VLA Computer Memorandum #185

Synchronous System Data Formats

K. Sowinski

March 5, 1986

This memo documents the VLA archive tape format in use from June 8, 1975 through December 1987. Tapes containing this data are known as "A" tapes. Data taken earlier is documented in VLA Computer Memo #126, which this memorandum supersedes. In general, the existence of programs to transform "A" tapes into the form of current VLA archive tapes makes detailed knowledge of the contents of this memorandum unnecessary; this publication is primarily for historical purposes. Data taken from January 1988 on is documented in a companion document: VLA Computer Memorandum #186.

Once every integration time, a record set will be written from BOSS to the fixed-head disk to be read by the DEC-10, and also stored on an archive tape. This set consists of one logical record, as described below, per active subarray.

The record written on the tape will be divided into units 1024 words long, a convenient buffer length. The format will be DEC-Magtape format (DMF), explained in appendix B. The record written on the FHD will be divided into segments 128 words long, the width of a sector. The format will be MODCOMP non-standard binary; a hardware conversion will read this into the DEC-10 as left-adjusted halfwords, with unused bits set to zero. The first sector (sector 0) of the FHD contains the following:

Halfword 0 = ptr to first sector of current 10-second data.
Halfword 1 = ptr to next available sector.
Halfword 2 = ptr to last sector written by monitor data logger.
Halfword 3 = ptr to last sector written by checker data logger.

We are only concerned with the first two halfwords in this memo. Only 2047 of the 4096 sectors are used for this data; when a record set would write beyond 2047, the pointer is reset to sector 1.

In either case, spanned records will be written. No blocking will be done. The first word of the tape record contains, right adjusted, the physical record length of each record on the tape. The first word of the disk record and the second word of the tape record contain information about the spanning. The leftmost sixteen bits of the right halfword will contain, right adjusted, the number of blocks in this logical record. Thus, end-of-record is detected by (left halfword = right halfword). Having described these control words, we shall omit all further reference to them. That is, we shall refer to the first halfword following the first spanning control word as Halfword 0, the right halfword of that word as Halfword 1, etc.; and if, say, the first block is 1000 halfwords counting the spanning control word, the first halfword following the spanning control word of the second block will be called Halfword 998 of the record. From now on, the word "record" will mean "logical record."

There are currently four record format types. Formats 3 and 4 are monitor-type data and do not concern us here. Listed below is the layout of formats 1 and 2. The numbers given in square brackets ([]) are revision levels. Items marked with an asterisk (*) are not yet implemented; items marked with a plus sign (+) are only partially implemented. Partially implemented items will contain harmless but unmeaningful data, but unimplemented items should be entirely ignored. Numeric data types are identified by symbols:

(FP) = 32-bit standard MODCOMP floating point.
(DP) = 48-bit double-precision MODCOMP floating point.
(HP) = 16-bit special VLA half-precision floating point.
(B+n) or (B-n) = 16-bit scaled binary integer.
(S+n) or (S-n) = 32-bit scaled binary integer.
(no symbol) = 16- or 32-bit binary integer.

See Appendix A for more detail of the numeric data types.

FORMAT 1 DATA. This is the visibility data. Each record is divided into several logical data areas, shown below. Any item not preceded by a revision level applies to all revisions after level 3 (all revisions before level 3 are considered obsolete).

The Record Control Area (RCA) is given below. This record area contains global data information and pointers.

Halfword	Description
0	Logical record length, halfwords.
1	Unused
2	Format type (description below is format 1).
3	Format revision level. [3] - 3 (implemented 76/06/28). [4] - 4 (" 77/01/24). [5] - 5 (" 78/05/17). [6] - 6 (" 79/04/05). [7] - 7 (" 82/04/01). [8] - 8 (" 83/06/08). [9] - 9 (" 83/06/17). [10] - 10 (" 83/08/10). [11] - 11 (" 84/02/16). [12] - 12 (" 86/03/05).
4,5	Date--MJAD.
6,7	Time--Count of 19.2 Hz. interrupts. [3] - Count at time of record assembly. [5] - Count at last ten-second interrupt.
8	Control program I.D. [3] - Not implemented. [6] - "CO" = continuum, "LI" = spectral line. Note: For the interim spectral line system a different set of Modcomp programs must be loaded for spectrometer observations than those used for continuum observations. Note further, that in the future this field may be used for other purposes. The Correlator Mode field should be used to distinguish between Line and Continuum data.
9	Unused.
10	Pointer to Subarray Data Area (SDA).
11	Length of each Antenna Data Area (ADA).
12	Pointer to ADA.
13	Number of antennas in this subarray.
14	Pointer to Bad Correlator Data Area (BCDA). +
15	Number of bad correlators (zero if not present). +
16	Pointer to first correlator data area.
17	Number of correlators.
18	Pointer to second correlator data area. [3] - Duplicate of word 16 above. [8] - Actual BD data implemented.
19	Number of correlators. [3] - Duplicate of word 17 above. [8] - Actual BD data implemented.

The format of the Subarray Data Area (SDA) is given below. This area contains information peculiar to each subarray, but global to all data coming from this subarray. It begins in the halfword pointed to by halfword 10 of the RCA.

Halfword	Description
0	Subarray I.D.
1	Unused.
2 to 5	Source name (8-bit ASCII, leftmost bit 0).
6	Source name numeric qualifier.
7	[3] - Unused. [8] - Array configuration (ASCII).
8,9	Observer or program I.D. (ASCII).
10	[3] - Observer's DEC-10 number (octal). [8] - Continuation of observer I.D. wds 8,9.
11	[3] - Unused. [8] - Observer's DEC-10 number (octal).
12,13	Observation mode descriptors. [11] - Mode descriptors no longer distinguish Line from

	Continuum.
14	Array status byte.
15	Gain code.
16	Number of complex correlators per baseline. [3] - 4 (Continuum system). [6] - 4 if continuum mode; number specified by bandwidth code if line mode.
17	[3] - Unused. [6] - Averaging time (number of ten-second intervals per dump (1 if continuum mode)). [8] - Averaging time in w.g. cycles.
18 to 19	Stop time, LST, radians (FP).
20 to 21	Start time, LST, radians (FP).
22 to 25	Source RA, 1950 or other epoch, radians (DP).
26 to 29	Source DEC, 1950 or other epoch, radians (DP).
30 to 33	Source RA, current time, radians (DP).
34 to 37	Source DEC, current time, radians (DP).
38 to 41	LO 1, GHz (DP).
42 to 45	LO 2, GHz (DP).
46 to 49	[3] - LO 2, GHz (DP). [8] - LO 3, GHz (DP).
50 to 53	[3] - LO 2, GHz (DP). [8] - LO 4, GHz (DP).
54 to 57	IAT time, end of interval, radians (DP).
58 to 61	LST time, end of interval, radians (DP).
62 to 63	Current surface refractivity N-1 (FP).
64 to 65	Estimated zenith atmospheric phase path, nsec. (FP).
66	Sin(h), end of interval.
67	Cos(h), end of interval.
68	Cos(A), end of interval.
69	Sin(A), end of interval.
70	Cos(eta), end of interval.
71	Sin(eta), end of interval.
72,73	Bandwidth selection (coded, see Appendix C).
74	Array control bits (logical binary).
75 to 79	[3] - Nonexistent. [4] - Weather station information.
80 to 83	[3] - Nonexistent. [10] - Velocity of source, I.F. A, Km/s (DP).
84 to 87	[3] - Nonexistent. [10] - Velocity of source, I.F. B, Km/s (DP).
88 to 91	[3] - Nonexistent. [10] - Line rest frequency, A, MHz (DP).
92 to 95	[3] - Nonexistent. [10] - Line rest frequency, B, MHz (DP).
96	[3] - Nonexistent. [10] - Velocity rest system code (ASCII). [12] - Velocity definition code (ASCII).
97 to 98	[3] - Nonexistent. [10] - I.F. / Correlator mode (ASCII). [11] - This field now contains meaningful information.
99 to 100	[3] - Nonexistent. [10] - A.P. options (ASCII).
101	[3] - Nonexistent. [10] - Epoch year.
102	[3] - Nonexistent. [10] - Autophasing I.F. designation (ASCII).
103 to 107	[3] - Nonexistent. [10] - No. / Beginning channel for five I.F.'s.

The Antenna Data Areas (ADA's) follow, one for each antenna on the subarray. The first ADA begins in the halfword pointed to by halfword 12 of the RCA, with the rest following in sequence, ordered (smallest DCS address) to (greatest DCS address).

Halfword	Description
0	Antenna I.D., DCS address (right byte of each).
1	U, current time, nsec.
2	V, current time, nsec.
3	W, current time, nsec.
4	I.F. status (I.F. A in first nibble, etc.).
5 to 8	I.F. nominal sensitivity (I.F. A,B,C,D). [3] - in binary scaled B+0. [7] - (HP).

9 to 12	I.F. peculiar phase (I.F. A,B,C,D).
13 to 20	Polarization information (I.F. A,B,C,D).
21	Antenna control bits (logical binary).
22 to 23	[3] - Nonexistent. [5] - U, precessed back to epoch, nsec. (FP).
24 to 25	[3] - Nonexistent. [5] - V, precessed back to epoch, nsec. (FP).

The Bad Correlator Data Area (BCDA) would appear next, except that it was never implemented. Hence, halfword 14 of the RCA has always been equal to halfword 16. If ever implemented, each entry will consist of two halfwords; the first is the correlator status byte and a bit saying which correlator set the error refers to; the second is the correlator number (pointer to the correlator data group).

Each correlator data area is organized in sequence by baseline pairs. The following loop will decode the data associated with a given antenna or baseline:

```
N = 0;
Do for I = 1 to (no. of antennas)-1 by 1;

     Do for J = I+1 to (no. of antennas) by 1;

          N = N + 1;
          Comment correlator data for baseline I X J is
                  located here;
          Comment the following formulae can be used to find
                  the ADA's for antennas I and J:
                  (ADA pointer) + (I-1)*(ADA length) for I,
                  (ADA pointer) + (J-1)*(ADA length) for J;

     end;
end;
```

data for each baseline will look like:

correlator for I.F. A of antenna I times I.F. A of antenna J.
correlator for I.F. C of antenna I times I.F. C of antenna J.
correlator for I.F. A of antenna I times I.F. C of antenna J.
correlator for I.F. C of antenna I times I.F. A of antenna J.

Each correlator mentioned above will consist of three halfword numbers:

real part.
imaginary part.
modified variance.

NOTE: In revision [8], an A.P error was made, making these three halfwords

imaginary part.
real part.
modified variance.

This error was corrected in revision [9].

Therefore, if you know the value of N above, the data for that baseline will begin at

(correlator data area pointer) + (N-1)*12.

The modified variance is the computed variance modified to take account of the source amplitude scattered into the variance by the atmospheric phase fluctuations. This process of modification has not yet been defined, but the intent is to make a single comparison detect high variances due to interference or malfunction.

If the specified mode is a spectral line mode, the bandwidth code is examined to determine the number (call it M) of complex correlators that will be accumulated per baseline. The data for each baseline (antenna I times antenna J) will look like:

I (these two words are pointers to be used in the pipeline system).
J " " " "

```
     correlator channel 1, real part.
     correlator channel 1, imaginary part.
     correlator channel 2, real part.
     correlator channel 2, imaginary part.
     :
     correlator channel M, real part.
     correlator channel M, imaginary part.
```

Again, knowing the value of N above, the data will begin at location

```
     (correlator data area pointer) + (N-1)*(M+1)*2.
```

FORMAT 2 DATA. This is a copy of the source files.

Format 2 was invented to put elements from a source library onto tape. Specifically, we use it to dump certain elements of the observing library onto the visibility data tape for later reference.

The following files are dumped, if present in the library; else they are ignored:

1) file ARRAY.
2) file ANTENNAS.
3) [0] empty.
   [1] file BASELINE.
4) [0] empty.
   [1] file POINTING.
5) files SUB1, SUB2, SUB3, SUB4, and SUB5.
6) files containing observation request cards mentioned on the first card of SUBx.
7) files mentioned in columns 61 or 71 of the default LO cards in SUBx.
8) files mentioned in columns 61 or 71 of the frequency set (//LO) cards in the files of item 6).

The Format 2 data are written in DMF (see Appendix B). The tape record length and spanning information are the same as Format 1. The logical record looks like:

Halfword	Description
0	Logical record length, halfwords.
1	Unused.
2	Format type (description below is format 2).
3	Format revision level.
	[0] - 0 (implemented 76/06/28).
	[1] - 1 (" 82/03/24).
4,5	Date--MJAD.
6,7	Time--Count of 19.2 Hz. interrupts.
8	Reserved (=0).
9	Unused.
10 to 13	ASCII file name of subrecord 1.
14	Subrecord sequence count, this file.
15 to 18	ASCII file name of subrecord 2.
19	Subrecord sequence count, this file.
20 to 44	(Same as above for subrecords 3 through 7.)
45 to 63	Unused.
64 to 191	Subrecord 1.
192 to 319	Subrecord 2.
321 to 959	(Same as above for subrecords 3 through 7.)

Subrecords 1-7 are 128-word compressed ASCII buffers containing the contents of the files from the observing library.

APPENDIX A

MODCOMP Numeric Data Types

The MODCOMP and the DEC-10, like all computers, have their own individual ways of representing and storing numbers. Also, the VLA has created some new formats and conventions. Trying to link all this together has gone far to ensure that the number one computer language used on site is profanity. However, we will make the best of it and attempt to explain the differences.

The MODCOMP is at base a 16-bit machine; the DEC-10 is a 36-bit machine. Two MODCOMP words are packed into each DEC-10 word, one in each DEC-10 halfword. This introduces our first problem: a DEC-10 halfword is 18 bits long. The leftmost 16 bits are therefore a MODCOMP word, and the right two bits of the 18 are zeroed. Note in the preceding report, the word "halfword" is used to designate a MODCOMP 16-bit word or the 18-bit DEC-10 halfword it is packed into. We will continue that convention in the appendices. Note that even MODCOMP double- or triple-words are packed 16 bits at a time, with two trailing zeroes per halfword.

The numeric types used on the MODCOMP's are explained below. Bit designations are from left to right, bit 0 to bit 15 in a 16-bit word. All numbers have the sign in the leftmost bit (bit 0); negative numbers are two's complement.

INTEGERS are either 16-bit (single-precision, the MODCOMP standard), or 32-bit (double-precision). Each assumes a binary point after the rightmost mantissa bit (the assumed scaling is B+15 or S+31; see scaled binary integers below).

SCALED BINARY INTEGERS are arithmetically equivalent to integers, but the binary point is not assumed to be at the far right. The convention is to call the bit position just to the right of the sign bit (between bit 0 and bit 1) bit position 0 and count positively to the right. "B" represents a 16-bit (single-precision) number; "S" a 32-bit (double-precision) number. For example, given the 16-bit binary number

```
0100 0000 0000 0001
```

the following scalings would yield the following values:

```
B+0           0.100 0000 0000 0001
B+10          0100 0000 000.0 0001
B-3     00.00 0100 0000 0000 0001
B+20          0100 0000 0000 0001 0000 0.000
```

FLOATING POINT numbers are of three basic types. The first two are standard MODCOMP numbers. SINGLE-PRECISION FLOATING POINT (FP) numbers have bit 0 as the sign, bits 1-9 as the exponent, and bits 10-31 as the mantissa. Note each floating point number uses two MODCOMP words. The binary point is assumed before bit 10. Therefore

```
0eee eeee eemm mmmm mmmm mmmm mmmm mmmm
```

decodes into

```
0.mm mmmm mmmm mmmm mmmm mmmm * 2**(eee eeee ee - 100 0000 00).
```

Floating point numbers are usually normalized, i.e., the first mantissa bit will be the complement of the sign bit. DOUBLE-PRECISION FLOATING POINT (DP) numbers act in the same way, but contain 16 more mantissa bits. Therefore, DP numbers take three MODCOMP words with bits 10-47 as the mantissa. For convenience, all DP numbers are actually packed into FOUR halfwords, the rightmost halfword being zeros.

HALF-PRECISION FLOATING POINT (HP) is our own invention. It is a 16-bit floating point consisting of sign, five exponent bits, and ten mantissa bits. As these numbers are ALWAYS normalized, the first bit of the true mantissa is hidden, i.e., it is equal to the complement of the sign bit and was dropped in the formation of the HP number. Therefore

```
    0eee eemm mmmmm mmmmm
```

decodes into

```
    0.1mm mmmmm mmmmm * 2**(eee ee - 100 00).
```

Actually, there are occasions where the exponent displacement (100 00 in the above formula) will be different; these will be so noted.

APPENDIX B

DEC-10 Magnetic Tape Format

The DEC-10 is a 36-bit machine; the MOCOMP's are 16-bit. Two MODCOMP 16-bit words will be packed into each DEC-10 36-bit word; each MODCOMP word is left-adjusted and is placed in a DEC-10 halfword with the unused bits set to zero. However, 36 bits do not fit easily onto 9-track tapes. Therefore, the tapes are written in DEC-Magtape Format (DMF), viz.:

Track Byte	1	2	3	4	5	6	7	8	9
1	F0	F1	F2	F3	F4	F5	F6	F7	P
2	F8	F9	F10	F11	F12	F13	F14	F15	P
3	F16*	F17*	R0	R1	R2	R3	R4	R5	P
4	R6	R7	R8	R9	R10	R11	R12	R13	P
5	0	0	0	0	R14	R15	R16*	R17*	P

* usually 0
Fx is the x'th bit of the left halfword; Rx is the x'th bit of the right halfword. P is parity.

Since it is easier for the MODCOMP to handle words instead of bytes, the record length will be rounded up to the nearest multiple of four halfwords. Therefore, four MODCOMP words will be converted to five MODCOMP words in DMF and written onto tape. The tape is now directly readable by the DEC-10.

A program exists on the MODCOMP, DMCOPY, to convert a tape written in DMF back to binary; a subroutine, DMCSUB, converts a buffer passed to it.

APPENDIX C

Table of Bandwidth Codes

In case you are interested, this is a table of bandwidth codes, channels (correlators per baseline), etc.

				Maximum no. allowed		Video bandwidth		
BW	BW	# of	Channel	-------------------		---------------		
code	analyzed	chan.	width	baselines	antennas	new	units	old
0	50 MHz	8	6 MHz	640	27	50	MHz	50
1	25 "	16	1.5 "	320	25	25	"	12
2	12 "	32	400 kHz	160	18	12		1.5
3	6 "	64	100 "	80	13	6	"	1.5
4	3 "	128	25 "	40	9	3	"	-
5	1.5 "	256	6 "	20	6	1.5	"	-
6	0.8 "	256	3 "	20	6	0.8	"	-
7	0.4 "	256	1.5 "	- - Not yet to be implemented - -				
8	0.2 "	256	0.8 "	"	"		"	
9	0.1 "	256	0.4 "	"	"		"	

The maximums given in the table are software maxima (yes, I know). The computers are only capable of handling a total of 5K correlators; therefore (no. baselines)*(no. channels/baseline) must be less than 5120. For the time being, all observing will be done with one subarray; if more were specified, the summation of all baselines would be limited.