
Subject: Re: Double precision data into caldat
Posted by [Paul Van Delst\[1\]](#) on Tue, 08 Sep 2015 20:58:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

On 09/08/15 15:37, Larry H. wrote:

```
> Hi,
>
> I'm trying to convert some Julian dates back into standard dates
> using caldat. (In fact, I am testing what I got out of julday in the
> first place.) There should be hours and minutes in the results. If
> I take the output of julday as a variable, say TEST, and put it into
> caldat as
>
> caldat, test, m, d, y, h, mm, s
>
> I get the right answer. However, if I just use the actual value of
> test, I get odd results. So, for the Julian day 2456658.56250000, I
> should get
>
> 2014      1      1      1      30      0
>
> in year, month, day, hour, min, sec form. If I use
>
> caldat, 2456658.56250000D, m, d, y, h, mm, s
>
> the results are correct, but if I do the type conversion using
> double(), it doesn't. So I have
>
> caldat, double(2456658.56250000), m, d, y, h, mm, s
>
> and
>
> p = double(2456658.56250000) caldat, p, m, d, y, h, mm, s
>
> both giving the result
>
> 2014      1      1      0      0      0
>
> Does anyone know why this is? As far as I know, all of those input
> values are the same.
```

Nope. They are not.

```
IDL> p=2456658.56250000
IDL> help, p
P      FLOAT    = 2.45666e+06
```

```
IDL> print, p, format='(f25.10)'
2456658.5000000000
```

Single precision has about 7-8 significant digits.

So for a value you write as "2456658.56250000" -- which is a single precision literal constant -- you've only got "2456658.5" of useful digits.

When you use DOUBLE, as in:

```
IDL> p=DOUBLE(2456658.56250000)
```

you are taking a SINGLE PRECISION literal constant (2456658.56250000) and converting it to double. So you still lose the extra information (the "X.X625") stuff, e.g.

```
IDL> p=DOUBLE(2456658.56250000)
IDL> help, p
P      DOUBLE  =    2456658.5
IDL> print, p, format='(f25.10)'
2456658.5000000000
```

Now, if you declare a double precision literal constant, like so:

```
IDL> p=2456658.56250000d0
IDL> help, p
P      DOUBLE  =    2456658.6
```

you are "declaring" a double precision literal so you have about 15 digits of precision-y goodness which you see when you print it out:

```
IDL> print, p, format='(f25.10)'
2456658.5625000000
```

Basically, if you ever declare literal constants in code where those values can have more than 8 useful digits, always use double precision declarations. Converting a single precision 8+ sig fig number to double will have no effect.

Hope that helps.

cheers,

paulv
