
Subject: Re: Decomposing a bit field?

Posted by [Chris\[6\]](#) on Wed, 21 Jul 2010 22:41:04 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jul 21, 1:25 am, Rob <rj...@le.ac.uk> wrote:

> On Jul 21, 11:29 am, Steve <f...@k.e> wrote:

>

>

>

>> Rob wrote:

>>> for i=0, 31 do begin test[i]=(ISHFT((15982561),-i) AND 1B)

>

>>> However, for example bits 1-4 need to be combined (and contain data

>>> from 0-15). As I don't quite understand what the code above is doing

>>> I'm not sure how to (or if it can be) modified to not just operate on

>>> the individual bits but on groups of bits.

>

>> ISHFT - moves bits right or left so that the bits you want are at the

>> end of the value. AND selects the bits that you want.

>

>>> What I need to extract is the information in the following form:

>

>>> Bit 0: 0 or 1

>

>> bit0= ishft(15982561,0) and 1B

>

>>> Bits 1-4: 0-15

>

>> bits1to4 = ishft(15982561,-1) and 17B

>

>>> Bit 5: 0 or 1

>

>> bit5 = ishft(15982561,-5) and 1B

>

>>> Bits 6-8: 0-8

>

>> bits6to8 = ishft(15982561,-6) and 7B

>

>>> etc

>

>>> Thanks in advance

>

> Thanks everyone, that makes much more sense now.

>

> I just wanted to confirm where the 7B comes from above.

>

> I assume it's because I'm checking 3 bits so it's 1+2+4. If I had a

> group of 4 bits it's be 1+2+4+8 (15 like in Chris's example below),

> right?

right- 7 is (000000....111) in binary. So if you take "x and 7" it will set all but the last three bits to zero. So it throws away all the leading bits, and returns the last 3 bits as some number (0-7)

15 (0000.....1111) does the same thing, but with 4 bits instead of 3.
chris
