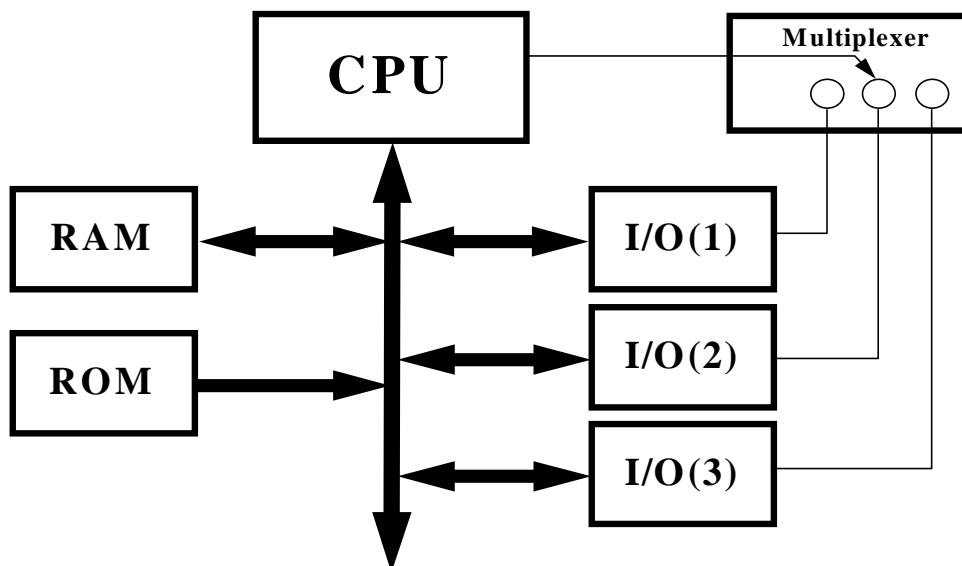
- *polling*;
- *interrupt*.

2

Polling

- La gestione del polling viene fatta attraverso un ciclo software di lettura dello stato dei dispositivi di I/O: la CPU interroga periodicamente i dispositivi
- Se il dispositivo interrogato ha richiesto servizio viene servito
- La maggior parte del tempo è impiegata dal programma principale nell'esecuzione del ciclo di polling.
- Caratteristiche:
 - ☺ Semplice da realizzare
 - ☹ *Elevata latenza*: un dispositivo deve attendere il suo turno prima di essere servito.

3

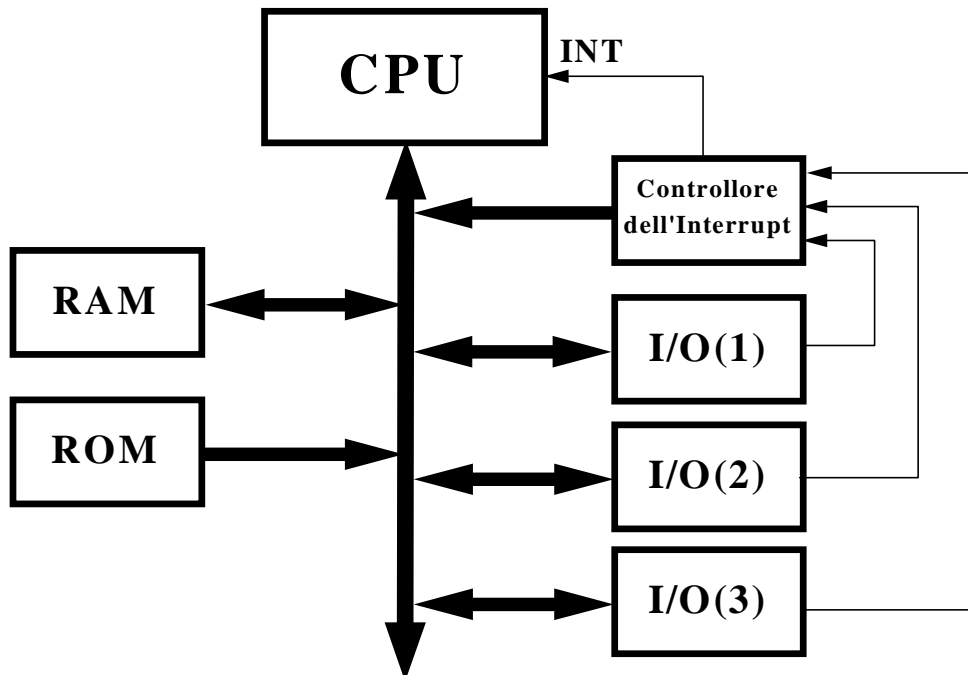


4

Interrupt

- La CPU non interroga i dispositivi di I/O
- quando un dispositivo ha necessità di essere servito attiva la richiesta di interrupt
- al termine dell'esecuzione dell'istruzione corrente, la CPU riconosce la richiesta di interrupt e la serve
- il servizio dell'interrupt avviene iniziando ad eseguire la procedura di servizio dell'interrupt (Interrupt Service Routine o ISR)
- un circuito apposito gestisce le richieste di interrupt (*Interrupt Controller*).

5



6

Tipi di Interrupt

L'8086/8088 supporta 256 tipi di interrupt, suddivisi in 4 gruppi:

- interrupt hardware esterni
- interrupt software
- interrupt interni
- interrupt non mascherabili.

7

Priorità

In ordine di priorità decrescente si hanno:

- interrupt interni
- interrupt non mascherabili
- interrupt software
- interrupt hardware esterni.

All'interno di un gruppo ogni interrupt ha un numero; gli interrupt con numero inferiore hanno priorità maggiore.

Il numero assegnato agli interrupt interni e a quello non mascherabile non può essere modificato.

8

Vector Table

Contiene i puntatori alle routine di servizio di ciascuno dei 256 tipi di interrupt possibili.

Si trova in memoria agli indirizzi più bassi, da 00000H a 003FEH.

Ad ogni tipo di interrupt sono associati 4 byte:

- i primi 2 byte contengono l'offset a cui si trova la procedura all'interno del segmento
- i 2 byte successivi contengono l'indicazione del segmento di codice in cui si trova la procedura.

L'indirizzo assoluto del primo dei 4 byte associati ad un interrupt n è pari a $n*4$.

9

Abilitazione/Disabilitazione degli Interrupt

Il flag IF permette di abilitare/disabilitare gli interrupt hardware esterni.

IF può essere modificato via software attraverso le istruzioni CLI e STI.

IF viene automaticamente resettato all'attivazione di una routine di servizio dell'interrupt.

10

Interrupt software

Sono attivati da programma tramite l'istruzione INT.

11

Interrupt Non Mascherabili

Sono attivati tramite il pin NMI.

Non sono mascherabili usando il flag IF.

Sono *positive edge triggered*. Per essere rilevati, il segnale NMI deve restare attivo per 2 colpi di clock consecutivi.

Ad essi corrisponde il vettore di interrupt all'indirizzo 0008H.

Sono utilizzati di solito in connessione con eventi quali la caduta di alimentazione o un errore di lettura in memoria.

12

Interrupt Interni

- *Divisione per Zero*: quando il quoziente di una DIV o IDIV non può essere rappresentato nel campo destinazione, viene attivata la routine corrispondente al vettore di interrupt in 0000H
- *Overflow*: la istruzione INTO causa l'attivazione della routine di interrupt corrispondente a 0010H se OF vale 1
- *Single Step*: se il flag TF vale 1 al termine di ogni istruzione viene attivata la routine di interrupt corrispondente a 0004H
- *Breakpoint*: se l'istruzione INT non ha parametri, viene codificata su un solo byte e causa l'attivazione della routine di interrupt corrispondente a 000CH.

13

Mappa degli Interrupt

Interrupt	0	→	Divide Error
	1	→	Single Step
	2	→	NonMaskable Interrupt
	3	→	Breakpoint
	4	→	Overflow
	5 ÷ 31	→	Reserved
	32 ÷ 255	→	Liberi

14

Protocollo di Interrupt

- Un dispositivo esterno invia una richiesta di interrupt sul pin INTR (*level triggered*)
- Durante l'ultimo periodo di clock di una istruzione la CPU rileva la presenza di un valore 1 su INTR
- La CPU invia un impulso su INTA per segnalare che l'interrupt è stato rilevato
- La CPU invia un secondo impulso su INTA per chiedere al dispositivo di scrivere sul data bus il numero corrispondente al tipo dell'interrupt
- La CPU legge dal data bus il tipo n dell'interrupt (1 byte)

15

Protocollo di Interrupt (II)

- La CPU salva nello stack il valore del registro dei flag (PSW) e l'indirizzo di ritorno (registri CS e IP)
- La CPU azzerava i flag IF e TF per disabilitare gli interrupt hardware esterni ed il *trap mode*.
- La CPU accede all'elemento (4*n)-esimo nella Interrupt Vector Table
- Viene attivata la corrispondente procedura di Interrupt.

16