Floating Point Data Requests

Just the engineering units, please!

Mar 3, 1989

The VME systems include a local database of the scale factors for every analog channel. Normally, a data request for the reading of a channel returns the raw channel reading value. The user is expected to separately collect the scale factors and use them to scale the results. This approach has always been taken in consideration of the limited floating point computational abilities of the local stations, which used an 8 MHz 68000 processor. A floating add took about 100 μs while a floating multiply took 150 μs.

Now that the stations are being constructed out of 20 MHz 68020 processors plus a 68881 floating point co-processor, it should not require much time to do such conversions to engineering units. The computation required is a conversion of the raw data value to a floating point fraction of full scale, a floating multiply by the fullscale value for that channel, and an optional floating addition of an offset value.

The types of data which need these conversions are readings, settings, and nominal and tolerance values. Four new listypes are designed for this purpose. Only systems updated with this feature will be able to handle such requests, as each local station which sources a channel's data must do the conversions; it cannot be done only by the requesting node, of course, as the scale factors only reside in the local data base of the original station. But any station will be able to make such requests, even if it doesn't support the feature itself.

In the listype table, the entries for the four new listypes should indicate the table# for the `ADATA` table, and the `ADESC` entries must be inferred from it. When these two tables are combined into one, this will be easier. Let the new listypes be 40, 41, 42, and 43, in parallel with 0, 1, 2, and 3. Listype 44 could support the delta setting referred to below.

Systems not using a 68020 could use their software floating point routines for the computations, assuming the use of the feature would not be extensive. Or, such systems could merely decline to support such requests.

Setting support should also be added for these four new listypes. This involves working the linear formulas backwards and watching for the possibility of dividing by a fullscale value of zero. If overflow is reached, the setting should either be clipped to fullscale or not executed at all. A delta setting should also be included for completeness.

The number of bytes associated with the listype used must be 4, or the request is not processed. It could be expanded to allow for 8 bytes also, if double precision values are desired. This would be easy to do, as the 68881 does all the hard work. One would probably have to implement double precision scale factors for this to be meaningful, however.

Note that neither the tolerance conversion nor the delta setting uses the offset value. Reading, nominal, and tolerance use the A/D scale factors, while setting and delta setting use the D/A scale factors. Out-of-range setting values are clipped to the nearest end of range. Or, one might choose to ignore settings which are *way* out of range.

Internal details:

A new read type routine #11 is used to process the engineering units scaling. The new set type routine #13 handles settings with this extra processing, while set type routine #14 handles the delta setting case.

The use of register D5 to hold the field offset in the ADATA entry must be presumed on entry to the setting handler SETENG or SETENGD. This provides the offset needed by the set routine to determine the type of processing required to perform the setting.

This engineering units feature is supported by the three procedures in the (new) READENG module: READENG, SETENG, and SETENGD.