

DIAGNOSTIC PROGRAM MANUAL
SIGMA 5
CPU DIAGNOSTIC PROGRAM
(SUFFIX)

PROGRAM NO. 704174B

October 1968

This publication supersedes SDS 901519A
dated February 1968

LIST OF EFFECTIVE PAGES

Total number of pages is 64, as follows:

Page No.	Issue	Page No.	Issue
Title.....	Original		
A.....	Original		
i thru ii	Original		
1-1 thru 1-2	Original		
2-1 thru 2-4	Original		
3-1 thru 3-6	Original		
4-1 thru 4-48.....	Original		

CONTENTS

Section	Title	Page
I	INTRODUCTION	1-1
	1-1 Scope of Manual	1-1
	1-2 Program Objectives	1-1
	1-3 General Specifications	1-1
II	OPERATING INSTRUCTIONS	2-1
	2-1 General	2-1
	2-2 Loading Procedure	2-1
	2-3 Program Loader	2-1
	2-4 Operating Procedures	2-1
	2-5 Success Indications	2-1
	2-6 Error Indications	2-1
	2-7 Options	2-3
	2-8 Sense Switches	2-3
	2-9 Control Panel Interrupt	2-3
	2-10 Test Selection	2-4
	2-11 Restarting the Program	2-4
III	PROGRAM DESCRIPTION	3-1
	3-1 General	3-1
	3-2 Test Module	3-1
	3-3 Flow Chart	3-1
IV	PROGRAM LISTING	4-1

RELATED PUBLICATIONS

The following publications contain information, supplementary to but not required, for a complete understanding of the Sigma 5 CPU Diagnostic Suffix Program.

<u>Publication Title</u>	<u>Publication No.</u>
Sigma 5 Computer, Reference Manual	900959
Sigma 5 Computer, Technical Manual	901172
Sigma Symbol and Meta-Symbol, Reference Manual	900952
Sigma 5/7 CPU Format Converter/CPU Loader Documentation, Reference Manual	901584*

*Not yet released

SECTION I
INTRODUCTION

1-1 SCOPE OF MANUAL

This manual describes the suffix program designed for the Sigma 5 Computer manufactured by Scientific Data Systems, Santa Monica, California.

This manual is made up of four sections. Section I is a general introduction to the suffix program. Section II contains a discussion of operating procedures. Section III is comprised of a detailed description of the program operation. Section IV contains the program's complete symbolic listing as generated by the Sigma metasymbol assembler.

1-2 PROGRAM OBJECTIVES

The purpose of the suffix program is to detect and diagnose malfunctions of the Sigma 5 CPU pertaining to multiple operand and push-down stack instructions. Specifically, the instruction set tested by suffix consists of the following:

LM	Load Multiple
STM	Store Multiple
MSP	Modify Stack Pointer
PSW	Push Word
PLW	Pull Word
PSM	Push Multiple
PLM	Pull Multiple
MMC	Move to Memory Control

Note that the MMC instruction is tested only to the extent that it modifies registers correctly. The memory protect diagnostic test verifies that it loads the write protection locks.

1-3 GENERAL SPECIFICATIONS

Table 1-1 lists the general specifications for this program.

Table 1-1. General Specifications

Computer configuration	Any Sigma 5 computer with card reader or paper tape reader for program input
Memory size	8K minimum (8192 words)
Optional equipment	Keyboard printer or line printer for printed output

Table 1-2 shows the testing that must have been successfully completed before the suffix program is run. Also listed are the testing prerequisites for the other Sigma 5 CPU diagnostic programs.

Table 1-2. Testing Prerequisites

Program	Prerequisite Program
Verify	None
Pattern	Verify
Auto	Verify, Pattern*
Suffix	Auto
Float	Auto
Interrupt	Auto
Memory protect	Suffix

*For the auto test to run, the block 0 register must be functioning correctly, as tested by the pattern program.

SECTION II
OPERATING INSTRUCTIONS

2-1 GENERAL

The suffix diagnostic program employs a data-gathering technique for its operation. The program consists of a driver or control section followed by a number of test modules. Each module contains eight or twelve words of data prescribing a test to be performed. The driver program accesses each module in sequence, sets up the prescribed conditions, executes the specified instruction, and then tests the results for possible errors. A report of each test or error may be printed out or displayed on the control panel indicators.

2-2 LOADING PROCEDURE

Table 2-1 shows the control panel switch settings to be used for loading the program. After the switches have been set up as indicated, the following procedure is required:

- a. Clear the memory.
- b. Perform the standard load procedure described in Section 5 of the Sigma 5 Computer reference manual.

Table 2-1. Switch Settings for Program Loading

Switch	Setting
CONTROL MODE	LOCAL
WATCHDOG TIMER	NORMAL
INTERLEAVE SELECT	NORMAL
PARITY ERROR MODE	CONT
AUDIO	ON
CLOCK MODE	CONT
ADDR STOP	Off
SENSE Switches	0

If the program is loaded with the switches set according to table 2-1, it will automatically branch to the starting location and begin running.

If SS1 is set to 1 when the program is loaded, a wait will occur at X'100' and the count pulse interrupts will not be armed (see R5 in table 2-4). To continue, set the COMPUTE switch to IDLE and then back to RUN.

2-3 PROGRAM LOADER

The suffix diagnostic program uses the dual loader described in appendix A. This program loader allows two modes of operation when furnished on punched cards to provide the best possible chance of a successful load.

2-4 OPERATING PROCEDURES

2-5 SUCCESS INDICATIONS

Provided that no errors occur, the program will run continuously through all test modules. After completing the last module it will start over, making another pass. A pass counter and error counter are maintained by the program. These may be examined by setting SENSE switch 3.

After completing each test module, the program reads SENSE switch 3 and makes a report if the switch is set. Reports are normally made through the keyboard printer, device address 001. However, if no device responds to a TIO to that address, the program halts at location X'1F6' with the report information contained in registers 1 through 4. The registers contain the following information:

- R1 The memory address of most recently completed test module
- R2 The error count
- R3 The pass count (bits 0-15), module count (bits 16-31)
- R4 The instruction tested

Figure 2-1 shows a typical printout from the keyboard printer resulting from setting SS3 after the program has made a number of successful passes.

2-6 ERROR INDICATIONS

When running with the sense switches on 0, the program will halt upon detecting an error. Before halting, however, the error will be reported via the keyboard printer. If the printing device does not respond to a TIO, the program will merely halt at location X'211' with the ALARM on. During the error halt, registers R1 through R8 contain the information shown in figure 2-2.

LIST	ERRORS	PASSES	SUFFIX ERROR DISPLAY INST IDENTIFIER	IS	SHOULD BE	DIFF
000063A	00000000	00280037	0A0002F6			
0000646	00000000	00280038	0A0002F6			
0000652	00000000	00280039	0A0002F6			
000065E	00000000	0028003A	0A0002F6			
000066A	00000000	0028003B	0A8E02F4			

901519A.201

Figure 2-1. Sample of Printout, No Errors

R1	CURRENT MODULE ADDRESS
R2	ERROR COUNTER
R3	PASS COUNTER (PASSES IN BITS 0-15, MODULES IN BITS 16-31)
R4	INSTRUCTION UNDER TEST
R5	ERROR IDENTIFIER AND ADDRESS: 10000000 = INSTRUCTION 20000000 = LOCATION+1 OF THE EXECUTION LOCATION 3000WXYZ = INDIRECT ADDRESS LOCATION 5000000X = PROGRAM STATUS WORD X; X=1 OR 2 6000000X = REGISTER X; X=0 THRU F 7000WXYZ = MEMORY WORD IN LOCATION WXYZ; WXYZ=0000 THRU FFFF 8000WXYZ = STACK POINTER DOUBLEWORD LOCATIONS
R6	ERRONEOUS RESULT (IS)
R7	PREDETERMINED RESULT (SHOULD BE)
R8	DIFFERENCE BETWEEN R6 AND R7 (RESULT OF EXCLUSIVE-OR OF R6 WITH R7)

901519A.202

Figure 2-2. Error Halt Information, Registers R1 Through R-8

A typical error printout is shown in figure 2-3.

The halt-on-error feature may be disabled by setting SENSE switch 4. However, if the machine is operating in this mode and no printout device is available, there will be no error indications except for brief flickers of the ALARM indicator (which may be too fast to see when only one or a few modules are failing).

The error indications described in the previous paragraphs occur only for faults that result from executing the test instruction. If a trap or spurious interrupt occurs at any other time during the operation of the control program, a WAIT is executed with the program halting at location X'B6'. If the wait is cleared, the program will attempt to reinitialize and resume testing with the current test module.

If a memory parity error occurs, the program is interrupted to location X'56'. The interrupt routine reads the memory

fault indicators and leaves the result in register 4. A WAIT is then executed, causing the program to halt at location X'C0'. If the wait is cleared, the program will attempt to reinitialize and resume testing with the current test module.

Table 2-2 summarizes the various halt locations in the program.

Table 2-2. Halt Locations

Location (Hex)	Reason for Halting
B6	Erroneous trap or interrupt
C0	Memory fault interrupt
CC	Control panel interrupt
100	SSI set to 1 when loading
1F6	Report halt
211	Error halt

LIST	ERRORS	PASSES	SUFFIX ERROR DISPLAY INST IDENTIFIER	IS	SHOULD BE	DIFF
0000040E	00000001	000A0005	2B0002E6 700002E6	00000000	000C0804	000C0804

901519A.203

Figure 2-3. Sample of Error Printout

2-7 OPTIONS

Several optional features are incorporated into the suffix program to give the operator a more flexible tool for diagnosing failures while providing a quick means of detecting faults with a minimum of operator intervention.

2-8 Sense Switches

The uses of SS3 and SS4 have already been mentioned. Further control is provided through SS1 and SS2, which allow the operator to repeatedly loop on a single test. Table 2-3 summarizes the functions of all four switches.

Table 2-3. Sense Switch Functions

Switch	Function
SS1	<u>Short Loop.</u> When SS1 is set, the program continuously repeats the same test module. A minimum of instructions are executed to set up the necessary register and memory areas. No testing of results or other sense switches takes place. If SS1 is set to 1 while the program is being loaded, the count pulse interrupts will not be automatically armed on the 100th successful pass as is usually done
SS2	<u>Long Loop.</u> When SS2 is set and SS1 reset, the program repeats the same module. All testing of results takes place and other sense switches are read
SS3	<u>Report.</u> When SS3 is set, the program reports at the completion of each test not otherwise reported as the result of an error
SS4	<u>Suppress Error Halt.</u> When SS4 is set, the program will not halt on errors. Errors will still be reported via the keyboard printer, if available. Printing may be suppressed by turning the device off

2-9 Control Panel Interrupt

Pressing the control panel INTERRUPT button at any time when the program is running allows the operator to conveniently change certain parameters in the program. When the INTERRUPT button is pressed, the computer comes to a wait with address X'CC' in the instruction address register. Changes may be made by entering information into any of the registers described in table 2-4, then reading out the instruction from the instruction address location and returning the COMPUTE switch to RUN.

Table 2-4. Register Contents at Time of Interrupt

Register	Contents
R0	The IOP device address of the unit used for report or error messages. The address is in bits 16-31 and is initially set to 1 when the program is loaded <u>Device Selection.</u> The program tests bits 0-15 to determine whether to use the keyboard printer or the line printer output routines. These bits are initially set to zero indicating that the output device is the keyboard printer. Setting a nonzero value in bits 0-15 and changing the address in bits 16-31 causes all messages to be put out on the line printer
R1	The memory address of the current or most recently completed test module <u>Test Selection.</u> When the machine is returned to RUN the program will begin testing with the module addressed by R1. The operator may insert any valid module address he wishes and then set SS1 or SS2 to loop on that module
R5	Register bits 16-19, used to arm and enable the count pulse interrupts. <u>Count Pulse Interrupt Level Selection.</u> Normally, if no errors have occurred, the program automatically arms and enables the four count pulse interrupt levels

(Continued)

Table 2-4. Register Contents at Time of Interrupt (Cont.)

Register	Contents
R5 (Cont.)	<p>upon making the one hundredth pass. The operator may suppress this feature by clearing R5 when the wait occurs for the control panel interrupt, or by loading the program with SS1 set. He may selectively disable only some of the count pulse interrupts if he wishes. He also has the option of later re-enabling the levels by again interrupting and changing R5</p> <p>Bits other than 16-19 of R5 have no effect on interrupt</p>

2-10 TEST SELECTION

The operator may select any test module to begin running by using the control panel INTERRUPT as described in

paragraph 2-9 and table 2-4. He may likewise loop on a selected test by setting SENSE switch 1 or 2 before running the test again.

The operator may also loop on a particular test by setting SS1 or SS2 when the test is being reported either by the keyboard printer or by a report halt or error halt.

2-11 RESTARTING THE PROGRAM

The program may be restarted from location X'100'. When loaded, the program inserts an unconditional branch to that location in address X'26' so that it can normally be restarted by pressing the CPU RESET button and putting the COMPUTE switch to RUN.

SECTION III
PROGRAM DESCRIPTION

3-1 GENERAL

A simplified flow chart is given in figure 3-1 to illustrate the general philosophy of the program's operation. When loaded, the suffix program automatically branches to its starting location and begins running. Initialization takes place where parameters are set up and a branch instruction is inserted in location X'26' to facilitate restarting if the CPU RESET button should be depressed.

The program accesses each test module in turn and sets up all the test conditions as prescribed by the module. The instruction contained in the module is executed and then SENSE switch 1 is tested to determine whether the short loop mode is in effect. If SS1 is set, the program repeats a minimum of the setup procedure for the same test module and again executes the instruction. No results are examined in the short-loop mode.

If not operating in the short loop (that is, if SS1 is reset), the program proceeds to compare the results of the test with the expected results. The resultant contents of all registers and memory operands (and stack pointer doubleword, if applicable) are prescribed by the test module. Other items are also tested, such as the instruction location, the location following the instruction, the indirect address location, and the program status doubleword.

When an error is detected or if SS3 is set, the program makes a report - normally through the keyboard printer. The reports may be switched over to a line printer, if desired (see Control Panel Interrupt in section II). When no printing device is available, the program makes its reports by executing a wait with the information stored in general registers 1 through 8.

After results have been checked and reports, if any, completed, SENSE switch 2 is read. If SS2 is set to 1, the program goes through the entire setup, test, and report procedure for the same test module that it just finished. If SS2 is reset to 0, then the module pointer is updated so that a new test is performed on the next cycle.

3-2 TEST MODULE

Each test performed by the suffix diagnostic program requires a test module in the following format:

Word 0 A negative count indicating the number of words in the module, including the count word. If word 0 has a value of zero, it indicates the end of the module list

Word 1 The instruction to be tested

Word 2 Bits 0-11. These bits of the PSW1 are set up prior to executing the test instruction. The remainder of PSW1 is automatically set up by the program.
Bits 12-31. This is a linkage address given to the program enabling it to set up for expected traps

Word 3 The value of PSW1 expected to be found in the location labeled RETURN after the test instruction has been executed (plus the subsequent XPSD)

Word 4 A pointer which enables the program to access the appropriate table and load data into the registers before executing the test instruction

Bits 0-15: Address of beginning of table

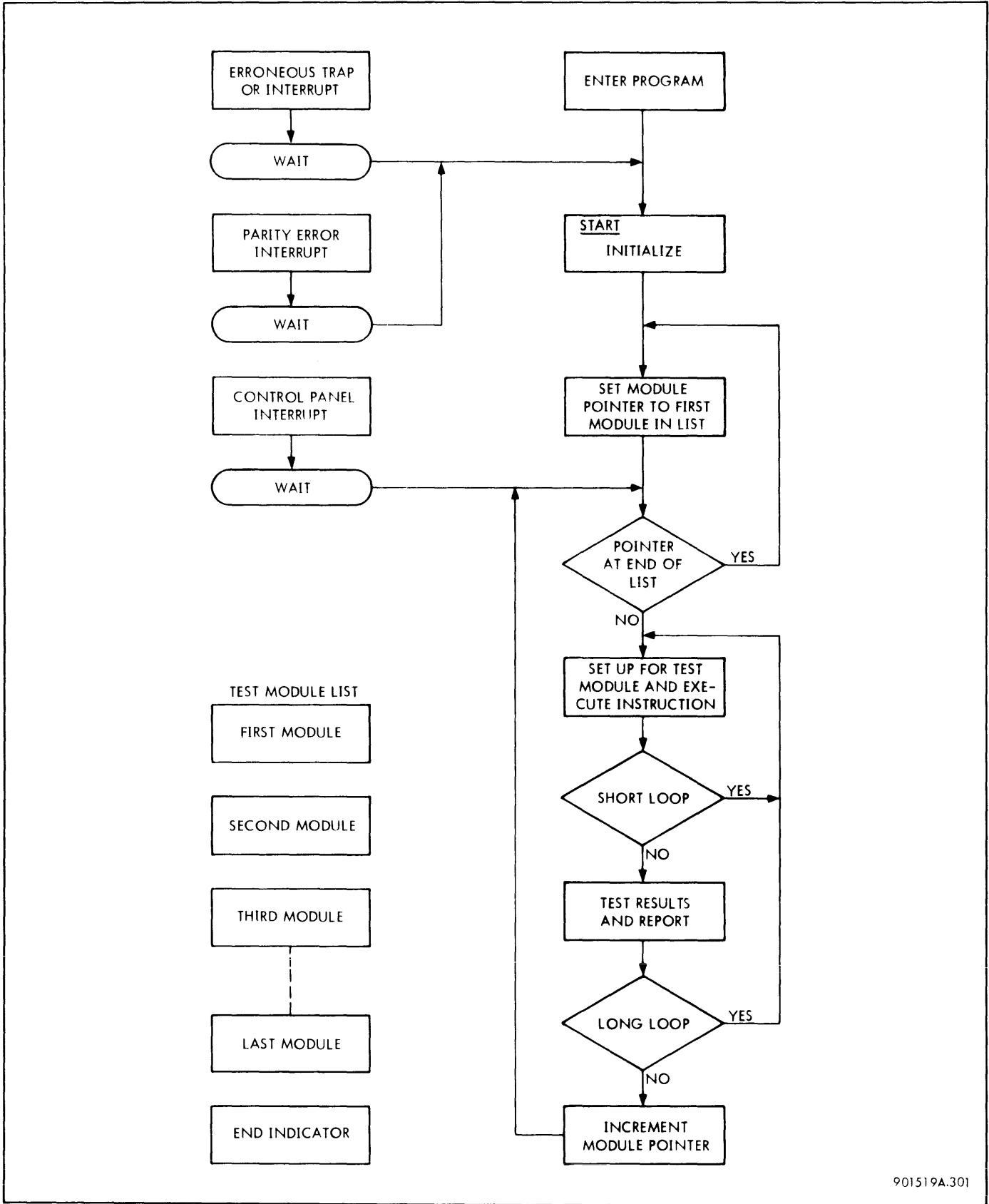
Bits 16-23: Negative count of the number of registers to be loaded

Bits 24-31: First register to be loaded

The pointer causes from 0 to 16 registers to be loaded. This pointer, as well as the pointers in words 5, 6, and 7 of the module, treat the load area as a circular set of 16 locations. Thus, if the pointer specifies that four locations are to be loaded beginning with the last location (in this case register 15), then the program loads location 15 from the first word of the data table and then loads the next three words of the data table into locations 0, 1, and 2 (registers 0, 1, and 2)

Word 5 A pointer having the same format as word 4, but used to specify the expected contents of registers after the test instruction is executed

Word 6 A pointer having the same format as word 4 but used to specify up to 16 memory operands. As with word 4, the load area is treated as a circular set of 16 locations



901519A.301

Figure 3-1. Sigma 5 Suffix Program, Simplified Flow Chart

- Words 7 A pointer having the same format as word 4 but used to specify expected memory results of the test

- Words 8, 9 The initial stack pointer doubleword for push-down instructions

- Words 10, 11 The final stack pointer doubleword expected after a push-down instruction test

Not all words are required for every module. In the suffix programs test modules consist of words 0 through 5, words 0 through 7, or words 0 through 11. All data areas not specifically called out by the module are cleared to zero.

3-3 FLOW CHART

A detailed flow diagram of the entire program (excluding test modules) is given in figure 3-2. A study of figure 3-2 along with the program listing in section IV should provide the user with a clear understanding of the suffix program's operation.

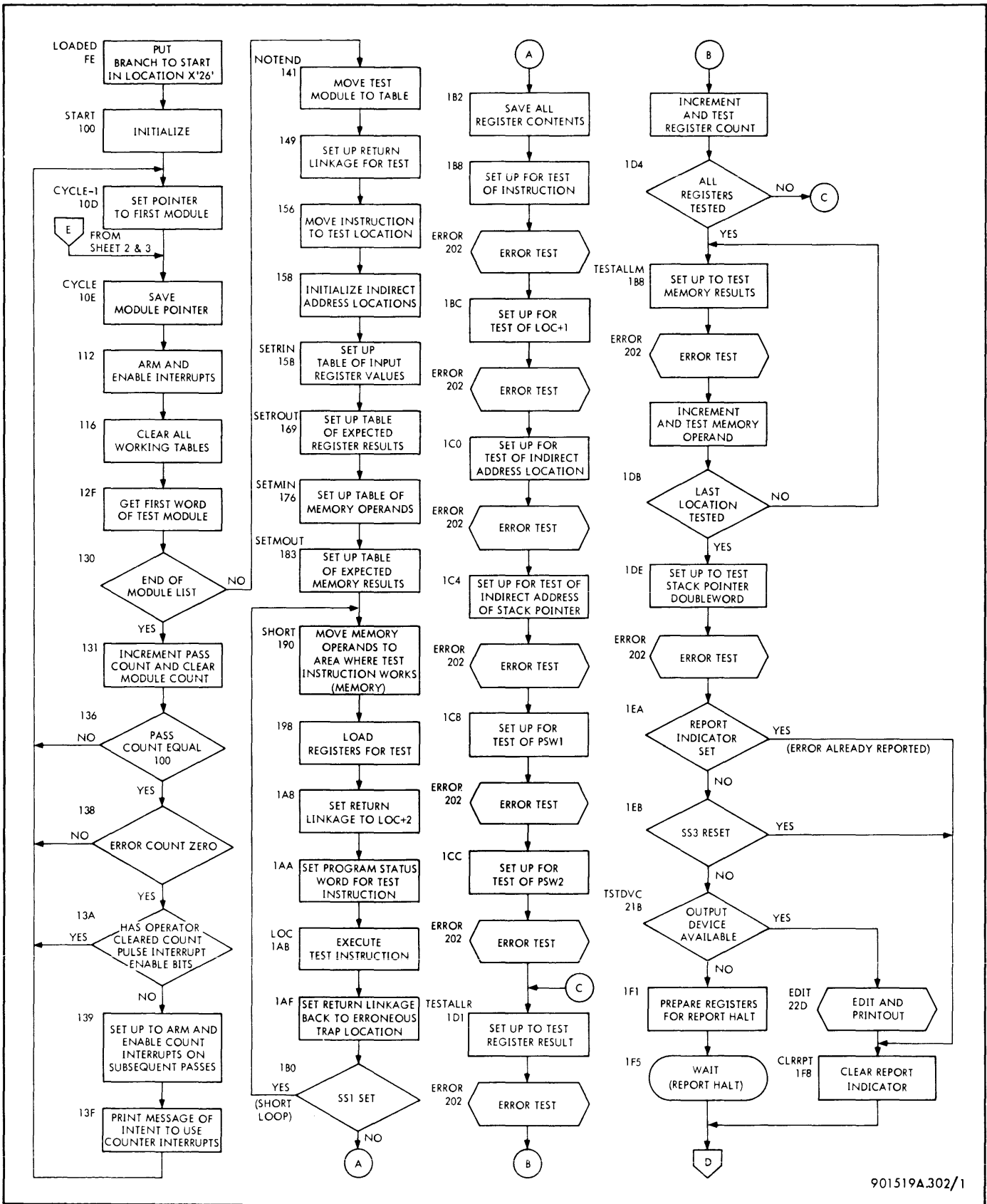


Figure 3-2. Sigma 5 Suffix Program, Detailed Flow Chart (Sheet 1 of 3 sheets)

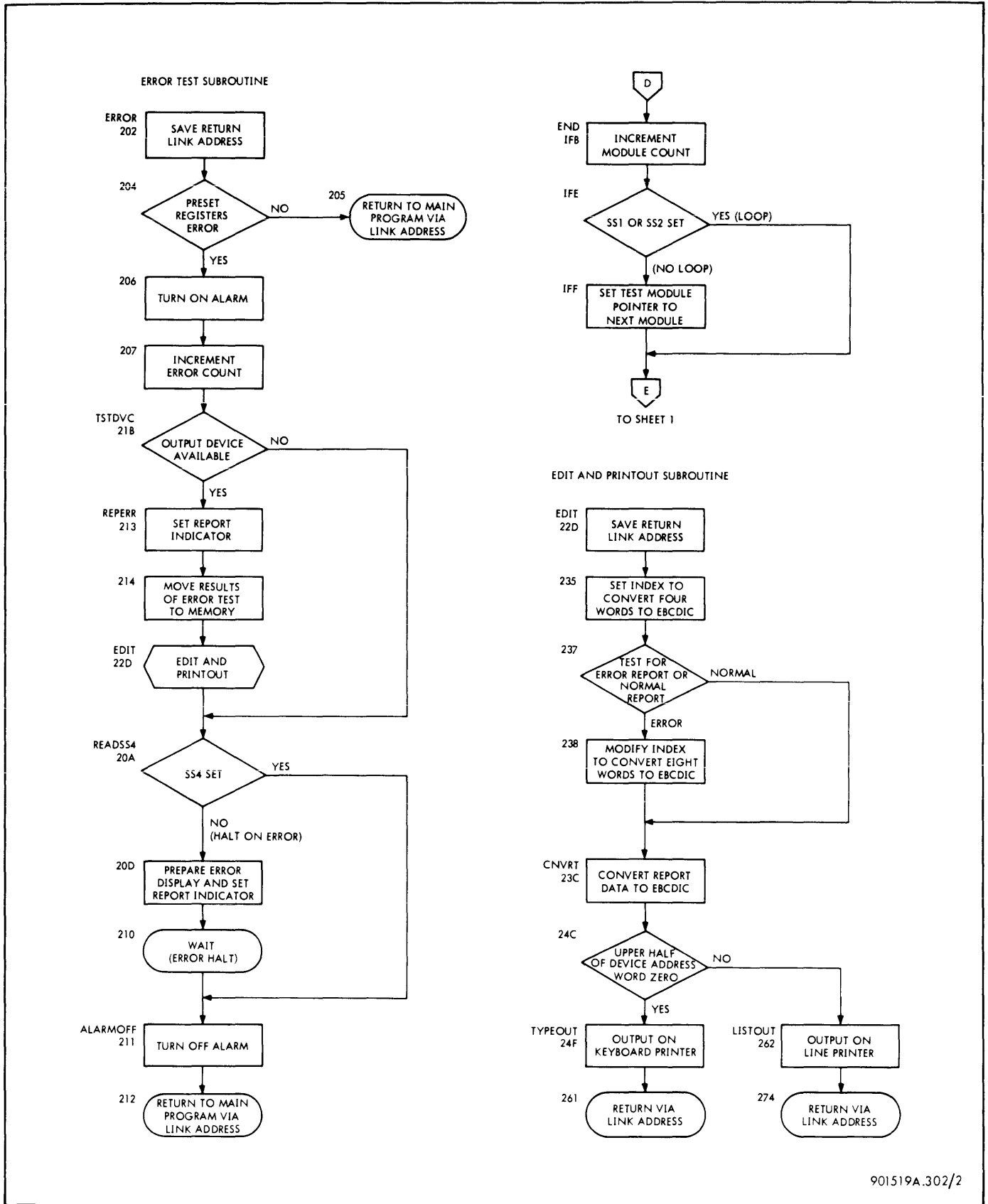


Figure 3-2. Sigma 5 Suffix Program, Detailed Flow Chart (Sheet 2 of 3 sheets)

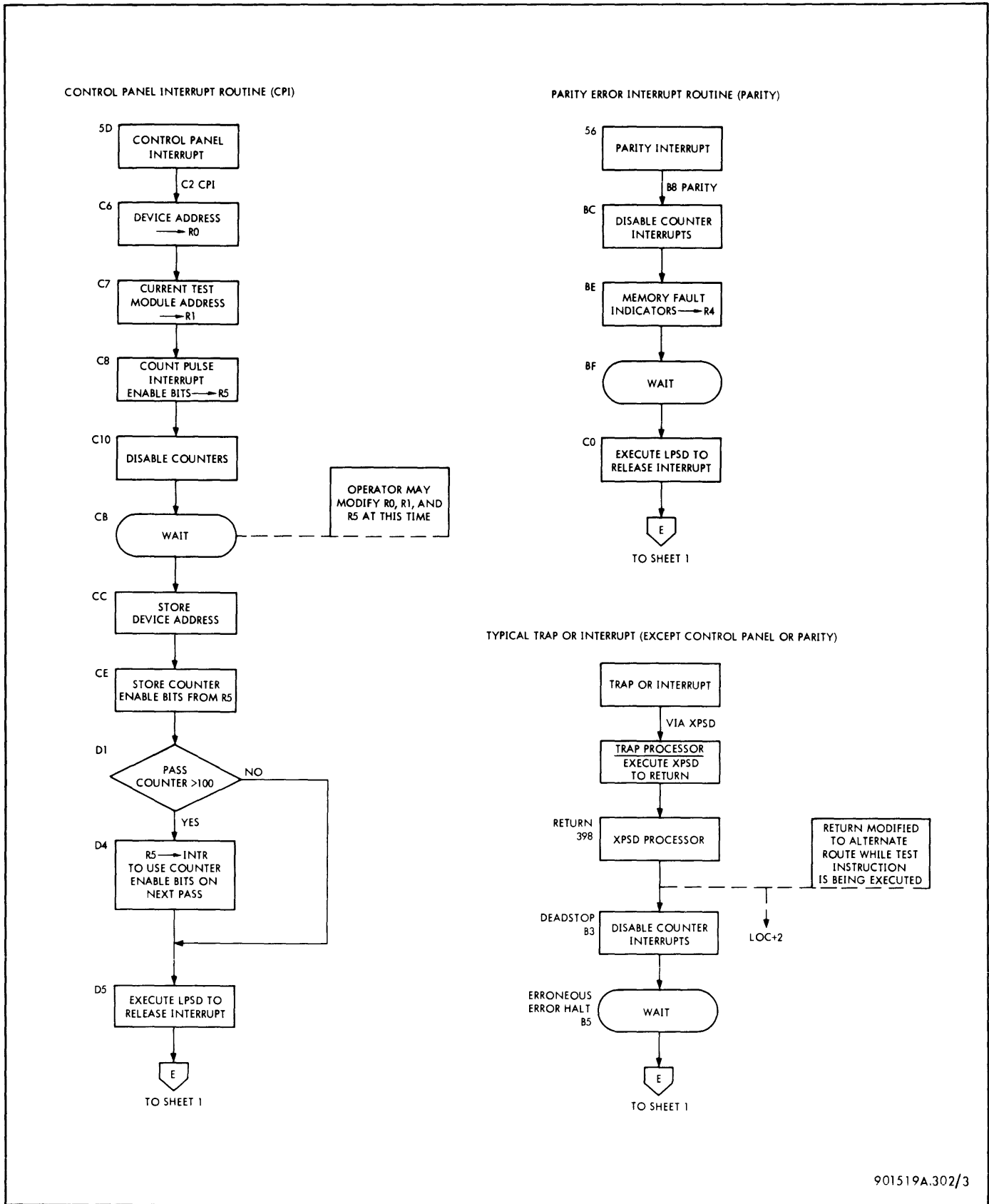


Figure 3-2. Sigma 5 Suffix Program, Detailed Flow Chart (Sheet 3 of 3 sheets)

SDS 901519

SECTION IV
PROGRAM LISTING

SIGMA 5 CPU DIAGNOSTIC - SUFFIX 704174-51300

46				PAGE		
47	01	00040		BRG	X1401	TRAP LOCATIONS
48						
49	01	00040	0F400060	NABTR	XPSD,4	NAB
50	01	00041	0F000074	JII TR	XPSD,0	JII
51	01	00042	0F00007A	SLTR	XPSD,0	SL
52	01	00043	0F000080	FXP8TR	XPSD,0	FXP8
53	01	00044	0F000086	FLPFTR	XPSD,0	FLPF
54	01	00045	0F00008C	DFTR	XPSD,0	DF
55	01	00046	0F000092	WDTRTR	XPSD,0	WDTR
56	01	00047	0F000098	BRANCH	XPSD,0	RETJRN
57	01	00048	0F00009E	CAL1TR	XPSD,0	CAL1
58	01	00049	0F0000A4	CAL2TR	XPSD,0	CAL2
59	01	0004A	0F0000AA	CAL3TR	XPSD,0	CAL3
60	01	00043	0F0000AA	CAL4TR	XPSD,0	CAL4
61						
62						
63	01	00052		BRG	X1521	INTERRUPT LOCATIONS
64						
65	01	00052	33100399	MTW,1	CNT1CP	COUNT PULSE INTERRUPT 1
66	01	00053	3310039A	MTW,1	CNT2CP	COUNT PULSE INTERRUPT 2
67	01	00054	3310039B	MTW,1	CNT3CP	COUNT PULSE INTERRUPT 3
68	01	00055	3310039C	MTW,1	CNT4CP	COUNT PULSE INTERRUPT 4
69	01	00056	0F0000B8	XPSD,0	PARITY	MEMORY PARITY
70	01	00057	0F0000BE	XPSD,0	RETJRN	
71	01	00058	0F0000BE	XPSD,0	RETJRN	
72	01	00059	0F0000BE	XPSD,0	RETJRN	
73	01	0005A	0F0000BE	XPSD,0	RETJRN	
74	01	0005B	0F0000BE	XPSD,0	RETJRN	
75	01	0005C	0F0000BE	XPSD,0	RETJRN	
76	01	0005D	0F0000C2	XPSD,0	CPI	I/O INTERRUPT IS NEVER ENABLED
77	01	0005E	0F0000BE	XPSD,0	RETJRN	CONTROL PANEL INTERRUPT
78	01	0005F	0F0000BE	XPSD,0	RETJRN	

SIGMA 5 CPU DIAGNOSTIC - SUFFIX 704174-51300

79				PAGE		
80						***NON-ALLOWED OPERATION TRAP***
81	01	00060	00000000 A	NAB	PZE	
82	01	00061	00000000 A		PZE	
83	01	00062	00000064		PZE,0	\$+2
84	01	00063	00000000 A		PZE	
85	01	00064	0F00003BE	NABRET	XPSD,0	RETJRN
86	01	00065	0F00003BE	MPVRET	XPSD,0	RETJRN
87	01	00066	0F00003BE	MVRET	XPSD,0	RETJRN
88	01	00067	0F00003BE		XPSD,0	RETJRN
89	01	00068	0F00003BE	NEARET	XPSD,0	RETJRN
90	01	00069	0F00003BE		XPSD,0	RETJRN
91	01	0006A	0F00003BE		XPSD,0	RETJRN
92	01	0006B	0F00003BE		XPSD,0	RETJRN
93	01	0006C	0F00003BE	NEIRET	XPSD,0	RETJRN
94	01	0006D	0F00003BE		XPSD,0	RETJRN
95	01	0006E	0F00003BE		XPSD,0	RETJRN
96	01	0006F	0F00003BE		XPSD,0	RETJRN
97	01	00070	0F00003BE		XPSD,0	RETJRN
98	01	00071	0F00003BE		XPSD,0	RETJRN
99	01	00072	0F00003BE		XPSD,0	RETJRN
100	01	00073	0F00003BE		XPSD,0	RETJRN
101						***UNIMPLEMENTED INSTRUCTION TRAP***
102	01	00074	00000000 A	JII	PZE	
103	01	00075	00000000 A		PZE	
104	01	00076	00000078		PZE,0	\$+2
105	01	00077	00000000 A		PZE	
106	01	00078	0F00003BE	JIIRET	XPSD,0	RETJRN
107						***STACK LIMIT REACHED TRAP***
108						
109	01	0007A	00000000 A	SL	B3JND 8	
110	01	0007B	00000000 A		PZE	
111	01	0007C	0000007E		PZE,0	\$+2
112	01	0007D	00000000 A		PZE	
113	01	0007E	7730007A		LCF,2	SL
114	01	0007F	0F00003BE	SLRET	XPSD,0	RETJRN

115				PAGE	
116			*		***FIXED ARITHMETIC OVERFLOW TRAP***
117				BOUND 8	
118	01 00080	00000000 A	FXP0	PZE	
119	01 00081	00000000 A		PZE	
120	01 00082	00000084		PZE,0	*+2
121	01 00083	00000000 A		PZE	
122	01 00084	70300080		LCF,2	FXP0
123	01 00085	0F0003BE	FP0RET	XPS0,0	RETURN
124			*		***FLOATING POINT ARITHMETIC FAULT TRAP***
125				BOUND 8	
126	01 00086	00000000 A	FLPF	PZE	
127	01 00087	00000000 A		PZE	
128	01 00088	0000008A		PZE,0	*+2
129	01 00089	00000000 A		PZE	
130	01 0008A	70300086		LCF,2	FLPF
131	01 0008B	0F0003BE	FPFRET	XPS0,0	RETURN
132			*		***UNUSED TRAP LOCATION***
133				BOUND 8	
134	01 0008C	00000000 A	DF	PZE	(DECIMAL TRAP ON SIGMA 7)
135	01 0008D	00000000 A		PZE	
136	01 0008E	00000090		PZE,0	*+2
137	01 0008F	00000000 A		PZE	
138	01 00090	0F0003BE	DFRET	XPS0,0	RETURN
139			*		***WATCHDOG TIMER RUNOUT TRAP***
140				BOUND 8	
141	01 00092	00000000 A	WDTR	PZE	
142	01 00093	00000000 A		PZE	
143	01 00094	00000096		PZE,0	*+2
144	01 00095	00000000 A		PZE	
145	01 00096	0F0003BE	WDTRET	XPS0,0	RETURN

146				PAGE	
147			*		
148			*		***CALL 1 TRAP***
149				BOUND 8	
150	01 00098	00000000 A	CAL1	PZE	
151	01 00099	00000000 A		PZE	
152	01 0009A	0000009C		PZE,0	*+2
153	01 0009B	00000000 A		PZE	
154	01 0009C	0F0003BE	C1RET	XPS0,0	RETURN
155			*		
156			*		***CALL 2 TRAP***
157				BOUND 8	
158	01 0009E	00000000 A	CAL2	PZE	
159	01 0009F	00000000 A		PZE	
160	01 000A0	000000A2		PZE,0	*+2
161	01 000A1	00000000 A		PZE	
162	01 000A2	0F0003BE	C2RET	XPS0,0	RETURN
163			*		
164			*		***CALL 3 TRAP***
165				BOUND 8	
166	01 000A4	00000000 A	CAL3	PZE	
167	01 000A5	00000000 A		PZE	
168	01 000A6	000000A8		PZE,0	*+2
169	01 000A7	00000000 A		PZE	
170	01 000A8	0F0003BE	C3RET	XPS0,0	RETURN
171			*		
172			*		***CALL 4 TRAP***
173				BOUND 8	
174	01 000AA	00000000 A	CAL4	PZE	
175	01 000AB	00000000 A		PZE	
176	01 000AC	000000AE		PZE,0	*+2
177	01 000AD	00000000 A		PZE	
178	01 000AE	0F0003BE	C4RET	XPS0,0	RETURN

```

179                                     PAGE
180
181                                     *
182                                     * MODIFY STACK LIMIT TRAP PROCESSOR FOR THOSE MODULES EXPECTING TO TRAP
183                                     * THIS INSERTS CONDITION CODE, FS, FZ, FN, MS, DN AND AN BITS INTO
184                                     * THE PSW1 WHICH IS LOADED AFTER A TRAP
185                                     *
185 01 000AF 484000B2 SLSW EBR,4 SLAD
186 01 00090 3540007C STW,4 SL*2
187 01 00031 690C0151 B SETPSW
188 01 00032 0000007E SLAD PZE SL*4
189
190
191
192                                     *
193                                     * ERRONEOUS TRAP DURING OPERATION OF DRIVER PROGRAM STOPS HERE
194                                     *
194 01 00033 22ACF000 A DEADSTOP LI,10 X'F000'
195 01 00034 6DA01500 A WD,10 X'1500' DISABLE COUNT PULSE INTERRUPTS
196 01 00035 2E000000 A WAIT 0
197 01 00036 321002AB LW,1 SAVE
198 01 00037 0E0003BB LPSD,0 REPEAT

```

```

199                                     PAGE
200
201                                     *
202                                     * PARITY INTERRUPT SERVICE ROUTINE IS ENTERED FROM INTERRUPT L9C X'156'.
203                                     * MEMORY FAULT INDICATORS ARE STORED IN R4. CONTENTS OF OTHER REGISTERS
204                                     * ARE UNPREDICTABLE. LOCATION LABELED 'PARITY1' CONTAINS PROGRAM ADDRESS
205                                     * AT TIME OF INTERRUPT, BUT THIS MAY BE A SHORT WAY AFTER THE INSTR
206                                     * WHERE THE FAULT ACTUALLY OCCURED. UPON CLEARING THE WAIT, AN ATTEMPT
207                                     * IS MADE TO REPEAT THE SAME TEST MODULE.
208                                     *
209 01 00038 00000000 A PARITY BOUND B
210 01 00039 00000000 A PZE
211 01 0003A 000000BC PZE,0 $+2
212 01 0003B 00000000 A PZE
213 01 0003C 22ACF000 A LI,10 X'F000'
214 01 0003D 6DA01500 A WD,10 X'1500' DISABLE COUNT PULSE INTERRUPTS
215 01 0003E 6C400010 A RD,4 X'10' RECORD PARITY ERROR INDICATORS
216 01 0003F 2E000000 A WAIT,0 0
217 01 000C0 0E3003BB LPSD,3 REPEAT RELEASE INTERRUPT

```

```

218          PAGE
219          * CONTROL PANEL INTERRUPT ROUTINE IS ENTERED FROM LOCATION X'5D'.
220          * REGISTERS R0, R1, AND R5 ARE LOADED AND THEN A WAIT OCCURS, THE
221          * OPERATOR MAY CHANGE THE CONTENTS OF ANY OF THESE REGISTERS TO
222          * MODIFY THE OPERATION OF THE PROGRAM, THEN CLEAR THE WAIT TO CONTINUE.
223          *          R0 BITS 21-31      OUTPUT DEVICE ADDRESS;
224          *          BITS 0-16         ZERO VALUE INDICATES TYPEWRITER;
225          *                          NONZERO VALUE INDICATES LINE PRINTER
226          *
227          *          R1                CURRENT TEST MODULE ADDRESS,
228          *                          PROGRAM BEGINS WITH THIS MODULE
229          *                          WHEN WAIT IS CLEARED.
230          *
231          *          R5 BITS 16-19     CNT PLS INTRPT ARM AND ENABLE BITS.
232          *
233          *
234          *          BRJND B
235          *          PZE
236          *          PZE
237          *          PZE,0      **2
238          *          LWA,0     DVC
239          *          LWA,1     SAVE
240          *          LWA,5     INTRC
241          *          LI,10    X'F000'
242          *          WD,10    X'1500'
243          *          WAIT
244          *          STW,0     DVC
245          *          AND,5     CNTRMASK
246          *          STW,5     INTRC
247          *          LWA,4     PASSES
248          *          SLS,4     *16
249          *          CI,4      100
250          *          BLE      **3
251          *          AI,5     X'810'
252          *          STW,5     INTR
253          *          LPSD,3   REPEAT
    
```

```

254          PAGE
255          * *****CONTROL PROGRAM BEGINS OPERATION HERE*****
256          * *****
257          * *****
258          * *****
259          *          BRG      X'F9'
260          *
261          *          LWA,0     BT9100
262          *          STW,0     X'26'
263          *          RD,0      0
264          *          BCR,8     START
265          *          LI,0      0
266          *          STW,0     INTRC
267          *          WAIT
268          *          LI,0      DEADST9P
269          *          STW,0     RETJRN*2
270          *          LI,1      4
271          *          LI,0      0
272          *          STW,0     DISPLAY=1,1
273          *          BDR,1     **1
274          *          LI,0      X'810'
275          *          STW,0     INTR
276          *          LI,2     *51
277          *          STW,2     LINE
278          *          LI,2     *2
279          *          STW,2     FIRST
280          *          STW,2     FIRSTL
281          *          BAL,15    C9RSIZE
282          *          LI,1     LIST
283          *          LI,0     DEADST9P
284          *          STW,0     RETJRN*2
285          *          STW,1     SAVE
286          *          WD,0      X'40'
287          *          LI,10    *1
288          *          WD,10    X'1100'
289          *          LWA,10   INTR
    
```

291				PAGE		
292				WORKING TABLES PRIOR TO LOADING AS SPECIFIED IN MODULE		
293						
294	01	00117	22000000 A	LI,0	0	
295	01	00118	22100000 A	LI,1	12	
296	01	00119	3502029F	STW,0	TABLE=1,1	CLEAR MODULE STORAGE TABLE
297	01	0011A	64100119	BDR,1	0=1	
298						
299	01	00113	22100010 A	LI,1	16	
300	01	0011C	350202B2	STW,0	RTABLIN=1,1	CLEAR RTABLIN
301	01	0011D	6410011C	BDR,1	0=1	
302						
303	01	0011E	22100010 A	LI,1	16	
304	01	0011F	350202C2	STW,0	RTABLOUT=1,1	CLEAR RTABLOUT
305	01	00120	6410011F	BDR,1	0=1	
306						
307	01	00121	22100010 A	LI,1	16	
308	01	00122	350202D2	STW,0	MTABLIN=1,1	CLEAR MTABLIN
309	01	00123	6410012F	BDR,1	0=1	
310						
311	01	00124	22100012 A	LI,1	18	
312	01	00125	350202E2	STW,0	MTABLOUT=1,1	CLEAR MTABLOUT AND STACK POINTER
313	01	00126	64100125	BDR,1	0=1	
314						
315	01	00127	22100010 A	LI,1	16	
316	01	00128	350202F2	STW,0	RRESULT=1,1	CLEAR RESULT
317	01	00129	6410012A	BDR,1	0=1	
318						
319	01	0012A	22100014 A	LI,1	20	
320	01	00123	35020303	STW,0	MEMORY=1,1	CLEAR MEMORY, STACK POINTER AND IA
321	01	0012C	6410012B	BDR,1	0=1	
322						
323	01	0012D	22100004 A	LI,1	4	
324	01	0012E	350202AE	STW,0	DISPLAY=3,1	CLEAR ERROR INDICATIONS FROM DISPLAY
325	01	0012F	6410012E	BDR,1	0=1	

PAGE

- PICK UP FIRST WORD IN MODULE (COUNT WORD) AND TEST FOR END OF
- MODULE LIST. THE END OF LIST IS INDICATED BY AN 'ALL-ZEROS' WORD
- AFTER THE LAST MODULE.

331	01	00130	824002A8	LW#4	*SAVE	PICK UP COUNT
332	01	00131	69300142	BNEZ	NOTEND	TEST FOR END INDICATOR
333	01	00132	323002AD	LW#3	PASSES	GET PASS COUNTER
334	01	00133	20310000 A	AI#3	X'10000'	INCREMENT PASS COUNT
335	01	00134	493003AB	AND#3	WOT15	CLEAR MODULE COUNT WITH MASK
336	01	00135	353002AD	STW#3	PASSES	PUT AWAY NEW COUNT

- TEST ERROR AND PASS COUNTERS. IF 100 PASSES HAVE BEEN COMPLETED
- WITHOUT ERROR, MODIFY THE REGISTER WORD WHICH NO INSTRUCTION USES
- SO THAT COUNT PULSE INTERRUPTS WILL BE ARMED AND ENABLED FOR ALL
- SUBSEQUENT PASSES.

343	01	00136	313003AF	CW#3	NJM	
344	01	00137	6930010E	BNE	CYCLE=1	TEST PASS COUNT
345	01	00138	322002AC	LW#2	ERRORS	
346	01	00139	6930010E	BNEZ	CYCLE=1	TEST ERROR COUNT
347	01	0013A	3250039E	LW#5	INTRC	TEST INTRC TO SEE IF OPERATOR HAS
348	01	0013B	6830010E	BEZ	CYCLE=1	ALREADY CLEARED ENABLING BITS
349	01	0013C	4950039F	AND#5	CNTRMASK	
350	01	0013D	20500810 A	AI#5	X'810'	INSERT BITS FOR CONTROL PANEL
351	01	0013E	3550039D	STW#5	INTR	AND PARITY INTERRUPTS
352	01	0013F	220001ED	LI#0	DA(CNTRDW)	
353	01	00140	6AF00278	BAL#15	PRINT	
354	01	00141	6800010E	B	CYCLE=1	START NEW PASS

PAGE

- MOVE MODULE TO TABLE (COUNT WORD IS CURRENTLY IN R6)

359	01	00142	326002AR	NOTEND	LW#6	SAVE	GET MODULE POINTER
360	01	00143	38600C04 A		SW#6	4	ADD COUNT
361	01	00144	22100000 A		LI#1	0	
362	01	00145	356003AR		STW#6	NEXT	SAVE POINTER TO NEXT MODULE
363	01	00146	327803AB		LW#7	*NEXT#4	MOVE CURRENT MODULE TO TABLE
364	01	00147	3572029F		STW#7	TABLE#1	
365	01	00148	20100001 A		AI#1	1	
366	01	00149	65400146		BIR#4	8=3	

- SET UP TO EXECUTE TEST MODULE WHICH IS NOW RESIDING IN TABLE.

370	01	0014A	324003AA		LW#4	XPSD	
371	01	0014B	354001AD		STW#4	LSC#1	INITIALIZE RETURN VERTICAL IN LOC#1
372	01	0014C	324002A1		LW#4	TABLE#2	PICK UP PSW#1 IN FROM MODULE
373	01	0014D	434003AD		AND#4	LINKAD	SELECT LINK ADDRESS; DELETE MS BIT
374	01	0014E	354003BA		STW#4	TEMP	
375	01	0014F	434003AE		AND#4	CBND	CLEAR BUT ADDRESS
376	01	00150	0E0003BA		LPSD#0	TEMP	GO TO SETPSW OR SET UP EXPECTED TRAP
377	01	00151	324002A1	SETPSW	LW#4	TABLE#2	PICK UP PSW#1 IN FROM MODULE
378	01	00152	434003AE		AND#4	CBND	CLEAR ADDRESS
379	01	00153	434003BC		EGR#4	LOCADD	INSERT ADDRESS OF LOC
380	01	00154	354003BC		STW#4	PSW1	SAVE PSW#1 IN (WITH ADDRESS INSERTED)
381	01	00155	324002A0		LW#4	TABLE#1	PICK UP INSTRUCTION
382	01	00156	354002AE		STW#4	DISPLAY#3	INSERT INSTRUCTION IN DISPLAY TABLE
383	01	00157	354001AC		STW#4	LOC	INSERT INSTRUCTION IN TEST LOCATION
384	01	00158	325003B1		LW#5	INDA	
385	01	00159	35500316		STW#5	IA	INITIALIZE INDIRECT ADDRESS
386	01	0015A	325003B2		LW#5	INDASP	
387	01	0015B	35500317		STW#5	IASP	INIT. STACK POINTER INDIRECT ADDRESS

```

388                                     PAGE
389
390 * SET-UP REGISTER-IN TABLE (RTABLIN)
391 *
392 01 0015C 22700001 A SETRIN LI,7 1 HALFWORD INDEX
393 01 0015D 522002A3 L#2 2 TABLE+4 PICK UP REGISTER-IN POINTER (ADDRESS)
394 01 0015E 524E02A3 L#4 4 TABLE+4,7 PICK UP COUNT AND 1ST REG INDICATOR
395 01 0015F 6830016A BEZ SETROUT
396 01 00160 25400578 A SAD,4 8 COUNT IN R4
397 01 00161 25500068 A SLS,5 24 FIRST REG INDICATOR IN R5
398 01 00162 38200004 A S#2 4 ADD COUNT TO ADDRESS
399 01 00163 32680002 A FETCHRIN L#6 2,4 FETCH DATA
400 01 00164 21500010 A CI,5 16
401 01 00165 69100167 BL 2,2
402 01 00166 22500000 A LI,5 0 WRAP AROUND TO 0 IF R EXCEEDS 15
403 01 00167 356A02B3 STW,6 RTABLIN,5 FILL REGISTER-IN TABLE
404 01 00168 20500001 A AI,5 1
405 01 00169 65400163 BIR,4 FETCHRIN
406
407 * SET-JP REGISTER-OUT TABLE (RTABLOUT)
408 *
409 01 0016A 522002A4 SETROUT L#2 5 PICK UP REGISTER-OUT POINTER (ADDR)
410 01 0016B 524E02A4 L#4 4 TABLE+5,7 PICK-UP COUNT AND 1ST REG INDICATOR
411 01 0016C 68300177 BEZ SETMIN
412 01 0016D 25400578 A SAD,4 8 COUNT IN R4
413 01 0016E 25500068 A SLS,5 24 FIRST REGISTER INDICATOR IN R5
414 01 0016F 38200004 A S#2 4 ADD COUNT TO ADDRESS
415 01 00170 32680002 A FETCHRR L#6 2,4 FETCH DATA
416 01 00171 21500010 A CI,5 16
417 01 00172 69100174 BL 2,2
418 01 00173 22500000 A LI,5 0 WRAP AROUND TO 0 IF R EXCEEDS 15
419 01 00174 356A02C3 STW,6 RTABLOUT,5 FILL REGISTER-OUT TABLE
420 01 00175 20500001 A AI,5 1
421 01 00176 65400170 BIR,4 FETCHRR

```

```

422                                     PAGE
423
424 * SET UP MEMORY-IN TABLE (MTABLIN)
425 *
426 01 00177 522002A5 SETMIN L#2 6 PICK UP MEMORY-IN POINTER (ADDRESS)
427 01 00178 524E02A5 L#4 4 TABLE+6,7 PICK UP COUNT AND 1ST WORD INDICATOR
428 01 00179 68300184 BEZ SETROUT
429 01 0017A 25400578 A SAD,4 8 COUNT IN R4
430 01 0017B 25500068 A SLS,5 24 FIRST WORD INDICATOR IN R5
431 01 0017C 38200004 A S#2 4 ADD COUNT TO ADDRESS
432 01 0017D 32680002 A FETCHMI L#6 2,4 FETCH DATA
433 01 0017E 21500010 A CI,5 16
434 01 0017F 69100181 BL 2,2
435 01 00180 22500000 A LI,5 0 WRAP AROUND IF WORD IND. EXCEEDS 15
436 01 00181 356A02D3 STW,6 MTABLIN,5 FILL MEMORY-IN TABLE
437 01 00182 20500001 A AI,5 1 INCREMENT WORD INDICATOR (INDEX)
438 01 00183 6540017D BIR,4 FETCHMI
439
440 * SET UP MEMORY-OUT TABLE (MTABLOUT)
441 *
442 01 00184 522002A6 SETROUT L#2 7 PICK UP MEMORY-OUT POINTER (ADDRESS)
443 01 00185 524E02A6 L#4 4 TABLE+7,7 PICK UP COUNT AND 1ST WORD INDICATOR
444 01 00186 68300191 BEZ SETROUT
445 01 00187 25400578 A SAD,4 8 COUNT IN R4
446 01 00188 25500068 A SLS,5 24 FIRST WORD INDICATOR IN R5
447 01 00189 38200004 A S#2 4 ADD COUNT TO ADDRESS
448 01 0018A 32680002 A FETCHMO L#6 2,4 FETCH DATA
449 01 0018B 21500010 A CI,5 16
450 01 0018C 6910018E BL 2,2
451 01 0018D 22500000 A LI,5 0 WRAP AROUND IF WORD IND. EXCEEDS 15
452 01 0018E 356A02E3 STW,6 MTABLOUT,5 FILL MEMORY-OUT TABLE
453 01 0018F 20500001 A AI,5 1 INCREMENT WORD INDICATOR (INDEX)
454 01 00190 6540018A BIR,4 FETCHMO
455

```



```

PAGE
456 * SHORT LOOP OPERATION RETURNS HERE TO BEGIN EACH LOOP
457 *
458 *
459 01 00191 221FFFF0 A SHORT LI#1 *16
460 01 00192 322202E3 L#2 RTABLIN+16,1 MOVE MEMORY DATA
461 01 00193 35220314 STW#2 MEMORY+16,1
462 01 00194 65100192 BR#1 *2
463 01 00195 322002A7 L#2 TABLE+8 MOVE
464 01 00196 35200314 STW#2 SP STACK
465 01 00197 322002A8 L#2 TABLE+9 POINTER
466 01 00198 35200315 STW#2 SP+1 DOUBLEWORD IN
467 *
468 *
469 * LOAD REGISTERS FROM RTABLIN
470 01 00199 32FC02C2 L#15 RTABLIN+15
471 01 0019A 32E002C1 L#14 RTABLIN+14
472 01 0019B 32D002C0 L#13 RTABLIN+13
473 01 0019C 32C002BF L#12 RTABLIN+12
474 01 0019D 32B002BE L#11 RTABLIN+11
475 01 0019E 32A002BD L#10 RTABLIN+10
476 01 0019F 329002BC L#9 RTABLIN+9
477 01 001A0 328002BB L#8 RTABLIN+8
478 01 001A1 327002BA L#7 RTABLIN+7
479 01 001A2 326002B9 L#6 RTABLIN+6
480 01 001A3 325002B8 L#5 RTABLIN+5
481 01 001A4 324002B7 L#4 RTABLIN+4
482 01 001A5 323002B6 L#3 RTABLIN+3
483 01 001A6 322002B5 L#2 RTABLIN+2
484 01 001A7 321002B4 L#1 RTABLIN+1
485 *
486 * SET UP RETURN TO LOC#2
487 *
488 01 001A8 220001AE LI#0 LOC#2
489 01 001A9 350003C0 STW#0 RETURN+2
490 01 001AA 320002B3 L#0 RTABLIN
    
```

```

PAGE
491 *
492 * EXECUTION OF TEST INSTRUCTION PROCEEDS AS FOLLOWS:
493 * THE PROGRAM STATUS DOUBLE WORD IS SET WITH ADDRESS OF LOC---OTHER
494 * BITS OF PSW1 ARE SET AS SPECIFIED IN TEST MODULE. THE TEST
495 * INSTRUCTION IS THEN EXECUTED IN LOCATION LOC. ALL RESULTS ARE SAVED
496 * FOR TESTING UNLESS OPERATING IN THE SHORT LOOP MODE.
497 *
498 *
499 01 001A3 0E0003BC LPS#0 PSW1
500 *****
501 01 001AC 00000000 A LOC PZE TEST INSTRUCTION INSERTED HERE
502 *****
503 01 001AD 0F0003BE XPS#0 RETURN SAVE RESULTING PSD#
504 01 001AE 350002F3 STW#0 RRESULT SAVE REGISTER 0 RESULTS
505 *
506 * MODIFY RETURN SO THAT ANY TRAPS AFTER THIS POINT GO TO DEADSTOP
507 *
508 01 001AF 220000B3 LI#0 DEADSTOP
509 01 00130 350003C0 STW#0 RETURN+2 SET RETURN TO DEADSTOP
510 *
511 01 001B1 6C000000 A RD#0 0 READ SS1
512 01 00132 69800191 BCS#8 SHORT SHORT LOOP
513 *
514 * SAVE REGISTER 1 THRU 15 RESULTS
515 *
516 01 001B3 320003B3 L#0 MOVER
517 01 001B4 350001B5 STW#0 *+1
518 01 001B5 351002F4 STW#1 RRESULT+1 INSTRUCTION MODIFIED WHILE RUNNING
519 01 001B6 300003B4 A#0 MOVERMOD
520 01 00137 310003B5 C#0 MOVERFIN
521 01 00138 691001B4 BL *+4
    
```

```

522                                     PAGE
523                                     *
524                                     * BEGIN TESTING RESULTS
525                                     *
526                                     * IN EACH TEST, THE IDENTIFIER IS PLACED IN R5 AND THE TWO ITEMS TO
527                                     * BE COMPARED ARE PLACED IN R6 (ACTUAL RESULT) AND R7 (EXPECTED RESULT).
528                                     * A BRANCH IS THEN MADE TO THE ERROR ROUTINE WHICH COMPARES R6 AND R7.
529                                     *
530                                     *
531                                     * TEST INSTRUCTION
532 01 001B9 325003C6 L#5 INSTID PICK UP IDENTIFIER
533 01 001BA 326001AC L#6 LOC IS
534 01 001BB 327002A0 L#7 TABLE+1 SHOULD BE
535 01 001BC 6AF00202 BAL,15 ERROR
536
537                                     *
538                                     * TEST LOCATION+1
539 01 001BD 325003C7 L#5 XPSDID PICK UP IDENTIFIER
540 01 001BE 326001AD L#6 LOC+1 IS
541 01 001BF 327003AA L#7 XPSD SHOULD BE
542 01 001C0 6AF00202 BAL,15 ERROR
543
544                                     *
545                                     * TEST INDIRECT ADDRESS LOCATION
546 01 001C1 325003C8 L#5 IAID PICK UP IDENTIFIER
547 01 001C2 32600316 L#6 IA IS
548 01 001C3 327003B1 L#7 INDA SHOULD BE
549 01 001C4 6AF00202 BAL,15 ERROR
550
551                                     *
552                                     * TEST STACK INDIRECT ADDRESS LOCATION
553 01 001C5 2C500001 A AI,5 1 INCREMENT IDENTIFIER
554 01 001C6 32600317 L#6 IASP IS
555 01 001C7 327003B2 L#7 INDASP SHOULD BE
556 01 001C8 6AF00202 BAL,15 ERROR

```

```

557                                     PAGE
558                                     * TEST PSW1
559                                     *
560 01 001C9 325003C9 L#5 PSDWID PICK UP IDENTIFIER
561 01 001CA 326003BE L#6 RETURN IS
562 01 001CB 327002A2 L#7 TABLE+3 SHOULD BE
563 01 001CC 6AF00202 BAL,15 ERROR
564
565                                     *
566                                     * TEST PSW2
567 01 001CD 2C500001 A AI,5 1 ADD 1 TO IDENTIFIER
568 01 001CE 326003BF L#6 RETURN+1 IS
569 01 001CF 327003B6 L#7 PSW2 SHOULD BE (CONSTANT FOR ALL MODULES)
570 01 001D0 6AF00202 BAL,15 ERROR
571
572                                     *
573                                     * TEST ALL REGISTER RESULTS
574 01 001D1 325003CA L#5 REGID PICK UP IDENTIFIER
575 01 001D2 326A02F3 TESTALLR L#6 RRESULT,5 IS
576 01 001D3 327A02C3 L#7 RTABLOUT,5 SHOULD BE
577 01 001D4 6AF00202 BAL,15 ERROR
578 01 001D5 2C500001 A AI,5 1 INCREMENT IDENTIFIER
579 01 001D6 315003CD C#5 REGIDFIN
580 01 001D7 691001D2 BL TESTALLR NOT FINISHED
581
582                                     *
583                                     * TEST ALL MEMORY RESULTS IN TABLE LABELED MEMORY
584 01 001D8 325003CB L#5 MEMID PICK UP IDENTIFIER
585 01 001D9 326A0000 A TESTALLM L#6 0,5 IS
586 01 001DA 3273FF3F A L#7 MTABLOUT-MEMORY,5 SHOULD BE
587 01 001DB 6AF00202 BAL,15 ERROR
588 01 001DC 2C500001 A AI,5 1 INCREMENT IDENTIFIER
589 01 001DD 315003CE C#5 MEMIDFIN TEST FOR END OF MEMORY TABLE
590 01 001DE 693001D2 BL TESTALLM NOT FINISHED

```

591			PAGE		
592			*		
593			*	TEST STACK POINTER DOUBLEWORD	
594			*		
595	01 001DF	325003CC	LW,5	SPID	PICK UP IDENTIFIER
596	01 001EO	32600314	LW,6	SP	IS
597			* DELETED ONE INSTRUCTION		*B
598	01 001E1	327002A9	LW,7	TABLE+10	SHOULD BE
599	01 001E2	6AF00202	BAL,15	ERROR	
600	01 001E3	20500001 A	A,5	1	INCREMENT IDENTIFIER
601	01 001E4	32600315	LW,6	SP+1	IS
602	01 001E5	327002AA	LW,7	TABLE+11	SHOULD BE
603	01 001E6	6AF00202	BAL,15	ERROR	

604			PAGE		
605			*		
606			* ALL TESTS HAVE BEEN COMPLETED AT THIS POINT. CHECK REPORT INDICATOR		
607			* TO SEE IF ANY ERRORS WERE REPORTED ON CURRENT MODULE. IF YES, BYPASS		
608			* READING OF SS3. IF NO, READ SS3 AND PRINT REPORT IF SS3 SET. CLEAR		
609			* REPORT INDICATOR.		
610			*		
611	01 001E7	22500000 A	L,5	0	CLEAR ERROR IDENTIFIER
612	01 001E8	355002AF	STW,5	DISPLAY**	
613	01 001E9	32E003CF	LW,14	RPTIND	PICK UP REPORT INDICATOR
614	01 001EA	693001F8	BNEZ	CLRRPT	TEST REPORT INDICATOR
615	01 001EB	6C000000 A	RD,0	0	READ SS3 (REQUEST FOR REPORT)
616	01 001EC	682001F8	BCR,2	CLRRPT	
617	01 001ED	6AF0021B	BAL,15	TSTOVC	TEST OUTPUT DEVICE
618	01 001EE	680001F7	B	OUTRPT	RETURN HERE IF DEVICE IS AVAILABLE
619	01 001EF	2280FC00 A	L,11	X'F000'	RETURN HERE IF DEVICE IS UNAVAIL.
620	01 001F0	6DB01500 A	WD,11	X'1500'	DISABLE COUNTER INTERRUPTS
621	01 001F1	6AF00220	BAL,15	LOAD4	LOAD REG. R1-R4 WITH DISPLAY INF9
622	01 001F2	22600000 A	L,6	0	CLEAR R6
623	01 001F3	22700000 A	L,7	0	CLEAR R7
624	01 001F4	22800000 A	L,8	0	CLEAR R8
625	01 001F5	2E000000 A	WAIT	0	REPORT HALT
626	01 001F6	680001F8	B	END	
627	01 001F7	6AF00220	OUTRPT	BAL,15	EDIT
628	01 001F8	22E00000 A	CLRRPT	0	OUTPUT REPORT
629	01 001F9	35E003CF	L,14	RPTIND	CLEAR REPORT INDICATOR
630	01 001FA	321002AB	LW,1	SAVE	GET CURRENT MODULE ADDRESS
631	01 001FB	22200001 A	L,2	1	
632	01 001FC	531402AD	MTH,1	PASSES,2	INCREMENT MODULE COUNT
633	01 001FD	6C000000 A	RD,0	0	
634	01 001FE	67C0010F	BCS,12	CYCLE	READ SS1 AND SS2 (LOOP)
635	01 001FF	321003AB	LW,1	NEXT	GET NEXT MODULE ADDRESS IF NO LOOP
636	01 00200	6800010F	B	CYCLE	

```

637                                     PAGE
638
639                                     *
640                                     * ERROR ROUTINE COMPARES R6 AND R7 USING AN EOR INSTRUCTION. IF NO
641                                     * ONES RESULT FROM EOR (IE1 NO ERROR) RETURN IS MADE TO TEST SEQUENCE.
642                                     * IF AN ERROR IS DETECTED, AN ATTEMPT TO REPORT IS MADE AND THE ALARM
643                                     * IS TURNED ON.
644                                     *
645 01 00201 00000000 A ERRLINK PZE ERRLINK SAVE RETURN LINK
646 01 00202 35F00201 ERRBR STW,15 ERRLINK PICK UP RESULT
647 01 00203 37800006 A LW,8 6 COMPARE WITH PREDETERMINED RESULT
648 01 00204 48800007 A EOR,8 7 NO ERROR
649 01 00205 E8300201 BEZ *ERRLINK TURN ON ALARM
650 01 00206 60000041 A WD,0 X'41' INCREMENT ERROR COUNTER
651 01 00207 331002AC MTW,1 ERRORS TEST DEVICE
652 01 00208 64F0021B BAL,15 TSTDVC RETURN HERE IF DEVICE AVAILABLE
653 01 00209 68000213 B REPERR RETURN HERE IF DEVICE UNAVAIL.
654 01 0020A 6C000000 A READSS4 RD,0 0 READ SS4
655 01 0020B 67100211 BCS,1 ALRM0FF SET REPORT INDICATOR
656 01 0020C 2280FC00 A LI,11 X'F000' DISABLE COUNTER INTERRUPTS
657 01 0020D 331003CF MTW,1 RPTIND LOAD REG. R1-R4 WITH DISPLAY INFO
658 01 0020E 60801500 A WD,11 X'1500' LOAD R5 THRU R8
659 01 0020F 64F00220 BAL,15 LOAD4 HALT ON ERROR
660 01 00210 2E000000 A WAIT 0 TURN OFF ALARM
661 01 00211 60000040 A ALRM0FF WD,0 X'40' NEXT TEST
662 01 00212 E8000201 B *ERRLINK SET REPORT INDICATOR
663 01 00213 331003CF REPERR MTW,1 RPTIND
664 01 00214 355002AF STW,5 DISPLAY+4
665 01 00215 356002B0 STW,6 DISPLAY+5
666 01 00216 357002B1 STW,7 DISPLAY+6
667 01 00217 358002B2 STW,8 DISPLAY+7
668 01 00218 64F0022D BAL,15 EDIT
669 01 00219 64F0022F BAL,15 LOAD5
670 01 0021A 6800020A B READSS4

```

```

670                                     PAGE
671
672                                     *
673                                     * TSTDVC TESTS OUTPUT DEVICE AVAILABILITY. IF AVAILABLE, RETURN IS TO
674                                     * ADDRESS IN LINK. IF UNAVAILABLE, RETURN IS TO ADDRESS+1.
675                                     *
676 01 0021B CF0003B7 TSTDVC HI0,0 *DVC
677 01 0021C C0B003B7 T18,11 *DVC
678 01 0021D E8C0000F A BCR,12 *15 SIO POSSIBLE
679 01 0021E 27F00001 A AI,15 1
680 01 0021F E800000F A B *15 UNAVAILABLE OR NOT OPERATIONAL
681
682                                     *
683                                     * THIS ROUTINE LOADS REGISTERS R1-R4 FROM THE FIRST 4 WORDS OF DISPLAY.
684                                     *
685 01 00220 321002A9 LOAD4 LW,1 DISPLAY
686 01 00221 322002AC LW,2 DISPLAY+1
687 01 00222 323002AD LW,3 DISPLAY+2
688 01 00223 324002AE LW,4 DISPLAY+3
689 01 00224 E800000F A B *15
690
691                                     *
692                                     * THIS ROUTINE LOADS REGISTERS R5-R8 FROM THE LAST 4 WORDS OF DISPLAY
693                                     *
694 01 00225 325002AF LOADS LW,5 DISPLAY+4
695 01 00226 326002B0 LW,6 DISPLAY+5
696 01 00227 327002B1 LW,7 DISPLAY+6
697 01 00228 328002B2 LW,8 DISPLAY+7
698 01 00229 E800000F A B *15

```

```

698
699
700
701
702
703
704
705
706
707
708
709 01 0022A 00000000 A EDITLINK PZE
710 01 0022B 358002B2 A EDMLAST STW,8 DISPLAY+7
711 01 0022C 354002AE A EDMB/E STW,4 DISPLAY+3
712 01 0022D 35F0022A A EDIT STW,15 EDITLINK SAVE RETURN LINK
713 01 0022E 3200022C A LWB,0 EDMBVE
714 01 0022F 300003B4 A AW,0 MBVERM0D
715 01 00230 35000232 A STW,0 *+2
716 01 00231 3100022B A CW,0 EDMLAST
717 01 00232 00000000 A PZE MOVE R5-R8 TO DISPLAY TABLE FOR CONV
718 01 00233 6810022F A BGE *+4 NOT FINISHED
719 01 00234 22E002AF A LI,14 DISPLAY+4 SET UP TO CONVERT 4 WORDS
720 01 00235 222FFFFC A LI,2 *+
721 01 00236 32F002AF A LW,15 DISPLAY+4 GET ERROR IDENTIFIER
722 01 00237 6830023A A BCR,3 *+3 TEST FOR ERROR
723 01 00238 20E00004 A AI,14 4 IF ERROR, MODIFY SET UP FOR 8 WORDS
724 01 00239 202FFFFC A AI,2 *+
725 01 0023A 22500000 A LI,5 0 SET BYTE INDEX FOR STORING IMAGE
726 01 00233 32000350 A LWB,0 BLANK
727 01 0023C 22400002 A CNVRT LI,4 2
728 01 0023D 75A033C A STB,0 IMAGE+1,5 INSERT 2 BLANKS BETWEEN WORDS
729 01 0023E 20500001 A AI,5 1 INCREMENT BYTE INDEX
730 01 0023F 6440023D A BDR,4 *+2
731 01 00240 22600008 A LI,6 8
732 01 00241 82A4000E A LW,10 *14,2 GET WORD TO BE CONVERTED

```

```

733
734
735 01 00242 22800000 A CNVRTMR LI,11 0 CLEAR R11
736 01 00243 25A00304 A SCD,10 4 HEX CHAR, INTO R11
737 01 00244 21800009 A CI,11 9
738 01 00245 69200247 A BCS,2 *+2
739 01 00246 20800039 A AI,11 X'39'
740 01 00247 20800087 A AI,11 X'87' CONVERT HEX CHAR TO EBCDIC BYTE
741 01 00248 75BA033C A STB,11 IMAGE+1,5
742 01 00249 27500001 A AI,5 1 INCREMENT BYTE INDEX
743 01 0024A 64600242 A BDR,6 CNVRTMR
744 01 00243 6520023C A BIR,2 CNVRT BRANCH TO CONVERT ANOTHER WORD
745
746
747
748
749
750
751
752 01 0024C 526003B7 A LW,6 DVC
753 01 0024D 69300262 A BNEZ LISTBUT USE LINE PRINTER
754 01 0024E 6A00024F A B TYPEBUT USE TYPEWRITER

```

```

755                                     PAGE
756                                     *   TYPEWRITER OUTPUT ROUTINE
757                                     *
758                                     * THE FOLLOWING CODE TESTS THE FIRST PASS INDICATOR AND THE LINE COUNT.
759                                     * IF FIRST TIME THRU, TITLE AND HEADING ARE PRINTED. IF AT BOTTOM OF
760                                     * PAGE, PAPER IS UPSPACED TO NEW PAGE THEN TITLE AND HEADING ARE PRINTED
761                                     *
762 01 0024F 32C003A0 TYPEOUT LW,12 LINE GET LINE COUNT (INITIALLY=*51)
763 01 00250 32C003A1 LW,13 FIRST GET FIRST PASS INDICATOR
764 01 00251 65C00255 BIR,13 SKIP6 FIRST TIME THRU
765 01 00252 65C0025A BIR,12 MSGOUT LINE COUNT NOT ZERO
766 01 00253 22C001E8 LI,0 DA(DSIXNL)
767 01 00254 6AF00278 BAL,15 PRINT UPSPACE 6 LINES
768 01 00255 22C001E9 SKIP6 LI,0 DA(DTITLE)
769 01 00256 6AF00278 BAL,15 PRINT NEW PAGE TITLE
770 01 00257 22C001EA LI,0 DA(DHEAD)
771 01 00258 6AF00278 BAL,15 PRINT NEW HEADING
772 01 00259 22CFFFC0 A LI,12 *51 RESET LINE COUNT
773 01 0025A 22C001EB MSGOUT LI,0 DA(DSHRTL)
774 01 0025B 321002AF LW,1 DISPLAY++ GET IDENTIFIER
775 01 0025C 6830025E BEZ *+2 REPORT AND NO ERROR
776 01 0025D 200000C1 A AI,0 1
777 01 0025E 6AF00278 BAL,15 PRINT PRINT REPORT OR ERROR
778 01 0025F 35C003A0 STW,12 LINE SAVE NEW LINE COUNT
779 01 00260 35C003A1 STW,13 FIRST SAVE NEW FIRST PASS INDICATOR
780 01 00261 E8C0022A B *EDITLINK

```

```

781                                     PAGE
782                                     *   LINE PRINTER OUTPUT ROUTINE
783                                     *
784                                     * THE FIRST PASS INDICATOR IS MODIFIED AND TESTED. IF ON FIRST PASS,
785                                     * THE PRINTER IS SET TO TOP OF PAGE AND THE TITLE AND HEADING ARE
786                                     * PRINTED. IF NOT ON FIRST PASS, THE PRINTER IS TESTED FOR TOP OF
787                                     * PAGE SO THAT THE TITLE AND HEADING MAY BE PRINTED AT THE TOP OF
788                                     * EVERY PAGE. THE ERROR OR REPORT MESSAGE IS THEN PRINTED.
789                                     *
790 01 00262 32C003A2 LISTOUT LW,13 FIRSTL GET FIRST PASS INDICATOR
791 01 00263 65C00275 BIR,13 TSPSET FIRST PASS
792 01 00264 C2B003B7 TD,11 *JVC
793 01 00265 68800268 BCR,8 *+3
794 01 00266 68400264 BCR,4 *+2 STOP BUSY, TRY AGAIN
795 01 00267 E8C0022A B *EDITLINK NO DEVICE RECOGNITION
796 01 00268 31B003A5 CW,11 MVEBIT TEST FOR PAPER IN MOTION
797 01 00269 69400264 BCS,4 *+5
798 01 0026A 31B003A6 CW,11 TSPBIT TEST FOR TOP OF PAGE
799 01 0026B 6940026E BCR,4 LISTMSG
800 01 0026C 22C001EF LI,0 DA(LTTL)
801 01 0026D 6AF00278 BAL,15 PRINT PRINT TITLE AND HEADING
802 01 0026E 22C001F4 LISTMSG LI,0 DA(LMSG) LOAD RO FOR REPORT
803 01 0026F 32B002AF LW,11 DISPLAY++ TEST ERROR IDENTIFIER
804 01 00270 68300272 BEZ *+2
805 01 00271 200000C2 A AI,0 2 REVISE RO FOR ERROR MESSAGE
806 01 00272 6AF00278 BAL,15 PRINT PRINT MESSAGE
807 01 00273 35C003A2 STW,13 FIRSTL SAVE FIRST PASS INDICATOR
808 01 00274 E8C0022A B *EDITLINK
809 01 00275 22C001EE TSPSET LI,0 DA(LTOP) GO TO TOP OF PAGE AND THEN
810 01 00276 6AF00278 BAL,15 PRINT PRINT TITLE AND HEADING
811 01 00277 68C0026E B LISTMSG NO PRINT THE REPORT

```

812							
813							
814							
815							
816							
817							
818							
819							
820	01	00278	CC0003B7	PRINT	S10,0	*0VC	
821	01	00279	C08003B7	TEST10	T10,11	*0VC	
822	01	0027A	688C027D		BCR,8	*+3	
823	01	0027B	68400279		BCR,4	*+2	BRANCH IF S10P BUSY
824	01	0027C	E800000F A		9	*+15	NO DEVICE RECOGNITION
825	01	0027D	48B003A3		AND,11	BUSYSTAT	
826	01	0027E	318C03A3		CW,11	BUSYSTAT	
827	01	0027F	68300279		BE	TEST10	
828	01	00280	E800000F A		B	*+15	EXIT

829								*B
830								*B
831								*B
832								*B
833								*B
834	01	00281	3200029D	CORSIZE	LW,0	ADDTRAP	SET NON-EXISTENT MEMORY RETURN	*B
835	01	00282	35000068		STW,0	NEARET		*B
836	01	00283	22002000 A		LI,0	X'2000'	GREATER THAN 8K	*B
837	01	00284	221FFFFFF A		LI,1	*1		*B
838	01	00285	22200000 A		LI,2	0		*B
839	01	00286	223FFFFFF4 A		LI,3	*12	MODULE COUNT	*B
840	01	00287	32400000 A	ADDTEST	LW,4	*0	TRY ADDRESS	*B
841	01	00288	3240035A		LW,4	DTA1+1	ADDRESS 0K	*B
842	01	00289	35400000 A		STW,4	*0	STORE STACK	*B
843	01	0028A	32400359		LW,4	DTA1		*B
844	01	0028B	35420000 A		STW,4	*0,1		*B
845	01	0028C	6704C299		EXU	STBCNT,2	STORE MODULE COUNT	*B
846	01	0028D	20200001 A		A1,2	1		*B
847	01	0028E	25000001 A		SLS,0	1	DOUBLE TEST ADDRESS	*B
848	01	0028F	68000287		B	ADDTEST		*B
849	01	00290	3200029E	ADDRET	LW,0	ADDTRAP+1	RESTORE NON-EXISTENT MEMORY RETURN	*B
850	01	00291	35000068		STW,0	NEARET		*B
851	01	00292	32400359		LW,4	DTA1	STORE P19 & P20 STACKS	*B
852	01	00293	354007FF A		STW,4	X'7FFF'		*B
853	01	00294	35400FFF A		STW,4	X'FFFF'		*B
854	01	00295	3240035A		LW,4	DTA1+1		*B
855	01	00296	35400800 A		STW,4	X'800'		*B
856	01	00297	35401000 A		STW,4	X'1000'		*B
857	01	00298	E800000F A		B	*+15		*B
858								*B
859								*B
860								*B
861	01	00299	353007BE	STBCNT	STW,3	DECP18		*B
862	01	0029A	353007CA		STW,3	DECP17		*B
863	01	0029B	353007D6		STW,3	DECP16		*B
864	01	0029C	353007E2		STW,3	DECP15		*B
865								*B

```

866 *
867 *
868 01 0029D 0F0003C2 ADDTRAP XPSD,0 SIZRET
869 01 0029E 0F0003BE XPSD,0 RETURN

```

*B
*B
*B
*B

```

870 PAGE
871 *
872 * *****CONSTANTS AND WORKING STORAGE*****
873 *
874 * MODULE UNDER TEST IS MOVED TO THIS TABLE BEFORE USING
875 *
876 01 0029F 00000000 A TABLE DATA 0,0,0,0,0,0,0,0,0,0,0
      01 002A0 00000000 A
      01 002A1 00000000 A
      01 002A2 00000000 A
      01 002A3 00000000 A
      01 002A4 00000000 A
      01 002A5 00000000 A
      01 002A6 00000000 A
      01 002A7 00000000 A
      01 002A8 00000000 A
      01 002A9 00000000 A
      01 002AA 00000000 A

877 *
878 * TABLE WHERE CONTENTS OF REGISTERS R1 THRU R8 ARE STORED
879 *
880 01 002AB 00000000 A DISPLAY DATA 0,0,0,0,0,0,0,0
      01 002AC 00000000 A
      01 002AD 00000000 A
      01 002AE 00000000 A
      01 002AF 00000000 A
      01 002B0 00000000 A
      01 002B1 00000000 A
      01 002B2 00000000 A

881 *
882 01 002AB SAVE EQU DISPLAY
883 01 002AC ERRORS EQU DISPLAY+1
884 01 002AD PASSES EQU DISPLAY+2

```



```

885
886
887 01 002B3 00000000 A
      01 002B4 00000000 A
      01 002B5 00000000 A
      01 002B6 00000000 A
      01 002B7 00000000 A
      01 002B8 00000000 A
      01 002B9 00000000 A
      01 002BA 00000000 A
      01 002BB 00000000 A
      01 002BC 00000000 A
      01 002BD 00000000 A
      01 002BE 00000000 A
      01 002BF 00000000 A
      01 002C0 00000000 A
      01 002C1 00000000 A
      01 002C2 00000000 A

888
889 01 002C3 00000000 A
      01 002C4 00000000 A
      01 002C5 00000000 A
      01 002C6 00000000 A
      01 002C7 00000000 A
      01 002C8 00000000 A
      01 002C9 00000000 A
      01 002CA 00000000 A
      01 002CB 00000000 A
      01 002CC 00000000 A
      01 002CD 00000000 A
      01 002CE 00000000 A
      01 002CF 00000000 A
      01 002D0 00000000 A
      01 002D1 00000000 A
      01 002D2 00000000 A

```

PAGE
* TABLE USED TO STORE REGISTER CONTENTS PRIOR TO TEST
RTABLIN DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

PAGE
* TABLE USED TO STORE EXPECTED REGISTER VALUES
RTABLOUT DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

```

890
891
892 01 002D3 00000000 A
      01 002D4 00000000 A
      01 002D5 00000000 A
      01 002D6 00000000 A
      01 002D7 00000000 A
      01 002D8 00000000 A
      01 002D9 00000000 A
      01 002DA 00000000 A
      01 002DB 00000000 A
      01 002DC 00000000 A
      01 002DD 00000000 A
      01 002DE 00000000 A
      01 002DF 00000000 A
      01 002E0 00000000 A
      01 002E1 00000000 A
      01 002E2 00000000 A

893
894 01 002E3 00000000 A
      01 002E4 00000000 A
      01 002E5 00000000 A
      01 002E6 00000000 A
      01 002E7 00000000 A
      01 002E8 00000000 A
      01 002E9 00000000 A
      01 002EA 00000000 A
      01 002EB 00000000 A
      01 002EC 00000000 A
      01 002ED 00000000 A
      01 002EE 00000000 A
      01 002EF 00000000 A
      01 002F0 00000000 A
      01 002F1 00000000 A
      01 002F2 00000000 A

```

PAGE
* TABLE USED TO STORE UP TO 16 MEMORY OPERANDS USED BY INSTRUCTION
RTABLIN DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

PAGE
* TABLE USED TO STORE UP TO 16 MEMORY RESULTS EXPECTED AFTER TEST
RTABLOUT DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

```

895                                     PAGE
896                                     *
897                                     * TABLE WHERE REGISTER RESULTS ARE STORED AFTER TEST
898                                     *
899 01 002F3 00000000 A  RESULT DATA  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
01 002F4 00000000 A
01 002F5 00000000 A
01 002F6 00000000 A
01 002F7 00000000 A
01 002F8 00000000 A
01 002F9 00000000 A
01 002FA 00000000 A
01 002FB 00000000 A
01 002FC 00000000 A
01 002FD 00000000 A
01 002FE 00000000 A
01 002FF 00000000 A
01 00300 00000000 A
01 00301 00000000 A
01 00302 00000000 A
    
```

```

900                                     PAGE
901                                     *
902                                     *
903                                     * TABLE OF MEMORY OPERANDS
904                                     * INSTRUCTION UNDER TEST OPERATES ON DATA IN THIS AREA
905                                     *
906                                     *
907 01 00304 00000000 A  MEMORY DATA  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
01 00305 00000000 A
01 00306 00000000 A
01 00307 00000000 A
01 00308 00000000 A
01 00309 00000000 A
01 0030A 00000000 A
01 0030B 00000000 A
01 0030C 00000000 A
01 0030D 00000000 A
01 0030E 00000000 A
01 0030F 00000000 A
01 00310 00000000 A
01 00311 00000000 A
01 00312 00000000 A
01 00313 00000000 A
908 01 00314 00000000 A  SP      PZE,0  0      STACK POINTER DOUBLE WORD
909 01 00315 00000000 A      PZE
910 01 00316 00000000 A  IA      PZE      INDIRECT ADDRESS LOCATION
911 01 00317 00000000 A  IASP   PZE      STACK POINTER INDIRECT ADDRESS
    
```

912				PAGE		
913	01 00318	15151515 A	TT	TEXT	'NNNNNNN	SUFFIX'
	01 00319	15151540 A				
	01 0031A	40404040 A				
	01 0031B	40404040 A				
	01 0031C	40404040 A				
	01 0031D	40404040 A				
	01 0031E	40404040 A				
	01 0031F	40404040 A				
	01 00320	40404040 A				
	01 00321	4040E2E4 A				
	01 00322	C6C6C9E7 A				
914	01 00323	40C5D9D9 A		TEXT	' ERROR DISPLAY'	
	01 00324	D6D940C4 A				
	01 00325	C9E2D7D3 A				
	01 00326	C1E84040 A				
915	01 00327	15404040 A	H0G	TEXT	'N LIST ERRORS PASSES INST'	
	01 00328	40D3C9E2 A				
	01 00329	E3404040 A				
	01 0032A	4040C5D9 A				
	01 0032B	D9D6D9E2 A				
	01 0032C	40404040 A				
	01 0032D	D7C1E2E2 A				
	01 0032E	C5E24040 A				
	01 0032F	404040C9 A				
	01 00330	D5E2E340 A				
916	01 00331	4040C9C4 A		TEXT	' IDENTIFIER IS SHOULD BE DIFFN'	
	01 00332	C5D5E3C9 A				
	01 00333	C6C9C5D9 A				
	01 00334	40404040 A				
	01 00335	C9E24040 A				
	01 00336	4040E2C8 A				
	01 00337	D6E4D3C4 A				
	01 00338	40C2C540 A				
	01 00339	404040C4 A				
	01 0033A	C9C6C615 A				

917				PAGE		
918						
919						
920						
921	01 00333	40404015 A		IMAGE	DATA	X'40404015'
922	01 0033C	00000000 A			DATA	0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
	01 0033D	00000000 A				
	01 0033E	00000000 A				
	01 0033F	00000000 A				
	01 00340	00000000 A				
	01 00341	00000000 A				
	01 00342	00000000 A				
	01 00343	00000000 A				
	01 00344	00000000 A				
	01 00345	00000000 A				
	01 00346	00000000 A				
	01 00347	00000000 A				
	01 00348	00000000 A				
	01 00349	00000000 A				
	01 0034A	00000000 A				
	01 0034B	00000000 A				
	01 0034C	00000000 A				
	01 0034D	00000000 A				
	01 0034E	00000000 A				
	01 0034F	00000000 A				
923	01 00350	40404040 A	BLANK	DATA	X'40404040'	BLANK EBCDIC CHARACTERS
924						
925	01 00351	000C0804 A	DTAO	DATA	X'000C0804'	DATA TABLE 0
926	01 00352	C10D0905 A		DATA	X'C10D0905'	
927	01 00353	C20E0A06 A		DATA	X'C20E0A06'	
928	01 00354	C30F0B07 A		DATA	X'C30F0B07'	
929	01 00355	C4000C08 A		DATA	X'C4000C08'	
930	01 00356	C5010D09 A		DATA	X'C5010D09'	
931	01 00357	C6020E0A A		DATA	X'C6020E0A'	
932	01 00358	C7030F0B A		DATA	X'C7030F0B'	

Address	Op	OpCode	OpName	Page	Comments
985					
986					
987	01	00399	00000000 A CNT1CP	DATA	0
988	01	0039A	00000000 A CNT2CP	DATA	0
989	01	0039B	00000000 A CNT3CP	DATA	0
990	01	0039C	00000000 A CNT4CP	DATA	0
991	01	0039D	00000000 A INTR	DATA	0
992	01	0039E	0000F000 A INTRC	DATA	X'0000F000' CNT PULSE INTR, ARM AND ENABLE BITS
993	01	0039F	0000F000 A CNTRMASK	DATA	X'0000F000'
994	01	003A0	FFFFFFFFC A LINE	DATA	*51 LINE COUNTER
995	01	003A1	FFFFFFFFE A FIRST	DATA	*2 FIRST PASS INDICATOR
996	01	003A2	FFFFFFFFE A FIRSTL	DATA	*2
997	01	003A3	60000000 A BUSYSTAT	DATA	X'60000000'
998	01	003A4	10000000 A AUTOSTAT	DATA	X'10000000' AUTOMATIC STATUS BIT
999	01	003A5	08000000 A MBVEBIT	DATA	X'08000000' PAPER IN MOTION STATUS BIT
1000	01	003A6	10000000 A TSPBIT	DATA	X'10000000' TOP OF PAGE STATUS BIT
1001	01	003A7	F1c10000 A TSPFORM	DATA	X'F1c10000' LINE PRINTER FORMAT CHARACTERS
1002	01	003A8	00000000 A NEXT	DATA	0 ADDRESS OF NEXT MODULE SAVED HERE
1003	01	003A9	60000100 BT0100	B	START
1004	01	003AA	0F0003BE XPSD	XPSD,0	RETJRN
1005	01	003AB	FFFF0000 A WOT15	DATA	X'FFFF0000' MASK
1006	01	003AC	0001FFFF A W15T31	DATA	X'1FFFF' MASK
1007	01	003AD	FF3FFFFF A LINKAD	DATA	X'FF3FFFFF' MASK
1008	01	003AE	FFF00000 A CBND	DATA	X'FFF00000' MASK
1009	01	003AF	00640000 A NUM	GEN,16,16	100,0
1010	01	003B0	000001AC L0CA0D	PZE,0	L0C
1011	01	003B1	00000300 A INDA	PZE,0	MEMORY
1012	01	003B2	00000314 A INDASP	PZE,0	SP
1013	01	003B3	351002F4 A MBVER	STW,1	RRESULT+1
1014	01	003B4	00100001 A MBVERMBD	DATA	X'00100001'
1015	01	003B5	36000303 A MBVERFIN	GEN,16,16	X'3600',RRESULT+16
1016	01	003B6	00000000 A PSW2	PZE	
1017	01	003B7	00000001 A DVC	DATA	1 OUTPUT DEVICE ADDRESS

1018					
1019					
1020	01	003B8	0000010F REPEAT	PZE,0	CYCLE
1021	01	003B9	00000000 A	PZE	0
1022	01	003BA	00000000 A TEMP	PZE	
1023	01	003BB	00000000 A	PZE	
1024	01	003BC	00000000 A PSW1	PZE	
1025	01	003BD	00000000 A	PZE	
1026	01	003BE	00000000 A RETURN	PZE	
1027	01	003BF	00000000 A	PZE	
1028	01	003C0	000000B3	PZE,0	DEADST0P
1029	01	003C1	00000000 A	PZE	
1030	01	003C2	00000000 A SIZRET	PZE	
1031	01	003C3	00000000 A	PZE	
1032	01	003C4	0000029C	PZE,0	ADDRET
1033	01	003C5	00000000 A	PZE	
1034					
1035					
1036					
1037	01	003C6	100001AC INSTID	GEN,4,28	1,L0C INSTRUCTION IDENTIFIER
1038	01	003C7	200001AD XPSDID	GEN,4,28	2,L0C+1 L0C+1 IDENTIFIER
1039	01	003C8	30000316 IAID	GEN,4,28	3,1A INDIRECT ADDRESS IDENTIFIER
1040	01	003C9	50000001 A PSDWID	DATA	X'50000001' PSDW IDENTIFIER
1041	01	003CA	60000000 A REGID	DATA	X'60000000' REGISTER IDENTIFIER
1042	01	003CB	70000304 A MEMID	GEN,4,28	7,MEMORY MEMORY WORD IDENTIFIER
1043	01	003CC	80000314 A SPID	GEN,4,28	8,SP STACK POINTER IDENTIFIER
1044					
1045	01	003CD	60000010 A REGIDFIN	DATA	X'60000010' IDENTIFIES END OF REGISTER BLOCK
1046	01	003CE	70000314 A MEMIDFIN	GEN,4,28	7,SP IDENTIFIES END OF MEMORY TABLE
1047					
1048	01	003CF	00000000 A RPTIND	DATA	0 ERROR REPORTED INDICATOR

		PAGE			
1049					
1050					
1051			*		
1052			*	COMMAND DOUBLEWORDS FOR TYPEOUT	
1053			*		
1054					
1055	01 003D0	05000C60	DSIXNL	BOUND 8 GEN,8,24 5,BA(TTL)	SIX NEW LINE CHARACTERS
1056	01 003D1	0800006A		GEN,8,24 8,6	
1057	01 003D2	05000C60	DTITLE	GEN,8,24 5,BA(TTL)	TITLE
1058	01 003D3	0800003A		GEN,8,24 8,58	
1059	01 003D4	05000C9C	DHEAD	GEN,8,24 5,BA(HDG)	HEADING
1060	01 003D5	08000060		GEN,8,24 8,80	
1061	01 003D6	05000CEC	DSHRTL	GEN,8,24 5,BA(IMAGE)	SHORT LINE (4 WORDS)
1062	01 003D7	0800002C		GEN,8,24 8,44	
1063	01 003D8	05000CEC		GEN,8,24 5,BA(IMAGE)	LONG LINE (8 WORDS)
1064	01 003D9	08000054		GEN,8,24 8,84	
1065	01 003DA	05000E14	CNTRDW	GEN,8,24 5,BA(CNTRMSG)	COMMAND DOUBLEWORD FOR
1066	01 003DB	0200004E	DATA	X'0200004E'	COUNTER INTERRUPT MESSAGE

		PAGE			
1067					
1068					
1069			*		
1070			*	COMMAND DOUBLEWORDS FOR LINE PRINTER	
1071			*		
1072	01 003DC	03000E9C	LTOP	GEN,8,24 3,BA(TOPFORM)	TOP OF PAGE ORDER
1073	01 003DD	28000001		DATA X'28000001'	COMMAND CHAIN
1074	01 003DE	01000040	LTTL	GEN,8,24 1,BA(BLANK)	SKIP
1075	01 003DF	83000006		DATA X'83000006'	DATA CHAIN
1076	01 003E0	01000067		GEN,8,24 1,BA(TTL)+7	TITLE
1077	01 003E1	2A000025		DATA X'2A000025'	COMMAND CHAIN
1078	01 003E2	01000040	LHEAD	GEN,8,24 1,BA(BLANK)	SKIP
1079	01 003E3	83000006		DATA X'83000006'	DATA CHAIN
1080	01 003E4	0100009D		GEN,8,24 1,BA(HDG)+1	HEADING
1081	01 003E5	2A00004E		DATA X'2A00004E'	COMMAND CHAIN
1082	01 003E6	03000E9D		GEN,8,24 3,BA(TOPFORM)+1	SPACE 1 LINE
1083	01 003E7	08000001		DATA X'08000001'	
1084	01 003E8	01000040	LMSG	GEN,8,24 1,BA(BLANK)	NORMAL REPORT ORDERS
1085	01 003E9	83000006		DATA X'83000006'	SKIP, DATA CHAIN
1086	01 003EA	010000F0		GEN,8,24 1,BA(IMAGE+1)	
1087	01 003EB	0A000028		DATA X'0A000028'	
1088	01 003EC	01000040		GEN,8,24 1,BA(BLANK)	ERROR REPORT ORDERS
1089	01 003ED	83000006		DATA X'83000006'	SKIP, DATA CHAIN
1090	01 003EE	010000F0		GEN,8,24 1,BA(IMAGE+1)	
1091	01 003EF	0A000050		DATA X'0A000050'	

				PAGE			
1148				*****			LM
1149				*			
1150				*		CC=1	
1151				*		LOAD MEMORY+2 INTO R4	
1152	01	00400	FFFFFFFF A	DATA	=8	COUNT	
1153	01	00401	24400306	LM,4	MEMORY+2	INSTRUCTION	
1154	01	00402	10000151	K	1,0,0,0,SETPSW	PSW1 IN	
1155	01	00403	100001AE	K	1,0,0,0,LBC+2	PSW1 OUT	
1156	01	00404	03840000 N	P	ZERBS,0,0	R IN	
1157	01	00405	0359FF04 N	P	DTA1,-1,4	R OUT	
1158	01	00406	0359FF02 N	P	DTA1,-1,2	MEM IN	
1159	01	00407	0359FF02 N	P	DTA1,-1,2	MEM OUT	
1160				*****			LM
1161				*		CC=0 (COUNT OF 16)	
1162				*		LOAD ALL REGISTERS	
1163				*		BEGINNING WITH R0	
1164	01	00408	FFFFFFFF A	DATA	=8	COUNT	
1165	01	00409	2A000304	LM,0	MEMORY	INSTRUCTION	
1166	01	0040A	00000151	K	0,0,0,0,SETPSW	PSW1 IN	
1167	01	0040B	000001AE	K	0,0,0,0,LBC+2	PSW1 OUT	
1168	01	0040C	03840000 N	P	ZERBS,0,0	R IN	
1169	01	0040D	0359F000 N	P	DTA1,-16,0	R OUT	
1170	01	0040E	0359F000 N	P	DTA1,-16,0	MEM IN	
1171	01	0040F	0359F000 N	P	DTA1,-16,0	MEM OUT	
1172				*****			LM
1173				*		CC=0 (COUNT OF 16)	
1174				*		LOAD ALL REGISTERS	
1175				*		BEGINNING WITH R9	
1176	01	00410	FFFFFFFF A	DATA	=8	COUNT	
1177	01	00411	2A900304	LM,9	MEMORY	INSTRUCTION	
1178	01	00412	00000151	K	0,0,0,0,SETPSW	PSW1 IN	
1179	01	00413	000001AE	K	0,0,0,0,LBC+2	PSW1 OUT	
1180	01	00414	03840000 N	P	ZERBS,0,0	R IN	
1181	01	00415	0359F009 N	P	DTA1,-16,9	R OUT	
1182	01	00416	0359F000 N	P	DTA1,-16,0	MEM IN	
1183	01	00417	0359F000 N	P	DTA1,-16,0	MEM OUT	

				PAGE			
1184				*****			LM
1185				*			
1186				*		CC=2 INDEXED	
1187				*		LOAD R5,R6 INDEX REGISTER=R4	
1188				*			
1189	01	00418	FFFFFFFF A	DATA	=8	COUNT	
1190	01	00419	2A500304	LM,5	MEMORY,4	INSTRUCTION	
1191	01	0041A	20000151	K	2,0,0,0,SETPSW	PSW1 IN	
1192	01	0041B	200001AE	K	2,0,0,0,LBC+2	PSW1 OUT	
1193	01	0041C	036BF004 N	P	DTA2+2,-1,4	R IN	
1194	01	0041D	036BF004 N	P	DTA2+2,-3,4	R OUT	
1195	01	0041E	036CF002 N	P	DTA2+3,-2,2	MEM IN	
1196	01	0041F	036CF002 N	P	DTA2+3,-2,2	MEM OUT	
1197				*****			LM
1198				*		CC=4 INDIRECTLY ADDRESSED	
1199				*		LOAD R7=R10	
1200				*			
1201	01	00420	FFFFFFFF A	DATA	=8	COUNT	
1202	01	00421	AA200316	LM,7	*1A	INSTRUCTION	
1203	01	00422	40000151	K	4,0,0,0,SETPSW	PSW1 IN	
1204	01	00423	400001AE	K	4,0,0,0,LBC+2	PSW1 OUT	
1205	01	00424	03840000 N	P	ZERBS,0,0	R IN	
1206	01	00425	0359F007 N	P	DTA1,-4,7	R OUT	
1207	01	00426	0359F000 N	P	DTA1,-4,0	MEM IN	
1208	01	00427	0359F000 N	P	DTA1,-4,0	MEM OUT	
1209				*****			LM
1210				*		CC=8 INDIRECTLY ADDRESSED, INDEXED	
1211				*		LOAD R2=R9 INDEX REGISTER=R1	
1212	01	00428	FFFFFFFF A	DATA	=8	COUNT	
1213	01	00429	AA200316	LM,2	*1A,1	INSTRUCTION	
1214	01	0042A	87300151	K	8,7,3,0,SETPSW	PSW1 IN	
1215	01	0042B	873001AE	K	8,7,3,0,LBC+2	PSW1 OUT	
1216	01	0042C	0369F001 N	P	DTA2,-1,1	R IN	
1217	01	0042D	0369F001 N	P	DTA2,-9,1	R OUT	
1218	01	0042E	036AF001 N	P	DTA2+1,-8,1	MEM IN	
1219	01	0042F	036AF001 N	P	DTA2+1,-8,1	MEM OUT	

		PAGE		
1220				STM
1221				
1222				CC#1
1223				STORE R4 IN MEMORY+2
1224	01 00430	FFFFFFF8 A	DATA #8	COUNT
1225	01 00431	23400306	STM,4 MEMORY+2	INSTRUCTION
1226	01 00432	10000151	K 1,0,0,SETPSW	PSW1 IN
1227	01 00433	100001AE	K 1,0,0,LBC+2	PSW1 OUT
1228	01 00434	0359FF04 N	P DT11,=1,4	R IN
1229	01 00435	0359FF04 N	P DT11,=1,4	R OUT
1230	01 00436	03840000 N	P ZEROS,0,0	MEM IN
1231	01 00437	0359FF02 N	P DT11,=1,2	MEM OUT
1232				STM
1233				CC#0 (COUNT OF 16)
1234				STORE ALL REGISTERS
1235				BEGINNING WITH R0
1236	01 00438	FFFFFFF8 A	DATA #8	COUNT
1237	01 00439	23000304	STM,0 MEMORY	INSTRUCTION
1238	01 0043A	00000151	K 0,0,0,SETPSW	PSW1 IN
1239	01 0043B	000001AE	K 0,0,0,LBC+2	PSW1 OUT
1240	01 0043C	0359F000 N	P DT11,=16,0	R IN
1241	01 0043D	0359F000 N	P DT11,=16,0	R OUT
1242	01 0043E	03840000 N	P ZEROS,0,0	MEM IN
1243	01 0043F	0359F000 N	P DT11,=16,0	MEM OUT
1244				STM
1245				CC#0 (COUNT OF 16)
1246				STORE ALL REGISTERS
1247				BEGINNING WITH R9
1248	01 00440	FFFFFFF8 A	DATA #8	COUNT
1249	01 00441	23900304	STM,9 MEMORY	INSTRUCTION
1250	01 00442	00000151	K 0,0,0,SETPSW	PSW1 IN
1251	01 00443	000001AE	K 0,0,0,LBC+2	PSW1 OUT
1252	01 00444	0359F000 N	P DT11,=16,0	R IN
1253	01 00445	0359F000 N	P DT11,=16,0	R OUT
1254	01 00446	03840000 N	P ZEROS,0,0	MEM IN
1255	01 00447	0359F007 N	P DT11,=16,7	MEM OUT

		PAGE		
1256				STM
1257				
1258				CC#2 INDEXED
1259				STORE R5,R6 INDEX REGISTER=R3
1260				
1261	01 00448	FFFFFFF8 A	DATA #8	COUNT
1262	01 00449	23560304	STM,5 MEMORY,3	INSTRUCTION
1263	01 0044A	20000151	K 2,0,0,SETPSW	PSW1 IN
1264	01 0044B	200001AE	K 2,0,0,LBC+2	PSW1 OUT
1265	01 0044C	0363FC03 N	P DT2+2,=4,3	R IN
1266	01 0044D	0363FC03 N	P DT2+2,=4,3	R OUT
1267	01 0044E	03840000 N	P ZEROS,0,0	MEM IN
1268	01 0044F	0363FE02 N	P DT2+4,=2,2	MEM OUT
1269				STM
1270				CC#4 INDIRECTLY ADDRESSED
1271				STORE R7,R10
1272				
1273	01 00450	FFFFFFF8 A	DATA #8	COUNT
1274	01 00451	A3700316	STM,7 #1A	INSTRUCTION
1275	01 00452	40000151	K 4,0,0,SETPSW	PSW1 IN
1276	01 00453	400001AE	K 4,0,0,LBC+2	PSW1 OUT
1277	01 00454	0359FC07 N	P DT11,=4,7	R IN
1278	01 00455	0359FC07 N	P DT11,=4,7	R OUT
1279	01 00456	03840000 N	P ZEROS,0,0	MEM IN
1280	01 00457	0359FC00 N	P DT11,=4,0	MEM OUT
1281				STM
1282				CC#8 INDIRECTLY ADDRESSED, INDEXED
1283				STORE R2-R9 INDEX REGISTER=R1
1284	01 00458	FFFFFFF8 A	DATA #8	COUNT
1285	01 00459	A3220316	STM,2 #1A,1	INSTRUCTION
1286	01 0045A	87300151	K 8,7,3,SETPSW	PSW1 IN
1287	01 0045B	873001AE	K 8,7,3,LBC+2	PSW1 OUT
1288	01 0045C	0369F701 N	P DT2,=9,1	R IN
1289	01 0045D	0369F701 N	P DT2,=9,1	R OUT
1290	01 0045E	03840000 N	P ZEROS,0,0	MEM IN
1291	01 0045F	036AF801 N	P DT2+1,=8,1	MEM OUT

			PAGE			
1345						
1346						
1347						
1348						
1349						
1350	01	00484	FFFFFFF4 A	DATA	=12	COUNT
1351	01	00485	13400314	MSP,4	SP	INSTRUCTION
1352	01	00486	F0000151	K	15,0,0,SETPSW	PSW1 IN
1353	01	00487	400001AE	K	4,0,0,LBC+2	PSW1 OUT
1354	01	00488	037FFF04 N	P	DTA3+1,=1,4	R IN
1355	01	00489	037FFF04 N	P	DTA3+1,=1,4	R OUT
1356	01	0048A	0359F000 N	P	DTA1,=16,0	MEM IN
1357	01	0048B	0359F000 N	P	DTA1,=16,0	MEM OUT
1358	01	0048C	0000030C	DATA	MEMORY+8	STACK POINTER
1359	01	0048D	8001FFFFE A	DATA	X'8001FFFFE'	DOUBLEWORD IN
1360	01	0048E	0000030D	DATA	MEMORY+9	STACK POINTER
1361	01	0048F	8000FFFFF A	DATA	X'8000FFFFF'	DOUBLEWORD OUT
1362						
1363						
1364						
1365						
1366	01	00490	FFFFFFF4 A	DATA	=12	COUNT
1367	01	00491	13000314	MSP,0	SP	INSTRUCTION
1368	01	00492	00000151	K	0,0,0,SETPSW	PSW1 IN
1369	01	00493	100001AE	K	1,0,0,LBC+2	PSW1 OUT
1370	01	00494	037DFF00 N	P	DTA3=1,=1,0	R IN
1371	01	00495	037DFF00 N	P	DTA3=1,=1,0	R OUT
1372	01	00496	0359F000 N	P	DTA1,=16,0	MEM IN
1373	01	00497	0359F000 N	P	DTA1,=16,0	MEM OUT
1374	01	00498	0000030C	DATA	MEMORY+8	STACK POINTER
1375	01	00499	FFFE8001 A	DATA	X'FFFE8001'	DOUBLEWORD IN
1376	01	0049A	0000030B	DATA	MEMORY+7	STACK POINTER
1377	01	0049B	FFFF8000 A	DATA	X'FFFF8000'	DOUBLEWORD OUT

			PAGE			
1378						
1379						
1380						
1381						
1382	01	0049C	FFFFFFF4 A	DATA	=12	COUNT
1383	01	0049D	13F00314	MSP,15	SP	INSTRUCTION
1384	01	0049E	00000151	K	13,0,0,SETPSW	PSW1 IN
1385	01	0049F	200001AE	K	2,0,0,LBC+2	PSW1 OUT
1386	01	004A0	0382FF0F N	P	DTA3+4,=1,15	R IN
1387	01	004A1	0382FF0F N	P	DTA3+4,=1,15	R OUT
1388	01	004A2	0359F000 N	P	DTA1,=16,0	MEM IN
1389	01	004A3	0359F000 N	P	DTA1,=16,0	MEM OUT
1390	01	004A4	0000030C	DATA	MEMORY+8	STACK POINTER
1391	01	004A5	0750FFFA A	DATA	X'0750FFFA'	DOUBLEWORD IN
1392	01	004A6	0000030C	DATA	MEMORY+8	STACK POINTER
1393	01	004A7	0750FFFA A	DATA	X'0750FFFA'	DOUBLEWORD OUT
1394						
1395						
1396						
1397	01	004A8	FFFFFFF4 A	DATA	=12	COUNT
1398	01	004A9	13B00314	MSP,11	SP	INSTRUCTION
1399	01	004AA	00000151	K	13,0,0,SETPSW	PSW1 IN
1400	01	004AB	200001AE	K	2,0,0,LBC+2	PSW1 OUT
1401	01	004AC	037AFF0B N	P	DTA3=4,=1,11	R IN
1402	01	004AD	037AFF0B N	P	DTA3=4,=1,11	R OUT
1403	01	004AE	0359F000 N	P	DTA1,=16,0	MEM IN
1404	01	004AF	0359F000 N	P	DTA1,=16,0	MEM OUT
1405	01	004B0	0000030C	DATA	MEMORY+8	STACK POINTER
1406	01	004B1	05008004 A	DATA	X'05008004'	DOUBLEWORD IN
1407	01	004B2	0000030C	DATA	MEMORY+8	STACK POINTER
1408	01	004B3	05008004 A	DATA	X'05008004'	DOUBLEWORD OUT

1409				PAGE		
1410				*****		MSP
1411				*		MODIFIER=4; SPACE COUNT UNDERFLOW
1412				*		TRAP INHIBITED BY TS
1413	01	00434	FFFFFFF4 A	DATA	=12	COUNT
1414	01	00435	13300314	MSP,3	SP	INSTRUCTION
1415	01	00436	70000151	K	7,0,0,SETPSW	PSW1 IN
1416	01	00437	800001AE	K	8,0,0,LBC+2	PSW1 OUT
1417	01	00438	0381FF03 N	P	DTA3+3,-1,3	R IN
1418	01	00439	0381FF03 N	P	DTA3+3,-1,3	R OUT
1419	01	0043A	03840000 N	P	ZEROS,0,0	MEM IN
1420	01	0043B	03840000 N	P	ZEROS,0,0	MEM OUT
1421	01	0043C	0000030C	DATA	MEMBRY+8	STACK POINTER
1422	01	0043D	80020050 A	DATA	X'80020050'	DOUBLEWORD IN
1423	01	0043E	0000030C	DATA	MEMBRY+8	STACK POINTER
1424	01	0043F	80020050 A	DATA	X'80020050'	DOUBLEWORD OUT
1425				*****		MSP
1426				*		MODIFIER=4; SPACE COUNT OVERFLOW
1427				*		TRAP INHIBITED BY TS
1428	01	004C0	FFFFFFF4 A	DATA	=12	COUNT
1429	01	004C1	13700314	MSP,7	SP	INSTRUCTION
1430	01	004C2	70000151	K	7,0,0,SETPSW	PSW1 IN
1431	01	004C3	800001AE	K	8,0,0,LBC+2	PSW1 OUT
1432	01	004C4	0373FF07 N	P	DTA3+3,-1,7	R IN
1433	01	004C5	0373FF07 N	P	DTA3+3,-1,7	R OUT
1434	01	004C6	03840000 N	P	ZEROS,0,0	MEM IN
1435	01	004C7	03840000 N	P	ZEROS,0,0	MEM OUT
1436	01	004C8	0000030C	DATA	MEMBRY+8	STACK POINTER
1437	01	004C9	FFFF0050 A	DATA	X'FFFF0050'	DOUBLEWORD IN
1438	01	004CA	0000030C	DATA	MEMBRY+8	STACK POINTER
1439	01	004CB	FFFF0050 A	DATA	X'FFFF0050'	DOUBLEWORD OUT

1440				PAGE		
1441				*****		MSP
1442				*		MODIFIER= MAX POSITIVE VALUE
1443				*		WORD COUNT OVERFLOWS BY 1
1444				*		TRAP
1445	01	004CC	FFFFFFF4 A	DATA	=12	COUNT
1446	01	004CD	13E00314	MSP,14	SP	INSTRUCTION
1447	01	004CE	000000AF	K	0,0,0,SLSW	PSW1 IN
1448	01	004CF	00000080	K	0,0,0,SLRET+1	PSW1 OUT
1449	01	004D0	0383FF0E N	P	DTA3+5,-1,14	R IN
1450	01	004D1	0383FF0E N	P	DTA3+5,-1,14	R OUT
1451	01	004D2	0359F000 N	P	DTA1,-16,0	MEM IN
1452	01	004D3	0359F000 N	P	DTA1,-16,0	MEM OUT
1453	01	004D4	00000304	DATA	MEMBRY	STACK POINTER
1454	01	004D5	FFFF0001 A	DATA	X'FFFF0001'	DOUBLEWORD IN
1455	01	004D6	00000304	DATA	MEMBRY	STACK POINTER
1456	01	004D7	FFFF0001 A	DATA	X'FFFF0001'	DOUBLEWORD OUT
1457				*****		MSP
1458				*		MODIFIER= MAX NEGATIVE VALUE
1459				*		WORD COUNT UNDERFLOWS BY 1
1460				*		TRAP
1461	01	004D8	FFFFFFF4 A	DATA	=12	COUNT
1462	01	004D9	13000314	MSP,13	SP	INSTRUCTION
1463	01	004DA	F73000AF	K	15,7,3,SLSW	PSW1 IN
1464	01	004DB	F7300080	K	15,7,3,SLRET+1	PSW1 OUT
1465	01	004DC	0379FF0D N	P	DTA3+5,-1,13	R IN
1466	01	004DD	0379FF0D N	P	DTA3+5,-1,13	R OUT
1467	01	004DE	0359F000 N	P	DTA1,-16,0	MEM IN
1468	01	004DF	0359F000 N	P	DTA1,-16,0	MEM OUT
1469	01	004E0	00000304	DATA	MEMBRY	STACK POINTER
1470	01	004E1	80007FFF A	DATA	X'80007FFF'	DOUBLEWORD IN
1471	01	004E2	00000304	DATA	MEMBRY	STACK POINTER
1472	01	004E3	80007FFF A	DATA	X'80007FFF'	DOUBLEWORD OUT

			PAGE			
1473						MSP
1474						
1475						
1476						
1477						
1478	01	004E4	FFFFFFF4 A	DATA	=12	
1479	01	004E5	13900314	MSP,9	SP	
1480	01	004E6	373000AF	K	11,7,3,SLSW	
1481	01	004E7	37300080	K	11,7,3,SLRET+1	
1482	01	004E8	0383FF09 N	P	DTA3+5,-1,9	
1483	01	004E9	0383FF09 N	P	DTA3+5,-1,9	
1484	01	004EA	0359F000 N	P	DTA1,-16,0	
1485	01	004EB	0359F000 N	P	DTA1,-16,0	
1486	01	004EC	00000304	DATA	MEMORY	
1487	01	004ED	7FFE0000 A	DATA	X'7FFE0000'	
1488	01	004EE	00000304	DATA	MEMORY	
1489	01	004EF	7FFE0000 A	DATA	X'7FFE0000'	
1490						
1491						
1492						
1493						
1494	01	004F0	FFFFFFF4 A	DATA	=12	
1495	01	004F1	13500314	MSP,5	SP	
1496	01	004F2	500000AF	K	5,0,0,SLSW	
1497	01	004F3	50000080	K	5,0,0,SLRET+1	
1498	01	004F4	0379FF05 N	P	DTA3+5,-1,5	
1499	01	004F5	0379FF05 N	P	DTA3+5,-1,5	
1500	01	004F6	03840000 N	P	ZEROS,0,0	
1501	01	004F7	03840000 N	P	ZEROS,0,0	
1502	01	004F8	00000304	DATA	MEMORY	
1503	01	004F9	0000FFFF A	DATA	X'0000FFFF'	
1504	01	004FA	00000304	DATA	MEMORY	
1505	01	004FB	0000FFFF A	DATA	X'0000FFFF'	

			PAGE			
1506						MSP
1507						
1508						
1509						
1510	01	004FC	FFFFFFF4 A	DATA	=12	
1511	01	004FD	138E0312	MSP,8	SP=2,7	
1512	01	004FE	80000151	K	11,0,0,SETPSW	
1513	01	004FF	400001AE	K	4,0,0,LBC+2	
1514	01	00500	037FFE07 N	P	DTA3+1,-2,7	
1515	01	00501	037FFE07 N	P	DTA3+1,-2,7	
1516	01	00502	0359F000 N	P	DTA1,-16,0	
1517	01	00503	0359F000 N	P	DTA1,-16,0	
1518	01	00504	00000304	DATA	MEMORY	
1519	01	00505	00020000 A	DATA	X'00020000'	
1520	01	00506	00000306	DATA	MEMORY+2	
1521	01	00507	00000002 A	DATA	X'00000002'	
1522						
1523						
1524						
1525	01	00508	FFFFFFF4 A	DATA	=12	
1526	01	00509	93100317	MSP,1	*IASP	
1527	01	0050A	00000151	K	0,0,0,SETPSW	
1528	01	0050B	100001AE	K	1,0,0,LBC+2	
1529	01	0050C	037CFE01 N	P	DTA3+2,-2,1	
1530	01	0050D	037CFE01 N	P	DTA3+2,-2,1	
1531	01	0050E	03840000 N	P	ZEROS,0,0	
1532	01	0050F	03840000 N	P	ZEROS,0,0	
1533	01	00510	00000306	DATA	MEMORY+2	
1534	01	00511	00000002 A	DATA	X'00000002'	
1535	01	00512	00000304	DATA	MEMORY	
1536	01	00513	00020000 A	DATA	X'00020000'	

				PAGE			MSP
1537				*****			
1538				*		INDEXED/	INDIRECTLY ADDRESSED
1539				*		MODIFIER=4	INDEX VALUE=8 IN R3
1540							
1541	01	00514	FFFFFFF4 A	DATA	=12	COUNT	
1542	01	00515	93260316	MSP,2	*1A,3	INSTRUCTION	
1543	01	00516	90000151	K	11,0,0,SETPSW	PSW1 IN	
1544	01	00517	400001AE	<	4,0,0,L0C+2	PSW1 OUT	
1545	01	00518	0381FE02 N	P	DTA3+3,*2,2	R IN	
1546	01	00519	0381FE02 N	P	DTA3+3,*2,2	R OUT	
1547	01	0051A	03840000 N	P	ZER0S,0,0	MEM IN	
1548	01	0051B	03840000 N	P	ZER0S,0,0	MEM OUT	
1549	01	0051C	00000304	DATA	MEMBRY	STACK POINTER	
1550	01	0051D	80048000 A	DATA	X'80048000'	DOUBLEWORD IN	
1551	01	0051E	0000030F	DATA	MEMBRY+4	STACK POINTER	
1552	01	0051F	80008004 A	DATA	X'80008004'	DOUBLEWORD OUT	

				PAGE			PSW
1553				*****			
1554				*		PUSH WORD INTO STACK FROM R6,	
1555				*		NO OVERFLOW OR UNDERFLOW	
1556							
1557	01	00520	FFFFFFF4 A	DATA	=12	COUNT	
1558	01	00521	03600314	PSW,6	SP	INSTRUCTION	
1559	01	00522	F7300151	<	15,7,3,SETPSW	PSW1 IN	
1560	01	00523	073001AE	<	0,7,3,L0C+2	PSW1 OUT	
1561	01	00524	0359F000 N	P	DTA1,*16,0	R IN	
1562	01	00525	0359F000 N	P	DTA1,*16,0	R OUT	
1563	01	00526	03840000 N	P	ZER0S,0,0	MEM IN	
1564	01	00527	035FFF01 N	P	DTA1*6,*1,1	MEM OUT	
1565	01	00528	00000304	DATA	MEMBRY	STACK POINTER	
1566	01	00529	4000BFFF A	DATA	X'4000BFFF'	DOUBLEWORD IN	
1567	01	0052A	00000305	DATA	MEMBRY+1	STACK POINTER	
1568	01	0052B	3FFFC000 A	DATA	X'3FFFC000'	DOUBLEWORD OUT	
1569				*****			PSW
1570				*		PUSH WORD INTO STACK FROM R5	
1571				*		NO OVERFLOW OR UNDERFLOW, TS SET	
1572				*		SPACE COUNT GOES TO ZERO	
1573	01	0052C	FFFFFFF4 A	DATA	=12	COUNT	
1574	01	0052D	03600314	PSW,5	SP	INSTRUCTION	
1575	01	0052E	97300151	<	11,7,3,SETPSW	PSW1 IN	
1576	01	0052F	473001AE	<	4,7,3,L0C+2	PSW1 OUT	
1577	01	00530	0359F000 N	P	DTA1,*16,0	R IN	
1578	01	00531	0359F000 N	P	DTA1,*16,0	R OUT	
1579	01	00532	03840000 N	P	ZER0S,0,0	MEM IN	
1580	01	00533	035FFF02 N	P	DTA1*5,*1,2	MEM OUT	
1581	01	00534	0000030F	DATA	MEMBRY+1	STACK POINTER	
1582	01	00535	80017FFE A	DATA	X'80017FFE'	DOUBLEWORD IN	
1583	01	00536	00000306	DATA	MEMBRY+2	STACK POINTER	
1584	01	00537	80007FFF A	DATA	X'80007FFF'	DOUBLEWORD OUT	

		PAGE						
1585								PSW
1586								
1587				*			SPACE COUNT UNDERFLOW	
1588				*			TS SET, NO TRAP	
1589	01	00538	FFFFFFF4 A	DATA	=12		COUNT	
1590	01	00539	09800314	PSW,8	SP		INSTRUCTION	
1591	01	0053A	20000151	K	2,0,0,SETPSW		PSW1 IN	
1592	01	0053B	000001AE	K	13,0,0,L0C+2		PSW1 OUT	
1593	01	0053C	0359F000 N	P	DTA1,=16,0		R IN	
1594	01	0053D	0359F000 N	P	DTA1,=16,0		R OUT	
1595	01	0053E	03840000 N	P	ZER0S,0,0		MEM IN	
1596	01	0053F	03840000 N	P	ZER0S,0,0		MEM OUT	
1597	01	00540	00000304	DATA	MEMORY		STACK POINTER	
1598	01	00541	80000000 A	DATA	X'80000000'		DOUBLEWORD IN	
1599	01	00542	00000304	DATA	MEMORY		STACK POINTER	
1600	01	00543	80000000 A	DATA	X'80000000'		DOUBLEWORD OUT	
1601								PSW
1602							SPACE COUNT UNDERFLOW	
1603				*			TS NOT SET, TRAP	
1604	01	00544	FFFFFFF4 A	DATA	=12		COUNT	
1605	01	00545	09FC0314	PSW,15	SP		INSTRUCTION	
1606	01	00546	000000AF	K	0,0,0,SLSW		PSW1 IN	
1607	01	00547	00000080	K	0,0,0,SLRET+1		PSW1 OUT	
1608	01	00548	0359F000 N	P	DTA1,=16,0		R IN	
1609	01	00549	0359F000 N	P	DTA1,=16,0		R OUT	
1610	01	0054A	03840000 N	P	ZER0S,0,0		MEM IN	
1611	01	0054B	03840000 N	P	ZER0S,0,0		MEM OUT	
1612	01	0054C	00000304	DATA	MEMORY		STACK POINTER	
1613	01	0054D	00008000 A	DATA	X'00008000'		DOUBLEWORD IN	
1614	01	0054E	00000304	DATA	MEMORY		STACK POINTER	
1615	01	0054F	00008000 A	DATA	X'00008000'		DOUBLEWORD OUT	

		PAGE						
1616								PSW
1617							WORD COUNT OVERFLOW	
1618				*			TS SET, NO TRAP	
1619				*			COUNT	
1620	01	00550	FFFFFFF4 A	DATA	=12		INSTRUCTION	
1621	01	00551	09000314	PSW,0	SP		PSW1 IN	
1622	01	00552	00000151	K	13,0,0,SETPSW		PSW1 OUT	
1623	01	00553	200001AE	K	2,0,0,L0C+2		R IN	
1624	01	00554	03840000 N	P	ZER0S,0,0		R OUT	
1625	01	00555	03840000 N	P	ZER0S,0,0		MEM IN	
1626	01	00556	0359F000 N	P	DTA1,=16,0		MEM OUT	
1627	01	00557	0359F000 N	P	DTA1,=16,0		STACK POINTER	
1628	01	00558	00000304	DATA	MEMORY		DOUBLEWORD IN	
1629	01	00559	000FFFFF A	DATA	X'000FFFFF'		STACK POINTER	
1630	01	0055A	00000304	DATA	MEMORY		DOUBLEWORD OUT	
1631	01	0055B	000FFFFF A	DATA	X'000FFFFF'			PSW
1632							WORD COUNT OVERFLOW	
1633				*			TS NOT SET, TRAP	
1634				*			COUNT	
1635	01	0055C	FFFFFFF4 A	DATA	=12		INSTRUCTION	
1636	01	0055D	09000314	PSW,0	SP		PSW1 IN	
1637	01	0055E	F73000AF	K	15,7,3,SLSW		PSW1 OUT	
1638	01	0055F	F7300080	K	15,7,3,SLRET+1		R IN	
1639	01	00560	0359F000 N	P	DTA1,=16,0		R OUT	
1640	01	00561	0359F000 N	P	DTA1,=16,0		MEM IN	
1641	01	00562	03840000 N	P	ZER0S,0,0		MEM OUT	
1642	01	00563	03840000 N	P	ZER0S,0,0		STACK POINTER	
1643	01	00564	00000304	DATA	MEMORY		DOUBLEWORD IN	
1644	01	00565	000F7FFF A	DATA	X'000F7FFF'		STACK POINTER	
1645	01	00566	00000304	DATA	MEMORY		DOUBLEWORD OUT	
1646	01	00567	000F7FFF A	DATA	X'000F7FFF'			

Address	Op	OpCode	Operand	Page	Register	Value	Instruction
1647							
1648							WORD COUNT OVERFLOW
1649							SPACE COUNT UNDERFLOW
1650							TS NOT SET, TW SET, TRAP
1651							COUNT
1652	01	00568	FFFFFFF4 A		DATA	=12	INSTRUCTION
1653	01	00569	09100314		PSW,1	SP	PSW1 IN
1654	01	0056A	F73000AF		K	15,7,3,SLSW	PSW1 OUT
1655	01	0056B	F7300080		K	15,7,3,SLRET+1	R IN
1656	01	0056C	03840000 N		P	ZERBS,0,0	R OUT
1657	01	0056D	03840000 N		P	ZERBS,0,0	MEM IN
1658	01	0056E	0359F000 N		P	DTA1,-16,0	MEM OUT
1659	01	0056F	0359F000 N		P	DTA1,-16,0	STACK POINTER
1660	01	00570	00000304		DATA	MEMORY	DOUBLEWORD IN
1661	01	00571	0000FFFF A		DATA	X'0000FFFF'	STACK POINTER
1662	01	00572	00000304		DATA	MEMORY	DOUBLEWORD OUT
1663	01	00573	0000FFFF A		DATA	X'0000FFFF'	DOUBLEWORD OUT
1664							
1665							WORD COUNT OVERFLOW
1666							SPACE COUNT UNDERFLOW
1667							TS SET, TW NOT SET, TRAP
1668	01	00574	FFFFFFF4 A		DATA	=12	COUNT
1669	01	00575	09100314		PSW,1	SP	INSTRUCTION
1670	01	00576	073000AF		K	0,7,3,SLSW	PSW1 IN
1671	01	00577	07300080		K	0,7,3,SLRET+1	PSW1 OUT
1672	01	00578	03840000 N		P	ZERBS,0,0	R IN
1673	01	00579	03840000 N		P	ZERBS,0,0	R OUT
1674	01	0057A	0359F000 N		P	DTA1,-16,0	MEM IN
1675	01	0057B	0359F000 N		P	DTA1,-16,0	MEM OUT
1676	01	0057C	00000304		DATA	MEMORY	STACK POINTER
1677	01	0057D	80007FFF A		DATA	X'80007FFF'	DOUBLEWORD IN
1678	01	0057E	00000304		DATA	MEMORY	STACK POINTER
1679	01	0057F	80007FFF A		DATA	X'80007FFF'	DOUBLEWORD OUT

Address	Op	OpCode	Operand	Page	Register	Value	Instruction
1680							
1681							INDEXED, INDEX VALUE OF 1 IN R1
1682							PUSH WORD INTO STACK FROM R2
1683							COUNT
1684	01	00580	FFFFFFF4 A		DATA	=12	INSTRUCTION
1685	01	00581	09220312		PSW,2	SP=2,1	PSW1 IN
1686	01	00582	30000151		K	11,0,0,SETPSW	PSW1 OUT
1687	01	00583	400001AE		K	4,0,0,LBC+2	R IN
1688	01	00584	037FFE01 N		P	DTA3+1,-2,1	R OUT
1689	01	00585	037FFE01 N		P	DTA3+1,-2,1	MEM IN
1690	01	00586	03840000 N		P	ZERBS,0,0	MEM OUT
1691	01	00587	0380FF01 N		P	DTA3+2,-1,1	STACK POINTER
1692	01	00588	00000304		DATA	MEMORY	DOUBLEWORD IN
1693	01	00589	00010000 A		DATA	X'00010000'	STACK POINTER
1694	01	0058A	00000305		DATA	MEMORY+1	DOUBLEWORD OUT
1695	01	0058B	00000001 A		DATA	X'00000001'	DOUBLEWORD OUT
1696							
1697							INDIRECTLY ADDRESSED
1698							PUSH WORD FROM R0
1699	01	0058C	FFFFFFF4 A		DATA	=12	COUNT
1700	01	0058D	89000317		PSW,0	=IASP	INSTRUCTION
1701	01	0058E	F0000151		K	15,0,0,SETPSW	PSW1 IN
1702	01	0058F	070001AE		K	0,0,0,LBC+2	PSW1 OUT
1703	01	00590	0359F000 N		P	DTA1,-16,0	R IN
1704	01	00591	0359F000 N		P	DTA1,-16,0	R OUT
1705	01	00592	03840000 N		P	ZERBS,0,0	MEM IN
1706	01	00593	0359FF01 N		P	DTA1,-1,1	MEM OUT
1707	01	00594	00000304		DATA	MEMORY	STACK POINTER
1708	01	00595	80078007 A		DATA	X'80078007'	DOUBLEWORD IN
1709	01	00596	00000305		DATA	MEMORY+1	STACK POINTER
1710	01	00597	80068008 A		DATA	X'80068008'	DOUBLEWORD OUT

			PAGE			
1711						INDEXED, INDEX VALUE OF 8 IN R5
1712						INDIRECTLY ADDRESSED
1713						PSW
1714						
1715	01	00598	FFFFFF4	A	DATA	=12
1716	01	00599	896A0316		PSW,6	=1A,5
1717	01	0059A	17300151		K	1,7,3, SETPSW
1718	01	0059B	073001AE		K	0,7,3, LBC+2
1719	01	0059C	0382FE05	N	P	DTA3+4, =2,5
1720	01	0059D	0382FE05	N	P	DTA3+4, =2,5
1721	01	0059E	03840000	N	P	ZEROS, 0,0
1722	01	0059F	0383FF00	N	P	DTA3,5, =1,0
1723	01	005A0	00000303		DATA	MEMORY+1
1724	01	005A1	80028000	A	DATA	X'80028000'
1725	01	005A2	00000304		DATA	MEMORY
1726	01	005A3	80018001	A	DATA	X'80018001'
1727						PLW
1728						PULL WORD INTO R0
1729						NO OVERFLOW OR UNDERFLOW
1730	01	005A4	FFFFFF4	A	DATA	=12
1731	01	005A5	08000314		PLW,0	SP
1732	01	005A6	F7300151		K	15,7,3, SETPSW
1733	01	005A7	073001AE		K	0,7,3, LBC+2
1734	01	005A8	03840000	N	P	ZEROS, 0,0
1735	01	005A9	035AFF00	N	P	DTA1+1, =1,0
1736	01	005AA	0359F000	N	P	DTA1, =16,0
1737	01	005AB	0359F000	N	P	DTA1, =16,0
1738	01	005AC	00000305		DATA	MEMORY+1
1739	01	005AD	00020003	A	DATA	X'00020003'
1740	01	005AE	00000304		DATA	MEMORY
1741	01	005AF	00030002	A	DATA	X'00030002'

			PAGE			
1742						PULL WORD INTO R15
1743						SPACE COUNT GOES TO MAXIMUM
1744						WORD COUNT GOES TO ZERO
1745						PLW
1746						
1747	01	005B0	FFFFFF4	A	DATA	=12
1748	01	005B1	08F00314		PLW,15	SP
1749	01	005B2	E0000151		K	14,0,0, SETPSW
1750	01	005B3	100001AE		K	1,0,0, LBC+2
1751	01	005B4	03840000	N	P	ZEROS, 0,0
1752	01	005B5	0368FF0F	N	P	DTA1+15, =1,15
1753	01	005B6	0359F000	N	P	DTA1, =16,0
1754	01	005B7	0359F000	N	P	DTA1, =16,0
1755	01	005B8	00000313		DATA	MEMORY+15
1756	01	005B9	FFFFE8001	A	DATA	X'FFFFE8001'
1757	01	005BA	00000312		DATA	MEMORY+14
1758	01	005BB	FFFFE8000	A	DATA	X'FFFFE8000'
1759						PLW
1760						SPACE COUNT OVERFLOW
1761						TS SET, ABORT
1762	01	005BC	FFFFFF4	A	DATA	=12
1763	01	005BD	08100314		PLW,1	SP
1764	01	005BE	70000151		K	7,0,0, SETPSW
1765	01	005BF	800001AE		K	8,0,0, LBC+2
1766	01	005C0	0359F000	N	P	DTA1, =16,0
1767	01	005C1	0359F000	N	P	DTA1, =16,0
1768	01	005C2	03840000	N	P	ZEROS, 0,0
1769	01	005C3	03840000	N	P	ZEROS, 0,0
1770	01	005C4	00000304		DATA	MEMORY
1771	01	005C5	FFFF0001	A	DATA	X'FFFF0001'
1772	01	005C6	00000304		DATA	MEMORY
1773	01	005C7	FFFF0001	A	DATA	X'FFFF0001'

			PAGE		
1774			*****		PLW
1775			* * * * *		
1776					SPACE COUNT OVERFLOW
1777					TS NOT SET, TRAP
1778	01	005C8	DATA	=12	COUNT
1779	01	005C9	PLW,1	SP	INSTRUCTION
1780	01	005CA	K	0,0,0,SLSW	PSW1 IN
1781	01	005CB	K	0,0,0,SLRET+1	PSW1 OUT
1782	01	005CC	P	DTA1,=16,0	R IN
1783	01	005CD	P	DTA1,=16,0	R OUT
1784	01	005CE	P	ZEROS,0,0	MEM IN
1785	01	005CF	P	ZEROS,0,0	MEM OUT
1786	01	005D0	DATA	MEMORY	STACK POINTER
1787	01	005D1	DATA	X'7FFF0001'	DOUBLEWORD IN
1788	01	005D2	DATA	MEMORY	STACK POINTER
1789	01	005D3	DATA	X'7FFF0001'	DOUBLEWORD OUT
1790			*****		PLW
1791			* * * * *		
1792					WORD COUNT UNDERFLOW
1793	01	005D4	DATA	=12	COUNT
1794	01	005D5	PLW,1	SP	INSTRUCTION
1795	01	005D6	K	8,0,0,SETPSW	PSW1 IN
1796	01	005D7	K	7,0,0,L8C+2	PSW1 OUT
1797	01	005D8	P	ZEROS,0,0	R IN
1798	01	005D9	P	ZEROS,0,0	R OUT
1799	01	005DA	P	DTA1,=16,0	MEM IN
1800	01	005DB	P	DTA1,=16,0	MEM OUT
1801	01	005DC	DATA	MEMORY	STACK POINTER
1802	01	005DD	DATA	X'00008000'	DOUBLEWORD IN
1803	01	005DE	DATA	MEMORY	STACK POINTER
1804	01	005DF	DATA	X'00008000'	DOUBLEWORD OUT

			PAGE		
1805			*****		PLW
1806			* * * * *		
1807					WORD COUNT UNDERFLOW
1808					TS NOT SET, TRAP
1809	01	005E0	DATA	=12	COUNT
1810	01	005E1	PLW,1	SP	INSTRUCTION
1811	01	005E2	K	15,0,0,SLSW	PSW1 IN
1812	01	005E3	K	15,0,0,SLRET+1	PSW1 OUT
1813	01	005E4	P	ZEROS,0,0	R IN
1814	01	005E5	P	ZEROS,0,0	R OUT
1815	01	005E6	P	DTA1,=16,0	MEM IN
1816	01	005E7	P	DTA1,=16,0	MEM OUT
1817	01	005E8	DATA	MEMORY	STACK POINTER
1818	01	005E9	DATA	X'00000000'	DOUBLEWORD IN
1819	01	005EA	DATA	MEMORY	STACK POINTER
1820	01	005EB	DATA	X'00000000'	DOUBLEWORD OUT
1821			*****		PLW
1822			* * * * *		
1823					WORD COUNT UNDERFLOW
1824					SPACE COUNT OVERFLOW
1825	01	005EC	DATA	=12	COUNT
1826	01	005ED	PLW,1	SP	INSTRUCTION
1827	01	005EE	K	0,7,3,SLSW	PSW1 IN
1828	01	005EF	K	0,7,3,SLRET+1	PSW1 OUT
1829	01	005F0	P	ZEROS,0,0	R IN
1830	01	005F1	P	ZEROS,0,0	R OUT
1831	01	005F2	P	DTA1,=16,0	MEM IN
1832	01	005F3	P	DTA1,=16,0	MEM OUT
1833	01	005F4	DATA	MEMORY	STACK POINTER
1834	01	005F5	DATA	X'FFFFFF0000'	DOUBLEWORD IN
1835	01	005F6	DATA	MEMORY	STACK POINTER
1836	01	005F7	DATA	X'FFFFFF0000'	DOUBLEWORD OUT

			PAGE			
1837						PLW
1838						
1839						
1840						
1841						
1842	01	005F8	FFFFFFF4 A	DATA	=12	COUNT
1843	01	005F9	08100314	PLW,1	SP	INSTRUCTION
1844	01	005FA	F73000AF	K	15,7,3,SLSW	PSW1 IN
1845	01	005FB	F7300080	K	15,7,3,SLRET+1	PSW1 OUT
1846	01	005FC	03840000 N	P	ZEROS,0,0	R IN
1847	01	005FD	03840000 N	P	ZEROS,0,0	R OUT
1848	01	005FE	0359F000 N	P	DTA1,=16,0	MEM IN
1849	01	005FF	0359F000 N	P	DTA1,=16,0	MEM OUT
1850	01	00600	00000304	DATA	MEMORY	STACK POINTER
1851	01	00601	7FFF8000 A	DATA	X'7FFF8000'	DOUBLEWORD IN
1852	01	00602	00000304	DATA	MEMORY	STACK POINTER
1853	01	00603	7FFF8000 A	DATA	X'7FFF8000'	DOUBLEWORD OUT
1854						PLW
1855						
1856						
1857						
1858	01	00604	FFFFFFF4 A	DATA	=12	COUNT
1859	01	00605	08000314	PLW,0	SP	INSTRUCTION
1860	01	00606	10000151	K	1,0,0,SETPSW	PSW1 IN
1861	01	00607	800001AE	K	11,0,0,L0C+2	PSW1 OUT
1862	01	00608	03840000 N	P	ZEROS,0,0	R IN
1863	01	00609	03840000 N	P	ZEROS,0,0	R OUT
1864	01	0060A	0359F000 N	P	DTA1,=16,0	MEM IN
1865	01	0060B	0359F000 N	P	DTA1,=16,0	MEM OUT
1866	01	0060C	00000305	DATA	MEMORY+1	STACK POINTER
1867	01	0060D	FFFF8000 A	DATA	X'FFFF8000'	DOUBLEWORD IN
1868	01	0060E	00000305	DATA	MEMORY+1	STACK POINTER
1869	01	0060F	FFFF8000 A	DATA	X'FFFF8000'	DOUBLEWORD OUT

			PAGE			
1870						PLM
1871						
1872						
1873						
1874						
1875	01	00610	FFFFFFF4 A	DATA	=12	COUNT
1876	01	00611	0A800314	PLM,8	SP	INSTRUCTION
1877	01	00612	10000151	K	1,0,0,SETPSW	PSW1 IN
1878	01	00613	000001AE	K	0,0,0,L0C+2	PSW1 OUT
1879	01	00614	03840000 N	P	ZEROS,0,0	R IN
1880	01	00615	035A0108 N	P	DTA1,+1,1,8	R OUT
1881	01	00616	0359F000 N	P	DTA1,=16,0	MEM IN
1882	01	00617	0359F000 N	P	DTA1,=16,0	MEM OUT
1883	01	00618	00000305	DATA	MEMORY+1	STACK POINTER
1884	01	00619	003F7FFF A	DATA	X'003F7FFF'	DOUBLEWORD IN
1885	01	0061A	00000304	DATA	MEMORY	STACK POINTER
1886	01	0061B	00407FFE A	DATA	X'00407FFE'	DOUBLEWORD OUT
1887						PLM
1888						
1889						
1890						
1891	01	0061C	FFFFFFF4 A	DATA	=12	COUNT
1892	01	0061D	0A400314	PLM,4	SP	INSTRUCTION
1893	01	0061E	27300151	K	2,7,3,SETPSW	PSW1 IN
1894	01	0061F	073001AE	K	0,7,3,L0C+2	PSW1 OUT
1895	01	00620	03840000 N	P	ZEROS,0,0	R IN
1896	01	00621	035AFE04 N	P	DTA1,+1,=2,4	R OUT
1897	01	00622	0359F000 N	P	DTA1,=16,0	MEM IN
1898	01	00623	0359F000 N	P	DTA1,=16,0	MEM OUT
1899	01	00624	00000306	DATA	MEMORY+2	STACK POINTER
1900	01	00625	00008312 A	DATA	X'00008312'	DOUBLEWORD IN
1901	01	00626	00000304	DATA	MEMORY	STACK POINTER
1902	01	00627	00028310 A	DATA	X'00028310'	DOUBLEWORD OUT

			PAGE			PLM	
1903							
1904							
1905							
1906							
1907	01	00628	FFFFFFF4	A	DATA	=12	COUNT
1908	01	00629	0A200314		PLM,2	SP	INSTRUCTION
1909	01	0062A	47300151		K	4,7,3,SETPSW	PSW1 IN
1910	01	0062B	073001AE		K	0,7,3,LBC+2	PSW1 OUT
1911	01	0062C	03840000	N	P	ZEROS,0,0	R IN
1912	01	0062D	0359FC02	N	P	DTA1,-4,2	R OUT
1913	01	0062E	0359F000	N	P	DTA1,-16,0	MEM IN
1914	01	0062F	0359F000	N	P	DTA1,-16,0	MEM OUT
1915	01	00630	00000307		DATA	MEMORY+3	STACK POINTER
1916	01	00631	807F7766	A	DATA	X'807F7766'	DOUBLEWORD IN
1917	01	00632	00000303		DATA	MEMORY+1	STACK POINTER
1918	01	00633	80837762	A	DATA	X'80837762'	DOUBLEWORD OUT
1919							
1920							
1921							
1922	01	00634	FFFFFFF4	A	DATA	=1P	COUNT
1923	01	00635	0A100314		PLM,1	SP	INSTRUCTION
1924	01	00636	87300151		K	8,7,3,SETPSW	PSW1 IN
1925	01	00637	073001AE		K	0,7,3,LBC+2	PSW1 OUT
1926	01	00638	03840000	N	P	ZEROS,0,0	R IN
1927	01	00639	035AF801	N	P	DTA1+1,-8,1	R OUT
1928	01	0063A	0359F000	N	P	DTA1,-16,0	MEM IN
1929	01	0063B	0359F000	N	P	DTA1,-16,0	MEM OUT
1930	01	0063C	0000030C		DATA	MEMORY+8	STACK POINTER
1931	01	0063D	800A800A	A	DATA	X'800A800A'	DOUBLEWORD IN
1932	01	0063E	00000304		DATA	MEMORY	STACK POINTER
1933	01	0063F	80128002	A	DATA	X'80128002'	DOUBLEWORD OUT

			PAGE			PLM	
1934							
1935							
1936							
1937							
1938	01	00640	FFFFFFF4	A	DATA	=12	COUNT
1939	01	00641	0A000314		PLM,0	SP	INSTRUCTION
1940	01	00642	07300151		K	0,7,3,SETPSW	PSW1 IN
1941	01	00643	173001AE		K	1,7,3,LBC+2	PSW1 OUT
1942	01	00644	03840000	N	P	ZEROS,0,0	R IN
1943	01	00645	0359F000	N	P	DTA1,-16,0	R OUT
1944	01	00646	0359F000	N	P	DTA1,-16,0	MEM IN
1945	01	00647	0359F000	N	P	DTA1,-16,0	MEM OUT
1946	01	00648	00000313		DATA	MEMORY+15	STACK POINTER
1947	01	00649	80008010	A	DATA	X'80008010'	DOUBLEWORD IN
1948	01	0064A	00000303		DATA	MEMORY+1	STACK POINTER
1949	01	0064B	80108000	A	DATA	X'80108000'	DOUBLEWORD OUT
1950							
1951							
1952							
1953							
1954	01	0064C	FFFFFFF4	A	DATA	=12	COUNT
1955	01	0064D	0A100314		PLM,1	SP	INSTRUCTION
1956	01	0064E	00000151		K	0,0,0,SETPSW	PSW1 IN
1957	01	0064F	100001AE		K	1,0,0,LBC+2	PSW1 OUT
1958	01	00650	03840000	N	P	ZEROS,0,0	R IN
1959	01	00651	0359F001	N	P	DTA1,-16,1	R OUT
1960	01	00652	0359F000	N	P	DTA1,-16,0	MEM IN
1961	01	00653	0359F000	N	P	DTA1,-16,0	MEM OUT
1962	01	00654	00000313		DATA	MEMORY+15	STACK POINTER
1963	01	00655	7FEF0010	A	DATA	X'7FEF0010'	DOUBLEWORD IN
1964	01	00656	00000303		DATA	MEMORY+1	STACK POINTER
1965	01	00657	7FFF0000	A	DATA	X'7FFF0000'	DOUBLEWORD OUT

			PAGE		
1966					PULL 1 WORD
1967					WORD COUNT UNDERFLOW
1968					TS=1, TW=0 TRAP
1969					COUNT
1970					INSTRUCTION
1971	01	00658	FFFFFFF4 A	DATA =12	PSW1 IN
1972	01	00659	0A000314	PLM,0 SP	PSW1 OUT
1973	01	0065A	100000AF	K 1,0,0,SLSW	R IN
1974	01	0065B	10000080	K 1,0,0,SLRET+1	R OUT
1975	01	0065C	03840000 N	P ZEROS,0,0	MEM IN
1976	01	0065D	03840000 N	P ZEROS,0,0	MEM OUT
1977	01	0065E	0359F000 N	P DTAL,-16,0	STACK POINTER
1978	01	0065F	0359F000 N	P DTAL,-16,0	DOUBLEWORD IN
1979	01	00660	00000304	DATA MEMORY	STACK POINTER
1980	01	00661	F2100000 A	DATA X'F2100000'	DOUBLEWORD OUT
1981	01	00662	00000304	DATA MEMORY	
1982	01	00663	F2100000 A	DATA X'F2100000'	
1983					PULL 1 WORD
1984					WORD COUNT UNDERFLOW
1985					TS=1, TW=1 ABORT
1986					COUNT
1987	01	00664	FFFFFFF4 A	DATA =12	INSTRUCTION
1988	01	00665	0A000314	PLM,0 SP	PSW1 IN
1989	01	00666	10000151	K 1,0,0,SETPSW	PSW1 OUT
1990	01	00667	300001AE	K 3,0,0,LBC+2	R IN
1991	01	00668	03840000 N	P ZEROS,0,0	R OUT
1992	01	00669	03840000 N	P ZEROS,0,0	MEM IN
1993	01	0066A	0359F000 N	P DTAL,-16,0	MEM OUT
1994	01	0066B	0359F000 N	P DTAL,-16,0	STACK POINTER
1995	01	0066C	00000304	DATA MEMORY	DOUBLEWORD IN
1996	01	0066D	F2108000 A	DATA X'F2108000'	STACK POINTER
1997	01	0066E	00000304	DATA MEMORY	DOUBLEWORD OUT
1998	01	0066F	F2108000 A	DATA X'F2108000'	

			PAGE		
1999					PULL 1 WORD
2000					SPACE COUNT OVERFLOW
2001					TS=0, TW=1 TRAP
2002					COUNT
2003					INSTRUCTION
2004	01	00670	FFFFFFF4 A	DATA =12	PSW1 IN
2005	01	00671	0A000314	PLM,0 SP	PSW1 OUT
2006	01	00672	173000AF	K 1,7,3,SLSW	R IN
2007	01	00673	17300080	K 1,7,3,SLRET+1	R OUT
2008	01	00674	03840000 N	P ZEROS,0,0	MEM IN
2009	01	00675	03840000 N	P ZEROS,0,0	MEM OUT
2010	01	00676	0359F000 N	P DTAL,-16,0	STACK POINTER
2011	01	00677	0359F000 N	P DTAL,-16,0	DOUBLEWORD IN
2012	01	00678	00000304	DATA MEMORY	STACK POINTER
2013	01	00679	7FFF8100 A	DATA X'7FFF8100'	DOUBLEWORD IN
2014	01	0067A	00000304	DATA MEMORY	STACK POINTER
2015	01	0067B	7FFF8100 A	DATA X'7FFF8100'	DOUBLEWORD OUT
2016					PULL 1 WORD
2017					SPACE COUNT OVERFLOW
2018					TS=1, TW=0 ABORT
2019					COUNT
2020	01	0067C	FFFFFFF4 A	DATA =12	INSTRUCTION
2021	01	0067D	0A000314	PLM,0 SP	PSW1 IN
2022	01	0067E	17300151	K 1,7,3,SETPSW	PSW1 OUT
2023	01	0067F	873001AE	K 8,7,3,LBC+2	R IN
2024	01	00680	03840000 N	P ZEROS,0,0	R OUT
2025	01	00681	03840000 N	P ZEROS,0,0	MEM IN
2026	01	00682	0359F000 N	P DTAL,-16,0	MEM OUT
2027	01	00683	0359F000 N	P DTAL,-16,0	STACK POINTER
2028	01	00684	00000304	DATA MEMORY	DOUBLEWORD IN
2029	01	00685	FFFF0001 A	DATA X'FFFF0001'	STACK POINTER
2030	01	00686	00000304	DATA MEMORY	DOUBLEWORD OUT
2031	01	00687	FFFF0001 A	DATA X'FFFF0001'	

		PAGE		
2032				PLM
2033				
2034				WORD COUNT UNDERFLOW
2035				SPACE COUNT OVERFLOW
2036				TS=1 TW=1 ABBRT
2037	01 00688	FFFFFFF4 A	DATA =12	COUNT
2038	01 00689	0A000314	PLM,0 SP	INSTRUCTION
2039	01 0068A	10000151	< 1,0,0,SETPSW	PSW1 IN
2040	01 0068B	300001AE	< 1,0,0,LBC+2	PSW1 OUT
2041	01 0068C	03840000 N	P ZEROS,0,0	R IN
2042	01 0068D	03840000 N	P ZEROS,0,0	R OUT
2043	01 0068E	0359F000 N	P DTA1,=16,0	MEM IN
2044	01 0068F	0359F000 N	P DTA1,=16,0	MEM OUT
2045	01 00690	00000304	DATA MEMBRY	STACK POINTER
2046	01 00691	FFFF8000 A	DATA X'FFFF8000'	DOUBLEWORD IN
2047	01 00692	00000304	DATA MEMBRY	STACK POINTER
2048	01 00693	FFFF8000 A	DATA X'FFFF8000'	DOUBLEWORD OUT
2049				PLM
2050				WORD COUNT UNDERFLOW
2051				SPACE COUNT OVERFLOW
2052				TS=0 TW=0 TRAP
2053	01 00694	FFFFFFF4 A	DATA =12	COUNT
2054	01 00695	0A000314	PLM,0 SP	INSTRUCTION
2055	01 00696	07300CAF	< 0,7,3,SLSW	PSW1 IN
2056	01 00697	07300080	< 0,7,3,SLRET+1	PSW1 OUT
2057	01 00698	03840000 N	P ZEROS,0,0	R IN
2058	01 00699	03840000 N	P ZEROS,0,0	R OUT
2059	01 0069A	0359F000 N	P DTA1,=16,0	MEM IN
2060	01 0069B	0359F000 N	P DTA1,=16,0	MEM OUT
2061	01 0069C	00000304	DATA MEMBRY	STACK POINTER
2062	01 0069D	7FF0000F A	DATA X'7FF0000F'	DOUBLEWORD IN
2063	01 0069E	00000304	DATA MEMBRY	STACK POINTER
2064	01 0069F	7FF0000F A	DATA X'7FF0000F'	DOUBLEWORD OUT

		PAGE		
2065				PLM
2066				
2067				INDEXED, INDEX VALUE=1 IN R7
2068				PULL 15 WORDS; START LOADING INTS
2069				REGISTER RS; TS=0, TW=C
2070	01 006A0	FFFFFFF4 A	DATA =12	COUNT
2071	01 006A1	0A8E0312	PLM,8 SP=2,7	INSTRUCTION
2072	01 006A2	F0000151	< 15,0,0,SETPSW	PSW1 IN
2073	01 006A3	100001AE	< 1,0,0,LBC+2	PSW1 OUT
2074	01 006A4	0369FF07 N	P DTA2,=1,7	R IN
2075	01 006A5	0369FF07 N	P DTA2,=16,7	R OUT
2076	01 006A6	036AF000 N	P DTA2+1,=16,0	MEM IN
2077	01 006A7	036AF000 N	P DTA2+1,=16,0	MEM OUT
2078	01 006A8	00000312	DATA MEMBRY+16	STACK POINTER
2079	01 006A9	0000000F A	DATA X'0000000F'	DOUBLEWORD IN
2080	01 006AA	00000303	DATA MEMBRY+1	STACK POINTER
2081	01 006AB	000F0000 A	DATA X'000F0000'	DOUBLEWORD OUT
2082				PLM
2083				INDIRECTLY ADDRESSED
2084				WORD COUNT UNDERFLOW
2085				TS=0, TW=0, TRAP
2086	01 006AC	FFFFFFF4 A	DATA =12	COUNT
2087	01 006AD	8A000317	PLM,8 SP=1,ASP	INSTRUCTION
2088	01 006AE	100000AF	< 1,0,0,SLSW	PSW1 IN
2089	01 006AF	10000080	< 1,0,0,SLRET+1	PSW1 OUT
2090	01 006B0	03840000 N	P ZEROS,0,0	R IN
2091	01 006B1	03840000 N	P ZEROS,0,0	R OUT
2092	01 006B2	0359F000 N	P DTA1,=16,0	MEM IN
2093	01 006B3	0359F000 N	P DTA1,=16,0	MEM OUT
2094	01 006B4	00000305	DATA MEMBRY+1	STACK POINTER
2095	01 006B5	00000000 A	DATA 00000000	DOUBLEWORD IN
2096	01 006B6	00000305	DATA MEMBRY+1	STACK POINTER
2097	01 006B7	00000000 A	DATA 00000000	DOUBLEWORD OUT

```

2098                                     PAGE
2099                                     ..... PSM
2100                                     *
2101                                     * PUSH 1 WORD FROM R8
2102                                     * TS=0, TW=0
2103                                     * STARTS WITH 0 WORD COUNT
2103 01 006B8 FFFFFFF4 A DATA =12 COUNT
2104 01 006B9 03800314 PSM,8 SP INSTRUCTION
2105 01 006BA 10000151 K 1,0,0,SETPSW PSW1 IN
2106 01 006BB 000001AE K 0,0,0,LBC+2 PSW1 OUT
2107 01 006BC 0359F000 N P DTA1,=16,0 R IN
2108 01 006BD 0359F000 N P DTA1,=16,0 R OUT
2109 01 006BE 03840000 N P ZERBS,0,0 MEM IN
2110 01 006BF 0361FF01 N P DTA1+8,=1,1 MEM OUT
2111 01 006C0 00000304 DATA MEMORY STACK POINTER
2112 01 006C1 01000000 A DATA X'01000000' DBLEWORD IN
2113 01 006C2 00000305 DATA MEMORY+1 STACK POINTER
2114 01 006C3 00FF0001 A DATA X'00FF0001' DBLEWORD OUT
2115                                     ..... PSM
2116                                     *
2117                                     * PUSH 2 WORDS FROM R4,R5
2118                                     * TS=0, TW=1
2119                                     * STARTS WITH MAXIMUM SPACE COUNT
2119 01 006C4 FFFFFFF4 A DATA =12 COUNT
2120 01 006C5 03400314 PSM,4 SP INSTRUCTION
2121 01 006C6 27300151 K 2,7,3,SETPSW PSW1 IN
2122 01 006C7 073001AE K 0,7,3,LBC+2 PSW1 OUT
2123 01 006C8 0359F000 N P DTA1,=16,0 R IN
2124 01 006C9 0359F000 N P DTA1,=16,0 R OUT
2125 01 006CA 03840000 N P ZERBS,0,0 MEM IN
2126 01 006CB 035DFE01 N P DTA1+4,=2,1 MEM OUT
2127 01 006CC 00000304 DATA MEMORY STACK POINTER
2128 01 006CD 7FFFFFFD A DATA X'7FFFFFFD' DBLEWORD IN
2129 01 006CE 00000306 DATA MEMORY+2 STACK POINTER
2130 01 006CF 7FFDFFFF A DATA X'7FFDFFFF' DBLEWORD OUT
    
```

```

2131                                     PAGE
2132                                     ..... PSM
2133                                     *
2134                                     * PUSH 4 WORDS FROM R2-R5
2135                                     * TS=1, TW=0
2136 01 006D0 FFFFFFF4 A DATA =12 COUNT
2137 01 006D1 03200314 PSM,2 SP INSTRUCTION
2138 01 006D2 47300151 K 4,7,3,SETPSW PSW1 IN
2139 01 006D3 073001AE K 0,7,3,LBC+2 PSW1 OUT
2140 01 006D4 0359F000 N P DTA1,=16,0 R IN
2141 01 006D5 0359F000 N P DTA1,=16,0 R OUT
2142 01 006D6 03840000 N P ZERBS,0,0 MEM IN
2143 01 006D7 0353FC01 N P DTA1+2,=4,1 MEM OUT
2144 01 006D8 00000304 DATA MEMORY STACK POINTER
2145 01 006D9 80050001 A DATA X'80050001' DBLEWORD IN
2146 01 006DA 00000308 DATA MEMORY+4 STACK POINTER
2147 01 006DB 80010005 A DATA X'80010005' DBLEWORD OUT
2148                                     ..... PSM
2149                                     *
2150                                     * PUSH 8 WORDS FROM R1-R8
2151                                     * TS=1, TW=1
2152                                     * COUNT
2152 01 006DC FFFFFFF4 A DATA =12 INSTRUCTION
2153 01 006DD 03100314 PSM,1 SP INSTRUCTION
2154 01 006DE 87300151 K 8,7,3,SETPSW PSW1 IN
2155 01 006DF 073001AE K 0,7,3,LBC+2 PSW1 OUT
2156 01 006E0 0359F000 N P DTA1,=16,0 R IN
2157 01 006E1 0359F000 N P DTA1,=16,0 R OUT
2158 01 006E2 03840000 N P ZERBS,0,0 MEM IN
2159 01 006E3 035AF801 N P DTA1+1,=8,1 MEM OUT
2160 01 006E4 00000304 DATA MEMORY STACK POINTER
2161 01 006E5 FFFF8000 A DATA X'FFF8000' DBLEWORD IN
2162 01 006E6 0000030C DATA MEMORY+8 STACK POINTER
2162 01 006E7 FFF78008 A DATA X'FFF78008' DBLEWORD OUT
    
```

		PAGE		
2163				***** PSM
2164				
2165				PUSH 16 WORDS FROM R0-R15
2166				TS=0, TW=0
2167	01 006E8	FFFFFFF4 A	DATA #12	COUNT
2168	01 006E9	03000314	PSM,0 SP	INSTRUCTION
2169	01 006EA	00000151	< 0,0,0,SETPSW	PSW1 IN
2170	01 006EB	400001AE	< 4,0,0,L0C+2	PSW1 OUT
2171	01 006EC	0359F000 N	P DTA1,-16,0	R IN
2172	01 006ED	0359F000 N	P DTA1,-16,0	R OUT
2173	01 006EE	03840000 N	P ZER0S,0,0	MEM IN
2174	01 006EF	0359F000 N	P DTA1,-16,0	MEM OUT
2175	01 006F0	00000303	DATA MEMBRY=1	STACK POINTER
2176	01 006F1	00100000 A	DATA X'00100000'	DOUBLEWORD IN
2177	01 006F2	00000313	DATA MEMBRY+15	STACK POINTER
2178	01 006F3	00000010 A	DATA X'00000010'	DOUBLEWORD OUT
2179				***** PSM
2180				PUSH 16 WORDS FROM R0-R15 BEGINNING
2181				WITH R15
2182				TS=1, TW=1
2183	01 006F4	FFFFFFF4 A	DATA #12	COUNT
2184	01 006F5	03F00314	PSM,15 SP	INSTRUCTION
2185	01 006F6	00000151	< 0,0,0,SETPSW	PSW1 IN
2186	01 006F7	400001AE	< 4,0,0,L0C+2	PSW1 OUT
2187	01 006F8	0359F000 N	P DTA1,-16,0	R IN
2188	01 006F9	0359F000 N	P DTA1,-16,0	R OUT
2189	01 006FA	03840000 N	P ZER0S,0,0	MEM IN
2190	01 006FB	0359F001 N	P DTA1,-16,1	MEM OUT
2191	01 006FC	00000303	DATA MEMBRY=1	STACK POINTER
2192	01 006FD	8010FFFF A	DATA X'8010FFFF'	DOUBLEWORD IN
2193	01 006FE	00000313	DATA MEMBRY+15	STACK POINTER
2194	01 006FF	8000FFFF A	DATA X'8000FFFF'	DOUBLEWORD OUT

		PAGE		
2195				***** PSM
2196				
2197				PUSH 1 WORD
2198				WORD COUNT OVERFLOW
2199				TS=1, TW=0, TRAP
2200	01 00700	FFFFFFF4 A	DATA #12	COUNT
2201	01 00701	03100314	PSM,1 SP	INSTRUCTION
2202	01 00702	173000AF	< 1,7,3,SLSW	PSW1 IN
2203	01 00703	17300080	< 1,7,3,SLRET+1	PSW1 OUT
2204	01 00704	0359F000 N	P DTA1,-16,0	R IN
2205	01 00705	0359F000 N	P DTA1,-16,0	R OUT
2206	01 00706	03840000 N	P ZER0S,0,0	MEM IN
2207	01 00707	03840000 N	P ZER0S,0,0	MEM OUT
2208	01 00708	00000304	DATA MEMBRY	STACK POINTER
2209	01 00709	80007FFF A	DATA X'80007FFF'	DOUBLEWORD IN
2210	01 0070A	00000304	DATA MEMBRY	STACK POINTER
2211	01 0070B	80007FFF A	DATA X'80007FFF'	DOUBLEWORD OUT
2212				***** PSM
2213				PUSH 3 WORDS
2214				WORD COUNT OVERFLOW
2215				TS=1, TW=1, ABR7
2216	01 0070C	FFFFFFF4 A	DATA #12	COUNT
2217	01 0070D	03100314	PSM,1 SP	INSTRUCTION
2218	01 0070E	37300151	< 3,7,3,SETPSW	PSW1 IN
2219	01 0070F	E73001AE	< 14,7,3,L0C+2	PSW1 OUT
2220	01 00710	0359F000 N	P DTA1,-16,0	R IN
2221	01 00711	0359F000 N	P DTA1,-16,0	R OUT
2222	01 00712	03840000 N	P ZER0S,0,0	MEM IN
2223	01 00713	03840000 N	P ZER0S,0,0	MEM OUT
2224	01 00714	00000304	DATA MEMBRY	STACK POINTER
2225	01 00715	8000FFFF A	DATA X'8000FFFF'	DOUBLEWORD IN
2226	01 00716	00000304	DATA MEMBRY	STACK POINTER
2227	01 00717	8000FFFF A	DATA X'8000FFFF'	DOUBLEWORD OUT

		PAGE		
2228				PSM
2229				
2230				PUSH 1 WORD
2231				SPACE COUNT UNDERFLOW
2232				TS=0, TW=1 TRAP
2233	01 00718	FFFFFFF4 A	DATA =12	COUNT
2234	01 00719	09200314	PSM,2 SP	INSTRUCTION
2235	01 0071A	100000AF	K 1,0,0,SLSW	PSW1 IN
2236	01 0071B	10000080	K 1,0,0,SLRET+1	PSW1 OUT
2237	01 0071C	0359F000 N	P DTAL,=16,0	R IN
2238	01 0071D	0359F000 N	P DTAL,=16,0	R OUT
2239	01 0071E	03840000 N	P ZERBS,0,0	MEM IN
2240	01 0071F	03840000 N	P ZERBS,0,0	MEM OUT
2241	01 00720	00000304	DATA MEMORY	STACK POINTER
2242	01 00721	0000FFFF0 A	DATA X'0000FFFF0'	DOUBLEWORD IN
2243	01 00722	00000304	DATA MEMORY	STACK POINTER
2244	01 00723	0000FFFF0 A	DATA X'0000FFFF0'	DOUBLEWORD OUT
2245				PSM
2246				
2247				PUSH 16 WORDS
2248				SPACE COUNT UNDERFLOW
2249				TS=1, TW=1 ABORT
2249	01 00724	FFFFFFF4 A	DATA =12	COUNT
2250	01 00725	09600314	PSM,6 SP	INSTRUCTION
2251	01 00726	00000151	K 0,0,0,SETPSW	PSW1 IN
2252	01 00727	000001AE	K 13,0,0,LBC+2	PSW1 OUT
2253	01 00728	0359F000 N	P DTAL,=16,0	R IN
2254	01 00729	0359F000 N	P DTAL,=16,0	R OUT
2255	01 0072A	03840000 N	P ZERBS,0,0	MEM IN
2256	01 0072B	03840000 N	P ZERBS,0,0	MEM OUT
2257	01 0072C	00000304	DATA MEMORY	STACK POINTER
2258	01 0072D	80008000 A	DATA X'80008000'	DOUBLEWORD IN
2259	01 0072E	00000304	DATA MEMORY	STACK POINTER
2260	01 0072F	80008000 A	DATA X'80008000'	DOUBLEWORD OUT

		PAGE		
2261				PSM
2262				
2263				WORD COUNT OVERFLOW
2264				SPACE COUNT UNDERFLOW
2265				TS=1, TW=1 ABORT
2266	01 00730	FFFFFFF4 A	DATA =12	COUNT
2267	01 00731	09100314	PSM,1 SP	INSTRUCTION
2268	01 00732	10000151	K 1,0,0,SETPSW	PSW1 IN
2269	01 00733	E00001AE	K 14,0,0,LBC+2	PSW1 OUT
2270	01 00734	0359F000 N	P DTAL,=16,0	R IN
2271	01 00735	0359F000 N	P DTAL,=16,0	R OUT
2272	01 00736	03840000 N	P ZERBS,0,0	MEM IN
2273	01 00737	03840000 N	P ZERBS,0,0	MEM OUT
2274	01 00738	00000304	DATA MEMORY	STACK POINTER
2275	01 00739	8000FFFF A	DATA X'8000FFFF'	DOUBLEWORD IN
2276	01 0073A	00000304	DATA MEMORY	STACK POINTER
2277	01 0073B	8000FFFF A	DATA X'8000FFFF'	DOUBLEWORD OUT
2278				PSM
2279				
2280				WORD COUNT OVERFLOW
2281				SPACE COUNT UNDERFLOW
2282				TS=0, TW=1 TRAP
2282	01 0073C	FFFFFFF4 A	DATA =12	COUNT
2283	01 0073D	09100314	PSM,1 SP	INSTRUCTION
2284	01 0073E	200000AF	K 2,0,0,SLSW	PSW1 IN
2285	01 0073F	20000080	K 2,0,0,SLRET+1	PSW1 OUT
2286	01 00740	0359F000 N	P DTAL,=16,0	R IN
2287	01 00741	0359F000 N	P DTAL,=16,0	R OUT
2288	01 00742	03840000 N	P ZERBS,0,0	MEM IN
2289	01 00743	03840000 N	P ZERBS,0,0	MEM OUT
2290	01 00744	00000304	DATA MEMORY	STACK POINTER
2291	01 00745	0001FFFE A	DATA X'0001FFFE'	DOUBLEWORD IN
2292	01 00746	00000304	DATA MEMORY	STACK POINTER
2293	01 00747	0001FFFE A	DATA X'0001FFFE'	DOUBLEWORD OUT

```

2294                                     PAGE
2295                                     ***** PSM
2296                                     *
2297                                     *      WORD COUNT OVERFLOW
2298                                     *      SPACE COUNT UNDERFLOW
2299                                     *      TS=1, TW=0, TRAP
2300 01 00748 FFFFFFF4 A      DATA      =12      COUNT
2301 01 00749 09100314      PSM,1      SP      INSTRUCTION
2302 01 0074A 100000AF      <      1,0,0,SLSW      PSW1 IN
2303 01 0074B 10000080      <      1,0,0,SLRET+1      PSW1 BUT
2304 01 0074C 0359F000 V      P      DTA1,=16,0      R IN
2305 01 0074D 0359F000 V      P      DTA1,=16,0      R BUT
2306 01 0074E 03840000 V      P      ZERBS,0,0      MEM IN
2307 01 0074F 03840000 V      P      ZERBS,0,0      MEM BUT
2308 01 00750 00000304      DATA     MEMORY      STACK POINTER
2309 01 00751 80007FFF A      DATA     X'80007FFF'      DOUBLEWORD IN
2310 01 00752 00000304      DATA     MEMORY      STACK POINTER
2311 01 00753 80007FFF A      DATA     X'80007FFF'      DOUBLEWORD BUT
2312                                     ***** PSM
2313                                     *
2314                                     *      INDEXED, INDEX=4 IN R7
2315                                     *      TS=0, TW=0
2316 01 00754 FFFFFFF4 A      DATA      =12      COUNT
2317 01 00755 098E030C      PSM,8      SP=8,7      INSTRUCTION
2318 01 00756 50000151      <      5,0,0,SETPSW      PSW1 IN
2319 01 00757 000001AE      <      0,0,0,LBC+2      PSW1 BUT
2320 01 00758 0372FA07 V      P      DTA2+9,=6,7      R IN
2321 01 00759 0372FA07 V      P      DTA2+9,=6,7      R BUT
2322 01 0075A 03840000 V      P      ZERBS,0,0      MEM IN
2323 01 0075B 0373F301 V      P      DTA2+10,=-5,1      MEM BUT
2324 01 0075C 00000304      DATA     MEMORY      STACK POINTER
2325 01 0075D 00067FFA A      DATA     X'00067FFA'      DOUBLEWORD IN
2326 01 0075E 00000309      DATA     MEMORY+5      STACK POINTER
2327 01 0075F 00017FFF A      DATA     X'00017FFF'      DOUBLEWORD BUT

```

```

2326                                     PAGE
2327                                     ***** PSM
2328                                     *
2329                                     *      INDIRECTLY ADDRESSED
2330                                     *      TS=0, TW=0
2331                                     *      PUSHES FROM R15 AND R0
2332 01 00760 FFFFFFF4 A      DATA      =12      COUNT
2333 01 00761 83F00317      PSM,15     *IASP      INSTRUCTION
2334 01 00762 20000151      <      2,0,0,SETPSW      PSW1 IN
2335 01 00763 000001AE      <      0,0,0,LBC+2      PSW1 BUT
2336 01 00764 0359F00F V      P      DTA1,=16,15      R IN
2337 01 00765 0359F00F V      P      DTA1,=16,15      R BUT
2338 01 00766 03840000 V      P      ZERBS,0,0      MEM IN
2339 01 00767 0359FE00 V      P      DTA1,=2,0      MEM BUT
2340 01 00768 00000303      DATA     MEMORY+1      STACK POINTER
2341 01 00769 00800080 A      DATA     X'00800080'      DOUBLEWORD IN
2342 01 0076A 00000305      DATA     MEMORY+1      STACK POINTER
2343 01 0076B 007E0082 A      DATA     X'007E0082'      DOUBLEWORD BUT

```

			PAGE		
2343					
2344					MMc
2345					
2346					
2347	01	0076C	FFFFFFF8	A	
2348	01	0076D	6F420000	A	
2349	01	0076E	00000151	A	
2350	01	0076F	000001AE	A	
2351	01	00770	03F0FE04	N	
2352	01	00771	03F2FE04	N	
2353	01	00772	03840000	N	
2354	01	00773	03840000	N	
2355					MMc
2356					
2357					
2358	01	00774	FFFFFFF8	A	
2359	01	00775	6F420000	A	
2360	01	00776	F7300151	A	
2361	01	00777	F73001AE	A	
2362	01	00778	03F4FE04	N	
2363	01	00779	03F6FE04	N	
2364	01	0077A	0359F000	N	
2365	01	0077B	0359F000	N	
2366					MMc
2367					
2368					
2369	01	0077C	FFFFFFF8	A	
2370	01	0077D	6F420000	A	
2371	01	0077E	00800151	A	
2372	01	0077F	20000067	A	
2373	01	00780	03F0FE04	N	
2374	01	00781	03F0FE04	N	
2375	01	00782	0359F000	N	
2376	01	00783	0359F000	N	

			PAGE		
2377					
2378					MMc
2379					
2380					
2381	01	00784	FFFFFFFA	A	
2382	01	00785	6F420000	A	
2383	01	00786	00000151	A	
2384	01	00787	000001AE	A	
2385	01	00788	03F8FE04	N	
2386	01	00789	03FAFE04	N	
2387					MMc
2388					
2389					
2390	01	0078A	FFFFFFFA	A	
2391	01	0078B	6F020000	A	
2392	01	0078C	F0000151	A	
2393	01	0078D	F00001AE	A	
2394	01	0078E	03FCFE00	N	
2395	01	0078F	03FEFE00	N	
2396					MMc
2397					
2398					
2399	01	00790	FFFFFFFA	A	
2400	01	00791	EF420000	A	
2401	01	00792	00000151	A	
2402	01	00793	000001AE	A	
2403	01	00794	03F0FE04	N	
2404	01	00795	03F2FE04	N	

2405			PAGE		
2406			*****		MMC
2407			*		ILLEGAL INSTRUCTION TRAP
2408	01 00796	FFFFFFF8 A	DATA	=8	COUNT
2409	01 00797	6F440000 A	MMC,4	2	INSTRUCTION
2410	01 00798	00000151	K	0,0,0,SETPSW	PSW1 IN
2411	01 00799	8000006D	K	8,0,0,NEIRET+1	PSW1 BUT
2412	01 0079A	03F0FE04 V	P	MMCR1,=2,4	R IN
2413	01 0079B	03F0FE04 V	P	MMCR1,=2,4	R BUT
2414	01 0079C	0359F000 V	P	DTA1,=16,0	MEM IN
2415	01 0079D	0359F000 V	P	DTA1,=16,0	MEM BUT
2416			*****		MMC
2417			*		ILLEGAL INSTRUCTION TRAP
2418	01 0079E	FFFFFFF8 A	DATA	=8	COUNT
2419	01 0079F	6F440000 A	MMC,4	4	INSTRUCTION
2420	01 007A0	00000151	K	0,0,0,SETPSW	PSW1 IN
2421	01 007A1	8000006D	K	8,0,0,NEIRET+1	PSW1 BUT
2422	01 007A2	03F0FE04 V	P	MMCR1,=2,4	R IN
2423	01 007A3	03F0FE04 V	P	MMCR1,=2,4	R BUT
2424	01 007A4	0359F000 V	P	DTA1,=16,0	MEM IN
2425	01 007A5	0359F000 V	P	DTA1,=16,0	MEM BUT
2426			*****		

2427			PAGE		
2428			* THE FOLLOWING 6 MODULES WILL TEST THE P-REGISTER COUNT DOWN LOGIC		*B
2429			*****		PLM
2430			*		PULL 2 WORDS FROM 7FF & 800
2431			*		NO OVERFLOW OR UNDERFLOW
2432			*		R/P20=NP21, NP22, PDC22
2433			*		PDC22=PDC29, P23292
2434			*		PDC29=PDC31, PDC3033, NP30,
2435			*		NP31
2436	01 007A6	FFFFFFF4 A	DECP20	DATA	=12
2437	01 007A7	0A000314	PLM,0	SP	INSTRUCTION
2438	01 007A8	20000151	K	2,0,0,SETPSW	PSW1 IN
2439	01 007A9	100001AE	K	1,0,0,LBC+2	PSW1 BUT
2440	01 007AA	03840000 V	P	ZERBS,0,0	R IN
2441	01 007AB	0359FE00 V	P	DTA1,=2,0	R BUT
2442	01 007AC	0359F000 V	P	DTA1,=16,0	MEM IN
2443	01 007AD	0359F000 V	P	DTA1,=16,0	MEM BUT
2444	01 007AE	00000800 A	DATA	X'800'	STACK POINTER
2445	01 007AF	00000002 A	DATA	X'00000002'	DOUBLEWORD IN
2446	01 007B0	000007FE A	DATA	X'7FE'	STACK POINTER
2447	01 007B1	00020000 A	DATA	X'00020000'	DOUBLEWORD BUT
2448			*****		PLM
2449			*		PULL 2 WORDS FROM 7FF & 1000
2450			*		NO OVERFLOW OR UNDERFLOW
2451			*		R/P19=NP20, NP21, NP22, PDC22
2452	01 007B2	FFFFFFF4 A	DECP19	DATA	=12
2453	01 007B3	0A000314	PLM,0	SP	INSTRUCTION
2454	01 007B4	20000151	K	2,0,0,SETPSW	PSW1 IN
2455	01 007B5	100001AE	K	1,0,0,LBC+2	PSW1 BUT
2456	01 007B6	03840000 V	P	ZERBS,0,0	R IN
2457	01 007B7	0359FE00 V	P	DTA1,=2,0	R BUT
2458	01 007B8	0359F000 V	P	DTA1,=16,0	MEM IN
2459	01 007B9	0359F000 V	P	DTA1,=16,0	MEM BUT
2460	01 007BA	00001000 A	DATA	X'1000'	STACK POINTER
2461	01 007BB	00000002 A	DATA	X'00000002'	DOUBLEWORD IN
2462	01 007BC	000007FE A	DATA	X'7FE'	STACK POINTER
2463	01 007BD	00020000 A	DATA	X'00020000'	DOUBLEWORD BUT

```

2464          PAGE
2465          *..... PLM *B
2466          * PULL 2 WORDS FROM 1FFF & 2000 *B
2467          * NO OVERFLOW OR UNDERFLOW *B
2468          * R/P18*PDC18*PDC29*P2329Z*NP19 *B
2469          * NP20*NP21*NP22 *B
2470 01 0073E 00000000 A DECP18 DATA 0 COUNT SUPPLIED BY DRIVER *B
2471 01 0073F 0A000314 PLM,0 SP INSTRUCTION *B
2472 01 007C0 20000151 K 2,0,0,SETPSW PSW1 IN *B
2473 01 007C1 100001AE K 1,0,0,LBC+2 PSW1 OUT *B
2474 01 007C2 03840000 N P ZERBS,0,0 R IN *B
2475 01 007C3 0359FE00 N P DTAL,=2,0 R OUT *B
2476 01 007C4 0359F000 N P DTAL,=16,0 MEM IN *B
2477 01 007C5 0359F000 N P DTAL,=16,0 MEM OUT *B
2478 01 007C6 00002000 A DATA X'2000' STACK POINTER *B
2479 01 007C7 00000002 A DATA X'00000002' DOUBLEWORD IN *B
2480 01 007C8 00001FFE A DATA X'1FFE' STACK POINTER *B
2481 01 007C9 00020000 A DATA X'00020000' DOUBLEWORD OUT *B
2482          *..... PLM *B
2483          * PULL 2 WORDS FROM 3FFF & 4000 *B
2484          * NO OVERFLOW OR UNDERFLOW *B
2485          * R/P17*NP18*PDC18 *B
2486          * COUNT SUPPLIED BY DRIVER *B
2487 01 007CA 00000000 A DECP17 DATA 0 *B
2488 01 007CB 0A000314 PLM,0 SP INSTRUCTION *B
2489 01 007CC 20000151 K 2,0,0,SETPSW PSW1 IN *B
2490 01 007CD 100001AE K 1,0,0,LBC+2 PSW1 OUT *B
2491 01 007CE 03840000 N P ZERBS,0,0 R IN *B
2492 01 007CF 0359FE00 N P DTAL,=2,0 R OUT *B
2493 01 007D0 0359F000 N P DTAL,=16,0 MEM IN *B
2494 01 007D1 0359F000 N P DTAL,=16,0 MEM OUT *B
2495 01 007D3 00004000 A DATA X'4000' STACK POINTER *B
2496 01 007D4 00000002 A DATA X'00000002' DOUBLEWORD IN *B
2497 01 007D5 00003FFE A DATA X'3FFE' STACK POINTER *B
2498          *..... PLM *B
2499          * PULL 2 WORDS FROM 7FFF & 8000 *B
2500          * NO OVERFLOW OR UNDERFLOW *B
2501          * R/P16*NP17*NP18*PDC18 *B
2502          * COUNT SUPPLIED BY DRIVER *B
2503 01 007D6 00000000 A DECP16 DATA 0 *B
2504 01 007D7 0A000314 PLM,0 SP INSTRUCTION *B
2505 01 007D8 20000151 K 2,0,0,SETPSW PSW1 IN *B
2506 01 007D9 100001AE K 1,0,0,LBC+2 PSW1 OUT *B
2507 01 007DA 03840000 N P ZERBS,0,0 R IN *B
2508 01 007DB 0359FE00 N P DTAL,=2,0 R OUT *B
2509 01 007DC 0359F000 N P DTAL,=16,0 MEM IN *B
2510 01 007DD 0359F000 N P DTAL,=16,0 MEM OUT *B
2511 01 007DE 00008000 A DATA X'8000' STACK POINTER *B
2512 01 007DF 00000002 A DATA X'00000002' DOUBLEWORD IN *B
2513 01 007E0 00007FFE A DATA X'7FFE' STACK POINTER *B
2514 01 007E1 00020000 A DATA X'00020000' DOUBLEWORD OUT *B
2515          *..... PLM *B
2516          * PULL 2 WORDS FROM FFFF & 10000 *B
2517          * NO OVERFLOW OR UNDERFLOW *B
2518          * R/P15*NP16*NP17*PDC18 *B
2519          * COUNT SUPPLIED BY DRIVER *B
2520 01 007E2 00000000 A DECP15 DATA 0 *B
2521 01 007E3 0A000314 PLM,0 SP INSTRUCTION *B
2522 01 007E4 20000151 K 2,0,0,SETPSW PSW1 IN *B
2523 01 007E5 100001AE K 1,0,0,LBC+2 PSW1 OUT *B
2524 01 007E6 03840000 N P ZERBS,0,0 R IN *B
2525 01 007E7 0359FE00 N P DTAL,=2,0 R OUT *B
2526 01 007E8 0359F000 N P DTAL,=16,0 MEM IN *B
2527 01 007E9 0359F000 N P DTAL,=16,0 MEM OUT *B
2528 01 007EA 00010000 A DATA X'10000' STACK POINTER *B
2529 01 007EB 00000002 A DATA X'00000002' DOUBLEWORD IN *B
2530 01 007EC 00007FFE A DATA X'7FFE' STACK POINTER *B
2531 01 007ED 00020000 A DATA X'00020000' DOUBLEWORD OUT *B
2532          * END OF TEST MODULE LIST *
2533          * DATA 0 END OF LIST INDICATOR *
2534          *.....

```

```

2498          PAGE
2499          *..... PLM *B
2500          * PULL 2 WORDS FROM 7FFF & 8000 *B
2501          * NO OVERFLOW OR UNDERFLOW *B
2502          * R/P16*NP17*NP18*PDC18 *B
2503          * COUNT SUPPLIED BY DRIVER *B
2504 01 007D6 00000000 A DECP16 DATA 0 *B
2505 01 007D7 0A000314 PLM,0 SP INSTRUCTION *B
2506 01 007D8 20000151 K 2,0,0,SETPSW PSW1 IN *B
2507 01 007D9 100001AE K 1,0,0,LBC+2 PSW1 OUT *B
2508 01 007DA 03840000 N P ZERBS,0,0 R IN *B
2509 01 007DB 0359FE00 N P DTAL,=2,0 R OUT *B
2510 01 007DC 0359F000 N P DTAL,=16,0 MEM IN *B
2511 01 007DD 0359F000 N P DTAL,=16,0 MEM OUT *B
2512 01 007DE 00008000 A DATA X'8000' STACK POINTER *B
2513 01 007DF 00000002 A DATA X'00000002' DOUBLEWORD IN *B
2514 01 007E0 00007FFE A DATA X'7FFE' STACK POINTER *B
2515 01 007E1 00020000 A DATA X'00020000' DOUBLEWORD OUT *B
2516          *..... PLM *B
2517          * PULL 2 WORDS FROM FFFF & 10000 *B
2518          * NO OVERFLOW OR UNDERFLOW *B
2519          * R/P15*NP16*NP17*PDC18 *B
2520          * COUNT SUPPLIED BY DRIVER *B
2521 01 007E2 00000000 A DECP15 DATA 0 *B
2522 01 007E3 0A000314 PLM,0 SP INSTRUCTION *B
2523 01 007E4 20000151 K 2,0,0,SETPSW PSW1 IN *B
2524 01 007E5 100001AE K 1,0,0,LBC+2 PSW1 OUT *B
2525 01 007E6 03840000 N P ZERBS,0,0 R IN *B
2526 01 007E7 0359FE00 N P DTAL,=2,0 R OUT *B
2527 01 007E8 0359F000 N P DTAL,=16,0 MEM IN *B
2528 01 007E9 0359F000 N P DTAL,=16,0 MEM OUT *B
2529 01 007EA 00010000 A DATA X'10000' STACK POINTER *B
2530 01 007EB 00000002 A DATA X'00000002' DOUBLEWORD IN *B
2531 01 007EC 00007FFE A DATA X'7FFE' STACK POINTER *B
2532 01 007ED 00020000 A DATA X'00020000' DOUBLEWORD OUT *B
2533          * END OF TEST MODULE LIST *
2534          * DATA 0 END OF LIST INDICATOR *

```

SIGMA 5 CPU DIAGNOSTIC * SUFFIX 704174*51300
2535 01 000F9

END

LOADED

91



READER SURVEY

PUBLICATION NO. _____ TITLE: _____

IS MATERIAL PRESENTED PROPERLY:

- FULLY COVERED ?
- CLEARLY EXPLAINED ?
- WELL ILLUSTRATED ?
- WELL ORGANIZED ?
- OTHER _____

HOW DID YOU USE THIS PUBLICATION?

- FOR TROUBLESHOOTING AND REPAIR
- FOR PROGRAMMING INFORMATION
- FOR OPERATING INFORMATION
- AS A STUDENT
- AS AN INSTRUCTOR
- OTHER _____

WHAT IS YOUR POSITION?

CUSTOMER PERSONNEL

CUSTOMER ORGANIZATION _____

- TECHNICIAN
- ANALYST
- MANAGER
- OPERATOR
- PROGRAMMER
- STUDENT
- OTHER _____

SDS PERSONNEL

- CUSTOMER ENGINEER
- SALES REPRESENTATIVE
- SYSTEMS ENGINEER
- INSTRUCTOR
- STUDENT
- OTHER _____

COMMENTS: _____

STAPLE

STAPLE

FOLD

FIRST CLASS
PERMIT NO. 1026
SANTA MONICA, CALIF.

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

POSTAGE WILL BE PAID BY

SCIENTIFIC DATA SYSTEMS
701 So. Aviation Boulevard
El Segundo, California 90245

ATTN: TECHNICAL PUBLICATIONS DEPT.



CUT ALONG LINE

FOLD