Subject: Re: Most significant digit and formatting floating point output
Posted by Helder Marchetto on Wed, 08 Aug 2012 20:20:28 GMT

View Forum Message <> Reply to Message

On Tuesday, August 7, 2012 5:37:31 PM UTC+2, Craig Markwardt wrote:
> On Tuesday, August 7, 2012 9:33:13 AM UTC-4, Helder wrote:
>
>> Dear all,
>
>>
>
>> well, I'm tangled up with something quite stupid, but I don't want to reinvent the wheel (for the 10th time today...).
>
>>
>
>> Here are two versions of the same question:
>
>>
>
>> 1) Short version: given a number such as 0.003456789 how do I get the most significant digit position? I need this number to produce formatted output that would convert the number to simply 0.003. In general I can figure this out with some IFs and stuff like that, but I'm hoping there is a more "elegant" way for this.
>
>>
>
>>
>
>>
>
>> 2) Long version: I'm analyzing some images and I get results with errors. These numbers are floating point and I would like to format the output so that it looks something like this: 'My results are (0.123 +/- 0.003) units' and the source data is:
>
>>
>
>> Result = 0.123456789
>
>>
>
>> Error = 0.003456789
>
>>
>
>> The point is that I need to find the first non-zero element in Error. Then I would use this number in the FORMAT parameter like this: MyFormat = '(f0'+STRTRIM(FirstNonZeroElement+2,2)+'.'+STRTRIM(FirstNonZ eroElement,2)+')'

>
>>
>
>> and use "MyFormat" as formatting for the string conversion of both.
>
>>
>
>> (in the above example FirstNonZeroElement should be 3 and MyFormat would be '(f05.3)'.
>
>
>
> ALOG10(ABS(X)) should give you this information. But to be general, you need to handle ABS(X) LT 1 and ABS(X) GT 1 separately. And also, the X LT 0 case demands one extra digit for the '-' symbol.
>
>
>
> Some extra rounding logic may be needed.
>
>
>
> Craig

Hi Craig,
thanks, that is what I was looking for.
Cheers,
Helder