Subject: Re: signed character
Posted by Dick Jackson on Thu, 25 Apr 2002 13:19:58 GMT
View Forum Message <> Reply to Message

"Carmen" <cambicambi@hotmail.com> wrote in message
news:8128b197.0204250239.509a20fb@posting.google.com...
> Hello,
>
> I am trying to access information stored in a matrix, byte per byte.
> If I were using C, I should read every byte as SIGNED CHARACTERS.
> But, IDL has not that definition.
>
> I would appreciate a lot if you can tell me what should I do in order
> to read the data correctly.

Signed characters represent values -128 to 127, but IDL Bytes are treated as 0 to 255. I'd convert them to (short) integer, the simplest IDL data type that can express -128 to 127, as in this example:

```
IDL> a = [128B, 255B, 0B, 127B]
IDL> b = fix(a) + ([0, -256])[a GT 127B]
```

(convert to integer, subtract 256 from elements where needed... I think this will be the fastest method of doing this)

```
IDL> Print, b
   -128      -1       0     127
IDL> Help, b
B               INT       = Array[4]
```

If you've read data into a byte array, you may want a function to convert it:

```
FUNCTION SignedCharToFix, a

Return, fix(a) + ([0, -256])[a GT 127B]

END
```

Hope this helps.

Cheers,
--
-Dick

Dick Jackson / dick@d-jackson.com
D-Jackson Software Consulting / http://www.d-jackson.com
Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392

Subject: Re: signed character
Posted by Mark Rivers on Thu, 25 Apr 2002 13:24:52 GMT
View Forum Message <> Reply to Message

Carmen <cambicambi@hotmail.com> wrote in message
news:8128b197.0204250239.509a20fb@posting.google.com...
> Hello,
>
> I am trying to access information stored in a matrix, byte per byte.
> If I were using C, I should read every byte as SIGNED CHARACTERS.
> But, IDL has not that definition.
>
> I would appreciate a lot if you can tell me what should I do in order
> to read the data correctly.

If A is the matrix containing an IDL byte array (which IDL assumes is
unsigned) then you could do

```
; First convert A to a signed 16-bit integer array
a = fix(a)
; Find the indices of the negative elements (MSB of 8-bit value set)
negs = where(a and '80'x)
; Extended the sign to 16 bits
a[negs]=a[negs] or 'ff80'x
```

The drawbacks of this approach are that your matrix is twice as big (16 bits
vs 8 bits), and that you need to create the temporary array "negs". If your
matrix is only 2-D and not terribly huge then this should be fine.

Mark Rivers