

# IL LINGUAGGIO BINARIO

**Ing. Daniele Corti**



copyright

all rights reserved

Copyright © Ing. Daniele Corti 2013

[www.ingdanielecorti.it](http://www.ingdanielecorti.it)

Tutti i diritti sono riservati a norma di legge e a norma delle convenzioni internazionali.

Ver.1.0

## **PREREQUISITI**

- ✓ Dato e informazione.

## **OBIETTIVI**

- ✓ Conoscere il linguaggio binario dei calcolatori.
- ✓ Conoscere il bit, il Byte e i loro multipli.

## **ARGOMENTI**

- ✓ Elettronica analogica e digitale.
- ✓ L'elettronica dei calcolatori e i transistori.
- ✓ Il linguaggio dei calcolatori: il codice binario.
- ✓ Bit e Byte.
- ✓ Significato fisico del Byte.
- ✓ Esperienza pratica.
- ✓ Unità di misura e multipli.

## CAP 2 – IL LINGUAGGIO BINARIO

### INTRODUZIONE

Così come il metro viene utilizzato per misurare le distanze ed il grammo permette di stabilire il peso di un corpo, allo stesso modo anche il mondo dell'informatica ha **una propria unità di misura**, impiegata nello specifico per misurare **la quantità di informazioni**.

E proprio come il metro dispone di vari multipli (fra cui il più usato è sicuramente il chilometro), allo stesso anche l'unità di misura in informatica avrà bisogno di unità multiple che consentano di esprimere in forma più compatta quantità molto elevate: è sicuramente più pratico utilizzare "100 Kilometri" piuttosto che "100.000 metri".

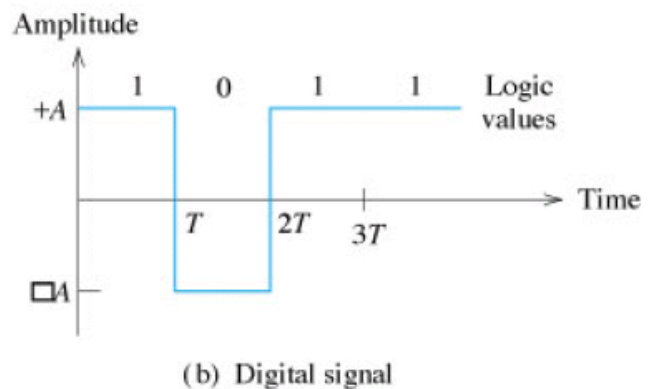
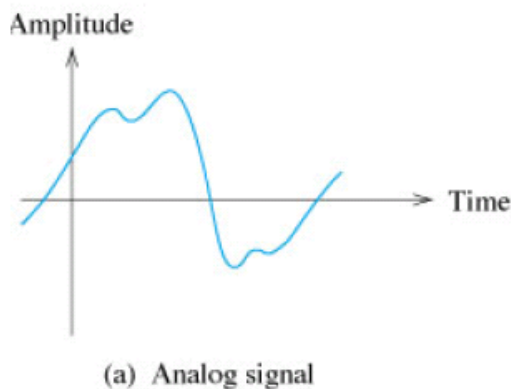
Ma le analogie non finiscono qui: proprio come esiste il "Kilometro al secondo" per trattare le velocità, allo stesso modo il mondo dei computer ha avuto bisogno di realizzare unità di misura composte per trattare problemi simili.

### ELETTRONICA ANALOGICA E DIGITALE

L'elettronica è una scienza che studia i fenomeni elettrici e, può essere classificata in analogica e digitale.

Nei sistemi analogici, i fenomeni sono rappresentabili con segnali continui e hanno quindi le seguenti caratteristiche:

- **l'ampiezza varia con continuità in un intervallo continuo.** Il segnale è continuo sia in ampiezze che nello spazio e/o nel tempo, come si evince dal grafico sinistro della seguente figura:



**Un segnale analogico è un segnale che può assumere infiniti valori di ampiezza in un intervallo di tempo e/o spazio.**

I segnali analogici sono così denominati poiché nel rappresentare una grandezza, come il suono per esempio, variano seguendo l'andamento di quest'ultima, ovvero << in analogia >> con essa.

I **segnali digitali** (dall'inglese *digit*, cifra), o **numerici**, sono segnali che possono assumere solo un numero limitato di valori. Un caso particolare si ha quando i valori possibili sono solo due: in tal caso si parla di segnale digitale **binario**.

Il segnale digitale ha le seguenti caratteristiche:

- **Assume solo un insieme finito di valori:** è discreto in ampiezze.
- **È una sequenza di numeri:** è discreto nel tempo (e/o nello spazio).

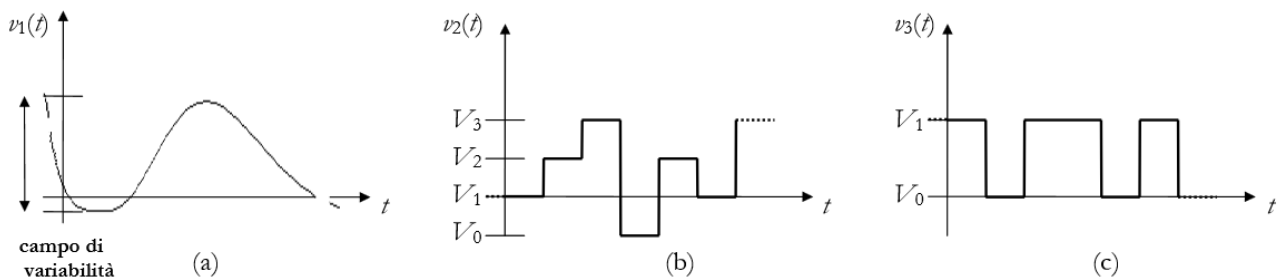


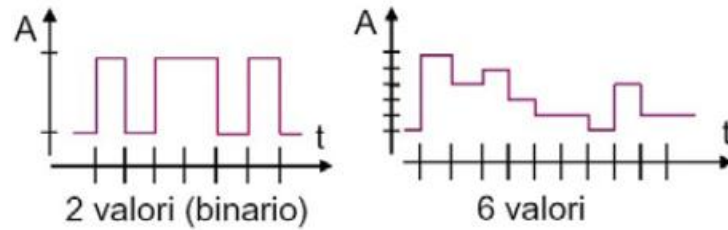
Fig (a) segnale analogico; (b) segnale digitale; (c) segnale digitale binario

Un segnale digitale può essere così rappresentato:



### Esempio

Nel disegno seguente possiamo osservare due segnali digitali; in quello di sinistra l'ampiezza assume solo due valori (**segnale binario**), mentre quello di destra 6 valori.



In un segnale **digitale** l'ampiezza può assumere soltanto due valori.

Nei sistemi digitali, le informazioni sono rappresentate da **variabili logiche** che possono assumere soltanto due valori: 1 (high, true, on) o 0 (low, false, off).

### **Il computer può lavorare soltanto con grandezze di tipo digitale e finite.**

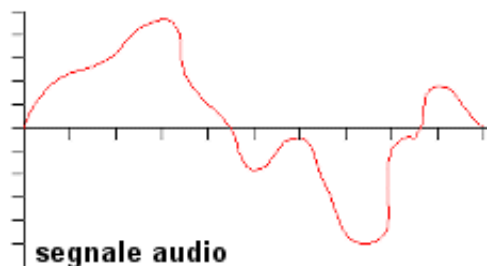
I fenomeni fisici per loro natura sono rappresentabili con segnali continui: il suono, la temperatura, la pressione, etc. se volessimo rappresentare queste informazioni analogiche al computer dovremmo trasformarle al fine di consentire al computer stesso di trattarle. Trasformare le informazioni significa passare dal mondo analogico al mondo digitale.

### **Codifica: trasformazione di un'informazione da analogica a digitale**

La codifica consente di rappresentare al computer le informazioni sotto forma di sequenze binarie.

#### **Esempio**

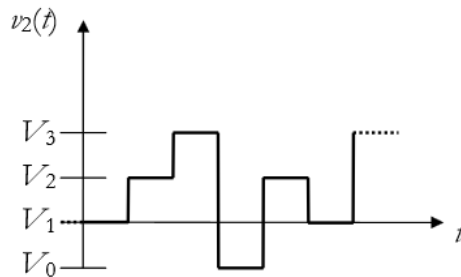
Il segnale audio è per sua natura un segnale analogico, ovvero un segnale che varia in modo continuo nel tempo. Se volessimo riporre su un diagramma la variazione dell'intensità sonora nel tempo, otterremmo un andamento di questo tipo:



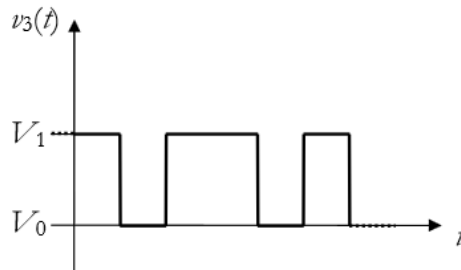
La caratteristica che rende questo segnale continuo è legata alla ampiezza che assume un'infinità di valori passando da un valore minimo ad uno massimo.

Se volessimo rappresentare il segnale audio con un segnale digitale, al fine da poterlo gestire con un calcolatore, dovremmo effettuare una serie di operazioni di trasformazioni (conversione dall'analogico al digitale) che sono argomento alla base della **teoria dell'informazione**.

I segnali digitali sono così definiti in quanto idonei a rappresentare sequenze di cifre associate ai possibili livelli. Il segnale della seguente figura può essere interpretato come la sequenza 1 – 2 – 3 – 0 – 2 - ...



Il segnale invece della seguente figura può essere interpretato come la sequenza 1 – 0 – 1 – 1 – 0 - ...



Le cui cifre, per la caratteristica che hanno di assumere solo due valori, sono dette **cifre binarie** o **bit**.

### Esercizio

Ricercare nella rete la definizione dei termini analogico e digitale proponendo degli esempi di segnali che rappresentano grandezze fisiche.

## L'ELETTRONICA DEI CALCOLATORI

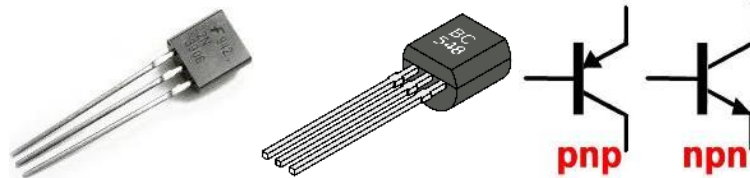
I sistemi di elaborazione sono realizzati con tecnologia digitale: le informazioni sono rappresentate mediante segnali elettrici a 2 valori di tensione ( $V_{LOW}$ ,  $V_{HIGH}$ ) oppure (0, 1).

Il computer è formato da un complesso di circuiti elettrici-elettronici e l'unica cosa che è in grado di riconoscere è lo stato elettrico: circuito chiuso (passa corrente) o circuito aperto (non passa corrente) dei suoi componenti.

Per questo motivo, l'unità di memorizzazione (e, in generale, di rappresentazione delle informazioni) è il **bit**.

## TRANSISTOR

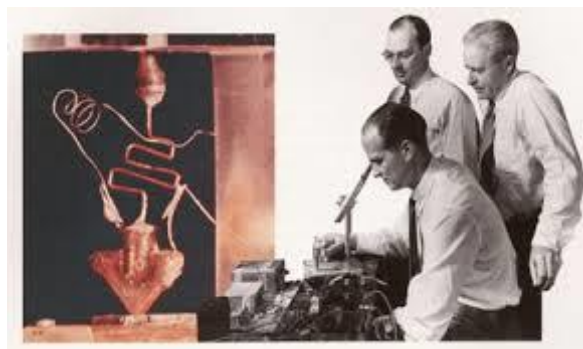
Nel campo digitale, il dispositivo elettronico di base che consente il salvataggio dello stato logico 0 o 1 (un bit) è detto transistor. In elettronica digitale il transistor è utilizzato come interruttore, mentre in elettronica analogica come amplificatore di segnale.



### 1947: Invention of the Transistor

The first transistor was invented at Bell Laboratories on December 16, 1947 by William Shockley (seated at Brattain's laboratory bench), John Bardeen (left) and Walter Brattain (right). This was perhaps the most important electronics event of the 20th century, as it later made possible the integrated circuit and microprocessor that are the basis of modern electronics. Prior to the transistor the only alternative to its current regulation and switching functions (TRANSfer resISTOR) was the vacuum tube, which could only be miniaturized to a certain extent, and wasted a lot of energy in the form of heat. Although video was possible with vacuum tube equipment, as was the case with the Ampex VRX-1000, without the transistor video products would never have gotten very small.

The picture on the left above shows the first point contact transistor built by Walter Brattain. It consisted of a plastic triangle lightly suspended above a germanium crystal which itself was sitting on a metal plate attached to a voltage source. A strip of gold was wrapped around the point of the triangle with a tiny gap cut into the gold at the precise point it came in contact with the germanium crystal. The germanium acted as a semiconductor so that a small electric current entering on one side of the gold strip came out the other side as a proportionately amplified current.



### Esercizio

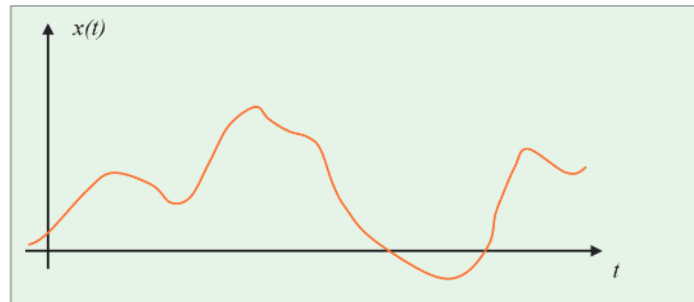
Ricerca nella rete la definizione, la storia e gli inventori del transistor.

### Esercizio

Con quale elemento chimico si producono i transistor e in quale zona lo si ricava.

### Campionamento

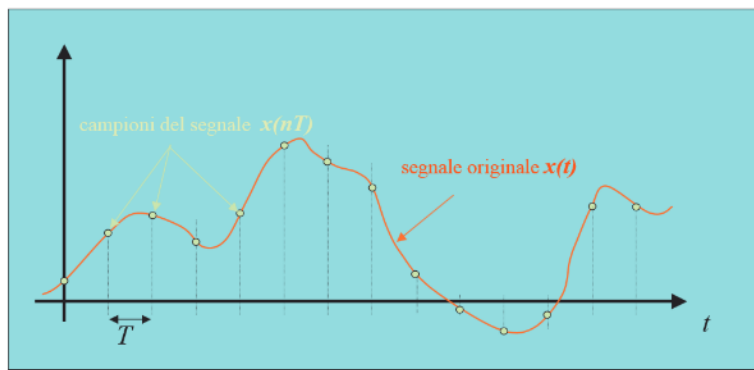
Sia dato il seguente segnale continuo sia nel tempo sia nelle ampiezze:



La rappresentazione di un segnale continuo con un segnale digitale richiede di discretizzare sia il tempo sia le ampiezze.

Il **campionamento** è una tecnica di conversione di un segnale continuo in un **segnale discreto nel tempo**.

Campionare quindi significa considerare del segnale continuo originale solo quelle ampiezze corrispondenti a determinati istanti di tempo discreti (per esempio 0 sec, 1 sec, 2 sec, 3 sec, etc):



### Esempio

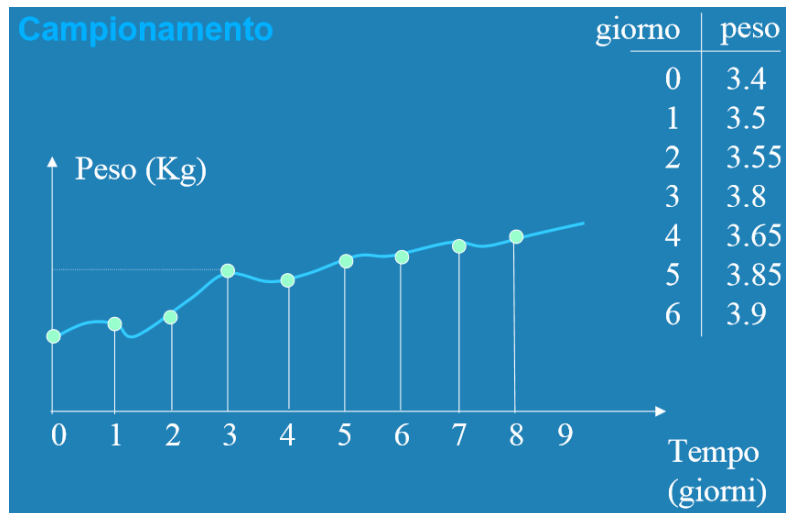
Sappiamo che nel mondo reale tutti i segnali sono analogici. Consideriamo ad esempio il peso di un neonato.



Si tratta ovviamente di una grandezza continua nel tempo in quanto ad ogni istante temporale il bambino avrà un peso ben preciso.



Tuttavia, il peso del bambino viene misurato una volta al giorno, ad esempio alle 20:00 di ogni giorno. Ogni valore di peso misurato si chiama **campione** e l'operazione di pesatura giornaliera prende il nome di **campionamento**.



In generale, campionare un segnale è un metodo per registrare un valore istantaneo di quel segnale.

Possiamo osservare nel nostro esempio che i rilievi del peso sono stati fatti a intervalli di tempo regolari. Possiamo quindi definire la seguente grandezza:

**$f_c$  = frequenza di campionamento**

**La frequenza di campionamento indica il numero di campioni registrati al secondo.**

**La frequenza di campionamento si misura in Hz [1/sec]**

Per riprodurre digitalmente un segnale analogico la frequenza di campionamento deve essere sufficientemente elevata.

### Esercizio

Sapendo che si fanno 150 passi in un minuto, qual è il valore della frequenza.

### Esempio

Si vuole studiare un fenomeno fisico di cui sono note le seguenti informazioni:

- Il segnale continuo rappresenta un suono.
- Il segnale ha una durata di 5 secondi.
- Il segnale viene campionato con frequenza di campionamento pari a 30 Hz.
- Ogni campione occupa 6 Byte.

Determinare quanti Byte occupa tale segnale campionato.

Risoluzione

Se  $f_c = 30$  Hz allora in un 1 secondo ho 30 campioni.

In 5 secondi avrò  $30 * 5 = 150$  campioni.

Se un campione occupa 6 Byte allora 150 campioni occupano  $6 * 150 = 900$  Byte.

## IL LINGUAGGIO DEI CALCOLATORI: IL CODICE BINARIO

Il linguaggio di un calcolatore, detto **linguaggio macchina**, è un codice binario, formato cioè dai due soli simboli 1 e 0 che rappresentano rispettivamente il circuito chiuso e quello aperto. Ai simboli 0 e 1 è stato dato il nome di **bit** (**B**inary **d**igIT, cifra binaria). Parliamo di stati logici 0 (basso) e 1 (alto).

Il bit è l'elemento minimo di memoria.



Il bit si può trovare solo in due possibili stati che convenzionalmente sono rappresentati con le cifre 0 e 1.

Dominio di variazione del bit:  $\{0,1\}$ .

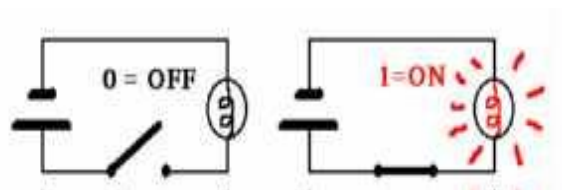
A livello elettronico questi due stati logici (0 e 1) corrispondono a “circuito aperto” (assenza di corrente elettrica) e a “circuito chiuso” (presenza di corrente elettrica).

## BIT (b)

Il linguaggio naturale utilizzato dall'uomo è molto diverso dal linguaggio binario impiegato da un computer. Il computer è una macchina che utilizza un codice binario, formato cioè dai due soli simboli 1 e 0 che rappresentano rispettivamente il circuito chiuso e quello aperto. Ai simboli 0 e 1 è stato dato il nome di **bit** (da **binary digit**, cifra binaria). Parliamo di stati logici 0 (basso) e 1 (alto). Il bit rappresenta l'unità elementare di informazione, una scelta tra "sì" e "no", tra "vero" e "falso", tra "acceso" e "spento", due risposte che possono essere associate ai valori "0" e "1".



Il funzionamento dei circuiti elettrici di tutti i calcolatori moderni è basato su due stati elementari: la presenza oppure l'assenza di un segnale elettrico.



Il bit può assumere 2 valori:

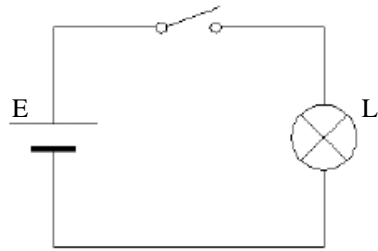


A seconda della posizione il bit assume un peso differente e viene moltiplicato per potenze di 2 crescenti, partendo dalla posizione a destra (**bit meno significativo**):

- 1<sup>a</sup> posizione: la cifra viene moltiplicata per 1 ( $2^0$ ).
- 2<sup>a</sup> posizione: la cifra viene moltiplicata per 2 ( $2^1$ ).
- 3<sup>a</sup> posizione: la cifra viene moltiplicata per 4 ( $2^2$ ).
- Etc.

### Esempio 1

Si vuole codificare (tradurre, trasformare) l'informazione naturale lampada accesa o lampada spenta in codice binario.



Codice Binario	Interruttore	Lampada L
0	Aperto	Spenta
1	Chiuso	Accesa

Questa situazione può essere rappresentata attraverso il seguente circuito elettrico utilizzato per l'accensione di una lampada (L) mediante la chiusura di un interruttore; il circuito viene alimentato da una pila E. quando l'interruttore è aperto (stato logico 0) la lampada è spenta; quando l'interruttore viene chiuso (stato logico 1) la lampada si accende.

Per codificare l'informazione "Lampada Accesa" o "Lampada spenta" ho, quindi, bisogno di un bit.

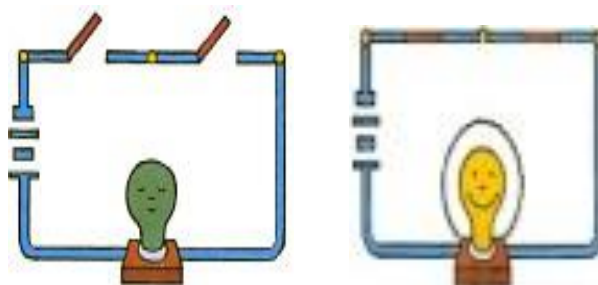
Dal punto di vista logico digitale questo circuito è equivalente ad un dispositivo elettronico digitale detto transistor. Il transistor è utilizzato per salvare un bit, cioè lo stato logico 0 o lo stato logico 1. Il transistor è la più piccola unità fisica per il salvataggio delle informazioni.

### Esercizio

Rappresentare un segnale digitale nel tempo che rappresenti lo stato di intermittenza di una lampada.

### Esempio 2

Studiamo ora la situazione di due lampade comandate da due interruttori indipendenti.



Codice Binario	Lampada L1	Lampada L2
0 0	Spenta	Spenta
0 1	Spenta	Accesa
1 0	Accesa	Spenta
1 1	Accesa	Accesa

In questo caso per poter codificare le quattro possibili combinazioni ho bisogno di due bit.

In generale:

$$C \leq 2^n$$

Dove:

C = Numero di combinazioni

n = numero\_bit

NB:  $C \leq 2^n \rightarrow n \geq \log_2 C \rightarrow n_{\min} = \log_2 C$

### Esempio 3

Se dobbiamo discriminare fra 256 possibili situazioni differenti, abbiamo bisogno di almeno

$\log_2 C = \log_2 256 = \log_2 2^8 = 8$  bit.

NB Per risolvere la disequazione si può utilizzare un metodo pratico basato su tentativi:

Proviamo per  $n=0$  e trovo che la disequazione  $256 \leq 2^0 = 1$  non è vera.

Proviamo per  $n=1$  e trovo che la disequazione  $256 \leq 2^1 = 2$  non è vera.

Proviamo per  $n=2$  e trovo che la disequazione  $256 \leq 2^2 = 4$  non è vera.

...

Proviamo per  $n=8$  e trovo che la disequazione  $256 \leq 2^8 = 256$  è vera e quindi il risultato è  $n_{\min} = 8$ .

In informatica ed in teoria dell'informazione, la parola bit ha due significati molto diversi, a seconda del contesto in cui rispettivamente la si usa:

- un bit è l'**unità di misura dell'informazione** (dall'inglese "binary unit"), definita come la quantità minima di informazione che serve a discernere tra due possibili alternative equiprobabili.
- un bit è una **cifra binaria**, (in inglese "binary digit") ovvero uno dei due simboli del sistema numerico binario, classicamente chiamati zero (0) e uno (1).

A seconda del contesto la parola bit può avere significati diversi:

- Il bit come quantità di informazione: in questo contesto il bit rappresenta l'unità di misura dell'informazione (**Binary uniT**), definita come la quantità minima di informazione che server a discernere tra due possibili alternative equiprobabili.
- Il bit come cifra binaria (**Binary digiT**): in questo contesto il bit rappresenta una cifra binaria (come visto prima), l'unità di definizione di uno stato logico. È anche definito come l'unità elementare dell'informazione trattata da un elaboratore. La rappresentazione logica del bit avviene attraverso i soli valori (0,1). È usuale trovare nel campo informatico non tanto il bit ma sequenze di bit al fine di poter rappresentare più di due alternative possibili.

#### Esempio 4

Ognuna delle 7 note musicali possono essere rappresentate (codificate) con una sequenza di 3 bit, infatti, per discriminare fra 7 situazioni diverse occorrono 3 bit come riportato nella seguente tabella:

Nota Musicale	Codifica
DO	000
RE	001
MI	010
FA	011
SOL	100
LA	101
SI	110

Come si può osservare dalla tabella ad ogni nota è stata assegnata una sequenza di bit differente. Il calcolatore in questo modo potrà discernere, in base al codice digitale, una nota dalle altre.

I raggruppamenti contengono in genere un numero di stringhe binarie pari ad una potenza binaria, cioè  $2^n$ , dove  $n$  è il numero di bit costituenti una stringa. Nel esempio delle note musicali  $n = 3$  e quindi con 3 bit possiamo discernere fra 8 combinazioni differenti (noi ne abbiamo sfruttate solo 7 dato che le note sono solo 7). Il raggruppamento più noto è il Byte.

### Esempio 5

Indichiamo con  $P$  la variabile che rappresenta lo stato del tempo in una zona, cioè in particolare se  $P=1$  significa che piove e con  $P=0$  che non piove.

Indichiamo anche con  $S$  la situazione di sereno ( $S=1$ ) e nuvoloso ( $S=0$ ).

Le possibili situazioni che si possono creare sono:

<b>Codice Binario</b>	<b>P</b>	<b>S</b>
0 0	Non piove	È nuvoloso
0 1	Non piove	È sereno
1 0	Piove	È nuvoloso
1 1	Piove	È sereno

### Esempio 6

Nel seguente esempio possiamo osservare come sia possibile ottenere il numero decimale conoscendo il corrispondente numero binario (decodifica). Approfondiremo in un altro capitolo la codifica da decimale a binario e la decodifica da binario a decimale.

$$101_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 4 + 0 + 1 = 5_{10}$$

### Esercizio

Costruire la tabella nel caso si aggiunge alle precedenti variabili anche  $W$  per esprimere se il vento soffia più di 20 KM/h ( $W=1$ ) oppure no ( $W=0$ ).

### Esempio 7

Rappresentiamo con l'abaco alcuni numeri in base 2:

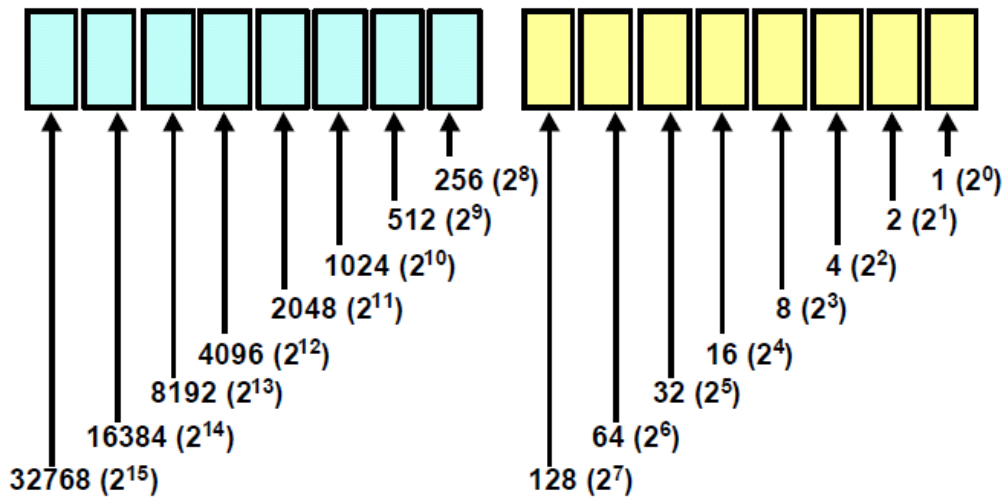
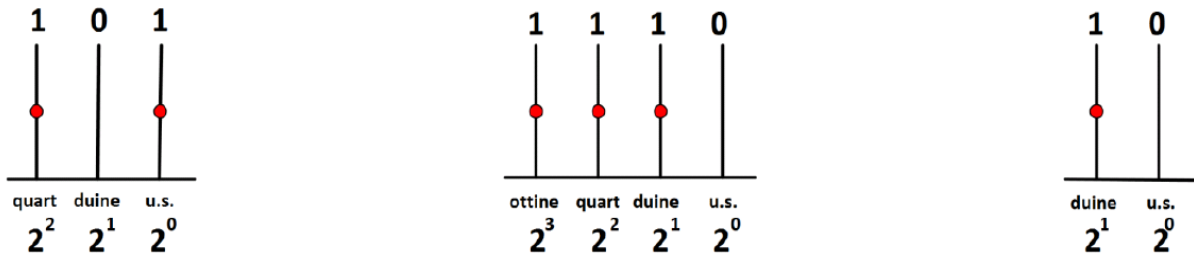
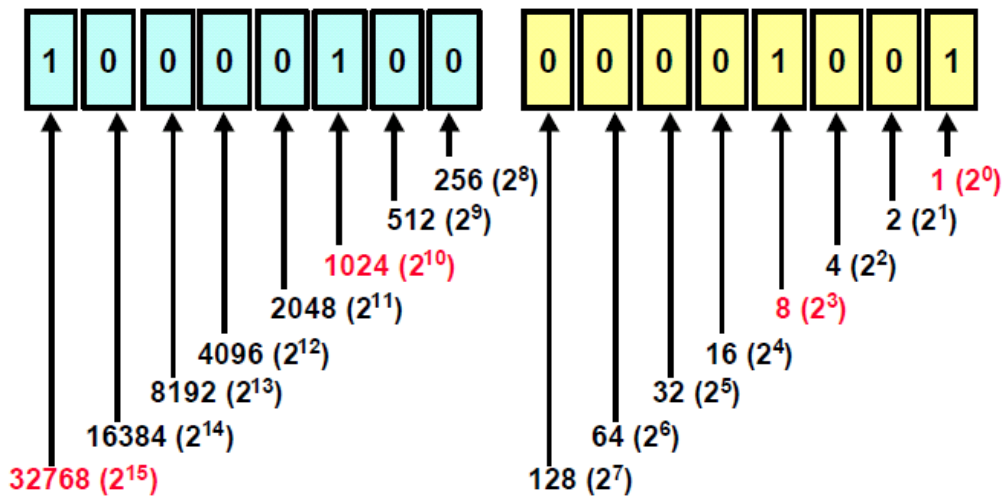


Figura 1 - Pesi binari



VALORE  $32768 + 1024 + 8 + 1 = 33801$

Figura 2 - calcolo di un valore



## IL BYTE (B)

Abbiamo quindi visto che con un bit il calcolatore è in grado di discriminare tra due situazioni diverse, e con una sequenza di  $n$  bit possiamo rappresentare  $2^n$  informazioni differenti. Ovviamente la definizione di solo due diverse informazioni non è molto utile e quindi si usano più bit per rappresentarle, in modo analogo al nostro metodo di accostare più lettere per comporre le parole e le frasi. Aumentando il numero di bit si aumenta il numero di informazioni diverse rappresentabili.

### Esempio 8

Rappresentiamo lo stato dei freni e dell'olio di un veicolo:

Codice	Freni	Olio
00	KO	KO
01	KO	OK
10	OK	KO
11	OK	OK

E quindi in generale, avendo a disposizione  $n$  bit, si potranno rappresentare  $2^n$  sequenze diverse e quindi codificare altrettante informazioni.

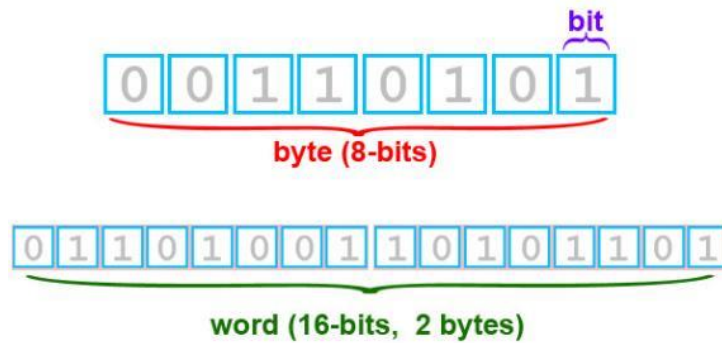
Si è deciso, a livello internazionale, di utilizzare sequenze di 8 bit per formare l'unità effettiva di misura della memoria e si è attribuito a questa quantità il nome **Byte** (**B**inary **T**Erminale, elemento binario, ma di etimologia incerta) e quindi:

$$8 \text{ bit} = 1 \text{ Byte}$$

Utilizzando un Byte è possibile ottenere 256 diverse combinazioni di bit (0 o 1) ovvero è possibile codificare  $256 = 2^8$  diverse informazioni, compresi tra 0 e 255.

È pertanto possibile rappresentare tutti i simboli (caratteri, cifre e simboli particolari) digitabili da tastiera.

Gli informatici di lingua francese utilizzano il più preciso termine **octet** (ovvero ottetto): distinzione utile soprattutto quando erano ancora diffusi computer con byte di dimensione diversa.



Ad esempio alla lettera “a” corrisponde la sequenza 01100001 nel codice di rappresentazione dei caratteri più usato, il codice ASCII (American Standard Code for Information Interchange).

Il Byte è un multiplo del bit:

$$1 \text{ Byte} = 1 \text{ B} = 8 \text{ bit} = 2^3 \text{ bit}$$

Ricordiamo che da tastiera si possono digitare 256 caratteri (caratteri, cifre, simboli) differenti).

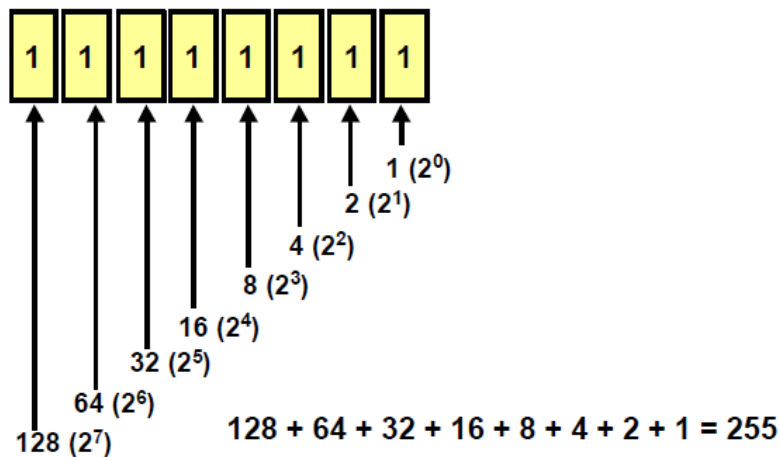
$$256 \text{ simboli} = 2^8 \text{ simboli}$$

Se digito, quindi, il carattere “A” esso occuperà in memoria RAM 1 Byte.

In una memoria RAM da 4 GByte possiamo salvare (circa) 4 miliardi di caratteri (char).

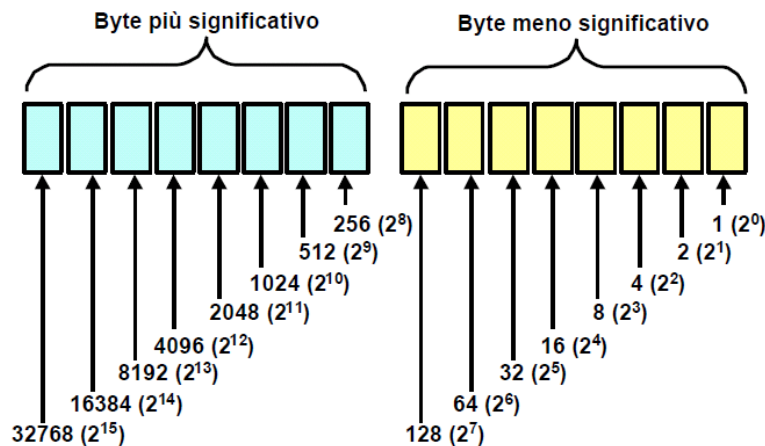
Osserviamo che il simbolo utilizzato per il Byte come unità di misura della quantità di informazione è **B**; la lettera maiuscola sarebbe riservata alle sole unità di misura tratte dai cognomi degli ideatori, ma la Commissione Elettrotecnica Internazionale (IEC) ha deciso di fare un’eccezione dato che **b** è generalmente usato per indicare il bit (il cui simbolo standard sarebbe "bit" per esteso).

### Esempio 9



### Sequenza di Byte

- Il byte più significativo è il primo a sinistra
- Il byte meno significativo è l'ultimo a destra



Un Byte è una quantità di informazione "elementare" per molti scopi: servono ad esempio circa 200 livelli di grigio per passare dal bianco al nero perché l'occhio umano pensi di vedere una sfumatura continua, servono circa 200 caratteri diversi (lettere accentate comprese) per scrivere un qualsiasi testo di una lingua occidentale, servono circa 200 livelli di intensità per riprodurre un suono abbastanza fedele.

Il simbolo utilizzato per il Byte come unità di misura della quantità di informazione è **B**; la lettera maiuscola sarebbe riservata alle sole unità di misura tratte dai cognomi degli ideatori, ma la Commissione Elettrotecnica Internazionale (IEC) ha deciso di fare un'eccezione dato che **b** è generalmente usato per indicare il bit (il cui simbolo standard sarebbe "bit" per esteso).

## SIGNIFICATO FISICO DEL BYTE

Il significato di ogni singolo Byte del file dipende poi dalla natura dei dati, così ad esempio:

- In un file di puro testo ogni singolo carattere (lettera maiuscola, lettera minuscola, cifra, punteggiatura, spazio, ecc.) viene rappresentato con un byte, in accordo con una tabella convenzionale detta ASCII (American Standard Code for Information Interchange);
- In un file che contiene un programma eseguibile ogni singola istruzione viene rappresentata con un Byte (o talvolta con più Byte), secondo la convenzione stabilita dai produttori dei microprocessori;
- In un file di dati numerici i Byte sono usati per rappresentare numeri (interi o decimali) secondo le regole dell'aritmetica binaria (esiste tutta una teoria sui numeri binari, che comunque riguarda soltanto gli specialisti; l'utente normale può tranquillamente ignorarla).
- E così via.

In pratica, il Byte può essere pensato equivalente alla pressione di un tasto. Per esempio, la pressione del tasto A genera la sequenza di bit 01000001:



Ad ognuno delle 256 combinazioni diverse di 0 e 1 che si ottengono con un Byte viene associato un preciso carattere (anche lo spazio bianco occupa un Byte di memoria).

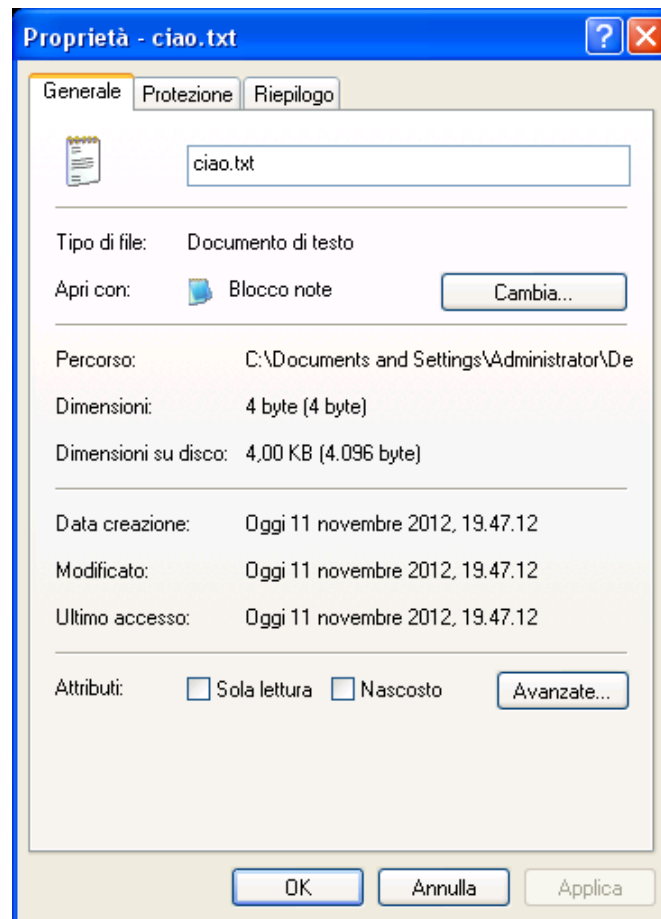
Vengono così definite delle tabelle standard di caratteri che sono utilizzabili dal computer per rappresentare/codificare le informazioni in modo da poter scambiare facilmente dati tra loro.

Uno degli standard più conosciuti è quello usato dai comuni PC: il codice ASCII.

Ecco quindi che una frase del tipo “CIAO MAMMA!” occupa 11 Byte: 4 per la parola CIAO, 1 per lo spazio, 5 per la parola MAMMA e, 1 per il simbolo del punto esclamativo.

## ESPERIENZA PRATICA

Una esperienza concreta per studiare questo concetto è quella valutare le dimensioni di un file di testo. Aprite il programma Blocco Note, digitate la stringa **ciao**, e quindi salvate il documento assegnandogli un nome. Cliccate ora con il tasto destro sul file e dal menu contestuale che appare scegliete la voce Proprietà. Analizzate le dimensioni del file ricordandovi che avete salvato in questo file 4 caratteri e, ognuno di essi è pari ad un Byte.



Le dimensioni del file sono quindi pari a 4 Byte.

Se volessimo conoscere la stessa grandezza espressa in termini di bit, possiamo moltiplicare 4 Byte x 8 bit = 32 bit. Potremmo indifferentemente dire quindi che il nostro documento occupa **4 Byte** oppure **32 bit**.

## BIT OPPURE BYTE

Come appena visto, esprimere una grandezza in bit oppure in Byte è indifferente: è la stessa cosa affermare che *una distanza misura 2,54 centimetri oppure 1 pollice*? Certamente sì, la sostanza è la stessa, cambia solo l'unità di misura di riferimento.

Per una consuetudine non scritta però si preferisce utilizzare **il Byte per le misure spaziali** ("questo documento occupa 5.000 Byte" oppure "questa memoria ha una capienza di 64.000.000 di Byte") ed **il bit per quelle velocistiche** ("questo collegamento ad Internet permette di scaricare a 56.000 bit al secondo"). Si tratta però solamente di consuetudini.

## UNITA' DI MISURA E MUTIPLI (B = Byte; b = bit)

Per comodità di calcolo i multipli del Byte vengono generalmente arrotondati a potenze di 2 (benché questo sia formalmente sbagliato), invece che di 10; tale ambiguità ha portato l'IEC a definire nuovi prefissi per multipli binari; tali valori non sono comunque entrati nell'uso comune.

Questa ambiguità viene sfruttata a fini commerciali dai produttori di hard disk, utilizzando i "corretti ma poco in uso" multipli decimali, in modo tale da far figurare quantità maggiori: ad esempio, un hard disk da 80 GB nominali potrà effettivamente contenere solo 74.5 GiB.

Potenze in base a 2 del Byte.

$$1 \text{ Byte} = 1\text{B} = 8 \text{ b} = 1 \text{ CARATTERE} / \text{SIMBOLO}$$

Nella tabella seguente sono elencati i prefissi ufficiali IEC, i prefissi decimali spesso usati e la differenza percentuale tra i due.

Quantità	Fattore	Nome corretto	Sigla	Altri nomi in uso	Sigla	Errore
	$2^{-3} \text{ B}$	bit	b			
	$2^{-1} \text{ B}$	nibble				
1 B =	$2^0 \text{ B}$	byte	B	byte	B	0
1.024 B =	$2^{10} \text{ B}$	Kibibyte	KiB	Kilobyte	kB	+2,4%
1.024 KiB =	$2^{20} \text{ B}$	Mebibyte	MiB	Megabyte	MB	+4,9%
1.024 MiB =	$2^{30} \text{ B}$	Gibibyte	GiB	Gigabyte	GB	+7,4%
1.024 GiB =	$2^{40} \text{ B}$	Tebibyte	TiB	Terabyte	TB	+10,0%
1.024 TiB =	$2^{50} \text{ B}$	Pebibyte	PiB	Petabyte	PB	+12,6%
1.024 PiB =	$2^{60} \text{ B}$	Exbibyte	EiB	Exabyte	EB	+15,3%

Multipli del byte					
Prefissi SI			Prefissi binari		
Nome	Simbolo	Multiplo	Nome	Simbolo	Multiplo
kilobyte	kB	$10^3$	kibibyte	KiB	$2^{10}$
megabyte	MB	$10^6$	mebibyte	MiB	$2^{20}$
gigabyte	GB	$10^9$	gibibyte	GiB	$2^{30}$
terabyte	TB	$10^{12}$	tebibyte	TiB	$2^{40}$
petabyte	PB	$10^{15}$	pebibyte	PiB	$2^{50}$
exabyte	EB	$10^{18}$	exbibyte	EiB	$2^{60}$
zettabyte	ZB	$10^{21}$	zebibyte	ZiB	$2^{70}$
yottabyte	YB	$10^{24}$	yobibyte	YiB	$2^{80}$

**NB:** Per semplicità noi utilizzeremo il prefisso binario senza la i: KiB → KB.

1 YottaByte = 1 YB	= 1024 ZB	= $2^{80}$ B = $1024^8$ B
1 ZettaByte = 1 ZB	= 1024 PB	= $2^{70}$ B = $1024^7$ B
1 ExaByte = 1 EB	= 1024 PB	= $2^{60}$ B = $1024^6$ B
1 PetaByte = 1 PB	= 1024 TB	= $2^{50}$ B = $1024^5$ B
1 TeraByte = 1 TB	= 1024 GB	= $2^{40}$ B = $1024^4$ B
1 GigaByte = 1 GB	= 1024 MB	= $2^{30}$ B = $1024^3$ B
1 MegaByte = 1 MB	= 1024 KB	= $2^{20}$ B = $1024^2$ B
1 KiloByte = 1 KB	= 1024 B	= $2^{10}$ B = $1024^1$ B

Prefisso	Derivazione (greca)	Indicazione
Yotta	Okto	ottava potenza di 1000
Zetta	Sept	settima potenza di 1000
Exa	Hex	sesta potenza di 1000
Peta	Penta	quinta potenza di 1000
Tera	Tetra	quarta potenza di 1000
Giga	Gigante	terza potenza di 1000
Mega	Grande	seconda potenza di 1000
Kilo	Mille	prima potenza di 1000

Multipli del byte					
Prefissi SI			Prefissi binari		
Nome	Simbolo	Multiplo	Nome	Simbolo	Multiplo
kilobyte	kB	$10^3$	kibibyte	KiB	$2^{10}$
megabyte	MB	$10^6$	mebibyte	MiB	$2^{20}$
gigabyte	GB	$10^9$	gibibyte	GiB	$2^{30}$
terabyte	TB	$10^{12}$	tebibyte	TiB	$2^{40}$
petabyte	PB	$10^{15}$	pebibyte	PiB	$2^{50}$
exabyte	EB	$10^{18}$	exbibyte	EiB	$2^{60}$
zettabyte	ZB	$10^{21}$	zebibyte	ZiB	$2^{70}$
yottabyte	YB	$10^{24}$	yobibyte	YiB	$2^{80}$

### SOTTOMULTIPLI

1 milli = 1 m = 0,001 =  $10^{-3}$

1 micro = 1  $\mu$  = 0,001 m =  $10^{-6}$

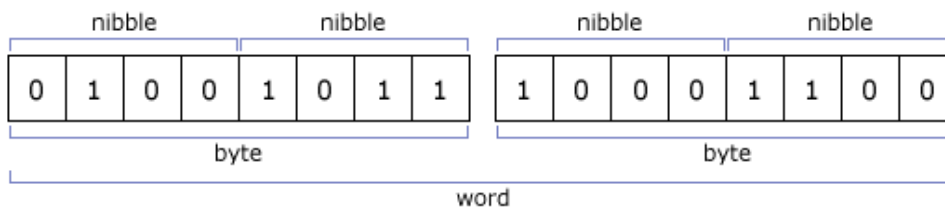
1 nano = 1 n = 0,001  $\mu$  =  $10^{-9}$

1 pico = 1 p = 0,001 n =  $10^{-12}$

### NIBBLE - WORD

Mentre un gruppo contiguo di 8 bit è detto Byte, un gruppo contiguo di 4 bit, cioè la metà di un Byte, è detto **Nibble** (o Half-Byte). L'insieme di due Byte adiacenti è detto **Word** (parola). Esiste anche il termine **Double-Word** per indicare l'insieme di due word adiacenti.

Il bit meno significativo di un gruppo di bit (sia esso un byte, nibble o word) è detto LSB (Less Significant Bit), mentre quello più significativo è detto MSB (Most Significant Bit).



\* 4 bit = metà di un Byte, formano un nibble.



\* 2 Byte = 16 bit, sono chiamati anche word.

\* 4 Byte = 32 bit = 2 word, sono chiamati anche double word.

\* 8 Byte = 64 bit = 2 double word, sono chiamati anche quad word.

È importante sottolineare che i valori di word, double word e quad word qui riportati sono indicativi e fanno riferimento ad una architettura di computer in cui una word è uguale a 2 Byte.

## UNITÀ DI MISURA PER LA MEMORIZZAZIONE DELLE INFORMAZIONI

Simbolo	in Bit	in Byte	in potenze di 2	Equivalente a
1 b (bit)	1	1/8	$2^1 = 2$ stati (acceso - spento)	
1 B (Byte)	8	1	$2^8 = 256$ caratteri	Un carattere alfanumerico
1 KB (KiloByte)	8.192	1.024	$2^{10}$ Byte	Un terzo di pagina di testo
1 MB (MegaByte)	8.388.608	1.048.576	$2^{20}$ Byte	Circa 300 pagine di testo
1 GB (GigaByte)	8.589.934.592	1.073.741.824	$2^{30}$ Byte	Circa 300.000 pagine di testo
1 TB (TeraByte)	8.796.093.302.400	1.099.511.628.000	$2^{40}$ Byte	Circa 300 milioni di pagine di testo

## UNITÀ DI MISURA PIÙ DIFFUSE

Simbolo	Unità di misura	Descrizione
Bd	Baud	Equivale a 1 bps (Bit Per Secondo)
bps	Bit Per Secondo	Misura il trasferimento dei dati tra computer.
MIPS	Milioni di Istruzioni Per Secondo	Indicano la quantità di istruzioni elementari del processore che vengono elaborati in un secondo all'interno della CPU.
Hz	Hertz	Misura la velocità del clock interno. Il clock sincronizza i vari componenti del sistema.
dpi	Dots Per Inch	Stabilisce la quantità di punti per pollice quadrato (2,54 x 2,54 cm) in una immagine e in una stampa.
px	Pixel	Minimo puntino luminoso, la cui quantità o densità identifica la risoluzione di un monitor.

## TEMPO DI TRASFERIMENTO (DOWNLOAD) DI 1 MB

- a 14.400 bps (1.8 KB/s) = 582 sec.
- a 28.800 bps (3.6 KB/s) = 291 sec.
- a 33.600 bps (4.2 KB/s) = 250 sec.
- a 56.000 bps (7.0 KB/s) = 150 sec.
- a 64.000 bps (8.0 KB/s) = 131 sec. (ISDN)
- a 128.000 bps (16.0 KB/s) = 65 sec.
- a 256.000 bps (32.0 KB/s) = 33 sec.
- a 382.000 bps (47.7 KB/s) = 22 sec.
- a 640.000 bps (75.0 KB/s) = 14 sec. (ADSL)
- a 1.000.000 bps (125.0 KB/s) = 8 sec.

### ***Molto importante:***

La velocità di trasferimento dei modem viene espressa in bps (bit per secondo).

Si parla ad es. di modem a 56 Kbps o 56.000 bps, o di una linea ADSL da 20 Mbps.

Poiché le dimensioni dei file vengono espresse in Byte (B) e multipli (KB, MB, GB, TB), per ottenere la velocità espressa in KB dobbiamo dividere per 8.

Ad es: un modem a 33.600 bps trasferisce  $33.600/8 = 4.200$  Byte/sec = 4,2 KB/sec.