

[Home Page](#) [Informazioni](#) [Aiuto](#)

## Informazioni aggiuntive sulle architetture di memoria Little-Endian e Big-Endian

[http://www.vbsimple.net/info/info\\_13.htm](http://www.vbsimple.net/info/info_13.htm)

I processori della maggior parte di PC sono del modello **Intel x86**. Infatti i moderni computer rispettano gli stessi modelli architetture dei loro predecessori.

È fondamentale però tenere a mente che al mondo non esistono soltanto i processori di tipo Intel x86, ma ci sono tantissimi altri computer con architetture molto diverse dai normali PC, quali sono i processori Motorola.

Una differenza sostanziale tra le due architetture è la modalità di memorizzazione dei dati in memoria: le piattaforme Intel infatti memorizzano i dati in una maniera detta **Little-Endian** (detta talvolta Little-Ending o Little-End In), mentre le piattaforme Motorola trattano i dati secondo il metodo **Big-Endian** (detto anche Big-Ending o Big End In).

La differenza fondamentale tra le due modalità è l'ordine di memorizzazione dei dati.

Il formato **Little-Endian** memorizza i dati a coppie ordinate ma in posizione invertita. Ciò significa che i due bytes meno significativi sono memorizzati per primi, mentre i due bytes con valori più significativi sono memorizzati per ultimi.

Ad esempio il numero esadecimale **0x12345678** nelle piattaforme little-endian viene memorizzato nei bytes

**0x78 (bit dallo 0 al 7)**

**0x56 (bit dall'8 al 15)**

**0x34 (bit dal 16 al 23)**

**0x12 (bit dal 24 al 31)**

I bytes sono invertiti, ma i bit dei bytes sono ordinati.

Il formato **Big-Endian** memorizza i dati in maniera ordinata. I due bytes più significativi sono memorizzati per primi e quelli meno significativi sono memorizzati per ultimi. Lo stesso numero esadecimale **0x12345678** nelle piattaforme big-endian è memorizzato nei bytes

**0x12 (bit dallo 0 al 7)**

**0x34 (bit dall'8 al 15)**

**0x56 (bit dal 16 al 23)**

**0x78 (bit dal 24 al 31)**

I bytes ed i bit in essi sono ordinati.

Pertanto quando si utilizzano diretti accessi alla memoria, quale la funzione API CopyMemory (RtlMoveMemory), è importante ricordare che per [estrarre i bytes contenuti in un numero long](#) adoperando gli accessi alla memoria, è necessario invertire le coppie di dati per ottenere il numero originale.

Alcune piattaforme, quali le MIPS RISC o le DEC Alpha o le Sparc permettono l'utilizzo di entrambe le modalità Little-Endian e Big-Endian. Windows NT in ogni caso, se viene

eseguito su una di queste macchine, lavora sempre nel formato Little-Endian, ignorando l'impostazione predefinita della macchina.

[Fibia FBI](#)

27 Marzo 2001

---