

SDS 901076A
\$2.50

DIAGNOSTIC PROGRAM MANUAL
SIGMA 5 AND 7
SYSTEMS TEST MONITOR

February 1967

SCIENTIFIC DATA SYSTEMS • 1649 Seventeenth Street • Santa Monica, Calif. • (213) 871-0960

©1967, Scientific Data Systems, Inc.

LIST OF EFFECTIVE PAGES

Total number of effective pages is 54, as follows:

Page No.	Issue	Page No.	Issue
Title	Original		
A	Original		
i thru iv	Original		
1-1 thru 1-4	Original		
2-1 thru 2-8	Original		
3-1 thru 3-12	Original		
4-1 thru 4-24	Original		

TABLE OF CONTENTS

Section	Title	Page
I	INTRODUCTION	1-1
1-1	Scope of Manual	1-1
1-4	Program Objectives	1-1
1-7	General Specifications	1-1
II	OPERATING INSTRUCTIONS	2-1
2-1	Program Loading Procedure	2-1
2-4	Program Operating Procedure	2-1
2-6	Loading The Device Programs	2-1
2-11	System Control	2-2
2-14	Parameter Input	2-3
2-15	Assignment of Parameters	2-3
2-16	Standard Parameters	2-3
2-17	Changing of Parameters	2-3
2-18	Run List	2-3
2-19	Constructing the Run List	2-3
2-20	Additions to or Deletions from the Run List	2-4
2-21	Operator Control of Message Printouts	2-4
2-22	Error Messages	2-4
2-23	Profile Printouts	2-4
2-24	Execution of Device Programs	2-5
2-26	Program Initializing Procedure	2-5
2-28	Program Delays	2-5
2-31	Program Halts and Loops	2-6
2-33	Error and Profile Printouts	2-7
2-34	Error Messages	2-7
2-35	Operator-Requested Profile Printouts	2-8
2-37	Profile Printout Format	2-8
III	PROGRAM DESCRIPTION	3-1
3-1	Operation of the Systems Test Monitor Program, General	3-1
3-2	Tables of the Monitor Program	3-1
3-4	A Typical Operation Sequence	3-1
3-6	Operation's Start	3-1
3-11	Operation's Run	3-2
3-16	Operation's Conclusion	3-3

TABLE OF CONTENTS (Cont.)

Section	Title	Page
3-18	The System Evaluation (SEVA) Interface Used With All Device Programs	3-3
3-19	Prerequisites for All Device Programs	3-3
3-20	Origin	3-3
3-22	Instruction Format for All Programs	3-3
3-24	Explanation	3-3
3-25	Example	3-4
3-28	Mode	3-4
3-30	Monitor Subroutines, General	3-6
3-32	Subroutine Descriptions	3-6
3-33	Error Reporting	3-6
3-35	Start I/O	3-6
3-37	The HIO, TIO, and TDV Functions	3-6
3-39	AIO	3-6
3-41	Delay Routine	3-6
3-43	Profile Routine	3-7
3-45	Monitor Subroutine Call Sequences	3-7
3-46	Start I/O Subroutine	3-7
3-47	Call Sequence	3-7
3-48	Description of SIO Call Sequence	3-7
3-50	The HIO, TIO, and TDV Routines	3-8
3-53	Error Routine	3-8
3-54	Calling Sequence	3-8
3-55	Description	3-9
3-57	Equate Table	3-9
3-59	Delay Routine	3-9
3-62	Profile Routine	3-11
3-63	Calling Sequence	3-11
3-64	Description	3-11
IV	PROGRAM LISTING	4-1
4-1	General	4-1

LIST OF ILLUSTRATIONS

Figure	Title	Page
1-1	Flow Chart of the Sigma 5/7 Systems Test Monitor	1-2
3-1	Sample Program Listing of a Device Program	3-5
3-2	Sample Listing of an Equate Table	3-10

LIST OF TABLES

Table	Title	Page
1-1	General Specifications	1-3
2-1	SENSE Switch Options (Device Program Loading)	2-3
2-2	Possible Input Messages (Device Program Loading)	2-4
2-3	SENSE Switch Options (After Initial Device Program Loading)	2-6
2-4	Program Halts and Loops (After Initial Device Program Loading)	2-6
2-5	Error Message Printouts	2-7
3-1	Call Sequence of the Start I/O Routine.	3-7
3-2	Calling Sequences for HIO, TIO, and TDV	3-8
3-3	Calling Sequence for Error Routine	3-9
3-4	Calling Sequence of Delay Routine	3-11

LIST OF RELATED PUBLICATIONS

<u>Publication Title</u>	<u>Publication No.</u>
SDS Sigma 7 Computer Reference Manual	900950
SDS Sigma 5 Computer Reference Manual	900959
SDS Sigma Computer Systems Interface Design Manual	900973
SDS Sigma Keyboard/Printer Reference Manual	900974
Symbol and Metasymbol Reference Manual	900952
Sigma 5 and 7 Relocatable Diagnostic Program Loader Diagnostic Program Manual	900972

SECTION I
INTRODUCTION

1-1 SCOPE OF MANUAL

1-2 This document describes the functions of the Sigma 5/7 Systems Test Monitor. The manual includes the program loading procedure, the monitor operating procedures, a detailed description of the method employed for operator communication and control of the monitor, and program interface requirements which must be followed so that the device test program may be controlled by the systems test monitor. The manual concludes with a complete program listing of the systems test monitor program.

1-3 Figure 1-1 depicts a flow diagram of the program as a whole.

1-4 PROGRAM OBJECTIVES

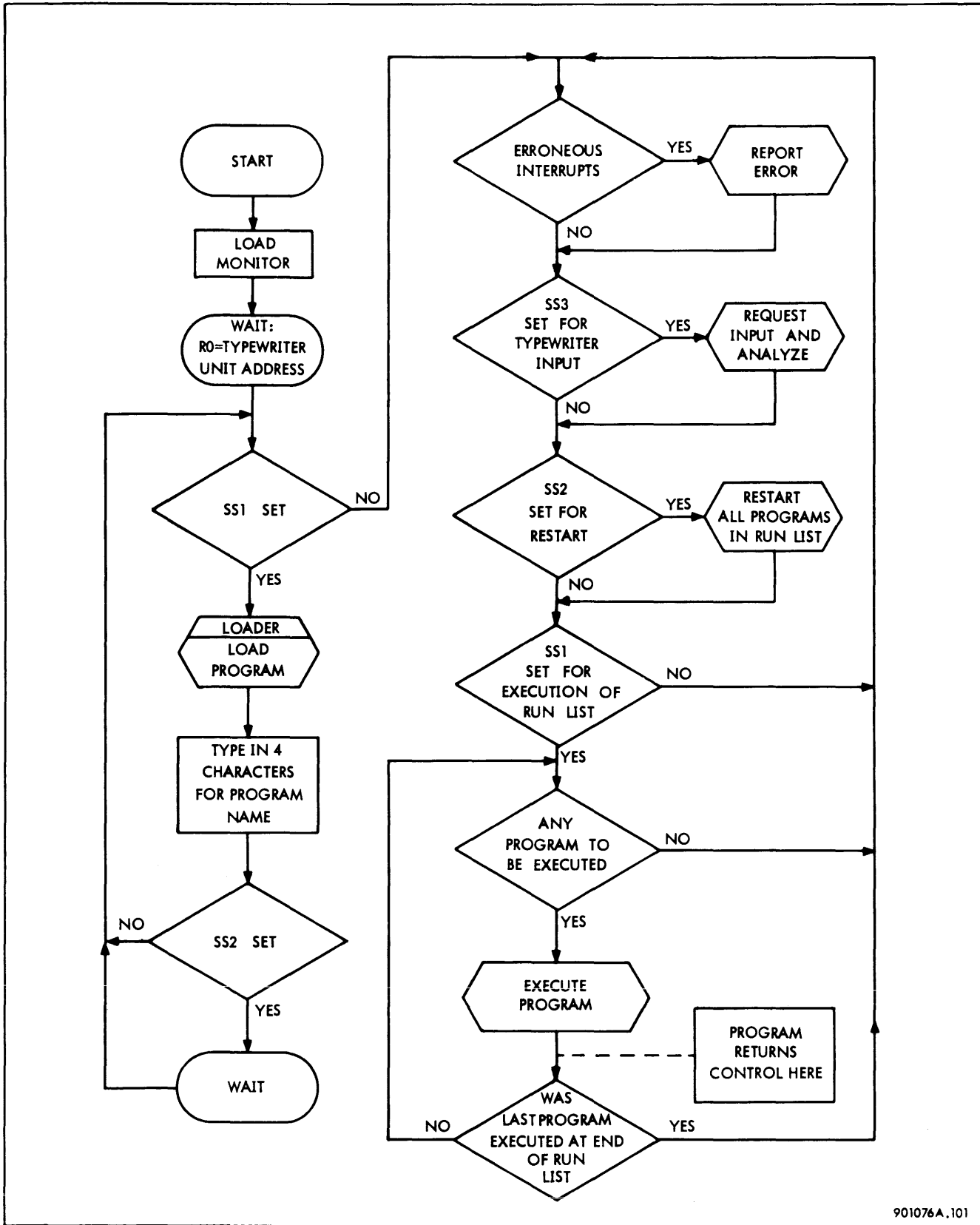
1-5 The system test is an offline program written specifically to evaluate the operation of a computer system as a unit. It emphasizes operation of the total processing system, including all devices, under varying sequences and with as heavy a load as possible. No attempt is made to isolate an error except for that which is already provided by the machine detection circuitry; however, the environment of the machine at the time of the error is recorded for subsequent use in error isolation.

1-6 To accomplish this objective, the following principles have been made part of this systems evaluation program:

- a. Overlapping of input-output and central processor operation to approach a maximum throughput
- b. Use of most machine instructions in the program operation
- c. Use of random data patterns for all data transfer operations
- d. Provision of methods for detecting errors beyond the capability of the machine.

1-7 GENERAL SPECIFICATIONS

1-8 The general specifications for this program are given in table 1-1.



901076A.101

Figure 1-1. Flow Chart of the Sigma 5/7 Systems Test Monitor

Table 1-1. General Specifications

COMPUTER CONFIGURATION	Sigma 5 or Sigma 7 computer with 4K of memory
REQUIRED EQUIPMENT	Card reader or paper tape reader as the program media input device, keyboard/printer as the operator-communication device
OPTIONAL EQUIPMENT	Card reader, card punch, paper tape reader, paper tape punch, magnetic tape (9 channel), line printer, and medium-speed disc (RAD)
PREREQUISITES	Sigma 5 or Sigma 7 computer, memory, I/O interface, keyboard/printer and program media input device (card reader or paper tape reader), all of which must be operational. Peripheral devices must conform to their respective design specifications
STORAGE	Resides in low-order memory (absolute origin 4016) and occupies approximately 1200 decimal locations
SOURCE LANGUAGE	Sigma Metasymbol (see Symbol and Metasymbol Reference Manual, No. 900952)
PROGRAM MEDIA	80-column punched cards and 8-level paper tape with SDS Sigma 5/7 Relocatable Diagnostic Program Loader, Model No. 704356

SECTION II
OPERATING INSTRUCTIONS

2-1 PROGRAM LOADING PROCEDURE

2-2 The Sigma 5/7 systems test monitor object program media are loaded, using the SDS Sigma 5/7 Relocatable Diagnostic Program Loader, Model No. 704356. The loader is supplied as a physical part of the object program media. Refer to the Relocatable Diagnostic Program Loader Manual, No. 901128, and follow the detailed procedures given there to load the systems test monitor into memory.

2-3 A successful load operation is indicated by a wait at location X'00173'.

2-4 PROGRAM OPERATING PROCEDURE

2-5 The following paragraphs (2-6 through 2-23) describe in detail the procedures for loading the systems test monitor and device programs, the means of controlling the systems test, and the options available at run time.

2-6 LOADING THE DEVICE PROGRAMS

2-7 The systems test monitor is placed behind the loader. The device programs have to be placed behind the systems test monitor; these device programs can be in any sequence. The number of device programs which can be loaded is limited to ten. The following listing shows the sequence of operations to follow after the wait at X'00173', which had occurred after the loading of the monitor:

- a. Enter the unit address of the keyboard/printer into register 0 (R0).
- b. Reset SS2, SS3, and SS4.
- c. Set SS1 as indicated in table 2-1.
- d. Place the first device program in paper tape or card reader.
- e. Clear wait.

The device program is now loaded.

2-8 The SENSE switch options for the device loading procedure are given in table 2-1.

Table 2-1. SENSE Switch Options (Device Program Loading)

Switch	Status	Function
SS1	1 (Set)	Monitor reads in a device program
	0 (Reset)	Monitor discontinues loading device program
SS2	1 (Set)	(Not used)
	0 (Reset)	Correct position of switch at load time
SS3	1 (Set)	(Not used)
	0 (Reset)	Correct position of switch at load time
SS4	1 (Set)	(Not used)
	0 (Reset)	Correct position of switch at load time

2-9 When the keyboard is addressed for input:

- a. Reset SS1 if the program just loaded is the last one.
- b. If the program just loaded is not the last program, leave SS1 set. Place the next device program in paper tape or card reader.
- c. Type four alphabetic characters, which will be the device program name used throughout the systems test operation. This name must be different for each device program. If the name is three or less characters, type spaces to complete the four-character requirement.

2-10 After the fourth character is typed, the monitor will proceed as follows:

- a. If SS1 is set, it will read the next device program. The operator can address the keyboard again (paragraph 2-9) for further instructions.
- b. If SS1 is reset, it will discontinue loading programs and start idling.

2-11 SYSTEM CONTROL

2-12 The following paragraphs (2-13 through 2-23) describe how to control the systems test in order to accomplish the specific test objective desired by the operator.

2-13 Most communications between the operator and the systems test is via the keyboard. In order for the operator to gain control of the typewriter, SS3 must be set. Termination of keyboard input is accomplished by typing the EOM (end of message) character.

2-14 Parameter Input

2-15 Assignment of Parameters. The first input via the keyboard must be to assign parameters for the device program to be run. To accomplish this, the following information should be typed:

P MMMM, XXX, ABC, D2+NNNN, XXX, 4 EOM

- a. P stands for "Assign Parameters"; it must be the first character.
- b. A blank space must be used between the Assign Parameter (P) character and those which follow. A blank space can be used between any of the following characters.
- c. MMMM identifies a device program name to which the following parameters are to be assigned.
- d. XXX is a three-character hexadecimal parameter which specifies the unit address that the named device is to use.
- e. , ABC, D2 and , 4 are examples of information that follow the unit address. This information is special for each program and will be defined in the specifications for that particular test.
- f. + indicates that a new test will be named with new parameters.
- g. EOM terminates the keyboard input.

2-16 Standard Parameters. Some device programs have a standard set of parameters which can be requested by typing an asterisk (*) after the unit address. For example,

P NNNN , UA , * EOM

will cause the program named NNNN to assign standard parameters peculiar to that device program. The description of the standard parameters can be found in the particular documentation for the various device programs.

2-17 Changing of Parameters. Once parameters have been assigned to device programs, new parameters can be entered at any time.

2-18 Run List

2-19 Constructing the Run List. After parameters have been assigned, the unit tests can be executed. However, since P (Assign Parameters) does not cause any program to be executed, additional typewriter input will be required by the monitor to enable it to run the program.

If, for example,

R MMMM + NNNN + PPPP EOM

is typed, this command will tell the monitor to run programs MMMM, NNNN, and PPPP. These program names will be entered into the run list after the run list is cleared.

2-20 Additions to or Deletions from the Run List. Once device programs are running, others may be added to or deleted from the list, or a new list may be specified. If programs are to be added to the list, the first character typed should be A. (For example, if the statement A MMMM + NNNN EOM is typed, the monitor will add programs MMMM and NNNN to the run list.) If programs are to be deleted, the first character typed should be a D. (Statement D MMMM + NNNN EOM will delete programs MMMM and NNNN from the run list.) The first character of the statement is followed by the test names to be added or deleted.

2-21 Operator Control of Message Printouts

2-22 Error Messages. When the monitor or the device programs encounter system or operating errors, error messages will be typed out. These error messages can be suppressed for all programs or for any particular program. See table 2-2 for possible error messages.

2-23 Profile Printouts. The profile printout after the error messages can be suppressed or initiated by the operator. Table 2-2 also contains a list of input messages to control profile printouts.

Table 2-2. Possible Input Messages (Device Program Loading)

Message	Explanation
PROF EOM	Type out the profile whenever an error, made by any program (including the monitor), is reported
NPRO EOM	Do not type the profile when an error occurs
SERR EOM	Suppress all error typeouts
PERR EOM	Reinstate all error typeouts (previously suppressed)
S NAM1 + NAM2 + NAM3 EOM	Suppress all error typeouts from programs NAM1, NAM2, and NAM3. (The monitor error typeouts cannot be suppressed with this directive.)
T NAM1 + NAM2 + NAM3 EOM	Reinstate error typeouts of programs NAM1, NAM2, and NAM3.

2-24 EXECUTION OF DEVICE PROGRAMS

2-25 Before any program can be executed, two prerequisites must be met:

- a. The program must be in the run list.
- b. SS1 must be set. (If SS1 is reset at any time, no device program will be executed until SS1 is set again.)

2-26 Program Initializing Procedure

2-27 Because programs get lost at times, due to malfunctions in the equipment, it may be desirable to initialize the program to its original state after loading. To accomplish this, proceed as follows:

- a. Set COMPUTE switch to IDLE.
- b. Press SYST CLEAR pushbutton on console.
- c. Set SS2 (=1); reset SS1, SS3, and SS4 (all = 0).
- d. Set COMPUTE switch to RUN.

The monitor will now be in an idle condition and formal operations can be resumed. (See table 2-3.)

2-28 Program Delays

2-29 In order to create delays between the starting of the devices, the console INTERRUPT and SS4 are used to establish a delay constant to be used with count pulse 3. For incrementing the delay constant, set SS4 to 1 and press the console INTERRUPT once for each increment to the count pulse constant desired. For decrementing the delay constant, reset SS4 to 0; press the console INTERRUPT for each decrement to the count pulse constant desired. When the count pulse constant reaches zero during decrementing, no delay is started. The console INTERRUPT is then ineffective when pressed.

2-30 A summary of the SENSE switch options after initial loading as described in paragraphs 2-27 through 2-29 is given in table 2-3.

Table 2-3. SENSE Switch Options (After Initial Device Program Loading)

Switch	Status	Function
SS1	1 (Set)	Causes the monitor to execute the test selected by the R or A directives
	0 (Reset)	Causes the monitor to loop in the program load section (that is, the monitor is in an idle condition)
SS2	1 (Set)	Restarts test program
	0 (Reset)	Will not restart test program
SS3	1 (Set)	Permits typewriter input
	0 (Reset)	Will not allow typewriter input
SS4*	1 (Set)	Increments the delay constant
	0 (Reset)	Decrements the delay constant

*This SENSE switch is used in conjunction with the console INTERRUPT.

2-31 Program Halts and Loops

2-32 Table 2-4 lists the program halts and loops that may occur in the monitor during its operation. The recovery procedure for occurrence of any loop is specified in the restart procedure.

Table 2-4. Program Halts and Loops (After Initial Device Program Loading)

Item No.	Location	Indication	Description and Recovery Procedure
1	X'00114'	Loop to itself (BRU\$)	An invalid OP code trap occurred. R0 shows the address where the trap occurred
2	X'0011D'	Loop to itself (BRU\$)	An unimplemented instruction trap occurred. R0 contains the address where trap occurred
3	X'00125'	Loop to itself (BRU\$)	A watchdog timer trap occurred. R0 shows the address where the trap occurred
4	X'0012A'	Loop to itself (BRU\$)	A power-on trap occurred
5	X'0012B'	Wait	A power-off trap occurred
6	X10012C'	Loop to itself (BRU\$)	The program will end up in this instruction if the wait instruction after the power-off trap has been cleared

Table 2-4. Program Halts and Loops (After Initial Device Program Loading) (Cont.)

Item No.	Location	Indication	Description and Recovery Procedure
7	X'0012F'	Loop to itself (BRU\$)	A memory parity trap occurred. R0 contains the address where the trap occurred
8	X'00173'	Wait	Takes place after the loading of the monitor. The keyboard address is placed in R0. After the appropriate SENSE switches are set, clear the wait to continue
9	X'0019E'	Wait	Takes place during the device program load time if SS2 is set. Clear halt to continue
10	X'00201'	Wait	Takes place during restart procedure. Press I/O RESET pushbutton on console, reset SS2, clear halt, and continue
11	X'004DC'	Loop to itself (BRU\$)	Error timeout not possible because keyboard did not accept SIO command. Use restart procedure

2-33 Error and Profile Printouts

2-34 Error Messages. Table 2-5 shows the error printouts initiated by the monitor.

Table 2-5. Error Message Printouts

Error Message	Description and Recovery Procedure
* ERROR MON 00000001	Program called for is not present in memory. Correct name and type in again. Reload programs if input line is correct and error timeouts persist
* ERROR MON 00000002	Incorrect command given in input line. Correct input line and type in again. Reload programs if input line is correct and error timeouts persist
* ERROR MON 00000003 A0000000 BBBBBBBB	The monitor received an I/O interrupt which it did not expect. The second word contains the condition codes in bits 0 through 4 (as shown by the letter A). Word 3 (B) contains the AIO response. There is no recovery necessary
* ERROR MON 00000004	Too many parameters were entered in the input line for one of the programs. Correct input line and type in again. Reload program if input line is correct and error timeouts persist

2-35 Operator-Requested Profile Printouts. Before a device is started, the device program reports to a monitor routine, which records the program name and its identification (ID) in a profile table. When an error occurs, this profile is typed out, if so requested.

2-36 If the operator desires a profile printout, he sets SS3 to enable him to input a request; and types the following message:

PROF EOM

This will result in a profile typeout whenever an error by any program (including the monitor) is reported. If no profile is desired, the operator types: NPRO EOM

2-37 Profile Printout Format. An example of a profile table that is printed after every error message (if requested by the operator) is as follows:

```
PROFILE MT1 = RD** LP1 = CH** MT1 = RD** LP1 = PR** MT1 = RD** MT1 = RD**
      MT1 = RD** LP1 = CH** MT1 = RD** MT1 = RD** MT1 = RD** MT1 = RD**
      LP1 = AF** MT1 = RD** LP1 = IS** MT1 = RD** MT1 = RD** MT1 = RD**
```

2-38 The program name (MT1 and LP1) precedes the ID of the function that was last marked in that program. Eighteen (18) such functions of various programs are recorded. The proper identification for the various programs can be found in the documentation of these programs.

SECTION III
PROGRAM DESCRIPTION

3-1 OPERATION OF THE SYSTEMS TEST MONITOR PROGRAM, GENERAL

3-2 TABLES OF THE MONITOR PROGRAM

3-3 The systems test monitor constructs two tables which make up the basic skeleton of the test. One table, the address table, contains the beginning address of each device program loaded. The other table, the run table, contains the names of the programs to be run at any time. It is loaded or cleared upon command by the operator. The contents of these two tables govern the manner in which the systems test monitor performs.

3-4 A TYPICAL OPERATION SEQUENCE

3-5 Presume that two tests: one for the printer, called PR1, and the other for the card reader, called CR1, are loaded after the monitor itself was loaded. The monitor run table will be empty at this time, and the monitor idles.

3-6 Operation's Start

3-7 To start operation, set SS3. The input light on the typewriter will go on. Reset SS3. Then type in the following:

```
P CR1 ,XXX, *+PR1,XXX* EOM
```

This statement will assign standard parameters to the tests CR1 and PR1.

3-8 Once the monitor senses a program switch set, indicating that the interrupt at channel end has been put through, it analyzes the input data. Since P is the first character, it proceeds to assign parameters if possible. After identifying the program name (CR1), the monitor attempts to find the program by means of the address table. If it cannot find the program, an error message will be typed. However, since the CR1 program has been loaded, the monitor stores the parameters in the card reader program. The parameters for PR1 are stored in the printer program. Once the monitor has finished storing the parameters, it interrogates the SENSE switches again. If SS3 is on, it asks for more input.

3-9 SS1 is then interrogated by the monitor to see whether any programs have to be run (SS1 could be called a circuit breaker.) As long as SS1 is reset, the monitor does not look at its run table and no programs run, even though some programs were selected through the typewriter. If SS1 is set, the monitor looks at the run table. Since it is empty, no programs are run.

3-10 SS3 must then be set to transfer control to the typewriter for input. If the operator types in the line

R PR1 + CR1 EOM

programs PR1 and CR1 will be put in the run list. If SS1 remains set (=1), the monitor looks at the run table again, where it will now find PR1 and CR1. The monitor then interrogates another table in PR1, called the program status table, to find out whether there is an address in this table or not. Since this is the first time the monitor entered this program, there will be an address in the program status table. The monitor branches to this address and program PR1 starts execution.

3-11 Operation's Run

3-12 The program prints one line. After the device program has started this line, it removes the subroutine address from its program status table, and control returns to the monitor. The monitor then looks at the next entry in the run table, CR1, and enters the CR1 program. After a card read operation has been initiated, this device program also transfers control back to the monitor.

3-13 The monitor continuously looks at the run table (providing SS1 is set) to see whether there are any addresses in the program status tables of the programs selected. Since the two programs have started their routines, their tables will be empty. As soon as one of the programs receives an interrupt, indicating that the operation started on the device has been completed, it puts an address back in its program status table. When the monitor sees this address, it transfers control back to the device program. The program can then initiate another I/O operation; subsequently it transfers control back to the monitor again.

3-14 The execution of device programs are thus done at random, since the various devices take varying amounts of time to complete their operations. Thus it is possible, for example, for two or more cards to be read before another line is printed.

3-15 If, after the programs are running for a while, the operator notices that one of the devices has stopped operating, it may be advisable for him to clear the I/O and initialize the program again. To accomplish this, he uses the restart procedure, causing the program status tables in all the programs to be loaded again with the programs' beginning addresses.

3-16 Operation's Conclusion

3-17 After the restart procedure, the monitor starts sampling the run table again, since the run table was not cleared. The PR1 and CR1 programs start printing and reading cards, respectively, again.

3-18 THE SYSTEM EVALUATION (SEVA) INTERFACE USED WITH ALL DEVICE PROGRAMS

3-19 PREREQUISITES FOR ALL DEVICE PROGRAMS

3-20 Origin

3-21 All programs run under the SEVA monitor control must be relocatable. The program origin must be at zero.

3-22 Instruction Format For All Programs

3-23 The first instructions for each program appear in a format similar to the following example:

0		ORG	0	
1	*	PROGRAM		
2	PST	TEXT	'PR1'	Name
3		DATA	LLOC	Last location of program
4		DATA	0	Unit address
5	PAR	RES	10	
6		GEN, 32	RESTART	
7	PST1	DATA	LABELI, 0, -1	

3-24 Explanation. The program instructions in paragraph 3-23 are as follows:

- a. Line 0 puts the program origin at zero.
- b. Line 1 is a comment line (it is not necessary).
- c. Line 2 contains a text statement with a four-letter name. If the name is less than 4 letters, blanks must follow to make up the four-character requirement. This name may be changed at load time through typewriter input.
- d. Line 3 contains a statement which will generate the address of the program location that follows the last location. This is done by putting the following statement before the END card:

LLOC EQU \$

This also means that no literals can be used in the program.

e. Line 4 is a data statement. When parameters are assigned, the monitor puts the unit address of the device to be used by the program into this location.

f. Line 5 reserves 10 locations. When parameters are assigned to this program, the monitor clears these locations to zero and stores the given parameters into this area. The first parameter stands at the first byte of the first word; new parameters always start at the first byte of the next word.

g. Line 6 contains a restart address. The routine at this address initializes the program and puts the beginning address of the program into line 7. This restart address is used by the monitor when the operator wants to reinitialize the programs after a catastrophic systems failure.

h. Line 7 generates several words of code. One of these statements must be the beginning address of the program. The number of these data words is variable, but the last one must be a -1. The monitor interrogates these locations for address. If it finds an address in any one of these locations, it branches to that address after setting the location to zero. Before the program returns control to the monitor, it makes provision for storing another address into the address table, either immediately or by interrupt routines.

3-25 Example. If, for example, the following statement was typed:

P NAME,001,ABCDEF,12,3 EOM

a unit address of X'00000001' would be stored in line 4 of the format shown in paragraph 3-23.

Starting in the first byte of the parameter table (PAR), this table would look like this:

```
PAR X'C1C2C3C4'
    X'C5C60000'
    X'F1F20000'
    X'F3000000'
    X'00000000'
```

3-26 Figure 3-1 is a sample program listing of a device program.

3-27 The instruction CALI,0 0 transfers control from the program back to the monitor. The monitor does not save any registers; the device program must do this before transferring control.

3-28 Mode

3-29 The device programs are always in the slave mode.

```

45                                     PAGE
46                                     *
47                                     *
48                                     *
49      1 00000      D7D9F140 A   PST   TEXT   *PRI *   PST TABLE
50      1 00001      C0000224      DATA  LL0C      PROGRAM LOCATIONS
51      1 00002      C0000000 A      DATA  0        UA
52      1 00003      C0000000 A      RES    10
53      1 00000      C000003F      GEN,32  REST
54      1 0000E      C0000012      PST1   DATA  PRI,0,0,-1
      1 0000F      C0000000 A
      1 00010      C0000000 A
      1 00011      FFFFFFFF A
55
56
57
58      1 00012      04500000 A   PR1   CAL1,5  0        TEST DELAY
59      1 00013      C000003C      GEN,32  PRIA     BUSY RETURN
60      1 00014      221FFFFFF A      LI,1   -1
61      1 00015      31100002      CW,1   PST+2
62      1 00016      68300019      BCR,3  PRI,8
63      1 00017      32200002      LW,2   PST+2   SET UP UA
64      1 00018      3520019C      STA,2  UA
65      1 00019      35100002      PR1,8  STW,1   PST+2
66      1 0001A      22700003      LI,7   PST+3   X7 = TABLE ADDRESS
67      1 0001B      3570019F      STW,7  RUNT,7
68      1 0001C      32700190      PR6    LW,7   RUNT,7
69      1 0001D      32100002      LW,1   PST+2   TEST FOR CHANGE OF UA
70      1 0001E      211FFFFFF A      CI,1   -1
71      1 0001F      6930003C      BCR,3  PRIA     RETURN TO MONITOR
72      1 00020      321E0000 A      LW,1   0,7
73      1 00021      21100000 A      CI,1   0
74      1 00022      6830003C      BCR,3  PRIA     TEST FOR END OF INPUT
75      1 00023      211FFFFFF A      CI,1   -1
76      1 00024      6830003C      BCR,3  PRIA     TEST FOR END OF TABLE
77      1 00025      25100070 A      SLS,1  -16
78      1 00026      211003C2 A      CI,1   X^C3C2^   TEST FOR CE
    
```

Figure 3-1. Sample Program Listing of a Device Program

3-30 MONITOR SUBROUTINES, GENERAL

3-31 The following monitor subroutines serve as an interface with the device programs. There are eight subroutines in all.

3-32 SUBROUTINE DESCRIPTIONS

3-33 Error Reporting

3-34 Any time a program wishes to report an error, it calls upon a monitor subroutine. The program indicates the program name, the number of words to be typed, and the mode (decimal or hexadecimal). This will be typed out as part of the error message.

3-35 Start I/O

3-36 No unit test can start an I/O operation; the monitor handles all I/O operations. The program calls on the monitor SIO routine and provides it with a unit address, command words, and the address of an interrupt routine for the unit test.

3-37 The HIO, TIO, and TDV Functions

3-38 These three functions are accommodated in the monitor. The unit address information is supplied by the calling program.

3-39 AIO

3-40 All I/O interrupts are handled by the monitor. Interrupts from devices which have already been started cause the AIO routine to branch to subroutines, the addresses of which have been provided in the calling sequence for the SIO routine. Any spurious interrupts from devices are reported as errors by the monitor.

3-41 Delay Routine

3-42 After being initially loaded, all device programs are executed as fast as possible. However, it may be desirable to delay the start of a program to let another program progress up to a certain point. To accomplish this, the console INTERRUPT is used. If the console INTERRUPT is activated, the monitor will interrogate SS4. If SS4 is set, the delay count is incremented; if SS4 is reset, the delay count is decremented. The delay for the program is accomplished by setting the counter 3 pulse to the desired value. The delay in the program is a function of the delay count and the counter 3 pulse frequency. If the counter value, derived by the setting of the console

INTERRUPT and SS4, is larger than zero, the monitor will start the counter 3 pulse to get a delay before the program is started.

3-43 Profile Routine

3-44 The profile routine puts the program name and identification into a profile table in order to help the operator evaluate any errors which occur. The profile routine should be called before an I/O operation is started.

3-45 MONITOR SUBROUTINE CALL SEQUENCES

3-46 Start I/O Subroutine

3-47 Call Sequence. See table 3-1.

Table 3-1. Call Sequence of the Start I/O Routine

Line No.	Call Sequence	Explanation
1	CAL, 1 0	Call to branch to this routine
2	GEN, 32 UNITAD	Address of unit address
3	GEN, 32 BUSY	Busy return
4	GEN, 32 NOTBUSY	Not busy return
5	GEN, 32 NOTACC	Not accepted return
6	GEN, 32 INTERRUPT	Interrupt routine address
7	GEN, 32 DA (COMDW)	Command doubleword address used with SIO instruction

3-48 Description of SIO Call Sequence. Line 1 transfers control to the Start I/O routine. The routine returns to the address given in line 3 if another program has started the device requested. It returns to the address given in line 4 if the device has not already been used and the SIO is accepted by the device. (XR14 and XR15 contain the status response during the SIO, and B0 through B3 of XR14 contain the condition codes.) The Start I/O routine will return to the address given in line 5 if the SIO is not accepted by the device; that is, if CC1 or CC2 are set. XR14 and XR15 contain the same information as in line 4.

3-49 When a particular device that has started sends an interrupt, the monitor branches to line 6, the interrupt routine address. At this time, the program is in the master mode, and I/O instructions, such as TIO, HIO, and TDV can be performed. When the monitor branches to the

interrupt routine, XR14 contains the AIO response, and XR13 (bits 0 through 3) contains the condition code bits CC1 through CC4. Before returning to the monitor, XR13 should be loaded with a -1 if the monitor is to consider the I/O function of the device to be finished. If more interrupts are expected from the device without further SIO instructions, register 13 should not be -1.

3-50 The HIO, TIO, and TDV Routines

3-51 Table 3-2 contains the calling sequences for these I/O instructions.

Table 3-2. Calling Sequences for HIO, TIO, and TDV

Line No.	Calling Sequence	Explanation
HIO Routine		
1	CAL,2 0	Address of unit, address location
2	DATA UNITADDR	
TIO Routine		
1	CAL,3 0	Address of unit, address location
2	DATA UNITADDR	
TDV Routine		
1	CAL,4 0	Address of unit, address location
2	DATA UNITADDR	
Note: The monitor will return to the location following the data statement.		

3-52 Registers 14 and 15 contain the status information received when executing the specified instruction. Bits 0 through 3 of register 14 contain the condition codes received.

3-53 Error Routine

3-54 Calling Sequence. Table 3-3 contains the calling sequence for this routine.

Table 3-3. Calling Sequence for Error Routine

Line No.	Calling Sequence	Explanation
1	BAL, 1 * ERROR	Branch to this routine
2	GEN, 32 BUSYRT	Busy return address
3	GEN, 32 NOTBUSY	Not busy return address (The output was accepted)
4	DATA NUMBER	Number of words to be typed
5	DATA X'N'	Conversion key
6	TEXT 'NAME'	Program name
7	GEN, 32 ER	Address of first word of error output

3-55 Description. The label ERROR (in line 1) is explained in the equate table as described in paragraph 3-57. The error routine takes N number of words (given in line 4) and, starting at the location given in line 7, converts each word as specified in line 5. These words are then typed out. Line 6 also is typed to identify the error typeout.

3-56 Line 5 tells the error routine whether to convert the words to hexadecimal or decimal output. Bit 0 of the data statement of line 5 corresponds to the first word in the error output; bit 1 to the second word, etc. A one (1) in those positions will cause a decimal conversion; a zero (0) will cause a hexadecimal conversion.

3-57 Equate Table. The equate table consists of a number of EQU statements (as for example: ERROR EQU XXX), some of which refer to constants which are stored in the monitor. Thus, a program can use constants out of this table without generating its own data statements. In the typeout

B21 EQU XXX

B21 stands for bit 21. If this label is used, the constant is X'00000400'.

3-58 Figure 3-2 is a sample printout of an equate table.

3-59 Delay Routine

3-60 Table 3-4 contains the calling sequence for this routine.

```

1          *CATALOG NO.704349 BUFFERED LINE PRINTER SYSTEM TEST
2          SYSTEM   SIG7FDP
3
4          *
5          *          EQU  TABLE
6          *
7          00000092  ERROR  EQU    X°92°
8          00000080  THREE  EQU    X°80°
9          000000E0  ONE5   EQU    X°E0°
10         000000E6  ZERO   EQU    X°E6°
11         000000E7  B31   EQU    X°E7°
12         000000EA  B30   EQU    B31+1
13         000000E9  B29   EQU    B31+2
14         000000EA  B28   EQU    B31+3
15         000000E8  B27   EQU    B31+4
16         000000EC  B26   EQU    B31+5
17         000000ED  B25   EQU    B31+6
18         000000EE  B24   EQU    B31+7
19         000000EF  B23   EQU    B31+8
20         000000F0  B22   EQU    B31+9
21         000000F1  B21   EQU    B31+10
22         000000F2  B20   EQU    B31+11
23         000000F3  B19   EQU    B31+12
24         000000F4  B18   EQU    B31+13
25         000000F5  B17   EQU    B31+14
26         000000F6  B16   EQU    B31+15
27         000000F7  B15   EQU    B31+16
28         000000FA  B14   EQU    B31+17
29         000000F9  B13   EQU    B31+18
30         000000FA  B12   EQU    B31+19
31         000000FB  B11   EQU    B31+20
32         000000FC  B10   EQU    B31+21
33         000000FD  B9    EQU    B31+22
34         000000FE  B8    EQU    B31+23
35         000000FF  B7    EQU    B31+24
36         00000100  B6    EQU    B31+25
37         00000101  B5    EQU    B31+26
38         00000102  B4    EQU    B31+27
39         00000103  B3    EQU    B31+28
40         00000104  B2    EQU    B31+29
41         00000105  B1    EQU    B31+30
42         00000106  B0    EQU    B31+31
43         000000E7  ONE   EQU    B31
44         000000E8  TWO   EQU    B31+1
         000000E9  FOUR  EQU    B31+2
  
```

Figure 3-2. Sample Listing of an Equate Table

Table 3-4. Calling Sequence of Delay Routine

Line No.	Calling Sequence	Explanation
1	CAL, 5 0	Call delay routine
2	GEN, 32 NOTSTART	Return for not starting

3-61 The delay routine returns to the line following the GEN, 32 statement if the delay requested by the operator has elapsed. If not, the delay routine returns to the address given in line 2.

3-62 Profile Routine

3-63 Calling Sequence. See table 3-5.

Table 3-5. Calling Sequence for Profile Routine

Line No.	Calling Sequence	Explanation
1	CAL, 6 0	Branch to this routine
2	DATA ADDRNAME	Address of program name
3	TEXT 'bXXX'	Profiled ID

3-64 Description. Line 2 contains the address where the name of the program is to be found. Line 3 contains a text statement with 4 characters, the first character of which has to be a blank. The other three characters are optional as to content.

3-65 If this routine returned to a busy address on the first try, control must return to the monitor before entering the monitor routine again. (This is true of all device programs; the program should always return control to the monitor if they are waiting for completion of some input-output operation or a delay. All subroutines in the monitor will save all registers.)

SECTION IV
PROGRAM LISTING

4-1 GENERAL

4-2 The program listing which follows details the contents of this systems test monitor program.

4-3 Below is a sample printout of a line from the program listing, with an explanation of what is contained in each column. There can be as many as nine columns in the program listing, but not every column will appear on every line.

EXAMPLE:

61	1	00015	04500000	A	PT1	CAL1,5	0	TEST DELAY
a	b	c	d	e	f	g	h	i

EXPLANATION:

- a. Line number
- b. Indication of memory protection key (applies to Sigma 5 and 7 only)
- c. Memory address
- d. Routine instruction and data
- e. Indication whether of absolute origin or not
- f. Field label
- g. Operation
- h. Operand
- i. Comments

```

1          *CATALOG NO 704138 SYSTEM TEST MONITOR
2          SYSTEM SIG7FDP
3          *
4          *
5          *
6          1 00040          BRG          X'40'
7          1 00040          OF000042      XPSD,0  NA8PD          NOT ALL. OPERATION
8          1 00041          OF00004C      XPSD,0  NIPID          NOT IMPL. INSTR.
9          *
10         1 00042          00000000 A    NA8PD  DATA  0,0
11         1 00043          00000000 A
12         1 00044          00000112      GEN,32  NA8P
13         1 00045          01000000 A    DATA  X'01000000'
14         *
15         1 00046          OF00005E      XPSD,0  WDTID          WATCHDOG TIMER
16         1 00047          00000000 A    DATA  0
17         1 00048          CF400062      XPSD,4  CALD1          CALL 1
18         1 00049          00000000 A    REST   DATA  0
19         1 0004A          00000000 A    FINT   DATA  0
20         1 0004B          00000000 A    TIMST  DATA  0
21         *
22         1 0004C          00000000 A    NIPID  DATA  0,0
23         1 0004D          00000000 A
24         1 0004E          0000011B      GEN,32  NIPI
25         1 0004F          01000000 A    DATA  X'01000000'
26         *
27         1 00050          OF0000E2      XPSD,0  P8W8ND        POWER ON
28         1 00051          CF000070      XPSD,0  P88FFD        POWER OFF
29         1 00052          00000000 A    CPUL3  DATA  0
30         1 00053          00000000 A    PRID   DATA  0
31         1 00054          00000000 A    TEMP   DATA  0
32         1 00055          33F00052      MTW,15  CPUL3
33         1 00056          OF000074      XPSD,0  MEMPD          MEMORY PARITY
34         1 00057          00000000 A    TYPE   DATA  0
35         1 00058          000001A4      IN14   GEN,32  M5
36         1 00059          01000000 A    DATA  X'01000000'
  
```

```

35         1 0005A          00000000 A    TYPERR DATA  0
36         1 0005B          OF000078      XPSD,0  CBUND
37         1 0005C          OF00007C      XPSD,0  AI8D          I8 INT
38         1 0005D          OF000080      XPSD,0  CBND          CONTRL PANEL INT.
39         1 0005E          00000000 A    WDTID  DATA  0,0
40         1 0005F          00000000 A
41         1 00060          00000123      GEN,32  WDTI
42         1 00061          01000000 A    DATA  X'01000000'
43         1 00062          00000000 A    CALD1  DATA  0,0
44         1 00063          00000000 A
45         1 00064          00000366      GEN,32  CALD11
46         1 00065          01000000 A    DATA  X'01000000'
47         *
48         1 00066          680001F0      CALD11  B          M11X          CC 0
49         1 00067          68000372      B          S18          CC 1
50         1 00068          680003D8      B          H18          CC 2
51         1 00069          680003DB      B          T18          CC 3
52         1 0006A          640003DE      B          TDV          CC 4
53         1 0006B          68000133      B          DELAY        CC 5
54         1 0006C          68000402      B          PR8FILE      CC 6
55         1 0006D          40404040 A    BLANK   DATA  X'40404040'
56         1 0006E          00000000 A    TYPERR  DATA  0
57         1 0006F          00000169      RTAA   GEN,32  RTA
58         1 00070          00000000 A    P88FFD DATA  0,0          POWER OFF RT
59         1 00071          00000000 A
60         1 00072          0000012B      GEN,32  P88FF
61         1 00073          01000000 A    DATA  X'01000000'
62         1 00074          00000000 A    MEMPD  DATA  0,0          MEMORY PARITY
63         1 00075          00000000 A
64         1 00076          0000012D      GEN,32  MEMP
65         1 00077          01000000 A    DATA  X'01000000'
66         1 00078          00000000 A    CBUND  DATA  0,0          COUNTER 3 ZERO INT.
67         1 00079          00000000 A
68         1 0007A          0000015C      GEN,32  CBUN
69         1 0007B          01000000 A    DATA  X'01000000'
70         1 0007C          00000000 A    AI8D   DATA  0,0          I8 INTERRUPT
71         1 0007D          00000000 A
  
```

```

66 1 0007E 000003A9 GEN,32 A18
67 1 0007F 01000000 A DATA X'01000000'
68 1 00080 00000000 A COND DATA 0,0 CONTROL PANEL INT.
    1 00081 00000000 A
69 1 00082 00000151 GEN,32 C0N
70 1 00083 01000000 A DATA X'01000000'
71
72 1 00084 00000000 A PRSD DATA 0
73 1 00085 01000000 A DATA X'01000000'
    
```

```

74
75 1 00090 PAGE
    1 00090 BRG 144
76 1 00090 000007FF A X7FF DATA X'7FF'
77 1 00091 00000000 A ER DATA 0
78 1 00092 0000046F ERROR GEN,32 ERR0R1
79 1 00093 0000033C BIDEK GEN,32 BIDEK1
80 1 00094 00000000 A SIBT DATA 0,0,0,0,0,0,0,0,0
    1 00095 00000000 A
    1 00096 00000000 A
    1 00097 00000000 A
    1 00098 00000000 A
    1 00099 00000000 A
    1 0009A 00000000 A
    1 0009B 00000000 A
    1 0009C 00000000 A
    1 0009D 00000000 A
81 1 0009E FFFFFFFF A DATA =1
82 1 0009F 0000035F BINHE GEN,32 BINHE1
83 1 000A0 00000000 A SIBTA DATA 0,0,0,0,0,0,0,0,0
    1 000A1 00000000 A
    1 000A2 00000000 A
    1 000A3 00000000 A
    1 000A4 00000000 A
    1 000A5 00000000 A
    1 000A6 00000000 A
    1 000A7 00000000 A
    1 000A8 00000000 A
    1 000A9 00000000 A
84 1 000AA FFFFFFFF A DATA =1
85 1 000AB 00000000 A SAX7 DATA 0
86 1 000AC FFF00000 A XFFFOU DATA X'FFF00000'
87 1 000AD 00000000 A TYPEC DATA 0
88 1 000AE 00000000 A RTSA DATA 0
89 1 000AF 00000000 A TEND DATA 0
90
91 1 000B0 00000003 A THREE DATA 3
    
```

92	1	00CB1	00800000	A	PRSD1	DATA	X'00800000'
93	1	00CB2	00000000	A	SX0	DATA	0
94	1	00CB3	00000000	A	SX1	DATA	0
95	1	00CB4	00000000	A	SX2	DATA	0
96	1	00CB5	00000004	A	SX4	DATA	4
97	1	00CB6	00000003	A	SX3	DATA	3
98	1	00CB7	00000000	A	SX7	DATA	0
99	1	00CB8	00000000	A	SX14	DATA	0
100	1	00CB9	00000000	A	SX15	DATA	0
101	1	00CBA			SAVE	RES	16
102	1	00CCA	00000000	A	TIR	DATA	0
103	1	00CCB	00000000	A	MTR	DATA	0,0,0,0,0,0,0,0,0
	1	00CCC	00000000	A			
	1	00CCD	00000000	A			
	1	00CCE	00000000	A			
	1	00CCF	00000000	A			
	1	00CC0	00000000	A			
	1	00CC1	00000000	A			
	1	00CC2	00000000	A			
	1	00CC3	00000000	A			
	1	00CC4	00000000	A			
104	1	00CC5	FFFFFFFF	A		DATA	-1
105	1	00CC6	00000000	A	MTA	DATA	0,0,0,0,0,0,0,0,0
	1	00CC7	00000000	A			
	1	00CC8	00000000	A			
	1	00CC9	00000000	A			
	1	00CCA	00000000	A			
	1	00CCB	00000000	A			
	1	00CC0	00000000	A			
	1	00CCD	00000000	A			
	1	00CCE	00000000	A			
	1	00CCF	00000000	A			
106	1	00CE0	FFFFFFFF	A	ONES	DATA	-1
107	1	00CE2				BRUN'D	8
108	1	00CE2	00000000	A	P8W8ND	DATA	0,0
	1	00CE3	00000000	A			
109	1	00CE4	0000012A			GEN,32	P8W8N

110	1	00CF5	01000000	A		DATA	X'01000000'
111	1	00CE6	00000000	A	ZER8	DATA	0
112	1	00CE7	00000001	A	B31	DATA	1,2,4,8,16,32,64,128,256,512,1024,2048
	1	00CE8	00000002	A			
	1	00CE9	00000004	A			
	1	00CEA	00000008	A			
	1	00CEB	00000010	A			
	1	00CEC	00000020	A			
	1	00CED	00000040	A			
	1	00CEE	00000080	A			
	1	00CEF	00000100	A			
	1	00CF0	00000200	A			
	1	00CF1	00000400	A			
	1	00CF2	00000800	A			
113	1	00CF3	00001000	A		DATA	4096,8192,16384,32768,65536,X'200000'
	1	00CF4	00002000	A			
	1	00CF5	00004000	A			
	1	00CF6	00008000	A			
	1	00CF7	00010000	A			
	1	00CF8	00020000	A			
114	1	00CF9	00040000	A		DATA	X'400000',X'800000',X'1000000',X'2000000'
	1	00CFA	00080000	A			
	1	00CFB	00100000	A			
	1	00CFC	00200000	A			
115	1	00CFD	00400000	A		DATA	X'4000000',X'8000000',X'10000000',X'20000000'
	1	00CFE	00800000	A			
	1	00CFF	01000000	A			
	1	0C100	02000000	A			
116	1	0C101	04000000	A		DATA	X'40000000',X'80000000',X'100000000'
	1	0C102	08000000	A			
	1	0C103	10000000	A			
117	1	0C104	20000000	A		DATA	X'200000000',X'400000000',X'800000000'
	1	0C105	40000000	A			
	1	0C106	80000000	A			
118		1 000E7			8NE	EQU	B31
119		1 000E8			TW8	EQU	B31*1
120		1 000E9			FOUR	EQU	B31*2


```

121 1 00107 00000000 A ERSUPT DATA 0,0,0,0,0,0,0,0,0,-1
122 1 00108 00000000 A
123 1 00109 00000000 A
124 1 0010A 00000000 A
125 1 0010B 00000000 A
126 1 0010C 00000000 A
127 1 0010D 00000000 A
128 1 0010E 00000000 A
129 1 0010F 00000000 A
130 1 00110 00000000 A
131 1 00111 FFFFFFFF A
132 *
133 * BAD OP CODE
134 *
135 1 00112 35000119 NABP STW,0 NABP1
136 1 00113 32000042 LW,0 NABPD LOAD ADDR.
137 1 00114 68400114 BCR,4 $ CONTINUE IF MAE
138 1 00115 3310011A MTW,1 MAE
139 1 00116 33100042 MTW,1 NABPD GO TO NEXT ADDRESS
140 1 00117 32000119 LW,0 NABP1
141 1 00118 0E000042 LPSC,0 NABPD
142 1 00119 00000000 A NABP1 DATA 0
143 1 0011A 00000000 A MAE DATA 0
144 *
145 * NOT IMPL. INCTR.
146 *
147 1 0011B 35000122 NIP1 STW,0 NIP11
148 1 0011C 3200004C LW,0 NIPID LOAD ADDR.
149 1 0011D 6800011D BCR,0 $
150 1 0011E 3310004C MTW,1 NIPID
151 1 0011F 32000122 LW,0 NIP11
152 1 00120 3310004C MTW,1 NIPID RETURN
153 1 00121 0E00004C LPSC,0 NIPID
154 *
155 1 00122 00000000 A NIP11 DATA 0
156 *
157 * WATCH-DBG TIMER
158 *
159 *
160 *
161 *
162 *
163 *
164 *
165 *
166 *
167 *
168 *
169 *
170 *
171 *
172 *
173 *
174 *
175 *
176 *
177 *
178 *
179 *
180 *
181 *
182 *
183 *
184 *

```

```

148 1 00123 35000129 WDTI STW,0 WDTI1
149 1 00124 3200005E LW,0 WDTID LOAD ADDR.
150 1 00125 68000125 BCR,0 $
151 1 00126 3310005E MTW,1 WDTID
152 1 00127 32000129 LW,0 WDTI1
153 1 00128 0E00005F LPSC,0 WDTID RETURN
154 *
155 1 00129 00000000 A WDTI1 DATA 0
156 *
157 1 0012A 6AF0012A P0W0N BAL,15 $
158 *
159 1 0012B 2E000000 A P00FF WAIT
160 1 0012C 6AF0012C BAL,15 $
161 *
162 1 0012D 35000132 MEMP STW,0 MEMP1
163 1 0012E 32000074 LW,0 MEMPD LOAD ADDR.
164 1 0012F 6800012F BCR,0 $
165 *
166 1 00130 32000132 LW,0 MEMP1
167 1 00131 0E000074 LPSC,0 MEMPD RETURN
168 *
169 1 00132 00000000 A MEMP1 DATA 0
170 *
171 *
172 * DELAY ROUTINE
173 *
174 1 00133 350000B2 DELAY STW,0 SX0 SAVE XR
175 1 00134 351000B3 STW,1 SX1
176 *
177 1 00135 32000147 LW,0 TIMER TEST FOR DELAY
178 1 00136 68300145 BCR,3 DEL1
179 1 00137 3200004B LW,0 TIMST TEST FOR TIMER STARTED
180 1 00138 69300142 BCS,3 DEL2
181 1 00139 6A000148 BAL,0 DEL3 START TIMER
182 1 0013A 33100062 DEL4 MTW,1 CALD1
183 *
184 1 0013B 32100062 LW,1 CALD1

```

```

185 1 0013C 4B1004E9 AND,1 L(X'FFFE0000') SET UP NOT READY RETURN ADDRESS
186 1 0013D C9100062 BR,1 *CALD1
187 1 0013E 35100062 STW,1 CALD1
188 1 0013F 320000B2 DEL5 LW,0 SX0 RESTORE XR
189 1 00140 321000B3 LW,1 SX1
190
191 1 00141 0E000062 * LPSD,0 CALD1
192 1 00142 3200004A DEL2 LW,0 FINT TEST FOR TIMER FINISHED
193 1 00143 6830013A BCR,3 DEL4
194 1 00144 6A000148 BAL,0 DEL3 START TIMER
195 1 00145 33200062 DEL1 MTW,2 CALD1
196 1 00146 6A00013F BAL,0 DEL5
197
198 1 00147 00000000 A * TIMER DATA 0
199
200 1 00148 3310004B DEL3 MTW,1 TIMST SET TIMER START
201 1 00149 22100000 A LI,1 0
202 1 0014A 3510004A STW,1 FINT NOT FINISHED
203 1 0014B 32100147 LW,1 TIMER
204 1 0014C 2C100001 A AI,1 1
205 1 0014D 35100052 STW,1 CPUL3 ARM AND ENABLE TIMER
206 1 0014E 22101040 A LI,1 X'1040'
207 1 0014F 6D101200 A WD,1 X'1200'
208 1 00150 EA100000 A BAL,1 *C
209
210
211
212
213 1 00151 3510015B C0N STW,1 C0N1 CONTROL PANEL INTERRUPT
214 1 00152 6C000000 A RD,0 0 SAVE XR
215 1 00153 65100157 BCR,1 C0N2
216 1 00154 33700147 MTW,7 TIMER TIMER = TIMER+7
217 1 00155 3210015B C0N3 LW,1 C0N1
218 1 00156 0E300080 LPSD,3 C0N0
219 1 00157 32100147 C0N2 LW,1 TIMER TIMER = 0
220 1 00158 63300155 BCR,3 C0N3
221 1 00159 33900147 MTW,9 TIMER TIMER = TIMER-7
    
```

```

222 1 0015A 6A100155 BAL,1 C0N3
223
224 1 0015B 00000000 A C0N1 DATA 0
225
226 1 0015C 35100162 * C0UN STW,1 C0UN1 COUNT ZERO INTERRUPT
227 1 0015D 22101000 A LI,1 X'1000' SAVE
228 1 0015E 6D101100 A WD,1 X'1100' DISARM
229 1 0015F 3510004A STW,1 FINT
230 1 00160 32100162 LW,1 C0UN1 RESTORE
231 1 00161 0E300078 LPSD,3 C0UN0
232
233 1 00162 00000000 A C0UN1 DATA 0
234 1 00164 BBUND 8
235 1 00164 010005A0 M2EX GEN,8,24 X'11',BA(M2EX1)
236 1 00165 2C000004 A GEN,8,24 X'2C',4
237 1 00166 86C0005A4 GEN,8,24 X'86',BA(RTA)
238 1 00167 5E0000C8 A GEN,8,24 X'5E',200
239 1 00168 40155B40 A M2EX1 DATA X'40155B40'
240 1 00169 RTA RES 10
241
    
```

```

242                                     PAGE
243                                     *
244                                     *
245                                     *
246 1 0C173 2E000000 A M1AA WAIT
247 1 0C174 35000057 STW,0 TYPE
248 1 0C175 321000E6 LW,1 ZERO
249 1 0C176 3510011A STW,1 MAE
250 1 0C177 22500000 A LI,5 CALCULATE MEMORY SIZE
251 1 0C178 20501000 A M1BBB AI,5 X'1000'
252 1 0C179 32FA0000 A LW,15 0,5
253 1 0C17A 32FC011A LW,15 MAE
254 1 0C17B 6930017F BCS,3 M1BBA TEST FOR TRAP
255 1 0C17C 2151F000 A C1,5 X'1F000'
256 1 0C17D 69300178 BCS,3 M1BBB
257 1 0C17E 680C0180 B *+2
258 1 0C17F 3A5000F3 M1BBA SW,5 B31+12 GET CORRECT MEMORY SIZE
259 1 0C180 6650018A AWM,5 M1BBC
260 1 0C181 6C000000 A M1BB RD,0 TEST BP
261 1 0C182 688001A5 BCR,8 M1A
262 1 0C183 320004E8 LW,0 PRBL
263 1 0C184 33100000 A MTW,1 0
264 1 0C185 2500007F A SLS,0 -1
265 1 0C186 25000001 A SLS,0 1
266 1 0C187 350004E8 STW,0 PRBL
267 1 0C188 350200C2 STW,0 MTR,1 SET MTR TABLE
268 1 0C189 35100054 STW,1 TEMP SAVE XR1
269 1 0C18A 6A800F84 A M1BBC BAL,11 X'F84' LOAD TEST
270 1 0C18B 32100054 LW,1 TEMP RESTORE XR1
271 1 0C18C 320004E8 LW,0 PRBL
272 1 0C18D 25000002 A SLS,0 2
273 1 0C18E 322001A0 LW,2 MX1 SET UP TY DA
274 1 0C18F 482000AC AND,2 XFFFF0
275 1 0C190 49200000 A BR,2 0
276 1 0C191 352001A0 STW,2 MX1 MERGE BITS
277 1 0C192 22000000 LI,0 DA(MX1) CA
278 1 0C193 C0000057 SI0,0 *TYPE
    
```

```

279 1 0C194 C0000057 TI0,0 *TYPE REQUEST INPUT
280 1 0C195 69000194 BCS,12 *-1 WAIT FOR TY READY
281 1 0C196 331004E8 MTW,1 PRBL
282 1 0C197 B22004E8 LW,2 *PRBL FETCH NEXT PROGRAM LOCATION
283 1 0C198 352004E8 STW,2 PRBL
284 1 0C199 33100001 A MTW,1 1 XR1=XR1+1
285 1 0C19A 32FC01A2 LW,15 BM1A INIT. LOC. X'26'
286 1 0C19B 35FC0026 A STW,15 X'26'
287 1 0C19C 60000000 A RD,0 0
288 1 0C19D 6840019F BCR,4 *-2 TEST BP2
289 1 0C19F 2E000000 A WAIT
290 1 0C19F 6AF00181 BAL,15 M1BB
291
292 1 0C1A0 * BOUND 8
293 1 0C1A0 86000000 A MX1 GEN,8,24 X'86',0
294 1 0C1A1 0E000004 A GEN,8,24 X'E1',4
295
296 1 0C1A2 680001A5 BM1A B M1A
297 *
    
```

298 PAGE
 299 *
 300 *
 301 *
 302 1 001A3 2E000000 A M1 WAIT
 303 1 001A4 35000057 STW,0 TYPE XO = TYPEWRITER ADDRESS
 304 *
 305 *
 306 1 001A5 221008B0 A M1A LI,1 X'8B0'
 307 1 001A6 6D101200 A WD,1 X'1200'
 308 1 001A7 6C000000 A RD,0 0
 309 1 001A8 694001F2 BCS,4 M3
 310 1 001A9 0E000058 LPSD,0 IN14
 311 * FIND RUN REQUEST,BREAKPOINTS
 312 1 001AA 3200005A M5 LW,0 TYPER TEST INPUT START
 313 1 001AB 69300216 BCS,3 M2
 314 1 001AC 321003D3 LW,1 ERINT
 315 1 001AD 683001B2 BCR,3 \$+5
 316 1 001AE 350003D3 STW,0 ERINT
 317 1 001AF 22100003 A LI,1 3 REPORT ERRONEOUS INTERRUPT
 318 1 001B0 351001CC STW,1 M2AD
 319 1 001B1 64FC01D5 BAL,15 TERINT
 320 1 001B2 6C000000 A RD,0 0 READ BP
 321 1 001B3 69200216 BCS,2 M2 BP3,TYPE REQ
 322 1 001B4 6C000000 A M12 RD,0 0 BP2,RESTART
 323 1 001B5 694001F2 BCS,4 M3 BP1,RUN BK IF SET
 324 1 001B6 688001AA BCR,8 M5
 325 * IDENTIFY TEST TO BE RUN
 326 *
 327 1 001B7 327000E6 M4 LW,7 ZER0 XR7=0
 328 1 001B8 320E00D6 M6 LW,0 MTA,7
 329 1 001B9 683001C2 BCR,3 M9 TEST FOR ZER0 ENTRY
 330 1 001BA 210FFFFF A CI,0 -1 TEST FOR TERMINATOR
 331 1 001BB 683001AA BCR,3 M5
 332 1 001BC 326000E6 M10 LW,6 ZER0 XR6=0
 333 1 001BD 321C00C9 M7 LW,1 MTR,6 GET LOCATION OF TEST
 334 1 001BE 211FFFFF A CI,1 -1 TEST FOR TERMINATOR

335 1 001BF 693001D0 BCS,3 M7A
 336 1 001C0 22100001 A LI,1 1 BRANCH IF NO TERMINATOR
 337 1 001C1 64FC01C4 BAL,15 M2AAA REPORT ERROR
 338 1 001C2 33100007 A M9 MTR,1 7
 339 1 001C3 64F001B8 BAL,15 M6
 340 *
 341 *
 342 *
 343 * REPORT ERROR WRONG ID
 344 1 001C4 351001CC M2AAA STW,1 M2AD
 345 1 001C5 EA100092 M2AB BAL,1 *ERROR CALL ERROR RT
 346 1 001C6 000001C5 GEN,32 M2AB BSY RT
 347 1 001C7 000001CF GEN,32 M2AC MBSY RT
 348 1 001C8 00000001 A DATA 1
 349 1 001C9 00000000 A DATA 0 HEX
 350 1 001CA D4C6D540 A DATA X'D4D6D540' PR0GR. NAME = MON
 351 1 001CB 000001CC GEN,32 M2AD
 352 *
 353 1 001CC 00000000 A M2AD DATA 0
 354 1 001CD 00000000 A M2ADX DATA 0,0
 355 1 001CE 00000000 A *
 356 1 001CF EAF0000F A M2AC BAL,15 *15
 357 *
 358 1 001D0 B2200001 A M7A LW,2 *1
 359 1 001D1 31200000 A CW,2 0 COMPARE PROGRAM TITLE
 360 1 001D2 683001DC BCR,3 M7C
 361 1 001D3 331C0006 A MTR,1 6 X6=X6+1
 362 1 001D4 64F001BD BAL,15 M7
 363 *
 364 1 001D5 EA100092 TERINT BAL,1 *ERROR REPORT ERRONEOUS INT.
 365 1 001D6 000001D5 GEN,32 TERINT
 366 1 001D7 000001CF GEN,32 M2AC
 367 1 001D8 00000003 A DATA 3
 368 1 001D9 00000000 A DATA 0
 369 1 001DA D4C6D540 A DATA X'D4D6D540'
 370 1 001DB 000001CC GEN,32 M2AD

371

16

372				PAGE			
373			*				
374			*		GO TO PROGRAM IF DESIRED		
375			*				
376	1	001DC	201000CE A	M7C	AI,1	14	XRI=XR1+14
377	1	001DD	322C0049		LW,2	REST	TEST RESTART
378	1	001DF	683C01E2		BCR,3	M8	
379	1	001DF	33F00C01 A		MTW,15	1	XRI=XR1-1
380	1	0C1E0	B2200001 A		LW,2	*1	LOAD ADDRESS
381	1	0C1E1	6AF001E8		BAL,15	M11	
382	1	0C1E2	B2200001 A	M8	LW,2	*1	
383	1	0C1E3	693C01E6		BCS,3	M7D	
384	1	0C1E4	33100001 A		MTW,1	1	XRI=XR1+1,GET NEXT PST ENTRY
385	1	0C1E5	6AF001E2		BAL,15	M8	
386	1	001E6	212FFFFF A	M7D	CI,2	-1	TEST FOR TERMINATOR
387	1	0C1E7	683001C2		BCR,3	M9	GET NEXT PROGRAM
388	1	001E8	357000AB	M11	STW,7	SAX7	SAVE X7
389	1	0C1E9	492C00B1		BR,2	PRSD1	STORE PROSDWORD ADDR FOR PROGRAM
390	1	0C1EA	35200084		STW,2	PRSD	PRSD1 =X'07B00000'
391	1	001EB	320000E6		LW,0	ZER0	
392	1	0C1EC	32200049		LW,2	REST	TEST FOR RESTART
393	1	0C1ED	693001EF		BCS,3	M7E	
394	1	001EE	B5000C01 A		STW,0	*1	CLEAR PST
395	1	0C1EF	0E000084	M7E	LPSD,0	PRSD	ENTER PROGRAM
396				*			RETURN BY CAL1,CC=0
397	1	001FO	327C00AB	M11X	LW,7	SAX7	
398	1	001F1	6AF001C2		BAL,15	M9	
399				*			

```

400                                     PAGE
401                                     *
402                                     *
403 1 001F2 221008B0 A M3 LI,1 RESTART PROCEDURE
404 1 001F3 6D101100 A WD,1 X'8B0'
405 1 001F4 6D101300 A WD,1 X'1100' DISARM
406 1 001F5 32100049 LW,1 X'1300' DISABLE
407 1 001F6 693001FF BCS,3 M3A SET RESTART AND TEST FOR BSY
408 1 001F7 220FFFFF A LI,0 -1
409 1 001FR 35000049 STW,0 REST
410 1 001F9 3510004A STW,1 FINT RESET TIMER
411 1 001FA 3510004B STW,1 TIMST TIMER NOT STARTED
412 1 001FB 3510005A STW,1 TYPER RESET TYPEREQUEST
413 1 001FC 35100091 STW,1 ER RESET ERROR RT
414 1 001FD 351003D3 STW,1 ERINT CLEAR ERR, INTERRUPT
415 1 001FE 6AF00203 BAL,15 M3B GO TO RESTART
416 1 001FF 22000000 A M3A LI,0 0
417 1 00200 35000049 STW,0 REST
418 1 00201 2E000000 A WAIT FOR RESET IO
419 1 00202 6AFC01A5 BAL,15 M1A
420                                     *
421                                     *
422 1 00203 22200000 A M3B LI,2 0
423 1 00204 32040094 M3D LW,0 SI0T,2
424 1 00205 210FFFFF A CI,0 -1 CLEAR SI0 TABLE
425 1 00206 68300208 BCR,3 M3C
426 1 00207 35140107 STW,1 ERSUPT,2
427 1 00208 35140094 STW,1 SI0T,2
428 1 00209 33100002 A MTW,1 2
429 1 0020A 6AF00204 BAL,15 M3D
430 1 0020B 351003FF M3C STW,1 ERRT
431 1 0020C 32100215 LW,1 BLANKX
432 1 0020D 22200036 A LI,2 PRBE-PR0FT SET PROFILE TABLE TO BALNKS
433 1 0020E 35140417 STW,1 PR0FT-1,2
434 1 0020F 6420020E BDR,2 *-1
435 1 00210 321003D6 LW,1 AI0SA*2
436 1 00211 351003D4 STW,1 AI0SA RESTORE REGISTER SAVE
    
```

```

437 1 00212 321003D7 LW,1 AI0SA+3
438 1 00213 351003D6 STW,1 AI0SA+1
439 1 00214 680001B7 B M4
440 1 00215 435C5C4C A BLANKX TEXT I ** I
441                                     *
442                                     *
443                                     *
444 1 00216 3200005A M2 LW,0 TYPER HAS INPUT BEEN STARTED
445 1 00217 69300223 BCS,3 M2A
446 1 00218 350000AD STW,0 TYPEC INIT, TYPECOMPLETE
447 1 00219 04100000 A CALL,1 0 CALL SI0
448 1 0021A 00000057 GEN,32 TYPE JA OF TYPEWRITER
449 1 0021B 000001B4 GEN,32 M12 BUSY RETURN
450 1 0021C 00000222 GEN,32 M2A1 NOT BSY RT
451 1 0021D 000001B4 GEN,32 M12 NOT ACCEPTED RT
452 1 0021E 000003EC GEN,32 M2B INTERRUPT ADDRESS
453 1 0021F 000000B2 GEN,32 DA(M2EX) COMMAND DOUBLE WORD ADDRESS
454                                     *
455                                     *
456 1 00220 6AF00220 M2D BAL,15 M2D TYPEWRITER HANGUP
457 1 00221 6AF001B4 BAL,15 M12
458 1 00222 33F0005A M2A1 MTW,15 TYPER
459 1 00223 3P0000AD M2A LW,0 TYPEC TEST FOR TYPE COMPLETED
460 1 00224 683001B4 BCR,3 M12 BRANCH IF NOT COMPLETED
461 1 00225 226C0000 A LI,6 0
462 1 00226 3560005A STW,6 TYPER
463 1 00227 3200006E LW,0 TYPERR
464 1 00228 6830022C BCR,3 M2B5
465 1 00229 321000CA LW,1 TIR
466 1 0022A 6AF001C4 BAL,15 M2AAA
467 1 0022B 6AF001AA BAL,15 M5
468                                     *
469                                     *
470                                     *
471                                     *
472                                     *
473                                     *
    
```

474	1	0022C	32000169	M2B5	LW,0	RTA	
475	1	0022D	31000400		CW,0	SERR	
476	1	0022E	69300232		BCS,3	**4	
477	1	0022F	220FFFFFF A		LI,0	-1	
478	1	00230	350003FF	M2B5A	STW,0	ERRT	
479	1	00231	680001B4		B	M12	
48C	1	00232	31000401		CW,0	PERR	TEST FOR PRINTING ERRORS
481	1	00233	69300236		BCS,3	**3	
482	1	00234	22000000 A		LI,0	0	
483	1	00235	68000230		B	M2B5A	
484	1	00236	31000456		CW,0	PRBC0	TEST FOR PRINTING PROFILE
485	1	00237	6930023A		BCS,3	**3	
486	1	00238	3500046E		STW,0	PROFID	
487	1	00239	680001B4		B	M12	TYPE PROFILE ON ERROR
488	1	0023A	31000457		CW,0	NPR0C0	
489	1	0023B	6930023F		BCS,3	**4	
49C	1	0023C	22000000 A		LI,0	0	
491	1	0023D	3500046E		STW,0	PROFID	CLEAR PROFILE PRINT OUT
492	1	0023E	680001B4		B	M12	
493	1	0023F	22700000 A		LI,7	0	
494	1	00240	720E0169		LB,0	RTA,7	
495	1	00241	350000AE		STW,0	RTSA	SAVE ORDER
496	1	00242	210000D7 A		CI,0	X'D7'	TEST FOR P
497	1	00243	683002D9		BCR,3	M13	
498	1	00244	210000C1 A		CI,0	X'C1'	TEST FOR A
499	1	00245	68300251		BCR,3	M2AA	
50C	1	00246	210000C4 A		CI,0	X'C4'	TEST FOR D
501	1	00247	68300251		BCR,3	M2AA	
502	1	00248	210000D9 A		CI,0	X'D9'	TEST FOR R
503	1	00249	68300251		BCR,3	M14	
504	1	0024A	210000E2 A		CI,0	X'E2'	TEST FOR S
505	1	0024B	68300251		BCR,3	M14	
506	1	0024C	210000E3 A		CI,0	X'E3'	TEST FOR T
507	1	0024D	68300251		BCR,3	M14	
508				*			
509	1	0024E	22100002 A	M2B4	LI,1	2	REPORT ERROR 2, NO ID
51C	1	0024F	6AF001C4		BAL,15	M2AAA	

511	1	00250	6AF001AA		BAL,15	M5	
512				*			
513				*			SET UP TABLE
514				*			
515	1	00251	33100007 A	M14	MTW,1	7	X7=X7+1
516	1	00252	721E0169		LB,1	RTA,7	
517	1	00253	31700CAF		CW,7	TEND	TEST FOR END
518	1	00254	6830025C		BCR,3	M20A	
519	1	00255	21100040 A		CI,1	X'40'	TEST FOR SPACE
52C	1	00256	68300251		BCR,3	M14	
521	1	00257	2110004E A		CI,1	X'4E'	TEST FOR +
522	1	00258	6830025C		BCR,3	M20A	
523	1	00259	751C0169		STB,1	RTA,6	STORE BYTE IN TABLE
524	1	0025A	33100006 A	M14B	MTW,1	6	
525	1	0025B	6AF00251		BAL,15	M14	
526		1 00251		M2AA	EQU	M14	
527				*			
528	1	0025C	22300040 A	M20A	LI,3	X'40'	X3 = SPACE
529	1	0025D	32200006 A	M14C	LW,2	6	
530	1	0025E	2520001E A		SLS,2	30	
531	1	0025F	21200000 A		CI,2	0	TEST FOR X6= X00
532	1	00260	68300264		BCR,3	M14AA	
533	1	00261	753C0169		STB,3	RTA,6	
534	1	00262	20600001 A		AI,6	1	X6=X6+1
535	1	00263	6AF00250		BAL,15	M14C	
536				*			
537	1	00264	32200006 A	M14AA	LW,2	6	
538	1	00265	2520007E A		SLS,2	-2	GET TABLE ADDRESS
539	1	00266	20200168		AI,2	RTA-1	
54C	1	00267	B2200002 A		LW,2	*2	X2 = PROGRAM NAME
541	1	00268	22400000 A		LI,4	0	
542	1	00269	325800CB	M14AA2	LW,5	MTR,4	TEST WHETHER PROGRAM IS IN CORE
543	1	0026A	215FFFFFF A		CI,5	-1	
544	1	0026B	6930026E		BCS,3	M14AA1	
545	1	0026C	22100001 A		LI,1	1	
546	1	0026D	6800024F		B	M2B4+1	REPORT ERROR 1 IF NOT PRESENT
547	1	0026E	B2500005 A	M14AA1	LW,5	*5	

548	1	0026F	31500002 A		CW,5	2		COMPARE NAMES
549	1	00270	68300273		BCR,3	M14A		
550	1	00271	20400001 A		LI,4	1		
551	1	00272	68000269		B	M14AA2		
552				*				
553	1	00273	317000AF	M14A	CW,7	TEND		
554	1	00274	69300251		BCS,3	M14		GET NEXT NAME
555	1	00275	2560007E A		SLS,6	-2		
556	1	00276	222FFFFF A		LI,2	-1		STORE 0 IN LAST LOCATION
557	1	00277	352C0169		STW,2	RTA,6		
558	1	00278	320000AE		LW,0	RTSA		
559	1	00279	210000D9 A		CI,0	X'D9'		TEST FOR R
560	1	0027A	6930028C		BCS,3	M20B		
561	1	0027B	22700000 A		LI,7	0		X7=0
562	1	0027C	321E0169	M20E	LW,1	RTA,7		
563	1	0027D	211FFFFF A		CI,1	-1		TEST FOR TERMINATOR IN RTA
564	1	0027E	69300286		BCS,3	M20C		
565	1	0027F	321E00D6	M20D	LW,1	MTA,7		
566	1	00280	211FFFFF A		CI,1	-1		TEST FOR TERMINATOR IN MTA
567	1	00281	683001B4		BCR,3	M12		
568	1	00282	22100000 A		LI,1	0		STORE ZERO IN MTA
569	1	00283	351E00D6		STW,1	MTA,7		
570	1	00284	33100007 A		MTW,1	7		
571	1	00285	6AF0027F		BAL,15	M20D		
572				*				
573	1	00286	322E00D6	M20C	LW,2	MTA,7		TEST FOR TERMINATOR IN MTA
574	1	00287	212FFFFF A		CI,2	-1		
575	1	00288	683001B4		BCR,3	M12		
576	1	00289	351E00D6		STW,1	MTA,7		STORE RTA IN MTA
577	1	0028A	33100007 A		MTW,1	7		
578	1	0028B	6AF0027C		BAL,15	M20E		
579				*				
580				*				
581	1	0028C	210000C1 A	M20B	CI,0	X'C1'		TEST FOR A
582	1	0028D	6930029D		BCS,3	M16		
583	1	0028E	22600000 A		LI,6	0		
584	1	0028F	22700000 A		LI,7	0		

585	1	00290	321E00D6	M15B	LW,1	MTA,7		TEST FOR MTA = 0
586	1	00291	68300296		BCR,3	M15A		
587	1	00292	211FFFFF A		CI,1	-1		TEST FOR TERM. IN MTA
588	1	00293	683001B4		BCR,3	M12		
589	1	00294	33100007 A		MTW,1	7		
590	1	00295	6AF00290		BAL,15	M15B		
591				*				
592	1	00296	322C0169	M15A	LW,2	RTA,6		TEST FOR RTA TERMINATOR
593	1	00297	212FFFFF A		CI,2	-1		
594	1	00298	683001B4		BCR,3	M12		STORE RTA IN MTA
595	1	00299	352E00D6		STW,2	MTA,7		X6=X6+1
596	1	0029A	33100006 A		MTW,1	6		X7=X7+1
597	1	0029B	33100007 A		MTW,1	7		
598	1	0029C	6AF00290		BAL,15	M15B		
599				*				
600				*				
601	1	0029D	210000C4 A	M16	CI,0	X'C4'		TEST FOR D
602	1	0029E	693002AF		BCS,3	M33		
603	1	0029F	22600000 A		LI,6	0		
604	1	002A0	22700000 A	M16B	LI,7	0		
605	1	002A1	321C0169	M16C	LW,1	RTA,6		TEST FOR RTA TERM.
606	1	002A2	211FFFFF A		CI,1	-1		
607	1	002A3	683001B4		BCR,3	M12		COMPARE NAMES
608	1	002A4	311E00D6		CW,1	MTA,7		
609	1	002A5	693002AA		BCS,3	M16A		
610	1	002A6	22200000 A		LI,2	0		SET MTA = 0
611	1	002A7	352E00D6		STW,2	MTA,7		X6=X6+1
612	1	002A8	33100006 A		MTW,1	6		
613	1	002A9	6AF002AC		BAL,15	M16B		
614				*				
615	1	002AA	321E00D6	M16A	LW,1	MTA,7		TEST FOR TERMINATOR IN MTA
616	1	002AB	211FFFFF A		CI,1	-1		
617	1	002AC	683001B4		BCR,3	M12		
618	1	002AD	33100007 A		MTW,1	7		
619	1	002AE	6AF002A1		BAL,15	M16C		
620				*				
621	1	002AF	210000E2 A	M33	CI,0	X'E2'		TEST FOR S

622	1	002B0	693002C8		BCS,3	M44	
623	1	002B1	22600000 A		LI,6	0	
624	1	002B2	322C0169	M334	LW,2	RTA,6	TEST NAME
625	1	002B3	212FFFFF A		CI,2	-1	TEST FOR TERMINATOR
626	1	002B4	683001B4		BCR,3	M12	
627	1	002B5	22700000 A		LI,7	0	
628	1	002B6	323E0107	M333	LW,3	ERSUPT,7	
629	1	002B7	213FFFFF A		CI,3	-1	TEST FOR TERMINATOR
630	1	002B8	683002BF		BCR,3	M331	
631	1	002B9	312E0107		CW,2	ERSUPT,7	COMPARE NAMES
632	1	002BA	683002BD		BCR,3	M332	
633	1	002BB	20700001 A		AI,7	1	GET NEXT ENTRY
634	1	002BC	680002B6		B	M333	
635	1	002BD	20600001 A	M332	AI,6	1	GET NEXT NAME
636	1	002BE	680002B2		B	M334	
637	1	002BF	22700000 A	M331	LI,7	0	
638	1	002C0	323EC107	M335	LW,3	ERSUPT,7	
639	1	002C1	693002C4		BCS,3	*+3	
640	1	002C2	352E0107		STW,2	ERSUPT,7	
641	1	002C3	680002BD		B	M332	
642	1	002C4	213FFFFF A		CI,3	-1	
643	1	002C5	683002BD		BCR,3	M332	GET NEXT NAME
644	1	002C6	20700001 A		AI,7	1	
645	1	002C7	680002C0		B	M335	
646				*			
647	1	002C8	22600000 A	M44	LI,6	0	
648	1	002C9	322C0169	M442	LW,2	RTA,6	GET NAME
649	1	002CA	212FFFFF A		CI,2	-1	
650	1	002CB	683001B4		BCR,3	M12	TEST FOR LAST NAME
651	1	002CC	22700000 A		LI,7	0	
652	1	002CD	323E0107	M441	LW,3	ERSUPT,7	
653	1	002CE	213FFFFF A		CI,3	-1	TEST FOR LAST ENTRY
654	1	002CF	683002D7		BCR,3	M443	
655	1	002D0	312E0107		CW,2	ERSUPT,7	
656	1	002D1	693002D5		BCS,3	*+4	
657	1	002D2	22200000 A		LI,2	0	STORE ZERO IN TABLE
658	1	002D3	352E0107		STW,2	ERSUPT,7	

659	1	002D4	680002D7		B	M443	
660	1	002D5	20700001 A		AI,7	1	LOOK AT NEXT NAME
661	1	002D6	680002C0		B	M441	
662	1	002D7	20600001 A	M443	AI,6	1	GO THROUGH TABLE
663	1	002D8	680002C9		B	M442	
664				*			
665				*			
666				*			
667				*			IDENTIFY PARAMETERS
668				*			
669	1	002D9	22700000 A	M13	LI,7	0	X7=0
670	1	002DA	22600000 A	M131	LI,6	0	X6=0
671	1	002DB	33100007 A	M18	MTW,1	7	
672	1	002DC	721E0169		LB,1	RTA,7	
673	1	002DD	317000AF		CW,7	TEND	TEST FOR END INPUT
674	1	002DE	683001B4		BCR,3	M12	
675	1	002DF	21100040 A		CI,1	X'40'	TEST FOR SPACE
676	1	002E0	683002DB		BCR,3	M18	
677	1	002E1	21100036B A		CI,1	X'6B'	TEST FOR COMMA
678	1	002E2	693002E9		BCS,3	M18A	
679	1	002E3	22200040 A		LI,2	X'40'	
680	1	002E4	21600004 A	M18C	CI,6	4	TEST FOR END OF NAME
681	1	002E5	683002ED		BCR,3	M21	
682	1	002E6	752C0053		STB,2	PRID,6	
683	1	002E7	33100006 A		MTW,1	6	X6=X6+1
684	1	002E8	6AF002E4		BAL,15	M18C	
685	1	002E9	751C0053	M18A	STB,1	PRID,6	SET UP PRID
686	1	002EA	33100006 A		MTW,1	6	
687	1	002EB	21600004 A		CI,6	4	
688	1	002EC	693002DB		BCS,3	M18	
689				*			
690				*			TEST FOR PROGRAM AVAILABILITY
691				*			
692	1	002ED	22400000 A	M21	LI,4	0	
693	1	002EE	322800CB	M21B	LW,2	MTR,4	
694	1	002EF	212FFFFF A		CI,2	-1	
695	1	002FO	683002F6		BCR,3	M211	TEST FOR TERMINATOR

696	1 002F1	32140000 A		LW,1	0,2	FETCH PROGRAM ID
697	1 002F2	31100053		CW,1	PRID	TEST FOR PROGRAM ID
698	1 002F3	683002F9		BCR,3	M21A	
699	1 002F4	33100004 A		MTW,1	4	XR4=XR4+1
700	1 002F5	680002EE		B	M21B	
701			*			
702			*		REPORT ERROR	PROGRAM NOT AVAILABLE
703			*			
704	1 002F6	22100001 A	M211	LI,1	1	
705	1 002F7	6AF001C4		BAL,15	M2AAA	BRANCH TO ERROR REPORTING ROUTINE
706	1 002F8	6AF001B4		BAL,15	M12	
707			*			
708	1 002F9	20200002 A	M21A	AI,2	2	XR2= XR2 + 2
709	1 002FA	22600000 A		LI,6	0	X6=0
710	1 002FB	33100007 A	M22	MTW,1	7	
711	1 002FC	317000AF		CW,7	TEND	TEST FOR +,OR TEND
712	1 002FD	68300313		BCR,3	M22C	
713	1 002FE	721E0169		LB,1	RTA,7	
714	1 002FF	2110004E A		CI,1	X'4E'	TEST FOR +
715	1 00300	6830030A		BCR,3	M181	
716	1 00301	21100040 A		CI,1	X'40'	TEST FOR SPACE
717	1 00302	683002FB		BCR,3	M22	
718	1 00303	2110006B A		CI,1	X'6B'	TEST FOR COMMA
719	1 00304	68300319		BCR,3	M22A	
720	1 00305	2160002C A		CI,6	44	
721	1 00306	69100310		BCS,1	M22B	BRANCH IF NOT EXC. LIMIT
722			*			
723			*		REPORT ERROR	PARAMETER TABLE EXCEEDED
724			*			
725	1 00307	22100004 A	M22D	LI,1	4	ERROR 4
726	1 00308	6AF001C4		BAL,15	M2AAA	
727	1 00309	6AF001B4		BAL,15	M12	
728			*			
729	1 0030A	2160002C A	M181	CI,6	44	
730	1 0030B	683002DA		BCR,3	M131	
731	1 0030C	22100000 A		LI,1	0	
732	1 0030D	F51C0002 A		STB,1	*2,6	

733	1 0030E	33100006 A		MTW,1	6	
734	1 0030F	6800030A		B	M181	
735			*			
736	1 00310	F51C0002 A	M22B	STB,1	*2,6	STORE BYTE IN APPROPRIATE SPACE
737	1 00311	33100006 A		MTW,1	6	
738	1 00312	6AF002FB		BAL,15	M22	
739			*			
740	1 00313	2160002C A	M22C	CI,6	44	
741	1 00314	683001B4		BCR,3	M12	RETURN
742	1 00315	22100000 A		LI,1	0	
743	1 00316	F51C0002 A		STB,1	*2,6	CLEAR TABLE
744	1 00317	33100006 A		MTW,1	6	
745	1 00318	6AF00313		BAL,15	M22C	
746			*			
747			*			
748	1 00319	21600003 A	M22A	CI,6	3	TEST FOR UA
749	1 0031A	69300336		BCS,3	M22D1	
750	1 0031B	32140000 A		LW,1	0,2	LOAD UAR
751	1 0031C	25100078 A		SLS,1	-8	
752	1 0031D	22000000 A		LI,0	0	
753	1 0031E	32400001 A		LW,4	1	
754	1 0031F	4B4004EA		AND,4	L(X'FOF0F')	
755	1 00320	2540017C A		SLD,4	-4	
756	1 00321	2540007C A		SLS,4	-4	
757	1 00322	2540017C A		SLD,4	-4	ELIMINATE BITS
758	1 00323	2540007C A		SLS,4	-4	
759	1 00324	25400108 A		SLD,4	8	
760	1 00325	2500010C A		SLD,0	12	
761	1 00326	4B0000E7		AND,0	0NE	TEST FOR CHARACTERS
762	1 00327	69300329		BCS,3	M22F	
763	1 00328	20400900 A		AI,4	X'900'	
764	1 00329	22000000 A	M22F	LI,0	0	
765	1 0032A	25000108 A		SLD,0	8	
766	1 0032B	4B0000E7		AND,0	0NE	
767	1 0032C	6930032E		BCS,3	M22G	
768	1 0032D	20400090 A		AI,4	X'90'	
769	1 0032E	22000000 A	M22G	LI,0	0	

770	1	0032F	25000108	A		SLD,0	8	
771	1	00330	48C000E7			AND,0	8NE	
772	1	00331	69300333			BCS,3	M22H	
773	1	00332	20400009	A		AI,4	X'9'	
774	1	00333	35440000	A	M22H	STW,4	0,2	RESTORE UA
775	1	00334	33100006	A		MTW,1	6	X6= 4
776	1	00335	6AF002FB			BAL,15	M22	
777					*			
778	1	00336	32100006	A	M22D1	LW,1	6	
779	1	00337	4B1000B0			AND,1	THREE	TEST FOR END OF WORD
780	1	00338	683002FB			BCR,3	M22	
781	1	00339	22100000	A		LI,1	0	
782	1	0033A	F51000C2	A		STB,1	*2,6	STORE ZER0ES
783	1	0033R	33100006	A		MTW,1	6	X6=X6+1
784	1	0033C	6AF00336			BAL,15	M22D1	
785					*			
786					*			

787								PAGE
788					*			
789					*			BINARY TO DECIMAL CONVERSION ROUTINE
790					*			
791	1	0033D	35200358		BIDEC1	STW,2	BID	SAVE XR'IS
792	1	0033E	35700359			STW,7	BID+1	
793					*			
794	1	0033F	2270000A	A		LI,7	10	
795	1	00340	21300000	A		CI,3	0	TEST FOR PLUS OR MINUS
796	1	00341	68100346			BCR,1	BID1	BRANCH IF POSITIVE
797	1	00342	22200060	A		LI,2	X'60'	
798	1	00343	F5220004	A		STB,2	*4,1	
799	1	00344	3A300003	A		LCH,3	3	MAKE NUMBER POSITIVE
800	1	00345	6A200348		BID2	BAL,2	BID3	
801	1	00346	2220004E	A	BID1	LI,2	X'4E'	LOAD PLUS (+)
802	1	00347	F5220004	A		STB,2	*4,1	
803	1	00348	2010000A	A	BID3	AI,1	10	X1= X1+10
804					*			
805	1	00349	22200000	A	BID4	LI,2	0	
806	1	0034A	3620035A			OW,2	TEN	FETCH REMAINDER
807	1	0034B	7224035B			LB,2	HEXT,2	
808	1	0034C	F5220004	A		STB,2	*4,1	
809	1	0034D	33FC0001	A		MTW,15	1	X1=X1-1
810	1	0034E	64700349			BDR,7	BID4	LOOP 9 TIMES
811					*			
812	1	0034F	2010000B	A		AI,1	11	
813	1	00350	22700040	A		LI,7	X'40'	
814	1	00351	F5720004	A		STB,7	*4,1	STORE TWO SPACES
815	1	00352	33100001	A		MTW,1	1	
816	1	00353	F5720004	A		STB,7	*4,1	
817	1	00354	33100001	A		MTW,1	1	
818	1	00355	32200358			LW,2	BID	RETURN
819	1	00356	32700359			LW,7	BID+1	
820	1	00357	EAF0000F	A		BAL,15	*15	
821					*			
822	1	00358	00000000	A	BID	DATA	0,0	
	1	00359	00000000	A				

823	1	0035A	0000000A	A	TEN	DATA	10
824	1	0035B	FOF1F2F3	A	HEXT	DATA	X'FOF1F2F3',X'F4F5F6F7'
		1	0035C				
			F4F5F6F7	A			
825	1	0035D	F8F9C1C2	A		DATA	X'F8F9C1C2',X'C3C4C5C6'
		1	0035E				
			C3C4C5C6	A			

826							PAGE
827					*		
828					*		
829					*		BINARY TO HEXADECIMAL CONVERSION
830	1	0035F	35200370		BINHE1	STW,2	BINH
831	1	00360	35700371			STW,7	SAVE XR
832	1	00361	22700008	A		LI,7	BINH+1
833	1	00362	22200000	A	BINH1	LI,2	8
834	1	00363	25200104	A		SLD,2	X7= 8
835	1	00364	72240358			LB,2	X2= 0
836	1	00365	F5220004	A		STB,2	4
837	1	00366	33100001	A		MTW,1	HEXT,2
838	1	00367	64700362			BCR,7	STORE BYTE IN OUTPUT
839	1	00368	22700040	A		LI,7	*4,1
840	1	00369	F5720004	A		STB,7	1
841	1	0036A	33100001	A		MTW,1	BINH1
842	1	0036B	F5720004	A		STB,7	X'40'
843	1	0036C	33100001	A		MTW,1	1
844	1	0036D	32200370			LW,2	STORE TWO SPACES
845	1	0036E	32700371			LW,7	1
846	1	0036F	EAF0000F	A		BAL,15	BINH
847					*		BINH+1
848	1	00370	00000000	A	BINH	DATA	*15
849	1	00371	00000000	A			0,0

```

PAGE
850
851
852
853
854 1 00372 350003A5 *
855 1 00373 351003A6 *
856 1 00374 352003A7 *
857 1 00375 357003A8 *
858
859 1 00376 32700062 *
860 1 00377 33100007 A *
861 1 00378 82100007 A *
862 1 00379 22200000 A *
863 1 0037A 32040094 *
864 1 0037B 6830038A *
865 1 0037C 210FFFFFF A *
866 1 0037D 6830038C *
867 1 0037E 48000090 *
868 1 0037F 61000001 A *
869 1 00380 6930038A *
870
871
872
873 1 00381 22100002 A *
874 1 00382 487000AC *
875 1 00383 C9720062 *
876 1 00384 35700062 *
877 1 00385 320003A5 *
878 1 00386 321003A6 *
879 1 00387 322003A7 *
880 1 00388 327003A8 *
881 1 00389 0E000062 *
882
883 1 0038A 33100002 A *
884 1 0038B 6A00037A *
885
886 1 0038C 22200000 A *

```

```

32
887 1 0038D 32040094 *
888 1 0038E 68300393 *
889 1 0038F 210FFFFFF A *
890 1 00390 68300381 *
891 1 00391 33100002 A *
892 1 00392 6A00038D *
893
894 1 00393 82100001 A *
895 1 00394 491000FB *
896 1 00395 35140094 *
897 1 00396 357400A0 *
898 1 00397 22100005 A *
899 1 00398 82020007 A *
900 1 00399 22100000 A *
901 1 0039A 32740094 *
902 1 0039B CCE00007 A *
903 1 0039C 7402000E A *
904 1 0039D 68C003A2 *
905 1 0039E 35140094 *
906 1 0039F 32700062 *
907 1 003A0 22100004 A *
908 1 003A1 6A000382 *
909 1 003A2 32700062 *
910 1 003A3 22100003 A *
911 1 003A4 6A000382 *
912
913 1 003A5 00000000 A *
914 1 003A6 00000000 A *
915 1 003A7 00000000 A *
916 1 003A8 00000000 A *
917

```

```

918                                     PAGE
919                                     *
920                                     *
921                                     *
922 1 003A9 02200000 A A107 LCI 0
923 1 003AA 080003D4 PSM,0 A10SA SAVE XRS
924 1 003AB 22100000 A LI,1 0
925 1 003AC 351003D2 STW,1 A102 INIT, FIRST ENTRY
926 1 003AD 22000000 A A107 LI,13 0
927 1 003AE 22100000 A LI,1 0
928 1 003AF 6E000000 A A10,14 0 ACKN, INTERRUPT
929 1 003B0 74020000 A STCF 13,1
930 1 003B1 698003C9 BCS,8 A1011 NO ACKN,
931 1 003B2 331003D2 MTW,1 A102 SET NOT FIRST TIME
932 1 003B3 32200000 A LW,2 14
933 1 003B4 48200090 AND,2 X7FF FIND INTERRUPT ROUTINE
934 1 003B5 32320094 A103 LW,3 SI0T,1
935 1 003B6 683003BC BCR,3 A104
936 1 003B7 213FFFFF A CI,3 -1
937 1 003B8 683003CB BCR,3 A108 NO DEVICE ADDRESS
938 1 003B9 48300090 AND,3 X7FF
939 1 003BA 31300002 A CW,3 2
940 1 003BB 683003BE BCR,3 A105
941 1 003BC 33100001 A A104 MTW,1 1
942 1 003BD 6AF003B5 BAL,15 A103
943                                     *
944 1 003BF 351003D1 A105 STW,1 A106 SAVE X1
945 1 003BF 321200A0 LW,1 SI0TA,1
946 1 003C0 33400001 A MTW,4 1
947 1 003C1 82100001 A LW,1 *1 GO TO INT. ROUTINE
948 1 003C2 EAF00001 A BAL,15 *1
949 1 003C3 21DFFFFFF A CI,13 -1
950 1 003C4 693003AD BCS,3 A107 TEST FOR SETTING UA TO ZERO
951 1 003C5 321003D1 LW,1 A106
952 1 003C6 22200000 A LI,2 0
953 1 003C7 35220094 STW,2 SI0T,1
954 1 003C8 6AF003AD BAL,15 A107 BRANCH FOR ANOTHER AIO
    
```

```

955                                     *
956 1 003C9 321003D2 A1011 LW,1 A102 TEST FOR FIRST ENTRY
957 1 003CA 693003CE BCS,3 A101
958 1 003CB 331003D3 A108 MTW,1 ERINT
959 1 003CC 350001CD STW,13 M2ADX SAVE AIO STATUS
960 1 003CD 35E001CE STW,14 M2ADX+1
961 1 003CE 02200000 A A101 LCI 0
962 1 003CF 0A0003D4 PLM,0 A10SA RESTORE XRS
963 1 003D0 0E30007C LPSD,3 A10D
964                                     *
965 1 003D1 00000000 A A106 DATA 0
966 1 003D2 00000000 A A102 DATA 0
967 1 003D3 00000000 A ERINT DATA 0
968                                     *
969 1 003D4 BBUND 8
970 1 003D4 000000B9 A10SA GEN,32 SAVE=1
971 1 003D5 80108000 A GEN,1,15,1,15 1,16,1,0 PUSH AND PULL FOR XR
972 1 003D6 000000B9 GEN,32 SAVE=1
973 1 003D7 80108000 A GEN,1,15,1,15 1,16,1,0
974                                     *
    
```

```

975                                     PAGE
976                                     *
977                                     *
978                                     *
979 1 003C8 351003E8 H10 STW,1 SX13 SAVE X1
980 1 003D9 22100002 A LI,1 2
981 1 003DA 6AF003E0 BAL,15 HI01
982                                     *
983                                     *
984 1 003DB 351003E8 T10 STW,1 SX13 SAVE X1
985 1 003DC 22100001 A LI,1 1
986 1 003DD 6AF003E0 BAL,15 HI01
987                                     *
988                                     *
989 1 003DE 351003E8 TDV STW,1 SX13 SAVE X1
990 1 003DF 22100000 A LI,1 0
991 1 003E0 33100062 HI01 MTW,1 CALD1
992 1 003E1 B2E00062 LW,14 *CALD1 FETCH UA
993 1 003E2 B2E0000E A LW,14 *14 R14 = UA
994 1 003E3 670203E9 EXU HTT,1
995 1 003E4 7400000E A STCF 14
996 1 003E5 321003E8 LW,1 SX13 RESTORE X1
997 1 003E6 33100062 MTW,1 CALD1
998 1 003E7 0E000062 LPSC,0 CALD1
999                                     *
1000 1 003E8 00000000 A SX13 DATA 0
1001                                     *
1002 1 003E9 CEE0000E A HTT TDV,14 *14
1003 1 003EA CDE0000E A TI0,14 *14
1004 1 003EB CFE0000F A HI0,14 *14
1005                                     *
1006                                     *
    
```

```

1007                                     PAGE
1008                                     *
1009                                     *
1010                                     *
1011                                     *
1012 1 003EC 221FFFFFF A M2B LI,1 -1
1013 1 003ED 22200000 A LI,2 0 B31+30 = X140000000
1014 1 003EE 4BEC0105 AND,14 B31+30
1015 1 003EF 683003F4 BCR,3 M2B1 TEST FOR UNUSUAL END
1016 1 003F0 35E0006E STW,14 TYPERR
1017 1 003F1 22DFFFFFF A M2B3 LI,13 -1 SET UP ERROR CONDITION
1018 1 003F2 35D000AD STW,13 TYPEC
1019 1 003F3 EAF000CF A BAL,15 *15
1020 1 003F4 3520006E STW,2 TYPERR
1021 1 003F5 35E000CA STW,14 TIR TIR = TYPE INT. RESPONSE
1022 1 003F6 CD900057 TI0,9 *TYPE
1023 1 003F7 223000C8 A LI,3 200
1024 1 003F8 4B900090 AND,9 X7FF FIND LAST BYTE READ
1025 1 003F9 3B300009 A SW,3 9
1026 1 003FA 693003FC BCS,3 M2B2
1027 1 003FB 35300169 STW,3 RTA TEST FOR NB BYTES READ
1028                                     *
1029 1 003FC 353000AF M2B2 STW,3 TEND
1030 1 003FD 33F000AF MTW,15 TEND
1031 1 003FE 6A3003F1 BAL,0 M2B3
1032 1 003FF 00000000 A ERRT DATA 0
1033 1 00400 E2C5D9D9 A SERR TEXT 'SERR'
1034 1 00401 D7C5D9D9 A PERR TEXT 'PERR'
    
```

			PAGE		TYPE OUT PROFILE
1034					
1035					
1036				*	
1037	1 00402	3510044F	PR0FILE	STW,1	PRS1 SAVE XR1,XR2
1038	1 00403	35200450		STW,2	PRS2
1039	1 00404	22100033 A		LI,1	PR0E-PR0FT=3
1040	1 00405	32220417		LW,2	PR0FT-1,1
1041	1 00406	3522041A		STW,2	PR0FT+2,1
1042	1 00407	64100405		BDR,1	*=2
1043	1 00408	33100062		MTW,1	CALD1
1044	1 00409	B2200062		LW,2	*CALD1
1045	1 0040A	32240000 A		LW,2	0,2
1046	1 0040B	35200418		STW,2	PR0FT
1047	1 0040C	33100062		MTW,1	CALD1
1048	1 0040D	B2200062		LW,2	*CALD1
1049	1 0040E	43200451		AND,2	MAS
1050	1 0040F	49200452		OR,2	MAS1
1051	1 00410	35200419		STW,2	PR0FT+1
1052	1 00411	33100062		MTW,1	CALD1
1053	1 00412	3210044F		LW,1	PRS1
1054	1 00413	32200450		LW,2	PRS2
1055	1 00414	0E000062		LPSD,0	CALD1
1056				*	
1057				*	
1058	1 00415	15C7D9D6 A	PR0FT1	DATA	X'15D7D9D6'
1059	1 00416	C6C9D3C5 A		TEXT	'FILE : '
	1 00417	407A4040 A			
1060	1 00418	00000036	PR0FT	EGU	\$
1061				D0	54
1062	1 00418	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00419	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0041A	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0041B	405C5C40 A		TEXT	' ** '
1063				FIN	

1062	1 0041C	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0041D	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0041E	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0041F	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00420	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00421	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00422	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00423	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00424	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00425	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00426	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00427	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00428	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00429	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0042A	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0042B	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0042C	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0042D	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0042E	405C5C40 A		TEXT	' ** '

1063				FIN	
1062	1 0042F	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00430	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00431	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00432	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00433	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00434	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00435	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00436	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00437	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00438	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00439	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0043A	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0043B	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0043C	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0043D	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0043E	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0043F	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00440	405C5C40 A		TEXT	' ** '
1063				FIN	

1062	1 00441	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00442	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00443	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00444	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00445	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00446	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00447	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00448	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 00449	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0044A	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0044B	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0044C	405C5C40 A		TEXT	' ** '
1063				FIN	
1062	1 0044D	405C5C40 A		TEXT	' ** '
1063				FIN	
1064	1 0044E	FFFFFFFF A	PR0E	DATA	=1
1065	1 0042A		PR0FT2	EQU	PR0FT+18
1066	1 0044F	00000000 A	PRS1	DATA	0
1067	1 00450	00000000 A	PRS2	DATA	0
1068	1 00451	00FFFFFF A	MAS	DATA	X'0FFFFFFF'
1069	1 00452	7E000000 A	MAS1	DATA	X'7E000000'
1070	1 00453	15404040 A	BLANKSP	DATA	X'15404040',X'40404040',X'40404040'
	1 00454	40404040 A			
	1 00455	40404040 A			
1071	1 00456	D7D9D6C6 A	PR0C8	DATA	X'07D9D6C6'
1072	1 00457	D5D7D9D6 A	NPR0C8	TEXT	'NPR0'

1073	1 00458			BBOUND	8	
1074	1 00458	05001054	PR0D	GEN,8,24	X'05',BA	(PR0FT1)
1075	1 00459	8C000054	A	GEN,8,24	X'8C',84	
1076	1 0045A	0000114C		GEN,8,24	X'00',BA	(BLANKSP)
1077	1 0045B	8C00000C	A	GEN,8,24	X'8C',12	
1078	1 0045C	000010A8		GEN,8,24	X'00',BA	(PR0FT2)
1079	1 0045D	8C000048	A	GEN,8,24	X'8C',72	
1080	1 0045E	0000114C		GEN,8,24	X'00',BA	(BLANKSP)
1081	1 0045F	8C00000C	A	GEN,8,24	X'8C',12	
1082	1 00460	000010F0		GEN,8,24	X'00',BA	(PR0FT2+18)
1083	1 00461	1C000048	A	GEN,8,24	X'1C',72	

1084				PAGE		
1085			*			PRINT PROFILE TABLE
1086			*			
1087	1 00462	04100000	A	PR0FP	CAL1,1	0
1088	1 00463	00000057		GEN,32	TYPE	UNIT ADDR.
1089	1 00464	00000462		GEN,32	PR0FP	BUSY RT
1090	1 00465	00000469		GEN,32	PR0FRET	N0T BUSY RETURN
1091	1 00466	00000462		GEN,32	PR0FP	N0T ACC. RT
1092	1 00467	0000046A		GEN,32	PR0FI	INTERRUPT
1093	1 00468	0000022C		GEN,32	DA(PR0D)	COMM. DOUBLE WORD ADDR.
1094			*			
1095	1 00469	680E0000	A	PR0FRET	B	0,7
1096			*			
1097	1 0046A	220FFFFF	A	PR0FI	LI,13	-1
1098	1 0046B	3500046D		STW,13	PR0FEND	
1099	1 0046C	E8C0000F	A	B		*15
1100			*			
1101	1 0046D	00000000	A	PR0FEND	DATA	0
1102	1 0046E	00000000	A	PR0FID	DATA	0
1103			*			

```

1104                                     PAGE
1105
1106                                     *
1107                                     *
1107 1 0046F 351000B3 ERROR1 STW,1 ERROR ROUTINE
1108 1 00470 350000B2          STW,0 SX1
1109 1 00471 6A1004B9          BAL,1 SX0 SAVE XR
1110                                     *
1111                                     *
1112                                     *
1113                                     *
1114 1 00472 3220005A          LW,2 TYPER TEST FOR TYPER BUSY
1115 1 00473 68300477          BCR,3 ERR01
1116 1 00474 6A1004D1          BAL,1 ERR011. RESTORE XR
1117 1 00475 B21000B3          LW,1 *SX1
1118 1 00476 EA100001 A          BAL,1 *1 RETURN TO BUSY ADDR.
1119
1120 1 00477 32200091          ERR01 LW,2 ER
1121 1 00478 69300474          BCS,3 ERR03 TEST FOR ERROR ROUTINE BUSY
1122
1123 1 00479 321003FF          LW,1 ERRT TEST FOR NO ERROR TYPEOUT
1124 1 0047A 693004CD          BCS,3 ERR019A
1125 1 0047B 27100004 A          LI,1 4
1126 1 0047C B20200B3          LW,0 *SX1,1 TEST FOR SPPR. ERROR TYPEOUT
1127 1 0047D 22100000 A          LI,1 0
1128 1 0047E 222FFFFF A          LI,2 -1
1129 1 0047F 31020107          CW,0 ERSUPT,1
1130 1 00480 683004CD          BCR,3 ERR019A
1131 1 00481 31220107          CW,2 ERSUPT,1 TEST FOR END OF TABLE
1132 1 00482 68300485          BCR,3 *+3
1133 1 00483 22100001 A          AI,1 1
1134 1 00484 6800047F          B *+5
1135 1 00485 04300000 A          CAL1,3 0
1136 1 00486 00000057          GEN,32 TYPE
1137 1 00487 48E00105          AND,14 B31+30 TEST FOR BUSY TY
1138 1 00488 69300474          BCS,3 ERR03
1139 1 00489 22200004 A          LI,2 4
1140 1 0048A 321404E3          LW,1 BUTP=1,2 STORE IN OUTPUT
    
```

```

1141 1 0048B 35140168          STW,1 RTA=1,2
1142 1 0048C 6420048A          BDR,2 ERR04
1143 1 0048D 22200004 A          LI,2 4
1144 1 0048E B27400B3          LW,7 *SX1,2 X7= NO. OF PROGRAM
1145 1 0048F 3570016B          STW,7 RTA+2
1146 1 00490 22200002 A          LI,2 2
1147 1 00491 B28400B3          LW,8 *SX1,2 X7 = NO. OF WORDS
1148 1 00492 22A00000 A          LI,10 0
1149 1 00493 22100010 A          LI,1 16 X1=16
1150 1 00494 22200003 A          LI,2 3
1151 1 00495 B29400B3          LW,9 *SX1,2 FETCH WORD ID
1152 1 00496 3240006F          LW,4 RTAA
1153 1 00497 22200005 A          LI,2 5
1154 1 00498 B27400B3          LW,7 *SX1,2 FETCH ADDR. OF ERROR OUTPUT
1155 1 00499 323E0000 A          LW,3 0,7
1156 1 0049A 32900009 A          LW,9 9
1157 1 0049B 6910049E          BCS,1 ERR05 BRANCH IF R9= 1
1158 1 0049C EAF0009F          BAL,15 *BINHE HEX CONVERSION
1159 1 0049D 6AF0049F          BAL,15 ERR06
1160
1161 1 0049E EAF00093          ERR05 BAL,15 *BIDEC DEC CONVERSION
1162 1 0049F 25900001 A          ERR06 SLS,9 1
1163 1 004A0 33100007 A          MTW,1 7 X7=X7+1
1164 1 004A1 21A00005 A          CI,10 5
1165 1 004A2 693004AC          BCS,3 ERR07 TEST FOR WORDS ON ONE LINE
1166 1 004A3 22A00010 A          LI,10 16 X10 = 16
1167 1 004A4 22F00C15 A          LI,15 X'15' STORE CARRIAGE RETURN
1168 1 004A5 75F20169          STB,15 RTA,1
1169 1 004A6 22F00040 A          LI,15 X'140' SPACE
1170 1 004A7 33100001 A          MTW,1 1
1171 1 004A8 75F20169          STB,15 RTA,1
1172 1 004A9 33100001 A          MTW,1 1 STORE SPACES
1173 1 004AA 64A004A8          BDR,10 ERR08
1174 1 004AB 6AF004AD          BAL,15 ERR07+1
1175 1 004AC 3310000A A          MTW,1 10
1176 1 004AD 64800499          BDR,8 ERR030 LOOP FOR MORE WORDS
1177 1 004AE 328004DF          LW,8 ERR014+1
    
```

1178	1	004AF	488004EB		AND,8	L(X'FF000000')
1179	1	004B0	49800001 A		OR,8	1
1180	1	004B1	358004DF		STW,8	ERR014+1
1181				*		
1182	1	004B2	04100000 A	ERR010	CAL,1	0
1183	1	004B3	00000057		GEN,32	TYPE
1184	1	004B4	000004B2		GEN,32	ERR010
1185	1	004B5	000004C6		GEN,32	ERR019
1186	1	004B6	000004DC		GEN,32	ERR012
1187	1	004B7	000004E0		GEN,32	ERR013
1188	1	004B8	0000026F		GEN,32	DA(ERR014)
1189				*		
1190	1	004B9	352000B4	SAVER	STW,2	SX2
1191	1	004BA	353000B6		STW,3	SX3
1192	1	004BB	354000B5		STW,4	SX4
1193	1	004BC	357000B7		STW,7	SX7
1194	1	004BD	358004C3		STW,8	SX8
1195	1	004BE	359004C4		STW,9	SX9
1196	1	004BF	35E000B8		STW,14	SX14
1197	1	004C0	35F000B9		STW,15	SX15
1198	1	004C1	35A004C5		STW,10	SX10
1199	1	004C2	EA100001 A		BAL,1	*1
1200				*		
1201	1	004C3	00000000 A	SX8	DATA	0
1202	1	004C4	00000000 A	SX9	DATA	0
1203	1	004C5	00000000 A	SX10	DATA	0
1204				*		
1205	1	004C6	3210046E	ERR019	LW,1	PR0FID
1206	1	004C7	683004CD		BCR,3	ERR019A
1207	1	004C8	22100000 A		LI,1	0
1208	1	004C9	3510046D		STW,1	PR0FEND
1209	1	004CA	6A700462		BAL,7	PR0FF
1210	1	004CB	3210046D		LW,1	PR0FEND
1211	1	004CC	683004CB		BCR,3	*=1
1212	1	004CD	6A1004D1	ERR019A	BAL,1	ERR011
1213	1	004CE	331000B3		MTW,1	SX1
1214	1	004CF	B21000B3		LW,1	*SX1

CALLING SEQUENCE FOR SID
NOT BUSY RETURN
NOT ACCEPTED RETURN.
INT.ADDR,

TEST FOR PR0FID PRINT
PRINT PR0FID

1215	1	004D0	EA100001 A		BAL,1	*1
1216	1	004D1	320000B2	ERR011	LW,0	SX0
1217	1	004D2	322000B4		LW,2	SX2
1218	1	004D3	323000B6		LW,3	SX3
1219	1	004D4	324000B5		LW,4	SX4
1220	1	004D5	327000B7		LW,7	SX7
1221	1	004D6	328004C3		LW,8	SX8
1222	1	004D7	329004C4		LW,9	SX9
1223	1	004D8	32A004C5		LW,10	SX10
1224	1	004D9	32E000B8		LW,14	SX14
1225	1	004DA	32F000B9		LW,15	SX15
1226	1	004DB	EA100001 A		BAL,1	*1
1227				*		
1228	1	004DC	6A1004DC	ERR012	BAL,1	*
1229				*		
1230	1	004DE			B0UND	8
1231	1	004DE	010005A4	ERR014	GEN,8,24	1,BA(RTA)
1232	1	004DF	4C000000 A		GEN,8,24	X'4C',0
1233				*		
1234				*		
1235	1	004E0	22100000 A	ERR013	LI,1	0
1236	1	004E1	35100091		STW,1	ER
1237	1	004E2	22DFFFFFF A		LI,13	=1
1238	1	004E3	EAF0000F A		BAL,15	*15
1239				*		
1240	1	004E4	155CC5D9 A	0UTP	DATA	X'155CC5D9',X'D9D6D940',0,X'40404040'
	1	004E5	D9D6D940 A			
	1	004E6	00000000 A			
	1	004E7	40404040 A			
1241	1	004E8	000004EC	PR0L	DATA	L(X'ABCDEF12')
1242				*		
1243		1	00173	END	M1AA	
	1	004E9	FFFE0000 A			
	1	004EA	000F0F0F A			
	1	004EB	FF000000 A			
	1	004EC	ABCDEF12 A			

RETURN