From: STSCIC::FITCH 11-JAN-1993 13:45:15.15
To: BALZANO,CHANCE
CC:
Subj: GIMP status buffer correction, vicki the dacs are 12 bit signed

From: CASS::RICK 11-JAN-1993 10:57:37.64
To: STSCIC::FITCH,CASS05::BAITY,RICK
CC:
Subj: Status buffer messages occuring in GIMP Fix tests

John:

Since our phone conversation Thursday, I have done further analysis of the situation. This has confirmed the basics of our discussion, with some additional clarification. A summary of the matter, with proposed corrective action, follows.

As you are aware (but included here for completeness), the present microprocessor code for the GIMP fix is uplinked as:

: XO PARA 8 + C@ 8SEX + ; ' XO XHIST !<SPACE>
: YO PARA 9 + C@ 8SEX + ; ' YO YHIST !<CR>

with the ASCII <CR> sent following the entire text, then verifying receipt of the ok<CR><LF> in the Firmware Status Bytes at the appropriate place in the Telemetry Major Frame. I've shown it here on two lines so that it would fit, but it's treated as a single line of 78 characters (less than 80, as required) including the <CR>.

The GIMP correction code takes each of the appropriate computed deflection values and adds the small bipolar GIMP corrections (8 bit signed, range -128 -> 127, uplinked through the PARA tables) to them before being passed to the deflection DACs by the DAC Load and Verify (DL&V) firmware procedure. Note that the DACs receive input as 12 bit signed quantities (range -2048 -> 2047), and that in normal operation (without the GIMP fix) the values are computed as 16 bit signed quantities (range -16384 -> 16383) and then masked to 12 bits before passing to DL&V, whose job it is to verify that the DAC hardware has actually received and loaded those 12 bits correctly. There is no internal overflow checking for this computation; that is, if BASE, STEPS, RANGE, PITCH, etc. are such that DL&V would receive data for some steps that would be beyond 2047, they effectively "wrap around" (i.e. 2045, 2046, 2047, -2048, -2047, ...) and no errors are generated.

Recognizing great desire to limit the overhead required for the addition of a GIMP correction term, pains were taken to keep the uplinked microprocessor code below the 80 character limit, so that only one "line" needs to be sent, acknowledged, and verified. Note, for example, that the 12 bit "pre-GIMP" value is not sign-extended for (16 bit) addition and then re-masked for DL&V, since the 12 bits of the calculated value sent to the DAC are correct without this treatment. The operational requirement remains as before, to not ask (through deflection pattern parameters) for deflections outside the capability of the DACs; for example, a "pre-GIMP" value of 2045 with a GIMP correction of 5 should normally be avoided operationally since it cannot be handled by the DAC without wraparound.

However, it now appears that this does not take into account that the verification done by DL&V compares all 16 bits of the computed value given to it with the DAC readback (the masking to 12 bits was done pre-GIMP before passing the value to DL&V. Hence for cases of deflection near zero such that the small GIMP correction of a deflection value causes a change in sign (either

positive => negative or negative => positive), DL&V reports an error (since even though the 12 bits sent to the DAC are correct, the high order 4 bits of the computed value will not all be zero in this case) and a status buffer message results.

The problem arises since the spot available for insertion of the G P correction (which, by the way, was originally intended for hysteresis deperm rather than computation) lies between the 12 bit masking (done as part of the normal deflection computation) and DL&V. It seems that the most effective patch will be to re-mask the GIMP-corrected computed value as part of the GIMP correction microprocessor code (which unfortunately will push it above the 80 character limit, and thereby require more overhead). Such revised code, including everything as discussed above, would be:

```
HEX : XO 12SEX PARA 8 + C@ 8SEX + 0FFF AND ; ' XO XHIST !<CR>
: YO 12SEX PARA 9 + C@ 8SEX + 0FFF AND ; ' YO YHIST !<CR>
```

Two lines are now required, and therefore two corresponding separate verifications of the Firmware Status Bytes (probably requiring two Major Frames). The sign extension of the incoming value (12SEX) could be eliminated (as was done before) as unnecessary, since the 12 bit computed result is correct regardless (addition is nice that way); however, it may serve in the interest of clarity, and we already need two lines anyway to do the masking. Also, the leading "HEX" is to ensure that the newly added 0FFF will be interpreted correctly (the PARA offsets 8 and 9 were cleverly chosen for the earlier code to be interpretable without worrying about numeric base), and could be eliminated with further effort. The "briefest" code could then be either:

```
: O PARA 8 + C@ 8SEX + 4095 AND ; ' XO XHIST !<CR>
: YO PARA 9 + C@ 8SEX + 4095 AND ; ' YO YHIST !<CR>
```

or

```
: XO PARA 8 + C@ 8SEX + 0FFF AND ; ' XO XHIST !<CR>
: YO PARA 9 + C@ 8SEX + 0FFF AND ; ' YO YHIST !<CR>
```

depending on which number base is applicable at this point; this could be determined with further testing, perhaps on the simulator here. Again though, the small amount of brevity that these eliminations afford may not be necessary at this point -- unless someone thinks strongly otherwise, I recommend the full earlier stated lines for the "belt and suspenders" approach, to help avoid possible confusion later.

One final note: overflow checking of the computed deflection values could potentially be performed here too (with some more code), if desired, but only to see if the GIMP correction caused an overflow (any overflow caused by PATTERN specification has already been masked at this point); let me know if that is an issue, and we can look into details.

Rick

P.S. Please also let me know what happens with the rest of the G P testing. A possibility occurs to me: On the issue of the science data showing residual GIMP, was it properly taken into account that the GIMP correction values are in direct DAC units, and therefore need to be PITCH corrected?