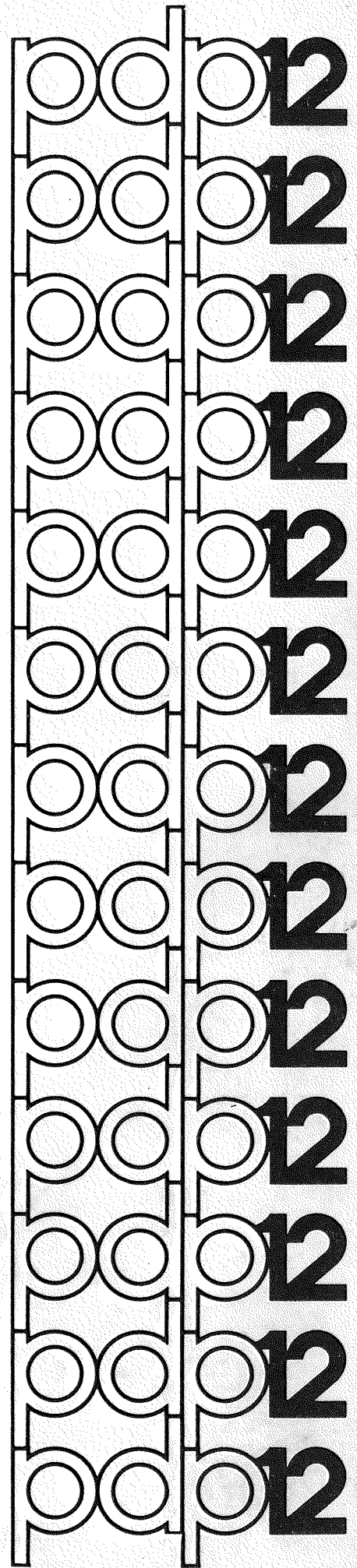
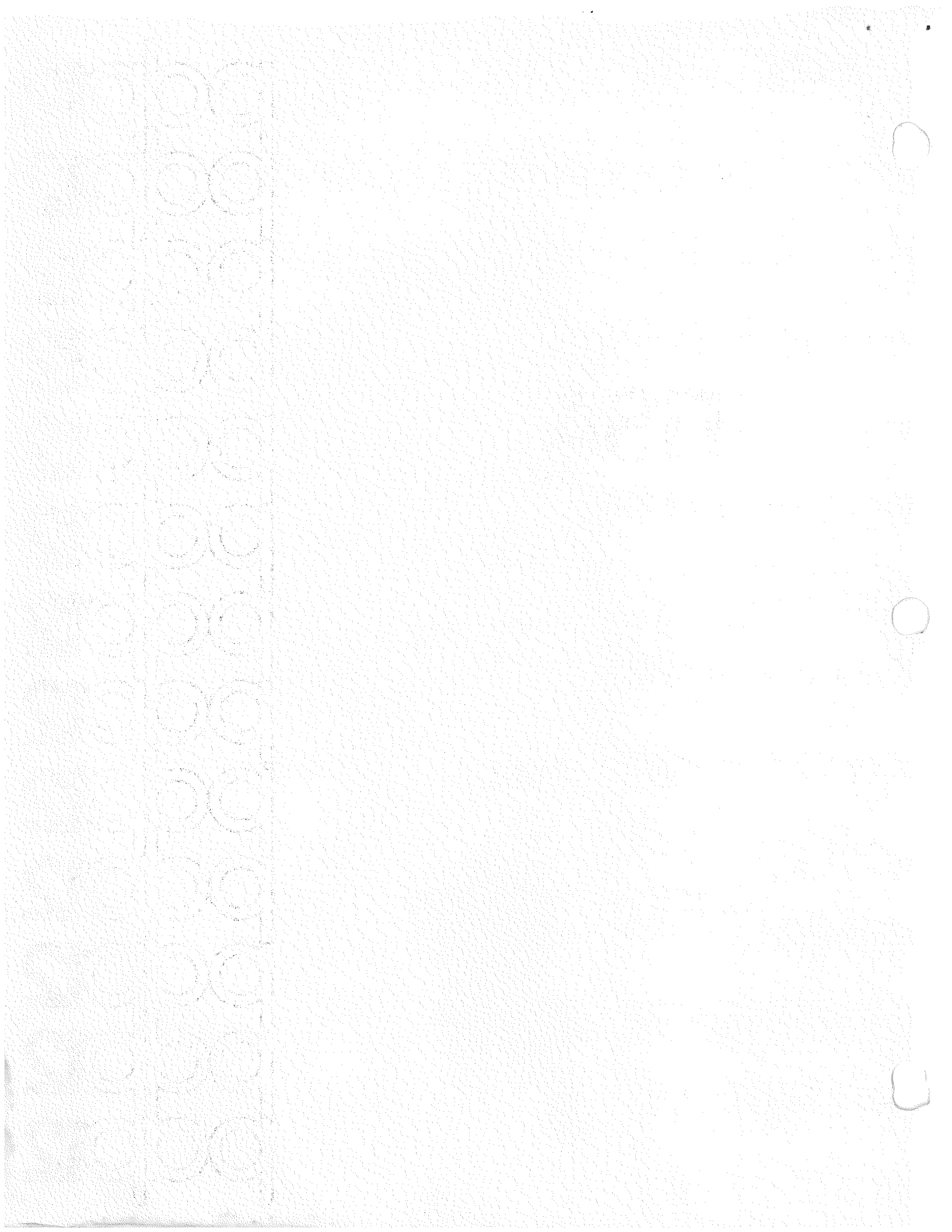


digital

# TISA





DEC-12-UW3A-D  
June, 1970

Copyright © 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	PDP
FLIP CHIP	FOCAL
DIGITAL	COMPUTER LAB

The equipment described herein is covered by patents and patents pending.

For additional copies order DEC-12-UW3A-D from Program Library, Digital Equipment Corporation, 146 Main Street, Maynard, Mass. 01754 Price \$2.00



## TABLE OF CONTENTS

1.0	INTRODUCTION	1
2.0	MINIMUM REQUIREMENTS	1
3.0	OPERATING PROCEDURE	1
3.1	Loading TISA	1
3.2	Operating Modes	2
	Sense Switch $\emptyset$ Options	2
3.4	Knob Options	3
4.0	SETUP MODE	3
4.1	Using Setup Mode	3
4.2	Message 1	4
4.3	Message 2	5
4.4	Message 3	7
5.0	HELP FRAME AND KEYBOARD COMMANDS	8
5.1	DIAL	8
5.2	Halt All Instruments	8
5.3	Halt Instrument n	9
5.4	Polarize Display	9
5.5	Freeze	10
5.6	Go Instrument n	10
5.7	Setup Mode	10
5.8	Write	10
5.9	CATACAL	12
5.10	MAGSPY	12
5.11	Load	12
5.12	Polarize	12
5.13	Freeze	12
6.0	TELETYPE USAGE	12
7.0	DATA ACQUISITION	13
8.0	PIN ASSIGNMENTS	14
9.0	DATA BUFFER DISPLAY	16
9.1	Moving Window	16
9.2	Cursor Display	17
9.3	X-Y Coordinates	18
10.0	INSTRUMENT INTERFACE	20
11.0	GLOSSARY	23
Appendix A	Internal Description	A-1
Appendix B	Flowcharts	B-1



## 1.0 INTRODUCTION

TISA, the Time Independent Spectrum Acquisition program, acquires asynchronous data from interfaced instruments, displays that data on the scope and stores it on LINCTape. TISA is compatible with any instrument that transmits X-Y data at rates that do not exceed 2 milliseconds/point and is capable of acquiring data simultaneously from five instruments. The number of data points that TISA can accept depends on the core size of the PDP-12A being used; the maximum number is 512 points for a 4K machine, 4608 for an 8K machine, and 29,184 for a 32K machine.

The external asynchronous instrument must be interfaced to the PDP-12 by either two potentiometers or one shaft encoder and one potentiometer, such that one input transmits X-axis data (independent variable), and the other transmits Y-axis data (dependent variable). For a given instrument TISA will generate the function  $Y=f(X)$ . Thus, a typical application for TISA is generating a series of absorption spectra from infra-red spectrophotometers.

Concepts and terminology peculiar to this manual are defined in the glossary (refer to section 11).

## 2.0 MINIMUM REQUIREMENTS

- a. PDP-12A with 4K of core<sup>1</sup> and KW12A clock.
- b. LAP6-DIAL<sup>2</sup> (version 2) tape with the TISA program.

## 3.0 OPERATING PROCEDURE

### 3.1 Loading TISA

TISA is a "load and go" program and is called from tape by the DIAL command

→LO TISA, unit)

where unit is the tape unit with the TISA tape. The initial starting address is 4001; the restart address is 4020.

To obtain correct results, the X-Y inputs for an interfaced instrument must be connected as described in section 8.0, PIN ASSIGNMENTS.

---

<sup>1</sup>For a data buffer of reasonable length, 8K of core is suggested.

<sup>2</sup>LAP6-DIAL is hereafter referred to as DIAL.

### 3.2 Operating Modes

TISA has three modes of operation:

- a. Setup mode - experiment parameters are defined
- b. A/D mode - data is being acquired
- c. Pause mode - no data is being acquired

When setup mode is terminated, TISA automatically enters pause mode and the data buffer display is activated. Any of the available commands may be issued at this time (refer to section 5.0).

### 3.3 Sense Switch $\emptyset$ Options

Sense Switch  $\emptyset$  (SSW $\emptyset$ ) is used in all three operating modes.

In the setup mode, its settings and effects are:

<u>Value</u>	<u>Effect</u>
$\emptyset$	none
1	erase <u>all</u> previous parameter input and initialize setup mode

Note that SSW $\emptyset$  is active during all of setup mode and that all user input is ignored when SSW $\emptyset$ =1. The only function of SSW $\emptyset$  during setup mode is to allow the user to restart parameter input from the beginning if he discovers an input error from which he cannot recover. During setup mode, raising and lowering SSW $\emptyset$  performs the same function as the following restart procedure, which may also be used:

1. Raise and lower the STOP switch
2. Set the mode switch to LINC
3. Depress the I/O PRESET switch
4. Depress the START 2 $\emptyset$  switch

During pause and A/D modes, the SSW $\emptyset$  settings and effects are

<u>Value</u>	<u>Effect</u>
$\emptyset$	display data buffer
1	display HELP frame

Raising SSW $\emptyset$  during A/D mode has no effect on data acquisition.

### 3.4 Knob Options

Knobs 0 and 4 provide flexibility for user control of the data buffer display.

Knob Movement		Effect
	KNOB 0	
Clockwise from center position		Move window forward; the farther away from center position, the faster the motion.
Counterclockwise from center position		Move window backward; the farther away from center position, the faster the motion.
Center position		Freeze window.
	KNOB 4	
Clockwise		Move cursor toward the right side of the scope.
Counterclockwise		Move cursor toward the left side of the scope.

### 4.0 SETUP MODE

#### 4.1 Using Setup Mode

Setup mode allows the user to define his particular experimental parameters, such as number of instruments, sampling frequency, etc., by displaying a series of scope messages. The user responds by typing the appropriate characters on the keyboard.

In setup mode, all scope messages are presented using the QANDA subroutine; the user must be familiar with the operating procedures of QANDA. Refer to DEC-12-FISA-D for a detailed explanation of QANDA. The following explanation deals only with those conventions that are peculiar to TISA.

Setup mode is initialized by any of the following:

1. Loading TISA from DIAL (refer to section 3.1)
2. Raising SSW0 during setup mode (refer to section 3.3)
3. Typing "S" during pause mode (refer to section 5.7)
4. Restarting at 4020 (refer to section 3.3)

When responding to any of the following messages, an unfilled blank is interpreted as a zero; leading zeros are ignored. Thus, a response of 1 \_ \_

is identical to a response of 001, \_ \_ \_ \_ is 0000. Note that all messages must be answered and terminated according to QANDA conventions except where noted. TISA does not interrogate any user reply until a terminator is typed.

#### 4.2 Message 1

Initializing setup mode clears the data buffers (fills them with zeros) and displays the following message:

```
PARAMETER INPUT
DO' _
1  LINCTAPE
2  KEYBOARD
```

Be usre that SSW0=0 now or all further keyboard input will be ignored. TISA alerts the user that SSW0=1 by causing message 1 to flicker.

Type 1 to select option 1, LINCTape, or type 2 to select option 2, keyboard. A response other than 1 or 2 and a terminator will redisplay the message.

Option 1 allows the user to recall from LINCTape a parameter list for a given experiment that was previously saved using the WRITE command in the HELP frame (refer to section 5.8). If option 1 is chosen, the following message is displayed:

```
READ TBLK' _ _ _
UNIT  _
```

Type the tape block and unit numbers that contain the desired parameter list, followed by a terminator. Only octal values are legal; nonoctal input will redisplay the read message. The user must know where his parameter list is stored; TISA has no file handling capabilities. After the tape block and unit have been specified, TISA tries to read in that parameter list. If the requested tape block contains a previously defined parameter list, setup mode is terminated and pause mode is entered. If the read message is redisplayed after TISA has read in the requested tape block, an illegal TBLK, i.e., one that does not contain a parameter list, has been specified. It is possible to request an illegal tape block that will not be caught. Because TISA cannot check for all possible errors, running TISA under such conditions could produce weird results. The user is cautioned to know the location of a stored parameter list.

Option 2 requires parameter input from the keyboard and will advance to message 2 for definition of each instrument.

#### 4.3 Message 2

Message 2 requests information defining the parameters for a particular instrument. The message has the following format:

```
INSTRUMENT K IS1 _  
(1 = analog, 2 = encoded)  
  
STARTING POINT= _ _ _ _ _  
TOTAL POINTS= _ _ _ _ _  
SAMPLING FREQUENCY= _ _
```

TISA will support up to five instruments, each of which must be defined in message 2. In the above display, K is the number of the instrument currently being defined and is always in the range 0-4. For example, the first instrument defined in message 2 would be displayed as:

```
INSTRUMENT 0 IS1 _
```

Four error conditions described below may be detected and noted by TISA when the values for message 2 are supplied.

All input to message 2 must be decimal; non-decimal input for any of the four responses will redisplay message 2. If TISA first detects any of the error conditions described below, the appropriate error message is displayed. An error in defining an instrument has no effect on the parameter input for a previously defined instrument; thus, if instruments 0 and 1 have been defined successfully but an error, such as non-decimal input, occurs in defining instrument 2, message 2 is redisplayed for instrument 2 and the parameter definitions for instruments 0 and 1 remain unchanged. Note that if an error in the definition of instrument 1 is detected by the user while defining instrument 2, it can be corrected only by raising SSW0 and starting with message 1 again.

The only legal values for the first response in message 2 are 1 for an analog instrument or 2 for an encoded instrument. Any other response will redisplay message 2.

TISA will support a maximum of three encoded instruments. If an attempt is made to define more than three instruments, the message

ERR ENC

is displayed and message 2 is redisplayed for the offending instrument when the error message is terminated.

The response for STARTING POINT can be any decimal number up to five digits long, i.e. 0-99999, and is the value of the initial X coordinate in the data buffer display routines (see DATA BUFFER DISPLAY, section 9.0).

TOTAL POINTS is the number of points that TISA will acquire and store in its data buffers, as distinguished from a detected point as described in SAMPLING FREQUENCY below.

The response for TOTAL POINTS for any instrument must be a number greater than 0 and less than or equal to the number of points left in the data buffer, which is the maximum number of points available minus the number that has been allotted to previously defined instruments. The maximum number of points available depends upon the amount of core in the system configuration, according to the following scheme:

<u>AMOUNT OF PDP-12 CORE</u>	<u>MAXIMUM NUMBER OF DATA POINTS AVAILABLE</u>
4K	512
8K	4608
12K	8704
16K	12800
20K	16896
24K	20992
28K	25088
32K	29184

Thus, if instruments 0 and 1 are defined as having 608 and 1000 points, respectively, for an 8K machine, the maximum number of points available for instrument 2 is 4608 minus 608 for instrument 0 and 1000 for instrument 1, or 3000 points for instrument 2.

If the user requests no points, the error message

ERR 0 PTS

is displayed.

If more points are requested than there are left, i.e. the capacity of the data buffer is exceeded, the error message

ERR BUF FULL

is displayed.

Terminating either error message reinitializes message 2 for the offending instrument.

The SAMPLING FREQUENCY determines the density or resolution of the spectrum. The X axis input is considered as a pulse. For an analog device a pulse is a one bit increment of an analog channel; for an encoded device a pulse is an interrupt on one of the KW12A clock channels. A point is said to be detected on the Y axis whenever TISA receives a pulse on the X axis. The ratio of detected points to acquired points is defined as the SAMPLING FREQUENCY. If the user defines TOTAL POINTS to be 10000 and the SAMPLING FREQUENCY to be 5, TISA will detect 50000 points and every fifth detected point will be an acquired point. SAMPLING FREQUENCIES of 0 or 1 mean acquire every detected point, a SAMPLING FREQUENCY of 2 means acquire every second detected point, etc.

The SAMPLING FREQUENCY must be a number from 0 to 10. Requested sampling frequencies outside of this range display the error message

ERR FREQ

Terminate this error message; message 2 is reinitialized.

#### 4.4 Message 3

When an instrument's parameter definition has been successfully completed in message 2, TISA displays the following message:

```
DO!_
1  MORE INSTRUMENTS
2  START
```

Type 1 to redisplay message 2 to define more instruments. TISA will redisplay message 2 with the instrument identifier (the K of message 2) incremented by one. Each instrument is defined in turn in this manner until option 2 of message 3 is selected. Type 2 if all the instruments in this experiment have been defined. Pause mode is entered immediately after typing 2. Terminating message 3 with a response other than 1 or 2 is illegal and redispays message 3.

If option 1 is selected, there are two possible error messages that may be displayed. If the user attempts to define more than five instruments, the error message

ERR INST

is displayed. If the user attempts to define additional instruments when all the data storage has been allotted to previously defined instruments, the following error message is displayed:

ERR BUF FULL

Terminate either error message display; message 3 is then redisplayed.

## 5.0 HELP FRAME AND KEYBOARD COMMANDS

When the user is in either pause or A/D mode and  $SSW\theta=1$ , a HELP frame is displayed that briefly summarizes all keyboard commands available to the user in either of these modes. The HELP frame display is:

D	A/D	DIAL MODE	
		CTRL/H	HALT ALL INSTRUMENTS
		HN	HALT INSTRUMENT N
		P	POLARITY
		F	FREEZE
		PAUSE MODE	
		GN	GO INSTRUMENT N
		S	SETUP
		W	WRITE TAPE
		C	CATACAL
		M	MAGSPY
		L	LOADER
		P	POLARITY
		F	FREEZE

All commands are issued by typing the letter(s) and then their action is implemented immediately.

### 5.1 DIAL

The D command is active in either mode and is used to restart DIAL. Refer to the LAP6-DIAL Programmer's Reference Manual, DEC-12-SE2B-D.

### 5.2 Halt All Instruments

Typing CTRL/H<sup>1</sup> when in A/D mode directs TISA to halt all instruments; data acquisition is terminated on all instruments and TISA enters pause mode. Whenever TISA enters pause mode, the message:

PAUSE MODE

is printed on the Teletype.

<sup>1</sup>CTRL/H is typed by pressing down the CTRL (control) key and simultaneously typing H, in the same manner as using the SHIFT key.

### 5.3 Halt Instrument n

HN is a two character command that allows the user to halt a given instrument (put it into pause mode) without affecting the mode status of the other defined instruments. The first character of the command is H (for halt); the second character, n, is a number in the range 0-4 and defines the number assigned to the instrument in message 2. Thus, the command H2 means halt instrument 2. Any number of intervening characters may be typed between H and N, the numeric argument, without interfering with the execution of the command HN, except for the character G. Thus, the command HAAA777BBB2 is identical to the command H2 because the nine intervening characters are illegal and ignored by TISA. The command H666PPP???2 directs TISA to halt instrument 2, but it is not identical to the command H2 because it executes the polarity command P (refer to section 5.4) three times before instrument 2 is halted. An HN command is acknowledged by the following message printed on the Teletype:

HALT INSTRUMENT N

where N is the user supplied value 0-4. If the HN command is typed for an instrument that has not been defined by the user or for an instrument that is in pause mode, it is ignored.

### 5.4 Polarize Display

Typing P polarizes or inverts the entire data buffer display. For example:



The P command does not destroy the integrity of the data in the data buffer but merely inverts it before it is displayed on the scope. If the data buffer is polarized when one P command has already been issued, the display is restored to its normal form. If the P command causes the display to be inverted, TISA responds with the Teletype message:

INVERTED

If the P command causes the display to be normalized, TISA outputs the message:

NORMAL

The P command can be issued as often and whenever the user chooses.

### 5.5 Freeze

The F command directs TISA to freeze the moving window display (see DATA BUFFER DISPLAY, section 9.0), thereby isolating the display on a given portion of the data buffer. Typing F when the window is frozen directs TISA to restart window motion. TISA acknowledges the F command and indicates the action taken by typing FREEZE or MOTION, whichever is appropriate. The user may issue the F command as often as he likes.

The P and F commands are active in both pause and A/D modes.

### 5.6 Go Instrument n

GN is a two character command issued in the same manner as HN (refer to section 5.3), but performs the opposite function; GN puts an instrument in A/D mode. Remember that arguments H and G cannot be intermixed in a command. For example, typing H followed by G erases the first half of an HN command and directs TISA to wait for the second argument of a GN command. Thus, the commands HG2, HHHHGG2, and G2 are all identical. If instrument N is in A/D mode, TISA will ignore the command GN.

When TISA initiates A/D mode for a given instrument, the message

```
GO INSTRUMENT N
```

is printed on the Teletype, where N is the instrument number 0-4. Whenever TISA enters A/D mode it outputs the following message on the Teletype:

```
A/D MODE
```

### 5.7 Setup Mode

Typing the key S causes TISA to enter setup mode so a new parameter list can be input. The contents of the data buffer are not preserved.

### 5.8 Write

The W command outputs the data buffers to LINctape. TISA acknowledges this command by displaying the following message:

```
EXPERIMENT NAME  
-----
```

The user can name his experiment with any combination of up to eight characters; input acceptable to QANDA is legitimate here.

The Write command will generate 2+K TBLKs of output where K is the smallest number of tape blocks needed to store the portion of the data defined by the experiment. If the user had defined 1000 points of the data buffer area, then K is equal to four. The first two blocks contain the experiment name and the parameter list for the experiment just completed; they contain no data but enable the user to exercise the LINctape input option during setup mode (refer to section 4.2). The experiment name will appear in the beginning of the first block of output exactly as it is typed in the name message above. For example, if the first block of output is assigned the name

ABCD\_ \_ \_ \_

then this block when viewed via MAGSPY<sup>1</sup> will appear as follows:

---

A B C D \_ \_ \_ \_  
\$%ASJWBE\*996%RYSSI&456%&J

After the name message, TISA displays the following Write message:

```
WRITE XYZ (OCT) TBLKS
FIRST TBLK= ' _ _ _
UNIT _
```

where XYZ is the number of octal tape blocks that will be output and is the number 2+K described above. The user specifies in octal the starting TBLK and unit number; a non-octal response redisplay the name message. If the response to the Write message is legitimate, TISA commences to write the data buffer plus the two header blocks. While writing, the display is maintained; however, all commands are ignored until this operation is finished. When the interrupt (ION) light comes on, the operation is finished. If this parameter list just output is ever to be recalled, remember the first TBLK specified in the Write message because it is this block that option 1 of message 1 (refer to section 4.2) requires. (This block number may also be determined using the MAGSPY program.) Note that TISA considers TBLK 777 to be contiguous to TBLK 000. For example, if a request is made to write 20 TBLKs starting at TBLK 770, the first 10 TBLKs are written on 770-777 and the last 10 TBLKs on 0-7. Data should not be output in this manner, however, because of difficulties created for related software, such as MAGSPY<sup>1</sup>.

---

<sup>1</sup>MAGSPY is a program used to examine data stored on LINctape. Refer to DEC-12-UZSA-D and section 5.10.

## 5.9 CATACAL

Typing C exits TISA and loads the program CATACAL which is very useful for data manipulation. Refer to DEC-12-UW1A-D for a description of CATACAL.

## 5.10 MAGSPY

Typing M exits TISA and loads the program MAGSPY.

The CATACAL and MAGSPY commands both assume that a DIAL-V2 tape (not DIAL-MS) with the binary files CATACAL and MAGSPY are on unit  $\emptyset$ . If the binary files are not present, the user will find himself in DIAL with the NO message on the scope. If a DIAL-V2 tape is not on unit  $\emptyset$ , the results are unpredictable.

## 5.11 Load

TISA can load an arbitrary program by issuing the command L. A DIAL V2 tape must be on unit  $\emptyset$ . The following load message is displayed:

```
LOAD |  
UNIT | _ - - - - -
```

Type at least one character for the program name in acceptable QANDA characters and then type the unit where this program is located. No error checking is done for the unit argument; thus, if the user requests unit A, he gets unit 1, etc. If the requested binary is a file on the requested unit, TISA will load it. If the file does not exist, the user will find himself in DIAL with the NO message on the scope.

## 5.12 Polarize

The P command in pause mode is identical to the P command, Polarize, in A/D mode (refer to section 5.4).

## 5.13 Freeze

The F command in pause mode is identical to the F command, Freeze, in A/D mode (refer to section 5.5).

## 6.0 TELETYPE USAGE

TISA outputs messages to the Teletype to make the user aware of the present action. The following set of messages are in TISA's repertoire:

```

HALT INSTRUMENT N      (N=0-4)
GO INSTRUMENT N        (N=0-4)
MOTION
FREEZE
NORMAL
INVERTED
PAUSE MODE
A/D MODE
HANDS OFF

```

A backlog of several messages may occur because the Teletype is a relatively slow device for TISA's environment. TISA services legal keyboard commands immediately; the message(s) appropriate to the command(s), however, is typed afterward. If a command is issued while a message is being typed, the command is executed immediately and the message(s) appropriate to the command is put on TISA's backlog. TISA will never overload the Teletype but the user may do so by leaning on the P key. In this case, the integrity of the messages printed on the Teletype may be destroyed, but execution of the keyboard commands is maintained. If the Teletype becomes too overloaded with commands, the message HANDS OFF is printed.

#### 7.0 DATA ACQUISITION

TISA has the ability to acquire data simultaneously from five instruments. A given device must not transmit X axis data at rates that are faster than 2 milliseconds/point or data will be lost. If all instruments (maximum of 3) are defined as encoded, the optimum rate obeys the following scheme:

<u>NO. OF ENCODED INSTRUMENT</u>	<u>OPTIMUM RATE</u>
1	500 $\mu$ s
2	1000 $\mu$ s
3	1500 $\mu$ s

For analog devices, a one bit increment in the X axis potentiometer indicates to TISA that a point has been detected on the Y axis. For encoded devices, an interrupt on a KW12A clock channel (X axis input) indicates a detected point. The data that TISA acquires comes from the Y axis input. Each data point represents a 10 bit A/D conversion on one of the Y axis analog inputs, and is stored as a 10 bit single precision word. Each defined instrument is assigned a sector of contiguous core locations in the data buffer area, beginning at location 7000 of PDP-8 field 0 and continuing to the last location of the last PDP-8 field. The data sectors for each instrument are adjacent to one another; there are no data gaps between instrument data sectors. The data sector for instrument 0 always begins at the initial PDP-8 data buffer location (07000). For example, if the user defined TOTAL POINTS for

instruments 0, 1, and 2 to be 512, 2000, and 3 (1000, 3720, 3 octal), respectively, then the allocation in the data area for an 8K of core configuration is:

<u>INSTRUMENT</u>	<u>DATA BUFFER ALLOCATION</u>			
	<u>FROM</u>		<u>TO</u>	
0	0	7000	0	7777
1	1	0000	1	3717
2	1	3720	1	3722

(1 3723 to 1 7777 is unused)

TISA starts acquiring data from an instrument upon receiving a Go command (refer to section 5.6). No action is taken on an instrument until the first pulse is received from the X axis. TISA does not require the data to be input in a periodic or time dependent fashion; pulses can be days or milliseconds apart.

TISA will acquire data from an instrument until any one of the following conditions occurs:

1. the requested number (TOTAL POINTS) of points has been acquired
2. the user halts the instrument from the keyboard
3. the X axis potentiometer (analog instruments only) decreases 20 bits from its last value read

Conditions 1 and 3 represent internal instrument halts. If a halt message is printed on the Teletype and the halt was not requested by the user from the keyboard, then the instrument halt was internal. All three conditions cease data acquisition for the given instrument and an appropriate message is printed on the Teletype.

## 8.0 PIN ASSIGNMENTS

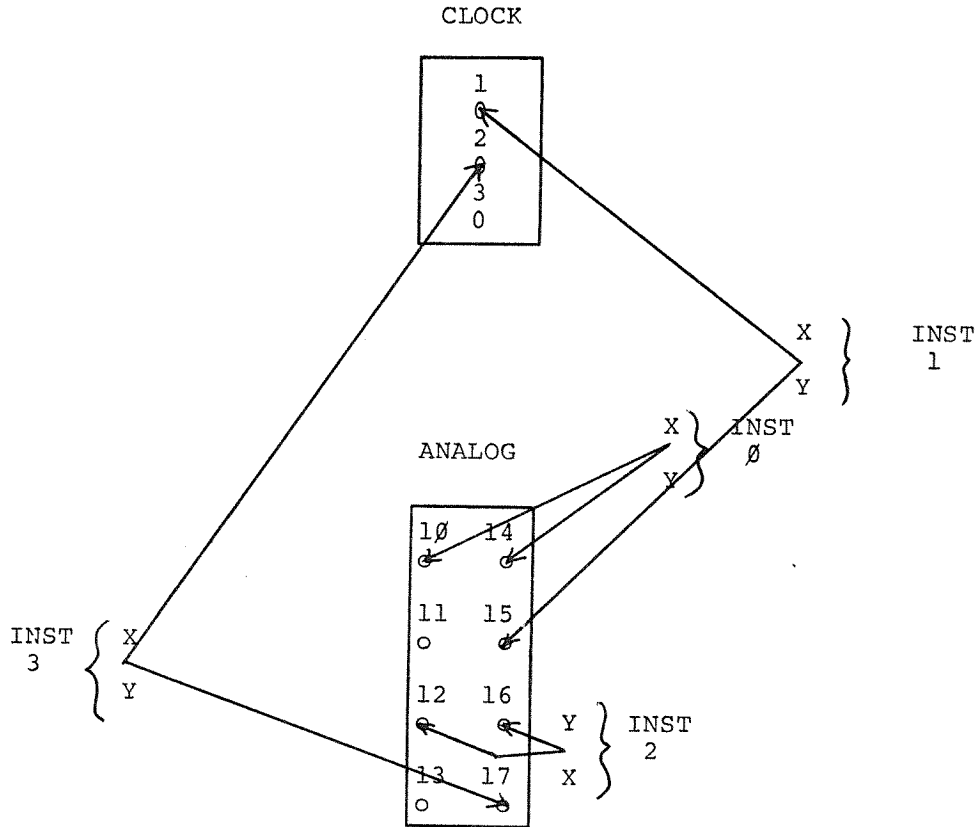
There is only one correct way to connect the X-Y inputs from an instrument to the PDP-12, regardless of the particular system configuration. Let instrument K be the Kth instrument defined during setup mode for  $K=0,1,2,3,4$  and let  $N_K$  be the number of analog instruments that were defined previous to instrument K. The X-Y inputs must then obey the following rules:

1. If instrument K is analog, its X axis input is connected to analog channel  $10+K$  for  $K<4$ . If  $K=4$ , the input is connected to analog channel 20.<sup>1</sup>

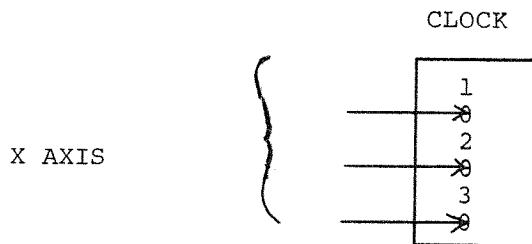
<sup>1</sup>The fifth channel requires the analog extension (channels 20-27) which is not standard PDP-12 hardware.

2. If instrument K is encoded, its X axis input is connected to clock channel  $1+K-N_K$ .
3. The Y axis input for instrument K is analog channel  $14+K$  for  $K < 4$ .<sup>1</sup> For  $K=4$ , the input is connected to analog channel 24.<sup>1</sup>

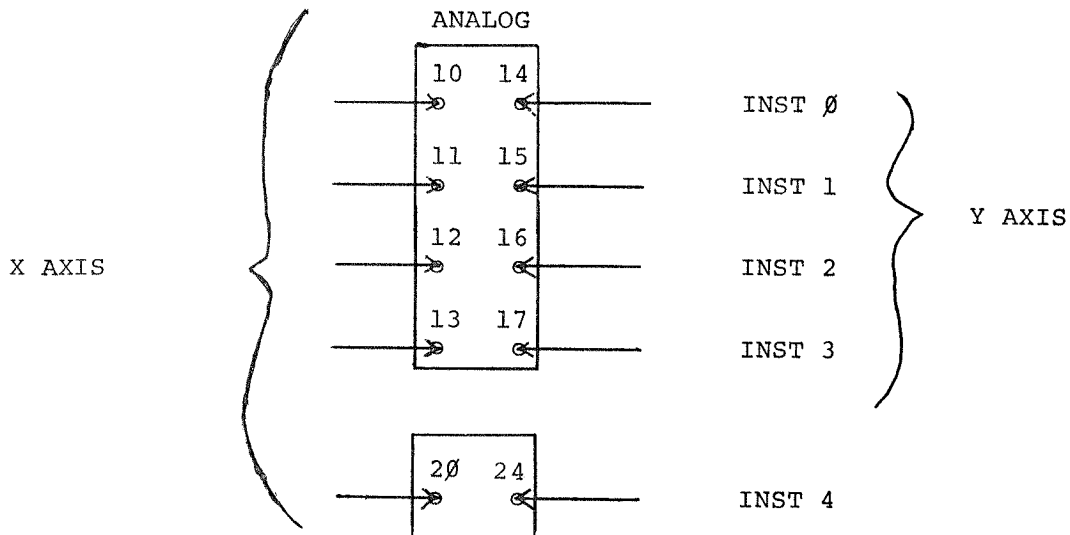
Consider, for example, the following configuration:



GENERAL CONFIGURATION



<sup>1</sup>The fifth channel requires the analog extension (channels 20-27) which is not standard PDP-12 hardware.



INST	INST TYPE	N <sub>K</sub>	X AXIS	Y AXIS
0	ANALOG	0	ANCH 10	ANCH 14
1	ENCODED	1	CLOCK 1	ANCH 15
2	ANALOG	1	ANCH 12	ANCH 16
3	ENCODED	2	CLOCK 2	ANCH 17

## 9.0 DATA BUFFER DISPLAY

The data buffer display has the three following features:

1. A moving window display controlled by A/D Knob 0 and the keyboard
2. A cursor superimposed on the data buffer display and controlled by A/D Knob 4
3. X-Y decimal coordinate display of cursor position

The 10 bit values in the data buffer are scaled to 9 bits prior to display to prevent scope "wrap-around."

### 9.1 Moving Window

Because the scope can display a maximum of 512 points at any one time and the data buffer has a maximum capacity of 29,184 points, the moving window display is used to view any 512 contiguous points of the data buffer, starting at any given point, simply by rotating A/D Knob 0. The moving window is very similar to that used in MAGSPY. Consider the data buffer as a ring

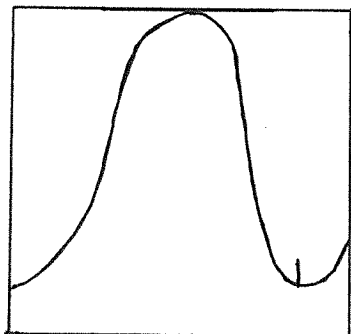
located behind the VR12 scope, with the initial and terminal data points being contiguous, that can be turned by a wheel, namely, A/D knob  $\emptyset$ ; the scope is the window. (The phrase moving window is actually a misnomer because it is the data buffer that moves.) When knob  $\emptyset$  is in its center position, the display is stationary.

When A/D knob  $\emptyset$  is moved clockwise from the center position, the data buffer (the ring) appears to move from right to left across the scope. The velocity of the moving window increases with the clockwise movement of A/D knob  $\emptyset$  from the center position. The exact opposite occurs for counterclockwise movement of A/D knob  $\emptyset$ . The data buffer appears to be moving from left to right across the scope. When the display is frozen under the keyboard command F, knob  $\emptyset$  has no effect. When TISA enters pause mode for the first time (after being loaded from LIN Ctape), the display is frozen and the leftmost scope point is the first point for instrument  $\emptyset$ .

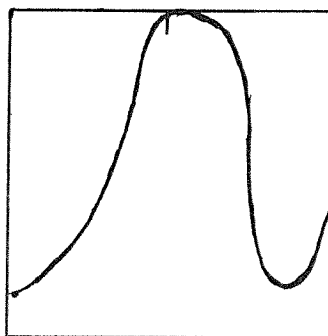
## 9.2 Cursor Display

A cursor is superimposed upon the data buffer display and can be positioned over any of the 512 points in the display by rotating A/D knob 4. If knob 4 is moved to its furthestmost counterclockwise (leftmost) position, the cursor will be imposed on the leftmost point of the scope display. Similarly, if knob 4 is positioned to its furthestmost clockwise (rightmost) position, the cursor will be imposed upon the rightmost scope data point. Intervening knob positions will yield intervening cursor positions. The cursor appears on the scope as a vertical line ( | ) and normally "sits" on top of the display point. However, for those cursor points that do not allow enough room at the top of the scope for the cursor to sit comfortably, the cursor is displayed beneath the cursor point as the following example shows.

CURSOR  
ABOVE



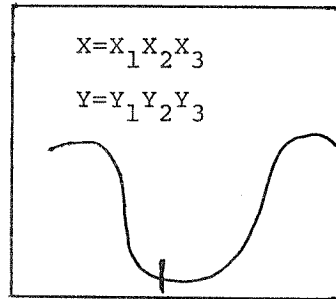
CURSOR  
BELOW



It should be noted that knob  $\emptyset$  has no control over the position of the cursor relative to the scope. If the window is frozen and the cursor is I inches from the left side of the scope, the cursor will remain fixed at I inches from the left of the scope if the window movement is started. The relationship of the cursor to the data buffer, however, changes because the data buffer is moving across the scope.

### 9.3 X-Y Coordinates

If the cursor is sitting on a defined portion of the data buffer, its X-Y coordinate position is displayed on the scope in the following manner:



Both X and Y are non-negative decimal integers; the range for X is  $\emptyset$ -391829; the range for Y is  $\emptyset$ -511. Leading zeros and the point  $\emptyset\emptyset\emptyset\emptyset$  are not displayed. If both the X and Y coordinates are  $\emptyset$ , the coordinates are displayed as:

X=  
Y=

If the cursor is positioned over an undefined portion of the data buffer, the X-Y coordinates are not displayed at all. The Y coordinate reads the amplitude (height) of the cursor; the value  $\emptyset$  is the bottom of the scope and the value 511 is the top of the scope.

The value of the X-coordinate depends upon the cursor's absolute position within the data buffer and the starting points and sampling frequencies assigned by the user during setup mode. Each instrument was assigned a sector of the data buffer after replying to TOTAL POINTS in message 2 (refer to section 4.3). When the STARTING POINT value was input, an initial X coordinate to the first data point of the data sector for that instrument was assigned.

Consider the case where the sampling frequency is 1, the STARTING POINT is  $9\emptyset\emptyset\emptyset$  for instrument K, and the cursor is positioned over the initial data point of the data sector for the instrument K. If the cursor is moved

from left to right with the window frozen and the cursor at the far left of the scope, the X coordinate will read successively:

9000, 9001, . . . . , 9009, 9010, . . . . , 9510, 9511

assuming TOTAL POINTS is greater than 511. Because the STARTING POINT (initial X coordinate) is arbitrary and data sectors are adjacent to one another in the data buffer, it is not unusual for the X coordinate to jump as the cursor crosses data sectors. Suppose the user defined TOTAL POINTS=1 for instruments 0, 1, and 2 and STARTING POINTS=9000, 5, and 47311, respectively. If the cursor is positioned over the first (and last) data point for instrument 0 and moved from left to right, the X coordinate will read respectively (and discontinuously):

9000, 5, 47311

As long as the cursor stays within a data sector and moves forward across it via A/D Knob 0 or 4 or both, the X coordinate will increase in value. The difference between any two X coordinates within a data sector for a given instrument is the SAMPLING FREQUENCY of that instrument. In the same case, if the SAMPLING FREQUENCY is three, then the successive X coordinate readings are:

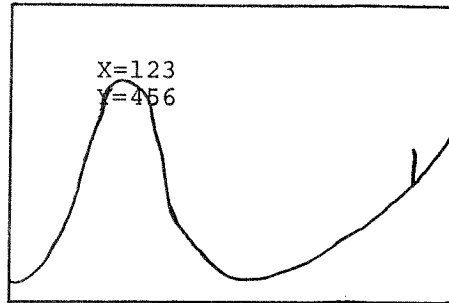
9000, 9003, . . . . , 9027, 9030, . . . . , 10530, 10533

For a SAMPLING FREQUENCY of 10, the readings are:

9000, 9010, . . . . , 9090, 9100, . . . . , 14100, 14110

With a SAMPLING FREQUENCY of 10, the data acquisition routines acquire 1 point for every 10 detected and they are stored as successive data points in the data buffer. This difference, therefore, is reflected in the X coordinate display. There is no X-Y display over undefined areas of the data buffer because there are no arguments (STARTING POINT, etc.) for the X-Y display routines to use. The X-Y coordinates are displayed on hardware channel 1, and the data buffer is displayed on hardware channel 2. To see both displays the channel select knob should be set to the position 1 & 2. In the event that the data buffer display and X-Y readout overlap, they can be viewed separately by selecting the appropriate hardware scope channel.

### OVERLAP CONDITION



### 10.0 INSTRUMENT INTERFACE

TISA plots the incoming data from each of up to five instruments as a dependent variable versus an independent variable. The dependent variable, for example the intensity at a given frequency for an infra-red spectrophotometer, must be an analog signal within the range of the analog-to-digital converter contained in the particular PDP-12. The standard voltage range is -1 volt to +1 volt.

The independent variable may be a monotonically increasing voltage as shown in figure 1 or a pulse train as shown in figure 2. Note that the fastest sampling rate on any and all instruments is 2 milliseconds per point. The maximum slope of an analog dependent variable can be found by the following formula:

$$\text{maximum slope} \cong \frac{X \text{ volts}}{10^3 \text{ samples}} \cdot \frac{1}{2 \text{ msec}}$$

where X = voltage range of particular analog-to-digital converter system

For the encoded or pulsed dependent variable the input waveform need only pass through the desired threshold whenever an analog sample is required. The desired trigger level is set by the threshold knob on the KW12A input panel. The threshold knob is not calibrated but the limits are approximately  $\pm 6$  volts and the potentiometer controlled by the threshold knob is approximately linear. The slope control defined as shown in figure 2 is also on the KW12A input panel.

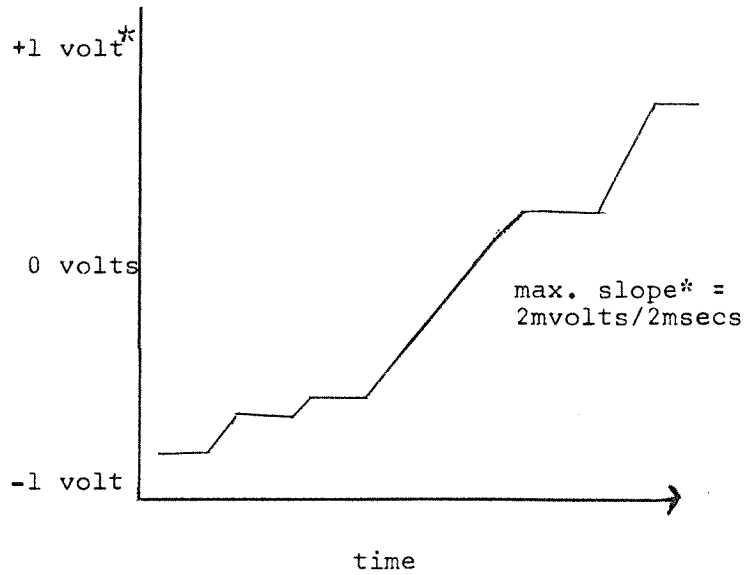
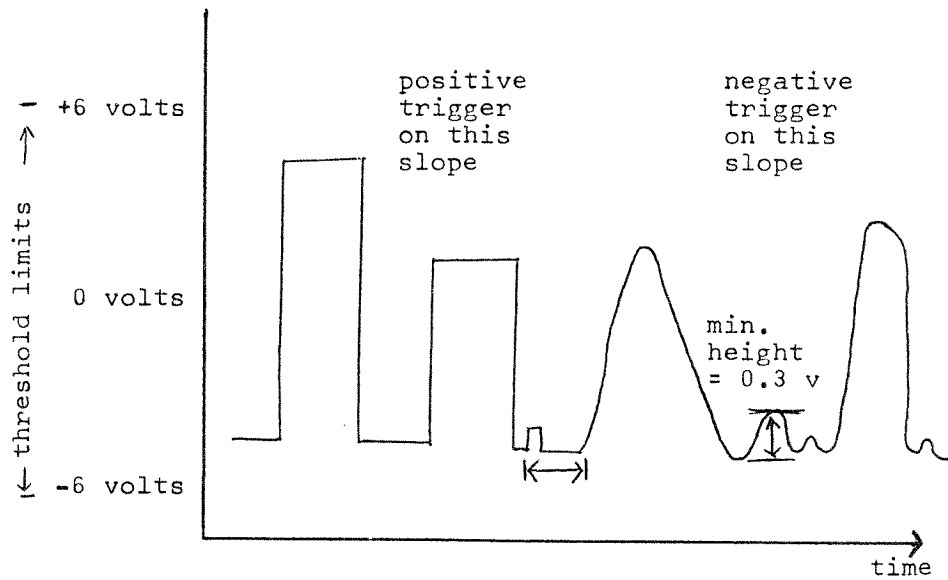


Figure 1 Monotonically Increasing Voltage  
 \*voltage range for standard analog to digital converter



minimum time between  
 triggering points = 2 msecs

Figure 2 Pulse Train

The setup procedure for instrument N, where N = 0, 1, 2, 3, 4, is described in section 8.0. The setup procedure details which input channels are to be used.

Typical analog independent variable sources are retransmitting pots or ramp generators while pulsed sources are typically shaft encoders or oscillators.

Several examples of typical interfaces were tried in the PDP-12 engineering laboratory. A Perkin Elmer model 621 infrared spectrophotometer was interfaced by placing a retransmitting pot on the pen drive gear and either a retransmitting pot or an incremental shaft encoder on the main chart drive gear. In the case of the Perkin Elmer instrument, the pickup devices were attached via extra chart drive gears supplied with the instrument. The potentiometers were Bourns 3400S series devices. The incremental shaft encoder was a Data Technology model #C25-100-40 with two one-hundred cycle/turn pulse outputs 90° out of phase<sup>1</sup>. The two outputs can be used to detect the direction of rotation and/or produce a 400 cycle/turn output. However, for this application, either of the outputs alone was sufficient.

---

<sup>1</sup>The specification of these part numbers in this manual does not constitute DIGITAL EQUIPMENT CORP. sanction or guarantee, but is intended as an aid for the user.

## 11.0 GLOSSARY

ACQUIRED POINT	an acquired point is one that has been retrieved from the Y axis of a defined instrument and stored in TISA's data buffers (see also SAMPLING FREQUENCY and DETECTED POINTS).
A/D MODE INSTRUMENT	an instrument is said to be in A/D mode (running) when TISA is accepting information from its X axis.
TISA	TISA is said to be in A/D mode when at least one of its defined instruments is in A/D mode.
ANALOG INSTRUMENT	one of two types of instruments that may be interfaced to TISA. An instrument is defined as analog if its X axis input runs from a potentiometer to a PDP-12A analog channel.
CURSOR	a vertical line (   ) superimposed upon the data buffer display and controlled by A/D knob 4.
CURSOR POINT	that point in the data buffer display upon which the cursor is sitting.
DATA BUFFER	the amount of core available on the PDP-12 that has been reserved for acquiring points. The data buffer is a function of core size and ranges from 512 to 29,184 words.
DATA POINT	a signed 10 bit single precision A/D conversion or word.
DATA SECTOR	a contiguous set of core locations in the data buffer that has been assigned to a defined instrument. A data sector can be anywhere in length from 1 to 29,184 core locations.
DEFINED DATA AREA	the portion of the data buffer that contains the data sectors of all defined instruments.
DEFINED INSTRUMENT	an instrument is defined if it has been assigned a set of parameters during setup mode.
DETECTED POINT	for a given instrument, a point on the Y axis is detected if TISA has received a pulse from the corresponding X axis.

ENCODED INSTRUMENT one of two types of instruments that may be interfaced to TISA. An instrument is defined as encoded if its X axis input runs from a shaft encoder to a KW12A clock channel.

HELP FRAME a message that can be displayed on the scope when TISA is in either pause or A/D mode which briefly summarizes the keyboard commands that are available to the user.

INSTRUMENT interfaced to TISA for data acquisition. Instruments are either analog or encoded, depending upon the manner in which they are interfaced.

INSTRUMENT K the Kth instrument defined during setup mode; K = 0,1,2,3,4.

INTERNAL HALT an instrument in A/D mode that is halted by a means other than by an HN or CTRL/H keyboard command is said to have been internally halted.

MODE INSTRUMENT the state of an instrument at any point in time or the relationship that it bears to TISA is called its mode. An instrument is either in A/D mode or pause mode. Undefined instruments are in pause mode by default.

TISA the state of TISA's operation or its relationship to the user at any given point in time is called the mode. TISA has three modes: A/D, pause, and setup.

PARAMETER a numeric value that is used to define an instrument, e.g. TOTAL POINTS.

PARAMETER LIST the set of all parameters for all defined instruments.

PAUSE MODE INSTRUMENT an instrument is in pause mode whenever TISA is not accepting information from its X axis.

TISA TISA is in pause mode whenever all of its defined instruments are in pause mode.

PULSE X axis information transmitted by a defined instrument. For an analog instrument, a pulse is a one bit increment on the X axis potentiometer. For an encoded instrument, it is an interrupt on one of the KW12A clock channels.

QANDA a PDP-12 subroutine that facilitates question and answer type of dialogue via the scope and keyboard. (Refer to DEC-12-FISA-D.)

SAMPLING FREQUENCY an instrument parameter that is the ratio of DETECTED POINTS to ACQUIRED POINTS. SAMPLING FREQUENCY determines an acquired spectrum's resolution.

SETUP MODE TISA is said to be in setup mode when the user is defining his instruments (messages 1-3).

SPECTRUM the sum total of an instrument's acquired points is called the spectrum of that instrument.

SSW0 Sense Switch 0.

STARTING POINT an instrument parameter that defines the initial X coordinate for the data buffer display of the first data point of the instrument's data sector.

TBLK an abbreviation for LINctape block.

TOTAL POINTS an instrument parameter which equals the total number of points that TISA will attempt to acquire when the instrument is in A/D mode. The length of an instrument's data sector is equal to its TOTAL POINTS.

X AXIS one of two of an instrument's inputs to TISA that are required to generate the instrument's spectrum. The X axis input is the independent variable.

Y AXIS one of two of an instrument's inputs to TISA that are required to generate the instrument's spectrum. The Y axis input is the dependent variable.



This appendix assumes that the reader is familiar with TISA's external characteristics as described in sections 1 through 11 of this manual.

- A.1 Interrupt Routines
- A.2 Data Acquisition Routines
- A.3 Teletype Routines
- A.4 Keyboard Routines
- A.5 LINCTape Routines
- A.7 Setup Routines
- A.8 Memory Check Routine
- A.9 Flow Charts
- A.10 Core Allocation



## A.1 INTERRUPT ROUTINES

The interrupt routines handle interrupts in both modes. In either mode, only the following central processor registers are preserved: AC, LINK, PC, IF, and DF. The entry and exit routines for P-mode and L-mode interrupts are different but the interrupt dispatching is common to both. Interrupts remain off until the routines exit. I.e., the interrupt routines can not be interrupted. An L-mode interrupt goes to a routine INTL which saves the AC and LINK, sets up a LINKmode jump at RTNJMP, an LDF instruction at RTNLDF and an LIF instruction at RTNLIF, and sets the interrupt mode switch (INTMOD) to 0. A P-mode interrupt saves the AC, link and sets the interrupt mode switch to 7777, before going to interrupt dispatch.

The interrupt dispatch routines always check three devices in the following order: clock, keyboard, and Teletype. If the clock flag is set, the routine dispatches a call to the routine TISA for data acquisition. If the keyboard flag is up, a character is read in and stored in location KBDBUF. If the Teletype flag is up, it is cleared, then location ADMESS is checked to see if a character needs to be typed (C(ADMESS)NE to 0 means type C(ADMESS)). If a character needs to be typed, it is; the location ADMESS is advanced by one; and the location TTYFLG is set to C(ADMESS) - non zero - to indicate that the Teletype is busy. If no characters need to be typed, then TTYFLG is set to 0 to indicate that the Teletype is not busy. When all devices have been either check or serviced, the AC and link are restored and the INTMOD switch is ISZed to implement one of the two following exits:

P-MODE	L-MODE
RMF	LINC
ION	DJR
JMP I 0	RTNLIF, 0
	RTNLDF, 0
	IOB
	ION
	RTNJMP, 0

## A.2 DATA ACQUISITION

When the data acquisition routines are called they check all five devices (even if some devices are not defined or not selected) because it takes only 14.4 microseconds to determine that a device is not selected. The first task of the data acquisition routines is to preserve the clock bits by executing:

```
CLSA
DCA STATUS
```

An analog device must be checked whenever there is a clock overflow, not because the device is ready but because it might be ready. The only way to determine a 1 bit increase of an X-axis analog channel is to read it periodically. The clock is set to overflow and causes an interrupt on overflow once every two milliseconds. Analog devices, therefore, are checked once every 2 milliseconds (optimum speed for analog devices under the worst circumstances). An encoded device is checked whenever its clock channel causes an interrupt indicating it is ready. Note that encoded devices can run at speeds much faster than 2 milliseconds. For instance, TISA could handle one encoded device and no others without loss of any data, if the rate of interrupt was faster than 1/2 millisecond/point. Because the optimum speed varies among all possible user defined configurations, the upper limit in the user's manual is a worst possible case restriction. There are only 4 bits in the clock status register that are of interest to the data acquisition routines:

1. Clock overflow, bit 0
2. Interrupt on channel 1, bit 6
3. Interrupt on channel 2, bit 8
4. Interrupt on channel 3, bit 10

Each device has a select word associated to it whose function is to mask out all status bits except the one that determines if the device is ready to be checked. For example, the select word for an analog device is 4000. The coding that determines if a device needs to be checked is identical for all devices. For device K, the coding is:

```

TAD      STATUS
AND      SELDVK
SNA
JMP(I)   NXTDV
DCA      DVTYP
JMS I    KINPUT

```

Devices are deselected by clearing their select words. The select word for an undefined device is always 0. Every device has a set of parameters associated to it that are required for checking and servicing the device. If a device needs to be checked, the subroutine INPUT is called; the parameters for the device are directly beneath the call:

```

        JMS  I    KINPUT
          100  XAXDVK
          100  YAXDVK
          .
          .
          .
    EHIDVK,  0
            JMP (I) NXTDV          /END OF LIST
                                     /CONTINUE RTN
                                     /HLT RTN
          .
          .
    NXTDV,   TAD  STATUS
          .
          .
          .

```

The INPUT routine retrieves the parameter list, checks the devices, and returns. If the device is to be halted, input returns to the halt return; otherwise it returns to the continue return. Each device is checked in an identical manner; the last device checked returns to the interrupt dispatch routines so that the keyboard and Teletype may also be checked.

After retrieving the parameter list, INPUT checks the high order bit of DVTYP to determine if the device is analog or encoded. DVTYP = 4000 for analog and 0XXX for encoded devices. If the device is analog, the routine ANALOG is called to check for a 1 bit increment of the X axis. ANALOG samples the X axis (NEWSAM) and compares the value read with OLDSAM, which holds the X axis A/D value for the last detected point. If NEWSAM is greater than OLDSAM, a point has been detected. At this juncture, NEWSAM replaces OLDSAM and ANALOG returns in line (ANARTN) for further checking of the device. If ANALOG finds no increase, NEWSAM is checked for a 20 bit decrease. If the X axis has dropped 20 bits, ANALOG exits to the halt return; otherwise, it exits to the continue to next device return. For encoded devices or analog devices that make it to ANARTN, INPUT advances and checks the detected point counter. When this counter goes to zero, a point is ready to be acquired; otherwise, INPUT exits to the continue return:

```

    ISZ  I  FREQ
    JMP  I  INPUT
    .
    .
    .

```

If the device is ready, INPUT resets the detected point counter, sets up a 15 bit buffer address (CDF INST and 12 bit address), SAMS the Y axis, stores away the value, and resets the field to 0 (CDF0). INPUT then compares the

current device address to the terminal device address to determine if all points have been acquired and, if so, exits to the halt return. If the device is not finished, INPUT increments the current address by 1 and exits to the continue return. Returning to the halt return affects the values of 2 parameters 0 is put into SELDVK to deselect and the device number (0-4) is put into HSWITK for subsequent use by the routine CHKSWI.

### A.3 TELETYPE ROUTINES

The Teletype routines work in conjunction with a que stack capable of holding the addresses (of the first character of each message) of twenty messages. Messages go on the que via the pointer QUEIN, and come off via the pointer QUEOUT; QUECNT always holds the number of untyped messages on the que. The que stack is ringed so that the address of message 20 is adjacent to the address of message 1. Each message consists of a string of 8 bit ASCII characters followed by the message terminator 0000. The subroutine MESCHK takes messages off the que, and the subroutine QUPUT puts them on the que.

#### A.3.1 MESCHK

MESCHK first looks at QUECNT to see if it is zero. When QUECNT is zero, there are no more messages waiting to be typed so MESCHK exits in this case. Location ADMESS is the pointer to the current message being typed; once ADMESS is initialized by MESCHK, it is controlled by the interrupt routines (see above). If QUECNT is non-zero (message waiting), MESCHK looks at ADMESS (TAD I ADMESS). If the AC is non-zero, a message is in progress and MESCHK exits. If the AC is zero, MESCHK must verify that the last character of the message has been typed; this is done by checking the location TTYFLG. If TTYFLG is non-zero, the TTY is busy and MESCHK exits. If the TTYFLG is zero, MESCHK takes the next message off the que:

```
TAD I QUEOUT
DCA ADMESS
```

QUEOUT is advanced to the next message; if it is at the end of the que, it is reset to the start of the que. The QUECNT is decreased by 1 and the first character of the message is typed to initiate the Teletype interrupt routines that will finish the message. MESCHK then exits.

#### A.3.2 QUPUT

When QUPUT is called, the address of the message is in the AC. QUPUT saves the message in QSAVE and checks QUECNT to see if the que has room for only one more message and, if so, the message in QSAVE is ignored and the busy message (HANDS OFF) is put on the que instead. If QUPUT makes it by the busy message, it checks to see if the que is full and, if so, QUPUT exits. If the que is not

full, the message is put on the que via QUEIN; QUEIN is advanced to the next location and QUECNT is incremented by 1; QUPUT then exits.

#### A.4 KEYBOARD ROUTINES

All incoming characters are dispositioned by the subroutine KEYBRD. When KEYBRD is called it immediately checks location KBDBUF to see if the interrupt routines have retrieved a character. If KBDBUF is zero, no character is waiting and KEYBRD exits. If KBDBUF is non zero, a character is waiting. The waiting character is put into location CHAR which is part of the character look up table, and KBDBUF is cleared (set to zero) so that the waiting character will be dispositioned only once. KEYBRD next searches the character table. Each defined character is assigned a consecutive pair of core locations. The first location in the pair is the 8 bit ASCII code of the defined character, the second location is JMP to a routine appropriate to the character. The last pair in the table is:

```
CHAR, 0
      JMP I KEYBRD
```

The reason for this pair is to ensure that all keyboard characters will generate a match in the table search routine. All undefined characters cause KEYBRD to exit. Starting at the top of the table, KEYBRD compares each defined character with the waiting character in location CHAR. The search routine ISZ's by the jump instructions in the table location. JMP X is never compared with CHAR. When a match is found at location A, KEYBRD executes the jump instruction at location A+1 via:

```
TAD      (LOC A+1)
DCA      .+1
0                               /JMP INST
```

The following is an explanation of the various routines that are called from the character table.

##### A.4.1 D Routine

This routine restarts DIAL with IOF and XOB clear by jumping to the following bootstrap:

```
4015      LDF 3
4016      RCG I
4017      7300
```

##### A.4.2 CTRLH Routine

The function of this routine is to put TISA into Pause mode. Location MODE determines whether TISA is in pause or A/D mode; 0 = pause, 7777 = A/D.

CTRLH first checks MODE; if it is zero, CTRLH exits (JMP I KEYBRD). If TISA is in A/D mode, CTRLH sets MODE to 0, sends 0 to CLEN to disable all clock interrupts, sets the select words for all devices to 0, puts the address of the pause mode message in the AC, calls QUPUT, and then exits.

#### A.4.3 H Routine

The character H is the first in a two character command of the form HN(N=0,4). The function of the H routine is to enable the N half of the command. H does this by setting the word HSWIT to 7777. H also clears the go switch GSWIT (sets it to 0) because there is another two character command GN. HSWIT and GSWIT cannot be simultaneously enabled, i.e., TISA waits for either the N half of G or the N half of H, but not both. After setting HSWIT and GSWIT, H exits.

#### A.4.4 G Routine

G has the opposite effect of H; i.e., GSWIT is set to 7777 and HSWIT is set to 0.

#### A.4.5 P Routine

The function of the P routine is to complement the polarity word POLAR. Depending upon the value of POLAR, the display routines do or do not complement a data word before it is displayed. The logic employed is:

```
                TAD    (DATA WORD)
                SRO I
POLAR,          /7777 COMPLEMENTS
                COM
```

After the polarity word is complemented, the P routine checks its value and sends either the normal or inverted message to QUPUT, then exits.

#### A.4.6 F Routine

Window motion depends upon the value of one word, FREEZE. The logic in the display routines is as follows:

```
                SRO I
FREEZE,        JMP (FREEZE)
                JMP (MOTION)
```

The F routine dispositions the "F" command and the freeze word in a manner exactly analogous to the P routine.

The function of the S routine is to initialize setup mode. If TISA is in A/D mode, S exits; if not, then the interrupts are turned off and S jumps to the setup routines at location SETUP.

#### A.4.8 W Routine

If TISA is in A/D mode, W exits. If TISA is in Pause mode, W calls the LINctape write routines (see LINctape Routines, section A.5.3).

#### A.4.9 L Routine

If TISA is in A/D mode, the L routine exits; otherwise, L jumps to a routine that displays the following QANDA message:

```
LOAD _ _ _ _ _
UNIT _ _
```

which allows the user to specify program name and unit. When the user terminates the load message, the first character of the answer buffer is checked for 00; 00 means the user input no program name to QANDA and will redisplay the load message. When the message is terminated, the routine KLOAD is called, whose function is to transfer the QANDA answer buffer to DIAL's monitor parameter table (for a full discussion of the table see DIAL EDITOR V2 internal description DEC-12-ZW7A-D) at locations E6+2 (2373) - E6+6 (2377). The program name is first transferred, one half word at a time, to an intermediate table (LODPTR). Eight characters must be transferred for the program name to meet the requirements of the E6 table. Unused portions of a QANDA answer buffer contain 00. Unused portions of the E6 table must contain 77. If the user requested the program name ABCD, the two answer buffers would look as follows:

QANDA BUFFER	KLOAD BUFFER
ANSWER, 74 01 /74A	LODPTR, 01 02 /AB
02 03 /B C	03 04 /CD
04 00 /D -	77 77
00 00 /- -	77 77
00 74 /-74	YYYY /UNIT
XX /UNIT	

The KLOAD routine skips over the first QANDA 74 and terminates name transfer when it uncovers the second 74. The response for unit in the QANDA answer buffer is stripped to 3 bits and stored in the LODPTR table after the name. 60 (0) is unit 0 and 01 (A) is unit 1, etc. When the LODPTR table has been set up, KLOAD puts LODPTR-1 into the AC and calls the routine LOADER.

The function of the routine LOADER is to transfer the table whose address-1 is in the AC to the E6 table and call the DIAL loader. All program calls in TISA to DIAL binary files realize that setting up the E6 table and calling the DIAL LOADER is sufficient work to accomplish the task. LOADER turns off the interrupts, clears the XOB and special functions, registers and reads the Editor's grid table into LDF1 in case the DIAL LOADER decides to display "NO". The AC table is transferred to the E6 table and then the DIAL LOADER is read in and started. The logic is highly similar to executing an LO command in the DIAL EDITOR.

A.4.10 C and M Routines

The function of the C and M routines is to call the programs CATALCAL and MAGSPY, respectively. Both routines will execute a JMP I KEYBRD if TISA is in A/D MODE. Otherwise, the address, CATPTR-1 or MAGPTR-1, is loaded into the AC and the LOADER routine is called.

CATPTR, 0301 /CA	MAGPTR, 1501 /MA
2401 /TA	0723 /GS
0301 /CA	2031 /PY
1477 /L	7777
0 /UNIT	0 /UNIT

C and M both require the programs to be on unit 0. C, M, and L all require a DIAL V2 tape to be on unit 0. DIAL MS will not work because the MS LOADER is logically different from the V2 LOADER.

A.4.11 NUM Routine

The function of the NUM routines is to interpret numeric characters. NUM first strips the ASCII and stores the number in TEMP. NUM compares TEMP with DEVNUM, the number of defined devices, to check for an undefined number, exiting in such a case. If the number in TEMP is a defined device number, NUM checks HSWIT for 7777. If HSWIT is set, NUM clears it and calls the HALT routine. If HSWIT is not set, NUM checks GSWIT. If GSWIT is set, NUM clears it and calls the GO routine; otherwise NUM exits.

A.4.12 HALT Routine

The HALT routine checks the device select word to see if the device is in Pause mode and, if so, HALT exits. If the device is in A/D mode, HALT sets the select word to 0 to deselect the device, calls the routine HLTMS to put the Halt message on the Teletype que, and then exits.

A.4.13 GO Routine

The GO routine first checks the device select word to determine if the device is in A/D mode and, if so, GO exits. If the device is in Pause mode, GO

calls GOMS to put the go message on the que. With the aid of 4 tables and relative to the device number in TEMP, the GO routine next initiates the device.

Device K has 2 tables associated to it: TIADK and TISTRK. TISTRK holds the parameters that define the device, and TIADK holds the addresses within the data acquisition routines of these parameters. To initiate a device, six parameters must be set up within the acquisition routines: starting address (2 words), select word, sampling frequency (variable), sampling frequency (fixed), and initial X axis SAM. There are two other tables that are shared by all devices VALTAB and ADRTAB:

VALTAB, TISTRO	ADRTAB, TIAD0
TISTR1	TIAD1
TISTR2	TIAD2
TISTR3	TIAD3
TISTR4	TIAD4

Note that for device K - whose value is in TEMP - VALTAB+K, ADRTAB+K point to TISTRK, TIADK respectively. The GO routine employs the above logic in selecting the tables appropriate to device K. GO further sets up a SAM instruction because analog devices require an initial X-axis SAM at location OSDVK. GO then sets HSWITK to 7777 for the routine CHKSWI, turns the interrupts off, and transfers the device parameters to the data acquisition routines. The interrupts must be off during parameter transfer because it is possible to get an interrupt from device J when device K is only partially initialized. Such a circumstance could produce adverse results for device K. The interrupts are off for 200 microseconds, which presents no problem for the data acquisition routines. Once the device is initialized all clock interrupts are enabled and MODE is checked. If TISA is in Pause mode, the A/D mode message is sent to the que. GO then sets MODE to 7777, turns the interrupts back on, and exits.

#### A.5 LINCTAPE ROUTINES

There are six LINCTape routines that enable the user to either read a parameter list from LINCTape or write his data buffers onto LINCTape.

##### A.5.1 READ Routine

If the user selects the LINCTape option during setup mode, TISA calls the READ routine, which displays the QANDA message:

```
READ   TBLK  _ _ _  
UNIT  _
```

When the message is terminated, READ sets a pointer to the QANDA answer buffer

and calls DORDC.

### A.5.2 DORDC Routine

DORDC calls the routine TPARGS whose function is to translate a QANDA message which holds LINCtape arguments. If TPARGS returns to P+1, the QANDA input was illegal and DORDC implements the error exit, which is simply a recall to the READ routine. If TPARGS returns to P+2, then location TBLK holds the requested tape block; UNIT holds the low order unit bit i.e., UNIT=10 or 0; and the XOB holds the high order unit bits. Using the arguments from TPARGS, DORDC reads the requested TBLK into locations 7000-7377 (start of the data buffer), executes an I/O preset to clear all flags and the extended unit bits, resets no pause, and transfers the parameter list at 7000-7377 to the setup initialization table at 6000-6144. If the user requests a TBLK that does not contain a legitimate parameter list, DORDC checks the only value in the table that could be fatal to TISA and the user, to wit, the number of defined devices. The function of setup mode is to fill the initialization table with the user's parameters. As the user defines more instruments, more information is input to this table. When setup mode is terminated, TISA takes the information from the initialization table and distributes it to appropriate locations within the program. The number of user defined devices is in the table and determines the length of the table transfer. If the number of defined devices is greater than the maximum number allowed, TISA will extend beyond the initialization table and transfer arbitrary values to arbitrary locations; such a situation is quite obviously fatal. If the number of defined devices is too large, DORDC executes the error exit; otherwise, DORDC goes to the last leg of the setup routines (QADUN+3), just as if input was from the keyboard.

### A.5.3 WRIT Routine

WRIT first shuts off the interrupts so that the user cannot inhibit the write sequence via the keyboard. WRIT then retrieves the number of blocks to write from the initialization table (loc. BLKCNT), translates it into DIAL codes, and puts it into the text field of the QANDA write message. The following message is then displayed:

```

WRITE      XYZ  (OCT)  TBLKS
FIRST     TBLK =  --  --  --
UNIT      _

```

where XYZ is the number described above. When the user terminates the message, WRIT sets a pointer to the QANDA answer field and calls DOWRC.

#### A.5.4 DOWRC

Like DORDC, DOWRC calls TPARGS; at the error return (P+1) is a recall to the routine WRIT. If there is no QANDA input error, DOWRC sets up the following arguments for DOMTP: WRI I (U) to ARGHI, MBLK\TBLK to ARGLO; then calls DOMTP. When DOMTP returns to DOWRC, the operation is finished and DOWRC turns the interrupts on and hangs up in the idle loop.

#### A.5.5 TPARGS

The function of TPARGS is to retrieve the TBLK and UNIT arguments from a QANDA answer field and to return to P+1 for input error and P+2 for no error. To achieve this end, TPARGS enlists the aid of the more general QANDA numeric decoder DIGCHK, a routine that is heavily exercised during setup mode. DIGCHK has 3 returns:

```
JMS I KDIG                                /CALL
                                           /NON DECIMAL CHAR RTN
                                           /QANDA END OF QUES. FIELD RTN
                                           /NORMAL RTN OCTAL NUM IN AC
```

TPARGS changes location UPLIM of DIGCHK to minus 10 (1's complement) so that non-octal input will be illegal. TPARGS then calls DIGCHK, retrieving in this manner one numeric argument at a time. If DIGCHK returns TPARGS to P+1, then TPARGS returns to the calling program at P+1 (the TPARGS error return). The arguments for TBLK are rolled and packed into location TBLK. DIGCHK is called until it executes the end of field return at which point TPARGS calls DIGCHK one more time for the UNIT argument. The low order unit bit is put into bit 8 of location UNIT and the two high order unit bits go to bits 10 and 11 of the XOB. After the tape arguments have been set up, TPARGS restores UPLIM to its regular value and returns to P+2. TPARGS always restores UPLIM before exiting, error or no error.

#### A.5.6 DOMTP

The function of DOMTP is to execute a series of tape transfers as indicated by a list of arguments. Currently, only DOWRC calls DOMTP, but DOMTP is general enough in nature to read data as well as write it. The arguments for DOMTP are:

```
BLKCNT = - (the number to do)
ARGHI  = a LINctape instruction
ARGLO  = initial MBLK\TBLK
LDFSTR = initial LDF
```

DOMTP first retrieves and sets up its arguments, then issues a tape instruction and calls TPWAIT, which is simply a routine to hang in the idle loop until the

tape operation is complete. When the tape is finished, DOMTP adds 1 to the TBLK and 1 to the MBLK, then checks the MBLK to see if it advanced from 7 to 0. When the MBLK goes to 0, DOMTP resets it to 4 and adds 1 to the LDF instruction at TPLDF. All tape transfers are done via the data field. DOMTP then iterates the tape process, counting each step. When the operation is finished, DOMTP stops the tape, clears the extended unit bits, and exits.

## A.6 DISPLAY ROUTINES

The Display Routines present the user with one of two displays, depending upon SSW0: Help Frame, or Data Buffer Display.

### A.6.1 DSCHLP Routine

The DSCHLP Routine (help frame) is driven by 4 tables: (1) HLPTAB, which contains the DIAL codes for all the characters displayed in the help frame plus a number of control characters that are not displayed; (2) VCTAB, which is a table of scope vertical coordinates and determines the vertical alignment of the help frame; (3) HCTAB, which is a table of scope horizontal coordinates and determines the horizontal alignment of the help frame; and (4) GRDTAB, which is the QANDA table of grid patterns. The following codes in HLPTAB are the control characters that direct DSCHLP to take certain actions:

<u>Symbol</u>	<u>Value</u>	<u>Meaning</u>
FULL	0600	Switch to full size characters
CRL	4300	End of a line; go to next line
ENDWD1	7700	End of help frame; exit
CHGHC	3600	Reset horizontal coordinate

The LINC data field is set to 2, where the GRID table is located.

DSCHLP first sets up pointers to the tables and falls through to LNNXT. LNNXT puts the next vertical coordinate in VCOORD, the next horizontal coordinate in BETAL, and then checks the first character of the line to determine full or half size characters. The link bit is set to 0 for half size characters and to 1 for full size characters. After setting character size, LNNXT goes to CHRNXT. CHRNXT loads a character from HLPTAB and checks for the control characters, CRL, ENDWD1, and CHGHC. CRL causes a jump to LNNXT, ENDWD1 effects an exit, and CHGHC causes the next value in HCTAB to be placed in BETAL. If CHRNXT finds no control characters, it calls DSCCHR routine to DSC the character whose DIAL code is in the AC.

### A.6.2 DSCCHR Routine

DSCCHR finds an address in GRDTAB relative to the contents of the AC and stores the address in GRDPTR (BETA14). VCOORD is loaded into the AC and the character is displayed. To make a horizontal gap between characters so they won't look pressed together on the scope, 2 or 4 is added to BETA1. DSCCHR checks the link bit which indicates character size. A gap of 2 is for half size and 4 is for full size

### A.6.3 Data Buffer Display

The Data Buffer Display is controlled by two knobs, 0 and 4. The high order 9 bits of knob 4 determines cursor position; 9 bits will cover 512 points. The A/D value 7400 is converted to -1, and the value 377 is converted to -1000. The converted value is called the cursor count and is stored in location CURCNT. Every time a point is displayed, CURCNT is ISZed. When CURCNT goes to zero, the cursor is displayed. The high order 6 bits of knob 0 is added to the initial scope address to find the address of the first point (leftmost scope point) to be displayed. Suppose the knob value is 10 and the initial scope address is 2000. The display routines display the data from 2010 to 3010 the first time they are called; the second time the display is called, locations 2020 to 3020 are displayed, and so on. Varying the initial scope address in the manner described above lends the appearance of motion to the display routines. Positive knob values move the display from right to left; negative values, from left to right.

When the Data Buffer Display is called, knob 4 is put into CURCNT, XCORD (loc 2017) is set -1000, and knob 0 is put into the AC. Next the FREEZE location is checked. If the display is frozen, the display routine (GODIS) is called immediately. If the display is not frozen, the routine SETMOT is called to determine the initial scope address relative to knob 0.

A.6.3.1 SETMOT - The locations BUFHI, BUFLO hold the initial 15 bit scope address that was used for the last call to the Data Buffer Display. The function of SETMOT is to update this address for each call. The routine further must be capable of making the initial and terminal points of the buffer contiguous. Suppose knob 0 has a value of 20 and BUFHI, BUFLO = 0001,7770 (8K of core). Adding 20 to this address extends 10 locations beyond the size of the data buffer, i.e., 0002,0010. There is no field 2 on an 8K PDP-12. This effect is called high end wrap around. For this case, SETMOT sets the initial address to 10 locations past the initial data buffer address, i.e., 0001,7010. Low end wrap around is handled in analogous fashion. When SETMOT exits, the proper address is in BUFHI, BUFLO. Note that all routines which must handle 15 bit

addresses treat the high order half as a PDP-8 field. An address at HI,LO is referenced in the following manner:

```

TAD HI
CLL RTL
RAL
TAD CDF0
DCA .+1
0
TAD I LO
/CORRECT CDF INST
/CORRECT ADDR

```

A.6.3.2 GODIS Routine - GODIS displays 512 consecutive data points, beginning with the address BUFHI,BUFLO. BUFHI goes to CURCDF, BUFLO goes to BUFPTR. The routine SETCDF sets the data field relative to CURCDF. The BETA register for display is XCORD (2017) and runs from 7000 to 7777. Note that hardware channel 1 is selected. When CURCNT goes to zero CURDIS is called to display the cursor. If BUFPTR goes to zero, a PDP-8 field boundary has been reached and FWDCDF is called to set the next data field. FWDCDF also checks for high end wraparound. BCNTR is set to -1000 when this location goes to zero, GODIS is finished, XYGET is called to display the X,Y coordinates.

A.6.3.3 CURDIS - When CURDIS is called, the cursor Y coordinate is in the AC and it is stored at location YCUR, not only for use by CURDIS but also for subsequent use by the X,Y coordinate routines.

The contents of the data field are put into XCURHI and BUFPTR is put into XCURLO. These two locations are the X coordinate and will be subsequently accessed by the routine XYGET. YCUR is compared to MAXCUR; if YCUR is greater than MAXCUR, -1 is put into TEMPHI and the cursor is displayed below the spectrum. Minus the length or the height of the cursor is in location CURLN. The cursor is displayed by putting YCUR into the AC, displaying it CURLN times via a DIS XCORD; prior to each DIS, TEMPHI is added to the AC.

A.6.3.4 XYGET Routines - The routines that determine the X,Y decimal coordinates use a table called LIMVL whose format is:

```

LIMVL0,    0      /-starting address device 0 (LO ORD)
           0      /-starting address device 0 (HI ORD)
           0      -sampling frequency device 0
           BCD0   /address of initial X-coordinate device 0
           .
           .
LIMVLK,    0      /-S.A. device K (LO ORD)
           0      /-S.A. device K (HI ORD)
           0      /-sam.freq. device K
           BCDK   /addr of X-COORD device K
           0      /-first unused core
           0      /location

```

The first task of XYGET is to find the data sector that contains the address XCURHI, XCURLO. This is done by adding the X coordinate to the starting addresses in the LIMVL table, always saving the sum for step k-1. If the sum for step k becomes minus, then the X coordinate lies in between the starting addresses used for steps k-1 and step k; and the sum for step k-1 is the address of the X coordinate, relative to the data sector that contains the X coordinate. Suppose that the data sector for device 0 is  $1000_8$  points and the X coordinate is 07500. The starting address for device 0 is 0 7000; for device 1 it is 1 0000. The X coordinate is the 500th point in the data sector for device 0. Subtracting the starting address for device 0 yields a sum of +500 which is saved. Because the result is positive, the process continues for device 1, yielding  $07500 - 10000 = -300$ , which is negative. The negative result means that the previous sum, step k-1, i.e., 500, is the desired address. If the entire table is scanned without a negative result, the X coordinate is in undefined data area and the XYGET routines exit. The relative address is then multiplied by the sampling frequency; this result is converted to a six digit BCD number. A routine to deflate by decimal radii is used for the conversion. The BCD number is stored, one digit to a word, at locations BCDVAL-BCDVAL+5. If the sampling frequency is 3, the value to convert to BCD is 1700. The decimal value for 1700 is 960, which is stored as:

```
BCDVAL, 0
         0
         0
         11    /9
         6     /6
         0     /0
```

960 is a relative decimal value and must be added to the initial X coordinate which is stored from low to high, starting at location BCDK:

```
BCDK,      /UNITS
           /TENS
           /HUNDREDS
           /THOUSANDS
           / 10 THOUSANDS
```

The number at BCDVAL is added as a BCD number to the number at BCDK. To obtain the DIAL code for each digit, add 60 to the sum for each digit. The DIAL codes are stored in the table DCODEX:

```
DCODEX, 3075 /X=
         4300 /CR 100K
         0000 /10K 1K
         0000 /100 10
         0043 /UNITS CR
```

These values will subsequently be displayed by the routine DSCXY.

After the table DCODEX has been set up for the X coordinate, the table DCODEY must be set up for the Y coordinate. First, 400 is added to YCUR to put it in the range 0-511(0-777); YCUR is then converted to a BCD value and then to DIAL codes, one digit at a time, and stored in the table DCODEY.

A.6.3.5 DSCXY Routine - DSCXY sets the LDF to 2 to access the grid table; sets half size characters, and sets the X,Y display coordinate for the top left of the scope (0,400). The routine always displays X=, Y=, but never displays leading zeroes or the value 0000. The non-numeric characters are separated from the numeric characters by the carriage return code (43). Display of numeric characters is inhibited until the first non-zero digit (DIAL code  $\neq$  60) is encountered. The characters are displayed by loading their codes into the AC and calling the routine DSCCHR. When DSCXY is finished, the entire scope display is complete. DSCXY exits to SCPRTN.

#### A.7 SETUP ROUTINES

The function of the Setup Routines is to decode QANDA responses and put them into the initialization table which runs from location INISTR to LAST, or to fill the table from a specified block of LINctape. QANDA and calls to QANDA are in segment 2. The Setup Routines start at location SETUP. For each QANDA setup message, the refresh return contains the instruction JMP CHKSNS:

```
CHKSNS,    SNS 0
           JMP QARFSH
SETUP,     PDP
```

thus activating SSW0 for all of Setup mode. The Setup Routines first call the routine FRSTR (fresh start), which clears the data buffer and sets all pointers to their initial values. Message 1 is then displayed. After message 1 comes either the READ message (see above) or message 2.

##### A.7.1 MESS2 Routine

MESS2 first calls the routine RESTR (restart a device), whose function is to reset all pointers to their initial values for defining device K. The initial values for device K are preserved throughout the entire definition of device K so that if an error is made in input the routine RESTR will be able to reset them.

All values go into the table via the index pointer INIPTR. When message 2 is terminated, the QANDA answer buffer is decoded and put into the initialization table. The first value is the device type. If the device is analog, the value

ANAVAL (4000) is put in the table; if the device is encoded, a value from the table ENCVAl is put into the initialization table. The next 3 values are the STARTING POINT, TOTAL POINTS, and SAMPLING FREQUENCY. The routine DIGCHK is used to convert the DIAL codes to digits. The routine DIGCHK expects that BETA14 (ANSPTR) is pointing to a QANDA answer field. When DIGCHK is called, it retrieves one QANDA character. If the QANDA character is 00 (a blank) it is ignored and another character is retrieved. If the QANDA character is QANDA end of question field (74,<) or QANDA end of answer buffer (34,\), DIGCHK exits to P+2. If the character is decimal (codes 60-71), DIGCHK exits to P+3 with the numeric value (0-11) in the AC. If the character is not decimal and not either 74 or 34, DIGCHK exits to the error return P+1. All DIGCHK error returns initialize the routine MESS2.

To setup the starting point, the routines DIGCHK and DSCL4 are alternately called until the end of question field return. DSCL4 takes the contents of the AC and rolls and packs them 4 bits each call into the double precision accumulator (DAC). After the starting point has been put into the DAC, the routine UPACK is called to take the BCD values out of the DAC and put them into the initialization table.

After STARTING POINT has been set up, TOTAL POINTS is set up. For TOTAL POINTS, the routines DIGCHK and MUL10 are alternately called until end of question field. MUL10 does a decimal to octal conversion with the DAC and AC. When TOTAL POINTS is in the DAC, the arguments appropriate to TOTAL POINTS are put into the table, e.g., starting address device K, 2's complement of the starting address, etc. The error conditions "no points" and "buffer full" are checked for at this time.

Like TOTAL POINTS, SAMPLING FREQUENCY is retrieved by alternately calling DIGCHK and MUL10. After the sampling frequency is in the DAC, the error condition greater than 10 is checked for. If the sampling frequency is zero, it is set to 1 before being input to the table. Sampling frequency terminates definition of the device and message 3 is displayed.

#### A.7.2 MESS3 Routine

If the user requests more devices, the error conditions "no more devices" and "buffer full" are checked first, then the routine NXSTR (start next device) is called, whose function is to set up all pointers for the definition of the next device; MESS2 is then recalled. If the user requests to start, an I/O preset is executed and the routine GBLK is called, whose function is to translate the defined data area into (-) the number of TBLKS and store it at location BLKCNT. After GBLK, the routine QADUN is called, whose function is to initialize TISA.

Associated to the initialization table is the table of addresses INIAD, which tells where all the initial values are to go. Using INIAD, QADUN distributes the initial values throughout TISA. The buffer preset and clock control registers are set at this time. The TTYQUE is initialized; pause mode is entered by setting the mode to A/D and putting the control H character into KBDBUF. The interrupts are turned on and the IDLE routine is called.

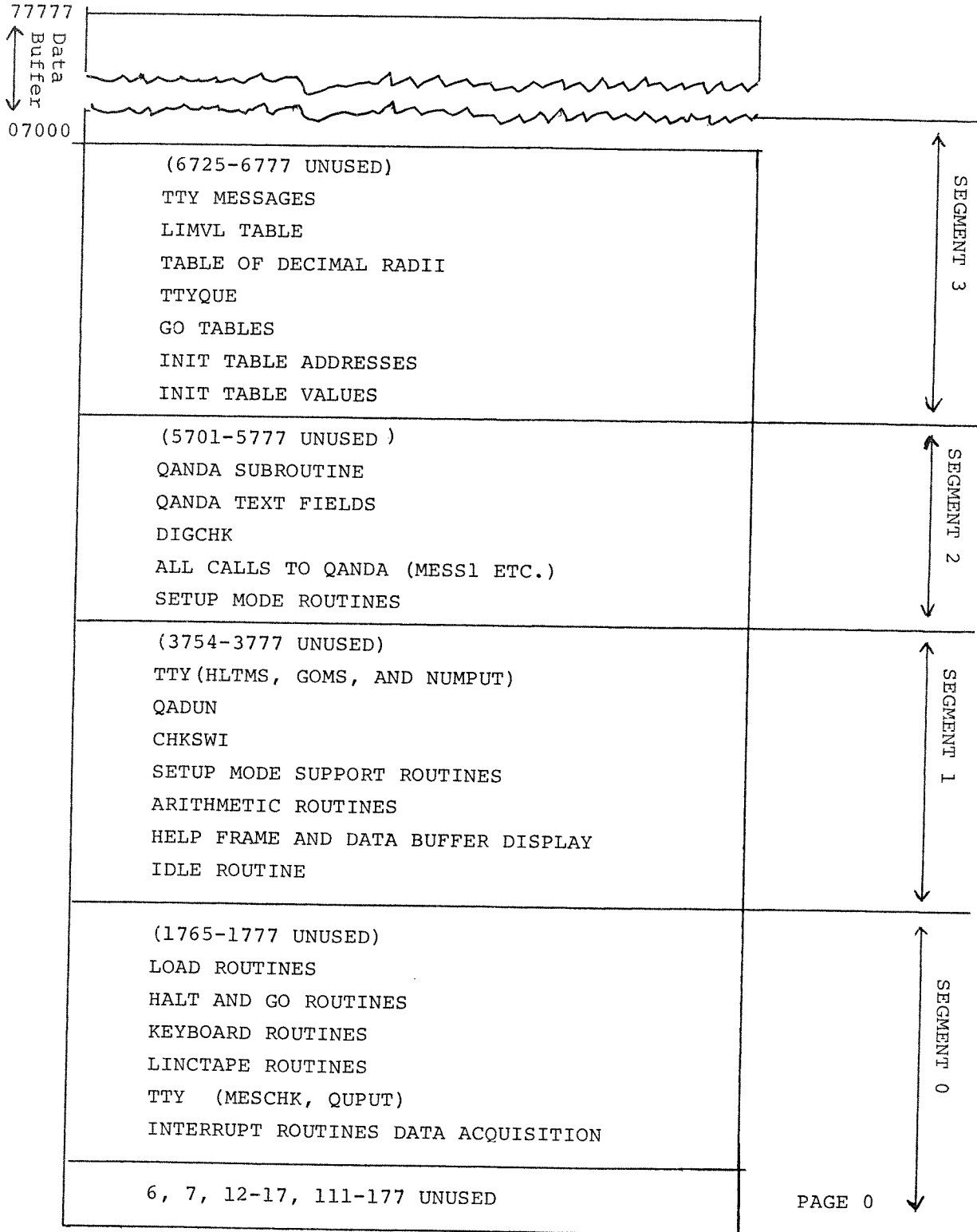
The function of the IDLE routine is to keep TISA busy, when it is not acquiring data, by calling four routines and exiting. The routines and the order in which they are called are

1. SCOPE DISPLAY
2. KEYBRD
3. CHKSWI
4. MESCHK

#### A.8 MEMORY CHECK ROUTINE

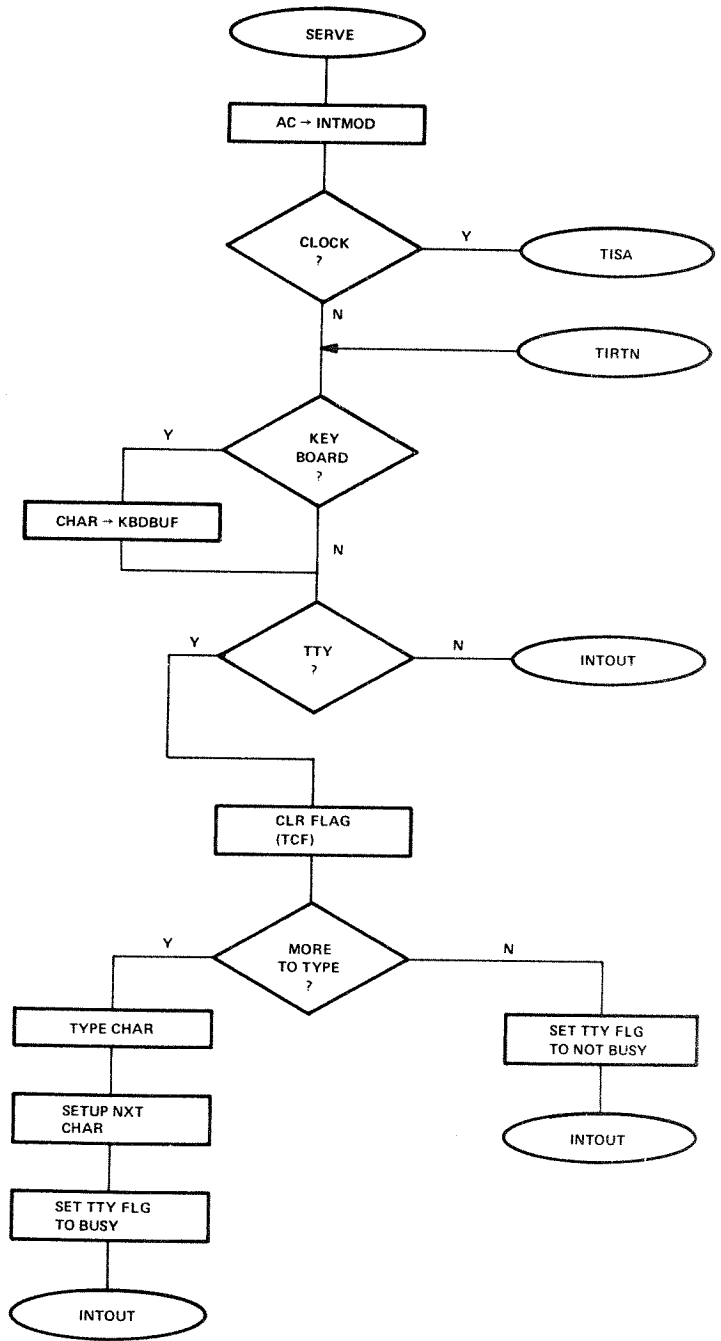
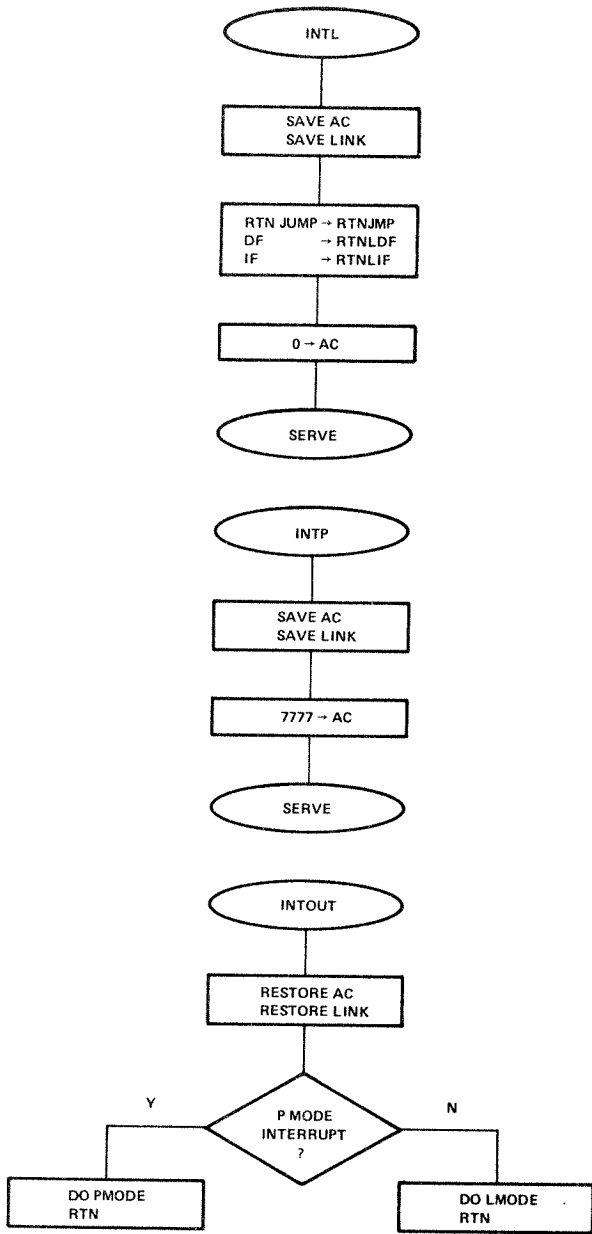
The routine MEMCHK lies in the data area and is called once only (when TISA is loaded from LINCTape). Its function is to determine the core limits of the PDP-12 on which TISA is running. MEMCHK puts the largest PDP-8 field in location MAXCDF and exits. This routine must be called for TISA to function properly and is the primary reason why the starting address for TISA is 4001 and the restart address is 4020.

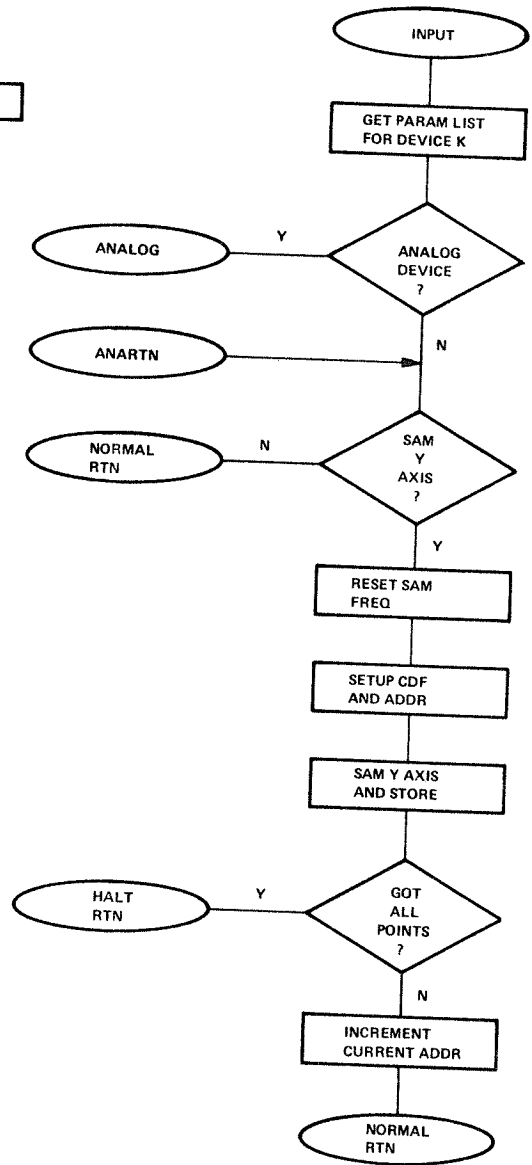
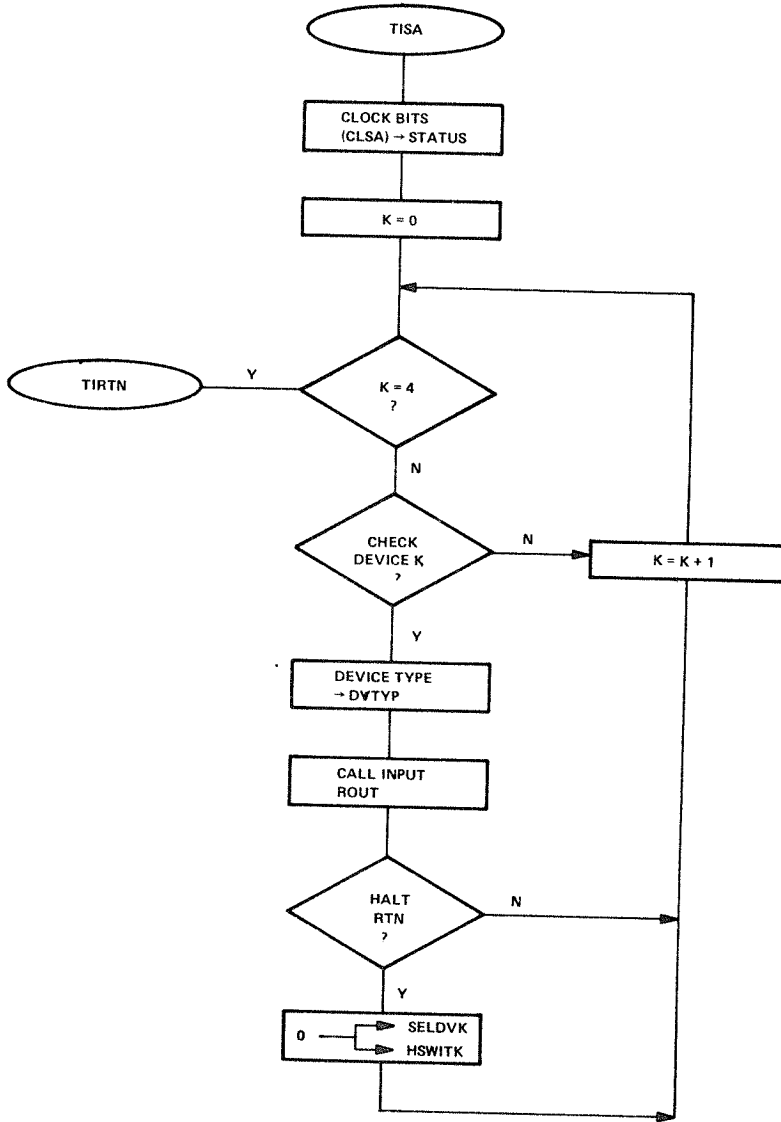
A.10 CORE ALLOCATION

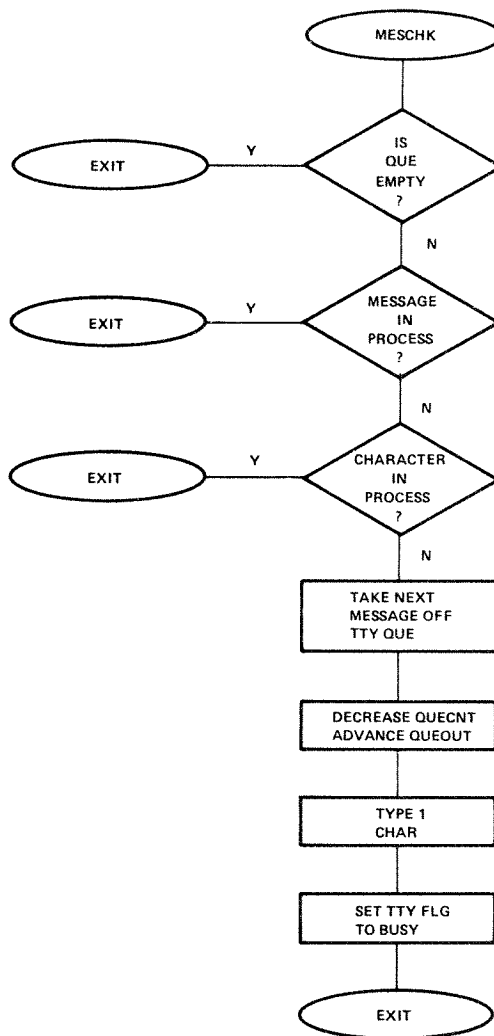
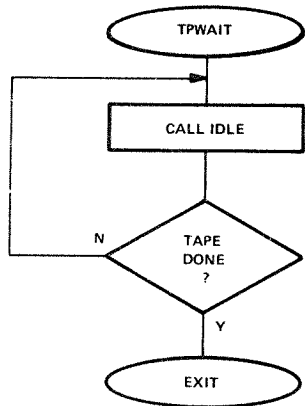
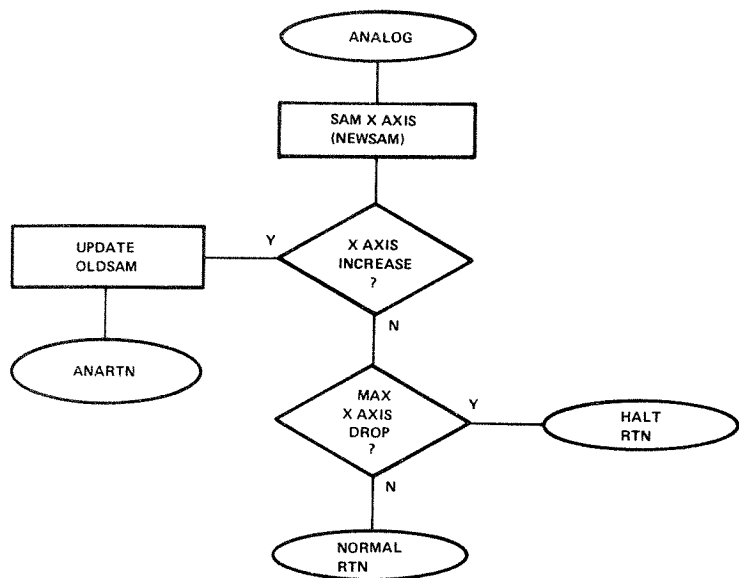


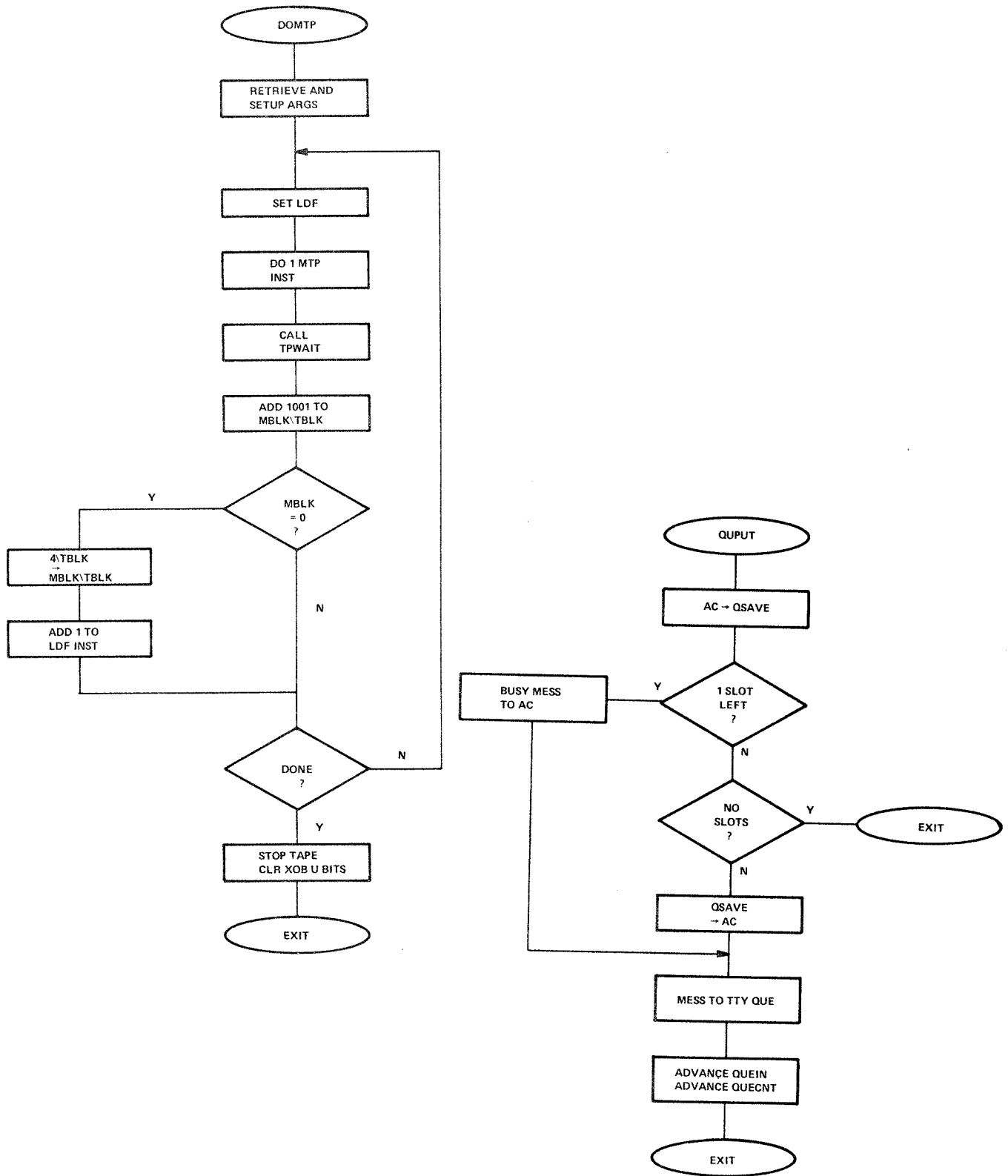


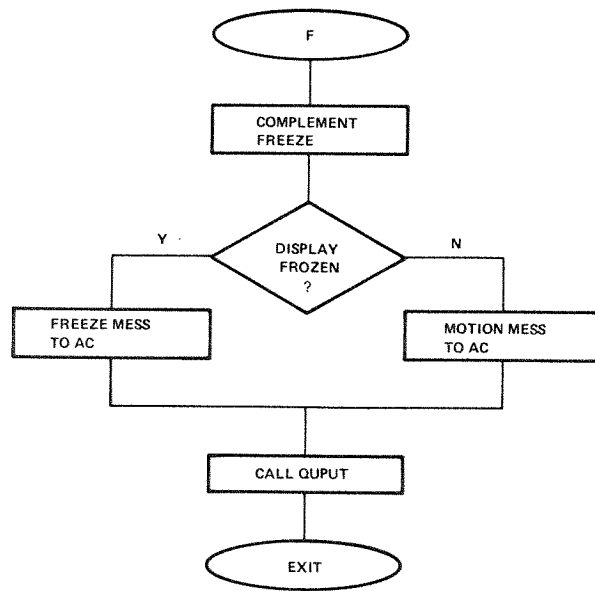
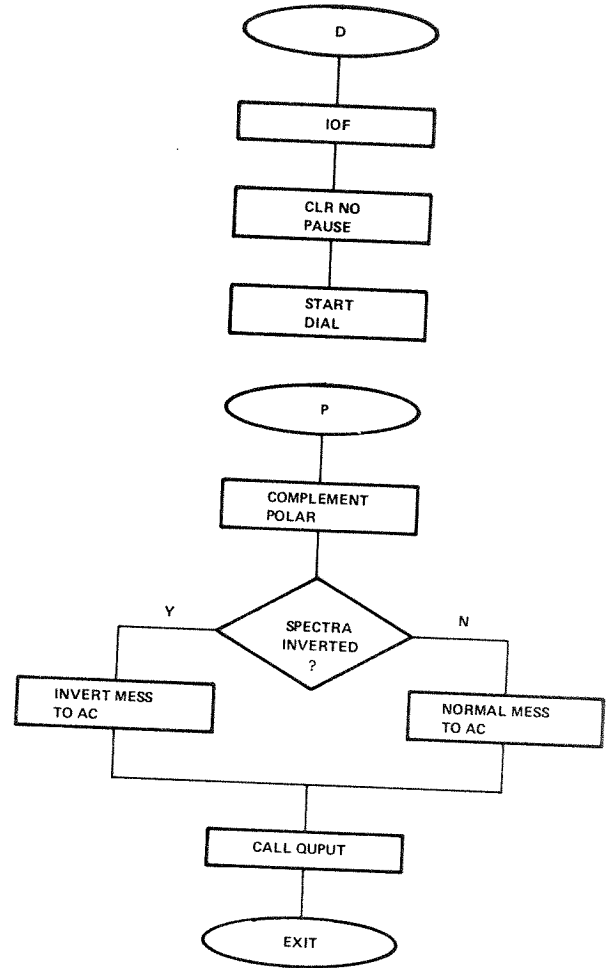
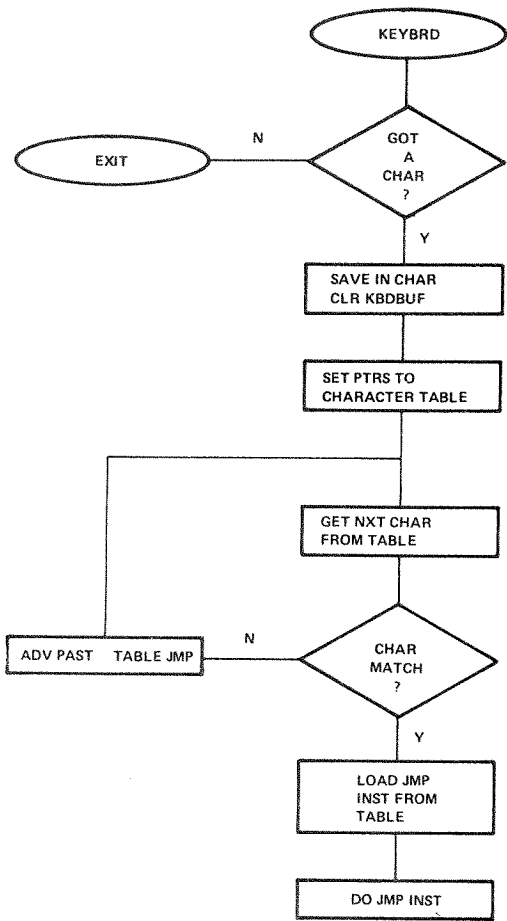
Appendix B  
Flowcharts

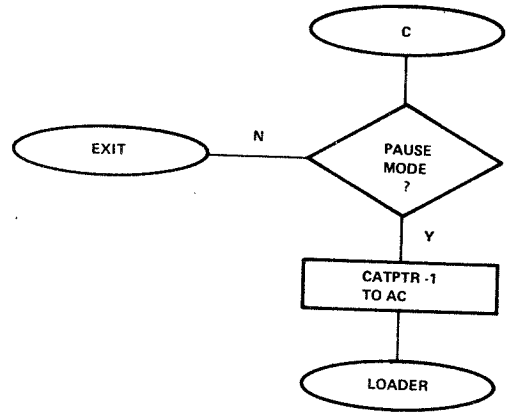
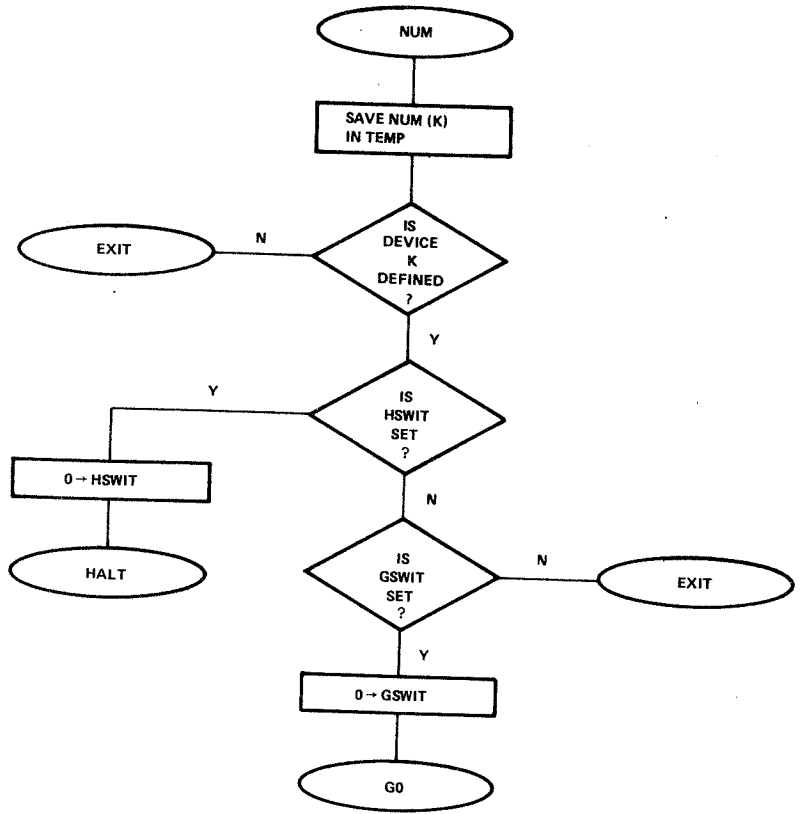
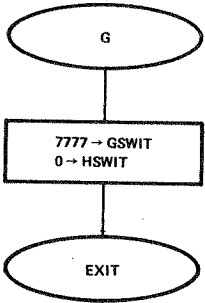
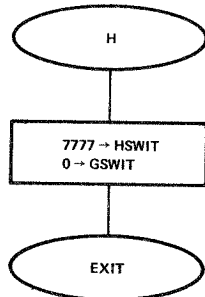
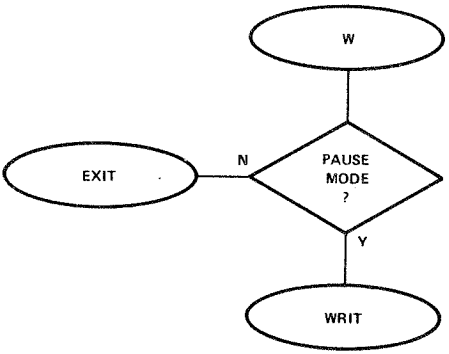
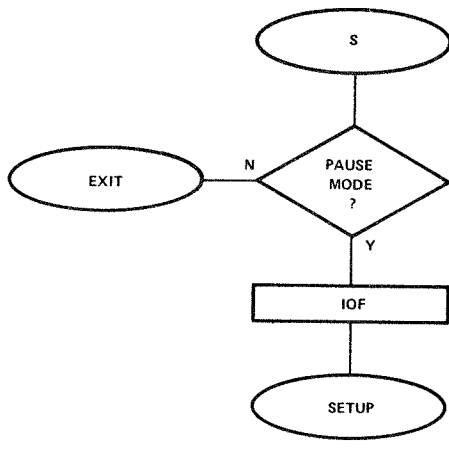


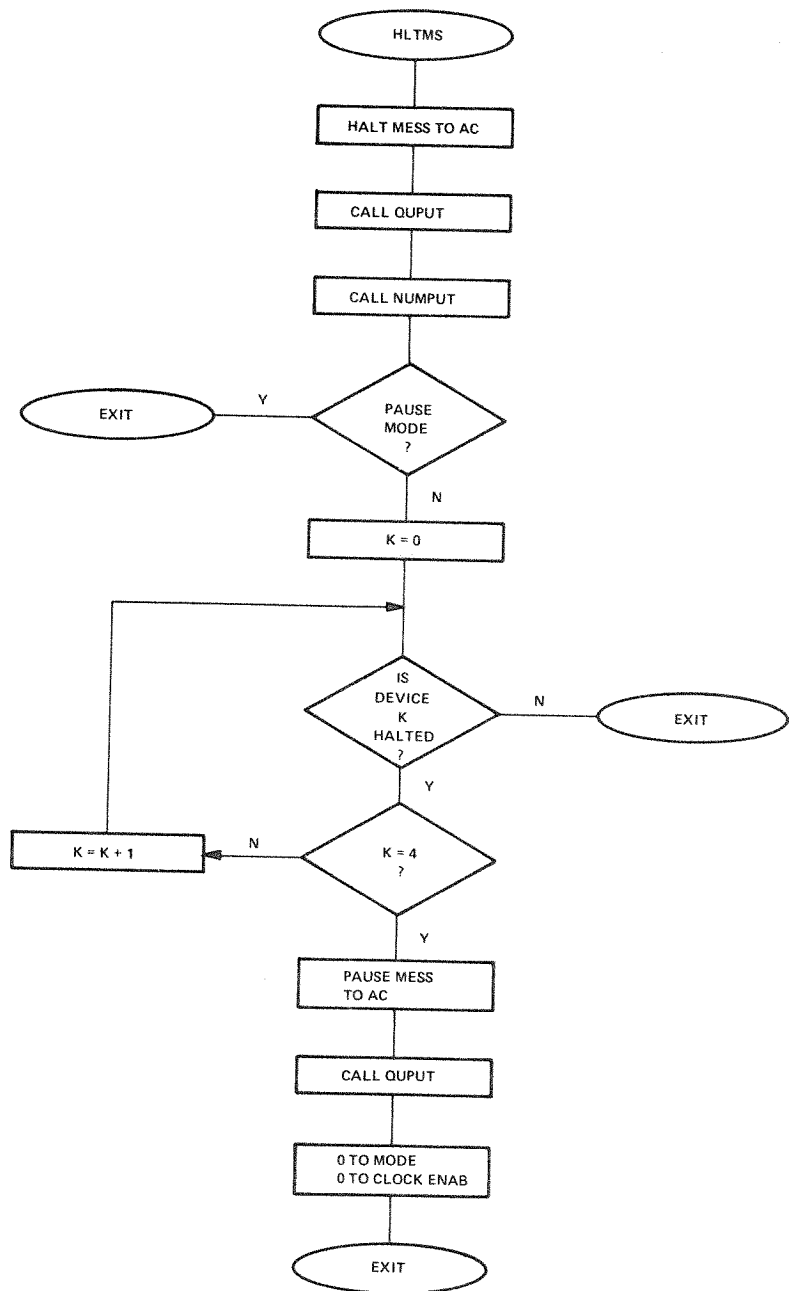
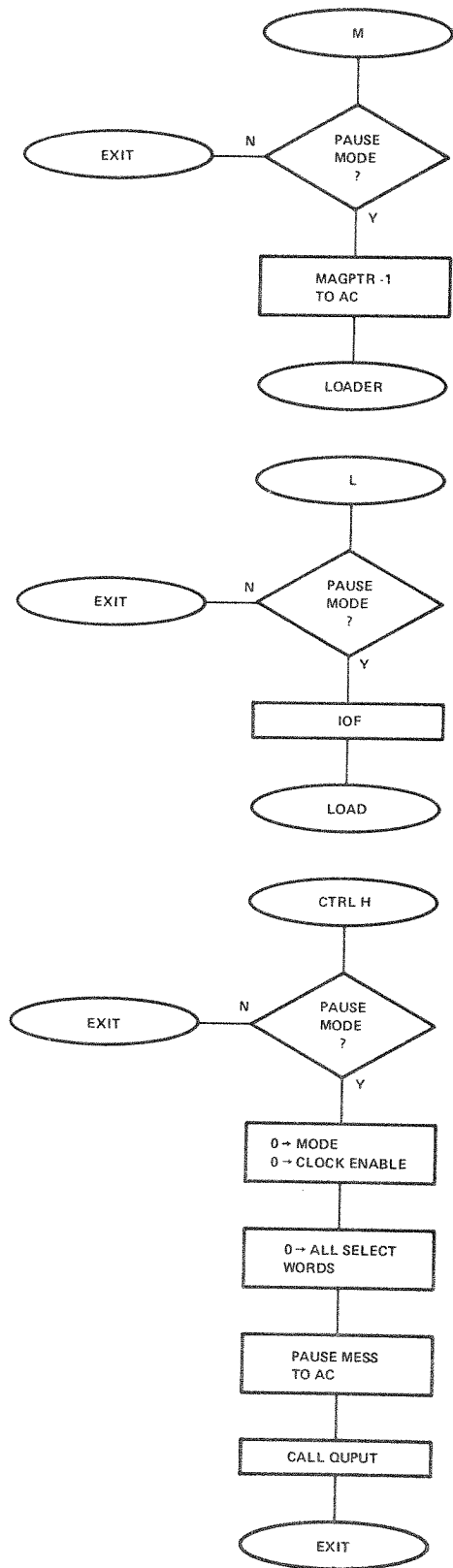


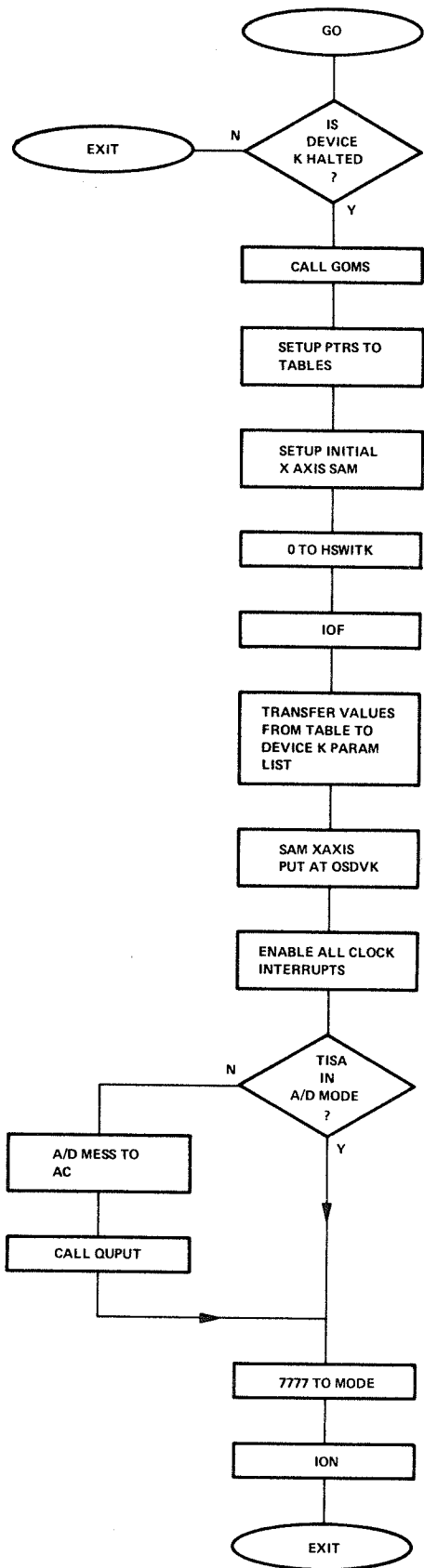
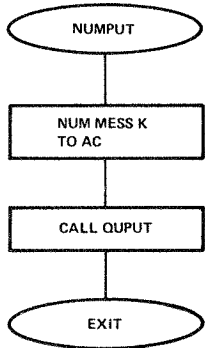
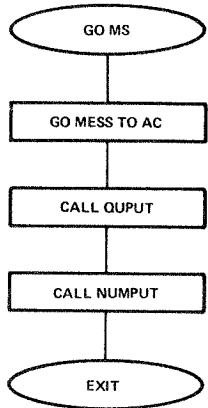
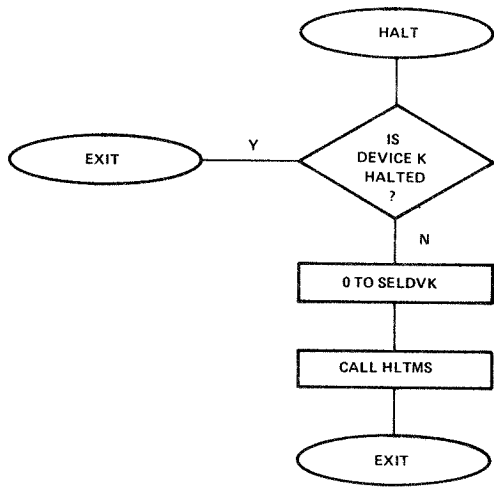


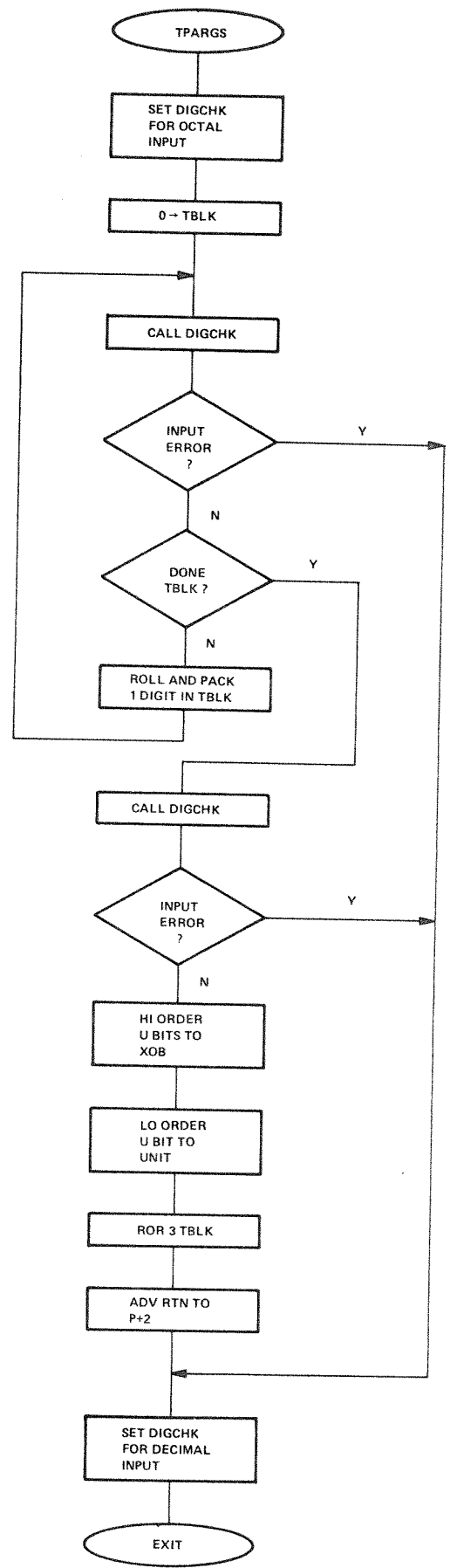
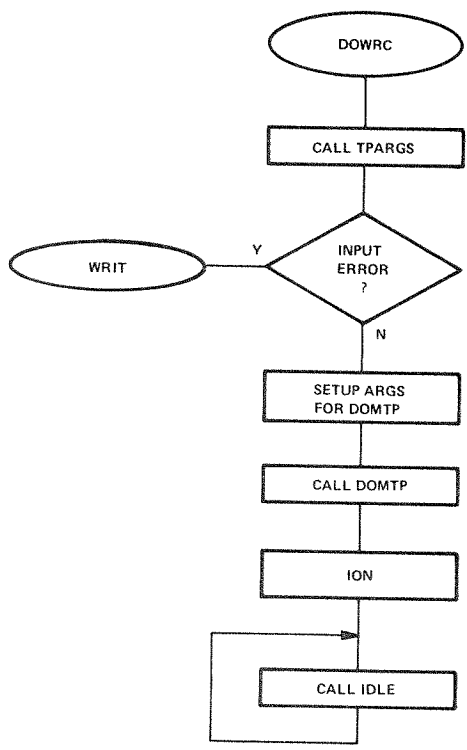
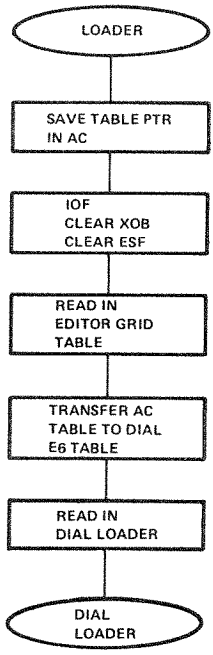


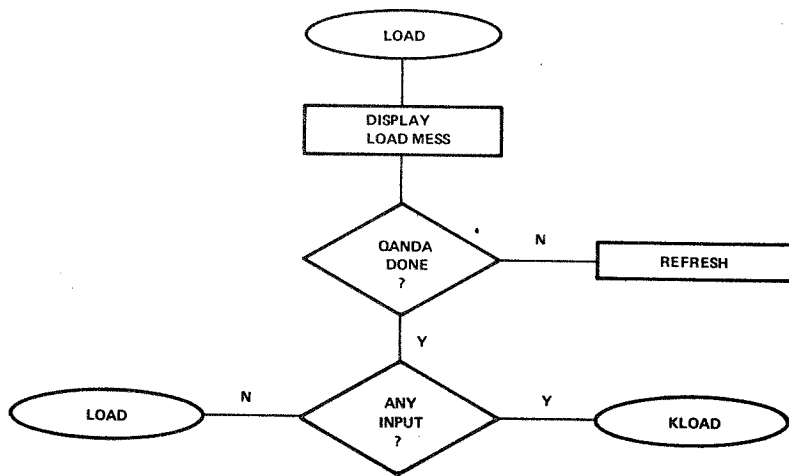
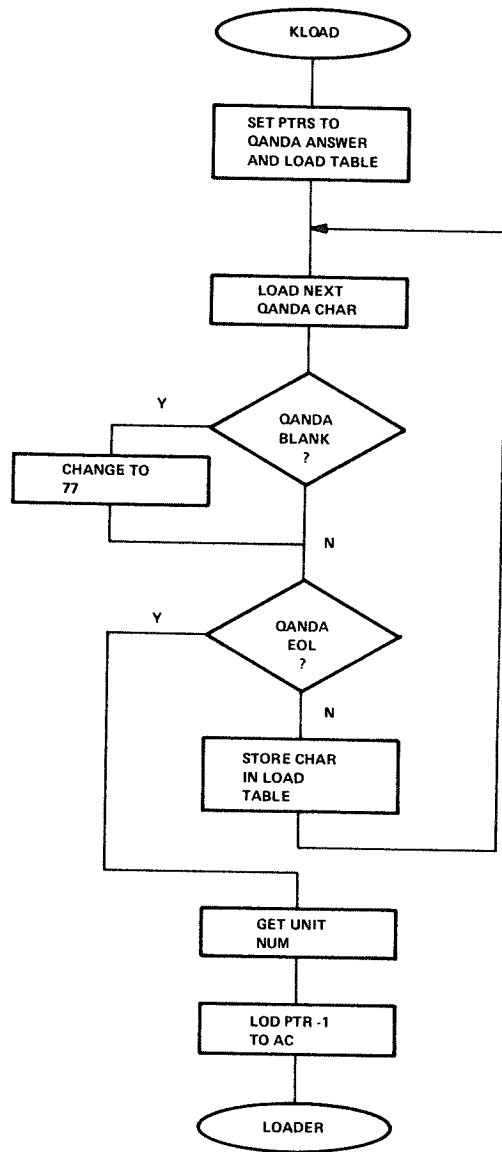
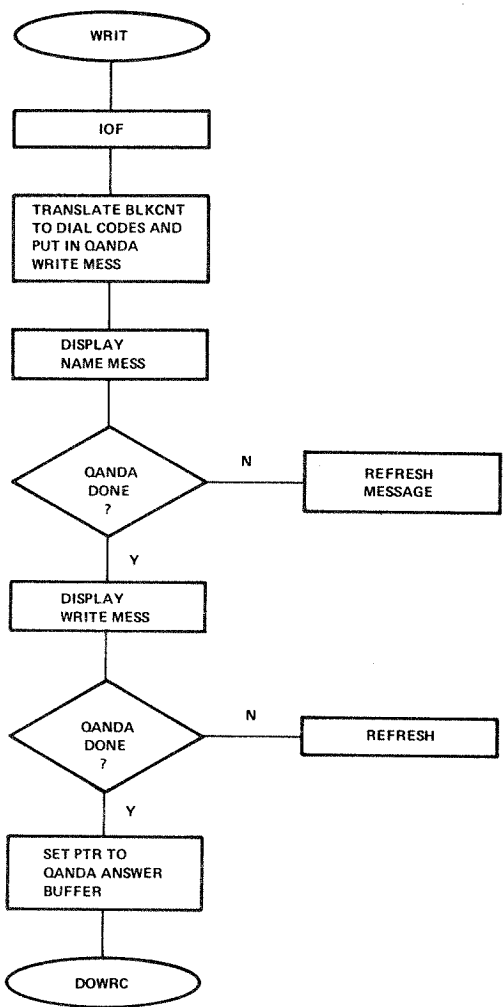


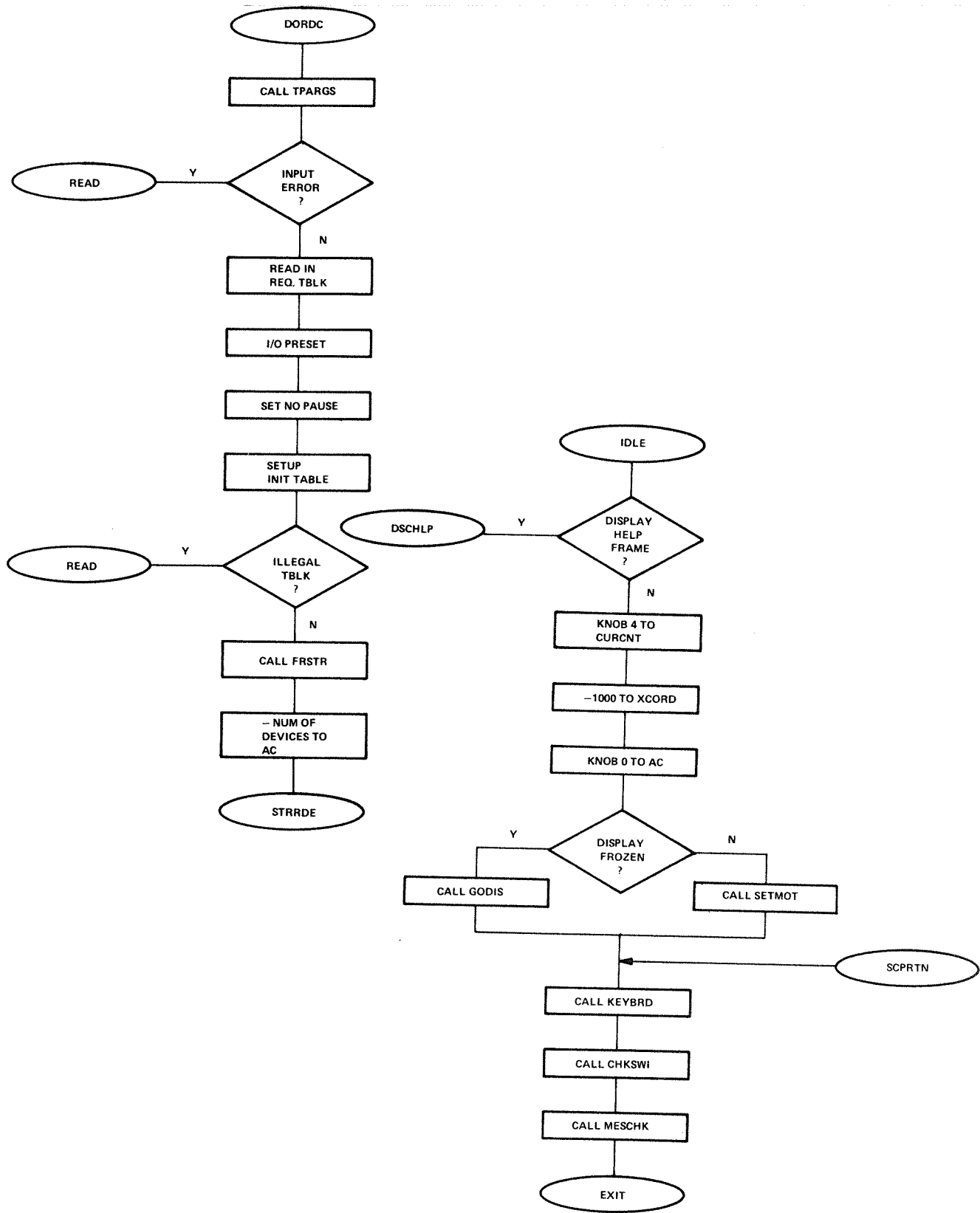


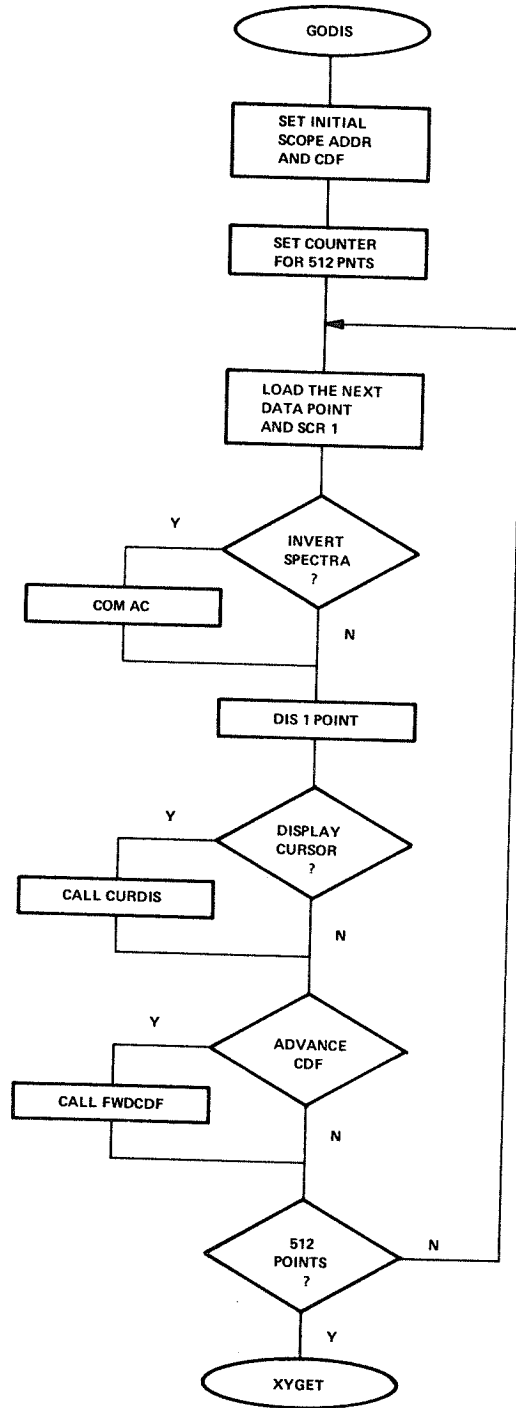
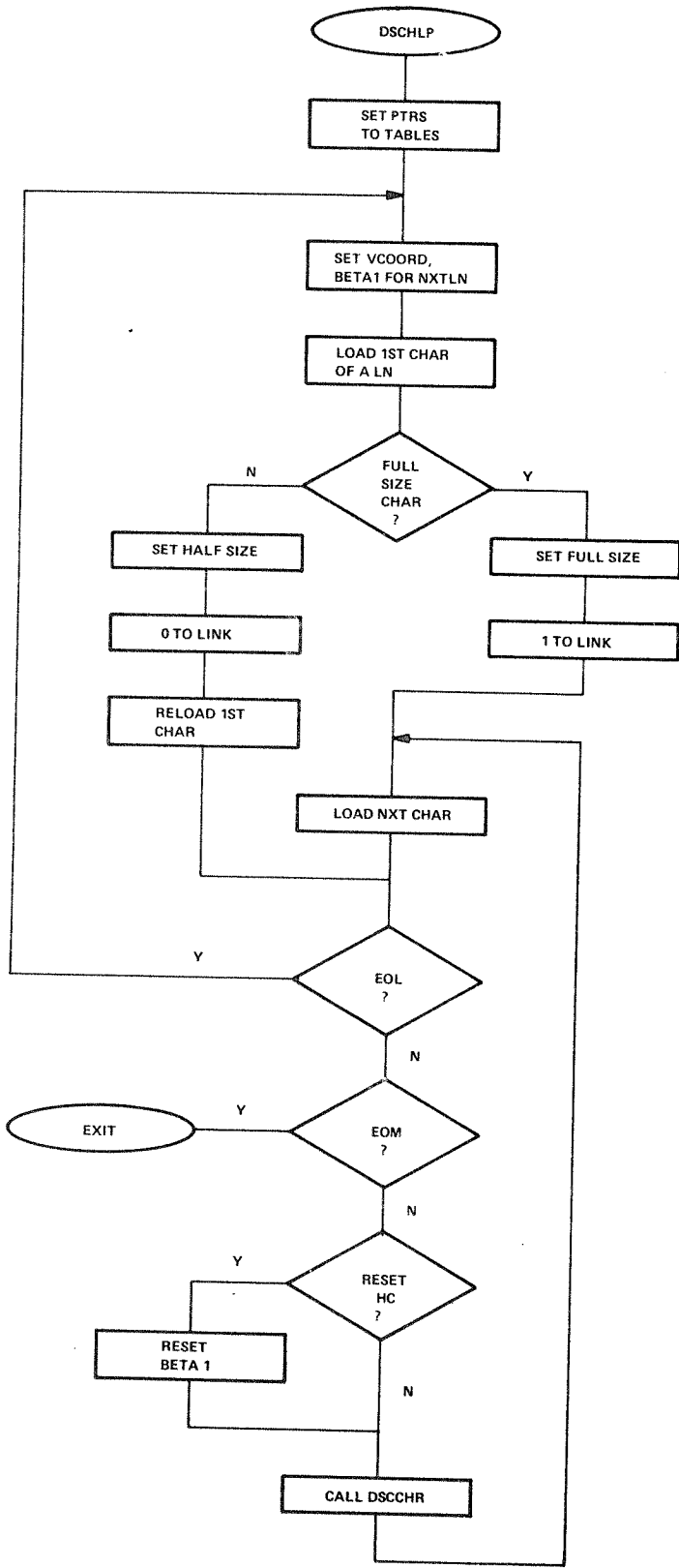


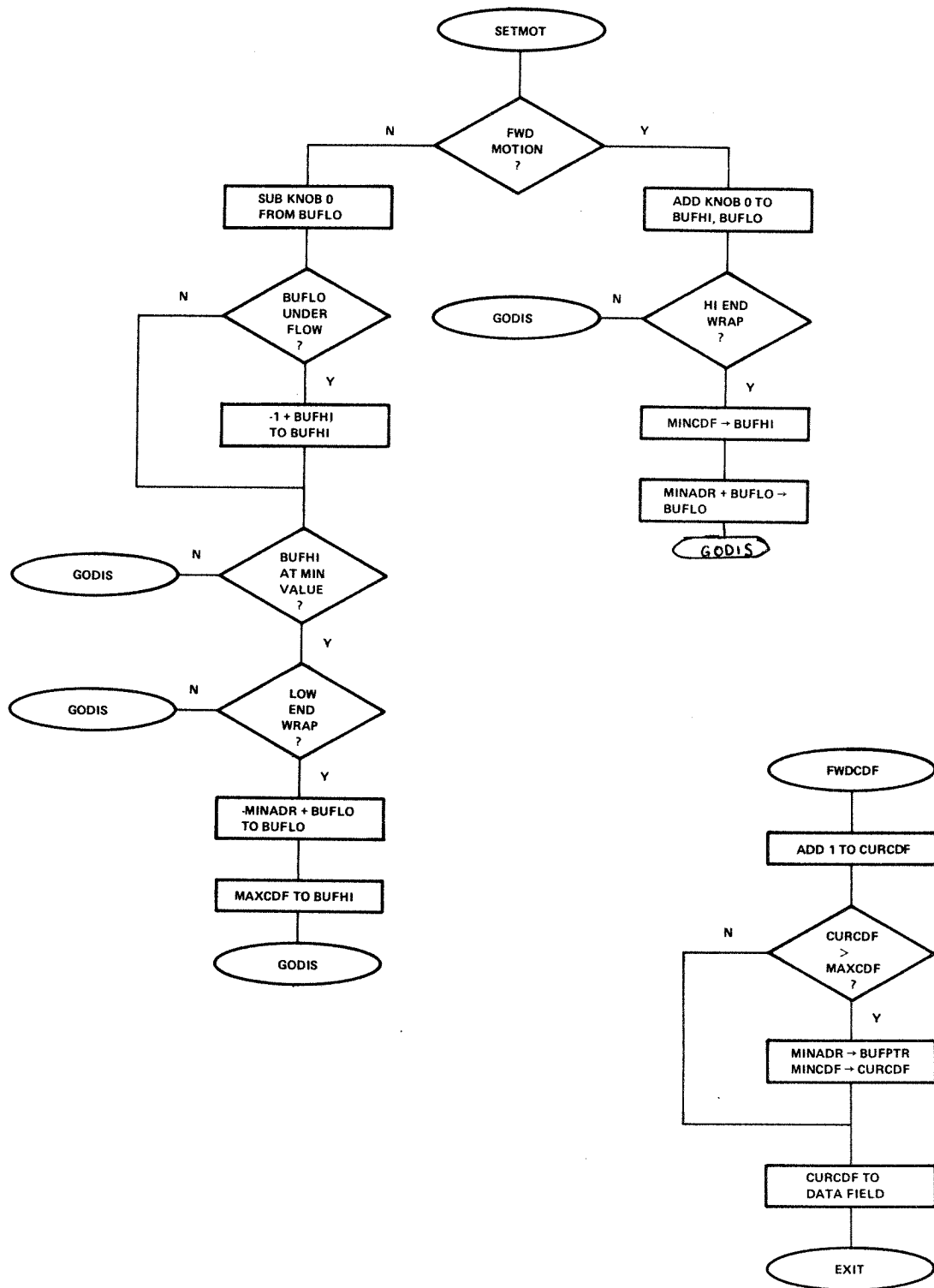


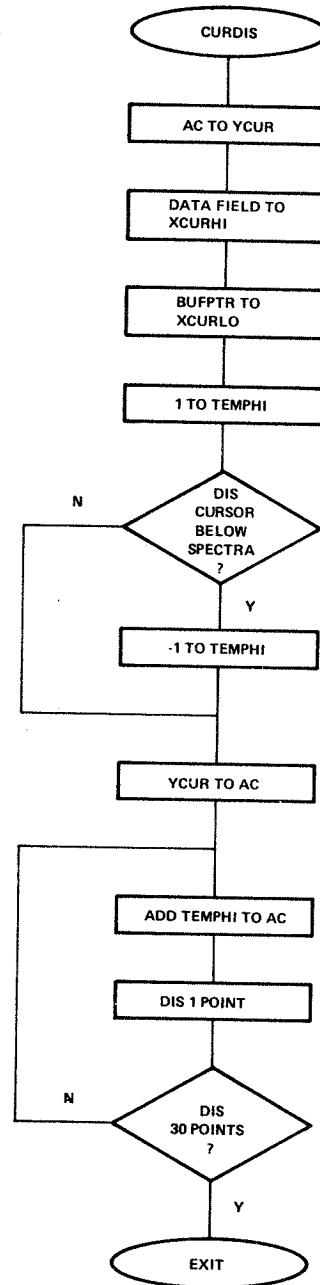
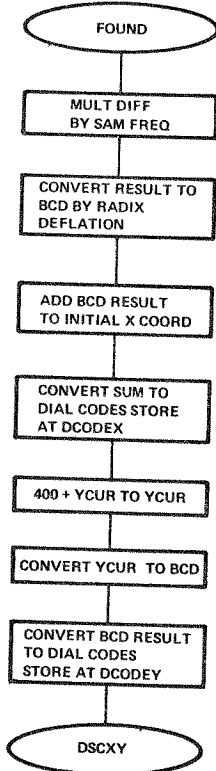
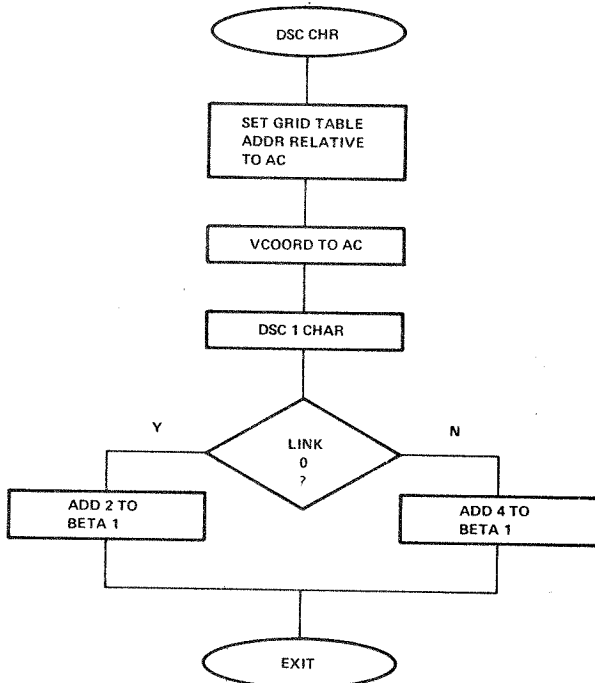


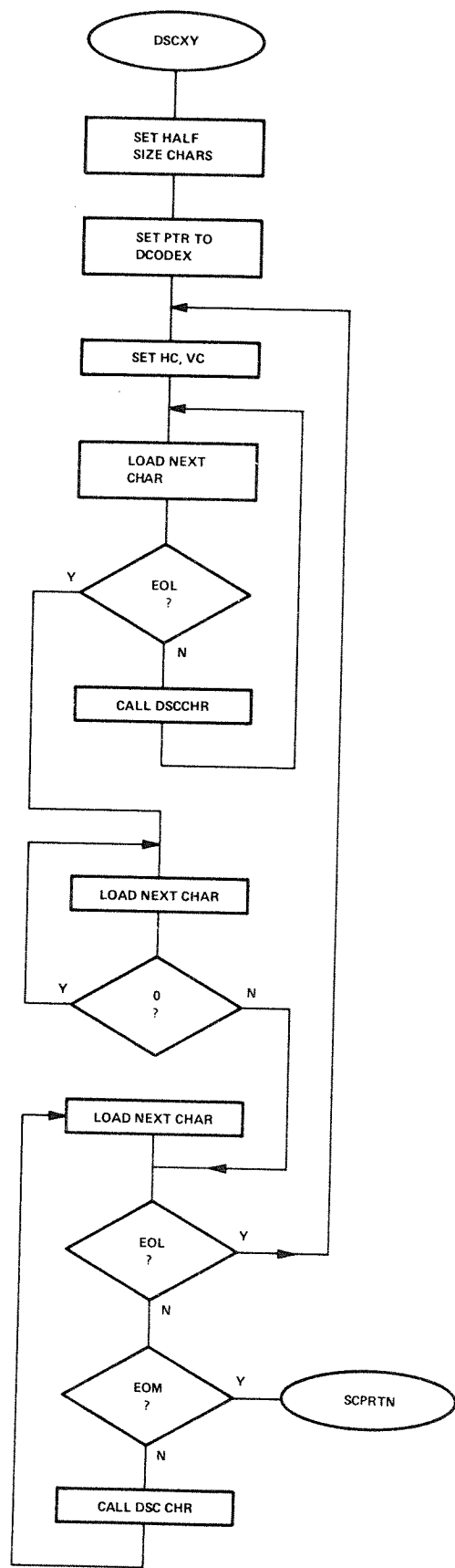
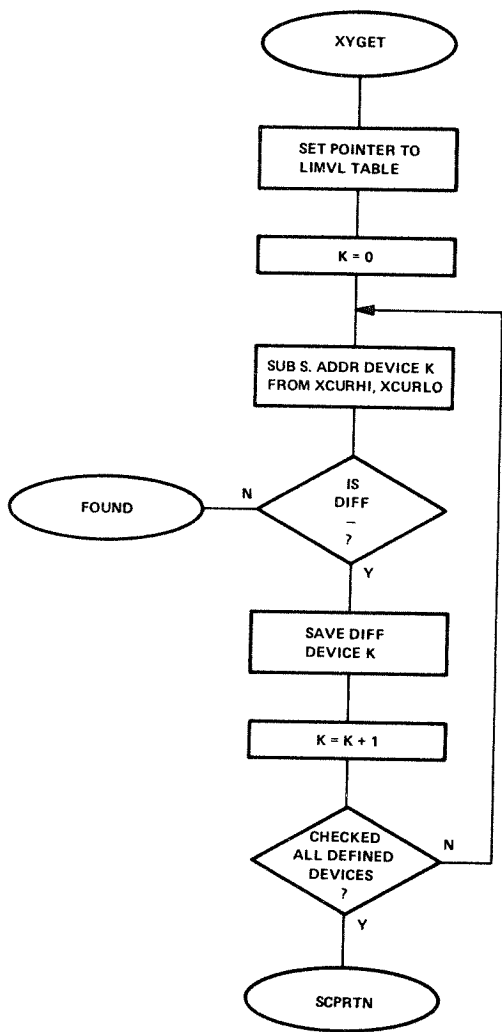


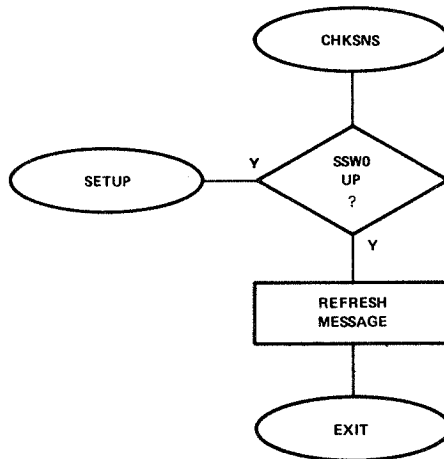
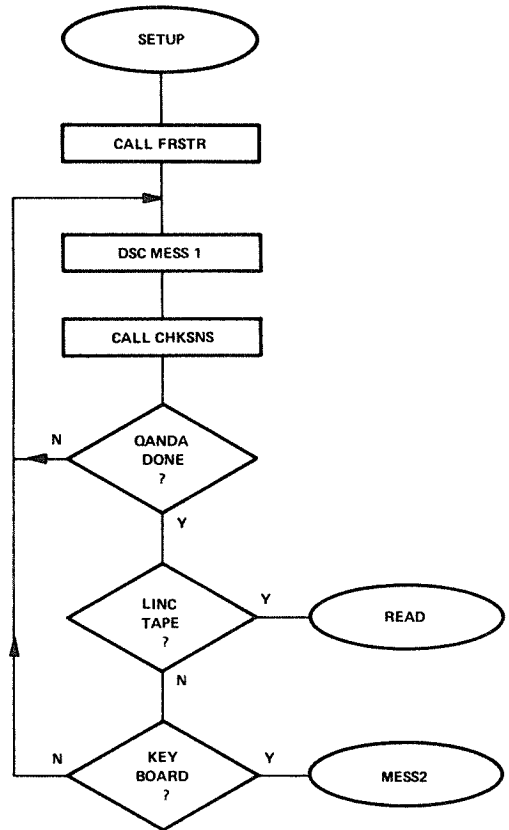
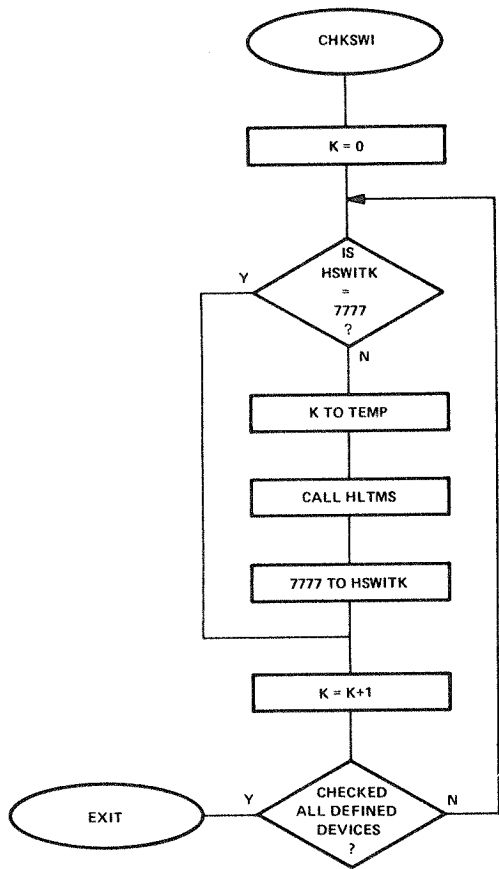


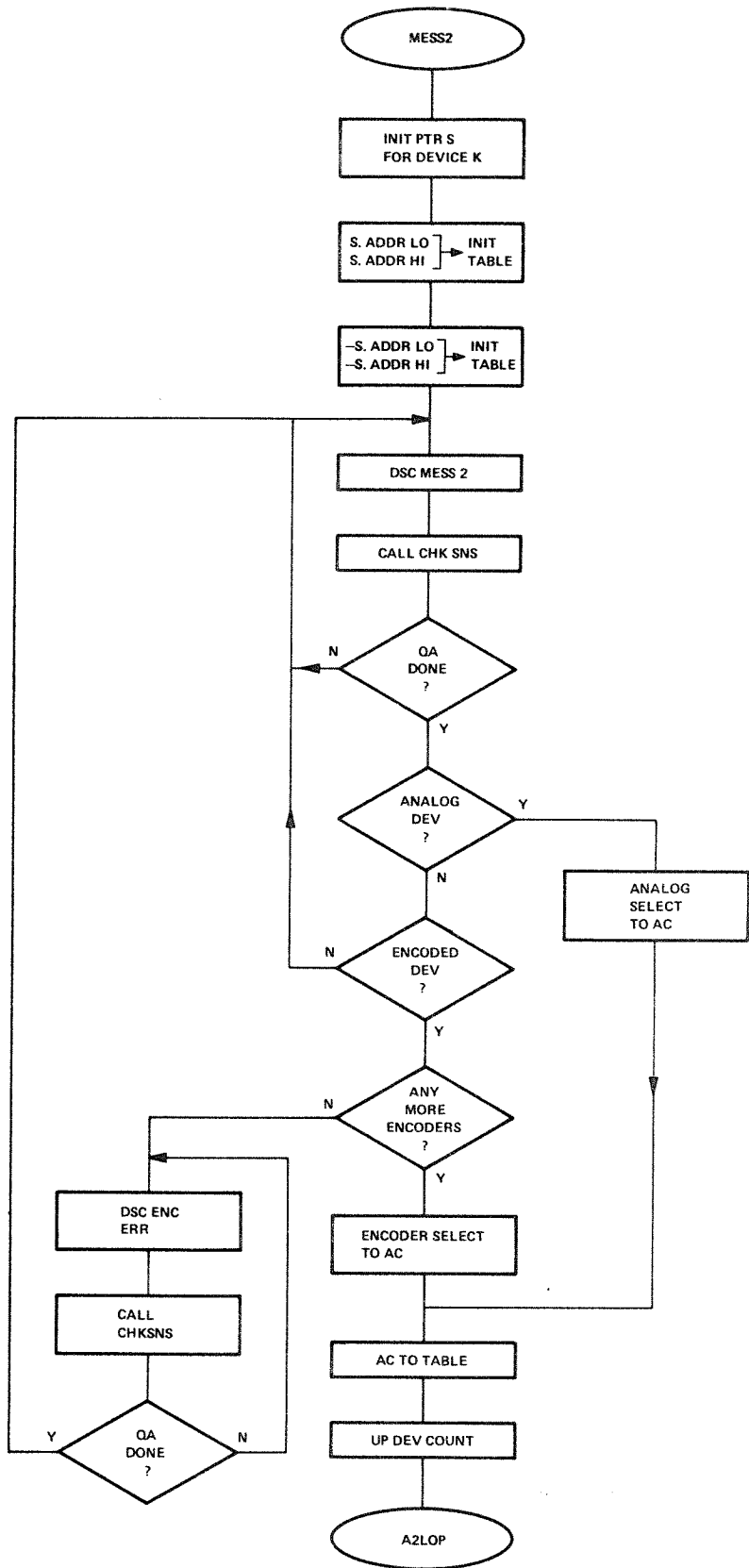
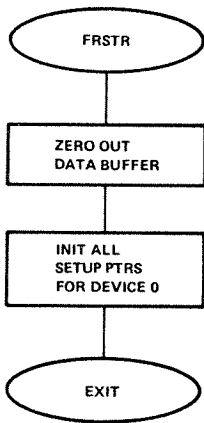
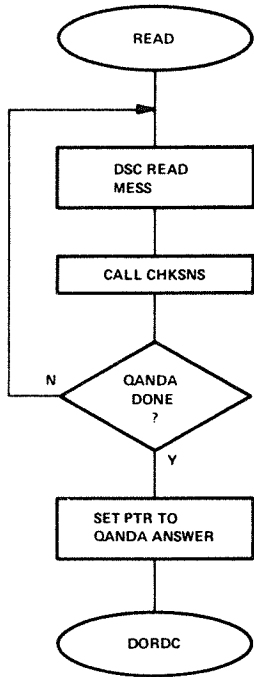


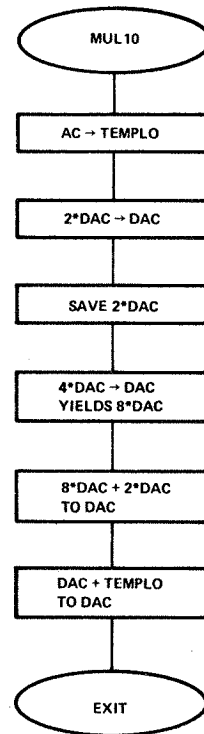
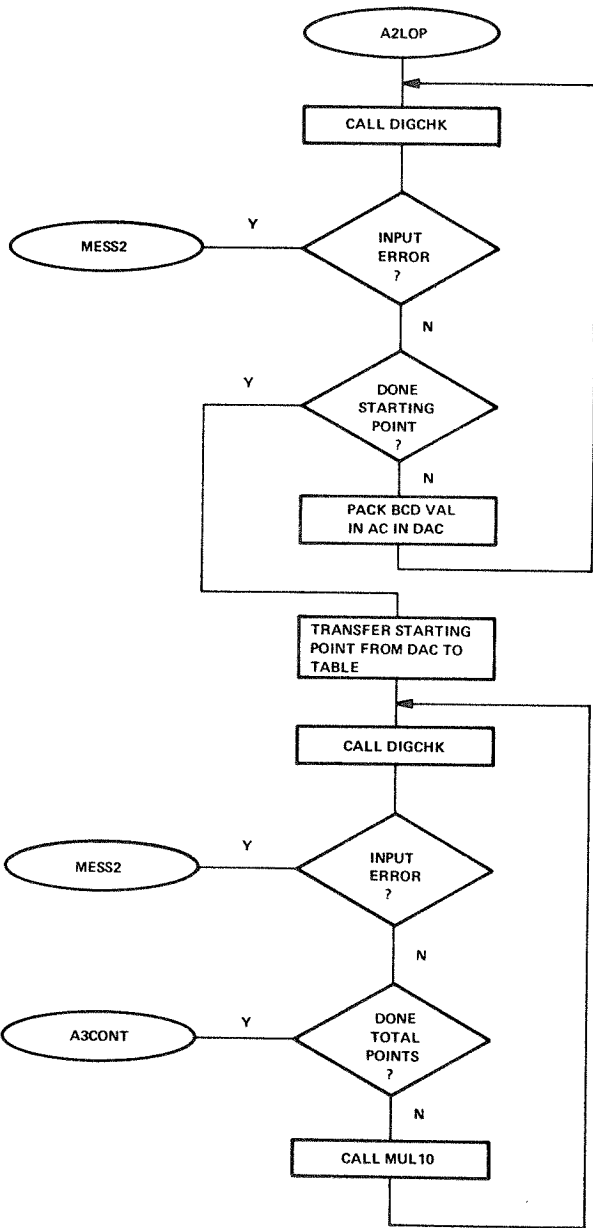


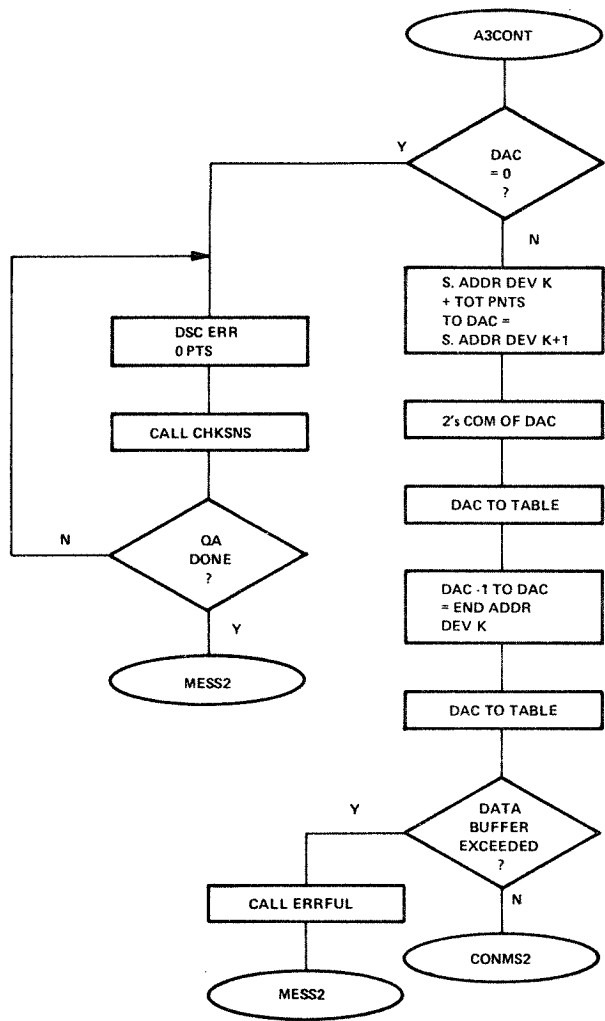


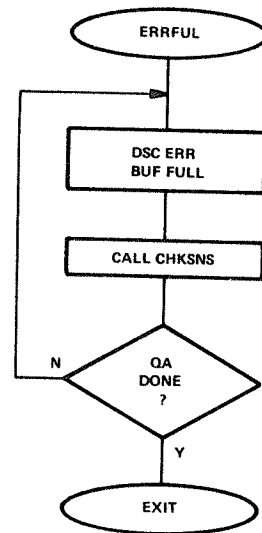
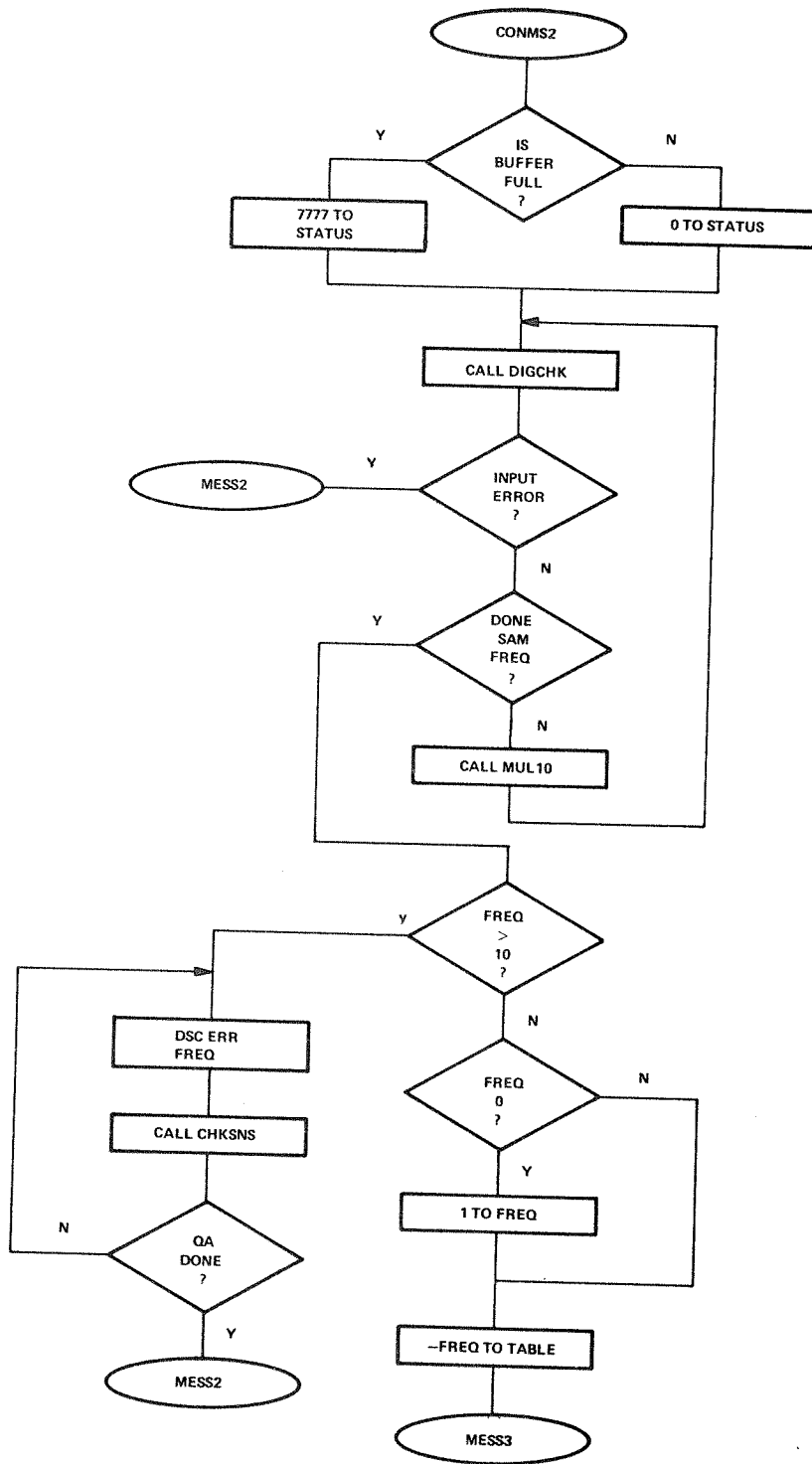


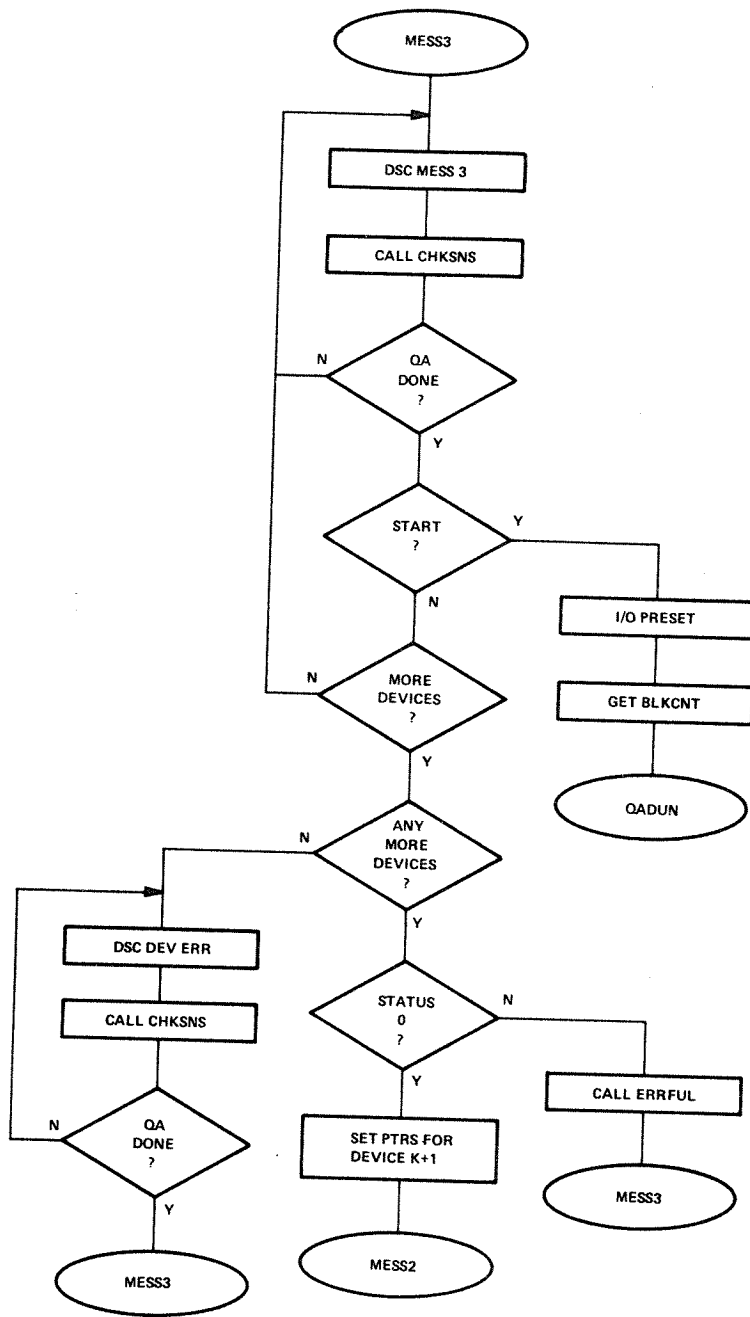


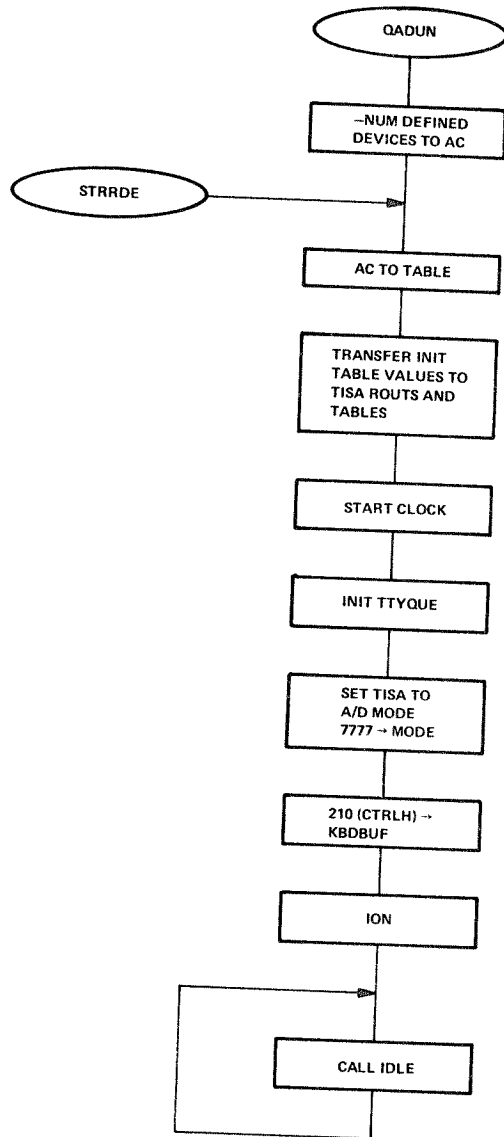
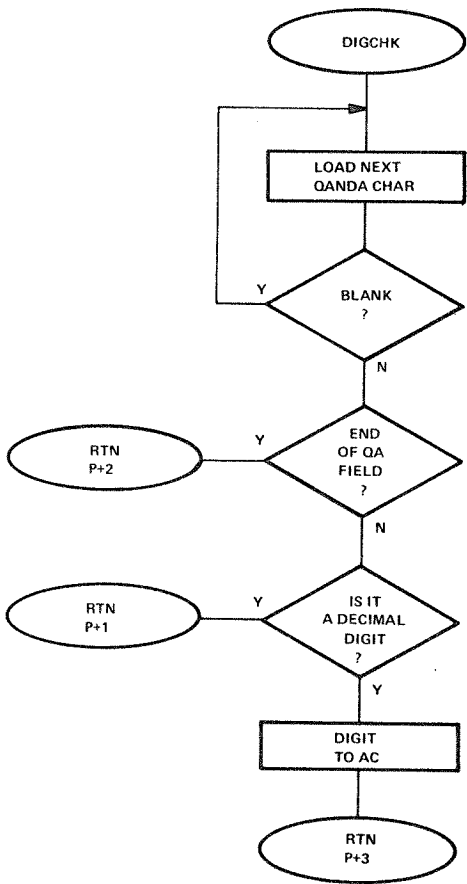












```

0000          *20
0001          /TISA (TIME INDEPENDENT
0002          /SPECTRUM ACQUISITION)
0003          /MULTIPLE DEVICES INTERRUPT DRIVEN
0004          /MOVING WINDOW DISPLAY
0005          /CURSOR AND X-Y DECIMAL READ OUT
0006          /USES QANDA FOR SET UP MODE
0007          /SUPPORTS UP TO 32K OF CORE
0010          PMODE          /INTERRUPT
0011          *1            /DISPATCH
0012          0001 5402      JMP I ,+1      /P MODE
0013          0002 0230      INTP
0014          0003 0000      BETA1, 0          /BETAS FOR LOAD
0015          0004 0000      BETA2, 0          /ROUTS
0016          0005 0000      BETA3, 0
0017          *10
0020          0010 0000      INIPTR, 0         /PTRS FOR TABLE
0021          0011 0000      ADPTR, 0         /SETUP
0022          /OF INITIAL VAL
0023          *20
0024          0020 0000      ADMESS, 0        /MESSAGE PTR
0025          /FOR TYPE OUT
0026          /ROUTINES
0027          0021 0000      KBDBUF, 0        /HOLDS LAST
0030          /KEYBOARD CHAR
0031          0022 0000      TTYFLG, 0        /HOLDS LAST
0032          /TTY CHAR
0033          /=0 IF NONE
0034          0023 0000      QUEIN, 0         /PTR TO TTY QUE
0035          /FOR IN COMING
0036          /MESSAGES
0037          0024 0000      QUEOUT, 0        /PTR FOR OUT
0040          /GOING MESS
0041          0025 0000      QUECNT, 0        /HOLDS NUMBER
0042          /OF UNTYPED
0043          /MESSAGES IN
0044          /TTY QUE
0045          0026 0000      STATUS, 0        /HOLDS CLOCK
0046          /BITS FROM CLSA
0047          0027 0000      DVTYP, 0        /TELLS WHETHER
0050          /A DEVICE IS
0051          /ANALOG OR
0052          /ENCODED
0053          /4000=ANALOG
0054          /0XXX=ENCODED
0055          0030 0600      KINPUT, INPUT    /PTR TO INPUT
0056          /ROUT
0057          0031 6201      CDF0, CDF 0
0060          0032 0000      MAXCDF, 0
0061          0033 0000      MINCDF, CDFMIN
0062          0034 7000      MINADR, ADRMIN    /BUFFER BOUNDS
0063          0035 0000      YCUR, 0          /Y COORD
0064          /OF CURSOR
0065          0036 0000      DEVNUM, 0        /-NUM OF DV
0066          /AT RUN TIME
0067          0037 2330      KDADD, DADD      /CALL TO DBL
0070          /PREC ADD
0071          *41            /L MODE
0072          0041 0002      2                /INTERRUPT
0073          0042 5443      JMP I ,+1      /DISPATCH
0074          0043 0200      INTL
0075          0044 0000      DBLLO, 0         /DBL PREC

```

0076	0045	0000	DBLHI, 0	/AC
0077	0046	0000	ARGLO, 0	/DBL PREC
0100	0047	0000	ARGHI, 0	/ARG
0101				/ALSO =
0102				/XCURLO, XCURHI
0103				/CURSOR X COORD
0104	0050	0000	TEMPLO, 0	
0105	0051	0000	TEMPHI, 0	
0106	0052	0000	CNTR, 0	
0107	0053	0000	BCNTR, 0	
0110	0054	0000	BUFPTR, 0	/FOR MOVING
0111				/WINDOW DISP
0112	0055	0000	LIMPTR, 0	/PTR TO ARGS
0113				/FOR SR XYGET
0114	0056	0000	BCDPTR, 0	/PTR TO BCD VAL
0115				/FOR SR XYGET
0116	0057	0000	RADPTR, 0	/PTR TO PWRS OF
0117				/10 FOR OCTAL
0120				/TO DECIMAL SR
0121	0060	0000	CURCDF, 0	
0122	0061	7773	MAXDVN, DVNMAX	/= -NUM OF DEV
0123				/TISA WILL
0124				/SUPPORT
0125	0062	0377	KCLEN, 0377	/GOES TO CLEN
0126				/INT ON OVFL
0127				/CHAN 1,2,3
0130	0063	4466	KDIG, DIGCHK	/CALL TO SR TO
0131				/CHK QANDA ANS
0132				/BUFER FOR
0133				/NUMERIC INPUT
0134	0064	3000	KMUL10, MUL10	/CALL TO DECIMAL
0135				/TO OCT CVERT
0136	0065	2020	KIDLE, IDLE	/CALL TO IDLE
0137				/LOOP
0140	0066	0000	SELDV0, 0	/DEVICE SELECT
0141	0067	0000	SELDV1, 0	/WORDS
0142	0070	0000	SELDV2, 0	
0143	0071	0000	SELDV3, 0	
0144	0072	0000	SELDV4, 0	
0145	0073	7777	HSWIT0, 7777	
0146	0074	7777	HSWIT1, 7777	
0147	0075	7777	HSWIT2, 7777	
0150	0076	7777	HSWIT3, 7777	
0151	0077	7777	HSWIT4, 7777	
0152	0100	0066	SELPTR, SELDV0	
0153	0101	3461	KHLTMS, HLTMS	/CAL SR TO PUT
0154				/HALT MESS IN
0155				/TTY QUE
0156	0102	3507	KGOMS, GOMS	/CALL GO MESS
0157	0103	0000	MODE, 0	/MODE WORD 0=
0160				/PAUSE 7777=A\D
0161	0104	0000	TEMP, 0	
0162	0105	1033	KQUPUT, QUPUT	/CALL SR TO PUT
0163				/A MESS IN QUE
0164	0106	6015	KINI, INISTR=1	/TOP INIT TABLE
0165	0107	6143	KBLKCT, BLKCNT	/PTR TO =NUM OF
0166				/TBLKS IN BUF
0167	0110	6144	KLAST, LAST	/ENTRY OF INIT
0170				/TABLE
0171	0111	0000	AAAP0, 0	/1ST UNUSED PG
0172				/0 CORE LOC
0173				/L MODE INTRUPT
0174	0200	3352	INTL, DCA ASAVE	

0175	0201	7010		RAR	
0176	0202	3351		DCA LSAVE	
0177	0203	1040		TAD 40	/MAKE LINC JMP
0200	0204	0355		AND P1777	
0201	0205	1354		TAD P6000	
0202	0206	3274		DCA RTNJMP	
0203	0207	6141		LINC	/SET UP LDF LIF
0204				LMODE	/FOR INTERRUPT
0205	0210	0500		IOB	/RTN
0206	0211	6234		6234	/RIB
0207	0212	1060		STA I	/SAVE IB
0210	0213	0000		0	
0211	0214	0242		ROL 2	/POSITION LDF
0212					/BITS
0213	0215	0002		PDP	
0214				PMODE	
0215	0216	0350		AND P37	
0216	0217	1347		TAD ANLDF	
0217	0220	3271		DCA RTNLDF	
0220	0221	1213		TAD ,=-6	
0221	0222	7012		RTR	/POSITION LIF
0222	0223	7010		RAR	/BITS
0223	0224	0350		AND P37	
0224	0225	1346		TAD ANLIF	
0225	0226	3270		DCA RTNLIF	
0226	0227	5234		JMP SERVE	/SERVICE INTRUP
0227	0230	3352	INTP,	DCA ASAVE	/P MODE INTRUP
0230	0231	7010		RAR	
0231	0232	3351		DCA LSAVE	
0232	0233	7040		CMA	/7777 FOR PMODE
0233					/0000 FOR LMODE
0234	0234	3345	SERVE,	DCA INTMOD	/INTERRUPT
0235	0235	6131		CLSK	/CLOCK ?
0236	0236	7410		SKP	
0237	0237	5275		JMP TISA	/YES SERVICE
0240					/DEVICES
0241	0240	7300	TIRTN,	CLA CLL	
0242	0241	6031		KSF	/KEYBOARD ?
0243	0242	5245		JMP CHKTTY	
0244	0243	6036		KRB	/YES GET CHAR
0245	0244	3021		DCA KBDBUF	
0246	0245	6041	CHKTTY,	TSF	/TTY ?
0247	0246	5256		JMP INTOUT	/NO EXIT
0250	0247	6042		TCF	/CLR FLAG
0251	0250	1420		TAD I ADDRESS	/NXT CHAR
0252	0251	7450		SNA	/MORE TO TYPE ?
0253	0252	5255		JMP ,+3	/NO
0254	0253	6046		TLS	
0255	0254	2020		ISZ ADDRESS	/SET PTR FOR
0256					/NEXT CHAR
0257	0255	3022		DCA TTYFLG	/SET TTY FLAG
0260					/0 MEANS TTY
0261					/NOT BUSY
0262	0256	1351	INTOUT,	TAD LSAVE	/RTN FROM
0263	0257	7004		RAL	/INTERRUPT
0264	0260	1352		TAD ASAVE	
0265	0261	2345		ISZ INTMOD	/P OR L MODE ?
0266	0262	5266		JMP OUTL	/L MODE
0267	0263	6244		RMF	
0270	0264	6001		ION	
0271	0265	5400		JMP I 0	
0272	0266	6141	OUTL,	LINC	
0273	0267	0006		6	/DJR

0274	0270	0000	RTNLIF, 0	
0275	0271	0000	RTNLDF, 0	
0276	0272	0500		500 /IOB
0277	0273	6001		ION
0300	0274	0000	RTNJMP, 0	
0301				/DATA
0302				/ACQUISITION
0303				/SERVICE
0304				/DEVICES
0305	0275	6135	TISA, CLSA	/SAVE
0306	0276	3026		DCA STATUS /CLOCK BITS
0307	0277	1026		TAD STATUS /CHK DV0
0310	0300	0066		AND SELDV0 /SELECT WD DV0
0311	0301	7450		SNA /READY ?
0312	0302	5321		JMP DV1 /NO GO TO DV1
0313	0303	3027		DCA DV1YP /SAVE TYPE
0314	0304	4430		JMS I KINPUT /CALL INPUT
0315				/PARAM LIST
0316				/DEVICE 0
0317	0305	0110		100 XAXDV0 /X AXIS SAM
0320	0306	0114		100 YAXDV0 /Y AXIS SAM
0321	0307	0000	SFDV0, 0	/SAM
0322				/FREQ (VAR)
0323	0310	0000	KSFDV0, 0	/SAM
0324				/FREQ (FIXED)
0325	0311	0000	HIDV0, 0	/15 BIT BUFFER
0326				/ADDR B9-B11
0327				/OF HI ORD
0330				/IS DATA FLD
0331	0312	0000	LODV0, 0	/12 BIT ADDR
0332				/WITHIN DATA
0333				/FIELD
0334	0313	0000	OSDV0, 0	/OLD
0335				/X AXIS SAM
0336	0314	0000	ELODV0, 0	/LAST
0337	0315	0000	EHIDV0, 0	/DATA ADDR
0340	0316	5321		JMP DV1 /HERE TO
0341				/CALL DV1
0342				/HERE TO HLT
0343				/DEVICE 0
0344	0317	3066	HLTDV0, DCA SELDV0	/0 DESELECTS
0345				/A DEVICE
0346	0320	3073		DCA HSWIT0 /SET HALT SWIT
0347	0321	1026	DV1, TAD STATUS	
0350	0322	0067		AND SELDV1 /SELECT WD DV1
0351	0323	7450		SNA /READY ?
0352	0324	5753		JMP I KDV2 /NO GO TO DV2
0353	0325	3027		DCA DV1YP /SAVE TYPE
0354	0326	4430		JMS I KINPUT /CALL INPUT
0355				/PARAM LIST
0356				/DEVICE 1
0357	0327	0111		100 XAXDV1 /X AXIS SAM
0360	0330	0115		100 YAXDV1 /Y AXIS SAM
0361	0331	0000	SFDV1, 0	/SAM
0362				/FREQ (VAR)
0363	0332	0000	KSFDV1, 0	/SAM
0364				/FREQ (FIXED)
0365	0333	0000	HIDV1, 0	/15 BIT BUFFER
0366				/ADDR B9-B11
0367				/OF HI ORD
0370				/IS DATA FLD
0371	0334	0000	LODV1, 0	/12 BIT ADDR
0372				/WITHIN DATA

0373					/FIELD
0374	0335	0000	OSDV1, 0		/OLD
0375					/X AXIS SAM
0376	0336	0000	ELODV1, 0		/LAST
0377	0337	0000	EHIDV1, 0		/DATA ADDR
0400	0340	5753	JMP I KDV2		/HERE TO
0401					/CALL DV2
0402					/HERE TO HLT
0403					/DEVICE 1
0404	0341	3067	HLTDV1, DCA SELDV1		/0 DESELECTS
0405					/A DEVICE
0406	0342	7001	IAC		/SET HALT SWIT
0407	0343	3074	DCA HSWIT1		
0410	0344	5753	JMP I KDV2		/NXT DEVICE
0411	0345	0000	INTMOD, 0		
0412	0346	0600	ANLIF, 600		
0413	0347	0640	ANLDF, 640		
0414	0350	0037	P37, 37		
0415	0351	0000	LSAVE, 0		
0416	0352	0000	ASAVE, 0		
0417	0353	0400	KDV2, DV2		
0420	0354	6000	P6000, 6000		
0421	0355	1777	P1777, 1777		
0422			*400		
0423	0400	1026	DV2, TAD STATUS		/CHK DV2
0424	0401	0070	AND SELDV2		/SELECT WD DV2
0425	0402	7450	SNA		/READY ?
0426	0403	5223	JMP DV3		/NO GO TO DV3
0427	0404	3027	DCA DVTYP		/SAVE TYPE
0430	0405	4430	JMS I KINPUT		/CALL INPUT
0431					/PARAM LIST
0432					/DEVICE 2
0433	0406	0112	100 XAXDV2		/X AXIS SAM
0434	0407	0116	100 YAXDV2		/Y AXIS SAM
0435	0410	0000	SFDV2, 0		/SAM
0436					/FREQ (VAR)
0437	0411	0000	KSFDV2, 0		/SAM
0440					/FREQ (FIXED)
0441	0412	0000	HIDV2, 0		/15 BIT BUFFER
0442					/ADDR B9-B11
0443					/OF HI ORD
0444					/IS DATA FLD
0445	0413	0000	LODV2, 0		/12 BIT ADDR
0446					/WITHIN DATA
0447					/FIELD
0450	0414	0000	OSDV2, 0		/OLD
0451					/X AXIS SAM
0452	0415	0000	ELODV2, 0		/LAST
0453	0416	0000	EHIDV2, 0		/DATA ADDR
0454	0417	5223	JMP DV3		/HERE TO
0455					/CALL DV3
0456					/HERE TO HLT
0457					/DEVICE 2
0460	0420	3070	HLTDV2, DCA SELDV2		/0 DESELECTS
0461					/A DEVICE
0462	0421	7105	CLL IAC RAL		/SET HALT SWIT
0463	0422	3075	DCA HSWIT2		
0464	0423	1026	DV3, TAD STATUS		
0465	0424	0071	AND SELDV3		/SELECT WD DV3
0466	0425	7450	SNA		/READY ?
0467	0426	5246	JMP DV4		/NO GO TO DEV4
0470	0427	3027	DCA DVTYP		/SAVE TYPE
0471	0430	4430	JMS I KINPUT		/CALL INPUT

0472					/PARAM LIST
0473					/DEVICE 3
0474	0431	0113		100 XAXDV3	/X AXIS SAM
0475	0432	0117		100 YAXDV3	/Y AXIS SAM
0476	0433	0000	SFDV3,	0	/SAM
0477					/FREQ (VAR)
0500	0434	0000	KSFDV3,	0	/SAM
0501					/FREQ (FIXED)
0502	0435	0000	HIDV3,	0	/15 BIT BUFFER
0503					/ADDR B9-B11
0504					/OF HI ORD
0505					/IS DATA FLD
0506	0436	0000	LODV3,	0	/12 BIT ADDR
0507					/WITHIN DATA
0510					/FIELD
0511	0437	0000	OSDV3,	0	/OLD
0512					/X AXIS SAM
0513	0440	0000	ELODV3,	0	/LAST
0514	0441	0000	EHIDV3,	0	/DATA ADDR
0515	0442	5246		JMP DV4	/HERE TO
0516					/CALL DV4
0517					/HERE TO HLT
0520					/DEVICE 3
0521	0443	3071	HLTDV3,	DCA SELDV3	/0 DESELECTS
0522					/A DEVICE
0523	0444	7125		STL IAC RAL	/SET HALT SWIT
0524	0445	3076		DCA HSWIT3	
0525	0446	1026	DV4,	TAD STATUS	
0526	0447	0072		AND SELDV4	/SELECT WD DV4
0527	0450	7450		SNA	/READY ?
0530	0451	5672		JMP I EXIT	/NO EXIT
0531	0452	3027		DCA DVTYP	/SAVE TYPE
0532	0453	4430		JMS I KINPUT	/CALL INPUT
0533					/PARAM LIST
0534					/DEVICE 4
0535	0454	0120		100 XAXDV4	/X AXIS SAM
0536	0455	0124		100 YAXDV4	/Y AXIS SAM
0537	0456	0000	SFDV4,	0	/SAM
0540					/FREQ (VAR)
0541	0457	0000	KSFDV4,	0	/SAM
0542					/FREQ (FIXED)
0543	0460	0000	HIDV4,	0	/15 BIT BUFFER
0544					/ADDR B9-B11
0545					/OF HI ORD
0546					/IS DATA FLD
0547	0461	0000	LODV4,	0	/12 BIT ADDR
0550					/WITHIN DATA
0551					/FIELD
0552	0462	0000	OSDV4,	0	/OLD
0553					/X AXIS SAM
0554	0463	0000	ELODV4,	0	/LAST
0555	0464	0000	EHIDV4,	0	/DATA ADDR
0556	0465	5672		JMP I EXIT	/HERE TO
0557					/EXIT
0560					/HERE TO HLT
0561					/DEVICE 4
0562	0466	3072	HLTDV4,	DCA SELDV4	/0 DESELECTS
0563					/A DEVICE
0564	0467	7107		CLL IAC RTL	/SET HALT SWIT
0565	0470	3077		DCA HSWIT4	
0566	0471	5672		JMP I EXIT	
0567	0472	0240	EXIT,	TIRTN	
0570	0473	6002	D,	IOF	/D KEY

0571	0474	6141		LINC	/START DIAL
0572				LMODE	
0573	0475	0001		AXO	
0574	0476	0602		LIF 2	
0575	0477	6015		JMP DIAL	
0576				Pmode	/P KEY
0577	0500	1740	P,	TAD I KPOLAR	/POLARITY WD
0600	0501	7040		CMA	/FLIP SPECTRUM
0601	0502	3740		DCA I KPOLAR	
0602	0503	1740		TAD I KPOLAR	/IS SPECTRUM
0603	0504	7640		SZA CLA	/INVERTED ?
0604	0505	5310		JMP ,+3	
0605	0506	1343		TAD NORMS	/NO NORM MESS
0606	0507	7410		SKP	/TO QUE
0607	0510	1344		TAD INVMS	/YES INVERT
0610					/MESS TO QUE
0611	0511	4505	KEYRTN,	JMS I KQUPUT	
0612	0512	5713	ZXIT,	JMP I ,+1	/CROSS PAGE
0613	0513	1203		OPKRTN	/RTN
0614	0514	1737	F,	TAD I KFREQ	/F KEY
0615	0515	7040		CMA	/COM FREEZE WD
0616	0516	3737		DCA I KFREQ	
0617	0517	1737		TAD I KFREQ	/IS SPECTRUM
0620	0520	7640		SZA CLA	/FROZEN ?
0621	0521	5324		JMP ,+3	
0622	0522	1341		TAD MOTMS	/NO MOTION
0623	0523	7410		SKP	/MESS TO QUE
0624	0524	1342		TAD FRMS	/YES FREEZE
0625	0525	5311		JMP KEYRTN	/MESS TO QUE
0626	0526	1103	S,	TAD MODE	/S KEY
0627	0527	7640		SZA CLA	/PAUSE MODE ?
0630	0530	5312		JMP ZXIT	/NO IGNORE S
0631	0531	6002		IOF	/YES CALL QA
0632	0532	5745		JMP I KSETUP	/SETUP ROUTS
0633	0533	1103	W,	TAD MODE	/W KEY
0634	0534	7640		SZA CLA	/PAUSE MODE ?
0635	0535	5312		JMP ZXIT	/NO IGNORE W
0636	0536	5746		JMP I KWRIT	/YES CALL
0637					/WRITE ROUTS
0640	0537	2040	KFREQ,	FREEZE	
0641	0540	2266	KPOLAR,	POLAR	
0642	0541	6613	MOTMS,	MSMOT	
0643	0542	6624	FRMS,	MSFR	
0644	0543	6635	NORMS,	MSNOR	
0645	0544	6646	INVMS,	MSINV	
0646	0545	4023	KSETUP,	SETUP	
0647	0546	4400	KWRIT,	WRIT	
0650				*600	
0651	0600	0000	INPUT,	0	/GET INPUT FROM
0652					/DEVICE K
0653					/GET PARAMETER
0654	0601	1600		TAD I INPUT	/LIST FROM CALL
0655	0602	3303		DCA SAMX	/X AXIS SAM
0656	0603	2200		ISZ INPUT	
0657	0604	1600		TAD I INPUT	
0660	0605	3254		DCA SAMY	
0661	0606	2200		ISZ INPUT	
0662	0607	1200		TAD INPUT	/PTR TO
0663	0610	3335		DCA FREQ	/SAMPLING FREQ
0664					/LAST COUNT
0665					/(VAR)
0666	0611	2200		ISZ INPUT	
0667	0612	1200		TAD INPUT	/PTR TO

0670	0613	3334	DCA	KFREQ	/SAM FREQ
0671					/(FIXED)
0672	0614	2200	ISZ	INPUT	/PTR TO
0673	0615	1200	TAD	INPUT	/BUFFER
0674	0616	3331	DCA	CDFVAL	/CDF BITS
0675	0617	2200	ISZ	INPUT	/PTR TO
0676	0620	1200	TAD	INPUT	/BUFFER
0677	0621	3330	DCA	ADDR	/ADDR
0700	0622	2200	ISZ	INPUT	
0701	0623	1200	TAD	INPUT	
0702	0624	3327	DCA	OLDSAM	/PTR TO OLDSAM
0703	0625	2200	ISZ	INPUT	
0704	0626	1600	TAD	I INPUT	/PTR TO LAST
0705	0627	3332	DCA	ENDLO	/DATA ADDR
0706	0630	2200	ISZ	INPUT	
0707	0631	1600	TAD	I INPUT	
0710	0632	3333	DCA	ENDHI	
0711	0633	2200	ISZ	INPUT	/FIRST RTN IS
0712					/JMP TO DV K+1
0713					/2ND IS HLT DVK
0714	0634	1027	TAD	DVTYP	/ANALOG
0715	0635	7710	SPA	CLA	/OR ENCODED ?
0716	0636	5302	JMP	ANALOG	
0717	0637	2735	ANARTN, ISZ	I FREQ	/READY TO SAM
0720					/Y AXIS ?
0721	0640	5600	JMP	I INPUT	/NO GO TO
0722					/DV K+1
0723	0641	1734	TAD	I KFREQ	/RESET FREQ
0724	0642	3735	DCA	I FREQ	/FOR NXT PASS
0725	0643	1730	TAD	I ADDR	
0726	0644	3336	DCA	HOLD	/BUFFER ADDR
0727	0645	1731	TAD	I CDFVAL	/SETUP CDF
0730	0646	7104	CLL	RAL	
0731	0647	7006	RTL		
0732	0650	1031	TAD	CONF	
0733	0651	3252	DCA	,+1	
0734	0652	0000	0		/CDF N
0735	0653	6141	LINC		
0736	0654	0000	SAMY, 0		/SAM Y AXIS
0737	0655	0002	2		/PDP
0740	0656	3736	DCA	I HOLD	/PUT IN DATA
0741					/BUFFER
0742	0657	6201	CDF	0	/RESTORE FLD 0
0743	0660	1332	TAD	ENDLO	/DVK IS FINI
0744					/WHEN CURRENT
0745	0661	1730	TAD	I ADDR	/ADDR
0746	0662	7640	SZA	CLA	/(CDFVAL, ADDR)
0747	0663	5273	JMP	NFDVK	
0750	0664	7004	RAL		
0751	0665	1333	TAD	ENDHI	/=TERMINAL ADDR
0752					/(ENDHI,
0753	0666	1731	TAD	I CDFVAL	/ENDLO)
0754	0667	7640	SZA	CLA	
0755	0670	5273	JMP	NFDVK	
0756	0671	2200	ISZ	INPUT	/FINI DV K
0757	0672	5600	JMP	I INPUT	
0760	0673	7101	NFDVK, CLL	IAC	/ADV CURRENT
0761	0674	1730	TAD	I ADDR	/ADDR
0762	0675	3730	DCA	I ADDR	
0763	0676	7004	RAL		
0764	0677	1731	TAD	I CDFVAL	
0765	0700	3731	DCA	I CDFVAL	
0766	0701	5600	JMP	I INPUT	/NORMAL RTN

0767	0702	6141	ANALOG, LINC		
0770	0703	0000	SAMX, 0		/SAM X AXIS
0771	0704	0002	2		/PDP
0772	0705	1326	TAD P1000		/MAKE POS
0773	0706	3325	DCA NEWSAM		
0774	0707	1727	TAD I OLDSAM		/CHK FOR 1 BIT
0775	0710	7041	CIA		/INCREASE
0776	0711	1325	TAD NEWSAM		
0777	0712	7540	SMA SZA		/X AXIS READY ?
1000	0713	5320	JMP CHGSAM		/YES
1001	0714	1324	TAD MAXDRP		/CHK FOR
1002	0715	7710	SPA CLA		/TOO FAR ?
1003	0716	2200	ISZ INPUT		/YES ADV TO
1004					/HALT RTN
1005	0717	5600	JMP I INPUT		/NO GO TO DVK+1
1006	0720	7200	CHGSAM, CLA		/UPDATE
1007	0721	1325	TAD NEWSAM		/OLDSAM
1010	0722	3727	DCA I OLDSAM		
1011	0723	5237	JMP ANARTN		/NOW CHK Y
1012					/AXIS
1013	0724	0020	MAXDRP, DRPMAX		/MAX X AX DROP
1014	0725	0000	NEWSAM, 0		
1015	0726	1000	P1000, 1000		
1016	0727	0000	OLDSAM, 0		
1017	0730	0000	ADDR, 0		
1020	0731	0000	CDFVAL, 0		
1021	0732	0000	ENDLO, 0		
1022	0733	0000	ENDHI, 0		
1023	0734	0000	KFREQ, 0		
1024	0735	0000	FREQ, 0		
1025	0736	0000	HOLD, 0		
1026			*1000		
1027	1000	0000	MESCHK, 0		/TTY CHK
1030	1001	1025	TAD QUECNT		/IS QUE EMPTY ?
1031	1002	7650	SNA CLA		
1032	1003	5600	JMP I MESCHK		/YES XIT
1033	1004	1420	TAD I ADMESS		/NO IS A MESS
1034	1005	7640	SZA CLA		/IN PROCESS ?
1035	1006	5600	JMP I MESCHK		/YES XIT
1036	1007	1022	TAD TTYFLG		/NO IS A CHAR
1037	1010	7640	SZA CLA		/IN PROCESS ?
1040	1011	5600	JMP I MESCHK		/YES XIT
1041	1012	1424	TAD I QUEOUT		/NO NEXT MESS
1042	1013	3020	DCA ADMESS		/TO BATTERS
1043					/BOX
1044	1014	1024	TAD QUEOUT		/ADV QUEOUT TO
1045	1015	2024	ISZ QUEOUT		/NEXT MESS
1046	1016	1265	TAD ENQUE		/IF AT END OF
1047	1017	7640	SZA CLA		/QUE ADV TO GO
1050	1020	5223	JMP ,+3		/DO NOT COLLECT
1051					/\$200
1052	1021	1264	TAD STQUE		
1053	1022	3024	DCA QUEOUT		
1054	1023	7240	STA		/BACK UP COUNT
1055	1024	1025	TAD QUECNT		
1056	1025	3025	DCA QUECNT		
1057	1026	1420	TAD I ADMESS		/KICK OFF 1ST
1060	1027	2020	ISZ ADMESS		/CHAR
1061	1030	6046	TL5		/INTERRUPTS
1062					/WILL CARRY THE
1063					/BALL FROM HERE
1064	1031	3022	DCA TTYFLG		/SET FLAG TTY
1065					/IS BUSY

1066	1032	5600		JMP I MESCHK	
1067					/INPUT MESS
1070	1033	0000	QUPUT,	0	/TO QUE
1071	1034	3266		DCA QSAVE	/SAVE ADDR
1072	1035	1025		TAD QUECNT	/CHK FOR QUE
1073	1036	1262		TAD QUEMAX	/FULL
1074	1037	7450		SNA	/1 SLOT LEFT ?
1075	1040	5257		JMP TBUSY	/YES INPUT BUSY
1076					/MESS
1077	1041	7700		SMA CLA	/FULL ?
1100	1042	5633		JMP I QUPUT	/YES INHIBIT MS
1101	1043	1266		TAD QSAVE	/NO INPUT MESS
1102	1044	3423	INMESS,	DCA I QUEIN	/TO QUE
1103	1045	1023		TAD QUEIN	/IS QUEIN AT
1104	1046	1265		TAD ENQUE	/GO ?
1105	1047	7640		SZA CLA	
1106	1050	5254		JMP ,+4	
1107	1051	1264		TAD STQUE	/YES COLLECT
1110	1052	3023		DCA QUEIN	/\$200
1111	1053	7410		SKP	
1112	1054	2023		ISZ QUEIN	/ADV PTR
1113	1055	2025		ISZ QUECNT	/ADV COUNT
1114	1056	5633		JMP I QUPUT	
1115	1057	7200	TBUSY,	CLA	/PUT BUSY
1116	1060	1263		TAD BUSYMS	/MESS IN QUE
1117	1061	5244		JMP INMESS	
1120	1062	7755	QUEMAX,	STRQUE=ENDQUE	/=MAX NUM OF
1121					/MESS+1
1122	1063	6676	BUSYMS,	MSBUSY	/ADDR OF BUSY
1123					/MESS
1124	1064	6373	STQUE,	STRQUE	
1125	1065	1362	ENQUE,	-ENDQUE	
1126	1066	0000	QSAVE,	0	
1127	1067	0000	DOMTP,	0	/WRITE TAPE
1130	1070	1507		TAD I KBLKCT	/=NUM TO DO
1131	1071	3266		DCA QSAVE	
1132	1072	1047		TAD ARGHI	
1133	1073	3302		DCA TPINST	
1134	1074	1046		TAD ARGLO	
1135	1075	3303		DCA TPWORD	
1136	1076	1372		TAD LDFSTR	
1137	1077	3301		DCA TPLDF	
1140	1100	6141	MTPLOP,	LINC	
1141				LMODE	
1142	1101	0000	TPLDF,	0	/LDF N
1143	1102	0000	TPINST,	0	/MTP INST
1144	1103	0000	TPWORD,	0	/MBLK\TBLK
1145	1104	7163		JMP TPWAIT	
1146	1105	0011		CLR	/LINK BIT
1147	1106	1020		LDA I	/ADV TBLK, MBLK
1150	1107	1001		1001	
1151	1110	1200		LAM	
1152	1111	1103		TPWORD	
1153	1112	1560		BCL I	
1154	1113	7000	P7K,	7000	
1155	1114	0450		AZE	/TBLK WRAP ?
1156	1115	7124		JMP GOODY	/NO
1157	1116	0011		CLR	/LINK
1160	1117	3113		ADD P7K	/BU MBLK
1161	1120	1200		LAM	
1162	1121	1103		TPWORD	
1163	1122	0450		AZE	/END OF DF ?
1164	1123	7137		JMP BYMBLK	/NO

1165	1124	0472	GOODY,	LZE I	/MBLK WRAP
1166					/AROUND ?
1167	1125	7137		JMP BYMBLK	/NO
1170	1126	1000		LDA	/YES RESET
1171	1127	1103		TPWORD	/MBLK
1172	1130	1620		BSE I	
1173	1131	4000		4000	
1174	1132	5103		STC TPWORD	
1175	1133	3101		ADD TPLDF	/ADV LDF
1176	1134	1120		ADA I	
1177	1135	0001		1	
1200	1136	5101		STC TPLDF	
1201	1137	0002	BYMBLK,	PDP	
1202				Pmode	
1203	1140	2266		ISZ QSAVE	/DONE ?
1204	1141	5300		JMP MTPLOP	/NO
1205	1142	6141		LINC	
1206				Lmode	
1207	1143	1000		LDA	
1210	1144	1102		TPINST	
1211	1145	1560		BCL I	
1212	1146	7767		7767	/U BIT
1213	1147	1120		ADA I	
1214	1150	0707		CHK	
1215	1151	5152		STC ,+1	
1216	1152	0000		0	/STOP TAPE
1217	1153	0270		270	
1220	1154	7163		JMP TPWAIT	
1221	1155	0004		ESF	
1222	1156	1020		LDA I	/AND CLR XOB
1223	1157	0010		EXTMTP	/U BITS
1224	1160	0001		AXO	
1225	1161	0002		PDP	
1226				Pmode	
1227	1162	5667		JMP I DOMTP	
1230				Lmode	
1231	1163	0002	TPWAIT,	PDP	
1232				Pmode	
1233	1164	4465		JMS I KIDLE	
1234	1165	6141		LINC	
1235				Lmode	
1236	1166	0006		DJR	
1237	1167	0416		STD	
1240	1170	7163		JMP TPWAIT	
1241	1171	6000		JMP 0	
1242				Pmode	
1243	1172	0643	LDFSTR,	FIRLDF	
1244				PAGE	
1245	1200	0000	KEYBRD,	0	/KEYBOARD SR
1246	1201	1021		TAD KBDBUF	/GOT A CHAR ?
1247	1202	7450		SNA	
1250	1203	5600	OPKRTN,	JMP I KEYBRD	/NO XIT
1251	1204	3265		DCA CHAR	/SAVE
1252	1205	3021		DCA KBDBUF	/CLR BUF
1253	1206	1223		TAD CHR TAB	/START OF TABLE
1254	1207	3224		DCA CHR PTR	
1255	1210	1624	CHRLOP,	TAD I CHR PTR	/SEARCH FOR
1256	1211	7041		CIA	/MATCH
1257	1212	1265		TAD CHAR	
1260	1213	2224		ISZ CHR PTR	/ADV PAST JMP
1261	1214	7650		SNA CLA	
1262	1215	5220		JMP MATCH	
1263	1216	2224		ISZ CHR PTR	/ADV TO NXT CHR

1264	1217	5210		JMP CHRLOP	
1265	1220	1624	MATCH,	TAD I CHRPTR	/LOAD A JMP
1266	1221	3222		DCA ,+1	/SERVE VIA
1267	1222	0000		0	/JMP INST
1270	1223	1225	CHRTAB,	KBDTAB	
1271	1224	0000	CHRPTR,	0	
1272				/KEYBOARD LOOKUP TABLE FORMAT IS:	
1273				/LOCA, ASCII CODE OF DEFINED CHAR	
1274				/LOCA+1, JMP TO ROUT FOR CHAR	
1275				/THE LOC CHAR IS IN TABLE SO IF NONE	
1276				/MATCH THE DEFAULT CASE IS JMP I KEYBRD	
1277	1225	0304	KBDTAB,	304	/D
1300	1226	5671		JMP I KD	/START DIAL
1301	1227	0210		210	/CNTRL\H
1302	1230	5274		JMP CTRLH	/HLT ALL DVS
1303	1231	0310		310	/H
1304	1232	5311		JMP H	/HALT
1305	1233	0320		320	/P
1306	1234	5667		JMP I KP	/POLARITY
1307	1235	0306		306	/F
1310	1236	5670		JMP I KF	/FREEZE
1311	1237	0307		307	/G
1312	1240	5315		JMP G	/GO
1313	1241	0323		323	/S
1314	1242	5672		JMP I KS	/SETUP
1315	1243	0327		327	/W
1316	1244	5673		JMP I KW	/WRITE TAPE
1317	1245	0303		303	/C
1320	1246	5342		JMP C	/CALL CATALCAL
1321	1247	0315		315	/M
1322	1250	5347		JMP M	/CALL MAGSPY
1323	1251	0314		314	/L
1324	1252	5354		JMP L	/CALL LOADER
1325					/NUMERIC ARGS
1326					/FOR H AND G
1327	1253	0260		260	/0
1330	1254	5321		JMP NUM	
1331	1255	0261		261	/1
1332	1256	5321		JMP NUM	
1333	1257	0262		262	
1334	1260	5321		JMP NUM	
1335	1261	0263		263	
1336	1262	5321		JMP NUM	
1337	1263	0264		264	
1340	1264	5321		JMP NUM	
1341	1265	0000	CHAR,	0	/DEFAULT CASE
1342	1266	5600		JMP I KEYBRD	
1343	1267	0500	KP,	P	
1344	1270	0514	KF,	F	
1345	1271	0473	KD,	D	
1346	1272	0526	KS,	S	
1347	1273	0533	KW,	W	
1350					/CTRLH KEY
1351	1274	1103	CTRLH,	TAD MODE	/=0 FOR PAUSE
1352					/=7777 FOR A\D
1353	1275	7650		SNA CLA	/PAUSE MODE ?
1354	1276	5600		JMP I KEYBRD	/YES IGNORE
1355					/CTRLH
1356	1277	3103		DCA MODE	/SET MODE TO
1357					/PAUSE
1360	1300	6134		CLEN	/DISABLE ALL
1361					/CLOCK
1362					/INTERRUPTS

1363	1301	3066		DCA SELDV0	
1364	1302	3067		DCA SELDV1	/DESELECT
1365	1303	3070		DCA SELDV2	/ALL DEVICES
1366	1304	3071		DCA SELDV3	
1367	1305	3072		DCA SELDV4	
1370	1306	1363		TAD HLTALL	/PUT MESS
1371	1307	4505		JMS I KQUPUT	/IN QUE
1372	1310	5600		JMP I KEYBRD	
1373					/H AND G SET A
1374					/SWITCH TO WAIT
1375					/FOR A 2ND CHAR
1376	1311	7040	H,	CMA	/SET FOR 2ND
1377	1312	3364		DCA HSWIT	/H AND G CANNOT
1400	1313	3365		DCA GSWIT	/BOTH BE SET
1401	1314	5600		JMP I KEYBRD	
1402	1315	7040	G,	CMA	/SET FOR 2ND
1403	1316	3365		DCA GSWIT	
1404	1317	3364		DCA HSWIT	
1405	1320	5600		JMP I KEYBRD	
1406	1321	1265	NUM,	TAD CHAR	/NUM KEY
1407	1322	0367		AND PP17	/STRIP ASCII
1410	1323	3104		DCA TEMP	
1411	1324	1104		TAD TEMP	/CHK FOR
1412	1325	1036		TAD DEVNUM	/UNDEFINED
1413	1326	7700		SMA CLA	/DEV NUM
1414	1327	5600		JMP I KEYBRD	/ILLEGAL XIT
1415	1330	1364		TAD HSWIT	/IS 2ND ARG
1416	1331	7650		SNA CLA	/FOR H ?
1417	1332	5335		JMP ,+3	/NO CHK G
1420	1333	3364		DCA HSWIT	/YES CLR SWIT
1421	1334	5766		JMP I KHALT	/GO TO HALT SR
1422	1335	1365		TAD GSWIT	/IS IT G ?
1423	1336	7650		SNA CLA	
1424	1337	5600		JMP I KEYBRD	/NO XIT
1425	1340	3365		DCA GSWIT	/YES CLR SWIT
1426	1341	5770		JMP I KGO	/GO TO GO SR
1427	1342	1103	C,	TAD MODE	/C KEY
1430	1343	7640		SZA CLA	/PAUSE MODE ?
1431	1344	5600		JMP I KEYBRD	/NO IGNORE C
1432	1345	6141		LINC	/YES LOAD
1433	1346	7517		6000+CATA	/CATACAL
1434	1347	1103	M,	TAD MODE	/M KEY
1435	1350	7640		SZA CLA	/PAUSE MODE ?
1436	1351	5600		JMP I KEYBRD	/NO IGNORE M
1437	1352	6141		LINC	/YES LOAD
1440	1353	7522		6000+MAG	/MAGSPY
1441	1354	1103	L,	TAD MODE	/L KEY
1442	1355	7640		SZA CLA	/PAUSE MODE ?
1443	1356	5600		JMP I KEYBRD	/NO XIT
1444	1357	6002		IOF	/YES GET PROG
1445	1360	6141		LINC	/NAME TO BE
1446				LMODE	/LOADED
1447	1361	0602		LIF 2	
1450	1362	6454		JMP LOAD	
1451				Pmode	
1452	1363	6661	HLTALL,	ALLHLT	
1453	1364	0000	HSWIT,	0	
1454	1365	0000	GSWIT,	0	
1455	1366	1400	KHALT,	HALT	
1456	1367	0017	PP17,	17	
1457	1370	1411	KGO,	GO	
1460					/HERE TO HALT
1461					/A DEVICE

1462				PAGE	
1463	1400	1104	HALT,	TAD TEMP	/HOLDS DV NUM
1464					/TO BE HALTED
1465	1401	1100		TAD SELPTR	/PTR TO 1ST
1466					/SELECT WORD
1467					/THEY OCCUR
1470					/SEQUENTIALLY
1471	1402	3306		DCA ATEMP	
1472	1403	1706		TAD I ATEMP	/IS DEVICE
1473	1404	7650		SNA CLA	/HALTED ?
1474	1405	5705		JMP I KEYXIT	/YES IGNORE
1475					/HALT COMMAND
1476	1406	3706		DCA I ATEMP	/0 T0 SELECT
1477					/WORD
1500					/PUT HALT
1501	1407	4501		JMS I KHLTMS	/MESS IN QUE
1502	1410	5705		JMP I KEYXIT	
1503	1411	1104	GO,	TAD TEMP	/START A DEVICE
1504	1412	1100		TAD SELPTR	/GET ADDR OF
1505	1413	3306		DCA ATEMP	/SELECT WORD
1506	1414	1706		TAD I ATEMP	/IS THE DV IN
1507	1415	7640		SZA CLA	/AND MODE ?
1510	1416	5705		JMP I KEYXIT	/YES IGNORE
1511					/GO COMMAND
1512					/PUT GO MESS
1513	1417	4502		JMS I KGOMS	/IN QUE
1514	1420	1104		TAD TEMP	/GET ADDR OF
1515	1421	1307		TAD TIVAL	/DV GO VALUES
1516	1422	3310		DCA FROM	
1517	1423	1710		TAD I FROM	
1520	1424	3310		DCA FROM	
1521	1425	1104		TAD TEMP	/GET PTR TO
1522	1426	1312		TAD TIADR	/WHERE THEY
1523	1427	3311		DCA TO	/GO
1524	1430	1711		TAD I TO	
1525	1431	3311		DCA TO	
1526	1432	7146		CLL CMA RTL	/-3
1527	1433	1104		TAD TEMP	/RELATIVE TO
1530	1434	7740		SMA SZA CLA	/SAM 10 ?
1531	1435	7107		CLL IAC RTL	/(4) NO SAM 20
1532	1436	1313		TAD ASAM	/AC=SAM 10 OR
1533					/SAM 14
1534	1437	1104		TAD TEMP	/SETUP 1ST SAM
1535	1440	3264		DCA INISAM	
1536	1441	1104		TAD TEMP	/CLR HLT SWIT
1537	1442	1316		TAD HLTPTR	
1540	1443	3104		DCA TEMP	
1541	1444	7040		CMA	
1542	1445	3504		DCA I TEMP	
1543	1446	1314		TAD M5	/TRANSFER 5
1544	1447	3104		DCA TEMP	/VALUES FROM
1545					/TABLE
1546	1450	6002		IOF	/INTERUPTS OFF
1547					/UNTIL DV IS
1550					/SELECTED
1551	1451	1711	GOLOP,	TAD I TO	/GET AN ADDR
1552	1452	3306		DCA ATEMP	/SAVE
1553	1453	1710		TAD I FROM	/GET A VALUE
1554	1454	3706		DCA I ATEMP	/STORE
1555	1455	2310		ISZ FROM	/ADV PTRS
1556	1456	2311		ISZ TO	
1557	1457	2104		ISZ TEMP	/DONE 5 ?
1560	1460	5251		JMP GOLOP	

1561	1461	1711	TAD I TO	/GET ADDR OF
1562	1462	3306	DCA ATEMP	/OLDSAM
1563	1463	6141	LINC	
1564	1464	0000	INISAM, 0	/SAM X AXIS
1565	1465	0002	2	/PDP
1566	1466	1315	TAD PL1000	/MAK +
1567	1467	3706	DCA I ATEMP	/STORE INITIAL
1570				/SAM
1571	1470	1062	TAD KCLEN	/ENABLE ALL
1572	1471	6134	CLEN	/CLOCK
1573				/INTERUPTS
1574	1472	7200	CLA	
1575	1473	1103	TAD MODE	
1576	1474	7710	SPA CLA	
1577	1475	5300	JMP ,+3	
1600	1476	1304	TAD ADMS	
1601	1477	4505	JMS I KQUPUT	
1602	1500	7240	STA	/SET A\D MODE
1603	1501	3103	DCA MODE	
1604	1502	6001	ION	/DV IS READY
1605	1503	5705	JMP I KEYXIT	
1606	1504	6712	ADMS, MSAD	
1607	1505	1203	KEYXIT, OPKRTRN	
1610	1506	0000	ATEMP, 0	
1611	1507	6361	TIVAL, VALTAB	
1612	1510	0000	FROM, 0	
1613	1511	0000	TO, 0	
1614	1512	6366	TIADR, ADRTAB	
1615	1513	0110	ASAM, 0110	/SAM 10
1616	1514	7773	M5, -5	
1617	1515	1000	PL1000, 1000	
1620	1516	0073	HLTPTR, HSWIT0	
1621			LMODE	
1622	1517	1020	CATA, LDA I	/LOAD CATAL
1623	1520	1575	CATPTR=1	
1624	1521	7550	JMP LOADER	
1625	1522	1020	MAG, LDA I	/LOAD MAGSPY
1626	1523	1602	MAGPTR=1	
1627	1524	7550	JMP LOADER	
1630	1525	0642	KLOAD, LDF 2	/LOAD PRGM X
1631	1526	0063	SET I BETA1	/GET NAME FROM
1632	1527	5607	LODPTR=HBIT	/QANDA ANS BUF
1633	1530	0064	SET I BETA2	
1634	1531	2524	ANSWER+2000	
1635	1532	1324	LODLOP, LDH I BETA2	/FROM QA
1636	1533	0470	AZE I	
1637	1534	0017	COM	/SET 0 TO 77
1640	1535	1420	SHD I	
1641	1536	7400	QAEOL1	/END OF NAME
1642	1537	7542	JMP GETU	/YES GET UNIT
1643	1540	1363	STH I BETA1	/TO LOD TABLE
1644	1541	7532	JMP LODLOP	
1645	1542	1324	GETU, LDH I BETA2	/UNIT WD
1646	1543	1560	BCL I	
1647	1544	7770	7770	
1650	1545	1063	STA I BETA1	
1651	1546	1020	LDA I	
1652	1547	1607	LODPTR=1	
1653	1550	4004	LOADER, STC BETA2	/PUT PROG AND
1654	1551	0500	IOB	/UNIT IN E6
1655				/TABLE FOR
1656				/LOADER
1657				/ARG IS IN AC

1660	1552	6002	6002	/I OF
1661	1553	0001	AXO	/CLR XOB
1662	1554	0004	ESF	
1663	1555	0641	LDF 1	
1664	1556	0720	RDC I	/DIAL GRD TAB
1665	1557	4314	4\314	
1666	1560	0063	SET I BETA1	/E6 TABLE FOR
1667	1561	2372	2372	/LOADER
1670	1562	0065	SET I BETA3	/NAME AND UNIT
1671	1563	7772	=5	/TAKE 5 WORDS
1672	1564	1024	LDA I BETA2	
1673	1565	1063	STA I BETA1	
1674	1566	0225	XSK I BETA3	/DONE ?
1675	1567	7564	JMP ,=3	/NO
1676	1570	0642	LDF 2	
1677	1571	0720	RDC I	/GET DIAL
1700	1572	4354	4\354	/LOADER
1701	1573	0641	LDF 1	
1702	1574	0602	LIF 2	
1703	1575	6020	JMP 20	/GO LOADER
1704	1576	0301	CATPTR, 0301	/CA
1705	1577	2401	2401	/TA
1706	1600	0301	0301	/CA
1707	1601	1477	1477	/L 77
1710	1602	0000	0	/UNIT
1711	1603	1501	MAGPTR, 1501	/MA
1712	1604	0723	0723	/GS
1713	1605	2031	2031	/PY
1714	1606	7777	7777	
1715	1607	0000	0	/UNIT
1716	1610	0000	LODPTR, 0	/N
1717	1611	0000	0	/A
1720	1612	0000	0	/M
1721	1613	0000	0	/E
1722	1614	0000	0	/UNIT
1723			PMODE	/HERE TO SETUP
1724				/READ WRITE SR
1725	1615	0000	TPARGS, 0	/GET MTP ARGS
1726	1616	7200	CLA	/FROM QANDA
1727	1617	1260	TAD M10	/SET DIGCHK
1730	1620	3661	DCA I KUPLIM	/TO ACCEPT
1731				/OCTAL INPUT
1732				/ONLY
1733	1621	3262	DCA TBLK	/CLR TBLK
1734	1622	4463	TPLOP, JMS I KDIG	/GET 1 DIGIT
1735	1623	5253	JMP ANONO	/INPUT ERROR
1736	1624	5232	JMP DUNBLK	/FINI TBLK
1737	1625	1262	TAD TBLK	/UPDATE
1740	1626	7106	RTL CLL	/TBLK
1741	1627	7004	RAL	
1742	1630	3262	DCA TBLK	
1743	1631	5222	JMP TPLOP	
1744	1632	4463	DUNBLK, JMS I KDIG	/GET UNIT
1745	1633	5253	JMP ANONO	/INPUT ERROR
1746	1634	7200	CLA	/ONLY 1 ARG
1747	1635	7110	CLL RAR	/SET U BITS
1750	1636	6141	LINC	
1751			LMODE	
1752	1637	1120	ADA I	
1753	1640	0010	EXTMTP	/EXT U BITS
1754	1641	0001	AXO	/TO XOB
1755	1642	0002	PDP	/NOW PUT LOW
1756			PMODE	/ORD U BIT

1757	1643	7206		CLA RTL	/IN BIT 8
1760	1644	7006		RTL	/OF UNIT WORD
1761	1645	3263		DCA UNIT	
1762	1646	1262		TAD TBLK	
1763	1647	7112		RTR CLL	
1764	1650	7010		RAR	/POS TO 9 BITS
1765	1651	3262		DCA TBLK	
1766	1652	2215		ISZ TPARGS	/ADV PAST
1767	1653	7200	ANONO,	CLA	/ERROR RTN
1770	1654	1257		TAD MM12	/FIX UP
1771	1655	3661		DCA I KUPLIM	/DIGCHK
1772	1656	5615		JMP I TPARGS	
1773	1657	7765	MM12,	-13	/=12 IS COM
1774	1660	7767	M10,	=11	/=10 IS COM
1775	1661	4506	KUPLIM,	UPLIM	
1776	1662	0000	TBLK,	0	
1777	1663	0000	UNIT,	0	
2000					/HERE TO READ
2001					/PARAM INPUT
2002	1664	4215	DORDC,	JMS TPARGS	/GET TBLK, UNIT
2003	1665	5737		JMP I KRDERR	/INPUT ERROR
2004	1666	1262		TAD TBLK	
2005	1667	1344		TAD RMBLK	
2006	1670	3277		DCA RDCWD	
2007	1671	1263		TAD UNIT	
2010	1672	1341		TAD ANRDC	
2011	1673	3276		DCA TPRDC	
2012	1674	6141		LINC	
2013				LMODE	
2014	1675	0643		LDF 3	
2015	1676	0000	TPRDC,	0	/RDC (U)
2016	1677	0000	RDCWD,	0	/MBLK\TBLK
2017	1700	0416		STD	
2020	1701	7700		JMP , -1	
2021	1702	1020		LDA I	
2022	1703	0020		20	
2023	1704	0004		ESF	/I/O PRESET
2024	1705	1020		LDA I	
2025	1706	0010		EXTMTP	/SET
2026	1707	0001		AXO	/NO PAUSE
2027	1710	0002		PDP	
2030				PMODE	
2031	1711	7200		CLA	
2032	1712	1342		TAD RDLEN	/LENGTH OF INIT
2033	1713	3052		DCA CNTR	/TABLE
2034	1714	1340		TAD KFIRST	/START OF TABLE
2035	1715	3010		DCA INIPTR	
2036	1716	1034		TAD MINADR	/INPUT STARTS
2037	1717	3011		DCA ADPTR	/BEG OF BUFFER
2040	1720	1411		TAD I ADPTR	
2041	1721	3410		DCA I INIPTR	
2042	1722	2052		ISZ CNTR	/DONE ?
2043	1723	5320		JMP , =3	
2044	1724	1510		TAD I KLAST	/CHK FOR FATAL
2045	1725	7500		SMA	
2046	1726	5737		JMP I KRDERR	
2047	1727	7041		CIA	/ERROR WRONG
2050	1730	1061		TAD MAXDVN	/TBLK AND TOO
2051	1731	7740		SMA SZA CLA	/MANY DEVS
2052	1732	5737		JMP I KRDERR	/ILLEGAL TBLK
2053	1733	4745		JMS I KFRESH	/CLR BUFFER
2054	1734	1510		TAD I KLAST	/=NUM OF DEVS
2055	1735	5736		JMP I , +1	/GO TO QADUN

2056	1736	3315	STRRDE	/ROUT
2057	1737	4343	KRDERR, READ-1	
2060	1740	6000	KFIRST, FIRST=1	
2061	1741	0700	ANRDC, 700	
2062	1742	7634	RDLEN, FIRST=LAST-1	
2063	1743	4000	WMBLK, WRMBLK	
2064	1744	6000	RMBLK, ROMBLK	
2065	1745	3200	KFRESH, FRSTR	
2066	1746	4215	DOWRC, JMS TPARGS	/GET TBLK, UNIT
2067	1747	5764	JMP I KWRERR	/INPUT ERROR
2070	1750	1262	TAD TBLK	
2071	1751	1343	TAD WMBLK	
2072	1752	3046	DCA ARGLO	/ARG1 FOR
2073				/SR DOMTP
2074	1753	1263	TAD UNIT	
2075	1754	1362	TAD AWRI	/ARG2 FOR
2076	1755	3047	DCA ARGHI	/DOMTP
2077	1756	4763	JMS I KDOMTP	/WRITE N BLKS
2100	1757	6001	ION	
2101	1760	4465	JMS I KIDLE	
2102	1761	5360	JMP .=1	
2103	1762	0726	AWRI, 726	/WRI I
2104	1763	1067	KDOMTP, DOMTP	
2105	1764	4400	KWRERR, WRIT	
2106	1765	0000	AAAS0, 0	
2107			/DISPLAY ROUTINES	
2110			/MOVING WINDOW, CURSOR	
2111			/X-Y DECIMAL READ OUT	
2112			/HELP FRAME	
2113			SCPFLD=2000	
2114			LMODE	
2115			SEGMNT 1	/=SCPFLD/2000
2116			*17	
2117	0017	0000	XCORD, 0	/DIS REGISTER
2120	0020	0000	IDLE, 0	/IDLE LOOP
2121				/SCOPE, KEYBRD
2122				/TTY
2123	0021	6141	6141	/LINC
2124	0022	0460	SNS I 0	/1 OF 2 DISPS
2125	0023	6052	JMP DSCHLP	/HELP FRAME
2126				/DATA BUFFER
2127	0024	0104	SAM 4	/CURSOR KNOB
2130	0025	0341	SCR 1	/9 BITS COVERS
2131				/ALL X VALUES
2132	0026	0002	PDP	/MAKE +
2133			PMODE	/WHEN CURCNT
2134	2027	1366	TAD P401	/GOES TO 0
2135	2030	7041	CIA	/DIS CURSOR
2136	2031	6141	LINC	
2137			LMODE	
2140	0032	4351	STC CURCNT=SCPFLD	
2141	0033	0077	SET I XCORD	
2142	0034	6777	=1000	/INIT X COORD
2143				/=1
2144	0035	0100	SAM 0	/WINDOW KNOB
2145	0036	0344	SCR 4	/HI ORD 6 BITS
2146				/DET MOTION +
2147				/IS FWD =IS BWD
2150				/APPEARANCE OF
2151				/MOTION IS
2152				/GIVEN BY
2153				/VARYING THE
2154				/INITIAL ADDR

2155	0037	1520		SRO I	
2156	0040	7777	FREEZE,	7777	/7777 MEANS
2157					/FREEZE WINDOW
2160	0041	6250		JMP GODIS=SCPFLD=1	
2161	0042	6200		JMP SETMOT=SCPFLD	
2162				PMODE	/HERE WHEN
2163					/SCOPE FINI
2164	2043	4647	SCPRTN,	JMS I KKEY	/KEYBOARD
2165	2044	4651		JMS I KSWI	/INTERNAL HLT
2166	2045	4650		JMS I KMES	/TTY
2167	2046	5620		JMP I IDLE	
2170	2047	1200	KKEY,	KEYBRD	
2171	2050	1000	KMES,	MESCHK	
2172	2051	3107	KSWI,	CHKSWI	
2173				LMODE	/DSC HELP
2174					/SEE HLPTAB
2175					/FOR FORMAT
2176	0052	0642	DSCHLP,	LDF 2	/GRID TABLE
2177					/IN LDF 2
2200	0053	0077		SET I MSPTR	/TO DIAL
2201	0054	5614		HLPTAB=HBIT	/ENCODED MESS
2202	0055	0076		SET I VCPTR	/TABLE OF
2203	0056	1537		VCTAB=1	/VERT COORDS
2204	0057	0075		SET I HCPTR	/TABLE OF
2205	0060	1556		HCTAB=1	/HORIZ COORDS
2206	0061	1036	LNNXT,	LDA I VCPTR	
2207	0062	4737		STC VCOORD	
2210	0063	1035		LDA I HCPTR	
2211	0064	4001		STC 1	
2212	0065	1337		LDH I MSPTR	/1ST CHAR OF LN
2213	0066	1420		SHD I	/IS LN FULL
2214	0067	0600		FUL1	/SIZE CHARS ?
2215	0070	6114		JMP SETFUL	/YES
2216	0071	0011		CLR	/L=0 FOR HAF
2217	0072	0004		ESF	/SET HAF SIZE
2220	0073	1317		LDH MSPTR	/RESTORE CHAR
2221	0074	0467		SKP	
2222	0075	1337	CHRNXT,	LDH I MSPTR	
2223	0076	1420		SHD I	
2224	0077	4300		CR1	/END OF LN ?
2225	0100	6061		JMP LNNXT	/YES
2226	0101	1420		SHD I	/END OF
2227	0102	7700		ENDWD1	/MESSAGE ?
2230	0103	6163		JMP SCPXIT	/YES
2231	0104	1420		SHD I	/RESET
2232	0105	3600		CHGHC	/HORIZ COORD ?
2233	0106	6111		JMP SETHC	/YES
2234	0107	6732		JMP DSCCHR	
2235	0110	6075		JMP CHRNXT	
2236	0111	1035	SETHC,	LDA I HCPTR	
2237	0112	4001		STC 1	
2240	0113	6075		JMP CHRNXT	
2241	0114	1020	SETFUL,	LDA I	
2242	0115	0200		200	
2243	0116	0004		ESF	/SET FULL SIZE
2244	0117	0017		COM	
2245	0120	0261		ROL I 1	/L=1 FOR FULL
2246	0121	6075		JMP CHRNXT	
2247	0122	0642	DSCXY,	LDF 2	/DSC: X=-----
2250	0123	0011		CLR	/ Y=---
2251	0124	0004		ESF	/HALF SIZE
2252	0125	2166		ADD P401=SCPFLD	
2253	0126	4737		STC VCOORD	

2254	0127	0077		SET I MSPTR	/DIAL ENCODED
2255	0130	5526		DCODEX-HBIT	/MESS
2256	0131	1020	NXTLN,	LDA I	
2257	0132	7757		-20	
2260	0133	1140		ADM	
2261	0134	0737		VCOORD	
2262	0135	0061		SET I 1	
2263	0136	0000		XYHC	
2264	0137	1337	INICHR,	LDH I MSPTR	/INIT CHARS
2265	0140	1420		SHD I	
2266	0141	4300		CR1	/EOL ?
2267	0142	6145		JMP .+3	/YES
2270	0143	6732		JMP DSCCHR	
2271	0144	6137		JMP INICHR	
2272	0145	1337		LDH I MSPTR	
2273	0146	1420		SHD I	/DO NOT DSC
2274	0147	6000		6000	/LEADING 0S
2275	0150	6145		JMP . =3	
2276	0151	0467		SKP	
2277	0152	1337	NXTCHR,	LDH I MSPTR	
2300	0153	1420		SHD I	
2301	0154	4300		CR1	/EOL ?
2302	0155	6131		JMP NXTLN	/YES
2303	0156	1420		SHD I	
2304	0157	7700		ENDWD1	/END OF MESS ?
2305	0160	6163		JMP SCPXIT	/YES
2306	0161	6732		JMP DSCCHR	
2307	0162	6152		JMP NXTCHR	
2310	0163	0002	SCPXIT,	PDP	/RTN
2311				Pmode	
2312	2164	7300		CLA CLL	
2313	2165	5243		JMP SCPRTN	
2314	2166	0401	P401,	401	
2315				*SCPFLD+200	/GET INIT BUF
2316					/ADDR FOR DISP
2317					/DELTA X IN AC
2320	2200	0002	SETMOT, 2		/PDP
2321	2201	7100		CLL	
2322	2202	7510		SPA	/FWD OR BWD ?
2323	2203	5224		JMP BWMOT	
2324	2204	1346		TAD BUFLO	/UPDATE 15 BIT
2325	2205	3346		DCA BUFLO	/ADDR
2326	2206	7004		RAL	
2327	2207	1347		TAD BUFHI	
2330	2210	3347		DCA BUFHI	
2331	2211	1347		TAD BUFHI	/CHK FOR HI
2332	2212	7041		CIA	/END WRAP
2333	2213	1032		TAD MAXCDF	/AROUND
2334	2214	7700		SMA CLA	/OK ?
2335	2215	5251		JMP GODIS	/YES
2336	2216	1033		TAD MINCDF	/RESET LO END
2337	2217	3347		DCA BUFHI	
2340	2220	1034		TAD MINADR	/NEW ADDR IS
2341	2221	1346		TAD BUFLO	/REL TO MINADR
2342	2222	3346		DCA BUFLO	
2343	2223	5251		JMP GODIS	
2344	2224	1346	BWMOT,	TAD BUFLO	/BU ADDR
2345	2225	3346		DCA BUFLO	
2346	2226	7430		SZL	/UNDER FLOW ?
2347	2227	5233		JMP CHKADR	/NO
2350	2230	7240		STA	/YES BU CDF
2351	2231	1347		TAD BUFHI	/BITS
2352	2232	3347		DCA BUFHI	

2353	2233	1033	CHKADR,	TAD	MINCDF	
2354	2234	7041		CIA		
2355	2235	1347		TAD	BUFHI	
2356	2236	7740		SMA	SZA CLA	/CDF BITS AT
2357						/LOW END ?
2360	2237	5251		JMP	GODIS	/NO
2361	2240	1034		TAD	MINADR	
2362	2241	7041		CIA		
2363	2242	1346		TAD	BUFLO	/ADDR BITS
2364	2243	7500		SMA		/OK ?
2365	2244	5251		JMP	GODIS	/YES
2366	2245	3346		DCA	BUFLO	/CORRECT ADDR
2367						/IS IN AC
2370	2246	1032		TAD	MAXCDF	/SET CDF BITS
2371	2247	3347		DCA	BUFHI	/BWD WRAP
2372	2250	0002		2		/SOMETIMES HERE
2373						/IN LMODE (PDP)
2374	2251	7300	GODIS,	CLA	CLL	
2375	2252	1346		TAD	BUFLO	/SET BUFPTR
2376	2253	3054		DCA	BUFPTR	/TO INIT ADDR
2377	2254	1347		TAD	BUFHI	/SET CDF BITS
2400	2255	3060		DCA	CURCDF	
2401	2256	4320		JMS	SETCDF	/SET DATA FLD
2402	2257	1350		TAD	M1000	/SET FOR
2403	2260	3053		DCA	BCNTR	/512 POINTS
2404	2261	7200	NXTPNT,	CLA		
2405	2262	1454		TAD	I BUFPTR	
2406	2263	6141		LINC		
2407				LMODE		
2410	0264	0341		SCR	1	/SCP MAX IS 9
2411						/BITS
2412	0265	1520		SRO	I	/7777 INVERTS
2413	0266	0000	POLAR,	0		/SPECTRUM
2414	0267	0017		COM		
2415	0270	0177		DIS	I XCORD	
2416	0271	0002		PDP		
2417				PMODE		
2420	2272	2351		ISZ	CURCNT	/READY TO DIS
2421						/CURSOR
2422	2273	7410		SKP		/NO
2423	2274	5752		JMP	I KCUR	
2424	2275	2054	CURRTN,	ISZ	BUFPTR	/CHNG FLDS ?
2425	2276	7410		SKP		/NO
2426	2277	5303		JMP	FWDCDF	
2427	2300	2053	FWDRTN,	ISZ	BCNTR	/512 PNTS ?
2430	2301	5261		JMP	NXTPNT	/NO
2431	2302	5753		JMP	I KXYGET	/YES GET
2432						/X-Y COORDS
2433	2303	7200	FWDCDF,	CLA		/ADV CDF
2434	2304	2060		ISZ	CURCDF	
2435	2305	1060		TAD	CURCDF	
2436	2306	7041		CIA		
2437	2307	1032		TAD	MAXCDF	
2440	2310	7700		SMA	CLA	/WRAP AROUND ?
2441	2311	5316		JMP	BYWRAP	/NO
2442	2312	1034		TAD	MINADR	
2443	2313	3054		DCA	BUFPTR	
2444	2314	1033		TAD	MINCDF	
2445	2315	3060		DCA	CURCDF	
2446	2316	4320	BYWRAP,	JMS	SETCDF	
2447	2317	5300		JMP	FWDRTN	
2450	2320	0000	SETCDF,	0		/SET CDF REL
2451						/TO CURCDF

2452	2321	1060		TAD CURCDF	
2453	2322	7104		RAL CLL	
2454	2323	7006		RTL	
2455	2324	1031		TAD CDF0	
2456	2325	3326		DCA .+1	
2457	2326	0000		Ø	
2460	2327	5720		JMP I SETCDF	
2461					/DBL PREC ADD
2462					/OF
2463					/(DBLHI, DBLLO)
2464					/(ARGHI, ARGLO)
2465					/ANSWER IN
2466					/(DBLHI, DBLLO)
2467					/L=0 IF +
2470					/L=1 IF =
2471	2330	0000	DADD,	Ø	
2472	2331	7300		CLA CLL	
2473	2332	1044		TAD DBLLO	
2474	2333	1046		TAD ARGLO	/=XCURL0
2475	2334	3044		DCA DBLLO	
2476	2335	7004		RAL	
2477	2336	1045		TAD DBLHI	
2500	2337	1047		TAD ARGHI	/=XCURHI
2501	2340	3045		DCA DBLHI	
2502	2341	7100		CLL	
2503	2342	1045		TAD DBLHI	
2504	2343	7710		SPA CLA	
2505	2344	7020		CML	
2506	2345	5730		JMP I DADD	
2507	2346	7000	BUFLO,	ADMIN	
2510	2347	0000	BUFHI,	CFDMIN	
2511	2350	7000	M1000,	=1000	
2512	2351	0000	CURCNT,	Ø	
2513	2352	2700	KCUR,	CURDIS	
2514	2353	2400	KXYGET,	XYGET	
2515				*SCPFLD+400	/FIND REL DEV
2516					/ADDR IN BUFFER
2517	2400	6201	XYGET,	CDF Ø	/RESTORE FIELD
2520	2401	7200		CLA	/HOLDS ADDR OF
2521	2402	1325		TAD KLIM	/TABLE DEV INFO
2522	2403	3055		DCA LIMPTR	
2523	2404	7240		STA	
2524	2405	1036		TAD DEVNUM	/= NUM DV
2525	2406	3052		DCA CNTR	
2526	2407	1455	LIMLOP,	TAD I LIMPTR	/=ST ADDR
2527	2410	3044		DCA DBLLO	/FOR DV K
2530	2411	2055		ISZ LIMPTR	/ADDR TO DBLLO
2531	2412	1455		TAD I LIMPTR	/CDF BITS
2532	2413	3045		DCA DBLHI	/TO DBLHI
2533					/XCOORD IS IN
2534					/(ARGHI, ARGLO)
2535	2414	4437		JMS I KDADD	/IS XCOORD GE
2536	2415	7430		SZL	/ST ADDR DV K ?
2537	2416	5232		JMP FOUND	/NO ADDR LIMITS
2540					/HAVE BEEN DET
2541	2417	1044		TAD DBLLO	/SAVE DIFF FOR
2542	2420	3050		DCA TEMPLO	/DV K IT IS REL
2543	2421	1045		TAD DBLHI	/ADDR FOR DV K+1
2544	2422	3051		DCA TEMPHI	
2545	2423	2055		ISZ LIMPTR	/ADV PTR TO
2546	2424	2055		ISZ LIMPTR	/NEXT ST
2547	2425	2055		ISZ LIMPTR	/ADDR
2550	2426	2052		ISZ CNTR	/SCANNED ALL

2551					/DEVICES ?
2552	2427	5207		JMP LIMLOP	/NO
2553	2430	5631		JMP I ,+1	/YES CURSOR IS
2554	2431	2043		SCPRTN	/IN VOID AREA
2555					/DO NOT DSC X=Y
2556	2432	3044	FOUND,	DCA DBLLO	/ANSWER IS IN
2557	2433	3045		DCA DBLHI	/TEMPLO,TEMPHI
2560					/CLR DBL AC
2561	2434	7346		STA CLL RTL	/BU PTR (=3)
2562	2435	1055		TAD LIMPTR	/TO DEVICE
2563	2436	3055		DCA LIMPTR	/SAM FREQ
2564	2437	1455		TAD I LIMPTR	/MULT ANSWER
2565	2440	3052		DCA CNTR	/BY SAM FREQ
2566	2441	1051		TAD TEMPHI	
2567	2442	3047		DCA ARGHI	
2570	2443	1050		TAD TEMPLO	
2571	2444	3046		DCA ARGLO	
2572	2445	4437		JMS I KDADD	
2573	2446	2052		ISZ CNTR	/DONE ?
2574	2447	5245		JMP ,=2	/NO
2575					/YES REL OCTAL
2576					/ADDR NOW IN
2577					/DBL AC CVERT
2600					/TO BCD
2601	2450	2055		ISZ LIMPTR	/ADV TO BCD
2602	2451	1455		TAD I LIMPTR	/ADDR FOR DV K
2603	2452	3055		DCA LIMPTR	/(LO TO HI)
2604					/5 DIGITS
2605	2453	1326		TAD KBCD	/ADDR OF BCD
2606	2454	3056		DCA BCDPTR	/ANSWER
2607	2455	1327		TAD KRAD	/PTR TO
2610	2456	3057		DCA RADPTR	/DECIMAL RADII
2611	2457	1330		TAD NEG5	
2612	2460	3052		DCA CNTR	
2613	2461	1457	DBLLOP,	TAD I RADPTR	/=100K,-10, HI
2614	2462	3047		DCA ARGHI	
2615	2463	2057		ISZ RADPTR	
2616	2464	1457		TAD I RADPTR	/=100K,-10, LO
2617	2465	3046		DCA ARGLO	
2620	2466	2057		ISZ RADPTR	
2621	2467	3053		DCA BCNTR	/COUNTS MULTS
2622					/OF PWRS OF 10
2623	2470	1044	DBLDEF,	TAD DBLLO	/SAVE LAST
2624	2471	3050		DCA TEMPLO	/DEFLATE
2625	2472	1045		TAD DBLHI	
2626	2473	3051		DCA TEMPHI	
2627	2474	4437		JMS I KDADD	/DEFLATE 1 TIME
2630	2475	2053		ISZ BCNTR	
2631	2476	7420		SNL	/MINUS YET ?
2632	2477	5270		JMP DBLDEF	/NO CONT
2633	2500	7240		STA	/CORRECT FOR
2634	2501	1053		TAD BCNTR	/OFF BY 1
2635	2502	3456		DCA I BCDPTR	/SAVE
2636	2503	2056		ISZ BCDPTR	
2637	2504	1050		TAD TEMPLO	/RESET TO
2640	2505	3044		DCA DBLLO	/LAST VAL
2641	2506	1051		TAD TEMPHI	
2642	2507	3045		DCA DBLHI	
2643	2510	2052		ISZ CNTR	/DONE BOTH ?
2644	2511	5261		JMP DBLLOP	/NO
2645	2512	1044		TAD DBLLO	/UNITS
2646	2513	3456		DCA I BCDPTR	
2647	2514	6141		LINC	

2650			LMODE	
2651	0515	0077	SET I XCORD	/SET PTR TO
2652	0516	1533	DCODEX+4	/X-Y TABLE
2653	0517	0002	PDP	
2654			PMODE	
2655	2520	1330	TAD NEG5	/SET FOR 5
2656	2521	3052	DCA CNTR	/BCD ADDS
2657	2522	3053	DCA BCNTR	/HOLDS BCD
2660				/CARRYS
2661	2523	5724	JMP I,+1	
2662	2524	2600	BCDLOP	/CONT NXT PAGE
2663	2525	6443	KLIM,	
2664	2526	6435	LIMVL0	
2665	2527	6417	KBCD,	
2666	2530	7773	BCDVAL	
2667			KRAD,	
2670			NEG5,	
2671			-5	/ADD ABSOLUTE
2672				/DECIMAL VIA
2673				/LIMPTR TO
2674				/DECIMAL XCOORD
2675				/VIA BCDPTR AND
2676				/STORE FINAL
2677				/RESULT IN
2700				/X-Y TABLE
2701			*SCPFLD+600	
2702	2600	7300	BCDLOP, CLA CLL	
2703	2601	1053	TAD BCNTR	/CARRY=0,1
2704	2602	1455	TAD I LIMPTR	/0=11
2705	2603	1456	TAD I BCDPTR	/0=11
2706	2604	3050	DCA TEMPLO	/0=23
2707	2605	1050	TAD TEMPLO	/CHK CARRY
2710	2606	1353	TAD M12	
2711	2607	7500	SMA	/CARRY ?
2712	2610	3050	DCA TEMPLO	/YES CORRECT
2713				/ANSER IS IN AC
2714	2611	7204	RAL CLA	/L=1 ON CARRY
2715	2612	3053	DCA BCNTR	/SET CARRY FOR
2716				/NEXT BCD ADD
2717	2613	7040	CMA	/RESET PTRS
2720	2614	1056	TAD BCDPTR	
2721	2615	3056	DCA BCDPTR	
2722	2616	2055	ISZ LIMPTR	
2723	2617	1050	TAD TEMPLO	/ANSWER+60
2724	2620	1354	TAD P60	/=DIAL CODE
2725	2621	6141	LINC	
2726			LMODE	/STORE IN
2727	0622	1357	STH XCORD	/X=Y TABLE
2730	0623	1020	LDA I	/BU XCORD
2731	0624	3777	=HBIT	
2732	0625	2017	ADD XCORD	
2733	0626	4017	STC XCORD	
2734	0627	0002	PDP	
2735			PMODE	
2736	2630	2052	ISZ CNTR	/DONE ?
2737	2631	5200	JMP BCDLOP	/NO
2740	2632	1456	TAD I BCDPTR	/YES GET 100K
2741	2633	1053	TAD BCNTR	/DIGIT+CARRY
2742	2634	1354	TAD P60	
2743	2635	6141	LINC	
2744			LMODE	
2745	0636	1357	STH XCORD	
2746	0637	0077	SET I XCORD	/NOW SET FOR
	0640	1535	DCODEY+1	/Y VALUE
	0641	0002	PDP	

2747				PMODE	
2750	2642	7344		CLL STA RAL	/DO 100S, 10S
2751	2643	3052		DCA CNTR	/Y BCD CVERT
2752	2644	1355		TAD PP400	/MAKE Y +
2753	2645	1035		TAD YCUR	
2754	2646	3035		DCA YCUR	
2755					/Y IS 0-777
2756	2647	1457	YLOP,	TAD I RADPTR	/-100, -10
2757	2650	3046		DCA ARGLO	
2760	2651	3053		DCA BCNTR	
2761	2652	2057		ISZ RADPTR	
2762	2653	1035		TAD YCUR	
2763	2654	1046		TAD ARGLO	/1 DEFLATE
2764	2655	2053		ISZ BCNTR	/BCD VALUE
2765	2656	7500		SMA	/DONE ?
2766	2657	5254		JMP ,=3	/NO
2767	2660	1457		TAD I RADPTR	/+100, +10
2770	2661	3035		DCA YCUR	/RESET VALUE
2771	2662	2057		ISZ RADPTR	
2772	2663	1053		TAD BCNTR	/OFF BY 1
2773	2664	1357		TAD P57	/60-1
2774	2665	6141		LINC	
2775				LMODE	/DIAL CODE
2776	0666	1377		STH I XCORD	/TO X-Y TABLE
2777	0667	0002		PDP	
3000				PMODE	
3001	2670	7200		CLA	
3002	2671	2052		ISZ CNTR	/DONE ?
3003	2672	5247		JMP YLOP	/NO
3004	2673	1035		TAD YCUR	/UNITS
3005	2674	1354		TAD P60	
3006	2675	6141		LINC	/STORE
3007				LMODE	/THE TABLE
3010	0676	1377		STH I XCORD	/IS FINALLY
3011	0677	6122		JMP DSCXY	/READY
3012					/HERE TO
3013				PMODE	/DIS CURSOR
3014	2700	3035	CURDIS,	DCA YCUR	/Y COORD
3015	2701	6214		RDF	
3016	2702	7110		CLL RAR	
3017	2703	7012		RTR	
3020	2704	3047		DCA XCURHI	/X COORD HI
3021	2705	1054		TAD BUFPTR	
3022	2706	3046		DCA XCURL0	/X COORD LO
3023	2707	7001		IAC	
3024	2710	3051		DCA TEMPHI	/DELTA V +, =1
3025	2711	1351		TAD CURLEN	/CURSOR IS VERT
3026	2712	3052		DCA CNTR	/LN OF LENGTH
3027					/=C(CURLEN)
3030	2713	1352		TAD MAXCUR	/IF Y COORD
3031	2714	1035		TAD YCUR	/TOO HI CURSOR
3032	2715	7750		SPA SNA CLA	/IS DIS BELOW
3033					/SPECTRUM
3034	2716	5321		JMP CURLOP-1	
3035	2717	7040		CMA	/TOO HI SET
3036	2720	3051		DCA TEMPHI	/DELTA V TO =1
3037	2721	1035		TAD YCUR	
3040	2722	1051	CURLOP,	TAD TEMPHI	
3041	2723	6141		LINC	
3042				LMODE	
3043	0724	0157		DIS XCORD	
3044	0725	0002		PDP	
3045				PMODE	

3046	2726	2052		ISZ CNTR	
3047	2727	5322		JMP CURLOP	
3050	2730	5731		JMP I ,+1	
3051	2731	2275		CURRTN	
3052				LMODE	
3053	0732	0241	DSCCHR,	ROL 1	/MULT BY 2
3054	0733	1120		ADA I	/ADDR IS REL
3055	0734	7330		GRDTAB-1+2000	/TO GRDTAB
3056	0735	4014		STC GRDPTR	
3057	0736	1120		ADA I	
3060	0737	0000	VCOORD,	0	
3061	0740	1774		DSC I GRDPTR	
3062	0741	1774		DSC I GRDPTR	
3063	0742	1020		LDA I	/MAKE HORIZ
3064	0743	0002	P2,	2	/GAP BET CHARS
3065	0744	0452		LZE	/2 FOR HALF
3066	0745	2743		ADD P2	/4 FOR FULL
3067	0746	1140		ADM	
3070	0747	0001		1	
3071	0750	6000		JMP 0	
3072				Pmode	
3073	2751	7750	CURLN,	KURLN	/=-LENGTH OF
3074					/CURSOR
3075	2752	7431	MAXCUR,	=377=KURLN	/MAX Y COORD
3076					/FOR CURSOR
3077					/ABOVE SPECTRUM
3100	2753	7766	M12,	-12	
3101	2754	0060	P60,	60	
3102	2755	0400	PP400,	400	
3103	2756	7774	MM4,	=4	
3104	2757	0057	P57,	57	
3105				PAGE	
3106	3000	0000	MUL10,	0	/DAC*10+AC
3107	3001	3050		DCA TEMPL0	/TO DAC
3110	3002	4217		JMS DSCL	/DAC*2
3111	3003	1044		TAD DBLLO	
3112	3004	3046		DCA ARGLO	/SAVE
3113	3005	1045		TAD DBLHI	
3114	3006	3047		DCA ARGHI	
3115	3007	4217		JMS DSCL	/YIELDS
3116	3010	4217		JMS DSCL	/DAC*8
3117	3011	4437		JMS I KDADD	/+DAC*2=DAC*10
3120	3012	3047		DCA ARGHI	/NOW ADD NEW
3121	3013	1050		TAD TEMPL0	/VAL
3122	3014	3046		DCA ARGLO	
3123	3015	4437		JMS I KDADD	/=DAC*10+AC
3124	3016	5600		JMP I MUL10	
3125	3017	0000	DSCCL,	0	/RAL DAC =DAC*2
3126	3020	7300		CLA CLL	
3127	3021	1044		TAD DBLLO	
3130	3022	7004		RAL	
3131	3023	3044		DCA DBLLO	
3132	3024	1045		TAD DBLHI	
3133	3025	7004		RAL	
3134	3026	3045		DCA DBLHI	
3135	3027	5617		JMP I DSCL	
3136	3030	0000	DCIA,	0	/2S COMP OF DAC
3137	3031	7200		CLA	
3140	3032	1045		TAD DBLHI	
3141	3033	7040		CMA	
3142	3034	3045		DCA DBLHI	
3143	3035	1044		TAD DBLLO	
3144	3036	7041		CIA	

3145	3037	7450		SNA	
3146	3040	2045		ISZ DBLHI	
3147	3041	3044		DCA DBLLO	
3150	3042	5630		JMP I DCIA	
3151	3043	0000	DSCL4,	0	/PACK BCD IN
3152	3044	3050		DCA TEMPLO	/DAC =ROL 4 +AC
3153	3045	4217		JMS DSCL	/MOVE 4 LEFT
3154	3046	4217		JMS DSCL	
3155	3047	4217		JMS DSCL	
3156	3050	4217		JMS DSCL	
3157	3051	1044		TAD DBLLO	
3160	3052	1050		TAD TEMPLO	
3161	3053	3044		DCA DBLLO	
3162	3054	5643		JMP I DSCL4	
3163	3055	0000	UPACK,	0	/UNPACK DAC
3164	3056	7346		STA CLL RTL	/PUT IN INITAB
3165	3057	3050		DCA TEMPLO	/5 VALS 3 IN
3166	3060	5264		JMP ,+4	/DBLLO
3167					/2 IN DBLHI
3170	3061	7012	PCKLOP,	RTR	/NXT VAL
3171	3062	7012		RTR	/TO LOW 4 BITS
3172	3063	3044		DCA DBLLO	/UPDATE
3173	3064	1044		TAD DBLLO	
3174	3065	0306		AND P17	
3175	3066	3410		DCA I INIPTR	/1 VAL TO TABLE
3176	3067	1044		TAD DBLLO	
3177	3070	2050		ISZ TEMPLO	/DONE ?
3200	3071	5261		JMP PCKLOP	/NO
3201	3072	7200		CLA	
3202	3073	1045		TAD DBLHI	/LAST VALS
3203	3074	0306		AND P17	
3204	3075	3410		DCA I INIPTR	
3205	3076	1045		TAD DBLHI	
3206	3077	7012		RTR	
3207	3100	7012		RTR	
3210	3101	0306		AND P17	
3211	3102	3410		DCA I INIPTR	
3212	3103	3044		DCA DBLLO	/CLR DAC
3213	3104	3045		DCA DBLHI	
3214	3105	5655		JMP I UPACK	
3215	3106	0017	P17,	17	
3216	3107	0000	CHKSWI,	0	/SEE IF TISA
3217	3110	1330		TAD HSWPTR	/HAS HALTED
3220	3111	3004		DCA BETA2	/ANY DEVS
3221	3112	1036		TAD DEVNUM	/=NUM DEFINED
3222	3113	3052		DCA CNTR	
3223	3114	1404	CHKLOP,	TAD I BETA2	
3224	3115	7510		SPA	/TISA HALT ?
3225	3116	5323		JMP BYHLT	/NO
3226	3117	3104		DCA TEMP	/HLT MESS
3227	3120	4501		JMS I KHLTMS	/TO QUE
3230	3121	7040		CMA	/CLR SWIT
3231	3122	3404		DCA I BETA2	
3232	3123	7200	BYHLT,	CLA	
3233	3124	2004		ISZ BETA2	
3234	3125	2052		ISZ CNTR	
3235	3126	5314		JMP CHKLOP	
3236	3127	5707		JMP I CHKSWI	
3237	3130	0073	HSWPTR,	HSWIT0	
3240				PAGE	
3241					/FRESH START
3242					/CLR BUFFER
3243					/INIT ALL ARGS

3244	3200	0000	FRSTR,	0	
3245	3201	7200		CLA	/PUT 0S FROM
3246	3202	1034		TAD MINADR	/MINCDF,MINADR
3247	3203	3046		DCA ARGLO	/TO MAXCDF,7777
3250	3204	1033		TAD MINCDF	
3251	3205	3060		DCA CURCDF	
3252	3206	4672	CLRLOP,	JMS I KSET	/SET CDF
3253	3207	3446		DCA I ARGLO	/A 0 TO BUFER
3254	3210	2046		ISZ ARGLO	/DONE FIELD ?
3255	3211	5207		JMP ,=2	/NO
3256	3212	2060		ISZ CURCDF	/ADV FLD
3257	3213	1060		TAD CURCDF	
3260	3214	7041		CIA	
3261	3215	1032		TAD MAXCDF	
3262	3216	7700		SMA CLA	/FINI ?
3263	3217	5206		JMP CLRLOP	/NO
3264	3220	6201		CDF 0	/RESET FLD
3265	3221	3053		DCA BCNTR	/DEV COUNT
3266	3222	1034		TAD MINADR	/1ST AVAIL
3267	3223	3274		DCA LSSALO	/BUF ADDR
3270	3224	1033		TAD MINCDF	
3271	3225	3275		DCA LSSAHI	/=SA DV 0
3272	3226	1106		TAD KINI	
3273	3227	3276		DCA LSINI	
3274	3230	1277		TAD KENC	/SA ENC TABLE
3275	3231	3300		DCA LSENC	
3276	3232	5600		JMP I FRSTR	
3277					/INIT ARGS FOR
3300					/CURRENT DEVICE
3301	3233	0000	RESTR,	0	
3302	3234	7200		CLA	
3303	3235	1053		TAD BCNTR	/=NUM OF
3304					/DEFINED DVS
3305	3236	3052		DCA CNTR	
3306	3237	1274		TAD LSSALO	/SA CURN DV
3307	3240	3021		DCA CURNLO	
3310	3241	1275		TAD LSSAHI	
3311	3242	3020		DCA CURNHI	
3312	3243	1276		TAD LSINI	/PTR TO INITAB
3313	3244	3010		DCA INIPTR	/CURN DV
3314	3245	1300		TAD LSENC	
3315	3246	3022		DCA ENCPTR	
3316	3247	3044		DCA DBLLO	
3317	3250	3045		DCA DBLHI	
3320	3251	1301		TAD P4060	/CURN DV NUM
3321	3252	1052		TAD CNTR	/TO QA TXT FLD
3322	3253	3702		DCA I KTX26	
3323	3254	5633		JMP I RESTR	
3324					/INIT ARGS NXT
3325					/DV REVERSE OF
3326					/RESTR
3327	3255	0000	NXSTR,	0	
3330	3256	7200		CLA	
3331	3257	1052		TAD CNTR	
3332	3260	3053		DCA BCNTR	
3333	3261	1021		TAD CURNLO	
3334	3262	3274		DCA LSSALO	
3335	3263	1020		TAD CURNHI	
3336	3264	3275		DCA LSSAHI	
3337	3265	1010		TAD INIPTR	
3340	3266	3276		DCA LSINI	
3341	3267	1022		TAD ENCPTR	
3342	3270	3300		DCA LSENC	

3343	3271	5655	JMP I NXSTR	
3344	3272	2320	KSET, SETCDF	
3345	3273	7757	INILEN, INISTR=INIEND-1	
3346	3274	0000	LSSALO, 0	
3347	3275	0000	LSSAHI, 0	
3350	3276	0000	LSINI, 0	
3351	3277	3306	KENC, ENCVAL	
3352	3300	0000	LENC, 0	
3353	3301	4060	P4060, 4060	
3354	3302	4573	KTXT26, TXT2+6	
3355	3303	6144	KADR, INIAD=1	
3356	3304	7776	CLKCNT, -2	
3357	3305	4100	CLKMOD, 4100	
3360	3306	0040	ENCVAL, 0040	/ENABLE CHAN 1
3361	3307	0010	0010	/" " 2
3362	3310	0002	0002	/" " 3
3363	3311	0000	0	/MUST HAVE 0
3364				/HERE
3365	3312	7200	QADUN, CLA	/DONE SETUP
3366	3313	1052	TAD CNTR	/NUM OF DEV
3367	3314	7041	CIA	
3370	3315	5036	STRRDE, DCA DEVNUM	
3371	3316	1036	TAD DEVNUM	
3372	3317	3510	DCA I KLAST	/VERY LAST
3373				/ENTRY IN
3374				/TABLE
3375	3320	1036	TAD DEVNUM	
3376	3321	3052	DCA CNTR	/TRANSFER ALL
3377	3322	1106	TAD KINI	/INITIAL VALS
3400	3323	5010	DCA INIPTR	
3401	3324	1503	TAD KADR	
3402	3325	3011	DCA ADPTR	
3403	3326	1273	LOP1, TAD INILEN	
3404	3327	5053	DCA BCNTR	
3405	3330	1411	LOP2, TAD I ADPTR	/1 ADDR
3406	3331	3104	DCA TEMP	/SAVE
3407	3332	1410	TAD I INIPTR	/1 VAL
3410	3333	3504	DCA I TEMP	/STORE
3411	3334	2053	ISZ BCNTR	/DONE A DEV ?
3412	3335	5330	JMP LOP2	/NO
3413	3336	2052	ISZ CNTR	/DONE ALL ?
3414	3337	5326	JMP LOP1	
3415	3340	1504	TAD CLKCNT	/SET CLOCK FOR
3416	3341	6133	CLAB	/2 MS COUNT
3417	3342	7200	CLA	/SET 1MS TICK
3420	3343	1305	TAD CLKMOD	/AND INTERRUPT
3421	3344	6132	CLLR	/ON OVERFLOW
3422	3345	7200	CLA	
3423	3346	3025	DCA QUECNT	/SET TTY QUE
3424	3347	1367	TAD KSTRQ	/PTRS AND CNT
3425	3350	3023	DCA QUEIN	
3426	3351	1367	TAD KSTRQ	
3427	3352	3024	DCA QUEOUT	
3430	3353	7040	CMA	
3431	3354	3103	DCA MODE	/SET A/D MODE
3432	3355	1366	TAD K210	/PUT CNTRL H
3433	3356	3021	DCA KBDBUF	/IN KEYBRD BUF
3434	3357	3022	DCA TTYFLG	/SET TTY FOR
3435	3360	1365	TAD KITTY	/NOT BUSY
3436	3361	5020	DCA ADMESS	/NO MESSAGES
3437	3362	6001	ION	/HANG IN IDLE
3440	3363	4465	JMS I KIDLE	/LOOP TISA WILL
3441	3364	5363	JMP ,=1	/IMMEDIATELY

3442	3365	0022	KTTY,	TTYFLG	/SWITCH INTO
3443	3366	0210	K210,	210	/PAUSE MODE
3444	3367	6373	KSTRQ,	STRQUE	
3445				PAGE	
3446	3400	0000	P1DAC,	0	/ADD 1 TO DAC
3447	3401	7201		CLA IAC	
3450	3402	3046		DCA ARGLO	
3451	3403	3047		DCA ARGHI	
3452	3404	4437		JMS I KDADD	
3453	3405	5600		JMP I P1DAC	
3454	3406	0000	LOHI,	0	/SA LO HI
3455	3407	1021		TAD CURNLO	/TO TABLE
3456	3410	3410		DCA I INIPTR	
3457	3411	1020		TAD CURNHI	
3460	3412	3410		DCA I INIPTR	
3461	3413	1021		TAD CURNLO	
3462	3414	3044		DCA DBLLO	
3463	3415	1020		TAD CURNHI	
3464	3416	3045		DCA DBLHI	
3465	3417	4627		JMS I KKDCIA	/DBL PREC CIA
3466	3420	1044		TAD DBLLO	/=SA LO HI
3467	3421	3410		DCA I INIPTR	/TO TABLE
3470	3422	1045		TAD DBLHI	
3471	3423	3410		DCA I INIPTR	
3472	3424	3044		DCA DBLLO	
3473	3425	3045		DCA DBLHI	
3474	3426	5606		JMP I LOHI	
3475	3427	3030	KKDCIA,	DCIA	
3476	3430	0000	GBLK,	0	/DETERMINE
3477					/BUFER LEN
3500					/AS A FCN OF
3501					/TBLKS FOR
3502					/TAPE ROUTS
3503	3431	7200		CLA	
3504	3432	1023		TAD QUEIN	/-LAST USED
3505	3433	3044		DCA DBLLO	/CORE LOC
3506	3434	1024		TAD QUEOUT	
3507	3435	3045		DCA DBLHI	
3510	3436	1034		TAD MINADR	
3511	3437	3046		DCA ARGLO	
3512	3440	1033		TAD MINCDF	
3513	3441	3047		DCA ARGHI	
3514	3442	4437		JMS I KDADD	/DAC NOW==
3515					/TOT CORE
3516					/LOCS USED
3517	3443	1260		TAD PL400	
3520	3444	3046		DCA ARGLO	
3521	3445	3047		DCA ARGHI	
3522	3446	3053		DCA BCNTR	
3523	3447	4437		JMS I KDADD	/ADD 400
3524	3450	2053		ISZ BCNTR	/COUNT TBLKS
3525	3451	7430		SZL	/DONE ?
3526	3452	5247		JMP ,=3	/NO
3527	3453	7005		IAC RAL	/2 BLKS OF HDR
3530	3454	1053		TAD BCNTR	
3531	3455	7041		CIA	
3532	3456	3507		DCA I KBLKCT	
3533	3457	5630		JMP I GBLK	
3534	3460	0400	PL400,	400	
3535	3461	0000	HLTMS,	0	/PUT HLT MESS
3536	3462	1314		TAD AHLTMS	/ADDR OF MESS
3537	3463	4505		JMS I KQUPUT	/INPUT MESS
3540	3464	4317		JMS NUMPUT	/DEV NUM TO QUE

3541	3465	1103		TAD MODE	
3542	3466	7700		SMA CLA	/PAUSE MODE ?
3543	3467	5661		JMP I HLTMS	/YES
3544	3470	1036		TAD DEVNUM	/CHK FOR PAUSE
3545	3471	3053		DCA BCNTR	/ALL SELECT
3546	3472	7040		CMA	/WORDS MUST
3547	3473	1100		TAD SELPTR	/BE 0
3550	3474	5011		DCA ADPTR	
3551	3475	1411	HLTLOP,	TAD I ADPTR	
3552	3476	7640		SZA CLA	/DEV K HALTED ?
3553	3477	5661		JMP I HLTMS	/NO EXIT
3554	3500	2053		ISZ BCNTR	/CHKED ALL ?
3555	3501	5275		JMP HLTLOP	/NO
3556	3502	1316		TAD HALTAL	/PAUSE MODE
3557	3503	4505		JMS I KQUPUT	/INPUT MESS
3560	3504	3103		DCA MODE	/0=PAUSE
3561	3505	6134		CLEN	/CLR CLOCK
3562	3506	5661		JMP I HLTMS	
3563	3507	0000	GOMS,	0	/PUT GO MESS
3564	3510	1315		TAD AGOMS	
3565	3511	4505		JMS I KQUPUT	/INPUT MESS
3566	3512	4317		JMS NUMPUT	
3567	3513	5707		JMP I GOMS	
3570	3514	6527	AHLTMS,	MSHLT	/ADDR OF HLT MS
3571	3515	6550	AGOMS,	MSGO	/ADDR OF GO MS
3572	3516	6661	HALTAL,	ALLHLT	/ADDR OF PAUSE
3573					/MODE MESS
3574	3517	0000	NUMPUT,	0	
3575	3520	1104		TAD TEMP	
3576	3521	1326		TAD KNUM	
3577	3522	5003		DCA BETA1	
3600	3523	1403		TAD I BETA1	
3601	3524	4505		JMS I KQUPUT	
3602	3525	5717		JMP I NUMPUT	
3603	3526	6522	KNUM,	NUM0	
3604					/X-Y TABLE
3605					/FOR DSC OF
3606					/X=----
3607					/Y=----
3610				LMODE	
3611	1527	3075	DCODEX,	3075	/X, =
3612	1530	4300		CR1 00	/EOL, 100K
3613	1531	0000		0000	/10K, 1K
3614	1532	0000		0000	/100, 10
3615	1533	0043		00 CR	/UNITS, EOL
3616	1534	3175	DCODEY,	3175	/Y, =
3617	1535	4300		CR1 00	/EOL, 100
3620	1536	0000		0000	/10, UNITS
3621	1537	7700		ENDWD1	/EOM
3622					/TABLE OF VERT
3623					/COORDS FOR
3624					/HELP FRAME
3625					/DISPLAY
3626	1540	0340	VCTAB,	340	/D DIAL
3627	1541	0240		240	/A/D MODE
3630	1542	0200		200	/CTRL/H
3631	1543	0160		160	/HN
3632	1544	0140		140	/P
3633	1545	0120		120	/F
3634	1546	0020		20	/PAUSE MODE
3635	1547	7757		=20	/GN
3636	1550	7737		=40	/S
3637	1551	7717		=60	/W

3640	1552	7677	-100	/C
3641	1553	7657	-120	/M
3642	1554	7637	-140	/L
3643	1555	7617	-160	/P
3644	1556	7577	-200	/F
3645				/TABLE OF HORIZ
3646				/COORDS
3647	1557	0020	HCTAB, 20	/D
3650	1560	0260	260	/DIAL
3651	1561	0164	164	/A/D
3652	1562	0354	354	/MODE
3653	1563	0310	310	/CTRL/H
3654	1564	0454	454	/HALT ALL
3655	1565	0310	310	/HN
3656	1566	0454	454	/HLT INST
3657	1567	0310	310	/P
3660	1570	0454	454	/POLARITY
3661	1571	0276	276	/F
3662	1572	0454	454	/FREEZE
3663	1573	0164	164	/PAUSE
3664	1574	0354	354	/MODE
3665	1575	0310	310	/GN
3666	1576	0454	454	/GO
3667	1577	0310	310	/S
3670	1600	0454	454	/SETUP
3671	1601	0310	310	/W
3672	1602	0454	454	/WRITE
3673	1603	0310	310	/C
3674	1604	0454	454	/CATACAL
3675	1605	0310	310	/M
3676	1606	0454	454	/MAGSPY
3677	1607	0310	310	/L
3700	1610	0454	454	/LOADER
3701	1611	0310	310	/P
3702	1612	0454	454	/POLARITY
3703	1613	0276	276	/F
3704	1614	0454	454	/FREEZE
3705				/DIAL ENCODED MESSAGE
3706				/FORMAT IS:
3707				/D DIAL
3710				/
3711				/
3712				/ A/D MODE
3713				/
3714				/ CTRL/H HALT ALL INSTRUMENTS
3715				/ HN HALT INSTRUMENT N
3716				/ P POLARITY
3717				/ F FREEZE
3720				/
3721				/ PAUSE MODE
3722				/ GN GO INSTRUMENT N
3723				/ S SETUP
3724				/ W WRITE TAPE
3725				/ C CATACAL
3726				/ M MAGSPY
3727				/ L LOADER
3730				/ P POLARITY
3731				/ F FREEZE
3732	1615	0604		
3732	1616	3604		
3732	1617	1101		
3732				HLPTAB, TEXT 9FD+DIAL
3733	1620	1443		

3733	1621	0601	
3733	1622	5704	
3733	1623	3615	
3733	1624	1704	
3733			FA/D+MODE
3734	1625	0543	
3734	1626	0324	
3734	1627	2214	
3734	1630	5710	
3734	1631	3610	
3734	1632	0114	
3734	1633	2440	
3734	1634	0114	
3734	1635	1440	
3734	1636	1116	
3734	1637	2324	
3734	1640	2225	
3734	1641	1505	
3734	1642	1624	
3734			CTRL/H+HALT ALL INSTRUMENTS
3735	1643	2343	
3735	1644	1016	
3735	1645	3610	
3735	1646	0114	
3735	1647	2440	
3735	1650	1116	
3735	1651	2324	
3735	1652	2225	
3735	1653	1505	
3735	1654	1624	
3735			HN+HALT INSTRUMENT N
3736	1655	4016	
3736	1656	4320	
3736	1657	3620	
3736	1660	1714	
3736	1661	0122	
3736	1662	1124	
3736			P+POLARITY
3737	1663	3143	
3737	1664	4006	
3737	1665	3606	
3737	1666	2205	
3737	1667	0532	
3737			F+FREEZE
3740	1670	0543	
3740	1671	0620	
3740	1672	0125	
3740	1673	2305	
3740	1674	3615	
3740	1675	1704	
3740			FPAUSE+MODE
3741	1676	0543	
3741	1677	0716	
3741	1700	3607	
3741	1701	1740	
3741	1702	1116	
3741	1703	2324	
3741	1704	2225	
3741	1705	1505	
3741	1706	1624	
3741			GN+GO INSTRUMENT N
3742	1707	4016	
3742	1710	4323	

3742	1711	3623	
3742	1712	0524	
3742			S+SETUP
3743	1713	2520	
3743	1714	4327	
3743	1715	3627	
3743	1716	2211	
3743	1717	2405	
3743	1720	4024	
3743	1721	0120	
3743			W+WRITE TAPE
3744	1722	0543	
3744	1723	0336	
3744	1724	0301	
3744	1725	2401	
3744	1726	0301	
3744			C+CATACAL
3745	1727	1443	
3745	1730	1536	
3745	1731	1501	
3745	1732	0723	
3745			M+MAGSPY
3746	1733	2031	
3746	1734	4314	
3746	1735	3614	
3746	1736	1701	
3746	1737	0405	
3746			L+LOADER
3747	1740	2243	
3747	1741	2036	
3747	1742	2017	
3747	1743	1401	
3747	1744	2211	
3747			P+POLARITY
3750	1745	2431	
3750	1746	4340	
3750	1747	0636	
3750	1750	0622	
3750	1751	0505	
3750			F+FREEZE
3751	1752	3205	
3751	1753	4377	
3751			?9
3752	1754	0000	AAAS1, 0
3753			/QANDA AND SETUP ROUTINES
3754			SEGMENT 2
3755			QAFLD=4000
3756			*1
3757	0001	1020	LDA I
3760	0002	0010	EXTMTP
3761	0003	0001	AXO
3762	0004	0707	CHK
3763	0005	0270	270
3764	0006	0603	LIF 3
3765	0007	7001	JMP MEMCHK
3766			/DETERMINE
3767			/CORE SIZE
3770	0015	0643	DIAL, LDF 3
3771	0016	0721	RCG I
3772	0017	7300	7300
3773	0020	6023	START, JMP SETUP
3774	0021	0440	CHKSNS, SNS 0
3775	0022	7066	JMP QARFSH
			/ON SSW0=1

3776	0023	0002	SETUP,	PDP	
3777				Pmode	
4000	4024	4763		JMS I KFRSTR	/INIT ALL
4001	4025	6141		LINC	/DSC
4002				Lmode	/PARAM INPUT
4003	0026	7013	MESS1,	JMP QAINIT	/DO -
4004	0027	0535		TXT1	/1 LINC TAPE
4005	0030	0524		ANSWER	/2 KEYBOARD
4006	0031	6021		JMP CHKSNS	
4007	0032	1300		LDH	
4010	0033	4524		ANSWER+HBIT	
4011	0034	1420		SHD I	
4012	0035	6100		6100	/READ ?
4013	0036	6344		JMP READ	/YES
4014	0037	1460		SAE I	
4015	0040	0062		62	/KEYBRD ?
4016	0041	6026		JMP MESS1	/NO ERROR
4017	0042	0002		PDP	
4020				Pmode	/INIT ARGS FOR
4021	4043	4764	MESS2,	JMS I KRESTR	/DEV K
4022	4044	4773		JMS I KLOHI	
4023	4045	6141		LINC	/DSC
4024				Lmode	/INST K IS -
4025	0046	7013	REMS2,	JMP QAINIT	/((1=ANA, 2=ENC)
4026	0047	0565		TXT2	/ST PNT =====
4027	0050	0524		ANSWER	/TOT PT =====
4030	0051	6021		JMP CHKSNS	/SAM FREQ =-
4031	0052	0074		SET I ANSPTR	
4032	0053	0525		ANSWER+1	
4033	0054	1300		LDH	/CHK ANA OR
4034	0055	4524		ANSWER+HBIT	/ENC
4035	0056	1420		SHD I	
4036	0057	6100		6100	/ANALOG ?
4037	0060	6073		JMP SETANA	/YES
4040	0061	1460		SAE I	
4041	0062	0062		62	/ENCODED ?
4042	0063	6046		JMP REMS2	/NO ERROR
4043	0064	0002		PDP	
4044				Fmode	
4045	4065	7200		CLA	
4046	4066	1422		TAD I ENCPTR	/ANY MORE
4047	4067	7450		SNA	/ENCODERS ?
4050	4070	5765		JMP I KERENC	/NO ENC ERROR
4051	4071	2022		ISZ ENCPTR	/SET FOR NXT
4052	4072	5276		JMP A1CONT	
4053				Lmode	
4054	0073	1020	SETANA,	LDA I	
4055	0074	4000		ANAVAL	
4056	0075	0002		PDP	
4057				Pmode	
4060	4076	3410	A1CONT,	DCA I INIPTR	/SELECT WD TO
4061					/INIT TABLE
4062	4077	2052		ISZ CNTR	/UP DV CNT
4063					/X COORD DV K
4064	4100	4463	A2LOP,	JMS I KDIG	/GET 1 DIGIT
4065					/FROM QA FLD
4066	4101	5243		JMP MESS2	/INPUT ERROR
4067	4102	5305		JMP A2CONT	/DONE QA FLD
4070	4103	4766		JMS I KDSCL4	/1 BCD VAL IS
4071					/AC SCALE AND
4072					/PACK IN DAC
4073	4104	5300		JMP A2LOP	
4074	4105	4767	A2CONT,	JMS I KUPACK	/UNPACK AND PUT

4075					/IN TABLE
4076					/TOTAL POINTS
4077	4106	4463	A3LOP,	JMS I KDIG	/GET 1 DIGIT
4100	4107	5243		JMP MESS2	/INPUT ERROR
4101	4110	5313		JMP A3CONT	/DONE QA FLD
4102	4111	4464		JMS I KMUL10	/DECIMAL TO
4103					/OCTAL CVERT
4104					/ANSWER IN DAC
4105	4112	5306		JMP A3LOP	
4106	4113	7200	A3CONT,	CLA	/CHK FOR
4107	4114	1044		TAD DBLLO	/NO POINTS
4110	4115	7640		SZA CLA	
4111	4116	5322		JMP NOT0	
4112	4117	1045		TAD DBLHI	
4113	4120	7650		SNA CLA	
4114	4121	5770		JMP I KER0PT	/ERR 0 PTS
4115	4122	1021	NOT0,	TAD CURNLO	/GET ST ADDR
4116	4123	3046		DCA ARGLO	/NXT DV
4117	4124	1020		TAD CURNHI	/=TOT PTS+
4120	4125	3047		DCA ARGHI	/ST ADDR DV K
4121	4126	4437		JMS I KDADD	
4122	4127	1044		TAD DBLLO	/SAVE NXT SA
4123	4130	3021		DCA CURNLO	
4124	4131	1045		TAD DBLHI	
4125	4132	3020		DCA CURNHI	
4126	4133	4771		JMS I KDCIA	/2S COM DAC
4127	4134	1044		TAD DBLLO	
4130	4135	3410		DCA I INIPTR	/=SA ADDR
4131	4136	1045		TAD DBLHI	/NEXT DEV
4132	4137	3410		DCA I INIPTR	
4133	4140	1044		TAD DBLLO	/SAVE FOR
4134	4141	3023		DCA QUEIN	/GBLK SR
4135	4142	1045		TAD DBLHI	
4136	4143	3024		DCA QUEOUT	
4137	4144	4774		JMS I KP1DAC	
4140	4145	1044		TAD DBLLO	/=END ADDR
4141	4146	3410		DCA I INIPTR	/DEV K
4142	4147	1045		TAD DBLHI	
4143	4150	3410		DCA I INIPTR	
4144					/CHK FOR
4145	4151	1032		TAD MAXCDF	/BUFFER FULL
4146	4152	3047		DCA ARGHI	
4147	4153	7040		CMA	
4150	4154	3046		DCA ARGLO	
4151	4155	4437		JMS I KDADD	/IF -END ADR IS
4152					/GT BUF LEN
4153					/BUF IS EXCEEDED
4154	4156	7420		SNL	/IS DAC = ?
4155	4157	5762		JMP I KCONMS	
4156	4160	4772		JMS I KERFUL	/YES BUF ERROR
4157	4161	5243		JMP MESS2	
4160	4162	4200	KCONMS,	CONMS2	
4161	4163	3200	KFRSTR,	FRSTR	
4162	4164	3233	KRESTR,	RESTR	
4163	4165	4303	KERENC,	ERRENC	
4164	4166	3043	KDSCL4,	DSCL4	
4165	4167	3055	KUPACK,	UPACK	
4166	4170	4311	KER0PT,	ERR0PT	
4167	4171	3030	KDCIA,	DCIA	
4170	4172	4317	KERFUL,	ERRFUL	
4171	4173	3406	KLOHI,	LOHI	
4172	4174	3400	KP1DAC,	P1DAC	
4173				*QAFLD+200	

4174	4200	1044	CONMS2,	TAD DBLLO	
4175	4201	7640		SZA CLA	/SET BUFF FULL
4176	4202	5207		JMP OK2	
4177	4203	1045		TAD DBLHI	/SWITCH
4200	4204	7640		SZA CLA	/7777=FULL
4201	4205	5207		JMP OK2	
4202	4206	7040		CMA	/ITS FULL
4203	4207	3026	OK2,	DCA STATUS	
4204	4210	3044		DCA DBLLO	/SAM FREQ
4205	4211	4463	A4,	JMS I KDIG	/1 DIGIT
4206	4212	5700		JMP I KMESS2	/INPUT ERROR
4207	4213	5216		JMP A4CONT	/DONE QA FLD
4210	4214	4464		JMS I KMUL10	/DEC TO OCT
4211	4215	5211		JMP A4	
4212	4216	7200	A4CONT,	CLA	
4213	4217	1044		TAD DBLLO	/CHK GT 10
4214	4220	1301		TAD NEG12	
4215	4221	7740		SMA SZA CLA	
4216	4222	5327		JMP ERFREQ	/ERROR FREQ
4217	4223	1044		TAD DBLLO	/SET 0=1
4220	4224	7450		SNA	
4221	4225	7001		IAC	
4222	4226	7041		CIA	/- FREQ TO TAB
4223	4227	3050		DCA TEMPLO	
4224	4230	1050		TAD TEMPLO	
4225	4231	3410		DCA I INIPTR	
4226	4232	1050		TAD TEMPLO	
4227	4233	3410		DCA I INIPTR	
4230	4234	1050		TAD TEMPLO	
4231	4235	3410		DCA I INIPTR	
4232	4236	6141		LINC	/DSC
4233				LMODE	/DO -
4234	0237	7013	MESS3,	JMP QAINIT	/1 MORE INSTS
4235	0240	0650		TXT3	/2 START
4236	0241	0524		ANSWER	
4237	0242	6021		JMP CHKSNS	
4240	0243	1300		LDH	
4241	0244	4524		ANSWER+HBIT	
4242	0245	1420		SHD I	
4243	0246	6200		6200	/START ?
4244	0247	6270		JMP ALLDUN	/YES
4245	0250	1460		SAE I	
4246	0251	0061		61	/MORE DVS ?
4247	0252	6237		JMP MESS3	/NO ERROR
4250	0253	0002		PDP	
4251				PMODE	/CHK FOR NO
4252	4254	7200		CLA	/MORE DEVICES
4253	4255	1052		TAD CNTR	/=NUM DEFINED
4254	4256	1061		TAD MAXDVN	/=NUM ALLOWED
4255	4257	7650		SNA CLA	
4256	4260	5335		JMP ERRDEV	/ERROR NO MORE
4257					/DEVICES
4260	4261	1026		TAD STATUS	/CHK FOR NO
4261	4262	7700		SMA CLA	/MORE PTS
4262	4263	5266		JMP OK1	
4263	4264	4317		JMS ERRFUL	/NONE LEFT FOR
4264	4265	5236		JMP MESS3-1	/NEXT DEV
4265	4266	4677	OK1,	JMS I KNXSTR	/SET ARGS FOR
4266					/DEVICE K+1
4267	4267	5700		JMP I KMESS2	/DO IT ALL
4270					/OVER AGAIN
4271				LMODE	
4272	0270	1020	ALLDUN,	LDA I	

4273	0271	0020		20	/I\O PRESET
4274	0272	0004		ESF	
4275	0273	0002		PDP	
4276				PMODE	
4277	4274	4702		JMS I KGBLK	/GET BLK CNT
4300	4275	5676		JMP I ,+1	
4301	4276	3312		QADUN	
4302	4277	5255	KNXSTR,	NXSTR	
4303	4300	4043	KMESS2,	MESS2	
4304	4301	7766	NEG12,	-12	
4305	4302	3430	KGBLK,	GBLK	
4306	4303	6141	ERRENC,	LINC	
4307				LMODE	/DSC
4310	0304	7013		JMP QAINIT	/ERR ENC
4311	0305	1006		TXT14	
4312	0306	0000		0	
4313	0307	6021		JMP CHKSNS	
4314	0310	6046		JMP REMS2	
4315	0311	6141	ERR0PT,	6141	/LINC
4316	0312	7013		JMP QAINIT	/DSC
4317	0313	0757		TXT10	/ERR 0 PTS
4320	0314	0000		0	
4321	0315	6021		JMP CHKSNS	
4322	0316	6042		JMP MESS2=1	
4323	0317	0000	ERRFUL,	0	
4324	0320	6141		6141	/LINC
4325	0321	7013		JMP QAINIT	/DSC
4326	0322	0765		TXT11	/ERR BUF FULL
4327	0323	0000		0	
4330	0324	6021		JMP CHKSNS	
4331	0325	0002		PDP	
4332				PMODE	
4333	4326	5717		JMP I ERRFUL	
4334				LMODE	
4335	0327	6141	ERFREQ,	6141	/DSC
4336	0330	7013		JMP QAINIT	/ERR FREQ
4337	0331	0774		TXT12	
4340	0332	0000	FREE,	0	
4341	0333	6021		JMP CHKSNS	
4342	0334	6042		JMP MESS2=1	
4343	0335	6141	ERRDEV,	6141	/DSC
4344	0336	7013		JMP QAINIT	/ERR DEV
4345	0337	1001		TXT13	
4346	0340	0000		0	
4347	0341	6021		JMP CHKSNS	
4350	0342	6237		JMP MESS3	
4351	0343	6141		6141	/LINC
4352	0344	7013	READ,	JMP QAINIT	/DSC
4353	0345	0705		TXT5	/READ TBLK ==-
4354	0346	0524		ANSWER	/UNIT -
4355	0347	6021		JMP CHKSNS	
4356	0350	0074		SET I ANSPTR	
4357	0351	0524		ANSWER	
4360	0352	0002		PDP	
4361				PMODE	
4362	4353	5754		JMP I ,+1	
4363	4354	1064		DOROC	
4364				PAGE	
4365	4400	6002	WRIT,	IOF	/IGNORE KBRD
4366					/DURING WRITE
4367	4401	1507		TAD I KBLKCT	/-NUM TO DO
4370	4402	7041		CIA	
4371	4403	6141		LINC	

4372			LMODE	
4373	0404	4332	STC FREE	
4374	0405	0074	SET I ANSPTR	/PUT TBLK
4375	0406	0723	TXT6+3	/NUM (3 DIGITS)
4376	0407	2332	ADD FREE	/IN QANDA TEXT
4377	0410	0306	ROR 6	/AT TXT6+3,
4400	0411	1560	BCL I	/TXT6+4,
4401	0412	7770	7770	/AND TXT6+5
4402	0413	2513	ADD PL60	
4403	0414	1374	STH I ANSPTR	
4404	0415	1000	LDA	
4405	0416	0332	FREE	
4406	0417	0303	ROR 3	
4407	0420	1560	BCL I	
4410	0421	7770	7770	
4411	0422	2513	ADD PL60	
4412	0423	1374	STH I ANSPTR	
4413	0424	1000	LDA	
4414	0425	0332	FREE	
4415	0426	1560	BCL I	
4416	0427	7770	7770	
4417	0430	2513	ADD PL60	
4420	0431	1374	STH I ANSPTR	
4421	0432	0643	LDF 3	/EXP NAME
4422	0433	7013	JMP QAINIT	/-----
4423	0434	0672	TXT4	
4424	0435	2005	ANS4+2000	
4425	0436	7066	JMP QARFSH	
4426	0437	1020	LDA I	/PATCH SPACE
4427	0440	0040	40	/AROUND NAME
4430	0441	1340	STH	/FOR MAGSPY
4431	0442	2005	ANS4+2000	
4432	0443	7013	JMP QAINIT	/DSC
4433	0444	0720	TXT6	/WRITE ---
4434	0445	0524	ANSWER	/TBLKS
4435	0446	7066	JMP QARFSH	/1ST TBLK ---
4436	0447	0074	SET I ANSPTR	/UNIT -
4437	0450	0524	ANSWER	
4440	0451	0002	PDP	
4441			PMODE	
4442	4452	5653	JMP I ,+1	
4443	4453	1746	DOWRC	
4444			LMODE	
4445	0454	7013	JMP QAINIT	/DSC
4446	0455	0747	TXT7	/LOAD -----
4447	0456	0524	ANSWER	/UNIT -
4450	0457	7066	JMP QARFSH	
4451	0460	1300	LDH	
4452	0461	4524	ANSWER+HBIT	
4453	0462	0470	AZE I	
4454	0463	6454	JMP LOAD	
4455	0464	0600	LIF 0	
4456	0465	7525	JMP KLOAD	
4457	0466	0000	DIGCHK, 0	/CHK NUMERIC
4460	0467	6141	6141	/INPUT
4461	0470	1334	DLOOP, LDH I ANSPTR	/FROM QA FLD
4462	0471	1420	SHD I	
4463	0472	7400	QAEOL1	
4464	0473	6517	JMP EOLXIT-QAFLD	
4465	0474	0470	AZE I	/BLANK INPUT ?
4466	0475	6470	JMP DLOOP	/YES IGNORE
4467	0476	1420	SHD I	
4470	0477	3400	QAEOM1	

4471	0500	6517		JMP EOLXIT-QAFLO	
4472	0501	1120		ADA I	
4473	0502	7720		=57	
4474	0503	0451		APO	/DIGIT ?
4475	0504	6522		JMP ERRXIT-QAFLO	/NO ILLEGAL
4476					/INPUT
4477	0505	1120		ADA I	
4500	0506	7765	UPLIM,	=12	/=10 OR =12
4501	0507	0471		APO I	/DIGIT ?
4502	0510	6522		JMP ERRXIT-QAFLO	
4503	0511	1314		LDH ANSPTR	
4504	0512	1560		BCL I	/PUT DIGIT
4505	0513	0060	PL60,	60	/IN AC
4506	0514	0002		PDP	
4507				PMODE	
4510	4515	2266		ISZ DIGCHK	/P+3 IS NORMAL
4511	4516	7410		SKP	
4512	4517	0002	EOLXIT,	2	/PDP
4513	4520	2266		ISZ DIGCHK	/P+2 IS EOL
4514	4521	7410		SKP	
4515	4522	0002	ERRXIT,	2	/P+1 IS ERR
4516	4523	5666		JMP I DIGCHK	/RTN
4517				LMODE	
4520	0524	0000	ANSWER,	0	
4521	0525	0000		0	
4522	0526	0000		0	
4523	0527	0000		0	
4524	0530	0000		0	
4525	0531	0000		0	
4526	0532	0000		0	
4527	0533	0000		0	
4528	0534	0000		0	
4531	0535	0620			
4531	0536	0122			
4531	0537	0110			
4531	0540	0224			
4531	0541	0222			
4531	0542	4011			
4531	0543	1020			
4531			TXT1,	TEXT ?FPARAMETER INPUT	
4532	0544	2524			
4532	0545	4306			
4532	0546	0417			
4532			F00<1		
4533	0547	7401			
4533	0550	4306			
4533	0551	0140			
4533	0552	1411			
4533	0553	1003			
4533	0554	4024			
4533	0555	0120			
4533			F1 LINC TAPE		
4534	0556	0543			
4534	0557	0002			
4534	0560	4013			
4534	0561	0531			
4534	0562	0217			
4534	0563	0122			
4534	0564	0434			
4534			F2 KEYBOARD\?		
4535	0565	0511			
4535	0566	1020			
4535	0567	2422			

4535	0570	2215
4535	0571	0210
4535	0572	2440
4535	0573	3031
4535	0574	4011
4535	0575	2374
4535		
4536	0576	0143
4536		
4537	0577	4043
4537	0600	4040
4537	0601	4040
4537	0602	2061
4537	0603	7001
4537	0604	1601
4537	0605	1417
4537	0606	0704
4537	0607	4062
4537	0610	7000
4537	0611	1603
4537	0612	1704
4537	0613	0004
4537		
4540	0614	0143
4540	0615	0023
4540	0616	2401
4540	0617	2224
4540	0620	1116
4540	0621	0740
4540	0622	2017
4540	0623	1116
4540	0624	2470
4540		
4541	0625	7465
4541	0626	4006
4541	0627	2417
4541	0630	2401
4541	0631	1440
4541	0632	2017
4541	0633	1116
4541	0634	2423
4541	0635	7574
4541		
4542	0636	6043
4542	0637	0623
4542	0640	0115
4542	0641	2014
4542	0642	1116
4542	0643	0740
4542	0644	0622
4542	0645	0021
4542	0646	7074
4542	0647	6234
4542		
4543	0650	0604
4543	0651	1774
4543		
4544	0652	6143
4544	0653	0661
4544	0654	4015
4544	0655	1722
4544	0656	0040
4544	0657	1116

TXT2, TEXT ?FINSTRUMENT XY IS<1

(1=ANALOG, 2=ENCODED)

FSTARTING POINT=<5

FTOTAL POINTS=<5

FSAMPLING FREQ=<2\?

TXT3, TEXT ?FDO<1

4544	0600	2324	
4544	0601	2223	
4544	0662	1505	
4544	0663	1024	
4544			F1 MORE INSTRUMENTS
4545	0664	2343	
4545	0665	0662	
4545	0666	4023	
4545	0667	2401	
4545	0670	2224	
4545	0671	3400	
4545			F2 START\?
4546	0672	0605	
4546	0673	3020	
4546	0674	0522	
4546	0675	1115	
4546	0676	0516	
4546	0677	2440	
4546	0700	1601	
4546			TXT4, TEXT ?FEXPERIMENT NAME
4547	0701	1505	
4547	0702	4306	
4547	0703	7470	
4547	0704	3400	
4547			F<8\?
4550	0705	0622	
4550	0706	0501	
4550	0707	0440	
4550	0710	2402	
4550	0711	1413	
4550			TXT5, TEXT ?FREAD TBLK<3
4551	0712	7463	
4551	0713	4306	
4551	0714	2516	
4551	0715	1124	
4551	0716	7461	
4551	0717	3400	
4551			FUNIT<1\?
4552	0720	0627	
4552	0721	2211	
4552	0722	2405	
4552	0723	4030	
4552	0724	3132	
4552	0725	4050	
4552	0726	1703	
4552	0727	2451	
4552	0730	4024	
4552	0731	0214	
4552			TXT6, TEXT ?FWRITE XYZ (OCT) TBLKS
4553	0732	1323	
4553	0733	4306	
4553	0734	0611	
4553	0735	2223	
4553	0736	2440	
4553	0737	2402	
4553	0740	1413	
4553	0741	7574	
4553			FFIRST TBLK<3
4554	0742	6343	
4554	0743	0625	
4554	0744	1611	
4554	0745	2474	
4554	0746	6134	

4554			FUNIT<1\?	
4555	0747	0614		
4555	0750	1701		
4555	0751	0474		
4555			TXT7, TEXT ?FLOAD<8	
4556	0752	7043		
4556	0753	0625		
4556	0754	1611		
4556	0755	2474		
4556	0756	6134		
4556			FUNIT<1\?	
4557	0757	0605		
4557	0760	2222		
4557	0761	4060		
4557	0762	4020		
4557	0763	2423		
4557	0764	3400		
4557			TXT10, TEXT ?FERR 0 PTS\?	
4560	0765	0605		
4560	0766	2222		
4560	0767	4002		
4560	0770	2506		
4560	0771	4006		
4560	0772	2514		
4560	0773	1434		
4560			TXT11, TEXT ?FERR BUF FULL\?	
4561	0774	0605		
4561	0775	2222		
4561	0776	4006		
4561	0777	2205		
4561	1000	2134		
4561			TXT12, TEXT ?FERR FREQ\?	
4562	1001	0605		
4562	1002	2222		
4562	1003	4011		
4562	1004	1623		
4562	1005	2434		
4562			TXT13, TEXT ?FERR INST\?	
4563	1006	0605		
4563	1007	2222		
4563	1010	4005		
4563	1011	1603		
4563	1012	3400		
4563			TXT14, TEXT ?FERR ENC\?	
4564			NOLIST	
5530	1701	0000	AAAS2, 0	
5531			SEGMENT 3	
5532			*1	
5533	0001	4040	FIRST, 4040	/SPACE FMT
5534	0002	4040	4040	/AROUND NAME
5535	0003	4040	4040	/FOR MAGSPY
5536	0004	4040	4040	
5537	0005	0000	ANS4, 0	/EXPERIMENT
5540	0006	0000	0	/NAME
5541	0007	0000	0	
5542	0010	0000	0	
5543	0011	0000	0	
5544	0012	4040	4040	
5545	0013	4040	4040	
5546	0014	4040	4040	
5547	0015	4040	4040	
5550			PMODE	
5551			/INITIALIZATION TABLE FILLED BY	

```

5552          /SETUP AND READ ROUTS
5553          /17 VALUES PER DEVICE 4 DEVICES
5554          6016 0001 INISTR, 1 /SA LO
5555          6017 0002          2 /SA HI
5556          6020 0003          3 /-SA LO
5557          6021 0004          4 /-SA HI
5560          6022 0005          5 /SELECT WORD
5561          6023 0006          6 /X COORD 1S
5562          6024 0007          7 /10S
5563          6025 0010          8 /100S
5564          6026 0011          9 /1000S
5565          6027 0010         10 /10000S
5566          6030 0011         11 /-SA LO NXT DV
5567          6031 0012         12 /-SA HI NXT DV
5570          6032 0013         13 /-(EA LO,EA HI)
5571          6033 0014         14 /
5572          6034 0015         15 /-SAM FREQ
5573          6035 0016         16 /-SAM FREQ
5574          6036 0017 INIEND, 17 /-SAM FREQ
5575          6037 0001          1 /DEVICE 1
5576          6040 0002          2
5577          6041 0003          3
5600          6042 0004          4
5601          6043 0005          5
5602          6044 0006          6
5603          6045 0007          7
5604          6046 0010          8
5605          6047 0011          9
5606          6050 0010         10
5607          6051 0011         11
5610          6052 0012         12
5611          6053 0013         13
5612          6054 0014         14
5613          6055 0015         15
5614          6056 0016         16
5615          6057 0017         17
5616          6060 0001          1 /DEVICE 2
5617          6061 0002          2
5620          6062 0003          3
5621          6063 0004          4
5622          6064 0005          5
5623          6065 0006          6
5624          6066 0007          7
5625          6067 0010          8
5626          6070 0011          9
5627          6071 0010         10
5630          6072 0011         11
5631          6073 0012         12
5632          6074 0013         13
5633          6075 0014         14
5634          6076 0015         15
5635          6077 0016         16
5636          6100 0017         17
5637          6101 0001          1 /DEVICE 3
5640          6102 0002          2
5641          6103 0003          3
5642          6104 0004          4
5643          6105 0005          5
5644          6106 0006          6
5645          6107 0007          7
5646          6110 0010          8
5647          6111 0011          9
5650          6112 0010         10

```

5651	6113	0011	11	
5652	6114	0012	12	
5653	6115	0013	13	
5654	6116	0014	14	
5655	6117	0015	15	
5656	6120	0016	16	
5657	6121	0017	17	
5660	6122	0001	1	/DEVICE 4
5661	6123	0002	2	
5662	6124	0003	3	
5663	6125	0004	4	
5664	6126	0005	5	
5665	6127	0006	6	
5666	6130	0007	7	
5667	6131	0010	8	
5670	6132	0011	9	
5671	6133	0010	10	
5672	6134	0011	11	
5673	6135	0012	12	
5674	6136	0013	13	
5675	6137	0014	14	
5676	6140	0015	15	
5677	6141	0016	16	
5700	6142	0017	17	
5701	6143	0000		BLKCNT, 0
5702	6144	0000		LAST, 0
5703				/TABLE OF ADDR THAT TELL WHERE
5704				/INIT VALS GO
5705	6145	6272		INIAD, TISTR0 /DEVICE 0
5706	6146	6273		TISTR0+1
5707	6147	6443		LIMVL0
5710	6150	6444		LIMVL0+1
5711	6151	6274		TISTR0+2
5712	6152	6471		BCD0
5713	6153	6472		BCD0+1
5714	6154	6473		BCD0+2
5715	6155	6474		BCD0+3
5716	6156	6475		BCD0+4
5717	6157	6447		LIMVL1
5720	6160	6450		LIMVL1+1
5721	6161	0314		ELODV0
5722	6162	0315		EHIOV0
5723	6163	6275		TISTR0+3
5724	6164	6276		TISTR0+4
5725	6165	6445		LIMVL0+2
5726	6166	6277		TISTR1 /DEVICE 1
5727	6167	6300		TISTR1+1
5730	6170	6447		LIMVL1
5731	6171	6450		LIMVL1+1
5732	6172	6301		TISTR1+2
5733	6173	6476		BCD1
5734	6174	6477		BCD1+1
5735	6175	6500		BCD1+2
5736	6176	6501		BCD1+3
5737	6177	6502		BCD1+4
5740	6200	6453		LIMVL2
5741	6201	6454		LIMVL2+1
5742	6202	0336		ELODV1
5743	6203	0337		EHIOV1
5744	6204	6302		TISTR1+3
5745	6205	6303		TISTR1+4
5746	6206	6451		LIMVL1+2
5747	6207	6304		TISTR2 /DEVICE 2

5750	6210	6305	TISTR2+1	
5751	6211	6453	LIMVL2	
5752	6212	6454	LIMVL2+1	
5753	6213	6306	TISTR2+2	
5754	6214	6503	BCD2	
5755	6215	6504	BCD2+1	
5756	6216	6505	BCD2+2	
5757	6217	6506	BCD2+3	
5760	6220	6507	BCD2+4	
5761	6221	6457	LIMVL3	
5762	6222	6460	LIMVL3+1	
5763	6223	0415	ELODV2	
5764	6224	0416	EHIDV2	
5765	6225	6307	TISTR2+3	
5766	6226	6310	TISTR2+4	
5767	6227	6455	LIMVL2+2	
5770	6230	6311	TISTR3	/DEVICE 3
5771	6231	6312	TISTR3+1	
5772	6232	6457	LIMVL3	
5773	6233	6460	LIMVL3+1	
5774	6234	6313	TISTR3+2	
5775	6235	6510	BCD3	
5776	6236	6511	BCD3+1	
5777	6237	6512	BCD3+2	
6000	6240	6513	BCD3+3	
6001	6241	6514	BCD3+4	
6002	6242	6463	LIMVL4	
6003	6243	6464	LIMVL4+1	
6004	6244	0440	ELODV3	
6005	6245	0441	EHIDV3	
6006	6246	6314	TISTR3+3	
6007	6247	6315	TISTR3+4	
6010	6250	6461	LIMVL3+2	
6011	6251	6316	TISTR4	/DEVICE 4
6012	6252	6317	TISTR4+1	
6013	6253	6463	LIMVL4	
6014	6254	6464	LIMVL4+1	
6015	6255	6320	TISTR4+2	
6016	6256	6515	BCD4	
6017	6257	6516	BCD4+1	
6020	6260	6517	BCD4+2	
6021	6261	6520	BCD4+3	
6022	6262	6521	BCD4+4	
6023	6263	6467	LIMVL5	
6024	6264	6470	LIMVL5+1	
6025	6265	0463	ELODV4	
6026	6266	0464	EHIDV4	
6027	6267	6321	TISTR4+3	
6030	6270	6322	TISTR4+4	
6031	6271	6465	LIMVL4+2	
6032				
6033			/TABLE OF VALUES FOR GO ROUTS	
6034	6272	0001	TISTR0, 1	/SALO
6035	6273	0002	2	/SAHI
6036	6274	0003	3	/SELECT WD
6037	6275	0004	4	/-SAM FREQ
6040	6276	0005	5	/-SAM FREQ
6041	6277	0001	TISTR1, 1	/DV1
6042	6300	0002	2	
6043	6301	0003	3	
6044	6302	0004	4	
6045	6303	0005	5	
6046	6304	0001	TISTR2, 1	/DV2

6047	6305	0002		2
6050	6306	0003		3
6051	6307	0004		4
6052	6310	0005		5
6053	6311	0001	TISTR3, 1	/DV3
6054	6312	0002		2
6055	6313	0003		3
6056	6314	0004		4
6057	6315	0005		5
6060	6316	0001	TISTR4, 1	
6061	6317	0002		2
6062	6320	0003		3
6063	6321	0004		4
6064	6322	0005		5
6065			/TABLE OF ADDR FOR GO VALUES	
6066			/6 PER DEVICE	
6067	6323	0312	TIAD0,	LODV0
6070	6324	0311		HIDV0
6071	6325	0066		SELDV0
6072	6326	0307		SFDV0
6073	6327	0310		KSFDV0
6074	6330	0313		OSDV0
6075	6331	0334	TIAD1,	LODV1
6076	6332	0333		HIDV1
6077	6333	0067		SELDV1
6100	6334	0331		SFDV1
6101	6335	0332		KSFDV1
6102	6336	0335		OSDV1
6103	6337	0413	TIAD2,	LODV2
6104	6340	0412		HIDV2
6105	6341	0070		SELDV2
6106	6342	0410		SFDV2
6107	6343	0411		KSFDV2
6110	6344	0414		OSDV2
6111	6345	0436	TIAD3,	LODV3
6112	6346	0435		HIDV3
6113	6347	0071		SELDV3
6114	6350	0433		SFDV3
6115	6351	0434		KSFDV3
6116	6352	0437		OSDV3
6117	6353	0461	TIAD4,	LODV4
6120	6354	0460		HIDV4
6121	6355	0072		SELDV4
6122	6356	0456		SFDV4
6123	6357	0457		KSFDV4
6124	6360	0462		OSDV4
6125			/ADDR OF START OF EACH TABLE	
6126			/FOR GO ROUTS	
6127	6361	6272	VALTAB,	TISTR0
6130	6362	6277		TISTR1
6131	6363	6304		TISTR2
6132	6364	6311		TISTR3
6133	6365	6316		TISTR4
6134	6366	6323	ADR TAB,	TIAD0
6135	6367	6331		TIAD1
6136	6370	6337		TIAD2
6137	6371	6345		TIAD3
6140	6372	6353		TIAD4
6141			/TTY QUE HOLDS 20 MESS ADDRS MAX	
6142	6373	0001	STRQUE,	1
6143	6374	0002		2
6144	6375	0003		3
6145	6376	0004		4

6146	6377	0005		5	
6147	6400	0006		6	
6150	6401	0007		7	
6151	6402	0010		8	
6152	6403	0011		9	
6153	6404	0010		10	
6154	6405	0011		11	
6155	6406	0012		12	
6156	6407	0013		13	
6157	6410	0014		14	
6160	6411	0015		15	
6161	6412	0016		16	
6162	6413	0017		17	
6163	6414	0020		18	
6164	6415	0021		19	
6165	6416	0020	ENDQUE,	20	
6166			/TABLES FOR SCOPE ROUTINES		
6167					/POWERS OF TEN
6170					/FOR OCTAL TO
6171					/DECIMAL
6172					/CONVERSION
6173	6417	7747	RADVAL,	7747	/=100,000
6174	6420	4540		4540	
6175	6421	7775		7775	/=10,000
6176	6422	4360		4360	
6177	6423	7777		7777	/=1,000
6200	6424	6030		6030	
6201	6425	7777		7777	/=100
6202	6426	7634		=144	
6203	6427	7777		7777	/=10
6204	6430	7766		-12	
6205	6431	7634		-144	/=100
6206	6432	0144		+144	/+100
6207	6433	7766		-12	/=10
6210	6434	0012		+12	/+10
6211					/HOLDS BCD VAL
6212					/OF X COORD
6213	6435	0000	BCDVAL,	0	/100K
6214	6436	0000		0	/10K
6215	6437	0000		0	/1K
6216	6440	0000		0	/100S
6217	6441	0000		0	/10S
6220	6442	0000		0	/UNITS
6221					/DEVICE INFO
6222					/TABLE FOR
6223					/XYGET SETUP
6224					/AT QANDA STAGE
6225	6443	0000	LIMVL0,	0	/=ST ADDR DV0
6226	6444	0000		0	/LO, HI
6227	6445	0000		0	/= SAM FREQ DV0
6230	6446	6471		BCD0	/ADDR OF BCD
6231					/VAL FOR DEV 0
6232	6447	0000	LIMVL1,	0	/DV1 ARGS
6233	6450	0000		0	
6234	6451	0000		0	
6235	6452	6476		BCD1	
6236	6453	0000	LIMVL2,	0	/DV2 ARGS
6237	6454	0000		0	
6240	6455	0000		0	
6241	6456	6503		BCD2	
6242	6457	0000	LIMVL3,	0	/DV3 ARGS
6243	6460	0000		0	
6244	6461	0000		0	

6245	6462	6510		BCD3	
6246	6463	0000	LIMVL4,	0	/DV4 ARGS
6247	6464	0000		0	
6250	6465	0000		0	
6251	6466	6515		BCD4	
6252	6467	0000	LIMVL5,	0	/1ST UNUSED
6253	6470	0000		0	/BUFFER CORE
6254					/LOC COMES
6255					/RIGHT AFT LAST
6256					/USER DEFINED
6257					/DEVICE
6260	6471	0000	BCD0,	0	/HOLDS USER
6261	6472	0000		0	/DEFINED INIT
6262	6473	0000		0	/X VALUE
6263	6474	0000		0	/5 DIGITS
6264	6475	0000		0	/1 EACH DV
6265	6476	0000	BCD1,	0	/UNITS
6266	6477	0000		0	/TENS
6267	6500	0000		0	/HUNDREDS
6270	6501	0000		0	/THOUSANDS
6271	6502	0000		0	/10 THOUS
6272	6503	0000	BCD2,	0	
6273	6504	0000		0	
6274	6505	0000		0	
6275	6506	0000		0	
6276	6507	0000		0	
6277	6510	0000	BCD3,	0	
6300	6511	0000		0	
6301	6512	0000		0	
6302	6513	0000		0	
6303	6514	0000		0	
6304	6515	0000	BCD4,	0	
6305	6516	0000		0	
6306	6517	0000		0	
6307	6520	0000		0	
6310	6521	0000		0	
6311	6522	6567	NUM0,	MS0	/NUM TAGS
6312	6523	6573	NUM1,	MS1	/MUST BE
6313	6524	6577	NUM2,	MS2	/CONTIGUOUS
6314	6525	6603	NUM3,	MS3	
6315	6526	6607	NUM4,	MS4	
6316			/TISA MESSAGES		
6317	6527	0310	MSHLT,	310	/H
6320	6530	0301		301	/A
6321	6531	0314		314	/L
6322	6532	0324		324	/T
6323	6533	0240		240	/SPACE
6324	6534	0311		311	/I
6325	6535	0316		316	/N
6326	6536	0323		323	/S
6327	6537	0324		324	/T
6330	6540	0322		322	/R
6331	6541	0325		325	/U
6332	6542	0315		315	/M
6333	6543	0305		305	/E
6334	6544	0316		316	/N
6335	6545	0324		324	/T
6336	6546	0240		240	/SPACE
6337	6547	0000		0	/EOM
6340	6550	0307	MSG0,	307	/G
6341	6551	0317		317	/O
6342	6552	0240		240	/SPACE
6343	6553	0311		311	/I

6344	6554	0316		316	/N
6345	6555	0323		323	/S
6346	6556	0324		324	/T
6347	6557	0322		322	/R
6350	6560	0325		325	/U
6351	6561	0315		315	/M
6352	6562	0305		305	/E
6353	6563	0316		316	/N
6354	6564	0324		324	/T
6355	6565	0240		240	/SPACE
6356	6566	0000		0	/EOM
6357	6567	0260	MS0,	260	
6360	6570	0215		215	
6361	6571	0212		212	
6362	6572	0000		0	
6363	6573	0261	MS1,	261	
6364	6574	0215		215	
6365	6575	0212		212	
6366	6576	0000		0	
6367	6577	0262	MS2,	262	
6370	6600	0215		215	
6371	6601	0212		212	
6372	6602	0000		0	
6373	6603	0263	MS3,	263	
6374	6604	0215		215	
6375	6605	0212		212	
6376	6606	0000		0	
6377	6607	0264	MS4,	264	
6400	6610	0215		215	
6401	6611	0212		212	
6402	6612	0000		0	
6403	6613	0315	MSMOT,	315	/M
6404	6614	0317		317	/O
6405	6615	0324		324	/T
6406	6616	0311		311	/I
6407	6617	0317		317	/O
6410	6620	0316		316	/N
6411	6621	0215		215	/CR
6412	6622	0212		212	/LF
6413	6623	0000		0	/EOM
6414	6624	0306	MSFR,	306	/F
6415	6625	0322		322	/R
6416	6626	0305		305	/E
6417	6627	0305		305	/E
6420	6630	0332		332	/Z
6421	6631	0305		305	/E
6422	6632	0215		215	
6423	6633	0212		212	
6424	6634	0000		0	
6425	6635	0316	MSNOR,	316	/N
6426	6636	0317		317	/O
6427	6637	0322		322	/R
6430	6640	0315		315	/M
6431	6641	0301		301	/A
6432	6642	0314		314	/L
6433	6643	0215		215	
6434	6644	0212		212	
6435	6645	0000		0	
6436	6646	0311	MSINV,	311	/I
6437	6647	0316		316	/N
6440	6650	0326		326	/V
6441	6651	0305		305	/E
6442	6652	0322		322	/R

6443	6653	0324		324	/T
6444	6654	0305		305	/E
6445	6655	0304		304	/D
6446	6656	0215		215	
6447	6657	0212		212	
6450	6660	0000		0	
6451	6661	0320	ALLHLT,	320	/P
6452	6662	0301		301	/A
6453	6663	0325		325	/U
6454	6664	0323		323	/S
6455	6665	0305		305	/E
6456	6666	0240		240	/SP
6457	6667	0315		315	/M
6460	6670	0317		317	/O
6461	6671	0304		304	/D
6462	6672	0305		305	/E
6463	6673	0215		215	
6464	6674	0212		212	
6465	6675	0000		0	
6466	6676	0310	MSBUSY,	310	/H
6467	6677	0301		301	/A
6470	6700	0316		316	/N
6471	6701	0304		304	/D
6472	6702	0323		323	/S
6473	6703	0240		240	
6474	6704	0317		317	/O
6475	6705	0306		306	/F
6476	6706	0306		306	/F
6477	6707	0215		215	
6500	6710	0212		212	
6501	6711	0000		0	
6502	6712	0301	MSAD,	301	/A
6503	6713	0257		257	//
6504	6714	0304		304	/D
6505	6715	0240		240	
6506	6716	0315		315	/M
6507	6717	0317		317	/O
6510	6720	0304		304	/D
6511	6721	0305		305	/E
6512	6722	0215		215	
6513	6723	0212		212	
6514	6724	0000		0	
6515	6725	0000	AAAS3,	0	
6516			LMODE		
6517			*1000		
6520	1000	0004	P4,	4	
6521					/FIND OUT HOW
6522					/MUCH EXT MEM
6523	1001	0640	MEMCHK,	LDF 0	/PUT ANYTHING
6524	1002	1020		LDA I	/EXCEPT LDF K
6525	1003	0640		LDF 0	/IN LDF 0
6526	1004	1040		STA	
6527	1005	2000		2000	
6530	1006	1040		STA	/SET LDF
6531	1007	1013		NLDF	/INST
6532	1010	5000	LOOP,	ADD P4	/PUT LDF K
6533	1011	1040		STA	/K=4,10,...,34
6534	1012	1013		NLDF	/IN LOC NLDF
6535	1013	0000	NLDF,	0	
6536	1014	1040		STA	/PUT LDF K
6537	1015	2000		2000	/IN LDF K
6540	1016	1000		LDA	/GET IT
6541	1017	2000		2000	/BACK

6542	1020	1440	SAE	/DID WE
6543	1021	1013	NLDF	/MAKE IT ?
6544	1022	7035	JMP DUN	/NO
6545	1023	0640	LDF 0	/CHK FOR
6546	1024	1440	SAE	/CONTROL LOGIC
6547	1025	2000	2000	/ONLY
6550	1026	0467	SKP	/STILL OK
6551	1027	7035	JMP DUN	/4K ONLY
6552	1030	1460	SAE I	/CHK FOR
6553	1031	0674	LDF 34	/MAX CORE
6554	1032	7010	JMP LOOP	/DO SOME MORE
6555	1033	3000	ADD P4	/WE HAVE 32K
6556	1034	5013	STC NLDF	
6557	1035	1000	LDA	/PUT FIRST NON
6560	1036	1013	NLDF	/EXISTANT LDF
6561				/IN AC
6562	1037	1120	ADA I	
6563	1040	7137	-640	
6564	1041	0302	ROR 2	/=1ST NON CDF
6565	1042	1120	ADA I	
6566	1043	7776	-1	
6567	1044	0470	AZE I	
6570	1045	0011	CLR	
6571	1046	0002	PDP	
6572			PMODE	
6573	7047	6201	CDF 0	
6574	7050	3032	DCA MAXCDF	
6575	7051	1255	TAD P4040	
6576	7052	3656	DCA I K6K	/FIXUP 6000
6577	7053	5654	JMP I ,+1	
6600	7054	4023	SETUP	
6601	7055	4040	P4040, 4040	
6602	7056	6000	K6K, 6000	
6603			/DEFINITIONS FOR INPUT ROUTINES	
6604			ADRMN=7000	/1ST BUF ADDR
6605			CDFMIN=0	
6606			DRPMAX=20	/MAX X AX DROP
6607			WRMBLK=4000	/1ST MBLK FOR
6610				/WRITE
6611			FIRLDF=643	/1ST BUF LDF
6612			ROMBLK=6000	/1ST MBLK FOR
6613				/READ
6614			/ANALOG ASSIGNMENTS FOR DEVICES	
6615			/X AXIS	
6616			XAXDV0=10	
6617			XAXDV1=XAXDV0+1	
6620			XAXDV2=XAXDV1+1	
6621			XAXDV3=XAXDV2+1	
6622			XAXDV4=20	
6623			/Y AXIS	
6624			YAXDV0=14	
6625			YAXDV1=YAXDV0+1	
6626			YAXDV2=YAXDV1+1	
6627			YAXDV3=YAXDV2+1	
6630			YAXDV4=24	
6631			/DEFINITIONS FOR SCOPE ROUTINES	
6632			CHGHC=3600	/CHANGE HORIZ
6633				/COORD
6634			CR=0043	/EOL RH
6635			CR1=4300	/EOL LH
6636			ENDWD=0077	/END OF MESS
6637			ENDWD1=7700	/END OF MESS
6640			FUL=0006	/FULL SIZ CHARS

```

6641          FUL1=0600
6642          GRDPTR=14          /PTR TO GRID
6643          /TABLE
6644          GRDTAB=QAV         /ADDR OF GRID
6645          /TABLE
6646          HBIT=4000
6647          HCPTR=15          /PTR TO TABLE
6650          /HORIZ COORDS
6651          KURLN=-30         /- LENGTH OF
6652          /CURSOR
6653          MSPTR=17          /=XCORD
6654          /PTR TO DIAL
6655          /ENCODED MESS
6656          VCPTR=16         /PTR TO TABLE
6657          /VERT COORDS
6660          XCURHI=ARGHI     /PAGE 0
6661          XCURL0=ARGLO     /PAGE 0
6662          XYHC=0           /INIT HORIZ
6663          /COORD FOR
6664          /X-Y DISP
6665          /DEFINITIONS FOR QANDA SETUP ROUTINES
6666          ANAVAL=4000       /ANALOG SELECT
6667          ANSPTR=14
6670          CURNHI=ADMESS    /PAGE 0
6671          CURNLO=KBDBUF    /PAGE 0
6672          DVNMAX=-5
6673          ENCPTR=TTYFLG    /PAGE 0
6674          EXTMTP=10        /XOB CONSTANT
6675          QAEOL1=7400
6676          QAEOM1=3400
6677          /TISA 15 APR 70

```

0000 ERRORS

AAAP0 0111  
AAAS0 1765  
AAAS1 3754  
AAAS2 5701  
AAAS3 6725  
ADDR 0730  
ADMESS 0020  
ADMS 1504  
ADPTR 0011  
ADRMIN 7000  
ADRTAB 6366  
AGOMS 3515  
AHLTMS 3514  
ALLDUN 4270  
ALLHLT 6661  
ANALOG 0702  
ANARTN 0637  
ANAVAL 4000  
ANLDF 0347  
ANLIF 0346  
ANONO 1653  
ANRDC 1741  
ANSPTH 0014  
ANSWER 4524  
ANS4 6005  
ARGHI 0047  
ARGLO 0046  
ASAM 1513  
ASAVE 0352

=

ATEMP 1506  
AWRI 1762  
A1CONT 4076  
A2CONT 4105  
A2LOP 4100  
A3CONT 4113  
A3LOP 4106  
A4 4211  
A4CONT 4216  
BCDLOP 2600  
BCDPTR 0056  
BCDVAL 6435  
BCD0 6471  
BCD1 6476  
BCD2 6503  
BCD3 6510  
BCD4 6515  
BCNTR 0053  
BETA1 0003  
BETA2 0004  
BETA3 0005  
BLKCNT 6143  
BUFHI 2347  
BUFLO 2346  
BUFPTR 0054  
BUSYMS 1063  
BWMOT 2224  
BYHLT 3123  
BYMBLK 1137  
BYWRAP 2316  
C 1342  
CATA 1517  
CATPTR 1576  
CDFMIN 0000  
CDFVAL 0731  
CDF0 0031  
CHAR 1265  
CHGHC 3600  
CHGSAM 0720  
CHKADR 2233  
CHKLOP 3114  
CHKSNS 4021  
CHKSWI 3107  
CHKTTY 0245  
CHRL0P 1210  
CHRNXT 2075  
CHRPTR 1224  
CHRTAB 1223  
CLKCNT 3304  
CLKMOD 3305  
CLRLOP 3206  
CNTR 0052  
CONMS2 4200  
CR 0043  
CR1 4300  
CTRLH 1274  
CURCDF 0060  
CURCNT 2351  
CURDIS 2700  
CURLN 2751  
CURLOP 2722  
CURNHI 0020  
CURNLO 0021

CURRTN 2275  
D 0473  
DADD 2330  
DBLDEF 2470  
DBLHI 0045  
DBLLO 0044  
DBLLOP 2461  
DCIA 3030  
DCODEX 3527  
DCODEY 3534  
DEVNUM 0036  
DIAL 4015  
DIGCHK 4466  
DLOOP 4470  
DOMTP 1067  
DORDC 1664  
DOWRC 1746  
DRPMAX 0020  
DSCCHR 2732  
DSCHLP 2052  
DSCL 3017  
DSCL4 3043  
DSCXY 2122  
DUN 7035  
DUNBLK 1632  
DVNMAX 7773  
DVTYP 0027  
DV1 0321  
DV2 0400  
DV3 0423  
DV4 0446  
EHIDV0 0315  
EHIDV1 0357  
EHIDV2 0416  
EHIDV3 0441  
EHIDV4 0464  
ELODV0 0314  
ELODV1 0336  
ELODV2 0415  
ELODV3 0440  
ELODV4 0463  
ENCPTR 0022  
ENCVAL 3306  
ENDHI 0733  
ENDLO 0732  
ENDQUE 6416  
ENDWD 0077  
ENDWD1 7700  
ENQUE 1065  
EOLXIT 4517  
ERFREQ 4327  
ERRDEV 4335  
ERRENC 4303  
ERRFUL 4317  
ERRXIT 4522  
ERROPT 4311  
EXIT 0472  
EXTMTP 0010  
F 0514  
FIRLDF 0643  
FIRST 6001  
FOUND 2432  
FREE 4332

FREEZE	2040
FREQ	0735
FRMS	0542
FROM	1510
FRSTR	3200
FUL	0006
FUL1	0600
FWDCDF	2303
FWDRTN	2300
G	1315
GBLK	3430
GETKBD	5534
GETU	1542
GO	1411
GODIS	2251
GOLOP	1451
GOMS	3507
GOODY	1124
GRDPTR	0014
GRDTAB	5331
GSWIT	1365
H	1311
HALT	1400
HALTAL	3516
HBIT	4000
HCPTR	0015
HCTAB	3557
HIDV0	0311
HIDV1	0333
HIDV2	0412
HIDV3	0435
HIDV4	0460
HLPTAB	3615
HLTALL	1363
HLTDV0	0317
HLTDV1	0341
HLTDV2	0420
HLTDV3	0443
HLTDV4	0466
HLTLOP	3475
HLTMS	3461
HLTPTR	1516
HOLD	0736
HSWIT	1364
HSWIT0	0073
HSWIT1	0074
HSWIT2	0075
HSWIT3	0076
HSWIT4	0077
HSWPTR	3130
IDLE	2020
INIAD	6145
INICHR	2137
INIEND	6036
INILEN	3273
INIPTR	0010
INISAM	1464
INISTR	6016
INMESS	1044
INPUT	0600
INTL	0200
INTMOD	0345
INTOUT	0256

-

INTP 0230  
INVMS 0544  
KADR 3303  
KBCD 2526  
KBDBUF 0021  
KBDTAB 1225  
KBLKCT 0107  
KCLEN 0062  
KCONMS 4162  
KCUR 2352  
KD 1271  
KDADD 0037  
KDCIA 4171  
KDIG 0063  
KDOMTP 1763  
KDSC4 4166  
KDV2 0353  
KENC 3277  
KERENC 4165  
KERFUL 4172  
KEROPT 4170  
KEYBRD 1200  
KEYRTN 0511  
KEYXIT 1505  
KF 1270  
KFIRST 1740  
KFREQ 0537  
KFREQ 0734  
KFRESH 1745  
KFRSTR 4163  
KGBLK 4302  
KGO 1370  
KGOMS 0102  
KHALT 1366  
KHLTMS 0101  
KIDLE 0065  
KINI 0106  
KINPUT 0030  
KKDCIA 3427  
KKEY 2047  
KLAST 0110  
KLIM 2525  
KLOAD 1525  
KLOHI 4173  
KMES 2050  
KMESS2 4300  
KMUL10 0064  
KNUM 3526  
KXSTR 4277  
KP 1267  
KPOLAR 0540  
KP1DAC 4174  
KQUPUT 0105  
KR1J 2527  
KRDERI 1737  
KRESTR 4164  
KS 1272  
KSET 1272  
KSETUP 1545  
KSFDV0 0310  
KSFDV1 0332  
KSFDV2 0411  
KSFDV3 0434

KSFDV4 0457  
KSTRQ 3367  
KSWI 2051  
KTTY 3365  
KTX26 3302  
KUPACK 4167  
KUPLIM 1661  
KURLEN 7750  
KW 1273  
KWRERR 1764  
KWRIT 0546  
KXYGET 2353  
K210 3366  
K6K 7056  
L 1354  
LAST 6144  
LDFSTR 1172  
LIMLOP 2407  
LIMPTR 0055  
LIMVL0 6443  
LIMVL1 6447  
LIMVL2 6453  
LIMVL3 6457  
LIMVL4 6463  
LIMVL5 6467  
LNNXT 2061  
LOAD 4454  
LOADER 1550  
LODLOP 1532  
LODPTR 1610  
LODV0 0312  
LODV1 0334  
LODV2 0413  
LODV3 0436  
LODV4 0461  
LOHI 3406  
LOOP 7010  
LOP1 3326  
LOP2 3330  
LSAVE 0351  
LSENC 3300  
LSINI 3276  
LSSAHI 3275  
LSSALO 3274  
M 1347  
MAG 1522  
MAGPTR 1603  
MATCH 1220  
MAXCDF 0032  
MAXCUR 2752  
MAXDRP 0724  
MAXDVN 0061  
MEMCHK 7001  
MESCHK 1000  
MESS1 4026  
MESS2 4043  
MESS3 4237  
MINADR 0034  
MINCDF 0033  
MM12 1657  
MM4 2756  
MODE 0103  
MOTMS 0541  
-

MSAD	6712
MSBUSY	6676
MSFR	6624
MSG0	6550
MSHLT	6527
MSINV	6646
MSMOT	6613
MSNOR	6635
MSPTR	0017
MS0	6567
MS1	6573
MS2	6577
MS3	6603
MS4	6607
MTPLOP	1100
MUL10	3000
M10	1660
M1000	2350
M12	2753
M5	1514
NEG12	4301
NEG5	2530
NEWSAM	0725
NFDVK	0673
NLDF	7013
NORMS	0543
NOT0	4122
NUM	1321
NUMPUT	3517
NUM0	6522
NUM1	6523
NUM2	6524
NUM3	6525
NUM4	6526
NXSTR	3255
NXTCHR	2152
NXTLN	2131
NXTPNT	2261
OK1	4266
OK2	4207
OLDSAM	0727
OPKRTN	1203
OSDV0	0313
OSDV1	0335
OSDV2	0414
OSDV3	0437
OSDV4	0462
OUTL	0266
P	0500
PCKLOP	3061
PL1000	1515
PL400	3460
PL60	4513
POLAR	2266
PP17	1367
PP400	2755
P10AC	3400
P1000	0726
P17	3106
P1777	0355
P2	2743
P37	0350
P4	7000

P401	2166
P4040	7055
P4060	3301
P57	2757
P60	2754
P6000	0354
P7K	1113
QAB	5017
QACA	5030
QACHAR	5670
QACKLF	5634
QACNTR	5617
QAD	5041
QADUN	3312
QAE	5063
QAEOL1	7400
QAEOM1	3400
QAEXIT	5650
QAF	5531
QAFLD	4000
QAG	5075
QAH	5127
QAI	5144
QAINIT	5013
QAJ	5151
QAK	5320
QAKRB	6036
QAL	5210
QALEGL	5610
QAM	5114
QAN	5236
QAO	5244
QAP	5255
QAQ	5276
QARFSH	5066
QAT	5303
QATLS	6046
QATPE	5657
QATSF	6041
QATY	5551
QAU	5521
QAV	5331
QAW	5525
QAX	5437
QAY	5425
QAZ	5314
QSAVE	1066
QUECNT	0025
QUEIN	0023
QUEMAX	1062
QUEOUT	0024
QUPUT	1033
RADPTR	0057
RADVAL	6417
RDCWD	1677
ROLEN	1742
ROMBLK	6000
READ	4344
REMS2	4046
RESTR	3233
RMBLK	1744
RTNJMP	0274
RTNLDF	0271

RTNLIF 0270  
S 0526  
SAMX 0703  
SAMY 0654  
SCPFLO 2000  
SCPRTN 2043  
SCPXIT 2163  
SELDV0 0066  
SELDV1 0067  
SELDV2 0070  
SELDV3 0071  
SELDV4 0072  
SELPTR 0100  
SERVE 0234  
SETANA 4073  
SETCDF 2320  
SETFUL 2114  
SETHC 2111  
SETMOT 2200  
SETUP 4023  
SFDV0 0307  
SFDV1 0331  
SFDV2 0410  
SFDV3 0433  
SFDV4 0456  
START 4020  
STATUS 0026  
STQUE 1064  
STRQUE 6373  
STRRDE 3315  
TBLK 1662  
TBUSY 1057  
TEMP 0104  
TEMPHI 0051  
TEMPL0 0050  
TIADR 1512  
TIAD0 6323  
TIAD1 6331  
TIAD2 6337  
TIAD3 6345  
TIAD4 6353  
TIRTN 0240  
TISA 0275  
TISTR0 6272  
TISTR1 6277  
TISTR2 6304  
TISTR3 6311  
TISTR4 6316  
TIVAL 1507  
TO 1511  
TPARGS 1615  
TPINST 1102  
TPLDF 1101  
TPLOP 1622  
TPRDC 1676  
TPWAIT 1163  
TPWORD 1103  
TTYFLG 0022  
TXT1 4535  
TXT10 4757  
TXT11 4765  
TXT12 4774  
TXT13 5001  
-

TXT14	5006
TXT2	4565
TXT3	4650
TXT4	4672
TXT5	4705
TXT6	4720
TXT7	4747
UNIT	1663
UPACK	3055
UPLIM	4506
VALTAB	6361
VCOORD	2737
VCPTR	0016
VCTAB	3540
W	0533
WMBLK	1743
WRIT	4400
WRMBLK	4000
XAXDV0	0010
XAXDV1	0011
XAXDV2	0012
XAXDV3	0013
XAXDV4	0020
XCORD	2017
XCURHI	0047
XCURLO	0046
XYGET	2400
XYHC	0000
YAXDV0	0014
YAXDV1	0015
YAXDV2	0016
YAXDV3	0017
YAXDV4	0024
YCUR	0035
YLOP	2647
ZXIT	0512

READER'S COMMENTS

TISA  
DEC-12-UW3A-D

Digital Equipment Corporation maintains a continuous effort to improve the quality and usefulness of its publications. To do this effectively we need user feedback – your critical evaluation of this manual.

Please comment on this manual's completeness, accuracy, organization, usability, and readability.

---

---

---

---

---

Did you find errors in this manual? \_\_\_\_\_

---

---

---

---

How can this manual be improved? \_\_\_\_\_

---

---

---

---

---

DEC also strives to keep its customers informed of current DEC software and publications. Thus, the following periodically distributed publications are available upon request. Please check the appropriate boxes for a current issue of the publication(s) desired.

- Software Manual Update, a quarterly collection of revisions to current software manuals.
- User's Bookshelf, a bibliography of current software manuals.
- Program Library Price List, a list of currently available software programs and manuals.

Please describe your position. \_\_\_\_\_

Name \_\_\_\_\_ Organization \_\_\_\_\_

Street \_\_\_\_\_ Department \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip or Country \_\_\_\_\_

----- **Fold Here** -----

----- **Do Not Tear - Fold Here and Staple** -----

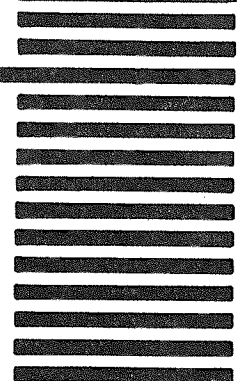
FIRST CLASS  
PERMIT NO. 33  
MAYNARD, MASS.

BUSINESS REPLY MAIL  
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

**digital**

Digital Equipment Corporation  
Software Information Services  
146 Main Street, Bldg. 3-5  
Maynard, Massachusetts 01754



## HOW TO OBTAIN SOFTWARE INFORMATION

Announcements for new and revised software, as well as programming notes, software problems, and documentation corrections are published by Software Information Service in the following newsletters.

Digital Software News for the PDP-8 Family  
Digital Software News for the PDP-9/15 Family  
PDP-6/PDP 10 Software Bulletin

These newsletters contain information applicable to software available from Digital's Program Library

Please complete the card below to place your name on the newsletter mailing list.

Questions or problems concerning DEC Software should be reported to the Software Specialist at your nearest DEC regional or district sales office. In cases where no Software Specialist is available, please send a Software Trouble Report form with details of the problem to

Software Information Service  
Digital Equipment Corporation  
146 Main Street, Bldg. 3-5  
Maynard, Massachusetts 01754

These forms, which are available without charge from the Program Library, should be fully filled out and accompanied by teletype output as well as listings or tapes of the user program to facilitate a complete investigation. An answer will be sent to the individual and appropriate topics of general interest will be printed in the newsletter.

New and revised software and manuals, Software Trouble Report forms, and cumulative Software Manual Updates are available from the Program Library. When ordering include the document number and a brief description of the program or manual requested. Revisions of programs and documents will be announced in the newsletters and a price list will be included twice yearly. Direct all inquiries and requests to:

Program Library  
Digital Equipment Corporation  
146 Main Street, Bldg. 3-5  
Maynard, Massachusetts 01754

Digital Equipment Computer Users Society (DECUS) maintains a user Library and publishes a catalog of programs as well as the DECUSCOPE magazine for its members and non-members who request it. For further information please write to

DECUS  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

---

Send Digital's software newsletters to:

Name \_\_\_\_\_  
Company Name \_\_\_\_\_  
Address \_\_\_\_\_  
\_\_\_\_\_

My computer is a

PDP-8/I <input type="checkbox"/>	PDP-8/L <input type="checkbox"/>	(zip code)
LINC-8 <input type="checkbox"/>	PDP-12 <input type="checkbox"/>	
PDP-9 <input type="checkbox"/>	PDP-15 <input type="checkbox"/>	Please specify
PDP-10 <input type="checkbox"/>	OTHER <input type="checkbox"/>	_____

My system serial number is \_\_\_\_\_ (if known)

Fold Here

Do Not Tear - Fold Here and Staple

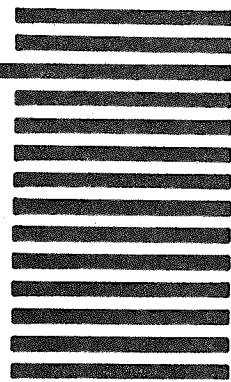
FIRST CLASS  
PERMIT NO. 33  
MAYNARD, MASS.

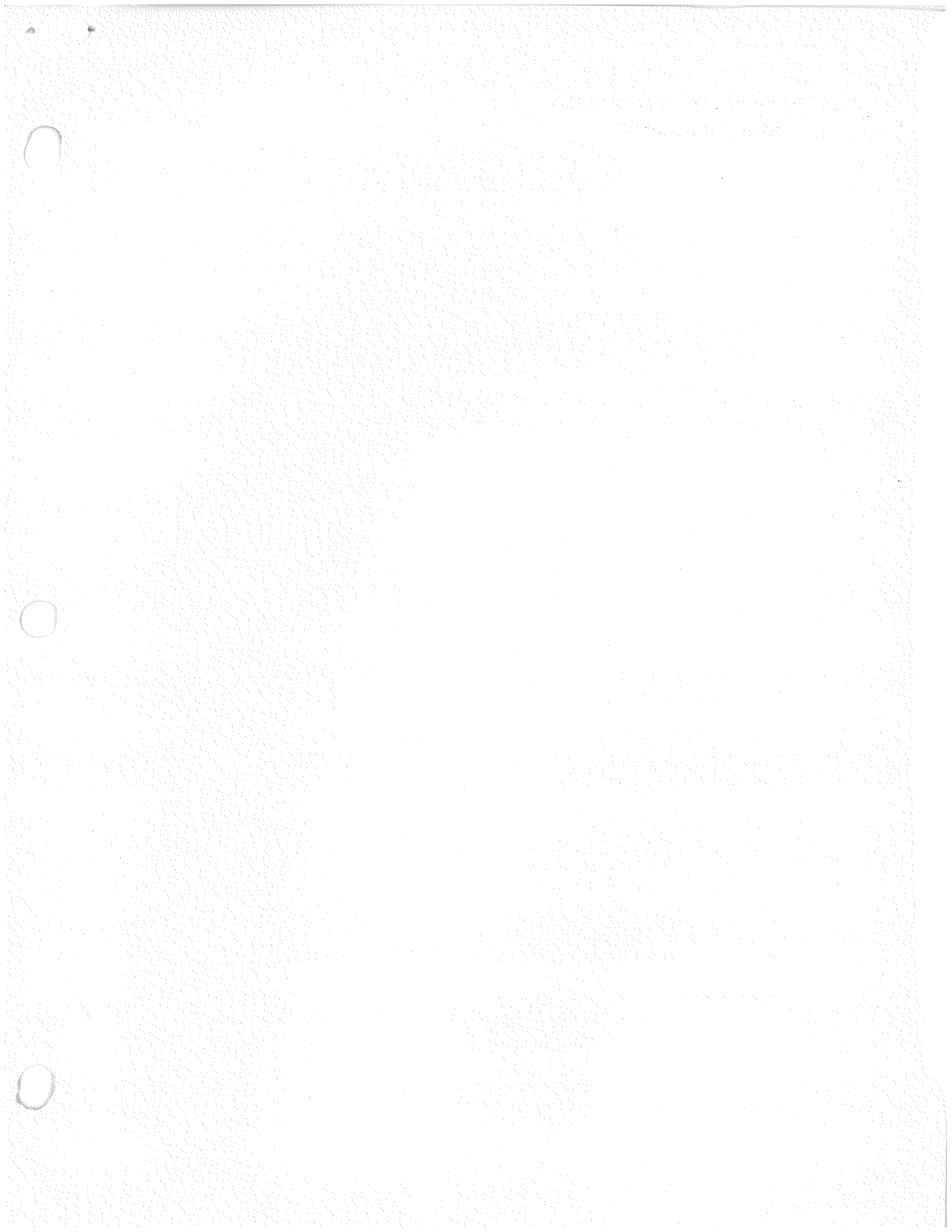
BUSINESS REPLY MAIL  
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

**digital**

Digital Equipment Corporation  
Software Information Services  
146 Main Street, Bldg. 3-5  
Maynard, Massachusetts 01754





**Digital Equipment Corporation  
Maynard, Massachusetts**

**digital**