

# BIG AND LITTLE ENDIAN

---

Little and big endian are two ways of storing multibyte data (int, float, etc) in memory.

– If machine is big endian, then first byte of binary representation of multi-byte data is stored first.

– If machine is little endian, then last byte of binary representation of multi-byte data is stored first.

## **Where does this term “endian” come from?**

Jonathan Swift was a satirist (he poked fun at society through his writings).

His most famous book is “Gulliver’s Travels“, and he talks about how certain people prefer to eat their hard boiled eggs from the little end first (thus, little endian), while others prefer to eat from the big end (thus, big endians) and how this lead to various wars.

## **Memory**

Memory is like one long array of addresses. Each address stores one element of the memory. Each element is typically one byte.

There are some memory configurations where each address stores a nybble (i.e every 4-bits have its own address). However, those are exceedingly rare, and have gone with the dinosaurs (floppy disks).

For this article, let's assume that all memory addresses store bytes. (i.e *byte-addressable memory*)

### **Big Endian**

In this format, what is stored in the memory will be consumed in the same sequence. Because memory stores the bytes in the actual sequence.

If a 32-bit word has value 0x0A0B0C0D. Then, that word will be stored as below in the memory (also showing how it will be in the register).

### **Little Endian**

In this format, the LSB is stored at the lowest address followed by higher significant bytes. This is opposite to how bytes were stored in BigEndian format.

If a 32-bit word has value 0x0A0B0C0D. Then, that word will be stored as below in the memory (also showing how it will be in the register).

To remember which is which, recall whether the least significant byte is stored first (thus, little endian) or the most significant byte is stored first (thus, big endian).

Note that it is “byte” instead of “bit”. LSB also means Least Significant Byte.

## **Issues:**

Endianess is a problem if we are opening a file in a system with different endianness than the system in which that file is created. Suppose you are storing int values to a file, then you send the file to a machine which uses the opposite endianness and read in the value. You'll run into problems because of endianness. You'll read in reversed values that won't make sense.

Similar problems will come while transferring a file over the network.

## **Impact of Endianess on file format:**

It may look like, but the Endianness is more to do with architecture of the machine than with the individual file format. However, file format may be using it there specific way. If a file format is dealing with individual bytes, then the Endianness of underline format actually does not matter (eg. If the file is ASCII file).

Other file format may use specific Endianness format. For Example:

Big Endian File Formats:

-----

Adobe Photoshop

IMG (GEM Raster)

JPEG

SGI (Silicon Graphics)

Sun Raster

MacPaint

WPG (WordPerfect Graphics Metafile)

Little Endian File Formats:

-----

BMP (Windows and OS/2 Bitmaps)

GIF

FLI (Autodesk Animator)

PCX (PC Paintbrush)

QTM (Quicktime Movies)

Microsoft RTF (Rich Text Format)

TGA (Targa)

NOT APPLICABLE (text format):

-----

PostScript

POV (Persistence of Vision ray-tracer)

Both

----

DXF (AutoCad)

Microsoft RIFF (.WAV & .AVI)

TIFF (Endian identifier encoded into file)

XWD (X Window Dump) (Endian identifier encoded into file)

**Code to Check Endianness:**

If the below function is run on Little Endian architecture then it will output “Little Endian” else it will print “Big Endian”.

```
1 void checkEndian()
2 {
3     unsigned int i = 1;
4     char *cptr = (char*)&i;
5
6     if (*cptr != 0)
7         printf("Little Endian");
8     else
9         printf("Big Endian");
10 }
```

**Which one is better:**

Which one is better, Unix or Windows ?

Which one is better, Apple or Microsoft?

Which one is better, Samsung Galaxy or iPad?

Its the same debate.. there is no fixed answer acceptable to all.

Source: <http://www.ritambhara.in/big-and-little-endian/>