
Subject: Re: Help: Byte-swapping program
Posted by [grunes](#) on Mon, 12 Dec 1994 15:12:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <schaep-1212941334430001@130.60.16.90> schaep@rsl.geogr.unizh.ch (Michael E. Schaepman) writes:

> Since we are moving our data from a VAX and a DecStation to Suns and Macs,
> I am looking for a program, that converts big <> low-endian byte order and
> IEEE <> VAX representation.

In IDL or PV-Wave, the bytes of each element of array A can be swapped
by
 BYTEORDER,A

I think IEEE is more difficult.

I once wrote a FORTRAN routine, but it has not been fully tested,
and is not portable to those fortran compilers that don't let you
equivalence numbers and characters:

-----CUT HERE-----

```
C-----
      function FromVaxR4(x)
c Function to convert Vax real*4 number to local floating point.
c Cannot handle NANs or numbers which are too small or too large.
c By mitchell r grunes.
      integer*4  x,y,i          ! Really Vax real*4--but
                                ! must be kept in integers
                                ! so won't be "normalized".

      character*1 a(4)
      equivalence (y,a)
      parameter (Mask23=2**23-1)
      parameter (ioffset=128+24)

      y=x
      i=      iand(ichar(a(2)),255)
      i=ior(ishft(i,8),iand(ichar(a(1)),255))
      i=ior(ishft(i,8),iand(ichar(a(4)),255))
      i=ior(ishft(i,8),iand(ichar(a(3)),255))

      iexponent=iand(ishft(i,-23),255)-ioffset
      mantissa=iand(i,Mask23)

      if(i.eq.0)then
        FromVaxR4=0
      else
        mantissa=ior(ishft(1,23),mantissa)
        if(i.gt.0)then
```

```

    FromVaxR4= mantissa*2.**iexponent
  else
    FromVaxR4=-mantissa*2.**iexponent
  endif
endif
end

```

C-----

```

function FromIEEEER4(x)
c Function to convert IEEE real*4 number to local floating point.
c Assumes number written on a "most significant byte first" machine like
c a Sun or SGI workstation.
c Cannot handle NaNs or numbers which are too small or too large.
c By mitchell r grunes.

```

```

integer*4 x,y,i      ! Really IEEE real*4--but
                     ! must be kept in integers
                     ! so won't be "normalized".

```

```

character*1 a(4)
equivalence (y,a)
parameter (Mask23=2**23-1)
parameter (ioffset=128+22)

```

```

y=x
i=      iand(ichar(a(1)),255)
i=ior(ishft(i,8),iand(ichar(a(2)),255))
i=ior(ishft(i,8),iand(ichar(a(3)),255))
i=ior(ishft(i,8),iand(ichar(a(4)),255))

```

```

iexponent=iand(ishft(i,-23),255)-ioffset
mantissa=iand(i,Mask23)

```

```

if(i.eq.0)then
  FromIEEEER4=0
else
  mantissa=ior(ishft(1,23),mantissa)
  if(i.gt.0)then
    FromIEEEER4= mantissa*2.**iexponent
  else
    FromIEEEER4=-mantissa*2.**iexponent
  endif
endif
end

```

C-----

```

function FromRIEEER4(x)
c Function to convert IEEE real*4 number to local floating point.
c Assumes number written on a "least significant byte first" machine like
c a PC.
c Cannot handle NaNs or numbers which are too small or too large.
c By mitchell r grunes.

```

```

integer*4  x,y,i      ! Really IEEE real*4--but
                    ! must be kept in integers
                    ! so won't be "normalized".

character*1 a(4)
equivalence (y,a)
parameter (Mask23=2**23-1)
parameter (ioffset=128+22)

y=x
i=      iand(ichar(a(4)),255)
i=ior(ishft(i,8),iand(ichar(a(3)),255))
i=ior(ishft(i,8),iand(ichar(a(2)),255))
i=ior(ishft(i,8),iand(ichar(a(1)),255))

iexponent=iand(ishft(i,-23),255)-ioffset
mantissa=iand(i,Mask23)

if(i.eq.0)then
  FromRIEEER4=0
else
  mantissa=ior(ishft(1,23),mantissa)
  if(i.gt.0)then
    FromRIEEER4= mantissa*2.**iexponent
  else
    FromRIEEER4=-mantissa*2.**iexponent
  endif
endif
end

```

Subject: Re: Help: Byte-swapping program
 Posted by [grunes](#) on Mon, 12 Dec 1994 15:39:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <schaep-1212941334430001@130.60.16.90> schaep@rsl.geogr.unizh.ch (Michael E. Schaepman) writes:

> Since we are moving our data from a VAX and a DecStation to Suns and Macs,
 > I am looking for a program, that converts big <> low-endian byte order and
 > IEEE <> VAX representation.

...

I may have been wrong!

I can't remember anymore, but I'm not sure a simple byte order reversal
 will fix the problem for 4 byte integers. VAX floating point numbers
 used a real strange byte order (see previous post), and it's possible
 that VAX 4 byte integers do too. In that case my previous suggestion
 of

BYTEORDER,A
will NOT work.

BTW, you could easily run into other problems from the fact that VAX and some other platform compilers frequently add extra little bytes into the file formats, and there is no agreement between platforms and sometimes compilers on what those bytes might be. For example, it is common to have a few bytes of header at the beginning, and some sort of record length before (and sometimes after) each record.

One of the nice things about IDL and PV-Wave is that (as far as I know) READU and WRITEU don't assume or add in any extra bytes. BUT I haven't used IDL/Wave much on the VAX, and it might be an exception.

(opinions expressed are mine alone)

Mitchell R Grunes (grunes@nrlvax.nrl.navy.mil)

Allied-Signal Technical Services

c/o Code 8140 Naval Research Lab

Subject: Re: Help: Byte-swapping program
Posted by [thompson](#) on Mon, 12 Dec 1994 16:04:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

grunes@news.nrl.navy.mil (Mitchell R Grunes) writes:

> In article <schaep-1212941334430001@130.60.16.90> schaep@rsl.geogr.unizh.ch (Michael E. Schaepman) writes:

>> Since we are moving our data from a VAX and a DecStation to Suns and Macs,
>> I am looking for a program, that converts big <> low-endian byte order and
>> IEEE <> VAX representation.

> In IDL or PV-Wave, the bytes of each element of array A can be swapped
> by
> BYTEORDER,A

> I think IEEE is more difficult.

The above statement is only correct if the array is short integers. For long integers one should use the keyword /LSWAP.

I don't know about PV-Wave, but IDL does contain the options /XDRTOF and /FTOXDR to convert between IEEE and the local host floating point representation. Similarly, one can use /XDRTOD and /DTOXDR for double precision.

The IDL astronomy user's library (anonymous ftp to idlastro.gsfc.nasa.gov) routines IEEE_TO_HOST and HOST_TO_IEEE does the appropriate conversions for all data types.

Bill Thompson

Subject: Re: Help: Byte-swapping program
Posted by [plutchak](#) on Mon, 12 Dec 1994 18:54:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <3chsab\$d78@paperboy.gsfc.nasa.gov> thompson@orpheus.nascom.nasa.gov (William Thompson) writes:

> I don't know about PV-Wave, but IDL does contain the options /XDRTOF and
> /FTOXDR to convert between IEEE and the local host floating point
> representation. Similarly, one can use /XDRTOD and /DTOXDR for double
> precision.
>
> The IDL astronomy user's library (anonymous ftp to idlastro.gsfc.nasa.gov)
> routines IEEE_TO_HOST and HOST_TO_IEEE does the appropriate conversions for all
> data types.

Yes, but neither of those seem to be sufficient. In my case (and I believe in the case of the original poster), we've got VAX floating point numbers to be read into IEEE format on a machine who's local format *is* IEEE. Unless I'm missing something in my reading of the manual, that's not covered by IDL's conversion functions.

I've got some code lying around somewhere that I wrote to go from VAX to IEEE floating point (single-precision), which I'll dig up if I don't see any other solution go by before I get back from my travels.

--

Joel Plutchak, Research Programmer/Analyst
Planetary Geology Group, Brown University, Providence, RI

Subject: Re: Help: Byte-swapping program
Posted by [grunes](#) on Mon, 12 Dec 1994 20:16:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

> Yes, but neither of those seem to be sufficient. In my case (and I
> believe in the case of the original poster), we've got VAX floating
> point numbers to be read into IEEE format on a machine who's local
> format *is* IEEE. Unless I'm missing something in my reading of the
> manual, that's not covered by IDL's conversion functions.
> I've got some code lying around somewhere that I wrote to go from
> VAX to IEEE floating point (single-precision), which I'll dig up if
> I don't see any other solution go by before I get back from my travels.

The Fortran code for routine FromVaxR4 that I posted in my original response would do the that part of the job. Since the person was

working on a Sun--in Sun Fortran, if you do
 open(1,file=name,access='direct',recl=1,form='unformatted')
and you want to look at the real*4 number at byte # n (where the first
byte is #1), you would say
 read(1,rec=n)i
and say
 x=FromVaxR4(i)
to convert it to local floating point. Similar open and write
statements could output to a local format file. Most other Fortrans
have some form of open to do a similar job.

However, I imagine most readers of this newsgroup prefer to use IDL
or PV-Wave. Perhaps someone with time on their hands would like
to translate my code to IDL/Wave?
(opinions expressed are mine alone)
Mitchell R Grunes (grunes@nrlvax.nrl.navy.mil)
Allied-Signal Technical Services
c/o Code 8140 Naval Research Lab

Subject: Re: Help: Byte-swapping program
Posted by [thompson](#) on Tue, 13 Dec 1994 15:50:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

grunes@news.nrl.navy.mil (Mitchell R Grunes) writes:

> One of the nice things about IDL and PV-Wave is that (as far as I know)
> READU and WRITEU don't assume or add in any extra bytes. BUT I haven't
> used IDL/Wave much on the VAX, and it might be an exception.

It is, but one can open up a file on the VAX with the /BLOCK qualifier and then
READU and WRITEU works just like it does in Unix. I use that in combination
with IEEE_TO_HOST and HOST_TO_IEEE to write OS-independent files.

Bill Thompson

Subject: Re: Help: Byte-swapping program
Posted by [thompson](#) on Tue, 13 Dec 1994 15:57:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

plutchak@lager.geo.brown.edu (Joel Plutchak) writes:

> In article <3chsab\$d78@paperboy.gsfc.nasa.gov> thompson@orpheus.nascom.nasa.gov
(William Thompson) writes:
>> I don't know about PV-Wave, but IDL does contain the options /XDRTOF and
>> /FTOXDR to convert between IEEE and the local host floating point

>> representation. Similarly, one can use /XDRTOD and /DTONDR for double
>> precision.
>>
>> The IDL astronomy user's library (anonymous ftp to idlastro.gsfc.nasa.gov)
>> routines IEEE_TO_HOST and HOST_TO_IEEE does the appropriate conversions for all
>> data types.

> Yes, but neither of those seem to be sufficient. In my case (and I
> believe in the case of the original poster), we've got VAX floating
> point numbers to be read into IEEE format on a machine who's local
> format *is* IEEE. Unless I'm missing something in my reading of the
> manual, that's not covered by IDL's conversion functions.
> I've got some code lying around somewhere that I wrote to go from
> VAX to IEEE floating point (single-precision), which I'll dig up if
> I don't see any other solution go by before I get back from my travels.

There are also some routines to handle this in the IDL astronomy user's
library. In the "misc" subdirectory there are the routines CONV_VAX_UNIX and
CONV_UNIX_VAX. They're not as fast as those mentioned above, but they'll do
what you want. Depending on how the file was written (i.e. the VMS record
attributes), you may also need the routine CONV_VAX_BLOCK.

Bill Thompson
