NATIONAL RADIO ASTRONOMY OBSERVATORY

Charlottesville, Virginia

July 31, 1974

VLA COMPUTER MEMORANDUM #114

MONITOR DATA DEMULTIPLEXING

Preliminary Specification

Barry Clark

There are throughout the VLA system a large number of monitor points, which can be checked for detection of malfunctions and used to localize and debug a detected fault. Because of the finite communication bandwidth provided to the distant antennas, and because of the expense of individual A/D converters, an elaborate scheme of multiplexing is employed. It would be possible to handle this time multiplexed serial data stream in several different ways. One would be to take each word as received, consult its multiplex address (that is, its origin), and, depending on that address, to perform whatever logic, limit checking, or logging is required for that test point on that device. However, I propose to implement a different system, which I believe to be rather more flexible. That is, to demultiplex all of the monitor point information, and to thereby construct, in computer core, an image of the VLA electronics system, with all of its test points present and computer readable. The limit checking and logging programs then operate independently and asynchronously on this digital analog of the VLA electronics.

However, it is anticipated that the limit checking and logging routines will be slow, and run relatively infrequently. Therefore, the demultiplexor routines, which are real-time and synchronous, will also perform a few additional services (on request), most conveniently described in terms of their electrical analogs -- time constants and peak detectors.

This memo concerns itself only with demultiplexing the antenna-associated data (that from local buffers 1 to 27). Other programs, yet to be specified, will deal with the various control room datasets.

The Digital Communication System (DCS) will send data to the computer system through the computer DMP, interupting at a 19.2 Hz rate when transmission is complete. This interupt will, among other actions, activate the demultiplexing task. The first action of the demultiplexing task is to check on the integrity of the transmission. The total number of computer words transmitted must be a multiple of six (each responding dataset sends two 40 bit DCS words, each of which is expanded into three 16 bit computer words), and the recognition pattern ($xx010101_2$) must be in the most significant byte of the last DCS word transmitted. The master parity error indicator may be checked, and, if parity is all correct, the word-by-word checking may be surpressed (if convenient) to save processing time.

The demultiplexor routines are called for each word which comes in from the DCS. They first test the 'no response' bit, and, if it is set, return immediately. They next check the parity bit, and, if it is set, branch to a special parity error routine. This last stores the monitor word in a rotating buffer (to preserve a little history) along with the time and date. The buffer pointer (without the cyclic feature) is also a useful error counter.

Then the serial line number is checked to verify that the word in question comes from an antenna local buffer. If not, special routines for the special devices are called. Each dataset returns two monitor words each cycle. These have rather different applications. The first monitor word comes from multiplex addresses according to a ROM generated cycle repeating once every ten seconds. This cycle will access all variables necessary for routine monitoring of the array performance. There may be other monitor points which are checked on a very much less frequent basis (pehaps every time an observation is set up, typically once an hour). These are accessed through monitor word two, which is also used to intensively monitor test points under suspicion of misbehavior. In its idle mode, monitor word two sequentially steps through all of the monitor points connected to the dataset. This provision is so that any monitor point is accessible to a data tap at least once every 10 seconds. Data taken in sequential mode for monitor word two will not be processed or logged by the computer system; it is presumed to be of no permanent interest.

Monitor word two is used primarily for engineer and technician testing and debugging. A typical usage is, for instance, that the engineer will insert monitor point random select commands in the manual buffer of command word 4 to monitor all of the test points he is interested in, and will watch the results on a data tap as various parameters are changed. He may, at the end of things, wish to review the data he has taken. Therefore, the following actions are specified for monitor word 2.

If the dataset is in sequential mode for word 2 (a bit for each dataset telling whether it will be kept in core) monitor word 2 will be ignored.

If the dataset has been placed in random mode by the initial observation setup routines, monitor word 2 will be passed to special purpose transient routines.

If the dataset has been put in random mode by a manual command from the operator, the data is streamed, without modification, onto a circular buffer on disk, so that it may be reviewed at will later.

Monitor word one contains the normal array monitor data. For monitor word one, a routine is called which tests the dataset address and returns (depending on the DSA) the addresses of two tables, one of which describes the analog monitor point processing, and the other of which describes the digital monitor points. The branch to the appropriate routine (analog or digital) is made on the most significant bit of the multiplex address.

For analog monitor items (detected by noting that the leftmost bit of the MPXA byte is reset), the 24 bit data word is split into two halves and the MPXA for each is used to offset from the table address to find a two-word description of how the word is to be handled. The first word of this description is a pointer to the buffer area for this monitor point. The second word is a description of what is to be done to the word, as follows:

Bit 0 -- set indicates a dummy or open MPXA--ignore word

Bit 1 -- set indicates peak detection--compare with stored high and low limits, replacing previous limit if it exceeds it.

Bits 2,3 -- stage one time constant--multiply, previous integrated value by 1-2**(-2*n + 1), where n is the value (1 to 3) of these two bits, then add current value of the variable.

Bits 4,5 -- stage two timeconstant--increment a counter; when it exceeds 32, multiply previous stage two integrated value by 1-2**(-2*n + 1), where n is the value (1 to 3) of bits 4,5; then add current value of stage one integrator after shifting it right 5 places. (This two stage integrator scheme allows time constants up to about 3 hours to be simulated.)

Bits 6,7 -- spare.

Bits 8-15 -- buffer entry length.

Each non-dummy monitor point will generate a one or two word buffer entry -- one for the current value of the variable and (if needed) one for an error count. The use of an error count allows the programmer to specify that an error message is generated only if the monitor point exceeds its limits at more than a specified rate. Whenever peak detection is specified, the entry length is increased by two words, one to hold peak high and one to hold peak low. Whenever TC1 is specified, the entry length is increased by one, to accomodate the integrated value. Whenever TC2 is specified, the entry length is incremented by two, one word for the counter, one for the integrated value. The total entry length required is entered into the right byte of the control word, and thus ranges from 0 (dummy words) to 7 (current value, error counter, time counter, two integrated values, and two peak values). To avoid confusion, it seems wise to restrict the possibilities somewhat as follows: first, TC2 may not be specified unless TC1 is 3. Second, (rather more opaquely) peak detection is permitted only if an error count is specified. With these rules, the address of the current value is

(buffer pointer) + (line number - 1)*(entry length)

The address of the error count (if any) is

(buffer pointer) + (line number)*(entry length) - 1

The address of the peak detection words, if any, are

(buffer pointer) + (line number)*(entry length) - 2 and -3

The address of the first stage time constant is

(buffer pointer) + (line number - 1)*(entry length) + 1

And the address of the second stage time constant and counter (if present) is

(buffer pointer) + (line number - 1)*(entry length) + 2 and +3

The digital monitor points appear to consist of three types: 1) the closed loop return of values sent, 2) the return of one-bit warning or error flags, and 3) the antenna axis positions. None of these seem to require the synthesized time constants that are appropriate for some analog quantities. Instead, in order to detect an intermittent flag in the warning bits of a frequently monitored digital point, it may be desireable to make an 'OR' of all of the monitor bitstrings received in a given time. Because the warning bits are hardware complemented, if necessary, to make normal operation return a zero bit, this should suffice. However, it is possible to imagine situations in which one wishes to know whether a flag has been continuously set. For this purpose we shall optionally make available the capability to complement the bit string before ORing it.

A ditital monitor word consists of a bit string (that is a string of one-bit logical variables) left adjusted in the 24 bit data word, followed by a digitally expressed value in the right half. These parts of the word are split apart before storing, at a point specified in the monitor point description word. The significance of this word is as follows:

Bit 0 -- bit string part of word is not present, discard left hand bits (if any).
Bit 1 -- value part of word is not present, discard right hand bits (if any).
Bit 2 -- the bit string part is to be ORed with the previous string.
Bit 3 -- the bit string complemented is to be ORed with a previous string.
Bit 4-8 -- the division point between bit string and value. Note that, if this number is greater than 16, two words must be allocated to hold the bit string; if it is less than 8, two words must be allocated to hold the value.
Bit 9-15 -- buffer entry length (as in the analog case).

Here also, we wish to make a restriction to make the buffer addressing easy. The restriction we impose is that the complement OR will not be called for unless the uncomplemented OR is called for. The string part of the word will be stored in

(buffer pointer) + (line number - 1)*(entry length)

The value is stored in this location, this location + 1 or this location + 2, depending if the control word bits 0, 5, or 4, respectively, are set. The ORed bit string is stored in location

(buffer pointer) + (line number)*(entry length) - 1 - (bit 4)

and the complement ORed string in location

(buffer pointer) + (line number)*(entry length) - 2 - 2* (bit 4)

The in-core buffer areas will start with a control word similar to the ones described above, and a second word with the identifying DSA and MPXA and a few bits of description of the logging required for the point. Specifications for the logging programs, and for other programs operating on the demultiplexed data will be worked up and described elsewhere.