

# Il Livello Logico Digitale (iii)

A cura di:

Luca Breveglieri      Giuseppe Pozzi

DEI, Politecnico di Milano  
luca.breveglieri/giuseppe.pozzi@polimi.it  
- versione dell'11 aprile 2003 -

13-04.-03

Informatica II - Livello logico (3)

1

# Banco dei registri e memoria

13-04.-03

Informatica II - Livello logico (3)

2

## Banco dei registri

- Spesso occorre utilizzare un certo numero di registri paralleli con funzione di caricamento, tutti aventi le stesse dimensioni e le stesse funzioni
- Invece di piazzare tanti registri paralleli indipendenti, è allora preferibile organizzare i registri in una struttura a vettore, chiamata banco dei registri
- Il banco dei registri permette anche di ridurre i collegamenti necessari

13-04.-03

Informatica II - Livello logico (3)

3

## Funzionamento del banco

- Si consideri un banco di 8 registri da 16 bit ciascuno, ovvero un banco  $8 \times 16$
- Ogni registro è identificato da un indirizzo: un numero compreso tra 0 e 7, estremi inclusi
- Gli 8 registri sono pertanto chiamati: R0, R1, ..., R7
- Ogni registro contiene 16 bit

13-04.-03

Informatica II - Livello logico (3)

4

## Funzionamento del banco

- Per specificare l'indirizzo di un registro occorrono 3 bit, sufficienti a codificare i numeri interi nell'intervallo 0, 1, ..., 7 (con  $n \geq 1$  registri occorrono  $\lceil \log_2 n \rceil$  bit)
- Le operazioni eseguibili sul banco sono:
  - lettura, ovvero si prelevano i 16 bit memorizzati nel registro indirizzato
  - scrittura, ovvero si inviano e memorizzano 16 bit nel registro indirizzato

13-04.-03

Informatica II - Livello logico (3)

5

## Funzionamento del banco

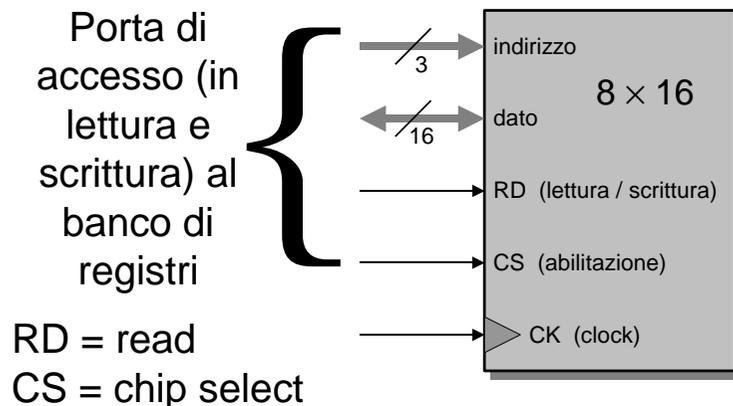
- Il banco dei registri ha:
  - 3 ingressi di indirizzo, su cui si manda l'indirizzo del registro da leggere o scrivere
  - 16 uscite/ingressi dati, su cui si riceve/invia il contenuto del registro da leggere/scrivere
  - un ingresso di comando: lettura o scrittura
  - un ingresso di abilitazione
  - e naturalmente il segnale di clock

13-04.-03

Informatica II - Livello logico (3)

6

## Rappresentazione del banco



13-04.-03

Informatica II - Livello logico (3)

7

## Operazioni del banco

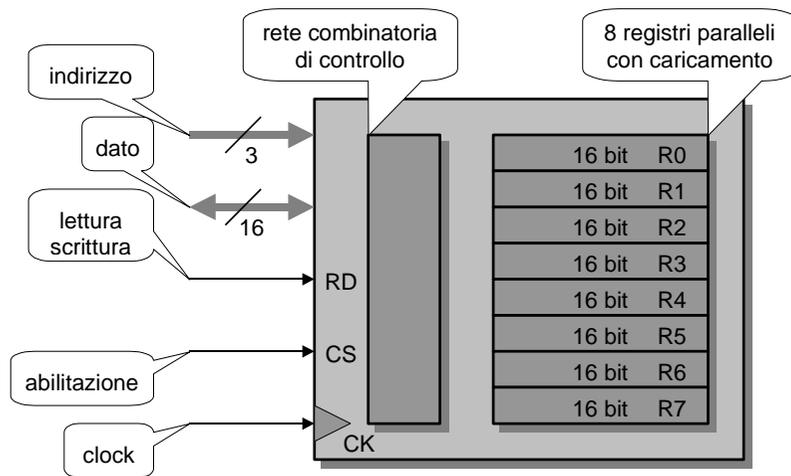
- Se l'abilitazione è attiva ( $CS = 1$ ):
  - Lettura: all'inizio del ciclo si inviano l'indirizzo del registro e il comando di lettura ( $RD = 1$ ); verso la fine del ciclo il contenuto del registro viene emesso
  - Scrittura: all'inizio del ciclo si inviano l'indirizzo del registro, il dato da scrivere e il comando di scrittura ( $RD = 0$ ); verso la fine del ciclo il dato viene memorizzato nel registro
- Se l'abilitazione è inattiva ( $CS = 0$ ), il banco mantiene il suo contenuto

13-04.-03

Informatica II - Livello logico (3)

8

## Struttura del banco



13-04.-03

Informatica II - Livello logico (3)

9

## Varianti e integrazioni

- I banchi di registri sono disponibili in svariate dimensioni:  $8 \times 8$ ,  $16 \times 16$ ,  $32 \times 32$ , e altre ancora
- I banchi più sofisticati possono avere più porte di accesso distinte, in lettura/scrittura, o anche in sola lettura o sola scrittura, per potere operare in parallelo su più registri
- In caso di più porte di scrittura, sono in grado di gestire i conflitti di scrittura

13-04.-03

Informatica II - Livello logico (3)

10

## Usi caratteristici

- Il banco dei registri è un blocco funzionale essenziale dei processori
- È usato per realizzare i registri interni al processore, che contengono gli operandi di tipo intero delle istruzioni aritmetico-logiche del processore
- Esistono anche banchi di registri adattati per operandi di tipo reale (standard IEEE 754)

13-04.-03

Informatica II - Livello logico (3)

11

## Memoria

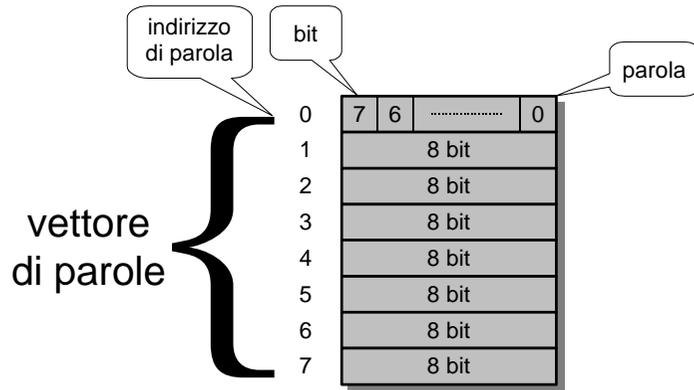
- La memoria è un blocco funzionale di tipo sequenziale complesso
- Serve per mantenere a tempo indefinito dati e programmi, e per permetterne l'accesso, in lettura o in scrittura
- Ha una struttura a vettore, i cui elementi sono le parole di memoria
- Ogni parola di memoria è una sequenza di bit, in numero fissato  $\geq 1$

13-04.-03

Informatica II - Livello logico (3)

12

## Organizzazione a parole



Capacità: 8 parole  $\times$  8 bit per parola = 64 bit

## Caratteristiche della memoria

- Un componente integrato (chip) di memoria si caratterizza specificandone:
  - la capacità, misurata in numero totale di bit memorizzabili: di solito si esprime come prodotto del numero di parole per il numero di bit contenuti nella parola
  - le funzioni: lettura e scrittura, solo lettura
  - il numero di porte di accesso
  - e il tempo necessario per l'accesso

## Interfaccia di memoria

- Il contenuto della memoria viene letto o scritto una parola per volta, in un ciclo di clock (più cicli in memorie lente)
- Si accede a una parola di memoria tramite la porta di accesso alla memoria
- La porta di accesso alla memoria può funzionare in lettura e scrittura (è il caso più frequente), solo in lettura e teoricamente anche solo in scrittura (caso poco frequente)

## Interfaccia di memoria

- La porta di accesso alla memoria è formata dai segnali seguenti:
  - Gli ingressi di indirizzo, che codificano in binario l'indirizzo della parola su cui si deve operare: se la memoria ha capacità di  $k \geq 1$  parole, occorrono  $\lceil \log_2 k \rceil$  ingressi di indirizzo
  - Le uscite / ingressi di dato, che servono per leggere / scrivere una parola: se la parola ha dimensione di  $h \geq 1$  bit, occorrono esattamente  $h$  uscite / ingressi di dato

## Interfaccia di memoria

- La porta di accesso alla memoria è formata dai segnali seguenti:
  - il comando di lettura / scrittura, RD (read): RD = 1 lettura; RD = 0 scrittura
  - il comando di abilitazione del componente, CS (chip select): CS = 1 chip attivo, si può accedere al contenuto; CS = 0 chip in stato di riposo, non si può né leggere né scrivere
  - il comando di abilitazione delle uscite dati, OE (output enable): OE = 1 le uscite / ingressi dati sono funzionanti; OE = 0 sono isolati

13-04.-03

Informatica II - Livello logico (3)

17

## Componente di memoria



- Componente integrato (chip) di memoria
- Capacità:  $2^n$  parole  $\times$  m bit per parola
- Spesso il clock non serve o è interno

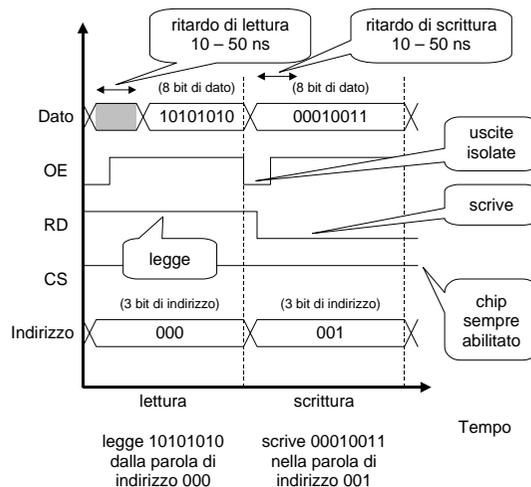
13-04.-03

Informatica II - Livello logico (3)

18

## Diagramma temporale

- Capacità:  $8 \times 8$
- # bit ind:  $\log_2 8 = 3$  bit
- # bit dato: 8 bit



13-04.-03

Informatica II - Livello logico (3)

19

## Ciclo di lettura

- All'inizio del ciclo occorre:
  - abilitare il componente (CS = 1)
  - inviare l'indirizzo della parola da leggere
  - attivare il comando di lettura (RD = 1)
  - non isolare gli ingressi/uscite dati (OE = 1)
- Dopo un certo intervallo di tempo (ritardo di lettura) il contenuto della parola è disponibile sulle uscite dati
- Ritardi tipici: 10 - 50 ns

13-04.-03

Informatica II - Livello logico (3)

20

## Ciclo di scrittura

- All'inizio del ciclo occorre:
  - abilitare il componente ( $CS = 1$ )
  - inviare l'indirizzo della parola da scrivere
  - disattivare il comando di lettura ( $RD = 0$ )
  - non isolare gli ingressi/uscite dati ( $OE = 1$ )
  - inviare sugli ingressi la parola da scrivere
- Dopo un certo intervallo di tempo (ritardo di scrittura, 10 - 50 ns) la parola viene scritta in memoria all'indirizzo indicato

## Struttura della memoria

- Le strutture interne delle memorie sono molto varie; sono disponibili numerose tecnologie di memoria
- Il modo più naturale (sebbene non adatto a memorie di grande capacità) per realizzare una memoria è costruire una matrice di bistabili, completandola con reti combinatorie di controllo per gestire l'accesso alle parole

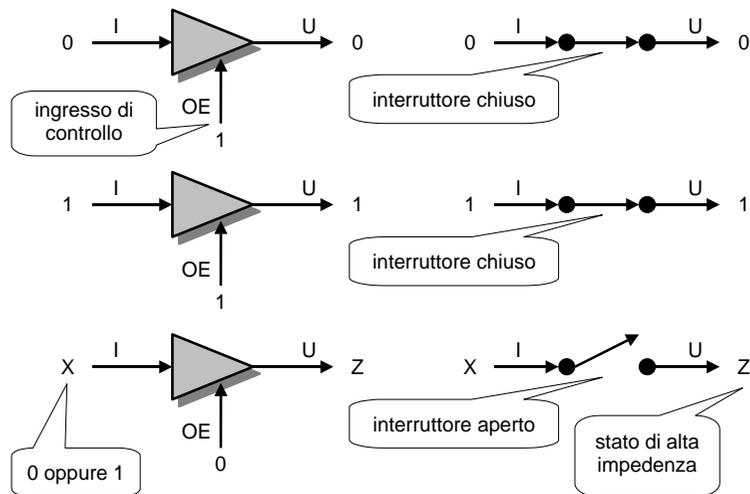
## Organizzazione a matrice

- Le memorie vengono viste come vettori (1-dimensionali) di parole
- Ma le memorie di elevata capacità possono in realtà avere una struttura interna a matrice (2-dimensionale), allo scopo di ridurre i collegamenti interni
- In tal caso sono di solito presenti due comandi aggiuntivi (RAS e CAS), per l'accesso alla memoria

## Buffer (tampone)

- Se due o più componenti di memoria sono collegati alle medesime linee di uscita, è possibile si verifichi un conflitto di scrittura
- Il buffer (tampone) ad alta impedenza è un elemento funzionale elementare utile per isolare elettricamente un'uscita
- Il buffer funziona come un interruttore: quando si trova in stato di alta impedenza (Z), la sua uscita è elettricamente isolata e non dà origine a conflitti di scrittura

## Funzionamento



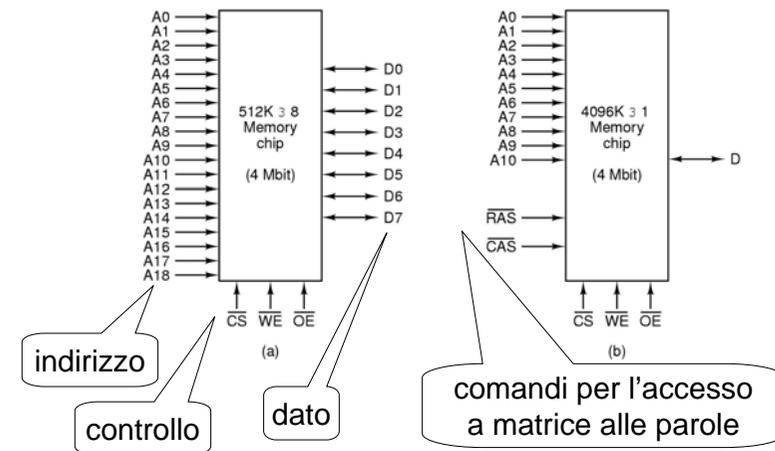
13-04.-03

Informatica II - Livello logico (3)

25

## Componente di memoria

Due componenti di memoria da  $4\text{ M} \times 1$



13-04.-03

Informatica II - Livello logico (3)

26

## Banco di memoria

- Sono disponibili componenti integrati di memoria di svariate capacità
- Per esempio:  $64\text{ K} \times 8$ ,  $1\text{ M} \times 8$ ,  $1\text{ M} \times 1$ ,  $256\text{ M} \times 1$ , ...
- Tuttavia, per ottenere memorie di capacità elevata, occorre aggregare più componenti di memoria, realizzando un banco di memoria

13-04.-03

Informatica II - Livello logico (3)

27

## Banco di memoria

- I banchi di memoria hanno una struttura a matrice di chip
- Per aumentare la lunghezza della parola di memoria, si compone una riga di chip di memoria, da usare in parallelo
- Per aumentare il numero di parole della memoria, si compone una colonna di chip di memoria, da usare in esclusione

13-04.-03

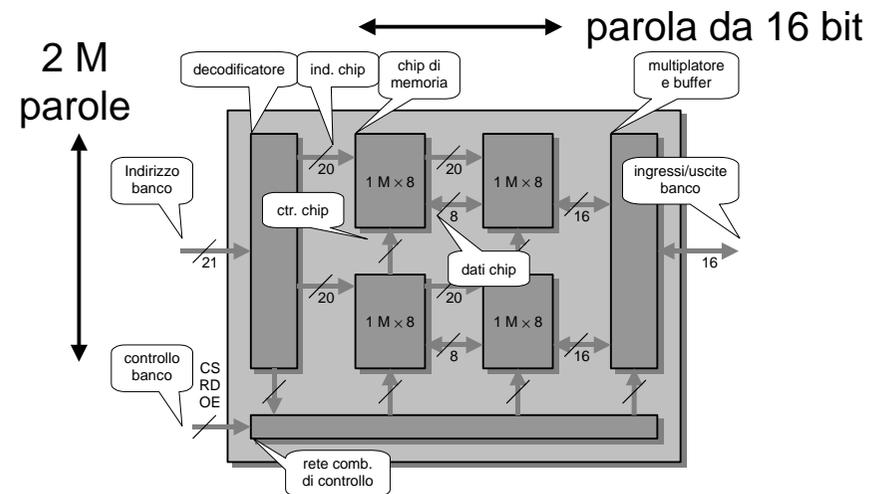
Informatica II - Livello logico (3)

28

## Esempio

- Si supponga di disporre di 4 chip di memoria da  $1\text{ M} \times 8$  ciascuno
- Si desidera ottenere un banco di memoria da  $2\text{ M} \times 16$
- Il numero di chip è sufficiente:  
 $4 (1\text{ M} \times 8) = 32\text{ Mbit}$  e  
 $2\text{ M} \times 16 = 32\text{ Mbit}$
- Occorre comporre una matrice  $2 \times 2$ : 2 righe di 2 chip di memoria ciascuna

## Struttura



13-04.-03

Informatica II - Livello logico (3)

29

13-04.-03

Informatica II - Livello logico (3)

30

## Tecnologie di memoria

- Esistono svariate tecnologie di memoria
- Dipendono:
  - dalla capacità della memoria
  - dal tempo di accesso a una parola (in lettura o in scrittura)
  - dalla politica di accesso: lettura e scrittura, sola lettura, programmabilità sul campo
  - dalla stabilità: volatile o persistente
  - e dal costo

## RAM Statica (SRAM)

- Memoria RAM (Random Access Memory) realizzata con bistabili
- Capacità medio-piccola
- Tempo di accesso molto breve
- Funziona in lettura e scrittura
- Volatile: senza alimentazione il contenuto della memoria svanisce
- Usi: svariati, in particolare come cache

13-04.-03

Informatica II - Livello logico (3)

31

13-04.-03

Informatica II - Livello logico (3)

32

## RAM Dinamica (DRAM)

- La memoria SRAM consuma parecchi transistor per bit memorizzato (circa 6 transistor per bit)
- La tecnologia DRAM usa circa 1 transistor per bit memorizzato
- Sfrutta il fenomeno dell'accumulo temporaneo di carica sul transistor
- Internamente contiene un circuito di rinfresco che rigenera le cariche sui transistor prima che queste svaniscano

13-04.-03

Informatica II - Livello logico (3)

33

## RAM Dinamica (DRAM)

- Memoria RAM realizzata come matrice di transistor, ad altissima densità
- Capacità grande-grandissima
- Tempo di accesso medio
- Funziona in lettura e scrittura
- Volatile: senza alimentazione il contenuto della memoria svanisce
- Usi: numerosissimi, la memoria centrale dei calcolatori normalmente è DRAM

13-04.-03

Informatica II - Livello logico (3)

34

## ROM

- Memoria ROM (Read Only Memory), realizzata come matrice di transistor
- Capacità grande-grandissima
- Tempo di accesso medio
- Funziona in sola lettura
- Persistente: il contenuto permane anche in assenza di alimentazione
- Usi: per memorizzare programmi permanenti, non modificabili; grandi volumi di produzione

13-04.-03

Informatica II - Livello logico (3)

35

## PROM, EPROM, EEPROM

- Capacità e tempo simili alla ROM
- Sola lettura e persistenti
- Sono programmabili sul campo, tramite un apposito programmatore:
  - PROM: programmabile una volta sola
  - EPROM: cancellabile con raggi UV
  - EEPROM: cancellabile elettricamente (si può anche scrivere un solo byte per volta)
- Usi: piccoli volumi di produzione, prototipi

13-04.-03

Informatica II - Livello logico (3)

36

## Memoria FLASH

- Capacità e tempo simili alla DRAM (o solo di poco inferiori)
- Funziona in lettura e scrittura (la scrittura però è a blocchi di byte)
- Persistente: il contenuto permane anche in assenza di alimentazione
- Usi: dati multimediali (p. es. immagini statiche, sequenze video), programmi fissi ma periodicamente aggiornabili

13-04.-03

Informatica II - Livello logico (3)

37

## Tabella riassuntiva

Tipo	Categoria	Modalità di cancellazione	Scrittura byte	Volatile	Usi specifici
SRAM	lett/scritt	elettrica	si	si	cache
DRAM	lett/scritt	elettrica	si	si	mem. centrale
ROM	sola lett	nessuna	no	no	grandi vol.
PROM	sola lett*	nessuna	no	no	piccoli vol.
EPROM	sola lett*	luce UV	no	no	prototipi
EEPROM	sola lett*	elettrica	si (lenta)	no	prototipi
FLASH	lett/scritt	elettrica	a blocchi	no	multimedia

\*Le memorie cancellabili vengono talvolta qualificate come “memorie prevalentemente a sola lettura” (read-mostly), invece che “a sola lettura” (read-only)

13-04.-03

Informatica II - Livello logico (3)

38

## Unità funzionali di I/O e CPU

13-04.-03

Informatica II - Livello logico (3)

39

## Unità funzionali di I/O

- Il calcolatore contiene svariati circuiti digitali aventi la funzione di gestire le comunicazioni con le periferiche collegate al calcolatore stesso
- Essi vengono chiamati unità funzionali di interfacciamento, o di I/O
- Essi sono blocchi funzionali, più o meno complessi, costituiti a loro volta da reti combinatorie, reti sequenziali e altri blocchi funzionali standard di libreria

13-04.-03

Informatica II - Livello logico (3)

40

## Unità di interfacciamento

- Le unità funzionali di interfacciamento (o di I/O) gestiscono:
  - disco magnetico e ottico
  - terminale
  - tastiera e puntatore (o mouse)
  - video, casse audio, cuffia e microfono
  - stampante e scanner
  - connessione di rete
  - e altre periferiche ancora ...

13-04.-03

Informatica II - Livello logico (3)

41

## Unità di interfacciamento

- Le unità funzionali di interfacciamento mettono in comunicazione il calcolatore (processore) con le periferiche
- Esse hanno pertanto due lati:
  - verso la periferica, con cui scambiano segnali specifici, dipendenti dalle caratteristiche della periferica
  - verso il calcolatore (processore), utilizzando una porta di accesso simile a quella della memoria

13-04.-03

Informatica II - Livello logico (3)

42

## Esempio

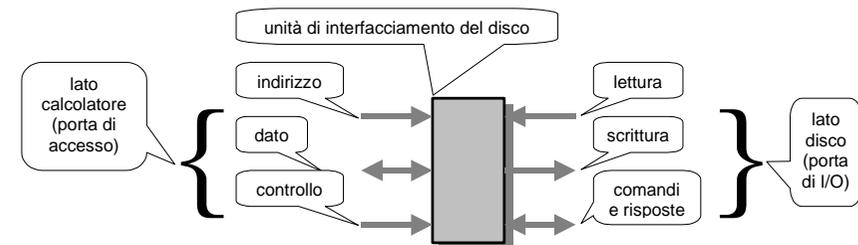
- L'unità funzionale di interfacciamento del disco rigido magnetico (hard disk), o controllore del disco rigido, ha:
  - un lato periferica, collegato al disco rigido, tramite cui manda al disco i comandi di azionamento disco e i dati da scrivere, e tramite cui riceve dal disco le risposte ai comandi e i dati da leggere
  - un lato calcolatore, costituito da una porta di accesso, simile a quella della memoria

13-04.-03

Informatica II - Livello logico (3)

43

## Controllore disco



Unità di interfacciamento del disco rigido: legge e scrive dati da e verso il disco, e li scambia con il calcolatore. I comandi al disco dipendono fortemente dal tipo di disco

13-04.-03

Informatica II - Livello logico (3)

44

## Controllore disco

- Il disco è organizzato in settori, facce (o testine di lett./scritt) e cilindri (o tracce)
- Le operazioni sono: lettura di un settore, scrittura di un settore
- L'interfaccia verso il lato calcolatore funziona come un banco di registri, tramite i quali si controllano le operazioni del disco
- Ogni registro ha un indirizzo, proprio come se fosse una parola di memoria

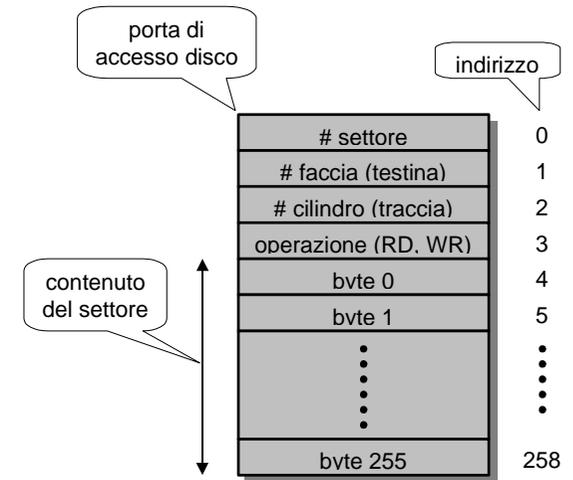
13-04.-03

Informatica II - Livello logico (3)

45

## Controllore disco

- Esempio (generico) di porta di accesso al disco
- I registri possono variare
- Settore: 256 byte



13-04.-03

Informatica II - Livello logico (3)

46

## Operazioni del controllore

- Lettura di un settore:
  - il calcolatore scrive nei registri i # di settore, faccia e cilindro, e il codice dell'operazione di lettura
  - dopo qualche tempo, il contenuto del settore è disponibile nei registri 0, ..., 255
  - il calcolatore legge il contenuto dei registri 0, ..., 255, e lo elabora
- Lettura e scrittura dei registri funzionano come per le parole di memoria

13-04.-03

Informatica II - Livello logico (3)

47

## Operazioni del controllore

- Scrittura di un settore:
  - il calcolatore scrive nei registri 0, ..., 255 il contenuto del settore, i # di settore, faccia e cilindro, e il codice dell'operazione di scrittura
  - dopo qualche tempo, il contenuto del settore sarà stato ricopiato nella posizione appropriata del disco
- Lettura e scrittura dei registri funzionano come per le parole di memoria

13-04.-03

Informatica II - Livello logico (3)

48

## Struttura interna

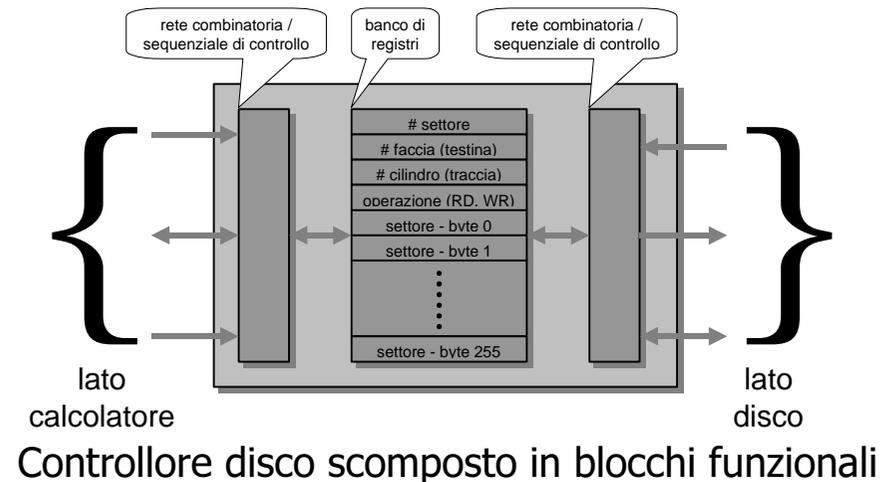
- La struttura interna di una generica unità di interfacciamento del disco è riconducibile, in stile di progetto funzionale, a blocchi funzionali combinatori e sequenziali noti
- Blocchi: banco di registri, rete combinatoria / sequenziale di controllo lato calcolatore, e rete combinatoria / sequenziale di controllo lato disco

13-04.-03

Informatica II - Livello logico (3)

49

## Struttura interna



13-04.-03

Informatica II - Livello logico (3)

50

## Considerazioni

- Le unità funzionali di interfacciamento con le periferiche dipendono fortemente dalla tecnologia delle periferiche stesse
- Esse sono comunque progettabili in stile funzionale, tramite blocchi funzionali
- Per il calcolatore un'unità di interfacciamento è essenzialmente un banco di registri, aventi però funzioni specializzate (a seconda della periferica)

13-04.-03

Informatica II - Livello logico (3)

51

## Unità di interfacciamento

- Numerose periferiche hanno caratteristiche di comunicazione e controllo simili
- Esiste una libreria di unità funzionali di interfacciamento, o "chip di I/O" (chip di ingresso / uscita), standardizzate, adatte a svariati tipi di periferica
- Ognuna di queste unità funzionali standard rispetta un determinato sistema (o protocollo) di comunicazione

13-04.-03

Informatica II - Livello logico (3)

52

## Unità standard

- Controllore del disco: i dischi obbediscono a svariati protocolli di comunicazione (p. es. gli standard IDE e SCSI); ognuno di questi dispone di unità funzionali standardizzate
- Controllore CRT (Controllore Catodhe ray Tube): è il nome generico delle unità funzionali di controllo e interfacciamento del terminale

13-04.-03

Informatica II - Livello logico (3)

53

## Unità standard

- UART (Universal Asynchronous Receiver Transmitter): è l'unità universale di trasmissione e ricezione asincrona:
  - scambia con il calcolatore un byte di dati alla volta (p. es. 1 byte per ciclo di clock)
  - comunica con la periferica in modo seriale, cioè inviandole o ricevendo un bit alla volta
  - è parzialmente programmabile: velocità di trasmissione (# bit / sec), metodo di rilevamento errori, formato dei dati, ...

13-04.-03

Informatica II - Livello logico (3)

54

## Unità standard

- USART (Universal Synchronous and Asynchronous Receiver Transmitter): è l'unità universale di trasmissione e ricezione sincrona e asincrona:
  - dispone di tutte le funzioni della UART
  - in aggiunta è in grado di comunicare con la periferica in modo seriale sincrono, cioè supponendo che la periferica e la USART dispongano di un segnale di clock comune

13-04.-03

Informatica II - Livello logico (3)

55

## Unità standard

- PIO (Parallel Input Output): è l'unità universale di trasmissione e ricezione parallela:
  - scambia con il calcolatore un byte di dati alla volta (p. es. 1 byte per ciclo di clock)
  - scambia con la periferica un byte di dati alla volta (p. es. 1 byte per ciclo di clock)
  - può operare in modo sincrono o asincrono
  - può comunicare con più periferiche

13-04.-03

Informatica II - Livello logico (3)

56

## Esempio di PIO

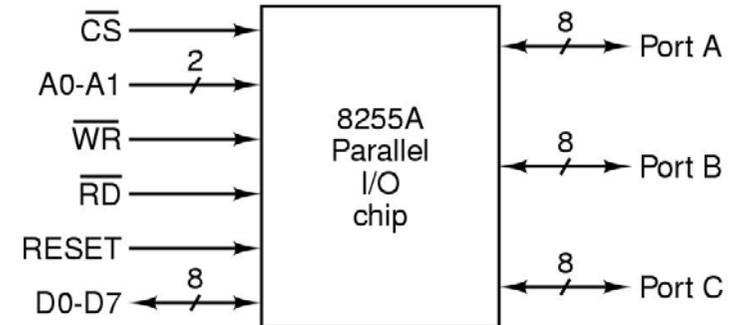
- Il chip di I/O Intel 8255A è una tipica unità funzionale PIO:
  - sul lato periferica dispone di tre porte da 8 bit ciascuna, per gestire fino a 3 periferiche
  - sul lato calcolatore la porta di accesso ha 8 bit di dato (D0-D7), 2 linee di indirizzo (A0, A1) (dunque gli indirizzi sono  $2^2 = 4$ ), il comando di selezione chip (CS), il comando di lettura (RD) e, separato da questo, il comando di scrittura (WR), e il comando di ripristino (RESET)

13-04.-03

Informatica II - Livello logico (3)

57

## Interfaccia PIO



Collegamenti della PIO (è un circuito integrato singolo, commercialmente disponibile)

13-04.-03

Informatica II - Livello logico (3)

58

## Operazioni della PIO

- La PIO contiene un registro parallelo a 8 bit (1 byte), con funzione di caricamento, per ogni porta di periferica
- Lettura della periferica:
  - La periferica A (B, C) invia 1 byte alla porta A (B, C) della PIO, che lo memorizza nel registro associato alla porta A (B, C)
  - In seguito, il calcolatore può leggere il contenuto del registro A (B, C) della PIO

13-04.-03

Informatica II - Livello logico (3)

59

## Operazioni della PIO

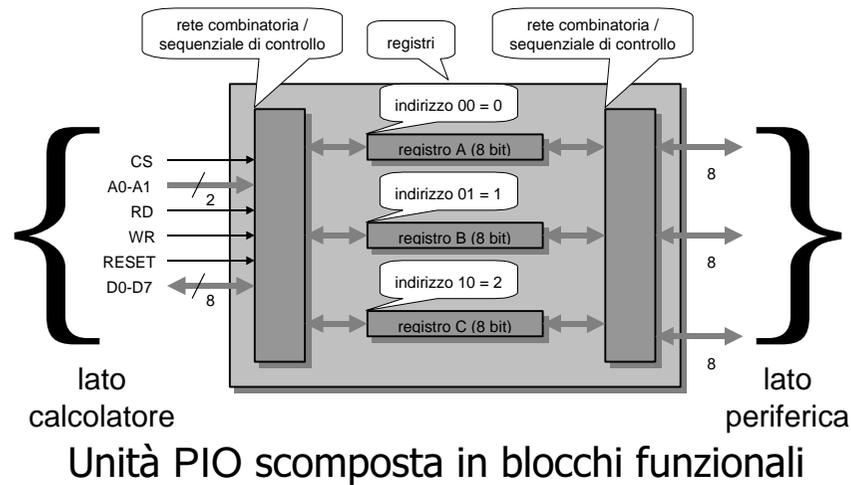
- Scrittura della periferica:
  - Il calcolatore scrive 1 byte nel registro A (B, C) della PIO
  - La PIO invia sulla porta A (B, C) il dato scritto nel registro A (B, C), che verrà così ricevuto dalla periferica
  - La PIO dispone anche di logica interna e di segnali esterni per gestire le periferiche non immediatamente disponibili a ricevere dati (tramite uno scambio di impulsi di sincronizzazione con la periferica)

13-04.-03

Informatica II - Livello logico (3)

60

## Struttura interna della PIO



13-04.-03

Informatica II - Livello logico (3)

61

## Porte di I/O del calcolatore

- Molti calcolatori (p. es. PC Intel) hanno interfacce standard verso le periferiche:
  - porta seriale (standard EIA RS 232)
  - porta parallela (standard CENTRONICS)
  - porta puntatore (o mouse, standard PS/2)
  - porta video (vari standard)
  - porte disco (standard IDE, SCSI, ...)
  - porta di rete (standard IEEE 802.3)
  - e altre porte ancora ...

13-04.-03

Informatica II - Livello logico (3)

62

## Porte del calcolatore

- Oggigiorno la maggior parte delle porte di I/O dei calcolatori è realizzata tramite chip di I/O disponibili come componenti standard di libreria:
  - porta seriale: chip UART 16550A (con i perfezionamenti successivi 16550B e C)
  - e così via ...
- Anche i connettori presenti sul calcolatore sono standardizzati

13-04.-03

Informatica II - Livello logico (3)

63

## Chipset

- I chip di I/O sono spesso organizzati in famiglie di circuiti integrati, chiamate "chipset" (insiemi di chip)
- Ogni chipset contiene svariati chip di I/O, ognuno dedicato a un particolare tipo di porta di I/O e a uno specifico standard di comunicazione
- I chipset commerciali più diffusi consentono di collegare tutte le periferiche più importanti e diffuse

13-04.-03

Informatica II - Livello logico (3)

64

## Mappa di indirizzamento

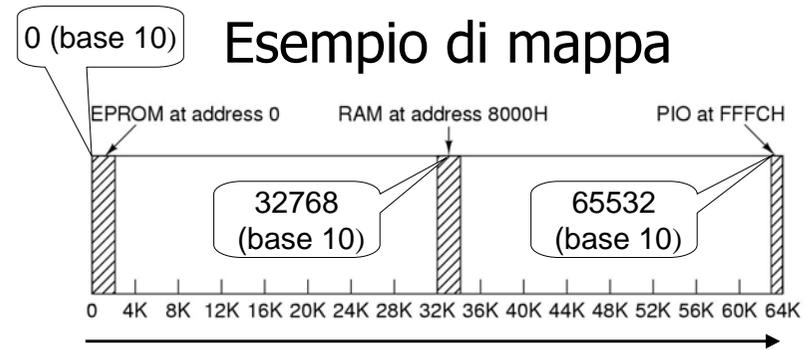
- Banche di memoria e unità funzionali di interfacciamento fanno entrambi uso di indirizzi per individuare le parole (di memoria) e i registri (di interfacciamento) su cui il calcolatore deve operare
- Pertanto ogni calcolatore definisce una propria "mappa di indirizzamento", specificando quali indirizzi sono attribuiti alla memoria e quali alle varie unità funzionali di interfacciamento presenti

13-04.-03

Informatica II - Livello logico (3)

65

## Esempio di mappa



successione di indirizzi, da 0 a  $64\text{ K} - 1 = 2^{16} - 1 = 65535$

EPROM (2K byte), RAM (2K byte) e PIO.  
Gli indirizzi non assegnati alla memoria o a unità funzionali sono disponibili per usi futuri.  
Gli indirizzi sono in formato esadecimale.

13-04.-03

Informatica II - Livello logico (3)

66

## Considerazioni

- Ogni calcolatore definisce una propria mappa di indirizzamento, garantendo assegnamenti univoci di indirizzi
- Classi di calcolatori aventi la stessa architettura adottano la stessa mappa di indirizzamento, per compatibilità
- Esempio: i calcolatori con architettura PC Intel adottano lo standard industriale MS-DOS per l'assegnamento degli indirizzi (con alcune estensioni)

13-04.-03

Informatica II - Livello logico (3)

67

## Processore

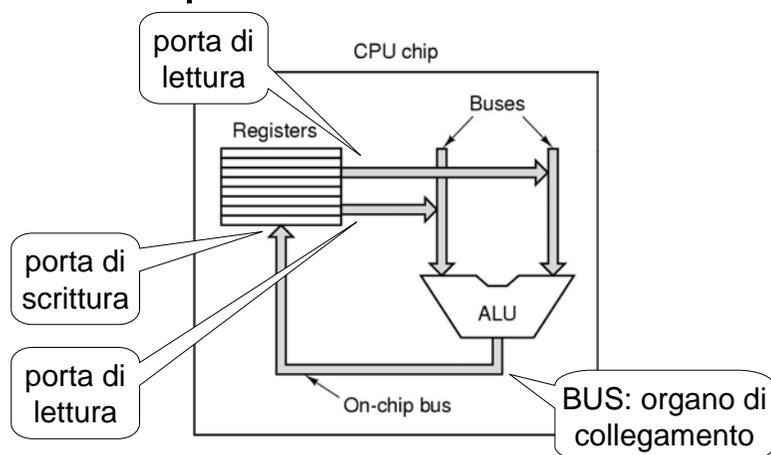
- Il processore, o CPU (Central Processing Unit, unità centrale di elaborazione) è l'unità funzionale più complessa e più raffinata presente nel calcolatore
- Tutti i processori moderni sono realizzati su un unico circuito integrato
- Il processore è un (complesso) sistema realizzato aggregando numerosi blocchi funzionali combinatori e sequenziali

13-04.-03

Informatica II - Livello logico (3)

68

## Esempio di struttura di CPU



Il cuore di una CPU: la ALU e i registri

## La CPU come blocco funzionale

- La CPU ha le funzioni seguenti:
  - preleva le istruzioni del programma dalla memoria del calcolatore e le esegue
  - legge e scrive dati dalla e nella memoria centrale del calcolatore, e li elabora, secondo le direttive del programma
  - riceve o invia dati dalle o verso le periferiche, e li elabora, secondo le direttive del programma

## La piedinatura della CPU

- Poiché la CPU è tutta integrata su un unico circuito integrato (chip), la sua interfaccia di comunicazione è completamente definita dalla piedinatura del chip ospitante la CPU
- I piedini della CPU sono divisi in tre grandi gruppi: piedini di indirizzo, di dato e di controllo
- Essi sono in corrispondenza con quelli di memoria e di unità di interfacciamento

## Esempio di operazione

- Ecco come la CPU preleva un'istruzione dalla memoria del calcolatore:
  - la CPU invia l'indirizzo della parola di memoria contenente l'istruzione ai piedini di indirizzo
  - poi attiva i piedini di controllo per comandare alla memoria un'operazione di lettura
  - la memoria invia la parola richiesta ai piedini dati della CPU e invia ai piedini di controllo la conferma che il comando è stata eseguito
  - ricevuta la conferma, la CPU accetta la parola ed esegue l'istruzione che questa codifica

## Considerazioni

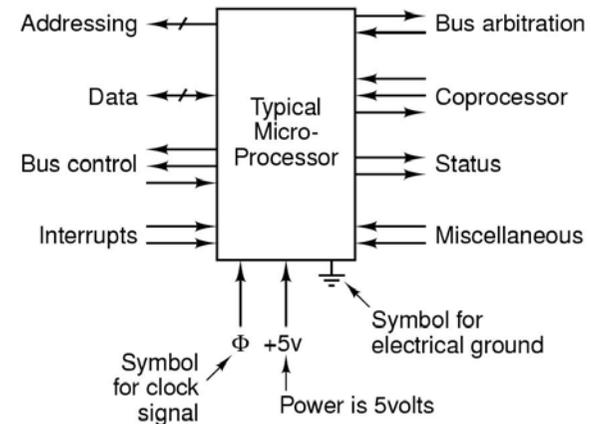
- La lettura e la scrittura di dati da o in memoria funzionano in modo del tutto simile al prelievo di un'istruzione
- La ricezione o l'invio di dati dalle o alle periferiche funzionano in modo analogo
- La CPU non ha altro modo di comunicare con il resto del calcolatore (e in definitiva con l'esterno) che quello appena descritto

13-04.-03

Informatica II - Livello logico (3)

73

## Piedinatura di una CPU tipica



13-04.-03

Informatica II - Livello logico (3)

74

## Il BUS del calcolatore

13-04.-03

Informatica II - Livello logico (3)

75

## Il BUS del calcolatore

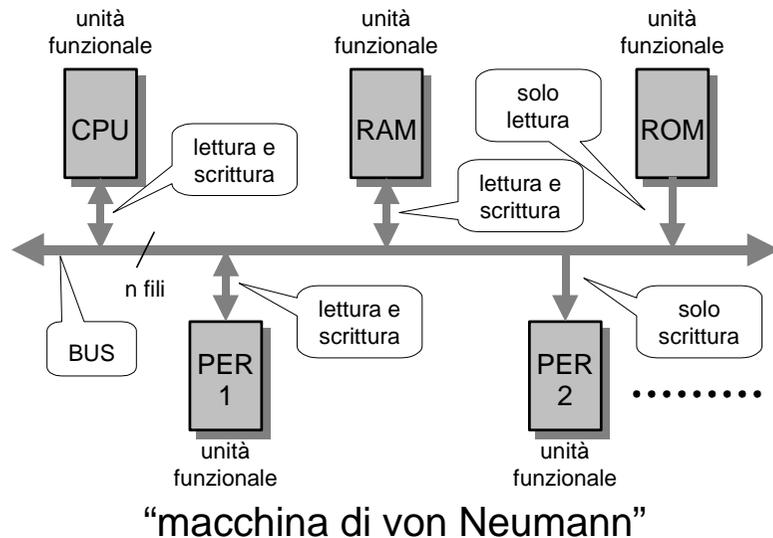
- Il calcolatore elettronico è un insieme di unità funzionali:
  - l'unità centrale di elaborazione (CPU), o processore
  - le unità funzionali di memoria, o banchi di memoria
  - le unità funzionali di interfacciamento alle periferiche
- Le unità sono interconnesse tramite un organo di collegamento: il BUS

13-04.-03

Informatica II - Livello logico (3)

76

## Schema del calcolatore



13-04.-03

Informatica II - Livello logico (3)

77

## BUS interni ed esterni

- I BUS del calcolatore si distinguono in:
  - BUS interni, confinati all'interno di una singola unità funzionale, e che collegano i blocchi funzionali contenuti nell'unità
  - BUS esterni, che si estendono all'esterno dell'unità funzionale, e che la collegano alle altre unità funzionali
- I BUS esterni del calcolatore sono solitamente standardizzati

13-04.-03

Informatica II - Livello logico (3)

78

## Tipi di BUS

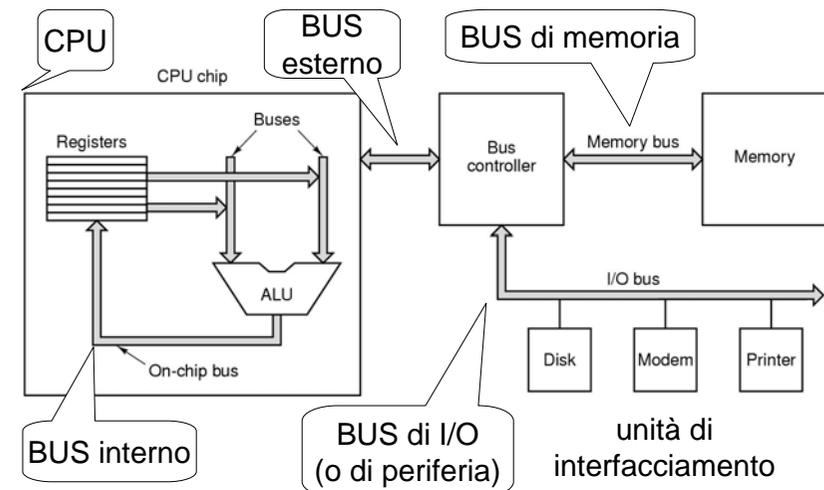
- I primi calcolatori erano dotati di un unico BUS esterno (BUS di sistema), collegante CPU, memoria e unità di I/O
- La maggior parte dei calcolatori odierni (2000) è dotata di due bus esterni:
  - BUS di memoria, collegante CPU e unità funzionali di memoria (banchi di memoria)
  - BUS di I/O, collegante CPU e unità funzionali di I/O
- Possono esserci più di due bus esterni

13-04.-03

Informatica II - Livello logico (3)

79

## Sistema con diversi BUS



13-04.-03

Informatica II - Livello logico (3)

80

## Funzioni dei BUS

- I BUS interni alle unità funzionali (specialmente i BUS interni alla CPU), non sono standardizzati, e spesso non sono nemmeno resi pubblici
- I BUS esterni del calcolatore sono invece standardizzati e pubblici: definiscono le regole di collegamento elettrico e di comunicazione tra le unità (o protocolli di comunicazione)

13-04.-03

Informatica II - Livello logico (3)

81

## Tecnologie dei BUS

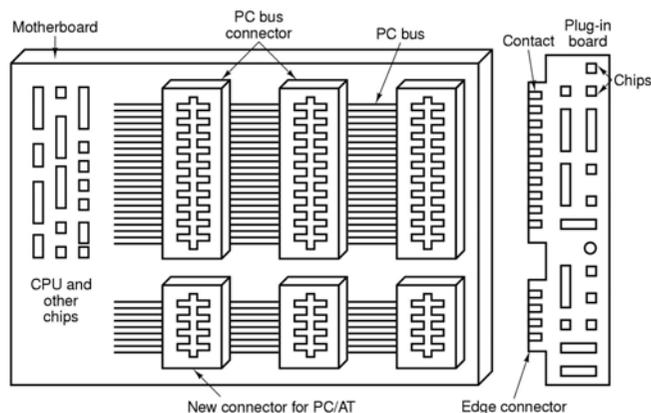
- La maggior parte dei BUS esterni è realizzata tramite collegamenti elettrici:
  - schede di BUS, con piste di collegamento e connettori montati sulla scheda
  - cavi elettrici flessibili connettorizzati
- Alcuni BUS, ad altissime prestazioni, sono realizzati in fibra ottica (FiberChannel)
- Alcuni BUS recenti e innovativi si basano su etere (onde radio, Bluetooth)

13-04.-03

Informatica II - Livello logico (3)

82

## Esempio: il BUS PC/AT



È un bus di sistema (CPU, memoria e I/O)

13-04.-03

Informatica II - Livello logico (3)

83

## Standard di BUS

- Standard di BUS esterno di sistema:
  - Omnibus (calcolatore PDP-8, obsoleto)
  - Unibus (calcolatore PDP-11, obsoleto)
  - Multibus (processore Intel 8086)
  - PC/XT BUS (calcolatore IBM PC XT)
  - BUS ISA (calcolatore Intel PC)
  - BUS EISA (calcolatore Intel PC 80386)
  - Microchannel (calcolatore IBM PC PS/2)
  - BUS PCI (calcolatore Intel PC odierno)
  - NuBUS (calcolatore Macintosh)

13-04.-03

Informatica II - Livello logico (3)

84

## Standard di BUS

- Standard di BUS esterno di I/O:
  - RS 232: unità di I/O seriali a bassa velocità (porta seriale esterna del PC)
  - CENTRONICS: unità di I/O parallele (porta parallela esterna del PC)
  - BUS IDE: dischi a media velocità
  - BUS SCSI: dischi ad alta velocità
  - FiberChannel: vettori di dischi (disk array), ad altissima velocità (BUS in fibra ottica)

13-04.-03

Informatica II - Livello logico (3)

85

## Standard di BUS

- Standard di BUS esterno di I/O:
  - USB, Universal Serial BUS: unità di I/O seriali a bassa / media / alta velocità
  - svariati standard di BUS esterno per il collegamento di video
  - FireWire: bus per elettronica consumer
  - BUS VME: attrezzature di laboratorio
  - BUS CAMAC: fisica delle alte energie
  - BUS CAN: applicazioni automobilistiche
  - e altri ancora ...

13-04.-03

Informatica II - Livello logico (3)

86

## Controllo del BUS

- In ogni istante, una sola unità funzionale possiede il controllo del BUS, cioè decide quali operazioni di comunicazione eseguire
- Generalmente la CPU possiede il controllo del BUS (o dei BUS, se ve n'è più d'uno), ma può anche cedere temporaneamente questo ruolo ad altre unità funzionali

13-04.-03

Informatica II - Livello logico (3)

87

## Master e slave

- L'unità funzionale che detiene il controllo del BUS si chiama MASTER:
  - decide quale operazione eseguire, lettura oppure scrittura, e in quale istante di tempo
  - decide qual è l'unità funzionale da leggere oppure da scrivere
- Le rimanenti unità funzionali, che non detengono il controllo del BUS, si chiamano SLAVE

13-04.-03

Informatica II - Livello logico (3)

88

## Relazione tra BUS e MASTER

- Ogni BUS è controllato da un unico MASTER
- Ogni MASTER può controllare  $n \geq 1$  BUS
- Ogni BUS può collegare più SLAVE
- In un sistema con  $n \geq 1$  BUS, vi possono essere al massimo  $n$  MASTER
- Il calcolatore deve possedere almeno un MASTER (di solito la CPU)
- Un calcolatore con un solo BUS deve contenere esattamente un MASTER

## Ruoli di MASTER e SLAVE

Master	Slave	Esempio
CPU	Memoria	Prelievo istruzioni e lettura/scrittura dati
CPU	Unità di I/O	Ricezione/invio dati da/a un'unità di I/O
CPU	Coprocessore	La CPU dà istruzioni al coprocessore
I/O	Memoria	Accesso diretto alla memoria (DMA)*
Coprocessore	CPU	Il coprocessore legge operandi dalla CPU

\*È una tecnica di trasferimento dati tra periferica e memoria che bypassa la CPU, escludendola

## Connessione al BUS

- Per potersi collegare al BUS, le unità MASTER e SLAVE sono dotate di un componente (o blocco funzionale) di interfaccia, chiamato BUS DRIVER
- Il BUS driver è in grado di:
  - collegarsi e scollegarsi elettricamente al o dal BUS (p. es. tramite la tecnica di alta impedenza, o anche tramite altre tecniche)
  - amplificare opportunamente i segnali da trasmettere / ricevere sul / dal BUS

## Problematiche del BUS

- Il progetto di BUS esterni per calcolatori comprende quattro aspetti principali:
  - struttura e dimensione, cioè numero di collegamenti presenti nel BUS
  - gestione della temporizzazione del BUS
  - arbitraggio degli accessi al BUS
  - protocollo di comunicazione del BUS
- Per progettare o descrivere un BUS occorre considerarli tutti

## Struttura del BUS

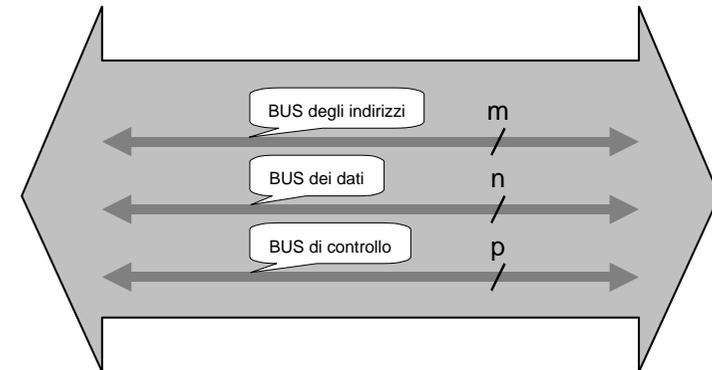
- Un generico BUS esterno di calcolatore ha una struttura simile alla piedinatura della CPU, cui va collegato
- È diviso in tre componenti:
  - BUS degli indirizzi (address BUS)
  - BUS dei dati (data BUS)
  - BUS di controllo (control BUS)
- La corrispondenza con i piedini della CPU non è però sempre uno-a-uno

13-04.-03

Informatica II - Livello logico (3)

93

## Componenti del BUS



m, n e p: larghezze dei componenti del BUS.  
Il BUS di controllo è quello meno omogeneo.

13-04.-03

Informatica II - Livello logico (3)

94

## Versioni del BUS

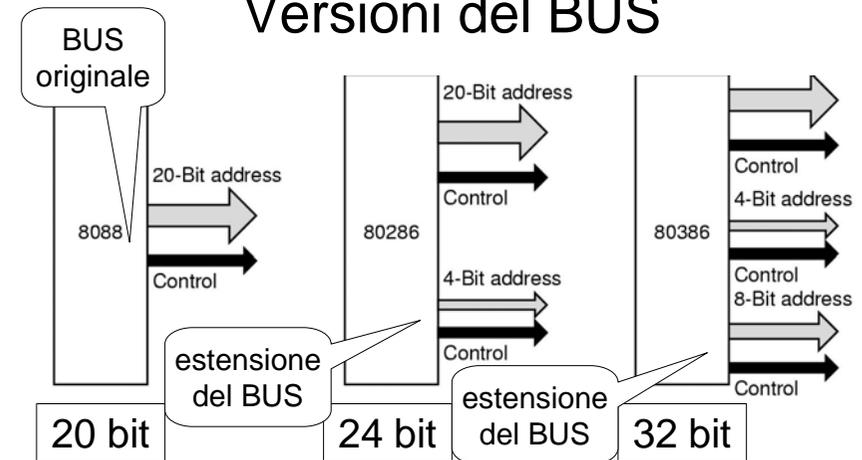
- La potenza di calcolo dei processori cresce a un tasso costante (secondo la legge di Moore)
- Di conseguenza, svariati tipi di BUS sono stati progressivamente potenziati:
  - aumento della larghezza del BUS indirizzi
  - aumento della larghezza del BUS dati
  - e anche qualche aggiunta di segnali al BUS di controllo

13-04.-03

Informatica II - Livello logico (3)

95

## Versioni del BUS



Raffinamento di un bus (a, b, c) al crescere della potenza del processore

13-04.-03

Informatica II - Livello logico (3)

96

## Dimensione del BUS

- Dimensione (o larghezza) del BUS indirizzi: con  $m \geq 1$  linee si ottengono  $2^m \geq 2$  indirizzi, numerati da 0 a  $2^m - 1$
- Dimensione (o larghezza) del BUS dati: con  $n \geq 1$  linee si possono scambiare (leggere o scrivere) parole da  $n$  bit
- Per installare nel calcolatore grandi quantità di memoria occorrono BUS indirizzi (e anche dati) molto larghi

13-04.-03

Informatica II - Livello logico (3)

97

## BUS multiplati

- Si possono riunire le funzioni del BUS di indirizzo e del BUS di dato in un unico gruppo di linee: il BUS multiplato
- Quando il processore deve leggere:
  - nel primo ciclo invia l'indirizzo sul BUS multiplato
  - nel secondo ciclo riceve il dato, sempre sul BUS multiplato
- La scrittura funziona in modo simile
- Meno collegamenti, ma velocità inferiore

13-04.-03

Informatica II - Livello logico (3)

98

## Funzionamento del BUS

- Le attività del calcolatore si sviluppano per cicli di BUS: in ogni ciclo avviene un'operazione (o transazione)
- Le operazioni sono governate dal MASTER che detiene il controllo del BUS
- Gli SLAVE non possono dare inizio a un'operazione in modo autonomo
- Operazioni:trasferimento di dati tra MASTER e SLAVE

13-04.-03

Informatica II - Livello logico (3)

99

## Operazioni del BUS

- Operazione di lettura: un dato viene trasferito da uno SLAVE al MASTER
- Operazione di scrittura: un dato viene trasferito dal MASTER a uno SLAVE
- Nota bene: il punto di vista per stabilire la direzione del trasferimento (lettura o scrittura) è sempre e solo quello del MASTER che detiene (in quel momento) il controllo del BUS

13-04.-03

Informatica II - Livello logico (3)

100

## Collegamento al BUS indirizzi

- Poiché è il MASTER a dare inizio a un'operazione di lettura o scrittura, solo il MASTER può immettere un indirizzo sul BUS degli indirizzi
- Gli SLAVE non possono generare indirizzi
- Gli SLAVE corrispondono agli indirizzi, secondo la mappa di indirizzamento del calcolatore

13-04.-03

Informatica II - Livello logico (3)

101

## Collegamento al BUS dati e al BUS di controllo

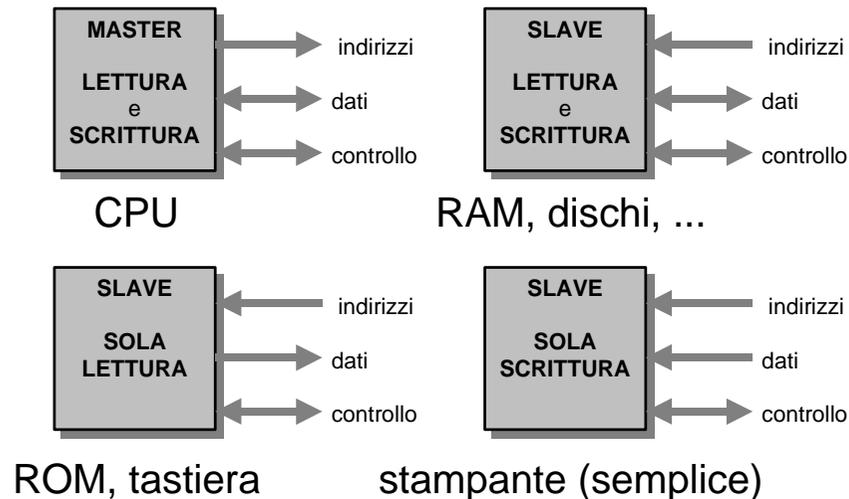
- I collegamenti delle unità funzionali al BUS dei dati possono variare, a seconda che l'unità funzionale (MASTER o SLAVE) sia in lettura e scrittura, sola lettura o sola scrittura
- I collegamenti delle unità funzionali al BUS di controllo sono quello meno regolari e classificabili: dipende dalle funzioni dell'unità considerata

13-04.-03

Informatica II - Livello logico (3)

102

## Schemi di collegamento BUS



13-04.-03

Informatica II - Livello logico (3)

103

## Cicli di BUS

- L'esecuzione di un programma da parte del calcolatore si esplica in una successione di cicli di BUS:
  - lettura di una parola di memoria
  - scrittura di una parola di memoria
  - lettura di un registro di I/O
  - scrittura di un registro di I/O
  - riposo: il BUS non viene usato

13-04.-03

Informatica II - Livello logico (3)

104

## Divisione in cicli di BUS

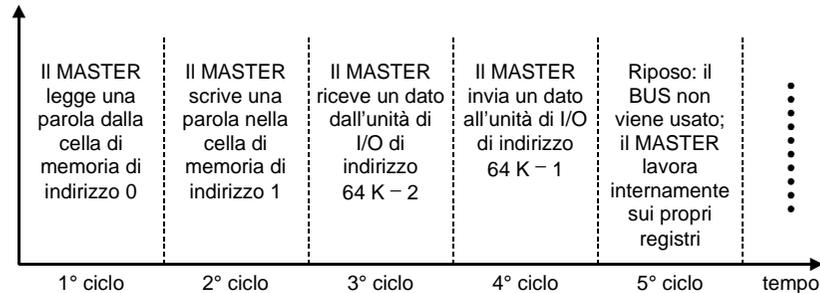


Diagramma temporale (schematico)  
dei cicli di BUS

## Sottocicli di BUS

- Una singola operazione di lettura o scrittura può anche svilupparsi su più cicli di BUS
- L'uso di più cicli di BUS può rendersi necessario quando un MASTER veloce deve trasferire dati con uno SLAVE lento
- Il numero di sottocicli può essere fisso o anche variabile

## Tipi di BUS di calcolatore ed esempi

## Temporizzazione del BUS

- Il funzionamento del BUS del calcolatore procede per cicli di BUS
- Esistono due metodi fondamentali per realizzare la scansione dei cicli di BUS:
  - BUS sincrono
  - BUS asincrono
- I due metodi si distinguono per la presenza o meno di un segnale di clock

## BUS sincrono

- Il BUS di controllo contiene una linea che trasporta un segnale di clock, a frequenza prestabilita
- Il clock viene distribuito a tutte le unità funzionali collegate al BUS
- Il segnale di clock scandisce le varie transizioni di segnale e il passaggio da un ciclo di BUS al ciclo successivo
- Tutte le unità sanno sempre quando si passa al ciclo successivo

13-04.-03

Informatica II - Livello logico (3)

109

## Esempio

- Lettura di una parola dalla memoria
- Bus sincrono, periodo di clock = 40 ns
- Frequenza =  $1 / 40 \text{ ns} = 25 \text{ MHz}$
- La memoria presenta un ritardo di lettura pari a 40 ns, cioè le occorrono 40 ns per presentare il dato richiesto alle uscite collegate al BUS dati
- Occorrono 3 cicli di clock (T1, T2 e T3) per portare a termine la lettura

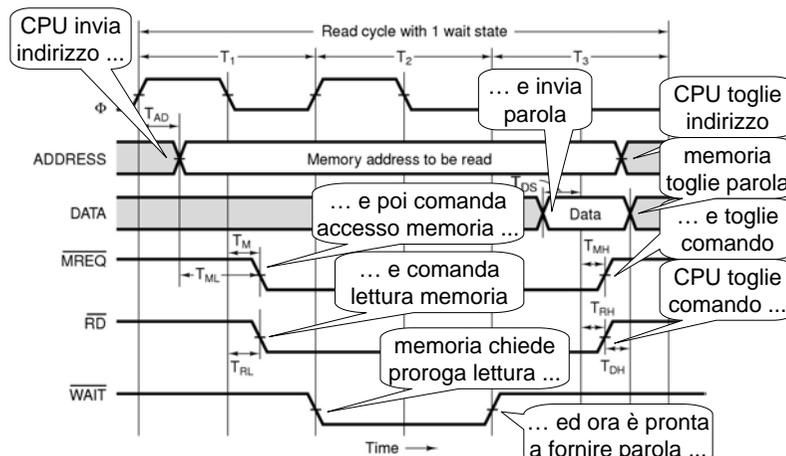
13-04.-03

Informatica II - Livello logico (3)

110

## Diagramma temporale

Animazione



La CPU prepara il dato a richiesta memoria

13-04.-03

Informatica II - Livello logico (3)

111

## Segnali di controllo

- MREQ (Memory Request): il MASTER comanda alla memoria un'operazione di accesso; è attivo basso
- RD (Read): il MASTER comanda una lettura; è attivo basso
- WAIT: lo SLAVE richiede al MASTER una proroga dell'operazione (cioè un ciclo di BUS aggiuntivo, o STATO DI ATTESA), perché ha bisogno di tempo; è attivo basso

13-04.-03

Informatica II - Livello logico (3)

112

## Fasi dell'operazione di lettura

- Ciclo T1: il MASTER (CPU) manda l'indirizzo sul BUS indirizzi e attiva MREQ e RD (dopo avere stabilizzato l'indirizzo)
- Ciclo T2: lo SLAVE (memoria) attiva WAIT e così richiede una proroga di un ciclo di BUS, per avere tempo bastante a fornire la parola
- Ciclo T3: lo SLAVE ha pronta la parola, perciò disattiva WAIT e fornisce la parola sul BUS dati; il MASTER legge la parola dal BUS dati, toglie l'indirizzo e disattiva MREQ e RD

13-04.-03

Informatica II - Livello logico (3)

113

## Considerazioni

- Poiché la memoria presenta un ritardo di lettura pari a 40 ns, non è in grado di fornire la parola richiesta entro il ciclo di clock T2
- Essa pertanto richiede una proroga, cioè uno STATO DI ATTESA, che è proprio il ciclo di clock T2
- L'operazione si conclude nel ciclo T3

13-04.-03

Informatica II - Livello logico (3)

114

## Ritardi

- Nel diagramma temporale di lettura nessun fronte di salita e discesa è disegnato perfettamente verticale
- Ogni transizione di segnale si svolge infatti con un certo ritardo di tempo
- Oltre ai ritardi di salita e discesa, sia il MASTER che lo SLAVE hanno dei ritardi interni, dovuti alla loro struttura
- In generale i ritardi dipendono dalla specifica tecnologia di BUS adottata

13-04.-03

Informatica II - Livello logico (3)

115

## Tabella riassuntiva dei ritardi

Symbol	Parameter	Min	Max	Unit
$T_{AD}$	Address output delay		11	nsec
$T_{ML}$	Address stable prior to $\overline{MREQ}$	6		nsec
$T_M$	$\overline{MREQ}$ delay from falling edge of $\Phi$ in $T_1$		8	nsec
$T_{RL}$	RD delay from falling edge of $\Phi$ in $T_1$		8	nsec
$T_{DS}$	Data setup time prior to falling edge of $\Phi$	5		nsec
$T_{MH}$	$\overline{MREQ}$ delay from falling edge of $\Phi$ in $T_3$		8	nsec
$T_{RH}$	$\overline{RD}$ delay from falling edge of $\Phi$ in $T_3$		8	nsec
$T_{DH}$	Data hold time from negation of $\overline{RD}$	0		nsec

Ritardi delle varie fasi dell'operazione di lettura in memoria (dipendono dalla tecnologia)

13-04.-03

Informatica II - Livello logico (3)

116

## Frequenze dei BUS sincroni

- Ecco alcune frequenze tipiche di BUS:
  - BUS ISA per PC: 8,33 MHz
  - BUS PCI per PC:
    - 33 MHz
    - 66 MHz
    - e a frequenze più elevate su WorkStation
- È comunque vero che la frequenza di BUS oggi è di solito inferiore alla frequenza del clock di CPU

13-04.-03

Informatica II - Livello logico (3)

117

## BUS asincrono

- Non esiste alcun segnale di clock comune alle unità funzionali collegate al BUS del calcolatore
- Le transizioni di segnale e il passaggio da un ciclo di BUS al ciclo successivo non sono sincronizzati
- Le unità funzionali osservano il BUS di controllo: quando avviene una transizione di segnale significa che si verifica un avanzamento dell'operazione

13-04.-03

Informatica II - Livello logico (3)

118

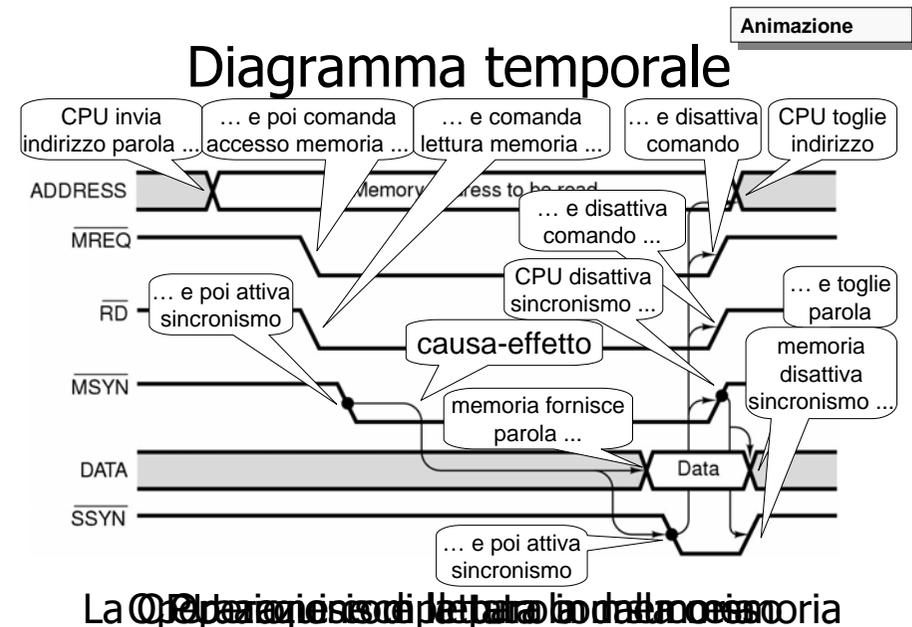
## Esempio

- Letture di una parola dalla memoria
- Bus asincrono
- La memoria presenta un ritardo di lettura pari a 40 ns, cioè le occorrono 40 ns per presentare il dato richiesto alle uscite collegate al BUS dati
- Non ha senso parlare di sottocicli, l'intera operazione avviene come un tutto unico, contenente varie transizioni

13-04.-03

Informatica II - Livello logico (3)

119



13-04.-03

Informatica II - Livello logico (3)

120

## Segnali di controllo

- Oltre ai segnali di controllo già visti per il BUS sincrono (tranne il clock, che qui manca), sono presenti:
- MSYN (Master Synchronisation): il MASTER segnala di avere impostato indirizzo e MREQ, RD; è attivo basso
- SSYN (Slave Synchronisation): lo SLAVE segnala di avere completato l'operazione; è attivo basso

13-04.-03

Informatica II - Livello logico (3)

121

## Fasi dell'operazione di lettura

- Il MASTER (CPU) manda l'indirizzo sul BUS indirizzi, attiva MREQ e RD e, dopo averli tutti stabilizzati, attiva MSYN
- Lo SLAVE (memoria) esegue l'operazione, nel minor tempo possibile, e fornisce la parola sul BUS dati; poi attiva SSYN
- Il MASTER legge la parola dal BUS dati, toglie l'indirizzo e disattiva MREQ e RD; poi disattiva anche MSYN
- Infine, lo SLAVE disattiva SSYN

13-04.-03

Informatica II - Livello logico (3)

122

## Considerazioni

- Poiché nel BUS asincrono non esiste un segnale di clock che marchi gli istanti di tempo in cui i vari segnali di controllo si possono attivare o disattivare, nel diagramma temporale vanno indicati i rapporti causa-effetto esistenti tra i vari segnali di controllo
- I segnali di controllo transitano di valore in reazione a transizioni precedenti

13-04.-03

Informatica II - Livello logico (3)

123

## Procedura full-handshake

- Lo schema asincrono appena visto è incentrato sulla seguente procedura:
  - MSYN viene attivato
  - SSYN viene attivato in risposta a MSYN
  - MSYN viene disattivato in risposta a SSYN
  - SSYN viene disattivato in risposta a MSYN
- La procedura ha nome full-handshake: è indipendente dal clock ed è insensibile alla presenza di SLAVE lenti

13-04.-03

Informatica II - Livello logico (3)

124

## Confronti

- Il BUS sincrono ha il vantaggio di essere semplice da progettare e da controllare
- Il BUS sincrono ha lo svantaggio di portare a sprechi di tempo, poiché ogni operazione si deve svolgere in un numero intero di cicli di clock:
  - quando un'operazione si potrebbe completare in meno di un ciclo di clock
  - quando un'operazione richiede un numero frazionario di cicli di clock

13-04.-03

Informatica II - Livello logico (3)

125

## Confronti

- Il BUS asincrono ha il vantaggio di essere più efficiente nell'uso dei cicli: l'operazione si completa esattamente nel tempo di cui abbisogna
- Il BUS asincrono ha lo svantaggio di essere complesso da progettare e da controllare
- I BUS di calcolatore sono in massima parte di tipo SINCRONO

13-04.-03

Informatica II - Livello logico (3)

126

## Esempi di BUS

- BUS ISA (Industry Standard Architecture): BUS di sistema sincrono; è un'estensione del BUS originale (BUS PC/AT) dei calcolatori di classe PC
- BUS PCI (Peripheral Component Interconnect): BUS di sistema sincrono; successore del BUS ISA per i calcolatori di classe PC
- BUS USB (Universal Serial BUS): BUS di I/O sincrono seriale per periferiche

13-04.-03

Informatica II - Livello logico (3)

127

## Arbitraggio del BUS e cenni alle tecniche di I/O

13-04.-03

Informatica II - Livello logico (3)

128

## Cambio di MASTER

- Normalmente la CPU ha il ruolo di MASTER tra le unità funzionali
- Tuttavia in determinate circostanze anche altre unità funzionali possono assumere temporaneamente il ruolo di MASTER, per scopi particolari:
  - Unità di I/O: possono diventare MASTER per trasferire dati direttamente con la memoria, senza bisogno della CPU (DMA)
  - Coprocessore: può diventare MASTER per prelevare operandi dalla memoria

13-04.-03

Informatica II - Livello logico (3)

129

## Arbitraggio del BUS

- Il BUS del calcolatore può avere, in ogni istante, un unico MASTER
- In caso di cessione del ruolo di MASTER da un'unità funzionale a un'altra, occorre dunque un meccanismo di ARBITRAGGIO DEL BUS, che ne regoli l'utilizzo da parte delle unità funzionali
- Esistono due meccanismi principali di arbitraggio: centralizzato e distribuito

13-04.-03

Informatica II - Livello logico (3)

130

## Arbitraggio centralizzato

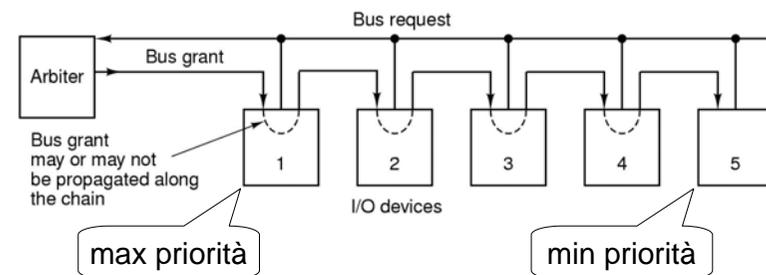
- Il meccanismo di arbitraggio centralizzato prevede:
  - un'apposita unità funzionale, che svolge la funzione di arbitro del BUS
  - alcune linee (appartenenti al BUS di controllo) colleganti l'arbitro del BUS alle unità funzionali potenziali richiedenti il controllo del BUS
- L'arbitro realizza il meccanismo di cessione del controllo del BUS, vale a dire del ruolo di MASTER

13-04.-03

Informatica II - Livello logico (3)

131

## Arbitraggio centralizzato



Meccanismo di arbitraggio centralizzato di tipo semplice, basato sullo scambio di un segnale di conferma (o BUS grant), per regolare i turni di utilizzo del BUS

13-04.-03

Informatica II - Livello logico (3)

132

## Funzionamento dell'arbitraggio

- Quando l'unità di arbitraggio riceve una richiesta di BUS (BUS request), attiva la linea di conferma (BUS grant)
- La conferma viene passata in cascata alle unità potenziali richiedenti:
  - se un'unità non ha una richiesta pendente, passa la conferma all'unità successiva
  - altrimenti, trattiene per sé la conferma, senza passarla all'unità successiva, e prende il controllo del BUS, comportandosi come MASTER

13-04.-03

Informatica II - Livello logico (3)

133

## Considerazioni

- Il meccanismo di arbitraggio centralizzato appena descritto viene chiamato COLLEGAMENTO A FESTONE (Daisy Chaining)
- È un modo cablato (cioè hardware) per assegnare una PRIORITÀ alle unità
- L'unità a contatto con l'arbitro è quella a priorità max, mentre quella all'estremo del festone è quella a priorità min
- Ogni unità è a un livello di priorità diverso

13-04.-03

Informatica II - Livello logico (3)

134

## Osservazioni

- Discutere:
  - inserimento nuova unità;
  - modifica priorità;
  - sensibilità ai guasti.

13-04.-03

Informatica II - Livello logico (3)

135

## Richiesta con priorità

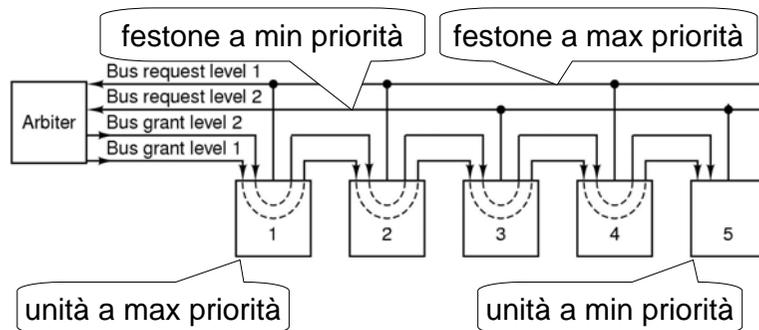
- Il meccanismo di collegamento a festone impone dei livelli di priorità prefissati
- Per introdurre una certa flessibilità nel meccanismo di arbitraggio, si può:
  - dotare il BUS di più festoni di richiesta / conferma
  - stabilire una scala di priorità tra i festoni
  - e collegare ogni unità a uno solo dei festoni
- L'unità usa il festone cui è collegata, e dunque opera al livello di priorità relativo

13-04.-03

Informatica II - Livello logico (3)

136

## Arbitraggio flessibile



Meccanismo di arbitraggio centralizzato flessibile: contiene più festoni di richiesta / conferma, posti a livelli di priorità differenti; ogni unità usa il festone cui è collegata

13-04.-03

Informatica II - Livello logico (3)

137

## Funzionamento dell'arbitraggio

- Se all'arbitro arrivano simultaneamente due richieste di controllo del BUS su due festoni distinti, l'arbitro manda la conferma solo sul festone al livello di priorità superiore
- Le unità collegate allo stesso festone sono ordinate in priorità secondo il meccanismo di collegamento a festone semplice già visto (daisy chaining)

13-04.-03

Informatica II - Livello logico (3)

138

## Funzionamento dell'arbitraggio

- Ecco un esempio di risoluzione dei conflitti di richiesta, in base alle priorità:
  - l'unità 2 ha priorità sull'unità 4: entrambe sono collegate allo stesso festone, ma 2 è più vicina all'arbitro di quanto non sia 4
  - l'unità 4 ha priorità sull'unità 3: 4 è collegata al festone 1, che ha priorità superiore al festone 2, cui è collegata 3
- È anche possibile collegare un'unità a più festoni, lasciandola scegliere il festone più adatto, secondo le situazioni

13-04.-03

Informatica II - Livello logico (3)

139

## Efficienza

- Nei meccanismi di arbitraggio descritti, le unità non sono in grado di sapere quando un'unità che aveva chiesto e ottenuto il controllo del BUS lo rilascia, rendendolo disponibile per altre unità
- Si migliora l'efficienza dell'arbitraggio aggiungendo una terza linea di controllo: il segnale di accettazione (BUS acknowledge); esso risolve il problema appena esposto

13-04.-03

Informatica II - Livello logico (3)

140

## Meccanismo di accettazione

- La linea di accettazione viene attivata dalle unità, non dall'arbitro. Ecco come funziona il meccanismo con accettazione:
  - un'unità richiede e ottiene dall'arbitro il controllo del BUS
  - non appena l'arbitro le invia la conferma, l'unità che prende il controllo del BUS attiva la linea di accettazione e disattiva la linea di richiesta
  - non appena l'arbitro vede attivarsi la linea di accettazione, disattiva la linea di conferma

13-04.-03

Informatica II - Livello logico (3)

141

## Meccanismo di accettazione

- così, subito dopo che l'unità ha preso il controllo del BUS, le linee di richiesta e di conferma sono già libere
- un'altra unità può dunque inviare subito una nuova richiesta, che resterà pendente; questa seconda unità riceverà la conferma non appena la prima unità avrà rilasciato il controllo del BUS, disattivando la linea di accettazione
- Il meccanismo di accettazione è più complesso e costoso, ma è più efficiente

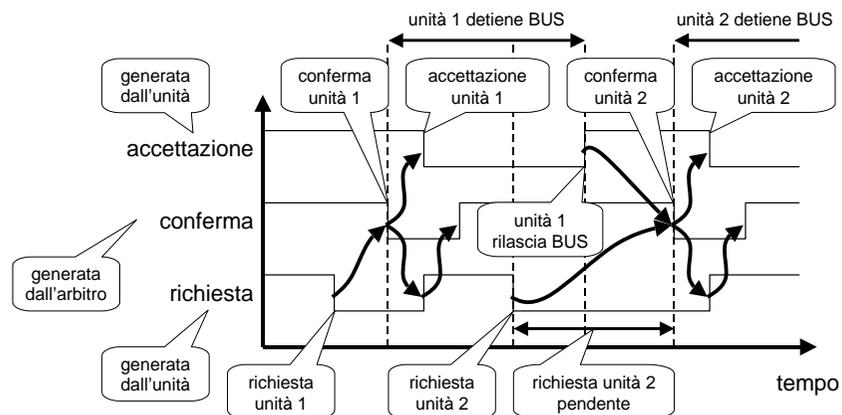
13-04.-03

Informatica II - Livello logico (3)

142

## Diagramma temporale

Animazione



Meccanismo di arbitrato distribuito con accettazione. L'arbitro controlla il BUS, con accettazione la richiesta di un'unità si attiva e si

13-04.-03

Informatica II - Livello logico (3)

143

## Arbitraggio distribuito

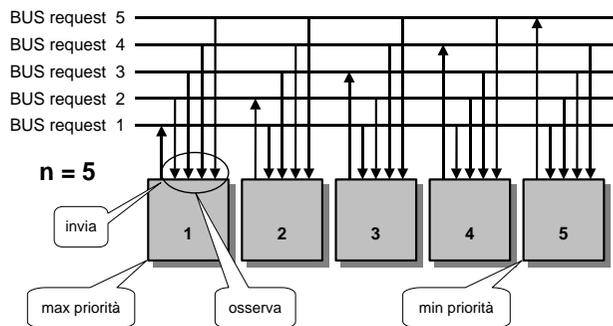
- Il meccanismo di arbitraggio distribuito a n linee prevede:
  - una linea di richiesta BUS per ogni unità
- Non esiste alcun arbitro del BUS
- Le unità hanno priorità diverse e fissate
- Le unità osservano tutte le linee di richiesta BUS
- Ogni unità può attivare solo la propria linea di richiesta BUS

13-04.-03

Informatica II - Livello logico (3)

144

## Arbitraggio distribuito



Ogni unità è dotata di una propria linea di richiesta BUS e osserva lo stato delle linee di richiesta BUS di tutte le altre unità

## Osservazioni

- Discutere:
  - inserimento nuova unità;
  - modifica priorità;
  - sensibilità ai guasti.

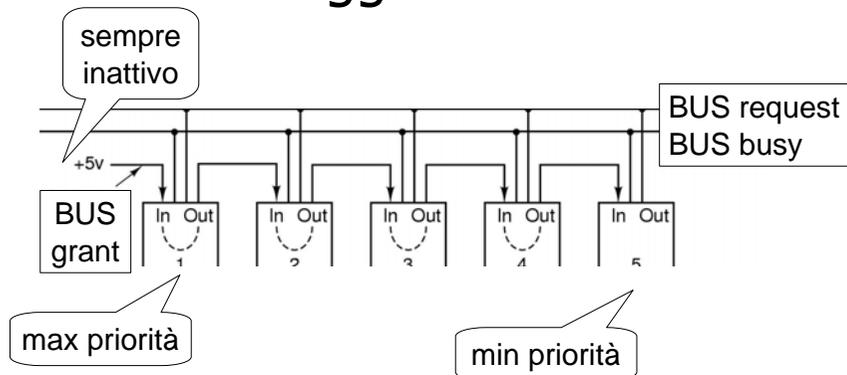
## Funzionamento dell'arbitraggio

- Quando l'unità deve diventare MASTER, si accerta che nessun'altra unità a priorità superiore alla sua abbia attivato la propria linea di richiesta BUS
- Se così è, l'unità attiva la sua linea di richiesta BUS e ottiene il controllo del BUS, diventando MASTER
- Confronto: si risparmia il costo dell'arbitro, ma il numero di linee di controllo cresce con quello delle unità

## Miglioramento

- Si può migliorare l'arbitraggio distribuito a n linee usando solo 3 linee di controllo:
  - richiesta BUS (BUS request): l'unità attiva questa linea per richiedere il BUS
  - impegno BUS (BUS busy): questa linea viene mantenuta attiva dall'unità che detiene correntemente il controllo del BUS
  - conferma BUS (BUS grant): è la linea di conferma, collegata a festone
- È simile al meccanismo di accettazione

## Arbitraggio distribuito



Arbitraggio distribuito con schema di collegamento a 3 sole linee di controllo (tutti i segnali sono attivi bassi)

## Funzionamento dell'arbitraggio

- Presupposto: un'unità che desideri diventare MASTER del BUS attende sempre di vedere:
  - il segnale di impegno BUS inattivo
  - il proprio ingresso IN inattivoprima di avanzare la propria richiesta di controllo del BUS
- In altri termini: l'unità si accerta che un'altra unità a priorità superiore non sia correntemente il MASTER del BUS

## Funzionamento dell'arbitraggio

- Inoltre tutte le unità osservano la regola seguente: se l'ingresso IN è attivo, anche l'uscita OUT va tenuta attiva, indipendentemente dal resto
- In altri termini: se un'unità vede che un'altra unità a priorità superiore detiene il controllo del BUS, passa questa informazione a tutte le unità con priorità inferiore alla propria

## Funzionamento dell'arbitraggio

- Quando si verificano le condizioni che consentono di avanzare una richiesta di controllo del BUS:
  - l'unità attiva la linea di richiesta BUS
  - l'unità attiva la linea di impegno BUS
  - l'unità attiva la propria uscita OUT
- Così l'unità ottiene il controllo del BUS, diventando MASTER, e notifica il fatto a tutte le unità di priorità inferiore alla propria (poi disattiva la richiesta BUS)

## Considerazioni

- Il meccanismo di arbitraggio distribuito a 3 linee di controllo è più semplice di quello a n linee di controllo
- Rispetto al meccanismo di arbitraggio centralizzato con collegamento a festone (daisy chain):
  - si risparmia il costo dell'arbitro
  - e dunque si riduce la probabilità di errori, malfunzionamenti, ...

13-04.-03

Informatica II - Livello logico (3)

153

13-04.-03

Informatica II - Livello logico (3)

154

13-04.-03

Informatica II - Livello logico (3)

155

13-04.-03

Informatica II - Livello logico (3)

156