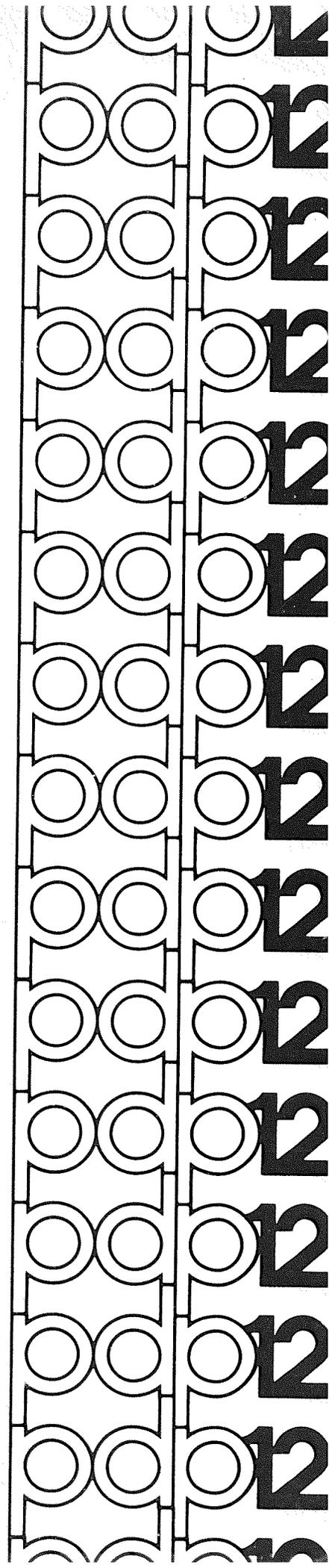


digital

SIGNAL AVERAGER USER'S GUIDE





DEC-12-UZ1A-D
1st Printing May 1970

Copyright (c) 1970 by Digital Equipment Corporation

The material in this handbook, including but not limited to instruction times and operating speeds, is for information purposes and is subject to change without notice.

The following are trademarks of Digital Equipment Corporation, Maynard Massachusetts:

DEC
FLIP CHIP
DIGITAL

PDP
FOCAL
COMPUTER LAB

For additional copies, order DEC-12-UZ1A-D from
Digital Equipment Corporation, Program Library, Maynard,
Massachusetts 01754 Price \$1.00

TABLE OF CONTENTS

	<u>Page</u>
Foreword	i
1.0 Abstract	1
2.0 Minimum Requirements	1
3.0 Loading Procedure	1
4.0 Using the Signal Averager	2
5.0 LINctape Option	6
Summary of Teletype Commands	8

APPENDIX A

Generation and Modification of Control Parameters for the Signal Averager

A.1 Introduction	A-1
A.2 Modifying the Symbols	A-3
A.3 Modifying the Job List	A-3
A.4 Modifying the Channel Display List	A-6
A.5 Other Possible Changes	A-8
A.6 An Example	A-10
A.7 How to Access Data Core Map	A-15
A.8 Data Block Structure	A-15
A.9 User Flowchart	A-17

FOREWORD

The PDP-12 Signal Averager is derived from the LAB-8 Basic Averager program (DEC-LB-U21B). The primary difference between the two is the manner in which each uses its hardware; the LAB-8 Basic Averager utilizes AXØ8 hardware and the PDP-12 Signal Averager runs via the KW12A clock and ADC. To the user, program operation and performance is almost identical.

1.0 ABSTRACT

The PDP-12 Signal Averager Program digitizes, displays, and averages analog signals at rates from 55 to 4095 microseconds/point/channel. The program is entirely core resident. The user can select and adjust, on-line, the sampling rate, the number of sweeps, and the delay via the Teletype.^(R) By modifying the Signal Averager source by the method described in Appendix A, the user can build the program to his particular needs.

2.0 MINIMUM REQUIREMENTS

PDP-12A computer with 4K of core memory
KW12A Clock
1 LAP6-DIAL¹ system tape with the Signal Averager source and binaries (DEC-12-YZAA-UO).

3.0 LOADING PROCEDURE

All binaries are loaded using the DIAL Loader by the command:

→LO NAME,UNIT ↵

Select the appropriate NAME for any one of the three versions of the program from the following chart.

<u>NAME</u>	<u># CHANNELS</u>	<u>DATA POINTS</u>
SIGAVG1	1	1000
SIGAVG2	2	500
SIGAVG4	4	250

^(R) Teletype is a registered trademark of Teletype Corporation.

¹LAP6-DIAL is hereafter referred to as DIAL

The UNIT is the number of the tape unit with the Signal Averager binaries. If the Signal Averager program has been properly loaded, it will respond on the Teletype with a carriage RETURN, LINE FEED and R:

4.0 USING THE SIGNAL AVERAGER

4.1 The leads from the experiment to the PDP-12A must be connected properly to assure correct functioning. The sync input should always be connected to channel one of the KW12A. If only one signal is being averaged, only analog input channel 10 is used. If more than one input is being sampled, the analog inputs should be connected sequentially, beginning with analog input channel 10. For example, if four inputs are being averaged, they should be connected according to the following scheme:

<u>ANALOG INPUT CHANNEL</u>	<u>INPUT #</u>
10	1
11	2
12	3
13	4

4.2 The user can create an individualized version of the program by modifying the Signal Averager source with the DIAL Editor. After the changes have been completed, the source is assembled and saved by DIAL. This procedure is detailed in Appendix A.

4.3 When the Signal Averager is initialized (either by typing CTRL/A, refer to Section 4.7, or by the DIAL Loader, refer to Section 3.0), it will respond by printing R: on the Teletype.

The sampling rate is being requested via the Teletype after R:. Type a value in the range 55 to 4095 microseconds/point and press the RETURN key.

4.4 The Signal Averager will always sample at rates that are multiples of 2.5 microseconds. The program sets the sampling rate to the smallest multiple of 2.5 that is greater than or equal to the requested rate. Consider the following table

<u>REQUESTED RATE</u>	<u>ACTUAL RATE</u>
60	60
61	62.5
62	62.5
63	65
64	65
65	65

The sampling rate on any channel is the clock rate times the number of channels in use.

The initial averaging operation and the occurrence of a sync are not simultaneous events; the elapsed time between occurrence of a sync and initialization of the first A to D conversion varies between 52 and 160 microseconds.

4.5 The Averager will print N:. Respond by typing the number of sweeps desired in the average, followed by a carriage RETURN. A value of zero is treated as 4096.

CAUTION: Because of space limitations, the Signal Averager program does not recognize the RUBOUT key nor does it check for illegal values on input. Be certain that arguments typed in are valid and accurate. If an error has been made in typing, type CTRL/Q which will cause all the questions to be repeated.

4.6 The program will now type D:. Answer with the delay in terms of the clock rate. Thus, if the clock rate is 60 microseconds and it is desired to delay 600 microseconds after the sync pulse before sampling begins, type 10 and press RETURN. Again, if a typing error is made, type CTRL/Q and the number of sweeps, rate, and delay can be respecified.

4.7 It is possible to alter these parameters when the program is averaging by typing CTRL/A. The sequence of requests, R, C, D, will be repeated.

4.8. The program will not begin looking at the input signal. A colon and two line feeds will be generated and the input signal will appear on the oscilloscope. If necessary, adjust the KWL2A threshold control to obtain a reliable sync pulse.

4.9 To begin averaging, press RETURN. The summation will then be shown as it accumulates.

4.10 The accumulated sum may be cleared at any time by typing CTRL/Z.

4.11 Type V at any time to switch from viewing the input to viewing the summation of the sweeps or vice versa. Type X to expand the viewing scale. Type C to contract the viewing scale of the summation. The input data cannot be scaled.

4.12. Press LINE FEED to pause averaging at any point and then press RETURN to resume. Typing LINE FEED always forces the display to the view inputs mode. The sampling will continue until the number of sweeps requested in the parameter setup have been taken. The average may be prematurely terminated by striking CTRL/Q.

4.13 If the average has been completed, it may be plotted by typing P to enter plot mode. Turn on the X-Y recorder, and press the RETURN or LINE FEED key to move the pen to the beginning of the curve. (The beginning of the curve is the X,Y origin defined in display word 2 of the job list for a given average. Refer to Appendix A.) Type RETURN or LINE FEED to commence plotting. To replot the data or to plot the next of a series of curves, press RETURN or LINE FEED to move the pen to the beginning of the curve. Press RETURN or LINE FEED again to start plotting. Terminate plot mode by typing P and turning off the recorder. Pressing RETURN will reactivate the display.

4.14 A completed average may be typed on the Teletype by typing the character T. T: will be printed in response. If all of the accumulated averages are to be printed, type Ø or a carriage RETURN. If it is desired to type selectively one entire channel or selective portions of a channel, type 1, 2, 3, or 4 plus RETURN to type out the first, second, third or fourth channel in the display. L: will then be printed. Respond with the low limit data point number and press RETURN. H: will then be printed. Answer by typing the high limit data point to be typed out and then pressing RETURN.

4.15 The first two numbers typed out represent the number of sweeps included in the average and the scale factor, respectively. The scale factor is the power of two by which the data has been scaled. If the number of sweeps is a power of two, then by typing C and X while the display is active, one can adjust the gain until the scale factor is appropriate for the number of sweeps because two raised to the scale factor power is equal to the number of sweeps. The typeout of data points will then be in millivolts as seen at the analog inputs.

4.16 If it is desired to terminate typeout at any time, type CTRL/Q.

4.17 If another set of sweeps is to be added to the average already accumulated after a group of sweeps has been completed, type CTRL/R to retain the present average and to reinitialize for more averaging.

4.18 To begin averaging after reinitialization has taken place, type RETURN.

5.0 LINCTape Option

The PDP-12 Signal Averager will allow the user to store the results of an average on LINCTape by typing W after an average has been taken. Note that when the program is accepting parameter input (sampling rates, etc.), typing results, or plotting, the WRITE command is inactive. After typing W, SIGAVG will respond with carriage RETURN, LINE FEED and W: indicating it is now waiting for the user to

specify a TBLK on unit 1. A tape must be on transport unit 1 which is WRITE ENABLED. Type in the TBLK and press RETURN. Characters are echoed as they are typed. If an error is made in typing, type CTRL/Q to have the question repeated.

When the Signal Averager receives the carriage RETURN, it writes out the averaged data only in signed two's complement double precision format in sequential TBLKs, starting with the requested TBLK. The data on tape runs sequentially from the first data word of the first data block to the last data word of the last data block. Unused words in the last TBLK contain all zeros. When the program has finished writing, it generates a carriage RETURN-LINE FEED and prints the last TBLK onto which it wrote. Once SIGAVG completes the W command, it reinitializes and prints R:. A new average may now be computed.

The LINCtape option will support SIGAVG1, SIGAVG2, and SIGAVG4, but will not support arbitrary user configurations. For user versions of SIGAVG, the symbol ADCALØ must have a value greater than 1ØØØ and the symbol ADBUF must have a value less than or equal to 4ØØ. (Refer to Appendix A).

SUMMARY OF TELETYPE COMMANDS

CTRL/A	Initialize averaging parameters
CTRL/D	Restart DIAL
CTRL/Q	Quit current operation.
CTRL/R	Rerun the last average.
CTRL/Z	Zero out all previous results.
Carriage RETURN	Argument terminator Commence averaging Position plotter pen Commence plotting
LINE FEED	Pause Averaging and start view input mode (same as carriage RETURN in plot mode).
C	Contract averaged data by a power of two.
P	Enter Plot mode Terminate Plot mode
T	Type out average
V	Switch from view input to view average or vice versa
X	Expand averaged data by a power of two
W	Write on LINCTape

APPENDIX A
GENERATION AND MODIFICATION OF CONTROL
PARAMETERS FOR THE SIGNAL AVERAGER

A.1 INTRODUCTION

The Signal Averager program is very general in nature in that it can be adapted to a variety of situations. The control parameters accomplish this specific adaptation.

There are three different binaries available on the LINtape DEC-12- UZ1A-UO supplied by the Digital Equipment Corporation Program Library. Each allows the maximum number of points to be taken as the number of channels are varied. In some cases, the user may want to decrease the number of points taken to realize a shorter period of stimulus repetition or to allow core for his own programs. He may want to sample some number of channels not covered by the existing control parameters. If the machine contains more than 4K of memory, the user may wish to fully utilize the core available to him. Each of these situations is easily implemented with an understanding of what control parameters are and how they can be generated or modified. Before modification is attempted, a knowledge of the DIAL Editor and Assembler is required and may be gained from the LAP6-DIAL Manual, DEC-12-SE2B-D.

Note that two source programs are also included on the Signal Averager LINtape: CNTRL2 is for a two-channel averager and CNTRL4 is for a four-channel averager. These two sources demonstrate the modifications made to the Signal Averager to obtain a two or four channel average.

Briefly, the procedure involves using the Editor program to make the necessary changes in the source which is then assembled and the binary saved on a DIAL tape. The detailed procedure follows and assumes that the user is familiar with PDP-8 programming techniques.

To edit the PDP-12 Signal Averager, bring the source into the DIAL Working Area by executing the command

```
→AP SIGAVG,UNIT)
```

The relevant portion of the source starts with the comment USER MODIFICATION STARTS HERE. It begins at approximately line 2657. Refer to the listing at the end of this manual.

A.2 Modifying the Symbols

The symbol NCHAN defines the number of analog inputs and the symbol POINTS defines the number of data points per channel (analog input). The maximum number of channels is five; the maximum number of points is 1000. There is a further restriction that the product of the number of channels and the number of points be less than or equal to 1000 on a 4K PDP-12. For example, if the number of channels (NCHAN) were set equal to 4, then the maximum allowable number for the symbol POINTS would be 250.

The symbol LNBUF defines the total number of points across all channels and it is equal to the product of NCHAN and POINTS. For a three channel signal averager LNBUF would be defined as:

$$\text{LNBUF}=\text{POINTS}+\text{POINTS}+\text{POINTS}$$

Symbols cannot be defined as products, such as LNBUF=POINTS*NCHAN, because the multiplication operation is not recognized by the DIAL Assembler.

A.3 Modifying the Job List

The PDP-12 Signal Averager uses "job list processing". Each average taken has a job list associated with it that defines the parameters of that particular average. Job lists are seven words long. If there is more than one, they follow each other sequentially in memory. Following the last list there must be a word containing zeros. The symbol "JLIST" designates the total length of the string of job lists. This is equal to 7*NCHAN+1 and must be specified as the result of a summation or one decimal number.

The symbol MEMTOT defines the amount of core available. It appears in the source as

```
*MEMTOT
0
```

The number following MEMTOT indicates the additional 4K stacks of memory existent on this machine so that an 8K machine would have

```
*MEMTOT
1
```

The next area that may be edited is the job list itself. Until this point, all variables that have been defined have been common to all the averages in this experiment. Now, each average must have its particular parameters defined. The first list will always start at location LOCORE (0230_8). After the end of the last job list, a zero word is inserted to indicate the end of the lists.

```
Word 1:      Bit 0      :      always = 0
              Bits 1 - 5 :      specify the analog
                               input number per-
                               taining to this average
```

The configuration for bits 1-5 may be determined from this table:

<u>analog input channel</u>	<u>input (logical) channel</u>	<u>configuration of Bits 1-5</u>
10	0	00 000
11	1	00 001
12	2	00 010
13	3	00 011
14	4	00 100

(Any other bit configuration for a channel is illegal)

	Bit 6	:	always = 1
	Bits 7-11	:	indicate the position of this average in the sampling list (usually the same as bits 1-5)
Word 2:	AVG		(not to be altered by the user)
Word 3:	- POINTS		assembles to be the negative of the number of data points in the block of this average
Word 4:	62n1		change data field instruction, CDFN (=62n1) where n is the number of the data field (i.e., the block of 4K; n=0 for 1st 4K, n=1 for 2nd 4K) where the first data point for this average is to be stored.
Word 5:	ADCALN		where N is the number of the average. First average is always zero. This is the address -1 of the beginning of the data block belonging to this average and is defined as: $\text{ADCAL}_0 = \text{ADBUF} + \text{LNBUF}$ $\text{ADCAL}_1 = \text{ADCAL}_0 + \text{BLKLEN} + 1$ $\text{ADCAL}_2 = \text{ADCAL}_1 + \text{BLKLEN} + 1$ In general, for k greater than 0: $\text{ADCAL}_k = \text{ADCAL}_{(k-1)} + \text{BLKLEN} + 1$
Word 6:	Display Word 1		(see below)
Word 7:	Display Word 2		(see below)

A.4 Modifying the Channel Display List

The channel display list contains two words for each average. These are the same as words 6 and 7 of the job list. In assembling, the channel display list comes directly after the end-of-word list in the job list.

Display Word 1

The leftmost 8 bits of display word 1 are the x-increment used between points in the display cycle. The first four of these 8 bits are the integer part of DELTAX, the last four are the fractional part.

Since the display scope has a maximum resolution of 512 points in each direction, it is useful to be able to use a fractional increment. For example, if the average contained 1024 points, an increment of 0.5_{10} (0000.1000_2 , using 8 bits) would just fill the scope with the display. An increment of 0.5_{10} will display two successive data points at the same x location. An increment of 0.25_{10} (0000.0100_2) will display four successive data points at each x position, etc.

A simple formula can be used to determine the DELTAX value to use, given the number of POINTS and PC, the fraction of the screen that you wish to cover with the display:

$$\text{DELTAX} = \frac{512 * \text{PC}}{\text{POINTS}}$$

The last four bits of display word 1 are the Y scale factor (YS) for the particular average. As each data point is displayed, its value is divided by 2^{YS} .

Thus, for full height display, YS would be zero.
For half height it would be one, etc.

Display Word 2

The first six bits of this word are the X origin for the average display; the last six bits are the Y origin. Both of these numbers are multiplied by 10_8 before being used as starting points for the display. This makes it possible to start the display anywhere on the screen. The first bit of the Y origin is interpreted as a sign bit so that it is possible to start below zero. For instance, a Y origin of -200_8 would imply a 60_8 as the low order end of display word 2.

Generating the Display Words

The following is an example of how to set up the display words for a 500 point, 2 channel, signal averager. It is assumed that the averages will be displayed one above the other, and on the full width of the screen.

DELTA: Since there are 500 points in each sweep, the formula above gives $DELTA = 512/500 = 1.024_{10} = 0001.000_2$, to 8 bit accuracy.

YS: Each average should cover half the screen so each data point value must have been divided by 2^1 . Therefore YS should be 1. Display word 1 is now:

$$\begin{array}{r} 00010000 \\ DELTA \end{array} \quad \begin{array}{r} 0001_2 \\ YS \end{array} = 0401_8$$

XZ: Since both traces start at the left hand side of the scope, XZ will be \emptyset for both averages.

To define an address in upper core, the user must type in the pseudo-op FIELD n, where n is the data field. All addresses specified thereafter will be located in that data field (the first 4K is field \emptyset).

When a data block lies in both upper and lower core, the PDP-12 Signal Averager switches between data fields by a process called "linking." The linking is accomplished by a set of three linkage words at the end of the block lying in the lower field. These inform the Signal Averager of the number of data points (not core locations or values) that are needed to complete this average, the data field in which these points lie, and the location -1 of the first word in the next data block. These linkage words must be supplied by the user.

The upper limit of the data storage area in the lower 4K is 6177. The Signal Averager program resides above this, so at some point before this limit, the linkage words must be inserted. (After assembly, the user should check to see that these words were placed in locations lower than 6177.)

It is important to note here that Word 3 of the job list pertaining to this average should reflect only the number of points found in the first part of the data block (i.e., it must not include the point count contained in the part of the block that is in upper core). The CDF instruction and ADCAL word of the job list should refer to the address and field of the first data point contained in the part of the block which is in lower core. The linkage words should contain the information on the points that lie in upper core.

YZ:. The first average must cover the top half of the scope (0-377₈) and its zero point should then be 200₈. This means that the second half of display word 2 will be 20₈ for this average. Similarly, the second average must have the Y origin at -200₈, and the second half of the word will be -20₈ = 60₈. The channel display list will now appear as:

```
          /CHANNEL DISPLAY LIST
          *CHLIST + 1

0401      /THESE WORDS ARE FOR
0020      /FIRST AVERAGE
0401      /THESE ARE
0060      /FOR SECOND
0000      /THIS IS END OF LIST
```

The last symbols to be defined are the end of block symbols. There must be a zero word following each data block for the program to work properly. The coding is ADCALK+BLKLEN+1; k = 0,1,...,n-1 for a 4K n channel averager. (Refer to line numbers 2734 and 2735 of listing in the back of this manual.)

A.5 Other Possible Changes

The easiest change to make is the number of points taken in an average. The only changes necessary are POINTS and the DELTAX value. The Assembler effects the consequent changes.

It is possible to enable the PDP-12 Signal Averager to use more than 4K of core. Word 4 of the job list defines the data field of the first point in the data block. ADCALN (the address -1 of that data block) directs the PDP-12 Signal Averager within that data field. Thus, if a data block is in upper 4K, only the CDF instruction need be changed. (ADCALN may have to be defined differently, refer to page A-11.)

A.6 An Example

To set up a single channel, 2000 point signal averager, the changes required are:

1. POINTS = 2000
2. *MEMTOT
1
3. In the job list Word 3 should be minus the number of points contained in lower core. In this case, 500 points will be stored in lower core (although a value greater than 500 could have been chosen). Remember that the upper bound for the data buffer is 6177.

DECIMAL
-500₁₀
OCTAL

4. Word 6 - DELTAX must be changed to correct the display for the increased number of points. # of points = 2000 for full screen display (PC=1).

$$\text{DELTAX} = \frac{512 * 1}{2000} = 0.256_{10} = 0.203_8$$

In 8 bits accuracy this value is

0000.0100

integer fractional
part part

the rightmost four bits for Y scale are 0
(0000). Word 6 becomes:

DELTAX YS
0000.0100 0000 = 0100₈

5. The first channel display word must also be changed to 0100.
6. The linkage words must be located directly after the last value pertaining to the last point in this block. This locations is determined by starting with ADCAL0. This is

the address -1 of the first word of the data block. The first word of the data block contains the number of sweeps and the following words contain the data. Therefore, the last location is equal to:

$$\text{ADCAL0}+1+ (\# \text{ of locations needed for data storage})+1$$

In this case, 500 points are stored in this block. Each point needs two words for the sum. Therefore, the address of the word following the data is: $\text{ADCAL0} + 1 + 1000_{10} + 1$.

Linkage Word 1:	-1500_{10}	- # points in next block
Linkage Word 2	6211_8	CDF for next field
Linkage Word 3:	99_{10}	location -1 of first word

NOTE: The value of linkage word 3 was chosen arbitrarily.

Define these in the following manner:

```

DECIMAL
*ADCAL0+1+1000+1
-1500
OCTAL
6211
DECIMAL
99

```

If the origin (ADCAL instruction) is greater than 6175, the data buffer will overlap the core area reserved for this program.

Now the end of block word must be inserted. It lies in field 1, 1500 points or 300_{10} words from the start of the buffer. There is no sweep count included in this block so the location of the word following the last data word is:

(starting address -1; specified in linkage word
 $3 = 99_{10}) + 1 + (\text{number of words used for storage; in this data block} = 1500_{10} * 2) =$
 $99 + 1 + 3000 = 3100_{10}.$

In this case the location is 3100_{10} .

Define this by the following:

```
FIELD 1      /CHANGE FIELDS
DECIMAL
*3100
0
OCTAL
FIELD 0      /RESET FIELD FOR PROPER
              /ASSEMBLY OF REMAINING CODE
```

The next block of data (if any) will be entirely in field 1. Its starting address -1 is equal to the end-of-block word defined above, rather than at $\text{ADCAL1} = \text{ADCAL0} + \text{BLKLEN} + 1$ as is the case with 4K.

The easiest way to define the new ADCAL1 is by saying (for the above example)

```
DECIMAL
ADCAL1=99+1+1500+1500
OCTAL
```

Any data blocks after this one may have their ADCAL words defined as in 4K (e.g., $\text{ADCAL2} = \text{ADCAL1} + \text{BLKLEN} + 1$).

To assemble the modified source, use the DIAL Assembler.
 After assembly, the binary must be saved by the following
 command:

→SB NAME,UNIT,P00400)

The listing generated by these changes for a 2000 point,
 single channel averager follows.

```

2646          /USER MODIFICATION STARTS HERE
2647          /EXAMPLE OF A 1 CHANNEL 2000 POINT
2650          /SIGAVGER
2651          DECIMAL
2652          NCHAN=1          /=NUM OF ANALOG
2653          /INPUTS
2654          POINTS=2000     /=NUM OF POINTS
2655          /PER CHAN
2656          BLKLEN=POINTS+POINTS+1
2657          LNBUF=POINTS    /=POINTS*NCHAN
2660          JLIST=7+1      /=7 WORDS PER
2661          /JOB+EOL WORD
2662          CHLEN=NCHAN+NCHAN+1
2663          /2 WDS FOR EACH
2664          /CHAN+EOL
2665          CHLIST=LOCORE+JLIST-1
2666          ADBUF=CHLEN+CHLIST
2667          /BUFFER STARTS
2670          /AFT CH LIST
2671          *MEMTOT        /NUM OF ADD
2672          0020 0001      1          /4K STACKS
2673          0021 5744     HICORE-LOCORE-4
2674          /JOB LIST
2675          OCTAL
2676          *LOCORE
2677          0230 0040      0040       /0(1) CHAN(5)
2700          /1(1) CHORD(5)
2701          0231 0400     AVG        /TYPE(8)
2702          DECIMAL
2703          0232 7014     -500      /-NUM OF DATA
2704          /POINTS IN BLK
2705          OCTAL
2706          0233 6201     6201      /CDF 0
  
```

2707			ADCAL0=ADBUF+LNBUF	
2710				/LOC-1 OF 1ST
2711				/POINT
2712	0234	4162	ADCAL0	/IS RIGHT AFTER
2713				/ADC BUFFER
2714	0235	0100	0100	/DELTAX(8)
2715				/YS(4)
2716	0236	0000	0	/X0(6),Y0(6)
2717	0237	0000	0	/END OF LIST
2720				/CHANNEL
2721				/DISPLAY LIST
2722	0240	0100	0100	/DELTAX(8)
2723				/YS(4)
2724	0241	0000	0	/X0(6),Y0(6)
2725	0242	0000	0	
2726			DECIMAL	
2727			*ADCAL0+1+1000+1	/INSERT
2730				/LINKAGE WORDS
2731				/HERE
2732	6134	5044	-1500	/- NUM OF
2733				/POINTS IN
2734				/NEXT BLOCK
2735			OCTAL	
2736	6135	6211	6211	/CDF 1
2737			DECIMAL	
2740	6136	0143	99	/LOC-1 OF 1ST
2741				/WD IN NEXT
2742				/BLOCK
2743			FIELD 1	/CHANGE FIELDS
-				
2744			DECIMAL	
2745			*3100	
2746	6034	0000	0	/END OF BLOCK
2747			OCTAL	
2750			FIELD 0	
2751				/USER MODIFICATION ENDS HERE

A.7 HOW TO ACCESS DATA CORE MAP

Below is a table of the names of several lists and buffers and the location on page zero of their pointers. For example, the starting location -1 of the job lists is found in the location 23. C(23)+1 is read "the contents of location 23 plus 1."

<u>AREA</u>	<u>SIGNAL AVERAGER</u>
Job List	C(23)+1
CHAN Disp List	C(24)+1
ADC Buffers A sweep	C(27)+1
Calculation (storage) Buffers	the location of these is obtained from the job list

A.8 DATA BLOCK STRUCTURE

To provide for multiple field machines, the PDP-12 Signal Averager saves data in linked. In general, the Signal Averager will end one block and link to another whenever it comes to an end of field.

An example is shown for N sweeps and M data points broken into two data blocks.

Words J3-J5
of Job List
Entry for
This Job

BLOCK 1
(In Data Field U)

BLOCK 2
(In Data Field V)

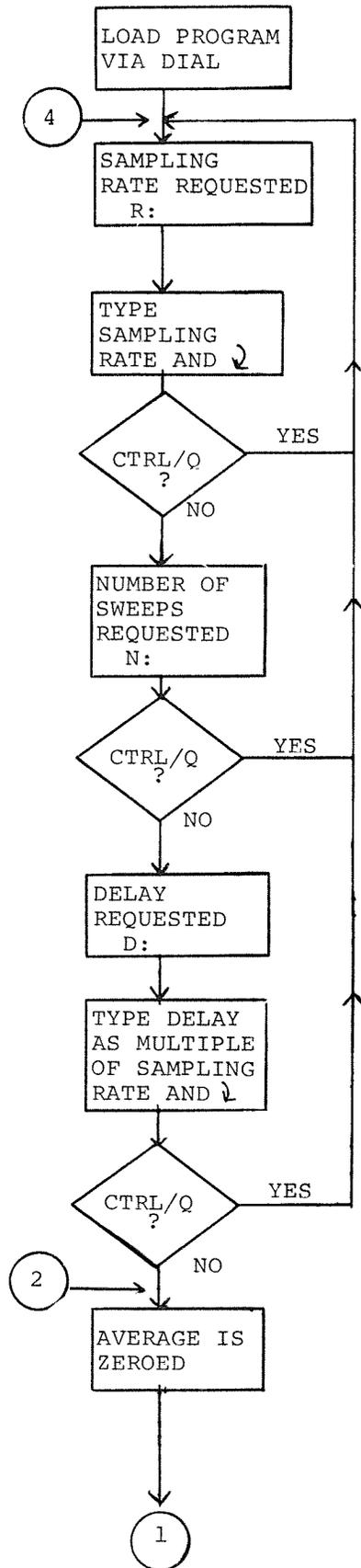
of Pts. in 1st Block:-L
CDF to 1st Block CDF U
Loc-1 of 1st Wrđ BLK1-1

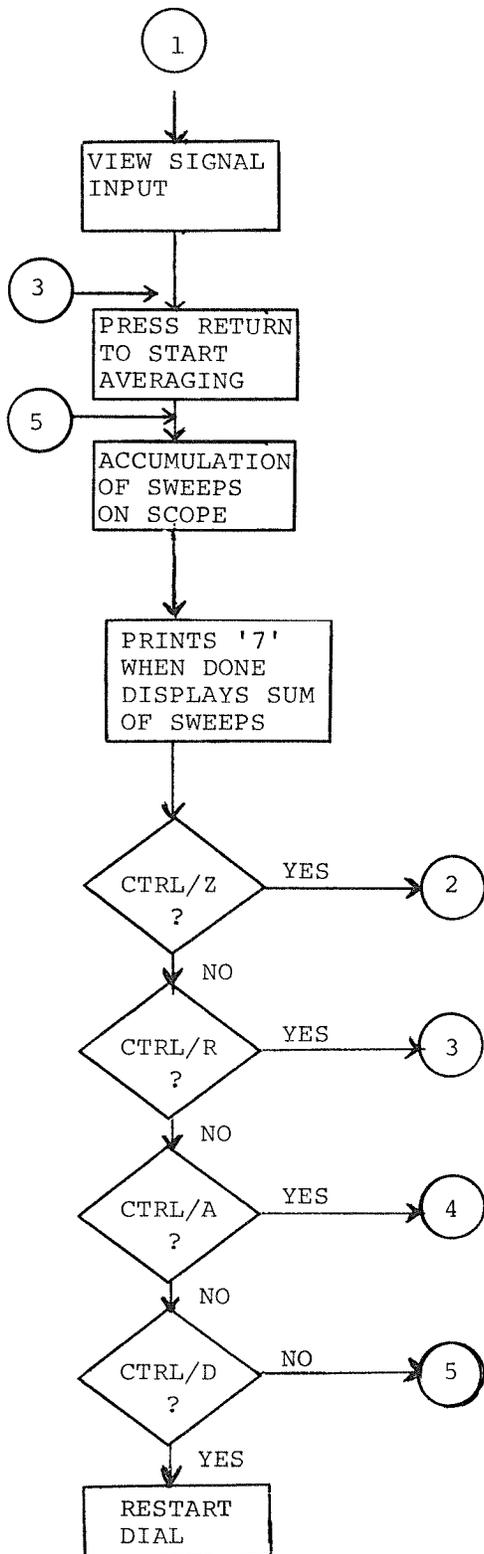
of sweeps: N
1st Data Point
2nd Data Point
3rd Data Point
.
.
.
.
.
.
.
.
.
Lth Data Point
-# of Pts in Nxt Blk:L-M
CDF to next Blk: CDFV
Loc-1 of 1st Wđ of Nxt Blk: BLK 2-1

L+1th Data Point
.
.
.
.
.
.
.
Mth Data Points
-# In Next Blk: Ø

A.9. USER FLOW CHART

Parameters:
 # of channels
 # of points





INDEX

Abstract	1
Analog Input Channels	2
Channels	1
Channel Display List	A-6
Commands	3, 8
Core Map	A-15
Data Block Structure	A-15
Delay	4
Flowchart	A-17
Initialization	2
Job List	A-4
LINCTape Storage Loading Procedure	6 1
Minimum Requirements	1
Modifying the Sig- nal Averager	A-1
Name	1
Number of Sweeps	3
Plot Mode	5
RUBOUT	4
Sampling Rate	3
Scale Factor	6
Sync Input	2
Type Average	5
Using the Signal Averager	2
View Mode	4, 5


```

0000          *20
0001          PMODE
0002          /PDP 12 SIGNAL AVERAGER
0003          /WITH MTP WRITE SR
0004          /9 APR 70
0005          /MEMORY BOUNDS FOR FIELD 0
0006          LOCORE=230      /LISTS, BUFERS
0007                          /DATA BLOCKS
0010                          /START HERE
0011          HICORE=6200    /PROTECTED AREA
0012                          /BEGINS HERE
0013                          /AVGING PARAMS
0014                          /LOCNS 20-64
0015                          /LIST ADDRESSES
0016          ADJLIS=23      /START OF JOB
0017                          /LIST-1
0020          ADCHNL=24      /START OF CHANEL
0021                          /DISPLAY LIST-1
0022          AVG=400        /JOB TYPES
0023          SD=1000
0024          TRN=1400
0025                          /JOINT SWEEP
0026                          /PARAMS 29 LOCS
0027          SMASK=25       /STIMULUS (SYNC)
0030                          /CHANNEL MASK
0031          NSWEP=26      /-NUM SWEEPS
0032                          /IN AVERAGE
0033          KSYTIM=27     /-( OF ASI FROM
0034                          /STIM TO SYNC
0035                          /POINT -1)
0036          ASI=37         /USEC PER ASI
0037          KMODE=42      /KW12 CNTRL REG
0040          CLKMOD=43    /KW12 ENAB REG
0041                          /SWEEP A LIST
0042                          /PARAMETERS
0043          SAMA=44       /-POINTS (ASI)
0044                          /IN SWEEP A
0045                          / (EACH CHANNEL)
0046          NCHA=45      /NUM OF CHAN
0047                          /IN SWEEP A
0050          ADBUFA=47    /LOCN -1 FOR
0051                          /START OF ADC
0052                          /BUFFER -A
0053          PLTDLY=50     /PLOTTER DELAY
0054                          /COUNTER
0055          CLKCNT=51    /CLOCK CNTR
0056                          /CHANEL COUNTER
0057                          /AND CONSTANT
0060          KCHCNT=52    /-NO. OF CHAN
0061                          /IN SWEEP CONST
0062          CHCNT=53     /-NO. OF CHAN
0063                          /LEFT IN SWEEP
0064                          / (VARIABLE)
0065                          /SWEEP A ON LN
0066                          /PARAMETER
0067          KBLA=60       /-NUM ASI FROM
0070                          /SYNC POINT TU
0071                          /LOGICAL END OF
0072                          /A-1
0073          OCSORT=130
0074          TEMP01=146   /TEMPORARY
0075          TEMP02=147   /STORAGE

```

0076			TEMP03=150	/REGISTERS
0077			TEMP04=151	/146-177
0100			TEMP05=152	
0101			TEMP06=153	
0102			TEMP07=154	
0103			TEMP10=155	
0104			TEMP11=156	
0105			TEMP12=157	
0106			TEMP13=160	
0107			TEMP14=161	
0110			TEMP15=162	
0111			TEMP16=163	
0112			TEMP17=164	
0113			TEMP20=165	
0114			TEMP21=166	
0115			ARITH0=167	/TEMP
0116			TEMP22=167	/STORAGE
0117			ARITH1=170	/AND
0120			TEMP23=170	/MULTIPLE
0121			ARITH2=171	/ACCUMULATORS
0122			TEMP24=171	
0123			ARITH3=172	
0124			TEMP25=172	
0125			ARITH4=173	
0126			TEMP26=173	
0127			ARITH5=174	
0130			TEMP27=174	
0131			KBDBUF=175	/TEMP
0132			TEMP30=175	/STORAGE
0133			TTYBUF=176	/AND
0134			TEMP31=176	/TTY-KBD
0135			TTYFLG=177	/BUFFERS
0136			TEMP32=177	
0137				/COMBINED
0140				/OPERATES
0141			MTH=CLA CMA CLL RTL	
0142			MTW=CLA CMA CLL RAL	
0143			TWO=CLA CLL CML RTL	
0144				/INSTALLATION
0145				/PARAMETERS
0146				/LOCNS 20-64
0147				/LIST ADDR
0150				/INT SERVICE
0151	0001	3066	*1	
0152	0002	7010	DCA ASAVE	
0153	0003	3067	RAR	
0154	0004	5465	DCA LSAVE	
0155	0005	0000	JMP I INTERX	
0156	0006	0000	Ø	
0157	0007	0000	Ø	
0160	0010	0000	LXCOR, Ø	
0161	0011	0000	BUFA, Ø	
0162	0012	0000	GETPNT, Ø	
0163	0013	0000	PUTPNT, Ø	
0164			JPNT, Ø	
0165			AXPNTR=JPNT	
0166			TMPNTR=17	
0167			*3Ø	
0170				/LINKS N CONSTS
0171	0030	7777	WSWITZ, 7777	/FOR MTP SR
0172	0031	7722	WDUNZ, WRIDUN	
0173	0032	7624	WRIZ, WRITE	
0174	0033	7662	INDATZ, INDATA	

0175	0034	7331	TMESSZ,	TMESS	
0176	0035	7342	TYPEZ,	TYPES	
0177	0036	0007	KZ7,	7	
0200				*56	
0201	0056	6605	ZCYCLE,	CYCLE	
0202				*104	
0203	0104	7600	KM200Z,	-200	
0204	0105	6275	STWZ,	ADDUN	
0205	0106	7431	ZLADR,	SQADS-2	
0206	0107	0377	KZ377,	377	
0207	0110	0260	KZ260,	260	
0210	0111	7775	M3Z,	-3	
0211			LMODE		
0212			*41		/LINC INTERRUPT
0213	0041	6200		JMP 200	/SERV AT 200
0214					
0215			PMODE		
0216			*50		
0217	0050	0000	PLTDLY,	0	
0220	0051	0000	CLKCNT,	0	
0221	0052	0000	KCHCNT,	0	
0222	0053	0000	CHCNT,	0	
0223	0054	7773	KM005,	-0005	/CONSTANTS
0224	0055	0100	K0100,	0100	
0225			*60		
0226	0060	0000	KBLA,	0	
0227	0061	0215	K0215,	0215	
0230	0062	0322	K0322,	0322	
0231	0063	0062	K0062,	0062	
0232	0064	4060	KM3720,	-3720	
0233	0065	0000	INTERX,	0	
0234	0066	0000	ASAVE,	0	
0235	0067	0000	LSAVE,	0	
0236					/LINKS TO SECN
0237					/3 SUBROUTINES
0240	0070	7114	JGET,	JGETS	/GET NEXT JOB
0241					/FROM JOB LIST
0242	0071	7147	BLKCNT,	BLKCNS	/MOVE THRU DATA
0243					/BLOCK
0244	0072	7200	SETPNT,	SETPNS	/SET ADC PNTRS
0245	0073	7170	IXPNT,	IXPNIS	/MOVE THRU
0246					/ADC BUFFERS
0247	0074	7000	SDIS,	SDISS	/SET UP DISPLAY
0250	0075	7066	DISP,	DISPS	/DISPLAY A PNT
0251	0076	0000	LDIS,	0	
0252	0077	6141		LINC	/LINC MODE DIS
0253				LMODE	/ROUT
0254	0100	0147		DIS LXCORD	/B7 IS H.C.
0255	0101	0002		POP	
0256				PMODE	
0257	0102	7200		CLA	
0260	0103	5476		JMP I LDIS	
0261				*112	
0262	0112	0004	K0004,	0004	/PAGE 0 CONSTS
0263	0113	0003	K0003,	0003	/USED BY
0264	0114	0002	K0002,	0002	/RESIDENT
0265	0115	7777	KM0001,	-001	/SUB RTS
0266	0116	0007	K0007,	0007	/DONT RELOCATE
0267	0117	7751	KM0027,	-0027	
0270	0120	0377	K0377,	0377	
0271	0121	7774	KM0004,	-0004	
0272					/TTY-LIST
0273	0122	0000	TTYLIST,	0	/-END OF LIST

0274	0123	0042	PROMRK,	42	/"-PROG OUTPUT
0275					/MARKER
0276	0124	0044	K44,	44	/-DISPLAY RESET
0277	0125	0047	TXMRK,	47	/-KEYBRD INPUT
0300					/MARKER
0301	0126	0045	KCR,	45	/CR-CARIG RTN
0302	0127	7735	KM0043,	-43	/LF-LINE FEED
0303					*OCSORT
0304	0130	0040	K0040,	40	/SPACE: K0040
0305	0131	7731	MTXMRK,	-47	/: MTXMRK
0306					/LINKAGES TO
0307					/BASIC SUB RT
0310	0132	7536	BRAN,	BRANS	/TABLE BRANCH
0311	0133	7272	SHFT,	SHFTS	/DBL PRECISION
0312					/ARITH SHIFT
0313	0134	7557	DADD,	DADDS	/DBL PREC ADD
0314					/TEMP STORAGE
0315	0135	0000	KLUG,	0	
0316	0136	1011		TAD GETPNT	
0317	0137	3012		DCA PUTPNT	
0320	0140	5535		JMP I KLUG	
0321					/REGS 146-177
0322				*200	/LINC INTERRUPT
0323				LMODE	/SERVICE ROUT
0324	0200	0060		SET I 0	/RTN TO LIRETN
0325	0201	0207		LIRETN	/WHEN FINISHED
0326	0202	4221		STC LASAVE	
0327	0203	0321		ROR I 1	
0330	0204	4227		STC LLSAVE	
0331	0205	0002		PDP	
0332				PMODE	
0333	0206	5001		JMP 1	/GO SERVICE
0334					/INTERRUPT
0335	0207	6002	LIRETN,	IOF	
0336	0210	7200		CLA	
0337	0211	1040		TAD 40	/GET RTN ADDR
0340	0212	6141		LINC	
0341				LMODE	
0342	0213	1620		BSE I	/MAKE A JUMP
0343	0214	6000		6000	
0344	0215	4226		STC LJUMP	
0345	0216	2227		ADD LLSAVE	
0346	0217	0261		ROL I 1	
0347	0220	1020		LDA I	
0350	0221	0000	LASAVE,	0	
0351	0222	0500		IOB	
0352	0223	6244		6244	/RMF
0353	0224	0500		IOB	
0354	0225	6001		6001	/ION
0355	0226	0000	LJUMP,	0	/JUMP BACK TO
0356					/MAIN PROGRAM
0357	0227	0000	LLSAVE,	0	
0360				PMODE	
0361				*400	/LOAD AND
0362	0400	7300		CLA CLL	/GO START
0363	0401	5602		JMP I .+1	
0364	0402	6511		START	
0365				*6200	
0366					/ON-LINE FAST
0367					/AVERAGE
0370					/RTS TO TYPE
0371					/DECIMAL WITH
0372					/FORMATTING

0373	6200	0000	DECTYP, 0		/TYPE SIGNED
0374					/DECIMAL IN
0375					/ARITH2
0376	6201	1171		TAD ARITH2	/GET SIGN
0377	6202	7710		SPA CLA	
0400	6203	1262		TAD K0015	/TYPE - FOR NEG
0401	6204	1264		TAD K0240	/SPACE FOR PLUS
0402	6205	4674		JMS I TYPEX	
0403	6206	1171		TAD ARITH2	
0404	6207	7510		SPA	
0405	6210	7041		CMA IAC	
0406	6211	3167		DCA ARITH0	/ABS OF NUM
0407	6212	1257		TAD KMD1K	/GET THOUSANDS
0410	6213	4230		JMS GDIGIT	/DIGIT
0411	6214	1260		TAD KMD100	/GET HUNDREDS
0412	6215	4230		JMS GDIGIT	/DIGIT
0413	6216	1261		TAD KMD010	/GET TENS DIGIT
0414	6217	4230		JMS GDIGIT	
0415	6220	7240		CLA CMA	/GET UNIT DIGIT
0416	6221	4230		JMS GDIGIT	
0417	6222	1264		TAD K0240	/TYPE SPACE
0420	6223	4674		JMS I TYPEX	
0421	6224	2273		ISZ CRCNT	/CHECK ITEM
0422					/COUNT FOR LINE
0423	6225	5600		JMP I DECTYP	/NOT 10 YET XIT
0424	6226	4247		JMS CRLFS	/10 ITEMS NEW
0425					/LINE
0426	6227	5600		JMP I DECTYP	
0427	6230	0000	GDIGIT, 0		/RADIX DEFLATE
0430					/AND TYPE DIGIT
0431	6231	3172		DCA ARITH3	/RADIX TO
0432					/DEFLATE BY
0433	6232	3146		DCA NDIGIT	
0434				NDIGIT=TEMP01	
0435	6233	1167		TAD ARITH0	/NUM TO DEFLAT
0436	6234	3167	GLOOP,	DCA ARITH0	/UPDATE DEFLATED
0437					/NUMBER
0440	6235	1167		TAD ARITH0	/TRIAL
0441	6236	1172		TAD ARITH3	/SUBTRACTION
0442	6237	2146		ISZ NDIGIT	
0443	6240	7500		SMA	/DO MORE?
0444	6241	5234		JMP GLOOP	/YES, CONTINUE
0445					/DEFLATION
0446	6242	7200		CLA	/NO
0447	6243	1263		TAD K0257	/NDIGIT IS
0450					/DIGIT+1
0451	6244	1146		TAD NDIGIT	/TYPE DIGIT
0452	6245	4674		JMS I TYPEX	
0453	6246	5630		JMP I GDIGIT	
0454	6247	0000	CRLFS, 0		
0455	6250	1061		TAD K0215	/TYPE CR
0456	6251	4674		JMS I TYPEX	
0457	6252	1266		TAD K212	/TYPE LF
0460	6253	4674		JMS I TYPEX	
0461	6254	1261		TAD KMD010	/SET COUNT
0462	6255	3273		DCA CRCNT	/-10 ITEMS
0463	6256	5647		JMP I CRLFS	
0464					/LOCAL CONSTS
0465	6257	6030	KMD1K, -1750		
0466	6260	7634	KMD100, -0144		
0467	6261	7766	KMD010, -0012		
0470	6262	0015	K0015, 0015		
0471	6263	0257	K0257, 0257		

0472	6264	0240	K0240,	0240	
0473					/TEXT OF CTRL/A
0474					/MESSAGES
0475	6265	0215	TXMESS,	215	/(CR)
0476	6266	0212	K212,	212	/(LF)
0477	6267	0000	TMCH,	0	/*
0500	6270	0272		272	/:
0501	6271	0240		240	/SPACE
0502	6272	0000		0	/END TEXT
0503	6273	7766	CRCNT,	-12	/LOCAL VAR
0504	6274	7342	TYPEX,	TYPES	/LOCAL
0505	6275	1162	ADDUN,	TAD ASWIT	/AVGING SWIT=0
0506					/IF OK TO AVG
0507	6276	7640		SZA CLA	
0510	6277	5377		JMP ADONE	/.NE. 0; DONT
0511					/UPDATE CALCS
0512	6300	1023		TAD ADJLIS	/START CALCS AT
0513					/TOP OF LIST
0514	6301	3013		DCA JPNT	
0515	6302	6201	CALJOB,	CDF 0	/FIELD 0 HOLDS
0516					/JOB LIST
0517	6303	4470		JMS I JGET	/GET JOB PARAMS
0520	6304	5365		JMP CALEND	/EOL DETECTED
0521					/END OF THIS
0522					/SWEEP
0523	6305	2013		ISZ JPNT	/MOVE OVER
0524	6306	2013		ISZ JPNT	/J6 AND J7
0525	6307	3324		DCA CALXM	/AC HELD CDF N
0526	6310	4472		JMS I SETPNT	/SET UP ADC
0527					/PNTRS N CNTRS
0530	6311	1011		TAD GETPNT	/SET UP TO
0531	6312	3012		DCA PUTPNT	/UPDATE JOBS
0532					/DATA REGION
0533	6313	1324		TAD CALXM	/UPDATE JOBS
0534	6314	3315		DCA .+1	/SWEEP CNTR
0535	6315	6201		CDF	/SET CDF
0536	6316	1411		TAD I GETPNT	/GET NUM OF
0537	6317	3154		DCA NSAVE	/SWEEPS
0540				NSAVE=TEMP07	
0541	6320	2412		ISZ I PUTPNT	/SAVE INCR,
0542					/NUM OF SWEEPS
0543	6321	6201	CALNXT,	CDF 0	/ADC BUFFER IS
0544					/IN FIELD 0
0545	6322	1553		TAD I TBUFAD	/GET SAMPLE
0546	6323	3150		DCA TADC	/FROM ADC BUFER
0547				TADC=TEMP03	
0550	6324	6201	CALXM,	CDF	/JOBS CDF
0551	6325	7100		CLL	
0552	6326	1150		TAD TADC	/ADC BUFFER
0553					/VALUE FOR PNT
0554	6327	4762		JMS I LADZ	/LIST DBL
0555					/PRECISION ADD
0556	6330	7240		CLA CMA	
0557	6331	1152		TAD TJTYPE	/S.D.?
0560	6332	7650		SNA CLA	
0561	6333	5355		JMP CALMOR	/NO TYPE 1
0562					/75/PNT
0563	6334	1150		TAD TADC	/ADC BUFER VAL
0564	6335	4763		JMS I SQADZ	/SQUARE N LIST
0565					/TRIPLE
0566					/PRECISION ADD
0567	6336	7344		MTW	
0570	6337	1152		TAD TJTYPE	/TREND?

0571	6340	7650	SNA CLA		
0572	6341	5355	JMP CALMOR		/NO TYPE 2
0573					/400/PNT
0574	6342	1411	TAD I GETPNT		/LAST SWEEPS
0575	6343	3146	DCA TSAVE		/ADC BUFFER VAL
0576			TSAVE=TEMP01		/FOR THIS PNT
0577	6344	1150	TAD TADC		/SAVE THIS
0600					/SWEEPS ADC
0601					/BUFFER VALUE
0602	6345	3412	DCA I PUTPNT		/FOR THIS PNT
0603	6346	1154	TAD NSAVE		/NUM OF SWEEPS
0604	6347	7650	SNA CLA		/0?
0605	6350	5354	JMP .+4		/YES
0606					/X(1)-X(0)=0
0607	6351	1146	TAD TSAVE		/THIS SWEEP-
0610	6352	7041	CMA IAC		/LAST SWEEP
0611	6353	1150	TAD TADC		/SQUARE N LIST
0612					/TRIPLE ADD
0613	6354	4763	JMS I SQADZ		/TYPE 3:750/PNT
0614	6355	4471	CALMOR, JMS I BLKCNT		/MORE DATA
0615					/IN JOB?
0616	6356	3324	DCA CALXM		/FIELD CHANGE
0617					/CDF INST IN AC
0620	6357	4473	JMS I IXPNT		/NOT COMPLETE
0621					/UPDATE ADC
0622					/POINTER
0623	6360	5302	JMP CALJOB		/JOB COMPLETE
0624					/GET NEXT JOB
0625	6361	5321	JMP CALNXT		/NOT COMPLETE
0626					/GET NEXT POINT
0627	6362	7412	LADZ, LADDS		
0630	6363	7433	SQADZ, SQADS		
0631	6364	0276	K0276, 0276		
0632	6365	1030	CALEND, TAD WSWITZ		
0633	6366	7650	SNA CLA		
0634	6367	5431	JMP I WDUNZ		
0635	6370	2165	ISZ NSWPS		/SWEEPS DONE?
0636	6371	5377	JMP ADONE		/NO
0637	6372	3161	DCA VSW		/YES SET VIEW
0640					/SWITCH TO
0641					/AVERAGES
0642	6373	3164	DCA XMASK		/DO NOT SAMPLE
0643	6374	1364	TAD K0276		/TYPE ">"
0644	6375	3175	DCA KBDBUF		
0645	6376	7410	SKP		
0646	6377	7000	ADONE, NOP		
0647	6400	5456	JMP I ZCYCLE		/START AT TOP
0650					/OF DISP CYCLE
0651			/SUBROUTINE TO SENSE AND BRANCH ON PSEUDO KEYBOARD: IKBRAN		
0652			/ EXIT 1		/NORMAL RETURN
0653			/ EXIT 2		/<LF>
0654			/ EXIT 3		/<CR>
0655					
0656	6401	0000	IKBRAS, 0		/PROTECT AGNST
0657					/BAD CNTRL TAPE
0660	6402	1175	TAD KBDBUF		/EXAMINE
0661					/INTERRUPT KBD
0662	6403	7450	SNA		/KEY STRUCK?
0663	6404	5601	JMP I IKBRAS		/NO NEW KEY
0664	6405	4532	JMS I BRAN		/YES WHICH ONE ?
0665	6406	7572	IKLIST		
0666	6407	5331	JMP CTRLZ		/Z ZERO AND
0667					/RESTART

0670	6410	5376		JMP CTRLR	/R - RESTART
0671	6411	5321		JMP CTRLA	/A ALTER PARAMS
0672	6412	5306		JMP CTRLQ	/Q TERM AVGING
0673	6413	2201		ISZ IKBRAS	/C.R. EXIT TO
0674					/CALL+3
0675	6414	5241		JMP IKCRLF	/L.F. ECHO AND
0676					/EXIT TO CALL+2
0677	6415	5231		JMP IKV	/V COMPLEMENT
0700					/VIEW SWITCH
0701	6416	7344		MTW	/C CONTRACT
0702					/VIEW SCALE
0703	6417	5235		JMP IKXC	/X EXPAND
0704					/VIEW SCALE
0705	6420	5303		JMP CTRLD	/D GET MONITOR
0706	6421	5432		JMP I WRIZ	/WRITE MTP
0707	6422	5254		JMP IKT	/T TYPE DATA
0710	6423	5245		JMP IKP	/P PLOT MODE
0711	6424	1175	IKCONT,	TAD KBDBUF	/OTHERS ECHO
0712	6425	6046		TLS	/TYPE
0713	6426	3177		DCA TTYFLG	/SET FLAG CHAR
0714					/IN PROGRESS
0715	6427	3175		DCA KBDBUF	/CLR KBD BUFER
0716	6430	5601	IKEXIT,	JMP I IKBRAS	/RTN
0717	6431	1161	IKV,	TAD VSW	/V COMP. VIEW
0720					/SWITCH
0721	6432	7040		CMA	
0722	6433	3161		DCA VSW	
0723	6434	5224		JMP IKCONT	/ECHO AND EXIT
0724	6435	7140	IKXC,	CLL CMA	/-2 GOES TO 1
0725					/0 GOES TO -1
0726	6436	1166		TAD NSHFT	/C: INC. NSHFT
0727					/X: DECR. NSHFT
0730	6437	3166		DCA NSHFT	/NSHFT IS NUM
0731					/OF PLACES TO
0732					/SHIFT RIGHT
0733	6440	5224		JMP IKCONT	/ECHO AND EXIT
0734	6441	1370	IKCRLF,	TAD K0212	/CR OR LF
0735					/TYPE LF LATER
0736	6442	3176		DCA TTYBUF	
0737	6443	2201		ISZ IKBRAS	/XIT TO CALL+2
0740					/OR CALL+3
0741	6444	5224		JMP IKCONT	/ECHO AND EXIT
0742	6445	1164	IKP,	TAD XMASK	/IS AVGING DONE?
0743	6446	7640		SZA CLA	
0744	6447	5227		JMP IKEXIT-1	/NO DONT EVEN
0745					/ECHO "P"
0746	6450	1163		TAD PSWIT	/YES, COMP
0747					/PLOT SWITCH
0750	6451	7040		CMA	
0751	6452	3163		DCA PSWIT	
0752	6453	5224		JMP IKCONT	/ECHO AND EXIT
0753	6454	1163	IKT,	TAD PSWIT	/PLOTING DONE?
0754	6455	1164		TAD XMASK	/AVERAGE DONE?
0755	6456	7640		SZA CLA	
0756	6457	5227		JMP IKEXIT-1	/NO DONT ECHO
0757	6460	3155		DCA TNJOB	/PRESET TO
0760					/DUMP MODE
0761				TNJOB=TEMP10	
0762	6461	3156		DCA TNMIN	
0763				TNMIN=TEMP11	
0764	6462	3157		DCA TNMAX	
0765				TNMAX=TEMP12	
0766	6463	1175		TAD KBDBUF	/"<CRLF>T: "

0767	6464	4762		JMS I RDKBDX	/GET AVERAGE
0770					/TO BE OUTPUT
0771	6465	7450		SNA	/0 TO TYPE ALL
0772	6466	5276		JMP IKTGO	/IN DUMP MODE
0773					/TYPE ALL DATA
0774	6467	3155		DCA TNJOB	/-JOB
0775	6470	1371		TAD K0314	/"<CRLF>L: "
0776	6471	4762		JMS I RDKBDX	/GET LOWER
0777					/LIMIT (BIN)
1000	6472	3156		DCA TNMIN	
1001	6473	1372		TAD K0310	/"<CRLF>H: "
1002	6474	4762		JMS I RDKBDX	/GET UPPER
1003					/LIMIT (BIN)
1004	6475	3157		DCA TNMAX	
1005	6476	7240	IKTGO,	CLA CMA	
1006	6477	3160		DCA TSWIT	
1007	6500	1061		TAD K0215	
1010	6501	3175		DCA KBDBUF	
1011	6502	5601		JMP I IKBRAS	
1012	6503	6002	CTRLD,	IOF	
1013	6504	5705		JMP I .+1	
1014	6505	7607		MONITR	
1015	6506	7240	CTRLQ,	CLA CMA	/DO 1 MORE SWEEP
1016	6507	3165		DCA NSWPS	/AND THEN QUIT
1017	6510	5227		JMP IKEXIT-1	
1020	6511	7040	START,	CMA	
1021	6512	3030		DCA WSWITZ	
1022	6513	1373		TAD K201	/AVGER STARTS
1023					/HERE
1024	6514	3175		DCA KBDBUF	
1025	6515	1055		TAD K0100	/GET LINC SPEC
1026	6516	6141		LINC	/FCNS CONST
1027				LMODE	
1030	0517	0004		ESF	/ENAB FAST SAM
1031	0520	0002		PDP	
1032				PMODE	
1033					
1034	6521	5765	CTRLA,	JMP I TCALX	/SET UP CLOCK
1035	6522	1367		TAD K0316	/"<CRLF>N: "
1036	6523	4762		JMS I RDKBDX	/-NUM OF SWEEPS
1037	6524	3026		DCA NSWEP	
1040	6525	1366		TAD K0304	/"<CRLF>D: "
1041	6526	4762		JMS I RDKBDX	/-DELAY (IN
1042					/SAMPLING INT)
1043	6527	1115		TAD KM0001	
1044	6530	3030		DCA KSYTIM+1	
1045	6531	1370	CTRLZ,	TAD K0212	/LFCRLF
1046	6532	4764		JMS I TMESSX	
1047	6533	1023		TAD ADJLIS	/START AT 1ST
1050	6534	3013		DCA JPNT	/JOB
1051	6535	6201	ZNXT,	CDF 0	
1052	6536	4470		JMS I JGET	/GET JOB PARAMS
1053	6537	5374		JMP ADINIT	/NO MORE JOBS
1054					/INIT SWEEP
1055					/CNTRL TO
1056					/ACCEPT DATA
1057	6540	2013		ISZ JPNT	/SKIP OVER DISP
1060	6541	2013		ISZ JPNT	/IN J6 N J7
1061	6542	3350		DCA ZXN	/CDF INST IN AC
1062	6543	4763		JMS I JCELLX	/GET NUM OF
1063					/CELLS -2 FOR
1064					/JOBS DATA POINT
1065	6544	1114		TAD K0002	/NUM OF CELLS

1066					/PER DATA POINT
1067					/IS N
1070	6545	7041		CMA IAC	
1071	6546	3147		DCA TZCELL	
1072				TZCELL=TEMP02	
1073	6547	7240		CLA CMA	/FIRST ITEM HAS
1074					/SWEEP CNT (NOT
1075					/CNTED IN LENTH)
1076	6550	6201	ZXM,	CDF	
1077	6551	1147		TAD TZCELL	/FOR EACH PNT 0
1100	6552	3146		DCA TZCNT	/N CELLS
1101				TZCNT=TEMP01	
1102	6553	3411		DCA I GETPNT	/GETPNT INIT
1103					/BY JGETS
1104	6554	2146		ISZ TZCNT	
1105	6555	5353		JMP .-2	
1106	6556	4471		JMS I BLKCNT	/NEXT POINT?
1107	6557	3350		DCA ZXM	/YES BUT CHANGE
1110					/CDF
1111	6560	5350		JMP ZXM	/CONTINUE
1112	6561	5335		JMP ZNXT	/END OF JOB
1113					/GET NEXT JOB
1114	6562	7211	RDKBDX,	RDKBDS	/LOCAL
1115	6563	6761	JCELLX,	JCELLS	/CROSSPAGE
1116	6564	7331	TMESSX,	TMESS	
1117	6565	7245	TCALX,	TCALS	
1120					/ASCII CHARS
1121	6566	0304	K0304,	304	/D
1122	6567	0316	K0316,	316	/N
1123	6570	0212	K0212,	212	/LF
1124	6571	0314	K0314,	314	/L
1125	6572	0310	K0310,	310	/H
1126	6573	0201	K201,	201	
1127					/INIT PNTRS
1130					/COUNTERS
1131					/AND BUFFERS
1132	6574	1113	ADINIT,	TAD K0003	/SET SCALE
1133					/TO 1/8
1134	6575	3166		DCA NSHFT	
1135				NSHFT=TEMP21	
1136	6576	1370	CTRLR,	TAD K0212	/PUT LFEED IN
1137	6577	3175		DCA KBDBUF	/KBD BUFF
1140	6600	1026		TAD NSWEP	/SET AUTO STOP
1141	6601	3165		DCA NSWPS	/COUNTER
1142				NSWPS=TEMP20	
1143	6602	1025		TAD SMASK	/SET AVGING
1144	6603	3164		DCA XMASK	/SWITCH
1145				XMASK=TEMP17	
1146	6604	3163		DCA PSWIT	/NO PLOTTING
1147				PSWIT=TEMP16	/PERMITTED
1150	6605	6133	CYCLE,	CLAB	
1151	6606	6132		CLLR	
1152	6607	1055		TAD K0100	
1153	6610	6132		CLLR	
1154	6611	6135		CLSA	
1155	6612	6141		LINC	
1156	6613	0110		110	/SAM 10
1157	6614	0002		2	/PDP
1160	6615	7200		CLA	
1161	6616	1051		TAD CLKCNT	
1162	6617	7041		CMA IAC	
1163	6620	6133		CLAB	
1164	6621	7200		CLA	

1165	6622	1043		TAD CLKMOD	/GET KW12 MODE
1166	6623	6134		CLEN	
1167	6624	7200		CLA	
1170	6625	1355		TAD ADRCLK	/SETUP INTERUPT
1171					/SERVICE
1172	6626	3065		DCA INTERX	/INTERRUPT GOES
1173					/TO SWEEP ADC
1174	6627	4760	KBDCHK,	JMS I IKBRAX	/LOOK AT
1175					/INTERRUPT KBRD
1176	6630	5236		JMP RSTART	/NO CHAR GO DISP
1177	6631	7240		CLA CMA	/L.F. COMES HERE
1200					/FOR 1ST SWEEP
1201					/SET MODE TO
1202					/PAUSE
1203	6632	3162		DCA ASWIT	/C.R. SET MODE
1204					/TO "COMPUTE"
1205				ASWIT=TEMP15	
1206	6633	1162		TAD ASWIT	/ASWIT=-1 FOR
1207					/PAUSE ASWIT=0
1210					/FOR COMPUTE
1211	6634	3161		DCA VSW	/INPUT FOR PAUSE
1212					/AVERAGES FOR
1213					/COMPUTE
1214				VSW=TEMP14	
1215	6635	3160		DCA TSWIT	/SET TO NO TYPE
1216				TSWIT=TEMP13	
1217	6636	6001	RSTART,	ION	/INIT
1220	6637	1161	DISWIT,	TAD VSW	/VIEW INPUT OR
1221					/PARTIAL SUMS?
1222	6640	7640		SZA CLA	
1223	6641	5340		JMP VINPUT	/VIEW INPUT
1224	6642	1023		TAD ADJLIS	/VEIW PART SUMS
1225	6643	3013		DCA JPNT	/START WITH
1226					/1ST JOB
1227	6644	6201	VJSTAR,	CDF 0	
1230	6645	4470		JMS I JGET	/VIEW PART SUMS
1231	6646	5227		JMP KBDCHK	/NO MORE JOBS
1232					/LOOK AT KBRD
1233	6647	3277		DCA VJXM	/CDF INST
1234	6650	4474		JMS I SDIS	/SETUP DISP
1235					/PARAMETERS
1236	6651	7000		NOP	/SKIPPED (KBUG)
1237	6652	4361		JMS JCELLS	/GET NUM OF LOC
1240					/PER POINT-2
1241	6653	3151		DCA TDSKIP	/DONT DISP S.D.
1242					/OR TREND
1243				TDSKIP=TEMP04	
1244	6654	1277		TAD VJXM	/SET UP FIELD
1245	6655	3256		DCA VJXMN	/TO GET N
1246	6656	6201	VJXMN,	CDF	
1247	6657	1411		TAD I GETPNT	/NUM OF SWEEPS
1250	6660	3171		DCA ARITH2	
1251	6661	1160		TAD TSWIT	/CHK TYP SWITCH
1252	6662	7650		SNA CLA	/NE. 0 TO TYPE
1253	6663	5277		JMP VJXM	/DONT TYPE
1254	6664	1155		TAD TNJOB	/DUMP MODE?
1255	6665	7650		SNA CLA	
1256	6666	5271		JMP .+3	/YES START
1257					/TYPING
1260	6667	2155		ISZ TNJOB	/NO IS THIS THE
1261					/JOB TO BE
1262					/TYPED?
1263	6670	5277		JMP VJXM	/NO DISP THIS 1

1264	6671	4756		JMS I CRLFX	/TYPE CRLF
1265					/SET ITEM CTR
1266	6672	4757		JMS I DECTYX	/TYPE N
1267	6673	1166		TAD NSHFT	/GET SCALE
1270	6674	3171		DCA ARITH2	/FACTOR SAV FOR
1271					/TYPING
1272	6675	4757		JMS I DECTYX	/TYPE SCALE
1273					/FACTOR
1274	6676	4756		JMS I CRLFX	/TYPE CRLF
1275					/SET CR COUNT
1276	6677	6201	VJXM,	CDF	/GET DATA FIELD
1277	6700	1411		TAD I GETPNT	/GET DATA PNT
1300					/(LO ORDER,
1301					/HI ORDER)
1302	6701	3171		DCA ARITH2	
1303	6702	1411		TAD I GETPNT	/AND PUT IN
1304	6703	3170		DCA ARITH1	/SHIFT REG
1305	6704	7344		MTW	/OUTPUT IN MV
1306					/DEC SF BY 4
1307	6705	1166		TAD NSHFT	
1310	6706	7041		CMA IAC	
1311	6707	4533		JMS I SHFT	
1312	6710	1160		TAD TSWIT	/CHK TYPE SWIT
1313	6711	7650		SNA CLA	
1314	6712	5327		JMP VJDIS	/TYPE NOT ENAB
1315	6713	1155		TAD TNJOB	/IS THIS JOB
1316	6714	7640		SZA CLA	/TO BE TYPED?
1317	6715	5327		JMP VJDIS	/NO JUST DISP
1320	6716	1156		TAD TNMIN	/REACHED THE
1321	6717	7650		SNA CLA	/FIRST POINT?
1322	6720	5323		JMP .+3	
1323	6721	2156		ISZ TNMIN	/NO COUNT DOWN
1324	6722	7410		SKP	/JUST DISPLAY
1325	6723	4757		JMS I DECTYX	/YES TYPE
1326					/DATA POINT
1327	6724	2157		ISZ TNMAX	/REACHED THE
1330					/LAST POINT?
1331	6725	7410		SKP	/NO CONTINUE
1332	6726	2155		ISZ TNJOB	/YES DISABLE
1333					/TYPEOUT
1334	6727	7326	VJDIS,	TWO	/SCALE TWO MORE
1335					/RIGHT (COUNTS
1336					/FROM MV)
1337	6730	4475		JMS I DISP	/JOB SCALE, BIAS
1340					/ DISPLAY
1341	6731	1151		TAD TDSKIP	/MOVE PNTR OVER
1342					/VAR AND TREND
1343	6732	1011		TAD GETPNT	
1344	6733	3011		DCA GETPNT	
1345	6734	4471		JMS I BLKCNT	/CHECK BLOCK
1346					/LENGTH
1347	6735	3277		DCA VJXM	/NEW DATA FIELD
1350	6736	5277		JMP VJXM	/FILE NOT YET
1351					/COMPLETE
1352	6737	5244		JMP VJSTAR	/GET NEXT JOB
1353					/AND DISPLAY IT
1354					/VIEW RAW INPUT
1355	6740	1024	VINPUT,	TAD ADCHNL	/GET START OF
1356					/CHANEL DISPLAY
1357	6741	3013		DCA JPNT	/LIST
1360	6742	3150		DCA TORD	/LOOK AT 1ST
1361					/OF LIST
1362					/(ORDER=0)

1363	6743	4472	VINEXT,	JMS I SETPNT	/SET SWEEP
1364					/PARAMS N PNTRS
1365	6744	4474		JMS I SDIS	/JPNT POINTS TO
1366					/DISP WDS D1,D2
1367	6745	5227		JMP KBDCHK	
1370	6746	2150		ISZ TORD	/MOVE CHANNEL
1371					/COUNTER
1372	6747	1553	VILoop,	TAD I TBUFAD	/GET DATA POINT
1373	6750	3171		DCA ARITH2	/LOAD IN LOW
1374					/ORDER SHFT REG
1375	6751	4475		JMS I DISP	/DISPLAY
1376	6752	4473		JMS I IXPNT	/DUN THIS CHAN?
1377	6753	5343		JMP VINEXT	/YES NEXT CHAN
1400	6754	5347		JMP VILoop	/NO GET NXT PNT
1401					/LOCAL CROSSPAGE
1402	6755	7465	ADRCLK,	CLKINT	
1403	6756	6247	CRLFX,	CRLFS	
1404	6757	6200	DECTYX,	DECTYP	
1405	6760	6401	IKBRAX,	IKBRAS	
1406					/SUBRT TO GET
1407					/(CELLS-2) FROM
1410					/JOB TYPE
1411	6761	0000	JCELLS,	0	/CELLS-2 PER
1412					/DATA POINT IN
1413					/CALC BUFFER
1414	6762	1152		TAD TJTYPE	/1, 2, OR 3
1415	6763	4532		JMS I BRAN	
1416	6764	0113		K0003	
1417	6765	1112		TAD K0004	/TYPE 3
1420					/7 CELLS+2
1421	6766	1113		TAD K0003	/TYPE 2
1422					/3 CELLS+2
1423	6767	5761		JMP I JCELLS	/TYPE 1
1424					/0 CELLS+2
1425				PAGE	
1426				/THIS SUBROUTINE DECODES DISPLAY WORDS: SDIS	
1427				/POINTER IS AUTOINDEX REGISTER "AXPNTR"	
1430				/CALL: SDIS	
1431				/	END OF LIST RETURN
1432				/	NORMAL RETURN
1433					AXPNTR=JPNT
1434	7000	0000	SDISS,	0	
1435	7001	1163		TAD PSWIT	/PLOT MODE?
1436	7002	7650		SNA CLA	
1437	7003	5213		JMP K0037+1	/NO, CONTINUE
1440	7004	1173	ZZ1,	TAD ARITH4	
1441	7005	3007		DCA LXCORD	
1442	7006	1261		TAD YZ	
1443	7007	4076		JMS LDIS	
1444	7010	4665		JMS I IKBRAY	/WAIT FOR
1445					/CR OR LF
1446	7011	5204		JMP ZZ1	
1447	7012	0037	K0037,	0037	/NOP
1450	7013	1413		TAD I AXPNT	/GET D1
1451					/DX(8),YS(4)
1452	7014	7450		SNA	
1453	7015	5600		JMP I SDISS	/EOL IF D1=0
1454	7016	3171		DCA ARITH2	
1455	7017	1171		TAD ARITH2	
1456	7020	0255		AND K0017	/HAVE Y SCALE
1457	7021	3260		DCA YS	/FACTOR
1460	7022	3170		DCA ARITH1	/0 TO HI ARITH0
1461	7023	1112		TAD K0004	/DELTA-X

1462	7024	4533		JMS I SHFT	/MOVE BINARY
1463					/POINT TO END
1464					/OF ARITH2
1465	7025	1170		TAD ARITH1	
1466	7026	3262		DCA DELTAX	
1467	7027	1171		TAD ARITH2	
1470	7030	3263		DCA DELTAX+1	
1471	7031	1413		TAD I AXPNTR	/GET D2
1472	7032	3171		DCA ARITH2	/XZ(6),YZ(6)
1473	7033	1113		TAD K0003	/MOVE YZ TO 9
1474	7034	4533		JMS I SHFT	/SIGNIF BITS
1475	7035	1171		TAD ARITH2	
1476	7036	3261		DCA YZ	
1477	7037	1264		TAD KM0006	/MOVE XZ TO 9
1500	7040	4533		JMS I SHFT	/BITS OF ARITH2
1501	7041	1171		TAD ARITH2	/GET RID OF LOW
1502	7042	0264		AND KM0006	/ORDER BITS
1503	7043	3173		DCA ARITH4	/FOP HOLDS
1504					/CURRENT X-VAL
1505	7044	1163		TAD PSWIT	/PLOT MODE?
1506	7045	7650		SNA CLA	
1507	7046	5256		JMP K0017+1	/NO, CONTINUE
1510	7047	1173	zz2,	TAD ARITH4	
1511	7050	3007		DCA LXCORD	
1512	7051	1261		TAD YZ	
1513	7052	4076		JMS LDIS	
1514	7053	4665		JMS I IKBRAY	/WAIT FOR
1515	7054	5247		JMP zz2	/CR OR LF
1516	7055	0017	K0017,	0017	/NOP
1517	7056	2200		ISZ SDISS	/XIT TO CALL+2
1520	7057	5600		JMP I SDISS	
1521					/LOCAL VARS
1522	7060	0000	YS,	0	/SCALE FACTOR
1523					/FOR Y
1524	7061	0000	YZ,	0	/SCP Y FOR Y=0
1525	7062	0000	DELTAX,	0	/INCREMENT BET
1526					/PNTS (INTEGER
1527					/PART)
1530	7063	0000		0	/((FRACTNL PART)
1531					/LOCAL CONSTANTS
1532	7064	7772	KM0006,	-0006	
1533	7065	6401	IKBRAY,	IKBRAS	
1534					/SUBROUTINE TO DISPLAY POINT OF DATA
1535					/(X-INCREMENTED):DISP
1536					/DATA VALUE SHOULD BE IN ARITH1
1537					/ARITH0+2
1540	7066	0000	DISPS,	0	/AC+(YS)=PLACES
1541	7067	1260		TAD YS	/TO SHIFT RIGHT
1542	7070	7041		CMA IAC	
1543	7071	4533		JMS I SHFT	/SCALE
1544	7072	1173		TAD ARITH4	/MOVE X TO NEW
1545	7073	3007		DCA LXCORD	/VALUE
1546	7074	1261		TAD YZ	/BIAS
1547	7075	1171		TAD ARITH2	
1550	7076	4076		JMS LDIS	/DO LINC MODE
1551					/DISPLAY
1552	7077	7200	DISRTN,	CLA	/RTN FRM PAGE 0
1553					/DISPLAY RT
1554	7100	1163		TAD PSWIT	/PLOT MODE?
1555	7101	7650		SNA CLA	
1556	7102	5306		JMP .+4	/NO, CONTINUE
1557	7103	3050		DCA PLTDLY	
1560	7104	2050		ISZ PLTDLY	/YES, WAIT A

```

1561      7105  5304      JMP  .-1          /WHILE
1562      7106  1262      TAD DELTAX       /LOAD X INCR
1563      7107  3170      DCA ARITH1
1564      7110  1263      TAD DELTAX+1
1565      7111  3171      DCA ARITH2
1566      7112  4534      JMS I DADD       /ADD TO CURNT X
1567      7113  5666      JMP I DISPS
1570
1571      /THIS SUBROUTINE DECODES JOB LIST
1572      /AND SETS JOB POINTERS (136)
1573      /      J1: A/B(1), CHAN(5), 1(1), CHAN ORDER(5)
1574      /      J2: JOB TYPE(4), CONTINGENCY(8)
1575      /      J3: LINKAGE WORD (L1) TO FIRST BLOCK
1576      /      J4: LINKAGE WORD (L2) TO FIRST BLOCK
1577      /      J5: LINKAGE WORD (L3) TO FIRST BLOCK
1600      /      J6: DISPLAY WORD D1
1601      /      J7: DISPLAY WORD D2
1602      /PARAMETERS OF JOB:
1603      /      TYPE: - TYPE IN "TJTYPE"
1604      /      CONTINGENCY: "TCMASK", CHAN ORDER: "TORD"
1605      /      COUNT FOR 1ST BLK: "TCBLK"
1606      /      LOCATION OF 1ST -1: "GETPNT"
1607      /      CDF INSTRUCTION: AC
1610      /      AUTOINDEX "JPNT" POINTS TO J5 AT EXIT (J6 IS NEXT)
1611      7114  0000      JGETS,  0
1612      7115  1413      TAD I JPNT       /GET J1
1613      7116  7450      SNA              /J1=0 FOR EOL
1614      7117  5714      JMP I JGETS      /J1=0 EXIT
1615      7120  3170      DCA ARITH1       /SAVE J1 IN
1616      7121  1413      TAD I JPNT       /SHIFT REG
1617      7122  3171      DCA ARITH2       /GET J2
1620      /LO-ORDER
1621      7123  1170      TAD ARITH1       /SHIFT REG
1622      /GET ORDER IN
1623      7124  0345      AND KK37         /MX LIST
1624      7125  3150      DCA TORD         /5 LOW ORDER
1625      TORD=TEMP03   /BITS OF J1
1626      7126  1112      TAD K0004        /GET JOB TYPE
1627      7127  4533      JMS I SHFT
1630      7130  1170      TAD ARITH1
1631      7131  0346      AND KK17         /4 HI ORDER OF
1632      /J2
1633      7132  3152      DCA TJTYPE       /--JOB TYPE
1634      TJTYPE=TEMP05
1635      7133  1413      TAD I JPNT       /GET -NUM OF
1636      7134  3344      DCA CNBLK        /ITEMS IN NEXT
1637      /BLOCK (J3)
1640      7135  1413      TAD I JPNT       /GET CDF (J4)
1641      7136  3146      DCA TCDF
1642      TCDF=TEMP01
1643      7137  1413      TAD I JPNT       /GET ST ADDR-1
1644      7140  3011      DCA GETPNT       /FOR DATA
1645      /BLOCK (J5)
1646      7141  2314      ISZ JGETS        /XIT TO CALL+2
1647      7142  1146      TAD TCDF         /CDF INST TO AC
1650      7143  5714      JMP I JGETS
1651      /VARS FOR JGETS
1652      /AND BLKCNS
1653      7144  0000      CNBLK,  0       /-NUM OF ITEMS
1654      /LEFT IN SWEEP
1655
1656      7145  0037      KK37,   37
1657      7146  0017      KK17,   17

```

```

1660 /THIS SUBROUTINE COUNTS DOWN BLOCK
1661 /ENTRIES AND LINKS TO NEXT BLOCK:BLKCNT
1662 /CALL: BLKCNT
1663 / NEW FIELD RETURN
1664 / NORMAL RETURN
1665 / END OF FILE RETURN
1666 7147 0000 BLKCNS, 0
1667 7150 2344 ISZ CNBLK /COUNT ITEMS
1670 /IN BLOCK
1671 7151 5366 JMP BLKOK /NO OVERFLOW
1672 7152 1411 TAD I GETPNT /OVERFLOW GET
1673 /NEXT BLOCK
1674 7153 7450 SNA /L1=0 IS EOF
1675 7154 5365 JMP BLKFIN /EOF EXIT
1676 7155 3344 DCA CNBLK /SET COUNT FOR
1677 /NEXT BLOCK
1700 7156 1411 TAD I GETPNT
1701 7157 3146 DCA TCDF /GET NEW CDF
1702 7160 1411 TAD I GETPNT /GET ADDRESS-1
1703 7161 3011 DCA GETPNT /FOR NEXT BLOCK
1704 7162 4135 JMS KLUG
1705 7163 1146 TAD TCDF /CDF TO AC.
1706 7164 5747 JMP I BLKCNS /NEW FIELD
1707 /EXIT TO CALL+1
1710 7165 2347 BLKFIN, ISZ BLKCNS /END OF FILE
1711 /EXIT TO CALL+3
1712 7166 2347 BLKOK, ISZ BLKCNS /CONTINUE IN
1713 /THIS BLOCK
1714 7167 5747 JMP I BLKCNS /EXIT TO CALL+2
1715 7170 0000 IXPNTS, 0
1716 7171 1045 TAD NCHA /NUM OF CELLS
1717 /IN DATA POINT
1720 7172 1153 TAD TBUFAD /+CURRENT
1721 /BUFFER ADDR
1722 7173 3153 DCA TBUFAD /UPDATE ADC
1723 /BUFFER
1724 7174 2377 ISZ NPOINT /IF CNTR DOESNT
1725 /OVERFLOW
1726 7175 2370 ISZ IXPNTS /EXIT TO CALL+2
1727 7176 5770 JMP I IXPNTS /OTHERWISE
1730 /TO CALL+1
1731 /VARS FOR
1732 /SETPNS, IXPNTS
1733 7177 0000 NPOINT, 0 /-NUM OF ITEMS
1734 /LEFT IN SWEEP
1735 PAGE
1736 /THIS SUBROUTINE SETS
1737 /SWEEP PARAMETERS: SETPNT
1740 7200 0000 SETPNS, 0
1741 7201 1044 TAD SAMA /GET -NUM
1742 7202 3610 DCA I KPOINT /OF DATA POINTS
1743 7203 1047 TAD ADBUFA /GET LOGICAL 1
1744 /OF BUFFER
1745 7204 7001 IAC
1746 7205 1150 TAD TORD /GET ORDER OF
1747 /CHANNEL IN
1750 /BUFFER
1751 /((SET BY JGET)
1752 7206 3153 DCA TBUFAD /SET ADDR OF
1753 /1ST DATA POINT
1754 TBUFAD=TEMP06
1755 7207 5600 JMP I SETPNS
1756 7210 7177 KPOINT, NPOINT

```

```

1757                                     /SR TO ACCEPT
1760                                     /DBL PREC
1761                                     /DECIMAL ARG
1762      7211  0000  RDKBDS,  0
1763      7212  4331                JMS TMESS          /ASK FOR ARG
1764      7213  3171  RDLOOP,  DCA ARITH2
1765      7214  6031                KSF
1766      7215  5214                JMP , -1
1767      7216  6034                KRS
1770      7217  4342                JMS TYPES          /ECHO
1771      7220  1147                TAD KSAVE          /CHECK FOR CR
1772      7221  7041                CMA IAC
1773      7222  1061                TAD K0215
1774      7223  7650                SNA CLA          /CR ?
1775      7224  5240                JMP RDONE          /YES ARG COMP
1776      7225  1147                TAD KSAVE          /NO GET DIGIT
1777      7226  0243                AND K017
2000      7227  3174                DCA ARITH5
2001      7230  7001                IAC          /ADD OLD SUM
2002      7231  4533                JMS I SHFT        /*10(10) TO NUM
2003      7232  4534                JMS I DADD
2004      7233  7326                TWO
2005      7234  4533                JMS I SHFT
2006      7235  4534                JMS I DADD
2007      7236  1174                TAD ARITH5
2010      7237  5213                JMP RDLOOP
2011      7240  1171  RDONE,  TAD ARITH2          /GET - ARGUMENT
2012      7241  7041                CMA IAC
2013      7242  5611                JMP I RDKBDS
2014      7243  0017  K017,  17
2015      7244  7773  M0005,  -5
2016                                     /SR TO SET UP
2017                                     /KW12 CNT TCALS
2020      7245  7200  TCALS,  CLA
2021      7246  1062                TAD K0322
2022      7247  4211                JMS RDKBDS          /GET CLOCK RATE
2023      7250  7200                CLA
2024      7251  3170                DCA ARITH1
2025      7252  3173                DCA ARITH4
2026      7253  3174                DCA ARITH5
2027      7254  3051                DCA CLKCNT          /CLEAR COUNT
2030      7255  4534                JMS I DADD          /MULT RATE BY 2
2031      7256  4534                JMS I DADD
2032      7257  1244                TAD M0005
2033      7260  3171                DCA ARITH2
2034      7261  7040                CMA
2035      7262  3170                DCA ARITH1
2036      7263  2051  DVLOOP,  ISZ CLKCNT
2037      7264  4534                JMS I DADD
2040      7265  1173                TAD ARITH4
2041      7266  7700                SMA CLA
2042      7267  5263                JMP DVLOOP
2043      7270  5671                JMP I , +1
2044      7271  6522                CTRLA+1
2045      /BASIC SUBROUTINES SHFT, DADD, AND BRAN [SU63AB]
2046      /SUBROUTINE TO SHIFT DOUBLE PRECISION WORD (SHFR): SHFT (10+6N)
2047      /CALL:  TAD KXXXX          /AC HOLDS SHFT COUNT, RIGHT IS NEGATIVE
2050      /
2051      /      SHFT
2052      /      RETURN          /LINK=0, AC=0
2053      /FORMAT OF DOUBLE WORD IS (HI,LO) HI(0)-ONLY-HOLDS SIGN
2054      /SIGN BIT WILL BE REPLICATED IN RIGHT SHIFTS
2055      /TEMPORARY STORAGE ALLOCATION
          SHCNT=TEMP01

```

```

2056          /ARITHMETIC REGISTER ALLOCATION
2057          SHFR=ARITH1          /ARITH1-2 ARE
2060          /FOR SHIFTING
2061      7272  0000  SHFTS,  0
2062      7273  7100          CLL
2063      7274  7450          SNA          /IF SHIFT COUNT
2064      7275  5672          JMP I SHFTS          /=0 EXIT
2065      7276  7500          SMA          /RIGHT OR LEF ?
2066      7277  7061          CML CMA IAC          /LEF SET LINK=1
2067          /AND COUNT NEG
2070      7300  3146          DCA SHCNT
2071      7301  7430          SZL          /RIGHT SHIFT?
2072      7302  5317          JMP SHLEFT          /NO-SHIFT LEFT
2073      7303  1170  SHRIHT, TAD SHFR          /SHIFT DONE
2074          /ON ARITH1-2
2075      7304  7510          SPA          /SET L=1 IF NEG
2076      7305  7020          CML
2077      7306  7010          RAR
2100      7307  3170          DCA SHFR          /SHFT WITH
2101          /SIGN REP
2102      7310  1171          TAD SHFR+1          /SHIFT LO ORDER
2103          /HALF
2104      7311  7010          RAR
2105      7312  3171          DCA SHFR+1
2106      7313  7100          CLL
2107      7314  2146          ISZ SHCNT          /ENOUGH SHIFTS?
2110      7315  5303          JMP SHRIHT          /NO-CONTINUE
2111      7316  5672          JMP I SHFTS          /YES-EXIT
2112      7317  1171  SHLEFT, TAD SHFR+1          /SHIFT LO-ORDER
2113      7320  7104          CLL RAL          /0 TO LSB
2114      7321  3171          DCA SHFR+1
2115      7322  1170          TAD SHFR          /SHIFT HI-ORDER
2116      7323  7004          RAL
2117      7324  3170          DCA SHFR
2120      7325  7100          CLL
2121      7326  2146          ISZ SHCNT          /ENOUGH?
2122      7327  5317          JMP SHLEFT          /NO-CONTINUE
2123      7330  5672          JMP I SHFTS
2124      7331  0000  TMESS,  0          /ROUT TO TYPE
2125          /<<CRLF>*:
2126      7332  3762          DCA I TMCHX          /*CHAR IN AC
2127      7333  1363          TAD ADMESS          /ADDR OF TEXT
2130      7334  3017          DCA TMPNTR
2131          TMPNTR=17
2132      7335  1417  TMLoop, TAD I TMPNTR          /GET CHARACTER
2133      7336  7450          SNA          /END OF TEXT?
2134      7337  5731          JMP I TMESS          /YES, EXIT
2135      7340  4342          JMS TYPES
2136      7341  5335          JMP TMLoop
2137      7342  0000  TYPES,  0          /SUBRT TO
2140      7343  6002          IOF          /TYPE CHAR
2141      7344  6046          TLS          /IN AC
2142      7345  6041          TSF
2143      7346  5345          JMP .-1
2144      7347  6042          TCF
2145      7350  3147          DCA KSAVE
2146          KSAVE=TEMP02
2147      7351  6031          KSF
2150      7352  5742          JMP I TYPES
2151      7353  6036          KRB
2152      7354  1364          TAD KMCTRL
2153      7355  7640          SZA CLA
2154      7356  5742          JMP I TYPES

```

2155	7357	6201		CDF 0	
2156	7360	5761		JMP I +1	
2157	7361	6627		KBDCHK	
2160	7362	6267	TMCHX,	TMCH	
2161	7363	6264	ADMESS,	TXMESS-1	
2162	7364	7557	KMCTRL,	-221	
2163	7365	7404	NTOUT,	INTOUT	
2164	7366	6031	INKBD,	KSF	/KEYBOARD UP?
2165	7367	5372		JMP INTTY	/NO, LOOK AT TTY
2166	7370	6036		KRB	/YES, READ CHAR
2167	7371	3175		DCA KBDBUF	
2170	7372	6041	INTTY,	TSF	/TTY DONE?
2171	7373	5765		JMP I NTOUT	/NO RESET AC.L
2172	7374	6042		TCF	/YES, CLEAR FLAG.
2173	7375	3177		DCA TTYFLG	/CLEAR SOFTWARE
2174					/FLAG TTY NOT
2175					/IN PROGRESS
2176	7376	1176		TAD TTYBUF	/MORE TO TYPE?
2177	7377	7450		SNA	
2200	7400	5204		JMP INTOUT	/NO, EXIT
2201	7401	6046		TLS	/YES, TYPE IT
2202	7402	3177		DCA TTYFLG	/SET SOFTWARE
2203					/FLAG: TTY IN
2204					/PROGRESS
2205	7403	3176		DCA TTYBUF	/CLEAR BUFFER
2206	7404	1067	INTOUT,	TAD LSAVE	/RESTORE LINK
2207	7405	7104		CLL RAL	
2210	7406	1066		TAD ASAVE	/RESTORE AC
2211	7407	6244		RMF	
2212	7410	6001		ION	
2213	7411	5400		JMP I 0	
2214	7412	0000	LADDS,	0	/SUBRT TO DO
2215					/DBL PREC ADD
2216					/TO PUTPNT
2217					/GETPNT LIST
2220					/(28)
2221	7413	3146		DCA TLAD	
2222				TLAD=TEMP01	
2223	7414	1030		TAD WSWITZ	
2224	7415	7650		SNA CLA	
2225	7416	5433		JMP I INDATZ	
2226	7417	1146		TAD TLAD	/ADD LO-ORDER
2227	7420	1411		TAD I GETPNT	/UPDATE LO
2230	7421	3412		DCA I PUTPNT	/ORDER CALC PNT
2231	7422	1146		TAD TLAD	/DO SIGN EXTEND
2232					/OF ADC WORD
2233					/TO HI-ORDER
2234	7423	7710		SPA CLA	
2235	7424	7040		CMA	
2236	7425	7430		SZL	/L=1 INDICATES
2237					/LO-ORDER
2240					/OVERFLOW
2241	7426	7001		IAC	/ADD IN OVRFLO
2242	7427	1411		TAD I GETPNT	/UPDATE HI
2243	7430	3412		DCA I PUTPNT	/ORDER CALC PNT
2244	7431	7100		CLL	
2245	7432	5612		JMP I LADDS	
2246	7433	0000	SQADS,	0	/SUBRT TO SQRE
2247					/AC AND ADD TO
2250					/TRIP PREC LIST
2251					/(62+56N
2252					/APPX 300)
2253	7434	7550		SPA SNA	/GET MAGNITUDE

2254	7435	7161		CLL CML CMA IAC	
2255	7436	3147		DCA TSQAD	
2256				TSQAD=TEMP02	
2257	7437	1147		TAD TSQAD	/LOAD DBL
2260	7440	3171		DCA ARITH2	/PREC. AC
2261	7441	3170		DCA ARITH1	
2262	7442	3174		DCA ARITH5	
2263	7443	3173		DCA ARITH4	
2264	7444	1147	SQLLOOP,	TAD TSQAD	/MULTIPLIER
2265	7445	7450		SNA	/ANYMORE TO ADD
2266					/TO PART PROD?
2267	7446	5256		JMP SQDONE	/NO-UPDATE
2270					/CALC BUFFER
2271					/AND EXIT
2272	7447	7110		CLL RAR	/LINK TELLS
2273					/WHETHER TO ADD
2274					/IN MORE
2275	7450	3147		DCA TSQAD	/ROTAT FOR NEXT
2276	7451	7430		SZL	/L=1, ADD
2277					/MULTIPLICAND
2300					/TO PART PROD
2301	7452	4534		JMS I DADD	
2302	7453	7001		IAC	/SHIFT PART
2303	7454	4533		JMS I SHFT	/PROD 1 LEFT
2304	7455	5244		JMP SQLLOOP	/DO SOME MORE
2305	7456	1174	SQDONE,	TAD ARITH5	/ADD DBL PREC
2306					/AC TO 3-WORD
2307					/ENTRY
2310	7457	1411		TAD I GETPNT	/LO-ORDER
2311	7460	3412		DCA I PUTPNT	
2312	7461	7004		RAL	/OVERFLOW BIT
2313	7462	1173		TAD ARITH4	/PLUS HI ORDER
2314	7463	4212		JMS LADDS	/ADD TO 2-WORD
2315					/ENTRY
2316	7464	5633		JMP I SQADS	/TRIP PREC
2317					/IN ALL
2320	7465	6135	CLKINT,	CLSA	/COME HERE ON
2321					/CLOCK INTERPT
2322	7466	0164		AND XMASK	/CHK FOR
2323	7467	7650		SNA CLA	/SYNC MASK
2324	7470	5733		JMP I INKBDX	/NO SYNC TRY
2325					/OTHER DEVICES
2326	7471	1060		TAD KBLA	/NUM OF SAMPLES
2327	7472	3146	STORWC,	DCA BLA	/TO ACCEPT
2330				BLA=TEMP01	
2331	7473	1047		TAD ADBUFA	/STRT OF BUFF-1
2332	7474	3010	STORMA,	DCA BUFA	
2333	7475	1030		TAD KSYTIM*1	/-NUM SAMPLING
2334					/INTERVALS TO
2335	7476	3147		DCA SYTIM	/WAIT
2336				SYTIM=TEMP02	
2337	7477	1042		TAD KMODE	/SET KW12 CNTRL
2340	7500	6132		CLLR	/REGISTER
2341	7501	6135	WLOOP,	CLSA	/WAIT FOR
2342	7502	7700		SMA CLA	/PRE-ANALYSIS
2343	7503	5301		JMP .-2	/DELAY
2344	7504	2147		ISZ SYTIM	
2345	7505	5301		JMP WLOOP	
2346	7506	5312		JMP CHNSET	/BYPASS WAIT
2347					/ACCEPT SWEEP
2350					/INTO BUFFER
2351	7507	6135	ADLOOP,	CLSA	/WAIT FOR
2352	7510	7700		SMA CLA	/OVRFLO

2353	7511	5307		JMP ADLOOP	/NOT READY
2354	7512	2322	CHNSET,	ISZ LSAM	/GET CHANNEL
2355	7513	1322		TAD LSAM	
2356	7514	1335		TAD ZZ	
2357	7515	7640		SZA CLA	
2360	7516	5321		JMP LSAM-1	
2361	7517	1334		TAD K0110	
2362	7520	3322		DCA LSAM	
2363	7521	6141		LINC	
2364	7522	0110	LSAM,	110	/SAM INST
2365	7523	0002		2	/PDP
2366	7524	7510		SPA	
2367	7525	7001		IAC	/CVERT 1S COM
2370	7526	3410		DCA I BUFA	/STORE POINT
2371	7527	2146		ISZ BLA	/HAVE ENOUGH?
2372	7530	5307		JMP ADLOOP	/NO GET NEXT
2373					/CONVERSION
2374	7531	5732		JMP I .+1	/YES EXIT
2375	7532	6275		ADDUN	
2376	7533	7366	INKBDX,	INKBD	
2377	7534	0110	K0110,	110	
2400	7535	7667	ZZ,	-110-NCHAN	
2401			/SUBROUTINE FOR BRANCHING ON MATCH OF AC AGAINST TABLE: BRAN		
2402			/	BRAN	
2403			/	ADDRESS OF TABLE	
2404			/	RETURN HERE IF FIRST ENTRY MEETS MATCH	
2405			/	ETC.	
2406			/	NONE MATCH	
2407			/TABLE,	FIRST ENTRY	
2410			/	SECOND ENTRY	
2411			/	-LAST ENTRY	
2412			/TEMPORARY STORAGE ALLOCATION		
2413				BPNT=TEMP01	
2414				BSAVE=TEMP02	
2415	7536	0000	BRANS,	0	
2416	7537	3147		DCA BSAVE	
2417	7540	1736		TAD I BRANS	/GET ADDR OF 1ST
2420					/ENTRY OF MATCH
2421					/LIST
2422	7541	3146		DCA BPNT	
2423	7542	1546	BRLOOP,	TAD I BPNT	/LOOK AT ENTRY
2424	7543	7500		SMA	/GET MAGNITUDE
2425	7544	7041		CMA IAC	
2426	7545	2336		ISZ BRANS	/INDEX RETN ADDR
2427	7546	1147		TAD BSAVE	/MATCH FOUND?
2430	7547	7650		SNA CLA	
2431	7550	5736		JMP I BRANS	/YES-EXIT TO RTN
2432					/AS CALCULATED
2433	7551	1546		TAD I BPNT	/NO-TEST FOR
2434					/LAST ENTRY.
2435	7552	2146		ISZ BPNT	/INDEX ENTRY
2436					/POINTER
2437	7553	7700		SMA CLA	/-INDICATES THIS
2440					/ WAS LAST ENTRY
2441	7554	5342		JMP BRLOOP	/NOT LAST CONT
2442	7555	2336		ISZ BRANS	/EXIT NOT IN
2443					/LIST NO MATCH
2444	7556	5736		JMP I BRANS	
2445			/SUBROUTINE TO DO DOUBLE PRECISION ADD OF ARITH1-2, AND 4-5: DADD (21)		
2446			/ARITHMETIC REGISTER ALLOCATION		
2447				DBLAC=ARITH1	
2450				DBLARG=ARITH4	
2451	7557	0000	DADDS,	0	/ADD LO-ORDER

2452	7560	7300		CLL	CLA	
2453	7561	1171		TAD	DBLAC+1	
2454	7562	1174		TAD	DBLARG+1	
2455	7563	3174		DCA	DBLARG+1	
2456	7564	7004		RAL		/CARRY
2457	7565	1170		TAD	DBLAC	/ADD HI-ORDER
2460	7566	1173		TAD	DBLARG	
2461	7567	3173		DCA	DBLARG	/LEAVE IN
2462						/ARITH4-5.
2463	7570	7100		CLL		
2464	7571	5757		JMP	I DADDS	
2465	7572	0232	IKLIST,	+232		/Z
2466	7573	0222		+222		/R
2467	7574	0201		+201		/A
2470	7575	0221		+221		/Q
2471	7576	0215		+215		/CR
2472	7577	0212		+212		/LF
2473	7600	0326		+326		/V
2474	7601	0303		+303		/C
2475	7602	0330		+330		/X
2476	7603	0204		+204		/D
2477	7604	0327		+327		/W
2500	7605	0324		+324		/T
2501	7606	7460		-320		/P - PLOT MODE
2502	7607	6141	MONITR,	LINC		/SR TO SET UP
2503				LMODE		/DIAL BOOT STRAP
2504	1610	0076		SET	I 16	
2505	1611	2015		2015		
2506	1612	0642		LDF	2	
2507	1613	1020		LDA	I	
2510	1614	0721		RCG	I	
2511	1615	1076		STA	I 16	/TO LDF 2
2512	1616	1020		LDA	I	
2513	1617	7300		7300		
2514	1620	1076		STA	I 16	
2515	1621	0602		LIF	2	/RTN TO
2516	1622	0643		LDF	3	
2517	1623	6016		JMP	16	/DIAL
2520				PMODE		
2521	7624	1175	WRITE,	TAD	KBDBUF	
2522	7625	4434		JMS	I TMESSZ	
2523	7626	3171	LOOPW,	DCA	ARITH2	
2524	7627	6031		KSF		
2525	7630	5227		JMP	.-1	
2526	7631	6034		KRS		
2527	7632	4435		JMS	I TYPEZ	
2530	7633	1147		TAD	KSAVE	
2531	7634	7041		CMA	IAC	
2532	7635	1061		TAD	K0215	
2533	7636	7650		SNA	CLA	
2534	7637	5246		JMP	ARGDUN	
2535	7640	1147		TAD	KSAVE	
2536	7641	0036		AND	KZ7	
2537	7642	1171		TAD	ARITH2	
2540	7643	7106		RTL	CLL	
2541	7644	7004		RAL		
2542	7645	5226		JMP	LOOPW	
2543	7646	1171	ARGDUN,	TAD	ARITH2	
2544	7647	7112		RTR	CLL	
2545	7650	7010		RAR		
2546	7651	1357		TAD	KK5K	
2547	7652	3311		DCA	TPWORD	
2550	7653	3030		DCA	WSWITZ	

2551	7654	3162		DCA ASWIT
2552	7655	1104		TAD KM200Z
2553	7656	3360		DCA CTR
2554	7657	1107		TAD KZ377
2555	7660	3015		DCA 15
2556	7661	5505		JMP I STWZ
2557	7662	6214	INDATA,	RDF
2560	7663	1272		TAD KCDF0
2561	7664	3274		DCA HERE
2562	7665	1274		TAD HERE
2563	7666	3300		DCA THERE
2564	7667	1274		TAD HERE
2565	7670	3320		DCA WHERE
2566	7671	1411		TAD I GETPNT
2567	7672	6201	KCDF0,	CDF 0
2570	7673	3415		DCA I 15
2571	7674	0000	HERE,	0
2572	7675	1411		TAD I GETPNT
2573	7676	6201		CDF 0
2574	7677	3415		DCA I 15
2575	7700	0000	THERE,	0
2576	7701	2360		ISZ CTR
2577	7702	5506		JMP I ZLADR
2600	7703	4305		JMS DOTAPE
2601	7704	5506		JMP I ZLADR
2602	7705	0000	DOTAPE,	0
2603	7706	6141		LINC
2604	7707	0640		640
2605	7710	0714		714
2606	7711	0000	TPWORD,	0
2607	7712	0002		2
2610	7713	2311		ISZ TPWORD
2611	7714	1104		TAD KM200Z
2612	7715	3360		DCA CTR
2613	7716	1107		TAD KZ377
2614	7717	3015		DCA 15
2615	7720	0000	WHERE,	0
2616	7721	5705		JMP I DOTAPE
2617	7722	6201	WRIDUN,	CDF 0
2620	7723	3415		DCA I 15
2621	7724	3415		DCA I 15
2622	7725	2360		ISZ CTR
2623	7726	5323		JMP .-3
2624	7727	4305		JMS DOTAPE
2625	7730	7340		STA CLL
2626	7731	1311		TAD TPWORD
2627	7732	7006		RTL
2630	7733	7006		RTL
2631	7734	3311		DCA TPWORD
2632	7735	1111		TAD M3Z
2633	7736	3360		DCA CTR
2634	7737	4756	LOOPW1,	JMS I KCRLF
2635	7740	1311		TAD TPWORD
2636	7741	7006		RTL
2637	7742	7004		RAL
2640	7743	3311		DCA TPWORD
2641	7744	1311		TAD TPWORD
2642	7745	0036		AND KZ7
2643	7746	1110		TAD KZ260
2644	7747	4435		JMS I TYPEZ
2645	7750	2360		ISZ CTR
2646	7751	5340		JMP LOOPW1+1
2647	7752	7300		CLA CLL

```

2650      7753  6201      CDF 0
2651      7754  5755      JMP I .+1
2652      7755  6511      START
2653      7756  6247      KCRLF, CRLFS
2654      7757  5000      KK5K, 5000
2655      7760  0000      CTR, 0
2656      7761  0000      AAAEND, 0
2657                                     /USER MODIFICATION STARTS HERE
2660      DECIMAL
2661      NCHAN=1           /=NUM OF ANALOG
2662                                     /INPUTS
2663      POINTS=1000       /=NUM OF POINTS
2664                                     /PER CHAN
2665      BLKLEN=POINTS+POINTS+1
2666      LNBUF=POINTS      /=POINTS*NCHAN
2667      JLIST=7+1        /=7 WORDS PER
2670                                     /JOB+EOL WORD
2671      CHLEN=NCHAN+NCHAN+1
2672                                     /2 WDS FOR EACH
2673                                     /CHAN+EOL
2674      CHLIST=LOCORE+JLIST-1
2675      ADBUF=CHLEN+CHLIST
2676                                     /BUFFER STARTS
2677                                     /AFT CH LIST
2700      *MEMTOT          /NUM OF ADD
2701      0020  0000      0           /4K STACKS
2702      0021  5744      HICORE-LOCORE-4
2703      /JOB LIST
2704      OCTAL
2705      *LOCORE
2706      0230  0040      0040          /0(1) CHAN(5)
2707                                     /1(1) CHORD(5)
2710      0231  0400      AVG           /TYPE(8)
2711      0232  6030      -POINTS        /-NUM OF DATA
2712                                     /POINTS IN BLK
2713      0233  6201      6201          /62N1=CDF N
2714                                     /=CDF 0 (N=0)
2715      ADCAL0=ADBUF+LNBUF
2716                                     /LOC-1 OF 1ST
2717                                     /POINT
2720      0234  2212      ADCAL0        /IS RIGHT AFTER
2721                                     /ADC BUFFER
2722      0235  0200      0200          /DELTA(8)
2723                                     /YS(4)
2724      0236  0000      0           /X0(6),Y0(6)
2725      0237  0000      0           /END OF LIST
2726                                     /CHANNEL
2727                                     /DISPLAY LIST
2730      0240  0200      0200          /DELTA(8)
2731                                     /YS(4)
2732      0241  0000      0           /X0(6),Y0(6)
2733      0242  0000      0
2734      *ADCAL0+BLKLEN+1
2735      6134  0000      0           /END OF BLOCK
2736      /USER MODIFICATION ENDS HERE
2737      MEMTOT=20
2740      *ADJLIS
2741      0023  0227      LOCORE-1
2742      *ADCHNL
2743      0024  0237      CHLIST
2744      *SMASK           /SYNC ON
2745      0025  0040      0040          /CLOCK INPUT 1
2746      *ASI

```

2747	0037	7777	-1	
2750			*KMODE	
2751	0042	1500	1500	/KW12 CNTRL REG
2752				/400KHZ
2753				/MODE 101
2754	0043	0260	0260	/KW12 ENAB REG
2755				/SYNC ON
2756				/INPUT 1
2757			DECIMAL	
2760			*SAMA	
2761	0044	6030	-POINTS	
2762			*NCHA	
2763	0045	0001	NCHAN	
2764			*ADBUFA	
2765	0047	0242	ADBUF	
2766			*0042	
2767	0052	7777	-NCHAN	/-NO. OF CHANS
2770	0053	7777	-NCHAN	/-NO. OF CHANS
2771			*KBLA	
2772	0060	6030	-LNBUF	/-POINTS*NCHAN
2773			/PDP 12 SIGNAL AVERAGER	
2774			/WITH MTP SR	
2775			/9 APR 70	

0000 ERRORS

AAAEND 7761
ADBUF 0242
ADBUFA 0047
ADCAL0 2212
ADCHNL 0024
ADDUN 6275
ADINIT 6574
ADJLIS 0023
ADLOOP 7507
ADMESS 7363
ADONE 6377
ADRCLK 6755
ARGDUN 7646
ARITH0 0167
ARITH1 0170
ARITH2 0171
ARITH3 0172
ARITH4 0173
ARITH5 0174
ASAVE 0066
ASI 0037
ASWIT 0162
AVG 0400
AXPNTR 0013
BLA 0146
BLKCNS 7147
BLKCNT 0071
BLKFIN 7165
BLKLEN 3721
BLKOK 7166
BPNT 0146
BRAN 0132
BRANS 7536
BRLOOP 7542
BSAVE 0147
BUFA 0010
CALEND 6365
-

CALJOB 6302
CALMOR 6355
CALNXT 6321
CALXM 6324
CHCNT 0053
CHLEN 0003
CHLIST 0237
CHNSET 7512
CLKCNT 0051
CLKINT 7465
CLKMOD 0043
CNBLK 7144
CRCNT 6273
CRLFS 6247
CRLFX 6756
CTR 7760
CTRLA 6521
CTRLD 6503
CTRLQ 6506
CTRLR 6576
CTRLZ 6531
CYCLE 6605
DADD 0134
DADDS 7557
DBLAC 0170
DBLARG 0173
DECTYP 6200
DECTYX 6757
DELTAX 7062
DISP 0075
DISPS 7066
DISRTN 7077
DISWIT 6637
DOTAPE 7705
DVLOOP 7263
GDIGIT 6230
GETPNT 0011
GLOOP 6234
HERE 7674
HICORE 6200
IKBRAS 6401
IKBRAX 6760
IKBRAY 7065
IKCONT 6424
IKCRLF 6441
IKEXIT 6430
IKLIST 7572
IKP 6445
IKT 6454
IKTGO 6476
IKV 6431
IKXC 6435
INDATA 7662
INDATZ 0033
INKBD 7366
INKBDX 7533
INTERX 0065
INTOUT 7404
INTTY 7372
IXPNT 0073
IXPNTS 7170
JCELLS 6761
JCELLX 6563

JGET	0070
JGETS	7114
JLIST	0010
JPNT	0013
KBDBUF	0175
KBDCBK	6627
KBLA	0060
KCDF0	7672
KCHCNT	0052
KCR	0126
KCRLF	7756
KK17	7146
KK37	7145
KK5K	7757
KLUG	0135
KMCTRL	7364
KMD010	6261
KMD1K	6257
KMD100	6260
KMODE	0042
KM0001	0115
KM0004	0121
KM0006	7064
KM0027	0117
KM0043	0127
KM005	0054
KM200Z	0104
KM3720	0064
KPOINT	7210
KSAVE	0147
KSYTIM	0027
KZ260	0110
KZ377	0107
KZ7	0036
K0002	0114
K0003	0113
K0004	0112
K0007	0116
K0015	6262
K0017	7055
K0037	7012
K0040	0130
K0062	0063
K0100	0055
K0110	7534
K017	7243
K0212	6570
K0215	0061
K0240	6264
K0257	6263
K0276	6364
K0304	6566
K0310	6572
K0314	6571
K0316	6567
K0322	0062
K0377	0120
K201	6573
K212	6266
K44	0124
LADDS	7412
LADZ	6362
LASAVE	0221

-

LDIS	0076
LIRETN	0207
LJUMP	0226
LLSAVE	0227
LNBUF	1750
LOCORE	0230
LOOPW	7626
LOOPW1	7737
LSAM	7522
LSAVE	0067
LXCORD	0007
MEMTOT	0020
MONITR	7607
MTH	7346
MTW	7344
MTXMRK	0131
M0005	7244
M3Z	0111
NCHA	0045
NCHAN	0001
NDIGIT	0146
NPOINT	7177
NSAVE	0154
NSHFT	0166
NSWEP	0026
NSWPS	0165
NTOUT	7365
OCSORT	0130
PLTDLY	0050
POINTS	1750
PROMRK	0123
PSWIT	0163
PUTPNT	0012
RDKBDS	7211
RDKBOX	6562
RDLOOP	7213
RDONE	7240
RSTART	6636
SAMA	0044
SD	1000
SDIS	0074
SDISS	7000
SETPNS	7200
SETPNT	0072
SHCNT	0146
SHFR	0170
SHFT	0133
SHFTS	7272
SHLEFT	7317
SHRIHT	7303
SMASK	0025
SGADS	7433
SGADZ	6363
SQDONE	7456
SQLOOP	7444
START	6511
STORMA	7474
STORWC	7472
STWZ	0105
SYTIM	0147
TADC	0150
TBUFAD	0153
TCALS	7245

TCALX 6565
TCDF 0146
TDSKIP 0151
TEMP01 0146
TEMP02 0147
TEMP03 0150
TEMP04 0151
TEMP05 0152
TEMP06 0153
TEMP07 0154
TEMP10 0155
TEMP11 0156
TEMP12 0157
TEMP13 0160
TEMP14 0161
TEMP15 0162
TEMP16 0163
TEMP17 0164
TEMP20 0165
TEMP21 0166
TEMP22 0167
TEMP23 0170
TEMP24 0171
TEMP25 0172
TEMP26 0173
TEMP27 0174
TEMP30 0175
TEMP31 0176
TEMP32 0177
THERE 7700
TJTYPE 0152
TLAD 0146
TMCH 6267
TMCHX 7362
TMESS 7331
TMESSX 6564
TMESSZ 0034
TMLOOP 7335
TMPNTR 0017
TNJOB 0155
TNMAX 0157
TNMIN 0156
TORD 0150
TPWORD 7711
TRN 1400
TSAVE 0146
TSQAD 0147
TSWIT 0160
TTYBUF 0176
TTYFLG 0177
TTYLST 0122
TWO 7326
TXMESS 6265
TXMRK 0125
TYPES 7342
TYPEX 6274
TYPEZ 0035
TZCELL 0147
TZCNT 0146
VILOOP 6747
VINEXT 6743
VINPUT 6740
VJDIS 6727

VJSTAR 6644
VJXM 6677
VJXMN 6656
VSW 0161
WDUNZ 0031
WHERE 7720
WLOOP 7501
WRIDUN 7722
WRITE 7624
WRIZ 0032
WSWITZ 0030
XMASK 0164
YS 7060
YZ 7061
ZCYCLE 0056
ZLADR 0106
ZNXT 6535
ZXM 6550
ZZ 7535
ZZ1 7004
ZZ2 7047

