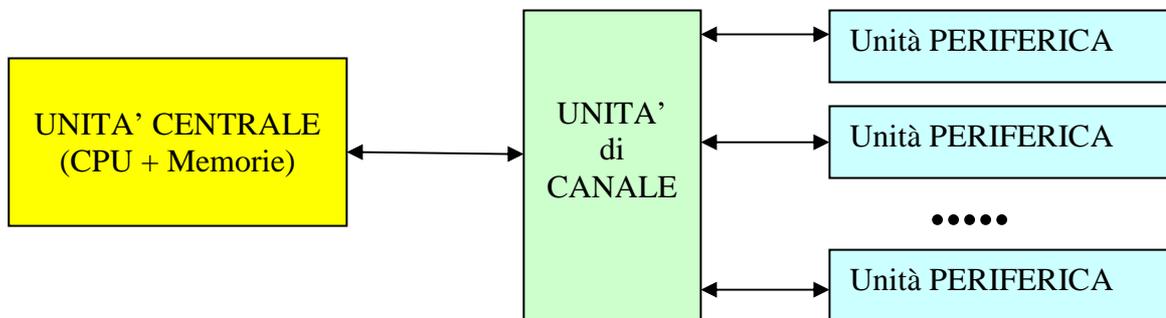


Sistema di Elaborazione

Un calcolatore convenzionale è un sistema di elaborazione nel quale viene eseguita un'istruzione alla volta e in ordine sequenziale ad eccezione di salti indicati dalla stessa istruzione.

Questo sistema si riferisce al modello di **VON NEUMANN**, dal nome del matematico che definì l'architettura nel 1952, che individuò le componenti fondamentali di un elaboratore:

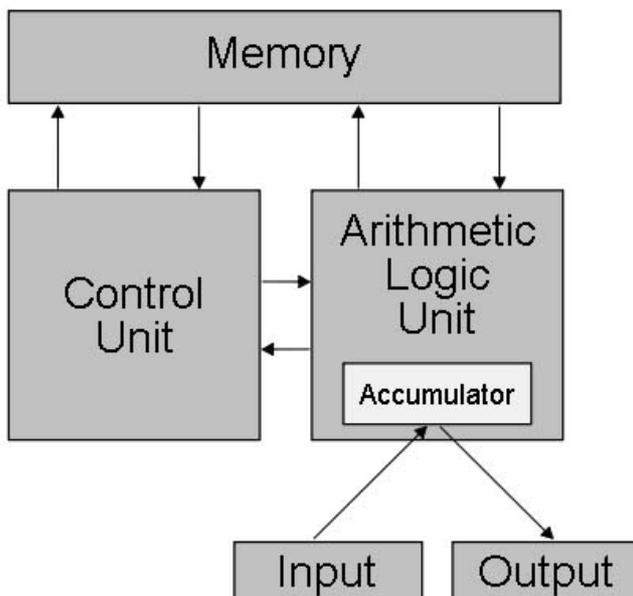
- Una **Unità centrale**, preposta all'elaborazione dei dati ed al controllo del sistema; essa contiene anche il sottosistema delle memorie;
- Delle **Unità Periferiche**, che consentono il colloquio tra il mondo esterno e il calcolatore;
- Delle **Unità di Canale**, che hanno il compito di connettere le periferiche all'unità centrale;



All'interno dell'Unità Centrale sono presenti la Memoria centrale e la CPU (Central Processing Unit o unità centrale di processamento) che a sua volta è costituita da:

- Una **Unità Aritmetico-Logica (ALU)**;
- Una **Unità di Controllo (Control Unit)**;
- Un insieme di **Registri** (memoria locale della CPU).

Dettagliando maggiormente lo schema della precedente figura avremo:



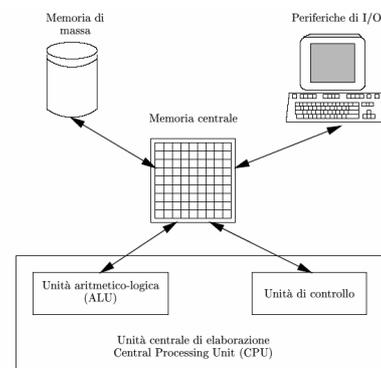
Lo schema si basa quindi su cinque componenti fondamentali:

1. **CPU** o unità di lavoro che si divide a sua volta in
 1. Unità operativa, nella quale uno dei sottosistemi più rilevanti è l'**ALU** (Arithmetic Logic Unit)
 2. Unità di controllo
2. **Unità di memoria**, intesa come memoria di lavoro o memoria principale (**RAM**, Random Access Memory) che ha il compito di memorizzare sia i programmi (istruzioni che li compongono) sia i dati che utilizzano i programmi stessi. Proprietà importanti della memoria sono:
 - Il **tempo di accesso** (ordine dei nano secondi), ossia il tempo che passa tra la richiesta di un dato e la sua disponibilità;
 - La **capacità di memorizzazione**, oggi nell'ordine di qualche Gbyte.
3. **Unità di input**, tramite la quale i dati vengono inseriti nel calcolatore per essere elaborati
4. **Unità di output**, necessaria affinché i dati elaborati possano essere restituiti all'operatore
5. **Bus**, un canale che collega tutti i componenti fra loro

All'interno dell'ALU è presente un registro detto **accumulatore**, che fa da buffer tra input e output grazie a una speciale istruzione che carica una parola dalla memoria all'accumulatore e viceversa.

È importante sottolineare che tale architettura, a differenza di altre, si distingue per la caratteristica di immagazzinare all'interno dell'unità di memoria, sia i dati dei programmi in esecuzione che il codice di questi ultimi.

Altra visione del modello di Von Neumann, vedi figura accanto, in cui si vede che la memoria è l'area di lavoro: nulla può essere eseguito dalla CPU se non è prima caricato in memoria centrale:



STATI DI UN ELABORATORE

Gli stati in cui può trovarsi un elaboratore si possono riassumere in:

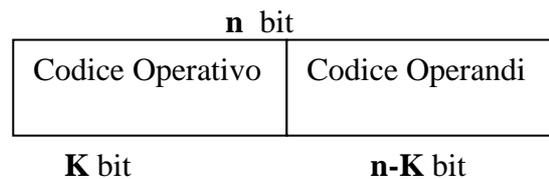
- Stato di attesa;
- Stato di Fetch-Decode: viene prelevata dalla memoria l'istruzione da eseguire e decodificata, cioè per associare le microistruzioni del processore e tutti le azioni da mettere in atto;
- Stato di execute: l'istruzione precedentemente decodificata viene eseguita.

CODIFICA DELLE ISTRUZIONI

Le istruzioni (e anche i dati) all'interno del computer sono codificate in codice binario e possono occupare una o più *parole* (byte o multipli del byte, normalmente 2 byte).

Un'istruzione è quindi logicamente divisa in due parti:

- Il **Codice Operativo**, che definisce l'istruzione da eseguire (ad esempio somma); Se il codice operativo è composto di K bit, allora si possono codificare fino a 2^k istruzioni (set di istruzioni del processore).
- **Codice Operandi**, che fornisce al processore le informazioni (indirizzi in memoria) per recuperare i dati degli operandi coinvolti nell'operazione da effettuare.



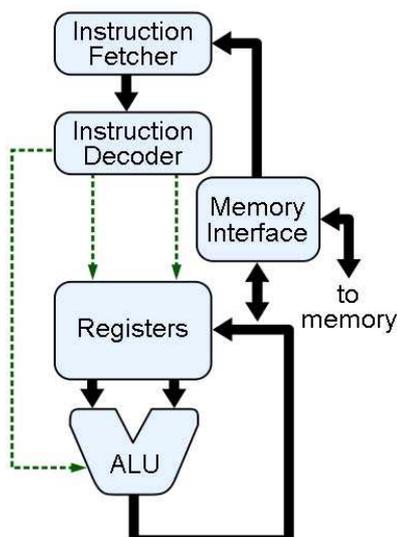
IL PROCESSORE

L'unità centrale di elaborazione, più conosciuta come CPU, anche chiamata processore, ha il compito di eseguire le istruzioni di un programma (che deve essere presente in memoria) e di interagire con le unità di I/O. Durante l'esecuzione del programma, la CPU legge o scrive dati in memoria; il risultato dell'esecuzione dipende dal dato su cui opera e dallo stato interno della CPU stessa, che tiene traccia delle passate operazioni.

Le funzioni tipiche di un processore, per ogni istruzione da eseguire, sono le seguenti:

1. lettura di un'istruzione dalla memoria;
2. Decodifica dell'istruzione;
3. calcolo degli indirizzi di memoria degli operandi;
4. caricamento degli operandi dalla memoria nei registri della cpu;
5. esecuzione dell'istruzione;
6. analisi delle interruzioni, cioè prima di eseguire la prossima istruzione la cpu vede se ci sono eventi (generati dall'utente o da altre unità) prioritari da gestire;
7. calcolo dell'indirizzo di memoria in cui è presente la prossima istruzione da eseguire.

Il processore è l'interprete che esegue le istruzioni in linguaggio macchina e che segue ciclicamente per ogni istruzione secondo gli stati detti in precedenza:



- **Acquisizione dell'istruzione (Instruction Fetch):** il processore preleva l'istruzione dalla memoria, presente nell'indirizzo specificato da un registro "speciale" (Program Counter o PC);
- **Decodifica:** una volta che la word è stata prelevata, viene determinata quale operazione debba essere eseguita e come ottenere gli operandi, in base al codice operativo che ne determina il microprogramma (insieme di microistruzioni) da eseguire;
- **Esecuzione (Execute):** viene eseguita la computazione desiderata. Nell'ultimo passo

dell'esecuzione viene incrementato il PC: tipicamente di uno se l'istruzione non era un salto condizionale, altrimenti l'incremento dipende dall'istruzione e dall'esito di questa.

Questo ciclo elementare può essere migliorato in vari modi nel caso di processori scomposti in più unità, ottenendo un parallelismo nell'elaborazione delle istruzioni di uno stesso programma:

- per esempio, la decodifica di una istruzione può essere fatta contemporaneamente all'esecuzione della precedente e alla lettura dalla memoria della prossima (*instruction prefetch*) e lo stesso può essere fatto con i dati che si prevede saranno necessari alle istruzioni (*data prefetch*).
- La stessa esecuzione delle istruzioni può essere suddivisa in passi più semplici, da eseguire in stadi successivi, organizzando la unità di controllo e la ALU in stadi consecutivi, come delle catene di montaggio (*pipeline*): in questo modo più istruzioni possono essere eseguite "quasi contemporaneamente", ciascuna occupando ad un certo istante uno stadio diverso della pipeline.

UNITA' DI INPUT E OUTPUT

Le **unità di I/O** si possono classificare in tre principali categorie :

1. Unità di **interazione** che permettono all'utente di interagire con la macchina: tastiera, mouse, scanner, monitor, webcam, microfoni, cuffie, ecc...
2. unità di **memorizzazione** che permettono la memorizzazione permanente delle informazioni (memoria di massa): Hard Disk, CDROM, DVD, Nastri, ecc...
3. Unità di **Comunicazione** che permettono di collegare più computer per la realizzazione di una rete di elaboratori: modem, Schede di rete LAN, schede ISDN.

Le unità I/O si occupano di gestire il colloquio tra Unità Centrale (CPU+Memoria) e le periferiche. Queste unità possono essere sofisticate al punto da comprendere processori dedicati (**Canali I/O**) in modo da eseguire direttamente le istruzioni di I/O sollevando la CPU da tale compito. Operazioni tipiche eseguite da tali canali di I/O sono:

- Lettura/scrittura di caratteri, stringhe, record;
- Interfacciamento con l'unità centrale;
- Ecc...

La comunicazione tra il sistema operativo e la periferica è mediata da una componente software chiamata driver.

Comunicazione tra memoria centrale e periferiche

Viene realizzata in uno dei seguenti modi:

1. (*soluzione software*) a **controllo di programma**: è il processore che trasferisce ogni dato, uno alla volta. Si ha massima flessibilità ma poca efficienza in quanto la CPU "perde tempo" non potendo eseguire invece altre istruzioni;
2. realizzazione in Hardware tramite un **accesso diretto alla memoria (Direct Memory Access, DMA)**; il processore è sollevato dai trasferimenti di dati in quanto se ne occuperebbe l'unità di controllo **DMAC (Direct Memory Access Controller)**. Questa soluzione è quindi più veloce della precedente proprio perché la gestione dell'I/O è affidata al DMAC: la CPU si limita a dare avvio al trasferimento rilasciando il bus dati, mentre il trasferimento vero e proprio è svolto dal DMAC.

Il DMA è usato da molti sistemi hardware come controller di unità a disco (esempio ATA133, Serial ATA o SCSI), schede grafiche e schede audio.

3. con il **Bus Mastering** in cui gli stessi controller delle periferiche gestiscono la comunicazione (come vedremo i chipset per i Bus PCI sono di questo tipo).

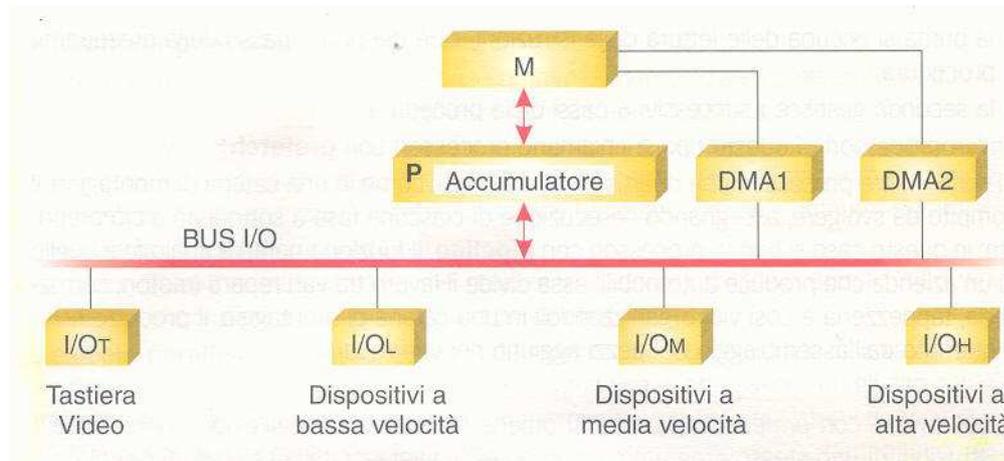


Figura 1: comunicazione tra unità di I/O, DMA e unità centrale tramite BUS I/O

Come si può vedere in figura le informazioni tra unità centrale (M,P) e periferiche passano per una via comune, il Bus di I/O. Quindi il **bus mastering** è una funzionalità supportata da alcune architetture bus che permette ad un controller collegato al bus di comunicare direttamente con altri dispositivi sul bus senza passare attraverso il processore centrale. La maggior parte delle moderne architetture bus supportano il bus mastering, ottenendo in questo modo notevoli miglioramenti prestazionali.

NB:

Per alcune funzioni, oltre alle soluzioni software e hardware esiste un modo intermedio detto **firmware**: il funzionamento dell'unità di I/O dipende da un programma memorizzato in una memoria veloce. Ad esempio il programma di controllo del personal computer, il **BIOS** (Basic Input Output System), è realizzato con soluzione firmware e risiede nella ROM-BIOS, usando i dati della memoria **CMOS** (Complementary Metal-Oxide Semiconductor) la quale contiene i parametri di configurazione del sistema. Perché è flessibile? Se ad esempio la scheda madre del mio pc non è in grado di gestire il mio nuovo Hard Disk da 400GB, basta aggiornare il BIOS.

OSSERVAZIONI

Il firmware è un programma, inteso come sequenza di istruzioni, integrato direttamente in un componente elettronico nel senso più vasto del termine. Il termine deriva dall'unione di "firm" (stabile) e "ware" (componente), indica che il programma non è immediatamente modificabile dall'utente finale, ovvero risiede stabilmente nell'hardware integrato in esso, e che si tratta del punto di incontro fra componenti logiche e fisiche, ossia fra hardware e software.

Quando si parla di firmware per una scheda elettronica (come una scheda di espansione per computer) questo generalmente trova posto all'interno di una memoria ROM o flash. Quando invece il firmware è integrato all'interno di un processore (come ad esempio il Pentium 4) in italiano viene detto anche microcodice.

Il firmware forse più conosciuto in ambito informatico è il BIOS, responsabile del corretto avvio del computer, ma quasi sempre sono dotati di proprio firmware anche i singoli

componenti di un computer, come dischi fissi, lettori o masterizzatori di CD e DVD, schede di espansione in genere.

LE MEMORIE

La memoria è un dispositivo in grado di immagazzinare e restituire la informazioni provenienti da altre unità.

Le memorie possono essere classificate in base al:

1. Tipo di accesso:

- a. Memorie ad **accesso casuale**, in cui il tempo per raggiungere un dato (di cui si conosce l'indirizzo in memoria) è indipendente dalla posizione occupata dal dato all'interno della memoria.
- b. Ad **accesso sequenziale**, le informazioni sono ritrovate nello stesso ordine in cui sono state memorizzate (es. nastro magnetico).
- c. Ad **accesso diretto**, nelle quali è possibile selezionare in modo casuale l'area (settore, cluster) in cui è contenuta l'informazione e poi al suo interno una ricerca sequenziale.

2. Funzione Svolta:

- a. **Registri**, interni alla CPU, con accesso casuale e dimensioni piuttosto ridotte;
- b. **Memoria Centrale** (RAM e ROM): di dimensione più ampia e ad accesso casuale;
- c. **Memoria di Massa** (Dischi magnetici, ottici, pen drive): con accesso sequenziale o diretto, per la conservazione permanente dei dati.

3. Volatilità:

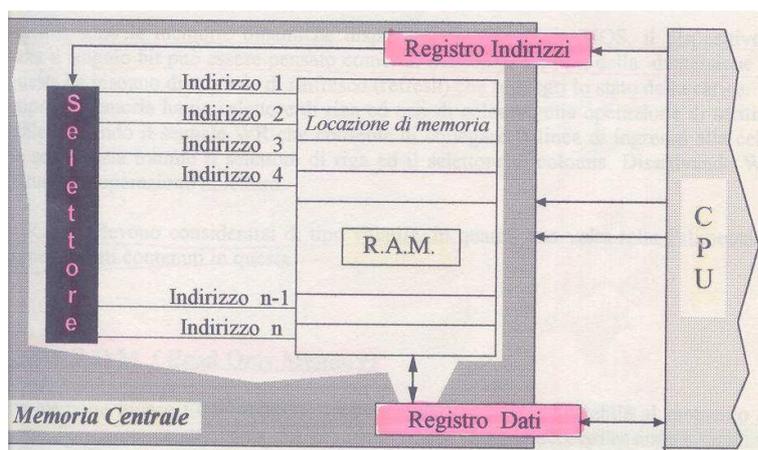
- a. **Memorie volatili**: in grado di conservare i dati per un tempo limitato (RAM);
- b. **Memorie permanenti**: in grado di conservare i dati anche dopo lo spegnimento del computer (massa e ROM).

RAM (Random Access Memory – memoria ad accesso casuale)

La RAM è utilizzata dalla CPU per memorizzare temporaneamente programmi, dati di input e risultati dell'elaborazione della CPU.

È costituita da celle (o locazioni) ognuna individuata con un indirizzo. Il contenuto di ogni cella può essere letto o scritto in un tempo detto tempo di accesso. Fisicamente è realizzata tramite semiconduttore in forma integrata.

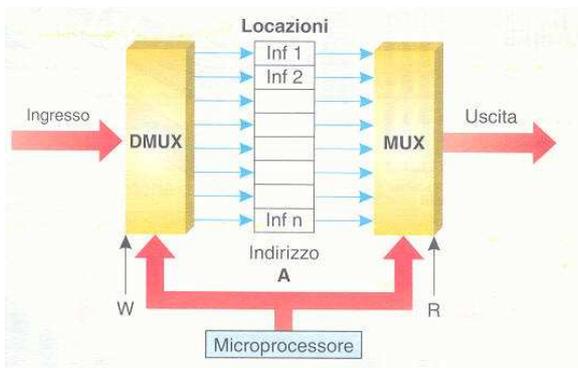
Il numero delle celle è molto elevato per cui serve un selettore il quale, ricevendo dalla cpu l'indirizzo, sia in grado di selezionare la linea della cella selezionata.



Al selettore l'indirizzo arriva dal Registro Indirizzi della CPU. Una volta che il selettore seleziona la cella:

- se l'operazione è di output, il dato viene trasferito dal Registro Dati verso la memoria (scrittura);
- se invece è di input, dalla memoria verso il Registro

Dati (lettura) e quindi verso la CPU. Il verso dipende dal segnale che arriva attraverso il control bus.



In effetti il selettore è costituito da un Multiplexer (MUX), che in caso di lettura seleziona la cella e copia il dato dalla memoria al registro dati, e da un demultiplexer (DMUX) per l'operazione inversa.

Figura 2: accesso in lettura o in scrittura in qualche cella di memoria

Tipologie di RAM

Da un punto di vista costruttivo le RAM si distinguono in:

1. **S.R.A.M** (Static RAM): sono dei flip flop che mantengono il dato finché non viene riscritto. Sono più costose rispetto alle D.R.A.M.
2. **D.R.A.M.** (Dynamic RAM): realizzate in tecnologia C-MOS, ossia micro condensatori, nei quali 1 corrisponde al condensatore carico e 0 a quello scarico. È il tipo di memoria usato nella pratica per due motivi:
 - a. Elevata capacità di integrazione
 - b. Costi più bassi

Problema: tende a disperdere la carica elettrica (come tutti i condensatori) => occorre un ciclo di refresh che reintegri la carica.

Negli attuali PC si utilizzano le **DDR2 SDRAM** (Double Data Rate Synchronous Dynamic RAM):

- sincronizzate con il bus di sistema con frequenze reali di 400 Mhz e 533 Mhz che trasferiscono rispettivamente a 800 Mhz e 1066 Mhz, in quanto i dati sono trasferiti sia sul fronte di salita che di discesa del clock => come se avessero il doppio della frequenza reale del bus di sistema.
- I chip sono identificati con la sigla **DDR2-XXX**, dove:
 - XXX indica frequenza di trasferimento (doppia rispetto alla reale).
 - Esempio, DDR2-800 lavora con frequenza a 400 Mhz (ma lavora come se fosse a 800 Mhz)
- Le schede di memoria, chiamati moduli, si identificano con la sigla **PC2-XXX**:



- XXX, indica la banda in MBps.
- Esempio, PC2-8500 indica una banda di 8,5 GBps
- I moduli attualmente usati sono le DIMM (Dual In-line Memory Module) a 168 pin. Precedentemente le SIMM (Serial In-line Memory Module) a 72 pin.

INSERIRE IMMAGINE LIBRO PAG 210

MEMORIE ROM (Read Only Memory)

Memoria permanente ad accesso casuale destinata a ospitare programmi che il calcolatore deve eseguire senza l'intervento dell'utente, come il BIOS (Basic Input Output System) del sistema di elaborazione.

Le ROM si dividono in:

- **PROM** (Programmable ROM): le prime ROM venivano programmate solo una volta dal costruttore. Le PROM permetteva all'utente di programmare autonomamente ma una sola volta.
- **EPROM** (Erasable PROM): è possibile rimuovere la programmazione esponendo i chip di memoria a radiazione ultravioletta per 15-20 min attraverso una finestrella sul dorso. Questa riprogrammazione poteva avvenire:
 - Un numero limitato di volte;
 - Off-line: occorre togliere il chip dalla scheda madre;
 - Tutto-niente: non consente di cancellare selettivamente le celle.
- **EEPROM** (Electrically Erasable PROM): riprogrammabili, ma con netti vantaggi rispetto alla EPROM:
 - Tempi di cancellazione bassi: millisecondi contro i 15-20 minuti della EPROM;
 - On-line: non occorre togliere il chip
 - Tensioni basse
 - Si possono cancellare anche singole celle: in questo caso si chiamano **EAPROM** (Electrically Alterable PROM).



Contenuto della ROM

Il nome di **Basic Input/Output System** deriva dal fatto che la sua funzione principale, in origine, era quella di intermediario tra l'hardware della macchina e il sistema operativo, che a sua volta interfaccia con il software applicativo.

In pratica, il BIOS è nato come un insieme di device driver, moduli software che pilotano i dispositivi hardware.

Agli inizi, quando il PC era quello di IBM, il software del BIOS conteneva tutti i device driver per l'intero sistema, ed era registrato in uno o più chip di una **ROM** (Read Only Memory) non volatile (manteneva i contenuti anche dopo lo spegnimento del computer) montata sulla scheda madre. Questa ROM, oltre ai device driver, conteneva dei programmi che esistono tuttora:

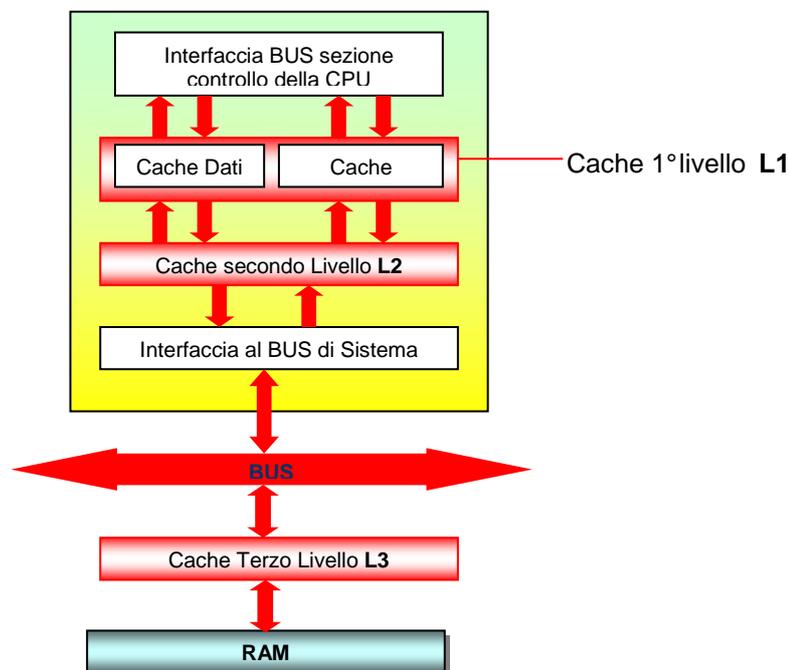
- **POST (Power On Self Test):** Istruzioni per il controllo del funzionamento elettronico dei dispositivi all'accensione del PC
- **SETUP del BIOS:** Istruzioni per la configurazione del sistema (subito dopo il POST);
- **BIOS:** parametri per la configurazione dei dispositivi verso il Sistema Operativo. I parametri del BIOS sono conservati in una speciale memoria chiamata **CMOS Memory** (che conserva la configurazione del computer determinata dai componenti installati e dalle personalizzazioni che l'utente ha apportato tramite il programma di setup) la quale è dotata di batteria tampone per conservare le informazioni anche a pc spento. Una volta caricato in memoria, il DOS dei primi tempi utilizzava le routine di basso livello del BIOS (i device driver) per interagire con l'hardware: tastiera, adattatore video, porte seriali e parallela, floppy e hard disk, joystick e orologio. Gli unici driver erano, insomma, quelli del BIOS.
- **BOOT LOADER:** il programma che avvia la sequenza di caricamento del sistema operativo partendo dai settori iniziali di un disco (DOS, OS/2, WINDOWS).

MEMORIA CACHE

È una memoria veloce in cui vengono memorizzate le informazioni più frequentemente usate. Se il processore non trova l'informazione in cache la ricerca nella RAM e quindi su disco.

Nei PC si hanno tre livelli di cache:

- Cache di livello 1 (**L1**) e di livello 2 (**L2**) montate dentro il microprocessore;
- Cache di livello 3 (**L3**) montata sulla scheda madre.



Perché la CACHE

Il processore lavora a frequenze f molto elevate (qualche GHz) mentre le altre parti del computer lavorano a frequenze più basse dettate dal BUS Interface Unit (BIU), ossia a frequenze del tipo: 133, 266, 533 e 1066 MHz.

Il processore accede direttamente alla Cache con la sua frequenza f , mentre quando deve prelevare dalla RAM deve accedere al BUS e quindi rallentare.

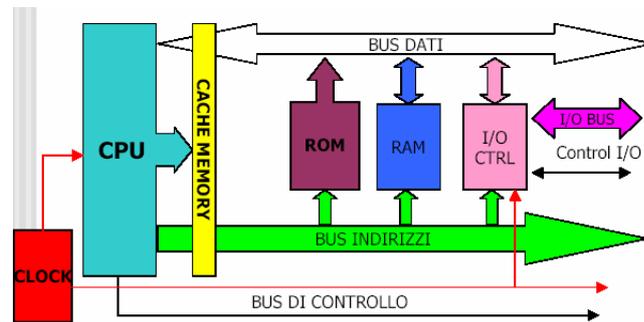


Figura 3: la CPU accede direttamente alla Cache alla RAM tramite Bus

Come lavora la CACHE

Se il processore deve leggere un dato:

- Se il dato si trova in cache di primo livello L1 => il processore lo legge direttamente;
- Se non si trova in L1 viene cercato in L2; se è qui => letto direttamente e copiato in L1;
- Se non si trova nemmeno in L2 viene cercato in L3; se è qui => letto direttamente e copiato in L2;
- Se non è in cache lo cerca in RAM; se lo trova qui => copia il blocco (contenente il nostro dato) dalla RAM in tutti e tre le cache;

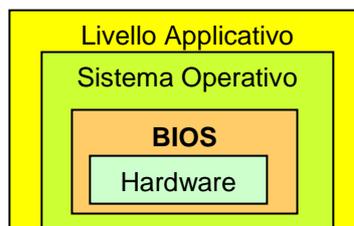
IL BIOS (Basic Input Output System)

Se si esegue un programma al computer esso utilizza le routine del sistema operativo il quale utilizza il BIOS per interfacciarsi direttamente con l'hardware. In questo modo si garantisce la compatibilità verso il basso sia Hardware che software.

Il BIOS personalizza l'Hardware e ne nasconde le caratteristiche fisiche al sistema operativo il quale crea delle macchine virtuali per le applicazioni.

Nella CMOS sono memorizzati i dati relativi alle periferiche Plug&Play e all'Hardware installato. I sistemi Operativi accedono a questi dati per verificare lo stato dei componenti e la presenza di nuovo Hardware (POST).

È possibile aggiornare il BIOS così da utilizzare le prestazioni di nuovi dispositivi hardware, scaricandolo dal sito del produttore.



SHADOW RAM

Il **BIOS** (Basic Input Output System) è un insieme di routine, a livello firmware, che operano da interfaccia tra Hardware e Sistema Operativo per permettere il trasferimento di informazioni tra le diverse unità del sistema.

Il BIOS è residente in EEPROM, ma per migliorare le prestazioni del sistema esso viene copiato, totalmente o in parte, in RAM (che è più veloce rispetto alla EEPROM): questa zona di RAM è chiamata Shadows RAM (ombra).

BUS

I Bus sono le linee di comunicazione che permettono il trasferimento di dati o di controllo tra le varie parti del sistema di elaborazione.

Essi sono composti da più linee così da permettere una trasmissione parallela.

I BUS si distinguono in:

- **BUS DATI** (Data Bus): linee bidirezionali per il trasporto dei dati tra le varie unità. Il bus dati può avere 8, 16, 32, 64, 80 linee, ossia il numero di bit che possono essere trasferiti parallelamente, ossia la massima dimensione di trasferimento di un dato. Questo numero di bit del bus dati rappresenta il **parallelismo** del microprocessore.
- **BUS INDIRIZZI** (Address Bus): linee unidirezionali tramite le quali il processore invia gli indirizzi per individuare la cella di memoria o l'unità di I/O con cui il processore vuole comunicare (trasferimento di dati).
- **BUS DI CONTROLLO** (Control Bus): linee bidirezionali riservate alla trasmissione dei segnali di controllo. Due linee molto importanti sono:
 - **INTR** (Interrupt Request): richiesta di interruzione delle attività del microprocessore da parte di qualche unità I/O,
 - **INTA** (Interrupt Acknowledge): riconoscimento dell'interruzione da parte del processore.

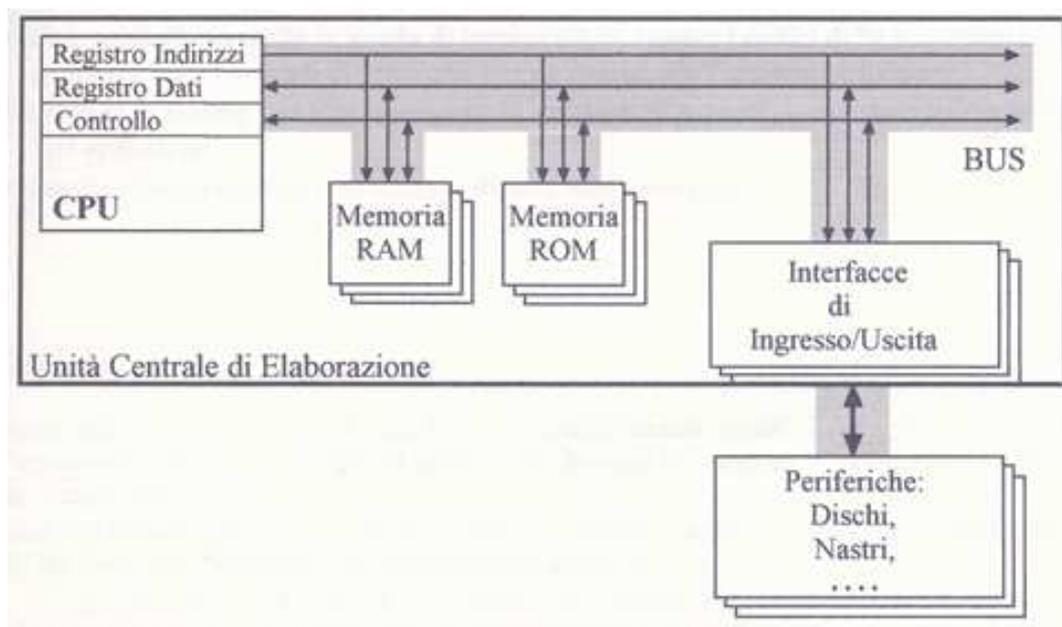
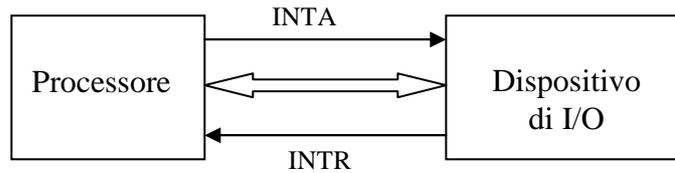


Figura 4: connessione dei componenti del pc tramite vari BUS

Il protocollo di comunicazione tra processore e dispositivo segue questi passi:

1. il dispositivo, quando vuole trasmettere/ricevere un dato al/dal microprocessore, attiva la linea INTR (*ad esempio durante la lezione, qualcuno bussata alla porta*);
2. il microprocessore attende di terminare l'istruzione corrente (*il prof. Conclude il concetto che sta spiegando*);
3. il microprocessore attiva la linea INTA (*il prof. Dice "avanti"*);
4. la periferica si identifica trasmettendo il proprio codice di tipo (*chi ha bussato alla porta si presenta e fa la sua richiesta*);
5. la periferica trasmette/riceve il dato;
6. il microprocessore continua l'esecuzione del programma interrotto eseguendo la prossima istruzione utile (*il prof. continua con la lezione*).



BUS dedicati all'I/O

Ci sono sistemi che utilizzano due distinti gruppi di BUS:

- bus dati, controllo e indirizzi dedicati esclusivamente alle memorie;
- bus dati, controllo e indirizzi dedicati esclusivamente alle Unità di I/O;

Quindi abbiamo uno spazio di indirizzamento della memoria distinto da quello delle periferiche e quindi si possono avere contemporaneamente operazioni di I/O e in memoria. Purtroppo sono complicati da realizzare.

BUS con distinti indirizzi di memoria e di I/O

Nei sistemi con processori Intel, memorie e unità di I/O condividono i bus. Siccome in un istante il processore può comunicare o con la memoria o con una periferica, per capire se un indirizzo presente sul bus indirizzi si riferisce o a una cella o a una periferica si utilizza una linea di controllo chiamata **Mem/IO**. Se Mem/IO è alta allora l'indirizzo si riferisce a una periferica e il contenuto del bus dati sarà scambiato tra processore e periferica, se invece è bassa allora tra processore e memoria.

Evoluzione dei Bus di Dati Standard

Per espandere un pc si possono aggiungere schede utilizzando gli Slot di espansione del sistema. Essi fanno parte dei BUS e su di essi sono inserite le schede.

Nel tempo questi Bus standard si sono evoluti in quanto la velocità di trasferimento dalla scheda al sistema dipende dalla frequenza di funzionamento dello stesso BUS.

- **ISA** (Industry Standard Architecture), presente nelle schede madri fino al Pentium II:
 - bus a 8 o 16 bit (trasportava un byte alla volta);
 - lavorava ad una frequenza di 8,33 MHz.
- **EISA** (Enhanced ISA):
 - Fu una evoluzione dell'ISA
 - bus a 32 bit e frequenza a 8,33 Mhz.
 - Può accogliere anche una scheda ISA a 16 bit (userà metà connettore).
- **VESA** (Video Electronics Standard Association): usato nelle schede con processore Intel 80486, abbandonato con i Pentium:
 - Inizio delle GUI (Windows) => ISA collo di bottiglia => serve maggiore banda => creato nuovo BUS per incrementare solo la velocità di trasferimento tra video e processore;
 - Parallelismo a 32 bit;
 - Frequenza 33,3 MHz;
 - Larghezza di banda = 127,02 MBps (mega Byte per secondo);
 - Problema: non supporta il Plug&Play.

OSS: Calcolo larghezza Banda = Frequenza * n°bit / 8 = 33,3 * 10 ⁶ * 32 / 8 =
--

$133,2 * 10^6$ byte. Per avere la velocità in MBps devo dividere per 1 mega (2^{20})
 $\Rightarrow 133,2 * 10^6 / 2^{20} = 127,02$ MBps.

- **PCI** (più avanti);
- **AGP** (più avanti);
- **PCI-express** (più avanti);

Il CHIPSET della scheda madre

La scheda madre è l'infrastruttura che permette il collegamento delle varie parti del computer.

Il Chipset:

- si trova sulla scheda madre.
- fissa le caratteristiche del sistema: quale tipo di processore, memorie, ecc..., possono essere montate sulla scheda madre.
- Offre supporto al processore, alla memoria e permette la comunicazione tra i vari componenti tramite BUS. Cioè gestisce materialmente il bus.

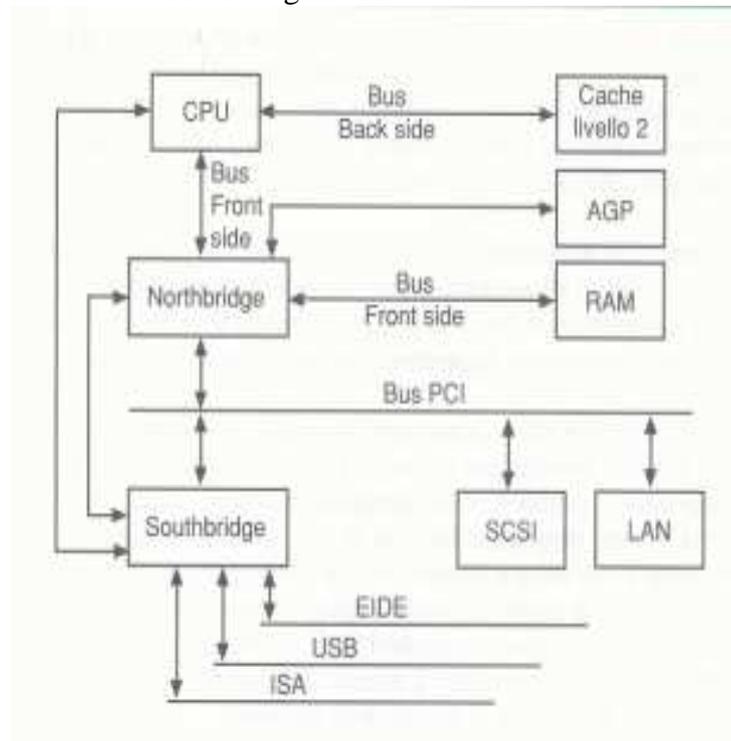


Figura 5: struttura del chipset

- è formato da due processori che coordinano lo scambio di informazioni tra le varie parti del pc:
 - **North Bridge:**
 - è collegato al microProcessore tramite bus veloce (Front Side o Bus di Sistema);
 - coordina quindi lo scambio di dati tra CPU, memoria e Interfaccia Grafica (a volte ingrata nel chipset stesso);
 - nei processori Intel si chiama MCH (Memory controller Hub).
 - **South Bridge:**
 - serve per controllare le Periferiche

- a esso sono collegati i canali EIDE (dischi), i bus USB, ISA e un Bus Dedicato per tastiera, mouse, porte parallele e seriali, ecc...
- nei processori Intel si chiama ICH (I/O controller Hub).

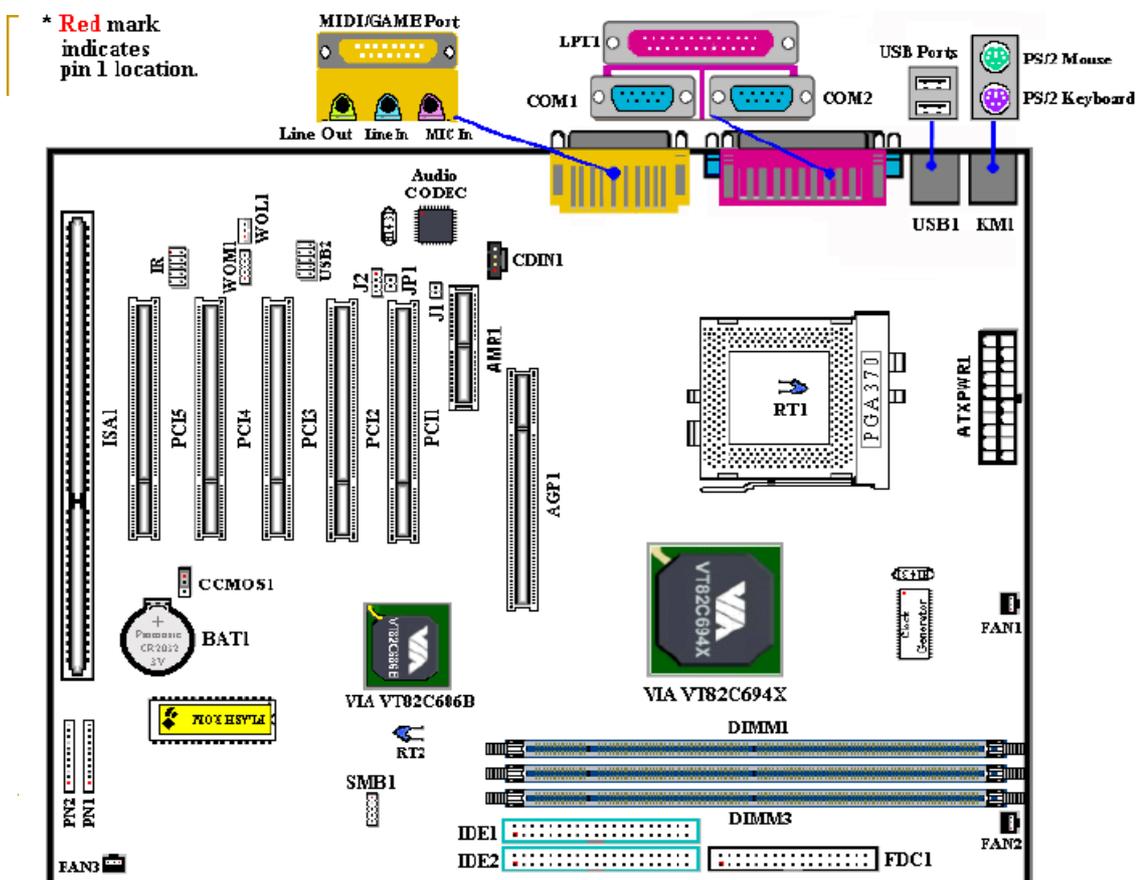


Figura 6: scheda madre. Slot PCI, ISA, AGP, Chipset, Banchi di memoria, ROM

Il Plug and Play

A partire da Windows 95, grazie alla standardizzazione delle schede, è stato introdotto il Plug&Play (Connetti e fai funzionare), ossia la possibilità di inserire una nuova scheda che viene automaticamente riconosciuta dal sistema operativo, e quindi immediatamente funzionante (ovviamente dopo aver eventualmente installato i suoi driver). Prima del Plug&Play occorre configurare manualmente le schede con forti rischi di conflitti.

Una volta inserita la scheda in uno slot libero il sistema operativo assegna automaticamente i **parametri** che gli permetteranno di identificare successivamente la scheda:

- indirizzo di interfaccia (**porto**), cioè come raggiungere la scheda (abbiamo detto che nel bus indirizzi ci può stare un indirizzo che individua una cella di memoria o un dispositivo di I/O).
- numero di **interrupt**, cioè la linea che deve usare la periferica per chiedere alla CPU di poter trasmettere/ricevere. Ad esempio il centralino di un ufficio in cui ogni impiegato ha un numero di telefono interno: il numero rappresenta l'interrupt, mentre il porto rappresenta la posizione fisica nell'ufficio.
- Modalità di comunicazione: tramite DMA o processore. Nel caso del DMA occorre specificare anche il canale di I/O, visto che collegati con la memoria ci possono essere più canali.

BUS PCI

PCI (Peripheral Component Interconnect), sviluppato da Intel, è il Bus:

- utilizzato per Collegare il chip NorthBridge con quello SouthBridge.
- utilizzato per la connessione (tramite SLOT) a schede PCI (di rete, audio, ecc...)
- Inizialmente a 32 bit e frequenza di lavoro a 33 MHz.
- Supporta il **Bus Mastering**: il controllo del Bus (per il trasferimento di I/O) è gestito dal Chipset e non più dalla CPU.
- Arbitraggio del Bus centralizzato.

Il bus PCI ha avuto nel tempo una evoluzione:

Frequenza sul Bus PCI	Parallelismo a 32 bit	Parallelismo a 64 bit
33 MHz PCI	133 MBps	266 MBps
66 MHz PCI	266 MBps	532 MBps
100 MHz PCI-X	-	800 MBps
133 MHz PCI-X	-	1 GBps

LIMITI:

1. non si possono avere altri aumenti di banda in quanto alle alte frequenze, l'elevato parallelismo provoca problemi di cross talk (interferenze).
2. problemi di scalabilità: con il bus condiviso le schede devono stare sempre in ascolto, quindi massimo 5 schede.

Funzionamento del BUS PCI

L'arbitraggio del bus è centralizzato ed è gestito dall'Host, un sistema che interconnette il bus PCI al chipset:

- ogni interfaccia che vuole comunicare chiede all'Host il controllo del bus attivando il segnale Bus Request;
- quando l'host concede il controllo del bus alla periferica attiva il segnale Bus Grant (concesso).
- La comunicazione è Half Duplex, cioè la trasmissione nei due versi non può avvenire contemporaneamente.
- Non è in grado di fornire QoS (Quality of Service), ossia garanzia di banda per una certa scheda in un dato momento, in quanto l'arbitraggio centralizzato non permette di definire livelli di priorità nel controllo del bus. Quindi il Bus PCI è poco adatto per applicazioni real time, audio-video streaming e audio-video editing.

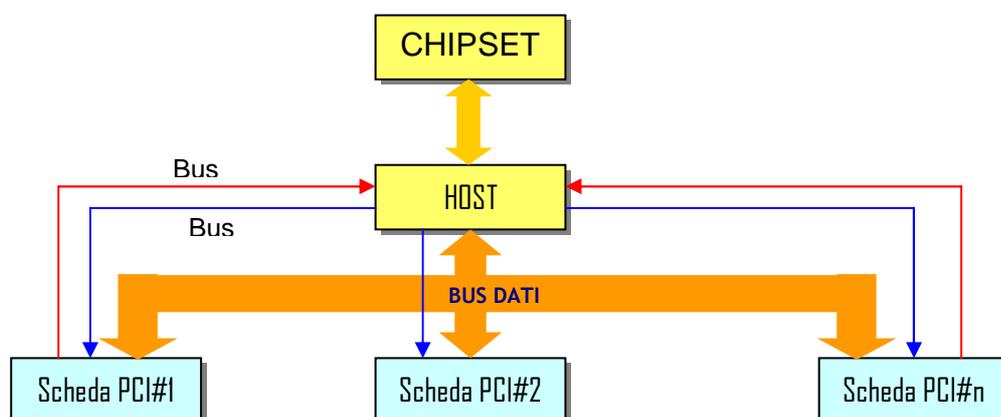


Figura 7: gestione del BUS PCI tramite Host

Quando un dispositivo ottiene il controllo del bus diventa Bus Master, cioè può dialogare con la CPU o la Memoria attraverso il chipset. Per questo motivo si sta affermando il PCI-Express in sostituzione del PCI.

Porta AGP (Advanced Graphics Port)

Con l'utilizzo di applicazioni audio-video e grafiche in 3D era inutile sfornare semplicemente processori più potenti in quanto il collo di bottiglia era nel bus tra scheda video e processore. Da qui la creazione del porto AGP: non si chiama bus in quanto collega solo due unità, video e processore.

Evoluzione dell'AGP:

BUS AGP	Massima Velocità di Trasferimento
AGP 1x	266 MBps
AGP 2x	533 MBps
AGP 3x	1,066 GBps
AGP 4x	2,133 GBps

PCI-Express (PCI-e)

È uno **standard aperto** (specifiche di pubblico dominio) sviluppato da Intel che definisce le caratteristiche elettriche, meccaniche, la topologia e i protocolli per il **collegamento punto-punto**, in fibra o in rame, tra due dispositivi.

Rispetto a PCI introduce alcune importanti vantaggi:

1. velocità di trasferimento tra 5 Gbps e 80 Gbps;
2. trasferimento seriale, con eliminazione dei problemi di cross talk dovuti al parallelismo;
3. collegamento punto-punto con comunicazione full-duplex;
4. scalabilità: posso aggregare sia un certo numero di canali (**lane**) sia di aumentare la banda. Le attuali schede madri hanno in dotazione 20 lane (che crescerà) aggregabili a piacere; normalmente sono così suddivisi: 16 lane per la scheda grafica collegata al north bridge e 4 collegati in combinazioni di PCI-e X1, X2 (oppure X4) collegati al South Bridge.
5. introduce il QoS tramite schedulazione per livelli di priorità: viene così garantita la banda per una certa scheda in un certo istante, per le applicazioni isocrone.
6. si possono inserire e togliere schede a caldo (Hot Swap).
7. è PCI compatibile: posso usare questi slot per inserire schede PCI.

Nella seguente figura vediamo un possibile schema di una scheda madre con bus PCI-Express. Come si può vedere c'è un raggruppamento di 16 lane per la scheda video e 4 per altre schede PCI-e (due X1 e uno X2).

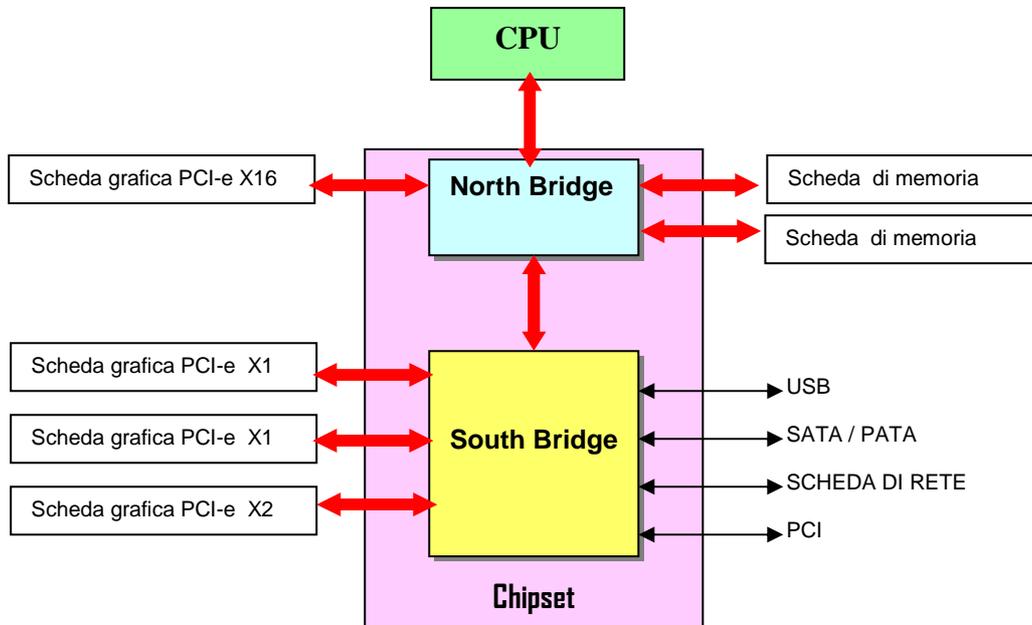


Figura 8: esempio di schema con PCI-e

Le possibili combinazioni del PCI-Express sono qui elencate:

Tipologia di Bus PCI-Express	Frequenza	Max Velocità Trasferim. x Direzione
PCI-e X1	5 GHz	250 MBps
PCI-e X2	10 GHz	500 MBps
PCI-e X4	20 GHz	1 GBps
PCI-e X8	40 GHz	2 GBps
PCI-e X16	80 GHz	4 GBps
PCI-e X32	160 GHz	8 GBps

Le velocità di trasferimento si calcolano tenendo conto del fatto che a livello fisico il PCI-e usa un codice 8bit/10 bit, ossia per trasmettere un gruppo di 8 bit in effetti ne utilizza 10 in modo da avere una sincronizzazione più stabile, in questo modo l'efficienza di trasmissione si abbassa all'80% => per trasferire 4GB ne occorrono 5 GB:

A 5 GHz => posso trasmettere 5Gbps, ma con rendimento 80% => 4 Gbps
 Ma 4Gbps = 4 Gbps/8 = 500 MBps, ma essendo full duplex => 250 MBps per ogni verso.

Quindi con PCI-e X1, lavorando a 5 GHz possiamo trasmettere 250 MBps per ogni verso.

INSERIRE IMMAGINI pag 228

BUS SCSI (Small Computer System Interface)

È uno standard per il collegamento di periferiche e personal computer, in particolare per Hard Disk, ma anche Drive CDROM, masterizzatori, dischi zip, scanner, ecc...

Le periferiche dialogano con il Bus SCSI attraverso il Controller SCSI (**SCSI Host adapter** o HBA). Nei PC il controller SCSI è collegato al bus PCI mentre nelle Workstation direttamente sulla scheda madre.

Esistono tre standard SCSI di riferimento:

1. **SCSI-1:**
 - Definisce il cablaggio, i comandi e il modo di trasferimento dei dati;
 - Clock a 5 MHz => banda passante: **4,77 MBps** ($5 \cdot 10^6 / 2^{20} = 4,77$);
 - Si possono collegare fino a 8 dispositivi
2. **SCSI-2**
 - Clock a 10 MHz
 - Fino a 16 dispositivi
3. **SCSI-3** (Ultra Wide SCSI)
 - Supporta diversi mezzi fisici (fibre ottiche, rame);
 - Clock da 20 a 40 MHz

Partendo dai tre standard si sono implementate diverse versioni di SCSI:

SCSI-1	Bus a 8 bit; velocità trasferimento: 5MBps
SCSI-2	Come SCSI1, ma connettore a 50 pin (25 in SCSI1) e supporta vari tipi di dispositivi
Wide SCSI	Bus a 16 bit
Fast SCSI	Bus a 8 bit, usa clock sia fronte salita che discesa => 10 MBps
Fast Wide SCSI	Bus a 16 bit e velocità di 20MBps
Ultra SCSI	Bus a 8 bit e velocità di 20MBps
SCSI-3	Bus a 16 bit e velocità di 40MBps (Ultra Wide SCSI)
Ultra2 SCSI	Bus a 8 bit e velocità di 40MBps
Wide Ultra2 SCSI	Bus a 16 bit e velocità di 80MBps

Lo standard USB (Universal Serial Bus)

Standard aperto per utilizzare vari tipi di periferiche di costruttori diversi.

Principali caratteristiche:

- Porte seriali ad alta velocità:
 - 12 Mbps per standard USB1.1;
 - 480 Mbps per standard USB2.0;
- fino a 127 dispositivi in cascata;
- connessione e configurazione a pc acceso (Hot Swap);
- supporto del real time per servizi isocroni (trasmissione di voce, audio e video).

Un **sistema USB** si basa su tre elementi:

1. Host USB

- Sistema di elaborazione dove è installato il controller USB (interfaccia tra pc e sistema USB); cioè è il pc in cui è installato il controller USB.

- Anovera l'**Hub-root**, cioè le porte principali in cui è possibile collegare Dispositivi Usb o Hub Usb.
- Interagisce con i dispositivi attraverso l'**Host Controller** che si occupa di:
 - Riconoscere il dispositivo USB collegato (o scollegato);
 - Gestire il collegamento tra il dispositivo USB e l'Host;
 - Gestire lo scambio di dati tra il dispositivo USB e l'Host;
 - Fornire alimentazione al dispositivo USB.

2. Interconnessione USB

- Come sono collegati e come comunicano i dispositivi con l'Host
- L'Interconnessione comprende in particolare:
 - La Topologia: schema di collegamento fisico tra dispositivi e host;
 - La politica di Arbitraggio del bus (essendo condiviso, come viene assegnato ad una periferica);

3. Dispositivi USB. Comprendono:

- Hub USB:
 - Permette di estendere il numero di porte al sistema USB in quanto è un concentratore in grado di convertire (multiplex are) un singolo punto in più punti di accesso;
 - Ogni cavo è un collegamento punto-punto tra l'Host (Hub-root) e un Hub o un Dispositivo.
- Dispositivi USB (Dispositivi semplici o funzioni):
 - Oggetto in grado di scambiare dati di controllo sul bus
 - Esempi di dispositivi Usb:
 - Dispositivi per l'interfaccia utente: mouse, tastiere;
 - Dispositivi per le immagini: scanner, stampanti, videocamere;
 - Memorie di massa: Hard Disk, CD-ROM drive, Floppy drive, DVD drive.

Velocità di collegamento

Un sistema USB può lavorare a diverse velocità in base al tipo di dispositivo collegato:

- **Low speed:**
 - 10-100 Kbps
 - Mouse e tastiere
- **Full speed:**
 - 0,5-10 Mbps
 - microfoni, video compresso
- **High speed:**
 - 25-480 Mbps
 - applicazioni in banda larga (servizi isocroni).

Caratteristiche elettriche

Usb usa 4 fili:

- D+ e D- in cui viaggiano i dati:
 - secondo la codifica NRZI (a 0 corrispondono $-D$ volt e a 1 corrispondono $+D$ volt);
 - con tecnica del bit stuffing per sincronizzare il clock tra trasmettitore e ricevitore;
 - sono binati in modo da eliminare interferenze;
- VBus e GND che forniscono alimentazione ai dispositivi non autoalimentati.
 - VBus = 5V



Caratteristiche meccaniche

Si occupano della forma dei connettori:

- I connettori verso l'Host vengono detti Upstream
- I connettori verso il dispositivo vengono detti Downstream
- Non sono intercambiabili, ossia non possono essere invertiti per errore.

Inserire Immagine pag 236

Bus IEEE 1394 (Firewire)

Bus seriale sviluppato da Apple per applicazioni audio e video e che richiedono quindi banda elevata e garantita. È diventato uno standard per le applicazioni audio e video.

Caratteristiche:

- È una interfaccia seriale e prevede due modalità di trasferimento dei dati:
 - Asincrona: il dato viene re-inviato se la linea non fosse libera;
 - Isocrona: invio di un flusso continuo di dati in real-time.
- Non richiede uso del pc: può essere usato per stampare direttamente fotoda una macchina fotografica o per trasferire file tra due videocamere;
- Plug&Play;
- Collegamento a caldo (Hot swap);
- Può collegare fino a 63 dispositivi su un singolo bus (topologia a stella o albero)
- Versioni:
 - IEEE 1394a:
 - Approvato nel 2000
 - Supporta velocità di 100 Mbps, 200 Mbps e 400 Mbps
 - IEEE 1394b:
 - Velocità di trasferimento fino a 3,2 Gbps.
- CABLAGGIO:
 - Massima lunghezza del cavo:
 - 4,5 m per dispositivi full-speed
 - 14 m per dispositivi low-speed
 - due tipi di cavi:
 - 6 fili:
 - 2 coppie di fili binati e schermati per trasferire i dati;
 - 2 fili per alimentare i dispositivi (fino a 60w)
 - 4 fili:

- 2 coppie di fili binati e schermati per trasferire i dati;
- non sono presenti i cavi di alimentazione dei dispositivi.

Inserire Immagine pag 237

La porta seriale (RS232/C)

Lo standard RS232/C (Recommended Standard 232) rappresenta la prima interfaccia che ha permesso lo scambio di dati tra dispositivi digitali.

Il termine seriale si riferisce alla modalità di trasmissione dei bit: i bit vengono inviati uno alla volta in un unico filo.

RS232 segue lo standard EIA/TIA (Electronic Industry Association/Telecommunication Industry Association) che lo definisce come l'interfaccia:

- tra DTE (Data terminal Equipment), cioè il computer;
- e il DCE (Data Circuit Equipment), cioè il modem.
- Tramite trasmissione seriale asincrona.

Lo standard definisce: le caratteristiche funzionali, elettriche e meccaniche nello scambio di dati tra DTE e DCE tramite collegamenti punto-punto.

Velocità di trasferimento:

- Max 20 Kbps
- Usando la linea telefonica PSTN, max 30Kbps. Limite aumentato tramite algoritmi di compressione.

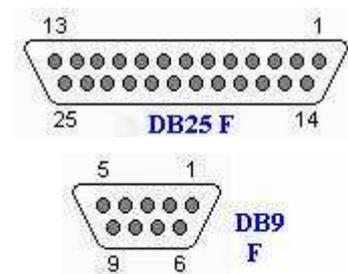
Caratteristiche meccaniche

Nel pc sono chiamate COM1 e COM2.

Connettori:

- Lato DTE (computer):
 - Cannon DB9S a 9 Pin, Maschio
 - Cannon DB25S a 25 Pin, Maschio
- Lato DCE (Modem):
 - Cannon DB9F a 9 Pin, Femmina
 - Cannon DB25F a 25 Pin, Femmina

Nel pc erano presenti normalmente una o due DB9S maschio.



Caratteristiche funzionali

Vedi libro pag. 239-240

La porta parallela

Nel pc è presente una sola porta parallela chiamata anche LPT1, PRN o interfaccia Centronics. Fu sviluppata nel 1981 da IBM per collegare stampanti ad aghi.

Caratteristiche:

- Trasferisce 1 byte alla volta (parallelo)

- Nel pc è presente un connettore DB25 Femmina
- Usa 17 linee per i segnali:
 - 8 per la trasmissione dei dati
 - Altre per il controllo: disponibilità nuovo byte, mancanza di carta, ecc...
- Inizialmente era unidirezionale successivamente fu standardizzata una nuova porta EPP (Enhanced Parallel Port) più veloce e bidirezionale.

Caratteristiche elettriche e meccaniche

Tensioni:

- Livello logico 1 da 2,4V a 5V
- Livello logico 0 da 0V a 0,8V

Cavi: max 5 m

La porta IDE-EIDE e la gestione dei Dischi

L'Hard Disk è una memoria permanente di tipo magnetico con capacità fino a qualche TB.

Fisicamente è così costituito:

- Pila di dischi magnetici che ruotano attorno ad un asse comune;
- Le informazioni sono memorizzate, in ogni faccia di ogni disco, in:
 - **TRACCE** concentriche suddivise in **SETTORI** separati da **GAP**.
 - ogni settore ha la stessa dimensione (normalmente 512 byte)
 - le tracce tra i diversi piatti disegnano dei Cilindri virtuali
 - delle testine che leggono dai settori. Ci sono testine:
 - Testine Fisse: solo movimento rotatorio del disco
 - Testine Mobili: movimento delle testine oltre a quello del disco

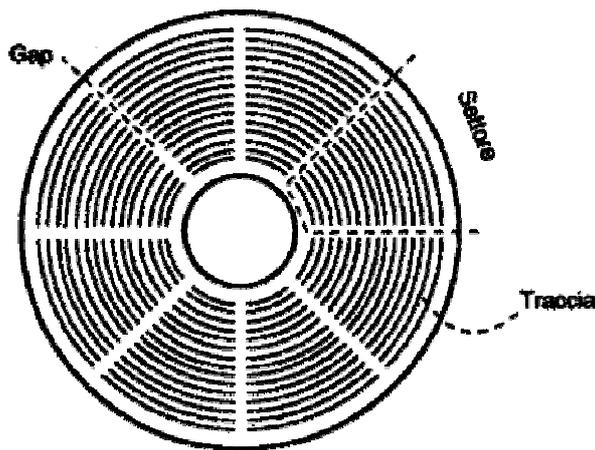


Figura 10: tracce, settori e gap sulla superficie di un disco

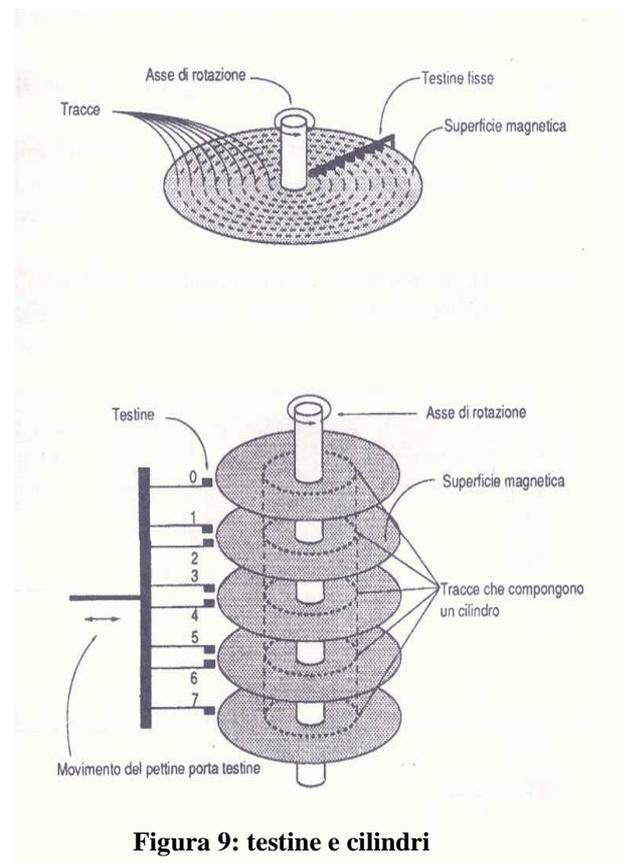


Figura 9: testine e cilindri

Le parti meccaniche che permettono di accedere ad ogni disco si chiamano **Disk Drive** e comprende:

- L'asse di rotazione;
- Il motore che fa girare il disco. Oggi velocità tipiche sono 7200 rpm (rear per minute);
- Le testine montate su un pettine (n° testine=n°facciate dei piatti)
- Un secondo motore per il posizionamento delle testine sul disco
- Circuiti di controllo che sincronizzano la lettura/scrittura