Subject: Re: modulo reset
Posted by Wayne Landsman on Tue, 23 Nov 2004 15:52:53 GMT
View Forum Message <> Reply to Message

Ralf Schaa wrote:
> I'm reading out some binary data and the description says this about a
> data field :
>
> in the case of a modulo reset add 2^32
>
> What is this about? googling for "modulo reset" didn't get me far;
> I only imagine that I need some kind of overflow to get things fixed,
> but it is very unclear to me what is happening and why ...

Well, my first guess is that you are working with unsigned 32bit
integers which have a maximum value of 2^32-1, and then resets to zero.

```
STIS>print,2UL^31,2UL^32
  2147483648           0
```

The easiest way to correct for this reset would be to use 64 bit integers

```
data = ulong64(datafield) + 2ULL^32
```

A less likely possiblity is that "modulo reset" refers to a 32bit
checksum, e.g. as implemented in
http://idlastro.gsfc.nasa.gov/ftp/pro/misc/checksum32.pro
which includes links to more documentation.

Good Luck, --Wayne Landsman

Subject: Re: modulo reset
Posted by Ralf Schaa on Tue, 23 Nov 2004 16:39:58 GMT
View Forum Message <> Reply to Message

Wayne Landsman wrote:
> Ralf Schaa wrote:
>
>> I'm reading out some binary data and the description says this about a
>> data field :
>>
>> in the case of a modulo reset add 2^32
>>
>> What is this about? googling for "modulo reset" didn't get me far;
>> I only imagine that I need some kind of overflow to get things fixed,
>> but it is very unclear to me what is happening and why ...
>

>
> Well, my first guess is that you are working with unsigned 32bit
> integers which have a maximum value of 2^32-1, and then resets to zero.
>
> STIS>print,2UL^31,2UL^32
> 2147483648 0
>
> The easiest way to correct for this reset would be to use 64 bit integers
>
> data = ulong64(datafield) + 2ULL^32
>
> A less likely possiblity is that "modulo reset" refers to a 32bit
> checksum, e.g. as implemented in
> http://idlastro.gsfc.nasa.gov/ftp/pro/misc/checksum32.pro
> which includes links to more documentation.

thanks wayne,

but the problem is this:

I am reading binary data (not longer than 32 bit, and I store it in ULL as suggested)
and the 'modulo reset' I talked about may appear at one datafield: that is in an accumulated
"Doppler" cycle count.
By differentiating with respect to time, one can get the true doppler count.
Than the documentation says, when a modula reset occurs , add 2^32.
I think, this means when the counter is full and is starting with zero again. than add the 2^32.
But I don't see what adding 2^32 exactly would do ...

further suggestions?

-Ralf

Subject: Re: modulo reset
Posted by James Kuyper on Tue, 23 Nov 2004 18:51:30 GMT
View Forum Message <> Reply to Message

Ralf Schaa wrote:
...
> I am reading binary data (not longer than 32 bit, and I store it in ULL
> as suggested)
> and the 'modulo reset' I talked about may appear at one datafield: that
> is in an accumulated
> "Doppler" cycle count.
> By differentiating with respect to time, one can get the true doppler
> count.

"Differencing", not "Differentiating". You differentiate a continuous

function of time. For a discontinuously sampled function, you can't differentiate, you can only calculate finite differences.

> Than the documentation says, when a modula reset occurs , add 2^32.
> I think, this means when the counter is full and is starting with zero
> again. than add the 2^32.
> But I don't see what adding 2^32 exactly would do ...

Let's assume that the current cycle count is t0=2^32-5. 20 cyles later the true count would be 2^32+15. However, because it reset at 2^32, the actual number in the cycle count would be t1=15. If you calculate the time difference as dt = t1-t2 while storing the value in, for instance, a 64 byte integer or floating point type, then the dt will be 15-(2^32-5) = 20-2^32. To get the correct number of cycles, you have to add in 2^32, leaving you with dt = 20.
