Composite Digital Status Words

Acnet adaptations

Nov 13, 1991

Digital data: Local station vs Acnet

The local station software provides access to digital status by Bit#, by Byte#, and by Chan#. The latter uses the associated digital status and control fields in the analog descriptor to provide the parameter page on/off and reset support. Binary alarm scans are made based upon the individual Bit#, so an alarm directs attention to a single bit.

In contrast, Acnet treats a device as including both the analog channel reading and a digital status word reading. An alarm can refer to either the digital status or the analog reading of a single device.

Local station planned changes

Two schemes are planned as additions to the local station. The first is the feature of building words of digital status and assigning them to pseudo analog channel reading words. With a new flag bit in the analog alarm flags word, the analog alarm scan logic is modified to treat the nominal and tolerance words not as numbers, but as a nominal bit pattern and mask. In this way, a set of bits (in the reading word of an analog channel) can produce a single alarm message.

A second scheme is one of enhancing the current associated status and control support to collect up to 8 (or 16) bits of status together. This would permit requesting such status, but it would not include alarm scanning logic on such status. This note focuses on the first scheme.

The problem

The problem at hand is to find some way that combines these efforts into one which can fulfill what Acnet expects and also what the local station expects. These are the requirements:

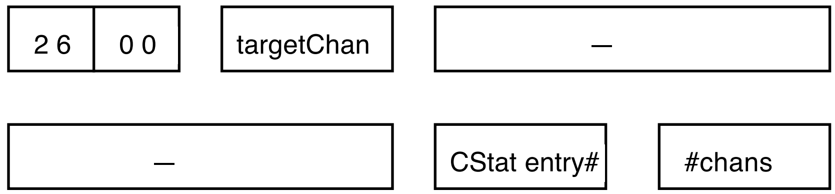
1. Build status words by device.
2. Report single alarm for the status word.
3. Report analog/digital alarms for single device.
4. Relate status word to analog channel and *vice versa*.
5. Support multiple digital controls for single device.
6. Provide text for each status word bit.
7. Provide state text for each status/control bit.
8. Support named status word if no related analog channel.

The solution

Suppose that associated status words are constructed each cycle and written into "reading" words of pseudo-channels, according to the first scheme. Also, suppose that a word in the pseudo-channel's analog descriptor refers to an analog channel that relates to the pseudo-channel. Then the pseudo-channel can be used to construct the EMC for the alarm system.

If these status words are used in the alarm scan, they must be built each cycle, not merely upon a user's request. This implies that it is good to do this efficiently. A table of instructions that describe how to do this efficiently can be interpreted by an offline database uploading utility program in the same way that it is interpreted by processing the data access table in the local station system in order to discover which raw status bits occupy positions in the composite status word.

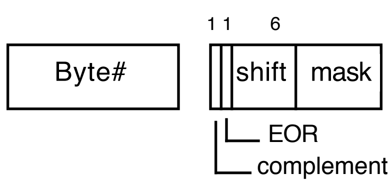
The data access table entry that supports this assembly of status bits is as follows:



The "targetChan" is the first analog channel whose reading word is to be filled with combined status bits. This word will be accessed using the Basic Status property of the central database. The "CStat entry#" is the entry# in the CSTAT system table that contains the list of 4-byte specifications that describe how to assemble the bits together for the first target channel. The system table directory specifies what is the maximum size of each list and hence the offset from the start of one list to the next. The "#chans" word specifies the number of resulting composite status words to be assigned to consecutive channel readings and also the number of lists that are to be interpreted.

If we assume that lists need no more than sixteen 4-byte specifications, as there are only 16 bits in a word, then the stepSize parameter could be 64 bytes. But in practice, it is likely that much fewer than 16 specifications will be needed, as one specification can operate on multiple bits occurring in a raw status byte. One might choose a list size of 32 bytes, for example, to permit up to 8 specs each.

The format of one 4-byte specification is as follows:



The Byte# refers to the raw status byte read by the usual "0405" data access table entry. Raw status byte values are placed into the BBYTE table (#5) by that entry. The mask is applied to the status byte reading as it is positioned in the lo byte of a word with zero in the hi byte. This word is shifted left by the amount of the shift. To shift right, use (16–shift), since it is actually a circular left shift.

The two option bits allow a complement of the status byte before masking and/or the use of exclusive-OR logic rather than the default inclusive-OR for combining the results of multiple 4-byte specs in a list into one composite status word. These two options should *not* be used for filling these composite status words in channel reading words, because they will complicate the job of uploading bit-based text to the central database. They are included for use in building composite status bytes that are placed into table #5 by using $2605 in the data access table entry, rather than $2600. (Table #0 is the analog data table.) In that case, new status bit text can be entered, and such new bits can be further combined into channel reading words as above. The whole matter of these two option bits can be considered an advanced topic that is unneeded for awhile.

In forming these 4-byte specs, the Byte# word must have a value in a valid range. The value zero is not considered valid, even though Byte #0 exists. The reason is that it is likely that many spaces reserved for specs will be unused and left zero.

It will take some care in devising the appropriate specs, but that is the nature of what is needed for doing the job of constructing these composite status words. Interpretation of the specs is expected to be quite efficient, which is desirable as it has to be done at 15 Hz.

Note that a single bit can be used as a composite status word in a trivial way using only one spec. This can give that bit an Acnet-style name. Raw status bits in a local station do not currently have names *a priori*.

To relate a composite status word "channel" to a real analog name, one could use the family word in the analog descriptor for that purpose. Alternatively, special (but similar) names can be defined for these composite status words.