

Lezione 1 – Modalità di I/O a DMA (*Direct Memory Access*)

Architettura degli elaboratori

Modulo 3 - Architettura del calcolatore

Unità didattica 5 - Input/Output a DMA

Nello Scarabottolo

Università degli Studi di Milano - Ssri - CDL ONLINE

Fenomeni che usano DMA

Come detto, sono spesso presenti nel calcolatore fenomeni di I/O che si ripetono ad alta frequenza:

- trasferimento di settori da/verso memoria di massa;
- trasmissione/ricezione di *frames* da rete.

Questi fenomeni richiedono generalmente il trasferimento da periferica a memoria o viceversa di sequenze di dati (celle).

Solo quando il trasferimento dell'intera sequenza è terminato, si può procedere con l'elaborazione.

Limiti della CPU

Anche qualora sia dedicata a questo tipo di operazioni di I/O, la CPU è penalizzata perché:

- essendo un componente *general purpose*, deve scoprire (*fetch* di istruzioni macchina) passo passo cosa le si chiede di fare;
- per trasferire un dato da periferica a memoria o viceversa, il singolo accesso utile (trasferimento del dato) è penalizzato da un elevato numero di accessi "inutili":
 - fetch delle istruzioni;
 - incremento del puntatore all'area di memoria da/in cui trasferire i dati;
 - aggiornamento del contatore di dati trasferiti...

Direct Memory Access

Questa tecnica prevede la possibilità che altri dispositivi - oltre alla CPU - possano accedere a memoria:

- diventare quindi temporaneamente Master del bus.

Per far questo, serve poter richiedere alla CPU la possibilità di utilizzare il bus:

- serve linea dedicata del bus di controllo: HOLDREQ.

Anche gli stadi di uscita della CPU che pilotano le linee dell'Address Bus e del Control Bus devono essere TRI STATE (per lasciare agli altri Master la possibilità di pilotarle).

Linea HOLDREQ

In un calcolatore possono esistere più periferiche che richiedono DMA.

Non c'è possibilità di sincronizzazione fra le richieste di DMA:

- ogni gestore di DMA chiede i bus quando la propria periferica deve trasferire un dato.

La linea di richiesta HOLDREQ deve essere gestita mediante porte OPEN COLLECTOR (U.D.1, Lez.1):

- linea attiva bassa;
- chi vuole i bus forza a 0 a bassa impedenza la linea;
- normalmente la linea è tenuta a 1 dalla resistenza di *pull-up*.

DMAC (*DMA Controller*)

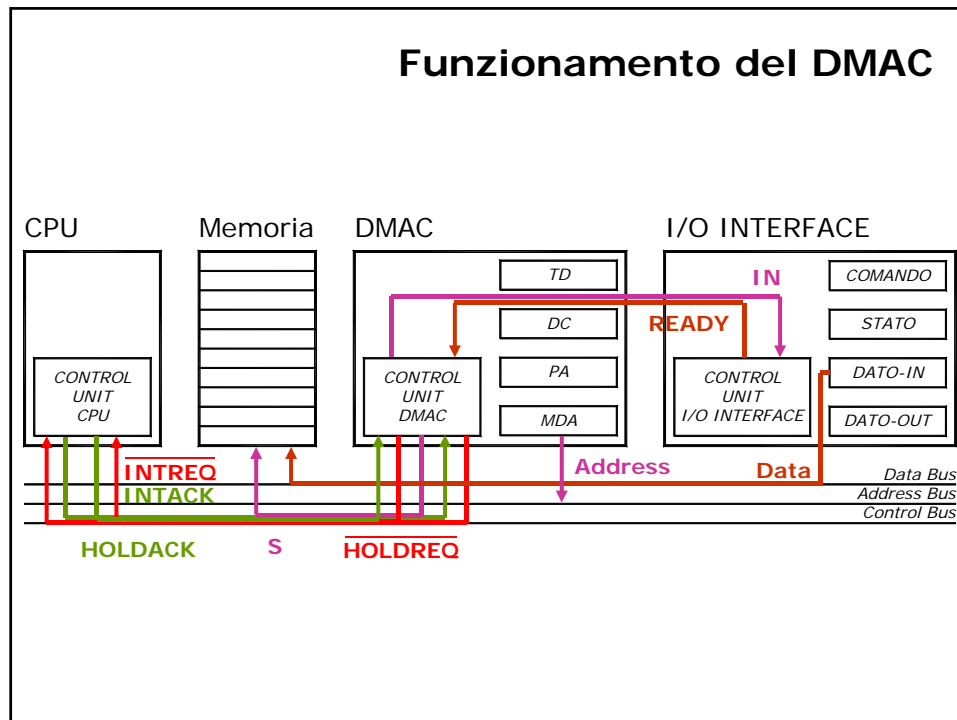
Integrato di supporto alla gestione del DMA.

La CPU programma il DMAC comunicando:

- indirizzo della zona di memoria da/in cui trasferire i dati;
- numero di dati da trasferire;
- identificativo della periferica e senso di trasferimento.

Quando la periferica segnala di essere pronta:

- il DMAC richiede i bus con il segnale HOLDREQ;
- quando la CPU ne ha terminato l'eventuale uso in corso, ne segnala il rilascio con HOLDACK;
- il DMAC effettua il trasferimento e aggiorna puntatori e contatori;
- finito l'intero trasferimento, genera interrupt.



Registri del DMAC

Registri accessibili alla CPU come normali registri di interfaccia (componente programmabile):

- PA** *Peripheral Address*: contiene l'identificativo dell'interfaccia a periferica con cui interagire per scambiare i dati;
- MDA** *Memory Data Address*: contiene l'indirizzo della prossima cella di memoria in cui inserire o da cui prelevare il dato;
- DC** *Data Counter*: contiene il numero di dati ancora da trasferire;
- TD** *Transfer Direction*: indica se l'operazione è una lettura (IN) o una scrittura (OUT).

In sintesi...

La tecnica di I/O mediante DMA ha le seguenti caratteristiche:

- è possibile trasferire dati da/verso la memoria sotto il controllo di un componente diverso dalla CPU;
- tale componente - il DMAC - ha il vantaggio di essere realizzato a questo scopo, e non perde tempo per scoprirlo da programma;
- il trasferimento avviene in modo "trasparente" al programma in esecuzione sulla CPU (che viene semplicemente rallentata perché deve occasionalmente rilasciare i bus);
- al termine dell'intera attività, il programma che aveva richiesto I/O viene avvisato con interrupt.

Chiusura

**Fine della
lezione**

