Subject: Re: Endian-ness
Posted by David Fanning on Fri, 08 Feb 2002 21:14:25 GMT
View Forum Message <> Reply to Message

Jonathan Joseph (jj21@cornell.edu) writes:

> Is there a system variable that gives the endian-ness of the current
> hardware? I am reading a file which tells me the endian-ness of the
> data, and I'd like to swap_endian if it is different from the current
> hardware. In lieu of finding a system variable to compare to, I have
> done this:
>
> test_int = 1
> byteorder,test_int,/ntohs
> big_endian = test_int eq 1
>
> "network" byte order is big-endian, so I convert a 1 to the host
> byte-order and see if it's still a 1.
>
> This way works fine, but it seems as thought I'm missing something.
> Is there a better way? Using the byteorder routine to convert the data
> is not an option (unless it's been improved for 5.5).

I don't know of a system variable. I've always used this
little function:

```
FUNCTION IS_LITTLE_ENDIAN
little_endian = (BYTE(1, 0, 1))[0]
IF (little_endian) THEN RETURN, 1 ELSE RETURN, 0
END

IDL> IF Is_Little_Endian() THEN Print, 'This is little endian, bro.'
```

I've always had success with the SWAP_IF_BIG_ENDIAN and
SWAP_IF_LITTLE_ENDIAN keywords on OPEN statements, however.
I presume these use the BYTEORDER routine, which I have
also never had a moments trouble with. (I live a clean
and wholesome life, though, which may explain it.)

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Endian-ness
Posted by Liam E. Gumley on Fri, 08 Feb 2002 21:18:15 GMT
View Forum Message <> Reply to Message

Jonathan Joseph wrote:
> Is there a system variable that gives the endian-ness of the current
> hardware? I am reading a file which tells me the endian-ness of the
> data, and I'd like to swap_endian if it is different from the current
> hardware. In lieu of finding a system variable to compare to, I have
> done this:
>
> test_int = 1
> byteorder,test_int,/ntohs
> big_endian = test_int eq 1
>
> "network" byte order is big-endian, so I convert a 1 to the host
> byte-order and see if it's still a 1.
>
> This way works fine, but it seems as thought I'm missing something.
> Is there a better way? Using the byteorder routine to convert the data
> is not an option (unless it's been improved for 5.5).

```
;---
FUNCTION BIG_ENDIAN

;- Returns true (1B) if the host platform is big endian
;- (most significant byte first)

return, 1B - byte(1L, 0L)

END
;---
FUNCTION LITTLE_ENDIAN

;- Returns true (1B) if the host platform is little endian
;- (least significant byte first)

return, byte(1L, 0L)

END
;---

IDL Version 5.3 (IRIX mipseb). (c) 1999, Research Systems, Inc.
IDL> print, big_endian(), little_endian()
  1   0

IDL Version 5.3 (Win32 x86). (c) 1999, Research Systems, Inc.
IDL> print, big_endian(), little_endian()
  0   1
```

Cheers,
Liam.
Practical IDL Programming
http://www.gumley.com/

Subject: Re: Endian-ness
Posted by Jonathan Joseph on Mon, 11 Feb 2002 15:16:30 GMT
View Forum Message <> Reply to Message

Thank you David and Liam,

Liam's result seems more aesthetically pleasing (no offense David). Unfortunately, I don't think I can use the swap_if_big_endian and swap_if_little_endian keywords to OPEN, because whether I swap or not depends on BOTH the hardware and the file. I have no a-priori knowledge of the endian-ness of the file until I've already opened it. Although, I guess I could open the file, figure out what it is, close it and then re-open it. The file has a text header (impervious to byte-order issues) that will indicate the endian-ness of the file, followed by binary data.

Am I correct in assuming that Liam's functions don't need to be specifically cast to Long? In other words,

byte(1,0) would yield the same result as byte(1L, 0L)

Thanks.

-Jonathan

"Liam E. Gumley" wrote:
>
> Jonathan Joseph wrote:
>> Is there a system variable that gives the endian-ness of the current
>> hardware? I am reading a file which tells me the endian-ness of the
>> data, and I'd like to swap_endian if it is different from the current
>> hardware. In lieu of finding a system variable to compare to, I have
>> done this:
>>
>> test_int = 1
>> byteorder,test_int,/ntohs
>> big_endian = test_int eq 1
>>
>> "network" byte order is big-endian, so I convert a 1 to the host
>> byte-order and see if it's still a 1.

>>
>> This way works fine, but it seems as thought I'm missing something.
>> Is there a better way? Using the byteorder routine to convert the data
>> is not an option (unless it's been improved for 5.5).
>
> ;---
> FUNCTION BIG_ENDIAN
>
> ;- Returns true (1B) if the host platform is big endian
> ;- (most significant byte first)
>
> return, 1B - byte(1L, 0L)
>
> END
> ;---
> FUNCTION LITTLE_ENDIAN
>
> ;- Returns true (1B) if the host platform is little endian
> ;- (least significant byte first)
>
> return, byte(1L, 0L)
>
> END
> ;---
>
> IDL Version 5.3 (IRIX mipseb). (c) 1999, Research Systems, Inc.
> IDL> print, big_endian(), little_endian()
> 1 0
>
> IDL Version 5.3 (Win32 x86). (c) 1999, Research Systems, Inc.
> IDL> print, big_endian(), little_endian()
> 0 1
>
> Cheers,
> Liam.
> Practical IDL Programming
> http://www.gumley.com/

Subject: Re: Endian-ness
Posted by george.mccabe on Wed, 20 Feb 2002 19:48:11 GMT
View Forum Message <> Reply to Message

jonathan,

i don't know how asthetically pleasing, but in my systems i enclude
the following lines in the IDL setup session batch procedure -

; big_endian platform?
a=1 & b=a & byteorder,a,/swap_if_little
DEFSYSV, '!BIGENDIAN', (a eq b), 1

i do like the (global) system variable approach for things such as
this,
george

Jonathan Joseph <jj21@cornell.edu> wrote in message
news:<3C67E04E.998AD84C@cornell.edu>...
> Thank you David and Liam,
>
> Liam's result seems more aesthetically pleasing (no offense David).
> Unfortunately, I don't think I can use the swap_if_big_endian and
> swap_if_little_endian keywords to OPEN, because whether I swap or
> not depends on BOTH the hardware and the file. I have no a-priori
> knowledge of the endian-ness of the file until I've already
> opened it. Although, I guess I could open the file, figure out
> what it is, close it and then re-open it. The file has a text header
> (impervious to byte-order issues) that will indicate the endian-ness
> of the file, followed by binary data.
>
> Am I correct in assuming that Liam's functions don't need to be
> specifically cast to Long? In other words,
>
> byte(1,0) would yield the same result as byte(1L, 0L)
>
> Thanks.
>
> -Jonathan