

4. COME FUNZIONA IL PC

Quando ho pensato questo corso ho cercato di capire quali potessero essere i bisogni di persone che conoscevano il mondo dei personal computer a macchia di leopardo, con esperienze più a meno approfondite su certi temi e quasi nulle su altri, ne è uscito un qualcosa che più che un corso è un insieme di “dritte” utili per entrare con un certa sicurezza nel mondo informatico per poi approfondire da solo le problematiche che servono. Lo scopo è quello di creare le competenze dell'oggetto PC, sia esso desktop, portatile, tablet o smartphone, per affrontarlo con tranquillità avendo come strumenti le conoscenze di base comuni a tutti questi oggetti.

Analogamente per il software non si approfondiranno i singoli programmi ma ci si intratterrà su quelle caratteristiche che consentiranno un approccio da “dominatori” anche a programmi mai usati. Un'ultima considerazione fondamentale: spesso persone che si ritengono di scarse competenze hanno paura toccando un tasto di fare un disastro, vedremo che un PC è un oggetto che non ha niente di sovranaturale, non ha intelligenza propria e per fare disastri occorre proprio impegnarsi perché per ogni operazione potenzialmente pericolosa il nostro PC chiederà il permesso e anche se per distrazione glielo daremo a sproposito, raramente tutto è perduto. Per cui coraggio ed impariamo a trattare il nostro Pc come il nostro schiavetto personale e non come una misteriosa entità che ci terrorizza.

4.1 PRINCIPI DI FUNZIONAMENTO

La prima cosa che dobbiamo acquisire è che un computer non è un essere onnipotente ma solo una macchina che pur nella sua complessità è piuttosto stupida. Si discute molto di intelligenza artificiale e in molti racconti o film si inventano losche trame in cui un potente calcolatore diventa super- intelligente e vuole dominare il mondo. In una recente pellicola addirittura il protagonista si innamora di un PC. Bene:

UN CALCOLATORE SA FARE SOLO QUELLO CHE GLI HANNO INSEGNATO E FATTO
MEMORIZZARE, LA FA BENE E NON SBAGLIA MAI.

Chiarito questo concetto un calcolatore, per quanto potente, è una macchinetta che sa fare solo somme e sottrazioni e per giunta con un suo linguaggio particolare formato da 0 e 1. Quando accenneremo alla programmazione questo sarà più chiaro.

Ovviamente per comunicare dovremo “parlargli” nella sua lingua e con le sue regole.

Iniziamo pertanto a capire questo linguaggio come è fatto e come si usa.

Consideriamo un interruttore, ad esempio uno di casa per accendere la luce, diciamo che il suo stato vale **0** quando è **aperto**, ovvero quando la luce è spenta, e che vale **1** quando è **chiuso**, ovvero quando la luce è accesa. Per inciso: ormai praticamente su tutti gli apparati elettrici si trova il simbolo 0 per spento e I per acceso.

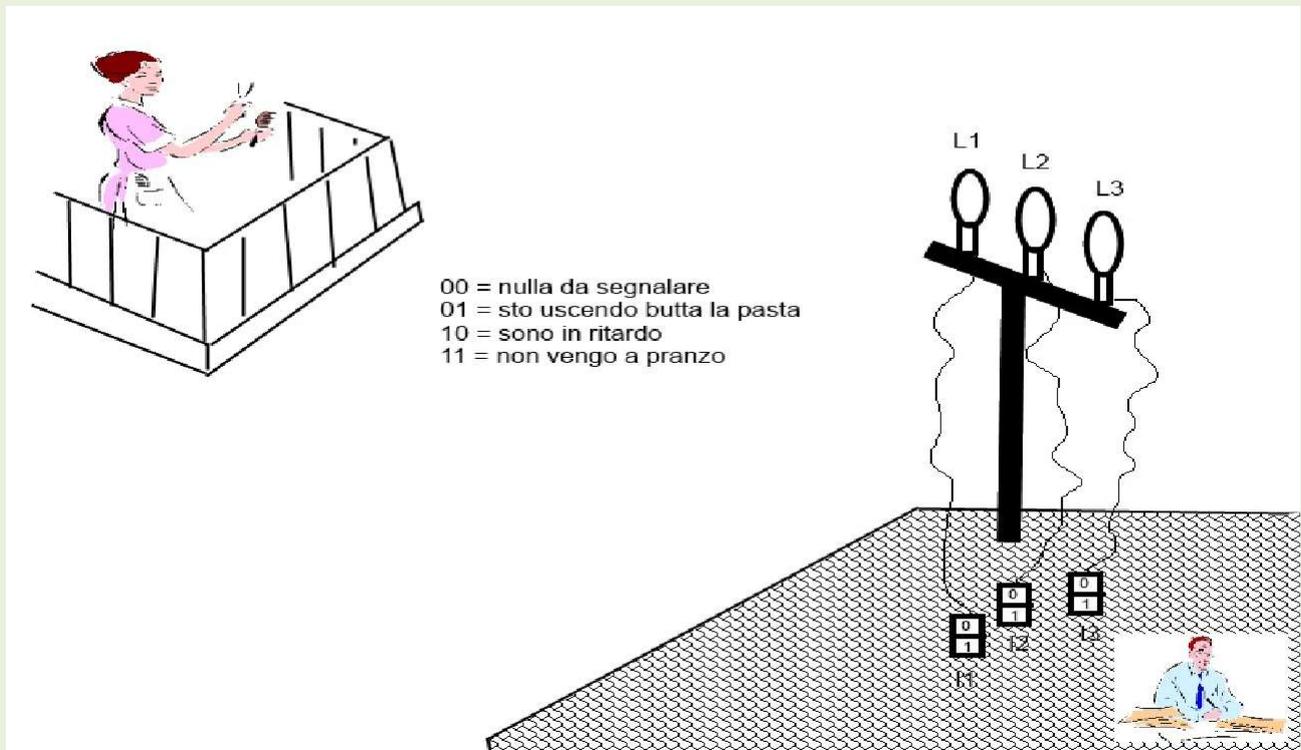
Supponiamo di avere tre lampadine (L1, L2, L3) installate su un palo rizzato sul tetto della ditta in cui lavora il signor Rossi che può accenderle tramite tre interruttori (I1,I2,I3). Per ora consideriamone solo due: L1 e L2. La signora Rossi può vederle dal suo balcone.

Un semplice modo di comunicare informazioni (ovvero **un codice** di comunicazione) potrebbe essere così costruito:

- | | | |
|---------------------------|-----|----------------------------|
| 1. L1 spenta L2 spenta => | 0 0 | niente da segnalare |
| 2. L1 spenta L2 accesa => | 0 1 | sto uscendo butta la pasta |
| 3. L1 accesa L2 spenta => | 1 0 | sono in ritardo |
| 4. L1 accesa L2 accesa => | 1 1 | non vengo a pranzo |

Ad un'ora prestabilita il Sig. Rossi manovra gli interruttori e accende le lampadine secondo il

codice, le tiene accese per 5 minuti poi le spegne tutte.



Una comunicazione potrebbe essere:

ore 12 1 0 = sono in ritardo

ore 12,05 00 = niente da segnalare (il sig. Rossi ha spento le lampadine)

ore 12,30 01 = sto uscendo butta la pasta

ore 12,35 00 = niente da segnalare (il sig. Rossi ha spento le lampadine)

Come si vede si è introdotta una nuova variabile: il **tempo**.

In qualsiasi elaboratore la temporizzazione è fondamentale: tutto funziona sulla temporizzazione fornita da un orologio digitale interno (in inglese **clock**) pena il caos totale.

Riprendiamo il nostro esempio, se i due messaggi 10 e 01 arrivassero a caso si avrebbe:

01 seguito da 10 la moglie butta la pasta e il signor Rossi la mangerà scotta,

viceversa se arrivasse 10 seguito da 01 tutto sarebbe ok.

Immaginate cosa succederebbe in un PC dove i segnali trasmessi sono centinaia al secondo.

Occorre pertanto un sistema di orologi sincronizzato tra trasmittente (sig, Rossi) e ricevente (sig.ra

Rossi).

Riprendiamo il nostro esempio e supponiamo che la sig.ra Rossi sia in ritardo di circa 5 minuti. Abbiamo detto che il sig.Rossi dopo 5 minuti spegne le lampadine, diciamo che ci mette 1 minuto per spegnere tutto:

alle 12,06 la signora Rossi vedrebbe L1 spenta (perché nel frattempo è stata spenta) e L2 spenta perché lo era già e leggerebbe: 00 niente da segnalare

alle 12,36 vedrebbe L1 spenta L2 spenta leggerebbe 00 niente da segnalare e la pasta non la butta.

Se viceversa fosse in anticipo, alle 11,55 leggerebbe 00 e alle 12,,25 ancora 00 il sig. Rossi non mangia.

Ne consegue che l'ora letta deve essere uguale per tutti e due ad esempio prendendo quella del campanile (orologio di sistema).

Ma la sincronizzazione non basta, bisogna essere sicuri che il messaggio sia stato ricevuto correttamente occorre pertanto aggiungere un sistema di segnalazione a ritroso con cui il ricevente confermi la ricezione stessa. Nel nostro caso la sig.ra Rossi potrebbe avere una lampadina che accende o un fazzoletto rosso da sventolare se è tutto ok e il sig. Rossi una terza lampadina che se accesa indica "trasmissione in corso".

Ricapitolando ci servono:

1. un codice
2. un orologio comune
3. un sistema di segnalazione

Approfondiamo il concetto di codice. Intanto diamo un paio di definizioni che ci serviranno più tardi:

digitale = espresso con numeri ovvero con sequenze di 0 e 1 (in inglese **digit** significa cifra)

bit = quantità minima di informazione ovvero 0 oppure 1

byte= sequenza di 8 bit

Se nel nostro esempio aumentiamo il numero di lampadine possiamo definire più parole, vediamo perché:

1 lampadina = 2 stati = 2 significati

2 lampadine = 4 stati = 00- 01- 10- 11 = 4 significati =2 x 2

3 lampadine = 8 stati = 000- 001 - 010 - 011-100 - 101- 110 - 111 = 8 significati = $2 \times 2 \times 2$

2^N Se cioè ho N lampadine avrò $2 \times 2 \times \dots \times 2 =$
--N volte----

Ricordando che 0 e 1 sono bit abbiamo;

2^N bit.

Nei calcolatori i messaggi tra le varie parti e verso l'esterno avvengono con messaggi di “parole” formate da sequenze di 0 e 1. I primi computer parlavano con parole di 4 bit, successivamente si è passati a 8 bit , 16, 32, 64 e certi processori usano anche 128 e 256 bit. Tanto per capirci:

$$2^8 = 256 \quad 2^{16} = 65536 \quad 2^{32} = 4.294.967.296 \quad 2^{64} = 18.446.744.073.709.551.616$$

Questo aumento di “parole” è quello che da una parte ha aumentato la velocità del sistema in quanto parole più lunghe si traducono nella possibilità di trasmettere più informazioni in contemporanea (un po' come parlare con una frase invece che con una parola alla volta) e dall'altra parte di codificare oggetti molto complessi come immagini in movimento; nonostante questo la “parola” base resta il **Byte** cioè una “parola” di 8 bit, le “parole” più lunghe sono espresse in numero di Byte. Vedremo meglio questi aspetti più avanti, per ora ci basti sapere che oggi i sistemi che abbiamo disponibili nelle nostre case funzionano a 32 o 64 bit.

Fissiamo i concetti :

1. i calcolatori usano un sistema digitale di 0 e 1
2. nei calcolatori si scambiano messaggi di 8 bit o più multipli di 8 bit
3. prima di un messaggio devo avvertire che lo sto mandando
4. devo dare conferma della ricezione del messaggio

Per lo svolgimento delle operazioni in un PC si usa la matematica binaria basata su due soli simboli: 0 e 1 e con regole uguali al sistema decimale di uso corrente, ricordando che anche le operazioni complesse si riducono a somme e sottrazioni che il nostro calcolatore fa a velocità elevatissime.

Detti simboli rappresentano gli “interruttori” aperti o chiusi, abbiamo detto che sa fare solo somme e differenze, abbiamo detto che usa i byte: mettendo insieme questi concetti risulta che i byte rappresentano “gli interruttori “ aperti e chiusi, o meglio per fare un altro passo avanti

rappresentano **impulsi di corrente “alta” (cioè 1) e “bassa” (cioè 0)** che vengono emessi dai circuiti elettronici che compongono il nostro PC. Nella realtà vi sono numeri elevatissimi di calcoli da fare in tempi brevissimi resi possibili da decine di migliaia di componenti elettronici miniaturizzati e raccolti in un piccolo contenitore di pochi centimetri , chiamato processore; nel PC il processore principale è la **CPU =Central Processing Unit = Unità di Elaborazione Centrale**

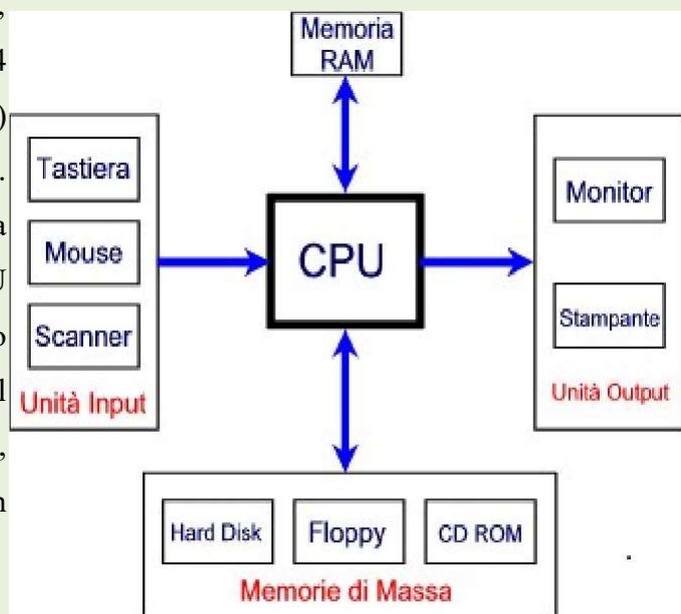
La **CPU** è il centro del sistema, il “cervello” che tutto governa; PC, Tablet, Smartphone; a seconda della complessità e della velocità della macchina gestisce tutto direttamente o tramite “schiavetti” a cui delega funzioni specializzate (ad esempio scheda video con processore dedicato, integrato per l'audio etc). Invia e riceve velocissimamente dati da e verso tutte le periferiche e le altre componenti.

Per sollevare la CPU da questi compiti gravosi esistono opportuni circuiti che gestiscono i flussi di messaggi, un po' come fa un vigile col traffico stradale, indirizzando i messaggi verso, ad esempio, il monitor, la stampante etc.

Le CPU sono il pezzo più costoso e delicato del PC, richiedono molte cautele nel maneggiarle. Hanno milioni di circuiti ultraminiaturizzati e, ricordando che le CPU più potenti sono a 64 bit, un numero considerevole di contatti (**pin**) che consente di usare “parole” molto lunghe. Le ultime generazioni sono CPU che sono a loro volta sono formate da 2, 4 o più CPU integrate che lavorano insieme. Un altro parametro da considerare è la “velocità” nel fare le operazioni ovvero la “frequenza”, misurata come detto precedentemente in multipli di Hertz.

Richiedono dei dissipatori di calore molto

efficaci in quanto la concentrazione dei circuiti e **la frequenza** di uso producono molto calore.



Abbiamo detto che il tempo è fondamentale per il funzionamento di un sistema digitale, quando parliamo di tempo in un processore, in una memoria, etc. si intende in **il numero di volte al secondo** che emetto / ricevo un segnale 0 oppure 1. Questo numero si chiama **frequenza** e si

misura in **Hertz** (abbreviato **Hz**) ed ha come multipli il **Kilohertz** (**kHZ**=mille hertz), il

MegaHertz (**Mhz** = un milione di hertz), il **Gigahertz** (**Ghz** = un miliardo di hertz).

La frequenza a cui lavora un processore è fondamentale per capire quanto veloce esso sarà.

Le prime CPU avevano frequenze di qualche Mhz, oggi funzionano a valori di qualche Ghz.

E' importante sapere che non tutti i componenti dei PC possono lavorare alla stessa

frequenza; ad esempio i dischi rigidi sono molto più lenti della CPU. Poiché le varie parti

devono funzionare in modo sincronizzato la soluzione, detto in modo brutale, è che **il più**

veloce aspetta il più lento ;come? Semplicemente i vari messaggi vengono infilati in un

“deposito elettronico” (la **memoria** disistema o **RAM**) uno dietro l'altro e poi ripresi nella

giusta sequenza; nel frattempo il più veloce fa altre cose oppure “perde tempo” non facendo

niente. Le memorie hanno frequenze fino a 4 Ghz. Per inciso, ormai dovrebbe essere chiaro

che quando pigio un tasto nervosamente perché apparentemente il mio PC non mi risponde

non faccio che aggravare la situazione in quanto la tastiera continua ad inviare messaggi

alla CPU che essendo impegnata a fare altro non risponde subito, continuando ad

immagazzinare in memoria i messaggi e rallentando ulteriormente il sistema quindi: **CI**

VUOLE PAZIENZA.

-----*****-----

Nella **versione estesa** consultabile nel successivo capitolo è possibile approfondire sia il

funzionamento intrinseco del Pc sia la composizione delle parti principali e accessorie

con le relative modalità di interazione con la CPU.

4.2 APPROFONDIMENTI

A cura SILVIO MARRESE

COS' E' UN PERSONAL COMPUTER E COME RAGIONA

La prima cosa che dobbiamo acquisire è che un computer non è un essere onnipotente ma solo una macchina che pur nella sua complessità è piuttosto stupida. Si discute molto di intelligenza artificiale e in molti racconti o film si inventano losche trame in cui un potente calcolatore diventa super- intelligente e vuole dominare il mondo. In una recente pellicola addirittura il protagonista si innamora di un PC. Bene:

UN CALCOLATORE SA FARE SOLO QUELLO CHE GLI HANNO INSEGNATO E FATTO MEMORIZZARE, LA FA BENE E NON SBAGLIA MAI.

Chiarito questo concetto un calcolatore, per quanto potente, è una macchinetta che sa fare solo somme e sottrazioni e per giunta con un suo linguaggio particolare formato da 0 e 1. Quando accenneremo alla programmazione questo sarà più chiaro.

Ovviamente per comunicare dovremo “parlargli” nella sua lingua e con le sue regole.

Iniziamo pertanto a capire questo linguaggio come è fatto e come si usa.

Consideriamo un interruttore, ad esempio uno di casa per accendere la luce, diciamo che il suo stato vale 0 quando è aperto, ovvero quando la luce è spenta, e che vale 1 quando è chiuso, ovvero quando la luce è accesa. Per inciso: ormai praticamente su tutti gli apparati elettrici si trova il simbolo 0 per spento e I per acceso.

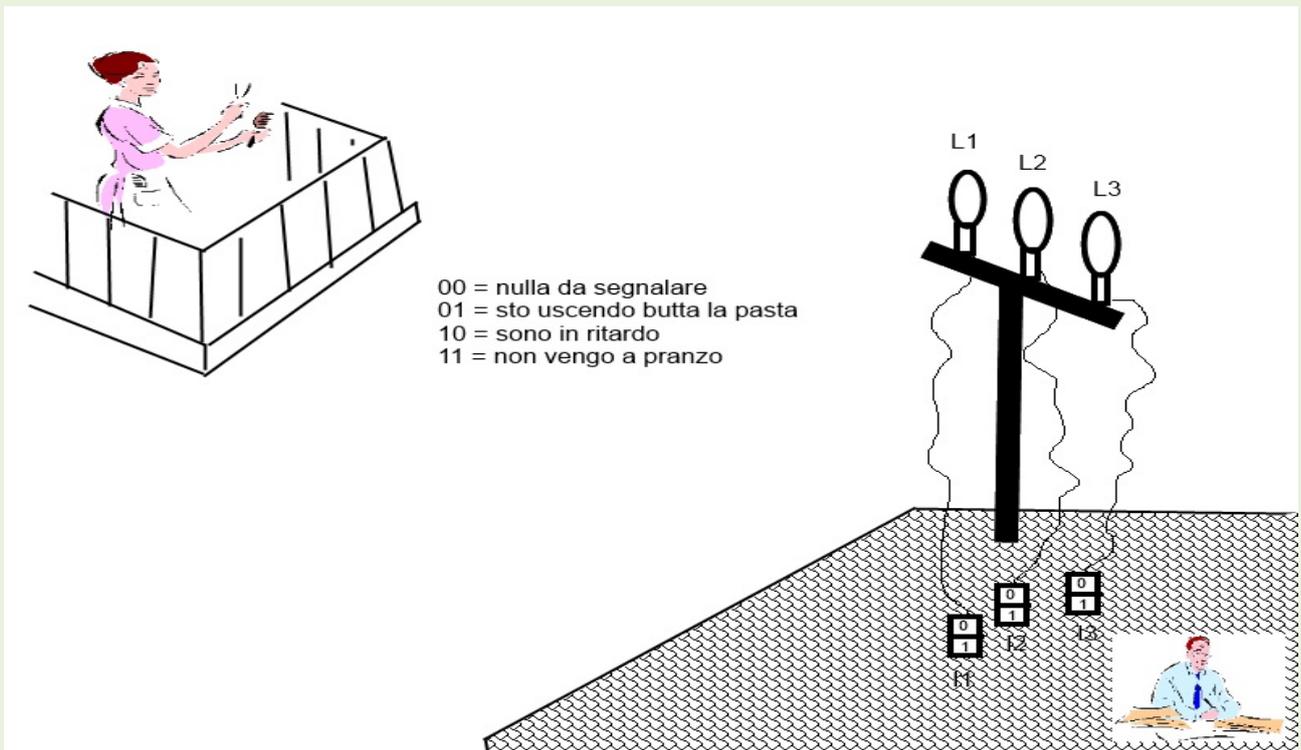
Supponiamo di avere tre lampadine (L1, L2, L3) installate su un palo rizzato sul tetto della ditta in cui lavora il signor Rossi che può accenderle tramite tre interruttori (I1,I2,I3). Per ora consideriamone solo due: L1 e L2. La signora Rossi può vederle dal suo balcone.

Un semplice modo di comunicare informazioni (ovvero un codice di comunicazione) potrebbe essere così costruito:

1. L1 spenta L2 spenta => 0 0 niente da segnalare

2. L1 spenta L2 accesa => 0 1 sto uscendo butta la pasta
3. L1 accesa L2 spenta => 1 0 sono in ritardo
4. L1 accesa L2 accesa => 1 1 non vengo a pranzo

Ad un'ora prestabilita il Sig. Rossi manovra gli interruttori e accende le lampadine secondo il codice, le tiene accese per 5 minuti poi le spegne tutte.



Una comunicazione potrebbe essere:

ore 12 1 0 = sono in ritardo

ore 12,05 00 = niente da segnalare (il sig. Rossi ha spento le lampadine)

ore 12,30 01 = sto uscendo butta la pasta

ore 12,35 00 = niente da segnalare (il sig. Rossi ha spento le lampadine)

Come si vede si è introdotta una nuova variabile: il **tempo**.

In qualsiasi elaboratore la temporizzazione è fondamentale: tutto funziona sulla temporizzazione fornita da un orologio digitale interno (in inglese **clock**) pena il caos totale.

Riprendiamo il nostro esempio, se i due messaggi 10 e 01 arrivassero a caso si avrebbe:

01 seguito da 10 la moglie butta la pasta e il signor Rossi la mangerà scotta, viceversa se arrivasse 10 seguito da 01 tutto sarebbe ok.

Immaginate cosa succederebbe in un PC dove i segnali trasmessi sono centinaia al secondo.

Occorre pertanto un sistema di orologi sincronizzato tra trasmittente (sig, Rossi) e ricevente (sig.ra Rossi).

Riprendiamo il nostro esempio e supponiamo che la sig.ra Rossi sia in ritardo di circa 5 minuti. Abbiamo detto che il sig.Rossi dopo 5 minuti spegne le lampadine, diciamo che ci mette 1 minuto per spegnere tutto: alle 12,06 la signora Rossi vedrebbe L1 spenta (perché nel frattempo è stata spenta) e L2 spenta perché lo era già e leggerebbe: 00 niente da segnalare

alle 12,36 vedrebbe L1 spenta L2 spenta leggerebbe 00 niente da segnalare e la pasta non la butta.

Se viceversa fosse in anticipo, alle 11,55 leggerebbe 00 e alle 12,25 ancora 00 il sig. Rossi non mangia.

Ne consegue che l'ora letta deve essere uguale per tutti e due ad esempio prendendo quella del campanile (orologio di sistema).

Ma la sincronizzazione non basta, bisogna essere sicuri che il messaggio sia stato ricevuto correttamente occorre pertanto aggiungere un sistema di segnalazione a ritroso con cui il ricevente confermi la ricezione stessa. Nel nostro caso la sig.ra Rossi potrebbe avere una lampadina che accende o un fazzoletto rosso da sventolare se è tutto ok e il sig. Rossi una terza lampadina che se accesa indica "trasmissione in corso".

Ricapitolando ci servono:

1. un codice
2. un orologio comune
3. un sistema di segnalazione

----- ***** -----

Approfondiamo il concetto di codice. Intanto diamo un paio di definizioni che ci serviranno più tardi:

digitale = espresso con numeri ovvero con sequenze di 0 e 1 (in inglese **digit** significa cifra)

bit = quantità minima di informazione ovvero 0 oppure 1

byte= sequenza di 8 bit

Se nel nostro esempio aumentiamo il numero di lampadine possiamo definire più parole, vediamo perché:

1 lampadina = 2 stati = 2 significati

2 lampadine = 4 stati = 00- 01- 10- 11 = 4 significati =2 x 2

3 lampadine = 8 stati = 000- 001 - 010 - 011-100 - 101- 110 - 111 = 8 significati = 2 x 2 x2

2^N Se cioè ho N lampadine avrò $2 \times 2 \times \dots \times 2 =$

--N volte----

Ricordando che 0 e 1 sono bit abbiamo;

2^N bit.

Nei calcolatori i messaggi tra le varie parti e verso l'esterno avvengono con messaggi di “parole” formate da sequenze di 0 e 1. I primi computer parlavano con parole di 4 bit, successivamente si è passati a 8 bit , 16, 32, 64 e certi processori usano anche 128 e 256 bit. Tanto per capirci:

$$2^8 = 256 \quad 2^{16} = 65536 \quad 2^{32} = 4.294.967.296 \quad 2^{64} = 18.446.744.073.709.551.616$$

Questo aumento di “parole” è quello che da una parte ha aumentato la velocità del sistema in quanto parole più lunghe si traducono nella possibilità di trasmettere più informazioni in contemporanea (un po' come parlare con una frase invece che con una parola alla volta) e dall'altra parte di codificare oggetti molto complessi come immagini in movimento; nonostante questo la “parola” base resta il **Byte** cioè una “parola” di 8 bit, le “parole” più lunghe sono espresse in numero di Byte.

Vedremo meglio questi aspetti più avanti, per ora ci basti sapere che oggi i sistemi che abbiamo disponibili nelle nostre case funzionano a 32 o 64 bit.

Fissiamo i concetti :

1. i calcolatori usano un sistema digitale di 0 e 1
2. nei calcolatori si scambiano messaggi di 8 bit o più multipli di 8 bit
3. prima di un messaggio devo avvertire che lo sto mandando
4. devo dare conferma della ricezione del messaggio

-----*****-----

Facciamo un altro passo.

Diamo una nuova definizione:

CPU= unità di processo centrale (in inglese **Central Processing Unit**)

La CPU è il centro del sistema il “cervello” che tutto governa. In seguito vedremo meglio cos'è.

Abbiamo detto che in un calcolatore si scambiano tra le varie parti centinaia di messaggi, cosa succede se arrivano più messaggi prima che la CPU possa eseguire le operazioni richieste? Come ad uno sportello vengono messe in coda e gestite in sequenza tramite un nuovo oggetto “**la memoria**” in cui vengono

posteggiati i messaggi che sono costituiti da operazioni da fare e dati da elaborare. Per sollevare la CPU da questi compiti gravosi esistono opportuni circuiti che gestiscono i flussi di messaggi, un po' come fa un vigile col traffico stradale, indirizzando i messaggi verso, ad esempio, il monitor, la stampante etc. vedremo più diffusamente questi processi più tardi

Per inciso, ormai dovrebbe essere chiaro che quando pigio un tasto nervosamente perché apparentemente il mio PC non mi risponde non faccio che aggravare la situazione in quanto la tastiera continua ad inviare messaggi alla CPU che essendo impegnata a fare altro non risponde subito, continuando ad immagazzinare in memoria i messaggi e rallentando ulteriormente il sistema quindi: **CI VUOLE PAZIENZA.**

Affrontiamo ora un passaggio che richiede attenzione: la matematica binaria.

Cominciamo col dire che nel mondo moderno ormai il sistema di misura è quello metrico decimale, cioè quello che usa dieci simboli da 0 a 9, che ci viene insegnato a scuola e che usiamo ogni giorno. Esistono però sistemi di misura basati su un numero di simboli diverso, ad esempio nel mondo anglosassone la sterlina si divideva ancora pochi anni fa in 12 penny non in 10 come ora, nel mondo dei PC i simboli sono 2: 0 e 1. Ma allora i numeri maggiori di 1 come si esprimono? Torniamo al sistema decimale: ho dieci simboli per cui posso assegnare “dieci quantità”, all'undicesima quantità uso 0 più 1 cioè 10 che significa che ho una unità che vale di dieci e 0 quantità delle unità. Così se voglio indicare 22 quantità userò 2 decine e 2 unità e così via, sono meccanismi che abbiamo assimilato da tempo ma ora li stiamo guardando in modo particolare.

Torniamo al sistema binario :

binario		decimale
0	=	0
1	=	1
10	=	2

Cosa ho fatto? Ho usato tutti i simboli disponibili poi ho detto che dell'insieme base (equivalente alle unità) non prendo niente, cioè 0, e prendo la prima dell'insieme superiore cioè 1.

Facciamo qualche altro esempio:

11	=	3
100	=	4 (qui ho fatto il passaggio analogo a quando passo dalle decine alle centinaia)
101	=	5

Fissiamo il concetto:

1. i simboli che rappresentano numeri possono essere diversi da dieci
2. nel sistema binario sono due

Ovviamente anche il sistema binario può essere utilizzato per fare i calcoli con le stesse regole del sistema decimale, ricordando che anche le operazioni complesse si riducono a somme e sottrazioni che il nostro calcolatore fa a velocità elevatissime.

Facciamo un esempio di come una moltiplicazione di fatto non è che una somma:

$$5 \times 3 = 15 \text{ cioè } 5+5+5 = 15 \text{ ovvero sommo } 5 \text{ per tre volte.}$$

Facciamo anche per completezza del ragionamento un esempio di somma (ma valgono tutte le operazioni possibili nel decimale) nel sistema binario:

ad esempio :

$$\begin{array}{r} \text{decimale} \quad \quad \quad 9 + \\ \quad \quad \quad \quad \quad 1 = \\ \hline \end{array}$$

0 riporto 1 cioè 10

$$\begin{array}{r} \text{binario} \quad 00001001 + \text{ (9 in decimale)} \\ \quad \quad \quad 00000001 = \text{ (1 in decimale)} \\ \hline \quad \quad \quad 00001010 \quad \text{ (10 in decimale)} \end{array}$$

vediamo cosa abbiamo fatto:

partendo da destra : $1+1 = 0$ riporto 1 (perché ho solo 2 simboli) quindi prima cifra di destra =0

2 posizione 1 di riporto + $0+0=1$ quindi seconda cifra = 1

3 e 4 cifra $0+0=0$ e $1+0=1$

5 6 7 cifra $0+0=0$

Il nostro risultato 10 in decimale in binario diventa 00001010

Si sarà notato che sono stati scritti anche degli 0 apparentemente inutili sulla sinistra, se facciamo mente locale a quanto detto precedentemente i due numeri che abbiamo sommato sono due **byte** che sono **sempre** formati da 8 cifre, da cui la presenza degli zeri a sinistra.

A questo punto ci si chiede perché parliamo di matematica binaria, a cosa serve? Vediamo perché è importante capire questi concetti.

Abbiamo detto che un calcolatore ha disponibile solo 2 simboli che rappresentano gli “interruttori” aperti o chiusi, abbiamo detto che sa fare solo somme e differenze, abbiamo detto che usa i byte:

mettendo insieme questi concetti risulta che i byte rappresentano “gli interruttori “ aperti e chiusi, o meglio per fare un altro passo avanti rappresentano **impulsi di corrente “alta” (cioè 1) e “bassa” (cioè 0)** che vengono emessi dai circuiti elettronici che compongono il nostro PC.

Nella nostra somma binaria di cui sopra, **fisicamente** siamo partiti da una situazione di “ 8 lampadine “ che partendo da sinistra sono:

spenta, spenta, spenta, spenta, **accesa**, spenta, spenta, **accesa**
e sommando il secondo byte la situazione è diventata:
spenta, spenta, spenta, spenta, **accesa**, spenta, **accesa**, spenta

cioè il messaggio è diverso, e ricordando ancora che un messaggio fisicamente equivale ad emissioni di impulsi, abbiamo fatto fare all'insieme di circuiti che compongono il PC qualcosa, ad esempio scrivere un puntino sullo schermo. Ma il byte che abbiamo sommato cos'è? ad esempio potrebbe essere un insieme di 8 impulsi (altro byte) inviati dalla tastiera.

Senza andare oltre ci basti sapere che quando si entra nel mondo digitale le azioni fatte dai vari circuiti vengono rappresentate con sequenze di 0 e 1 che corrispondono ad impulsi elettrici che a loro volta causano azioni che nel loro insieme fanno le “cose” che ci aspettiamo da un calcolatore.

----- **** -----

Quanto detto finora da solo i principi base su cui ci si muove, nella realtà vi sono numeri elevatissimi di calcoli da fare in tempi brevissimi, nei primordi dell'informatica per farli si usavano calcolatori elettronici che occupavano stanze intere oggi la super miniaturizzazione dei componenti ha consentito di costruire circuiti integrati che contengono decine di migliaia di componenti elettronici ultraminiaturizzati e racchiusi in un unico contenitore di pochi centimetri, chiamati processori, che fanno le stesse cose..Nei PC il processore principale è la CPU (**Central Processing Unit** =Unità di elaborazione centrale)

Tornando al digitale abbiamo detto che tutti i calcoli sono fatti in binario ovvero tramite impulsi di corrente, stante la mole delle operazioni che un processore fa, sia i costruttori che i programmatori usano dei linguaggi simbolici che **rappresentano** il funzionamento fisico dei circuiti utilizzando come sopra detto il sistema binario.

Dobbiamo fare ancora uno sforzo per avere tutti gli elementi che ci servono per capire il funzionamento del nostro PC: il concetto di **frequenza**.

Abbiamo detto che il tempo è fondamentale per il funzionamento di un sistema digitale, quando parliamo di tempo in un processore, in una memoria, etc. si intende in **il numero di volte al secondo** che emetto / ricevo un segnale 0 oppure 1. Questo numero si chiama **frequenza** e si misura in **Hertz** (abbreviato **Hz**) ed ha come multipli il **Kilohertz** (**kHZ**=mille hertz), il **MegaHertz** (**Mhz** = un milione di hertz), il **Gigahertz** (**Ghz** = un miliardo di hertz).

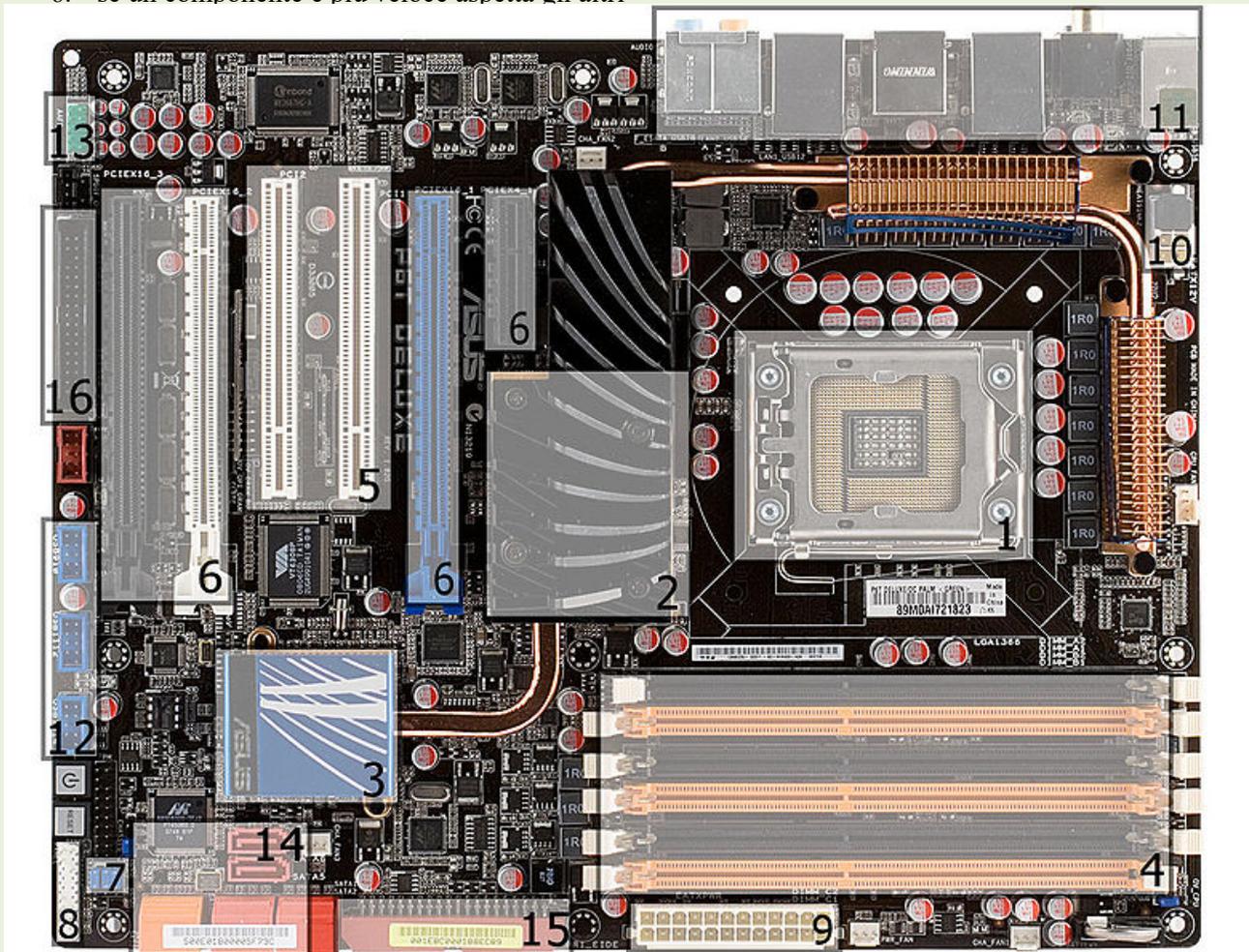
La frequenza a cui lavora un processore è fondamentale per capire quanto veloce esso sarà. Le prime CPU avevano frequenze di qualche Mhz, oggi funzionano a valori di qualche Ghz.

E' importante sapere che non tutti i componenti dei PC possono lavorare alla stessa frequenza; ad esempio i dischi rigidi sono molto più lenti della CPU. Ricordando che abbiamo detto che le varie parti devono

funzionare in modo sincronizzato la soluzione, detto in modo brutale, è che **il più veloce aspetta il più lento**, come? Semplicemente i vari messaggi vengono infilati in un “deposito elettronico” (la **memoria** di sistema o **RAM**) uno dietro l'altro e poi ripresi nella giusta sequenza, nel frattempo il più veloce o fa altre cose oppure “perde tempo” non facendo niente. Le memorie hanno frequenze fino a 4 Ghz.

Abbiamo a questo punto acquisito tutti i concetti che ci servono, ricapitoliamoli brevemente.

1. Nei PC il linguaggio simbolico è fatto di 0 e 1
2. gli 0 e 1 fisicamente sono impulsi “alti”=1 e “bassi”=0
3. gli 0 e 1 sono organizzati in un codice
4. le operazioni matematiche fatte coi codici rappresentano azioni fisiche fatte nei circuiti
5. i vari componenti del PC devono essere sincronizzati
6. se un componente è più veloce aspetta gli altri



Abbiamo visto come parla un PC, vediamo adesso quali sono le sue componenti principali che trovano posto su una scheda elettronica detta **motherboard** (=scheda madre) abbreviata in **MB**.

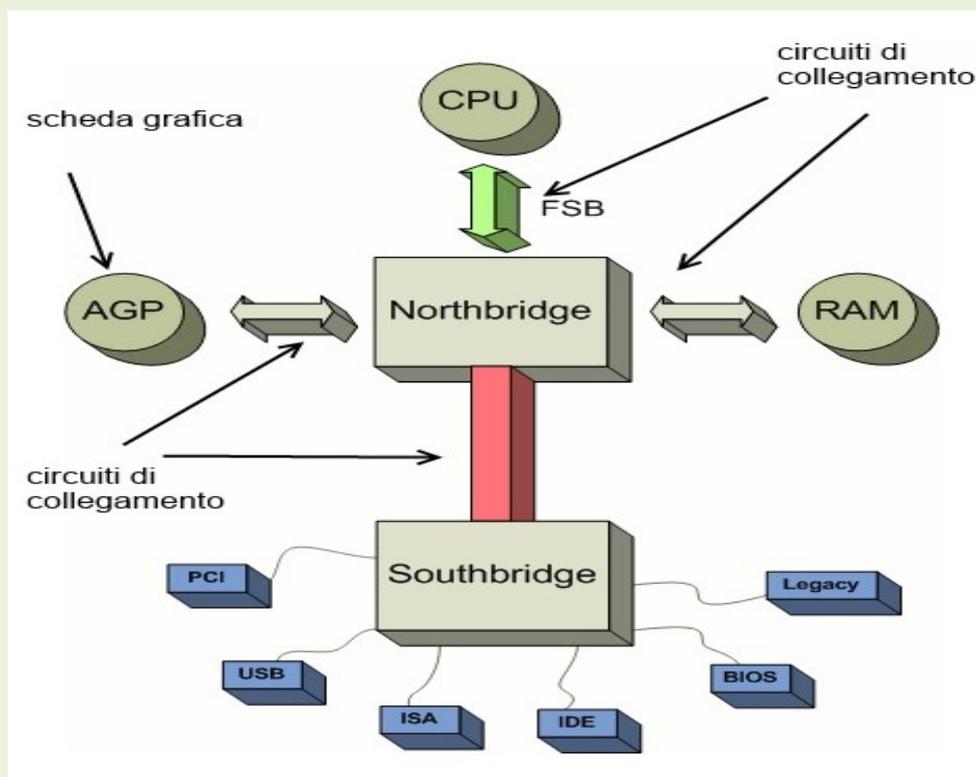
La **scheda madre** o **scheda di sistema** è una parte fondamentale di un moderno PC, sotto forma di scheda elettronica principale raccoglie in sé tutti i circuiti ed i collegamenti di interfaccia tra i vari componenti interni principali del PC (CPU, memoria, scheda grafica e etc montate o alloggiare sopra) e le interfacce verso le periferiche esterne (monitor, stampante etc).

È responsabile dunque della trasmissione e temporizzazione corretta di molte centinaia di segnali diversi, tutti ad alta frequenza e sensibili ai disturbi, tra processore e periferiche interne e viceversa.

La sua buona realizzazione è quindi un fattore chiave per le prestazioni e l'affidabilità dell'intero PC.

In quest'immagine di una scheda madre si notano i componenti principali (descritti nel paragrafo successivo): a destra il Socket (=zoccolo cioè sede di inserzione) della CPU (1), al centro il Northbridge (2) e il Southbridge (3), i banchi (connettori di inserzione) per la RAM (4), i connettori PCI (5), i connettori PCI express(6). In basso sono presenti: il connettore IDE(15), quello dell'alimentazione elettrica a 24 pin (9) e le porte SATA (14).

SCHEMA FUNZIONALE DELLA MOTHERBOARD



Northbridge
e (=ponte nord)

gestisce (come un vigile che gestisce il traffico) i circuiti verso la CPU, la scheda grafica e la memoria. Il northbridge su una scheda madre è il fattore che determina tipologia, numero e velocità della CPU, quantità e velocità della RAM che può essere utilizzata. Inoltre svolge il ruolo di regolazione della tensione e del numero di connessioni con le altre periferiche.

Southbridge (= ponte sud) gestisce i collegamenti verso i vari connettori presenti sulla MB e verso il

northbridge

PCI, PCI express, ISA sono connettori di espansione per schede aggiuntive (audio, grafica, etc)

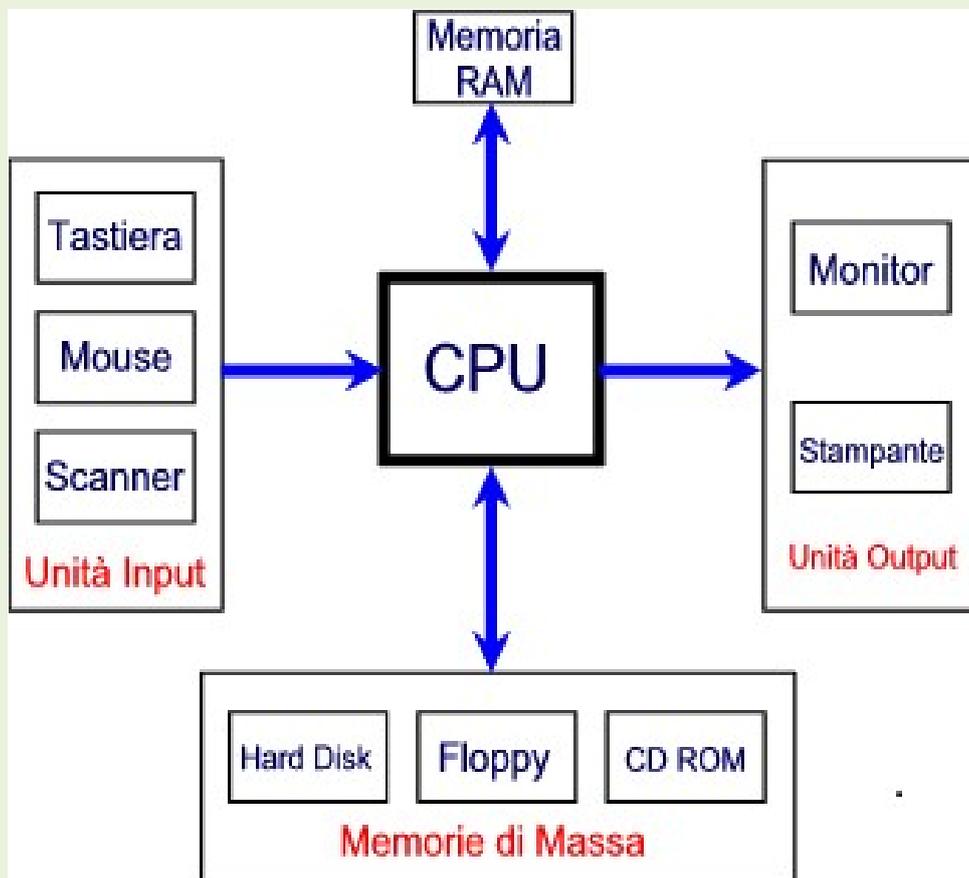
IDE è il connettore a cui si attaccano gli hard disk di vecchia generazione, oggi sostituito dai connettori **SATA**

USB è il gestore delle porte USB

BIOS è la piccola memoria che gestisce l'avvio del PC fornendo semplici funzioni di base

BUS si chiamano così i collegamenti tra i vari componenti. Sono specializzati per funzione in bus dati, bus indirizzi, bus controllo. Gli indirizzi servono ad individuare la posizione in cui immagazzinare / inviare un segnale, i controlli a definire quando inviare e ricevere le informazioni e la disponibilità al colloquio della CPU o della componente interpellata

.Nella figura successiva viene schematizzato un PC con le sue componenti principali che andremo a conoscere meglio.



LA CPU: ovvero "**central processing unit**" (in italiano unità di elaborazione centrale).

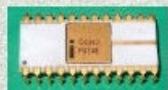
È il "cervello" di ogni PC, Tablet, telefonino etc, a seconda della complessità e della velocità della macchina gestisce tutto direttamente o tramite "schiavetti" a cui delega funzioni specializzate (ad esempio scheda video con processore dedicato, integrato per l'audio etc). Invia e riceve velocissimamente dati da e verso tutte le periferiche e le altre componenti.

Le CPU sono il pezzo più costoso e delicato del PC, richiedono molte cautele nel maneggiarle. Hanno

milioni di circuiti ultraminiaturizzati e, ricordando che le CPU più potenti sono a 64 bit, un numero considerevole di contatti (**pin**) che consente di usare “parole” molto lunghe. Le ultime generazioni sono CPU che sono a loro volta sono formate da 2, 4 o più CPU integrate che lavorano insieme. Un altro parametro da considerare è la “velocità” nel fare le operazioni ovvero la “frequenza”, misurata come detto precedentemente in multipli di Hertz.

Richiedono dei dissipatori di calore molto efficaci in quanto la concentrazione dei circuiti e la frequenza di uso producono molto calore.

Ecco alcuni esempi che danno un'idea della estrema complessità dei circuiti integrati in una CPU e della sua evoluzione nel tempo.

C4040	
	740 KHz 24 pin
D8086-2	
	8 MHz 40 pin
A80486DX-33	
	33 MHz 168 pin
PENTIUM IV	
	1400 / 600 MHz 478-pin



INTEL® i7-920
INTEL® CORE™ i7
SLEBJ COSTA RICA
Z. 66GHZ/8M1M, 80/08
3938A591

INTEL CORE i7

4 core

2,6 / 3,5 Ghz

731 milioni di transistors



1150 pin

UNITA' di INPUT (input= ingresso)

sono il modo in cui il PC comunica con l'esterno: quelle sempre presenti sono la tastiera e il mouse.

La **tastiera ha 105** tasti che rappresentano lettere maiuscole e minuscole, cifre e simboli speciali. Esistono inoltre 12 tasti speciali denominati F1 ... F12 a cui sono associate particolari funzioni (ad esempio F1 tradizionalmente è usata come Help=aiuto).

La tastiera riproduce quella di un macchina da scrivere ed ha un uso analogo.

Esiste un numero molto alto di tastiere, personalizzate con i caratteri delle varie lingue, inoltre per scopi

speciali vengono realizzate tastiere particolari.

Il **mouse** è stato introdotto dapprima dai sistemi APPLE e poi si è diffuso su tutti i PC. Ha due o tre tasti e (ma non sempre) una rotellina centrale che funge da terzo tasto. Funziona fissando una posizione sullo schermo del monitor e traducendo le coordinate del punto letto in informazioni che inviate alla CPU danno luogo ad azioni. Quando parleremo dei sistemi operativi studieremo più a fondo questi meccanismi.

Da poco si stanno diffondendo i cosiddetti “**touch screen**” (schermi sensibili al tocco) soprattutto nei tablet e negli smartphone. In pratica sono un incrocio tra il mouse e la tastiera combinandone le funzioni.



UNITA' di OUTPUT (output= uscita)

LA STAMPANTE

Le informazioni elaborate nel PC vengono visualizzate in due modi su un video (monitor, televisore etc) oppure stampate. Quest'ultimo modo è stato il primo utilizzato quando ancora i monitor non erano ancora diffusi. I primi centri di calcolo utilizzavano sia in ingresso che in uscita schede perforate che riportavano le informazioni codificate opportunamente.

Le stampanti sono di tre tipi:

ad aghi :sono le più vecchie ma sono ancora usate per i moduli con più copie. Utilizzano aghi sottilissimi che percuotono un nastro inchiostrato lasciando la traccia di un puntino. L'insieme di tanti puntini opportunamente combinati forma le lettere ma anche immagini. Normalmente hanno solo il colore nero.

A **getto d'inchiostro** (in inglese **inkjet**) tramite microscopici fori schizzano gocce di inchiostro di vari colori sulla carta formando caratteri e immagini.

Esistono stampanti col solo colore nero altre che hanno un serbatoio (cartuccia) per il nero ed uno per i tre colori base (verde, giallo e magenta) altre ancora hanno un serbatoio distinto per ogni colore.

Sono le più diffuse stante l'economicità, anche se il basso costo iniziale viene annullato dal costo delle cartucce di ricambio.

Laser: un raggio laser infrarosso viene modulato per formare ciò che si vuole stampare e viene inviato su un rullo ricoperto di materiale fotosensibile che si elettrizza nei punti dove viene colpito. La pagina passando sul rullo raccoglie della polvere nera o colorata (toner) che si ancora nei punti prescelti a causa dell'elettricità statica. Da ultimo la pagina viene scaldata su un rullo fusore, la polvere si fonde e si imprime indelebilmente sul foglio. La stampanti laser sono molto veloci.

MONITOR e VIDEO

E' il principale mezzo di output di un PC tramite esso si possono veder testi, immagini, filmati oltre ai messaggi di gestione del PC stesso.

Più che parlare però del monitor, del resto del tutto simile ad un normale televisore salvo il fatto di avere una miglior definizione ed una velocità di risposta più elevata, dobbiamo parlare di quello che è il vero deus ex machina del video: la **scheda grafica**.

Le prime schede grafiche erano monocromatiche successivamente si sono evolute su tre percorsi diversi ma integrati, il numero di colori che si possono mostrare, la velocità di esecuzione e la presenza di CPU integrate, ma sempre supervisionate dalla CPU principale, che prendono il nome di **GPU** (=graphic unit processor). Si è passati da due colori a sedici, 256, 64536... fino agli attuali 2^{32} colori. La creazione di immagini sempre più ad alta definizione che hanno bisogno di tempi relativamente lunghi per “manifestarsi” proprio per come sono costruiti i monitor, richiede che la scheda grafica sia equipaggiata con un sua memoria dedicata pena un rallentamento marcato di tutto il PC che deve rimanere in attesa che venga formata l'immagine sul monitor. Questo effetto permane ad esempio nei collegamenti internet lenti, quando un'immagine viene visualizzata riga per riga con tempi a volte di minuti mentre il PC sembra “impallato”. Ma non è tutto, ormai le immagini in movimento sono normalissime e la simulazione della realtà sempre più perfetta, quindi occorre anche fare una gestione delle ombre, della sovrapposizione delle figure, dei particolari che non sono visibili ma che devono essere presenti come “immagine nascosta” ma che al momento opportuno devono diventare visibili; tutto questo richiede una velocità di calcolo per le posizioni, la parti nascoste etc. che lasciate alla sola CPU rallenterebbero troppo il sistema da qui l'introduzione delle GPU che fanno solo questo; gestire le immagini. Questo ha portato a schede grafiche che nei modelli più evoluti hanno costi elevatissimi con RAM di diversi Gbyte di memoria dedicata e **bus dati** (=gruppi di conduttori che trasmettono in parallelo le parole di 0 e1) a 128 bit e 256 bit, ovvero messaggi formati da 128 e 256 simboli 0 o 1, nonché processori velocissimi dedicati.

Nei PC di fascia bassa la scheda grafica è integrata nella motherboard e condivide la memoria con le altre applicazioni, in quelli di maggiori prestazioni è installata tramite appositi connettori presenti sulla Motherboard chiamati **slot**.

Nei PC usati per l'elaborazione di filmati e fotografie o per i video – giochi esistono schede grafiche che sono costituite da due schede che lavorano insieme e che di fatto sono due calcolatori grafici con GPU a frequenze elevatissime e memorie di alcuni Gigabyte.

LA GESTIONE DELLE INFORMAZIONI

Abbiamo detto che quando arrivano più messaggi in contemporanea la CPU li deposita nella **memoria** in attesa di lavorarli, vediamo di capire il suo funzionamento e come è fatta.

Sostanzialmente una memoria è un insieme di circuiti integrati che ragionano sempre in termini di 0 e 1 sempre raggruppati in byte o multipli di byte, ma che possono mantenere il loro stato per tutto il tempo necessario. In altre parole immaginiamo di avere un casellario formato da 10 file di 8 caselle numerate da 1 a 10. Ogni tanto arrivano gruppi di 8 pratiche che devono essere lavorate insieme, se gli addetti non riescono ad espletare uno dei gruppi che arrivano, esso viene immagazzinato nel casellario nella prima riga libera, viene memorizzata la riga e la priorità del gruppo, al momento opportuno viene recuperata.

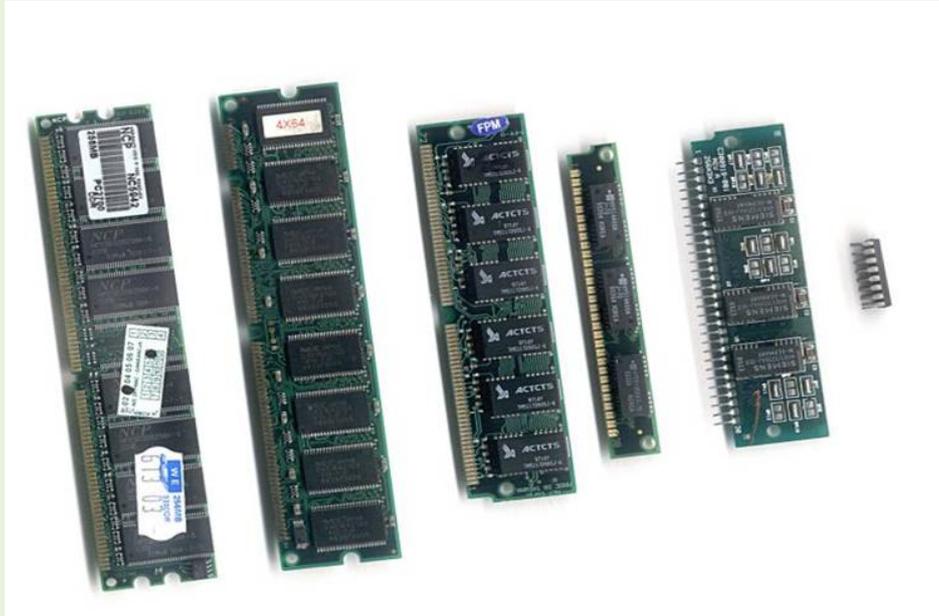
Possiamo perciò immaginare che una memoria sia un casellario formato da moltissime righe e da un numero di caselle per riga uguale alla lunghezza della “parola” che usa la mia CPU: 8 , 16, 32 , 64 bit ed oltre, ovvero “parole” di 1,2, 4, 8 byte. Si noti che i numeri sono sempre potenze di 2 cioè

$$2^3 = 8 \quad 2^4 = 16 \quad 2^5 = 32 \quad 2^6 = 64 \text{ etc.}$$

La memoria è indispensabile per il funzionamento di qualsiasi elaboratore per piccolo che sia

Vedremo nel capitolo sulla programmazione come la memoria interagisca con la CPU e le altre componenti del PC, per ora guardiamo come è fatta.

Nella fotografia sotto riportata abbiamo alcuni tipi di memorie.



Così come le

CPU si sono evolute nel tempo anche le memorie si sono dovute adattare per poter ospitare “parole” sempre più lunghe adottando non solo circuiti sempre più complessi ma anche più veloci nelle risposte dato che anche le CPU vanno a frequenze sempre più elevate.

Esistono vari tipi di memorie: quelle così dette “volatili” che appena si spegne l'apparato si azzerano e quelle statiche che mantengono i dati anche senza alimentazione. Queste ultime sono usate per memorizzare ad esempio le istruzioni di partenza del PC (**BIOS**)

Un po' di terminologia:

RAM (random access memory = memoria ad accesso casuale) sono quelle più diffuse e sono volatili. Si dicono ad accesso casuale perché le informazioni non sono immagazzinate occupando una dopo l'altra le “righe” ma occupandole in modo casuale purché libere (esistono comunque regole di occupazione che consentono di recuperare le informazioni in modo corretto). Hanno capacità da pochi megabyte alle odierne di alcuni gigabyte.

ROM (read only memory = memorie a sola lettura) sono organizzate come le RAM ma le locazioni di memoria contenenti le informazioni vengono “cristallizzate” e non sono modificabili (in gergo: sono state bruciate) sono usate per contenere dati o piccoli programmi che servono ad esempio per l'avvio del PC. Nei PC questi programmi sono contenuti nel cosiddetto **BIOS** (Basic Input-Output System = sistema di base per l'input e l'output). Una particolare ROM è la **EPROM** che può essere riscritta un certo numero di volte ma solo con apparecchi particolari non presenti nei PC. Hanno capacità molto contenute dell'ordine dei megabyte.

MEMORIE FLASH sono memorie statiche che però però sono leggibili e scrivibili in tempo reale, sono utilizzate in tutti i dispositivi portatili (tablet, smartphone, chiavette, schede SD etc) e hanno sostituito nei PC desktop (=scrivania) e nei portatili anche le ROM che contengono il BIOS in quanto l'evoluzione dei sistemi richiede aggiornamenti anche nei processi di input output.

Esistono dei particolari circuiti (gestori della memoria) che gestiscono l'ingresso e l'uscita dei dati e/o delle istruzioni da eseguire sempre rispettando la temporizzazione data dall'orologio di sistema.

Prima di passare oltre fissiamo qualche concetto:

1. la memoria è indispensabile per il funzionamento del PC
2. le memorie immagazzinano “parole” composte da uno o più byte in celle i cui indirizzi sono memorizzati in parti dedicate della RAM
3. le memorie sono volatili (RAM) o statiche (ROM, FLASH)
4. le memorie immagazzinano i dati dei programmi e le istruzioni da eseguire
5. le memorie sono lette e scritte sotto la supervisione del gestore della memoria a sua volta pilotato dalla CPU

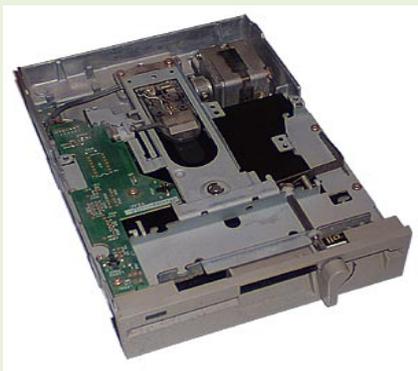
FLOPPY DISK

Il problema di trasportare e conservare i dati anche dopo lo spegnimento del PC ha trovato nel tempo alcune soluzioni:

1. **Floppy disk** : all'inizio la sua capacità era di soli 160 Kbyte sufficiente per i PC dei primi anni 80 ma divenuto presto inadeguato la sua capacità è salita sino a 2,5 Mbyte. Oggi è praticamente inutilizzato. Nei PC attuali non è più neanche presente il lettore (drive).

Serviva per salvare / utilizzare programmi e dati. In particolare agli inizi serviva per caricare il sistema operativo che avviava il PC stesso, il famoso **DOS**.

E' costituito da un disco di materiale plastico su cui è stato applicato uno strato ferromagnetico che, sempre nella logica 0 e 1, viene magnetizzato in due forme una che viene letta come 1 ed una che viene letta come 0. Questo disco è racchiuso in una custodia di protezione di cartoncino (quelli più vecchi) o di plastica. La custodia ha una finestrella attraverso cui si può leggere e scrivere sul disco interno tramite una testina magnetica analoga, come funzione, a quella di un registratore. Il disco viene inserito in un apposito lettore che lo fa girare molto velocemente. I dati sono “depositati” in cerchi concentrici tramite la testina che si muove radialmente. Le misure sono espresse in pollici essendo nati nel mondo anglosassone e sono 8”, 5” ¼, 3” 1/2



drive per floppy disk da 5¼ pollici



3,5 pollici floppy disk drive

floppy disk da 5¼ pollici



Un floppy da 3½ pollici

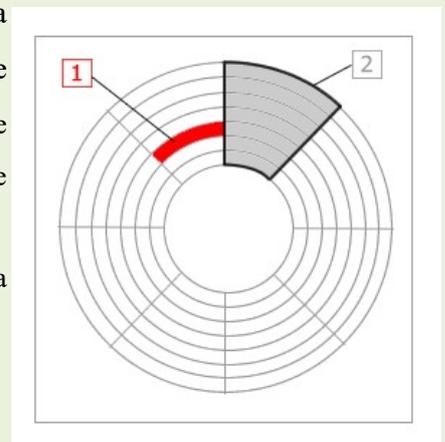
Da
sinistra
floppy da
8” 5” ¼
3” ½



La superficie del dischetto è organizzata in tracce e settori. Le tracce sono le piste circolari (1) del disco. Un settore comprende più tracce all'interno di una stessa sezione (2) del disco. I primi floppy disk della storia consentivano la registrazione dei dati soltanto su una faccia del disco.

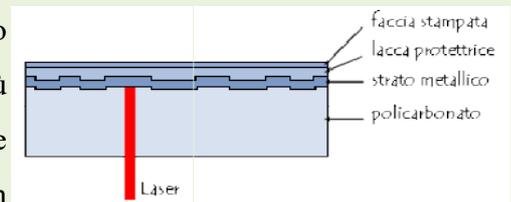
Successivamente sono stati prodotti floppy disk con doppia superficie magnetizzabile, in grado di registrare i dati su entrambe le facce del dischetto magnetico. In sostanza è come se avessimo tante caselle disposte una dietro l'altra sulle piste circolari le cui coordinate sono gli “indirizzi”

In sostanza è come se avessimo tante caselle disposte una dietro l'altra sulle piste circolari le cui coordinate sono gli “indirizzi”



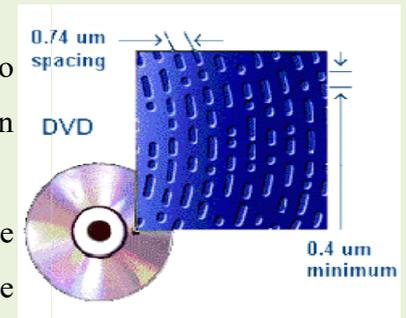
CD e DVD BLU-RAY

Il CD-ROM è un disco formato da tre strati. Il primo strato è formato da policarbonato, ed è lo strato più spesso che dà consistenza al CD. Su questo strato viene posto uno strato riflettente (di alluminio) ed un



terzo strato trasparente con funzioni protettive. I dati vengono scritti da un laser che fonde il supporto creando micro-avallamenti (=1) oppure lasciando la superficie piana non fusa (=0)

La superficie del CD-ROM, se esaminata con forte ingrandimento, si presenta formata da una traccia a spirale (dall'interno verso l'esterno), composta da una miriade di "pits",



(=fossi) e "lands" (spazi piani), la cui alternanza corrisponde agli stati 1 e 0 che il computer riconosce come informazioni attraverso la lettura effettuata sulla base della riflessione del raggio laser del lettore CD-ROM. Le informazioni contenute nei CD-ROM commerciali (dati, programmi, musica, ecc.) vengono "stampate" sul CD già in sede di fabbrica analogamente a quanto accadeva per i dischi in vinile.

Analogo funzionamento e costituzione hanno i DVD e i Bluray. La differenza sta nella maggior quantità di dati: i CD contengono 650 Megabyte mentre i DVD immagazzinano 4,7 Gigabyte. ed i Bluray fino a 25 Gigabyte. La crescita della capacità è dovuta allo sviluppo dei multimedia che richiedono quantità altissime di dati soprattutto per l'alta definizione.

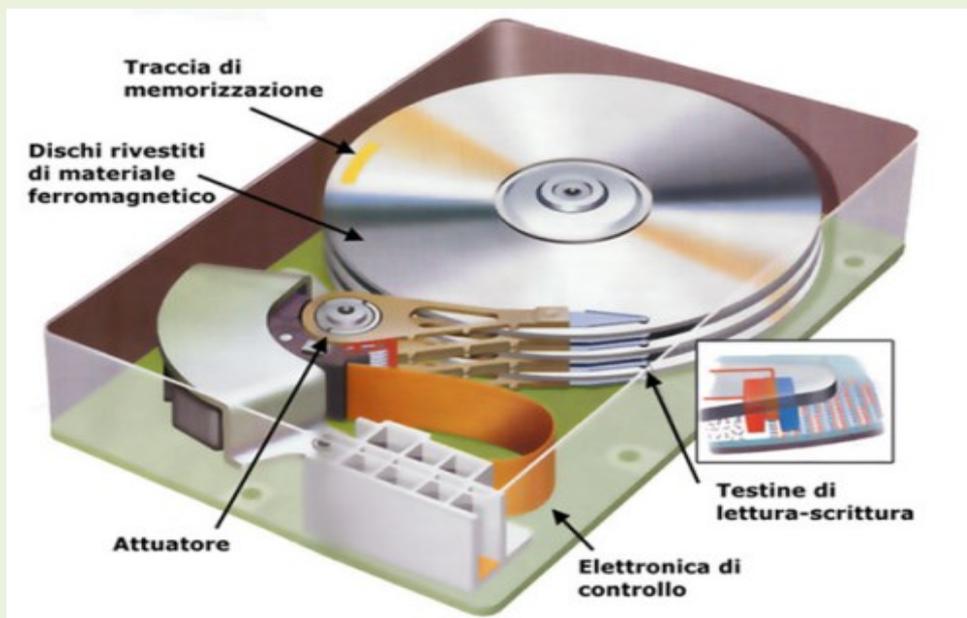
Di questi supporti esistono versioni che possono essere scritte una sola volta ed altre che possono essere cancellate e riscritte più volte.

2. **HARD DISK**

I PC, ormai lo abbiamo capito, richiedono quantità enormi di dati da conservare, caricare, modificare e soprattutto di elaborarli sempre più velocemente. A questo scopo sono preposti degli oggetti Hard Disk (abbreviato HD) o in italiano Disco Rigido.

Il disco rigido è costituito fondamentalmente da uno o più **piatti** in rapida rotazione, realizzati in **alluminio** o **vetro**, rivestiti di **materiale ferromagnetico** e da due **testine** di lettura / registrazione per ogni disco (una per faccia), le quali, durante il funzionamento "volano" alla distanza di poche decine di nanometri (=un milionesimo di millimetro) dalla superficie del disco leggendo o scrivendo i dati. La testina è tenuta sollevata dall'aria mossa dalla rotazione stessa dei dischi la cui frequenza o velocità di rotazione può superare i 15.000 giri al minuto.

Nome	Simbolo	Multiplo
kilobyte	kB	10^3
megabyte	MB	10^6
gigabyte	GB	10^9
terabyte	TB	10^{12}
petabyte	PB	10^{15}
exabyte	EB	10^{18}



La capacità degli HD è passata negli anni da una decina di Megabyte ai 2 Terabyte (=mille miliardi)

Il modo di scrivere i dati è analogo a quello dei floppy riportandoli su tracce e settori in caselline contigue, la differenza è che queste ultime sono moltissime.



un Hard disk multiplattello



Particolare delle testine

I BUS sono i conduttori che fanno comunicare tra loro le varie parti del computer.

Sono di tre tipi:

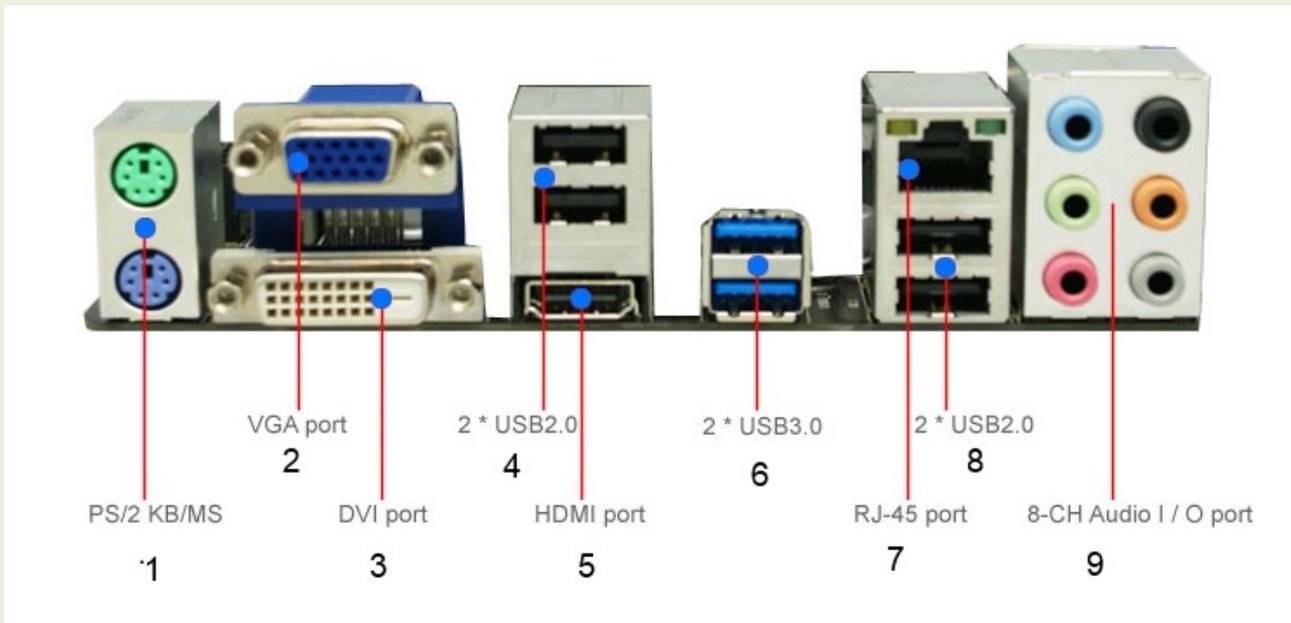
Bus dati è bidirezionale e trasporta le informazioni da un punto all'altro.

Bus indirizzi È (unidirezionale) e attraverso di esso la [CPU](#) decide in quale indirizzo andare a scrivere o a leggere informazioni; sia le celle di memoria ([RAM](#)) sia le periferiche di I/O (Input/Output) sono infatti divise in zone, ognuna delle quali ha un dato indirizzo. Dopo aver comunicato l'indirizzo tramite questo bus, la scrittura o lettura avviene normalmente tramite il bus dati. E' usabile in scrittura **solo** dalla [CPU](#) ed in lettura dagli altri componenti, in quanto tramite questo bus viene dato solo l'indirizzo della cella, che però è deciso dalla [CPU](#).

Bus controlli Il bus controlli è un insieme di collegamenti il cui scopo è coordinare le attività del sistema; tramite esso, la [CPU](#) può decidere quale componente deve scrivere sul bus dati in un determinato momento, quale indirizzo leggere sul bus indirizzi, quali celle di memoria devono scrivere e quali invece leggere, etc. Infatti la memoria e tutti gli altri componenti comunicano con la CPU attraverso un unico bus condiviso; questo significa che senza un controllo da parte della [CPU](#) si verrebbero a creare dei [conflitti](#)

CONNETTORI ESTERNI

I PC che di solito sono racchiusi in un contenitore con tutti i loro circuiti e schede varie hanno però necessità di vari contatti con il mondo esterno i connettori che servono a questo scopo sono in gergo chiamati **porte**. Vediamone sinteticamente aspetto e uso.



Porte 1 sono per il collegamento di mouse e tastiera

Porta 2 = VGA= collegamento del monitor (in questo caso sulla MB ma può essere su una scheda aggiuntiva nei PC più performanti)

Porta 3 =DVI= collegamento con i monitor digitali

Porte 4 , 6, 8 =USB servono per tutti gli “aggeggi” che usano USB per connettersi quali chiavette, HD esterni, stampanti, tastiere, mouse etc e quasi sempre forniscono loro anche l'alimentazione elettrica. E' importante non collegare ad una porta troppi aggeggi per evitare due effetti il rallentamento del sistema (ma non causa danni) e il **sovraccarico** di corrente con **possibili guasti al PC stesso non riparabili.**

Porta 5 = HDMI= uscita verso apparati ad alta definizione contengono sia segnali video che audio

Porta 7= Ethernet= connettore per la LAN e modem ADSL

Porte 9 = uscite audio a 8 canali (ovvero 7 altoparlanti dislocati intorno all'ascoltatore e 1 per i bassi)

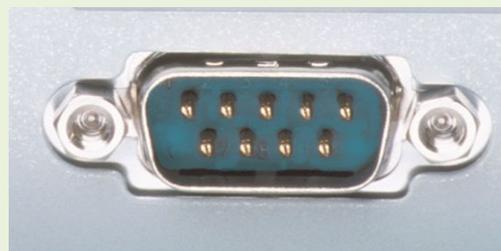
Nel nostro esempio abbiamo considerato un PC moderno con grafica ed audio integrati sulla MB non sempre inoltre sono presenti tutti questi connettori anche in base all'anzianità del PC.

Citiamo anche se ormai disusate due tipi di porte la **parallela** e la **seriale**. La parallela è ancora presente su molti PC non recentissimi e serviva per collegare le stampanti che oggi essendo diventate USB non la utilizzano più. La porta seriale (spesso ve ne erano due sulla MB) serviva per collegarsi con i modem che

utilizzavano la rete telefonica commutata o linee telefoniche dirette per comunicare con Internet o centri di calcolo, ormai non è più utilizzata.



Porta parallela



Porta seriale

AUDIO

Nei primi PC i suoni erano emessi da un altoparlantino incorporato nel contenitore del PC ed erano molto rudimentali anche se l'inventiva dei programmatori riusciva ad ottenere musicchette e suoni suggestivi in particolare nei videogiochi.

Musiche e parlato vennero successivamente che schede aggiuntive sia mono che stereo che col tempo divennero ad alta fedeltà. He siano sulla MB o su scheda dedicata, i suoni sono prodotti da processori dedicati che, analogamente alle schede grafiche, gestiscono in autonomia l'audio.

LA PROGRAMMAZIONE

Possiamo definire programmazione l'insieme di tutte quelle indicazioni che diamo alla CPU per ottenere il risultato che vogliamo.

Un primo livello di programmazione è quello che fa il produttore della CPU per darle certe funzioni che non andremo di certo ad esaminare sia perché sono segrete, sia perché non ci interessano per i nostri scopi. Noi considereremo la nostra CPU come una scatola che se riceve certe istruzioni e fa certe cose (ricordiamoci ancora una volta che stiamo parlando per simboli e che nella realtà si stanno gestendo circuiti elettronici anche se molto complicati). Ogni CPU ha sue caratteristiche in funzione dell'uso e ormai nel suo interno sono state assiemate anche funzioni che prima erano svolte da componenti specializzati come ad esempio le funzioni grafiche. Queste istruzioni sono rigidamente “cristallizzate” nelle CPU e non sono modificabili.

Per capire però come funziona la programmazione dobbiamo per prima cosa ricordarci quello che abbiamo detto all'inizio:

UN CALCOLATORE SA FARE SOLO QUELLO CHE GLI HANNO INSEGNATO E FATTO MEMORIZZARE, LO FA BENE E NON SBAGLIA MAI.

Il che ci porta a queste considerazioni:

se non gli dico tutto non farà quello che ho pensato.

se fa una cosa diversa da quella che avevo pensato ho sbagliato io

Cominciamo il nostro percorso con un linguaggio usabile da tastiera e che rappresenta simbolicamente le azioni elettroniche che vengono effettuate.

Questo linguaggio si chiama "**linguaggio macchina**" il nome dice già a cosa serve. E' usato da specialisti ed è il più veloce in assoluto e se, quando le CPU erano poco complesse, anche appassionati potevano usarlo, oggi è diventato materia pressoché riservata agli specialisti.

Cominciamo a dire che sostanzialmente utilizziamo due cose: le istruzioni e la memoria.

Pensiamo ad una calcolatrice tascabile (per inciso anche questa è un piccolo calcolatore) e di voler fare il seguente calcolo:

$$5 + 4 / 2$$

la sequenza corretta delle operazioni è:

tasto 5 > tasto (M +) > tasto 4 > tasto / > tasto 2 > tasto + > tasto (MRC) > tasto = > risultato (7)

quella scorretta sarebbe:

tasto 5 > tasto + > tasto 4 > tasto / > tasto 2 > tasto = > risultato (4,5)

Facciamo queste osservazioni.

1. Con una similitudine un po' azzardata possiamo dire che le funzioni fatte dai tasti sono le istruzioni "cristallizzate" nella CPU
2. La memoria ci serve per immagazzinare dei dati che dovremo usare poi

La logica del linguaggio macchina è più o meno così (attenzione ricordarsi che tutto è temporizzato):

la CPU riceve dalla tastiera il byte che nel codice ha significato di 5

5 viene messo in memoria

viene messo in memoria l'indirizzo della cella che occupa

la CPU riceve dalla tastiera il byte che nel codice ha significato di +

+ viene messo nella memoria delle istruzioni da fare

la CPU riceve dalla tastiera il byte che nel codice ha significato di 4

4 viene messo in memoria

viene messo in memoria l'indirizzo della cella che occupa

la CPU riceve dalla tastiera il byte che nel codice ha significato di /

/ viene messo nella memoria delle istruzioni da fare

la CPU riceve dalla tastiera il byte che nel codice ha significato di 2

2 viene messo in memoria

viene messo in memoria l'indirizzo della cella che occupa

la CPU riceve dalla tastiera il byte che nel codice ha significato di =

= viene messo nella memoria delle istruzioni da fare

La CPU capisce che deve dare il risultato e valuta quale operazione fare per prima (la regola è che si fanno prima divisioni e moltiplicazioni)

la CPU legge l'indirizzo del dividendo (4)

la CPU legge l'indirizzo del divisore (2)

la CPU esegue la divisione

la CPU pone il risultato in memoria

la CPU legge la seconda operazione da fare (+)

la CPU legge l'indirizzo del primo addendo (5)

la CPU esegue l'addizione tra il primo risultato ottenuto e il 5

la CPU emette il risultato

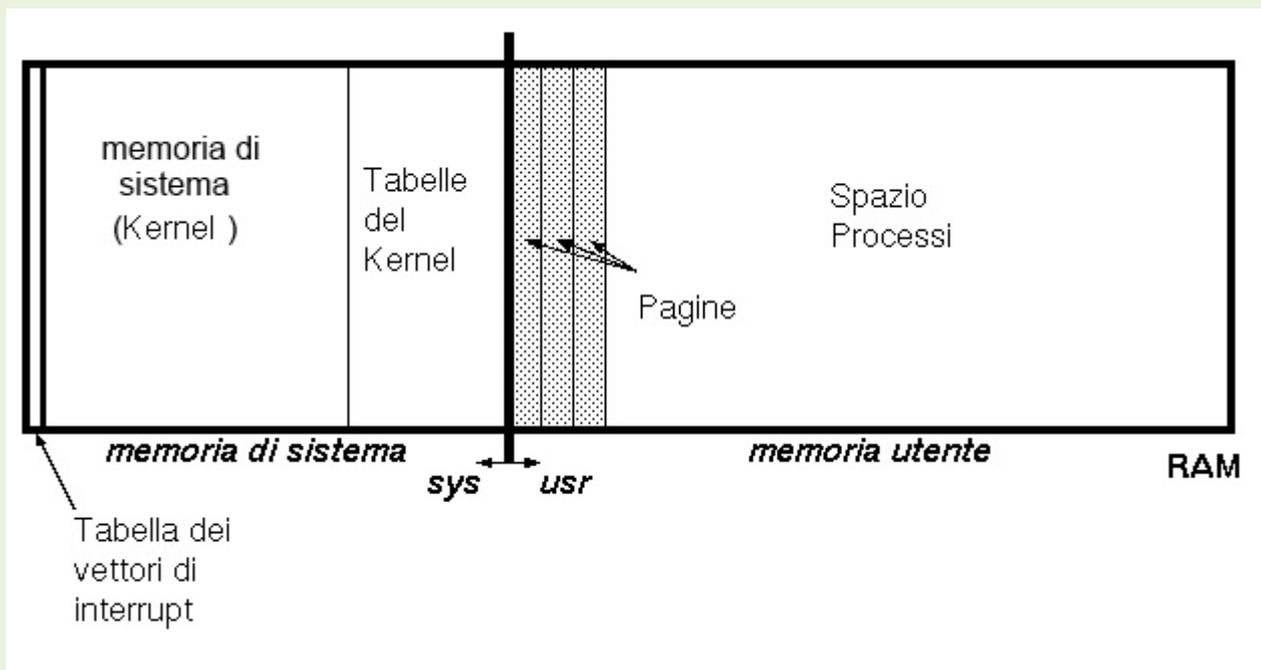
la CPU libera la memoria e attende il prossimo compito.

I termini del linguaggio non sono ovviamente questi e le operazioni sono più dettagliate però la logica è questa.

A questo punto vediamo di chiarire alcuni concetti:

abbiamo detto che la memoria è come un grande casellario formato da righe e colonne se ogni riga la codifichiamo con un numero, ovviamente in binario, e teniamo conto che noi scriviamo / leggiamo solo parole di lunghezza fissa (ad, esempio 32 bit) quel numero ci consente di identificare dove abbiamo messo quella parola che a sua volta può essere un dato o il nome di un' istruzione da fare: questo numero si chiama **indirizzo**.

In un mondo così complesso serve molto ordine per cui la memoria viene idealmente divisa in settori specializzati per uso, più in dettaglio:



La memoria centrale (*core memory*) e' il nome dato alla RAM di sistema. E' suddivisa in:

- **memoria di sistema** - anche chiamata area del **kernel**, che e' sempre residente in memoria a indirizzi noti, viene caricata all'avvio e contiene il sistema operativo
- **memoria utente** - in continuo stato di cambiamento, che contiene le pagine attive dei processi in esecuzione

La memoria di sistema e' suddivisa in:

- la **tabella dei vettori di interrupt**, una tabella di indirizzi di programmi del kernel che gestiscono i device drivers (programmi specializzati che nel linguaggio della CPU fanno funzionare le varie componenti, scheda grafica, audio, stampante etc). Essi vengono caricati in memoria all'avvio del pc oppure quando lancio un programma, di norma sono "proprietary" (cioè del costruttore del dispositivo). Ad esempio viene caricato il driver della tastiera della lingua scelta, della scheda video etc.
- la regione di testo del **kernel**, con il codice binario eseguibile del kernel
- le **tabelle del kernel**, che equivalgono alle regioni di dati e di stack (è l'elenco delle istruzioni da eseguire in sequenza) del kernel

La memoria utente e' suddivisa in **pagine**, tipicamente di 4 Kbyte ciascuna. Queste sono occupate da processi utente o da dati.

Un **interrupt** (= interruzione) è il segnale che manda un componente (ad esempio Hard disk, memoria.....) per ottenere "udienza" dalla CPU.

Il linguaggio macchina è un linguaggio "di basso livello".

Come detto, il linguaggio macchina è diventato molto complesso per cui nell'uso comune si usano linguaggi di "alto livello", i più comuni sono Basic, C, Java, Pascal,. Di "alto livello" significa che un termine del linguaggio (ad esempio Basic) corrisponde a diverse operazioni di dettaglio del linguaggio macchina. Nell'esempio della calcolatrice il tasto "=" fa alcune operazioni sopra descritte ma usa un solo comando "=" appunto.

Con **istruzione** si intende un comando dato alla macchina descritto nel **programma**, che è l'insieme delle istruzioni necessarie per conseguire lo scopo prefisso, ad esempio un editor di testi ha lo scopo di gestire documenti di testo, stamparli etc.

In pratica possiamo immaginare che la programmazione sia articolata in più livelli in cui il più basso è contenuto nel più alto, più precisamente:

nel processore	le istruzioni sono “cristallizzate” e definite dal costruttore
nel linguaggio macchina	le istruzioni vengono tradotte in parole convenzionali che si traducono in comandi fisici
nel linguaggio di alto livello	le istruzioni sintetizzano inglobandole istruzioni più semplici che danno perciò corso a piu' comandi in sequenza che vengono convertiti nel linguaggio “cristallizzato”

Fissiamo qualche concetto:

1. i processori incorporano le istruzioni per il funzionamento previsto e non sono modificabili
2. posso programmare un processore con linguaggio a basso od alto livello
3. un driver è un programmino che fa funzionare una componente del PC
4. Appositi indirizzi dividono la RAM in zone specializzate utilizzate dal sistema operativo (kernel), dai driver e dai programmi utente.
5. Un interrupt è la richiesta di udienza che un componente fa alla CPU

IL SISTEMA OPERATIVO

Il sistema operativo è un insieme di programmi ad alto livello che utilizzano una interfaccia a basso livello (**KERNEL**) che colloquia direttamente con la CPU . E' il primo insieme di programmi che viene caricato dal BIOS ed è essenziale per l'uso del PC,

I sistemi operativi sono andati crescendo di completezza di pari passo con l'accrescersi delle potenzialità delle CPU. I primi si chiamavano DOS e consentivano solo alcune funzioni di base: scrivere brevi programmi per personalizzare l'avvio, caricare driver, stampare, gestire floppy disk e hard disk, gestire le porte di comunicazione del modem, caricare la tastiera etc.

L'Apple ad un certo punto cominciò ad usare il mouse che divenne poi uno standard in ogni sistema operativo.

Va detto che il sistema operativo è strettamente legato al processore della macchina in cui è installato proprio perché al suo interno c'è l'interfaccia a basso livello che colloquia con la CPU. Non è pertanto possibile utilizzare un sistema operativo su una CPU diversa a meno di riscrivere il kernel; questo spiega perché a volte programmi scritti per una versione di Windows non funzionano più su una successiva o perché quando devo aggiornare un driver di una scheda grafica devo stare attento a scegliere proprio quello adatto al mio

sistema operativo.

I sistemi operativi più diffusi sono Unix, Mac OS X, Microsoft Windows (nelle versioni XP, 7, 8,10), Linux a cui si aggiungono quelli per sistemi portatili come IOS per Ipad, Android e la miriade di altri utilizzati negli SmartPhone.

Una curiosità: moltissimi appassionati o aziende si sono cimentati con nuovi sistemi operativi per diletto o per necessità ma noi non ne parleremo. Noi considereremo solo i sistemi operativi per PC e di questi solo Windows 10, comunque va sottolineato che queste funzioni sono presenti in ogni sistema operativo anche se accessibili in modi diversi..

Vediamo le funzioni principali.

Funzioni di input: tastiera, mouse tramite appositi driver incorporati nel kernel.

Funzioni di output: video, stampanti

Funzione di conservazione / gestione delle informazioni: gestione file,hard disk

Funzioni di gestione del computer: pannello di controllo, strumenti di amministrazione

Funzioni di colloquio con altri PC: LAN e Internet

Funzioni di elaborazione testi: wordpad

Funzioni di grafica e immagine: visualizzatore immagini e Paint

Queste funzioni ed altre ancora sono gestite da programmi di alto livello che si appoggiano su programmi a basso livello incorporati nel kernel.

Per cominciare dobbiamo parlare di come i programmi ed i dati vengono gestiti.

Innanzitutto qualche definizione:

Prima di usare il nostro PC dobbiamo fare un ultimo passo: capire come è organizzato un HD.

In primo luogo dobbiamo definire quella che si chiama “**partizione**”.

Una **partizione** indica una suddivisione fittizia di un'unità di memorizzazione fisica (per esempio un hard disk) che viene vista dal sistema operativo come un altro HD fisico (partizione logica). Le singole unità logiche vengono viste dal sistema operativo come unità separate, cioè come se fossero dischi diversi: possono essere formattate e gestite in modo indipendente. Su ogni disco rigido è sempre presente almeno una partizione.

ATTENZIONE: cancellare una partizione cancella tutto

Formattazione: è il processo con cui vengono create tracce e settori nella partizione. Un disco senza formattazione non può funzionare. Il tipo di formattazione è proprio di ogni sistema operativo. La formattazione in pratica costruisce il “casellario” vuoto e attribuisce ad ogni cella un indirizzo

File (= dossier, schedario): è un insieme di dati binari definito dal numero di byte che contiene. Il suo nome è creato dal programmatore o dall'utente e non può superare i 260 caratteri, inoltre possiede una cosiddetta

“**estensione**” lunga 3 caratteri e separata del nome da un punto. Essa specifica il tipo di dati presenti nel file stesso: ad esempio **.txt** si riferisce ad un documento di testo, **.doc** si riferisce ad un documento di testo con immagini e struttura definita dall'utente, **.xls** ad un foglio di calcolo, **.bpm** e **.jpg**, si riferiscono a file grafici ed immagini, **.exe** e **.com** si riferiscono a programmi eseguibili. Le estensioni sono parecchie, legate ai vari tipi di programmi. L'estensione viene letta automaticamente dal sistema operativo ed associata al programma che ha generato il file.

Cartella: è un contenitore di file che si riferiscono ad un certo argomento o programma. Hanno nomi senza estensione

Sottocartella: è una cartella contenuta in un'altra cartella. Ci possono essere più livelli di sottocartelle una dentro l'altra, ciascuna col suo nome.

Directory (= indice): è la struttura ad albero che evidenzia le cartelle, le sottocartelle ed i file.

FILE SYSTEM è il modo in cui i file sono organizzati sui dischi. Si riferisce cioè a come sono memorizzati i file negli Hard Disk e negli altri dispositivi di memorizzazione.

In pratica serve a creare una mappa degli indirizzi di dove sono collocati i file, da dove iniziano e finiscono, identificandoli sulle tracce e settori associati alla posizione fisica sul disco.

Un esempio di directory:

