**BIG ENDIAN**

Os bytes são guardados por ordem decrescente do seu “peso numérico” em endereços sucessivos da memória.

**VANTAGENS**

É mais natural para maioria das pessoas e, assim, torna mais fácil a leitura de despejos de memória em hexadecimal. Tendo o byte de mais alta ordem no deslocamento zero. Máquinas big endian armazenam inteiros e strings na mesma ordem e são mais rápidas em certas operações de string. A maioria das representações gráficas em bitmap são organizados com o esquema de “o bit mais significativo na esquerda”, o que significa que o trabalho com elementos gráficos maiores do que um byte pode ser feita pela própria arquitetura. Na decodificação de dados compactados codificados com esquemas como Huffman e LZW, a própria palavra de código pode ser usada como um índice para uma pesquisa na tabela se estiver armazenada em big endian.

**DESVANTAGENS**

A conversão de um endereço de 32 bits para um endereço inteiro de 16 bits requer que uma máquina big endian realize uma adição. A maioria das arquiteturas que usam o esquema big endian não permite que palavras sejam escritas em limites de endereço que não sejam palavras (por exemplo, se uma palavra possui 2 ou 4 bytes, ela deve sempre iniciar em um byte de endereço de numeração par). Isto desperdiça espaço. Se um programador escreve uma instrução para ler um valor de tamanho errado de palavra, em uma máquina big endian ele é sempre lido como um valor incorreto. Rede de computadores são big endian, o que significa que quando computadores little endian vão enviar inteiros através da rede (endereços de dispositivos de rede, por exemplo), eles necessitam convertê-los para ordem de bytes de rede. Do mesmo modo, quando recebem valores inteiros da rede, eles precisam convertê-los de volta para a sua representação inteira.

**LITTLE ENDIAN**

Os bytes são guardados por ordem crescente do seu “peso numérico” em endereços sucessivos da memória.

**VANTAGENS**

Enquanto big endian precisa realizar uma adição para converter um endereço inteiro. A aritmética de alta precisão em máquinas little endia é mais rápida e mais fácil. Arquiteturas litte endian, tal como a Intel, permitem leituras e escritas em endereços ímpares, o que torna a programação destas máquinas muito mais fácil.

**DESVANTAGENS**

Precisa-se saber onde se deve saber o tamanho do número e, depois, saltar sobre bytes para encontrar aquele que contém a informação de sinal. Diferente do big que as representações gráficas são organizados de bit mais significativos a esquerda o que significa que o trabalho com elementos gráficos maiores, no little isso é uma limitação de performance para computadores little endian porque eles devem continuamente reverter a ordem dos bytes ao trabalhar com grandes objetos gráficos.