

## Little Endian vs Big Endian

Nell'ambito dell'automazione industriale (ma non solo) uno dei problemi che spesso si affrontano riguarda lo scambio di informazioni tra dispositivi che memorizzano quantità numeriche con modalità differenti. In particolare, parliamo di Endianness presentando due modalità di memorizzazione delle informazioni.

### Little Endian

È un sistema di memorizzazione in cui quantità numeriche a 16 o più bits vengono memorizzate partendo dal byte meno significativo, fino al più significativo.

A titolo di esempio supponiamo di voler memorizzare una quantità a 16 bits (word). In questo caso, gli 8 bits meno significativi (0 fino a 7) verranno memorizzati nella prima locazione di memoria, mentre quelli più significativi (8 fino a 15) verranno memorizzati nella locazione successiva.

Sia 2000 l'indirizzo di memoria dove si vuole memorizzare la nostra quantità numerica (es. 1000)

In binario questa quantità equivale a 00000011 11101000 ovvero, 03E8 in esadecimale.

La parte più significativa (byte più significativo o byte alto) è 03, mentre quella meno significativa (byte meno significativo o byte basso) vale E8. Di seguito come viene memorizzata.

Indirizzo 2000 (bits 0-7)	Indirizzo 2001 (bits 8-15)
E8	03

### Big Endian

È un sistema di memorizzazione in cui quantità numeriche a 16 o più bits vengono memorizzate partendo dal byte più significativo, fino a quello meno significativo.

Considerando l'esempio di prima, la nostra quantità numerica 1000, verrà memorizzata come segue:

Indirizzo 2000 (bits 8-15)	Indirizzo 2001 (bits 0-7)
03	E8

Ovvero in modalità esattamente opposta rispetto al sistema Little Endian.

Il ragionamento può essere esteso sia per le quantità a 32 che a 64 bits.

La difficoltà nell'avere questi due sistemi di rappresentazione diversi, nasce quando si vuole effettuare uno scambio di informazioni tra due sistemi che adottano Endianness diverse.

Un caso molto comune è dato dal collegamento tra HMI (Human Machine Interface) che sono prevalentemente sistemi Little Endian con PLC (Programmable Logic Controller) che in alcuni casi possono adottare un sistema Big Endian.

Supponiamo ad esempio che il nostro PLC abbia al suo interno un'informazione numerica (1000) memorizzata in uno dei suoi registri a 16 bits e che, tale informazione sia in formato Big Endian.

Indirizzo 2000 (bits 8-15)	Indirizzo 2001 (bits 0-7)
03	E8

Il nostro HMI è connesso al PLC mediante ad esempio una porta seriale utilizzando un determinato protocollo.

Quando l'HMI chiederà il contenuto delle celle 2000 e 2001 al PLC, otterrà rispettivamente i valori 03 ed E8.

Essendo però l'HMI un sistema Little Endian, esso considererà 03 come byte meno significativo ed E8 come byte più significativo.

$E803 = 11101000\ 00000011 = 59395$  (-6141 se è una quantità con segno)

La quantità visualizzata dal nostro HMI (59395 o -6141) non corrisponde minimamente a ciò che invece è memorizzato nel PLC (1000).

Affinché il nostro valore sia visualizzato correttamente dall'HMI, occorre effettuare un'operazione di swap dei dati che provengono dal campo.

Lo swap è un'operazione che scambia la posizione di due o più bytes contigui nella memoria.

Ad esempio:

Indirizzo 2000 (bits 8-15)	Indirizzo 2001 (bits 0-7)
03	E8

Dopo lo swap diventa:

Indirizzo 2000 (bits 0-7)	Indirizzo 2001 (bits 8-15)
E8	03

Riprendendo l'esempio dell'HMI, quando viene ricevuta dal PLC la coppia di bytes (03, E8), essi verranno memorizzati come (E8, 03) mediante l'operazione di swap.

In questo modo l'HMI visualizzerà correttamente il valore 1000.

Può anche succedere il caso in cui due sistemi adottino la stessa Endianness, ma che il protocollo di comunicazione utilizzato per trasferire informazioni tra i 2 sistemi adotti una Endianness differente.

Ritornando al nostro esempio di prima, HMI e PLC adottano entrambi il sistema Little Endian ma il protocollo di comunicazione usato è invece Big Endian. È questo il caso del protocollo Modbus, uno dei più diffusi nell'ambito dell'automazione industriale.

Di seguito uno schema semplificato della connessione HMI/PLC.



Sulla sinistra il nostro HMI e sulla destra un PLC. Entrambi comunicano mediante il protocollo di Modbus.

La quantità numerica contenuta nel registro 2000 del PLC (es. velocità del motore) è memorizzata nella forma (E8, 03) in quanto Little Endian. Tuttavia, quando viene trasferita verso il nostro HMI, essa sarà nella forma (03, E8) ovvero in Big Endian.

Si rende quindi necessaria l'operazione di swap da parte del nostro HMI affinché, l'Endianness sia rispettata.

Lo stesso ragionamento si applica quando si vuole trasferire un'informazione numerica dal sistema HMI al PLC (ad esempio il set point di velocità del motore).

Riprendendo l'esempio del protocollo Modbus di cui sopra, prima trasferire il valore 500 (01F4 in esadecimale), l'HMI, effettuerà un'operazione di swap in modo che i bytes arrivino al PLC nell'ordine 01, F4 anziché F4, 01 (come sono effettivamente memorizzati nel nostro HMI).

## **Networking**

Quando si fa programmazione in ambito networking, si utilizzano le sockets, ovvero oggetti resi disponibili dai sistemi operativi, con cui è possibile creare applicazioni che comunicano con altri sistemi mediante i protocolli TCP/IP e UDP/IP. In generale occorre specificare l'indirizzo IP e la porta del nodo con cui si vuole comunicare. Prendendo come riferimento la versione IPv4 del protocollo IP, sappiamo che un indirizzo IP è una quantità a 32 bits, mentre la porta è una quantità a 16 bits.

In ambito networking, indirizzi IP e porte sono sempre in formato Big Endian.

Se il sistema dove gira la nostra applicazione è un sistema Little Endian, dovremo utilizzare due apposite funzioni per convertire da Little Endian a Big Endian. Esse sono:

- `htonl` – converte una quantità a 32 bits dalla modalità di rappresentazione del sistema host (nel nostro caso Little Endian) alla modalità di networking (Big Endian).
- `htons` – converte una quantità a 16 bits dalla modalità di rappresentazione del sistema host (nel nostro caso Little Endian) alla modalità di networking (Big Endian).

La prima verrà utilizzata per l'indirizzo IP, mentre la seconda per la porta.

**E per concludere un aneddoto.**

La terminologia Little Endian/Big Endian venne introdotto nel 1980 da Danny Cohen nel suo memorandum 'Internet Engineering Note 137'. In particolare, si è ispirato al racconto dei viaggi di Gulliver in Lilliput. In esso, si racconta di una disputa tra gli abitanti di Lilliput e quelli di un altro regno su come rompere (prima di mangiarle) le uova. Una fazione le rompe dalla parte piccola (Little Endians), mentre l'altra le rompe dalla parte più grande (Big Endians).

Questo è il link dove si può scaricare il memorandum: <https://www.ietf.org/rfc/ien/ien137.txt>