

Re: program logic based on endianness

Source: <http://unix.derkeiler.com/Newsgroups/comp.unix.programmer/2007-08/msg00209.html>

- *From:* Eric Sosman <esosman@xxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Fri, 10 Aug 2007 08:13:15 -0400
-

sam_cit@xxxxxxxxxxx wrote:

Hi Everyone,

I have a program unit which does >> and << of an integer which is of 4 bytes length. The logic of shifting and action based on the result, assumes that the system is big-endian.

Accordingly, if i need the program to work fine in a little-endian system. I understand that the code needs to be changed.

I wanted to know if the above holds true for bitwise and (&) and bitwise or (|). I think, the processor should take care of the operation a&b or a|b irrespective of the endianness of the system.

Please provide your comments.

Most C operators -- including the shift operators and the bitwise operators -- work on the values of their operands, not on the representations. `16 << 2' is always 64, never 4 (and never anything even weirder).

--
Eric Sosman
esosman@xxxxxxxxxxxxxxxxxxxxxxxx