

CMR11
CMV11 AA

CMR11 HOST EXERCISER
CZCMJBO

AH F914B MC
FICHE 1 OF 1

MAY 95
COPYR
MADE IN USA



Microfilm frame containing a grid of frames with illegible data.

IDENTIFICATION

B 1

SEQ 0001

PROGRAM CODE: AC-F912B-MC
PROGRAM NAME: CZCMJBO CMR-11 HOST EXERCISER
DATE CREATED: 30-MAY-79
MODIFIED: 14-JAN-80, 28-NOV-80
MAINTAINER: SPECIAL SYSTEMS KANATA

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1979,1981 BY DIGITAL EQUIPMENT OF CANADA LIMITED.

HISTORY

MOD DATE	NEW NUMBER	REASON FOR MOD.	BY:
30-MAY-80	YC-Z018E-OZ	ORIGINAL PROGRAM TO TEST CMR HOST LOGIC.	D.G.W.
14-JAN-80	AC-F912A-MC(CZCMJAO)	ORIGINAL RELEASE TO S.D.C. (U.S.) PRODUCT BECOMES CATEGORY 'A'	D.G.W.
28-NOV-80	AC-F912B-MC(CZCMJBO)	.CHANGE CNTRL CHAR FOR SFTWR SWITCH REG FROM CNTRL-S TO CNTRL-G .PRINT DEFAULT PARAMETERS AT START-UP. .ADD A CNTRL CHAR. AND A FLAG TO ALLOW PRINTING A PORT CONFIGURATION STATUS TABLE WHEN AVAILABLE. .PRINT ELAPSED TIME AT END-OF-PASS IN THE PORT LOOP BACK TEST AND IN THE EXERCISE PORTION. .CORRECT THE ORDER OF BAUD-RATE PRINTOUTS AT THE COMPLETION OF THE FIRST PASS OF THE LOGIC TEST SO THAT THE CORRECT BAUD IS SHOWN WITH THE CORRECT PORT NO. .ADD CODE WHICH PRINTS THE 'STATE' OF EACH 'REGULAR' REMOTE ON A PORT IN THE PORT CONFIGURATION STATUS PRINTOUT.	D.G.W.

TABLE OF CONTENTS

<u>HEADING</u>	<u>SECTION</u>
GENERAL	1.
REQUIREMENTS	2.
RESTRICTIONS	3.
CAUTIONS	4.
ASSUMPTIONS	5.
OPTIONS / CONTROLS	6.
LOADING	7.
STARTING / RESTARTING	8.
TEST DESCRIPTION	9.
START-UP	9.1
TEST SELECTOR	9.2
LOGIC TEST	9.3
PORT LOOP-BACK	9.4
HOST-REMOTE EXERCISE	9.5
MAINTENANCE ROUTINE	9.6
END OF PASS	9.7
ERRORS	10

1. GENERAL

THIS PROGRAM IS TO BE USED TO TEST THE COMPACT MICRO REMOTE HOST HARDWARE (CMR-11 M8990 OR CMV11 M7181-M7182) AND TO BASICALLY CHECK COMMUNICATIONS BETWEEN THE HOST HARDWARE ON THE UNIBUS AND THE REMOTE HARDWARE.

THE PROGRAM IS DIVIDED INTO THREE PARTS, PLUS A FOURTH MULTI-PURPOSE 'MAINTENANCE' ROUTINE:

- HOST LOGIC TEST
- PORT LOOP-BACK TEST
- CMR EXERCISE ROUTINE
- 'MAINTENANCE' ROUTINE

THE HOST LOGIC TEST IS INTENDED TO BE RUN WITH ALL REMOTE COMMUNICATION DISABLED OR DISCONNECTED AS IS THE PORT LOOP-BACK TEST. THE CMR EXERCISE ROUTINE, HOWEVER, CAN BE RUN WITHOUT CHANGING ANY OF THE HARDWARE FROM ITS 'INSTALLED' CONDITION. FOR THE DETAILS OF EACH TEST, SEE THE ACCOMPANYING TEXT.

2. REQUIREMENTS

HARDWARE:

PDP-11 OR LSI-11
CMR HOST MODULE(S) ON THE UNIBUS (Q-BUS)
CONSOLE TERMINAL ON LINE.

PORT JUMPERS:

FOR PORT LOOP BACK TEST A
10 PIN FEMALE BERG PLUG IS REQUIRED
WITH PINS 9 TO 2 AND 7 TO 10 JUMPED
FOR EACH PORT TO BE TESTED.

OTHER:

THE PROCESSOR MAINDECS AND CONSOLE TERMINAL TESTS
MUST HAVE BEEN RUN SUCCESSFULLY BEFORE RUNNING
THIS PROGRAM.

3. RESTRICTIONS

 (NONE)

4. CAUTIONS

 IT IS NOT ADVISABLE TO RUN THE LOGIC TEST OR THE PORT LOOP-BACK TEST ON AN 'ON-LINE' SYSTEM WITH THE REMOTES ATTACHED AS 'ON-LINE' DEVICES MAY BE AFFECTED. THE CMR EXERCISE ROUTINE, HOWEVER, MAY BE RUN UNDER THESE CONDITIONS.

COMMUNICATION PORTIONS OF THESE TESTS FROM HOST TO REMOTE AND REMOTE TO HOST MAY TAKE A CONSIDERABLE LENGTH OF TIME IF THE PORT BAUD RATES ARE SET LOW (300 BAUD).

CAUTION MUST BE EXERCISED WHEN USING THE 'MAINTENANCE' ROUTINE SO THAT FUNCTIONS ARE NOT STARTED WHICH WILL COMPROMISE ANY 'ON-LINE' SITUATIONS.

5. ASSUMPTIONS

5.1 IN THE PORT LOOP-BACK TEST, IT IS ASSUMED THAT WHEN THE OPERATOR TYPES <CR> INDICATING 'READY', HE HAS LOOPED A PORT FROM OUTPUT TO INPUT PHYSICALLY, AND HAS SELECTED THAT PORT TO BE TESTED.

5.2 IT IS ASSUMED, IN THE CMR EXERCISE ROUTINE, THAT THERE ARE REMOTES ATTACHED (AT LEAST ONE) AND THAT THEY ARE 'REGULAR'. IF THIS IS NOT THE CASE, THE ROUTINE WILL BE IDLE.

6. OPTIONS / CONTROLS

6.1 SWITCHABLE OPTIONS (COMBINABLE)

SWR =	MEANS
-----	-----
100000	DON'T HALT AFTER ERRORS
040000	DON'T PRINT ERRORS
020000	LOOP ON ERRONEOUS ROUTINE
004000	INHIBIT PROGRAM ITERATIONS
002000	DON'T PRINT END OF PASS

6.2 CONTROL CHARACTERS

CNTRL-C	RESTART TEST SELECTOR
CNTRL-T	
CNTRL-G	SHOW 'SWR' OPTIONS
CNTRL-P	PROCEED
CNTRL-O	ENTER 'MAINTENANCE' DEBUG ROUTINE
CNTRL-F	SHOW CONSOLE FILL COUNT
CNTRL-A	PRINT PORT CONFIG TABLE

6.3 NON-SWR PROCESSORS

G 1

SEQ 0006

ON PROCESSORS WITH NO PHYSICAL SWITCH REGISTER THE OPTIONS CAN BE EXAMINED OR CHANGED BY THE USE OF 'CNTRL-G' AND AT THE TIME OF AN ERROR HALT. THE PRINTOUT: SWR = XXXXXX : WILL SHOW THE OPTIONS AND THE OPERATOR CAN LEAVE THEM (<CR>) OR CHANGE THEM (YYYY)Y<CR>), THEN PROCEED (P)

7. LOADING

THE PROGRAM IS SUPPLIED IN .ABS FORMAT AND IS LOADED BY MEANS OF THE ABSOLUTE LOADER OR THE 'LOAD' COMMAND OF THE DIAGNOSTIC MONITOR XXDP.

8. STARTING / RESTARTING

STARTING ADDRESS : 200
RESTARTING ADDRESS : 204

9. TEST DESCRIPTION

9.1 START UP:

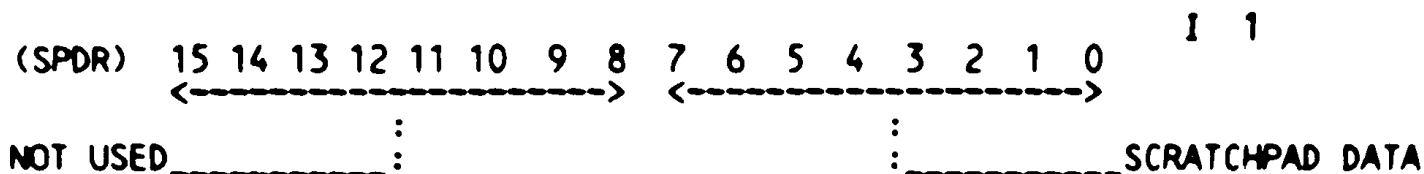
INTERFACE ADDRESS: (NNNNNN)

INTERFACE VECTOR: (NNN)

INTERFACE PRIORITY: (NN)

THE VALUE SHOWN IN PARENTHASIS ARE DEFAULT VALUES WHICH WILL BE USED IF ONLY <CR> IS ENTERED. IF A NEW VALUE IS ENTERED (XXX), IT WILL BE USED AS THE DEFAULT AT FUTURE START-UPS.

IF THE PROCESSOR HAS NO SWITCH REGISTER, THE SWITCH OPTIONS WILL NOW BE SHOWN AND MAY BE CHANGED (AS IN #6).



THE FOLLOWING TESTS ARE PERFORMED:

- TEST OF THE PWR-UP SEQUENCE
- TEST THE SCPD ADDRESS REG.
- TEST THE SCPD MEMORY
- TEST ALL OF THE CSR
- TEST INTERRUPT
- TEST PROGRAM INIT.
- CHECK SCPD ACCESS KEYS
- SIMPLE FUNCTIONS TEST

9.4 PORT LOOP-BACK TEST

THE PORT LOOP-BACK TEST USES THE PORT LOOP MAINTENANCE FUNCTION OF THE HOST FIRMWARE. THE TEST WILL RUN WITH ONE PORT OR WITH ALL PORTS SELECTED:

PORT # (<<CR> = ALL):

THE OPERATOR MUST ENSURE THAT THE PORT(S) BEING TESTED IS(ARE) LOOPED FROM XMITR TO RCVR. THIS MAY BE DONE BY PLUGGING JUMPER PLUGS INTO THE HOST PORT PLUGS OR BY SWITCHING THE LDM (M8996) EITHER AT THE HOST, OR AT ANY ONE REMOTE ON A GIVEN PORT, TO 'LOOP-BACK'. (SEE CMR-11 SYSTEMS OPTION DESCRIPTION SEC. 2.5.1).

9.5 CMR HOST-REMOTE EXERCISE

THIS ROUTINE STARTS A BACKGROUND SCAN AND WAITS FOR THE END OF A SCAN OF ALL REMOTES ON ALL PORTS ('THRU').

DURING THIS WAIT, IF ANY ALARMS OCCUR, THE BELL ON THE CONSOLE WILL RING (4 TIMES PER ALARM).

AT THE COMPLETION OF THE SCAN, THE PROGRAM SHOWS THE PORT CONFIGURATION STATUS AS FOLLOWS:

PORT #0	0(1)	1(1)	2(1)	3(1)	...
PORT #1	100(1)	101(1)	102(0)	103(0)	...
ETC.					

SHOWING EACH PORT ONLY IF THERE ARE REMOTES ON THE PORT WHICH ARE 'REGULAR' AND SHOWING THE REMOTE 'STATE' WHERE:

- (0) = INITIALIZED
- (1) = PASSIVE
- (2) = ACTIVE

***** THIS PRINTOUT ALSO OCCURS IF CNTRL-A *****
 ***** IS TYPED ANY TIME AFTER THE FIRST *****
 ***** TABLE IS PRINTED. *****

FOLLOWING THIS PRINTOUT, THE 'DIAGNOSTIC' FUNCTION IS ATTEMPTED ON EACH REMOTE PRESENT.

DURING THE REMAINDER OF THE EXERCISE, ALL REMOTES WILL BE EXERCISED WITH THE 'DIAGNOSTIC' FUNCTION AND ALL ALARMS WILL BE INDICATED (BY BELLS) AND ANY CHANGES OF REMOTE STATUS WILL BE REPORTED:

*** REMOTE STATUS CHANGE ***
 REMOTE #X

REGULAR
 REMOTE STATUS NOW IRREGULAR

J 1

\
YYYY

SEQ 0009

CAUTION SHOULD BE EXERCISED WHEN USING THIS ROUTINE
SEE SECT. 4.

THIS ROUTINE OPERATES LIKE A SIMPLE 'ODT'. HOWEVER,
IT ALLOWS EXAMINING AND/OR DEPOSITING ONLY IN THE
CMR-11 SCRATCHPAD MEMORY. FOLLOWING ARE THE OPERATING
CHARACTERS:

XXXX/	(SLASH)	OPEN AND DISPLAY CONTENTS OF THE SCPD ADDR 'XXXX'
<CR>	(RETURN)	CLOSE THE CURRENT ADDRESS.
<LF>	(LINE FEED)	CLOSE THE CURRENT ADDR. AND OPEN THE NEXT SEQUENTIAL ONE.
XXXX<CR>		DEPOSIT 'XXXX' IN THE CURRENTLY OPEN ADDRESS AND CLOSE IT.
XXXX<LF>		DEPOSIT 'XXXX' IN THE CURRENTLY OPEN ADDRESS, CLOSE IT AND OPEN THE NEXT SEQUENTIAL ONE.
G		SET 'STFUN' OR 'MAIN' TO START THE CURRENTLY LOADED FUNCTION
P		PROCEED OUT OF THE 'DE-BUG' ROUTINE AND RETURN TO THE SPOT WHICH WAS LEFT WHEN CNTRL-O WAS TYPED. (ANY PROMPTS WILL BE RE-ASKED).
(CSR ADDR)/		ALLOWS EXAMINATION OF THE HOST CSR REGISTER ONLY. NO OTHER UNIBUS ADDRESSES MAY BE EXAMINED IN THIS FASHION.

AT THE END OF EACH PASS OF ANY OF THE THREE TESTS, AN 'END PASS' MESSAGE IS PRINTED UNLESS INHIBITED BY SWITCH OPTION BIT 10.

THE PRINTOUT IS IN THE FOLLOWING FORM:

END PASS # X ERR. CNT. = Y

WHERE:

X = THE PASS NUMBER IN DECIMAL
Y = THE TOTAL NUMBER OF ERRORS SINCE START (DECIMAL).

IF THE NUMBER OF ERRORS BECOMES LARGER THAN 64K, THEN '** OVERFLOW **' WILL BE PRINTED INSTEAD.

IF SWITCH OPTION BIT 11 = 0, THIS MESSAGE WILL BE PRINTED ONLY AFTER 10 (OCTAL) PASSES THROUGH WHICHEVER TEST IS RUNNING. OTHERWISE, IT IS PRINTED EVERY PASS.

*** IN THE PORT LOOP BACK TEST AND THE CDR EXERCISE ROUTINE, THE ELAPSED TIME SINCE START-UP (OR CNTRL-C) IS PRINTED ALONG WITH THE END OF PASS MESSAGE PROVIDING THAT THE PROCESSOR IS EQUIPPED WITH A REGULAR LINE-TIME CLOCK. THE FORMAT OF THE TIME PRINTOUT IS AS FOLLOWS:

EL.TIM. HH:MM:SS:TT

(HOURS, MINUTES, SECONDS, AND TICKS WHERE ONE TICK = 1/60 SEC.)

10. ERRORS

FOR THE MOST PART, ERROR PRINTOUTS ARE SELF-EXPLANATORY. THE ERROR NUMBERS AND ADDRESSES CAN BE USED FOR REFERENCE TO THE PROGRAM LISTING FOR FURTHER DETAILS.

1-	2	SYMBOLIC DEFINITIONS
2-	1	MACRO DEFINITIONS
3-	1	TRAP CATCHERS
4-	1	SYMBOLIC LISTING
6-	1	SUPERVISOR DISPATCH
7-	1	HOST LOGIC TEST
18-	1	PORT LGOP-BACK TEST
19-	1	HOST-REMOTE EXERCISE
27-	1	MAINTENANCE ROUTINE
28-	1	UTILITY ROUTINES
29-	1	'SWITCH' MONITOR ROUTINE
30-	1	MESSAGE PRINT ROUTINE
31-	1	KEYBOARD INPUT ROUTINE
34-	1	BINARY TO ASCII CONVERSION ROUTINES
36-	1	CONSTANTS, VARIABLES & ASCII

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54

.TITLE CZCMJBO CMR(V)-11 HOST EXERCISER
.SBTTL SYMBOLIC DEFINITIONS
.NLIST TTM
.LIST ME

:INTERNAL DEFINITIONS:

177776 PS = 177776
177570 HWSWR = 177570
027636 SSWR = SWREG

:REGISTER DEFINITIONS:

000000 R0 = %0
000001 R1 = %1
000002 R2 = %2
000003 R3 = %3
000004 R4 = %4
000005 R5 = %5
000006 SP = %6
000007 PC = %7

:BUS REQUEST DEFINITIONS:

000340 P7 = 340
000300 P6 = 300
000240 P5 = 240
000200 P4 = 200
000140 P3 = 140
000100 P2 = 100
000040 P1 = 40
000000 P0 = 0

:BIT DEFINITIONS:

100000 B15 = 100000
040000 B14 = 40000
020000 B13 = 20000
010000 B12 = 10000
004000 B11 = 4000
002000 B10 = 2000
001000 B09 = 1000
000400 B08 = 400
000200 B07 = 200
000100 B06 = 100
000040 B05 = 40
000020 B04 = 20
000010 B03 = 10
000004 B02 = 4
000002 B01 = 2
000001 B00 = 1
002000 ISP = BEGIN

:INITIAL STACK POINTER

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57

.SBTTL MACRO DEFINITIONS

000001

N = 1 ;INITIAL ERROR NUMBER
 ;ERROR MACROS

.MACRO ERROR P,MSG ;***** ERROR P *****

BIT #B14,@SR
 BNE .+20
 MOV #MSG,R0
 MOV #P,ERRNUM
 JSR PC,ERR
 = N+1
 .ENDM

.MACRO DATERR P,MSG ;***** ERROR P *****

BIT #B14,@SR
 BNE .+20
 MOV #MSG,R0
 MOV #P,ERRNUM
 JSR PC,DERR
 = N+1
 .ENDM

.MACRO HLT
 JSR PC,SUHLT
 .ENDM

;PRINT MACRO (MSG ADDR IN R0)

.MACRO PNTM A ;PRINT MESSAGE
 MOV #A,R0 ;POINTED TO BY A
 JSR PC,TYPOUT
 .ENDM

;SCOPE LOOP MACRO

.MACRO SCOPE X
 JSR R5,SCRPTN
 X
 .ENDM

;INTER-PDP-11 COMPATABLE MOVE TO PS

.MACRO MTPS SRC,?LLL
 MOV SRC,-(SP)
 MOV #LLL,-(SP)
 RTI
 LLL:
 .ENDM

;REGISTER SAVE AND RESTORE MACROS

.MACRO REGSAV

58
59
60
61
62
63

JSR R5,REGSAV
.ENDM

.MACRO REGRES
JSR R5,REGRES
.ENDM

1		.SBTTL TRAP CATCHERS	
2	000000	.ENABLE ABS	
3		=	0
4	000000 000002	.WORD	.+2
5	000002 000000	.WORD	0
6	000004 016150	.WORD	DVATST
7	000006 000340	.WORD	340
8		.REPT	126.
9		.WORD	.+2
10		.WORD	0
11		.ENDR	
	000010 000012	.WORD	.+2
	000012 000000	.WORD	0
	000014 000016	.WORD	.+2
	000016 000000	.WORD	0
	000020 000022	.WORD	.+2
	000022 000000	.WORD	0
	000024 000026	.WORD	.+2
	000026 000000	.WORD	0
	000030 000032	.WORD	.+2
	000032 000000	.WORD	0
	000034 000036	.WORD	.+2
	000036 000000	.WORD	0
	000040 000042	.WORD	.+2
	000042 000000	.WORD	0
	000044 000046	.WORD	.+2
	000046 000000	.WORD	0
	000050 000052	.WORD	.+2
	000052 000000	.WORD	0
	000054 000056	.WORD	.+2
	000056 000000	.WORD	0
	000060 000062	.WORD	.+2
	000062 000000	.WORD	0
	000064 000066	.WORD	.+2
	000066 000000	.WORD	0
	000070 000072	.WORD	.+2
	000072 000000	.WORD	0
	000074 000076	.WORD	.+2
	000076 000000	.WORD	0
	000100 000102	.WORD	.+2
	000102 000000	.WORD	0
	000104 000106	.WORD	.+2
	000106 000000	.WORD	0
	000110 000112	.WORD	.+2
	000112 000000	.WORD	0
	000114 000116	.WORD	.+2
	000116 000000	.WORD	0
	000120 000122	.WORD	.+2
	000122 000000	.WORD	0
	000124 000126	.WORD	.+2
	000126 000000	.WORD	0
	000130 000132	.WORD	.+2
	000132 000000	.WORD	0
	000134 000136	.WORD	.+2
	000136 000000	.WORD	0
	000140 000142	.WORD	.+2
	000142 000000	.WORD	0

;TRAP BAD DEVICE ADDRESSES

000144	000146	.WORD	.+2
000146	000000	.WORD	0
000150	000152	.WORD	.+2
000152	000000	.WORD	0
000154	000156	.WORD	.+2
000156	000000	.WORD	0
000160	000162	.WORD	.+2
000162	000000	.WORD	0
000164	000166	.WORD	.+2
000166	000000	.WORD	0
000170	000172	.WORD	.+2
000172	000000	.WORD	0
000174	000176	.WORD	.+2
000176	000000	.WORD	0
000200	000202	.WORD	.+2
000202	000000	.WORD	0
000204	000206	.WORD	.+2
000206	000000	.WORD	0
000210	000212	.WORD	.+2
000212	000000	.WORD	0
000214	000216	.WORD	.+2
000216	000000	.WORD	0
000220	000222	.WORD	.+2
000222	000000	.WORD	0
000224	000226	.WORD	.+2
000226	000000	.WORD	0
000230	000232	.WORD	.+2
000232	000000	.WORD	0
000234	000236	.WORD	.+2
000236	000000	.WORD	0
000240	000242	.WORD	.+2
000242	000000	.WORD	0
000244	000246	.WORD	.+2
000246	000000	.WORD	0
000250	000252	.WORD	.+2
000252	000000	.WORD	0
000254	000256	.WORD	.+2
000256	000000	.WORD	0
000260	000262	.WORD	.+2
000262	000000	.WORD	0
000264	000266	.WORD	.+2
000266	000000	.WORD	0
000270	000272	.WORD	.+2
000272	000000	.WORD	0
000274	000276	.WORD	.+2
000276	000000	.WORD	0
000300	000302	.WORD	.+2
000302	000000	.WORD	0
000304	000306	.WORD	.+2
000306	000000	.WORD	0
000310	000312	.WORD	.+2
000312	000000	.WORD	0
000314	000316	.WORD	.+2
000316	000000	.WORD	0
000320	000322	.WORD	.+2
000322	000000	.WORD	0
000324	000326	.WORD	.+2

000326	000000	.WORD	0
000330	000332	.WORD	+2
000332	000000	.WORD	0
000334	000336	.WORD	+2
000336	000000	.WORD	0
000340	000342	.WORD	+2
000342	000000	.WORD	0
000344	000346	.WORD	+2
000346	000000	.WORD	0
000350	000352	.WORD	+2
000352	000000	.WORD	0
000354	000356	.WORD	+2
000356	000000	.WORD	0
000360	000362	.WORD	+2
000362	000000	.WORD	0
000364	000366	.WORD	+2
000366	000000	.WORD	0
000370	000372	.WORD	+2
000372	000000	.WORD	0
000374	000376	.WORD	+2
000376	000000	.WORD	0
000400	000402	.WORD	+2
000402	000000	.WORD	0
000404	000406	.WORD	+2
000406	000000	.WORD	0
000410	000412	.WORD	+2
000412	000000	.WORD	0
000414	000416	.WORD	+2
000416	000000	.WORD	0
000420	000422	.WORD	+2
000422	000000	.WORD	0
000424	000426	.WORD	+2
000426	000000	.WORD	0
000430	000432	.WORD	+2
000432	000000	.WORD	0
000434	000436	.WORD	+2
000436	000000	.WORD	0
000440	000442	.WORD	+2
000442	000000	.WORD	0
000444	000446	.WORD	+2
000446	000000	.WORD	0
000450	000452	.WORD	+2
000452	000000	.WORD	0
000454	000456	.WORD	+2
000456	000000	.WORD	0
000460	000462	.WORD	+2
000462	000000	.WORD	0
000464	000466	.WORD	+2
000466	000000	.WORD	0
000470	000472	.WORD	+2
000472	000000	.WORD	0
000474	000476	.WORD	+2
000476	000000	.WORD	0
000500	000502	.WORD	+2
000502	000000	.WORD	0
000504	000506	.WORD	+2
000506	000000	.WORD	0

000510	000512	.WORD	+2
000512	000000	.WORD	0
000514	000516	.WORD	+2
000516	000000	.WORD	0
000520	000522	.WORD	+2
000522	000000	.WORD	0
000524	000526	.WORD	+2
000526	000000	.WORD	0
000530	000532	.WORD	+2
000532	000000	.WORD	0
000534	000536	.WORD	+2
000536	000000	.WORD	0
000540	000542	.WORD	+2
000542	000000	.WORD	0
000544	000546	.WORD	+2
000546	000000	.WORD	0
000550	000552	.WORD	+2
000552	000000	.WORD	0
000554	000556	.WORD	+2
000556	000000	.WORD	0
000560	000562	.WORD	+2
000562	000000	.WORD	0
000564	000566	.WORD	+2
000566	000000	.WORD	0
000570	000572	.WORD	+2
000572	000000	.WORD	0
000574	000576	.WORD	+2
000576	000000	.WORD	0
000600	000602	.WORD	+2
000602	000000	.WORD	0
000604	000606	.WORD	+2
000606	000000	.WORD	0
000610	000612	.WORD	+2
000612	000000	.WORD	0
000614	000616	.WORD	+2
000616	000000	.WORD	0
000620	000622	.WORD	+2
000622	000000	.WORD	0
000624	000626	.WORD	+2
000626	000000	.WORD	0
000630	000632	.WORD	+2
000632	000000	.WORD	0
000634	000636	.WORD	+2
000636	000000	.WORD	0
000640	000642	.WORD	+2
000642	000000	.WORD	0
000644	000646	.WORD	+2
000646	000000	.WORD	0
000650	000652	.WORD	+2
000652	000000	.WORD	0
000654	000656	.WORD	+2
000656	000000	.WORD	0
000660	000662	.WORD	+2
000662	000000	.WORD	0
000664	000666	.WORD	+2
000666	000000	.WORD	0
000670	000672	.WORD	+2

000672	000000	.WORD	0
000674	000676	.WORD	.+2
000676	000000	.WORD	0
000700	000702	.WORD	.+2
000702	000000	.WORD	0
000704	000706	.WORD	.+2
000706	000000	.WORD	0
000710	000712	.WORD	.+2
000712	000000	.WORD	0
000714	000716	.WORD	.+2
000716	000000	.WORD	0
000720	000722	.WORD	.+2
000722	000000	.WORD	0
000724	000726	.WORD	.+2
000726	000000	.WORD	0
000730	000732	.WORD	.+2
000732	000000	.WORD	0
000734	000736	.WORD	.+2
000736	000000	.WORD	0
000740	000742	.WORD	.+2
000742	000000	.WORD	0
000744	000746	.WORD	.+2
000746	000000	.WORD	0
000750	000752	.WORD	.+2
000752	000000	.WORD	0
000754	000756	.WORD	.+2
000756	000000	.WORD	0
000760	000762	.WORD	.+2
000762	000000	.WORD	0
000764	000766	.WORD	.+2
000766	000000	.WORD	0
000770	000772	.WORD	.+2
000772	000000	.WORD	0
000774	000776	.WORD	.+2
000776	000000	.WORD	0

```

1          .SBTTL SYMBOLIC LISTING
2
3          =          200
4 000200 000167 001574  JMP      BEGIN
5 000204 012706 002000  MOV     #ISP,SP
6 000210 000005          RESET
7 000212 000167 002276  JMP     RESTRT
8
9          =          2000
10
11 002000 012706 002000  BEGIN: MOV     #ISP,SP          ;INITIALIZE THE STACK
12 002004 000005          RESET          ;CLEAR THE WORLD
13 002006          MTPS     #P7          ;STOP INTERRUPTS
14 002006 012746 000340  MOV     #P7, -(SP)
15 002012 012746 002020  MOV     #64$, -(SP)
16 002016 000002          RTI
17
18 002020 005067 025544  64$:  CLR     DEVER
19 002024 105067 025510  CLRB   LSIFLG          ;PREPARE TO CHECK FOR LSI-11
20 002030 012737 016122 000004  MOV     #LSTST, @#4
21 002036 012737 000340 000006  MOV     #P7, @#6
22 002044 005737 177776          TST     @#177776          ;CHECK FOR PSW & SET FLAG
23 002050 005067 025564          CLR     SWRFLG
24 002054 012737 016132 000004  MOV     #SRTST, @#4          ;SET UP TO TRAP IF NO SWR.
25 002062 012737 000340 000006  MOV     #P7, @#6
26 002070 012767 177570 025536  MOV     #HWSR, SR          ;SET UP TO CHECK IF ANY HD SWR
27 002076 005777 025532          TST     @SR          ;SEE IF IT'S THERE
28
29 002102          PNTM    TSTHDR
30 002102 012700 020712  MOV     #TSTHDR, R0          ;PRINT MESSAGE
31 002106 004767 015160  JSR    PC, TYP0UT          ;POINTED TO BY TSTHDR
32
33 002112          PROMT:  PNTM    FRAD          ;ASK FOR 1ST UNIBUS ADDR OF CMR-11
34 002112 012700 021277  MOV     #FRAD, R0          ;PRINT MESSAGE
35 002116 004767 015150  JSR    PC, TYP0UT          ;POINTED TO BY FRAD
36 002122 016767 025530 016110  MOV     DEVADR, KBBUF          ;LOAD DEFAULT ADDRESS
37 002130 004767 014172          JSR    PC, PRDFLT          ;GO PRINT DEFAULT
38 002134 012767 002112 025444  MOV     #PROMT, PCONT          ;LOAD RETURN ADDRESS
39 002142 004767 015166          JSR    PC, INPKB          ;GET KBD INPUT
40 002146 016767 016066 025502  MOV     KBBUF, DEVADR          ;REPLACE CMR-11 BASE ADDR
41
42 002154 026727 016060 164000  CMP     KBBUF, #164000          ;IS IT WITHIN LIMIT?
43 002162 103006          BHIS   PRMT1          ;YES, TEST IT
44 002164          PNTM    AGAIN          ;NO, ERROR & ASK AGAIN
45 002164 012700 021324  MOV     #AGAIN, R0          ;PRINT MESSAGE
46 002170 004767 015076  JSR    PC, TYP0UT          ;POINTED TO BY AGAIN
47
48 002174 000167 177712          JMP     PROMT
49 002200 012737 016150 000004  PRMT1: MOV     #DVATST, @#4
50 002206 005777 025444          TST     @DEVADR          ;IS IT A GOOD ADDR?
51 002212 012737 016370 000004  MOV     #NDR, @#4          ;SET UP FOR FURTHER TRAPS
52 002220 005037 000006          CLR     @#6          ;ALLOW CLOCK TO FUNCTION
53
54 002224          PRMT2:  PNTM    DVECT          ;YES, ASK FOR DEVICE VECTOR
55 002224 012700 021357  MOV     #DVECT, R0          ;PRINT MESSAGE
56 002230 004767 015036  JSR    PC, TYP0UT          ;POINTED TO BY DVECT
57 002234 016767 025414 015776  MOV     VECTOR, KBBUF          ;LOAD DEFAULT VECTOR
58 002242 004767 014060          JSR    PC, PRDFLT          ;GO PRINT DEFAULT
59 002246 012767 002224 025332  MOV     #PRMT2, PCONT          ;LOAD RETURN ADDRESS
60 002254 004767 015054          JSR    PC, INPKB          ;GET KBD INPUT
61 002260 016767 015754 025366  MOV     KBBUF, VECTOR          ;REPLACE CMR-11 VECTOR
    
```

46	002266	026727	015746	000776		CMP	KBBUF,#776	: IS IT WITHIN LIMITS?
47	002274	101406				BLOS	PRMT3	: YES, GET PRIORITY
48	002276					PNTM	AGAIN	: NO, ERROR AND ASK AGAIN
	002276	012700	021324			MOV	#AGAIN,R0	: PRINT MESSAGE
	002302	004767	014764			JSR	PC,TYP0UT	: POINTED TO BY AGAIN
49	002306	000167	177712			JMP	PRMT2	
50	002312				PRMT3:	PNTM	DVPRIO	: ASK FOR CMR-11 PRIORITY
	002312	012700	021403			MOV	#DVPRIO,R0	: PRINT MESSAGE
	002316	004767	014750			JSR	PC,TYP0UT	: POINTED TO BY DVPRIO
51	002322	016767	025332	015710		MOV	FKPRIO,KBBUF	: LOAD DEFAULT PRIORITY
52	002330	004767	013772			JSR	PC,PRDFLT	: GO PRINT DEFAULT
53	002334	012767	002312	025244		MOV	#PRMT3,PCONT	: LOAD RETURN ADDRESS
54	002342	004767	014766			JSR	PC,INPKB	: GET KBD INPUT
55	002346	026727	015666	000007		CMP	KBBUF,#7	: IS IT WITHIN LIMITS?
56	002354	003406				BLE	PRMT4	: LOW ENOUGH, OKAY
57	002356					PNTM	AGAIN	: NO, ERROR AND ASK AGAIN
	002356	012700	021324			MOV	#AGAIN,R0	: PRINT MESSAGE
	002362	004767	014704			JSR	PC,TYP0UT	: POINTED TO BY AGAIN
58	002366	000167	177720			JMP	PRMT3	
59	002372	026727	015642	000004	PRMT4:	CMP	KBBUF,#4	: HIGH ENOUGH?
60	002400	002006				BGE	PRMT5	: YES, OKAY
61	002402					PNTM	AGAIN	: NO, ERROR & ASK AGAIN
	002402	012700	021324			MOV	#AGAIN,R0	: PRINT MESSAGE
	002406	004767	014660			JSR	PC,TYP0UT	: POINTED TO BY AGAIN
62	002412	000167	177674			JMP	PRMT3	
63	002416	006367	015616		PRMT5:	ASL	KBBUF	
64	002422	006367	015612			ASL	KBBUF	
65	002426	006367	015606			ASL	KBBUF	
66	002432	006367	015602			ASL	KBBUF	
67	002436	006367	015576			ASL	KBBUF	
68	002442	016767	015572	025144		MOV	KBBUF,PRIO	: SHIFT IT INTO PLACE
69	002450	004767	013606			JSR	PC,DEVGEN	: LOAD NEW PRIORITY
70								: GENERATE CMR-11 ADDRESSES
71								
72								: TEST FOR CLOCK CSR ADDRESS AND SET
73								: CLOCK FLAG ACCORDINGLY:
74	002454	112767	177777	025047		MOVB	#-1,CLKFLG	: SET CLOCK FLAG
75	002462	012737	002566	000004		MOV	#CLKTRP,@#4	: SET UP TO TRAP
76	002470	005777	025172			TST	@LCS	: TEST LINE CLOCK STAT.
77								: A TRAP WILL CLEAR FLAG.
78								
79								: CLEAR OUT ALL TABLES FOR START-UP
80								
81	002474	005067	023764			CLR	EXTBL1	
82	002500	005067	024160			CLR	EXTBL2	
83	002504	005067	024354			CLR	EXTBL3	
84	002510	005067	024550			CLR	EXTBL4	
85	002514	012737	016370	000004	RESTR:	MOV	#HDER,@#4	: RESET TRAPS
86								
87	002522	012737	002574	000100		MOV	#CLKINT,@#100	: SET CLOCK VECTOR
88	002530	012737	000340	000102		MOV	#340,@#102	
89	002536					MTPS	#P7	: RAISE C.P. PRIORITY
	002536	012746	000340			MOV	#P7,-(SP)	
	002542	012746	002550			MOV	#64\$,-(SP)	
	002546	000002				RTI		
90	002550				64\$:			

91	002550	005767	025064		TST	SWRFLG	:NON-SWR-PROCESSOR?
92	002554	001402			BEQ	1\$	
93	002556	004767	014136		JSR	PC,SWDMP	:YES, DUMP SWITCH LOCATION
94	002562	000167	000076	1\$:	JMP	BCONT	
95							
96	002566	105067	024737	CLKTRP:	CLRB	CLKFLG	:CLEAR CLOCK FLAG
97	002572	000002			RTI		:AND RETURN
98							

```
1  
2  
3  
4 002574 112767 177777 024726 CLKINT: MOVB #1,CLKFL1 ;SET 'CLK TICKING' FLAG  
5 002602 105267 024743 INCB TICKS ;UPDATE TICKS  
6 002606 126727 024737 000074 CMPB TICKS,#60. ;REACHED A SECOND?  
7 002614 103422 BLO CLKEXT ;NO, JUST RETURN  
8 002616 105067 024727 CLRB TICKS ;YES, 0 - TICKS  
9 002622 105267 024720 INCB SECS ;AND UPDATE SECONDS  
10 002626 126727 024714 000074 CMPB SECS,#60. ;REACHED A MINUTE?  
11 002634 103412 BLO CLKEXT ;NO, JUST RETURN  
12 002636 105067 024704 CLRB SECS ;YES, 0 - SECONDS  
13 002642 105267 024673 INCB MINS ;AND UPDATE MINUTES  
14 002646 126727 024667 000074 CMPB MINS,#60. ;REACHED AN HOUR?  
15 002654 103402 BLO CLKEXT ;NO, JUST RETURN  
16 002656 105267 024654 INCB HOURS ;YES, UPDATE HOURS  
17 002662 000002 CLKEXT: RTI ;RETURN FROM CLOCK INTERRUPT
```

```

1                                     .SBTTL SUPERVISOR DISPATCH
2
3
4 002664 000005          BCONT:  RESET          :ALWAYS DO RESET UPON ARRIVAL
5 002666          MTPS          #P7           :KILL INTERRUPTS
6 002666 012746 000340  MOV          #P7,-(SP)
7 002672 012746 002700  MOV          #64$,-(SP)
8 002676 000002          RTI
9 002700          64$:
10 002700 105067 024627  CLRB         CNFGFL          :CLEAR CONFIG FLAG
11 002704 105067 024620  CLRB         CLKFL1         :CLR CLOCK TICKING FLAG
12 002710 105067 024635  CLRB         TICKS          :CLEAR RUN TIME CLOCK TIME
13 002714 105067 024626  CLRB         SECS
14 002720 105067 024615  CLRB         MINS
15 002724 105067 024606  CLRB         HOURS
16 002730 105067 024611  CLRB         SAFE          :CLEAR 'FILL CNT SAFETY FLAG'
17 002734 105067 024574  CLRB         DBFLG         :CLEAR 'DEBUG' FLAG
18 002740          PNTM        TSTSEL         :PRINT 'SELECT TEST (<CR>=HELP)''
19 002740 012700 021676  MOV          #TSTSEL,R0    :PRINT MESSAGE
20 002744 004767 014322  JSR          PC,TYPOUT     :POINTED TO BY TSTSEL
21 002750 012767 000001 015262  MOV          #1,KBBUF      :DEFAULT LOGIC TEST
22 002756 012767 002664 024622  MOV          #BCONT,PCONT  :LOAD RETURN ADDRESS
23 002764 004767 014344  JSR          PC,INPKB      :GET KBD INPUT
24 002770 005067 024612  CLR          PCONT
25 002774 026727 015240 000001  CMP          KBBUF,#1      :CHECK IT FOR '1'
26 003002 001010          BNE          BCNT1       :IF = 1, DO LOGIC TESTS
27 003004 105767 024521  TSTB        CLKFLG        :IS THERE A CLOCK CSR?
28 003010 001403          BEQ          BCNT0       :NO, FORGET IT.
29 003012 042777 000100 024646  BIC          #100,@LCS     :YES, ENSURE IT'S IDLE
30 003020 000167 000120  BCNT0:  JMP          LOGTST  :GO TO LOGIC TEST
31 003024 026727 015210 000002  BCNT1:  CMP          KBBUF,#2 :CHECK IT FOR '2'
32 003032 001010          BNE          BCNT2       :IS THERE A CLOCK CSR?
33 003034 105767 024471  TSTB        CLKFLG        :NO, FORGET IT
34 003040 001403          BEQ          BCNT3       :YES, ENABLE IT!!
35 003042 052777 000100 024616  BIS          #100,@LCS     :IF '2' DO PORT LOOP TEST
36 003050 000167 006056  BCNT3:  JMP          PORTLP  :CHECK IT FOR '3'
37 003054 026727 015160 000003  BCNT2:  CMP          KBBUF,#3 :IF NOT, GIVE HELP
38 003062 001010          BNE          TSTRSK     :IS THERE A CLOCK CSR?
39 003064 105767 024441  TSTB        CLKFLG        :NO, FORGET IT
40 003070 001403          BEQ          BCNT4       :YES, ENABLE IT!!
41 003072 052777 000100 024566  BIS          #100,@LCS     :IF '3' DO EXERCISE.
42 003100 000167 007374  BCNT4:  JMP          CMREXC
43 003104 026727 015130 000004  TSTRSK:  CMP          KBBUF,#4 :DID HE TYPE A 4?
44 003112 001006          BNE          HLPASK      :NO, BUT HE HAD BETTER!!
45 003114          PNTM        HELPM          :PRINT HELPFUL INFO
46 003114 012700 021725  MOV          #HELPM,R0     :PRINT MESSAGE
47 003120 004767 014146  JSR          PC,TYPOUT     :POINTED TO BY HELPM
48 003124 000167 177534  JMP          BCONT        :AND RE-ASK
49 003130          HLPASK:  PNTM        HLPDSR     :PRINT 'IF HELP IS REQ'D...'
50 003130 012700 022714  MOV          #HLPDSR,R0    :PRINT MESSAGE
51 003134 004767 014132  JSR          PC,TYPOUT     :POINTED TO BY HLPDSR
52 003140 000167 177520  JMP          BCONT

```

```

1          .SBTTL  HOST LOGIC TEST
2
3
4          ;CHECK POWER UP OF THE HOST MICRO FIRMWARE.
5
6 003144 005067 024500          LOGTST: CLR      VERFLG          ;CLEAR VERSION PRINT FLAG
7 003150 012767 000010 024416  MOV      #10,ITER        ;SET ITERATION COUNT
8 003156 005067 024420          CLR      PASCNT         ;CLEAR PASS COUNT
9 003162 105067 024354          CLR      OFLOF         ;CLEAR 'OVERFLOW' FLAG
10 003166 005067 024400         CLR      ERRCNT        ;CLEAR ERROR COUNT
11 003172          PNTM      LGKTSM          ;PRINT 'LOGIC TEST'
    003172 012700 022761         MOV      #LGKTSM,R0     ;PRINT MESSAGE
    003176 004767 014070         JSR      PC,TYPEOUT     ;POINTED TO BY LGKTSM
12 003202 000005          STO:   RESET          ;INIT
13 003204 012767 000000 024352  MOV      #0,CNTR        ;SET UP TIMER
14 003212 116767 177774 177772  MOV      MOV,MOV        ;MOV  GIVE THE MICRO TIME
15 003220 116767 177766 177764  MOV      MOV,MOV        ;MOV  TO GET GOING.
16 003226 116767 177760 177756  MOV      MOV,MOV
17 003234 116767 177752 177750  MOV      MOV,MOV
18 003242 116767 177744 177742  MOV      MOV,MOV
19 003250 116767 177736 177734  MOV      MOV,MOV
20 003256 032777 002000 024302  BIT      #B10,@CSR      ;IS 'PWRUP' SET?
21 003264 001016          BNE     ST1            ;IF YES, OKAY
22 003266          ERROR   W,N,NOGO        ;ERROR:HOST U.P. FAILED TO START.
    003266 032777 040000 024340  BIT      #B14,@SR      ;***** ERROR 1 *****
    003274 001007          BNE     .+20
    003276 012700 024031          MOV      #NOGO,R0
    003302 012767 000001 013300  MOV      #1,ERRNUM
    003310 004767 013126          JSR      PC,ERR
    N          =          N+1
23 003314          SCOPE  STO
    003314 004567 012670          JSR      R5,SCRPTN
    003320 003202          STO
24 003322 012767 102000 013262  ST1:   MOV      #102000,GOOD        ;NOW WATCH FOR INTR
25 003330 017767 024232 013256  MOV      @CSR,BAD
26 003336 042767 075777 013250  BIC      #75777,BAD
27 003344 026767 013242 013242  CMP      GOOD,BAD
28 003352 001421          BEQ     ST2            ;INTR & PWRUP SET; OKAY
29 003354 005367 024204          DEC     CNTR          ;WAIT A BIT
30 003360 001360          BNE     ST1
31 003362          DATERR W,N,NOTUP        ;ERROR:PWR UP SEQ NOT COMPLETED(BAD CSR)
    003362 032777 040000 024244  BIT      #B14,@SR      ;***** ERROR 2 *****
    003370 001007          BNE     .+20
    003372 012700 024066          MOV      #NOTUP,R0
    003376 012767 000002 013204  MOV      #2,ERRNUM
    003404 004767 013116          JSR      PC,DERR
    N          =          N+1
32 003410          SCOPE  STO
    003410 004567 012574          JSR      R5,SCRPTN
    003414 003202          STO
33 003416 017767 024144 013170  ST2:   MOV      @CSR,BAD        ;NOW CHK CSR FOR 0'S
34 003424 042767 102000 013162  BIC      #102000,BAD   ;EXCEPT PWRUP & INTR
35 003432 005067 013154          CLR     GOOD
36 003436 026767 013150 013150  CMP      GOOD,BAD
37 003444 001416          BEQ     ST3            ;CSR = 0'S, O.K.
    
```



```

1
2
3
4
5 003542 005077 024062          ST6:  CLR      @SPAR          ;TEST CLEAR OF SPAR
6 003546 005067 013040          CLR      GOOD
7 003552 017767 024052 013034  MOV      @SPAR,BAD        ;SPAR CLEAR?
8 003560 001416                BEQ      ST60             ;YES, OK
9 003562                DATERR  W,SPRCLR        ;ERROR:CAN'T CLEAR SPAR
                                ;***** ERROR 4 *****

    003562 032777 040000 024044  BIT      #B14,@SR
    003570 001007                BNE      .+20
    003572 012700 024164          MOV      #SPRCLR,R0
    003576 012767 000004 013004  MOV      #4,ERRNUM
    003604 004767 012716          JSR      PC,DERR
                                =
                                N
                                SCOPE
10 003610                JSR      ST6
    003610 004567 012374          JSR      R5,SCPRTN
    003614 003542                ST6
11 003616 012767 177777 023760  ST60:  MOV      #-1,PAT        ;SET UP FOR SLIDING 0 TEST
12 003624 012767 000001 023746  MOV      #B00,MASK        ;SET BIT MASK
13 003632 016767 023746 012752  ST61:  MOV      PAT,GOOD
14 003640 042767 176000 012744  BIC      #176000,GOOD
15 003646 016777 012740 023754  MOV      GOOD,@SPAR        ;LOAD PATTERN
16 003654 017767 023750 012732  MOV      @SPAR,BAD        ;AND READ IT BACK
17 003662 042767 000400 012724  BIC      #B08,BAD
18 003670 042767 000400 012714  BIC      #B08,GOOD
19 003676 026767 012710 012710  CMP      GOOD,BAD
20 003704 001416                BEQ      ST62
21 003706                DATERR  W,SPRDAT        ;ERROR:BAD BITS IN SPAR
                                ;***** ERROR 5 *****

    003706 032777 040000 023720  BIT      #B14,@SR
    003714 001007                BNE      .+20
    003716 012700 024211          MOV      #SPRDAT,R0
    003722 012767 000005 012660  MOV      #5,ERRNUM
    003730 004767 012572          JSR      PC,DERR
                                =
                                N
                                SCOPE
22 003734                JSR      ST61
    003734 004567 012250          JSR      R5,SCPRTN
    003740 003632                ST61
23 003742 032767 001000 023634  ST62:  BIT      #B09,PAT
24 003750 001414                BEQ      ST7
25 003752 012767 177777 023624  MOV      #-1,PAT
26 003760 046767 023614 023616  BIC      MASK,PAT
27 003766 006367 023606          ASL      MASK
28 003772 004767 012670          JSR      PC,MONIT
29 003776 000167 177630          JMP      ST61
                                ;AND CONTINUE

```



```

1          ;SCRATCH PAD MEMORY TEST
2          ; (1) UP COUNT IN REQ SCPD WHILE FILLING RES SCPD WITH 377'S
3          ; (2) UP COUNT IN RES SCPD WHILE FILLING REQ SCPD WITH 377'S
4
5 004002 005077 023622 ST7: CLR @SPAR ;ADDRESS 0
6 004006 005067 023640 CLR WORD
7 004012 016777 023634 023612 ST70: MOV WORD,@SPDR ;LOAD REQUEST SCRATCH PAD
8 004020 052777 001000 023602 BIS #B09,@SPAR
9 004026 012777 000377 023576 MOV #377,@SPDR
10 004034 042777 001000 023566 BIC #B09,@SPAR
11 004042 027727 023562 000377 CMP @SPAR,#377 ;DONE.
12 004050 103005 BHIS ST71
13 004052 005277 023552 INC @SPAR
14 004056 105267 023570 INCB WORD
15 004062 000753 BR ST70
16 004064 005077 023540 ST71: CLR @SPAR ;NOW...CHECK IT
17 004070 005067 023556 CLR WORD
18 004074 016767 023552 012510 ST80: JN WORD,GOOD
19 004102 017767 023524 012504 MOV @SPDR,BAD
20 004110 026767 012476 012476 CMP GOOD,BAD
21 004116 001432 BEQ ST81 ;O.K. CONTINUE
22 004120 DATERR W,ISATE ;ERROR:REQ SCPD ADR TEST ERROR
    ;***** ERROR 6 *****
    004120 032777 040000 023506 BIT #B14,@SR
    004126 001007 BNE .+20
    004130 012700 024240 MOV #ISATE,R0
    004134 012767 000006 012446 MOV #6,ERRNUM
    004142 004767 012360 JSR PC,DERR
    000007 = N+1
23 004146 032777 040000 023460 N BIT #B14,@SR ;PRINT ALLOWED?
24 004154 001010 BNE ST80S ;NO, SKIP IT
25 004156 PNTM SCPAD
    004156 012700 022776 MOV #SCPAD,R0 ;PRINT MESSAGE
    004162 004767 013104 JSR PC,TYP0UT ;POINTED TO BY SCPAD
26 004166 017700 023436 MOV @SPAR,R0 ;PRINT SCPD ADDRESS OF ERROR
27 004172 004767 014120 JSR PC,OC1PNT
28 004176 ST80S: SCOPE ST80
    004176 004567 012006 JSR R5,SCPRTN
    004202 004074 ST80
29 004204 027727 023420 000377 ST81: CMP @SPAR,#377
30 004212 103005 BHIS ST82
31 004214 005277 023410 INC @SPAR
32 004220 105267 023426 INCB WORD
33 004224 000723 BR ST80
34 004226 012777 001000 023374 ST82: MOV #1000,@SPAR
35 004234 012767 000377 012350 MOV #377,GOOD
36 004242 017767 023364 012344 ST90: MOV @SPDR,BAD
37 004250 026767 012336 012336 CMP GOOD,BAD ;RES SCPD ALL 1'S?
38 004256 001432 BEQ ST91 ;YES, O.K.
39 004260 DATERR W,OSDTE ;ERROR:RESULT SCPD DATA TEST ERROR
    ;***** ERROR 7 *****
    004260 032777 040000 023346 BIT #B14,@SR
    004266 001007 BNE .+20
    004270 012700 024271 MOV #OSDTE,R0
    004274 012767 000007 012306 MOV #7,ERRNUM
    004302 004767 012220 JSR PC,DEKR
    000010 = N
    
```

40	004306	032777	040000	023320		BIT	#B14,@SR		:PRINT ALLOWED?
41	004314	001010				BNE	ST90S		:NO, SKIP IT
42	004316					PNTM	SCPAD		
	004316	012700	022776			MOV	#SCPAD,RO		:PRINT MESSAGE
	004322	004767	012744			JSR	PC,TYP0UT		:POINTED TO BY SCPAD
43	004326	017700	023276			MOV	@SPAR,RO		:PRINT SCPD ADDRESS OF ERROR
44	004332	004767	013760			JSR	PC,OC1PNT		
45	004336				ST90S:	SCOPE	ST90		
	004336	004567	011646			JSR	R5,SCPRTN		
	004342	004242				ST90			
46	004344	027727	023260	001377	ST91:	CMP	@SPAR,#1377		:ALL DONE?
47	004352	001403				BEQ	ST10		:IF YES, REVERSE IN & OUT
48	004354	005277	023250			INC	@SPAR		
49	004360	000730				BR	ST90		
50									
51	004362	012777	001000	023240	ST10:	MOV	#B09,@SPAR		:DO IT AGAIN WITH A SWITCH
52	004370	005067	023256			CLR	WORD		
53	004374	016777	023252	023230	ST100:	MOV	WORD,@SPDR		:COUNT DOWN UNTIL HALF-WAY
54	004402	042777	001000	023220		BIC	#B09,@SPAR		
55	004410	012777	000377	023214		MOV	#377,@SPDR		:AND LOAD REQ SCPD WITH 1'S
56	004416	052777	001000	023204		BIS	#B09,@SPAR		
57	004424	027727	023200	001377		CMP	@SPAR,#1377		:DONE?
58	004432	103005				BHIS	ST101		:YES
59	004434	005277	023170			INC	@SPAR		:NO, KEEP GOING DOWN
60	004440	105267	023206			INCB	WORD		
61	004444	000753				BR	ST100		
62	004446	012777	001000	023154	ST101:	MOV	#B09,@SPAR		:NOW, ...CHECK IT
63	004454	005067	023172			CLR	WORD		
64	004460	016767	023166	012124	ST110:	MOV	WORD,GOOD		
65	004466	017767	023140	012120		MOV	@SPDR,BAD		
66	004474	026767	012112	012112		CMP	GOOD,BAD		
67	004502	001432				BEQ	ST111		:OUT ADDR OK?
68	004504					DATERR	W,OSATE		:ERROR:RESULT SCPD ADR TEST ERROR
									:***** ERROR 10 *****
	004504	032777	040000	023122		BIT	#B14,@SR		
	004512	001007				BNE	.+20		
	004514	012700	024326			MOV	#OSATE,RO		
	004520	012767	000010	012062		MOV	#10,ERRNUM		
	004526	004767	011774			JSR	PC,DERR		
		000011				=	N+1		
69	004532	032777	040000	023074	N	BIT	#B14,@SR		:PRINT ALLOWED?
70	004540	001010				BNE	ST110S		:NO, SKIP IT
71	004542					PNTM	SCPAD		
	004542	012700	022776			MOV	#SCPAD,RO		:PRINT MESSAGE
	004546	004767	012520			JSR	PC,TYP0UT		:POINTED TO BY SCPAD
72	004552	017700	023052			MOV	@SPAR,RO		:PRINT SCPD ADR OF ERROR
73	004556	004767	013534			JSR	PC,OC1PNT		
74	004562				ST110S:	SCOPE	ST110		
	004562	004567	011422			JSR	R5,SCPRTN		
	004566	004460				ST110			
75	004570	027727	023034	001377	ST111:	CMP	@SPAR,#1377		
76	004576	103005				BHIS	ST112		
77	004600	005277	023024			INC	@SPAR		
78	004604	105267	023042			INCB	WORD		
79	004610	000723				BR	ST110		
80	004612	012767	000377	011772	ST112:	MOV	#377,GOOD		:NOW CHECK REQ SCPD
81	004620	005077	023004			CLR	@SPAR		

82	004624	017767	023002	011762	ST120:	MOV	@SPDR,BAD	
83	004632	026767	011754	011754		CMP	GOOD,BAD	:ALL 1'S?
84	004640	001432				BEQ	ST121	:YES, CHECK THE REST
85	004642					DATERR	W,ISDTE	:ERROR:REQ SCPD DATA TEST ERR
	004642	032777	040000	022764		BIT	#B14,@SR	:***** ERROR 11 *****
	004650	001007				BNE	.+20	
	004652	012700	024362			MOV	#ISDTE,RO	
	004656	012767	000011	011724		MOV	#11,ERRNUM	
	004664	004767	011636			JSR	PC,DERR	
		000012			N	=	N+1	
86	004670	032777	040000	022736		BIT	#B14,@SR	:PRINT ALLOWED?
87	004676	001010				BNE	ST120S	:NO, SKIP IT
88	004700					PNTM	SCPAD	
	004700	012700	022776			MOV	#SCPAD,RO	:PRINT MESSAGE
	004704	004767	012362			JSR	PC,TYPOUT	:POINTED TO BY SCPAD
89	004710	017700	022714			MOV	@SPAR,RO	:PRINT SCPD ADR OF ERROR
90	004714	004767	013376			JSR	PC,OCIPNT	
91	004720				ST120S:	SCOPE	ST120	
	004720	004567	011264			JSR	R5,SCPRTN	
	004724	004624				ST120		
92	004726	027727	022676	000377	ST121:	CMP	@SPAR,#377	:DONE REQ SCPD?
93	004734	001403				BEQ	ST13	:YES, CHECK CLR SC. PAD
94	004736	005277	022666			INC	@SPAR	
95	004742	000730				BR	ST120	:NO, KEEP CHECKING

```

1
2
3
4
5 004744 005077 022660 ST13: CLR @SPAR ;START AT 0
6 004750 012777 000000 022654 ST130: MOV #0,@SPDR ;THIS LOC (#0) CAN BE CHANGED
7 004756 027727 022646 000377 CMP @SPAR,#377 ; TO ANYTHING AND THIS RTN
8 004764 001403 BEQ ST131A ; CAN BE USED FOR ANY DATA
9 004766 005277 022636 INC @SPAR ; FILL-UP & TEST
10 004772 000766 BR ST130
11 004774 012777 001000 022626 ST131A: MOV #B09,@SPAR ;ADDR RES. SCPD
12 005002 012777 000000 022622 ST131B: MOV #0,@SPDR ;THIS '#0' MUST ALSO BE CHANGED
13 005010 027727 022614 001377 CMP @SPAR,#1377
14 005016 001403 BEQ ST131 ;FILL UP WHOLE RES. SCPD
15 005020 005277 022604 INC @SPAR
16 005024 000766 BR ST131B
17 005026 005077 022576 ST131: CLR @SPAR ;NOW CHECK IT
18 005032 012767 000000 011552 MOV #0,GOOD ;THIS LOC (#0) MUST BE CHANGED
19 005040 017767 022566 011546 ST132: MOV @SPDR,BAD ; TO CORRESPOND TO ST130
20 005046 026767 011540 011540 CMP GOOD,BAD ; IN ORDER TO CHECK THE
21 005054 001432 BEQ ST133 ; CORRECT DATA FILL-UP
22 005056 DATERR VN,RQSPCL ;ERROR:REQ. SCPD NOT ALL 0'S
;***** ERROR 12 *****

005056 032777 040000 022550 BIT #B14,@SR
005064 001007 BNE .+20
005066 012700 024414 MOV #RQSPCL,R0
005072 012767 000012 011510 MOV #12,ERRNUM
005100 004767 011422 JSR PC,DERR
000013 = N+1
23 005104 032777 040000 022522 N BIT #B14,@SR ;PRINT ALLOWED?
24 005112 001010 BNE ST132S ;NO, SKIP IT
25 005114 PNTM SCPAD ;PRINT MESSAGE
005114 012700 022776 MOV #SCPAD,R0 ;POINTED TO BY SCPAD
005120 004767 012146 JSR PC,TYP0UT ;PRINT SCPD ADR OF ERROR
26 005124 017700 022500 MOV @SPAR,R0
27 005130 004767 013162 JSR PC,OC1PNT
28 005134 ST132S: SCOPE ST13
005134 004567 011050 JSR R5,SCPRTN
005140 004744 ST13
29 005142 027727 022462 000377 ST133: CMP @SPAR,#377 ;DONE CHECKING REQ. SCPD?
30 005150 001404 BEQ ST134 ;YES, CHECK RES. SCPD
31 005152 005277 022452 INC @SPAR ;NO, FINISH UP
32 005156 000167 177656 JMP ST132
33 005162 012777 001000 022440 ST134: MOV #1000,@SPAR ;NOW CHECK RES. SCPD
34 005170 012767 000000 011414 MOV #0,GOOD ;THIS '#0' ALSO MUST BE CHANGED
35 005176 017767 022430 011410 ST135: MOV @SPDR,BAD
36 005204 026767 011402 011402 CMP GOOD,BAD ;RES SCPD ALL 0'S?
37 005212 001432 BEQ ST136
38 005214 DATERR VN,RSSPCL ;ERROR:RES. SCPD NOT ALL 0'S
;***** ERROR 13 *****

005214 032777 040000 022412 BIT #B14,@SR
005222 001007 BNE .+20
005224 012700 024444 MOV #RSSPCL,R0
005230 012767 000013 011352 MOV #13,ERRNUM
005236 004767 011264 JSR PC,DERR
000014 = N
39 005242 032777 040000 022364 N BIT #B14,@SR ;PRINT ALLOWED?
    
```

40	005250	001010			BNE	ST135S		:NO, SKIP IT
41	005252				PNTM	SCPAD		
	005252	012700	022776		MOV	#SCPAD,RO		:PRINT MESSAGE
	005256	004767	012010		JSR	PC, TYP0UT		:POINTED TO BY SCPAD
42	005262	017700	022342		MOV	@SPAR,RO		:PRINT SCPD ADDR OF ERROR
43	005266	004767	013024		JSR	PC, OCTPNT		
44	005272			ST135S:	SCOPE	ST13		
	005272	004567	010712		JSR	R5, SCPRTN		
	005276	004744			ST13			
45	005300	027727	022324	001377	ST136:	CMP	@SPAR,#1377	:DONE RES. SCPD?
46	005306	001403			BEQ	ST14		:YES, NEXT ROUTINE
47	005310	005277	022314		INC	@SPAR		
48	005314	000730			BR	ST135		

```

1      ;TEST 'CLR TOP BYTE'
2      ;
3      ; SETTING 'CLRTPB' SHOULD RESULT IN ALL BITS OF THE
4      ; HIGH BYTE OF THE CSR BEING CLEARED. (LOW BYTE ALREADY = 0)
5 005316 012777 000020 022242 ST14:  MOV    #B04,@CSR      ;SET 'CLRTPB'
6 005324 005067 011262          CLR    GOOD        ;ALL CSR SHOULD BE 0
7 005330 017767 022232 011256  MOV    @CSR,BAD
8 005336 026767 011250 011250  CMP    GOOD,BAD    ;ALL CLEAR?
9 005344 001416          BEQ    ST15
10 005346          DATERR  W,CLTBER    ;ERROR:CLR TPB FAILED
          ;***** ERROR 14 *****
          005346 032777 040000 022260  BIT    #B14,@SR
          005354 001007          BNE    .+20
          005356 012700 024474          MOV    #CLTBER,R0
          005362 012767 000014 011220  MOV    #14,ERRNUM
          005370 004767 011132          JSR    PC,DERR
          N      =      N+1
11 005374          SCOPE  ST14
          005374 004567 010610          JSR    R5,SCRPTN
          005400 005316          ST14

12
13      ;NOW H A N G AND CHECK CSR
14      ; 'HANG' BY USING THE CMR-11 'DIE' FUNCTION
15      ; THEN CHECK THAT THE CSR WAS RIGHT
16      ; FOR THE 'DIE' FUNC. THEN CHECK ALL R/W BITS OF THE CSR
17
18 005402 005077 022222          ST15:  CLR    @SPAR      ;ADDR REQ. SCPD
19 005406 012777 000213 022216  MOV    #213,@SPDR  ;LOAD MAINT FUNC SUBF #3
20 005414 005277 022210          INC    @SPAR
21 005420 012777 000001 022204  MOV    #1,@SPDR    ;DATA LEN. = 1
22 005426 005277 022176          INC    @SPAR
23 005432 012777 000001 022172  MOV    #1,@SPDR    ;ONE REQUEST
24 005440 012777 000200 022120  MOV    #200,@CSR   ;SET 'MAIN' SHD CAUSE U.P. HANG
25 005446 012767 007777 022110  MOV    #7777,CNTR  ;WAIT FOR THE MICRO
26 005454 005777 022106          1$:   TST    @CSR      ;MICRO DONE?
27 005460 100403          BMI    Z$        ;NO, WAIT A BIT
28 005462 005367 022076          DEC    CNTR
29 005466 001372          BNE    1$
30 005470 012767 137600 011114  2$:   MOV    #137600,GOOD
31 005476 017767 022064 011110  MOV    @CSR,BAD    ;READ CSR REGISTER
32 005504 026767 011102 011102  CMP    GOOD,BAD    ;DID IT HANG CORRECTLY?
33 005512 001432          BEQ    ST16        ;YES, GOOD
34 005514 012767 000000 022042  MOV    #0,CNTR
35 005522          DATERR  W,DIERR    ;ERROR:CSR NOT RIGHT FOR 'DIE'
          ;***** ERROR 15 *****
          005522 032777 040000 022104  BIT    #B14,@SR
          005530 001007          BNE    .+20
          005532 012700 024536          MOV    #DIERR,R0
          005536 012767 000015 011044  MOV    #15,ERRNUM
          005544 004767 010756          JSR    PC,DERR
          N      =      N+1
36 005550 012777 000002 022010  ST150: MOV    #2,@CSR    ;PROGRAM INIT
37 005556 005367 022002          DEC    CNTR      ;WAIT FOR INIT
38 005562 001375          BNE    ST150
39 005564 012777 000020 021774  MOV    #B04,@CSR  ;CLR TOP BYTE
40 005572          SCOPE  ST15
          005572 004567 010412          JSR    R5,SCRPTN
    
```


CZCMJBO CMR(V)-11 HOST EXERCISE MACRO V03.02B10-DEC-80 10:41:54 PAGE 11-1
HOST LOGIC TEST

005576 005402

ST15

SEQ 0035

```

1          ;CHECK CSR WITH MICRO HUNG
2
3 005600 012767 137755 011004 ST16:  MOV    #137755,GOOD          ;CHECK R/W BITS OF CSR
4 005606 012777 000355 021752      MOV    #355,@CSR
5 005614 017767 021746 010772      MOV    @CSR,BAD
6 005622 026767 010764 010764      CMP    GOOD,BAD          ;DO THEY SET?
7 005630 001416                BEQ    ST160             ;YES, TRY CLEARING THEM
8 005632                DATERR  \N,CSRWE          ;ERROR:BAD BITS IN CSR WHILE HUNG
                                ;***** ERROR 16 *****

    005632 032777 040000 021774      BIT    #B14,@SR
    005640 001007                BNE    .+20
    005642 012700 024571                MOV    #CSRWE,R0
    005646 012767 000016 010734      MOV    #16,ERRNUM
    005654 004767 010646                JSR    PC,DERR
                                =
                                N+1
9 005660                SCOPE  ST16
    005660 004567 010324                JSR    R5,SCPRTN
    005664 005600                ST16

10 005666 012767 137400 010716 ST160: MOV    #137400,GOOD          ;NOW, SEE IF THEY CLEAR
11 005674 012777 000000 021664      MOV    #0,@CSR
12 005702 017767 021660 010704      MOV    @CSR,BAD
13 005710 026767 010676 010676      CMP    GOOD,BAD          ;DID THEY CLR?
14 005716 001416                BEQ    ST161             ;YES, NEXT
15 005720                DATERR  \N,CSRCE          ;ERROR:CAN'T CLR LOW CSR WHILE HUNG
                                ;***** ERROR 17 *****

    005720 032777 040000 021706      BIT    #B14,@SR
    005726 001007                BNE    .+20
    005730 012700 024623                MOV    #CSRCE,R0
    005734 012767 000017 010646      MOV    #17,ERRNUM
    005742 004767 010560                JSR    PC,DERR
                                =
                                N+1
16 005746                SCOPE  ST160
    005746 004567 010236                JSR    R5,SCPRTN
    005752 005666                ST160

17 005754 012777 000200 021604 ST161: MOV    #200,@CSR          ;CHECK ACCESS KEYS TO SCPD
18 005762 012777 000200 021640      MOV    #200,@SPAR        ;WITH MAIN SET
19 005770 017767 021636 021654      MOV    @SPDR,WORD        ;READ AN REQ SCPD WORD
20 005776 005167 021650                COM    WORD               ;COMPLEMENT IT
21 006002 042767 177400 021642      BIC    #177400,WORD      ;FORGET TOP BYTE
22 006010 016777 021636 021614      MOV    WORD,@SPDR        ;WRITE COMPL. BACK
23 006016 027767 021610 021626      CMP    @SPDR,WORD        ;READ IT BACK AGAIN
24 006024 001016                BNE    ST162             ;IT SHOULDN'T BE THE SAME
25 006026                ERROR   \N,SPAVME          ;ERROR:ACCESSED REQ SCPD WITH 'MAIN' SET
                                ;***** ERROR 20 *****

    006026 032777 040000 021600      BIT    #B14,@SR
    006034 001007                BNE    .+20
    006036 012700 024662                MOV    #SPAVME,R0
    006042 012767 000020 010540      MOV    #20,ERRNUM
    006050 004767 010366                JSR    PC,ERR
                                =
                                N+1
26 006054                SCOPE  ST161
    006054 004567 010130                JSR    R5,SCPRTN
    006060 005754                ST161

27 006062 012777 000001 021476 ST162: MOV    #1,@CSR          ;TRY AGAIN WITH 'ST FUN' SET
28 006070 012777 000200 021532      MOV    #200,@SPAR
29 006076 017767 021530 021546      MOV    @SPDR,WORD
30 006104 005167 021542                COM    WORD               ;COMPLEMENT IT
    
```

31	006110	042767	177400	021534	BIC	#177400,WORD	:FORGET TOP BYTE
32	006116	016777	021530	021506	MOV	WORD,@SPDR	:WRITE IT BACK
33	006124	026777	021522	021500	CMP	WORD,@SPDR	:READ IT BACK AGAIN
34	006132	001016			BNE	ST17	:IT SHOULDN'T BE THE SAME
35	006134				ERROR	\N,SPAVFE	:ERROR:ACCESSED REQ SCPD WITH 'STFUN' SET
	006134	032777	040000	021472	BIT	#B14,@SR	:***** ERROR 21 *****
	006142	001007			BNE	.+20	
	006144	012700	024726		MOV	#SPAVFE,R0	
	006150	012767	000021	010432	MOV	#21,ERRNUM	
	006156	004767	010260		JSR	PC,ERR	
		000022			=	N+1	
36	006162				SCOPE	ST162	
	006162	004567	010022		JSR	R5,SCPRTN	
	006166	006062			ST162		

N

```

1          ;TEST INTERRUPT WITH MICRO HUNG
2
3 006170 012777 006302 021456 ST17:  MOV    #ERINT,@VECTOR      ;SET UP VECTOR FOR ERROR
4 006176 012746 000000          MTPS    #0                ;ALLOW INTERRUPTS
5 006202 012746 006210          MOV    #0,-(SP)
6 006206 000002          MOV    #64$,-(SP)
7 006210          RTI
8          64$:
9 006210 000240          NOP
10 006212 000240          NOP
11 006214 012777 006344 021432 ST170: MOV    #GDINT,@VECTOR
12 006222 012777 000100 021336 MOV    #100,@CSR
13 006230 000240          NOP
14 006232 000240          NOP
15 006234          ERROR    W,NOINTR      ;ERROR:HOST CAN'T INTERRUPT UNIBUS
16          ;***** ERROR 22 *****
17 006234 032777 040000 021372 BIT    #B14,@SR
18 006242 001007          BNE    .+20
19 006244 012700 024773          MOV    #NOINTR,R0
20 006250 012767 000022 010332 MOV    #22,ERRNUM
21 006256 004767 010160          JSR    PC,ERR
22          =
23          N
24 006262 042777 000100 021276 BIC    #100,@CSR      ;CLEAR INTR ENAB
25 006270          SCOPE  ST170
26 006270 004567 007714          JSR    R5,SCPRTN
27 006274 006214          ST170
28 006276 000167 000054 ST170R: JMP    INTCNT
29 006302 022626 ERINT:  CMP    (SP)+,(SP)+
30 006304          ERROR    W,ERINTR      ;ERROR:ERRONEOUS HOST INTERRUPT
31          ;***** ERROR 23 *****
32 006304 032777 040000 021322 BIT    #B14,@SR
33 006312 001007          BNE    .+20
34 006314 012700 025031          MOV    #ERINTR,R0
35 006320 012767 000023 010262 MOV    #23,ERRNUM
36 006326 004767 010110          JSR    PC,ERR
37          =
38          N
39 006332          SCOPE  ST17
40 006332 004567 007652          JSR    R5,SCPRTN
41 006336 006170          ST17
42 006340 000167 177650          JMP    ST170
43 006344 012716 006276 GDINT:  MOV    #ST170R,(SP)
44 006350 005077 021212          CLR    @CSR
45 006354 000002          RTI
46 006356 012777 006422 021270 INTCNT: MOV    #ERINT1,@VECTOR
47 006364 042777 000100 021174 BIC    #100,@CSR
48 006372          MTPS    PRIO
49 006372 016746 021216          MOV    PRIO,-(SP)
50 006376 012746 006404          MOV    #64$,-(SP)
51 006402 000002          RTI
52          64$:
53 006404 012777 000100 021154 MOV    #100,@CSR
54 006412 000240          NOP
55 006414 000240          NOP
56 006416 000167 000046 INTCNR: JMP    ST18
57 006422 022626 ERINT1: CMP    (SP)+,(SP)+
58 006424 005077 021136          CLR    @CSR
59 006430          ERROR    W,WPODI      ;ERROR:WRONG PRIORITY;DEVICE IS HIGHER

```

```
006430 032777 040000 021' '5      BIT      #B14,BSR
006436 001007      BNE      .+20
006440 012700 025064      MOV      #WPODI,R0
006444 012767 000024 010136      MOV      #24,ERRNUM
006452 004767 007764      JSR      PC,ERR
          000025      =      N+1
32 006456      SCOPE  INTCNT
    006456 004567 007526      JSR      R5,SCRPTN
    006462 006356      INTCNT
33 006464 000167 000000      JMP      ST18
```

;***** ERROR 24 *****

```

1          ;NOW CHECK PROGRAM INIT.
2
3
4 006470 012777 000002 021070 ST18:  MOV    #2,@CSR           ;DO HOST 'INIT'
5 006476 016767 177774 177772 MOV:   MOV    MOV,MOV        ;WAIT A LITTLE
6 006504 016767 177766 177764 MOV    MOV,MOV
7 006512 016767 177760 177756 MOV    MOV,MOV
8 006520 016767 177752 177750 MOV    MOV,MOV
9 006528 016767 177744 177742 MOV    MOV,MOV
10 006534 016767 177736 177734 MOV    MOV,MOV
11 006542 012767 002000 010042 MOV    #2000,GOOD      ;ONLY PWR-UP SHOULD BE SET
12 006550 017767 021012 010036 MOV    @CSR,BAD        ;READ CSR
13 006556 026767 010030 010030 CMP    GOOD,BAD        ;CSR CORRECT AFTER INIT?
14 006564 001416                                BEQ    ST180           ;YES, CONTINUE
15 006566                                DATERR VN,HITER      ;ERROR:PROGRAM INIT FAILED
                                           ;***** ERROR 25 *****

006566 032777 040000 021040 BIT    #B14,@SR
006574 001007 BNE    .+20
006576 012700 025125 MOV    #HI,ER,RO
006602 012767 000025 010000 MOV    #25,ERRNUM
006610 004767 007712 JSR    PC,ERR
                                =
                                N+1
16 006614                                SCOPE ST18
006614 004567 007370 JSR    R5,SCRPTN
006620 006470 ST18
17 006622 032777 100000 020736 ST180: BIT    #B15,@CSR           ;WAIT FOR PWR-UP COMPL
18 006630 001774 BEQ    ST180
19 006632 012777 001000 020770 MOV    #1000,@SPAR     ;NOW READY TO READ VERSION
20 006640 012701 027464 MOV    #VERS@,R1       ;POINT AT ORIGINAL VERSION
21 006644 012702 027506 MOV    #VERSUN,R2      ;AND SAVE THIS VERSION
22 006650 117712 020756 ST181: MOVB   @SPDR,(R2)
23 006654 001016 BNE    ST182
24 006656 ERROR  VN,VISF      ;ERROR:MICRO DIDN'T STORE VERSION STUFF
                                           ;***** ERROR 26 *****

006656 032777 040000 020750 BIT    #B14,@SR
006664 001007 BNE    .+20
006666 012700 025165 MOV    #VISF,RO
006672 012767 000026 007710 MOV    #26,ERRNUM
006700 004767 007536 JSR    PC,ERR
                                =
                                N+1
25 006704                                SCOPE ST180
006704 004567 007300 JSR    R5,SCRPTN
006710 006622 ST180
26 006712 111167 007674 ST182: MOVB   (R1),GOOD      ;NOW COMPARE VERSION WITH ORIG.
27 006716 111267 007672 MOVB   (R2),BAD
28 006722 042767 177400 007662 BIC    #177400,GOOD    ;CLR TOP BYTE
29 006730 042767 177400 007656 BIC    #177400,BAD
30 006736 026767 007650 007650 CMP    GOOD,BAD        ;SAME AS FIRST TIME?
31 006744 001416 BEQ    ST183
32 006746 DATERR VN,VIDER  ;ERROR:2ND VERSION NOT SAME AS 1ST
                                           ;***** ERROR 27 *****

006746 032777 040000 020660 BIT    #B14,@SR
006754 001007 BNE    .+20
006756 012700 025225 MOV    #VIDER,RO
006762 012767 000027 007620 MOV    #27,ERRNUM
006770 004767 007532 JSR    PC,ERR
                                =
                                N+1

```



```

33 006774          SCOPE  ST182
    006774 004567 007210  JSR   R5,SCPRTN
    007000 006712          ST182
34 007002 122122          ST183: CMPB  (R1)+,(R2)+      ;UPDATE POINTERS
35 007004 027727 020620 001020  CMP   @SPAR,#1020      ;DONE?
36 007012 001403          BEQ   ST19
37 007014 005277 020610  INC   @SPAR            ;NEXT WORD
38 007020 000713          BR    ST181
39
40
41                ;CHECK ACCESS TO RESULT SCPD WITH INTERRUPT CLEAR
42
43
44 007022 012777 000020 020536  ST19:  MOV   #20,@CSR      ;CLEAR TOP BYTE OF CSR
45 007030 012777 001200 020572  MOV   #1200,@SPAR     ;ADDRESS RESULT SCPD
46 007036 017767 020570 020606  MOV   @SPDR,WORD      ;READ A WORD FROM IT
47 007044 005167 020602          COM   WORD             ;COMPLEMENT IT
48 007050 042767 177400 020574  BIC   #177400,WORD     ;IGNORE TOP 8 BITS
49 007056 016777 020570 020546  MOV   WORD,@SPDR      ;AND WRITE IT BACK
50 007064 026777 020562 020540  CMP   WORD,@SPDR      ;NOW READ IT BACK
51 007072 001016          BNE   ST20             ;IT SHOULDN'T BE THE SAME
52 007074          ERROR  \N,SPAVIC      ;ERROR:RES. SCR. PAD ACCESSED WITH INTR CLR
                                ;***** ERROR 30 *****

    007074 032777 040000 020532  BIT   #B14,@SR
    007102 001007          BNE   .+20
    007104 012700 025274          MOV   #SPAVIC,R0
    007110 012767 000030 007472  MOV   #30,ERRNUM
    007116 004767 007320          JSR   PC,ERR
                                =
                                N+1
53 007122          SCOPE  ST19
    007122 004567 007062  JSR   R5,SCPRTN
    007126 007022          ST19
54

```

```

1
2
3
4
5
6
7
8 007130 012767 000000 020426 ST20: MOV #0,CNTR
9 007136 012777 000020 020422 MOV #20,@CSR ;CLEAR TOP BYTE
10 007144 005077 020460 CLR @SPAR ;ADDRESS REQ SCPD
11 007150 012777 000210 020454 MOV #210,@SPDR ;LOAD 'MAIN' FUNCTION
12 007156 005277 020446 INC @SPAR ; TO EXAMINE HOST
13 007162 012777 000004 020442 MOV #4,@SPDR ; MEMORY LOCATIONS
14 007170 005277 020434 INC @SPAR
15 007174 012777 000001 020430 MOV #1,@SPDR ;ONE REQUEST
16 007202 005277 020422 INC @SPAR
17 007206 012777 000002 020416 MOV #2,@SPDR ;# OF BYTES TO READ = 2
18 007214 005277 020410 INC @SPAR
19 007220 012777 000377 020404 MOV #377,@SPDR ;START ADDR = DFFF (H)
20 007226 005277 020376 INC @SPAR
21 007232 012777 000337 020372 MOV #337,@SPDR
22 007240 012777 000001 020320 ST200: MOV #1,@CSR ;SET 'ST FUN' ON 'MAINT' FUNC
23 007246 012767 140000 007336 MOV #B14+B15,GOOD ;CHECK FOR FUNCOM & INTR
24 007254 012747 020306 007332 ST201: MOV @CSR,BAD
25 007262 026767 007324 007324 CMP GOOD,BAD ;'FUNCOM' AND 'INTR' SET?
26 007270 001421 BEQ ST202 ;IF NOT, WAIT A BIT
27 007272 005367 020266 DEC CNTR
28 007276 001366 BNE ST201
29 007300 DATERR W,STESTF ;ERROR:STATUS WRONG WITH 'STFUN' ON 'MAIN'
;***** ERROR 31 *****

007300 032777 040000 020326 BIT #B14,@SR
007306 001007 BNE .+20
007310 012700 025337 MOV #STESTF,R0
007314 012767 000031 007266 MOV #31,ERRNUM
007322 004767 007200 JSR PC,DEERR
000032 = N+1
N SCOPE ST20
JSR R5,SCPRTN
ST20

30 007326 004567 006656 ST202: MOV #1000,@SPAR ;ADDRESS RESULT SCPD
31 007334 012777 001000 020266 MOV #210,GOOD ;CHECK ON FUNCTION
32 007342 012767 000210 007242 MOV @SPDR,BAD
33 007350 017767 020256 007236 CMP GOOD,BAD ;FUNCTION CORRECT?
34 007356 026767 007230 007230 BEQ ST203
35 007364 001416 DATERR W,RSPEF ;ERROR:RESULT FUNC. CODE WRONG ON MAIN
36 007366 ;***** ERROR 32 *****

007366 032777 040000 020240 BIT #B14,@SR
007374 001007 BNE .+20
007376 012700 025370 MOV #RSPEF,R0
007402 012767 000032 007200 MOV #32,ERRNUM
007410 004767 007112 JSR PC,DEERR
000033 = N+1
N SCOPE ST20
JSR R5,SCPRTN
ST20

37 007414 004567 006570 ST203: MOV #1002,@SPAR ;NOW ADDRESS 'USDF'
38 007422 012777 001002 020200 MOV #377,GOOD ;SHOULD SHOW 'TOTAL FAILURE' CODE
39 007430 012767 000377 007154

```

40	007436	017767	020170	007150	MOV	@SPDR,BAD		
41	007444	026767	007142	007142	CMP	GOOD,BAD		:USOF O.K.?
42	007452	001416			BEQ	ST204		
43	007454				DATERR	W,USOF E1		:ERROR:'STFUN' ON 'MAIN' GAVE BAD USOF
								:***** ERROR 33 *****
	007454	032777	040000	020152	BIT	#B14,@SR		
	007462	001007			BNE	.+20		
	007464	012700	025431		MOV	#USOF E1,R0		
	007470	012767	000033	007112	MOV	#33,ERRNUM		
	007476	004767	007024		JSR	PC,DERR		
		000034			=	N+1		
44	007502				N	SCOPE	ST20	
	007502	004567	006502		JSR	R5,SCRPTN		
	007506	007130			ST20			
45	007510	012767	000000	020046	ST204:	MOV	#0,CNTR	
46	007516	012777	000020	020042	MOV	#20,@CSR		:SET 'CLRTPB'
47	007524	012777	000200	020034	MOV	#200,@CSR		:SET 'MAIN' WITH SAME FUN LOADED
48	007532	012767	140000	007052	MOV	#B14+B15,GOOD		:CHECK FOR 'FUNCOM' & 'INTR'
49	007540	017767	020022	007046	ST205:	MOV	@CSR,BAD	
50	007546	026767	007040	007040	CMP	GOOD,BAD		:FUN COMPLETE?
51	007554	001421			BEQ	ST206		
52	007556	005367	020002		DEC	CNTR		:WAIT A BIT
53	007562	001366			BNE	ST205		
54	007564				DATERR	W,STEMF		:ERROR:STATUS WRONG ON 'MAIN' FUNC
								:***** ERROR 34 *****
	007564	032777	040000	020042	BIT	#B14,@SR		
	007572	001007			BNE	.+20		
	007574	012700	025472		MOV	#STEMF,R0		
	007600	012767	000034	007002	MOV	#34,ERRNUM		
	007606	004767	006714		JSR	PC,DERR		
		000035			=	N+1		
55	007612				N	SCOPE	ST204	
	007612	004567	006372		JSR	R5,SCRPTN		
	007616	007510			ST204			
56	007620	012777	WJ1002	020002	ST206:	MOV	#1002,@SPAR	:POINT AT 'USOF'
57	007626	012767	000001	006756	MOV	#1,GOOD		:SHOULD SHOW SUCCESS
58	007634	017767	017772	006752	MOV	@SPDR,BAD		
59	007642	026767	006744	006744	CMP	GOOD,BAD		:USOF O.K.?
60	007650	001416			BEQ	ST207		
61	007652				DATERR	W,USOF E2		:ERROR:BAD 'USOF' ON 'MAIN' FUNCTION
								:***** ERROR 35 *****
	007652	032777	040000	017754	BIT	#B14,@SR		
	007660	001007			BNE	.+20		
	007662	012700	025523		MOV	#USOF E2,R0		
	007666	012767	000035	006714	MOV	#35,ERRNUM		
	007674	004767	006626		JSR	PC,DERR		
		000036			=	N+1		
62	007700				N	SCOPE	ST204	
	007700	004567	006304		JSR	R5,SCRPTN		
	007704	007510			ST204			
63	007706	012777	001001	017714	ST207:	MOV	#1001,@SPAR	:POINT AT DATA LENGTH
64	007714	012767	000003	006670	MOV	#3,GOOD		:SHOULD BE 3
65	007722	017767	017704	006664	MOV	@SPDR,BAD		
66	007730	026767	006656	006656	CMP	GOOD,BAD		:WAS IT 3?
67	007736	001416			BEQ	ST21		
68	007740				DATERR	W,DATLER		:ERROR:'MAIN' FUNC GAVE BAD DATA LEN.
								:***** ERROR 36 *****

	007740	032777	040000	017666		BIT	#B14,@SR	
	007746	001007				BNE	+.20	
	007750	012700	025560			MOV	#DATLER,R0	
	007754	012767	000036	006626		MOV	#36,ERRNUM	
	007762	004767	006540			JSR	PC,DERR	
		000037			N	=	N+1	
69	007766					SCOPE	ST204	
	007766	004567	006216			JSR	R5,SCPRTN	
	007772	007510				ST204		
70	007774	012700	027556		ST21:	MOV	#BAUD,R0	:POINT AT BAUD TABLE
71	010000	012777	001003	017622		MOV	#1003,@SPAR	:GET RESULTS OF EXAMINE
72	010006	117720	017620			MOVB	@SPDR,(R0)+	
73	010012	005277	017612			INC	@SPAR	
74	010016	117710	017610			MOVB	@SPDR,(R0)	:STORE BAUD RATES

```

1          ;CHECK RESULTS OF STARTING A REGULAR FUNCTION WITH 'MAIN'
2          ; TRY DOING AN 'ACTIVATE REMOTE' FIRST USING 'MAIN', THEN 'STFUN'
3
4
5 010022 012777 000020 017536 ST22:  MOV    #20,@CSR           ;CLEAR TOP BYTE OF CSR
6 010030 012777 000000 017572      MOV    #0,@SPAR
7 010036 012777 000060 017566      MOV    #60,@SPDR           ;LOAD 'ACTIVATE REMOTE' FUNC
8 010044 005277 017560      INC    @SPAR
9 010050 012777 000002 017554      MOV    #2,@SPDR           ;DATA LEN. = 2
10 010056 005277 017546      INC    @SPAR
11 010062 012777 000001 017542      MOV    #1,@SPDR           ;ONE REQUEST
12 010070 005277 017534      INC    @SPAR
13 010074 012777 000001 017530      MOV    #1,@SPDR           ;REMOTE #1
14 010102 012777 000200 017456 ST220: MOV    #200,@CSR         ;SET 'MAIN'
15 010110 032777 040000 017450 ST221: BIT    #B14,@CSR         ;FUNC. COMPL?
16 010116 001774      BEQ    ST221              ;NO, WAIT
17 010120 012777 001000 017502      MOV    #1000,@SPAR        ;ADDRESS RESULT
18 010126 017767 017500 006460      MOV    @SPDR,BAD          ;READ RESULT FUNC. CODE
19 010134 012767 000060 006450      MOV    #60,GOOD
20 010142 026767 006444 006444      CMP    GOOD,BAD           ;O.K.?
21 010150 001416      BEQ    ST222
22 010152      DATERR  VN,RSPEMF        ;ERROR:RES. FUNC. CODE WRONG FOR 'MAIN'
                                ;***** ERROR 37 *****

    010152 032777 040000 017454      BIT    #B14,@SR
    010160 001007      BNE    .+20
    010162 012700 025620      MOV    #RSPEMF,R0
    010166 012767 000037 006414      MOV    #37,ERRNUM
    010174 004767 006326      JSR    PC,DERR
                                =
                                N
                                SCOPE ST22
    23 010200      JSR    R5,SCRPTN
    010200 004567 006004      ST22
    010204 010022      ST22
    24 010206 012777 001002 017414 ST222: MOV    #1002,@SPAR        ;ADDR 'USOF'
    25 010214 012767 000377 006370      MOV    #377,GOOD         ;SHOULD SHOW TOTAL FAILURE
    26 010222 017767 017404 006364      MOV    @SPDR,BAD
    27 010230 026767 006356 006356      CMP    GOOD,BAD         ;USOF O.K.?
    28 010236 001416      BEQ    ST223
    29 010240      DATERR  VN,USOF E3        ;ERROR:'MAIN' ON REG FUNC GAVE WRONG USOF
                                ;***** ERROR 40 *****

    010240 032777 040000 017366      BIT    #B14,@SR
    010246 001007      BNE    .+20
    010250 012700 025662      MOV    #USOF E3,R0
    010254 012767 000040 006326      MOV    #40,ERRNUM
    010262 004767 006240      JSR    PC,DERR
                                =
                                N
                                SCOPE ST22
    30 010266      JSR    R5,SCRPTN
    010266 004567 005716      ST22
    010272 010022      ST22
    31 010274 012777 000020 017264 ST223: MOV    #20,@CSR           ;CLR TOP BYTE
    32 010302 012777 000001 017256      MOV    #1,@CSR           ;SET 'STFUN'
    33 010310 032777 040000 017250 ST224: BIT    #B14,@CSR         ;FUNCOM SET?
    34 010316 001774      BEQ    ST224              ;WAIT
    35 010320 012777 001002 017302      MOV    #1002,@SPAR        ;ADDRESS USOF
    36 010326 012767 000364 006256      MOV    #364,GOOD         ;SHOULD SHOW REM. ERROR
    37 010334 017767 017272 006252      MOV    @SPDR,BAD
    38 010342 026767 006244 006244      CMP    GOOD,BAD         ;USOF O.K.?
    39 010350 001416      BEQ    ST23

```

40 010352

010352 032777 040000 017254
010360 001007
010362 012700 025725
010366 012767 000041 006214
010374 004767 006126
000042

N

41 010400

010400 004567 0504
010404 010274

DATERR W,USOFE4
BIT #B14,BSR
BNE .+20
MOV #USOFE4,R0
MOV #41,ERRNUM
JSR PC,DEERR
=
SCOPE ST223
JSR R5,SCRPTN
ST223

;ERROR:WRONG USOF ON REGULAR FUNCTION
;***** ERROR 41 *****

```

1                                     ;END OF LOGIC TEST. PRINT END OF PASS AND VERSION STUFF
2
3
4 010406 005767 017236          ST23:  TST      VERFLG      ;IS FLAG SET?
5 010412 001402                BEQ      ST23A      ;NO
6 010414 000167 000350          JMP      ST24      ;YES, JUST PRINT 'END PASS'
7 010420 012767 177777 017222 ST23A: MOV      #-1,VERFLG ;NO, SET FLAG
8 010426 012701 027506          MOV      #VERSION,R1 ;GET VERSION DATA
9 010432 012702 027556          MOV      #BAUD,R2   ;AND BAUD RATES
10 010436 012700 023017          PNTM    VERSM      ;PRINT 'VERSION '
    010436 012700 023017          MOV      #VERSM,R0  ;PRINT MESSAGE
    010442 004767 006624          JSR     PC, TYP0UT ;POINTED TO BY VERSM
11 010446 012767 000021 017110 ST230: MOV     #21,CNTR ;DUMP ALL VERSION INFO
12 010454 112100                MOV     (R1)+,R0
13 010456 004767 010140          JSR     PC, T0
14 010462 005367 017076          DEC     CNTR
15 010466 001372                BNE     ST230
16 010470 012700 023052          PNTM    BAUDM      ;PRINT 'BAUD RATES:'
    010470 012700 023052          MOV     #BAUDM,R0  ;PRINT MESSAGE
    010474 004767 006572          JSR     PC, TYP0UT ;POINTED TO BY BAUDM
17 010500 012700 023070          PNTM    PORT0M     ;PRINT 'PORT 0 = '
    010500 012700 023070          MOV     #PORT0M,R0 ;PRINT MESSAGE
    010504 004767 006562          JSR     PC, TYP0UT ;POINTED TO BY PORT0M
18 010510 111267 017136          MOV     (R2),WORD  ;SORT OUT CORRECT PART
19 010514 042767 177617 017130 BIC     #177617,WORD
20 010522 006267 017124          ASR     WORD
21 010526 006267 017120          ASR     WORD
22 010532 006267 017114          ASR     WORD ;PUT IT IN L.S. BIT PLACES
23 010536 006267 017110          ASR     WORD
24 010542 012703 026302          MOV     #BDRTBL,R3 ;R3 POINTS AT TABLE OF RATES
25 010546 006367 017100          ASL     WORD ;DOUBLE OFFSET
26 010552 066703 017074          ADD     WORD,R3   ;USE CORRECT OFFSET
27 010556 011300                MOV     (R3),R0   ;GET BAUD
28 010560 004767 006506          JSR     PC, TYP0UT
29 010564 012700 023104          PNTM    PORT1M     ;NOW PRINT 'PORT 1 = '
    010564 012700 023104          MOV     #PORT1M,R0 ;PRINT MESSAGE
    010570 004767 006476          JSR     PC, TYP0UT ;POINTED TO BY PORT1M
30 010574 112267 017052          MOV     (R2)+,WORD ;GET NEXT BAUD RATE CODE
31 010600 042767 177770 017044 BIC     #177770,WORD ;SORT OUT CORRECT BAUD RATE
32 010606 012703 017030          MOV     #BDRTBL,R3
33 010612 006367 017034          ASL     WORD ;DOUBLE OFFSET
34 010616 066703 017030          ADD     WORD,R3   ;USE IT AS TABLE OFFSET
35 010622 011300                MOV     (R3),R0
36 010624 004767 006442          JSR     PC, TYP0UT ;PRINT RATE FOR PORT 1
37 010630 012700 023120          PNTM    PORT2M     ;PRINT 'PORT 2 = '
    010630 012700 023120          MOV     #PORT2M,R0 ;PRINT MESSAGE
    010634 004767 006432          JSR     PC, TYP0UT ;POINTED TO BY PORT2M
38 010640 111267 017006          MOV     (R2),WORD  ;GET BAUD RATE CODE
39 010644 042767 177617 017000 BIC     #177617,WORD ;GET CORRECT PART.
40 010652 006267 016774          ASR     WORD
41 010656 006267 016770          ASR     WORD
42 010662 006267 016764          ASR     WORD
43 010666 006267 016760          ASR     WORD
44 010672 012703 026302          MOV     #BDRTBL,R3
45 010676 006367 016750          ASL     WORD ;DOUBLE OFFSET
46 010702 066703 016744          ADD     WORD,R3   ;USE IT AS TABLE OFFSET
47 010706 011300                MOV     (R3),R0
    
```

```

48 010710 004767 006356      JSR      PC,TYPOUT      ;PRINT BAUD RATE FOR PORT 2
49 010714      PNTM      PORT3M      ;PRINT 'PORT 3 = '
      010714 012700 023134      MOV      #PORT3M,R0      ;PRINT MESSAGE
      010720 004767 006346      JSR      PC,TYPOUT      ;POINTED TO BY PORT3M
50 010724 111267 016722      MOV      (R2),WORD
51 010730 042767 177770 016714      BIC      #177770,WORD      ;SORT OUT CORRECT PART
52 010736 012703 026302      MOV      #BDRtbl,R3
53 010742 006367 016704      ASL      WORD      ;DOUBLE OFFSET
54 010746 066703 016700      ADD      WORD,R3      ;USE IT AS TABLE OFFSET
55 010752 011300      MOV      (R3),R0
56 010754 004767 006312      JSR      PC,TYPOUT      ;PRINT RATE FOR PORT 3
57 010760 004767 007514      JSR      PC,NULLS      ;NULLS AFTER PRINTOUT
58 010764 000167 000016      JMP
59
60 010770 005367 016600      ST24:   DEC      ITER      ;UPDATE ITERATION
61 010774 001404      BEQ      ST25      ;DONE 10?
62 010776 032777 004000 016630      BIT      #B11,@SR      ;SINGLE ITERATION?
63 011004 001415      BEQ      ST26
64 011006 012767 000010 016560      ST25:   MOV      #10,ITER
65 011014 005267 016562      INC      PASCNT      ;UPDATE PASS COUNT
66 011020 032777 002000 016606      BIT      #B10,@SR      ;END PASS INHIBIT?
67 011026 001004      BNE      ST26
68 011030 004767 000014      JSR      PC,ENPRTN
69 011034 004767 007440      JSR      PC,NULLS
70 011040 004767 005622      ST26:   JSR      PC,MONIT
71 011044 000167 172132      JMP      ST0
72
73
74 011050      ENPRTN: PNTM      ENDPS      ;PRINT 'END PASS # '
      011050 012700 023150      MOV      #ENDPS,R0      ;PRINT MESSAGE
      011054 004767 006212      JSR      PC,TYPOUT      ;POINTED TO BY ENDPS
75 011060 016700 016516      MOV      PASCNT,R0      ;GET PASS COUNT
76 011064 004767 007332      JSR      PC,DECPNT      ;PRINT IT IN DECIMAL
77 011070      PNTM      ERCTM      ;PRINT 'ERROR COUNT = '
      011070 012700 023177      MOV      #ERCTM,R0      ;PRINT MESSAGE
      011074 004767 006172      JSR      PC,TYPOUT      ;POINTED TO BY ERCTM
78 011100 105767 016436      TSTB     OFLOF      ;CHECK FOR ERROR CNT OVERFLOW
79 011104 001405      BEQ      EN01      ;IT DID'T, PRINT COUNT
80 011106      PNTM      OVFLM      ;IT DID, PRINT '**OVERFLOW**'
      011106 012700 023215      MOV      #OVFLM,R0      ;PRINT MESSAGE
      011112 004767 006154      JSR      PC,TYPOUT      ;POINTED TO BY OVFLM
81 011116 000404      BR
82 011120 016700 016446      EN01:   MOV      ERRCNT,R0      ;GET ERROR COUNT
83 011124 004767 007272      JSR      PC,DECPNT      ;PRINT IT IN DECIMAL
84 011130 000207      ENRET:   RTS      PC
    
```



```

1
2
3
4
5
6
7 011132 012767 000000 016424 PORTLP: MOV #0,CNTR ;BETTER CHECK FOR PWR-UP COMPL
8 011140 027727 016422 102000 PRLC: CMP @CSR,#102000 ;POWER UP COMPLETE?
9 011146 001420 BEQ POK ;YES, CARRY ON
10 011150 005367 016410 DEC CNTR ;NO, WAIT A BIT
11 011154 001371 BNE PRLC
12 011156 ERROR VN,NOGO ;ERROR:HOST MICRO FAILED TO START
;***** ERROR 42 *****

011156 032777 040000 016450 BIT #B14,@SR
011164 001007 BNE .+20
011166 012700 024031 MOV #NOGO,R0
011172 012767 000042 005410 MOV #42,ERRNUM
011200 004767 005236 JSR PC,ERR
000043 = N+1

13 011204 HLT ;RUN LOGIC TEST.
011204 004767 004760 JSR PC,SUHLT
14 011210 012767 000010 016356 POK: MOV #10,ITER ;LOAD ITERATION COUNT
15 011216 005067 016332 CLR ALLFLG ;CLEAR 'ALL' FLAG
16 011222 005067 016354 CLR PASCNT ;CLEAR PASS COUNT
17 011226 005067 016340 CLR ERRCNT ;CLEAR ERROR COUNT
18 011232 105067 016304 CLRB OFLOF ;CLR OVERFLOW FLAG
19 011236 PNTM PLTM ;PRINT 'PORT LOOP-BACK TEST'
011236 012700 023232 MOV #PLTM,R0 ;PRINT MESSAGE
011242 004767 006024 JSR PC,TYP0UT ;POINTED TO BY PLTM
20 011246 PNTM PORTN ;PRINT 'PORT # (<CR>=ALL) '
011246 012700 023260 MOV #PORTN,R0 ;PRINT MESSAGE
011252 004767 006014 JSR PC,TYP0UT ;POINTED TO BY PORTN
21 011256 012767 000004 006754 MOV #4,KBBUF ;DEFAULT IS 'ALL'
22 011264 012767 011246 016314 MOV #PST0,PCONT ;LOAD RETURN ADDRESS
23 011272 004767 006036 JSR PC,INPKB ;GET KBD INPUT
24 011276 005067 016304 CLR PCONT
25 011302 026727 006732 000004 CMP KBBUF,#4 ;WAS IT DEFAULT (OR 4)?
26 011310 001007 BNE PST1 ;NO
27 011312 012767 177777 016234 MOV #-1,ALLFLG ;YES, SET 'ALL' FLAG
28 011320 005067 016266 CLR PORT ; START WITH PORT 0
29 011324 000167 000034 JMP PST2A

30
31 011330 101004 PST1: BHI PSTE ;WAS IT HIGHER THAN 4?
32 011332 005067 016216 CLR ALLFLG ;NO, MUST BE A PORT #
33 011336 000167 000014 JMP PST2
34 011342 PSTE: PNTM AGAIN ;WRONG!, TRY AGAIN
011342 012700 021324 MOV #AGAIN,R0 ;PRINT MESSAGE
011346 004767 005720 JSR PC,TYP0UT ;POINTED TO BY AGAIN
35 011352 000167 177670 JMP PST0
36 011356 016767 006656 016226 PST2: MOV KBBUF,PORT ;SAVE PORT #
37 011364 012777 000020 016174 PST2A: MOV #20,@CSR ;CLR TOP BYTE
38 011372 005077 016232 CLR @SPAR ;ADDRESS REQ. SCR. PAD
39 011376 012777 000212 016226 MOV #212,@SPDR ;LOAD MAINT FUNC.
40 011404 005277 016220 INC @SPAR ;TO DO PORT LOOP-BACK
41 011410 012777 000223 016214 MOV #23,@SPDR ;DATA LEN. = 23
42 011416 005277 016206 INC @SPAR
43 011422 012777 000001 016202 MOV #1,@SPDR ;ONE REQUEST
    
```

```

44 011430 005277 016174          INC      @SPAR
45 011434 016777 016152 016170  MOV      PORT,@SPDR      ;LOAD PORT # (XMTR)
46 011442 005277 016162          INC      @SPAR
47 011446 016777 016140 016156  MOV      PORT,@SPDR      ;LOAD PORT # (RCVR)
48 011454 012767 000020 016102  MOV      #20,CNTR        ;CHARACTER COUNTER
49 011462 012701 026424          MOV      #DATBL,R1      ;LOOP DATA FROM 'DATBL'
50 011466 005277 016136          PST3:   INC      @SPAR
51 011472 012177 016134          MOV      (R1)+,@SPDR    ;LOAD DATA BYTE
52 011476 005367 016062          DEC      CNTR           ;LOADED ALL?
53 011502 001371          BNE      PST3          ;NO, KEEP LOADING
54 011504          PST5:   PNTM     TYP CR          ;PRINT 'TYPE <CR> WHEN READY'
      011504 012700 023312          MOV      #TYP CR,R0     ;PRINT MESSAGE
      011510 004767 005556          JSR      PC,TYP OUT     ;POINTED TO BY TYP CR
55 011514 012767 011504 016064  PST4:   MOV      #PST5,PCONT   ;INDICATE PROMPT PENDING
56 011522 012767 001236 006510  MOV      #1236,KBBUF    ;LOAD ANY DEFAULT VALUE
57 011530 004767 005600          JSR      PC,INPKB       ;GET KEYBOARD INPUT
58 011534 005067 016046          CLR      PCONT
59 011540 026727 006474 001236  CMP      KBBUF,#1236    ;<CR> ONLY TYPED?
60 011546 001356          BNE      PST5          ;IF NOT, KEEP LOOKING FOR IT
61
62          ;START MAINTENANCE FUNCTION
63
64 011550          PST5:   MTPS     #P0           ;ALLOW LINE CLOCK INTERRUPTS
      011550 012746 000000          MOV      #P0,-(SP)
      011554 012746 011562          MOV      #64$,-(SP)
      011560 000002          RTI
      011562          64$:
65 011562 012777 000020 015776  MOV      #20,@CSR
66 011570 012777 000200 015770  MOV      #200,@CSR
67 011576 005777 015764          PST6:   TST      @CSR
68 011602 100403          BMI      PST6          ;LOOK FOR INTERRUPT BIT
69 011604 004767 005056          JSR      PC,MONIT
70 011610 000772          BR       PST6          ;AND WATCH FOR CNTRL CHARS.
71 011612 012777 001002 016010  PST60:  MOV      #1002,@SPAR
72 011620 027727 016006 000001  CMP      @SPDR,#1
73 011626 001004          BNE      PST7
74 011630 004767 005032          JSR      PC,MONIT
75 011634 000167 000430          JMP      ENDCHK
76 011640 027727 015766 000375  PST7:   CMP      @SPDR,#375
77 011646 103434          BLO      PST8
78 011650          ERROR   W,PLFFE          ;ERROR:TOTAL FAILURE OF 'MAIN' FUNCTION
      ;***** ERROR 43 *****
      011650 032777 040000 015756  BIT      #B14,@SR
      011656 001007          BNE      .+20
      011660 012700 025762          MOV      #PLFFE,R0
      011664 012767 000043 004716  MOV      #43,ERRNUM
      011672 004767 004544          JSR      PC,ERR
      ;
      000044          =
      N
79 011676 032777 040000 015730  BIT      #B14,@SR
80 011704 001010          BNE      PST7S
81 011706          PNTM     PORTMS
      011706 012700 023556          MOV      #PORTMS,R0
      011712 004767 005354          JSR      PC,TYP OUT
82 011716 016700 015670          MOV      PORT,R0
83 011722 004767 006370          JSR      PC,OC TPNT
84 011726          PST7S: SCOPE
      011726 004567 004256          JSR      R5,SCRPTN

```

85	011732	011550			PST5			
	011734	000167	000330		JMP	ENDCHK		
86	011740	027727	015666	000374	PST8:	CMP @SPDR,#374		;XMTR HUNG UP?
87	011746	103434			BLO	PST80		
88	011750				ERROR	W,PLTXTO		;ERROR:XMTR TIMEOUT ERROR
								;***** ERROR 44 *****
	011750	032777	040000	015656	BIT	#B14,@SR		
	011756	001007			BNE	.+20		
	011760	012700	026025		MOV	#PLTXTO,R0		
	011764	012767	000044	004616	MOV	#44,ERRNUM		
	011772	004767	004444		JSR	PC,ERR		
		000045			=	N+1		
89	011776	032777	040000	015630	N	BIT #B14,@SR		;PRINT ALLOWED?
90	012004	001010			BNE	PST8S		;NO, SKIP IT
91	012006				PNTM	PORTMS		;PRINT 'PORT # ''
	012006	012700	023556		MOV	#PORTMS,R0		;PRINT MESSAGE
	012012	004767	005254		JSR	PC,TYPOUT		;POINTED TO BY PORTMS
92	012016	016700	015570		MOV	PORT,R0		
93	012022	004767	006270		JSR	PC,OCTPNT		
94	012026				PST8S:	SCOPE		
	012026	004567	004156		JSR	R5,SCPRTN		
	012032	011550			PST5			
95	012034	000167	000230		JMP	ENDCHK		
96	012040	027727	015566	000373	PST80:	CMP @SPDR,#373		;RCVR HUNG UP?
97	012046	103434			BLO	PST9		
98	012050				ERROR	W,PLRCTO		;ERROR:RCVR TIMEOUT ERROR
								;***** ERROR 45 *****
	012050	032777	040000	015556	BIT	#B14,@SR		
	012056	001007			BNE	.+20		
	012060	012700	026053		MOV	#PLRCTO,R0		
	012064	012767	000045	004516	MOV	#45,ERRNUM		
	012072	004767	004344		JSR	PC,ERR		
		000046			=	N+1		
99	012076	032777	040000	015530	N	BIT #B14,@SR		;PRINT ALLOWED?
100	012104	001010			BNE	PST80S		;NO, SKIP IT
101	012106				PNTM	PORTMS		;PRINT 'PORT # ''
	012106	012700	023556		MOV	#PORTMS,R0		;PRINT MESSAGE
	012112	004767	005154		JSR	PC,TYPOUT		;POINTED TO BY PORTMS
102	012116	016700	015470		MOV	PORT,R0		
103	012122	004767	006170		JSR	PC,OCTPNT		
104	012126				PST80S:	SCOPE		
	012126	004567	004056		JSR	R5,SCPRTN		
	012132	011550			PST5			
105	012134	000167	000130		JMP	ENDCHK		
106	012140	012777	001001	015462	PST9:	MOV #1001,@SPAR		;ADDRESS DATA TO FIND BAD WORD
107	012146	067777	015460	015454	ADD	@SPDR,@SPAR		
108	012154	017767	015452	004432	MOV	@SPDR,BAD		;GET BAD WORD
109	012162	042777	001000	015440	BIC	#B09,@SPAR		
110	012170	062777	000002	015432	ADD	#2,@SPAR		
111	012176	017767	015430	004406	MOV	@SPDR,GOOD		;GET GOOD WORD
112	012204				DATERR	W,PLDERR		;ERROR:LOOP TEST DETECTED DATA ERROR
								;***** ERROR 46 *****
	012204	032777	040000	015422	BIT	#B14,@SR		
	012212	001007			BNE	.+20		
	012214	012700	026101		MOV	#PLDERR,R0		
	012220	012767	000046	004362	MOV	#46,ERRNUM		
	012226	004767	004274		JSR	PC,DERR		

113	012232	000047	040000	015374	N	=	N+1	
114	012240	032777				BIT	#B14,@SR	:PRINT ALLOWED?
115	012242	001010				BNE	PST9S	
	012242	012700	023556			PNTM	PORTMS	:PRINT 'PORT # ''
	012246	004767	005020			MOV	#PORTMS,R0	:PRINT MESSAGE
116	012252	016700	015334			JSR	PC,TYPOUT	:POINTED TO BY PORTMS
117	012256	004767	006034			MOV	PORT,R0	
118	012262				PST9S:	JSR	PC,OCTPNT	:PRINT #
	012262	004567	003722			SCOPE	PST5	
	012266	011550				JSR	R5,SCPRTN	
						PST5		
119								
120								
121	012270	052777	000020	015270	ENDCHK:	BIS	#B04,@CSR	:CLR TOP BYTE
122	012276	005767	015252			TST	ALLFLG	:DOING ALL PORTS?
123	012302	001441				BEQ	PNTEND	:NO, DONE A PASS
124	012304	026727	015302	000003		CMP	PORT,#3	:YES, DONE ALL?
125	012312	103416				BLO	END0	:NO, KEEP GOING
126	012314	005067	015272			CLR	PORT	:CLEAR PORT #
127	012320	012777	000003	015302		MOV	#3,@SPAR	: IN REQ. SCPD
128	012326	016777	015260	015276		MOV	PORT,@SPDR	
129	012334	005277	015270			INC	@SPAR	
130	012340	016777	015246	015264		MOV	PORT,@SPDR	
131	012346	000417				BR	PNTEND	
132	012350	005267	015236		END0:	INC	PORT	:NEXT PORT
133	012354	012777	000003	015246		MOV	#3,@SPAR	:PREPARE TO LOAD NEXT PORT #
134	012362	016777	015224	015242		MOV	PORT,@SPDR	
135	012370	005277	015234			INC	@SPAR	
136	012374	016777	015212	015230		MOV	PORT,@SPDR	
137	012402	000167	177142			JMP	PST5	
138								
139	012406	005367	015162		PNTEND:	DEC	ITER	:UPDATE ITERATION COUNT
140	012412	001404				BEQ	PNT0	
141	012414	032777	004000	015212		BIT	#B11,@SR	:SW 11 =, INH ITERATION
142	012422	001422				BEQ	PNTRTN	
143	012424	112767	000010	015142	PNT0:	MOV	#10,ITER	:RELOAD ITERATION COUNT
144	012432	005267	015144		PNT1:	INC	PASCNT	:UPDATE PASS COUNT
145	012436	032777	002000	015170		BIT	#B10,@SR	:SW 10 = INH END PASS PRINT
146	012444	001011				BNE	PNTRTN	
147	012446	004767	176376			JSR	PC,ENPRTN	:PRINT END PASS
148	012452	105767	015052			TSTB	CLKFL1	:IS CLOCK TICKING?
149	012456	001402				BEQ	1\$:NO, SKIP PRINTOUT
150	012460	004767	004504			JSR	PC,PRNTIM	:PRINT ELAPSED TIME
151	012464	004767	006010		1\$:	JSP	PC,NULLS	:PRINT NULLS
152	012470	004767	004172		PNTRTN:	JSR	PC,MONIT	
153	012474	000167	177050			JMP	PST5	

.SBTTL HOST-REMOTE EXERCISE

```

1
2
3
4
5
6
7
8
9
10
11
12 012500
    012500 012746 000000
    012504 012746 012512
    012510 000002
    012512
13 012512 005067 015064
14 012516 012767 000010 015050
15 012524 105067 015012
16 012530 005067 015036
17 012534 012767 000000 015022
18 012542 027727 015020 102000
19 012550 001420
20 012552 005367 015006
21 012556 001371
22 012560
    012560 032777 040000 015046
    012566 001007
    012570 012700 024031
    012574 012767 000047 004006
    012602 004767 003634
    000050
23 012606
    012606 004767 003356
24 012612 005067 014742
25 012616 005077 014744
26 012622 012777 000020 014736
27 012630 012777 000044 014730
28 012636
    012636 012700 023342
    012642 004767 004424
29 012646 105067 014671
30 012652 105067 014672
31 012656 005067 014760
32 012662 005067 013576
33 012666 005067 013772
34 012672 005067 014166
35 012676 005067 014362
36
37
38
39 012702 004767 000574
40 012706 004767 003754
41 012712 032777 040000 014646
42 012720 001410
43 012722 052777 000020 014636

```

```

:START BACKGROUND SCAN, WAIT FOR 'THRU'
:REPORT ALARMS WHILE WAITING.
:WHEN 'THRU' SETS, DO A 'PORT CONFIGURATION STATUS' REPORT
:AND THEN RUN 'DIAGNOSTIC' FUNCTION ON ALL
:AVAILABLE REMOTES
:REPORT ALL CHANGES OF REMOTE STATUS

```

```

CMREXC: MTPS #0 ;INTERRUPT ON
MOV #0,-(SP)
MOV #648,-(SP)
RTI
648:
CLR PASCNT ;CLEAR PASS COUNTER
MOV #10,ITER ;LOAD ITERATION COUNT
CLRB OFLOF ;CLR FLAGS & COUNTERS
CLR ERRCNT
MOV #0,CNTR ;SET UP TO TIME POWER UP
CMP @CSR,#102000 ;PWR-UP & INTR SET?
BEQ EXS1 ;YES, CONTINUE
DEC CNTR ;NO, WAIT A BIT
BNE CELP
ERROR W,NOGO ;ERROR:HOST MICRO WON'T START
;***** ERROR 47 *****

```

```

BIT #B14,@CSR
BNE .+20
MOV #NOGO,RO
MOV #47,ERRNUM
JSR PC,ERR
=N
HLT
JSR PC,SUHLT ;RUN LOGIC TEST
EXS1: CLR CHNGFL ;CLEAR STATUS CHANGE FLAG
CLR @CSR ;START OFF CLEAR
MOV #20,@CSR ;CLEAR TOP BYTE
MOV #44,@CSR ;ENABLE BKG SCAN & THRU INTR
PNTM EXCM ;PRINT 'CMR EXERCISE ROUTINE'
MOV #EXCM,RO ;PRINT MESSAGE
JSR PC,TYP0UT ;POINTED TO BY EXCM
CLRB ONCE
CLRB THRU
CLR TALLY
CLR EXBL1
CLR EXBL2
CLR EXBL3
CLR EXBL4

```

```

:EXERCISER MAIN LOOP
EXS2: JSR PC,SCALM ;GO SCAN ALARMS
JSR PC,MONIT ;CHECK FOR CNTRL CHARS.
BIT #B14,@CSR ;IS FUN COMPL SET?
BEQ EXS2B ;NO.
BIS #20,@CSR ;CLR TOP BYTE

```

44	012730	005767	014624		TST	CHNGFL		:CHANGE FLAG SET?
45	012734	001022			BNE	EXS3		:IF YES, DO PCS
46	012736	004767	003724		JSR	PC,MONIT		:CHECK FOR CNTRL CHARS
47	012742	105767	014602	EXS2B:	TSTB	THRU		:IS 'THRU' SET?
48	012746	001015			BNE	EXS3		:IF YES, DO PORT CONFIG
49	012750	105767	014567		TSTB	ONCE		:DONE PORT CONFIG ONCE?
50	012754	001752			BEQ	EXS2		:NO, WAIT FOR THRU
51	012756	005767	014660		TST	TALLY		:YES, ANY REMOTES?
52	012762	001747			BEQ	EXS2		:NO, WAIT FOR CHANGE OR THRU
53	012764	005767	014570		TST	CHNGFL		:YES, ANY CHANGES PENDING?
54	012770	001004			BNE	EXS3		:YES, NEW PORT CONFIG
55	012772	004767	000230		JSR	PC,DIAGFU		:NO, GO DO DIAG FUNC ON AVAIL REMOTES
56	012776	000167	000406		JMP	SKD2		:THEN PRINT END PASS STUFF
57	013002	004767	000100	EXS3:	JSR	PC,PRTCNF		:DO A PORT CONFIG FUNC.
58	013006	005067	014546		CLR	CHNGFL		:CLR CHANGE FLAG
59	013012	112777	000074	014546	MOVB	#74,@CSR		:ENAB BKG SCN,CHGST,THRU INTR
60	013020	005767	014616		TST	TALLY		:ANY REMOTES?
61	013024	001411			BEQ	EXS3A		:NO, IDLE
62	013026	105767	014511		TSTB	ONCE		:ALREADY DONE REPORT?
63	013032	001017			BNE	EXS3B		:YES, SKIP IT THIS TIME
64	013034	004767	001334		JSR	PC,REPORT		:REPORT ALL REGULAR REMOTES
65	013040	112767	177777	014465	MOVB	#-1,CNFGFL		:SET PORT CONFIG FLAG
66	013046	000411			BR	EXS3B		
67	013050	105067	014467	EXS3A:	CLRB	ONCE		:NO REMOTES, WAIT FOR THRU
68	013054				PNTM	NOREM		:PRINT 'NO REMOTES..WAITING'
	013054	012700	023450		MOV	ANDREM,RO		:PRINT MESSAGE
	013060	004767	004206		JSR	PC,TYPOUT		:POINTED TO BY NOREM
69	013064	105067	014460		CLRB	THRU		:CLEAR THRU FLAG
70	013070	000704			BR	EXS2		
71	013072	105067	014452	EXS3B:	CLRB	THRU		:CLEAR THRU FLAG
72	013076	112767	177777	014437	MOVB	#-1,ONCE		:SET 'ONCE' FLAG
73	013104	000676			BR	EXS2		

```

1          :PORT CONFIGURATION STATUS SUBROUTINE
2          :PERFORMS A P.C.S. ON ALL 4 PORTS AND SAVES THE
3          :REMOTE #'S IN TABLES
4
5 013106 005067 014446      PRTCNF: CLR      CHNGFL      ;CLEAR 'CHANGE FLAG'
6 013112 005067 014524      CLR      TALLY      ;CLEAR REMOTE TALLY
7 013116 012767 026464 014520  MOV     #EXTBL1,TBLAD ;SET UP TO CONFIG PORT 0
8 013124 012767 000000 014456  MOV     #0,PORN
9 013132 004767 000734      JSR     PC,PORCNF      ;DO PORT CONFIG ON PORT 0
10 013136 012767 026664 014500  MOV     #EXTBL2,TBLAD
11 013144 012767 000001 014436  MOV     #1,PORN
12 013152 004767 000714      JSR     PC,PORCNF      ;DO PORT CONFIG ON PORT 1
13 013156 012767 027064 014460  MOV     #EXTBL3,TBLAD
14 013164 012767 000002 014416  MOV     #2,PORN
15 013172 004767 000674      JSR     PC,PORCNF      ;DO PORT CONFIG ON PORT 2
16 013176 012767 027264 014440  MOV     #EXTBL4,TBLAD
17 013204 012767 000003 014376  MOV     #3,PORN
18 013212 004767 000654      JSR     PC,PORCNF      ;DO PORT CONFIG ON PORT 3
19 013216 005767 014336      TST     CHNGFL      ;ANY CHANGES PENDING?
20 013222 001331      BNE     PRTCNF      ;IF SO, DO IT ALL OVER
21 013224 000207      RTS     PC
    
```

```

1          ;SUBROUTINE TO DO THE DIAGNOSTIC FUNCTION ON
2          ;ALL REMOTES WHICH ARE ON LINE AND REGULAR
3
4 013226 012701 026464   DIAGFU: MOV    #EXTBL1,R1      ;GET TABLE FOR PORT 0
5 013232 012167 014326   MOV    (R1)+,CNTR    ;GET # OF REMOTES ON PORT
6 013236 001002          BNE    SKP1          ;NONE, NEXT PORT
7 013240 000167 000016   JMP    SKP2
8 013244 012167 014350   SKP1:  MOV    (R1)+,REMNO ;GET 1ST REMOTE #
9 013250 004767 001510   JSR    PC,DIAGNF    ;DO DIAGNOSTIC FUNC. ON REM.
10 013254 005367 014304  DEC    CNTR         ;DO ALL REMOTES ON THIS PORT
11 013260 001371          BNE    SKP1
12
13 013262 012701 026664   SKP2:  MOV    #EXTBL2,R1 ;GET TABLE FOR PORT 1
14 013266 012167 014272   MOV    (R1)+,CNTR    ;1ST ENTRY IS # OF REMOTES
15 013272 001002          BNE    SKP3
16 013274 000167 000016   JMP    SKP4          ;IF NONE, NEXT PORT
17 013300 012167 014314   SKP3:  MOV    (R1)+,REMNO ;GET REMOTE NO.
18 013304 004767 001454   JSR    PC,DIAGNF    ;DO DIAGNOSTIC FUNCTION ON REM.
19 013310 005367 014250   DEC    CNTR
20 013314 001371          BNE    SKP3
21 013316 012701 027064   SKP4:  MOV    #EXTBL3,R1 ;GET TABLE FOR PORT 2
22 013322 012167 014236   MOV    (R1)+,CNTR    ;1ST ENTRY IS # OF REMOTES
23 013326 001002          BNE    SKP5
24 013330 000167 000016   JMP    SKP6          ;IF NONE, NEXT PORT
25 013334 012167 014260   SKP5:  MOV    (R1)+,REMNO ;GET REMOTE #
26 013340 004767 001420   JSR    PC,DIAGNF    ;DO DIAGNOSTIC FUNC. ON REMOTE
27 013344 005367 014214   DEC    CNTR
28 013350 001371          BNE    SKP5
29 013352 012701 027264   SKP6:  MOV    #EXTBL4,R1 ;GET TABLE FOR PORT 3
30 013356 012167 014202   MOV    (R1)+,CNTR    ;1ST ENTRY IS # OF REMOTES
31 013362 001002          BNE    SKP7
32 013364 000167 000016   JMP    SKP8          ;IF NONE, DONE!
33 013370 012167 014224   SKP7:  MOV    (R1)+,REMNO ;GET REMOTE #
34 013374 004767 001364   JSR    PC,DIAGNF    ;DO DIAGNOSTIC FUNCTION ON REM.
35 013400 005367 014160   DEC    CNTR         ;FINISH ALL REMOTES ON THIS PORT
36 013404 001371          BNE    SKP7
37 013406 000207          SKP8:  RTS     PC      ;RETURN
38
39
40 013410 005367 014160   SKPD2: DEC    ITER    ;UPDATE ITERATION COUNT
41 013414 001404          BEQ    SKPX0        ;IF 0, END PASS.
42 013416 032777 004000 014210 BIT    #B11,ASR     ;SW 11 = 1 == INH ITER.
43 013424 001422          BEQ    SKPX1
44 013426 012767 000010 014140 SKPX0: MOV    #10,ITER   ;RELOAD ITERATION COUNT
45 013434 005267 014142   INC    PASCNT       ;UPDATE PASS COUNT
46 013440 032777 002000 014166 BIT    #B10,ASR     ;SW 10 = INH END PASS PRINT
47 013446 001011          BNE    SKPX1
48 013450 004767 175374   JSR    PC,ENPRTM    ;PRINT END PASS STUFF
49 013454 105767 014050   TSTB  CLKFL1       ;IS CLOCK TICKING?
50 013460 001402          BEQ    18           ;NO, SKIP PRINTOUT
51 013462 004767 003502   JSR    PC,PRINTM    ;PRINT ELAPSED TIME
52 013466 004767 005006   18:   JSR    PC,LLLS
53 013472 004767 003170   SKPX1: JSR    PC,MONIT
54 013476 000167 177200   JMP    EXS?        ;KEEP GOING

```



```

1      ;SUBROUTINE TO CHECK FOR ALARMS AND CHANGES OF REMOTE STATE.
2      ; IF ALARM... RING 4 BELLS
3      ; IF CHANGE OF REMOTE STATE...SET A FLAG
4
5
6 013502 032777 010000 014056 SCALM: BIT #B12,@CSR ;ALARMS?
7 013510 001415 BEQ SCA1 ;NO, CHECK FOR CHG ST.
8 013512 012700 000007 MOV #7,R0 ;ALARM! RING TTY BELLS
9 013516 004767 005100 JSR PC,TTO
10 013522 004767 005074 JSR PC,TTO
11 013526 004767 005070 JSR PC,TTO
12 013532 004767 005064 JSR PC,TTO
13 013536 052777 000020 014022 BIS #B4,@CSR ;CLR ALARM
14 013544 000240 SCA1: NOP
15 013546 000240 NOP
16 013550 000240 NOP
17 013552 032777 020000 014006 BIT #B13,@CSR ;CHANGE OF STATE?
18 013560 001531 BEQ SCA3 ;NO, CHECK THRU
19 013562 012777 001000 014040 MOV #1000,@SPAR ;SEE IF THIS IS A SYS ERR
20 013570 027727 014036 000310 CMP @SPDR,#310 ; OR REMOTE CHANGE
21 013576 001415 BEQ SCA2 ;REMOTE CHANGE IS O.K.
22 013600 ERROR W,SYSER ;ERROR:HOST SYS ERROR. RUN LOGIC TEST.
    ;***** ERROR 50 *****
    013600 032777 040000 014026 BIT #B14,@SR
    013606 001007 BNE .+20
    013610 012700 026141 MOV #SYSER,R0
    013614 012767 000050 002766 MOV #50,ERRNUM
    013622 004767 002614 JSR PC,ERR
    N
23 013626 HLT
    013626 004767 002336 JSR PC,SUHLT
24 013632 SCA2: PNTM REMCHG ;PRINT 'REMOTE STATUS CHANGE'
    013632 012700 023657 MOV #REMCHG,R0 ;PRINT MESSAGE
    013636 004767 003430 JSR PC,TYPOUT ;POINTED TO BY REMCHG
25 013642 PNTM REMSG ;PRINT 'REMOTE #'
    013642 012700 023566 MOV #REMSG,R0 ;PRINT MESSAGE
    013646 004767 003420 JSR PC,TYPOUT ;POINTED TO BY REMSG
26 013652 012777 001003 013750 MOV #1003,@SPAR
27 013660 017700 013746 MOV @SPDR,R0 ;GET THE REMOTE NUMBER
28 013666 004767 004426 JSR PC,OCYPT ;PRINT IT
29 013670 PNTM REMSTM ;PRINT 'REMOTE STATUS NOW'
    013670 012700 023715 MOV #REMSTM,R0 ;PRINT MESSAGE
    013674 004767 003372 JSR PC,TYPOUT ;POINTED TO BY REMSTM
30 013700 005277 013724 INC @SPAR
31 013706 032777 000040 013720 BIT #B5,@SPDR ;NOW REGULAR?
32 013712 001405 BEQ SCA2A ;NO
33 013714 PNTM REGMG ;YES, PRINT 'REGULAR'
    013714 012700 023741 MOV #REGMG,R0 ;PRINT MESSAGE
    013720 004767 003346 JSR PC,TYPOUT ;POINTED TO BY REGMG
34 013724 000416 BR SCA2C
35 013726 032777 000100 013676 SCA2A: BIT #B6,@SPDR ;IRREGULAR?
36 013734 001405 BEQ SCA2B ;NO
37 013736 PNTM IRRMG ;YES, PRINT 'IRREGULAR'
    013736 012700 023752 MOV #IRRMG,R0 ;PRINT MESSAGE
    013742 004767 003324 JSR PC,TYPOUT ;POINTED TO BY IRRMG
38 013746 000410 BR SCA2D
39 013750 SCA2B: MOV @SPDR,R0 ;OTHERWISE, JUST PRINT THE STATUS
    
```

40	013754	004767	004336			JSR	PC,OCTPNT	
41	013760	000410				BR	SCA2E	:AND LEAVE THE TALLY ALONE
42	013762	005267	013654		SCA2C:	INC	TALLY	:UPDATE TALLY FOR ADDED REMOTE
43	013766	000405				BR	SCA2E	
44	013770	005767	013646		SCA2D:	TST	TALLY	:IS TALLY ALREADY =0?
45	013774	001402				BEQ	SCA2E	:YES, FORGET IT
46	013776	005367	013640			DEC	TALLY	:NO, MAKE IT ONE LESS
47	014002	005767	013634		SCA2E:	TST	TALLY	:IS TALLY NOW 0?
48	014006	001006				BNE	SCA2F	:NO, O.K.
49	014010	105067	013527			CLRB	ONCE	:YES, HAVE TO START OVER
50	014014					PNTM	ALROFF	:PRINT 'ALL REMOTES ARE IRREGULAR'
	014014	012700	023765			MOV	#ALROFF,R0	:PRINT MESSAGE
	014020	004767	003246			JSR	PC,TYPOUT	:POINTED TO BY ALROFF
51	014024	012767	177777	013526	SCA2F:	MOV	#-1,CHNGFL	:SET CHANGE FLAG
52	014032	052777	000020	013526		BIS	#20,@CSR	:CLEAR TOP BYTE
53	014040	000167	177500			JMP	SCA1	:GO SEE IF ANY MORE CHANGES.
54	014044	032777	004000	013514	SCA3:	BIT	#B11,@CSR	:IS THRU SET?
55	014052	001406				BEQ	SCART	:NO, LEAVE
56	014054	112767	177777	013466		MOVB	#-1,THRUFL	:YES, SET THE 'THRUFL' FLAG
57	014062	052777	000020	013476		BIS	#20,@CSR	:CLEAR TOP BYTE
58	014070	000207			SCART:	RTS	PC	

```

1      ;SUBROUTINE TO PERFORM 'PORT CONFIGURATION STATUS' ON A GIVEN
2      ;PORT AND STORE THE RESULTS IN A GIVEN TABLE. ALSO
3      ;STORED IS THE STATUS OF EACH REMOTE
4      ;FOR EACH REGULAR REMOTE FOUND, A TALLY IS INCREMENTED
5
6      ;TABLES ARE IN THE FOLLOWING FORMAT:
7
8      :      [ # OF ENTRIES ]
9      :      [ REM STAT][REM # ]
10     :      [      ][      ]
11
12     014072 016701 013546      PORCNF: MOV      TBLAD,R1      ;TABLE ADDR GIVEN BY CALLER
13     014076 052777 000020 013462  BIS      #B04,@CSR      ;CLR TOP BYTE OF CSR
14     014104 005077 013520      CLR      @SPAR          ;ADDR REQ. SCPD
15     014110 012777 000140 013514  MOV      #140,@SPDR     ;LOAD PORT CONF. ST. FUNC.
16     014116 005277 013506      INC      @SPAR
17     014122 012777 000002 013502  MOV      #2,@SPDR
18     014130 005277 013474      INC      @SPAR          ;DATA LEN. = 2
19     014134 012777 000001 013470  MOV      #1,@SPDR
20     014142 005277 013462      INC      @SPAR          ; ONE REQUEST
21     014146 016777 013436 013456  MOV      PORN,@SPDR     ;PORT # GIVEN BY CALLER
22     014154 004767 177322      JSR      PC,SCALM       ;CLEAR ALARMS & CHANGES
23     014160 052777 000001 013400  BIS      #B00,@CSR     ;START FUNCTION
24     014166 032777 040000 013372  PORO:  BIT      #B14,@CSR ;FUNCTION COMPLETE?
25     014174 001003      BNE      PORO0
26     014176 004767 177300      JSR      PC,SCALM
27     014202 000771      BR       PORO
28     014204 004767 002456 013412  PORO0: JSR      PC,MONIT
29     014210 012777 001000 000140  MOV      #1000,@SPAR   ;ADDR RESULT SCPD
30     014216 027727 013410      CMP      @SPDR,#140    ;THIS RESULT FROM PORT CONFIG?
31     014224 001415      BEQ      POR1
32     014226      ERROR      VN,YSER      ;ERROR:HOST SYS ERROR. RUN LOGIC TEST
33     014226 032777 040000 013400  BIT      #B14,@SR
34     014234 001007      BNE      .+20
35     014236 012700 026141      MOV      #YSER,R0
36     014242 012767 000051 002340  MOV      #51,ERRNUM
37     014250 004767 002166      JSR      PC,ERR
38     014254 000052      =      N+1
39
40     014254 004767 001710      N
41     014254 000167 000042      HLT
42     014260 005277 013344      JSR      PC,SUHLT
43     014264 017700 013342      POR1:  INC      @SPAR
44     014270 006200      MOV      @SPDR,R0
45     014272 006200      ASR      R0
46     014274 010021      ASR      R0
47     014276 001002      MOV      R0,(R1)+
48     014300 000167 000042      BNE      PORLD
49     014304 010067 013254      JMP      PORXT
50     014310 012777 001003 013312  PORLD: MOV      R0,CNTR
51     014316 117721 013310      MOV      #1003,@SPAR
52     014322 005277 013302      PORCN: MOV      @SPDR,(R1)+
53     014326 117721 013300      INC      @SPAR
54     014332 062777 000003 013270  MOV      @SPDR,(R1)+
55     014340 005367 013220      ADD      #3,@SPAR
56     014344 001364      DEC      CNTR
57     014346 005767 013236      BNE      PORCN
58     PORXT: TST      PORN
59
60     ;GET # OF REMOTES
61     ;DIVIDE BY 4
62     ;SAVE AS # OF ENTRIES
63     ;IF MORE THAN 0, LOAD THEM
64     ;IF NONE, EXIT
65     ;GET # OF ENTRIES
66     ;LOAD REMOTE #
67     ;LOAD REM. STATUS
68     ;GOT THEM ALL?
69     ;NO, GET THE REST
70     ;PORT 0?

```

50	014352	001002			BNE	1\$:NO, NEVER MIND
51	014354	005067	013262		CLR	TALLY	:YES, CLEAR TALLY
52	014360	060067	013256		ADD	RO, TALLY	:UPDATE TALLY
53	014364	052777	000020	013174	BIS	#B04,@CSR	:CLR TOP BYTE
54	014372	000207			RTS	PC	:AND RETURN

```

1
2
3
4 014374          :SUBROUTINE TO REPORT THE STATUS OF EACH PORT WHICH HAS ANY
      014374 012700 023600 :REMOTES ATTACHED AND REGULAR.
      014400 004767 002666
5 014404 105067 013134
6 014410 012767 000006 013160
7 014416 012701 026464
8 014422 012167 013136
9 014426 001407
10 014430 105267 013110
11 014434 012767 000000 013146
12 014442 004767 000130
13 014446 012701 026664
14 014452 012167 013106
15 014456 001407
16 014460 105267 013060
17 014464 012767 000001 013116
18 014472 004767 000100
19 014476 012701 027064
20 014502 012167 013056
21 014506 001407
22 014510 105267 013030
23 014514 012767 000002 013066
24 014522 004767 000050
25 014526 012701 027264
26 014532 012167 013026
27 014536 001407
28 014540 105267 013000
29 014544 012767 000003 013036
30 014552 004767 000020
31 014556 105767 012762
32 014562 001004
33 014564
      014564 012700 023765
      014570 004767 002476
34 014574 000207
    REPORT: PNTM      CONFMS
            MOV      #CONFMS,R0
            JSR      PC, TYPOUT
            CLRB     PNCFLG
            MOV      #6,LINLEN
            MOV      #EXTBL1,R1
            MOV      (R1)+,CNTR
            BEQ      REPO
            INCB     PNCFLG
            MOV      #0,PORN
            JSR      PC,PRNREM
            MOV      #EXTBL2,R1
            MOV      (R1)+,CNTR
            BEQ      REP1
            INCB     PNCFLG
            MOV      #1,PORN
            JSR      PC,PRNREM
            MOV      #EXTBL3,R1
            MOV      (R1)+,CNTR
            BEQ      REP2
            INCB     PNCFLG
            MOV      #2,PORN
            JSR      PC,PRNREM
            MOV      #EXTBL4,R1
            MOV      (R1)+,CNTR
            BEQ      REP20
            INCB     PNCFLG
            MOV      #3,PORN
            JSR      PC,PRNREM
            REP20: TSTB  PNCFLG
            BNE      REP3
            PNTM     ALROFF
            MOV      #ALROFF,R0
            JSR      PC, TYPOUT
            REP3:   RTS
            PC
    :PRINT '** PORT CONFIGURATION **'
    :PRINT MESSAGE
    :POINTED TO BY CONFMS
    :CLR 'PRINTED SOMETHING' FLAG
    :ALLOW 6 REM'S PER LINE.
    :GET PORT TABLE
    :GET # OF REMOTES
    :NONE HERE, NEXT PORT
    :SET FLAG
    :LOAD PORT NO.
    :GO PRINT REMOTES ON THIS PORT
    :GET # OF REMOTES
    :NONE HERE, NEXT PORT
    :SET FLAG
    :GO PRINT REMOTES ON THIS PORT
    :GET # OF REMOTES
    :NONE HERE, NEXT PORT
    :SET FLAG
    :GO PRINT REMOTES ON THIS PORT
    :PRINTED ANYTHING?
    :YES, FINE.
    :NO, SAY ALL ARE IRREGULAR.
    :PRINT MESSAGE
    :POINTED TO BY ALROFF
    :RETURN WHEN DONE
    
```

```

1      ;SUBROUTINE TO DO THE PRINTING PART OF PORT CONFIGURATION REPORT.
2      ;R1 POINTS AT FIRST REMOTE #
3      ;PORTN = PORT #
4      ;CNTR = # OF REMOTES ON THIS PORT
5
6
7      014576      PRNREM: PNTM      PORTMS      ;PRINT 'PORT # '
      014576      MOV          #PORTMS,R0      ;PRINT MESSAGE
      014602      JSR          PC, TYP0UT      ;POINTED TO BY PORTMS
8      014606      MOV          PORTN,R0      ;GET PORT #
9      014612      JSR          PC, OCTPNT      ;PRINT IT
10     014616      PREMO:  MOVB      (R1)+,R0      ;GET REMOTE #
11     014620      BIC          #177400,R0      ;CLR HIGH BYTE
12     014624      JSR          PC, OCTJSP      ;PRINT IT
13     014630      MOV          #'(',R0      ;BRACKET THE STATE
14     014634      JSR          PC, TTO
15     014640      MOVB      (R1)+,R0      ;GET REM. STATE
16     014642      BIC          #177400,R0      ;CLEAR THE JUNK OFF
17     014646      JSR          PC, OCTPNT      ;PRINT THE STATE
18     014652      MOV          #')',R0      ;CLOSE BRACKETS
19     014656      JSR          PC, TTO
20     014662      DEC          CNTR
21     014666      BEQ          PREML
22     014670      DEC          LINLEN
23     014674      BNE          PREMO
24     014676      MOV          #6,LINLEN      ;YES, RESTORE LINLEN
25     014704      MOV          #15,R0
26     014710      JSR          PC, TTO
27     014714      MOV          #7,CLK
28     014722      MOV          #40,R0
29     014726      PREM1:  JSR          PC, TTO
30     014732      DEC          CLK
31     014736      BNE          PREM1
32     014740      JMP          PREMO
33     014744      PREML:  MOV          #15,R0      ;ANOTHER <CR>
34     014750      JSR          PC, TTO
35     014754      MOV          #6,LINLEN
36     014762      RTS          PC      ;AND RETURN
    
```

```

1          ;SUBROUTINE TO DO DIAGNOSTIC FUNCTION ON A GIVEN REMOTE
2          ;ENTER WITH REMOTE # IN 'REMNO'
3
4
5 014764 012767 000020 012570 DIAGNF: MOV #20,CLK ;SET UP TO LOAD REQUEST
6 014772 012702 026424          MOV #DATBL,R2
7 014776 005077 012626          CLR @SPAR ;ADDRESS REQ SCR PAD
8 015002 012777 000100 012622 MOV #100,@SPDR ;LOAD 'DIAG' FUNC
9 015010 005277 012614          INC @SPAR
10 015014 012777 000022 012610 MOV #22,@SPDR ;DATA LEN. = 22
11 015022 005277 012602          INC @SPAR
12 015026 012777 000001 012576 MOV #1,@SPDR ;ONE REQUEST
13 015034 005277 012570          INC @SPAR
14 015040 016777 012554 012564 MOV REMNO,@SPDR ;LOAD REMOTE #
15 015046 005277 012556          DIAGO: INC @SPAR
16 015052 012277 012554          MOV (R2)+,@SPDR ;LOAD DATA PATTERN
17 015056 005367 012500          DEC CLK
18 015062 001371          BNE DIAGO
19 015064 004767 176412          DIAG00: JSR PC,SCALM ;CLEAR ALARMS ETC.
20 015070 052777 000020 012470 BIS #B04,@CSR ;CLR TOP BYTE
21 015076 052777 000001 012462 BIS #B00,@CSR ;SET START FUNC.
22 015104 032777 040000 012454 DIAG1: BIT #B14,@CSR ;FUN COMPL?
23 015112 001003          BNE DIAG10
24 015114 004767 176362          JSR PC,SCALM ;NO, WAIT FOR IT
25 015120 000771          BR DIAG1
26 015122 012777 001000 012500 DIAG10: MOV #1000,@SPAR ;ADDRESS RESULT SCR. PAD
27 015130 012767 000100 001454 MOV #100,GOOD ;RIGHT FUNCTION?
28 015136 017767 012470 001450 MOV @SPDR,BAD
29 015144 026767 001442 001442 CMP GOOD,BAD
30 015152 001416          BEQ DIAG2
31 015154          DATERR W,DIAGF ;ERROR:DIAG FUNC GAVE WRONG FUNC CODE
                                ;***** ERROR 52 *****

015154 032777 040000 012452          BIT #B14,@SR
015162 001007          BNE .+20
015164 012700 026202          MOV #DIAGF,R0
015170 012767 000052 001412 MOV #52,ERRNUM
015176 004767 001324          JSR PC,DERR
                                =
                                N
                                SCOPE
32 015202          JSR R5,SCRPTN
015202 004567 001002          DIAG00
015206 015064          DIAG00
33 015210 062777 000002 012412 DIAG2: ADD #2,@SPAR ;CHECK FUNCTION STATUS
34 015216 012767 000001 001366 MOV #1,GOOD ;SHD SHOW SUCCESS
35 015224 017767 012402 001362 MOV @SPDR,BAD
36 015232 026767 001354 001354 CMP GOOD,BAD
37 015240 001460          BEQ DIAG4
38 015242          DATERR W,DIAGUF ;O.K.
                                ;ERROR:USOF WRONG ON 'DIAG' FUNC
                                ;***** ERROR 53 *****

015242 032777 040000 012364          BIT #B14,@SR
015250 001007          BNE .+20
015252 012700 026245          MOV #DIAGUF,R0
015256 012767 000053 001324 MOV #53,ERRNUM
015264 004767 001236          JSR PC,DERR
                                =
                                N
39 015270 032777 040000 012336          BIT #B14,@SR ;PRINT ALLOWED?
40 015276 001036          BNE DIAG5S ;NO, SKIP IT
41 015300          PNTM REMSG ;PRINT 'REMOTE #'
    
```

```
015300 012700 023566      MOV      #REMSG,RO      ;PRINT MESSAGE
015304 004767 001762      JSR      PC,TYP0UT     ;POINTED TO BY REMSG
42 015310 005277 012314   INC      @SPAR
43 015314 017700 012312   MOV      @SPDR,RO      ;GET REM. NO.
44 015320 004767 002772   JSR      PC,OC1PNT     ;PRINT IT
45 015324 012700 023634   PNTM     STATEM       ;PRINT 'STATE '
015330 004767 001736      MOV      #STATEM,RO    ;PRINT MESSAGE
015334 005277 012270      JSR      PC,TYP0UT     ;POINTED TO BY STATEM
46 015334 005277 012270   INC      @SPAR
47 015340 017700 012266   MOV      @SPDR,RO      ;GET STAE OF REMOTE
48 015344 004767 002746   JSR      PC,OC1PNT
49 015350 012700 023646   PNTM     USORM        ;PRINT 'USOR = '
015354 004767 001712      MOV      #USORM,RO     ;PRINT MESSAGE
015360 005277 012244      JSR      PC,TYP0UT     ;POINTED TO BY USORM
50 015360 005277 012244   INC      @SPAR
51 015364 017700 012242   MOV      @SFDR,RO
52 015370 004767 002722   JSR      PC,OC1PNT
53 015374 012700 000610   DIAG3S: SCOPE         ;
015374 004567 000610      JSR      R5,SCPRTN
015400 015064 000020 012156  DIAG4: DIAG00
54 015402 052777 000020 012156  DIAG4: BIS      #B04,@CSR ;CLEAR TOP BYTE WHEN DONE
55 015410 000207 000020 012156  DIAG4: RTS      PC
```


.SBTTL MAINTENANCE ROUTINE

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17 015412 105767 012113      OOTPR:  TSTB    CLKFLG      ;STOP CLOCK WHILE WE'RE HERE
18 015416 001404              BEQ     $00$      ;BY CLEARING INTR ENAB
19 015420 017746 012242      MOV     @LCS,-(SP) ;IF IT'S SET
20 015424 005077 012236      CLR     @LCS
21 015430 112767 177777 012076 $00$:  MOV     #-1,DBFLG ;SET DEBUG FLAG
22 015436 017767 012124 012162 MOV     @CSR,SAVEC ;SAVE CSR FOR LOOKING AT
23 015444              PNTM    DBAID      ;PRINT 'CMR-11 DEBUG AID'
    015444 012700 021442      MOV     #DBAID,R0 ;PRINT MESSAGE
    015450 004767 001616      JSR     PC,TYPOUT ;POINTED TO BY DBAID
    015454 005067 012072      CLR     ADDR      ;GET ADDRESS
25 015460 016767 012066 002552 $0$:  MOV     ADDR,KBBUF ;FROM KEYBOARD
26 015466 004767 001642      JSR     PC,INPKB
27 015472 016767 002542 012052 MOV     KBBUF,ADDR
28 015500 026727 012046 001777 CMP     ADDR,#1777 ;WITHIN LIMIT?
29 015506 101423              BLOS   $2$        ;YES, O.K.
30 015510 026767 012036 012050 CMP     ADDR,CSR  ;IS IT THE CSR?
31 015516 001012              BNE    $10$       ;NO
32 015520 017700 012042      MOV     @CSR,R0
33 015524 004767 002614      JSR     PC,OCTJSP ;YES, SHOW CONTENTS OF CSR
34 015530 012700 000015      MOV     #15,R0
35 015534 004767 003062      JSR     PC,TTO
36 015540 000167 177710      JMP     $0$
37 015544              $10$:  PNTM    AGAIN
    015544 012700 021324      MOV     #AGAIN,R0 ;PRINT MESSAGE
    015550 004767 001516      JSR     PC,TYPOUT ;POINTED TO BY AGAIN
38 015554 000741              BR
39 015556 105767 011765      $2$:  TSTB    SLFL      ;SLASH HIT?
40 015562 001042              BNE    $11$       ;YES DUMP DATA
41 015564 105767 011745      TSTB    GOFLG     ;'G' TYPED?
42 015570 001462              BEQ    $20$       ;NO
43 015572 052777 000020 011766 $2A$:  BIS     #B04,@CSR ;INSURE TOP BYTE CLEAR
44 015600 005077 012024      CLR     @SPAR     ;ADDRESS FUNC. CODE
45 015604 027727 012022 000151 CMP     @SPDR,#151 ;IS FUNC NOT MAINT?
46 015612 101404              BLOS   $3$        ;IT'S NOT MAINT. DO IT
47 015614 052777 000200 011744 BIS     #B07,@CSR ;IT IS MAINT, SET 'MAIN'
48 015622 000403              BR     $4$
49 015624 052777 000001 011734 $3$:  BIS     #B00,@CSR ;SET 'STFUN'
50 015632 032777 100000 011726 $4$:  BIT     #B15,@CSR ;IS THERE AN INTR?
51 015640 001003              BNE    $5$        ;YES, SEE WHAT IT IS
52 015642 004767 001020      JSR     PC,MONIT
53 015646 000771              BR     $4$
    
```

54	015650	012767	001000	011674	\$5\$:	MOV	#1000, ADDR	:ADDRESS RESULT
55	015656	012700	000015			MOV	#15, R0	
56	015662	004767	002734			JSR	PC, ITO	:DO <CR>
57	015666	000430				BR	\$21\$	
58	015670	016777	011656	011732	\$11\$:	MOV	ADDR, @SPAR	:LOAD 'ADDRESS'
59	015676	017700	011730			MOV	@SPDR, R0	:GET 'CONTENTS'
60	015702	004767	002436			JSR	PC, OCTJSP	:AND PRINT IT
61	015706					PNTM	TWOSP	:SPACE AND PROMPT (:)
	015706	012700	021571			MOV	#TWOSP, R0	:PRINT MESSAGE
	015712	004767	001354			JSR	PC, TYP0UT	:POINTED TO BY TWOSP
62	015716	017767	011710	002314		MOV	@SPDR, KBBUF	
63	015724	004767	001404			JSR	PC, INPKB	:GET KBD INPUT
64	015730	016777	002304	011674		MOV	KBBUF, @SPDR	:LOAD ANY NEW INPUT
65	015736	105767	011575		\$20\$:	TSTB	INCF	:LINE FEED?
66	015742	001421				BEQ	\$13\$	
67	015744	005267	011602			INC	ADDR	:IF YES, INCREMENT ADDRESS
68	015750	016700	011576		\$21\$:	MOV	ADDR, R0	
69	015754	020027	001777			CMP	R0, #1777	:VALID ADDRESS?
70	015760	101402				BLOS	\$22\$:YES, O.K.
71	015762	000167	177556			JMP	\$10\$:NO GOOD, GET A NEW ONE!
72	015766	004767	002324		\$22\$:	JSR	PC, OCTPNT	:PRINT NEW ADDRESS
73	015772	012700	000057			MOV	#/, R0	:PRINT A SLASH
74	015776	004767	002620			JSR	PC, ITO	
75	016002	000167	177662			JMP	\$11\$:GO PRINT CONTENTS
76	016006	105767	011535		\$13\$:	TSTB	SLFL	:SLASH TYPED?
77	016012	001402				BEQ	\$23\$:NO, CHECK IF IT'S 'P'
78	016014	000167	177650			JMP	\$11\$:YES, SHOW CONTENTS
79	016020	105767	011506		\$23\$:	TSTB	CONFLG	: 'P' TYPED?
80	016024	001004				BNE	\$25\$:NO, MUST BE <CR>
81	016026	105767	011503			TSTB	G0FLG	: 'G' TYPED?
82	016032	001257				BNE	\$2A\$:YES, START FUNCTION
83	016034	000611				BR	\$1\$	
84	016036	105767	011467		\$25\$:	TSTB	CLKFLG	:CHECK ON CLOCK FLAG
85	016042	001402				BEQ	\$26\$:NO CLOCK, SKIP
86	016044	012677	011616			MOV	(SP)+, @LCS	:RESTORE CLOCK IF GOING
87	016050	105067	011460		\$26\$:	CLRB	DBFLG	:CLEAR DEBUG FLAG
88	016054	005767	011526			TST	PCONT	:ANY PROMPTS PENDING?
89	016060	001402				BEQ	\$24\$:NO, JUST RETURN TO CALLER
90	016062	016716	011520			MOV	PCONT, (SP)	:YES, RETYPE PROMPT.
91	016066	000207			\$24\$:	RTS	PC	:UNOFFICIAL RETURN

```

1
2
3
4
5
6
7
8 016070 010446
9 016072 010346
10 016074 010246
11 016076 010146
12 016100 010046
13 016102 000115
14
15 016104 030026
16 016106 012600
17 016110 012601
18 016112 012602
19 016114 012603
20 016116 012604
21 016120 000205
22
23
24
25 016122 112767 177777 011410
26 016130 000002
27
28
29
30 016132 012767 027636 011474
31 016140 012767 177777 011472
32 016146 000002
33
34
35
36 016150 012706 002000
37 016154
   016154 012700 021467
   016160 004767 001106
38 016164 000167 163722
39
40
41
42
43 016170 005767 011444
44 016174 001403
45 016176 004767 000516
46 016202 000207
47 016204 000000
48 016206 000207
49
50
51
52 016210 004767 000452
53 016214 005777 011414
54 016220 100402
55 016222

.SBTTL UTILITY ROUTINES

;ROUTINES TO SAVE AND RESTORE GENERAL REGISTERS
; CALLED BY JSR R5,REGSAV
; AND JSR R5,REGRES

REGSAV: MOV R4,-(SP)
        MOV R3,-(SP)
        MOV R2,-(SP)
        MOV R1,-(SP)
        MOV R0,-(SP)
        JMP @R5

REGRES: BIT R0,(SP)+ ;THROW AWAY OLD R5 VALUE
        MOV (SP)+,R0
        MOV (SP)+,R1
        MOV (SP)+,R2
        MOV (SP)+,R3
        MOV (SP)+,R4
        RTS R5

;ROUTINE TO SET LSI-11 FLAG (SOFTWARE)
LSTST: MOVB #1,LSIFLG ;SET LSI-11 SOFTWARE FLAG
        RTI ;RETURN

;ROUTINE TO SET SOFTWARE SWITCH REGISTER FLAG
SRTST: MOV #SSWR,SR ;NO HARDWARE SWR, USE
        MOV #1,SWRFLG ; MEMORY LOCATION
        RTI ; AND SET FLAG

;ROUTINE TO PRINT ILLEGAL DEVICE ADDRESS
DVATST: MOV #ISP,SP ;FIX THE STACK
        PNTM INVLAD ;PRINT NON-EXIST ADDR MSG
        MOV #INVLAD,R0 ;PRINT MESSAGE
        JSR PC,TYPOUT ;POINTED TO BY INVLAD
        JMP PROMT ;GO GET CORRECT ADDRESS

;ROUTINE TO REPLACE HALT WITH CALL TO SW MONITOR
;ON NON SWR PROCESSORS (CALLED BY MACRO 'HLT')
SWHLT: TST SWRFLG ;ANY HARDWARE SWR?
        BEQ IS ;IF YES, GO HALT
        JSR PC,SWDMP ;IF NO, DUMP SWITCHES
        RTS PC ;CONTINUE
IS: HALT ;REAL HALT
    RTS PC ;CONTINUE

;ROUTINE USED FOR LOOPING ON ERROR ROUTINE (CALLED BY MACRO 'SCOPE')
SCPRM: JSR PC,MONIT ;CHECK FOR ^G, OR ^C
        TST @SR ;IS SW 15 SET?
        BMI SEAK ;YES, DON'T HALT
        HLT ;COMMON ERROR HALT
    
```

```

56 016222 004767 177742          JSR    PC,SWHLT
57 016226 012500          SBK:   MOV    (R5)+,R0          ;GET DIRECTION FOR SCOPE LOOP
58 016230 032777 020000 011376  BIT    #B13,SR              ;SW 13 SET?
59 016236 001402          BEQ    SCNT                ;NO, DON'T LOOP
60 016240 012605          MOV    (SP)+,R5           ;YES, RESTORE R5
61 016242 000110          JMP    (R0)              ;AND LOOP
62 016244 005267 011322          SCNT:  INC    ERRCNT        ;UPDATE ERR CNT IF 'FALL THRU'
63 016250 001003          BNE    SCORTN            ;WATCH FOR ERRCNT OFLO
64 016252 112767 177777 011262  MOVB  #-1,OFLOF          ;SET OVERFLO FLAG.
65 016260 000205          SCORTN: RTS                ;AND RETURN
66
67                                     ;ROUTINE TO GENERATE CSC DEVICE ADDRESSES FROM 'DEVADR'
68
69 016262 016700 011370          DEVG:  MOV    DEVADR,R0    ;GET DEVICE CSR ADDRESS
70 016266 010067 011274          MOV    R0,CSR           ;GENERATE CSR
71 016272 062700 000002          ADD    #2,R0
72 016276 010067 011326          MOV    R0,SPAR         ;GENERATE SPAR
73 016302 062700 000002          ADD    #2,R0
74 016306 010067 011320          MOV    R0,SPDR         ;GENERATE SPDR
75 016312 016700 011336          MOV    VECTOR,R0
76 016316 012760 000340 000002  MOV    #340,2(R0)
77 016324 000207          RTS                    ;RETURN
78
79                                     ;ROUTINE TO PRINT DEFAULT VALUES FOR PROMPTS AT FRONT END.
80                                     ; ENTER WITH DEFAULT VALUE IN LOC. KBBUF
81
82 016326 012700 000050          PRDFT: MOV    #'(,R0      ;GET OPEN PARENTHESIS
83 016332 004767 002264          JSR    PC,ITO           ;PRINT IT
84 016336 016700 001676          MOV    KBBUF,R0        ;GET DEFAULT VALUE
85 016342 004767 001750          JSR    PC,OCTPNT       ;PRINT IT
86 016346 012700 000051          MOV    #'),R0         ;GET CLOSE PARENTHESIS
87 016352 004767 002244          JSR    PC,ITO           ;PRINT IT
88 016356 012700 000040          MOV    #' ,R0         ;AND PRINT A SPACE
89 016362 004767 002234          JSR    PC,ITO
90 016366 000207          RTS                    PC
91
92                                     ;ROUTINE TO CATCH ERROR TRAPS TO LOC 4
93
94 016370 011667 011252          HDR:   MOV    (SP),TEMP    ;SAVE STACK FOR ADDR OF TRAP
95 016374 012706 002000          MOV    #ISP,SP         ;RENEW STACK BEFORE USING
96 016400          PNTM  TRP4            ;PRINT 'TRAPPED TO 4 FROM LOC ..'
97 016400 012700 021533          MOV    #TRP4,R0       ;PRINT MESSAGE
98 016404 004767 000662          JSR    PC,TYPEOUT     ;POINTED TO BY TRP4
99 016410 162767 000002 011230  SUB    #2,TEMP
100 016416 016700 011224          MOV    TEMP,R0
101 016422 004767 001670          JSR    PC,OCTPNT      ;PRINT SOURCE OF TRAP
102 016426 004767 002046          JSR    PC,NULLS
103 016432 012706 002000          MOV    #ISP,SP        ;REFRESH STACK POINTER
104 016436 000167 164052          JMP    RESTART
105
106                                     ;ERROR ROUTINES: REGULAR ERRORS AND DATA ERRORS.
107
108 016442 011667 000140          ERR:   MOV    (SP),ERRAD  ;GET ADDRESS OF ERROR CALL
109 016446 162767 000026 000132  SUB    #2,ERRAD        ;CORRECT IT

```

```

110 016454 010046          ERR1:  MOV    RO, -(SP)          ;STACK ERROR MSG ADDR
111 016456                PNTM   ERRM          ;PRINT "**ERROR**"
      016456 012700 016616  MOV    #ERRM,RO      ;PRINT MESSAGE
      016462 004767 000604  JSR    PC, TYP0UT    ;POINTED TO BY ERRM
112 016466 016700 000116  MOV    ERRNUM,RO     ;PRINT ERROR NO. (P)
113 016472 004767 001620  JSR    PC, OCTPNT    ;PRINT " AT LOC. "
114 016476                PNTM   WDAT          ;PRINT MESSAGE
      016476 012700 016631  MOV    #WDAT,RO      ;POINTED TO BY WDAT
      016502 004767 000564  JSR    PC, TYP0UT
115 016506 016700 000074  MOV    ERRAD,RO      ;PRINT ADDRESS OF ERROR
116 016512 004767 001600  JSR    PC, OCTPNT    ;GET ADDR OF ERROR MSG
117 016516 012600                MOV    (SP)+,RO      ;PRINT ERR MSG
118 016520 004767 000546  JSR    PC, TYP0UT
119 016524 000207                RTS     PC          ;RETURN
120
121
122 016526 011667 000054    DERR:  MOV    (SP),ERRAD      ;GET ADDRESS OF ERROR CALL
123 016532 162767 000026 000046  SUB    #26,ERRAD     ;CORRECT IT
124 016540 004767 177710  JSR    PC,ERR1       ;PRINT REG. ERROR MSG
125 016544                PNTM   WDSDB          ;PRINT "SHOULD BE "
      016544 012700 016643  MOV    #WDSDB,RO     ;PRINT MESSAGE
      016550 004767 000516  JSR    PC, TYP0UT    ;POINTED TO BY WDSDB
126 016554 016700 000032  MOV    GOOD,RO       ;PRINT GOOD DATA
127 016560 004767 001532  JSR    PC, OCTPNT    ;PRINT " WAS "
128 016564                PNTM   WDWAS          ;PRINT MESSAGE
      016564 012700 016657  MOV    #WDWAS,RO     ;POINTED TO BY WDWAS
      016570 004767 000476  JSR    PC, TYP0UT
129 016574 016700 000014  MOV    BAD,RO        ;PRINT BAD DATA
130 016600 004767 001512  JSR    PC, OCTPNT
131 016604 000207                RTS     PC          ;RETURN
132
133
134
135
136 016606 000000          ;RELATED STORAGE:
137 016610 000000          ERRAD:  .WORD  0
138 016612 000000          ERRNUM: .WORD  0
139 016614 000000          GOOD:  .WORD  0
140
141
142
143
144 016616 046 046 052  ERRM:  .ASCII  /**ERROR */
      016621 052 105 122
      016624 122 117 122
      016627 040 100
145 016631 040 101 124  WDAT:  .ASCII  / AT LOC. */
      016634 040 114 117
      016637 103 056 040
      016642 100
146 016643 046 123 110  WDSDB: .ASCII  /SHOULD BE */
      016646 117 125 114
      016651 104 040 102
      016654 105 040 100
147 016657 054 040 127  WDWAS: .ASCII  /, WAS */
      016662 101 123 040
      016665 100
    
```

CZCMB0 CMR(V)-11 MOST EXERCISE MACRO V03.02B10-DEC-80 10:41:54 PAGE 28-3
UTILITY ROUTINES

148

.EVEN

SEQ 0070

```

1
2
3
4
5
6
7
8
9
10
11 016666 005000
12 016670 105777 010770
13 016674 100402
14 016676 000167 000264
15 016702 017700 010754
16 016706 042700 177600
17 016712 020027 000007
18 016716 001056
19 016720
    016720 012700 021575
    016724 004767 000342
20 016730 017700 010700
21 016734 004767 001356
22 016740
    016740 012700 021571
    016744 004767 000322
23 016750 017767 010660 001262
24 016756 012767 016720 010622
25 016764 004767 000344
26 016770 005067 010612
27 016774 016777 001240 010632
28 017002
    017002 012700 021650
    017006 004767 000260
29 017012 105777 010646
30 017016 100375
31 017020 017700 010636
32 017024 042700 177600
33 017030 020027 000007
34 017034 001731
35 017036 020027 000120
36 017042 001363
37 017044 012700 000015
38 017050 004767 001546
39 017054 020027 000024
40 017060 001004
41 017062 012706 002000
42 017066 000167 163572
43 017072 020027 000017
44 017076 001002
45 017100 000167 176306
46
47 017104 020027 000003
48 017110 001004
49 017112 012706 002000
50 017116 000167 163542
51 017122 020027 000006

.SBTTL 'SWITCH' MONITOR ROUTINE

:ENTER AT MONIT FROM EVERY SUB-TEST TO SEE IF CNTRL-G OR CNTRL-C WAS TYPED
:ENTER AT SWDMP FROM ERROR HALTS IF SW 15 = 0
:ALSO MONITORS THE FOLLOWING CONTROL FUNCTIONS:
:
:   P CONTINUE (PROCEED)
:   CNTRL-O GO TO CMR-11 DEBUG ROUTINE
:   CNTRL-A DUMP PORT CONFIG TABLES

MONIT: CLR      RO
        TSTB   @KBS
        BMI    MONIC
        JMP    EX6
MONIC: MOV     @KBD,RO
MONCH: BIC     #177600,RO
        CMP    RO,#07
        BNE   EX1
SWDMP: PNTM   SWMSG
        MOV   #SWMSG,RO
        JSR  PC, TYPDT
        MOV   @SR,RO
        JSR  PC, OCTPNT
        PNTM  TWOSP
        MOV   #TWOSP,RO
        JSR  PC, TYPDT
        MOV   @SR, KBBUF
        MOV   #SWDMP, PCONT
        JSR  PC, INPKB
        CLR  PCONT
        MOV   KBBUF, @SR
CCRTN: PNTM   TYPCTP
        MOV   #TYPCTP,RO
        JSR  PC, TYPDT
CONTW1: TSTB  @KBS
        BPL   CONTW1
        MOV   @KBD,RO
        BIC   #177600,RO
        CMP  RO,#07
        BEQ  SWDMP
        CMP  RO,#P
        BNE  CONTW1
        MOV  #15,RO
        JSR  PC, T10
EX1:   CMP    RO,#24
        BNE  EX2
        MOV  #ISP,SP
        JMP  BCONT
EX2:   CMP    RO,#17
        BNE  EX3
        JMP  OOTPR
EX3:   CMP    RO,#3
        BNE  EX4
        MOV  #ISP,SP
        JMP  BCONT
EX4:   CMP    RO,#6

:CHECK KEYBOARD FLAG
:IF SET, CHECK WHAT CHAR.
:OTHERWISE, EXIT

:TRIM OFF PARITY BIT
:WAS IT ^G?
:NO, EXIT
:PRINT 'SWR = '
:PRINT MESSAGE
:POINTED TO BY SWMSG
:GET CONTENTS OF SR
:PRINT IT
:SPACE AND PROMPT (:)
:PRINT MESSAGE
:POINTED TO BY TWOSP
:LOAD OLD SWITCHES
:LOAD RETURN ADDRESS
:GET KBD INPUT

:LOAD NEW SWITCHES
:PRINT 'P TO CONTINUE'
:PRINT MESSAGE
:POINTED TO BY TYPCTP

:TRIM OFF PARITY BIT
:^G?
:YES, GET SWR AGAIN
:P
:NO, KEEP LOOKING
:RETURN LINE

:WAS A ^T TYPED?
:NO, EXIT
:YES, RENEW STACK
:BACK TO DISPATCHER
:WAS CNTRL-O TYPED?
:NO
:AND GO TO DEBUG ROUTINE

:WAS CNTRL-C TYPED?
:NO
:YES, REFRESH STACK
:AND RESTART
:WAS IT ^F?

```

52	017126	001012		BNE	EX5	:NO, EXIT
53	017130	105767	010400	TSTB	DBFLG	:IS DEBUG FLAG SET?
54	017134	001007		BNE	EX5	:YES, JUST RETURN
55	017136	105767	010403	TSTB	SAFE	:ALREADY IN FILL RTN?
56	017142	001004		BNE	EX5	:YES, JUST RETURN
57	017144	005067	010436	CLR	PCONT	:CLEAR PROMPT CARRIER
58	017150	004767	000432	JSR	PC,KBRET	:SNEAK INTO FILL COUNT ROUTINE
59	017154	020027	000001	EX5: CMP	RO,#1	:WAS 'A' TYPED?
60	017160	001002		BNE	EX6	:NOPE, BYE!
61	017162	004767	001100	JSR	PC,PNTCNF	:YES, PRINT PORT CONFIG
62	017166	000207		EX6: RTS	PC	
63						
64						
65						:ROUTINE TO PRINT ELAPSED TIME SINCE START-UP OR CNTRL-C
66						
67	017170			PRNTIM: PNTM	ELTIM	:PRINT 'EL. TIM. '
	017170	012700	023164	MOV	#ELTIM,RO	:PRINT MESSAGE
	017174	004767	000072	JSR	PC,TYPNT	:POINTED TO BY ELTIM
68	017200	116700	010332	MOVB	HOURS,RO	:GET HOURS
69	017204	004767	001212	JSR	PC,DECPNT	:PRINT IN DECIMAL
70	017210	012700	000072	MOV	#:,RO	
71	017214	004767	001402	JSR	PC,TTO	:PRINT ':'
72	017220	116700	010315	MOVB	MINS,RO	:GET MINUTES
73	017224	004767	001172	JSR	PC,DECPNT	:PRINT IN DECIMAL
74	017230	012700	000072	MOV	#:,RO	
75	017234	004767	001362	JSR	PC,TTO	:ANOTHER ':'
76	017240	116700	010302	MOVB	SECS,RO	:GET SECS
77	017244	004767	001152	JSR	PC,DECPNT	:PRINT IN DECIMAL
78	017250	012700	000072	MOV	#:,RO	
79	017254	004767	001342	JSR	PC,TTO	:ANOTHER ':'
80	017260	116700	010265	MOVB	TICKS,RO	:GET TICKS (1/60 SEC)
81	017264	004767	001132	JSR	PC,DECPNT	:PRINT IN DECIMAL
82	017270	000207		RTS	PC	:RETURN


```

1          .SBTTL MESSAGE PRINT ROUTINE
2
3          :MESSAGE TYP0UT ROUTINE (CALLED BY MACRO PN1M A)
4          :MESSAG0S ARE IN THE FORMAT:
5          :   MESG:      .ASCII      /@MESSAGE@/
6          :
7          :WHERE: @ IS TRANSLATED INTO CR. AND LF.
8          :
9          :USES THE SUBROUTINE 'TTO'
10         :WHICH PRINTS CR. & LF. UPON SEEING A CR. CODE.
11         :AND  @ IS MESSAGE TERMINATOR
12         :
13         :ENTER WITH ADDRESS OF MESSAGE IN R0
14
15 017272 010046          TYP0UT: MOV      R0,-(SP)          :STACK ADDRESS OF MESSAGE
16 017274 117600 000000  TPOFCH: MOVB   @(SP),R0          :FETCH ASCII BYTE
17 017300 022700 000100          :CMP      #100,R0          :IS IT @ (TERMINATOR)?
18 017304 001411          :BEQ     TPOUTX          :YES-EXIT
19 017306 022700 000046          :CMP      #46,R0          :IS IT CRLF FLAG?
20 017312 001002          :BNE     TPCONT          :NO-TYPE CHARACTER
21 017314 012700 000015          :MOV      #15,R0          :YES, CHANGE DATA TO CR
22 017320 004767 001276  TPCONT: JSR     PC,TTO          :TYPE IT
23 017324 005216          :INC     (SP)            :MOVE POINTER TO NEXT BYTE
24 017326 000762          :BR      TPOFCH          :FETCH NEXT CHARACTER
25 017330 005726          TPOUTX: TST     (SP)+          :POP STACK TO REACH RETURN VECTOR
26 017332 000207          :RTS     PC
27

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56

.SBTTL KEYBOARD INPUT ROUTINE

```

:KEYBOARD INPUT ROUTINE CALLED BY JSR PC,INPKB
:ENTERED WITH OLD CONTENTS IN KBBUF
:IF JUST CR TYPED, EXIT WITH SAME CONTENTS IN KBBUF
:IF NEW NUMBER TYPED, EXIT WITH NEW CONTENTS IN KBBUF

INPKB: CLR NOKEFL ;CLEAR NO NUMBER FLAG
        MOV R1,-(SP) ;STACK OLD R1
        MOV KBBUF,-(SP) ;STACK 'OLD CONTENTS'
        CLR KBBUF ;CLEAR INPUT BUFFER
GETCHR: JSR PC,KBRD ;FETCH A CHARACTER IN R0
        JSR PC,TTO ;ECHO IT
        CMP R0,#12 ;WAS IT A CR OR LF?
        BNE 1$ ;NO
        JMP NRTRN ;YES, RETURN WITH PROPER KBBUF
1$: MOV R0,R1 ;SET UP TO CHECK FOR A NUMBER
     BIC #177407,R1 ;MASK ALL BUT # CODE
     CMP R1,#60 ;IS IT A # FROM 0-7?
     BEQ 3$ ;YES, PACK IT
     CMP R0,#177 ;WAS IT A DELETE/RUBOUT?
     BNE 2$ ;NO, MUST BE GARBAGE
     MOV #57,R0 ;YES, BUT PRINT '^'
     JSR PC,TTO
     CLC ;CLEAR THE C-BIT
     ROR KBBUF ;DELETE LAST DIGIT
     CLC
     ROR KBBUF ; THAT WAS STUFFED
     CLC
     ROR KBBUF ; INTO KBBUF
     TST KBBUF ;HAVE WE DELETED EVERYTHING?
     BNE 11$ ;NO
     CLR NOKEFL ;YES, BACK TO NO NUMBER INPUT
     JMP GETCHR ;GO FOR MORE INPUT
2$: MOV #207,R0 ;ECHO 'BELL' FOR ERRONEOUS INPUT
     JSR PC,TTO
     MOV #77,R0 ;THEN '?'
     JSR PC,TTO
     JMP GETCHR ;AND GET ANOTHER CHARACTER
000516 3$: MOV #-1,NOKEFL ;GOT A DIGIT. SET FLAG
        BIC #177770,R0 ;GET THE DIGIT PART OF THE CHARACTER
        ASL KBBUF ;SHIFT KBBUF BUFFER
        ASL KBBUF ; TO ACCEPT THE
        ASL KBBUF ; NEW DIGIT.
        BIS R0,KBBUF ;ADD THE NEW DIGIT
        JMP GETCHR ;GO FOR MORE INPUT

NRTRN: TST NOKEFL ;WAS THERE NEW DATA?
        BNE NEK ;YES, GO BACK WITH IT
        MOV (SP)+,KBBUF ;NO, RETRIEVE OLD DATA
        MOV (SP)+,R1 ;RESTORE R1
        RTS PC ;AND RETURN

NEK: TST (SP)+ ;DUMP OLD DATA
     MOV (SP)+,R1 ;RESTORE R1
     RTS PC ;AND RETURN
    
```

```

1      :KEYBOARD READ ROUTINE
2      :CHECKS FOR THE FOLLOWING SPECIAL CHARACTERS:
3      :
4      :       CNTRL-C      RESTART PROG.
5      :       CNTRL-F      CHANGE/SHOW FILL COUNT
6      :       CNTRL-A      PRINT PORT CONFIGURATION
7      :                   STATUS IF AVAILABLE.
8      :       CNTRL-O      ENTER 'DE-BUG' ROUTINE
9      :                   UNLESS ALREADY THERE
10     :                   OR UNLESS IN INITIAL DIALOGUE.
11     :
12     : IF IN 'DE-BUG' ROUTINE:
13     :       <CR>         CLOSE LOCATION
14     :       <LF>         CLOSE LOC. AND OPEN NEXT
15     :       '.'          OPEN LOCATION
16     :       'G'         SET START FUNC OR MAINT
17     :       'P'         RETURN AND CONTINUE
18
19 17 017574 105777 010064      KBRD:  TSTB   @KBS           :WAIT FOR INPUT FROM CONSOLE
20 18 017600 100375             BPL    KBRD
21 19 017602 017700 010054      MOV    @KBD,RO      :PUT THE CHAR INTO RO
22 20 017606 105067 007725      KBRET: CLRB   INCF           :CLEAR FLAGS
23 21 017612 105067 007717      CLRB   GOFLG
24 22 017616 105067 007725      CLRB   SLFL
25 23 017622 105067 007704      CLRB   CONFLG
26 24 017626 042700 177600      BIC    #177600,RO  :TRIM PARITY
27 25 017632 020027 000003      CMP    RO,#3       :WAS ^C TYPED?
28 26 017636 001004             BNE    1$          :YES, RESTART PROGRAM
29 27 017640 012706 002000      MOV    #ISP,SP
30 28 017644 000167 163014      JMP    BCONT
31 29 017650 105767 007660      1$:   TSTB   DBFLG           :DON'T ALLOW ^F IN 'DEBUG'
32 30 017654 001060             BNE    4$
33 31 017656 020027 000006      CMP    RO,#6       :WAS CNTRL-F TYPED?
34 32 017662 001055             BNE    4$          :NO
35 33 017664 105767 007655      2$:   TSTB   SAFE           :ALREADY IN FILL CNT RTN?
36 34 017670 001401             BEQ    20$         :NO, SAFE TO ENTER
37 35 017672 000560             BR     KRTN        :YES, IGNORE CHARACTER
38 36 017674             PNTM   FILMSG         :PRINT 'FILL COUNT = '
39 37 017704 112767 177777 007633  MOV    #FILMSG,RO  :PRINT MESSAGE
40 38 017712 016700 000324      JSR    PC,TYP0UT   :POINTED TO BY FILMSG
41 39 017716 004767 000374      MOVB   #-1,SAFE    :SET FLAG
42 40 017722             PNTM   TWOSP          :SHOW FILL COUNT
43 41 017722 012700 021571      MOV    #TWOSP,RO   :SPACE AND PROMPT
44 42 017726 004767 177340      JSR    PC,TYP0UT   :PRINT MESSAGE
45 43 017732 016767 000304 000300  MOV    FILL,KBUF   :POINTED TO BY TWOSP
46 44 017740 004767 177370      JSR    PC,INPKB    :GET NEW COUNT
47 45 017744 026727 000270 000500  CMP    KBUF,#500   :TOO BIG?
48 46 017752 101406             BLOS   3$          :NO GOOD, GET IT AGAIN
49 47 017754             PNTM   AGAIN          :PRINT MESSAGE
50 48 017754 012700 021324      MOV    #AGAIN,RO   :POINTED TO BY AGAIN
51 49 017760 004767 177306      JSR    PC,TYP0UT
52 50 017764 000167 177704      JMP    20$
53 51 017770 016767 000244 000244  3$:   MOV    KBUF,FILL   :LOAD NEW COUNT
54 52 017776 005767 007604      30$:  TST    PCUNT       :RETURNING TO A PROMPT?
55 53 020002 001514             BEQ    KRTN        :NO, CLEAN RETURN
56 54 020004 062706 000006      ADD    #6,SP       :YES, BREAK UP WITH INPKB
57 55 020010 016716 007572      MOV    PCONT,(SP)  :MODIFY RETURN ADDR

```

```

52 020014 000507
53 020016 020027 000001      4S:  BR      KRTN
54 020022 001012              CMP     RO,#1
55 020024 105767 007515      BNE    40$
56 020030 001101              TSTB   SAFE
57 020032 004767 000230      BNE    KRTN
58 020036 105767 007472      JSR    PC,PNTCNF
59 020042 001074              TSTB   DBFLG
60 020044 000167 177726      BNE    KRTN
61 020050 105767 007460      JMP    30$
62 020054 001021              40S:  TSTB   DBFLG
63 020056 020027 000017      BNE    6$
64 020062 001064              CMP     RO,#17
65 020064 005767 007516      BNE    KRTN
66 020070 001404              TST    PCONT
67 020072 026727 007510 002372 BEQ    5$
68 020100 103407              CMP     PCONT,#PRMT4
69 020102 105767 007437      5S:  BLO    6$
70 020106 001052              TSTB   SAFE
71 020110 062706 000006      BNE    KRTN
72 020114 000167 175272      ADD    #6,SP
73 020120 020027 000012      JMP    ODTPR
74 020124 001007              6S:  CMP     RO,#12
75 020126 012700 000015      BNE    7$
76 020132 112767 177777 007377 MOV    #15,RO
77 020140 000167 000070      MOVB   #-1,INCF
78 020144 020027 000057      JMP    KRTN
79 020150 001007              7S:  CMP     RO,#/
80 020152 112767 177777 007367 BNE    8$
81 020160 004767 000436      MOVB   #-1,SLFL
82 020164 012716 017550      JSR    PC,ITO
83 020170 020027 000107      MOV    #NRTRN,(SP)
84 020174 001007              8S:  CMP     RO,#'G'
85 020176 112767 177777 007331 BNE    9$
86 020204 004767 000412      MOVB   #-1,GOFLG
87 020210 012716 017550      JSR    PC,ITO
88 020214 020027 000120      MOV    #NRTRN,(SP)
89 020220 001005              9S:  CMP     RO,#'P'
90 020222 112767 177777 007302 BNE    KRTN
91 020230 012716 017550      MOVB   #-1,CONFLG
92 020234 000207      MOV    #NRTRN,(SP)
93
94
95
96
97 020236 000000      KRTN:  RTS    PC
98 020240 000000
99 020242 000010
100 020244 000000
101 020246 046 106 111
    020251 114 114 040
    020254 103 117 125
    020257 116 124 040
    020262 075 040 100
102

```

```

: ^A TYPED?
: NO, SKIP
: IN FILL COUNT ROUTINE?
: IF YES, JUST EXIT.
: YES, GO PRINT TABLES
: IN DEBUG??
: IF YES, NORMAL RETURN
: RE-PRINT ANY PROMPTS
: IS 'DEBUG' FLAG SET?
: YES, CHECK FOR OTHER FLAGS
: CNTRL-O TYPED?
: NO, EXIT
: TEST IF ODT IS ALLOWED
: YES, IF PCONT = 0 OR
: IF GREATER THAN PRIORITY
: PROMPT.
: ALSO, NOT WHILE IN FILL
: COUNT ROUTINE.
: BREAK ALL TIES WITH 'INPKB'
: A-O.K. GO TO DEBUG-AID
: CHECK FOR L.F.
: NOT A LINE FEED.
: CHANGE L.F. TO C.R.
: IT WAS, SET INCR FLAG
: AND LEAVE
: WAS IT A SLASH?
: NO.
: YES, SET SLASH FLAG
: ECHO THE SLASH
: AND EXIT.
: WAS 'G' TYPED?
: NO.
: YES, SET 'GO' FLAG
: ECHO THE 'G'
: WAS 'P' TYPED?
: NO, EXIT
: YES, SET 'CONTINUE' FLAG

```

```

: ASSOCIATED VARIABLE STORAGE:
NOKEFL: .WORD 0
KBBUF: .WORD 0
FILL: .WORD 10
FLCNT: .WORD 0
FILMSG: .ASCII /BFILL COUNT = 2/

```

.EVEN

```
1  
2  
3  
4  
5 020266 105767 007241 PNTCNF: TSTB CNFGFL ;CHECK IF CONFIG IS AVAILABLE  
6 020272 001404 BEQ PNTCO ;IF NOT, SAY SO.  
7 020274 004767 174074 JSR PC,REPORT ;IF SO, DUMP IT.  
8 020300 000167 000010 JMP PNTCX ;AND LEAVE  
9 020304 PNTCO: PNTM NOCNF ;PRINT '** STATUS NOT AVAILABLE **'  
020304 012700 021605 MOV #NOCNF,RO ;PRINT MESSAGE  
020310 004767 176756 JSR PC,TYPOUT ;POINTED TO BY NOCNF  
10 020314 000207 PNTCX: RTS PC ;RETURN  
11
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48

.SBTTL BINARY TO ASCII CONVERSION ROUTINES

:CONVERTS BINARY TO BINARY, BINARY TO
 :OCTAL, AND BINARY TO DECIMAL; EITHER
 :UNJUSTIFIED WITH LEADING ZERO'S SUPPRESSED
 :OR RIGHT JUSTIFIED WITH LEADING 0'S
 :SUPPRESSED

:REGULAR BIN-OCTAL UNJUSTIFIED:

OCTPNT: REGSAV.
 JSR R5,REGSAV
 CLR RJFLG ;CLEAR RIGHT JUSTIFY FLAG
 MOV #10,R1 ;SET RADIX FOR OCTAL
 JSR PC,NUMPNT ;CONVERT & PRINT
 REGRES
 JSR R5,REGRES
 RTS PC ;RETURN

:BIN-OCTAL JUSTIFIED:

OCTJSP: REGSAV
 JSR R5,REGSAV
 MOV #-1,RJFLG ;SET RIGHT JUSTIFY FLAG
 MOV #10,R1 ;SET RADIX FOR OCTAL
 JSR PC,NUMPNT ;CONVERT & PRINT
 REGRES
 JSR R5,REGRES
 RTS PC

:BIN-BIN

BINPNT: REGSAV
 JSR R5,REGSAV
 CLR RJFLG ;CLEAR RIGHT JUSTIFY FLAG
 MOV #2,R1 ;SET RADIX FOR BINARY
 JSR PC,NUMPNT ;CONVERT & PRINT
 REGRES
 JSR R5,REGRES
 RTS PC

:BIN-DECIMAL UNJUSTIFIED:

DECPNT: REGSAV
 JSR R5,REGSAV
 CLR RJFLG ;CLEAR RIGHT JUSTIFY FLAG
 MOV #12,R1 ;SET RADIX FOR DECIMAL
 JSR PC,NUMPNT ;CONVERT & PRINT
 REGRES
 JSR R5,REGRES
 RTS PC

:BIN-DECIMAL JUSTIFIED (6 PLACES)

DECJSP: REGSAV
 JSR R5,REGSAV

020316 004567 175546
 020316 005067 000360
 020322 012701 000010
 020326 004767 000156
 020332 004567 175542
 020336 000207
 020342
 020344 004567 175520
 020344 012767 177777 000330
 020350 012701 000010
 020356 004767 000126
 020362 004567 175512
 020366 000207
 020372
 020374 004567 175470
 020374 005067 000302
 020400 012701 000002
 020404 004767 000100
 020410 004567 175464
 020414 000207
 020420
 020422 004567 175442
 020422 005067 000254
 020426 012701 000012
 020432 004767 000052
 020436 004567 175436
 020442 000207
 020446
 020450
 020450 004567 175414

49	020454	012767	177777	000224	MOV	#-1,RJFLG	:SET RIGHT JUSTIFY FLAG
50	020462	012701	000012		MOV	#12,R1	:SET RADIX FOR DECIMAL
51	020466	004767	000022		JSR	PC,NUMPNT	:CONVERT & PRINT
52	020472				REGRES		
53	020472	004567	175406		JSR	R5,REGRES	
54	020476	000207			RTS	PC	
55							
56	020500	005000			NULLS: CLR	R0	
57	020502	004767	000114		JSR	PC,TTO	
58	020506	004767	000110		JSR	PC,TTO	
59	020512	000207			NULOUT: RTS	PC	

```

1      :UNSIGNED CONVERT-PRINT ROUTINE (BIN - ASCII)
2
3 020514 010167 000170  NUMPNT: MOV    R1,RADIX      ;SAVE RADIX
4 020520 005002          CLR    R2          ;CLEAR TAB COUNTER
5 020522 005001          DIVSET: CLR   R1          ;CLEAR WORK REGISTER
6 020524 020067 000160  DIDIV:  CMP    R0,RADIX      ;IS NUMBER BELOW RADIX?
7 020530 103404          BLO    GETDG          ;IF YES, STORE DIGIT
8 020532 166700 000152  SUB    RADIX,R0      ;ELSE, KEEP SUBTRACTING
9 020536 005201          INC    R1          ;AND KEEP COUNT
10 020540 000771          BR     DIVID
11 020542 010046          GETDG: MOV   R0,-(SP)      ;STACK REMAINDER
12 020544 010100          MOV   R1,R0
13 020546 001403          BEQ   PNTEXT        ;PRINT IF HIGHEST ORDER STACKED
14 020550 005202          INC   R2          ;ELSE COUNT DIGITS FOR R. JUSTIFY
15 020552 004767 177744  JSR   PC,DIVSET
16
17 020556 012703 000006  PNTEXT: MOV   #6,R3      ;GET DIGIT COUNT CONSTANT
18 020562 160203          SUB   R2,R3        ;HAVE WE PRODUCED 6 DIGITS?
19 020564 003413          BLE   PNT          ;YES, JUSTIFICATION UNNECESSARY
20 020566 005767 000114  TST   RJFLG        ;IS THE JUSTIFY FLAG SET?
21 020572 001410          BEQ   PNT          ;NO-DON'T JUSTIFY
22 020574 012700 000040  JUST:  MOV   #40,R0     ;YES, PRINT LEADING SPACES
23 020600 004767 000016  JSR   PC,TTO
24 020604 005303          DEC   R3
25 020606 001372          BNE   JUST
26 020610 005067 000072  CLR   RJFLG        ;CLEAR JUSTIFY FLG WHEN DONE
27 020614 012600          PNT:  MOV   (SP)+,R0    ;GET REST OF DIGITS OFF STACK
28 020616 052700 000060  BIS   #'0,R0       ;MAKE THEM ASCII
29
30      :TYPE OUT ROUTINE
31      :PRINTS A CHARACTER WHICH IS IN R0
32      :IF THE CHARACTER IS 'CR.', ALSO PRINT A 'LF.'.
33
34 020622 010077 007042  TTO:   MOV   R0,@TTB    ;PRINT CONTENTS OF R0
35 020626 105777 007040  TTOLP: TSTB  @TTS      ;WAIT TILL PRINT DONE
36 020632 100375          BPL   TTOLP
37 020634 022700 000015  CMP   #15,R0       ;WAS IT A CR?
38 020640 001401          BEQ   TTOLF        ;YES, ECHO A LF AS WELL
39 020642 000207          RTS   PC          ;NO, JUST RETURN
40 020644 012700 000012  TTOLF: MOV   #12,R0
41 020650 004767 177746  JSR   PC,TTO       ;ECHO A LINE FEED
42 020654 016767 177362 177362  MOV   FILL,FLCNT    ;PRINT NULLS FOR FILL
43 020662 012700 000000  MOV   #0,R0
44 020666 004767 177730  18:   JSR   PC,TTO
45 020672 005367 177346  DEC   FLCNT
46 020676 003373          BGT   18
47 020700 012700 000012  MOV   #12,R0      ;THEN RESTORE R0
48 020704 000207          RTS   PC          ;AND LEAVE
49
50
51      :ASSOCIATED VARIABLE STORAGE:
52
53 020706 000000          RJFLG: .WORD 0
54 020710 000000          RADIX: .WORD 0

```


1
2
3
4
5
6
7
8
9

.SBTTL CONSTANTS, VARIABLES & ACSII

:ASCII MESSAGES:

020712	046	103	132
020715	103	115	112
020720	102	060	040
020723	103	115	122
020726	050	126	051
020731	055	061	061
020734	040	110	117
020737	123	124	040
020742	105	130	105
020745	122	103	111
020750	123	105	122
020753	040	040	040
020756	062	070	055
020761	116	117	126
020764	055	070	060
020767	046	046	040
020772	040	040	040
020775	040	040	052
021000	052	052	040
021003	103	101	125
021006	124	111	117
021011	116	040	052
021014	052	052	
021016	046	124	110
021021	111	123	040
021024	120	122	117
021027	107	122	101
021032	115	040	111
021035	123	040	103
021040	101	120	101
021043	102	114	105
021046	040	117	106
021051	040	101	114
021054	124	105	122
021057	111	116	107
021062	040	124	110
021065	105	040	123
021070	124	101	124
021073	105		
021074	046	117	106
021077	040	101	116
021102	131	040	106
021105	111	105	114
021110	104	040	105
021113	121	125	111
021116	120	115	105
021121	116	124	040
021124	101	124	124
021127	101	103	110
021132	105	104	040
021135	124	117	040
021140	101	116	131

TSTHDR: .ASCII /&CZCMJBO CMR(V)-11 HOST EXERCISER 28-NOV-80/

.ASCII /&& *** CAUTION ***/

.ASCII /&THIS PROGRAM IS CAPABLE OF ALTERING THE STATE/

.ASCII /&OF ANY FIELD EQUIPMENT ATTACHED TO ANY OF THE/

	021143	040	117	106	
	021146	040	124	110	
	021151	105			
10	021152	046	122	105	.ASCII /&REMOTES ON THIS CMR-11./
	021155	115	117	124	
	021160	105	123	040	
	021163	117	116	040	
	021166	124	110	111	
	021171	123	040	103	
	021174	115	122	055	
	021177	061	061	056	
11	021202	046	046	120	.ASCII /&&PROCEED WITH CAUTION IF FIELD HARDWARE IS STILL ON-LINE!&&/
	021205	122	117	103	
	021210	105	105	104	
	021213	040	127	111	
	021216	124	110	040	
	021221	103	101	125	
	021224	124	111	117	
	021227	116	040	111	
	021232	106	040	106	
	021235	111	105	114	
	021240	104	040	110	
	021243	101	122	104	
	021246	127	101	122	
	021251	105	040	111	
	021254	123	040	123	
	021257	124	111	114	
	021262	114	040	117	
	021265	116	055	114	
	021270	111	116	105	
	021273	041	046	046	
	021276	100			
12	021277	046	111	116	FRAD: .ASCII /&INTERFACE ADDRESS: &/
	021302	124	105	122	
	021305	106	101	103	
	021310	105	040	101	
	021313	104	104	122	
	021316	105	123	123	
	021321	072	040	100	
13	021324	046	124	110	AGAIN: .ASCII /&THAT WON'T DO. TRY AGAIN&/
	021327	101	124	040	
	021332	127	117	116	
	021335	047	124	040	
	021340	104	117	056	
	021343	040	124	122	
	021346	131	040	101	
	021351	107	101	111	
	021354	116	046	100	
14	021357	046	111	116	DVECT: .ASCII /&INTERFACE VECTOR: &/
	021362	124	105	122	
	021365	106	101	103	
	021370	105	040	126	
	021373	105	103	124	
	021376	117	122	072	
	021401	040	100		
15	021403	046	111	115	DVPRI0: .ASCII /&INTERFACE PRIORITY (4 TO 7): &/
	021406	124	105	127	

	021411	106	101	103	
	021414	105	040	120	
	021417	122	111	117	
	021422	122	111	124	
	021425	131	040	050	
	021430	064	040	124	
	021433	117	040	067	
	021436	051	072	040	
16	021441	100			
	021442	046	103	115	DBAID: .ASCII /&CMR-11 DE-BUG AID&&&/
	021445	122	055	061	
	021450	061	040	104	
	021453	105	055	102	
	021456	125	107	040	
	021461	101	111	104	
17	021464	046	046	100	
	021467	046	111	116	INVLAD: .ASCII /&INVALID ADDRESS...(IT'S NOT THERE)&/
	021472	126	101	114	
	021475	111	104	040	
	021500	101	104	104	
	021503	122	105	123	
	021506	123	056	056	
	021511	056	050	111	
	021514	124	047	123	
	021517	040	116	117	
	021522	124	040	124	
	021525	110	105	122	
18	021530	105	051	100	
	021533	046	046	124	TRP4: .ASCII /&&TRAPPED TO 4 FROM LOCATION &/
	021536	122	101	120	
	021541	120	105	104	
	021544	040	124	117	
	021547	040	064	040	
	021552	106	122	117	
	021555	115	040	114	
	021560	117	103	101	
	021563	124	111	117	
19	021566	116	040	100	
	021571	040	040	072	TWOSP: .ASCII / :&/
20	021574	100			
	021575	046	123	127	SURMSG: .ASCII /&SUR = &/
	021600	122	040	075	
21	021603	040	100		
	021605	046	052	052	NOCNF: .ASCII /&** CONFIGURATION NOT AVAILABLE **&/
	021610	040	103	117	
	021613	116	106	111	
	021616	107	125	122	
	021621	101	124	111	
	021624	117	116	040	
	021627	116	117	124	
	021632	040	101	126	
	021635	101	111	114	
	021640	101	102	114	
	021643	105	040	052	
22	021646	052	100		
	021650	046	124	131	TYPCTP: .ASCII /&TYPE 'P' TO CONTINUE&/
	021653	120	105	040	

	021656	042	120	042	
	021661	040	124	117	
	021664	040	103	117	
	021667	116	124	111	
	021672	116	125	105	
	021675	100			
23	021676	046	123	105	TSTSEL: .ASCII /BSELECT TEST (4=HELP) a/
	021701	114	105	103	
	021704	124	040	124	
	021707	105	123	124	
	021712	040	050	064	
	021715	075	110	105	
	021720	114	120	051	
	021723	040	100		
24	021725	046	123	105	HELPM: .ASCII /BSELECT ONE OF THE FOLLOWING:8/
	021730	114	105	103	
	021733	124	040	117	
	021736	116	105	040	
	021741	117	106	040	
	021744	124	110	105	
	021747	040	106	117	
	021752	114	114	117	
	021755	127	111	116	
	021760	107	072	046	
25	021763	046	103	122	.ASCII /BCR OR 1 = LOGIC TEST/
	021766	040	117	122	
	021771	040	061	040	
	021774	075	040	114	
	021777	117	107	111	
	022002	103	040	124	
	022005	105	123	124	
26	022010	046	040	040	.ASCII /B 2 = PORT LOOP-BACK TEST/
	022013	040	040	040	
	022016	040	062	040	
	022021	075	040	120	
	022024	117	122	124	
	022027	040	114	117	
	022032	117	120	055	
	022035	102	101	103	
	022040	113	040	124	
	022043	105	123	124	
27	022046	046	040	040	.ASCII /B 3 = CMR-11 EXERCISER/
	022051	040	040	040	
	022054	040	063	040	
	022057	075	040	103	
	022062	115	122	055	
	022065	061	061	040	
	022070	105	130	105	
	022073	122	103	111	
	022076	123	105	122	
28	022101	046	040	040	.ASCII /B 4 = THIS HELP INFO./
	022104	040	040	040	
	022107	040	064	040	
	022112	075	040	124	
	022115	110	111	123	
	022120	040	110	105	
	022123	114	120	040	

	022126	111	116	106	
	022131	117	056		
29	022133	046	103	116	.ASCII /&CNTRL-C = TEST SELECT/
	022136	124	122	114	
	022141	055	103	040	
	022144	075	040	124	
	022147	105	123	124	
	022152	040	123	105	
	022155	114	105	103	
	022160	124			
30	022161	046	103	116	.ASCII /&CNTRL-G = SWITCH OPTION SELECT/
	022164	124	122	114	
	022167	055	107	040	
	022172	075	040	123	
	022175	127	111	124	
	022200	103	110	040	
	022203	117	120	124	
	022206	111	117	116	
	022211	040	123	105	
	022214	114	105	103	
	022217	124			
31	022220	046	103	116	.ASCII /&CNTRL-F = CONSOLE FILL COUNT SELECT/
	022223	124	122	114	
	022226	055	106	040	
	022231	075	040	103	
	022234	117	116	123	
	022237	117	114	105	
	022242	040	106	111	
	022245	114	114	040	
	022250	103	117	125	
	022253	116	124	040	
	022256	123	105	114	
	022261	105	103	124	
32	022264	046	103	116	.ASCII /&CNTRL-A = SHOW PORT CONFIG. STATUS/
	022267	124	122	114	
	022272	055	101	040	
	022275	075	040	123	
	022300	110	117	127	
	022303	040	120	117	
	022306	122	124	040	
	022311	103	117	116	
	022314	106	111	107	
	022317	056	040	123	
	022322	124	101	124	
	022325	125	123		
33	022327	046	040	040	.ASCII /& (ONLY WORKS IF TABLE ALREADY/
	022332	040	040	040	
	022335	040	040	040	
	022340	040	040	050	
	022343	117	116	114	
	022346	131	040	127	
	022351	117	122	113	
	022354	123	040	111	
	022357	105	040	124	
	022362	101	102	114	
	022365	105	040	101	
	022370	114	122	105	

	022373	101	104	131		
34	022376	046	040	040	.ASCII /&	PRINTED AUTOMATICALLY)./
	022401	040	040	040		
	022404	040	040	040		
	022407	040	040	120		
	022412	122	111	116		
	022415	124	105	104		
	022420	040	101	125		
	022423	124	117	115		
	022426	101	124	111		
	022431	103	101	114		
	022434	114	131	051		
	022437	056				
35	022440	046	103	116	.ASCII /&CNTRL-O = DEBUG ROUTINE WHICH ALLOWS/	
	022443	124	122	114		
	022446	055	117	040		
	022451	075	040	104		
	022454	105	102	125		
	022457	107	040	122		
	022462	117	125	124		
	022465	111	116	105		
	022470	040	127	110		
	022473	111	103	110		
	022476	040	101	114		
	022501	114	117	127		
	022504	123				
36	022505	046	040	040	.ASCII /&	LOADING SCR. PAD AND STARTING/
	022510	040	040	040		
	022513	040	040	040		
	022516	040	040	114		
	022521	117	101	104		
	022524	111	116	107		
	022527	040	123	103		
	022532	122	056	040		
	022535	120	101	104		
	022540	040	101	116		
	022543	104	040	123		
	022546	124	101	122		
	022551	124	111	116		
	022554	107				
37	022555	046	040	040	.ASCII /&	FUNCTIONS AND EXAMINING THE/
	022560	040	040	040		
	022563	040	040	040		
	022566	040	040	106		
	022571	125	116	103		
	022574	124	111	117		
	022577	116	123	040		
	022602	101	116	104		
	022605	040	105	130		
	022610	101	115	111		
	022613	116	111	116		
	022616	107	040	124		
	022621	110	105			
38	022623	046	040	040	.ASCII /&	RESULTS/
	022626	040	040	040		
	022631	040	040	040		
	022634	040	040	122		

	022637	105	123	125	
	022642	114	124	123	
39	022645	046	040	040	.ASCII /& 'P' = PROCEED FROM SW. MONITOR @/
	022650	040	040	042	
	022653	120	042	040	
	022656	075	040	120	
	022661	122	117	103	
	022664	105	105	104	
	022667	040	106	122	
	022672	117	115	040	
	022675	123	127	056	
	022700	040	115	117	
	022703	116	111	124	
	022706	117	122	040	
	022711	040	040	100	
40	022714	046	111	106	HLPDSR: .ASCII /&IF HELP IS REQUIRED, ENTER A '4' @/
	022717	040	110	105	
	022722	114	120	040	
	022725	111	123	040	
	022730	122	105	121	
	022733	125	111	122	
	022736	105	104	054	
	022741	040	105	116	
	022744	124	105	122	
	022747	040	101	040	
	022752	042	064	042	
	022755	040	040	040	
	022760	100			
41	022761	046	114	117	LGKTSM: .ASCII /&LOGIC TEST@/
	022764	107	111	103	
	022767	040	124	105	
	022772	123	124	046	
	022775	100			
42	022776	046	123	103	SCPAD: .ASCII /&SCR.PAD ADDR = @/
	023001	122	056	120	
	023004	101	104	040	
	023007	101	104	104	
	023012	122	040	075	
	023015	040	100		
43	023017	046	046	110	VERSM: .ASCII /&HOST FIRMWARE VERSION : @/
	023022	117	123	124	
	023025	040	106	111	
	023030	122	115	127	
	023033	101	122	105	
	023036	040	126	105	
	023041	122	123	111	
	023044	117	116	040	
	023047	072	040	100	
44	023052	046	046	102	BAUDM: .ASCII /&BAUD RATES:@/
	023055	101	125	104	
	023060	040	122	101	
	023063	124	105	123	
	023066	072	100		
45	023070	046	120	117	PORTOM: .ASCII /&PORT #0 = @/
	023073	122	124	040	
	023076	043	060	040	
	023101	075	040	100	

46	023104	046	120	117	PORT1M: .ASCII /&PORT #1 = @/
	023107	122	124	040	
	023112	043	061	040	
	023115	075	040	100	
47	023120	046	120	117	PORT2M: .ASCII /&PORT #2 = @/
	023123	122	124	040	
	023126	043	062	040	
	023131	075	040	100	
48	023134	046	120	117	PORT3M: .ASCII /&PORT #3 = @/
	023137	122	124	040	
	023142	043	063	040	
	023145	075	040	100	
49	023150	046	105	116	ENDPS: .ASCII /&END PASS #@/
	023153	104	040	120	
	023156	101	123	123	
	023161	040	043	100	
50	023164	040	105	114	ELTIM: .ASCII / EL. TIM.=@/
	023167	056	040	124	
	023172	111	115	056	
	023175	075	100		
51	023177	040	040	105	ERCTM: .ASCII / ERR. CNT = @/
	023202	122	122	056	
	023205	040	103	116	
	023210	124	040	075	
	023213	040	100		
52	023215	052	052	117	OVFLM: .ASCII /**OVERFLOW**@/
	023220	126	105	122	
	023223	106	114	117	
	023226	127	052	052	
	023231	100			
53	023232	046	120	117	PLTM: .ASCII /&PORT LOOP-BACK TEST#@/
	023235	122	124	040	
	023240	114	117	117	
	023243	120	055	102	
	023246	101	103	113	
	023251	040	124	105	
	023254	123	124	046	
	023257	100			
54	023260	046	124	131	PORTN: .ASCII /&TYPE PORT # (<CR>=ALL): @/
	023263	120	105	040	
	023266	120	117	122	
	023271	124	040	043	
	023274	040	050	074	
	023277	103	122	076	
	023302	075	101	114	
	023305	114	051	072	
	023310	040	100		
55	023312	046	124	131	TYP&CR: .ASCII /&TYPE <CR> WHEN READY: @/
	023315	120	105	040	
	023320	074	103	122	
	023323	076	040	127	
	023326	110	105	116	
	023331	040	122	105	
	023334	101	104	131	
	023337	072	040	100	
56	023342	046	103	115	EXCM: .ASCII /&CMR EXERCISE ROUTINE@/
	023345	122	040	105	

	023350	130	105	122	
	023353	103	111	123	
	023356	105	040	122	
	023361	117	125	124	
	023364	111	116	105	
	023367	046			
57	023370	046	050	107	.ASCII /&(GIVE ME A FEW MINUTES ALONE WITH THE REMOTES)@/
	023373	111	126	105	
	023376	040	115	105	
	023401	040	101	040	
	023404	106	105	127	
	023407	040	115	111	
	023412	116	125	124	
	023415	105	123	040	
	023420	101	114	117	
	023423	116	105	040	
	023426	127	111	124	
	023431	110	040	124	
	023434	110	105	040	
	023437	122	105	115	
	023442	117	124	105	
	023445	123	051	100	
58	023450	046	052	052	NOREM: .ASCII /&*** NO REMOTES ON HOST ***&/
	023453	052	040	116	
	023456	117	040	122	
	023461	105	115	117	
	023464	124	105	123	
	023467	040	117	116	
	023472	040	110	117	
	023475	123	124	040	
	023500	052	052	052	
	023503	046			
59	023504	046	050	040	.ASCII /&(I WILL KEEP LOOKING FOR ANY CHANGES)@/
	023507	111	040	127	
	023512	111	114	114	
	023515	040	113	105	
	023520	105	120	040	
	023523	114	117	117	
	023526	113	111	116	
	023531	107	040	106	
	023534	117	122	040	
	023537	101	116	131	
	023542	040	103	110	
	023545	101	116	107	
	023550	105	123	040	
	023553	051	046	100	
60	023556	046	120	117	PORTMS: .ASCII /&PORT #@/
	023561	122	124	040	
	023564	043	100		
61	023566	046	122	105	REMSG: .ASCII /&REMOTE #@/
	023571	115	117	124	
	023574	105	040	043	
	023577	100			
62	023600	046	040	052	CONFMS: .ASCII /& ** PORT CONFIGURATION **@/
	023603	052	040	120	
	023606	117	122	124	
	023611	040	103	117	

	023614	116	106	111	
	023617	107	125	122	
	023622	101	124	111	
	023625	117	116	040	
	023630	052	052	046	
	023633	100			
63	023634	046	123	124	STATEM: .ASCII /&STATE = @/
	023637	101	124	105	
	023642	040	075	040	
	023645	100			
64	023646	046	125	123	USORM: .ASCII /&USOR = @/
	023651	117	122	040	
	023654	075	040	100	
65	023657	046	052	052	REMCHG: .ASCII /&*** REMOTE STATUS CHANGE ***@/
	023662	052	040	122	
	023665	105	115	117	
	023670	124	105	040	
	023673	123	124	101	
	023676	124	125	123	
	023701	040	103	110	
	023704	101	116	107	
	023707	105	040	052	
	023712	052	052	100	
66	023715	046	122	105	REMSTM: .ASCII /&REMOTE STATUS NOW @/
	023720	115	117	124	
	023723	105	040	123	
	023726	124	101	124	
	023731	125	123	040	
	023734	116	117	127	
	023737	040	100		
67	023741	040	122	105	REGMG: .ASCII / REGULAR@/
	023744	107	125	114	
	023747	101	122	100	
68	023752	040	111	122	IRRGMG: .ASCII / IRREGULAR@/
	023755	122	105	107	
	023760	125	114	101	
	023763	122	100		
69	023765	046	052	052	ALROFF: .ASCII /&*** ALL REMOTES ARE IRREGULAR ***@/
	023770	052	040	101	
	023773	114	114	040	
	023776	122	105	115	
	024001	117	124	105	
	024004	123	040	101	
	024007	122	105	040	
	024012	111	122	122	
	024015	105	107	125	
	024020	114	101	122	
	024023	040	052	052	
	024026	052	046	100	

```

1          ;ERROR MESSAGES
2
3 024031    040    110    117  NOGO:  .ASCII / HOST MICRO FAILED TO START.@/
024034    123    124    040
024037    115    111    103
024042    122    117    040
024045    106    101    111
024050    114    105    104
024053    040    124    117
024056    040    123    124
024061    101    122    124
024064    056    100
4 024066    040    120    127  NOTUP:  .ASCII / PWR-UP SEQUENCE NOT COMPLETE.(BAD CSR).@/
024071    122    055    125
024074    120    040    123
024077    105    121    125
024102    105    116    103
024105    105    040    116
024110    117    124    040
024113    103    117    115
024116    120    114    105
024121    124    105    056
024124    050    102    101
024127    104    040    103
024132    123    122    051
024135    056    100
5 024137    040    103    123  CSRRST:  .ASCII / CSR REG. NOT RESET.@/
024142    122    040    122
024145    105    107    056
024150    040    116    117
024153    124    040    122
024156    105    123    105
024161    124    056    100
6 024164    040    103    101  SPRCLR:  .ASCII / CAN'T CLR SPAR REG.@/
024167    116    047    124
024172    040    103    114
024175    122    040    123
024200    120    101    122
024203    040    122    105
024206    107    056    100
7 024211    040    102    101  SPRDAT:  .ASCII / BAD DATA IN SPAR REG.@/
024214    104    040    104
024217    101    124    101
024222    040    111    116
024225    040    123    120
024230    101    122    040
024233    122    105    107
024236    056    100
8 024240    040    122    105  ISATE:  .ASCII / REQ SCPD ADR TST ERROR.@/
024243    121    040    123
024246    103    120    104
024251    040    101    104
024254    122    040    124
024257    123    124    040
024262    105    122    122
024265    117    122    056
024270    100
    
```

9	024271	040	122	105	OSDTE: .ASCII / RESULT SCPD DATA TST ERROR.@/
	024274	123	125	114	
	024277	124	040	123	
	024302	103	120	104	
	024305	040	104	101	
	024310	124	101	040	
	024313	124	123	124	
	024316	040	105	122	
	024321	122	117	122	
	024324	056	100		
10	024326	040	122	105	OSATE: .ASCII / RESULT SCPD ADR TST ERROR.@/
	024331	123	125	114	
	024334	124	040	123	
	024337	103	120	104	
	024342	040	101	104	
	024345	122	040	124	
	024350	123	124	040	
	024353	105	122	122	
	024356	117	122	056	
	024361	100			
11	024362	040	122	105	ISDTE: .ASCII / REQ SCPD DATA TST ERROR.@/
	024365	121	040	123	
	024370	103	120	104	
	024373	040	104	101	
	024376	124	101	040	
	024401	124	123	124	
	024404	040	105	122	
	024407	122	117	122	
	024412	056	100		
12	024414	040	122	105	RQSPCL: .ASCII / REQ. SCPD NOT ALL 0'S.@/
	024417	121	056	040	
	024422	123	103	120	
	024425	104	040	116	
	024430	117	124	040	
	024433	101	114	114	
	024436	040	060	047	
	024441	123	056	100	
13	024444	040	122	105	RSSPCL: .ASCII / RES. SCPD NOT ALL 0'S.@/
	024447	123	056	040	
	024452	123	103	120	
	024455	104	040	116	
	024460	117	124	040	
	024463	101	114	114	
	024466	040	060	047	
	024471	123	056	100	
14	024474	040	042	103	CLTBER: .ASCII / 'CLRTPB' FAILED TO CLR TOP BYTE.@/
	024477	114	122	124	
	024502	120	102	042	
	024505	040	106	101	
	024510	111	114	105	
	024513	104	040	124	
	024516	117	040	103	
	024521	114	122	040	
	024524	124	117	120	
	024527	040	102	131	
	024532	124	105	056	
	024535	100			

15	024536	040	127	122	DIERR: .ASCII / WRONG CSR FOR 'DIE' FUNC.@/
	024541	117	116	107	
	024544	040	103	123	
	024547	122	040	106	
	024552	117	122	040	
	024555	042	104	111	
	024560	105	042	040	
	024563	106	125	116	
	024566	103	056	100	
16	024571	040	102	101	CSRWE: .ASCII / BAD CSR BITS WHILE HUNG.@/
	024574	104	040	103	
	024577	123	122	040	
	024602	102	111	124	
	024605	123	040	127	
	024610	110	111	114	
	024613	105	040	110	
	024616	125	116	107	
	024621	056	100		
17	024623	040	103	101	CSRCE: .ASCII / CAN'T CLR LOW CSR WHILE HUNG.@/
	024626	116	047	124	
	024631	040	103	114	
	024634	122	040	114	
	024637	117	127	040	
	024642	103	123	122	
	024645	040	127	110	
	024650	111	114	105	
	024653	040	110	125	
	024656	116	107	056	
	024661	100			
18	024662	040	101	103	SPAVME: .ASCII / ACCESSED REQ SCPD WITH 'MAIN' SET.@/
	024665	103	105	123	
	024670	123	105	104	
	024673	040	122	105	
	024676	121	040	123	
	024701	103	120	104	
	024704	040	127	111	
	024707	124	110	040	
	024712	042	115	101	
	024715	111	116	042	
	024720	040	123	105	
	024723	124	056	100	
19	024726	040	101	103	SPAVFE: .ASCII / ACCESSED REQ SCPD WITH 'STFUN' SET.@/
	024731	103	105	123	
	024734	123	105	104	
	024737	040	122	105	
	024742	121	040	123	
	024745	103	120	104	
	024750	040	127	111	
	024753	124	110	040	
	024756	042	123	124	
	024761	106	125	116	
	024764	042	040	123	
	024767	105	124	056	
	024772	100			
20	024773	040	110	117	NOINTR: .ASCII / HOST CAN'T INTERRUPT UNIBUS.@/
	024776	123	124	040	
	025001	103	101	116	

	025004	047	124	040	
	025007	111	116	124	
	025012	105	122	122	
	025015	125	120	124	
	025020	040	125	116	
	025023	111	102	125	
	025026	123	056	100	
21	025031	040	105	122	ERINTR: .ASCII / ERRONEOUS HOST INTERRUPT.@/
	025034	122	117	116	
	025037	105	117	125	
	025042	123	040	110	
	025045	117	123	124	
	025050	040	111	116	
	025053	124	105	122	
	025056	122	125	120	
	025061	124	056	100	
22	025064	040	127	122	WPODI: .ASCII / WRONG PRIORITY. DEV. IS HIGHER.@/
	025067	117	116	107	
	025072	040	120	122	
	025075	111	117	122	
	025100	111	124	131	
	025103	056	040	104	
	025106	105	126	056	
	025111	040	111	123	
	025114	040	110	111	
	025117	107	110	105	
	025122	122	056	100	
23	025125	040	120	122	HITER: .ASCII / PROG INIT. FAILED TO INIT CSR.@/
	025130	117	107	040	
	025133	111	116	111	
	025136	124	056	040	
	025141	106	101	111	
	025144	114	105	104	
	025147	040	124	117	
	025152	040	111	116	
	025155	111	124	040	
	025160	103	123	122	
	025163	056	100		
24	025165	040	115	111	VISF: .ASCII / MICRO DIDN'T STORE VERSION NO.@/
	025170	103	122	117	
	025173	040	104	111	
	025176	104	116	047	
	025201	124	040	123	
	025204	124	117	122	
	025207	105	040	126	
	025212	105	122	123	
	025215	111	117	116	
	025220	040	116	117	
	025223	056	100		
25	025225	040	062	116	VIDER: .ASCII / 2ND VERSION INFO DIFFERS FROM PWR-UP.@/
	025230	104	040	126	
	025233	105	122	123	
	025236	111	117	116	
	025241	040	111	116	
	025244	106	117	040	
	025247	104	111	106	
	025252	106	105	122	

	025255	123	040	106	
	025260	122	117	115	
	025263	040	120	127	
	025266	122	055	125	
	025271	120	056	100	
26	025274	040	101	103	SPAVIC: .ASCII / ACCESSED RES. SCPD WITH INTR CLR.@/
	025277	103	105	123	
	025302	123	105	104	
	025305	040	122	105	
	025310	123	056	040	
	025313	123	103	120	
	025316	104	040	127	
	025321	111	124	110	
	025324	040	111	116	
	025327	124	122	040	
	025332	103	114	122	
	025335	056	100		
27	025337	040	127	122	STESTF: .ASCII / WRONG CSR WITH 'STFUN'.@/
	025342	117	116	107	
	025345	040	103	123	
	025350	122	040	127	
	025353	111	124	110	
	025356	040	042	123	
	025361	124	106	125	
	025364	116	042	056	
	025367	100			
28	025370	040	042	115	RSPEF: .ASCII / 'MAIN' RESULT FUNC. CODE WRONG.@/
	025373	101	111	116	
	025376	042	040	122	
	025401	105	123	125	
	025404	114	124	040	
	025407	106	125	116	
	025412	103	056	040	
	025415	103	117	104	
	025420	105	040	127	
	025423	122	117	116	
	025426	107	056	100	
29	025431	040	102	101	USOF1: .ASCII / BAD USOF FROM 'STFUN' ON MAINT.@/
	025434	104	040	125	
	025437	123	117	106	
	025442	040	106	122	
	025445	117	115	040	
	025450	042	123	124	
	025453	106	125	116	
	025456	042	040	117	
	025461	116	040	115	
	025464	101	111	116	
	025467	124	056	100	
30	025472	040	127	122	STEMF: .ASCII / WRONG CSR WITH 'MAINT'.@/
	025475	117	116	107	
	025500	040	103	123	
	025503	122	040	127	
	025506	111	124	110	
	025511	040	042	115	
	025514	101	111	116	
	025517	124	042	056	
	025522	100			

31	025523	040	102	101	USOFE2: .ASCII / BAD USOF FROM 'MAINT' FUNC.@/
	025526	104	040	125	
	025531	123	117	106	
	025534	040	106	122	
	025537	117	115	040	
	025542	042	115	101	
	025545	111	116	124	
	025550	042	040	106	
	025553	125	116	103	
	025556	056	100		
32	025560	040	042	115	DATLER: .ASCII / 'MAIN' FUNC GAVE BAD DATA LEN.@/
	025563	101	111	116	
	025566	042	040	106	
	025571	125	116	103	
	025574	040	107	101	
	025577	126	105	040	
	025602	102	101	104	
	025605	040	104	101	
	025610	124	101	040	
	025613	114	105	116	
	025616	056	100		
33	025620	040	122	105	RSPMF: .ASCII / RES. FUNC CODE WRONG FOR 'MAIN'.@/
	025623	123	056	040	
	025626	106	125	116	
	025631	103	040	103	
	025634	117	104	105	
	025637	040	127	122	
	025642	117	116	107	
	025645	040	106	117	
	025650	122	040	042	
	025653	115	101	111	
	025656	116	042	056	
	025661	100			
34	025662	040	102	101	USOFE3: .ASCII / BAD USOF FROM 'MAIN' ON REG FUNC.@/
	025665	104	040	125	
	025670	123	117	106	
	025673	040	106	122	
	025676	117	115	040	
	025701	042	115	101	
	025704	111	116	042	
	025707	040	117	116	
	025712	040	122	105	
	025715	107	040	106	
	025720	125	116	103	
	025723	056	100		
35	025725	040	127	122	USOFE4: .ASCII / WRONG USOF ON REGULAR FUNC.@/
	025730	117	116	107	
	025733	040	125	123	
	025736	117	106	040	
	025741	117	116	040	
	025744	122	105	107	
	025747	125	114	101	
	025752	122	040	106	
	025755	125	116	103	
	025760	056	100		
36	025762	040	124	117	PLFFE: .ASCII / TOTAL FAILURE ON 'MAIN' FUNCTION.@/
	025765	124	101	114	

	025770	040	106	101	
	025773	111	114	125	
	025776	122	105	040	
	026001	117	116	040	
	026004	042	115	101	
	026007	111	116	042	
	026012	040	106	125	
	026015	116	103	124	
	026020	111	117	116	
37	026023	056	100		
	026025	040	120	117	P.TXTO: .ASCII / PORT XMTR TIMED OUT.@/
	026030	122	124	040	
	026033	130	115	124	
	026036	122	040	124	
	026041	111	115	105	
	026044	104	040	117	
	026047	125	124	056	
38	026052	100			
	026053	040	120	117	PLRCTO: .ASCII / PORT RCVR TIMED OUT.@/
	026056	122	124	040	
	026061	122	103	126	
	026064	122	040	124	
	026067	111	115	105	
	026072	104	040	117	
	026075	125	124	056	
39	026100	100			
	026101	040	120	117	PLDERR: .ASCII / PORT-LOOP DETECTED DATA ERROR.@/
	026104	122	124	055	
	026107	114	117	117	
	026112	120	040	104	
	026115	105	124	105	
	026120	103	124	105	
	026123	104	040	104	
	026126	101	124	101	
	026131	040	105	122	
	026134	122	117	122	
	026137	056	100		
40	026141	040	110	117	SYSER: .ASCII / HOST SYS ERROR. RUN LOGIC TEST.@/
	026144	123	124	040	
	026147	123	131	123	
	026152	040	105	122	
	026155	122	117	122	
	026160	056	040	122	
	026163	125	116	040	
	026166	114	117	107	
	026171	111	103	040	
	026174	124	105	123	
	026177	124	056	100	
41	026202	040	042	104	DIAGF: .ASCII / 'DIAG' FUNC GAVE WRONG FUNC CODE.@/
	026205	111	101	107	
	026210	042	040	106	
	026213	125	116	103	
	026216	040	107	101	
	026221	126	105	040	
	026224	127	122	117	
	026227	116	107	040	
	026232	106	125	116	

	026235	103	040	103	
	026240	117	104	105	
	026243	056	100		
42	026245	040	125	123	DIAGUF: .ASCII / USOF WRONG ON 'DIAG' FUNC.@/
	026250	117	106	040	
	026253	127	122	117	
	026256	116	107	040	
	026261	117	116	040	
	026264	042	104	111	
	026267	101	107	042	
	026272	040	106	125	
	026275	116	103	056	
	026300	100			

43
44

.EVEN

```

1      ;TABLE STORAGE
2
3
4      ;TABLE OF BAUD RATE MESSAGE POINTERS
5
6      026302 026322      BDRTBL: .WORD   BD1
7      026304 026327      .WORD   BD2
8      026306 026334      .WORD   BD3
9      026310 026342      .WORD   BD4
10     026312 026350      .WORD   BD5
11     026314 026356      .WORD   BD6
12     026316 026364      .WORD   BD7
13     026320 026402      .WORD   BD8
14
15     ;TABLE OF ASCII STORAGE FOR BAUD RATES
16
17     026322      040      063      060      BD1:   .ASCII / 300a/
18     026325      060      100
19     026327      040      066      060      BD2:   .ASCII / 600a/
20     026332      060      100
21     026334      040      061      062      BD3:   .ASCII / 1200a/
22     026337      060      060      100
23     026342      040      062      064      BD4:   .ASCII / 2400a/
24     026345      060      060      100
25     026350      040      064      070      BD5:   .ASCII / 4800a/
26     026353      060      060      100
27     026356      040      071      066      BD6:   .ASCII / 9600a/
28     026361      060      060      100
29     026364      040      042      110      BD7:   .ASCII / 'HIGH' SPEEDa/
30     026367      111      107      110
31     026372      042      040      123
32     026375      120      105      105
33     026400      104      100
34     026402      040      042      123      BD8:   .ASCII / 'SPECIAL' SPEEDa/
35     026405      120      105      103
36     026410      111      101      114
37     026413      042      040      123
38     026416      120      105      105
39     026421      104      100
40
41     .EVEN
42
43     ;DATA TABLE FOR MAINTENANCE AND DIAGNOSTIC FUNCTION
44
45     DATBL: .WORD   001
46     .WORD   003
47     .WORD   007
48     .WORD   017
49     .WORD   037
50     .WORD   077
51     .WORD   177
52     .WORD   377
53     .WORD   376
54     .WORD   374
55     .WORD   370
56     .WORD   360
57     .WORD   340
58     .WORD   300

```

43	026460	000200	.WORD	200
44	026462	000000	.WORD	000
45				

```

1          ;PORT CONFIGURATION STATUS TABLES.
2
3 026464   EXTBL1: .BLKW 64.
4 026664   EXTBL2: .BLKW 64.
5 027064   EXTBL3: .BLKW 64.
6 027264   EXTBL4: .BLKW 64.
7
8          ;TABLE FOR ORIGINAL (PWR-UP) VERSION DATA
9
10 027464  VERSP: .BLKW 9.
11
12          ;TABLE FOR 2ND VERSION DATA
13
14 027506  VERSUN: .BLKW 9.
15
16          ;SOFTWARE FLAG BYTES
17
18 027530   000   CLKFL1: .BYTE 0           ;CLOCK TICKING
19 027531   000   CLKFLG: .BYTE 0          ;CLOCK PRESENT
20 027532   000   CONFLG: .BYTE 0          ;CONTINUE FLAG
21 027533   000   CNFGFL: .BYTE 0         ;CONFIGURATION AVAILABLE
22 027534   000   DBFLG: .BYTE 0          ;DEBUG FLAG
23 027535   000   GOFLG: .BYTE 0          ;'GO' FLAG
24 027536   000   HOURS: .BYTE 0          ;HOURS STORAGE
25 027537   000   INCF: .BYTE 0           ;LINE FEED FLAG
26 027540   000   LSIFLG: .BYTE 0         ;THIS IS AN 11/03
27 027541   000   MINS: .BYTE 0           ;MINUTES STORAGE
28 027542   000   OFLOF: .BYTE 0         ;ERROR COUNT OVERFLOW
29 027543   000   ONCE: .BYTE 0           ;INDICATOR THAT CONFIG HAS BEEN PRINTED
30 027544   000   PNCFLG: .BYTE 0        ;PRINTED ANY REMOTE INFO?
31 027545   000   SAFE: .BYTE 0           ;DON'T ALLOW ^F IN FILL COUNT RTN.
32 027546   000   SECS: .BYTE 0           ;SECONDS STORAGE
33 027547   000   SLFL: .BYTE 0           ;'SLASH' FLAG
34 027550   000   THRUF: .BYTE 0         ;'THRU' FLAG
35 027551   000   TICKS: .BYTE 0         ;1/60 SEC. TICKS STORAGE
36          .EVEN
    
```

CONSTANTS, VARIABLES & ACSII

```

1
2
3 027552 000000
4 027554 000000
5 027556 000000
6 027560 000000
7 027562 000000
8 027564 000000
9 027566 000000
10 027570 000000
11 027572 000000
12 027574 000000
13 027576 000000
14 027600 000000
15 027602 000000
16 027604 000000
17 027606 000000
18 027610 000000
19 027612 000000
20 027614 000000
21 027616 000000
22 027620 000000
23 027622 000000
24 027624 000000
25 027626 000000
26 027630 000000
27 027632 000000
28 027634 000000
29 027636 000000
30 027640 000000
31 027642 000000
32 027644 000000
33 027646 000000
34 027650 000000
35 027652 000000
36
37
38
39
40 027654 000170
41 027656 164070
42 027660 000004
43 027662 177562
44 027664 177560
45 027666 177546
46 027670 177566
47 027672 177564
48
49 000001

;VARIABLES:
ADDR: .WORD 0
ALLFLG: .WORD 0
BAUD: .WORD 0
CHNGFL: .WORD 0
CLK: .WORD 0
CNTR: .WORD 0
CSR: .WORD 0
DEVER: .WORD 0
ERRCNT: .WORD 0
ITER: .WORD 0
LINLEN: .WORD 0
MASK: .WORD 0
PASCNT: .WORD 0
PAT: .WORD 0
PCONT: .WORD 0
PORN: .WORD 0
PORT: .WORD 0
PRIO: .WORD 0
REMFLG: .WORD 0
REMNO: .WORD 0
REMTN: .WORD 0
REMST: .WORD 0
SAVEC: .WORD 0
SPAR: .WORD 0
SPDR: .WORD 0
SR: .WORD 0
SWREG: .WORD 0
SWRFLG: .WORD 0
TALLY: .WORD 0
TBLAD: .WORD 0
TEMP: .WORD 0
VERFLG: .WORD 0
WORD: .WORD 0

;CONSTANTS:
VECTOR: .WORD 170
DEVADR: .WORD 164070
FKPRIO: .WORD 4
KBD: .WORD 177562
KBS: .WORD 177560
LCS: .WORD 177546
TTB: .WORD 177566
TTS: .WORD 177564

.END

```

SYMBOL TABLE			
ADDR	027552	CSRCE	024623
AGAIN	021324	CSRST	024137
ALLFLG	027554	CSRWE	024571
ALROFF	023765	DATBL	026424
BAD	016614	DATLER	025560
BAUD	027556	DBAID	021442
BAUDM	023052	DBFLG	027534
BCNT0	003020	DECJSP	020450
BCNT1	003024	DECPNT	020422
BCNT2	003054	DERR	016526
BCNT3	003050	DEVADR	027656
BCNT4	003100	DEVER	027570
BCONT	002664	DEVGEM	016262
BDRTEL	026302	DIAGF	026202
BD1	026322	DIAGFU	013226
BD2	026327	DIAGNF	014764
BD3	026334	DIAGUF	026245
BD4	026342	DIAGO	015046
BD5	026350	DIAG00	015064
BD6	026356	DIAG1	015104
BD7	026364	DIAG10	015122
BD8	026402	DIAG2	015210
BEGIN	002000	DIAG3S	015374
BIMPNT	020374	DIAG4	015402
B00	= 000001	DIERR	024536
B01	= 000002	DIVID	020524
B02	= 000004	DIVSET	020522
B03	= 000010	DVATST	016150
B04	= 000020	DVECT	021357
B05	= 000040	DVPRI0	021403
B06	= 000100	ELTIM	023164
B07	= 000200	ENDCHK	012270
B08	= 000400	ENDPS	023150
B09	= 001000	END0	012350
B10	= 002000	ENPRTN	011050
B11	= 004000	ENRET	011130
B12	= 010000	ENO1	011120
B13	= 020000	ERCTM	023177
B14	= 040000	ERINT	006302
B15	= 100000	ERINTR	025031
CCRTN	017002	ERINT1	006422
CELP	012542	ERR	016442
CHNGFL	027560	ERRAD	016606
CLK	027562	ERRCNT	027572
CLKEXT	002662	ERRM	016616
CLKFLG	027531	ERRNUM	016610
CLKFL1	027530	ERR1	016454
CLKINT	002574	EXCM	023342
CLKTRP	002566	EXS1	012612
CLTBER	024474	EXS2	012702
CMREXC	012500	EXS2B	012742
CNFGFL	027533	EXS3	013002
CNTR	027564	EXS3A	013050
CONFLG	027532	EXS3B	013072
CONFMS	023600	EXTBL1	026464
CONTW1	017012	EXTBL2	026664
CSR	027566	EXTBL3	027064
EXTBL4	027264		
EX1	017054		
EX2	017072		
EX3	017104		
EX4	017122		
EX5	017154		
EX6	017166		
FILL	020242		
FILMSG	020246		
FKPRIO	027660		
FLCNT	020244		
FRAD	021277		
GDINT	004344		
GETCHR	017352		
GETDG	020542		
GOFLG	027535		
GOOD	016612		
HDR	016370		
HELPM	021725		
HITER	025125		
HLPASK	003130		
HLPDSR	022714		
HOURS	027536		
HWSWR	= 177570		
INCF	027537		
INPKB	017334		
INTCNR	006416		
INTCNT	006356		
INVLAD	021467		
IRRGMG	023752		
ISATE	024240		
ISDTE	024362		
ISP	= 002000		
ITER	027574		
JUST	020574		
KEBUF	020240		
KBD	027662		
KBRD	017574		
KBRET	017606		
KBS	027664		
KRTN	020234		
LCS	027666		
LGKTSM	022761		
LINLEN	027576		
LOGTST	003144		
LSIFLG	027540		
LSTST	016122		
MASK	027600		
MINS	027541		
MONCH	016706		
MONIC	016702		
MONIT	015666		
MOV	006476		
MOVB	003212		
N	= 000054		
NEK	017566		
NOCNF	021605		
NOGO	024031		
NOINTR	024773		
NOKEFL	020236		
NOREM	023450		
NOTUP	024066		
NRTRN	017550		
NULLS	020500		
MULOUT	020512		
MUMPNT	020514		
OCTJSP	020344		
OCTPNT	020316		
ODTPR	015412		
OFLOF	027542		
ONCE	027543		
OSATE	024326		
OSDTE	024271		
OVFLM	023215		
PASCNT	027602		
PAT	027604		
PCONT	027606		
PLDERR	026101		
PLFFE	025762		
PLRCTO	026053		
PLTM	023232		
PLTXTO	026025		
PNCFLG	027544		
PNT	020614		
PNTCNF	020266		
PNTCX	020314		
PNTCO	020304		
PNTEND	012406		
PNTXT	020556		
PNTRTN	012470		
PNT0	012424		
PNT1	012432		
POK	011210		
PORCN	014316		
PORCNF	014072		
PORLD	014304		
PORN	027610		
PORT	027612		
PORTLP	011132		
PORTMS	023556		
PORTN	023260		
PORTOM	023070		
PORT1M	023104		
PORT2M	023120		
PORT3M	023134		
PORXT	014346		
PORO	014166		
POROO	014204		
POR1	014260		
PRDFLT	016326		
PREML	014744		
PREMO	014616		
PREM1	014726		
PRI0	027614		
PRLC	011140		
PRMT1	002200		
PRMT2	002224		
PRMT3	002312		
PRMT4	002372		
PRMT5	002416		
PRNREM	014576		
PRNTIM	017170		
PROMT	002112		
PRTCNF	013106		
PS	= 177776		
PSTE	011342		
PSTS	011504		
PST0	011246		
PST1	011330		
PST2	011356		
PST2A	011364		
PST3	011466		
PST4	011514		
PST5	011550		
PST6	011576		
PST60	011612		
PST7	011640		
PST7S	011726		
PST8	011740		
PST8S	012026		
PST80	012040		
PST80S	012126		
PST9	012140		
PST9S	012262		
P0	= 000000		
P1	= 000040		
P2	= 000100		
P3	= 000140		
P4	= 000200		
P5	= 000240		
P6	= 000300		
P7	= 000340		
RADIX	020710		
REGMG	023741		
REGRES	016104		
REGSAV	016070		
REMCHG	023657		
REMFLG	027616		
REMNO	027620		
REMSG	023566		
REMST	027624		
REMSTM	023715		
REMTN	027622		
REPORT	014374		
REPO	014446		
REP1	014476		
REP2	014526		
REP20	014556		
REP3	014574		
RESTRT	002514		
RJFLG	020706		

RQSPCL	024414	SPAVME	024662	ST160	005666	ST60	003616	TYPCTP	021650
RSPEMF	025620	SPDR	027632	ST161	005754	ST61	003632	TYPOUT	017272
RSPESF	025370	SPRCLR	024164	ST162	006062	ST62	003742	USOFE1	025431
RSSPCL	024444	SPRDAT	024211	ST17	006170	ST7	004002	USOFE2	025523
SAFE	027545	SR	027634	ST170	006214	ST70	004012	USOFE3	025662
SAVEC	027626	SRTST	016132	ST170R	006276	ST71	004064	USOFE4	025725
SBAK	016226	SSWR =	027636	ST18	006470	ST80	004074	USORM	023646
SCALM	013502	STATEM	023634	ST180	006622	ST80S	004176	VECTOR	J27654
SCART	014070	STEMF	025472	ST181	006650	ST81	004204	VERFLG	027650
SCA1	013544	STESTF	025337	ST182	006712	ST82	004226	VERSM	023017
SCA2	013632	STO	003202	ST183	007002	ST90	004242	VERSP	027464
SCA2A	013726	ST1	003322	ST19	007022	ST90S	004336	VERSUN	027506
SCA2B	013750	ST10	004362	ST2	003416	ST91	004344	VIDER	025225
SCA2C	013762	ST100	004374	ST20	007130	SWDMP	016720	VISF	025165
SCA2D	013770	ST101	004446	ST200	007240	SWHLT	016170	WDAT	016631
SCA2E	014002	ST110	004460	ST201	007254	SWREG	027636	WDSDB	016643
SCA2F	014024	ST110S	004562	ST202	007334	SWRFLG	027640	WDWAS	016657
SCA3	014044	ST111	004570	ST203	007422	SWRMSG	021575	WORD	027652
SCONT	016244	ST112	004612	ST204	007510	YSER	026141	WPODI	025064
SCORTN	016260	ST120	004624	ST205	007540	TALLY	027642	\$0\$	015454
SCPAD	022776	ST120S	004720	ST206	007620	TBLAD	027644	\$00\$	015430
SCHRTN	016210	ST121	004726	ST207	007706	TEMP	027646	\$1\$	015460
SECS	027546	ST13	004744	ST21	007774	THRUF	027550	\$10\$	015544
SKPD2	013410	ST130	004750	ST22	010022	TICKS	027551	\$11\$	015670
SKPX0	013426	ST131	005026	ST220	010102	TPCONT	017320	\$13\$	016006
SKPX1	013472	ST131A	004774	ST221	010110	TPOFCH	017274	\$2A\$	015572
SKP1	013244	ST131B	005002	ST222	010206	TPOUTX	017330	\$2\$	015556
SKP2	013262	ST132	005040	ST223	010274	TRP4	021533	\$20\$	015736
SKP3	013300	ST132S	005134	ST224	010310	TSTHDR	020712	\$21\$	015750
SKP4	013316	ST133	005142	ST23	010406	TSTRSK	003104	\$22\$	015766
SKP5	013334	ST134	005162	ST23A	010420	TSTSEL	021676	\$23\$	016020
SKP6	013352	ST135	005176	ST230	010454	TTB	027670	\$24\$	016066
SKP7	013370	ST135S	005272	ST24	010770	TTO	020622	\$25\$	016036
SKP8	013406	ST136	005300	ST25	011006	TTOLF	020644	\$26\$	016050
SLFL	027547	ST14	005316	ST26	011040	TTOLP	020626	\$3\$	015624
SPAR	027630	ST15	005402	ST3	003502	TTS	027672	\$4\$	015632
SPAVFE	024726	ST150	005556	ST4	003522	TWOSP	021571	\$5\$	015650
SPAVIC	025274	ST16	005600	ST6	003542	TYPCR	023312		

. ABS. 027674 000
 000000 001
 ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 1395 WORDS (6 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 65 PAGES
 DK1:ZCJUBO,ZCJUBO/C=DK1:ZCJUBO

NEK	31-49	31-53#															
NOCNF	33-9	36-21#															
NOGO	7-22	18-12	19-22	37-3#													
NOINTR	13-11	37-20#															
NOKEFL	31-8*	31-33*	31-40*	31-48	32-97#												
NOREM	19-68	36-58#															
NOTUP	7-31	37-4#															
NRTRN	31-16	31-48#	32-82	32-87	32-91												
NULLS	17-57	17-69	18-151	21-52	28-100	34-56#											
NULOUT	34-59#																
NUMPNT	34-15	34-24	34-33	34-42	34-51	35-3#											
OCTJSP	25-12	27-33	27-60	34-21#													
OCTPNT	9-27	9-44	9-73	9-90	10-27	10-43	18-83	18-93	18-103	18-117	22-28	22-40	25-9	25-17			
	26-44	26-48	26-52	27-72	28-85	28-99	28-113	28-116	28-127	28-130	29-21	32-39	34-12#				
ODTPR	27-17#	29-45	32-72														
OFLOF	7-9*	17-78	18-18*	19-15*	28-63*	39-28#											
ONCE	19-29*	19-49	19-62	19-67*	19-72*	22-49*	39-29#										
OSATE	9-68	37-10#															
OSDTE	9-39	37-9#															
OVFLM	17-80	36-52#															
PC	1-32#	18-64															
P1	1-31#																
P2	1-30#																
P3	1-29#																
P4	1-28#																
P5	1-27#																
P6	1-26#																
P7	1-25#	4-13	4-17	4-21	4-89	6-5											
PASCNT	7-8*	17-65*	17-75	18-16*	18-144*	19-13*	21-45*	40-15#									
PAT	8-11*	8-13	8-23	8-25*	8-26*	40-16#											
PCONT	4-28*	4-43*	4-53*	6-16*	6-18*	18-22*	18-24*	18-55*	18-58*	27-88	27-90	29-24*	29-26*	29-57*			
	32-48	32-51	32-65	32-67	40-17#												
PLDERR	18-112	37-39#															
PLFFE	18-78	37-36#															
PLRCTO	18-98	37-38#															
PLTM	18-19	36-53#															
PLTXTO	18-88	37-37#															
PNCFLG	24-5*	24-10*	24-16*	24-22*	24-28*	24-31	39-30#										
PNT	35-19	35-21	35-27#														
PNT0	18-140	18-143#															
PNT1	18-144#																
PNTCO	33-6	33-9#															
PNTCNF	29-61	32-57	33-5#														
PNTCX	33-8	33-10#															
PNTEND	18-123	18-131	18-139#														
PNTXT	35-13	35-17#															
PNTRTN	18-142	18-146	18-152#														
POK	18-9	18-14#															
PORO	23-24#	23-27															
POROO	23-25	23-28#															
POR1	23-31	23-34#															
PORCN	23-43#	23-48															
PORCNF	20-9	20-12	20-15	20-18	23-12#												
PORLD	23-39	23-41#															
PORN	20-8*	20-11*	20-14*	20-17*	23-21	23-49	24-11*	24-17*	24-23*	24-29*	25-8	40-18#					
PORT	18-28*	18-36*	18-45	18-47	18-82	18-92	18-102	18-116	18-124	18-126*	18-128	18-130	18-132*	18-134			

	18-136	40-19#					
PORTOM	17-17	36-45#					
PORT1M	17-29	36-46#					
PORT2M	17-37	36-47#					
PORT3M	17-49	36-48#					
PORTLP	6-30	18-7#					
PORTMS	18-81	18-91	18-101	18-115	25-7	36-60#	
PORTN	18-20	36-54#					
PORXT	23-40	23-49#					
PRDFLT	4-27	4-42	4-52	28-82#			
PREMO	25-10#	25-23	25-32				
PREM1	25-29#	25-31					
PREML	25-21	25-33#					
PRI0	4-68*	13-24	40-20#				
PRLC	18-8#	18-11					
PRMT1	4-33	4-36#					
PRMT2	4-40#	4-43	4-49				
PRMT3	4-47	4-50#	4-53	4-58	4-62		
PRMT4	4-56	4-59#	32-67				
PRMT5	4-60	4-63#					
PRNREM	24-12	24-18	24-24	24-30	25-7#		
PRNTIM	18-150	21-51	29-67#				
PROMT	4-25#	4-28	4-35	28-38			
PRTCNF	19-57	20-5#	20-20				
PS	1-8#						
PST0	18-20#	18-22	18-35				
PST1	18-26	18-31#					
PST2	18-33	18-36#					
PST2A	18-29	18-37#					
PST3	18-50#	18-53					
PST4	18-55#						
PST5	18-64#	18-84	18-94	18-104	18-118	18-137	18-153
PST6	18-67#	18-70					
PST60	18-68	18-71#					
PST7	18-73	18-76#					
PST7S	18-80	18-84#					
PST8	18-77	18-86#					
PST80	18-87	18-96#					
PST80S	18-100	18-104#					
PST8S	18-90	18-94#					
PST9	18-97	18-106#					
PST9S	18-114	18-118#					
PSTE	18-31	18-34#					
PSTS	18-54#	18-55	18-60				
RADIX	35-3*	35-6	35-8	35-54#			
REGMG	22-33	36-67#					
REGRES	28-15#	34-16	34-25	34-34	34-43	34-52	
REGSAV	28-8#	34-12	34-21	34-30	34-39	34-48	
REMCHG	22-24	36-65#					
REMFLG	40-21#						
REMNO	21-8*	21-17*	21-25*	21-33*	26-14	40-22#	
REMSG	22-25	26-41	36-61#				
REMSI	40-24#						
REMSM	22-29	36-66#					
REMTN	40-23#						
REPO	24-9	24-13#					

DATERR	2-17#	7-31	7-38	8-9	8-21	9-22	9-39	9-68	9-85	10-22	10-38	11-10	11-35	12-8
	12-15	14-15	14-32	15-29	15-36	15-43	15-54	15-61	15-68	16-22	16-29	16-40	18-112	26-31
ERROR	26-38													
	2-7#	7-22	12-25	12-35	13-11	13-16	13-31	14-24	14-52	18-12	18-78	18-88	18-98	19-22
	22-22	23-32												
HLT	2-27#	18-13	19-23	22-23	23-33	28-55								
MTPS	2-47#	4-13	4-89	6-5	13-4	13-24	18-64	19-12						
PNTM	2-33#	4-24	4-25	4-34	4-40	4-48	4-50	4-57	4-61	6-14	6-40	6-42	7-11	9-25
	9-42	9-71	9-88	10-25	10-41	17-10	17-16	17-17	17-29	17-37	17-49	17-74	17-77	17-80
	18-19	18-20	18-34	18-54	18-81	18-91	18-101	18-115	19-28	19-68	22-24	22-25	22-29	22-33
	22-37	22-50	24-4	24-33	25-7	26-41	26-45	26-49	27-23	27-37	27-61	28-37	28-96	28-111
	28-114	28-125	28-128	29-19	29-22	29-28	29-67	32-36	32-40	32-45	33-9			
REGRES	2-61#	34-16	34-25	34-34	34-43	34-52								
REGSAV	2-57#	34-12	34-21	34-30	34-39	34-48								
SCOPL	2-40#	7-23	7-32	7-39	8-10	8-22	9-28	9-45	9-74	9-91	10-28	10-44	11-11	11-40
	12-9	12-16	12-26	12-36	13-13	13-17	13-32	14-16	14-25	14-33	14-53	15-30	15-37	15-44
	15-55	15-62	15-69	16-23	16-30	16-41	18-84	18-94	18-104	18-118	26-32	26-53		