

# ARCHITETTURA I

(nuovo ordinamento)

Sono alcune domande prese dai temi d'esame dello Scarabottolo – le domande simili sono state raggruppate

## 1) PROBLEMA DEL CALCOLO DEL RIPORTO NEI CIRCUITI DI SOMMA E LE POSSIBILI SOLUZIONI

Il circuito sommatore di 2 numeri a n bit che tiene conto di un eventuale riporto è il circuito FULL ADDER, formato da 3 ingressi, di cui 2 sono dedicati agli operandi e 1 al riporto precedente, e da 2 uscite di cui 1 dedicata alla somma e una al riporto. I full adder sono collegati a catena in modo tale che il riporto in uscita di un bit sia il riporto in entrata per il full adder successivo. I circuiti sommatore hanno però il problema del ritardo di propagazione del riporto (carry). Per questo problema è stata realizzata la tecnica di CARRY LOOK AHEAD in cui si riesce ad anticipare il calcolo del riporto e a velocizzare la somma.

## 2) IL FUNZIONAMENTO DI BASE DELLA MACCHINA DI VON NEUMAN

La macchina di Von Neuman è composta da 4 elementi fondamentali:

- cpu
- memoria di lavoro
- interfaccia di I/O
- bus di sistema

La cpu, dispositivo master, gestisce in modo sequenziale il sistema, coordina le attività degli altri dispositivi e preleva istruzioni di un programma dalla memoria le decodifica e le esegue.

La memoria contiene istruzioni e dati necessari per l'esecuzione del programma.

L'interfaccia di I/O consente alla cpu di dialogare con la periferica (mondo esterno), trasferendo informazioni.

Il bus di sistema è formato da fili/linee che permettono il collegamento tra i vari componenti.

## 3) MODI DI INDIRIZZAMENTO A MEMORIA (LC-2) :

- **COSA SONO**
- **QUALI SONO**
- **CRITERI DI SCELTA (??)**

Sono le diverse modalità di recupero dei dati necessari per l'esecuzione delle istruzioni.

I modi di indirizzamento sono:

- **immediato**: l'istruzione macchina contiene l'indirizzo della cella di memoria contenente il dato;
- **indiretto**: l'istruzione macchina fornisce l'indirizzo di una cella di memoria che contiene l'indirizzo di un'altra cella di memoria contenente il dato;
- **base + offset**: l'istruzione macchina indica un registro GPR cui sommare un offset per ottenere l'indirizzo della cella di memoria;

#### 4) **DISCUTERE LE CARATTERISTICHE DELL'INTERRUPT PER LA GESTIONE DELLE OPERAZIONI DI INPUT/OUTPUT**

L'interrupt è uno dei tre modi di sincronizzazione tra cpu e interfacce di I/O, e cioè una richiesta di interruzione, da parte dell'interfaccia alla cpu, dello svolgimento dell'attività della cpu. Consente alla periferica di segnalare lo stato di servizio alla cpu mediante la linea dedicata INTREQ del bus di controllo e la cpu risponde, quale riconoscimento di interruzione, con la linea di INTACK. Ha una visione copernicana della cpu in quanto i ritmi di lavoro sono dettati dagli eventi esterni. E' particolarmente adatto a gestire fenomeni urgenti che non possono attendere il tempo di interrogazione di una soluzione a controllo di programma. Un aspetto negativo è che risolve il problema dei fenomeni che si ripetono ad alta frequenza.

5)

- **STRUTTURA E FUNZIONAMENTO DEL PIC**
- **STRUTTURA INTERNA E FUNZIONAMENTO DELLE DIVERSE COMPONENTI PRESENTI IN UN CALCOLATORE CHE PREVEDE GESTIONE VETTORIZZATA DELLE INTERRUZIONI MEDIANTE PIC**
- **GESTIONE DELLE OPERAZIONI DI I/O MEDIANTE PIC**

## Struttura

E' formato da una unità di controllo e 3 registri accessibili dalla cpu

- IVR (INTERRUPT VECTOR REGISTER) : contiene l'identificativo associato a ciascuna periferica collegata
- IPR (INTERRUPT PRIORITY REGISTER) : contiene le informazioni necessarie per stabilire l'ordine di priorità delle periferiche
- IMR (INTERRUPT MASK REGISTER) : contiene le informazioni per sapere quale periferica può chiedere interruzioni e quali no
- 

## Funzionamento

La periferica chiede interrupt all'interfaccia e genera un suo "codice identificativo". L'interfaccia comunica al PIC la richiesta di interrupt e a sua volta il PIC richiede interrupt alla cpu attivando la linea INTREQ. Quando la cpu è pronta a gestire la richiesta di interrupt attiva INTACK e quando il PIC riceve INTACK deposita sul bus dati l'indice del vettore di interrupt relativo all'IVR che gestisce l'interazione con il dispositivo.

## 6) GESTIONE DELL'I/O MEDIANTE INTERRUPT: IL COMPORTAMENTO DELL'UNITA' CENTRALE IN CASO DI INTERRUPT VETTORIZZATO

Alla ricezione di un interrupt la cpu:

- salva il valore del PC nello stack
- disabilita il riconoscimento di ulteriori interrupt
- attiva l'INTACK
- attende sul databus la comparsa di un identificativo a 8 bit che usa come indice in un vettore di interrupt:
  1. tabella di celle di memoria, una associata ad ogni dispositivo sorgente di interrupt
  2. ogni cella contiene l'indirizzo iniziale della routine di risposta all'interrupt associato

## **7) QUANDO E PERCHE' HA SENSO RICORRERE ALLA TECNICA DEL DMA NELLE OPERAZIONI DI INPUT/OUTPUT DI UN CALCOLATORE**

### Quando

Si utilizza quando sono spesso presenti nel calcolatore fenomeni di I/O che si ripetono ad alta frequenza: trasferimento settori da/verso memoria di massa, trasmissione/ricezione di frames da rete, trasferimento periferica – memoria di celle/sequenza di dati.

### Perché

Questa tecnica prevede la possibilità che altri dispositivi, oltre alla cpu, possano accedere a memoria senza l'utilizzo della cpu. Serve per trasferire i dati direttamente da interfaccia a memoria, e quindi il trasferimento è molto più veloce appunto perché non viene utilizzata la cpu, che impiegherebbe molto più tempo perché deve sempre scoprire quello che le si chiede di fare, mediante fetch, decode ed execut, e quindi per svolgere un singolo accesso utile dovrebbe fare molti accessi inutili.

## **8)**

- **STRUTTURA E FUNZIONI DEL DMAC**
- **GESTIONE DELLE OPERAZIONI DI I/O MEDIANTE DMAC**
- **STRUTTURA HARDWARE E SOFTWARE DI UN SISTEMA DI GESTIONE DEGLI ACCESSI A DISCO RIGIDO MEDIANTE DMA**
- **STRUTTURA INTERNA E FUNZIONAMENTO DELLE DIVERSE COMPONENTI PRESENTI IN UN CALCOLATORE CHE PREVEDE ACCESSI A DISCO RIGIDO MEDIANTE DMA**
- **ACCESSO A MEMORIA MEDIANTE DMA: RUOLO E COMPORTAMENTO DEI DIVERSI ELEMENTI COINVOLTI**

### Struttura

E' formato da 4 registri ( a cui la cpu può accedere):

- PA (PERIPHERAL ADDRESS) : contiene l'identificativo dell'interfaccia a periferica con cui interagire per scambiare i dati;
- MDA (MEMORI DATA ADDRESS ) : contiene l'indirizzo della prossima cella di memoria in cui inserire o prelevare il dato;
- DC (DATA COUNTER) : contiene il numero dei dati ancora da trasferire;
- TD (TRANSFER DIRECTION) : indica se l'operazione è lettura (in) o scrittura (out)

### Funzionamento

Gli elementi coinvolti sono 3:

- cpu
- dmac
- interfaccia a disco

il comportamento della cpu a livello software è che inizializza il dmac e l'interfaccia mediante la routine readisk del sistema operativo. Il programma in esecuzione della cpu richiede lettura da memoria di massa a disco chiamando la routine readisk.

Questa routine inizializza il DMAC inserendo:

- nel PA l'identificativo dell'interfaccia
- nel MDA l'indirizzo della prossima cella di memoria
- nel DC il numero totale di dati da trasferire
- nel TD che si tratta di lettura

Inizializza l'interfaccia comunicandole il numero della traccia e del settore da leggere da disco, e che si tratta di lettura.

Il comportamento a livello hardware dei tre elementi è il seguente:

- l'interfaccia segnala dato pronto al dmac (ponendo il 1° bit meno significativo a 1 di status)
- il dmac chiede i bus alla cpu mediante la linea HOLDREQ (del bus di controllo)
- la cpu rilascia i bus mediante la linea HOLDACK (del control bus)

Il DMAC pone sull'address bus il contenuto dell'MDA e attiva in all'interfaccia e memwrite alla memoria. Viene così trasferito il dato direttamente da interfaccia a memoria mediante il databus. Viene disattivata sia la linea di HOLDREQ che HOLDACK; viene incrementato l'MDA e decrementato il DC. Se il  $DC > 0$  continua il trasferimento, se  $DC = 0$  finisce. A questo punto il dmac invia un segnale di interruzione alla cpu; la cpu attiva la routine di risposta all'interrupt, che segnala al readisk che è terminata l'operazione. Infine il sistema operativo riattiva il programma che aveva richiesto lettura da disco.

## **9) STRUTTURA E COMPORTAMENTO DELLA MEMORIA CACHE**

La cache è una memoria nascosta di dimensioni ridotte rispetto alla memoria di lavoro, è molto + vicina alla cpu, recentemente addirittura sullo stesso chip, in modo tale da ridurre notevolmente il tempo di accesso a memoria. Viene copiato nella cache il contenuto della cella di memoria richiesta dalla cpu e mediante indirizzo di memoria tutte le celle vicine (blocco) presenti nella memoria di lavoro. Così quando la cpu avrà bisogno della stessa parola di memoria o di altre, andrà a prelevarle direttamente dalla cache.

## **10) PRINCIPALI CARATTERISTICHE DI UNA CPU ORGANIZZATA A PIPELINE STRUTTURA INTERNA DI UNA CPU PIPELINE**

La cpu pipeline è una cpu che ha una struttura a catena di montaggio, cioè formata da stadi che operano in parallelo in modo tale di consentire di velocizzare l'esecuzione e aumentare il numero di istruzioni macchina completate per unità di tempo. Riesce quindi a completare un'istruzione ad ogni ciclo di clock. Gli stadi di cui è formata sono quattro:

- fetch: prelievo dell'istruzione dalla memoria da parte del processore
- decode: l'istruzione viene esaminata e decodificata (in base all'operando)
- execute: viene effettuata l'esecuzione delle istruzioni
- writeback: il risultato dell'istruzione viene copiato in un registro del processore o memorizzato

tra uno stadio e l'altro vi è un registro intermedio chiamato buffer o registro di buffer, poiché ciascuno stadio passa informazioni allo stadio consecutivo, le informazioni vengono scritte nel buffer dallo stadio precedente per permettere allo stadio successivo di utilizzarle. E' formata da memoria cache.