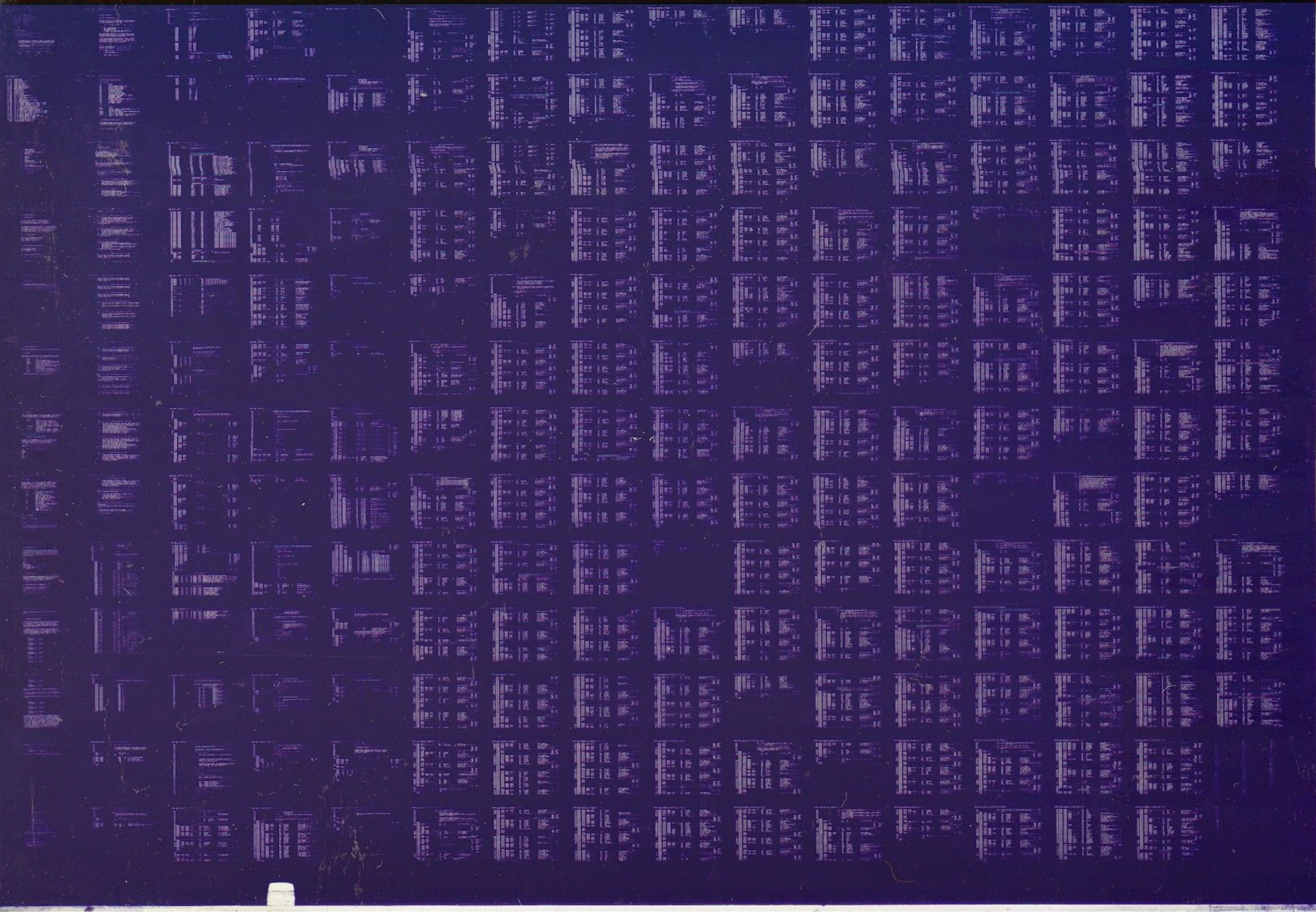


IEU11-A,  
IEQ11-A

IEU/IEQ STATIC DIAG  
CZIEABO

AH-T066B-MC  
FICHE 1 OF 2

MAY 1983  
COPYRIGHT © 82-83  
MADE IN USA





IEU11-A,  
IEQ11-A

IEU/IEQ STATIC DIAG  
CZIEABO

AH-T066B-MC  
FICHE 2 OF 2

MAY 1983  
COPYRIGHT © 82-83  
MADE IN USA



*[Faint, illegible text visible on the left edge of the page, possibly bleed-through from the reverse side.]*



.REM 8

IDENTIFICATION

PRODUCT CODE: AC-T064B-MC  
PRODUCT NAME: CZIEAB0 IEU/IEQ STATIC DIAG  
PRODUCT DATE: SEPTEMBER 1982  
MAINTAINER: CSS MUNICH  
AUTHOR: PETER SEEBACH

REMARKS TO VERSION B  
\*\*\*\*\*

INITCODE MODIFIED SO THAT SEVERAL UNITS CAN BE RUN .  
THE CHANGES ARE MARKED WITH \*B\*  
UPDATED TO VERSION B BY PETER SEEBACH, 6-SEP-82

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.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1982, 1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL  
DEC

PDP  
DECUS

UNIBUS  
DECTAPE

MASSBUS



PROGRAM HEADER AND TABLES  
TABLE OF CONTENTS

MACRO M1113 06-SEP-82 16:46

14-	1012	PROGRAM HEADER
15-	1085	DISPATCH TABLE
16-	1101	DEFAULT HARDWARE P-TABLE
18-	1132	SOFTWARE P-TABLE
19-	1171	GLOBAL EQUATES SECTION
20-	1234	GLOBAL DATA SECTION
21-	1400	GLOBAL TEXT SECTION
22-	1458	GLOBAL ERROR REPORT SECTION
23-	1570	LOCAL MACRO DEFINITIONS
23-	1603	GLOBAL SUBROUTINES SECTION
28-	2086	GLOBAL INTERRUPT HANDLING ROUTINES
36-	2349	REPORT CODING SECTION
37-	2417	PROTECTION TABLE
38-	2446	INITIALIZE SECTION
39-	2599	AUTODROP SECTION
40-	2633	CLEANUP CODING SECTION
41-	2672	DROP UNIT SECTION
42-	2716	ADD UNIT SECTION
43-	2767	TEST 1: REGISTER ADDRESSING TEST
44-	2902	TEST 2: INITIALIZATION TEST
45-	2989	TEST 3: R/W BIT TEST
46-	3065	TEST 4 : SYSTEM CONTROLLER COMMANDS TEST
47-	3246	TEST 5: INTERRUPT TEST
48-	3368	TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1
49-	3808	TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2
50-	4273	TEST 8: DATA TRANSFER TEST
51-	4556	TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)
52-	4737	TEST 10: SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)
53-	4916	TEST 11: SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)
54-	5095	TEST 12: SECONDARY ADDRESSING TEST OF CHANNEL 2 (TALKER)
55-	5286	TEST 13: DEVICE CLEAR INTERFACE FUNCTION TEST
56-	5522	TEST 14: DEVICE TRIGGER INTERFACE FUNCTION TEST
57-	5625	TEST 15: INCOMPLETE SOURCE HANDSHAKE TEST
58-	5747	TEST 16: CHANGING OF THE CONTROLLER CONFIGURATION
59-	5903	TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST
60-	6165	TEST 18: SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1
61-	6293	TEST 19: SERVICE REQUEST INTERFACE FUNTION TEST OF CHANNEL 2
62-	6421	TEST 20: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1
63-	6538	TEST 21: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 2
64-	6655	TEST 22: END OF A MESSAGE BLOCK TEST
65-	6874	TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2
66-	7327	TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1
67-	7779	TEST 25: MCR FUNCTION TEST OF CHANNEL 1
68-	7970	TEST 26: MCR FUNCTION TEST OF CHANNEL 2
69-	8163	TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST
70-	8493	TEST 28: ADDITIONAL STANDBY TEST
71-	8609	HARDWARE PARAMETER CODING SECTION
73-	8689	SOFTWARE PARAMETER CODING SECTION



## TABLE OF CONTENTS

1.0	GENERAL INFORMATION
1.1	PROGRAM ABSTRACT
1.2	SYSTEM REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
1.6	EXECUTION TIME
2.0	OPERATING INSTRUCTIONS
2.1	COMMANDS
2.2	SWITCHES
2.3	FLAGS
2.4	HARDWARE QUESTIONS
2.5	SOFTWARE QUESTIONS
2.6	EXTENDED P-TABLE DIALOGUE
2.7	QUICK STARTUP PROCEDURE
3.0	ERROR INFORMATION
3.1	TYPES OF ERROR MESSAGES AND REPORTS
3.2	SPECIFIC ERROR MESSAGES AND REPORTS
4.0	PERFORMANCE AND PROGRESS REPORTS
5.0	DEVICE INFORMATION TABLES
6.0	TEST SUMMARIES
7.0	PROGRAM LISTING



## 1.0 GENERAL INFORMATION

### 1.1 PROGRAM ABSTRACT

THE IEU-11/IEQ-11 DIAGNOSTIC PROGRAM PROVIDES A SERIES OF TESTS DESIGNED TO VERIFY THE INTEGRITY AND FUNCTIONALITY OF THE IEU-11 OR IEQ-11 INTERFACE. TEST 3, 27 AND 28 ARE DEPENDENT ON THE INTERFACE AND ON THE SOFTWARE P-TABLE ANSWERS (SEE 6.27 AND 6.28).

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

### 1.2 SYSTEM REQUIREMENTS

PDP-11/LSI-11 PROCESSOR WITH 32K MEMORY OR MORE  
IEQ-11 INTERFACE (M-8634) FOR THE LSI  
IEU-11 INTERFACE (M-5648) FOR THE PDP  
CONSOLE TERMINAL (VT100, LA36, ECT.)  
XXDP+ LOAD DEVICE (RX, RK, RL ECT.)

### 1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USER MANUAL (CHQUSA )  
IEU11-A OPTION DESCRIPTION (YG-C03KC-00) OR  
IEQ11-A OPTION DESCRIPTION  
IEU/IEQ DIAGNOSTIC LISTING

### 1.4 DIAGNOSTIC HIERARCY PREREQUISITES

ALL PDP-11 OR LSI-11 PROCESSOR DIAGNOSTIC SHOULD RUN SUCCESSFULLY

### 1.5 ASSUMPTIONS



PARAMETER CODING

MACRO M1113 06-SEP-82 16:46 PAGE 4-1

NONE

1.6 EXECUTION TIME

EXECUTION TIME IS DEPENDENT ON THE PROCESSOR SPEED AND THE TYPE OF  
TEST EXECUTION (QUICK VERIFY PASS OR NOT):  
THE FOLLOWING ARE TYPICAL EXECUTION TIMES OBSERVED ON A PDP-11/60:

QUICK VERIFY PASS ? YES ,EXECUTION TIME FOR THE WHOLE PASS IS 28 SEC.

QUICK VERIFY PASS ? NO ,EXECUTION TIME FOR THE WHOLE PASS IS 6.75 MIN.



## 2.0 OPERATING INSTRUCTIONS

-----

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

### 2.1 COMMANDS

-----

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
-----	-----
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME)
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".



2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

-----  
 START/TESTS:1-5/PASS:1000/EOP:100  
 -----

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					



## 2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDU	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

\*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

### 2.3.1 FLAG COMMANDS

-----  
 FLA(GS)            THIS COMMAND PRINTED THE CURRENT SETTING  
 -----  
                   OF ALL FLAGS

-----  
 ZFL(AGS)           THIS COMMAND CLEARED ALL FLAGS  
 -----

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

/FLAGS:LOE:IER:BOE



## 2.4 HARDWARE QUESTIONS

-----

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT.

EXAMPLE OF THE DIALOGUE:

```
CHANGE HW (L) ? Y
#UNITS (D) ? 1
UNIT 0
DEVICE ADDRESS          (O) 760150 ?
INTERRUPT VECTOR       (O) 420 ?
PRIORITY LEVEL         (O) 6 ?
DEVICE PRIMARY ADDRESS CH 0 (O) 0 ?
DEVICE PRIMARY ADDRESS CH 1 (O) 1 ?
IS TESTCABLE IN       (L)N/Y ?
```

## 2.5 SOFTWARE QUESTIONS

-----

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?". IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y". THE SOFTWARE QUESTIONS AND THE DEFAULT VALUE ARE :

QUICK VERIFY PASS (L) Y ?

IF 'Y', THEN EACH TEST WILL ITERATE ONLY ONCE BEFORE CONTINUING TO THE NEXT TEST IN SEQUENCE.

IF 'N', THEN EACH TEST WILL ITERATE 20 TIMES BEFORE CONTINUING ON THE NEXT TEST IN SEQUENCE.  
ALSO SOME TESTS WILL DO WITH MORE TESTPATTERNS .

NUMBER OF MATCH CHARACTER COUNTS (O) 63.?

AFTER THIS QUESTION YOU CAN CHANGE THE NUMBER OF DMA CYCLES

NUMBER OF BYTE COUNTS (O) 2047. ?



PARAMETER CODING

MACRO M1113 06-SEP-82 16:46 PAGE 9

AFTER THIS QUESTION YOU CAN SELECT THE ADDITIONAL STANDBY TEST WHICH CHECK EXTENDED ADDRESS BITS FOR THE Q-22 BUS WITHOUT MEMORY.

DO YOU WANT THE ADDITIONAL STANDBY TEST (L) ? N

## 2.6 EXTENDED P-TABLE DIALOGUE

-----

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THE SIMPLEST WAY TO BUILD THIS TABLE IS TO ANSWER ALL QUESTIONS FOR EACH UNIT TO BE TESTED. IF YOU HAVE A MULTIPLEXED DEVICE SUCH AS A MASS STORAGE CONTROLLER WITH SEVERAL DRIVES OR A COMMUNICATION DEVICE WITH SEVERAL LINES, THIS BECOMES TEDIOUS SINCE MOST OF THE ANSWERS ARE REPETITIOUS.

TO ILLUSTRATE A MORE EFFICIENT METHOD, SUPPOSE YOU ARE TESTING A FICTIONAL DEVICE, THE XY11. SUPPOSE THIS DEVICE CONSISTS OF A CONTROL MODULE WITH EIGHT UNITS (SUB-DEVICES) ATTACHED TO IT. THESE UNITS ARE DESCRIBED BY THE OCTAL NUMBERS 0 THROUGH 7. THERE IS ONE HARDWARE PARAMETER THAT CAN VARY AMONG UNITS CALLED THE Q-FACTOR. THIS Q-FACTOR MAY BE 0 OR 1. BELOW IS A SIMPLE WAY TO BUILD A TABLE FOR ONE XY11 WITH EIGHT UNITS.

# UNITS (D) ? 8<CR>

UNIT 1  
CSR ADDRESS (O) ? 160000<CR>  
SUB-DEVICE # (O) ? 0<CR>  
Q-FACTOR (O) 0 ? 1<CR>

UNIT 2  
CSR ADDRESS (O) ? 160000<CR>  
SUB-DEVICE # (O) ? 1<CR>  
Q-FACTOR (O) 1 ? 0<CR>

UNIT 3  
CSR ADDRESS (O) ? 160000<CR>  
SUB-DEVICE # (O) ? 2<CR>  
Q-FACTOR (O) 0 ? <CR>

UNIT 4  
CSR ADDRESS (O) ? 160000<CR>  
SUB-DEVICE # (O) ? 3<CR>  
Q-FACTOR (O) 0 ? <CR>

UNIT 5  
CSR ADDRESS (O) ? 160000<CR>  
SUB-DEVICE # (O) ? 4<CR>  
Q-FACTOR (O) 0 ? <CR>

UNIT 6  
CSR ADDRESS (O) ? 160000<CR>  
SUB-DEVICE # (O) ? 5<CR>



PARAMETER CODING

MACRO M1113 06-SEP-82 16:46 PAGE 9-1

Q-FACTOR (0) 0 ? &lt;CR&gt;

UNIT 7

CSR ADDRESS (0) ? 160000&lt;CR&gt;

SUB-DEVICE # (0) ? 6&lt;CR&gt;

Q-FACTOR (0) 0 ? 1&lt;CR&gt;

UNIT 8

CSR ADDRESS (0) 160000&lt;CR&gt;

SUB-DEVICE # (0) ? 7&lt;CR&gt;

Q-FACTOR (0) 1 ? &lt;CR&gt;

NOTICE THAT THE DEFAULT VALUE FOR THE Q-FACTOR CHANGES WHEN A NON-DEFAULT RESPONSE IS GIVEN. BE CAREFUL WHEN SPECIFYING MULTIPLE UNITS!

AS YOU CAN SEE FROM THE ABOVE EXAMPLE, THE HARDWARE PARAMETERS DO NOT VARY SIGNIFICANTLY FROM UNIT TO UNIT. THE PROCEDURE SHOWN IS NOT VERY EFFICIENT.

THE RUNTIME SERVICES CAN TAKE MULTIPLE UNIT SPECIFICATIONS HOWEVER. LET'S BUILD THE SAME TABLE USING THE MULTIPLE SPECIFICATION FEATURE.

# UNITS (D) ? 8&lt;CR&gt;

UNIT 1

CSR ADDRESS (0) ? 160000&lt;CR&gt;

SUB-DEVICE # (0) ? 0,1&lt;CR&gt;

Q-FACTOR (0) 0 ? 1,0&lt;CR&gt;

UNIT 3

CSR ADDRESS (0) ? 160000&lt;CR&gt;

SUB-DEVICE # (0) ? 2-5&lt;CR&gt;

Q-FACTOR (0) 0 ? 0&lt;CR&gt;

UNIT 7

CSR ADDRESS (0) ? 160000&lt;CR&gt;

SUB-DEVICE # (0) ? 6,7&lt;CR&gt;

Q-FACTOR (0) 0 ? 1&lt;CR&gt;

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 160000 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE "-" CONSTRUCT TELLS THE RUNTIME SERVICES TO INCREMENT THE DATA FROM THE FIRST NUMBER TO THE SECOND. IN THIS CASE, SUB-DEVICES 2, 3, 4 AND 5 WERE SPECIFIED. (IF THE SUB-DEVICE WERE SPECIFIED BY ADDRESSES, THE INCREMENT WOULD BE BY 2 SINCE ADDRESSES MUST BE ON AN EVEN BOUNDARY.) THE CSR ADDRESSES AND Q-FACTORS FOR THE FOUR ENTRIES ARE ASSUMED TO BE 160000 AND 0 RESPECTIVELY SINCE THEY WERE ONLY SPECIFIED ONCE. THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.



PARAMETER CODING

MACRO M1113 06-SEP-82 16:46 PAGE 9-2

THE WHOLE PROCESS COULD HAVE BEEN ACCOMPLISHED IN ONE PASS AS SHOWN BELOW.

# UNITS (D) ? 8&lt;CR&gt;

UNIT 1

CSR ADDRESS (O) ? 160000&lt;CR&gt;

SUB-DEVICE # (O) ? 0-7&lt;CR&gt;

Q-FACTOR (O) 0 ? 0,1,0,,,,,1,1&lt;CR&gt;

AS YOU CAN SEE FROM THIS EXAMPLE, NULL REPLIES (COMMAS ENCLOSING A NULL FIELD) TELL THE RUNTIME SERVICES TO REPEAT THE LAST REPLY.

2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK) QUESTIONS
3. TYPE "R NAME", WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE "START"
5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"
6. ANSWER ALL THE HARDWARE QUESTIONS
7. ANSWER THE "CHANGE SW" QUESTION WITH "N"

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

2.7.1 CONTROL CHARACTERS

A CONTROL C ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES A RETURN TO COMMAND MODE.

A CONTROL Z ENTERED DURING ONE OF THE OPERATOR DIALOGUES CAUSES THE DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.



### 3.0 ERROR INFORMATION

#### 3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE "IER" FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

NAME ,TYPE,NUMBER,UNIT NUMBER,TST NUMBER,PC:XXXXXX

,WHERE; NAME = DIAGNOSTIC NAME  
 TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)  
 NUMBER = ERROR NUMBER  
 UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)  
 TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED  
 PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBE" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE "IER", "IBR" OR "IXE" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

#### EXAMPLES:

CZIEA SOFT ERR 00301 ON UNIT 00 TST 003 SUB 000 PC: 001234  
 READ-WRITE BITS NOT CORRECT  
 REGISTER : IIR , CHANNEL : 1 , GOOD DATA :00000 ,BAD DATA :00001

NOTE THAT THE ERROR NUMBER IS IN THE FORMAT 'TNN' WHERE:

T IS THE TEST NUMBER AND  
 NN IS THE ERROR NUMBER WITHIN THE TEST

IE. 00302 = ERROR 2 IN TEST 03.  
 00504 = ERROR 4 IN TEST 05.

### 3.2 SPECIFIC ERROR MESSAGES AND REPORTS

---

#### ERROR MESSAGES:

CALL	MESSAGE
E101	REGISTER ADDRESSING ERROR - TRAP 4
E200	REGISTER INCORRECT AFTER BUS RESET
E301	READ - WRITE BITS INCORRECT
E302	BITS NOT CLEARED AFTER BUS RESET
E303	MUX BIT IN CSR NOT SETTABLE
E401	CSR CONTENTS INCORRECT
E402	NO INTERRUPT WHEN EXPECTED
E403	INCORRECT PRIORITY LEVEL
E501	BITS IN IIR REGISTER INCORRECT
E502	BITS IN ISR REGISTER INCORRECT
E801	DATA TRANSFER FROM CHANNEL 1 TO 2 INCORRECT
E802	DATA TRANSFER FROM CHANNEL 2 TO 1 INCORRECT
E901	ICR CONTENTS INCORRECT
E222	DIR CONTENTS INCORRECT
E250	RX BUFFER CONTENTS INCORRECT AFTER DMA (2 TO 1)
E231	RX BUFFER CONTENTS INCORRECT AFTER DMA (1 TO 2)
E232	NO INTERRUPT AFTER DMA
E233	NO INTERRUPT AFTER READ FROM A NXM ADDRESS
E234	BAR CONTENTS INCORRECT
E235	BCR CONTENTS INCORRECT

#### ERROR REPORTS:

CALL	MESSAGE
ERR101	REGISTER AT (AAAAAA) DOES NOT RESPOND
ERR201	REGISTER: CSR,CHA.:(CC) ,GOOD DATA:NNNN,BAD DATA:NNNN (FOR IEU11A IGNORE BIT 9-12)
ERR202	REGISTER: (RRR),CHA.:(CC) ,GOOD DATA:NNNN,BAD DATA:NNN
ERR401	CHAN.:(CC),CORRECT PRIORITY:(PPP) ,WRONG PRIORITY:(PPP)
ERR402	CHANNEL :(CC) IS SELECTED
ERR501	CHAN.:(CC),GOOD DATA:NNNN ,BAD DATA:NNNN,ITERATION:NN
ERR231	GOOD DATA TXADDR BAD DATA RXADDR BYTE CNT# DDDDDD AAAAAA DDDDD AAAAA CCCCC

### 4.0 PERFORMANCE AND PROGRESS REPORTS

---

AT THE END OF EACH PASS, THE PASS COUNT IS GIVEN ALONG WITH THE TOTAL NUMBER OF ERRORS REPORTED SINCE THE DIAGNOSTIC WAS STARTED. THE "EOP" SWITCH CAN BE USED TO CONTROL HOW OFTEN THE END OF PASS MESSAGE IS PRINTED. SECTION 2.2 DESCRIBES SWITCHES.

### 5.0 DEVICE INFORMATION TABLES

---



## 6.0 TEST SUMMARIES

-----

### 6.1 TEST 1 - IEX11 : REGISTER ADDRESSING TEST

---

VERIFY THAT ADDRESSING THE 8 BUS DEVICE REGISTERS DOES NOT CAUSE A NON-EXISTENT MEMORY TRAP.  
 AN ERROR IN THIS TEST COULD MEAN THAT THE DEVICE IS INCORRECTLY CONFIGURED OR THAT THE ADDRESS IS WRONG.  
 COMMUNICATION BETWEEN THE MAIN CPU AND THE IEX11 IS ACCOMPLISHED THROUGH A SET OF SIXTEEN REGISTERS. THE SIXTEEN REGISTERS ARE ASSIGNED ADDRESSES IN THE I/O PAGE.

### 6.2 IEX - TEST 2 : INITIALIZATION TEST

---

RESETS THE IEX AND ENSURES THAT REGISTERS CSR, IIR, ISR, ICR, IDR, MCR IN BOTH CHANNELS ARE IN THEIR PROPER INITIALIZATION STATE.  
 REGISTERS BAR AND BCR ARE NOT IN A DEFINITIVE STATE AFTER RESET SO THEY ARE NOT TESTED HERE.  
 THE MUX BIT IN CSR IS ALSO TESTED.

### 6.3 IEX - TEST 3 : R/W BIT TEST

---

THIS TEST CHECKS ALL R/W BITS OF CSR, BAR, BCR AND MCR REGISTERS IN BOTH CHANNELS. IT ALSO TESTS THE MASTER CLEAR FUNCTION IN CSR1 + CSR2.  
 THE TMS 9914 REGISTERS IIR, ISR, ICR, IDR ARE NOT CHECKED IN THIS TEST.

### 6.4 IEX - TEST 4 : SYSTEM CONTROLLER COMMANDS TEST

---

PART 1 CHANNEL 1 WHICH IS SELECTED AS SYSTEM CONTROLLER, SENDS THE IFC AND REN MESSAGE BY MEANS OF THE AUXILIARY COMMANDS SIC AND SRE.  
 ALSO, BOTH IIR AS WELL AS ISR REGISTERS ARE CHECKED.  
 PART 2 CHANNEL 2 WHICH IS SELECTED AS SYSTEM CONTROLLER, SENDS THE IFC AND REN MESSAGE BY MEANS OF THE AUXILIARY COMMANDS SIC AND SRE.  
 ALSO BOTH IIR AS WELL AS ISR REGISTERS ARE CHECKED.

### 6.5 IEX - TEST 5 : INTERRUPT TEST

---

PART 1 CHECKS THE DEVICE PRIORITY LEVEL AND THE FUNCTION OF INTERRUPT SEQUENCE IN CHANNEL 1.  
 INITIATING THIS SEQUENCE WILL BE DONE BY SETTING THE INT ENB, DMA ENB BITS IN CSR1 AND BO BIT IN IIR 1 REGISTER.  
 PART 2 CHECKS THE DEVICE PRIORITY LEVEL AND THE FUNCTION OF INTERRUPT SEQUENCE IN CHANNEL 2.  
 INITIATING THIS SEQUENCE WILL BE DONE BY SETTING THE INT ENB, DMA ENB

BITS IN CSR2 AND BO BIT IN IIR 2 REGISTER.

6.6 IEX - TEST 6 : ADDRESS REGISTER TEST (ICR) OF CHANNEL 1

---  
 PART 1 CHECKS THE CORRECT FUNCTION OF ADDRESS REGISTER 1 (ADR)  
 BY LOADING ITS DEVICE PRIMARY ADDRESS INTO BIT A1-A5 AND RECEIVING THE  
 ASSIGNED LISTEN OR TALKER ADDRESS VIA THE IEC/IEEE BUS.  
 NOTE: THE ULPA BIT IN THE ISR1 REGISTER IS DEPENDENT  
 FROM THE STATUS OF DPA1 (DPA1=ULPA IS SET)  
 PART 2 CHECKS THE FUNCTION OF THE "DAT", "DAL" AND "EDPA" BIT OF ADR1 REGISTER  
 IF THE QUICK VERIFY PASS IS NOT SELECTED, THE TEST ITERATION WILL DO  
 IT WITH DIFFERENT DPA'S.

6.7 IEX - TEST 7 : ADDRESS REGISTER TEST (ICR) OF CHANNEL 2

---  
 THIS TEST IS THE SAME TEST AS TEST 6 EXCEPT THE CHANNEL IS CHANGED  
 PART 1 CHECKS THE CORRECT FUNCTION OF ADDRESS REGISTER 2 (ADR)  
 BY LOADING ITS DEVICE PRIMARY ADDRESS INTO BIT A1-A5 AND RECEIVING THE  
 ASSIGNED LISTEN OR TALKER ADDRESS VIA THE IEC/IEEE BUS.  
 NOTE: THE ULPA BIT IN THE ISR2 REGISTER IS DEPENDENT  
 FROM THE STATUS OF DPA2 (DPA2=ULPA IS SET)  
 PART 2 CHECKS THE FUNCTION OF THE "DAT", "DAL" AND "EDPA" BIT OF ADR2 REGISTER  
 IF THE QUICK VERIFY PASS IS NOT SELECTED, THE TEST ITERATION WILL DO  
 IT WITH DIFFERENT DPA'S.

6.8 IEX - TEST 8 : DATA TRANSFER TEST

---  
 THIS TEST IS DIVIDED INTO TWO PARTS.  
 IT CHECKS THE DATA OUT (DOR) AND DATA IN (DIR) REGISTERS.  
 PART 1 CHECKS DOR AND DIR REGISTERS BY LOADING THE DOR1 WITH A DATA BYTE  
 AND READING IT FROM THE DIR2 (PROGRAMMED DATA TRANSFER FROM  
 CHAN.1 TO CHAN.2).  
 PART 2 CHECKS DOR AND DIR REGISTERS BY LOADING THE DOR2 WITH A DATA BYTE  
 AND READING IT FROM THE DIR1 (PROGRAMMED DATA TRANSFER FROM  
 CHAN.2 TO CHAN.1).  
 IF THE QUICK VERIFY PASS IS NOT SELECTED, THE TEST ITERATION IS  
 CARRIED OUT WITH A DIFFERENT DATA PATTERN

6.9 IEX - TEST 9 : SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)

---  
 THIS TEST CHECKS THE EXTENDED LISTENER INTERFACE FUNCTION .  
 PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING  
 FEATURE OF CHANNEL 1 BY MEANS OF RECEIVING A VALID AS WELL AN INVALID MY  
 SECONDARY ADDRESS (MSA1)

6.10 IEX - TEST 10 : SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)

---



PARAMETER CODING

MACRO M1113 06-SEP-82 16:46 PAGE 13-2

THIS TEST CHECKS THE EXTENDED TALKER INTERFACE FUNCTION.  
PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING  
FEATURE OF CHANNEL 1 BY MEANS OF RECEIVING A VALID AS WELL AN INVALID MY  
SECONDARY ADDRESS (MSA1)

6.11 IEX - TEST 11 : SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)  
---

THIS TEST CHECKS THE EXTENDED LISTENER INTERFACE FUNCTION .  
PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING  
FEATURE OF CHANNEL 2 BY MEANS OF RECEIVING A VALID AS WELL AN INVALID MY  
SECONDARY ADDRESS (MSA2)

6.12 IEX - TEST 12 : SECONDARY ADDRESSING TEST OF CHANNEL 2 (TALKER)  
---

THIS TEST CHECKS THE EXTENDED TALKER INTERFACE FUNCTION .  
PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING  
FEATURE OF CHANNEL 2 BY MEANS OF RECEIVING A VALID AS WELL AN INVALID MY  
SECONDARY ADDRESS (MSA2)

6.13 IEX - TEST 13 : DEVICE CLEAR INTERFACE FUNCTION TEST  
---

PART 1 CHECKS THE DEVICE CLEAR INTERFACE FUNCTION OF CHANNEL 2 BY MEANS OF  
RECEIVING A UNIVERSAL COMMAND (DCL) AS WELL AS AN ADDRESS COMMAND  
(SDC)  
PART 2 CHECKS THE DEVICE CLEAR INTERFACE FUNCTION OF CHANNEL 1 BY MEANS OF  
RECEIVING A UNIVERSAL COMMAND (DCL) AS WELL AS AN ADDRESS COMMAND  
(SDC)

6.14 IEX - TEST 14 : DEVICE TRIGGER INTERFACE FUNCTION TEST  
---

PART 1 CHECKS THE TRIGGER INTERFACE FUNCTION OF CHANNEL 2 BY MEANS OF  
RECEIVING THE ADDRESS COMMAND GET AS WELL AS THE AUXILIARY  
COMMAND NOT FGET.  
PART 2 CHECKS THE TRIGGER INTERFACE FUNCTION OF CHANNEL 1 BY MEANS OF  
RECEIVING THE ADDRESS COMMAND GET AS WELL AS THE AUXILIARY  
COMMAND NOT FGET.

6.15 IEX - TEST 15 : INCOMPLETE SOURCE HANDSHAKE TEST  
---

PART 1 CHECKS THE INCOMPLETE SOURCE HANDSHAKE OF CHANNEL 1.  
SOURCE HANDSHAKE DOES NOT OCCUR DURING THE DATA TRANSFER,  
BECAUSE CHANNEL 2 IS NOT SELECTED AS LISTENER.  
PART 2 CHECKS THE INCOMPLETE SOURCE HANDSHAKE OF CHANNEL 2.  
SOURCE HANDSHAKE DOES NOT OCCUR DURING THE DATA TRANSFER,  
BECAUSE CHANNEL 1 IS NOT SELECTED AS LISTENER.

6.16 IEX - TEST 16 : CHANGING OF THE CONTROLLER CONFIGURATION

- PART 1 CHECKS THE CHANGING OF THE CONTROLLER CONFIGURATION FROM 1 TO 2  
BY MEANS OF THE AUXILIARY COMMANDS RQC AND RLC.  
PART 2 CHECKS THE CHANGING OF THE CONTROLLER CONFIGURATION FROM 2 TO 1  
BY MEANS OF THE AUXILIARY COMMANDS RQC AND RLC.

6.17 IEX - TEST 17 : REMOTE/LOCAL INTERFACE FUNCTION TEST

- PART 1 CHECKS THE REMOTE/LOCAL FUNCTION OF CHANNEL 2 USING THE FOLLOWING  
COMMANDS GTL, LLO, NOT RTL.  
PART 2 CHECKS THE REMOTE/LOCAL FUNCTION OF CHANNEL 1 USING THE FOLLOWING  
COMMANDS GTL, LLO, NOT RTL.

6.18 IEX - TEST 18 : SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1

- THIS TEST CHECKS THE SERIAL POLL REGISTER OF CHANNEL 1  
PART 1 SETS AND CLEARS THE RSV BIT IN SPR REGISTER OF CHANNEL 1  
AND CHECKS THE SRQ BIT IN ISR2.  
PART 2 CHECKS THE SERIAL POLL SEQUENCE OF CHANNEL 1.  
IF QUICK VERIFY PASS IS NOT SELECTED, THE SERIAL POLL SEQUENCE IS CARRIED  
OUT WITH DIFFERENT DATA.

6.19 IEX - TEST 19 : SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 2

- THIS TEST CHECKS THE SERIAL POLL REGISTER OF CHANNEL 2.  
PART 1 SETS AND CLEARS THE RSV BIT IN SPR REGISTER OF CHANNEL 2 AND  
CHECKS THE SRQ BIT IN ISR1.  
PART 2 CHECKS THE SERIAL POLL SEQUENCE OF CHANNEL 2.  
IF QUICK VERIFY PASS IS NOT SELECTED, THE SERIAL POLL SEQUENCE IS CARRIED  
OUT WITH DIFFERENT DATA.

6.20 IEX - TEST 20 : PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1

- PART 1 CHECKS PARALLEL POLL SEQUENCE (LOCAL CONFIGURED).  
PART 2 CHECKS PARALLEL POLL SEQUENCE (REMOTE CONFIGURED).

6.21 IEX - TEST 21 : PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 2

- PART 1 CHECKS PARALLEL POLL SEQUENCE (LOCAL CONFIGURED).  
PART 2 CHECKS PARALLEL POLL SEQUENCE (REMOTE CONFIGURED).



6.22 IEX - TEST 22 : END OF A MESSAGE BLOCK TEST

- 
- PART 1 CHECKS THE END OF A MESSAGE BLOCK FROM CHANNEL 1. CHANNEL 2 SENDS THE  
EOI MESSAGE VIA THE IEC/IEEE BUS.
- PART 2 CHECKS THE END OF A MESSAGE BLOCK FROM CHANNEL 2. CHANNEL 1 SENDS THE  
EOI MESSAGE VIA THE IEC/IEEE BUS.

6.23 IEX - TEST 23 : DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

- 
- PART 1 SENDS DATA VIA THE IEC/IEEE BUS FROM CHAN. 1 TO 2 BY MEANS OF A DMA  
I.E. CHAN. 1 WHICH IS SELECTED AS TALKER PERFORMS A DATI CYCLE,  
WHEREAS CHAN.2 WHICH IS SELECTED AS A LISTENER PERFORMS A DATOB CYCLE.  
THE MAX. SELECTABLE BYTE COUNT FOR THIS DATA TRANSFER IS 2K BYTES  
AND THE HIGHEST BUS ADDRESS IS BELOW 32 K.
- PART 2 CHECKS THE NON EXISTENT MEMORY BIT OF CHANNEL 1.  
THIS IS DONE BY A DMA FROM A NON EXISTING I/O PAGE ADDRESS  
SELECTED IN THE BUS ADDRESS REGISTER OF CHAN 1 (DATI CYCLE).
- PART 3 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED  
OVER 32K (IF MEMORY MANAGEMENT IS AVAILABLE).
- PART 4 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED  
OVER 64K (IF MEMORY MANAGEMENT IS AVAILABLE).

6.24 IEX - TEST 24 : DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

- 
- PART 1 SENDS DATA VIA THE IEC/IEEE BUS FROM CHAN. 2 TO 1 BY MEANS OF A DMA  
I.E. CHAN. 2 WHICH IS SELECTED AS TALKER PERFORMS A DATI CYCLE,  
WHEREAS CHAN.1 WHICH IS SELECTED AS A LISTENER PERFORMS A DATOB CYCLE.  
THE MAX. SELECTABLE BYTE COUNT FOR THIS DATA TRANSFER IS 2K BYTES  
AND THE HIGHEST BUS ADDRESS IS BELOW 32 K.
- PART 2 CHECKS THE NON EXISTENT MEMORY BIT OF CHANNEL 1  
THIS IS DONE BY A DMA FROM A NON EXISTING I/O PAGE ADDRESS  
SELECTED IN THE BUS ADDRESS REGISTER OF CHAN 2 (DATOB CYCLE).
- PART 3 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED  
OVER 32K (IF MEMORY MANAGEMENT IS AVAILABLE).
- PART 4 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED  
OVER 64K (IF MEMORY MANAGEMENT IS AVAILABLE).

6.25 IEX - TEST 25 : MCR FUNCTION TEST OF CHANNEL 1

- 
- PART 1 CHANNEL 2 TRANSMITS 9 DATA BYTES (50) THAN  
A PREDEFINED QUANTITY (MC INPUT) OF SUCCESSIVE EOS CHARACTERS (177)  
VIA THE IEC/IEEE BUS TO CHANNEL 1. AFTER RECEIVING THESE CHARACTERS  
THE DMA DATA TRANSFER IS TERMINATED BY CHANNEL 1 (COMP END).
- PART 2 SAME AS PART 1 EXCEPT THAT A WRONG QUANTITY OF SUCCESSIVE EOS  
CHARACTERS ARE TRANSMITTED BEFORE THE CORRECT QUANTITY OF SUCCESSIVE  
EOS CHARACTER ARE TRANSMITTED.  
I.E. 2 EOS CHAR.(25),1 DATA BYTE (50) THAN THE PREDEFINED EOS CHAR.(25).

PARAMETER CODING

MACRO M1113 06-SEP-82 16:46 PAGE 13-5

6.26 IEX - TEST 26 MCR FUNCTION TEST OF CHANNEL 2  
 ----

- PART 1 CHANNEL 1 TRANSMITS 9 DATA BYTES (50) THAN  
 A PREDEFINED QUANTITY (MC INPUT) OF SUCCESSIVE EOS CHARACTERS (177)  
 VIA THE IEC/IEEE BUS TO CHANNEL 2. AFTER RECEIVING THESE CHARACTERS  
 THE DMA DATA TRANSFER IS TERMINATED BY CHANNEL 2 (COMP END).
- PART 2 SAME AS PART 1 EXCEPT THAT A WRONG QUANTITY OF SUCCESSIVE EOS  
 CHARACTERS ARE TRANSMITTED BEFORE THE CORRECT QUANTITY OF SUCCESSIVE  
 EOS CHARACTER ARE TRANSMITTED.  
 I.E. 2 EOS CHAR.(25),1 DATA BYTE (50) THAN THE PREDEFINED EOS CHAR.(25).

6.27 IEX - TEST 27 EXTENDED ADDRESS BIT (Q22-BUS) TEST  
 ----

THIS TEST IS ONLY BE CARRIED OUT IF A Q-BUS IS USED AND IF THE AVAILABLE  
 MEMORY IS GREATER THAN 128K (Q22-BUS).

- PART 1 FINDS OUT IF THE AVAILABLE MEMORY IS GREATER THAN 128K.  
 IF YES ,THEN A DMA IS CARRIED OUT BY SENDING DATA VIA THE  
 IEC/IEEE BUS FROM CHANNEL 1 TO 2.  
 THE SEQUENCE DESCRIBED ABOVE IS ALSO EXECUTED WITH 256K (BA 19 SET),  
 512K (BA 20 SET)AND 1024K (BA 21 SET).
- PART 2 SAME AS PART 1 EXCEPT THE CHANNELS .THE DMA DATA TRANSFER IS  
 CARRIED OUT FROM CHANNEL 2 TO 1.

6.28 IEX - TEST 28 ADDITIONAL STANDBY TEST  
 ----

THIS TEST CAN BE USED IF YOU WANT TO CHECK THE EXTENDED ADDRESS BITS  
 WITHOUT MEMORY.  
 IT MOVES A SLIDING ONE'S BIT PATTERN ACROSS THE ADDRESS LINE 16,17,  
 18,19,20,21 IGNORING NXM ERRORS BUT CHECKING THE ADDRESS REGISTER LINES TO  
 THE BUS.

THE PATTERN SHOULD BE CHECKED ON A LOGIC ANALYSER .  
 THE LOGIC ANALYSER HAS TO BE CONNECT TO THE ADDRESS LINES 16-21  
 THE TRIGGER HAS TO BE CONNECT TO THE SIGNAL ADREN L(E9, PIN 4).

THIS TEST IS ONLY CARRIED OUT IF A Q-BUS IS USED AND IF YOU ANSWER  
 THE SOFTWARE QUESTION.

7.0 PROGRAM LISTING



```

1011 .TITLE PROGRAM HEADER AND TABLES
1012 .SBTTL PROGRAM HEADER
1038
1040 000000 .ENABL AMA,ABS
1041 002000 = 2000
1043
1044
1045 002000 BGNMOD
1046
1047 :++
1048 : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
1049 : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
1050 :--
1051
1052 002000 POINTER BGNDU,BGNAU,BGNSETUP,BGNSW,BGNSFT,ERRTBL
1053
1054
1071
1072 002000
002000 HEADER CZIEA,B,0,0,0
002000 103 L$NAME:: ;DIAGNOSTIC NAME
002001 132 .ASCII /C/
002002 111 .ASCII /Z/
002003 105 .ASCII /I/
002004 101 .ASCII /E/
002005 000 .ASCII /A/
002006 000 .BYTE 0
002007 000 .BYTE 0
002010 L$REV:: ;REVISION LEVEL
002010 102 .ASCII /B/
002011 L$DEPO:: ;0
002011 060 .ASCII /O/
002012 L$UNIT:: ;NUMBER OF UNITS
002012 000001 .WORD T$PTHV
002014 L$TIML:: ;LONGEST TEST TIME
002014 000000 .WORD 0
002016 L$HPCP:: ;POINTER TO H.W. QUES.
002016 076724 .WORD L$HARD
002020 L$SPCP:: ;POINTER TO S.W. QUES.
002020 077336 .WORD L$SOFT
002022 L$HPTP:: ;PTR. TO DEF. H.W. PTABLE
002022 002216 .WORD L$HW
002024 L$SPTP:: ;PTR. TO S.W. PTABLE
002024 002234 .WORD L$SW
002026 L$LADP:: ;DIAG. END ADDRESS
002026 100004 .WORD L$LAST
002030 L$STA:: ;RESERVED FOR APT STATS
002030 000000 .WORD 0
002032 L$CO:: .WORD 0
002032 000000 .WORD 0
002034 L$DTYP:: ;DIAGNOSTIC TYPE
002034 000000 .WORD 0
002036 L$APT:: ;APT EXPANSION
002036 000000 .WORD 0
002040 L$DTP:: ;PTR. TO DISPATCH TABLE
002040 002124 .WORD L$DISPATCH
002042 L$PRIO:: ;DIAGNOSTIC RUN PRIORITY

```

PROGRAM HEADER AND TABLES  
PROGRAM HEADER

MACRO M1113 06-SEP-82 16:46 PAGE 14-1

002042 000000  
 002044 000000  
 002046 000000  
 002050 000000  
 002050 003  
 002051 003  
 002052 000000  
 002054 000000  
 002056 000000  
 002060 003352  
 002062 000000  
 002064 000000  
 002066 000000  
 002070 012422  
 002072 012340  
 002074 000000  
 002076 003414  
 002100 104035  
 002102 002504  
 002104 011120  
 002106 012314  
 002110 012230  
 002112 011112  
 002114 000000  
 002116 000000  
 002120 000000

L\$ENVI:: .WORD 0 ;FLAGS DESCRIBE HOW IT WAS SETUP  
 L\$EXP1:: .WORD 0 ;EXPANSION WORD  
 L\$MREV:: .WORD 0 ;SVC REV AND EDIT #  
 L\$EF:: .BYTE C\$REVISION ;DIAG. EVENT FLAGS  
           .BYTE C\$EDIT  
 L\$SPC:: .WORD 0  
 L\$DEVP:: .WORD 0 ; POINTER TO DEVICE TYPE LIST  
 L\$REPP:: .WORD L\$DVTYP ;PTR. TO REPORT CODE  
 L\$EXP4:: .WORD 0  
 L\$EXP5:: .WORD 0  
 L\$AUT:: .WORD 0 ;PTR. TO ADD UNIT CODE  
 L\$DUT:: .WORD L\$AU ;PTR. TO DROP UNIT CODE  
 L\$LUN:: .WORD L\$DU ;LUN FOR EXERCISERS TO FILL  
 L\$DESP:: .WORD 0 ;POINTER TO DIAG. DESCRIPTION  
 L\$LOAD:: .WORD L\$DESC ;GENERATE SPECIAL AUTOLOAD EMT  
           EMT E\$LOAD  
 L\$ETP:: .WORD L\$ERRTBL ;POINTER TO ERRTBL  
 L\$ICP:: .WORD L\$INIT ;PTR. TO INIT CODE  
 L\$CCP:: .WORD L\$CLEAN ;PTR. TO CLEAN-UP CODE  
 L\$ACP:: .WORD L\$AUTO ;PTR. TO AUTO CODE  
 L\$PRT:: .WORD L\$SAUTO ;PTR. TO PROTECT TABLE  
 L\$SPRT:: .WORD L\$SPROT  
 L\$TEST:: .WORD 0 ;TEST NUMBER  
 L\$DLY:: .WORD 0 ;DELAY COUNT  
 L\$HIME:: .WORD 0 ;PTR. TO HIGH MEM



PROGRAM HEADER AND TABLES  
DISPATCH TABLE

MACRO M1113 06-SEP-82 16:46 PAGE 15

1085  
 1086  
 1087  
 1088  
 1089  
 1090  
 1091  
 1092 002122  
 002122 000034  
 002124  
 002124 012430  
 002126 012704  
 002130 013424  
 002132 014076  
 002134 015574  
 002136 016762  
 002140 023062  
 002142 027416  
 002144 031764  
 002146 033550  
 002150 035316  
 002152 037066  
 002154 040744  
 002156 043116  
 002160 044022  
 002162 044776  
 002164 046344  
 002166 050722  
 002170 052032  
 002172 053142  
 002174 054212  
 002176 055262  
 002200 057250  
 002202 063270  
 002204 067270  
 002206 071012  
 002210 072544  
 002212 075456

.SBTTL DISPATCH TABLE

```

:++
: THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
: IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
:--

```

```

DISPATCH 28
.L$DISPATCH::
.WORD 28
.WORD T1
.WORD T2
.WORD T3
.WORD T4
.WORD T5
.WORD T6
.WORD T7
.WORD T8
.WORD T9
.WORD T10
.WORD T11
.WORD T12
.WORD T13
.WORD T14
.WORD T15
.WORD T16
.WORD T17
.WORD T18
.WORD T19
.WORD T20
.WORD T21
.WORD T22
.WORD T23
.WORD T24
.WORD T25
.WORD T26
.WORD T27
.WORD T28

```

1093

PROGRAM HEADER AND TABLES  
DEFAULT HARDWARE P-TABLE

MACRO M1113 06-SEP-82 16:46 PAGE 16

```

1101          .SBTTL  DEFAULT HARDWARE P-TABLE
1102
1103          :++
1104          : THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
1105          : THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
1106          : IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
1107          : AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
1108          :--
1109
1110          002214          BGNHW  DFPTBL
              002214          .WORD  L10000-L$HW/2
              002216          000006
              002216
              DFPTBL::
1111
1121          002216          160140          .WORD  160140          :1ST (OF 8) REGISTER ADDRESS
1122          002220          000420          .WORD  420          :1ST (OF 2) VECTOR ADDRESS
1123          002222          000300          .WORD  PRI06          :DEVICE PRIORITY LEVEL
1124          002224          000000          .WORD  0          :DEVICE PRIMARY ADDRESS FOR CH.1
1125          002226          000001          .WORD  1          :DEVICE PRIMARY ADDRESS FOR CH.2
1126          002230          000000          .WORD  0          :DEFAULT VALUE FOR TESTCABLE
1127
1128
1129          002232          ENDPHW
              002232

```

L10000:



PROGRAM HEADER AND TABLES  
SOFTWARE P-TABLE

MACRO M1113 06-SEP-82 16:46 PAGE 18

```

1132          .SBTTL  SOFTWARE P-TABLE
1133
1134          :++
1135          : THE SOFTWARE TABLE CONTAINS VARIOUS DATA USED BY THE
1136          : PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE
1137          : SET UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR
1138          : AT RUN TIME.
1139          :--
1140 002232          BGNSW  SFPTBL
          002232          .WORD  L10001-L$SW/2
          002234          000004
          002234
1141
1149
1150 002234          000001          QVP::  .WORD  1          ;QUICK VERIFY SWITCH
1151 002236          000077          MCINP:: .WORD  63.         ;NUMBER OF DEFAULT MATCH CHARACTER COUNTS
1152 002240          003777          BCINP:: .WORD  2047.       ;NUMBER OF DEFAULT BYTE COUNTS IS 2047 DECIMAL
1153 002242          000000          MAINB:: .WORD  0          ;STANDBY TEST 28,DEFAULT IS NO
1154 002244          002244          ENDSW
1155
1156 002244          ENDMOD
          L10001:

```

1159  
1170  
1171  
1199  
1209  
1210 002244  
1211  
1212  
1213  
1214  
1215  
1216  
1231  
1232 002244

.TITLE GLOBAL AREAS  
.SBTTL GLOBAL EQUATES SECTION

BGNMOD

::+  
: THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT  
: ARE USED IN MORE THAN ONE TEST.  
:--

EQUALS

::  
: BIT DIFINITIONS

100000	BIT15== 100000
040000	BIT14== 40000
020000	BIT13== 20000
010000	BIT12== 10000
004000	BIT11== 4000
002000	BIT10== 2000
001000	BIT09== 1000
000400	BIT08== 400
000200	BIT07== 200
000100	BIT06== 100
000040	BIT05== 40
000020	BIT04== 20
000010	BIT03== 10
000004	BIT02== 4
000002	BIT01== 2
000001	BIT00== 1

001000	BIT9== BIT09
000400	BIT8== BIT08
000200	BIT7== BIT07
000100	BIT6== BIT06
000040	BIT5== BIT05
000020	BIT4== BIT04
000010	BIT3== BIT03
000004	BIT2== BIT02
000002	BIT1== BIT01
000001	BIT0== BIT00

::  
: EVENT FLAG DEFINITIONS  
: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

000040	EF.START== 32.	: START COMMAND WAS ISSUED
000037	EF.RESTART== 31.	: RESTART COMMAND WAS ISSUED
000036	EF.CONTINUE== 30.	: CONTINUE COMMAND WAS ISSUED
000035	EF.NEW== 29.	: A NEW PASS HAS BEEN STARTED
000034	EF.PWR== 28.	: A POWER-FAIL/POWER-UP OCCURRED

::  
: PRIORITY LEVEL DEFINITIONS



GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 19-1  
GLOBAL EQUATES SECTION

000340	PRI07==	340
000300	PRI06==	300
000240	PRI05==	240
000200	PRI04==	200
000140	PRI03==	140
000100	PRI02==	100
000040	PRI01==	40
000000	PRI00==	0
	:	
	:	OPERATOR FLAG BITS
	:	
000004	EVL==	4
000010	LOT==	10
000020	ADR==	20
000040	IDU==	40
000100	ISR==	100
000200	UAM==	200
000400	BOE==	400
001000	PNT==	1000
002000	PRI==	2000
004000	IXE==	4000
010000	IBE==	10000
020000	IER==	20000
040000	LOE==	40000
100000	HOE==	100000

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 20  
GLOBAL DATA SECTION

```

1234          .SBTTL GLOBAL DATA SECTION
1235
1236          :++
1237          : THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1238          : IN MORE THAN ONE TEST.
1239          :--
1240
1253          :*****
1254          : IEX11 VECTOR AND REGISTER INDIRECT POINTERS
1255          :*****
1256          VECC1::          .WORD 0          : INTERRUPT VECTOR FOR CHANNEL 1
1257          VECC2::          .WORD 0          : INTERRUPT VECTOR FOR CHANNEL 2
1258          IIRX::          .WORD 0          : POINTER TO IEEE INTERRUPT REGISTER
1259          IIRLX::         .WORD 0          : POINTER TO LOW BYTE OF IIR REGISTER
1260          IIRHX::         .WORD 0          : POINTER TO HIGH BYTE OF INTERRUPT REGISTER
1261          ISRX::          .WORD 0          : POINTER TO IEEE STATUS REGISTER
1262          ISRLX::         .WORD 0          : POINTER TO LOW BYTE OF ISR REGISTER
1263          ISRHX::         .WORD 0          : POINTER TO HIGH BYTE OF STATUS REGISTER
1264          ICRX::          .WORD 0          : POINTER TO IEEE COMMAND REGISTER
1265          ICRLX::         .WORD 0          : POINTER TO LOW BYTE OF ICR REGISTER
1266          ICRHX::         .WORD 0          : POINTER TO HIGH BYTE OF COMMAND REGISTER
1267          IDRX::          .WORD 0          : POINTER TO IEEE DATA REGISTER
1268          IDRLX::         .WORD 0          : POINTER TO LOW BYTE OF IDR REGISTER
1269          IDRHX::         .WORD 0          : POINTER TO HIGH BYTE OF DATA REGISTER
1270          CSRX::          .WORD 0          : POINTER TO CONTROL & STATUS REGISTER
1271          BARX::          .WORD 0          : POINTER TO BUS ADDRESS REGISTER
1272          BCRX::          .WORD 0          : POINTER TO BYTE COUNT REGISTER
1273          MCRX::          .WORD 0          : POINTER TO MATCH CHARACTER REGISTER
1274          MCRHX::         .WORD 0          : POINTER TO HIGH BYTE OF MCR REGISTER
1275          DPA1::          .WORD 0          : POINTER TO DEVICE PRIMARY ADDRESS FOR CH.1
1276          DPA2::          .WORD 0          : POINTER TO DEVICE PRIMARY ADDRESS FOR CH.2
1277          PLEV::         .WORD 0          : POINTER TO THE PRIORITY LEVEL
1278
1281          :*****
1282          : PROGRAM CONTROL PARAMETERS
1283          :*****
1284          ITRDEF::         .WORD 20         : ITERATION DEFAULT
1285          ITRCNT::         .WORD 0          : ITERATION COUNTER
1286
1287          :*****
1288          : PROGRAM VARIABLES
1289          :*****
1290          PNTF::          .WORD 0          : FLAG FOR TEST HEADER PRINTOUT
1291          NXMFLG::         .WORD 0          : FLAG USED WHEN ADDRESS IS NXM.
1292          MM22::          .WORD 0          : FLAG INDICATING 22 BIT MMU
1293          MMFLG::         .WORD 0          : FLAG TO SEE IF MEMORY MANAGEMENT THERE
1294          PHHIGH::        .WORD 0          : LOCATION FOR MEMORY SIZE
1295          PHLOW::         .WORD 0          : LOCATION FOR MEMORY SIZE
1296          VIADD::         .WORD 0          : LOCATION FOR VIRTUAL MEMORY SIZE
1297          PHHSIZ::        .WORD 0          : LOCATION FOR MEMORY SIZE
1298          PHSIZ::         .WORD 0          : LOCATION FOR MEMORY SIZE
1299          SIZEPA::        .WORD 0          : LOCATION FOR MEMORY SIZE IN PAGE FORM
1300          MASK::          .WORD 0          : BIT MASK OF READ/WRITE BITS
1301          MASCOM::        .WORD 0          : COMPLEMENT OF MASK
1302          REGADD::        .WORD 0          : ADDRESS OF REGISTER TO BE TESTED

```



```

1303 002356 000000 BUFAB:: .WORD 0 ;LOCATION FOR START ADDR. OF BUFFER A
1304 002360 000000 BUFBB:: .WORD 0 ;LOCATION FOR START ADDR. OF BUFFER B
1305 002362 000000 CSRMSK:: .WORD 0 ;WORK LOCATION USED IN TEST 27
1306 002364 000000 CSRMS1:: .WORD 0 ;WORK LOCATION USED IN TEST 27
1307 002366 000000 CSRMS2:: .WORD 0 ;WORK LOCATION USED IN TEST 27
1308 002370 000000 ANS:: .WORD 0 ;STORE FOR OPERATOR ANSWER USED IN T28
1309 002372 000000 LOGDEV:: .WORD 0 ;LOGICAL DEVICE NUMBER
1310 002374 000000 CHAN:: .WORD 0 ;FLAG FOR CHANNEL
1311 002376 000000 INTFC1:: .WORD 0 ;INTERRUPT FLAG FOR CHA.1.
1312 002400 000000 INTFC2:: .WORD 0 ;INTERRUPT FLAG FOR CHA.2
1313 002402 000000 RSAVE:: .WORD 0 ;TEMPORARY LOCATION TO SAVE DATA
1314 002404 000000 CNT1:: .WORD 0 ;COUNTER USED IN TEST 23-26
1315 002406 000000 SDPA:: .WORD 0 ;TEMPORARY STORE TO SAVE DEVICE PRIM. ADDR.
1316 002410 000000 MLA1:: .WORD 0 ;STORE TO SAVE MY LISTENER ADDRESS
1317 002412 000000 MLA2:: .WORD 0 ;STORE TO SAVE MY LISTENER ADDRESS
1318 002414 000000 MTA1:: .WORD 0 ;STORE TO SAVE MY TALKER ADDRESS CH.1
1319 002416 000000 MTA2:: .WORD 0 ;STORE TO SAVE MY TALKER ADDRESS CH.2
1320 002420 000000 MSA1:: .WORD 0 ;STORE TO SAVE MSA
1321 002422 000000 RXADRH:: .WORD 0 ;LOCATION FOR RX HIGH ADDRESS
1322 002424 000000 RXADRL:: .WORD 0 ;LOCATION FOR RX LOW ADDRESS
1323 002426 000000 TXADRH:: .WORD 0 ;LOCATION FOR DMA TX HIGH ADDRESS
1324 002430 000000 TXADRL:: .WORD 0 ;LOCATION FOR DMA TX LOW ADDRESS
1325 002432 000000 ERNU:: .WORD 0 ;LINE COUNTER FOR ERROR PRINTOUT
1326 002434 000000 CDAT1:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1327 002436 000000 CDAT2:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1328 002440 000000 CDAT3:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1329 002442 000000 CDAT4:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1330 002444 000000 CDAT5:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1331 002446 000000 CDAT6:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1332 002450 000000 CDAT7:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1333 002452 000000 CDAT8:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1334 002454 000000 CDAT9:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1335 002456 000000 CDAT10:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1336 002460 000000 CDAT11:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1337 002462 000000 CDAT12:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1338 002464 000000 CDAT13:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1339 002466 000000 CDAT14:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1340 002470 000000 CDAT15:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1341 002472 000000 CDAT16:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1342 002474 000000 CDAT17:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1343 002476 000000 CDAT18:: .WORD 0 ;VARIABLE COMPARE DATA FOR ISR REGISTER
1344
1345
1346 ;*****
;ERROR VARIABLES
1347 ;*****
1348 002500 000000 GOOD:: .WORD 0 ;WORD USED FOR ERROR PRINTOUT
1349 002502 000000 BAD:: .WORD 0 ;WORD USED FOR ERROR PRINTOUT
1350 002504 000000 ERRTBL
;L$ERRTBL::
002504 000000 ERRTP:: .WORD 0
002506 000000 ERRNBR:: .WORD 0
002510 000000 ERRMSG:: .WORD 0
002512 000000 ERRBLK:: .WORD 0
1351
1352 ;*****
1353 ;TABLE AREA FOR MCR FUNTIONS (TEST 25,26)
1354 ;*****

```

```

1355
1356 002514 050 050 050 TABD:: .NLIST BEX .BYTE 50,50,50,50,50,50,50,50,50,50 :9DATA,63 EOS
1357 002525 125 125 125 .BYTE 125,125,125,125,125,125,125,125,125,125
1358 002537 125 125 125 .BYTE 125,125,125,125,125,125,125,125,125,125
1359 002551 125 125 125 .BYTE 125,125,125,125,125,125,125,125,125,125
1360 002563 125 125 125 .BYTE 125,125,125,125,125,125,125,125,125,125
1361 002575 125 125 125 .BYTE 125,125,125,125,125,125,125,125,125,125
1362 002607 125 125 125 .BYTE 125,125,125,125,125,125,125,125,125,125
1363 002621 125 125 125 .BYTE 125,125,125
1364 .EVEN
1365
1366 002624 000007 TABE:: .REPT 7 :72 DATA BYTES
1367 .BYTE 0,0,0,0,0,0,0,0,0,0,0
1368 .ENDR
1369 002732 000 000 .BYTE 0,0
1370 .EVEN
1371
1372 002734 050 050 050 TABF:: .BYTE 50,50,50,50,50,50,50,50,50 :9 DATA,63 EOS
1373 000006 .REPT 6
1374 .BYTE 177,177,177,177,177,177,177,177,177,177
1375 .ENDR
1376 003041 177 177 177 .BYTE 177,177,177
1377 .EVEN
1378
1379 003044 012 012 050 TABG:: .BYTE 12,12,50,12,12,12 :2EOS,1DATA,63EOS
1380 000006 .REPT 6
1381 .BYTE 12,12,12,12,12,12,12,12,12,12
1382 .ENDR
1383 .EVEN
1384
1385 003146 000 000 000 TABH:: .BYTE 0,0,0,0,0,0 :66 DATA BYTES
1386 000006 .REPT 6
1387 .BYTE 0,0,0,0,0,0,0,0,0,0,0
1388 .ENDR
1389 .EVEN
1390
1391 003250 025 025 050 TABK:: .BYTE 25,25,50,25,25,25 :2 EOS,1 DATABYTE,63 EOS
1392 000006 .REPT 6
1393 .BYTE 25,25,25,25,25,25,25,25,25,25
1394 .ENDR
1395 .EVEN
1396 .LIST BEX
1397
1398

```



GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 21  
GLOBAL TEXT SECTION

1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412

003352			
003352			
003352	111	105	125
003355	061	061	040
003360	106	117	122
003363	040	125	116
003366	111	102	125
003371	123	040	134
003374	111	105	121
003377	061	061	040
003402	106	117	122
003405	040	121	055
003410	102	125	123
003413	000		

.SBTTL GLOBAL TEXT SECTION  
:++  
: THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,  
: MESSAGES, AND ASCII INFORMATION THAT ARE USED IN  
: MORE THAN ONE TEST.  
:--

:\*\*\*\*\*  
: NAMES OF DEVICES SUPPORTED BY PROGRAM  
:\*\*\*\*\*

DEV TYP <IEU11 FOR UNIBUS \IEQ11 FOR Q-BUS>  
L\$DVTYP::  
.ASCIZ /IEU11 FOR UNIBUS \IEQ11 FOR Q-BUS/

.EVEN

:\*\*\*\*\*  
: TITEL OF PROGRAM  
:\*\*\*\*\*

1413  
1419  
1420  
1421  
1422  
1423

003414			
003414			
003414	111	105	125
003417	134	111	105
003422	121	040	104
003425	111	101	107
003430	116	117	123
003433	124	111	103
003436	040	101	103
003441	055	124	060
003444	066	064	101
003447	055	115	103
003452	000		

DESCRIPT <IEU\IEQ DIAGNOSTIC AC-T064A-MC>  
L\$DESC::  
.ASCIZ /IEU\IEQ DIAGNOSTIC AC-T064A-MC/

.EVEN

.EVEN

1424  
1425  
1426  
1433  
1434  
1435  
1436  
1447  
1455  
1456

000001  
000001

:  
: FORMAT STATEMENTS USED IN PRINT CALLS  
:

SVCGBL= 1  
SVCINS= 1

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 22  
 GLOBAL ERROR REPORT SECTION

```

1458          .SBTTL GLOBAL ERROR REPORT SECTION
1459
1460          :++
1461          : THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
1462          : USED BY MORE THAN ONE TEST TO OUTPUT ADDITIONAL ERROR INFORMATION. PRINTB
1463          : (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
1464          :--
1465
1466
1467 003454          BGNMSG ERR101
1468 003454          PRINTB #EMG101,R1
1469 003454          010146
1470 003456          012746 004116
1471 003462          012746 000002
1472 003466          010600
1473 003470          104414
1474 003472          062706 000006
1475 003476          ENDMSG
1476 003476          104423
1477
1478
1479 003500          BGNMSG ERR201
1480 003500          PRINTB #EMG201,CHAN,GOOD,BAD
1481 003500          013746 002502
1482 003504          013746 002500
1483 003510          013746 002374
1484 003514          012746 004170
1485 003520          012746 000004
1486 003524          010600
1487 003526          104414
1488 003530          062706 000012
1489 003534          PRINTB #EMG203
1490 003534          012746 004265
1491 003540          012746 000001
1492 003544          010600
1493 003546          104414
1494 003550          062706 000004
1495 003554          ENDMSG
1496 003554          104423
1497
1498
1499 003556          BGNMSG ERR202
1500 003556          PRINTB #EMG202,R2,CHAN,GOOD,BAD
1501 003556          013746 002502
1502 003562          013746 002500
1503 003566          013746 002374
1504 003572          010246
1505 003574          012746 004330
1506 003600          012746 000005
1507 003604          010600
1508 003606          104414
1509 003610          062706 000014
  
```

ERR101::

```

MOV R1,-(SP)
MOV #EMG101,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP
  
```

L10002:

TRAP C\$MSC

ERR201::

```

MOV BAD,-(SP)
MOV GOOD,-(SP)
MOV CHAN,-(SP)
MOV #EMG201,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #12,SP
  
```

L10003:

```

MOV #EMG203,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #4,SP
  
```

L10003:

TRAP C\$MSG

ERR202::

```

MOV BAD,-(SP)
MOV GOOD,-(SP)
MOV CHAN,-(SP)
MOV R2,-(SP)
MOV #EMG202,-(SP)
MOV #5,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #14,SP
  
```



GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 22-1  
 GLOBAL ERROR REPORT SECTION

1495	003614			ENDMSG		L10004:	TRAP	C\$MSG
	003614							
	003614	104423						
1496								
1497	003616			BGNMSG	ERR401	ERR401::		
	003616							
1498	003616			PRINTB	#EMG401,CHAN,GOOD,BAD			
	003616	013746	002502				MOV	BAD,-(SP)
	003622	013746	002500				MOV	GOOD,-(SP)
	003626	013746	002374				MOV	CHAN,-(SP)
	003632	012746	004425				MOV	#EMG401,-(SP)
	003636	012746	000004				MOV	#4,-(SP)
	003642	010600					MOV	SP,RO
	003644	104414					TRAP	C\$PNTB
	003646	062706	000012				ADD	#12,SP
1499	003652			ENDMSG		L10005:	TRAP	C\$MSG
	003652							
	003652	104423						
1500								
1501	003654			BGNMSG	ERR402	ERR402::		
	003654							
1502	003654			PRINTB	#EMG402,CHAN			
	003654	013746	002374				MOV	CHAN,-(SP)
	003660	012746	004521				MOV	#EMG402,-(SP)
	003664	012746	000002				MOV	#2,-(SP)
	003670	010600					MOV	SP,RO
	003672	104414					TRAP	C\$PNTB
	003674	062706	000006				ADD	#6,SP
1503	003700			ENDMSG		L10006:	TRAP	C\$MSG
	003700							
	003700	104423						
1504								
1505	003702			BGNMSG	ERR501	ERR501::		
	003702							
1506	003702			PRINTB	#EMG501,CHAN,GOOD,BAD,ITRCNT			
	003702	013746	002322				MOV	ITRCNT,-(SP)
	003706	013746	002502				MOV	BAD,-(SP)
	003712	013746	002500				MOV	GOOD,-(SP)
	003716	013746	002374				MOV	CHAN,-(SP)
	003722	012746	004560				MOV	#EMG501,-(SP)
	003726	012746	000005				MOV	#5,-(SP)
	003732	010600					MOV	SP,RO
	003734	104414					TRAP	C\$PNTB
	003736	062706	000014				ADD	#14,SP
1507	003742			ENDMSG		L10007:	TRAP	C\$MSG
	003742							
	003742	104423						
1508								
1509	003744			BGNMSG	ERR231	ERR231::		
	003744							
1510	003744			PRINTB	#DMAHAD			
	003744	012746	004715				MOV	#DMAHAD,-(SP)
	003750	012746	000001				MOV	#1,-(SP)
	003754	010600					MOV	SP,RO
	003756	104414					TRAP	C\$PNTB
	003760	062706	000004				ADD	#4,SP
1511	003764	006337	002426	ASL	TXADRH			

1512	003770	006337	002422		ASL	RXADRH		
1513	003774	005737	002430		TST	TXADRL		
1514	004000	002005			BGE	1\$		
1515	004002	042737	100000	002430	BIC	#100000, TXADRL		
1516	004010	005237	002426		INC	TXADRH		
1517	004014	005737	002424		1\$: TST	RXADRL		
1518	004020	002005			BGE	2\$		
1519	004022	042737	100000	002424	BIC	#100000, RXADRL		
1520	004030	005237	002422		INC	RXADRH		
1521	004034				2\$: PRINTB	#EMG231, GOOD, TXADRH, TXADRL, BAD, RXADRH, RXADRL, CNT1		
	004034	013746	002404				MOV	CNT1, -(SP)
	004040	013746	002424				MOV	RXADRL, -(SP)
	004044	013746	002422				MOV	RXADRH, -(SP)
	004050	013746	002502				MOV	BAD, -(SP)
	004054	013746	002430				MOV	TXADRL, -(SP)
	004060	013746	002426				MOV	TXADRH, -(SP)
	004064	013746	002500				MOV	GOOD, -(SP)
	004070	012746	004651				MOV	#L'EMG231, -(SP)
	004074	012746	000010				MOV	#10, -(SP)
	004100	010600					MOV	SP, R0
	004102	104414					TRAP	C\$PNTB
	004104	062706	000022				ADD	#22, SP
1522	004110				ENDMSG			
	004110						L10010:	
	004110	104423					TRAP	C\$MSG
1523								
1524	004112				EXIT	MSG		
	004112	000167					.WORD	J\$JMP
	004114	177772					.WORD	L10010-2-
1525								
1526								
1527					:	EXTENDED ERROR MESSAGES		
1528					:			
1529					:			
1530	004116	045	123	063	EMG101:	.ASCIZ	/S3%AREGISTER AT %06% DOES NOT RESPOND%/	
1531	004170	045	101	122	EMG201:	.ASCIZ	/%AREGISTER: CSR, CHA.:%01% ,GOOD DATA:%06% ,BAD DATA:%06%/	
1532	004265	045	101	050	EMG203:	.ASCIZ	/%(FOR IEU11-A IGNORE BIT 9-12).%/	
1533	004330	045	101	122	EMG202:	.ASCIZ	/%AREGISTER:%T% ,CHA.:%01% ,GOOD DATA:%06% ,BAD DATA:%06%/	
1534	004425	045	101	103	EMG401:	.ASCIZ	/%ACHAN.:%01% ,CORRECT PRIORITY:%03% ,WRONG PRIORITY:%03%/	
1535	004521	045	101	103	EMG402:	.ASCIZ	/%ACHANNEL :%01% IS SELECTED%/	
1536	004560	045	101	103	EMG501:	.ASCIZ	/%ACHAN.:%01% ,GOOD :%06% ,BAD :%06% ,ITERATION :%03%/	
1537	004651	045	117	066	EMG231:	.ASCIZ	/%06%S6%03%05%S6%06%S6%03%05%S6%06%/	
1538	004715	045	101	107	MAHAD:	.ASCIZ	/%AGOOD DATA TXADDR BAD DATA RXADDR BYTE CNT#%/	
1539								
1540					:	ERROR MESSAGES		
1541					:			
1542					:			
1543	005012	040	122	105	E101::	.ASCIZ	/ REGISTER ADDRESSING ERROR - TRAP 4 /	
1544	005057	040	122	105	E200::	.ASCIZ	/ REGISTER INCORRECT AFTER BUS RESET /	
1545	005124	040	122	105	E301::	.ASCIZ	/ READ - WRITE BITS INCORRECT /	
1546	005162	040	102	111	E302::	.ASCIZ	/ BITS NOT CLEARED AFTER MASTER RESET /	
1547	005230	040	115	125	E303::	.ASCIZ	/ MUX BIT IN CSR NOT SETABLE /	
1548	005265	040	103	123	E401::	.ASCIZ	/ CSR CONTENTS INCORRECT /	
1549	005316	040	116	117	E402::	.ASCIZ	/ NO INTERRUPT WHEN EXPECTED /	
1550	005353	040	111	116	E403::	.ASCIZ	/ INCORRECT PRIORITY LEVEL /	
1551	005406	040	102	111	E501::	.ASCIZ	/ BITS IN IIR REGISTER INCORRECT /	
1552	005447	040	102	111	E502::	.ASCIZ	/ BITS IN ISR REGISTER INCORRECT /	

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 22-3  
 GLOBAL ERROR REPORT SECTION

1553	005510	040	104	101	E801::	.ASCIZ	/ DATA TRANSFER FROM CHANNEL 1 TO 2 INCORRECT /
1554	005566	040	104	101	E802::	.ASCIZ	/ DATA TRANSFER FROM CHANNEL 2 TO 1 INCORRECT /
1555	005644	040	111	103	E901::	.ASCIZ	/ ICR CONTENTS INCORRECT /
1556	005675	040	104	111	E222::	.ASCIZ	/ DIR CONTENTS INCORRECT /
1557	005726	040	122	130	E250::	.ASCIZ	/ RX BUFFER CONTENTS INCORRECT AFTER DMA (2 TO 1) /
1558	006010	040	122	130	E231::	.ASCIZ	/ RX BUFFER CONTENTS INCORRECT AFTER DMA (1 TO 2) /
1559	006072	040	116	117	E232::	.ASCIZ	/ NO INTERRUPT AFTER DMA /
1560	006123	040	116	117	E233::	.ASCIZ	/ NO INTERRUPT AFTER READ FROM A NXM ADDRESS /
1561	006200	040	102	101	E234::	.ASCIZ	/ BAR CONTENTS INCORRECT /
1562	006231	040	102	103	E235::	.ASCIZ	/ BCR CONTENTS INCORRECT /
1563							
1564						.EVEN	
1565						.LIST BEX	
1566							
1567							
1568							



1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603

.SBTTL LOCAL MACRO DEFINITIONS

:++  
:THIS SECTION CONTAINS ONLY MACROS WHICH ARE USED  
:SEPARATE FROM THE MACRO LIBRARY (SVC34R)(LIBA.MLB)  
:---

:\*\*\*\*\*  
:PRIT MACRO - THIS MACRO CHECKS IF INTERRUPT OCCURES  
:AND WHAT IS THE PRIORITY.  
:\*\*\*\*\*

```

.MACRO PRIT ARG,ENUM,ERRM,?A
TST ARG ;INTERRUPT OCCURED?
BNE A ;BRANCH IF YES
DEC R1 ;CHECKSUM = 7
SETPRI #PRI06 ;CHANGE PROCESSOR PRIORITY TO 6
TST ARG ;INTERRUPT OCCURED?
BNE A ;BRANCH IF YES
DEC R1 ;CHECKSUM = 6
SETPRI #PRI05 ;CHANGE PROCESSOR PRIORITY TO 5
TST ARG ;INTERRUPT OCCURED?
BNE A ;BRANCH IF YES
DEC R1 ;CHECKSUM = 5
SETPRI #PRI04 ;CHANGE PROCESSOR PRIORITY TO 4
TST ARG ;INTERRUPT OCCURED?
BNE A ;BRANCH IF YES
DEC R1 ;CHECKSUM = 4
SETPRI #PRI03 ;CHANGE PROCESSOR PRIORITY TO 3
TST ARG ;INTERRUPT OCCURED?
BNE A ;BRANCH IF YES
ERRSOFT ENUM,E402,ERRM ;NO INTERRUPT ERROR
A: NOP
.ENDM PRIT
.SBTTL GLOBAL SUBROUTINES SECTION

```

1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661

```

:++
: THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
: THAT ARE USED IN MORE THAN ONE TEST.
:++
:*****
: SUBROUTINE REGTST - GENERAL PURPOSE REGISTER TEST.
:*****
:
: FUNCTIONAL DESCRIPTION:
:
:     CHECKS THAT ALL READ/WRITE BITS OF THE SELECTED REGISTER CAN BE
:     SET, CLEARED, AND INDIVIDUALLY SET (SLIDING ONES PATTERN).
:
: INPUTS:
:
:     IF ENTERED AT LOCATION REGTST, THE LOCATIONS FOLLOWING THE
:     SUBROUTINE CALL MUST CONTAIN THE READ/WRITE BIT MASK, THE
:     ADDRESS OF THE REGISTER TO BE TESTED, AND THE FIRST ERROR NUMBER
:     TO BE USED (SEE CALLING SEQUENCE).
:
:     IF ENTERED AT LOCATION REGTS1, THE READ/WRITE BIT MASK, REGISTER
:     ADDRESS TO BE TESTED, AND THE FIRST ERROR NUMBER MUST BE LOADED
:     INTO LOCATIONS MASK, REGADD, AND ERNBR RESPECTIVELY. THIS
:     ALLOWS THE ARGUMENTS TO BE VARIED AT RUN TIME.
:
: IMPLICIT INPUTS:      NONE.
:
: OUTPUTS:              ERROR MESSAGES IF ERRORS OCCUR.
:
: IMPLICIT OUTPUTS:
:
:     IF ENTERED AT LOCATION REGTST,
:
:     MASK   - CONTAINS THE READ/WRITE BIT MASK
:     REGADD - CONTAINS THE ADDRESS OF THE REGISTER BEING TESTED
:
:     ALWAYS,
:
:     MASCOM - CONTAINS THE COMPLEMENT OF THE MASK
:     GOOD   - CONTAINS LAST EXPECTED DATA
:     BAD    - CONTAINS LAST ACTUAL DATA
:     ERNBR  - CONTAINS THE INPUT ERROR NUMBER + 2
:     ERRTP  - CONTAINS 3 (SOFT ERROR)
:     ERRBLK - CONTAINS ADDRESS OF REGERR (REGISTER ERROR MESSAGE)
:     ERRMSG - CONTAINS 3RD REGISTER ERROR MESSAGE
:
: SUBORDINATE ROUTINES USED:  DRS ERROR MACRO
:
: FUNCTIONAL SIDE EFFECTS:   NONE.
:
: CALLING SEQUENCE:

```

```

1662      :           EITHER FIXED PARAMETERS FOLLOW THE SUBROUTINE CALL :
1663      :
1664      :           EG.      CALL      REGTST
1665      :                   177          : BIT MASK OF R/W BITS
1666      :                   CSR          : REGISTER ADDRESS
1667      :                   200.        : FIRST ERROR NUMBER
1668      :
1669      :           OR PARAMETERS ARE SET DYNAMICALLY :
1670      :
1671      :           EG.      MOV      #177,MASK : BIT MASK OF R/W BITS
1672      :                   MOV      CSR,REGADD : REGISTER ADDRESS
1673      :                   MOV      #200.,ERRNBR : FIRST ERROR NUMBER
1674      :                   CALL     REGTS1
1675      :
1676      : *****
1677      :
1678 006262 REGTST::
1679 006262 017637 000000 002350 MOV @ (SP),MASK : GET R/W BIT MASK
1680 006270 062716 000002      ADD #2,(SP) : JUMP OVER ARGUMENT
1681 006274 017637 000000 002354 MOV @ (SP),REGADD : GET REGISTER ADDRESS
1682 006302 062716 000002      ADD #2,(SP) : JUMP OVER ARGUMENT
1683 006306 017637 000000 002506 MOV @ (SP),ERRNBR : GET FIRST ERROR NUMBER
1684 006314 062716 000002      ADD #2,(SP) : JUMP OVER ARGUMENT
1685 006320
1686 006320 013737 002350 002352 REGTS1::
1687 006326 005137 002352      MOV MASK,MASCOM : SET UP COMPLEMENT
1688 006332 012737 000003 002504 COM MASCOM : OF R/W BIT MASK
1689 006340 012737 006612 002512 MOV #3,ERRTYP : SET UP FOR SOFT ERROR
1690 006346 012737 006660 002510 MOV #REGERR,ERRBLK : SET UP ERROR MESSAGE ROUTINE
1691      : MOV #RERR1,ERRMSG : FIRST ERROR MESSAGE
1692      :
1693      : CHECK THAT ALL R/W BITS CAN BE SET
1694 006354 013737 002350 002500 MOV MASK,GOOD : SET UP EXPECTED DATA
1695 006362 BGNSEG
1696 006364 104404      BIS GOOD,@REGADD : SET ALL R/W BITS TRAP C$BSEG
1697 006372 017737 173756 002502 MOV @REGADD,BAD : READ THE RESULT
1698 006400 043737 002352 002502 BIC MASCOM,BAD : KEEP ONLY R/W BITS
1699 006406 023737 002502 002500 CMP BAD,GOOD : ALL R/W BITS SET?
1700 006414 001401      BEQ 1$ : IF YES, BRANCH
1701 006416 ERROR : ELSE REPORT ERROR TRAP C$ERROR
1702 006420 104460
1703 006420 104405      10000$: TRAP C$ESEG
1704      :
1705      : CHECK THAT ALL R/W BITS CAN BE CLEARED
1706 006422 005037 002500      CLR GOOD : SET UP EXPECTED DATA
1707 006426 005237 002506      INC ERRNBR : NEXT ERROR NUMBER
1708 006432 012737 006732 002510 MOV #RERR2,ERRMSG : NEXT ERROR MESSAGE
1709      :
1710 006440 BGNSEG
1711 006440 104404      TRAP C$BSEG
1712 006442 043777 002350 173704 BIC MASK,@REGADD : CLEAR ALL R/W BITS
1713 006450 017737 173700 002502 MOV @REGADD,BAD : READ THE RESULT
1714 006456 043737 002352 002502 BIC MASCOM,BAD : KEEP ONLY R/W BITS

```



GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 24-2  
GLOBAL SUBROUTINES SECTION

GL  
GL

```

1714 006464 023737 002502 002500      CMP      BAD,GOOD      ; ALL R/W BITS CLEAR?
1715 006472 001401                      BEQ      2$             ; IF YES, BRANCH
1716 006474 104460                      ERROR                      ; ELSE REPORT ERROR
                                           TRAP      C$ERROR
1717 006476 104405      2$:      ENDSEG
                                           10001$: TRAP      C$ESEG
006476
1718                                     ; CHECK THAT EACH R/W BIT CAN BE SET
1719                                     ;
1720                                     ;
1721 006500 005237 002506                      INC      ERRNBR          ; NEXT ERROR NUMBER
1722 006504 012737 007010 002510          MOV      #RERR3,ERRMSG  ; NEXT ERROR MESSAGE
1723 006512 012737 000001 002500          MOV      #1,GOOD        ; FIRST BIT TO TEST
1724 006520 033737 002500 002350      3$:      BIT      GOOD,MASK ; R/W BIT?
1725 006526 001004                      BNE      5$             ; IF YES, TEST IT
1726 006530 006337 002500      4$:      ASL      GOOD        ; ELSE FIND NEXT R/W BIT
1727 006534 103425                      BCS      7$             ; IF ALL DONE, RETURN
1728 006536 000770                      BR       3$             ; ELSE CHECK IF NEXT IS R/W
1729
1730                                     5$:      BGNSEG
                                           TRAP      C$BSEG
1731 006540 104404
1731 006542 042777 177767 173604          BIC      #177767,@REGADD ; CLEAR ALL BITS EXCEPT THE MUX BIT
1732 006550 053777 002500 173576          BIS      GOOD,@REGADD   ; SET THE BIT
1733 006556 017737 173572 002502          MOV      @REGADD,BAD    ; READ IT BACK
1734 006564 043737 002352 002502          BIC      MASCOM,BAD     ; KEEP ONLY R/W BITS
1735 006572 023737 002502 002500          CMP      BAD,GOOD      ; ALL OTHER BITS CLEAR?
1736 006600 001401                      BEQ      6$             ; IF YES, BRANCH
1737 006602 104460                      ERROR                      ; ELSE REPORT ERROR
                                           TRAP      C$ERROR
1738 006604 104405      6$:      ENDSEG
                                           10002$: TRAP      C$ESEG
006604
1739 006606 000750                      BR       4$             ; TEST NEXT BIT
1740
1741 006610 000207      7$:      RETURN
1742
1743
1744
1745 006612      BGNMSG  REGERR
1746 006612      PRINTB  #REGMSG,REGADD,CHAN,GOOD,BAD,MASK
                                           REGERR::
006612 013746 002350                      MOV      MASK,-(SP)
006616 013746 002502                      MOV      BAD,-(SP)
006622 013746 002500                      MOV      GOOD,-(SP)
006626 013746 002374                      MOV      CHAN,-(SP)
006632 013746 002354                      MOV      REGADD,-(SP)
006636 012746 007077                      MOV      #REGMSG,-(SP)
006642 012746 000006                      MOV      #6,-(SP)
006646 010600                      MOV      SP,R0
006650 104414                      TRAP      C$PNTB
006652 062706 000016                      ADD      #16,SP
1747 006656      ENDMSG
                                           L10011: TRAP      C$MSG
006656 104423
1748
1749      .NL IST BEX
1750

```

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 24-3  
GLOBAL SUBROUTINES SECTION

1751 006660 122 105 107 RERR1: .ASCIZ %REGISTER READ/WRITE BITS COULD NOT BE SET%  
1752 006732 122 105 107 RERR2: .ASCIZ %REGISTER READ/WRITE BITS COULD NOT BE CLEARED%  
1753 007010 122 105 107 RERR3: .ASCIZ %REGISTER READ/WRITE BITS COULD NOT BE INDIVIDUALLY SET%  
1754  
1755 007077 045 101 101 REGMSG: .ASCIZ .%AADD: %06%A,CHAN %01%A,GOOD %06%A, BAD %06%A, R/W BITS %06%N.  
1756  
1757 .LIST BEX  
1758 .EVEN

1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816

```

*****
SUBROUTINE MEMINI - MEMORY SIZING AND MEMORY MANAGEMENT INIT ROUTINE.
*****
**
FUNCTIONAL DESCRIPTION:
    DETERMINES THE SIZE OF CONTIGUOUS USABLE MEMORY AND OUTPUTS IT
    TO THE CONSOLE. IF MEMORY MANAGEMENT IS AVAILABLE, THE KERNEL
    PAGE ADDRESS AND DESCRIPTOR REGISTERS ARE INITIALISED.

INPUTS:
    NONE.

IMPLICIT INPUTS:
    NONE.

OUTPUTS:
    PHHSIZ - HIGH WORD OF MEMORY SIZE
    PHLOW  - LOW WORD OF MEMORY SIZE
    MMFLG  - FLAG SET TO 1 IF MEMORY MANAGEMENT AVAILABLE
    MM22   - FLAG SET TO 1 IF 22 BIT MEMORY MANAGEMENT AVAILABLE

    MEMORY SIZE IN K WORDS IS PRINTED

IMPLICIT OUTPUTS:
    IF MEMORY MANAGEMENT IS AVAILABLE, THE KERNEL PAGE DESCRIPTOR
    AND PAGE ADDRESS REGISTERS ARE SET AS FOLLOWS :-

    KPDR0 TO KPDR7 ARE SET TO 77406

    KPAR0 IS SET TO 0
    KPAR1 IS SET TO 200
    KPAR2 IS SET TO 400
    KPAR3 IS SET TO 600
    KPAR4 IS SET TO 1000
    KPAR6 IS SET TO 1400

    KPAR5 POINTS TO THE HIGHEST ADDRESSABLE MEMORY BANK
    KPAR7 POINTS TO THE I/O PAGE

    IF 22 BIT MAPPING IS AVAILABLE, IT IS ENABLED VIA SR3. MEMORY
    MANAGEMENT IS LEFT DISABLED VIA SRO.

SUBORDINATE ROUTINES USED:
    NXM    - NON EXISTANT MEMORY TRAP SERVICE ROUTINE
    VPCON  - VIRTUAL TO PHYSICAL ADDRESS CONVERSION ROUTINE

FUNCTIONAL SIDE EFFECTS:
    NONE

```



GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 25-1  
GLOBAL SUBROUTINES SECTION

```

1817 ; CALLING SEQUENCE:
1818 ;
1819 ;     CALL     MEMINI
1820 ;
1821 ;--
1822 ;
1823 ; KT11 STATUS REGISTER ADDRESSES
1824 ;
1825 SR0= 177572
1826 SR1= 177574
1827 SR2= 177576
1828 SR3= 172516
1829 ;
1830 ; KERNEL PAGE DESCRIPTOR REGISTERS
1831 ;
1832 KPDR0= 172300
1833 KPDR1= 172302
1834 KPDR2= 172304
1835 KPDR3= 172306
1836 KPDR4= 172310
1837 KPDR5= 172312
1838 KPDR6= 172314
1839 KPDR7= 172316
1840 ;
1841 ; KERNEL PAGE ADDRESS REGISTERS
1842 ;
1843 KPAR0= 172340
1844 KPAR1= 172342
1845 KPAR2= 172344
1846 KPAR3= 172346
1847 KPAR4= 172350
1848 KPAR5= 172352
1849 KPAR6= 172354
1850 KPAR7= 172356
1851 ;
1852 MEMINI::
1853     MOV     R0,-(SP) ; SAVE REGISTERS USED IN
1854     MOV     R1,-(SP) ; THIS ROUTINE
1855     MOV     R2,-(SP) ;
1856     SETVEC #4,#NXM,#340 ; SET UP NON-EXISTENT MEMORY TRAP VEC.
1857     MOV     #340,-(SP)
1858     MOV     #NXM,-(SP)
1859     MOV     #4,-(SP)
1860     MOV     #3,-(SP)
1861     TRAP   C$SVEC
1862     ADD    #10,SP
1863     CLR    NXMFLG ; CLEAR NXM FLAG
1864     CLR    PHLSIZ ; START WITH 1ST 2K BANK OF MEMORY
1865     CLR    PHSIZ ;
1866 ;
1867 ; SIZE MEMORY UP TO 32K
1868 ;
1869 ;10$: TST    @PHLSIZ ; CHECK THIS BANK EXISTS
1870 ;     TST    NXMFLG ; WAS THERE AN NXM TRAP?
1871 ;     BNE    140$ ; IF YES, PRINT THE MEMORY SIZE
1872 ;     ADD    #10000,PHLSIZ ; ELSE GET NEXT 2K BANK
1873 ;     CMP    PHLSIZ,#160000 ; HAVE WE REACHED THE I/O PAGE?

```

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 25-2  
GLOBAL SUBROUTINES SECTION

```

1868 007274 001364          BNE      10$          ; IF NOT, TRY NEXT 2K BANK
1869
1870          : 28K OR MORE - CHECK IF MEMORY MANAGEMENT UNIT PRESENT.
1871          :
1872 007276 005037 002332    CLR      MMFLG        ; ASSUME NO MEMORY MANAGEMENT
1873 007302 005737 177572    TST      SRO          ; ADDRESS MMU STATUS REGISTER 0
1874 007306 005737 002326    TST      NXMFLG       ; WAS THERE AN NXM TRAP?
1875 007312 001131          BNE      140$        ; IF YES, PRINT THE MEMORY SIZE
1876
1877          : MEMORY MANAGEMENT AVAILABLE - SET UP MMU REGISTERS
1878          :
1879 007314 012737 000001 002332  MOV      #1,MMFLG     ; FLAG MEMORY MANAGEMENT AVAILABLE
1880 007322 012700 172340    MOV      #KPAR0,R0   ; LOAD FIRST KPAR ADDRESS
1881 007326 005001          CLR      R1          ; SET UP CONTENTS OF FIRST KPAR
1882 007330 012702 000006    MOV      #6,R2       ; SET UP FIRST 6 KPAR'S
1883 007334 010120 50$:    MOV      R1,(R0)+    ; LOAD KPAR VALUE
1884 007336 062701 000200    ADD      #200,R1     ; NEXT KPAR VALUE
1885 007342 005302          DEC      R2          ; ALL KPAR'S LOADED?
1886 007344 001373          BNE      50$        ; IF NOT, LOAD NEXT
1887 007346 012737 177600 172356  MOV      #177600,KPAR7 ; ELSE LOAD KPAR7 WITH I/O PAGE ADDRESS
1888
1889 007354 012700 172300    MOV      #KPDRO,R0   ; LOAD FIRST PDR ADDRESS
1890 007360 012701 000010    MOV      #10,R1      ; SET UP 8 PDR'S
1891 007364 012720 077406 60$:    MOV      #77406,(R0)+ ; LOAD ALL PDR'S WITH 77406
1892 007370 005301          DEC      R1          ; ALL LOADED?
1893 007372 001374          BNE      60$        ; IF NOT, LOAD NEXT
1894
1895          : USE THE MEMORY MANAGEMENT UNIT TO SIZE THE MEMORY UP TO 128K
1896          :
1897 007374 005037 172352    CLR      KPAR5       ; POINT KPAR5 TO FIRST PAGE
1898 007400 012737 000001 177572  MOV      #1,SRO      ; ENABLE MEMORY MANAGEMENT
1899 007406 005737 120000 70$:    TST      120000     ; ADDRESS PAGE POINTED TO BY KPAR5
1900 007412 005737 002326    TST      NXMFLG       ; WAS THERE AN NXM TRAP?
1901 007416 001053          BNE      130$        ; IF YES, PRINT THE MEMORY SIZE
1902 007420 062737 000200 172352  ADD      #200,KPAR5   ; ELSE POINT KPAR5 TO NEXT PAGE
1903 007426 023727 172352 007600  CMP      KPAR5,#7600 ; ARE WE IN THE I/O PAGE?
1904 007434 001364          BNE      70$        ; IF NOT, TEST NEXT PAGE
1905
1906          : 128 K OR MORE - CHECK IF 22 BIT MMU PRESENT
1907          :
1908 007436 005037 002330 90$:    CLR      MM22        ; ASSUME NO 22 BIT MAPPING
1909 007442 005000          CLR      R0          ; COMPARE DATA AT ADDRESS 0
1910 007444 012737 010000 172352  MOV      #10000,KPAR5 ; WITH DATA AT 128 K
1911 007452 012701 120000    MOV      #120000,R1  ; USING KPAR5
1912 007456 012702 000010    MOV      #10,R2      ; WILL CHECK 8 WORDS
1913 007462 012737 000020 172516  MOV      #20,SR3     ; ENABLE 22 BIT MAPPING
1914
1915 007470 022021 100$:    CMP      (R0)+,(R1)+ ; DATA IDENTICAL?
1916 007472 001006          BNE      110$        ; IF NOT, WE HAVE 22 BIT MMU
1917 007474 005302          DEC      R2          ; ELSE CHECKED 8 WORDS?
1918 007476 001374          BNE      100$       ; IF NOT, CHECK NEXT
1919 007500 012737 007600 172352  MOV      #7600,KPAR5 ; ELSE CAN ONLY USE 124 K
1920 007506 000417          BR       130$       ; PRINT OUT MEMORY SIZE
1921
1922          : 22 BIT MEMORY MANAGEMENT AVAILABLE - USE TO SIZE MEMORY UP TO 4 MBYTES
1923          :
1924 007510 012737 000001 002330 110$:  MOV      #1,MM22     ; FLAG 22 BIT MMU AVAILABLE

```

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 25-3  
GLOBAL SUBROUTINES SECTION

```

1925 007516 005737 120000      120$: TST      120000      : ADDRESS PAGE POINTED TO BY KPAR5
1926 007522 005737 002326      TST      NXMFLG      : WAS THERE AN NXM TRAP?
1927 007526 001007              BNE      130$        : IF YES, SAVE THE MEMORY SIZE
1928 007530 062737 000200 172352  ADD      #200,KPAR5  : ELSE POINT TO NEXT PAGE
1929 007536 023727 172352 177600  CMP      KPAR5,#177600 : REACHED THE I/O PAGE?
1930 007544 001364              BNE      120$        : IF NOT, TEST THE NEXT PAGE
1931
1932      : CONVERT VIRTUAL SIZE TO PHYSICAL SIZE
1933
1934 007546 005037 177572      130$: CLR      SRO        : DISABLE MEMORY MANAGEMENT
1935 007552 005037 002340      CLR      VIADD       : CONVERT KPAR5 TO PHYSICAL
1936 007556 004737 010024      JSR      PC,VPCON    : MEMORY SIZE
1937 007562 013737 002334 002342  MOV      PHHIGH,PHHSIZ : SAVE MEMORY SIZE
1938 007570 013737 002336 002344  MOV      PHLOW,PHLSIZ  :
1939
1940      : PRINT OUT MEMORY SIZE AND RETURN FROM THE SUBROUTINE
1941
1942 007576 013701 002342      140$: MOV      PHHSIZ,R1    : GET SIZE HIGH WORD
1943 007602 013700 002344      MOV      PHLSIZ,R0    : AND LOW WORD
1944      .REPT      5      : SHIFT HIGH AND LOW
1945      ASL      RO        : WORDS TO GET THE
1946      ROL      R1        : NUMBER OF K WORDS
1947      .ENDR
1948 007632              PRINT# #MSIZE,R1    : PRINT THE SIZE
1949      MOV      R1,-(SP)
1950      MOV      #MSIZE,-(SP)
1951      MOV      #2,-(SP)
1952      MOV      SP,RO
1953      TRAP    C$PNTF
1954      ADD      #6,SP
1955
1956      CLRVEC #4      : RESET THE NXM VECTOR
1957
1958      MOV      (SP)+,R2  : RESTORE THE REGISTERS
1959      MOV      (SP)+,R1  : USED BY THE ROUTINE
1960      MOV      (SP)+,R0
1961      RETURN
1962
1963      .NLIST BEX
1964      .ASCIZ /%MEMORY SIZE = %D4%A K%/
1965      .LIST BEX
1966      .EVEN

```



1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017

```

*****
: SUBROUTINE PVCON - PHYSICAL TO VIRTUAL ADDRESS CONVERSION ROUTINE
*****
**
: FUNCTIONAL DESCRIPTION:
:
: CONVERTS A PHYSICAL ADDRESS OF UP TO 22 BITS INTO A VIRTUAL
: ADDRESS USING KPAR 5.
:
: INPUTS:
:
: PHHIGH - HIGH WORD OF PHYSICAL ADDRESS
: PHLOW - LOW WORD OF PHYSICAL ADDRESS
:
: IMPLICIT INPUTS:
:
: NONE.
:
: OUTPUTS:
:
: VIADD - VIRTUAL ADDRESS USING KPAR5
: KPAR5 - POINTS TO PHYSICAL PAGE
:
: IMPLICIT OUTPUTS:
:
: NONE.
:
: SUBORDINATE ROUTINES USED:
:
: NONE.
:
: FUNCTIONAL SIDE EFFECTS:
:
: NONE.
:
: CALLING SEQUENCE:
:
: CALL PVCON
:
:--
PVCON::
: MOV R0,-(SP) ; SAVE REGISTERS USED IN
: MOV R1,-(SP) ; THIS ROUTINE
:
: PUT ADDRESS BITS 0 TO 6 INTO THE VIRTUAL ADDRESS
:
: MOV PHLOW,VIADD ; LOAD LOWEST 15 BITS OF PHYSICAL ADDR.
: BIC #177700,VIADD ; CLEAR PAGE AND BLOCK INFORMATION
: BIS #120000,VIADD ; SET THE PAGE REGISTER TO KPAR5
:
: PUT ADDRESS BITS 6 TO 21 INTO KPAR5
:
: MOV PHLOW,R0 ; INITIALISE OUR SHIFT REGISTER
: MOV PHHIGH,R1 ; HIGH WORD WILL BE SHIFTED IN
: .REPT 6
: ASR R1 ; SHIFT HIGH WORD

```

```

007724
007724 010046
007726 010146
007730 013737 002336 002340
007736 042737 177700 002340
007744 052737 120000 002340
007752 013700 002336
007756 013701 002334
000006

```

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 26-1  
GLOBAL SUBROUTINES SECTION

2018			ROR	RO	: INTO LOW WORD
2019			. ENDR		
2020	010012	010037	MOV	RO, KPAR5	: SAVE AS KPAR5
2021	010016	012601	MOV	(SP)+, R1	: RESTORE THE REGISTERS
2022	010020	012600	MOV	(SP)+, RO	: USED BY THIS ROUTINE
2023	010022	000207	RETURN		:

2025  
 2026  
 2027  
 2028  
 2029  
 2030  
 2031  
 2032  
 2033  
 2034  
 2035  
 2036  
 2037  
 2038  
 2039  
 2040  
 2041  
 2042  
 2043  
 2044  
 2045  
 2046  
 2047  
 2048  
 2049  
 2050  
 2051  
 2052  
 2053  
 2054  
 2055  
 2056  
 2057  
 2058  
 2059  
 2060  
 2061  
 2062  
 2063  
 2064  
 2065  
 2066  
 2067  
 2068  
 2069  
 2070  
 2071  
 2072  
 2073  
 2074  
 2075  
 2076  
 2077  
 2078  
 2079  
 2080  
 2081

```

.....
: SUBROUTINE VPCON - VIRTUAL TO PHYSICAL ADDRESS CONVERSION ROUTINE
.....
**
: FUNCTIONAL DESCRIPTION:
:
:     CONVERTS A VIRTUAL ADDRESS TOGETHER WITH KPAR 5 INTO A PHYSICAL
:     ADDRESS OF UP TO 22 BITS.
:
: INPUTS:
:
:     VIADD - VIRTUAL ADDRESS (BITS 13 TO 15 ARE IGNORED)
:     KPAR5 - PAGE ADDRESS REGISTER 5
:
: IMPLICIT INPUTS:
:
:     NONE.
:
: OUTPUTS:
:
:     PHHIGH - PHYSICAL ADDRESS HIGH WORD
:     PHLOW  - PHYSICAL ADDRESS LOW WORD
:
: IMPLICIT OUTPUTS:
:
:     NONE.
:
: SUBORDINATE ROUTINES USED:
:
:     NONE.
:
: FUNCTIONAL SIDE EFFECTS:
:
:     NONE.
:
: CALLING SEQUENCE:
:
:     CALL VPCON
:
:--
    
```

```

2067 010024
2068 010024 010046
2069 010026 013737 002340 002336
2070 010034 042737 160000 002336
2071 010042 013700 172352
2072 010046 005037 002334
2073      000006
2074
2075
2076
2077 010116 060037 002336
2078 010122 005537 002334
2079 010126 012600
2080 010130 000207
2081
    
```

```

VPCON::
MOV     R0, -(SP)           ; SAVE R0
MOV     VIADD, PHLOW       ; SET UP LOW WORD
BIC     #160000, PHLOW     ; DISCARD PAGE INFORMATION
MOV     KPAR5, R0          ; PAGE WILL GO IN HIGH WORD
CLR     PHHIGH             ; INITIALISE HIGH WORD
.REPT   6
ASL     R0                  ; SHIFT PAGE NUMBER
ROL     PHHIGH             ; INTO HIGH WORD
.ENDR
ADD     R0, PHLOW          ; ADD BLOCK NO. TO LOW WORD
ADC     PHHIGH             ; CARRY ACROSS TO HIGH WORD
MOV     (SP)+, R0          ; RESTORE R0
RETURN
    
```



2083  
2084  
2085  
2086  
2087  
2088  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115

010132  
010132  
010132 012737 000001 002326  
010140  
010140  
010140 000002

```

:*****:
: INTERRUPT SERVICE ROUTINES :
:*****:
.SBTTL GLOBAL INTERRUPT HANDLING ROUTINES

:++
:THE INTERRUPT HANDLING SECTION CONTAINS CODING REQUIRED TO USE
:THE 'SETVEC' MACRO. NOTE EVERY INTERRUPT ROUTINE SHOULD SAVE
:AND RESTOR RO.
:---

:*****:
:      NXM - INTERRUPT SERVICE ROUTINE
:*****:
:FUNCTION:      THIS ROUTINE IS ASSIGNED TO VECTOR 4
:                WHEN ADDRESSING THE IEX FOR THE FIRST TIME.
:                IF THIS INTERRUPT IS GENERATED THE IEX
:                IS INCORRECTLY ADDRESSED.
:                THIS ROUTINE IS ALSO USED IN MEMORY MANAGEMENT
:                SUBROUTINE

:ENTRY CONDITON:

:EXIT CONDITON: WHEN THIS INTERRUPT IS SERVICED THE
:                NXMFLG IS SET.

:USED IN TESTS: AUTO DROP,23,24,
:*****:

      BGNSRV  NXM
      MOV     #1,NXMFLG      ;SET FLAG IF MEMORY IS NON-EXISTENT.
      ENDSRV

      L10012:
      RTI

```

2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130

```
*****  
:      INTSC1 - INTERRUPT SERVICE ROUTINE  
:FUNTION:      GENERAL PURPOSE INTERRUPT ROUTINE FOR CHANNEL 1.  
:              SET A FLAG WHEN INTERRUPT WAS GENERATED  
:ENTRY CONDION:  
:EXIT CONDION: INTFC1 = FLAG, SET WHEN THIS INTERRUPT IS SERVID  
:CALLED BY TEST:5,23,24,25,26  
:*****
```

2131 010142  
010142  
2132 010142 012737 000001 002376  
2133 010150  
010150  
010150 000002

```
BGNSRV INTSC1  
MOV #1,INTFC1 ;SET INTERRUPT FLAG INTSC1::  
ENDSRV  
L10013:  
RTI
```

2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147

```
*****  
:      INTSC2 - INTERRUPT SERVICE ROUTINE  
:FUNTION:      GENERAL PURPOSE INTERRUPT ROUTINE FOR CHANNEL 2.  
:              SET A FLAG WHEN INTERRUPT IN CHANNEL 2 WAS  
:              GENERATED  
:ENTRY CONDION:  
:EXIT CONDION: INTFC2 = FLAG,SET WHEN THIS INTERRUPT IS SERVID  
:USED IN TEST:5,23,24,25,26  
:*****
```

2148 010152  
010152  
2149 010152 012737 000001 002400  
2150 010160  
010160  
010160 000002

```
BGNSRV INTSC2  
MOV #1,INTFC2 ;SET INTERRUPT FLAG INTSC2::  
ENDSRV  
L10014:  
RTI
```





```

2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184 010220
      010220 104404
2185 010222
      010222 104433
2186 010224 112777 000000 172036
2187 010232 012777 000010 172040
2188 010240 112777 000000 172022
2189 010246 112777 000000 172000
2190 010254 005077 172020
2191 010260 052777 000002 172012
2192 010266 112777 000217 171774
2193 010274 012701 000100
2194 010300 005301
2195 010302 001376
2196 010304 112777 000017 171756
2197 010312 112777 000040 171756
2198 010320 004737 011060
2199 010324 112777 000077 171744
2200 010332 004737 011060
2201 010336 112777 000200 171724
2202 010344 112777 000000 171716
2203 010352 112777 000002 171674
2204 010360 012777 000010 171712
2205 010366 052777 000002 171704
2206 010374 112777 000217 171666
2207 010402 012701 000100
2208 010406 005301
2209 010410 001376
2210 010412 112777 000017 171650
2211 010420 112777 000042 171650
2212 010426 004737 011060
2213 010432 112777 000077 171636
2214 010440 004737 011060
2215 010444 112777 000200 171616
2216 010452 112777 000000 171610
2217 010460 112777 000000 171572
2218 010466 112777 000000 171566
2219 010474 005077 171600
2220 010500 112777 000000 171552
2221 010506 112777 000000 171546
2222 010514 112777 000200 171546
2223 010522 112777 000000 171540
2224 010530
      010530
      010530 104405
2225 010532 000207

```

```

*****
SUBROUTINE CULPA
=====
FUNCTION: THIS SUBROUTINE CLEARS THE ULPA BIT IN ISR REGISTER
          AND THE MASK BITS IN IIR REGISTER FOR BOTH CHANNELS .
CALLING FORMAT: JSR PC,CULPA
CALLED BY TEST: ALL TESTS EXCEPT TEST 1,2,3,14,23,24,25,26,27
*****
CULPA:: BGNSEG
          TRAP CSBSEG
          BRESET :RESET HARDWARE TRAP C$RESET
          MOVB #0,@ICRHX :LOAD NOT SWRST INTO ACR 1
          MOV #10,@CSRX :SELECT CHANNEL 2
          MOVB #0,@ICRHX :LOAD NOT SWRST INTO ACR 2
          MOVB #0,@IIRHX :LOAD EVEN DPA INTO ADR 2
          CLR @CSRX :SELECT CHANNEL 1
          BIS #2,@CSRX :SELECT CHA. 1 AS SYSTEM CONTROLLER
          MOVB #217,@ICRHX :LOAD SIC INTO ACR1
          MOV #100,R1 :WAIT 100 US
2$: DEC R1 :...
   BNE 2$ :...
   MOVB #17,@ICRHX :LOAD NOT SIC INTO ACR1
   MOVB #40,@IDRHX :LOAD EVEN MLA 2 INTO DOR 1
   JSR PC,LOOP :WAIT A LITTLE
   MOVB #77,@IDRHX :LOAD UNL INTO DOR 1
   JSR PC,LOOP :WAIT A LITTLE
   MOVB #200,@ICRHX :SET SYSTEM CONTROLLER 1 IN IDLE STATE
   MOVB #0,@ICRHX :...
   MOVB #2,@IIRHX :LOAD EVEN DPA INTO ADR 1
   MOV #10,@CSRX :SELECT CHANNEL 2
   BIS #2,@CSRX :SELECT CHANNEL 2 AS SYS. CONTROLLER
   MOVB #217,@ICRHX :LOAD SIC INTO ACR2
   MOV #100,R1 :WAIT 100 US
3$: DEC R1 :...
   BNE 3$ :...
   MOVB #17,@ICRHX :LOAD NOT SIC INTO ACR2
   MOVB #42,@IDRHX :LOAD EVEN MLA INTO DOR 2
   JSR PC,LOOP :WAIT A LITTLE
   MOVB #77,@IDRHX :LOAD UNL INTO DOR 2
   JSR PC,LOOP :WAIT A LITTLE
   MOVB #200,@ICRHX :SET SYSTEM CONTROLLER 2 IN IDLE STATE
   MOVB #0,@ICRHX :...
   MOVB #0,@ISRLX :CLEAR LOW BYTE OF ISR2
   MOVB #0,@ISRHX :CLEAR HIGH BYTE OF ISR2
   CLR @CSRX :SELECT CHANNEL 1
   MOVB #0,@ISRLX :CLEAR LOW BYTE OF ISR1
   MOVB #0,@ISRHX :CLEAR HIGH BYTE OF ISR1
   MOVB #200,@ICRHX :SET SYSTEM CONTROLLER 1 IN IDEL STATE
   MOVB #0,@ICRHX :...
          ENDSEG
          10000$: TRAP C$ESEG
          RETURN

```

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 32-1  
GLOBAL INTERRUPT HANDLING ROUTINES

2226

HA  
ADI

```

2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241 010534
      010534 104404
2242 010536
      010536 104433
2243 010540 113777 002312 171506
2244 010546 012777 000010 171524
2245 010554 113777 002314 171472
2246 010562 005077 171512
2247 010566 112777 000000 171474
2248 010574 112777 000000 171456
2249 010602 112777 000000 171452
2250 010610 112777 000010 171462
2251 010616 112777 000000 171444
2252 010624 112777 000000 171426
2253 010632 112777 000000 171422
2254 010640 005077 171434
2255 010644 012737 000001 002374
2256 010652 052777 000002 171420
2257 010660 112777 000217 171402
2258 010666 012701 000100
2259 010672 005301
2260 010674 001376
2261 010676 112777 000017 171364
2262 010704
      010704
      010704 104405
2263 010706 000207

*****
SUBROUTINE BGIN1
=====
:FUNCTION: THIS SUBROUTINE LOADS THE DPA ADDRESS AND THE
           SOFTWARE RESET INTO BOTH CHANNELS,
           IT ALSO SELECTS CHANNEL 1 AS SYSTEM CONTROLLER
           AND CLEARS MASK0+MASK1 REGISTER .
:CALLING FORMAT: JSR PC,BGIN1
:CALLED BY TEST: 5,7,8,11,12,13,14,15,16,17,19,21,22,23,26
*****
BGIN1:: BGNSEG
           TRAP C$BSEG
BRESET
           TRAP C$RESET
MOV B DPA1,@IIRHX ;LOAD DEVICE PRIM. ADDR.1
MOV #10,@CSRX ;SELECT CHANNEL 2
MOV B DPA2,@IIRHX ;LOAD DEVICE PRIM. ADDR. 2
CLR @CSRX ;SELECT CHANNEL 1
MOV B #0,@ICRHX ;LOAD NOT SWRST IN ACR 1
MOV B #0,@ISRLX ;CLEAR LOW BYTE OF ISR1
MOV B #0,@ISRHX ;CLEAR HIGH BYTE OF ISR1
MOV B #10,@CSRX ;SELECT CHANNEL 2
MOV B #0,@ICRHX ;LOAD NOT SWRST IN ACR 2
MOV B #0,@ISRLX ;CLEAR LOW BYTE OF ISR2
MOV B #0,@ISRHX ;CLEAR HIGH BYTE OF ISR2
CLR @CSRX ;SELECT CHANNEL 1
MOV #1,CHAN ;LOAD CHANNEL NUMBER
BIS #2,@CSRX ;SELECT CHANNEL 1 AS SYSTEM CONTROLLER
MOV B #217,@ICRHX ;LOAD SIC IN ACR 1
MOV #100,R1 ;WAIT 100 US
1$: DEC R1 ;...
BNE 1$ ;...
MOV B #17,@ICRHX ;LOAD NOT SIC IN ACR1
ENDSEG
           10001$: TRAP C$ESEG
RETURN

```



2265  
2266  
2267  
2268  
2269  
2270  
2271  
2272  
2273  
2274  
2275  
2276  
2277  
2278 010710  
010710 104404  
2279 010712  
010712 104433  
2280 010714 113777 002312 171332  
2281 010722 012777 000010 171350  
2282 010730 113777 002314 171316  
2283 010736 005077 171336  
2284 010742 112777 000000 171320  
2285 010750 112777 000000 171302  
2286 010756 112777 000000 171276  
2287 010764 012777 000010 171306  
2288 010772 012737 000002 002374  
2289 011000 112777 000000 171262  
2290 011006 112777 000000 171244  
2291 011014 112777 000000 171240  
2292 011022 052777 000002 171250  
2293 011030 112777 000217 171232  
2294 011036 012701 000100  
2295 011042 005301  
2296 011044 001376  
2297 011046 112777 000017 171214  
2298 011054  
011054  
011054 104405  
2299 011056 000207

```

*****
SUBROUTINE BGIN2
=====
FUNCTION: THIS SUBROUTINE LOADS THE DPA ADDRESSES AND THE
          SOFTWARE RESET INTO BOTH CHANNELS
          IT ALSO SELECTS CHANNEL 2 AS SYSTEM CONTROLLER
          AND CLEARS MASK0+MASK1 REGISTER.

CALLING FORMAT: JSR PC,BGIN2

CALLED BY TEST: 5,6,8,9,10,13,14,15,16,17,18,20,22,24,25
*****
BGIN2:: BGNSEG
          BRESET ;RESET HARDWARE TRAP CSBSEG
          MOV B #10,@CSRX ;LOAD DEVICE PRIM. ADDR.1 TRAP CSRESET
          MOV B #10,@CSRX ;SELECT CHANNEL 2
          MOV B #10,@CSRX ;LOAD DEVICE PRIM. ADDR. 2
          CLR @CSRX ;SELECT CHANNEL 1
          MOV B #0,@ICRHX ;LOAD NOT SWRST IN ACR 1
          MOV B #0,@ISRLX ;CLEAR LOW BYTE OF ISR1
          MOV B #0,@ISRHX ;CLEAR HIGH BYTE OF ISR1
          MOV B #10,@CSRX ;SELECT CHANNEL 2
          MOV B #2,CHAN ;LOAD CHANNEL FLAG
          MOV B #0,@ICRHX ;LOAD NOT SWRST IN ACR 2
          MOV B #0,@ISRLX ;CLEAR LOW BYTE OF ISR2
          MOV B #0,@ISRHX ;CLEAR HIGH BYTE OF ISR2
          BIS #2,@CSRX ;SELECT CHANNEL 2 AS SYSTEM CONTROLLER
          MOV B #217,@ICRHX ;LOAD SIC IN ACR 2
          MOV #100,R1 ;WAIT 100 US
          DEC R1
          BNE 1$
          MOV B #17,@ICRHX ;LOAD NOT SIC IN ACR2
          ENDSEG
          10002$: TRAP CSESEG
          RETURN

```

GLOBAL AREAS MACRO M1113 06-SEP-82 16:46 PAGE 35  
GLOBAL INTERRUPT HANDLING ROUTINES

2301  
 2302  
 2303  
 2304  
 2305  
 2306  
 2307  
 2308  
 2309  
 2310  
 2311  
 2312 011060 012701 000001  
 2313 011064 005301  
 2314 011066 001376  
 2315 011070 000207  
 2316  
 2317  
 2318  
 2319  
 2320  
 2321  
 2322  
 2323  
 2324  
 2325  
 2326  
 2327  
 2328  
 2329 011072 012701 000100  
 2330 011076 005301  
 2331 011100 001376  
 2332 011102 000207  
 2333  
 2334 011104  
 2335

```

.....
SUBROUTINE LOOP
=====
:FUNCTION : ROUTINE FOR WAIT (AT LEAST) 1 US.
:CALLING FORMAT: JSR PC,LOOP
:CALLED BY TEST: ALL TESTS EXCEPT 1,2,3,4,5
.....
LOOP: MOV #1,R1 ;LOAD LOOP COUNTER
1$: DEC R1 ;DECREMENT LOOP COUNTER
BNE 1$ ;LOOP UNTIL DONE
RETURN

```

```

.....
SUBROUTINE WAIT
=====
:FUNCTION: ROUTINE FOR WAIT (AT LEAST) 100 US.
:CALLING FORMAT: JSR PC,WAIT
:CALLED BY TEST: 4,6,8,13,17
.....
WAIT:: MOV #100,R1
1$: DEC R1 ;DECREMENT COUNTER
BNE 1$ ;WAIT UNTIL DONE
RETURN
ENDMOD

```

MISCELLANEOUS SECTIONS MACRO M1113 06-SEP-82 16:46 PAGE 36  
 GLOBAL INTERRUPT HANDLING ROUTINES

```

2348          .TITLE MISCELLANEOUS SECTIONS
2349          .SBTTL REPORT CODING SECTION
2377
2378 011104          BGNMOD
2379
2380          :++
2381          : THE REPORT CODING SECTION CONTAINS THE
2382          : 'PRINTS' CALLS THAT GENERATE STATISTICAL REPORTS.
2383          :--
2384
2385 011104          BGNRPT
2386
2387
2388
2389
2398
2399 011104          EXIT    RPT
2399 011104 000167
2399 011106 000000
2400
2412
2413          .EVEN
2414
2415 011110          ENDRPT
2415 011110
2415 011110 104425
    
```

LSRPT::

.WORD JSJMP  
.WORD L10016-2-

L10016: TRAP CSRPT



MISCELLANEOUS SECTIONS MACRO M1113 06-SEP-82 16:46 PAGE 37  
PROTECTION TABLE

2417  
2418  
2419  
2420  
2421  
2422  
2423  
2424 011112  
011112  
2425  
2426 011112 000000  
2427 011114 177777  
2428 011116 177777  
2429  
2430 011120  
2431

.SBTTL PROTECTION TABLE

::++  
: THIS TABLE IS USED BY THE RUNTIME SERVICES  
: TO PROTECT THE LOAD MEDIA.  
:--

BGNPROT

L\$PROT::

.WORD 0  
.WORD -1  
.WORD -1

:CHECK CSR ADDRESS  
:DON'T CHECK MASSBUS UNIT NUMBER  
:DON'T CHECK DRIVE NUMBER

ENDPROT

MISCELLANEOUS SECTIONS MACRO M1113 06-SEP-82 16:46 PAGE 38  
INITIALIZE SECTION

```

2446          .SBTTL  INITIALIZE SECTION
2447
2448          : **
2449          : THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
2450          : AT THE BEGINNING OF EACH PASS.
2451          : --
2452
2453 011120      BGNINIT
2454 011120
2478
2479 011120      RFLAGS  RO          ;GET THE OPERATOR FLAGS
2480 011120 104421          ;
2481 011122 032700 001000      BIT    #1000,RO          ;IS PNT FLAG SET?
2482 011126 001404          BEQ    2$          ;BRANCH IF NO
2483 011130 012737 000001 002324  MOV    #1,PNTF          ;SET FLAG FOR TEST HEADER PRINTOUT
2484 011136 000402          BR     3$          ;
2485 011140 005037 002324 2$:  CLR    PNTF          ;CLEAR FLAG FOR DISABLE TEST HEADER PRINTOUT
2486 011144          3$:  READEF #EF.START          ;IS THIS JUST STARTED ?
2487 011144 012700 000040          MOV    #EF.START,RO
2488 011150 104447          TRAP   C$RFLA
2489 011152          BCOMPLETE STARST          ;IF YES - BRANCH.
2490 011152 103424          BCS    STARST
2491 011154          READEF #EF.RESTART          ;IS THIS A RESTART ?
2492 011154 012700 000037          MOV    #EF.RESTART,RO
2493 011160 104447          TRAP   C$REFG
2494 011162          BCOMPLETE NEWST          ;IF YES - BRANCH
2495 011162 103425          BCS    NEWST
2496 011164          READEF #EF.NEW          ;IS THIS A NEW PASS ?
2497 011164 012700 000035          MOV    #EF.NEW,RO
2498 011170 104447          TRAP   C$REFG
2499 011172          BCOMPLETE NEWST          ;IF YES - FIRST UNIT AGAIN
2500 011172 103421          BCS    NEWST
2501 011174          READEF #EF.CONTINUE          ;IS THIS A CONTINUE ?
2502 011174 012700 000036          MOV    #EF.CONTINUE,RO
2503 011200 104447          TRAP   C$REFG
2504 011202          BCOMPLETE EXINI          ;IF YES - DON'T INITIALIZE
2505 011202 103406          BCS    EXINI
2506 011204          READEF #EF.PWR          ;IS THIS A POWER FAIL
2507 011204 012700 000034          MOV    #EF.PWR,RO
2508 011210 104447          TRAP   C$REFG
2509 011212          BNCOMPLETE GETPRM          ;IF NOT-MUST BE A NEW UNIT
2510 011212 103014          BCC    GETPRM
2511 011214 004737 007176          JSR    PC,MEMINI          ;IF YES-MUST BE A POWER FAIL,RELOAD MMU
2512 011220          EXINI:  EXIT    INIT          ;
2513 011220 104432          TRAP   C$EXIT
2514 011222 001004          .WORD  L10020-.
2515 011224          STARST: SETPRI #PRI07          ;SET DIAGNOSTIC TO PRIORITY 7
2516 011224 012700 000340          MOV    #PRI07,RO
2517 011230 104441          TRAP   C$SPRI
2518 011232 004737 007176          JSR    PC,MEMINI          ;INITIATE MEMORY MANAGEMENT
2519 011236 012737 177777 002372  NEWST:  MOV    #-1,LOGDEV          ;INITIALIZE LOGICAL UNIT NUMBER.
2520 011244 005237 002372 002372  GETPRM:  INC    LOGDEV          ;NEXT LOGICAL UNIT TO BE TESTED ?
2521 011250 023737 002372 002012  CMP    LOGDEV,L$UNIT          ;IS THE MAXIMUM UNIT # EXCEEDED ?
2522 011256 002367          BGE    NEWST          ;IF YES - A NEW START
2523          :          SETVEC #14,#34716,#340          ;***JUST FOR DEBUG PROGRAM;:::::::::::
2524          :          NOP          ;SPACE FOR DEBUG PROGRAM

```

MISCELLANEOUS SECTIONS MACRO M1113 06-SEP-82 16:46 PAGE 38-1  
INITIALIZE SECTION

```

2505 011262 000240      NOP      :...
2506 011264 000240      NOP      :...
2507 011266 000240      NOP      :...
2508 011270 000240      NOP      :...
2509 011272 000240      NOP      :...
2510 011274 000240      GPHARD  LOGDEV,R1  ;GET THE P-TABLE POINTER INTO R1
      011274 013700 002372      MOV      LOGDEV,R0
      011300 104442      TRAP    C$GPHRD
      011302 010001      MOV      R0,R1
2511 011304 000240      BNCOMPLETE GETPRM ;IF NOT AVAILABLE ,GET THE NEXT ONE
      011304 103357      BCC     GETPRM
2512 011306 011100      MOV      (R1),R0 ;SAVE THE ADDRESS
2513 011310 032700 000007      BIT      #7,R0 ;DOES THIS DEVICE ADDRESS END IN NON-ZERO?
2514 011314 001414      BEQ     10$ ;IF NOT - OK (76XXX0)
2515 011316 042711 000007      BIC     #7,(R1) ;MAKE IT 76XXX0
2516 011322 000240      PRINTB #FINIT1,(R1),R0 ;INFORM THE USER
      011322 010046      MOV      R0,-(SP)
      011324 011146      MOV      (R1),-(SP)
      011326 012746 012046      MOV      #FINIT1,-(SP)
      011332 012746 000003      MOV      #3,-(SP)
      011336 010600      MOV      SP,R0
      011340 104414      TRAP    C$PNTB
      011342 062706      ADD     #10,SP
2517 011346 011137 002256      10$: MOV      (R1),ISRX ;LOAD ADDRESS 0
2518 011352 013737 002256 002260      MOV      ISRX,ISRLX ;LOAD LOW BYTE LABLE OF ADDRSS 0
2519 011360 011137 002262      MOV      (R1),ISRHX ;LOAD HIGH BYTE, ADDRESS 1
2520 011364 062737 000001 002262      ADD     #1,ISRHX ;...
2521 011372 011137 002250      MOV      (R1),IIRX ;LOAD ADDRESS 2
2522 011376 062737 000002 002250      ADD     #2,IIRX ;...
2523 011404 013737 002250 002252      MOV      IIRX,IIRLX ;LOAD LOW BYTE LABLE OF ADDRESS 2
2524 011412 011137 002254      MOV      (R1),IIRHX ;LOAD HIGH BYTE, ADDRESS 3
2525 011416 062737 000003 002254      ADD     #3,IIRHX ;...
2526 011424 011137 002264      MOV      (R1),ICRX ;LOAD ADDRESS 4
2527 011430 062737 000004 002264      ADD     #4,ICRX ;...
2528 011436 013737 002264 002266      MOV      ICRX,ICRLX ;LOAD LOW BYTE LABLE OF ADDRESS 4
2529 011444 011137 002270      MOV      (R1),ICRHX ;LOAD HIGH BYTE,ADDRESS 5
2530 011450 062737 000005 002270      ADD     #5,ICRHX ;...
2531 011456 011137 002272      MOV      (R1),IDRX ;LOAD ADDRESS 6
2532 011462 062737 000006 002272      ADD     #6,IDRX ;...
2533 011470 013737 002272 002274      MOV      IDRX,IDRLX ;LOAD LOW BYTE LABLE OF ADDRESS 6 (PPR)
2534 011476 011137 002276      MOV      (R1),IDRHX ;LOAD HIGH BYTE,ADDRESS 7 (DOR)
2535 011502 062737 000007 002276      ADD     #7,IDRHX ;...
2536 011510 011137 002300      MOV      (R1),CSRX ;LOAD ADDRESS OF CONTROL&STATUS REGISTER
2537 011514 062737 000010 002300      ADD     #10,CSRX ;...
2538 011522 011137 002302      MOV      (R1),BARX ;LOAD ADDRESS OF BUS ADDRESS REGISTER
2539 011526 062737 000012 002302      ADD     #12,BARX ;...
2540 011534 011137 002304      MOV      (R1),BCRX ;LOAD ADDRESS OF BYTE COUNT REGISTER
2541 011540 062737 000014 002304      ADD     #14,BCRX ;...
2542 011546 011137 002306      MOV      (R1),MCRX ;LOAD ADDRESS OF MATCH CHARACTER REGISTER
2543 011552 062737 000016 002306      ADD     #16,MCRX ;...
2544 011560 012137 002310      MOV      (R1)+,MCRHX ;LOAD HIGH BYTE OF MCR REGISTER
2545 011564 062737 000017 002310      ADD     #17,MCRHX ;...
2546 011572 011100      MOV      (R1),R0 ;GET VECTOR
2547 011574 032700 000007      BIT      #7,R0 ;DOES THIS VECTOR END IN NON - ZERO ?
2548 011600 001414      BEQ     11$ ;IF NOT - OK (XX0)
2549 011602 042711 000007      BIC     #7,(R1) ;MAKE IT XX0
2550 011606 000240      PRINTB #FINIT2,(R1),R0 ;INFORM THE USER

```





```

2599
2600
2601
2602
2603
2604
2605
2606
2607
2608 012230
      012230
2609
2616 012230
      012230 012746 000340
      012234 012746 010132
      012240 012746 000004
      012244 012746 000003
      012250 104437
      012252 062706 000010
2617 012256 005037 002326
2618 012262 005777 167762
2619
2620
2621
2622
2623
2624
2625 012266 005737 002326
2626 012272 001407
2627 012274
      012274 013700 002372
      012300 104451
2628 012302
      012302 104444
2629 012304
      012304 012700 000004
      012310 104436
2630 012312
2631 012312
      012312
      012312 104461

```

.SBTTL AUTODROP SECTION

```

:++
: THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
: THE "ADR" FLAG WAS SET. THE UNIT(S) UNDER TEST ARE CHECKED TO
: SEE IF THEY WILL RESPOND. THOSE THAT DON'T ARE IMMEDIATELY
: DROPPED FROM TESTING.
:--

```

BGNAUTO

LSAUTO::

SETVEC #4,#NXM,#PRI07 ;SET UP NON -EXISTENT MEMORY TRAP VECTOR.

```

MOV #PRI07,-(SP)
MOV #NXM,-(SP)
MOV #4,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP

```

```

CLR NXMFLG ;CLEAR NON -EXISTENT MEMORY FLAG
TST @IIRX ;REFERENCE MEMORY ADDRESS FOR THE DEVICE
           ;TO SEE IF IT EXISTS.

```

```

:*****
: IF THE DEVICE DOESN'T EXIST THE RESULTANT TRAP TO VECTOR 04 WILL
: CAUSE THE DEVICE TO BE DROPPED (SEE INTERRUPT ROUTINE)
: OTHERWISE THE MEMORY REFERENCE IS UNEVENTFUL AND THE DEVICE IS READY.
:*****

```

```

TST NXMFLG ;WAS THERE A TRAP ?
BEQ 10$ ;BR IF NOT
DODU LOGDEV ;DROP THE DEVICE

```

```

MOV LOGDEV,RO
TRAP C$DODU

```

DOCLN ;CLEAN UP CODE.

CLRVEC #4 ;RETURN VECTOR 04 TO NORMAL STATE

```

TRAP C$DCLN
MOV #4,RO
TRAP C$CVEC

```

10\$:

ENDAUTO

L10021:

TRAP C\$AUTO

MISCELLANEOUS SECTIONS MACRO M1113 06-SEP-82 16:46 PAGE 40  
 CLEANUP CODING SECTION

```

2633          .SBTTL  CLEANUP CODING SECTION
2634
2635          :++
2636          : THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
2637          : AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
2638          :--
2639
2640 012314          BGNCLN
      012314
2641
2650 012314          SETPRI  #PRI07          ;DISABLE INTERRUPTS
      012314 012700 000340
      012320 104441
2651 012322          BRESET          ;RESET THE IEX11
      012322 104433
2652 012324          CLRVEC  #4          ;RETURN VECTOR 04 TO NORMAL STATE
      012324 012700 000004
      012330 104436
2653
2654 012332          EXIT  CLN
      012332 104432
      012334 000002
2655
2667
2668          .EVEN
2669
2670 012336          ENDCLN
      012336
      012336 104412
  
```

L\$CLEAN::

```

      MOV  #PRI07,RO
      TRAP C$SPRI
      TRAP C$RESET
      MOV  #4,RO
      TRAP C$CVEC
      TRAP C$EXIT
      .WORD L10022-
  
```

L10022: TRAP C\$CLEAN



2672  
2673  
2674  
2675  
2676  
2677  
2678  
2679  
2680  
2681  
2682  
2683  
2684  
2685  
2686  
2695  
2696  
2697  
2698  
2710  
2711  
2712  
2713  
2714

012340  
012340  
012340  
012344  
012350  
012354  
012356  
012360  
012364  
012364  
012366  
012370  
012373  
012376  
012401  
012404  
012407  
012412  
012415  
012420  
012420  
012420

013746  
012746  
012746  
010600  
104417  
062706  
000167  
000030  
045  
101  
111  
045  
045  
104  
120  
104  
116  
125  
124  
104  
101  
122  
120  
000

.SBTTL DROP UNIT SECTION  
:++  
: THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE  
: TO NO LONGER BE TESTED.  
: SUPVSR DOES THE "DROPPING". THIS IS JUST TO TELL THE MAN.  
: "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADC"  
: COMMAND, OTHERWISE REMAIN INACTIVE. THE DISPLAY COMMAND  
: WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE  
: WHICH ARE STILL ACTIVE.  
:--

BGN DU  
PRINTF #FMDROP,LOGDEV ;UNIT DROPPED  
EXIT DU  
FMDROP: .ASCIZ /%N%AUNIT %D2%A DROPPED/  
.EVEN  
ENDDU

L\$DU::

MOV LOGDEV, -(SP)  
MOV #FMDROP, -(SP)  
MOV #2, -(SP)  
MOV SP, R0  
TRAP C\$PNTF  
ADD #6, SP  
.WORD J\$JMP  
.WORD L10023-2-

L10023:

TRAP C\$DU

MISCELLANEOUS SECTIONS MACRO M1113 06-SEP-82 16:46 PAGE 42  
ADD UNIT SECTION

2716  
2717  
2718  
2719  
2720  
2721  
2722  
2723  
2724  
2725  
2734  
2735  
2736  
2748  
2749  
2750  
2751  
2752  
2753

012422  
012422  
012422 000167  
012424 000000  
012426  
012426  
012426 104452  
012430

.SBTTL ADD UNIT SECTION

..++  
: THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES  
: TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK  
: TO THE TEST CYCLE.  
:--

BGNAU

L\$AU::

EXIT AU

.WORD JSJMP  
.WORD L10024-2-

.EVEN

ENDAU

L10024: TRAP C\$AU

ENDMOD

2756  
2767  
2803  
2804 012430  
2805  
2806  
2807  
2808  
2809  
2810  
2811  
2812  
2813  
2814  
2815  
2816  
2817  
2818  
2819  
2820  
2827  
2833  
2834 012430  
2835 012430  
2836 012434  
2837 012436  
012436  
012442  
012446  
012450  
012452  
2838  
2845 012456  
2846 012462  
012462  
012466  
012472  
012476  
012502  
012504  
2847 012510  
2848 012514  
2849 012520  
2850 012524  
2851 012526  
2852 012532  
012532  
2853 012534  
2854 012536  
2855 012540  
2856 012542  
2857 012544  
2858 012546  
012546  
012550  
012552

.TITLE HARDWARE TESTS  
.SBTTL TEST 1: REGISTER ADDRESSING TEST

BGNMOD

\*\*\*\*\*  
TEST 1 -IEX11  
:VERIFY THAT ADDRESSING THE 8 BUS DEVICE REGISTERS DOES NOT CAUSE  
:A NON-EXISTENT MEMORY TRAP.  
:AN ERROR IN THIS TEST COULD MEAN THAT THE DEVICE IS INCORRECTLY  
:CONFIGURED OR THAT THE ADDRESS IS WRONG.  
:COMMUNICATION BETWEEN THE MAIN CPU AND THE IEX11 IS ACCOMPLISHED  
:THROUGH A SET OF SIXTEEN REGISTERS.THE SIXTEEN REGISTERS ARE  
:ASSIGNED ADDRESSES IN THE I/O PAGE.  
\*\*\*\*\*

BGNTST

T1::  
TST PNTF ;IS THE PNT FLAG SET  
BEQ 7\$ ;IF YES, PRINT THE TEST HEADER  
PRINTF #TSHD1  
MOV #TSHD1,-(SP)  
MOV #1,-(SP)  
MOV SP,R0  
TRAP C\$PNTF  
ADD #4,SP  
7\$: CLR ITRCNT ;CLEAR ITERATION COUNTER  
SETVEC #4,#LOCATE,#PRI07 ;  
MOV #PRI07,-(SP)  
MOV #LOCATE,-(SP)  
MOV #4,-(SP)  
MOV #3,-(SP)  
TRAP C\$SVEC  
ADD #10,SP  
ITRAC1: MOV ISRX,R1 ;GET REGISTER ADDRESS  
SUB #2,R1 ;  
MOV #8.,R2 ;SET COUNTER FOR 8 REGISTER ADDRESSES  
CLR R3 ;  
10\$: ADD #2,R1 ;GET NEXT REGISTER ADDRESS  
BGNSEG TRAP C\$BSEG  
CLR R4 ;  
TST (R1) ;TEST REGISTER ADDRESS  
TST R4 ;WAS THERE A TRAP  
BEQ 20\$ ;BRANCH IF NO  
INC R3 ;  
ERRHRD 101,E101,ERR101 ;  
TRAP C\$ERHRD  
.WORD 101  
.WORD E101



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 43-1  
TEST 1: REGISTER ADDRESSING TEST

HA  
TE

```

2859 012554 003454                                .WORD  ERR101
                                20$:  ENDSEG
                                10000$:
2860 012556 104405                                TRAP  C$ESEG
012556 005302                                :ALL REGISTER ADDRESSES TESTED ?
2861 012560 001361                                :BRANCH IF YES
2862 012562 005703                                :
2863 012564 001404                                :
2864 012570 013700 002372                                :
012570 104451                                :
                                MOV  LOGDEV,RO
2865 012576 104444                                TRAP  C$DODU
012576 005737 002234                                :
2866 012600 001007                                DOCLN  :
                                TRAP  C$DCLN
2867 012604 005237 002322                                30$:  TST  QVP
                                :IS QUICK VERIFY PASS SELECTED ?
2868 012606 023737 002320 002322                                :IF YES EXIT TEST
2869 012612 000732                                :ITERATION COUNTER +1
2870 012620 012700 000004                                :DEFAULT ITERATION EXECUTED
2871 012622 104436                                :IF YES EXIT TEST
2872 012624 104436                                :IF NO TEST ITERATION
                                EXQV1: CLRVEC #4
                                MOV  #4,RO
                                TRAP  C$CVEC
2873 012632 104432                                EXIT TST
                                TRAP  C$EXIT
2874 012634 000046                                .WORD  L10025-.
                                BGNSRV LOCATE
                                :SERVICE ROUTINE LOCATE
                                LOCATE::
2875 012636 005204                                INC  R4
2876 012640 000002                                :
                                ENDSRV
                                L10026:
                                RTI
2877 012642 045 123 062 TSHD1: .NLIST BEX
2889 .ASCIZ /%S2%AREGISTER ADDRESSING TEST%N/
2891 .LIST BEX
2892 .EVEN
2893 012702 104401                                ENDTST
                                L10025:
2894 012702 104401                                TRAP  C$ETST
2895

```

```

2902 .SBTTL TEST 2: INITIALIZATION TEST
2903 .....
2904 IEX - TEST 2
2905 :RESET THE IEX AND ENSURE THAT REGISTERS CSR,IIR,ISR,ICR,IDR,MCR :
2906 :IN BOTH CHANNELS ARE IN THEIR PROPER INITIALIZATION STATE .
2907 .....
2908 :REGISTERS BAR & BCR ARE NOT IN A DEFINATE STATE AFTER RESET
2909 :SO THEY ARE NOT TESTED HERE
2910 .....
2911 :THE MUX BIT IN CSR IS ALSO TESTED
2912 .....
2913 BGNSTST
2914 012704 005737 002324 TST PNTF :IS THE PNT FLAG SET
2915 012710 001410 BEQ 7$ :IF YES, PRINT THE TEST HEADER
2916 012712 PRINTF #TSHD2 :...
      012712 012746 013367 MOV #TSHD2,-(SP)
      012716 012746 000001 MOV #1,-(SP)
      012722 010600 MOV SP,R0
      012724 104417 TRAP C$PNTF
      012726 062706 000004 ADD #4,SP
2917 012732 005037 002322 7$: CLR ITRCNT :CLEAR ITERATION COUNTER
2918 012736 ITRAC2: BRESET :CLEAR HARDWARE
      012736 104433 TRAP C$RESET
2919 012740 012737 000001 002374 MOV #1,CHAN :LOAD CANNEL NUM.
2920 012746 BGNSEG
      012746 104404 TRAP C$BSEG
2921 012750 017737 167324 002502 MOV @CSRX,BAD :TEST THE INITIALIATION STATE OF CSR REGISTER
2922 012756 042737 017060 002502 BIC #17060,BAD :BIT 9-12 AND BIT 4+5 ARE NOT TESTED
2923 012764 005737 002502 TST BAD :CSR SHOULD BE ZERO
2924 012770 001406 BEQ 10$ :BRANCH IF DATA CORRECT
2925 012772 005037 002500 CLR GOOD :SET UP DATA FOR ERROR MESSAGE
2926 012776 ERRSOF T 201,E200,ERR201 :ERROR HANDLER
      012776 104457 TRAP C$ERSOF T
      013000 000311 .WORD 201
      013002 005057 .WORD E200
      013004 003500 .WORD ERR201
2927 013006 104405 10000$: ENDSEG TRAP C$ESEG
2928 013010 104404 13$: BGNSEG TRAP C$BSEG
2929 013012 017737 167232 002502 MOV @IIRX,BAD :GET IIR CONTENTS
2930 013020 005737 002502 TST BAD :CONTENTS SHOULD BE ZERO
2931 013024 001410 BEQ 20$ :BRANCH IS YES
2932 013026 005037 002500 CLR GOOD :SET UP DATA FOR ERROR MESSAGE
2933 013032 012702 013336 MOV #IIRNAM,R2 :...
2934 013036 ERRSOF T 202,E200,ERR202 :ERROR HANDLER
      013036 104457 TRAP C$ERSOF T
      013040 000312 .WORD 202
      013042 005057 .WORD E200
      013044 003556 .WORD ERR202
2935 013046 104405 20$: ENDSEG 10001$: TRAP C$ESEG
2936 013050 BGNSEG TRAP C$BSEG
      013050 104404 TRAP C$BSEG

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 44-1  
TEST 2: INITIALIZATION TEST

2937	013052	017737	167200	002502	MOV	@ISR,X,BAD	:GET ISR CONTENTS		
2938	013060	042737	000001	002502	BIC	#1,BAD	:CLEAR ULPA BIT		
2939	013066	005737	002502		TST	BAD	:CONTENTS SHOULD BE ZERO		
2940	013072	001410			BEQ	30\$	:BRANCH IF YES		
2941	013074	005037	002500		CLR	GOOD	:SET UP DATA FOR ERROR MESSAGE		
2942	013100	012702	013343		MOV	#ISR,NAM,R2	:		
2943	013104				ERRSOFT	203,E200,ERR202	:ERROR HANDLER		
	013104	104457						TRAP	C\$ERSOFT
	013106	000313						.WORD	203
	013110	005057						.WORD	E200
	013112	003556						.WORD	ERR202
2944	013114				30\$:	ENDSEG			
	013114							10002\$:	
	013114	104405						TRAP	C\$ESEG
2945	013116				BGNSEG			TRAP	C\$BSEG
	013116	104404							
2946	013120	117737	167140	002502	MOVB	@ICR,X,BAD	:GET ICR CONTENTS		
2947	013126	005737	002502		TST	BAD	:CONTENTS SHOULD BE ZERO		
2948	013132	001410			BEQ	40\$	:BRANCH IF YES		
2949	013134	005037	002500		CLR	GOOD	:SET UP DATA FOR ERROR MESSAGE		
2950	013140	012702	013350		MOV	#ICR,NAM,R2	:		
2951	013144				ERRSOFT	204,E200,ERR202	:ERROR HANDLER		
	013144	104457						TRAP	C\$ERSOFT
	013146	000314						.WORD	204
	013150	005057						.WORD	E200
	013152	003556						.WORD	ERR202
2952	013154				40\$:	ENDSEG			
	013154							10003\$:	
	013154	104405						TRAP	C\$ESEG
2953	013156				BGNSEG			TRAP	C\$BSEG
	013156	104404							
2954	013160	017737	167122	002502	MOV	@MCR,X,BAD	:GET MCR CONTENTS		
2955	013166	005737	002502		TST	BAD	:CONTENTS SHOULD BE ZERO		
2956	013172	001410			BEQ	50\$	:BRANCH IF YES		
2957	013174	005037	002500		CLR	GOOD	:SET UP DATA FOR ERROR MESSAGE		
2958	013200	012702	013362		MOV	#MCR,NAM,R2	:		
2959	013204				ERRSOFT	205,E200,ERR202	:ERROR HANDLER		
	013204	104457						TRAP	C\$ERSOFT
	013206	000315						.WORD	205
	013210	005057						.WORD	E200
	013212	003556						.WORD	ERR202
2960	013214				50\$:	ENDSEG			
	013214							10004\$:	
	013214	104405						TRAP	C\$ESEG
2961	013216	022737	000002	002374	CMP	#2,CHAN	:LOOK AT CHANNEL FLAG		
2962	013224	001427			BEQ	61\$	:EXIT IF CHANNEL 2 WAS SELECTED		
2963	013226	052777	000010	167044	BIS	#10,@CSR,X	:SET MUX BIT		
2964	013234	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL FLAG		
2965	013242	017737	167032	002502	MOV	@CSR,X,BAD	:GET CSR2 CONTENTS		
2966	013250	042737	017060	002502	BIC	#17060,BAD	:IGNORE BIT 9-12 AND BIT 4+5		
2967	013256	022737	000010	002502	CMP	#10,BAD	:MUX BIT SHOULD BE SET		
2968	013264	001651			BEQ	13\$	:BRANCH IF YES		
2969	013266	012737	000010	002500	MOV	#10,GOOD	:GET GOOD DATA		
2970	013274				ERRSOFT	206,E200,ERR201	:ERROR HANDLER		
	013274	104457						TRAP	C\$ERSOFT
	013276	000316						.WORD	206
	013300	005057						.WORD	E200





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 45  
TEST 3: R/W BIT TEST

2989  
2990  
2991  
2992  
2993  
2994  
2995  
2996  
2997  
2998  
2999 013424  
3000 013424 005737 002324  
3001 013430 001410  
3002 013432  
013432 012746 014047  
013436 012746 000001  
013442 010600  
013444 104417  
013446 062706 000004  
3003 013452 005037 002322  
3004 013456  
013456 012700 000340  
013462 104441  
3005 013464  
013464 104433  
3006 013466 005037 002502  
3007 013472 012737 000001 002374  
3008 013500  
013500 104407  
3009 013502  
013502 103004  
3010 013504 012737 017167 002350  
3011 013512 000403  
3012 013514 012737 000167 002350  
3013 013522 013737 002300 002354  
3014 013530 012737 000455 002506  
3015 013536 004737 006320  
3016 013542  
013542 104404  
3017 013544 012777 000107 166526  
3018 013552 052777 000400 166520  
3019 013560 117737 166514 002502  
3020 013566 105777 166506  
3021 013572 001410  
3022 013574 005037 002500  
3023 013600 012702 014042  
3024 013604  
013604 104457  
013606 000460  
013610 005162  
013612 003556  
3025 013614  
013614  
013614 104405  
3026 013616 012737 177767 002350  
3027 013624 013737 002302 002354

```
.SBTTL TEST 3: R/W BIT TEST
:*****
: IEX - TEST 3
: THIS TEST CHECKS ALL R/W BITS OF CSR, BAR, BCR AND MCR REGISTER
: IN BOTH CHANNELS. IT ALSO TESTS THE MASTER CLEAR FUNCTION IN
: CSR1 + CSR2.
: THE TMS 9914 REGISTERS IIR, ISR, ICR, IDR ARE NOT CHECKED
: IN THIS TEST.
:*****
```

```
BGNTST
T3::
TST PNTF ; IS THE PNT FLAG SET
BEQ 7$ ; IF YES, PRINT THE TEST HEADER
PRINTF #TSHD3 ; ...
MOV #TSHD3, -(SP)
MOV #1, -(SP)
MOV SP, R0
TRAP C$PNTF
ADD #4, SP
7$: CLR ITRCNT ; CLEAR ITERATION COUNTER
SETPRI #PRI07 ; INTERRUPT NOT ALLOWED
MOV #PRI07, R0
TRAP C$SPRI
ITRAC3: BRESET ; RESET ALL HARDWARE
TRAP C$RESET
CLR BAD ; CLEAR ERROR LABEL BAD
MOV #1, CHAN ; LOAD CHANNEL NO.
T3SEC: READBUS ; DETERMINE BUS TYPE
TRAP C$RDBU
BNCOMPLETE UNIMSK ; BRANCH IF UNI-BUS
BCC UNIMSK
MOV #17167, MASK ; LOAD BIT MASK FOR Q-BUS DEVICE
BR .+10 ;
UNIMSK: MOV #167, MASK ; LOAD BIT MASK FOR UNI-BUS DEVICE
MOV CSRX, REGADD ; LOAD REGISTER ADDRESS
MOV #301, ERRNBR ; FIRST ERROR NUMBER
JSR PC, REGTST ; CALL REGISTER TEST
BGNSEG
TRAP C$BSEG
MOV #107, @CSRX ; SET ALL R/W BITS THAT CAN BE CLEARED BY MC
BIS #400, @CSRX ; MASTER CLEAR
MOVB @CSRX, BAD ; GET CSR CONTENTS
TSTB @CSRX ; ALL BITS CLEARED
BEQ 10$ ; BRANCH IF YES
CLR GOOD ; SET UP DATA FOR ERROR MESSAGE
MOV #CSRNAM, R2 ; ...
ERRSOFT 304, E302, ERR202
TRAP C$ERSOFT
WORD 304
WORD E302
WORD ERR202
10$: ENDSEG
10000$: TRAP C$ESEG
MOV #177767, MASK ; BIT MASK OF R/W BITS
MOV BARX, REGADD ; GET REGISTER ADDRESS OF BAR
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 45-1  
TEST 3: R/W BIT TEST

```

3028 013632 012737 000461 002506      MOV      #305,ERRNBR      ;FIRST ERROR NUMBER
3029 013640 004737 006320              JSR      PC,REGTS1       ;CALL REGISTER TEST
3030 013644 013737 002304 002354      MOV      BCRX,REGADD     ;GET REGISTER ADDRESS OF BCR
3031 013652 012737 000464 002506      MOV      #308,ERRNBR     ;FIRST ERROR NUMBER
3032 013660 004737 006320              JSR      PC,REGTS1       ;CALL REGISTER TEST
3033 013664 012737 137767 002350      MOV      #137767,MASK    ;BIT MASK OF R/W BITS
3034 013672 013737 002306 002354      MOV      MCRX,REGADD     ;GET REGISTER ADDRESS OF MCR
3035 013700 012737 000467 002506      MOV      #311,ERRNBR     ;FIRST ERROR NUMBER
3036 013706 004737 006320              JSR      PC,REGTS1       ;CALL REGISTER TEST
3037 013712 032737 000001 002374      BIT      #1,CHAN         ;WAS CHANNEL 2 SELECTED
3038 013720 001433                    BEQ      QVT3            ;IF YES JUMP TO QUICK VERIFY PASS
3039 013722 012737 000002 002374      MOV      #2,CHAN         ;SET CHANNEL FLAG
3040 013730 005037 002502              CLR      BAD             ;CLEAR ERROR LABEL BAD
3041 013734                    BGNSEG
3042 013734 104404                    TRAP    C$BSEG
3042 013736 012777 000003 166334      MOV      #3,@CSRX        ;SELECT CHANNEL 2
3043 013744 117737 166330 002502      MOVB     @CSRX,BAD       ;GET CSR CONTENTS
3044 013752 032737 000010 002502      BIT      #10,BAD         ;IS MUX BIT SET
3045 013760 001411                    BEQ      20$             ;BRANCH IF YES
3046 013762 012702 014042              MOV      #CSRNAM,R2     ;SET UP DATA FOR ERROR MESSAGE
3047 013766 012737 000010 002500      MOV      #10,GOOD       ;...
3048 013774 ERRSOFT 314,E303
3048 013774 104457                    TRAP    C$ERSOFT
3048 013776 000472                    .WORD  314
3048 014000 005230                    .WORD  E303
3048 014002 000000                    .WORD  0
3049 014004 20$: ENDSEG
3049 014004 10001$: TRAP C$ESEG
3049 014004 104405
3050 014006 000634
3051 014010 005737 002234      QVT3:   BR      T3SEC     ;REPEAT TEST WITH CHANNEL 2
3052 014014 001010      QVT3:   TST      QVP      ;IS QUICK VERIFY PASS SELECTED
3053 014016 005237 002322      QVT3:   BNE     EXQV3     ;IF YES EXIT TEST
3054 014022 023737 002322 002320      QVT3:   INC     ITRCNT    ;ITERATION COUNTER + 1
3055 014030 001402      QVT3:   CMP     ITRCNT,ITRDEF ;DEFAULT ITERATION EXECUTED
3056 014032 000137 013464      QVT3:   BEQ     EXQV3     ;IF YES EXIT TEST
3057 014036 014036 104432      QVT3:   JMP     ITRAC3    ;IF NO TEST ITERATION
3057 014040 000034      QVT3:   EXIT    TST
3058 014042 040 103 123 CSRNAM: .NLIST BEX
3059 014047 045 123 062 TSHD3: .ASCIZ / CSR/
3060 014074 104401      TSHD3: .ASCIZ /%S2%AR-W BIT TEST%N/
3061 .LIST BEX
3062 .EVEN
3063 014074 104401      TSHD3: .ENDTST
3063 014074 L10030: TRAP C$E1ST
3063 014074

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 46  
 TEST 4 : SYSTEM CONTROLLER COMMANDS TEST

```

3065 .SBTTL TEST 4 : SYSTEM CONTROLLER COMMANDS TEST
3066 .....
3067 IEX - TEST 4
3068 :PART 1 CHANEL 1 WHICH IS SELECTED AS SYSTEM CONTROLLER SENDS
3069 THE IFC AND REN MESSAGE BY MEANS OF THE AUXILIARY COMMANDS SIC AND
3070 SRE. ALSO BOTH IIR AS WELL AS ISR REGISTER ARE CHECKED.
3071 :PART 2 CHANEL 2 WHICH IS SELECTED AS SYSTEM CONTROLLER SENDS
3072 THE IFC AND REN MESSAGE BY MEANS OF THE AUXILIARY COMMANDS SIC AND
3073 SRE. ALSO BOTH IIR AS WELL AS ISR REGISTER ARE CHECKED.
3074 .....
3075 BGNTST
3076 014076 005737 002324 TST PNTF ;IS THE PNT FLAG SET
3077 014102 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
3078 014104 PRINTF #TSHD4 ;....
3079 014124 005037 000004 MOV #TSHD4,-(SP)
3080 014130 004737 010220 MOV #1,-(SP)
3081 014134 113777 002312 166112 MOV SP,RO
3082 014142 112777 000010 166130 TRAP C$PNTF
3083 014150 113777 002314 166076 ADD #4,SP
3084 014156 104404 ITRAC4: CLR ITRCNT ;CLEAR ITERATION COUNTER
3085 014160 105077 166114 JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
3086 014164 052777 000002 166106 MOVB DPA1,@IIRHX ;LOAD DEVICE PRIM. ADDR. 1 IN ADDR 1
3087 014172 112777 000217 166070 MOVB #10,@CSRX ;SELECT CHANNEL 2
3088 014200 004737 011072 JSR PC,WAIT ;LOAD DEVICE PRIM. ADDR 2 IN ADDR REG.
3089 014204 052777 000010 166066 MOVB DPA2,@IIRHX ;SELECT CHANNEL 1
3090 014212 012737 000002 002374 CLRB @CSRX ;SELECT CHANNEL 1 AS SYSTEM CONTROLLER
3091 014220 017737 166024 002502 BIS #2,@CSRX ;--SEND INTERFACE CLEAR TO CHANNEL 1--
3092 014226 022737 000400 002502 MOVB #217,@ICRHX ;WAIT 100 US.
3093 014234 001410 JSR PC,WAIT ;SELECT CHANNEL 2
3094 014236 012737 000400 002500 BIS #10,@CSRX ;LOAD CHANNEL NO.
3095 014244 104457 000400 002500 MOV #2,CHAN ;GET IIR2 CONTENTS
3096 014254 000621 000400 002500 MOV @IIRX,BAD ;IFC BIT IN IIR 2 SHOULD BE SET
3097 014256 017737 165774 002502 10$: BEQ 10$ ;BRANCH IF YES
3098 014264 022737 121040 002502 MOV #400,BAD ;SET UP DATA FOR ERROR MESSAGE
3099 014272 001410 BEQ 10$ ;ERROR HANDLER
3100 014274 012737 121040 002500 ERRSOFT 401,E501,ERR501
3101 014302 104457 000621 TRAP C$ERRSOFT
3102 014312 104406 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET
3103 014314 042777 000010 165756 20$: MOV @ISRX,BAD ;GET ISR2 CONTENTS
3104 014322 012737 000001 002374 CMP #121040,BAD ;ATN,NDAC,IFC,ATN SHOULD BE SET
    BEQ 20$ ;BRANCH IF YES
    MOV #121040,GOOD ;SET UP DATA FOR ERROR MESSAGE
    ERRSOFT 402,E502,ERR501 ;ERROR HANDLER
    TRAP C$ERRSOFT
    .WORD 402
    .WORD E502
    .WORD ERR501
    TRAP C$CLP1
    
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 46-1  
 TEST 4 : SYSTEM CONTROLLER COMMANDS TEST

3105	014330	017737	165714	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
3106	014336	022737	000020	002502		CMP	#20,BAD		:BO BIT IN IIR1 SHOULD BE SET
3107	014344	001410				BEQ	30\$		:BRANCH IF YES
3108	014346	012737	000020	002500		MOV	#20,GOOD		:SET UP DATA FOR ERROR MESSAGES
3109	014354					ERRSOFT	403,E501,ERR501		:ERROR HANDLER
	014354	104457							TRAP C\$ERSOFT
	014356	000623							.WORD 403
	014360	005406							.WORD E501
	014362	003702							.WORD ERR501
3110	014364					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	014364	104406							TRAP C\$CLP1
3111	014366	017737	165664	002502	30\$:	MOV	@ISRX,BAD		:GET ISR1 CONTENTS
3112	014374	022737	120040	002502		CMP	#120040,BAD		:ATN,NDAC,ATN IN ISR1 SHOULD BE SET
3113	014402	001410				BEQ	40\$		:BRANCH IF YES
3114	014404	012737	120040	002500		MOV	#120040,GOOD		:SET UP DATA FOR ERROR MESSAGES
3115	014412					ERRSOFT	404,E502,ERR501		:ERROR HANDLER
	014412	104457							TRAP C\$ERSOFT
	014414	000624							.WORD 404
	014416	005447							.WORD E502
	014420	003702							.WORD ERR501
3116	014422					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	014422	104406							TRAP C\$CLP1
3117	014424	112777	000017	165636	40\$:	MOVB	#17,@ICRHX		:----LOAD NOT SIC IN ACR-----
3118	014432	017737	165612	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
3119	014440	005737	002502			TST	BAD		:IIR1 CONTENTS SHOULD BE ZERO
3120	014444	001406				BEQ	50\$		:BRANCH IF YES
3121	014446	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGES
3122	014452					ERRSOFT	405,E501,ERR501		:ERROR HANDLER
	014452	104457							TRAP C\$ERSOFT
	014454	000625							.WORD 405
	014456	005406							.WORD E501
	014460	003702							.WORD ERR501
3123	014462				50\$:	ENDSEG			
	014462								10000\$:
	014462	104405							TRAP C\$ESEG
3124	014464					BGNSEG			TRAP C\$BSEG
	014464	104404							
3125	014466	042777	000010	165604		BIC	#10,@CSRX		:SELECT CHANNEL 1
3126	014474	112777	000220	165566		MOVB	#220,@ICRHX		:----LOAD SRE IN ICR1-----
3127	014502	004737	011072			JSR	PC,WAIT		:WAIT 100 US.
3128	014506	052777	000010	165564		BIS	#10,@CSRX		:SELECT CHANNEL 2
3129	014514	012737	000002	002374		MOV	#2,CHAN		:SET CHANNEL FLAG
3130	014522	017737	165522	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
3131	014530	005737	002502			TST	BAD		:IIR2 SHOULD BE ZERO
3132	014534	001407				BEQ	11\$		:BRANCH IF YES
3133	014536	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGES
3134	014542					ERRSOFT	406,E501,ERR501		:ERROR HANDLER
	014542	104457							TRAP C\$ERSOFT
	014544	000626							.WORD 406
	014546	005406							.WORD E501
	014550	003702							.WORD ERR501
3135	014552					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	014552	104406							TRAP C\$CLP1
3136	014554	017737	165476	002502	11\$:	MOV	@ISRX,BAD		:GET ISR2 CONTENTS
3137	014562	022737	120440	002502		CMP	#120440,BAD		:ATN,NDAC,REN,ATN IN ISR2 SHOULD SET
3138	014570	001410				BEQ	12\$		:BRANCH IF YES
3139	014572	012737	120440	002500		MOV	#120440,GOOD		:SET UP DATA FOR ERROR MESSAGES







HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 46-3  
TEST 4 : SYSTEM CONTROLLER COMMANDS TEST

```

3175 015042 017737 165210 002502 15$:  MOV @ISRX,BAD          ;GET ISR1 CONTENTS
3176 015050 022737 121040 002502      CMP #121040,BAD      ;ATN,NDAC,IFC,ATN SHOULD BE SET
3177 015056 001410                    BEQ 16$              ;BRANCH IF YES
3178 015060 012737 121040 002500      MOV #121040,GOOD    ;SET UP DATA FOR ERROR MESSAGES
3179 015066 104457                      ERRSOFT 411,E502,ERR501 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 411
                                .WORD E502
                                .WORD ERR501
3180 015076 104406                      CKLOOP              ;BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
3181 015100 052777 000010 165172 16$:  BIS #10,@CSRX        ;SELECT CHANNEL 2
3182 015106 012737 000002 002374      MOV #2,CHAN         ;LOAD CHANNEL FLAG
3183 015114 017737 165130 002502      MOV @IIRX,BAD       ;GET IIR2 CONTENTS
3184 015122 022737 000020 002502      CMP #20,BAD         ;BO BIT IN IIR2 SHOULD BE SET
3185 015130 001410                    BEQ 17$              ;BRANCH IF YES
3186 015132 012737 000020 002500      MOV #20,GOOD        ;SET UP DATA FOR ERROR MESSAGES
3187 015140 104457                      ERRSOFT 412,E501,ERR501 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 412
                                .WORD E501
                                .WORD ERR501
3188 015150 104406                      CKLOOP              ;BRANCH TO BGNSEG IF ERRLOPP IS SET
                                TRAP C$CLP1
3189 015152 112777 000017 165110 17$:  MOVB #17,@ICRHX     ;----LOAD NOT SIC IN ICR2 (ACR)----
3190 015160 017737 165064 002502      MOV @IIRX,BAD       ;GET IIR2 CONTENTS
3191 015166 005737 002502              TST BAD              ;IIR2 SHOULD BE ZERO
3192 015172 001406                    BEQ 21$              ;BRANCH IF YES
3193 015174 005037 002500              CLR GOOD             ;SET UP DATA FOR ERROR MESSAGES
3194 015200 104457                      ERRSOFT 413,E501,ERR501 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 413
                                .WORD E501
                                .WORD ERR501
3195 015210 104405                      21$:  ENDSEG              10002$:
3196 015212 104404                      BGNSEG              TRAP C$ESEG
                                TRAP C$BSEG
3197 015214 052777 000010 165056      BIS #10,@CSRX        ;SELECT CHANNEL 2
3198 015222 112777 000220 165040      MOVB #220,@ICRHX    ;----LOAD SRE IN ACR 2-----
3199 015230 004737 011072              JSR PC,WAIT          ;WAIT 100 US.
3200 015234 042777 000010 165036      BIC #10,@CSRX        ;SELECT CHANNEL 1
3201 015242 012737 000001 002374      MOV #1,CHAN         ;LOAD CHANNEL FLAG
3202 015250 017737 164774 002502      MOV @IIRX,BAD       ;GET IIR1 CONTENTS
3203 015256 005737 002502              TST BAD              ;IIR 1 SHOULD BE ZERO
3204 015262 001407                    BEQ 22$              ;BRANCH IF YES
3205 015264 005037 002500              CLR GOOD             ;SET UP DATA FOR ERROR MESSAGES
3206 015270 104457                      ERRSOFT 414,E501,ERR501 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 414
                                .WORD E501
                                .WORD ERR501
3207 015300 104406                      CKLOOP              ;BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
3208 015302 017737 164750 002502 22$:  MOV @ISRX,BAD          ;GET ISR1 CONTENTS
3209 015310 022737 120440 002502      CMP #120440,BAD     ;ATN,NDAC,REN,ATN BITS SHOULD BE SET

```

```

3210 015316 001410
3211 015320 012737 120440 002500 BEQ 23$ :BRANCH IF YES
3212 015326 104457 MOV #120440,GOOD :SET UP DATA FOR ERROR MESSAGES
ERRSOFT 415,E502,ERR501 :ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 415
                                .WORD E502
                                .WORD ERR501
3213 015336 104406 CKLOOP :BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
3214 015340 052777 000010 164732 23$: BIS #10,@CSRX :SELECT CHANNEL 2
3215 015346 012737 000002 002374 MOV #2,CHAN
3216 015354 017737 164670 002502 MOV @IIRX,BAD
3217 015362 005737 002502 TST BAD :GET IIR2 CONTENTS
3218 015366 001407 BEQ 24$ :IIR 2 SHOULD BE ZERO
3219 015370 005037 002500 CLR GOOD :BRANCH IF YES
3220 015374 104457 ERRSOFT 416,E501,ERR501 :SET UP DATA FOR ERROR MESSAGES
                                :ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 416
                                .WORD E501
                                .WORD ERR501
3221 015404 104406 CKLOOP :BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
3222 015406 017737 164644 002502 24$: MOV @ISRX,BAD :GET ISR2 CONTENTS
3223 015414 022737 120440 002502 CMP #120440,BAD :ATN,NDAC,REN,ATN SHOULD BE SET
3224 015422 001410 BEQ 25$ :BRANCH IF YES
3225 015424 012737 120440 002500 MOV #120440,GOOD :SET UP DATA FOR ERROR MESSAGES
3226 015432 104457 ERRSOFT 417,E502,ERR501 :ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 417
                                .WORD E502
                                .WORD ERR501
3227 015442 104406 CKLOOP :BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
3228 015444 112777 000020 164616 25$: MOVB #20,@ICRHX :----LOAD NOT SRE IN ICR2 (ACR)----
3229 015452 112777 000200 164610 MOVB #200,@ICRHX :----LOAD SWRST IN ICR2 (ACR)----
3230 015460 112777 000000 164602 MOVB #0,@ICRHX :----LOAD NOT SWRST IN ICR2 (ACR)----
3231 015466 104405 10003$:
3232 015470 005737 002234 TST QVP TRAP C$ESEG
3233 015474 001010 BNE EXQV4 :IS QUICK VERIFY PASS SELECTED
3234 015476 005237 002322 INC ITRCNT :IF YES EXIT TEST
3235 015502 023737 002320 002322 CMP ITRDEF,ITRCNT :ITERATION COUNTER + 1
3236 015510 001402 BEQ EXQV4 :DEFAULT ITERATION EXECUTED
3237 015512 000137 014130 JMP ITRAC4 :IF YES EXIT TEST
3238 015516 104432 EXQV4: EXIT TST :IF NO TEST ITERATION
                                :
                                TRAP C$EXIT
                                .WORD L10031-.
3239 015520 000052
3240 015522 045 123 062 TSHD4: .NLIST BEX
3241 .ASCIZ /%S2%ASYSTEM CONTROLLER COMMANDS TEST%/
3242 .LIST BEX
3243 .EVEN
3244 015572 104401 ENDTST
                                L10031:
                                TRAP C$ETST

```



```

3246 .SBTTL TEST 5: INTERRUPT TEST
3247 *****
3248 IEX - TEST 5
3249 :PART 1 CHECKS THE DEVICE PRIORITY LEVEL AND
3250 THE FUNCTION OF INTERRUPT SEQUENCE IN CHANNEL 1. INITIATING THIS
3251 SEQUENCE WILL BE DONE BY SETTING THE INT ENB, DMA ENB BITS IN CSR1
3252 AND BO BIT IN IIR1 REGISTER.
3253
3254 :PART 2 CHECKS THE DEVICE PRIORITY LEVEL AND
3255 THE FUNCTION OF INTERRUPT SEQUENCE IN CHANNEL 2. INITIATING THIS
3256 SEQUENCE WILL BE DONE BY SETTING THE INT ENB, DMA ENB BITS IN CSR2
3257 AND BO BIT IN IIR2 REGISTER.
3258 *****
3259 015574 BGNTST
015574
3260 015574 005737 002324 TST PNTF ;IS THE PNT FLAG SET
3261 015600 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
3262 015602 PRINTF #TSHD5 ;...
015602 012746 016732 MOV #TSHD5,-(SP)
015606 012746 000001 MOV #1,-(SP)
015612 010600 MOV SP,RO
015614 104417 TRAP C$PNTF
015616 062706 000004 ADD #4,SP
3263 015622 005037 002322 7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
3264 015626 004737 010220 ITRAC5: JSR PC,CULPA ;RESET HARDWARE
3265 015632 004737 010534 JSR PC,BGIN1 ;SET UP PARAMETER
3266 015636 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NO.
3267 015644 BGNSEG
015644 104404 TRAP C$BSEG
3268 015646 005037 002376 CLR INTFC1 ;CLEAR INTERRUPT FLAG FOR CHANNEL 1
3269 015652 005037 002400 CLR INTFC2 ;CLEAR INTERRUPT FLAG FOR CHANNEL 2
3270 015656 SETVEC VECC1,#INTSC1,#PRI07 ;SET VECTOR FOR CHA.1
015656 012746 000340 MOV #PRI07,-(SP)
015662 012746 010142 MOV #INTSC1,-(SP)
015666 013746 002244 MOV VECC1,-(SP)
015672 012746 000003 MOV #3,-(SP)
015676 104437 TRAP C$$VEC
015700 062706 000010 ADD #10,SP
3271 015704 SETVEC VECC2,#INTERR,#PRI07 ;
015704 012746 000340 MOV #PRI07,-(SP)
015710 012746 010162 MOV #INTERR,-(SP)
015714 013746 002246 MOV VECC2,-(SP)
015720 012746 000003 MOV #3,-(SP)
015724 104437 TRAP C$$VEC
015726 062706 000010 ADD #10,SP
3272 015732 SETPRI #PRI07 ;NO INTERRUPT
015732 012700 000340 MOV #PRI07,RO
015736 104441 TRAP C$$PRI
3273 015740 112777 000020 164312 MOVB #20,@ISRLX ;:::SET BO BIT IN ISR1
3274 015746 052777 000101 164324 BIS #101,@CSRX ;:::SET INT ENB AND DMA ENB BIT IN CSR
3275 015754 012701 000010 MOV #10,R1 ;:::SET PRIORITY COUNTER
3276 .LIST MEB
3277 015760 PRIT INTFC1,501,ERR402 ;:::TRY WITH INTERRUPT
015760 005737 002376 TST INTFC1 ;INTERRUPT OCCURED?
015764 001040 BNE 64$ ;BRANCH IF YES
015766 005301 DEC R1 ;CHECKSUM = 7
015770 012700 000300 MOV #PRI06,RO
    
```



```

015774 104441
015776 005737 002376      TST   INTFC1      ;INTERRUPT OCCURED?
016002 001031              BNE   64$         ;BRANCH IF YES
016004 005301              DEC   R1          ;CHECKSUM = 6
016006 012700 000240      MOV   #PRI05,RO
016012 104441              TRAP  C$SPRI
016014 005737 002376      TST   INTFC1      ;INTERRUPT OCCURED?
016020 001022              BNE   64$         ;BRANCH IF YES
016022 005301              DEC   R1          ;CHECKSUM = 5
016024 012700 000200      MOV   #PRI04,RO
016030 104441              TRAP  C$SPRI
016032 005737 002376      TST   INTFC1      ;INTERRUPT OCCURED?
016036 001013              BNE   64$         ;BRANCH IF YES
016040 005301              DEC   R1          ;CHECKSUM = 4
016042 012700 000140      MOV   #PRI03,RO
016046 104441              TRAP  C$SPRI
016050 005737 002376      TST   INTFC1      ;INTERRUPT OCCURED?
016054 001004              BNE   64$         ;BRANCH IF YES
016056 104457              TRAP  C$ERSOFT
016060 000765              .WORD 501
016062 005316              .WORD E402
016064 003654              .WORD ERR402
016066 000240
3278 64$: NOP
3279 .NLIST MEB
016070 012700 000340      SETPRI #PRI07      ;SET PRIORITY
016074 104441              MOV   #PRI07,RO
016076 106301              TRAP  C$SPRI
3280 016076 106301      ASLB  R1           ;CREATE CORRECT PRI. FOR COMPARISON
3281 016100 106301      ASLB  R1           ;...
3282 016102 106301      ASLB  R1           ;...
3283 016104 106301      ASLB  R1           ;...
3284 016106 106301      ASLB  R1           ;...
3285 016110 020137 002316      CMP   R1,PLEV     ;CHECK INTERRUPT PRIORITY
3286 016114 001412      BEQ   20$         ;BRANCH IF INTERRUPT WAS OK
3287 016116 010137 002502      MOV   R1,BAD     ;SET UP DATA FOR ERROR MESSAGE
3288 016122 013737 002316 002500      MOV   PLEV,GOOD
3289 016130      ERRSOFT 502,E403,ERR401 ;ERROR HANDLER
016130 104457              TRAP  C$ERSOFT
016132 000766              .WORD 502
016134 005353              .WORD E403
016136 003616              .WORD ERR401
3290 016140      CKLOOP           ;BRANCH BACK TO BGNSEG IF ERROR LOOP SET
016140 104406              TRAP  C$CLP1
3291 016142      20$:
3292 016142 017737 164132 002502      MOV   @CSRX,BAD  ;GET CSR1 CONTENTS
3293 016150 042737 017000 002502      BIC   #17000,BAD ;IGNORE BIT 9-12
3294 016156 022737 000202 002502      CMP   #202,BAD  ;INT,SYS CONT SHOULD BE SET
3295 016164 001410      BEQ   30$         ;BRANCH IF YES
3296 016166 012737 000202 002500      MOV   #202,GOOD ;SET UP DATA FOR ERROR MESSAGE
3297 016174      ERRSOFT 503,E401,ERR501 ;ERROR HANDLER
016174 104457              TRAP  C$ERSOFT
016176 000767              .WORD 503
016200 005265              .WORD E401
016202 003702              .WORD ERR501
3298 016204      CKLOOP           ;BRANCH TO BGNSEG IF ERRLOOP IS SET
016204 104406              TRAP  C$CLP1
3299 016206      30$:

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 47-2  
 TEST 5: INTERRUPT TEST

```

3300 016206 017737 164036 002502      MOV    @IIRX,BAD      ;GET IIR1 CONTENTS
3301 016214 022737 000220 002502      CMP    #220,BAD      ;INTO,BO SHOULD BE SET
3302 016222 001407                BEQ    33$            ;BRANCH IF YES
3303 016224 012737 000220 002500      MOV    #220,GOOD     ;SET UP DATA FOR ERROR MESSAGE
3304 016232                ERRSOFT 504,E501,ERR501 ;ERROR HANDLER
                                TRAP    CSERSOFT
                                .WORD   504
                                .WORD   E501
                                .WORD   ERR501
                                10000$:
3305 016242                33$:  ENDSEG
                                TRAP    C$ESEG
                                MOV     VECC2,RO
3306 016244 013700 002246      CLRVEC VECC2          ;RESTORE INTERRUPT VECTOR
                                TRAP    C$CVEC
                                MOV     VECC2,RO
                                TRAP    C$CVEC
                                ;(DISABLE INTERRUPT)
3307 016252                CLRVEC VECC1          ;...
3308 016252 013700 002244      MOV     VECC1,RO
                                TRAP    C$CVEC
                                MOV     VECC1,RO
                                TRAP    C$CVEC
3309
3310 :+++-----
3311 :PART 2 CHECK THE INTERRUPT FUNCTION AND THE PRIORITY LEVEL OF CHANNEL 2
3312 :+++-----
3312 016260 004737 010710      JSR    PC,BGIN2      ;SET UP PARAMETER FOR CHAN 2
3313 016264                SETVEC VECC2,#INTSC2,#PRI07 ;SET VECTOR FOR CHANNEL 2
                                MOV     #PRI07,-(SP)
                                MOV     #INTSC2,-(SP)
                                MOV     VECC2,-(SP)
                                MOV     #3,-(SP)
                                TRAP    C$SVEC
                                ADD     #10,SP
3314 016312                SETVEC VECC1,#INTERR,#PRI07 ;SET VECTOR FOR CHANNEL 1
                                MOV     #PRI07,-(SP)
                                MOV     #INTERR,-(SP)
                                MOV     VECC1,-(SP)
                                MOV     #3,-(SP)
                                TRAP    C$SVEC
                                ADD     #10,SP
3315 016340                BGNSEG
                                TRAP    C$BSEG
3316 016342 005037 002376      CLR    INTFC1        ;CLEAR OLD FLAG
3317 016346 005037 002400      CLR    INTFC2
3318 016352                SETPRI #PRI07        ;NO INTERRUPT
                                MOV     #PRI07,RO
                                TRAP    C$SPRI
3319 016360 112777 000020 163672      MOVB   #20,@ISRLX    ;:::SET BO BIT IN ISR2 REGISTER
3320 016366 052777 000101 163704      BIS    #101,@CSRX    ;:::SET INT ENB,DMA ENB IN CSR2
3321 016374 012701 000010                MOV    #10,R1        ;:::SET PRIORITY COUNTER
3322
3323 016400                .LIST MEB
                                PRIT   INTFC2,505,ERR402 ;:::TRY INTERRUPT
                                TST    INTFC2          ;INTERRUPT OCCURED?
                                BNE    65$            ;BRANCH IF YES
                                DEC    R1              ;CHECKSUM = 7
                                MOV     #PRI06,RO
                                TRAP    C$SPRI
3324 016400 005737 002400      TST    INTFC2        ;INTERRUPT OCCURED?
3325 016404 001040                BNE    65$            ;BRANCH IF YES
3326 016406 005301                DEC    R1              ;CHECKSUM = 6
3327 016410 012700 000300      TST    INTFC2
3328 016414 104441                BNE    65$
3329 016416 005737 002400      DEC    R1
3330 016422 001031
3331 016424 005301
    
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 47-3  
TEST 5: INTERRUPT TEST

```

016426 012700 000240                                MOV #PRI05,RO
016432 104441                                TRAP C$SPRI
016434 005737 002400                                TST INTFC2 ;INTERRUPT OCCURED?
016440 001022                                BNE 65$ ;BRANCH IF YES
016442 005301                                DEC R1 ;CHECKSUM = 5
016444 012700 000200                                MOV #PRI04,RO
016450 104441                                TRAP C$SPRI
016452 005737 002400                                TST INTFC2 ;INTERRUPT OCCURED?
016456 001013                                BNE 65$ ;BRANCH IF YES
016460 005301                                DEC R1 ;CHECKSUM = 4
016462 012700 000140                                MOV #PRI03,RO
016466 104441                                TRAP C$SPRI
016470 005737 002400                                TST INTFC2 ;INTERRUPT OCCURED?
016474 001004                                BNE 65$ ;BRANCH IF YES
016476 104457                                TRAP C$ERSOFT
016500 000771                                .WORD 505
016502 005316                                .WORD E402
016504 003654                                .WORD ERR402
016506 000240
3324 3325 016510 012700 000340 65$: NOP ;SET PRIORITY
.NLIST MEB SETPRI #PRI07
016510 012700 000340                                MOV #PRI07,RO
016514 104441                                TRAP C$SPRI
3326 016516 106301                                ASLB R1 ;CREATE CORRECT PRI. FOR COMPARISON
3327 016520 106301                                ASLB R1 ;...
3328 016522 106301                                ASLB R1 ;...
3329 016524 106301                                ASLB R1 ;...
3330 016526 106301                                ASLB R1 ;...
3331 016530 020137 002316                                CMP R1,PLEV ;CHECK INTERRUPT PRIORITY
3332 016534 001412                                BEQ 40$ ;BRANCH IF INTERRUPT PRI.WAS OK
3333 016536 010137 002502                                MOV R1,BAD ;SET UP DATA FOR ERROR MESSAGE
3334 016542 013737 002316 002500                                MOV PLEV,GOOD
3335 016550 ERRSOFT 506,E403,ERR401 ;ERROR HANDLER
016550 104457                                TRAP C$ERSOFT
016552 000772                                .WORD 506
016554 005353                                .WORD E403
016556 003616                                .WORD ERR401
3336 016560 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET
016560 104406                                TRAP C$CLP1
3337 016562 017737 163512 002502 40$: MOV @CSRX,BAD ;GET CSR2 CONTENTS
3338 016562 042737 017000 002502                                BIC #17000,BAD ;IGNORE BIT 9-12
3339 016570 022737 000212 002502                                CMP #212,BAD ;INT,SYS CONT,MUX SHOULD BE SET
3340 016576 001410                                BEQ 43$ ;BRANCH IF YES
3341 016604 012737 000212 002500                                MOV #212,GOOD ;SET UP DATA FOR ERROR MESSAGE
3342 016606 012737 000212 002500                                MOV #212,GOOD
3343 016614 ERRSOFT 507,E401,ERR501 ;ERROR HANDLER
016614 104457                                TRAP C$ERSOFT
016616 000773                                .WORD 507
016620 005265                                .WORD E401
016622 003702                                .WORD ERR501
3344 016624 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP SET
016624 104406                                TRAP C$CLP1
3345 016626 017737 163416 002502 43$: MOV @IIRX,BAD ;GET IIR2 CONTENTS
3346 016626 022737 000220 002502                                CMP #220,BAD ;INTO,BO BIT SHOULD BE SET
3347 016634 001407                                BEQ 50$ ;BRANCH IF YES
3348 016642 012737 000220 002500                                MOV #220,GOOD ;SET UP DATA FOR ERROR MESSAGE
3349 016644

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 47-4  
TEST 5: INTERRUPT TEST

```

3350 016652          ERRSOF T 508,E501,ERR501          ;ERROR HANDLER
      016652 104457
      016654 000774
      016656 005406
      016660 003702
3351 016662          50$:  ENDSEG
      016662
      016662 104405
3352 016664          CLRVEC  VECC1          ;RESTORE VECTOR (DICABLE INTERRUPT)
      016664 013700 002244
      016670 104436
3353 016672          CLRVEC  VECC2          ;...
      016672 013700 002246
      016676 104436
3354 016700          TST      QVP          ;IS QUICK VERIFY PASS SELECTED
3355 016704          BNE      EXQV5        ;IF YES EXIT TEST
3356 016706          INC      ITRCNT       ;ITERATION COUNTER + 1
3357 016712          CMP      ITRDEF,ITRCNT ;DEFAULT ITERATION EXECUTED
3358 016720          BEQ      EXQV5        ;IF YES EXIT TEST
3359 016722          JMP      ITRAC5       ;IF NO TEST ITERATION
3360 016726          EXQV5:  EXIT TST
      016726 104432
      016730 000030
3361
3362
3363 016732          045    123    062  TSHD5:  .NLIST  BEX
3364          .ASCIZ  /%S2%AINTERRUPT TEST%/
3365          .LIST  BEX
3366 016760          .EVEN
      016760 104401          .ENDTST
                                     L10032:
                                     TRAP  C$ETST

```

TRAP C\$ERSOFT  
.WORD 508  
.WORD E501  
.WORD ERR501

10001\$:  
TRAP C\$ESEG  
MOV VECC1,RO  
TRAP C\$CVEC  
MOV VECC2,RO  
TRAP C\$CVEC

: IS QUICK VERIFY PASS SELECTED  
: IF YES EXIT TEST  
: ITERATION COUNTER + 1  
: DEFAULT ITERATION EXECUTED  
: IF YES EXIT TEST  
: IF NO TEST ITERATION

TRAP C\$EXIT  
.WORD L10032-

L10032:  
TRAP C\$ETST

```

3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386 016762
      016762
3387 016762 005737 002324
3388 016766 001410
3389 016770
      016770 012746 023000
      016774 012746 000001
      017000 010600
      017002 104417
      017004 062706 000004
3390 017010 005037 002322
3391 017014 004737 010220
3392 017020 004737 010710
3393 017024 013701 002312
3394 017030 062701 000040
3395 017034 010137 002410
3396 017040 032737 000001 002312
3397 017046 001420
3398 017050 052737 000001 002434
3399 017056 052737 000001 002436
3400 017064 052737 000001 002440
3401 017072 052737 000001 002442
3402 017100 052737 000001 002444
3403 017106 000417
3404 017110 042737 000001 002434
3405 017116 042737 000001 002436
3406 017124 042737 000001 002440
3407 017132 042737 000001 002442
3408 017140 042737 000001 002444
3409 017146
      017146 104404
3410 017150 052777 000010 163122
3411 017156 012737 000002 002374
3412 017164 113777 002410 163104
3413 017172 004737 011060
3414 017176 017737 163046 002502
3415 017204 022737 000020 002502
3416 017212 001410
3417 017214 012737 000020 002500

```

```

.SBTTL TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1
.....
IEX - TEST 6
:PART 1 CHECKS THE CORRECT FUNCTION OF ADDRESS REGISTER 1 (ADR) BY LOADING ITS
:DEVICE PRIMARY ADDRESS INTO BIT A1-A5 AND RECEIVING THE ASSIGNED
:LISTEN OR TALK ADDRESS VIA THE IEC/IEEE BUS.
:NOTE: THE ULPA BIT IN THE ISR1 REGISTER IS DEPENDENT
:ON THE STATUS OF DPA1 (ODD DPA1 => ULPA IS SET)
:PART 2 CHECKS THE FUNCTION OF THE DAT BIT (DISABLES THE TALK FUNCTION),
:THE DAL BIT (DISABLES THE LISTEN FUNCTION) AND THE EDPA BIT (ENABLES
:THE DUAL PRIMARY ADDRESSING MODE, WHICH ARE ALSO PRESENT IN THE
:ADR1 REGISTER.
:IF THE QUICK VERIFY PASS IS NOT SELECTED, THE TEST ITERATION WILL DO
:IT WITH DIFFERENT DPA'S.
.....

```

```

BGNTST
T6::
TST PNTF ;IS THE PNT FLAG SET
BEQ 7$ ;IF YES, PRINT THE TEST HEADER
PRINTF #TSHD6 ;...
MOV #TSHD6,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #4,SP

7$: CLR ITRCNT ;CLEAR COUNTER
JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
JSR PC,BGIN2 ;SET UP PARAMETER
A1: MOV DPA1,R1 ;GET DPA1 ADDRESS
ADD #40,R1 ;CREATE MY LISTENER ADDRESS (MLA)
MOV R1,MLA ;STORE MLA
BIT #1,DPA1 ;IS DPA EVEN
BEQ 3$ ;BRANCH IF YES
BIS #1,CDAT1 ;SET ULPA BIT IN COMPARE DATA FOR ISR
BIS #1,CDAT2
BIS #1,CDAT3
BIS #1,CDAT4
BIS #1,CDAT5
BR +40 ;BRANCH TO BGNSEG
3$: BIC #1,CDAT1 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
BIC #1,CDAT2
BIC #1,CDAT3
BIC #1,CDAT4
BIC #1,CDAT5
BGNSEG
TRAP C$BSEG

BIS #10,@CSRX ;SELECT CHANNEL 2
MOV #2,CHAN ;LOAD CHANNEL NUMBER
MOVB MLA1,@IDRHX ;-----LOAD MLA1 INTO DOR 2-----
JSR PC,LOOP ;WAIT A LITTLE
MOV @IIRX,BAD ;GET IIR2 CONTENTS
CMP #20,BAD ;BO BIT SHOULD BE SET
BEQ 10$ ;BRANCH IF YES
MOV #20,GOOD ;SET UP DATA FOR ERROR MESSAGES

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-1  
 TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

3418	017222					ERRSOFT 601,E501,ERR501		:ERROR HANDLER	
	017222	104457						TRAP	C\$ERSOFT
	017224	001131						.WORD	601
	017226	005406						.WORD	E501
	017230	003702						.WORD	ERR501
3419	017232					CKLOOP		:BRANCH TO BGNSEG WHEN ERROR LOOP IS SET	
	017232	104406						TRAP	C\$CLP1
3420	017234	017737	163016	002502	10\$:	MOV @ISR2,BAD		:GET ISR2 CONTENTS	
3421	017242	022737	120040	002502		CMP #120040,BAD		:ATN,ATN,NDAC BIT OF ISR2 SHOULD BE SET	
3422	017250	001410				BEQ 20\$		:BRANCH IF YES	
3423	017252	012737	120040	002500		MOV #120040,GOOD		:SET UP DATA FOR ERROR MESSAGES	
3424	017260					ERRSOFT 602,E502,ERR501		:ERROR HANDLER	
	017260	104457						TRAP	C\$ERSOFT
	017262	001132						.WORD	602
	017264	005447						.WORD	E502
	017266	003702						.WORD	ERR501
3425	017270					CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET	
	017270	104406						TRAP	C\$CLP1
3426	017272	042777	000010	163000	20\$:	BIC #10,@CSRX		:SELECT CHANNEL 1	
3427	017300	012737	000001	002374		MOV #1,CHAN		:LOAD CHANNEL NUMBER	
3428	017306	017737	162736	002502		MOV @IIR1,BAD		:GET IIR1 CONTENTS	
3429	017314	022737	002401	002502		CMP #2401,BAD		:MA,MAC,IFC,BIT IN IIR1 SHOULD BE SET	
3430	017322	001410				BEQ 30\$		:BRANCH IF YES	
3431	017324	012737	002401	002500		MOV #2401,GOOD		:SET UP DATA FOR ERROR MESSAGES	
3432	017332					ERRSOFT 603,E501,ERR501		:ERROR HANDLER	
	017332	104457						TRAP	C\$ERSOFT
	017334	001133						.WORD	603
	017336	005406						.WORD	E501
	017340	003702						.WORD	ERR501
3433	017342					CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET	
	017342	104406						TRAP	C\$CLP1
3434	017344	017737	162706	002502	30\$:	MOV @ISR1,BAD		:GET ISR1 CONTENTS	
3435	017352	023737	002444	002502		CMP CDAT5,BAD		:ATN,ATN,LADS,NDAC LPAS,(ULPA) BIT SET	
3436	017360	001410				BEQ 40\$		:BRANCH IF YES	
3437	017362	013737	002444	002500		MOV CDAT5,GOOD		:SET UP DATA FOR ERROR MESSAGES	
3438	017370					ERRSOFT 604,E502,ERR501		:ERROR HANDLER	
	017370	104457						TRAP	C\$ERSOFT
	017372	001134						.WORD	604
	017374	005447						.WORD	E502
	017376	003702						.WORD	ERR501
3439	017400					CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET	
	017400	104406						TRAP	C\$CLP1
3440	017402	052777	000010	162670	40\$:	BIS #10,@CSRX		:SELECT CHANNEL 2	
3441	017410	012737	000002	002374		MOV #2,CHAN		:LOAD CHANNEL NUMBER	
3442	017416	112777	000077	162652		MOVB #77,@IDRHX		:-----LOAD UNL INTO DOR-----	
3443	017424	004737	011060			JSR PC,LOOP		:WAIT A LITTLE	
3444	017430	017737	162614	002502		MOV @IIR2,BAD		:GET IIR2 CONTENTS	
3445	017436	022737	000020	002502		CMP #20,BAD		:CHECK BO BIT IN IIR2	
3446	017444	001410				BEQ 50\$		:BRANCH IF YES	
3447	017446	012737	000020	002500		MOV #20,GOOD		:SET UP DATA FOR ERROR MESSAGES	
3448	017454					ERRSOFT 605,E501,ERR501		:ERROR HANDLER	
	017454	104457						TRAP	C\$ERSOFT
	017456	001135						.WORD	605
	017460	005406						.WORD	E501
	017462	003702						.WORD	ERR501
3449	017464					CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET	
	017464	104406						TRAP	C\$CLP1



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-2  
 TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

```

3450 017466 017737 162564 002502 50$: MOV @ISRX,BAD ;GET ISR2 CONTENTS
3451 017474 022737 120040 002502 CMP #120040,BAD ;ATN,ATN,NDAC BIT SHOULD BE SET
3452 017502 001410 BEQ 60$ ;BRANCH IF YES
3453 017504 012737 120040 002500 MOV #120040,GOOD ;SET UP DATA FOR ERROR MESSAGES
3454 017512 ERRSOFT 606,E502,ERR501 ;ERROR HANDLER
      017512 104457 TRAP CSERSOFT
      017514 001136 .WORD 606
      017516 005447 .WORD E502
      017520 003702 .WORD ERR501
3455 017522 CKLOOP ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      017522 104406 TRAP C$CLP1
3456 017524 042777 000010 162546 60$: BIC #10,@CSRX ;SELECT CHANNEL 1
3457 017532 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
3458 017540 017737 162504 002502 MOV @IIRX,BAD ;GET IIR1 CONTENTS
3459 017546 122737 000001 002502 CMPB #1,BAD ;MAC BIT IN IIR1 SHOULD BE SET
3460 017554 001410 BEQ 11$ ;BRANCH IF YES
3461 017556 012737 000001 002500 MOV #1,GOOD ;SET UP DATA FOR ERROR MESSAGES
3462 017564 ERRSOFT 607,E501,ERR501 ;ERROR HANDLER
      017564 104457 TRAP CSERSOFT
      017566 001137 .WORD 607
      017570 005406 .WORD E501
      017572 003702 .WORD ERR501
3463 017574 CKLOOP ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      017574 104406 TRAP C$CLP1
3464 017576 017737 162454 002502 11$: MOV @ISRX,BAD ;GET ISR1 CONTENTS
3465 017604 023737 002434 002502 CMP CDAT1,BAD ;ATN,ATN,NDAC,(ULPA) BIT SHOULD BE SET
3466 017612 001407 BEQ 12$ ;BRANCH IF YES
3467 017614 013737 002434 002500 MOV CDAT1,GOOD ;SET UP DATA FOR ERROR MESSAGES
3468 017622 ERRSOFT 608,E502,ERR501 ;ERROR HANDLER
      017622 104457 TRAP CSERSOFT
      017624 001140 .WORD 608
      017626 005447 .WORD E502
      017630 003702 .WORD ERR501
3469 017632 12$: ENDSEG
      017632 104405 10000$: TRAP C$ESEG
3470 017634 BGNSEG TRAP C$BSEG
      017634 104404
3471 017636 052777 000010 162434 BIS #10,@CSRX ;SELECT CHANNEL 2
3472 017644 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
3473 017652 013701 002312 MOV DPA1,R1 ;LOAD DPA1 IN R1
3474 017656 062701 000100 ADD #100,R1 ;CREATE MY TALKER ADDRESS (MTA)
3475 017662 010137 002414 MOV R1,MTA1 ;SAVE MTA
3476 017666 ENDSEG
      017666 104405 10001$: TRAP C$ESEG
3477 017670 BGNSEG TRAP C$BSEG
      017670 104404
3478 017672 113777 002414 162376 MOVB MTA1,@IDRHX ;---LOAD TALKER ADDR. INTO DOR 2-----
3479 017700 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
3480 017704 017737 162340 002502 MOV @IIRX,BAD ;GET IIR2 CONTENTS
3481 017712 022737 000020 002502 CMP #20,BAD ;BO BIT IN IIR2 SHOULD BE SET
3482 017720 001410 BEQ 13$ ;BRANCH IF YES
3483 017722 012737 000020 002500 MOV #20,GOOD ;SET UP DATA FOR ERROR MESSAGES
3484 017730 ERRSOFT 609,E501,ERR501 ;ERROR HANDLER
      017730 104457 TRAP CSERSOFT
      017732 001141 .WORD 609

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-4  
 TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

```

3519 020212 012737 120040 002500      MOV      #120040,GOOD      ;SET UP DATA FOR ERROR MESSAGES
3520 020220      ERRSOFT 614,E502,ERR501 ;ERROR HANDLER
      020220 104457      TRAP      C$ERSOFT
      020222 001146      .WORD    614
      020224 005447      .WORD    E502
      020226 003702      .WORD    ERR501
3521 020230      CKLOOP      ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      020230 104406      TRAP      C$CLP1
3522 020232 042777 000010 162040 22$: BIC      #10,@CSRX      ;SELECT CHANNEL 1
3523 020240 012737 000001 002374      MOV      #1,CHAN      ;LOAD CHANNEL NUMBER
3524 020246 017737 161776 002502      MOV      @IIRX,BAD     ;GET IIR1 CONTENTS
3525 020254 022737 000001 002502      CMP      #1,BAD       ;MAC BIT IN IIR1 SHOULD BE SET
3526 020262 001410      BEQ      23$          ;BRANCH IF YES
3527 020264 012737 000001 002500      MOV      #1,GOOD      ;SET UP DATA FOR ERROR MESSAGES
3528 020272      ERRSOFT 615,E501,ERR501 ;ERROR HANDLER
      020272 104457      TRAP      C$ERSOFT
      020274 001147      .WORD    615
      020276 005406      .WORD    E501
      020300 003702      .WORD    ERR501
3529 020302      CKLOOP      ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      020302 104406      TRAP      C$CLP1
3530 020304 017737 161746 002502 23$: MOV      @ISRX,BAD     ;GET ISR1 CONTENTS
3531 020312 023737 002434 002502      CMP      CDAT1,BAD     ;ATN,ATN,NDAC,(ULPA)BITS SHOULD BE SET
3532 020320 001407      BEQ      24$          ;BRANCH IF YES
3533 020322 013737 002434 002500      MOV      CDAT1,GOOD    ;SET UP DATA FOR ERROR MESSAGES
3534 020330      ERRSOFT 616,E502,ERR501 ;ERROR HANDLER
      020330 104457      TRAP      C$ERSOFT
      020332 001150      .WORD    616
      020334 005447      .WORD    E502
      020336 003702      .WORD    ERR501
3535 020340      24$: ENDSEG
      020340      10002$: TRAP      C$ESEG
      020340 104405
3536      ;-----
3537      ;PART 2 OF THE ADDRESS REGISTER TEST.
3538      ;THIS PART CHECKS THE "DAL","DAT","EDPA" BITS IN THE IEEE
3539      ; ADDRESS REGISTER.
3540      ;-----
3541 020342 013701 002312      MOV      DPA1,R1      ;GET DEVICE PRIM. ADDR. 1
3542 020346 052701 000040      BIS      #40,R1      ;ADD DAT BIT TO DAP1
3543 020352 110177 161676      MOV      R1,@IIRHX   ;LOAD DEVICE PRIM. ADDR.1 + DAT BIT
3544 020356      BGNSEG      TRAP      C$BSEG
      020356 104404
3545 020360 052777 000010 161712      BIS      #10,@CSRX   ;SELECT CHANNEL 2
3546 020366 012737 000002 002374      MOV      #2,CHAN     ;LOAD CHANNEL NUMBER
3547 020374 113777 002410 161674      MOV      MLA1,@IDRHX ;----LOAD MLA IN DOR REGISTER-----
3548 020402 004737 011060      JSR      PC,LOOP     ;WAIT A LITTLE
3549 020406 042777 000010 161664      BIC      #10,@CSRX   ;SELECT CHANNEL 1
3550 020414 012737 000001 002374      MOV      #1,CHAN     ;LOAD CHANNEL NUMBER
3551 020422 017737 161630 002502      MOV      @ISRX,BAD   ;GET ISR1 CONTENTS
3552 020430 023737 002444 002502      CMP      CDAT5,BAD   ;ATN,ATN,LADS,NDAC,LPAS,(ULPA)BIT SET
3553 020436 001410      BEQ      40$         ;BRANCH IF YES
3554 020440 013737 002444 002500      MOV      CDAT5,GOOD  ;SET UP DATA FOR ERROR MESSAGES
3555 020446      ERRSOFT 617,E502,ERR501 ;ERROR HANDLER
      020446 104457      TRAP      C$ERSOFT
      020450 001151      .WORD    617
      020452 005447      .WORD    E502
    
```

COPA1:





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-6  
 TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

	020766	104457								TRAP	C\$ERSOFT
	020770	001154								.WORD	620
	020772	005447								.WORD	E502
	020774	003702								.WORD	ERR501
3597	020776				36\$:	ENDSEG					
	020776								10005\$:		
3598	021000	104405				BGNSEG				TRAP	C\$ESEG
	021000	104404								TRAP	C\$BSEG
3599	021002	052777	000010	161270		BIS	#10,@CSRX			:SELECT CHANNEL 2	
3600	021010	012737	000002	002374		MOV	#2,CHAN			:GET CHANNEL NUMBER	
3601	021016	113777	002414	161252		MOVB	MTA1,@IDRHX			:----LOAD MTA1 INTO DOR 2-----	
3602	021024	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE	
3603	021030	042777	000010	161242		BIC	#10,@CSRX			:SELECT CHANNEL 1	
3604	021036	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER	
3605	021044	017737	161206	002502		MOV	@ISR1,BAD			:GET ISR1 CONTENTS	
3606	021052	023737	002440	002502		CMP	CDAT3,BAD			:ATN,ATN,TADS,NDAC,TPAS,(ULPA) BITS SET	
3607	021060	001410				BEQ	16\$			:BRANCH IF YES	
3608	021062	013737	002440	002500		MOV	CDAT3,GOOD			:SET UP DATA FOR ERROR MESSAGE	
3609	021070					ERRSOFT	621,E502,ERR501			:ERROR HANDLER	
	021070	104457								TRAP	C\$ERSOFT
	021072	001155								.WORD	621
	021074	005447								.WORD	E502
	021076	003702								.WORD	ERR501
3610	021100					CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET	
	021100	104406								TRAP	C\$CLP1
3611	021102	052777	000010	161170	16\$:	BIS	#10,@CSRX			:SELECT CHANNEL 2	
3612	021110	012737	000002	002374		MOV	#2,CHAN			:LOAD CHANNEL NUMBER	
3613	021116	112777	000137	161152		MOVB	#137,@IDRHX			:----LOAD UNT INTO DOR 2-----	
3614	021124	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE	
3615	021130	042777	000010	161142		BIC	#10,@CSRX			:SELECT CHANNEL 1	
3616	021136	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER	
3617	021144	017737	161106	002502		MOV	@ISR1,BAD			:GET ISR1 CONTENTS	
3618	021152	023737	002434	002502		CMP	CDAT1,BAD			:ATN,ATN,NDAC,(ULPA)BITS SHOULD BE SET	
3619	021160	001407				BEQ	24\$			:BRANCH IF YES	
3620	021162	013737	002434	002500		MOV	CDAT1,GOOD			:SET UP DATA FOR ERROR MESSAGES	
3621	021170					ERRSOFT	622,E502,ERR501			:ERROR HANDLER	
	021170	104457								TRAP	C\$ERSOFT
	021172	001156								.WORD	622
	021174	005447								.WORD	E502
	021176	003702								.WORD	ERR501
3622	021200				24\$:	ENDSEG					
	021200								10006\$:		
3623	021202	022737	000036	002312		CMP	#36,DPA1			:IS LAST DPA1 ADDRESS EXECUTED	
3624	021210	001002				BNE	25\$			:BRANCH IF NO	
3625	021212	000137	022636			JMP	TQVP6			:BRANCH IF YES	
3626	021216	042777	000010	161054	25\$:	BIC	#10,@CSRX			:SELECT CHANNEL 1	
3627	021224	013701	002312			MOV	DPA1,R1			:GET DPA1	
3628	021230	052701	000200			BIS	#200,R1			:SET EDPA BIT	
3629	021234	110177	161014			MOVB	R1,@IDRHX			:----LOAD DPA1 & EDPA BIT INTO ADR 1----	
3630	021240					BGNSEG					
	021240	104404								TRAP	C\$BSEG
3631	021242	052777	000010	161030		BIS	#10,@CSRX			:SELECT CHANNEL 2	
3632	021250	012737	000002	002374		MOV	#2,CHAN			:LOAD CHANNEL NUMBER	
3633	021256	113777	002410	161012		MOVB	MLA1,@IDRHX			:----LOAD MLA1 INTO DOR 2-----	
3634	021264	004737	011060		COPB1:	JSR	PC,LOOP			:WAIT A LITTLE	



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-7  
TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

3635	021270	042777	000010	161002	BIC	#10,@CSRX	:SELECT CHANNEL 1
3636	021276	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
3637	021304	017737	160746	002502	MOV	@ISR1,BAD	:GET ISR1 CONTENTS
3638	021312	023737	002444	002502	CMP	CDAT5,BAD	:ATN,ATN,LADS,NDAC,LPAS,(ULPA) BIT SET
3639	021320	001410			BEQ	40\$	:BRANCH IF YES
3640	021322	013737	002444	002500	MOV	CDAT5,GOOD	:SET UP DATA FOR ERROR MESSAGES
3641	021330				ERRSOFT	623,E502,ERR501	:ERROR HANDLER
	021330	104457					TRAP C\$ERSOFT
	021332	001157					.WORD 623
	021334	005447					.WORD E502
	021336	003702					.WORD ERR501
3642	021340				CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	021340	104406					TRAP C\$CLP1
3643	021342	052777	000010	160730	40\$: BIS	#10,@CSRX	:SELECT CHANNEL 2
3644	021350	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
3645	021356	112777	000077	160712	MOVB	#77,@IDRHX	:----LOAD UNL INTO DOR 2-----
3646	021364	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
3647	021370	042777	000010	160702	BIC	#10,@CSRX	:SELECT CHANNEL 1
3648	021376	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
3649	021404	017737	160646	002502	MOV	@ISR1,BAD	:GET ISR1 CONTENTS
3650	021412	023737	002434	002502	CMP	CDAT1,BAD	:ATN,ATN,NDAC,(ULPA)BITS SHOULD BE SET
3651	021420	001407			BEQ	12\$	:BRANCH IF YES
3652	021422	013737	002434	002500	MOV	CDAT1,GOOD	:SET UP DATA FOR ERROR MESSAGES
3653	021430				ERRSOFT	624,E502,ERR501	:ERROR HANDLER
	021430	104457					TRAP C\$ERSOFT
	021432	001160					.WORD 624
	021434	005447					.WORD E502
	021436	003702					.WORD ERR501
3654	021440				12\$: ENDSEG		
	021440						10007\$: TRAP C\$ESEG
	021440	104405					
3655	021442	052777	000010	160630	BIS	#10,@CSRX	:SELECT CHANNEL 2
3656	021450	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
3657	021456	113701	002410		MOVB	MLA1,R1	:GET MY LISTENER ADDRESS
3658	021462	032737	000001	002312	BIT	#1,DPA1	:IS DPA ODD
3659	021470	001021			BNE	13\$	:BRANCH IF YES
3660	021472	105201			INCB	R1	:CHANCH MLA1 TO ODD
3661	021474	052737	000001	002434	BIS	#1,CDAT1	:SET ULPA BIT IN COMPARE DATA FOR ISR
3662	021502	052737	000001	002436	BIS	#1,CDAT2	:...
3663	021510	052737	000001	002440	BIS	#1,CDAT3	:...
3664	021516	052737	000001	002442	BIS	#1,CDAT4	:...
3665	021524	052737	000001	002444	BIS	#1,CDAT5	:...
3666	021532	000420			BR	+42	:BRANCH TO BGNSEG
3667	021534	105301			DECB	R1	:CHANGE MLA1 TO EVEN
3668	021536	042737	000001	002434	BIC	#1,CDAT1	:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
3669	021544	042737	000001	002436	BIC	#1,CDAT2	:...
3670	021552	042737	000001	002440	BIC	#1,CDAT3	:...
3671	021560	042737	000001	002442	BIC	#1,CDAT4	:...
3672	021566	042737	000001	002444	BIC	#1,CDAT5	:...
3673	021574				BGNSEG		
	021574	104404					TRAP C\$BSEG
3674	021576	052777	000010	160474	BIS	#10,@CSRX	:SELECT CHANNEL 2
3675	021604	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
3676	021612	110177	160460		MOVB	R1,@IDRHX	:----LOAD NEW MLA1 INTO DOR 2-----
3677	021616	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
3678	021622	042777	000010	160450	BIC	#10,@CSRX	:SELECT CHANNEL 1
3679	021630	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER

COPC1:



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-8  
 TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

3680	021636	017737	160414	002502	MOV	@ISR1,BAD	:GET ISR1 CONTENTS
3681	021644	023737	002444	002502	CMP	CDAT5,BAD	:ATN,ATN,LADS,NDAC,(ULPA),LPAS BITS SET
3682	021652	001410			BEQ	40\$	:BRANCH IF YES
3683	021654	013737	002444	002500	MOV	CDAT5,GOOD	:SET UP DATA FOR ERROR MESSAGES
3684	021662				ERRSOFT	625,E502,ERR501	:ERROR HANDLER
	021662	104457					TRAP C\$ERSOFT
	021664	001161					.WORD 625
	021666	005447					.WORD E502
	021670	003702					.WORD ERR501
3685	021672				CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	021672	104406					TRAP C\$CLP1
3686	021674	052777	000010	160376	40\$: BIS	#10,@CSR1	:SELECT CHANNEL 2
3687	021702	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
3688	021710	112777	000077	160360	MOVB	#77,@IDRHX	:-----LOAD UNL INTO DOR 2-----
3689	021716	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
3690	021722	042777	000010	160350	BIC	#10,@CSR1	:SELECT CHANNEL 1
3691	021730	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
3692	021736	017737	160314	002502	MOV	@ISR1,BAD	:GET ISR1 CONTENTS
3693	021744	023737	002434	002502	CMP	CDAT1,BAD	:ATN,ATN,NDAC,(ULPA)BIT SHOULD BE SET
3694	021752	001407			BEQ	12\$	:BRANCH IF YES
3695	021754	013737	002434	002500	MOV	CDAT1,GOOD	:SET UP DATA FOR ERROR MESSAGES
3696	021762				ERRSOFT	626,E502,ERR501	:ERROR HANDLER
	021762	104457					TRAP C\$ERSOFT
	021764	001162					.WORD 626
	021766	005447					.WORD E502
	021770	003702					.WORD ERR501
3697	021772				12\$: ENDSEG		
	021772						10010\$: TRAP C\$ESEG
	021772	104405					
3698	021774	032737	000001	002312	BIT	#1,DPA1	:IS DPA EVEN
3699	022002	001420			BEQ	17\$	:BRANCH IF YES
3700	022004	052737	000001	002434	BIS	#1,CDAT1	:SET ULPA BIT IN COMPARE DATA FOR ISR
3701	022012	052737	000001	002436	BIS	#1,CDAT2	:...
3702	022020	052737	000001	002440	BIS	#1,CDAT3	:...
3703	022026	052737	000001	002442	BIS	#1,CDAT4	:...
3704	022034	052737	000001	002444	BIS	#1,CDAT5	:...
3705	022042	000417			BR	+40	:BRANCH TO BGNSEG
3706	022044	042737	000001	002434	17\$: BIC	#1,CDAT1	:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
3707	022052	042737	000001	002436	BIC	#1,CDAT2	:...
3708	022060	042737	000001	002440	BIC	#1,CDAT3	:...
3709	022066	042737	000001	002442	BIC	#1,CDAT4	:...
3710	022074	042737	000001	002444	BIC	#1,CDAT5	:...
3711	022102				BGNSEG		
	022102	104404					TRAP C\$BSEG
3712	022104	052777	000010	160166	BIS	#10,@CSR1	:SELECT CHANNEL 2
3713	022112	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
3714	022120	113777	002414	160150	MOVB	MTA1,@IDRHX	:-----LOAD MTA INTO DOR 2-----
3715	022126	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
3716	022132	042777	000010	160140	BIC	#10,@CSR1	:SELECT CHANNEL 1
3717	022140	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
3718	022146	017737	160104	002502	MOV	@ISR1,BAD	:GET ISR1 CONTENTS
3719	022154	023737	002440	002502	CMP	CDAT3,BAD	:ATN,ATN,TADS,NDAC,TPAS,(ULPA) BIT SET
3720	022162	001410			BEQ	16\$	:BRANCH IF YES
3721	022164	013737	002440	002500	MOV	CDAT3,GOOD	:SET UP DATA FOR ERROR MESSAGE
3722	022172				ERRSOFT	627,E502,ERR501	:ERROR HANDLER
	022172	104457					TRAP C\$ERSOFT
	022174	001163					.WORD 627





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 48-10  
TEST 6: ADDRESS REGISTER TEST OF CHANNEL 1

```

3766 022534          104406          CKLOOP          :BRANCH TO BGNSEG WHEN ERRLOOP IS SET
          022534          104406          TRAP          C$CLP1
3767 022536 052777 000010 157534 16$: BIS #10,@CSRX :SELECT CHANNEL 2
3768 022544 012737 000002 002374 MOV #2,CHAN :LOAD CHANNEL NUMBER
3769 022552 112777 000137 157516 MOVB #137,@IDRHX :----LOAD UNIT INTO DOR 2-----
3770 022560 004737 011060 JSR PC,LOOP :WAIT A LITTLE
3771 022564 042777 000010 157506 BIC #10,@CSRX :SELECT CHANNEL 1
3772 022572 012737 000001 002374 MOV #1,CHAN :LOAD CHANNEL NUMBER
3773 022600 017737 157452 002502 MOV @ISRX,BAD :GET ISR1 CONTENTS
3774 022606 023737 002434 002502 CMP CDAT1,BAD :ATN,ATN,NDAC,(ULPA)SHOULD BE SET
3775 022614 001407 BEQ 24$ :BRANCH IF YES
3776 022616 013737 002434 002500 MOV CDAT1,GOOD :SET UP DATA FOR ERROR MESSAGES
3777 022624 ERRSOFT 630,E502,ERR501 :ERROR HANDLER
          022624 104457 TRAP C$ERSOFT
          022626 001166 .WORD 630
          022630 005447 .WORD E502
          022632 003702 .WORD ERR501
3778 022634          24$: ENDSEG
          022634          10012$: TRAP C$ESEG
          022634 104405
3779 022636 005737 002234 TQVP6: TST QVP :IS QUICK VERIFY PASS SELECTED
3780 022642 001054 BNE EXQV6 :IF YES EXIT TEST
3781 022644 005737 002322 TST ITRCNT :IS THIS THE FIRST TIME THROUGH THE TEST
3782 022650 001007 BNE 1$ :BRANCH IF NO
3783 022652 013737 002312 002406 MOV DPA1,SDPA :SAVE ENTERED DPA1
3784 022660 005037 002312 CLR DPA1 :CLR DPA1
3785 022664 005237 002322 INC ITRCNT :SET FLAG TO SEE FIRST TIME PASS
3786 022670 005237 002312 1$: IN DPA1 :GET NEW DPA1
3787 022674 023737 002312 002314 CMP DPA1,DPA2 :IS DPA1 = DPA2
3788 022702 001002 BNE 2$ :BRANCH IF NO
3789 022704 005237 002312 INC DPA1 :INCREMENT DPA1
3790 022710 022737 000037 002312 2$: CMP #37,DPA1 :ALL DONE
3791 022716 001423 BEQ 3$ :BRANCH IF YES
3792 022720 052777 000010 157352 BIS #10,@CSRX :SELECT CHANNEL 2
3793 022726 112777 000217 157334 MOVB #217,@ICRHX :----LOAD SIC INTO ACR 2
3794 022734 004737 011072 JSR PC,WAIT :WAIT A LITTLE
3795 022740 112777 000017 157322 MOVB #17,@ICRHX :----LOAD NOT SIC INTO ACR 2
3796 022746 042777 000010 157324 BIC #10,@CSRX :SELECT CHANNEL 1
3797 022754 113777 002312 157272 MOVB DPA1,@IIRHX :LOAD NEW DPA1 INTO ADR1
3798 022762 000137 017024 JMP A1 :REPEAT THE TEST WITH THE NEW DPA1
3799 022766 013737 002406 002312 3$: MOV SDPA,DPA1 :RESTOR ENTERED DPA1
3800 022774 EXQV6: EXIT TST
          022774 104432 TRAP C$EXIT
          022776 000062 .WORD L10033-
3801
3802
3803 023000 045 123 062 TSHD6: .NLIST BEX
3804 .ASCIZ /%S2%ADDRESS REGISTER TEST (ICR) OF CHANNEL 1%N/
3805 .LIST BEX
3806 023060 .EVEN
          023060 .ENDTST
          023060 104401 L10033: TRAP C$ETST

```



```

3808
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3820
3821
3822
3823
3824
3825 023062
      023062
3826 023062 005737 002324
3827 023066 001410
3828 023070
      023070 012746 027334
      023074 012746 000001
      023100 010600
      023102 104417
      023104 062706 000004
3829 023110 005037 002322
3830 023114 004737 010220
3831 023120 004737 010534
3832 023124 013701 002314
3833 023130 062701 000040
3834 023134 010137 002412
3835 023140 032737 000001 002314
3836 023146 001420
3837 023150 052737 000001 002434
3838 023156 052737 000001 002436
3839 023164 052737 000001 002440
3840 023172 052737 000001 002442
3841 023200 052737 000001 002444
3842 023206 000417
3843 023210 042737 000001 002434
3844 023216 042737 000001 002436
3845 023224 042737 000001 002440
3846 023232 042737 000001 002442
3847 023240 042737 000001 002444
3848 023246
      023246 104404
3849 023250 042777 000010 157022
3850 023256 012737 000001 002374
3851 023264 113777 002412 157004
3852 023272 004737 011060
3853 023276 017737 156746 002402
3854 023304 022737 000020 002402
3855 023312 001413
3856 023314 012737 000020 002500
3857 023322 013737 002402 002502

```

```

.SBTTL TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2
:*****
:THIS TEST IS THE SAME TEST AS TEST 6 EXCEPT THE CHANNEL IS CHANGED
:
:PART 1 CHECKS THE CORRECT FUNCTION OF ADDRESS REGISTER 2 (ADR) BY
:LOADING ITS DEVICE PRIMARY ADDRESS INTO BIT A1-A5 AND RECEIVING
:THE ASSIGNED LISTEN OR TALK ADDRESS VIA THE IEC/IEEE BUS.
:NOTE: THE ULPA BIT IN THE ISR2 REGISTER IS DEPENDENT ON THE STATUS OF
:      DPA2 (ODD DPA2 => ULPA IS SET)
:
:PART 2 CHECKS THE FUNCTION OF THE DAT BIT (DISABLES THE TALK FUNCTION),
:THE DAL BIT (DISABLES THE LISTEN FUNCTION) AND THE EDPA BIT
:(ENABLES THE DUAL PRIMARY ADDRESSING MODE), WHICH ARE ALSO
:PRESENT IN THE ADR2 REGISTER.
:IF THE QUICK VERIFY PASS IS NOT SELECTED, THE TEST ITERATION
:WILL DO IT WITH DIFFERENT DPA'S.
:*****

```

```

      BGNST
      T7::
      TST     PNTF          :IS THE PNT FLAG SET
      BEQ     7$           :IF YES, PRINT THE TEST HEADER
      PRINTF  #TSHD7       :...
                               MOV     #TSHD7,-(SP)
                               MOV     #1,-(SP)
                               MOV     SP,R0
                               TRAP    C$PNTF
                               ADD     #4,SP
7$:   CLR     ITRCNT        :CLEAR ITERATION COUNTER
      JSR     PC,CULFA      :CLEAR ULPA BIT IN ISR 1 AND 2
      JSR     PC,BGIN1      :SET UP PARAMETER
A2:   MOV     DPA2,R1       :GET DPA2 ADDRESS
      ADD     #40,R1        :CREATE MY LISTENER ADDRESS (MLA)
      MOV     R1,MLA2       :STORE MLA
      BIT     #1,DPA2       :IS DPA EVEN
      BEQ     3$           :BRANCH IF YES
      BIS     #1,CDAT1      :SET ULPA BIT IN COMPARE DATA FOR ISR
      BIS     #1,CDAT2
      BIS     #1,CDAT3
      BIS     #1,CDAT4
      BIS     #1,CDAT5
      BR     +40           :BRANCH TO BGNSEG
3$:   BIC     #1,CDAT1      :CLEAR ULPA BIT IN COMPARE DATA FOR ISR
      BIC     #1,CDAT2
      BIC     #1,CDAT3
      BIC     #1,CDAT4
      BIC     #1,CDAT5
      BGNSEG
                               TRAP    C$BSEG
      BIC     #10,@CSRX     :SELECT CHANNEL 1
      MOV     #1,CHAN       :LOAD CHANNEL NUMBER
      MOVB    MLA2,@IDRHX   :LOAD LISTENER ADDRESS OF CHANNEL 1
      JSR     PC,LOOP       :WAIT A LITTLE
      MOV     @IIRX,RSAVE   :GET IIR1 CONTENTS
      CMP     #20,RSAVE     :IS BO BIT OF IIR1 SET
      BEQ     10$          :BRANCH IF YES
      MOV     #20,GOOD      :SET UP DATA FOR ERROR MESSAGES
      MOV     RSAVE,BAD
      :...

```









HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-3  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

3927	024060	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE
3928	024064	017737	156160	002402		MOV	@IIRX,RSAVE			:GET IIR1 CONTENTS
3929	024072	022737	000020	002402		CMP	#20,RSAVE			:BO BIT IN IIR1 SHOULD BE SET
3930	024100	001413				BEQ	13\$			:BRANCH IF YES
3931	024102	012737	000020	002500		MOV	#20,GOOD			:SET UP DATA FOR ERROR MESSAGES
3932	024110	013737	002402	002502		MOV	RSAVE,BAD			:...
3933	024116					ERRSOFT	709,E501,ERR501			:ERROR HANDLER
	024116	104457								TRAP CSERSOFT
	024120	001305								.WORD 709
	024122	005406								.WORD E501
	024124	003702								.WORD ERR501
3934	024126					CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	024126	104406								TRAP C\$CLP1
3935	024130	017737	156122	002402	13\$:	MOV	@ISRX,RSAVE			:GET ISR1 CONTENTS
3936	024136	022737	120040	002402		CMP	#120040,RSAVE			:ATN,ATN,NDAC BIT IN ISR1 SHOULD BE SET
3937	024144	001413				BEQ	14\$			:BRANCH IF YES
3938	024146	012737	120040	002500		MOV	#120040,GOOD			:SET UP DATA FOR ERROR MESSAGES
3939	024154	013737	002402	002502		MOV	RSAVE,BAD			:...
3940	024162					ERRSOFT	710,E502,ERR501			:ERROR HANDLER
	024162	104457								TRAP CSERSOFT
	024164	001306								.WORD 710
	024166	005447								.WORD E502
	024170	003702								.WORD ERR501
3941	024172					CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	024172	104406								TRAP C\$CLP1
3942	024174	052777	000010	156076	14\$:	BIS	#10,@CSRX			:SELECT CHANNEL 2
3943	024202	012737	000002	002374		MOV	#2,CHAN			:LOAD CHANNEL NUMBER
3944	024210	017737	156034	002402		MOV	@IIRX,RSAVE			:GET IIR2 CONTENTS
3945	024216	022737	002001	002402		CMP	#2001,RSAVE			:MA,MAC BIT IN IIR2 SHOULD BE SET
3946	024224	001413				BEQ	15\$			:BRANCH IF YES
3947	024226	012737	002001	002500		MOV	#2001,GOOD			:SET UP DATA FOR ERROR MESSAGES
3948	024234	013737	002402	002502		MOV	RSAVE,BAD			:...
3949	024242					ERRSOFT	711,E501,ERR501			:ERROR HANDLER
	024242	104457								TRAP CSERSOFT
	024244	001307								.WORD 711
	024246	005406								.WORD E501
	024250	003702								.WORD ERR501
3950	024252					CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	024252	104406								TRAP C\$CLP1
3951	024254	017737	155776	002402	15\$:	MOV	@ISRX,RSAVE			:GET ISR2 CONTENTS
3952	024262	023737	002440	002402		CMP	CDAT3,RSAVE			:ATN,ATN,TADS,NDAC,TPAS,(ULPA) BIT SET
3953	024270	001413				BEQ	16\$			:BRANCH IF YES
3954	024272	013737	002402	002502		MOV	RSAVE,BAD			:SET UP DATA FOR ERROR MESSAGES
3955	024300	013737	002440	002500		MOV	CDAT3,GOOD			:...
3956	024306					ERRSOFT	712,E502,ERR501			:ERROR HANDLER
	024306	104457								TRAP CSERSOFT
	024310	001310								.WORD 712
	024312	005447								.WORD E502
	024314	003702								.WORD ERR501
3957	024316					CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	024316	104406								TRAP C\$CLP1
3958	024320	042777	000010	155752	16\$:	BIC	#10,@CSRX			:SELECT CHANNEL 1
3959	024326	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER
3960	024334	112777	000137	155734		MOVB	#137,@IDRHX			:LOAD UNT INTO DOR
3961	024342	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE
3962	024346	017737	155676	002402		MOV	@IIRX,RSAVE			:GET IIR1 CONTENTS
3963	024354	022737	000020	002402		CMP	#20,RSAVE			:BO BIT IN IIR1 SHOULD BE SET

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-4  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

```

3964 024362 001413          BEQ      21$          :BRANCH IF YES
3965 024364 012737 000020 002500  MOV      #20,GOOD    :SET UP DATA FOR ERROR MESSAGES
3966 024372 013737 002402 002502  MOV      RSAVE,BAD   :
3967 024400          ERRSOFT 713,E501,ERR501 :ERROR HANDLER
      024400 104457          TRAP      C$ERSOFT
      024402 001311          .WORD    713
      024404 005406          .WORD    E501
      024406 003702          .WORD    ERR501
3968 024410          CKLOOP          :BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      024410 104406          TRAP      C$CLP1
3969 024412 017737 155640 002402 21$:  MOV      @ISRX,RSAVE  :GET ISR1 CONTENTS
3970 024420 022737 120040 002402  CMP      #120040,RSAVE :ATN,ATN,NDAC IN ISR1 SHOULD BE SET
3971 024426 001413          BEQ      22$          :BRANCH IF YES
3972 024430 012737 120040 002500  MOV      #120040,GOOD :SET UP DATA FOR ERROR MESSAGES
3973 024436 013737 002402 002502  MOV      RSAVE,BAD   :
3974 024444          ERRSOFT 714,E502,ERR501 :ERROR HANDLER
      024444 104457          TRAP      C$ERSOFT
      024446 001312          .WORD    714
      024450 005447          .WORD    E502
      024452 003702          .WORD    ERR501
3975 024454          CKLOOP          :BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      024454 104406          TRAP      C$CLP1
3976 024456 052777 000010 155614 22$:  BIS      #10,@CSRX   :SELECT CHANNEL 2
3977 024464 012737 000002 002374  MOV      #2,CHAN     :LOAD CHANNEL NUMBER
3978 024472 017737 155552 002402  MOV      @IIRX,RSAVE :GET IIR2 CONTENTS
3979 024500 022737 000001 002402  CMP      #1,RSAVE    :MAC BIT IN IIR2 SHOULD BE SET
3980 024506 001413          BEQ      23$          :BRANCH IF YES
3981 024510 012737 000001 002500  MOV      #1,GOOD     :SET UP DATA FOR ERROR MESSAGES
3982 024516 013737 002402 002502  MOV      RSAVE,BAD   :
3983 024524          ERRSOFT 715,E501,ERR501 :ERROR HANDLER
      024524 104457          TRAP      C$ERSOFT
      024526 001313          .WORD    715
      024530 005406          .WORD    E501
      024532 003702          .WORD    ERR501
3984 024534          CKLOOP          :BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      024534 104406          TRAP      C$CLP1
3985 024536 017737 155514 002402 23$:  MOV      @ISRX,RSAVE  :GET ISR2 CONTENTS
3986 024544 023737 002434 002402  CMP      CDAT1,RSAVE :ATN,ATN,NDAC,(ULPA)BIT SHOULD BE SET
3987 024552 001412          BEQ      24$          :BRANCH IF YES
3988 024554 013737 002434 002500  MOV      CDAT1,GOOD  :SET UP DATA FOR ERROR MESSAGES
3989 024562 013737 002402 002502  MOV      RSAVE,BAD   :
3990 024570          ERRSOFT 716,E502,ERR501 :ERROR HANDLER
      024570 104457          TRAP      C$ERSOFT
      024572 001314          .WORD    716
      024574 005447          .WORD    E502
      024576 003702          .WORD    ERR501
3991 024600          24$:  ENDSEG          10002$:
      024600          TRAP      C$ESEG
      024600 104405
3992          :-----:
3993          :PART 2 OF THE ADDRESS REGISTER TEST.
3994          :THIS PART CHECKS THE "DAL","DAT","EDPA" BITS IN THE ADDRESS REGISTER.
3995          :-----:
3996 024602 013701 002314  MOV      DPA2,R1     :GET DEVICE PRIM. ADDR. 2
3997 024606 052701 000040  BIS      #40,R1      :ADD DAT BIT TO DAP2-
3998 024612 110177 155436  MOVB    R1,@IIRHX   :LOAD DEVICE PRIM. ADDR.2 + DAT BIT
3999 024616  BGNSEG
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-5  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

```

024616 104404
4000 024620 042777 000010 155452 COPA2: BIC #10,@CSRX ;SELECT CHANNEL 1
4001 024626 113777 002412 155442 MOVB MLA2,@IDRHX ;----LOAD MLA IN DOR1 REGISTER-----
4002 024634 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
4003 024640 052777 000010 155432 BIS #10,@CSRX ;SELECT CHANNEL 2
4004 024646 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
4005 024654 017737 155376 002402 MOV @ISR2,RSAVE ;GET ISR2 CONTENTS
4006 024662 023737 002444 002402 CMP CDAT5,RSAVE ;ATN,ATN,LADS,NDAC,LPAS,(ULPA) BIT SET
4007 024670 001413 BEQ 40$ ;BRANCH IF YES
4008 024672 013737 002444 002500 MOV CDAT5,GOOD ;SET UP DATA FOR ERROR MESSAGES
4009 024700 013737 002402 002502 MOV RSAVE,BAD
4010 024706 ERRSOFT 717,E502,ERR501 ;ERROR HANDLER
                                TRAP CSBSEG
                                .WORD 717
                                .WORD E502
                                .WORD ERR501
024706 104457
024710 001315
024712 005447
024714 003702
4011 024716 CKLOOP ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
                                TRAP C$CLP1
024716 104406
4012 024720 042777 000010 155352 40$: BIC #10,@CSRX ;SELECT CHANNEL 1
4013 024726 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
4014 024734 112777 000077 155334 MOVB #77,@IDRHX ;----LOAD UNL INTO DOR-----
4015 024742 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
4016 024746 052777 000010 155324 BIS #10,@CSRX ;SELECT CHANNEL 2
4017 024754 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
4018 024762 017737 155270 002402 MOV @ISR2,RSAVE ;GET ISR2 CONTENTS
4019 024770 023737 002434 002402 CMP CDAT1,RSAVE ;ATN,ATN,NDAC,(ULPA) BIT SHOULD BE SET
4020 024776 001412 BEQ 12$ ;BRANCH IF YES
4021 025000 013737 002434 002500 MOV CDAT1,GOOD ;SET UP DATA FOR ERROR MESSAGES
4022 025006 013737 002402 002502 MOV RSAVE,BAD
4023 025014 ERRSOFT 718,E502,ERR501 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 718
                                .WORD E502
                                .WORD ERR501
025014 104457
025016 001316
025020 005447
025022 003702
4024 025024 12$: ENDSEG
                                10003$: TRAP C$ESEG
025024 104405
4025 025026 BGNSEG
                                TRAP CSBSEG
025026 104404
4026 025030 042777 000010 155242 BIC #10,@CSRX ;SELECT CHANNEL 1
4027 025036 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
4028 025044 113777 002416 155224 MOVB MTA2,@IDRHX ;---LOAD MTA2 IN DOR1 REGISTER (IDR1)-
4029 025052 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
4030 025056 052777 000010 155214 BIS #10,@CSRX ;SELECT CHANNEL 2
4031 025064 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
4032 025072 017737 155160 002402 MOV @ISR2,RSAVE ;GET ISR2 CONTENTS
4033 025100 023737 002436 002402 CMP CDAT2,RSAVE ;ATN,ATN,NDAC,TPAS,(ULPA) SHOULD SET
4034 025106 001412 BEQ 33$ ;BRANCH IF YES
4035 025110 013737 002402 002502 MOV RSAVE,BAD ;SET UP DATA FOR ERROR MESSAGES
4036 025116 013737 002436 002500 MOV CDAT2,GOOD
4037 025124 ERRSOFT 719,E502,ERR501 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 719
                                .WORD E502
                                .WORD ERR501
025124 104457
025126 001317
025130 005447
025132 003702
4038 025134 33$: ENDSEG
                                10004$:
025134
    
```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-7  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

	025466	104457								TRAP	C\$ERSOFT
	025470	001322								.WORD	722
	025472	005447								.WORD	E502
	025474	003702								.WORD	ERR501
4082	025476				24\$:	ENDSEG					
	025476										
	025476	104405								10006\$:	
4083	025500	022737	000036	002314		CMP	#36,DPA2			TRAP	C\$ESEG
4084	025506	001002				BNE	25\$			:IS LAST DPA ADDRESS SELECTED	
4085	025510	000137	027172			JMP	TQVP?			:BRANCH IF NO	
4086	025514	052777	000010	154556	25\$:	BIS	#10,@CSRX			:SELECT CHANNEL 2	
4087	025522	013701	002314			MOV	DPA2,R1			:GET DPA2	
4088	025526	052701	000200			BIS	#200,R1			:-----SET EDPA BIT-----	
4089	025532	110177	154516			MOVB	R1,@IIRHX			:LOAD DPA2 & EDPA BIT INTO ADDR.	
4090	025536					BGNSEG					
	025536	104404								TRAP	C\$BSEG
4091	025540	042777	000010	154532		BIC	#10,@CSRX			:SELECT CHANNEL 1	
4092	025546	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER	
4093	025554	113777	002412	154514		MOVB	MLA2,@IDRHX			:---LOAD MLA2 INTO DOR1 REGISTER---	
4094	025562	004737	011060		COPB2:	JSR	PC,LOOP			:WAIT A LITTLE	
4095	025566	052777	000010	154504		BIS	#10,@CSRX			:SELECT CHANNEL 2	
4096	025574	012737	000002	002374		MOV	#2,CHAN			:LOAD CHANNEL NUMBER	
4097	025602	017737	154450	002402		MOV	@ISR2,RSAVE			:GET ISR2 CONTENTS	
4098	025610	023737	002444	002402		CMP	CDAT5,RSAVE			:ATN,ATN,LADS,NDAC,LPAS,(ULPA) BIT SET	
4099	025616	001413				BEQ	40\$			:BRANCH IF YES	
4100	025620	013737	002444	002500		MOV	CDAT5,GOOD			:SET UP DATA FOR ERROR MESSAGES	
4101	025626	013737	002402	002502		MOV	RSAVE,BAD				
4102	025634					ERRSOFT	723,E502,ERR501			:ERROR HANDLER	
	025634	104457								TRAP	C\$ERSOFT
	025636	001323								.WORD	723
	025640	005447								.WORD	E502
	025642	003702								.WORD	ERR501
4103	025644					CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET	
	025644	104406								TRAP	C\$CLP1
4104	025646	042777	000010	154424	40\$:	BIC	#10,@CSRX			:SELECT CHANNEL 1	
4105	025654	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER	
4106	025662	112777	000077	154406		MOVB	#77,@IDRHX			:---LOAD UNL INTO DOR-----	
4107	025670	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE	
4108	025674	052777	000010	154376		BIS	#10,@CSRX			:SELECT CHANNEL 2	
4109	025702	012737	000002	002374		MOV	#2,CHAN			:LOAD CHANNEL NUMBER	
4110	025710	017737	154342	002402		MOV	@ISR2,RSAVE			:GET ISR2 CONTENTS	
4111	025716	023737	002434	002402		CMP	CDAT1,RSAVE			:ATN,ATN,NDAC,(ULPA) BIT SHOULD BE SET	
4112	025724	001412				BEQ	12\$			:BRANCH IF YES	
4113	025726	013737	002434	002500		MOV	CDAT1,GOOD			:SET UP DATA FOR ERROR MESSAGES	
4114	025734	013737	002402	002502		MOV	RSAVE,BAD				
4115	025742					ERRSOFT	724,E502,ERR501			:ERROR HANDLER	
	025742	104457								TRAP	C\$ERSOFT
	025744	001324								.WORD	724
	025746	005447								.WORD	E502
	025750	003702								.WORD	ERR501
4116	025752				12\$:	ENDSEG					
	025752										
	025752	104405								10007\$:	
4117	025754	042777	000010	154316		BIC	#10,@CSRX			TRAP	C\$ESEG
4118	025762	012737	000001	002374		MOV	#1,CHAN			:SELECT CHANNEL 1	
4119	025770	113701	002412			MOVB	MLA2,R1			:LOAD CHANNEL NUMBER	
4120	025774	032737	000001	002314		BIT	#1,DPA2			:GET MY LISTENER ADDRESS	
										:IS DPA EVEN	



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-8  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

4121	026002	001021			BNE	3\$			:BRANCH IF YES
4122	026004	105201			INCB	R1			:CHANGE MLA2 TO ODD
4123	026006	052737	000001	002434	BIS	#1,CDAT1			:SET ULPA BIT IN COMPARE DATA FOR ISR
4124	026014	052737	000001	002436	BIS	#1,CDAT2			:...
4125	026022	052737	000001	002440	BIS	#1,CDAT3			:...
4126	026030	052737	000001	002442	BIS	#1,CDAT4			:...
4127	026036	052737	000001	002444	BIS	#1,CDAT5			:...
4128	026044	000420			BR	+42			:BRANCH TO BGNSEG
4129	026046	105301			DECB	R1	3\$:		:CHANGE MLA2 TO EVEN
4130	026050	042737	000001	002434	BIC	#1,CDAT1			:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4131	026056	042737	000001	002436	BIC	#1,CDAT2			:...
4132	026064	042737	000001	002440	BIC	#1,CDAT3			:...
4133	026072	042737	000001	002442	BIC	#1,CDAT4			:...
4134	026100	042737	000001	002444	BIC	#1,CDAT5			:...
4135	026106				BGNSEG				
	026106	104404							TRAP C\$BSEG
4136	026110	042777	000010	154162	BIC	#10,@CSRX			:SELECT CHANNEL 1
4137	026116	110177	154154		MOVB	R1,@IDRHX			:-----LOAD NEW MLA1 INTO DOR-----
4138	026122	004737	011060		JSR	PC,LOOP	COPC2:		:WAIT A LITTLE
4139	026126	052777	000010	154144	BIS	#10,@CSRX			:SELECT CHANNEL 2
4140	026134	012737	000002	002374	MOV	#2,CHAN			:LOAD CHANNEL NUMBER
4141	026142	017737	154110	002402	MOV	@ISRX,RSAVE			:GET ISR2 CONTENTS
4142	026150	023737	002444	002402	CMP	CDAT5,RSAVE			:ATN,ATN,LADS,NDAC,(ULPA),LPAS BIT SET
4143	026156	001413			BEQ	40\$			:BRANCH IF YES
4144	026160	013737	002444	002500	MOV	CDAT5,GOOD			:SET UP DATA FOR ERROR MESSAGES
4145	026166	013737	002402	002502	MOV	RSAVE,BAD			:...
4146	026174				ERRSOFT	725,E502,ERR501			:ERROR HANDLER
	026174	104457							TRAP C\$ERSOFT
	026176	001325							.WORD 725
	026200	005447							.WORD E502
	026202	003702							.WORD ERR501
4147	026204				CKLOOP				:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	026204	104406							TRAP C\$CLP1
4148	026206	042777	000010	154064	BIC	#10,@CSRX	40\$:		:SELECT CHANNEL 1
4149	026214	012737	000001	002374	MOV	#1,CHAN			:LOAD CHANNEL NUMBER
4150	026222	112777	000077	154046	MOVB	#77,@IDRHX			:-----LOAD UNL INTO DOR-----
4151	026230	004737	011060		JSR	PC,LOOP			:WAIT A LITTLE
4152	026234	052777	000010	154036	BIS	#10,@CSRX			:SELECT CHANNEL 2
4153	026242	012737	000002	002374	MOV	#2,CHAN			:LOAD CHANNEL NUMBER
4154	026250	017737	154002	002402	MOV	@ISRX,RSAVE			:GET ISR2 CONTENTS
4155	026256	023737	002434	002402	CMP	CDAT1,RSAVE			:ATN,ATN,NDAC,(ULPA) BIT SHOULD BE SET
4156	026264	001412			BEQ	12\$			:BRANCH IF YES
4157	026266	013737	002434	002500	MOV	CDAT1,GOOD			:SET UP DATA FOR ERROR MESSAGES
4158	026274	013737	002402	002502	MOV	RSAVE,BAD			:...
4159	026302				ERRSOFT	726,E502,ERR501			:ERROR HANDLER
	026302	104457							TRAP C\$ERSOFT
	026304	001326							.WORD 726
	026306	005447							.WORD E502
	026310	003702							.WORD ERR501
4160	026312				12\$:	ENDSEG			
	026312	104405							10010\$:
4161	026314	032737	000001	002314	BIT	#1,DPA2			:IS DPA EVEN
4162	026322	001420			BEQ	3\$			:BRANCH IF YES
4163	026324	052737	000001	002434	BIS	#1,CDAT1			:SET ULPA BIT IN COMPARE DATA FOR ISR
4164	026332	052737	000001	002436	BIS	#1,CDAT2			:...
4165	026340	052737	000001	002440	BIS	#1,CDAT3			:...



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-9  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

4166	026346	052737	000001	002442	BIS	#1,CDAT4	:...
4167	026354	052737	000001	002444	BIS	#1,CDAT5	:...
4168	026362	000417			BR	+40	:BRANCH TO BGNSEG
4169	026364	042737	000001	002434	BIC	#1,CDAT1	:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4170	026372	042737	000001	002436	BIC	#1,CDAT2	:...
4171	026400	042737	000001	002440	BIC	#1,CDAT3	:...
4172	026406	042737	000001	002442	BIC	#1,CDAT4	:...
4173	026414	042737	000001	002444	BIC	#1,CDAT5	:...
4174	026422				BGNSEG		
	026422	104404					TRAP C\$BSEG
4175	026424	042777	000010	153646	BIC	#10,@CSRX	:SELECT CHANNEL 1
4176	026432	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
4177	026440	113777	002416	153630	MOVB	MTA2,@IDRHX	:---LOAD MTA INTO DOR1 REGISTER---
4178	026446	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
4179	026452	052777	000010	153620	BIS	#10,@CSRX	:SELECT CHANNEL 2
4180	026460	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
4181	026466	017737	153564	002402	MOV	@ISRX,RSAVE	:GET ISR2 CONTENTS
4182	026474	023737	002440	002402	CMP	CDAT3,RSAVE	:ATN,ATN,TADS,NDAC,TPAS,(ULPA) BIT SET
4183	026502	001413			BEQ	16\$	:BRANCH IF YES
4184	026504	013737	002402	002502	MOV	RSAVE,BAD	:SET UP DATA FOR ERROR MESSAGES
4185	026512	013737	002440	002500	MOV	CDAT3,GOOD	:...
4186	026520				ERRSOFT	727,E502,ERR501	:ERROR HANDLER
	026520	104457					TRAP C\$ERSOFT
	026522	001327					.WORD 727
	026524	005447					.WORD E502
	026526	003702					.WORD ERR501
4187	026530				CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	026530	104406					TRAP C\$CLP1
4188	026532	042777	000010	153540	BIC	#10,@CSRX	:SELECT CHANNEL 1
4189	026540	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
4190	026546	112777	000137	153522	MOVB	#137,@IDRHX	:---LOAD UNIT INTO DOR---
4191	026554	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
4192	026560	052777	000010	153512	BIS	#10,@CSRX	:SELECT CHANNEL 2
4193	026566	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
4194	026574	017737	153456	002402	MOV	@ISRX,RSAVE	:GET ISR2 CONTENTS
4195	026602	023737	002434	002402	CMP	CDAT1,RSAVE	:ATN,ATN,NDAC,(ULPA) SHOULD BE SET
4196	026610	001412			BEQ	24\$	:BRANCH IF YES
4197	026612	013737	002434	002500	MOV	CDAT1,GOOD	:SET UP DATA FOR ERROR MESSAGES
4198	026620	013737	002402	002502	MOV	RSAVE,BAD	:...
4199	026626				ERRSOFT	728,E502,ERR501	:ERROR HANDLER
	026626	104457					TRAP C\$ERSOFT
	026630	001330					.WORD 728
	026632	005447					.WORD E502
	026634	003702					.WORD ERR501
4200	026636				24\$:	ENDSEG	
	026636	104405					10011\$:
4201	026640				BGNSEG		TRAP C\$ESEG
	026640	104404					TRAP C\$BSEG
4202	026642	042777	000010	153430	BIC	#10,@CSRX	:SELECT CHANNEL 1
4203	026650	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER
4204	026656	013701	002416		MOV	MTA2,R1	:GET TALKER ADDRESS
4205	026662	032737	000001	002314	BIT	#1,DPA2	:IS DPA ODD
4206	026670	001021			BNE	25\$	:BRANCH IF YES
4207	026672	105201			INCB	R1	:CHANGE MTA2 TO ODD
4208	026674	052737	000001	002434	BIS	#1,CDAT1	:SET ULPA BIT IN COMPARE DATA FOR ISR
4209	026702	052737	000001	002436	BIS	#1,CDAT2	:...

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 49-10  
 TEST 7: ADDRESS REGISTER TEST OF CHANNEL 2

4210	026710	052737	000001	002440		BIS	#1,CDAT3		:...
4211	026716	052737	000001	002442		BIS	#1,CDAT4		:...
4212	026724	052737	000001	002444		BIS	#1,CDAT5		:...
4213	026732	000420				BR	+42		:BRANCH TO LOAD NEW MTA
4214	026734	105301			25\$:	DECB	R1		:CHANGE MTA2 TO EVEN
4215	026736	042737	000001	002434		BIC	#1,CDAT1		:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4216	026744	042737	000001	002436		BIC	#1,CDAT2		:...
4217	026752	042737	000001	002440		BIC	#1,CDAT3		:...
4218	026760	042737	000001	002442		BIC	#1,CDAT4		:...
4219	026766	042737	000001	002444		BIC	#1,CDAT5		:...
4220	026774	110177	153276			MOVB	R1,@IDRHX		:---LOAD NEW MTA INTO DOR REGISTER---
4221	027000	004737	011060		COPD2:	JSR	PC,LOOP		:WAIT A LITTLE
4222	027004	052777	000010	153266		BIS	#10,@CSRX		:SELECT CHANNEL 2
4223	027012	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4224	027020	017737	153232	002402		MOV	@ISRX,RSAVE		:GET ISR2 CONTENTS
4225	027026	023737	002440	002402		MOV	@ISRX,RSAVE		:ATN,ATN,TADS,NDAC,TPAS,(ULPA) BIT SET
4226	027034	001413				BEQ	16\$		:BRANCH IF YES
4227	027036	013737	002402	002502		MOV	RSAVE,BAD		:SET UP DATA FOR ERROR MESSAGES
4228	027044	013737	002440	002500		MOV	CDAT3,GOOD		:...
4229	027052					ERRSOFT	729,E502,ERR501		:ERROR HANDLER
	027052	104457							TRAP C\$ERSOFT
	027054	001331							.WORD 729
	027056	005447							.WORD E502
	027060	003702							.WORD ERR501
4230	027062					CKLOOP			:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	027062	104406							TRAP C\$CLP1
4231	027064	042777	000010	153206	16\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
4232	027072	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4233	027100	112777	000137	153170		MOVB	#137,@IDRHX		:---LOAD UNT INTO DOR---
4234	027106	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
4235	027112	052777	000010	153160		BIS	#10,@CSRX		:SELECT CHANNEL 2
4236	027120	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4237	027126	017737	153124	002402		MOV	@ISRX,RSAVE		:GET ISR2 CONTENTS
4238	027134	023737	002434	002402		MOV	@ISRX,RSAVE		:ATN,ATN,NDAC,(ULPA) SHOULD BE SET
4239	027142	001412				BEQ	24\$		:BRANCH IF YES
4240	027144	013737	002434	002500		MOV	CDAT1,GOOD		:SET UP DATA FOR ERROR MESSAGES
4241	027152	013737	002402	002502		MOV	RSAVE,BAD		:...
4242	027160					ERRSOFT	730,E502,ERR501		:ERROR HANDLER
	027160	104457							TRAP C\$ERSOFT
	027162	001332							.WORD 730
	027164	005447							.WORD E502
	027166	003702							.WORD ERR501
4243	027170				24\$:	ENDSEG			
	027170	104405							10012\$:
4244	027172	005737	002234		TQVP7:	TST	QVP		:IS QUICK VERIFY PASS SELECTED
4245	027176	001054				BNE	EXQV7		:IF YES EXIT TEST
4246	027200	005737	002322			TST	ITRCNT		:IS THIS THE FIRST TIME THRU THE TEST
4247	027204	001007				BNE	1\$		:BRANCH IF NO
4248	027206	013737	002314	002406		MOV	DPA2,SDPA		:SAVE ENTERED DPA2
4249	027214	005037	002314			CLR	DPA2		:CLR DPA2
4250	027220	005237	002322			INC	ITRCNT		:SET FLAG TO SEE FIRST TIME PASS
4251	027224	005237	002314		1\$:	INC	DPA2		:GET NEW DPA2
4252	027230	023737	002314	002312		CMP	DPA2,DPA1		:IS DPA2 = DPA1
4253	027236	001002				BNE	2\$		:BRANCH IF NO
4254	027240	005237	002314			INC	DPA2		:INCREMENT DPA2
4255	027244	022737	000037	002314	2\$:	CMP	#37,DPA2		:ALL DONE







```

4273
4274
4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290 027416
      027416
4291 027416 005737 002324
4292 027422 001410
4293 027424
      027424 012746 031730
      027430 012746 000001
      027434 010600
      027436 104417
      027440 062706 000004
4294 027444 005037 002322
4295 027450 004737 010220
4296 027454 004737 010534
4297 027460
      027460 104404
4298 027462 112777 000212 152600
4299 027470 017737 152562 002502
4300 027476 022737 120042 002502
4301 027504 001407
4302 027506 012737 120042 002500
4303 027514
      027514 104457
      027516 001441
      027520 005447
      027522 003702
4304 027524
      027524 104405
4305 027526 013701 002314
4306 027532 062701 000040
4307 027536 010137 002412
4308 027542 032737 000001 002314
4309 027550 001412
4310 027552 052737 000001 002434
4311 027560 052737 000001 002450
4312 027566 052737 000001 002444
4313 027574 000411
4314 027576 042737 000001 002434
4315 027604 042737 000001 002450
4316 027612 042737 000001 002444
    
```

```

.SBTTL TEST 8: DATA TRANSFER TEST
:.....
      IEX - TEST 8
:THIS TEST IS DIVIDED INTO TWO PARTS
:IT CHECKS THE DATA OUT (DOR) AND DATA IN (DIR) REGISTERS
:PART 1 CHECKS THE DOR AND DIR REGISTER BY LOADING THE DOR1 WITH A
      DATA BYTE AND READING IT FROM THE DIR2
      (PROGRAMMED DATA TRANSFER FROM CHAN.1 TO CHAN.2).
:PART 2 CHECKS THE DOR AND DIR REGISTER BY LOADING THE DOR2 WITH A
      DATA BYTE AND READING IT FROM THE DIR1
      (PROGRAMMED DATA TRANSFER FROM CHAN.2 TO CHAN.1).
:IF THE QUICK VERIFY PASS IS NOT SELECTED, THE TEST ITERATION IS
:CARRIED OUT WITH A DIFFERENT DATA PATTERN
:.....
      BGNSTST
:.....
      T8::
      TST      PNTF      ;IS THE PNT FLAG SET
      BEQ      7$        ;IF YES, PRINT THE TEST HEADER
      PRINTF   #TSHDB    ;...
:.....
      MOV      #TSHDB,-(SP)
      MOV      #1,-(SP)
      MOV      SP,R0
      TRAP    C$PNTF
      ADD      #4,SP
7$:   CLR      ITRCNT    ;CLEAR COUNTER
      JSR      PC,CULPA  ;CLEAR ULPA BIT IN ISR 1 AND 2
      JSR      PC,BGIN1  ;SET UP PARAMETER
      BGNSEG
:.....
      TRAP    C$BSEG
4298:  MOVB    #212,@ICRHX ;----LOAD TON IN ACR 1 (ICR1)-----
      MOV     @ISRX,BAD   ;GET ISR1 CONTENTS
      CMP     #120042,BAD ;ATN,NDAC,ATN,TADS SHOULD BE SET
      BEQ     4$          ;BRANCH IF YES
      MOV     #120042,GOOD ;SET UP DATA FOR ERROR MESSAGE
      ERRSOF 801,E502,ERR501 ;ERROR HANDLER
:.....
      TRAP    C$ERSOFT
      .WORD   801
      .WORD   E502
      .WORD   ERR501
4304:  4$:     ENDSEG
:.....
      10000$:
      TRAP    C$ESEG
4305:  MOV     DPA2,R1     ;GET DPA2 ADDRESS
      ADD     #40,R1     ;CREATE MY LISTENER ADDRESS (MLA)
      MOV     R1,MLA2    ;STORE MLA
      BIT     #1,DPA2    ;IS DPA EVEN
      BEQ     3$        ;BRANCH IF YES
      BIS     #1,CDAT1   ;SET ULPA BIT IN COMPARE DATA FOR ISR
:.....
      BR     +24        ;BRANCH TO BGNSEG
4314:  3$:   BIC     #1,CDAT1  ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
      BIC     #1,CDAT7
      BIC     #1,CDAT5
      ....
    
```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 50-2  
 TEST 8: DATA TRANSFER TEST

```

4356 030124 012737 000020 002500      MOV      #20,GOOD      ;SET UP DATA FOR ERROR MESSAGE
4357 030132 013737 002402 002502      MOV      RSAVE,BAD    ;
4358 030140      ERRSOFT 805,E501,ERR501 ;:ERROR HANDLER
      030140 104457      TRAP      C$ERSOFT
      030142 001445      .WORD    805
      030144 005406      .WORD    E501
      030146 003702      .WORD    ERR501
4359 030150      CKLOOP      ;BRANCH TO BGNSEG IF ERRLOOP WAS SET
      030150 104406      TRAP      C$CLP1
4360 030152 052777 000010 152120 42$:  BIS      #10,@CSRX    ;SELECT CHANNEL 2
4361 030160 012737 000002 002374      MOV      #2,CHAN      ;LOAD CHANNEL NUMBER
4362 030166 017737 152056 002402      MOV      @IIRX,RSAVE  ;GET IIR2 CONTENTS
4363 030174 022737 000040 002402      CMP      #40,RSAVE    ;BI BIT IN IIR2 SHOULD BE SET
4364 030202 001413      BEQ      43$         ;BRANCH IF YES
4365 030204 012737 000040 002500      MOV      #40,GOOD     ;SET UP DATA FOR ERROR MESSAGE
4366 030212 013737 002402 002502      MOV      RSAVE,BAD    ;
4367 030220      ERRSOFT 806,E501,ERR501 ;:ERROR HANDLER
      030220 104457      TRAP      C$ERSOFT
      030222 001446      .WORD    806
      030224 005406      .WORD    E501
      030226 003702      .WORD    ERR501
4368 030230      CKLOOP      ;
      030230 104406      TRAP      C$CLP1
4369 030232 017737 152034 002402 43$:  MOV      @IDRX,RSAVE  ;READ DATA FROM DIR REGISTER
4370 030240 105037 002402      CLRB    RSAVE         ;CLEAR LOW BYTE OF IDR
4371 030244 000337 002402      SWAB   RSAVE         ;SWAB DATA FOR COMPARE
4372 030250 120537 002402      CMPB   R5,RSAVE      ;CORRECT DATA RECEIVED
4373 030254 001414      BEQ      44$         ;BRANCH IF YES
4374 030256 000305      SWAB   R5            ;SWAB HIGH AND LOW BYTE
4375 030260 105005      CLRB   R5            ;CLEAR HIGH BYTE OF R5
4376 030262 000305      SWAB   R5            ;SWAB HIGH AND LOW BYTE
4377 030264 010537 002500      MOV      R5,GOOD     ;SET UP DATA FOR ERROR MESSAGE
4378 030270 013737 002402 002502      MOV      RSAVE,BAD    ;
4379 030276      ERRSOFT 807,E801,ERR501 ;:ERROR HANDLER
      030276 104457      TRAP      C$ERSOFT
      030300 001447      .WORD    807
      030302 005510      .WORD    E801
      030304 003702      .WORD    ERR501
4380 030306      ENDSEG      ;
      030306 104405      TRAP      C$ESEG
      030306 005737 002234      TST     QVP          ;IS QUICK VERIFY PASS SELECTED
4381 030310 001010      BNE     50$         ;BRANCH IF YES
4382 030314 062705 000021      ADD     #21,R5       ;CREATE NEW TRANSMIT DATA
4383 030316 005237 002322      INC     ITRCNT       ;INCREMENT ITERATION COUNTERB
4384 030322 023737 002320 002322      CMP     ITRDEF,ITRCNT ;ALL DONE
4385 030326 001251      BNE     A81         ;BRANCH IF NO
4386 030334 001251      BNE     A81
4387 030336      BGNSEG      ;
      030336 104404      TRAP      C$BSEG
4388 030340 042777 000010 151732      BIC     #10,@CSRX    ;SELECT CHANNEL 1
4389 030346 012737 000001 002374      MOV     #1,CHAN      ;LOAD CHANNEL NUMBER
4390 030354 112777 000014 151706      MOVB   #14,@ICRHX   ;-----LOAD ICA BIT INTO ACR1-----
4391 030362 004737 011060      JSR    PC,LOOP      ;WAIT A LITTLE FOR THE BO BIT
4392 030366 017737 151656 002502      MOV     @IIRX,BAD    ;GET IIR1 CONTENTS
4393 030374 022737 000020 002502      CMP     #20,BAD     ;BO IN IIR1 SHOULD BE SET
4394 030402 001410      BEQ     51$         ;BRANCH IF YES
4395 030404 012737 000020 002500      MOV     #20,GOOD     ;SET UP DATA FOR ERROR MESSAGE

```



```

4396 030412          ERRSOFT 808,E501,ERR501      ;ERROR HANDLER
      030412 104457          TRAP C$ERSOFT
      030414 001450          .WORD 808
      030416 005406          .WORD E501
      030420 003702          .WORD ERR501
4397 030422          CKLOOP                    ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      030422 104406          TRAP C$CLP1
4398 030424 112777 000012 151636 51$: MOVB #12,@ICRHX      ;----LOAD NOT TON INTO ACR1 ----
4399 030432 112777 000077 151636      MOVB #77,@IDRHX      ;----LOAD UNL INTO DOR1----
4400 030440 004737 011060      JSR PC,LOOP          ;WAIT A LITTLE
4401 030444 017737 151606 002502      MOV @ISRX,BAD        ;GET ISR1 CONTENTS
4402 030452 022737 120040 002502      CMP #120040,BAD      ;ATN,NDAC,ATN, SHOULD BET SET
4403 030460 001410      BEQ 53$              ;BRANCH IF YES
4404 030462 012737 120040 002500      MOV #120040,GOOD     ;SET UP DATA FOR ERROR MESSAGE
4405 030470          ERRSOFT 809,E502,ERR501      ;....
      030470 104457          TRAP C$ERSOFT
      030472 001451          .WORD 809
      030474 005447          .WORD E502
      030476 003702          .WORD ERR501
4406 030500          CKLOOP                    ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      030500 104406          TRAP C$CLP1
4407 030502 052777 000010 151570 53$: BIS #10,@CSRX        ;SELECT CHANNEL 2
4408 030510 012737 000002 002374      MOV #2,CHAN          ;LOAD CHANNEL NUMBER
4409 030516 017737 151534 002502      MOV @ISRX,BAD        ;GET ISR2 CONTENTS
4410 030524 023737 002434 002502      CMP CDAT1,BAD        ;ATN,NDAC,ATN,(ULPA) SHOULD BE SET
4411 030532 001410      BEQ 55$              ;BRANCH IF YES
4412 030534 013737 002434 002500      MOV CDAT1,GOOD       ;SET UP DATA FOR ERROR MESSAGE
4413 030542          ERRSOFT 810,E502,ERR501      ;ERROR HANDLER
      030542 104457          TRAP C$ERSOFT
      030544 001452          .WORD 810
      030546 005447          .WORD E502
      030550 003702          .WORD ERR501
4414 030552          CKLOOP                    ;BRANCH TO BGNSEG IF ERROR LOOP IS SET
      030552 104406          TRAP C$CLP1
4415 030554          55$: ENDSEG
      030554          10003$: TRAP C$ESEG
      030554 104405
4416          ;+-----+
4417          ;PART 2 OF DATA TRANSFER TEST
4418          ;THIS PART CHECKS THE DATA TRANSFER FROM CHANNEL 2 TO CHANNEL 1
4419          ;+-----+
4420
4421 030556 004737 010220      JSR PC,CULPA          ;CLEAR ULPA BIT IN ISR 1 AND 2
4422 030562 004737 010710      JSR PC,BGIN2          ;SET UP PARAMETER
4423 030566 005037 002322      CLR ITRCNT           ;CLEAR COUNTER
4424 030572 112777 000010 151500      MOVB #10,@CSRX       ;SELECT CHANNEL 2
4425 030600 012737 000002 002374      MOV #2,CHAN          ;LOAD CHANNEL NUMBER
4426 030606 052777 000002 151464      BIS #2,@CSRX         ;SELECT CHANNEL 2 AS SYSTEM CONTROLLER
4427 030614 112777 000217 151446      MOVB #217,@ICRHX     ;----LOAD SIC IN ACR 2----
4428 030622 004737 011072      JSR PC,WAIT          ;WAIT 100 US
4429 030626 112777 000017 151434      MOVB #17,@ICRHX      ;----LOAD NOT SIC IN ACR 2----
4430 030634          BGNSEG
      030634 104404          TRAP C$BSEG
4431 030636 112777 000212 151424      MOVB #212,@ICRHX     ;----LOAD TON IN ACR 2 (ICR2)----
4432 030644 017737 151406 002402      MOV @ISRX,RSAVE      ;GET ISR2 CONTENTS
4433 030652 022737 120042 002402      CMP #120042,RSAVE    ;ATN,NDAC,ATN,TADS, SHOULD BE SET
4434 030660 001412      BEQ 4$              ;BRANCH IF YES
    
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 50-4  
TEST 8: DATA TRANSFER TEST

4435	030662	012737	120042	002500		MOV	#120042,GOOD	:SET UP DATA FOR ERROR MESSAGE
4436	030670	013737	002402	002502		MOV	RSAVE,BAD	:...
4437	030676					ERRSOFT	811,E502,ERR501	:ERROR HANDLER
	030676	104457						TRAP
	030700	001453						.WORD
	030702	005447						.WORD
	030704	003702						.WORD
4438	030706				4\$:	ENDSEG		
	030706							10004\$:
	030706	104405						TRAP
4439	030710	013701	002312			MOV	DPA1,R1	:GET DPA1 ADDRESS
4440	030714	062701	000040			ADD	#40,R1	:CREATE MY LISTENER ADDRESS (MLA)
4441	030720	010137	002410			MOV	R1,MLA1	:STORE MLA
4442	030724	032737	000001	002312		BIT	#1,DPA1	:IS DPA EVEN
4443	030732	001412				BEQ	3\$	:BRANCH IF YES
4444	030734	052737	000001	002434		BIS	#1,CDAT1	:SET ULPA BIT IN COMPARE DATA FOR ISR
4445	030742	052737	000001	002450		BIS	#1,CDAT7	:...
4446	030750	052737	000001	002444		BIS	#1,CDAT5	:...
4447	030756	000411				BR	+24	:BRANCH TO BGNSEG
4448	030760	042737	000001	002434	3\$:	BIC	#1,CDAT1	:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4449	030766	042737	000001	002450		BIC	#1,CDAT7	:...
4450	030774	042737	000001	002444		BIC	#1,CDAT5	:...
4451	031002					BGNSEG		
	031002	104404						TRAP
4452	031004	052777	000010	151266		BIS	#10,@CSRX	:SELECT CHANNEL 2
4453	031012	012737	000002	002374		MOV	#2,CHAN	:LOAD CHANNEL NUMBER
4454	031020	113777	002410	151250		MOVB	MLA1,@IDRHX	:-----LOAD LISTENER ADDRESS OF CHANNEL 2
4455	031026	004737	011060			JSR	PC,LOOP	:WAIT A LITTLE
4456	031032	042777	000010	151240		BIC	#10,@CSRX	:SELECT CHANNEL 1
4457	031040	012737	000001	002374		MOV	#1,CHAN	:LOAD CHANNEL NUMBER
4458	031046	017737	151176	002502		MOV	@IIRX,BAD	:GET IIR1 CONTENTS
4459	031054	022737	002401	002502		CMP	#2401,BAD	:MA,MAC,IFC BIT IN IIR1 SHOULD BE SET
4460	031062	001410				BEQ	20\$	:BRANCH IF YES
4461	031064	012737	002401	002500		MOV	#2401,GOOD	:SET UP DATA FOR ERROR MESSAGES
4462	031072					ERRSOFT	812,E501,ERR501	:ERROR HANDLER
	031072	104457						TRAP
	031074	001454						.WORD
	031076	005406						.WORD
	031100	003702						.WORD
4463	031102					CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	031102	104406						TRAP
4464	031104	017737	151146	002502	20\$:	MOV	@ISRX,BAD	:GET ISR1 CONTENTS
4465	031112	023737	002444	002502		CMP	CDAT5,BAD	:ATN,ATN,LADS,NDAC,LPAS,(ULPA) BIT SET
4466	031120	001410				BEQ	30\$	:BRANCH IF YES
4467	031122	013737	002444	002500		MOV	CDAT5,GOOD	:SET UP DATA FOR ERROR MESSAGES
4468	031130					ERRSOFT	813,E502,ERR501	:ERROR HANDLER
	031130	104457						TRAP
	031132	001455						.WORD
	031134	005447						.WORD
	031136	003702						.WORD
4469	031140					CKLOOP		:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	031140	104406						TRAP
4470	031142	052777	000010	151130	30\$:	BIS	#10,@CSRX	:SELECT CHANNEL 2
4471	031150	012737	000002	002374		MOV	#2,CHAN	:LOAD CHANNEL NUMBER
4472	031156	112777	000013	151104		MOVB	#13,@ICRHX	:-----LOAD GTS INTO ACR2 (ICR2)-----
4473	031164	042777	000010	151106	40\$:	BIC	#10,@CSRX	:SELECT CHANNEL 1
4474	031172	012737	000001	002374		MOV	#1,CHAN	:LOAD CHANNEL NUMBER



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 50-5  
 TEST B: DATA TRANSFER TEST

4475	031200	017737	151052	002502		MOV	@ISRX,BAD		:GET ISR1 CONTENTS
4476	031206	023737	002450	002502		CMP	CDAT7,BAD		:NDAC,LADS,LPAS,(ULPA) SHOULD BE SET
4477	031214	001407				BEQ	41\$		:BRANCH IF YES
4478	031216	013737	002450	002500		MOV	CDAT7,GOOD		:SET UP DATA FOR ERROR MESSAGE
4479	031224					ERRSOFT	814,E502,ERR501		:
	031224	104457							TRAP C\$ERSOFT
	031226	001456							.WORD 814
	031230	005447							.WORD E502
	031232	003702							.WORD ERR501
4480	031234				41\$:	ENDSEG			
	031234								10005\$:
	031234	104405							TRAP C\$ESEG
4481	031236	112705	000252			MOVB	#252,R5		:GET DATA PATTERN
4482	031242				A82:	BGNSEG			TRAP C\$BSEG
	031242	104404							
4483	031244	052777	000010	151026		BIS	#10,@CSRX		:SELECT CHANNEL 2
4484	031252	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4485	031260	110577	151012			MOVB	R5,@IDRH		:----LOAD DATA PATTERN INTO DOR2-----
4486	031264	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
4487	031270	017737	150754	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
4488	031276	022737	000020	002502		CMP	#20,BAD		:BO BIT IN IIR2 SHOULD BE SET
4489	031304	001410				BEQ	42\$		:BRANCH IF YES
4490	031306	012737	000020	002500		MOV	#20,GOOD		:SET UP DATA FOR ERROR MESSAGE
4491	031314					ERRSOFT	815,E501,ERR501		:ERROR HANDLER
	031314	104457							TRAP C\$ERSOFT
	031316	001457							.WORD 815
	031320	005406							.WORD E501
	031322	003702							.WORD ERR501
4492	031324					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP WAS SET
	031324	104406							TRAP C\$CLP1
4493	031326	042777	000010	150744	42\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
4494	031334	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4495	031342	017737	150702	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
4496	031350	022737	000040	002502		CMP	#40,BAD		:BI BIT IN IIR1 SHOULD BE SET
4497	031356	001410				BEQ	43\$		:BRANCH IF YES
4498	031360	012737	000040	002500		MOV	#40,GOOD		:SET UP DATA FOR ERROR MESSAGE
4499	031366					ERRSOFT	816,E501,ERR501		:ERROR HANDLER
	031366	104457							TRAP C\$ERSOFT
	031370	001460							.WORD 816
	031372	005406							.WORD E501
	031374	003702							.WORD ERR501
4500	031376					CKLOOP			:
	031376	104406							TRAP C\$CLP1
4501	031400	017737	150666	002402	43\$:	MOV	@IDRX,RSAVE		:READ DATA FROM DIR1 REGISTER
4502	031406	105037	002402			CLRB	RSAVE		:CLEAR LOW BYTE OF IDR
4503	031412	000337	002402			SWAB	RSAVE		:SWAB DATA FOR COMPARE
4504	031416	120537	002402			CMPB	R5,RSAVE		:CORRECT DATA RECEIVED
4505	031422	001414				BEQ	44\$		:BRANCH IF YES
4506	031424	000305				SWAB	R5		:SWAB HIGH AND LOW BYTE
4507	031426	105005				CLRB	R5		:CLEAR HIGH BYTE OF R5
4508	031430	000305				SWAB	R5		:SWAB HIGH AND LOW BYTE
4509	031432	010537	002500			MOV	R5,GOOD		:SET UP DATA FOR ERROR MESSAGE
4510	031436	013737	002402	002502		MOV	RSAVE,BAD		:
4511	031444					ERRSOFT	817,E802,ERR501		:ERROR HANDLER
	031444	104457							TRAP C\$ERSOFT
	031446	001461							.WORD 817
	031450	005566							.WORD E802





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 50-7  
TEST 8: DATA TRANSFER TEST

4548 031724  
031724 104432  
031726 000034

EXIT TST

TRAP C\$EXIT  
.WORD L10035-

4549

4550

4551 031730 045

123

062

TSMDB:

.NLIST

BEX

.ASCIZ

/%S2%ADATA TRANSFER TEST%N/

.LIST

BEX

4552

4553

4554 031762

031762

031762 104401

.EVEN

ENDTST

L10035: TRAP C\$ETST

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 51  
 TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)

```

4556 .SBTTL TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)
4557 :*****
4558 : IEX - TEST 9
4559 : THIS TEST CHECKS THE EXTENDED LISTENER INTERFACE FUNCTION .
4560 :
4561 : PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING
4562 : FEATURE OF CHANNEL 1 BY MEANS OF RECEIVING A VALID AS WELL AS AN INVALID
4563 : MY SECONDARY ADDRESS (MSA1).
4564 :*****
4565 031764 BGNTST
      031764
4566 031764 005737 002324 TST PNTF ; IS THE PNT FLAG SET
4567 031770 001410 BEQ 7$ ; IF YES, PRINT THE TEST HEADER
4568 031772 PRINTF #TSHD9 ; ...
      031772 012746 033454 MOV #TSHD9,-(SP)
      031776 012746 000001 MOV #1,-(SP)
      032002 010600 MOV SP,R0
      032004 104417 TRAP C$PNTF
      032006 062706 000004 ADD #4,SP
4569 032012 005037 002322 7$: CLR ITRCNT ; CLEAR COUNTER
4570 032016 004737 010220 JSR PC,CULPA ; CLEAR ULPA BIT IN ISR 1 AND 2
4571 032022 004737 010710 JSR PC,BGIN2 ; SET UP PARAMETER
4572 032026 042777 000010 150244 A9: BIC #10,@CSRX ; SELECT CHANNEL 1
4573 032034 112777 000223 150226 MCVB #223,@ICRHX ; -----LOAD DAI INTO ACR1-----
4574 032042 112777 000020 150212 MOVB #20,@ISRHX ; -----LOAD APT BIT INTO ISR1-----
4575 032050 013701 002312 MOV DPA1,R1 ; GET DPA1 ADDRESS
4576 032054 062701 000040 ADD #40,R1 ; CREATE MY LISTENER ADDRESS (MLA)
4577 032060 010137 002410 MOV R1,MLA1 ; STORE MLA
4578 032064 032737 000001 002312 BIT #1,DPA1 ; IS DPA EVEN
4579 032072 001412 BEQ 3$ ; BRANCH IF YES
4580 032074 052737 000001 002446 BIS #1,CDAT6 ; SET ULPA BIT IN COMPARE DATA FOR ISR
4581 032102 052737 000001 002442 BIS #1,CDAT4 ; ...
4582 032110 052737 000001 002444 BIS #1,CDAT5 ; ...
4583 032116 000411 BR +24 ; BRANCH TO BGNSEG
4584 032120 042737 000001 002446 3$: BIC #1,CDAT6 ; CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4585 032126 042737 000001 002442 BIC #1,CDAT4 ; ...
4586 032134 042737 000001 002444 BIC #1,CDAT5 ; ...
4587 032142 BGNSEG
      032142 104404 TRAP C$BSEG
4588 032144 052777 000010 150126 BIS #10,@CSRX ; SELECT CHANNEL 2
4589 032152 012737 000002 002374 MOV #2,CHAN ; LOAD CHANNEL NUMBER
4590 032160 113777 002410 150110 MOVB MLA1,@IDRHX ; ----LOAD LISTENER ADDRESS OF CHAN. 1
4591 032166 004737 011060 JSR PC,LOOP ; WAIT A LITTLE
4592 032172 042777 000010 150100 BIC #10,@CSRX ; SELECT CHANNEL 1
4593 032200 012737 000001 002374 MOV #1,CHAN ; LOAD CHANNEL NUMBER
4594 032206 017737 150036 002502 MOV @IIRX,BAD ; GET IIR1 CONTENTS
4595 032214 022737 000400 002502 CMP #400,BAD ; IFC,BIT IN IIR1 SHOULD BE SET
4596 032222 001410 BEQ 30$ ; BRANCH IF YES
4597 032224 012737 000400 002500 MOV #400,GOOD ; SET UP DATA FOR ERROR MESSAGES
4598 032232 ERRSOFT 901,E501,ERR501 ; ERROR HANDLER
      032232 104457 TRAP C$ERRSOFT
      032234 001605 .WORD 901
      032236 005406 .WORD E501
      032240 003702 .WORD ERR501
4599 032242 CKLOOP ; BRANCH TO BGNSEG WHEN ERRLOOP IS SET
      032242 104406 TRAP C$CLP1
4600 032244 017737 150006 002502 30$: MOV @ISRX,BAD ; GET ISR1 CONTENTS
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 51-1  
 TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)

4601	032252	023737	002442	002502		CMP	CDAT4,BAD		:ATN,ATN,NDAC LPAS,(ULPA) BIT SET
4602	032260	001410				BEQ	40\$		:BRANCH IF YES
4603	032262	013737	002442	002500		MOV	CDAT4,GOOD		:SET UP DATA FOR ERROR MESSAGES
4604	032270					ERRSOFT	902,E502,ERR501		:ERROR HANDLER
	032270	104457							TRAP C\$ERSOFT
	032272	001606							.WORD 902
	032274	005447							.WORD E502
	032276	003702							.WORD ERR501
4605	032300					CKLOOP			:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	032300	104406							TRAP C\$CLP1
4606	032302	052777	000010	147770	40\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
4607	032310	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4608	032316	013701	002410			MOV	MLA1,R1		:GET MLA1
4609	032322	062701	000100			ADD	#100,R1		:CREATE MSA1
4610	032326	010137	002420			MOV	R1,MSA1		:STORE MSA1 TO LOCATION MSA1
4611	032332	113777	002420	147736		MOVB	MSA1,@IDRHX		:-----LOAD MSA1 INTO DOR2-----
4612	032340	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
4613	032344	017737	147700	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
4614	032352	022737	000000	002502		CMP	#0,BAD		:IIR2 SHOULD BE ZERO
4615	032360	001407				BEQ	41\$		:BRANCH IF YES
4616	032362	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGE
4617	032366					ERRSOFT	903,E501,ERR501		:ERROR HANDLER
	032366	104457							TRAP C\$ERSOFT
	032370	001607							.WORD 903
	032372	005406							.WORD E501
	032374	003702							.WORD ERR501
4618	032376					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032376	104406							TRAP C\$CLP1
4619	032400	042777	000010	147672	41\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
4620	032406	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4621	032414	017737	147630	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
4622	032422	022737	010100	002502		CMP	#10100,BAD		:APT,INT1 BIT IN IIR1 SHOULD BE SET
4623	032430	001410				BEQ	42\$		:BRANCH IF YES
4624	032432	012737	010100	002500		MOV	#10100,GOOD		:SET UP DATA FOR ERROR MESSAGE
4625	032440					ERRSOFT	904,E501,ERR501		:ERROR HANDLER
	032440	104457							TRAP C\$ERSOFT
	032442	001610							.WORD 904
	032444	005406							.WORD E501
	032446	003702							.WORD ERR501
4626	032450					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032450	104406							TRAP C\$CLP1
4627	032452	017737	147600	002502	42\$:	MOV	@ISRX,BAD		:GET ISR1 CONTENTS
4628	032460	023737	002446	002502		CMP	CDAT6,BAD		:ATN,DAV,NDAC,NRFD,ATN,LPAS(ULPA) IS SET
4629	032466	001410				BEQ	43\$		:BRANCH IF YES
4630	032470	013737	002446	002500		MOV	CDAT6,GOOD		:SET UP DATA FOR ERROR MESSAGE
4631	032476					ERRSOFT	905,E502,ERR501		:ERROR HANDLER
	032476	104457							TRAP C\$ERSOFT
	032500	001611							.WORD 905
	032502	005447							.WORD E502
	032504	003702							.WORD ERR501
4632	032506					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032506	104406							TRAP C\$CLP1
4633	032510	017737	147550	002502	43\$:	MOV	@ICRX,BAD		:GET ICR1 CONTENTS
4634	032516	123737	002420	002502		CMPB	MSA1,BAD		:MSA1 ADDRESS SHOULD BE SET
4635	032524	001410				BEQ	44\$		:BRANCH IF YES
4636	032526	013737	002420	002500		MOV	MSA1,GOOD		:SET UP DATA FOR ERROR MESSAGE
4637	032534					ERRSOFT	906,E901,ERR501		:ERROR HANDLER

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 51-2  
 TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)

	032534	104457									TRAP	C\$ERSOFT
	032536	001612									.WORD	906
	032540	005644									.WORD	E901
	032542	003702									.WORD	ERR501
4638	032544				CKLOOP							;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032544	104406									TRAP	C\$CLP1
4639	032546	052777	000010	147524	44\$:	BIS	#10,@CSRX					;SELECT CHANNEL 2
4640	032554	012737	000002	002374		MOV	#2,CHAN					;LOAD CHANNEL NUMBER
4641	032562	017737	147470	002502		MOV	@ISR2,BAD					;GET ISR2 CONTENTS
4642	032570	022737	170040	002502		CMP	#170040,BAD					;ATN,DAV,NDAC,ATN,NRFD SHOULD BE SET
4643	032576	001410				BEQ	45\$					;BRANCH IF YES
4644	032600	012737	170040	002500		MOV	#170040,GOOD					;SET UP DATA FOR ERROR MESSAGE
4645	032606					ERRSOFT	907,E502,ERR501					;ERROR HANDLER
	032606	104457									TRAP	C\$ERSOFT
	032610	001613									.WORD	907
	032612	005447									.WORD	E502
	032614	003702									.WORD	ERR501
4646	032616				CKLOOP							;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032616	104406									TRAP	C\$CLP1
4647	032620	042777	000010	147452	45\$:	BIC	#10,@CSRX					;SELECT CHANNEL 1
4648	032626	012737	000001	002374		MOV	#1,CHAN					;LOAD CHANNEL NUMBER
4649	032634	112777	000201	147426		MOVB	#201,@ICRHX					;-----LOAD DACR INTO ACR1-----
4650	032642	017737	147410	002402		MOV	@ISR1,RSAVE					;GET ISR1 CONTENTS
4651	032650	023737	002444	002402		CMP	CDAT5,RSAVE					;ATN,NDAC,ATN,LADS,LPAS (ULPA) IS SET
4652	032656	001413				BEQ	46\$					;BRANCH IF YES
4653	032660	013737	002444	002500		MOV	CDAT5,GOOD					;SET UP DATA FOR ERROR MESSAGE
4654	032666	013737	002402	002502		MOV	RSAVE,BAD					;ERROR HANDLER
4655	032674					ERRSOFT	908,E502,ERR501					
	032674	104457									TRAP	C\$ERSOFT
	032676	001614									.WORD	908
	032700	005447									.WORD	E502
	032702	003702									.WORD	ERR501
4656	032704				CKLOOP							;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032704	104406									TRAP	C\$CLP1
4657	032706	052777	000010	147364	46\$:	BIS	#10,@CSRX					;SELECT CHANNEL 2
4658	032714	012737	000002	002374		MOV	#2,CHAN					;LOAD CHANNEL NUMBER
4659	032722	017737	147322	002402		MOV	@IIR2,RSAVE					;GET IIR2 CONTENTS
4660	032730	022737	000020	002402		CMP	#20,RSAVE					;BO BIT SHOULD BE SET
4661	032736	001413				BEQ	50\$					;BRANCH IF YES
4662	032740	012737	000020	002500		MOV	#20,GOOD					;SET UP DATA FOR ERROR MESSAGE
4663	032746	013737	002402	002502		MOV	RSAVE,BAD					;ERROR HANDLER
4664	032754					ERRSOFT	909,E501,ERR501					
	032754	104457									TRAP	C\$ERSOFT
	032756	001615									.WORD	909
	032760	005406									.WORD	E501
	032762	003702									.WORD	ERR501
4665	032764				CKLOOP							;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	032764	104406									TRAP	C\$CLP1
4666	032766	112777	000077	147302	50\$:	MOVB	#77,@IDRHX					;---LOAD UNL INTO DOR2-----
4667	032774					ENDSEG						
	032774											10000\$:
	032774	104405									TRAP	C\$ESEG
4668	032776				COPA9:	BGNSEG					TRAP	C\$BSEG
	032776	104404										
4669	033000	052777	000010	147272		BIS	#10,@CSRX					;SELECT CHANNEL 2
4670	033006	012737	000002	002374		MOV	#2,CHAN					;LOAD CHANNEL NUMBER
4671	033014	113777	002410	147254		MOVB	MLA1,@IDRHX					;----LOAD LISTENER ADDRESS OF CHANNEL 2--



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 51-3  
 TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)

4672	033022	004737	011060		JSR	PC, LOOP				:WAIT A LITTLE
4673	033026	042777	000010	147244	BIC	#10, @CSRX				:SELECT CHANNEL 1
4674	033034	012737	000001	002374	MOV	#1, CHAN				:LOAD CHANNEL NUMBER
4675	033042	017737	147202	002502	MOV	@IIRX, BAD				:GET IIR1 CONTENTS
4676	033050	022737	000001	002502	CMP	#1, BAD				:MAC BIT IN IIR1 SHOULD BE SET
4677	033056	001410			BEQ	30\$				:BRANCH IF YES
4678	033060	012737	000001	002500	MOV	#1, GOOD				:SET UP DATA FOR ERROR MESSAGES
4679	033066				ERRSOFT	910, E501, ERR501				:ERROR HANDLER
	033066	104457								TRAP C\$ERSOFT
	033070	001616								.WORD 910
	033072	005406								.WORD E501
	033074	003702								.WORD ERR501
4680	033076				CKLOOP					:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	033076	104406								TRAP C\$CLP1
4681	033100	052777	000010	147172	30\$: BIS	#10, @CSRX				:SELECT CHANNEL 2
4682	033106	012737	000002	002374	MOV	#2, CHAN				:LOAD CHANNEL NUMBER
4683	033114	013701	002410		MOV	MLA1, R1				:GET MLA1
4684	033120	062701	000100		ADD	#100, R1				:CREATE MSA1
4685	033124	010137	002420		MOV	R1, MSA1				:STORE MSA1 TO LOCATION MSA1
4686	033130	113777	002420	147140	MOVB	MSA1, @IDRHX				:-----LOAD MSA1 INTO DOR2-----
4687	033136	004737	011060		JSR	PC, LOOP				:WAIT A LITTLE
4688	033142	042777	000010	147130	BIC	#10, @CSRX				:SELECT CHANNEL 1
4689	033150	012737	000001	002374	MOV	#1, CHAN				:LOAD CHANNEL NUMBER
4690	033156	112777	000001	147104	MOVB	#1, @ICRHX				:-----LOAD NOT DACR INTO ACR1-----
4691	033164	017737	147060	002502	MOV	@IIRX, BAD				:READ IIR1 FOR CLEAR THE BITS
4692	033172	017737	147060	002502	MOV	@ISRX, BAD				:GET ISR1 CONTENTS
4693	033200	023737	002442	002502	CMP	CDAT4, BAD				:ATN, NDAC, ATN, LPAS (ULPA) IS SET
4694	033206	001410			BEQ	46\$				:BRANCH IF YES
4695	033210	013737	002442	002500	MOV	CDAT4, GOOD				:SET UP DATA FOR ERROR MESSAGE
4696	033216				ERRSOFT	911, E502, ERR501				:ERROR HANDLER
	033216	104457								TRAP C\$ERSOFT
	033220	001617								.WORD 911
	033222	005447								.WORD E502
	033224	003702								.WORD ERR501
4697	033226				CKLOOP					:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	033226	104406								TRAP C\$CLP1
4698	033230	052777	000010	147042	46\$: BIS	#10, @CSRX				:SELECT CHANNEL 2
4699	033236	012737	000002	002374	MOV	#2, CHAN				:LOAD CHANNEL NUMBER
4700	033244	017737	147000	002502	MOV	@IIRX, BAD				:GET IIR2 CONTENTS
4701	033252	022737	000020	002502	CMP	#20, BAD				:BO BIT SHOULD BE SET
4702	033260	001410			BEQ	50\$				:BRANCH IF YES
4703	033262	012737	000020	002500	MOV	#20, GOOD				:SET UP DATA FOR ERROR MESSAGE
4704	033270				ERRSOFT	912, E501, ERR501				:ERROR HANDLER
	033270	104457								TRAP C\$ERSOFT
	033272	001620								.WORD 912
	033274	005406								.WORD E501
	033276	003702								.WORD ERR501
4705	033300				CKLOOP					:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	033300	104406								TRAP C\$CLP1
4706	033302	112777	000077	146766	50\$: MOVB	#77, @IDRHX				:---LOAD UNL INTO DOR2-----
4707	033310				ENDSEG					
	033310									10001\$:
	033310	104405								TRAP C\$ESEG
4708	033312	005737	002234		TST	QVP				:IS QUICK VERIFY PASS SELECTED
4709	033316	001054			BNE	EXQV9				:IF YES EXIT TEST
4710	033320	005737	002322		TST	IIRCNT				:IS THIS THE FIRST TIME THROUGH THE TEST
4711	033324	001007			BNE	1\$				:BRANCH IF NO



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 51-4  
 TEST 9: SECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)

4712	033326	013737	002312	002406		MOV	DPA1,SDPA	:SAVE ENTERED DPA1
4713	033334	005037	002312			CLR	DPA1	:CLR DPA1
4714	033340	005237	002322			INC	ITRCNT	:SET FLAG TO SEE FIRST TIME PASS
4715	033344	005237	002312		1\$:	INC	DPA1	:GET NEW DPA1
4716	033350	023737	002312	002314		CMP	DPA1,DPA2	:IS DPA1 = DPA2
4717	033356	001002				BNE	2\$	:BRANCH IF NO
4718	033360	005237	002312			INC	DPA1	:INCREMENT DPA1
4719	033364	022737	000037	002312	2\$:	CMP	#37,DPA1	:ALL DONE
4720	033372	001423				BEQ	3\$	:BRANCH IF YES
4721	033374	052777	000010	146676		BIS	#10,@CSRX	:SELECT CHANNEL 2
4722	033402	112777	000217	146660		MOVB	#217,@ICRHX	:----LOAD SIC INTO ACR 2-----
4723	033410	004737	011072			JSR	PC,WAIT	:WAIT A LITTLE
4724	033414	112777	000017	146646		MOVB	#17,@ICRHX	:----LOAD NOT SIC INTO ACR 2-----
4725	033422	042777	000010	146650		BIC	#10,@CSRX	:SELECT CHANNEL 1
4726	033430	113777	002312	146616		MOVB	DPA1,@IIRHX	:LOAD NEW DPA1 INTO ADR1
4727	033436	000137	032026			JMP	A9	:REPET THE TEST WITH THE NEW DPA1
4728	033442	013737	002406	002312	3\$:	MOV	SDPA,DPA1	:RESTORE ENTERED DPA1
4729	033450				EXQV9:	EXIT	TST	
	033450	104432						TRAP C\$EXIT
	033452	000074						.WORD L10036-
4730								
4731								
4732	033454	045	123	062	TSHD9:	.NLIST	BEX	
4733						.ASCIZ	/S2%ASECONDARY ADDRESSING TEST OF CHANNEL 1 (LISTENER)%N/	
4734						.LIST	BEX	
4735	033546					.EVEN		
	033546					ENDTST		
	033546	104401						L10036: TRAP C\$ETST

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 52  
 TEST 10: SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)

```

4737 .SBTTL TEST 10: SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)
4738 :*****
4739 : IEX - TEST 10
4740 : THIS TEST CHECKS THE EXTENDED TALKER INTERFACE FUNCTION .
4741 :
4742 : PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING
4743 : FEATURE OF CHANNEL 1 BY MEANS OF RECEIVING A VALID AS WELL AS AN INVALID
4744 : MY SECONDARY ADDRESS (MSA1).
4745 :*****
4746 033550 BGNTST
4747 033550 T10::
4748 033550 005737 002324 TST PNTF ; IS THE PNT FLAG SET
4749 033554 001410 BEQ 7$ ; IF YES, PRINT THE TEST HEADER
033556 012746 035224 PRINTF #TSHD10 ;...
033562 012746 000001 MOV #TSHD10,-(SP)
033566 010600 MOV #1,-(SP)
033570 104417 MOV SP,RO
033572 062706 000004 TRAP C$PNTF
4750 033576 005037 002322 7$: CLR ITRCNT ; CLEAR ITERATION COUNTER
4751 033602 004737 010220 JSR PC,CULPA ; CLEAR ULPA BIT IN ISR 1 AND 2
4752 033606 004737 010710 JSR PC,BGIN2 ; SET UP PARAMETER
4753 033612 042777 000010 146460 A10: BIC #10,@CSRX ; SELECT CHANNEL 1
4754 033620 112777 000223 146442 MOVB #223,@ICRHX ; ----LOAD DAI INTO ACR1-----
4755 033626 112777 000020 146426 MOVB #20,@ISRHX ; ----LOAD APT BIT INTO ISR1-----
4756 033634 013701 002312 MOV DPA1,R1 ; GET DPA1 ADDRESS
4757 033640 062701 000100 ADD #100,R1 ; CREATE MY TALKER ADDRESS (MTA)
4758 033644 010137 002414 MOV R1,MTA1 ; STORE MTA
4759 033650 032737 000001 002312 BIT #1,DPA1 ; IS DPA EVEN
4760 033656 001412 BEQ 3$ ; BRANCH IF YES
4761 033660 052737 000001 002452 BIS #1,CDAT8 ; SET ULPA BIT IN COMPARE DATA FOR ISR
4762 033666 052737 000001 002436 BIS #1,CDAT2 ;...
4763 033674 052737 000001 002440 BIS #1,CDAT3 ;...
4764 033702 000411 BR +24 ; BRANCH TO BGNSEG
4765 033704 042737 000001 002452 3$: BIC #1,CDAT8 ; CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4766 033712 042737 000001 002436 BIC #1,CDAT2 ;...
4767 033720 042737 000001 002440 BIC #1,CDAT3 ;...
4768 033726 104404 BGNSEG
4769 033730 052777 000010 146342 BIS #10,@CSRX ; SELECT CHANNEL 2
4770 033736 012737 000002 002374 MOV #2,CHAN ; LOAD CHANNEL NUMBER
4771 033744 113777 002414 146324 MOVB MTA1,@IDRHX ; ----LOAD TALKER ADDRESS OF CHANNEL 2---
4772 033752 004737 011060 JSR PC,LOOP ; WAIT A LITTLE
4773 033756 042777 000010 146314 BIC #10,@CSRX ; SELECT CHANNEL 1
4774 033764 012737 000001 002374 MOV #1,CHAN ; LOAD CHANNEL NUMBER
4775 033772 017737 146252 002502 MOV @IIRX,BAD ; GET IIR1 CONTENTS
4776 034000 022737 000400 002502 CMP #400,BAD ; IFC,BIT IN IIR1 SHOULD BE SET
4777 034006 001410 BEQ 30$ ; BRANCH IF YES
4778 034010 012737 000400 002500 MOV #400,GOOD ; SET UP DATA FOR ERROR MESSAGES
4779 034016 104457 ERRSOFT 1001,E501,ERR501 ; ERROR HANDLER
034016 104457 TRAP C$ERRSOFT
034020 001751 .WORD 1001
034022 005406 .WORD E501
034024 003702 .WORD ERR501
4780 034026 CKLOOP ; BRANCH TO BGNSEG WHEN ERRLOOP IS SET
034026 104406 TRAP C$CLP1
4781 034030 017737 146222 002502 30$: MOV @ISRX,BAD ; GET ISR1 CONTENTS
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 52-1  
 TEST 10: SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)

4782	034036	023737	002436	002502		CMP	CDAT2,BAD		:ATN,ATN,NDAC,TPAS,(ULPA) BIT SET
4783	034044	001410				BEQ	40\$		:BRANCH IF YES
4784	034046	013737	002436	002500		MOV	CDAT2,GOOD		:SET UP DATA FOR ERROR MESSAGES
4785	034054					ERRSOFT	1002,E502,ERR501		:ERROR HANDLER
	034054	104457							TRAP CSERSOFT
	034056	001752							.WORD 1002
	034060	005447							.WORD E502
	034062	003702							.WORD ERR501
4786	034064					CKLOOP			:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	034064	104406							TRAP C\$CLP1
4787	034066	052777	000010	146204	40\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
4788	034074	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4789	034102	013701	002414			MOV	MTA1,R1		:GET MTA1
4790	034106	062701	000040			ADD	#40,R1		:CREATE MSA1
4791	034112	010137	002420			MOV	R1,MSA1		:STORE MSA1 TO LOCATION MSA1
4792	034116	113777	002420	146152		MOVB	MSA1,@IDRHX		:-----LOAD MSA1 INTO DOR2-----
4793	034124	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
4794	034130	017737	146114	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
4795	034136	022737	000000	002502		CMP	#0,BAD		:IIR2 SHOULD BE ZERO
4796	034144	001407				BEQ	41\$		:BRANCH IF YES
4797	034146	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGE
4798	034152					ERRSOFT	1003,E501,ERR501		:ERROR HANDLER
	034152	104457							TRAP CSERSOFT
	034154	001753							.WORD 1003
	034156	005406							.WORD E501
	034160	003702							.WORD ERR501
4799	034162					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	034162	104406							TRAP C\$CLP1
4800	034164	042777	000010	146106	41\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
4801	034172	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4802	034200	017737	146044	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
4803	034206	022737	010100	002502		CMP	#10100,BAD		:APT,INT1 BIT IN IIR1 SHOULD BE SET
4804	034214	001410				BEQ	42\$		:BRANCH IF YES
4805	034216	012737	010100	002500		MOV	#10100,GOOD		:SET UP DATA FOR ERROR MESSAGE
4806	034224					ERRSOFT	1004,E501,ERR501		:ERROR HANDLER
	034224	104457							TRAP CSERSOFT
	034226	001754							.WORD 1004
	034230	005406							.WORD E501
	034232	003702							.WORD ERR501
4807	034234					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	034234	104406							TRAP C\$CLP1
4808	034236	017737	146014	002502	42\$:	MOV	@ISRX,BAD		:GET ISR1 CONTENTS
4809	034244	023737	002452	002502		CMP	CDAT8,BAD		:ATN,DAV,NDAC,NRFD,ATN,TPAS(ULPA) IS SET
4810	034252	001410				BEQ	43\$		:BRANCH IF YES
4811	034254	013737	002452	002500		MOV	CDAT8,GOOD		:SET UP DATA FOR ERROR MESSAGE
4812	034262					ERRSOFT	1005,E502,ERR501		:ERROR HANDLER
	034262	104457							TRAP CSERSOFT
	034264	001755							.WORD 1005
	034266	005447							.WORD E502
	034270	003702							.WORD ERR501
4813	034272					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	034272	104406							TRAP C\$CLP1
4814	034274	017737	145764	002502	43\$:	MOV	@ICRX,BAD		:GET ICR1 CONTENTS
4815	034302	123737	002420	002502		CMPB	MSA1,BAD		:MSA1 ADDRESS SHOULD BE SET
4816	034310	001410				BEQ	44\$		:BRANCH IF YES
4817	034312	013737	002420	002500		MOV	MSA1,GOOD		:SET UP DATA FOR ERROR MESSAGE
4818	034320					ERRSOFT	1006,E901,ERR501		:ERROR HANDLER





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 52-3  
 TEST 10: SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)

4853	034604	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4854	034612	017737	145432	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
4855	034620	022737	000001	002502		CMP	#1,BAD		:MAC BIT IN IIR1 SHOULD BE SET
4856	034626	001410				BEQ	30\$		:BRANCH IF YES
4857	034630	012737	000001	002500		MOV	#1,GOOD		:SET UP DATA FOR ERROR MESSAGES
4858	034636					ERRSOFT	1010,E501,ERR501		:ERROR HANDLER
	034636	104457							TRAP C\$ERSOFT
	034640	001762							.WORD 1010
	034642	005406							.WORD E501
	034644	003702							.WORD ERR501
4859	034646					CKLOOP			:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	034646	104406							TRAP C\$CLP1
4860	034650	052777	000010	145422	30\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
4861	034656	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4862	034664	013701	002414			MOV	MTA1,R1		:GET MLA1
4863	034670	062701	000040			ADD	#40,R1		:CREATE MSA1
4864	034674	010137	002420			MOV	R1,MSA1		:STORE MSA1 TO LOCATION MSA1
4865	034700	113777	002420	145370		MOVB	MSA1,@IDRHX		:-----LOAD MSA1 INTO DOR2-----
4866	034706	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
4867	034712	042777	000010	145360		BIC	#10,@CSRX		:SELECT CHANNEL 1
4868	034720	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4869	034726	112777	000001	145334		MOVB	#1,@ICRHX		:-----LOAD NOT DACR INTO ACR1-----
4870	034734	017737	145310	002502		MOV	@IIRX,BAD		:READ IIR1 FOR CLEAR THE BITS
4871	034742	017737	145310	002502		MOV	@ISRX,BAD		:GET ISR1 CONTENTS
4872	034750	023737	002436	002502		CMP	CDAT2,BAD		:ATN,NDAC,ATN,TPAS (ULPA) IS SET
4873	034756	001410				BEQ	46\$		:BRANCH IF YES
4874	034760	013737	002436	002500		MOV	CDAT2,GOOD		:SET UP DATA FOR ERROR MESSAGE
4875	034766					ERRSOFT	1011,E502,ERR501		:ERROR HANDLER
	034766	104457							TRAP C\$ERSOFT
	034770	001763							.WORD 1011
	034772	005447							.WORD E502
	034774	003702							.WORD ERR501
4876	034776					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	034776	104406							TRAP C\$CLP1
4877	035000	052777	000010	145272	46\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
4878	035006	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4879	035014	017737	145230	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
4880	035022	022737	000020	002502		CMP	#20,BAD		:BO BIT SHOULD BE SET
4881	035030	001410				BEQ	50\$		:BRANCH IF YES
4882	035032	012737	000020	002500		MOV	#20,GOOD		:SET UP DATA FOR ERROR MESSAGE
4883	035040					ERRSOFT	1012,E501,ERR501		:ERROR HANDLER
	035040	104457							TRAP C\$ERSOFT
	035042	001764							.WORD 1012
	035044	005406							.WORD E501
	035046	003702							.WORD ERR501
4884	035050					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	035050	104406							TRAP C\$CLP1
4885	035052	112777	000137	145216	50\$:	MOVB	#137,@IDRHX		:---LOAD UNT INTO DOR2-----
4886	035060					ENDSEG			
	035060								10001\$:
	035060	104405							TRAP C\$ESEG
4887	035062	005737	002234			TST	QVP		:IS QUICK VERIFY PASS SELECTED
4888	035066	001054				BNE	EXQV10		:IF YES EXIT TEST
4889	035070	005737	002322			TST	ITRCNT		:IS THIS THE FIRST TIME THROUGH THE TEST
4890	035074	001007				BNE	1\$		:BRANCH IF NO
4891	035076	013737	002312	002406		MOV	DPA1,SDPA		:SAVE ENTERED DPA1
4892	035104	005037	002312			CLR	DPA1		:CLR DPA1

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 52-4  
TEST 10: SECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)

```

4893 035110 005237 002322          INC      ITRCNT      :SET FLAG TO SEE FIRST TIME PASS
4894 035114 005237 002312          INC      DPA1        :GET NEW DPA1
4895 035120 023737 002312 002314 1$:    CMP      DPA1,DPA2   :IS DPA1 = DPA2
4896 035126 001002          BNE      2$         :BRANCH IF NO
4897 035130 005237 002312          INC      DPA1        :INCREMENT DPA1
4898 035134 022737 000037 002312 2$:    CMP      #37,DPA1   :ALL DONE
4899 035142 001423          BEQ      3$         :BRANCH IF YES
4900 035144 052777 000010 145126          BIS      #10,@CSRx  :SELECT CHANNEL 2
4901 035152 112777 000217 145110          MOVB     #217,@ICRHx :----LOAD SIC INTO ACR 2-----
4902 035160 004737 011072          JSR      PC,WAIT    :WAIT A LITTLE
4903 035164 112777 000017 145076          MOVB     #17,@ICR4x  :----LOAD NOT SIC INTO ACR 2---
4904 035172 042777 000010 145100          BIC      #10,@CSRx  :SELECT CHANNEL 1
4905 035200 113777 002312 145046          MOVB     DPA1,@IIRHx :LOAD NEW DPA1 INTO ADR1
4906 035206 000137 033612          JMP      A10        :REPEAT THE TEST WITH THE NEW DPA1
4907 035212 013737 002406 002312 3$:    MOV      SDPA,DPA1  :RESTORE ENTERED DPA1
4908 035220          EXQV10: EXIT TST
      035220 104432
      035222 000072          TRAP    C$EXIT
                                   .WORD    L10037-.

4909
4910
4911 035224      045      123      062  TSMD10: .NLIST  BEX
      .ASCIZ  /%S2%ASECONDARY ADDRESSING TEST OF CHANNEL 1 (TALKER)%N/
4912      .LIST  BEX
4913      .EVEN
4914 035314          ENDT$1
      035314
      035314 104401          L10037: TRAP    C$ETST

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 53  
TEST 11: SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)

```

4916 .SBTTL TEST 11: SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)
4917 :.....
4918 :                                IEX = TEST 11
4919 : THIS TEST CHECKS THE EXTENDED LISTENER INTERFACE FUNCTION .
4920 :
4921 : PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING
4922 : FEATURE OF CHANNEL 2 BY MEANS OF RECEIVING A VALID AS WELL AS
4923 : AN INVALID MY SECONDARY ADDRESS (MSA2).
4924 :.....
4925 035316          BGNTST
035316
4926 035316 005737 002324          TST      PNTF          : IS THE PNT FLAG SET
4927 035322 001410          BEQ      7$          : IF YES, PRINT THE TEST HEADER
4928 035324          PRINTF   #TSHD11      : ...
035324 012746 036772          MOV      #TSHD11,-(SP)
035330 012746 000001          MOV      #1,-(SP)
035334 010600          MOV      SP,R0
035336 104417          TRAP    C$PNTF
035340 062706 000004          ADD     #4,SP
4929 035344 005037 002322          7$: CLR     ITRCNT      : CLEAR COUNTER
4930 035350 004737 010220          JSR    PC,CULPA     : CLEAR ULPA IN ISR 1 AND 2
4931 035354 004737 010534          JSR    PC,BGIN1    : SET UP PARAMETER
4932 035360 052777 000010 144712          A11: BIS    #10,@CSRX   : SELECT CHANNEL 2
4933 035366 112777 000223 144674          MOV    #223,@ICRHX : ----LOAD DAI INTO ACR2-----
4934 035374 112777 000020 144660          MOV    #20,@ISRHX  : ----LOAD APT BIT INTO ISR2-----
4935 035402 013701 002314          MOV    DPA2,R1     : GET DPA2 ADDRESS
4936 035406 062701 000040          ADD    #40,R1      : CREATE MY LISTENER ADDRESS (MLA)
4937 035412 010137 002412          MOV    R1,MLA2     : STORE MLA
4938 035416 032737 000001 002314          BIT    #1,DPA2     : IS DPA EVEN
4939 035424 001412          BEQ    3$          : BRANCH IF YES
4940 035426 052737 000001 002446          BIS    #1,CDAT6    : SET ULPA BIT IN COMPARE DATA FOR ISR
4941 035434 052737 000001 002442          BIS    #1,CDAT4
4942 035442 052737 000001 002444          BIS    #1,CDAT5
4943 035450 000411          BR     .+24        : BRANCH TO BGNSEG
4944 035452 042737 000001 002446          3$: BIC    #1,CDAT6    : CLEAR ULPA BIT IN COMPARE DATA FOR ISR
4945 035460 042737 000001 002442          BIC    #1,CDAT4
4946 035466 042737 000001 002444          BIC    #1,CDAT5
4947 035474          BGNSEG
035474 104404          TRAP    C$BSEG
4948 035476 042777 000010 144574          BIC    #10,@CSRX   : SELECT CHANNEL 1
4949 035504 012737 000001 002374          MOV    #1,CHAN     : LOAD CHANNEL NUMBER
4950 035512 113777 002412 144556          MOV    MLA2,@IDRHX : ----LOAD LISTENER ADDRESS OF CHANNEL 1--
4951 035520 004737 011060          JSR    PC,LOOP     : WAIT A LITTLE
4952 035524 052777 000010 144546          BIS    #10,@CSRX   : SELECT CHANNEL 2
4953 035532 012737 000002 002374          MOV    #2,CHAN     : LOAD CHANNEL NUMBER
4954 035540 017737 144504 002502          MOV    @IIRX,BAD   : GET IIR2 CONTENTS
4955 035546 022737 000400 002502          CMP    #400,BAD    : IFC,BIT IN IIR2 SHOULD BE SET
4956 035554 001410          BEQ    30$        : BRANCH IF YES
4957 035556 012737 000400 002500          MOV    #400,GOOD   : SET UP DATA FOR ERROR MESSAGES
4958 035564          ERRSOFT 1101,E501,ERR501 : ERROR HANDLER
035564 104457          TRAP    C$ERRSOFT
035566 002115          .WORD  1101
035570 005406          .WORD  E501
035572 003702          .WORD  ERR501
4959 035574          CKLOOP          : BRANCH TO BGNSEG WHEN ERRLOOP IS SET
035574 104406          TRAP    C$CLP1
4960 035576 017737 144454 002502 30$: MOV    @ISRX,BAD   : GET ISR2 CONTENTS

```

HARDWARE TESTS MACRO M113 06-SEP-82 16:46 PAGE 53-1  
 TEST 11: SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)

4961	035604	023737	002442	002502		CMP	CDAT4,BAD		:ATN,ATN,NDAC,LPAS,(ULPA) BIT SET
4962	035612	001410				BEQ	40\$		:BRANCH IF YES
4963	035614	013737	002442	002500		MOV	CDAT4,GOOD		:SET UP DATA FOR ERROR MESSAGES
4964	035622					ERRSOFT	1102,E502,ERR501		:ERROR HANDLER
	035622	104457							TRAP C\$ERSOFT
	035624	002116							.WORD 1102
	035626	005447							.WORD E502
	035630	003702							.WORD ERR501
4965	035632					CKLOOP			:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	035632	104406							TRAP C\$CLP1
4966	035634	042777	000010	144436	40\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
4967	035642	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
4968	035650	013701	002412			MOV	MLA2,R1		:GET MLA2
4969	035654	062701	000100			ADD	#100,R1		:CREATE MSA2
4970	035660	010137	002420			MOV	R1,MSA1		:STORE MSA1 TO LOCATION MSA1
4971	035664	113777	002420	144404		MOVB	MSA1,@IDRHX		:-----LOAD MSA1 INTO DOR1-----
4972	035672	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
4973	035676	017737	144346	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
4974	035704	022737	000000	002502		CMP	#0,BAD		:IIR1 SHOULD BE ZERO
4975	035712	001407				BEQ	41\$		:BRANCH IF YES
4976	035714	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGE
4977	035720					ERRSOFT	1103,E501,ERR501		:ERROR HANDLER
	035720	104457							TRAP C\$ERSOFT
	035722	002117							.WORD 1103
	035724	005406							.WORD E501
	035726	003702							.WORD ERR501
4978	035730					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	035730	104406							TRAP C\$CLP1
4979	035732	052777	000010	144340	41\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
4980	035740	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
4981	035746	017737	144276	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
4982	035754	022737	010100	002502		CMP	#10100,BAD		:APT,INT1 BIT IN IIR2 SHOULD BE SET
4983	035762	001410				BEQ	42\$		:BRANCH IF YES
4984	035764	012737	010100	002500		MOV	#10100,GOOD		:SET UP DATA FOR ERROR MESSAGE
4985	035772					ERRSOFT	1104,E501,ERR501		:ERROR HANDLER
	035772	104457							TRAP C\$ERSOFT
	035774	002120							.WORD 1104
	035776	005406							.WORD E501
	036000	003702							.WORD ERR501
4986	036002					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	036002	104406							TRAP C\$CLP1
4987	036004	017737	144246	002502	42\$:	MOV	@ISRX,BAD		:GET ISR2 CONTENTS
4988	036012	023737	002446	002502		CMP	CDAT6,BAD		:ATN,DAV,NDAC,NRFD,ATN,LPAS(ULPA) IS SET
4989	036020	001410				BEQ	43\$		:BRANCH IF YES
4990	036022	013737	002446	002500		MOV	CDAT6,GOOD		:SET UP DATA FOR ERROR MESSAGE
4991	036030					ERRSOFT	1105,E502,ERR501		:ERROR HANDLER
	036030	104457							TRAP C\$ERSOFT
	036032	002121							.WORD 1105
	036034	005447							.WORD E502
	036036	003702							.WORD ERR501
4992	036040					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	036040	104406							TRAP C\$CLP1
4993	036042	017737	144216	002502	43\$:	MOV	@ICRX,BAD		:GET ICR2 CONTENTS
4994	036050	123737	002420	002502		CMPB	MSA1,BAD		:MSA1 ADDRESS SHOULD BE SET
4995	036056	001410				BEQ	44\$		:BRANCH IF YES
4996	036060	013737	002420	002500		MOV	MSA1,GOOD		:SET UP DATA FOR ERROR MESSAGE
4997	036066					ERRSOFT	1106,E901,ERR501		:ERROR HANDLER



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 53-2  
 TEST 11: SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)

036066	104457								TRAP	C\$ERSOFT
036070	002122								.WORD	1106
036072	005644								.WORD	E901
036074	003702								.WORD	ERR501
4998	036076			CKLOOP					:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET	TRAP C\$CLP1
	036076	104406								
4999	036100	042777	000010	144172	44\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1	
5000	036106	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER	
5001	036114	017737	144136	002502		MOV	@ISRX,BAD		:GET ISR1 CONTENTS	
5002	036122	022737	170040	002502		CMP	#170040,BAD		:ATN,DAV,NDAC,ATN,NRFD SHOULD BE SET	
5003	036130	001410				BEQ	45\$		:BRANCH IF YES	
5004	036132	012737	170040	002500		MOV	#170040,GOOD		:SET UP DATA FOR ERROR MESSAGE	
5005	036140					ERRSOFT	1107,E502,ERR501		:ERROR HANDLER	
	036140	104457							TRAP	C\$ERSOFT
	036142	002123							.WORD	1107
	036144	005447							.WORD	E502
	036146	003702							.WORD	ERR501
5006	036150			CKLOOP					:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET	TRAP C\$CLP1
	036150	104406								
5007	036152	052777	000010	144120	45\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2	
5008	036160	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER	
5009	036166	112777	000201	144074		MOVB	#201,@ICRHX		:-----LOAD DACR INTO ACR2-----	
5010	036174	017737	144056	002502		MOV	@ISRX,BAD		:GET ISR2 CONTENTS	
5011	036202	023737	002444	002502		CMP	CDAT5,BAD		:ATN,NDAC,ATN,LADS,LPAS (ULPA) IS SET	
5012	036210	001410				BEQ	46\$		:BRANCH IF YES	
5013	036212	013737	002444	002500		MOV	CDAT5,GOOD		:SET UP DATA FOR ERROR MESSAGE	
5014	036220					ERRSOFT	1108,E502,ERR501		:ERROR HANDLER	
	036220	104457							TRAP	C\$ERSOFT
	036222	002124							.WORD	1108
	036224	005447							.WORD	E502
	036226	003702							.WORD	ERR501
5015	036230			CKLOOP					:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET	TRAP C\$CLP1
	036230	104406								
5016	036232	042777	000010	144040	46\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1	
5017	036240	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER	
5018	036246	017737	143776	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS	
5019	036254	022737	000020	002502		CMP	#20,BAD		:BO BIT SHOULD BE SET	
5020	036262	001410				BEQ	50\$		:BRANCH IF YES	
5021	036264	012737	000020	002500		MOV	#20,GOOD		:SET UP DATA FOR ERROR MESSAGE	
5022	036272					ERRSOFT	1109,E501,ERR501		:ERROR HANDLER	
	036272	104457							TRAP	C\$ERSOFT
	036274	002125							.WORD	1109
	036276	005406							.WORD	E501
	036300	003702							.WORD	ERR501
5023	036302			CKLOOP					:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET	TRAP C\$CLP1
	036302	104406								
5024	036304	112777	000077	143764	50\$:	MOVB	#77,@IDRHX		:---LOAD UNL INTO DOR1-----	
5025	036312					ENDSEG				
	036312								10000\$:	
	036312	104405							TRAP	C\$ESEG
5026	036314			COPA11: BGNSEG					TRAP	C\$BSEG
	036314	104404								
5027	036316	042777	000010	143754		BIC	#10,@CSRX		:SELECT CHANNEL 1	
5028	036324	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER	
5029	036332	113777	002412	143736		MOVB	MLA2,@IDRHX		:---LOAD LISTENER ADDRESS OF CHANNEL 1---	
5030	036340	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE	
5031	036344	052777	000010	143726		BIS	#10,@CSRX		:SELECT CHANNEL 2	



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 53-3  
 TEST 11: SECONDARY ADDRESSING TEST OF CHANNEL 2 (LISTENER)

5032	036352	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
5033	036360	017737	143664	002502		MOV	@IIRX,BAD		:GET IIR2 CONTENTS
5034	036366	022737	000001	002502		CMP	#1,BAD		:MAC BIT IN IIR2 SHOULD BE SET
5035	036374	001410				BEQ	30\$		:BRANCH IF YES
5036	036376	012737	000001	002500		MOV	#1,GOOD		:SET UP DATA FOR ERROR MESSAGES
5037	036404					ERRSOFT	1110,E501,ERR501		:ERROR HANDLER
	036404	104457							TRAP C\$ERSOFT
	036406	002126							.WORD 1110
	036410	005406							.WORD E501
	036412	003702							.WORD ERR501
5038	036414					CKLOOP			:BRANCH TO BGNSEG WHEN ERRLOOP IS SET
	036414	104406							TRAP C\$CLP1
5039	036416	042777	000010	143654	30\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
5040	036424	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
5041	036432	013701	002412			MOV	MLA2,R1		:GET MLA2
5042	036436	062701	000100			ADD	#100,R1		:CREATE MSA1
5043	036442	010137	002420			MOV	R1,MSA1		:STORE MSA1 TO LOCATION MSA1
5044	036446	113777	002420	143622		MOVB	MSA1,@IDRHX		:-----LOAD MSA1 INTO DOR1-----
5045	036454	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
5046	036460	052777	000010	143612		BIS	#10,@CSRX		:SELECT CHANNEL 2
5047	036466	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
5048	036474	112777	000001	143566		MOVB	#1,@ICRX		:-----LOAD NOT DACR INTO ACR2-----
5049	036502	017737	143542	002502		MOV	@IIRX,BAD		:READ IIR2 FOR CLEAR THE BITS
5050	036510	017737	143542	002502		MOV	@ISRX,BAD		:GET ISR2 CONTENTS
5051	036516	023737	002442	002502		CMP	CDAT4,BAD		:ATN,NDAC,ATN,LPAS (ULPA) IS SET
5052	036524	001410				BEQ	46\$		:BRANCH IF YES
5053	036526	013737	002442	002500		MOV	CDAT4,GOOD		:SET UP DATA FOR ERROR MESSAGE
5054	036534					ERRSOFT	1111,E502,ERR501		:ERROR HANDLER
	036534	104457							TRAP C\$ERSOFT
	036536	002127							.WORD 1111
	036540	005447							.WORD E502
	036542	003702							.WORD ERR501
5055	036544					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	036544	104406							TRAP C\$CLP1
5056	036546	042777	000010	143524	46\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
5057	036554	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
5058	036562	017737	143462	002502		MOV	@IIRX,BAD		:GET IIR1 CONTENTS
5059	036570	022737	000020	002502		CMP	#20,BAD		:BO BIT SHOULD BE SET
5060	036576	001410				BEQ	50\$		:BRANCH IF YES
5061	036600	012737	000020	002500		MOV	#20,GOOD		:SET UP DATA FOR ERROR MESSAGE
5062	036606					ERRSOFT	1112,E501,ERR501		:ERROR HANDLER
	036606	104457							TRAP C\$ERSOFT
	036610	002130							.WORD 1112
	036612	005406							.WORD E501
	036614	003702							.WORD ERR501
5063	036616					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	036616	104406							TRAP C\$CLP1
5064	036620	112777	000077	143450	50\$:	MOVB	#77,@IDRHX		:---LOAD UNL INTO DOR1-----
5065	036626					ENDSEG			
	036626								10001\$:
	036626	104405							TRAP C\$ESEG
5066	036630	005737	002234			TST	QVP		:IS QUICK VERIFY PASS SELECTED
5067	036634	001054				BNE	EXQV11		:IF YES EXIT TEST
5068	036636	005737	002322			TST	ITRCNT		:IS THIS THE FIRST TIME THROUGH THE TEST
5069	036642	001007				BNE	1\$		:BRANCH IF NO
5070	036644	013737	002314	002406		MOV	DPA2,SDPA		:SAVE ENTERED DPA2
5071	036652	005037	002314			CLR	DPA2		:CLEAR DPA2



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 54  
TEST 12: SECONDARY ADDRESSING TEST OF CHANNEL 2 (TALKER)

```

5095 .SBTTL TEST 12: SECONDARY ADDRESSING TEST OF CHANNEL 2 (TALKER)
5096 :*****
5097 :IEX - TEST 12
5098 :THIS TEST CHECKS THE EXTENDED TALKER INTERFACE FUNCTION .
5099 :
5100 :PURPOSE OF THIS TEST IS TO CHECK THE SECONDARY ADDRESSING
5101 :FEATURE OF CHANNEL 2 BY MEANS OF RECEIVING A VALID AS WELL AS
5102 :AN INVALID MY SECONDARY ADDRESS (MSA2).
5103 :*****
5104 037066 BGNTST
5105 037066 005737 002324 TST PNTF ;IS THE PNT FLAG SET
5106 037072 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
5107 037074 PRINTF #TSHD12 ;...
037074 012746 040652 MOV #TSHD12,-(SP)
037100 012746 000001 MOV #1,-(SP)
037104 010600 MOV SP,R0
037106 104417 TRAP C$PNTF
037110 062706 000004 ADD #4,SP
5108 037114 005037 002322 7$: CLR ITRCNT ;CLEAR COUNTER
5109 037120 004737 010220 JSR PC,CULPA ;CLEAR ULPA IN ISR 1 AND 2
5110 037124 004737 010534 JSR PC,BGIN1 ;SET UP PARAMETER
5111 037130 052777 000010 143142 A12: BIS #10,@CSRX ;SELECT CHANNEL 2
5112 037136 112777 000223 143124 MOVB #223,@ICRHX ;-----LOAD DAI INTO ACR2-----
5113 037144 112777 000020 143110 MOVB #20,@ISRHX ;-----LOAD APT BIT INTO ISR2-----
5114 037152 013701 002314 MOV DPA2,R1 ;GET DPA2 ADDRESS
5115 037156 062701 000100 ADD #100,R1 ;CREATE MY TALKER ADDRESS (MTA)
5116 037162 010137 002416 MOV R1,MTA2 ;STORE MTA
5117 037166 032737 000001 002314 BIT #1,DPA2 ;IS DPA EVEN
5118 037174 001412 BEQ 3$ ;BRANCH IF YES
5119 037176 052737 000001 002452 BIS #1,CDAT8 ;SET ULPA BIT IN COMPARE DATA FOR ISR
5120 037204 052737 000001 002436 BIS #1,CDAT2
5121 037212 052737 000001 002440 BIS #1,CDAT3
5122 037220 000411 BR +24 ;BRANCH TO BGNSEG
5123 037222 042737 000001 002452 3$: BIC #1,CDAT8 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
5124 037230 042737 000001 002436 BIC #1,CDAT2
5125 037236 042737 000001 002440 BIC #1,CDAT3
5126 037244 BGNSEG ;...
037244 104404 TRAP C$BSEG
5127 037246 042777 000010 143024 BIC #10,@CSRX ;SELECT CHANNEL 1
5128 037254 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5129 037262 113777 002416 143006 MOVB MTA2,@IDRHX ;----LOAD TALKER ADDRESS OF CHANNEL 1---
5130 037270 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5131 037274 052777 000010 142776 BIS #10,@CSRX ;SELECT CHANNEL 2
5132 037302 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
5133 037310 017737 142734 002402 MOV @IIRX,RSAVE ;GET IIR2 CONTENTS
5134 037316 022737 000400 002402 CMP #400,RSAVE ;IFC,BIT IN IIR2 SHOULD BE SET
5135 037324 001413 BEQ 30$ ;BRANCH IF YES
5136 037326 012737 000400 002500 MOV #400,GOOD ;SET UP DATA FOR ERROR MESSAGES
5137 037334 013737 002402 002502 MOV RSAVE,BAD
5138 037342 ERRSOFT 1201,E501,ERR501 ;ERROR HANDLER
037342 104457 TRAP C$ERRSOFT
037344 002261 .WORD 1201
037346 005406 .WORD E501
037350 003702 .WORD ERR501
5139 037352 CKLOOP ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
037352 104406 TRAP C$CLP1

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 54-2  
 TEST 12: SECONDARY ADDRESSING TEST OF CHANNEL 2 (TALKER)

5177	037650	017737	142410	002402	43\$:	MOV	@ICRX,RSAVE		:GET ICR2 CONTENTS
5178	037656	123737	002420	002402		CMPB	MSA1,RSAVE		:MSA1 ADDRESS SHOULD BE SET
5179	037664	001413				BEQ	44\$		:BRANCH IF YES
5180	037666	013737	002420	002500		MOV	MSA1,GOOD		:SET UP DATA FOR ERROR MESSAGE
5181	037674	013737	002402	002502		MOV	RSAVE,BAD		
5182	037702					ERRSOFT	1206,E901,ERR501		:ERROR HANDLER
	037702	104457							TRAP CSERSOFT
	037704	002266							.WORD 1206
	037706	005644							.WORD E901
	037710	003702							.WORD ERR501
5183	037712					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	037712	104406							TRAP C\$CLP1
5184	037714	042777	000010	142356	44\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
5185	037722	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
5186	037730	017737	142322	002402		MOV	@ISR1,RSAVE		:GET ISR1 CONTENTS
5187	037736	022737	170040	002402		CMP	#170040,RSAVE		:ATN,DAV,NDAC,ATN,NRFD SHOULD BE SET
5188	037744	001413				BEQ	45\$		:BRANCH IF YES
5189	037746	012737	170040	002500		MOV	#170040,GOOD		:SET UP DATA FOR ERROR MESSAGE
5190	037754	013737	002402	002502		MOV	RSAVE,BAD		
5191	037762					ERRSOFT	1207,E502,ERR501		:ERROR HANDLER
	037762	104457							TRAP CSERSOFT
	037764	002267							.WORD 1207
	037766	005447							.WORD E502
	037770	003702							.WORD ERR501
5192	037772					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	037772	104406							TRAP C\$CLP1
5193	037774	052777	000010	142276	45\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
5194	040002	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
5195	040010	112777	000201	142252		MOVB	#201,@ICRHX		:-----LOAD DACR INTO ACR2-----
5196	040016	017737	142234	002402		MOV	@ISR2,RSAVE		:GET ISR2 CONTENTS
5197	040024	023737	002440	002402		CMP	CDAT3,RSAVE		:ATN,NDAC,ATN,TADS,TPAS (ULPA) IS SET
5198	040032	001413				BEQ	46\$		:BRANCH IF YES
5199	040034	013737	002440	002500		MOV	CDAT3,GOOD		:SET UP DATA FOR ERROR MESSAGE
5200	040042	013737	002402	002502		MOV	RSAVE,BAD		
5201	040050					ERRSOFT	1208,E502,ERR501		:ERROR HANDLER
	040050	104457							TRAP CSERSOFT
	040052	002270							.WORD 1208
	040054	005447							.WORD E502
	040056	003702							.WORD ERR501
5202	040060					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	040060	104406							TRAP C\$CLP1
5203	040062	042777	000010	142210	46\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
5204	040070	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
5205	040076	017737	142146	002402		MOV	@IIR1,RSAVE		:GET IIR1 CONTENTS
5206	040104	022737	000020	002402		CMP	#20,RSAVE		:BO BIT SHOULD BE SET
5207	040112	001413				BEQ	50\$		:BRANCH IF YES
5208	040114	012737	000020	002500		MOV	#20,GOOD		:SET UP DATA FOR ERROR MESSAGE
5209	040122	013737	002402	002502		MOV	RSAVE,BAD		
5210	040130					ERRSOFT	1209,E501,ERR501		:ERROR HANDLER
	040130	104457							TRAP CSERSOFT
	040132	002271							.WORD 1209
	040134	005406							.WORD E501
	040136	003702							.WORD ERR501
5211	040140					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	040140	104406							TRAP C\$CLP1
5212	040142	112777	000137	142126	50\$:	MOV3	#137,@IDRHX		:---LOAD UNT INTO DOR1-----
5213	040150					ENDSEG			



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 54-3  
TEST 12: SECONDARY ADDRESSING TEST OF CHANNEL 2 (TALKER)

```

040150
5214 040150 104405 COPA12: BGNSEG
040152 104404
5215 040154 042777 000010 142116 BIC #10,@CSRX ;SELECT CHANNEL 1
5216 040162 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5217 040170 113777 002416 142100 MOVB MTA2,@IDRHX ;----LOAD TALKER ADDRESS OF CHANNEL 1---
5218 040176 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5219 040202 052777 000010 142070 BIS #10,@CSRX ;SELECT CHANNEL 2
5220 040210 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
5221 040216 017737 142026 002402 MOV @IIRX,RSAVE ;GET IIR2 CONTENTS
5222 040224 022737 000001 002402 CMP #1,RSAVE ;MAC BIT IN IIR2 SHOULD BE SET
5223 040232 001413 BEQ 30$ ;BRANCH IF YES
5224 040234 012737 000001 002500 MOV #1,GOOD ;SET UP DATA FOR ERROR MESSAGES
5225 040242 013737 002402 002502 MOV RSAVE,BAD
5226 040250 ERRSOFT 1210,E501,ERR501 ;ERROR HANDLER
040250 104457 TRAP CSERSOFT
040252 002272 .WORD 1210
040254 005406 .WORD E501
040256 003702 .WORD ERR501
5227 040260 CKLOOP ;BRANCH TO BGNSEG WHEN ERRLOOP IS SET
040260 104406 TRAP C$CLP1
5228 040262 042777 000010 142010 30$: BIC #10,@CSRX ;SELECT CHANNEL 1
5229 040270 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5230 040276 013701 002416 MOV MTA2,R1 ;GET MTA2
5231 040302 062701 000040 ADD #40,R1 ;CREATE MSA1
5232 040306 010137 002420 MOV R1,MSA1 ;STORE MSA1 TO LOCATION MSA1
5233 040312 113777 002420 141756 MOVB MSA1,@IDRHX ;-----LOAD MSA1 INTO DOR1-----
5234 040320 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5235 040324 052777 000010 141746 45$: BIS #10,@CSRX ;SELECT CHANNEL 2
5236 040332 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
5237 040340 112777 000001 141722 MOVB #1,@ICRHX ;----LOAD NOT DACR INTO ACR2-----
5238 040346 017737 141676 002502 MOV @IIRX,BAD ;READ IIR2 FOR CLEAR THE BITS
5239 040354 017737 141676 002402 MOV @ISRX,RSAVE ;GET ISR2 CONTENTS
5240 040362 023737 002436 002402 CMP CDAT2,RSAVE ;ATN,NDAC,ATN,TPAS (ULPA) IS SET
5241 040370 001413 BEQ 46$ ;BRANCH IF YES
5242 040372 013737 002436 002500 MOV CDAT2,GOOD ;SET UP DATA FOR ERROR MESSAGE
5243 040400 013737 002402 002502 MOV RSAVE,BAD
5244 040406 ERRSOFT 1211,E502,ERR501 ;ERROR HANDLER
040406 104457 TRAP CSERSOFT
040410 002273 .WORD 1211
040412 005447 .WORD E502
040414 003702 .WORD ERR501
5245 040416 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
040416 104406 TRAP C$CLP1
5246 040420 042777 000010 141652 46$: BIC #10,@CSRX ;SELECT CHANNEL 1
5247 040426 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5248 040434 017737 141610 002402 MOV @IIRX,RSAVE ;GET IIR1 CONTENTS
5249 040442 022737 000020 002402 CMP #20,RSAVE ;BO BIT SHOULD BE SET
5250 040450 001413 BEQ 50$ ;BRANCH IF YES
5251 040452 012737 000020 002500 MOV #20,GOOD ;SET UP DATA FOR ERROR MESSAGE
5252 040460 013737 002402 002502 MOV RSAVE,BAD
5253 040466 ERRSOFT 1212,E501,ERR501 ;ERROR HANDLER
040466 104457 TRAP CSERSOFT
040470 002274 .WORD 1212
040472 005406 .WORD E501
040474 003702 .WORD ERR501

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 55  
 TEST 13: DEVICE CLEAR INTERFACE FUNCTION TEST

```

5286 .SBTTL TEST 13: DEVICE CLEAR INTERFACE FUNCTION TEST
5287 :*****
5288 : IEX - TEST 13
5289 : PART 1 CHECKS THE DEVICE CLEAR INTERFACE FUNCTION OF CHANNEL 2 BY MEANS
5290 : OF RECEIVING A UNIVERSAL COMMAND (DCL) AS WELL AS AN ADDRESS
5291 : COMMAND (SDC).
5292 :
5293 : PART 2 CHECKS THE DEVICE CLEAR INTERFACE FUNCTION OF CHANNEL 1 BY MEANS
5294 : OF RECEIVING A UNIVERSAL COMMAND (DCL) AS WELL AS AN ADDRESS
5295 : COMMAND (SDC).
5296 :*****
5297 040744 BGNTST
5298 040744 005737 002324 TST PNTF ;IS THE PNT FLAG SET
5299 040750 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
5300 040752 PRINTF #TSHD13 ;...
5301 040752 012746 043040 MOV #TSHD13,-(SP)
5302 040756 012746 000001 MOV #1,-(SP)
5303 040762 010600 MOV SP,RO
5304 040764 104417 TRAP C$PNTF
5305 040766 062706 000004 ADD #4,SP
5306 040772 005037 002322 7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
5307 040776 004737 010220 JSR PC,CULPA ;CLEAR ULPA IN ISR 1 AND 2
5308 041002 004737 010534 ITAC13: JSR PC,BGIN1
5309 041006 032737 000001 002314 BIT #1,DPA2
5310 041014 001404 BEQ 3$ ;IS DPA EVEN
5311 041016 052737 000001 002454 BIS #1,CDAT9 ;BRANCH IF YES
5312 041024 000403 BR +10 ;SET ULPA BIT IN COMPARE DATA FOR ISR
5313 041026 042737 000001 002454 3$: BIC #1,CDAT9 ;BRANCH TO BGNSEG
5314 041034 104404 BGNSEG ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
5315 041036 052777 000010 141234 TRAP C$BSEG
5316 041044 112777 000223 141216 BIS #10,@CSRX ;SELECT CHANNEL 2
5317 041052 112777 000010 141202 MOVB #223,@ICRHX ;----LOAD DAI INTO ACR 2-----
5318 041060 042777 000010 141212 MOVB #10,@ISRHX ;---LOAD DCAS BIT INTO ISR2 REGISTER
5319 041066 112777 000024 141202 BIC #10,@CSRX ;SELECT CHANNEL 1
5320 041074 004737 011060 JSR PC,LOOP ;----LOAD DCL INTO DOR 1-----
5321 041100 052777 000010 141172 ;WAIT A LITTLE
5322 041106 012737 000002 002374 BIS #10,@CSRX ;SELECT CHANNEL 2
5323 041114 017737 141130 002402 MOV #2,CHAN ;LOAD CHANNEL NUMBER
5324 041122 022737 004500 002402 MOV @IIRX,RSAVE ;GET IIR2 CONTENTS
5325 041130 001413 BEQ 10$ ;DCAS,INT1,IFC BIT SHOULD BE SET
5326 041132 012737 004500 002500 MOV #4500,GOOD ;BRANCH IF YES
5327 041140 013737 002402 002502 MOV RSAVE,BAD ;SET UP DATA FOR ERROR MESSAGE
5328 041146 104457 ERRSOFT 1301,E501,ERR501 ;...
5329 041150 002425 TRAP C$ERSOFT
5330 041152 005406 .WORD 1301
5331 041154 003702 .WORD E501
5332 041156 104406 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET
5333 041160 017737 141100 002402 10$: MOV @ICRX,RSAVE ;GET ICR2 CONTENTS
5334 041166 122737 000024 002402 CMPB #24,RSAVE ;ICR2 CONTENTS SHOULD BE 24
5335 041174 001413 BEQ 20$ ;BRANCH IF YES
5336 041176 012737 000024 002500 MOV #24,GOOD ;SET UP DATA FOR ERROR MESSAGE
5337 041204 013737 002402 002502 MOV RSAVE,BAD ;...
5338 041212 ERRSOFT 1302,E901,ERR501 ;ERROR HANDLER

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 55-2  
 TEST 13: DEVICE CLEAR INTERFACE FUNCTION TEST

```

5368 041510 013737 002454 002500      MOV      CDAT9,GOOD      ;SET UP DATA FOR ERROR MESSAGE
5369 041516 013737 002402 002502      MOV      RSAVE,BAD      ;
5370 041524 104457 000000 000000      ERRSOFT 1306,E502,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    1306
                                .WORD    E502
                                .WORD    ERR501
5371 041534 104406 000000 000000      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
5372 041536 017737 140522 002402 24$:  MOV      @ICRX,RSAVE      ;GET ICR2 CONTENTS
5373 041544 122737 000004 002402      CMPB     #4,RSAVE        ;CONTENTS SHOULD BE 4
5374 041552 001413 000000 000000      BEQ      25$            ;BRANCH IF YES
5375 041554 012737 000004 002500      MOV      #4,GOOD        ;SET UP DATA FOR ERROR MESSAGES
5376 041562 013737 002402 002502      MOV      RSAVE,BAD      ;
5377 041570 104457 000000 000000      ERRSOFT 1307,E901,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    1307
                                .WORD    E901
                                .WORD    ERR501
5378 041600 104406 000000 000000      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
5379 041602 112777 000001 140460 25$:  MOVB     #1,@ICRHX        ;----LOAD NOT DACR INTO ACR2-----
5380 041610 042777 000010 140462      BIC     #10,@CSRX        ;SELECT CHANNEL 1
5381 041616 012737 000001 002374      MOV     #1,CHAN          ;LOAD CHANNEL NUMBER
5382 041624 112777 000077 140444      MOVB    #77,@IDRHX       ;----LOAD UNL INTO DOR 1-----
5383 041632 004737 011060 000000      JSR     PC,LOOP          ;WAIT A LITTLE
5384 041636 017737 140406 002402      MOV     @IIRX,RSAVE      ;GET IIR1 CONTENTS
5385 041644 022737 000020 002402      CMP     #20,RSAVE        ;BO BIT SHOULD BE SET
5386 041652 001413 000000 000000      BEQ     26$            ;BRANCH IF YES
5387 041654 012737 000020 002500      MOV     #20,GOOD        ;SET UP DATA FOR ERROR MESSAGE
5388 041662 013737 002402 002502      MOV     RSAVE,BAD      ;
5389 041670 104457 000000 000000      ERRSOFT 1308,E501,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    1308
                                .WORD    E501
                                .WORD    ERR501
5390 041700 104406 000000 000000      CKLOOP      ;BRANCH BACK TO BGNSEG IS ERRLOOP IS SET
                                TRAP      C$CLP1
5391 041702 052777 000010 140370 26$:  BIS     #10,@CSRX        ;SELECT CHANNEL 2
5392 041710 012737 000002 002374      MOV     #2,CHAN          ;LOAD CHANNEL NUMBER
5393 041716 017737 140326 002402      MOV     @IIRX,RSAVE      ;GET IIR2 CONTENTS
5394 041724 022737 000001 002402      CMP     #1,RSAVE        ;MAC BIT SHOULD BE SET
5395 041732 001413 000000 000000      BEQ     27$            ;BRANCH IF YES
5396 041734 012737 000001 002500      MOV     #1,GOOD        ;SET UP DATA FOR ERROR MESSAGE
5397 041742 013737 002402 002502      MOV     RSAVE,BAD      ;
5398 041750 104457 000000 000000      ERRSOFT 1309,E501,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    1309
                                .WORD    E501
                                .WORD    ERR501
5399 041760 104406 000000 000000      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
5400 041762 104405 000000 000000 27$:  ENDSEG
                                10000$:  TRAP    C$ESEG
5401 041762 104405
5402

```

-----  
 ;PART 2 CHECKS THE DEVICE CLEAR INTERFACE FUNCTION OF CHANNEL 2

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 55-3  
 TEST 13: DEVICE CLEAR INTERFACE FUNCTION TEST

```

5403 ;++++-----
5404 041764 004737 010220 JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
5405 041770 004737 010710 JSR PC,BGIN2 ;SET UP PARAMETER
5406 041774 032737 000001 002312 BIT #1,DPA1 ;IS DPA EVEN
5407 042002 001404 BEQ 4$ ;BRANCH IF YES
5408 042004 052737 000001 002454 BIS #1,CDAT9 ;SET ULPA BIT IN COMPARE DATA FOR ISR
5409 042012 000403 BR +10 ;
5410 042014 042737 000001 002454 4$: BIC #1,CDAT9 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
5411 042022 PSEU1: BGNSEG ;
                    TRAP C$BSEG
5412 042024 104404 052777 000010 140246 BIS #10,@CSRX ;SELECT CHANNEL 2
5413 042032 112777 000217 140230 MOVB #217,@ICRHX ;LOAD SIC INTO ACR 2
5414 042040 004737 011072 JSR PC,WAIT ;WAIT A LITTLE
5415 042044 112777 000017 140216 MOVB #17,@ICRHX ;LOAD NOT SIC IN ACR 2
5416 042052 042777 000010 140220 BIC #10,@CSRX ;SELECT CHANNEL 1
5417 042060 112777 000223 140202 MOVB #223,@ICRHX ;----LOAD DAI INTO ACR 1-----
5418 042066 112777 000010 140166 MOVB #10,@ISRHX ;---LOAD DCAS BIT INTO ISR1 REGISTER
5419 042074 052777 000010 140176 BIS #10,@CSRX ;SELECT CHANNEL 2
5420 042102 112777 000024 140166 MOVB #24,@IDRHX ;----LOAD DCL INTO DOR 2-----
5421 042110 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5422 042114 042777 000010 140156 BIC #10,@CSRX ;SELECT CHANNEL 1
5423 042122 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5424 042130 017737 140114 002402 MOV @IIRX,RSAVE ;GET IIR1 CONTENTS
5425 042136 022737 004500 002402 CMP #4500,RSAVE ;DCAS,IFC,INT1 BIT SHOULD BE SET
5426 042144 001413 BEQ 10$ ;BRANCH IF YES
5427 042146 012737 004500 002500 MOV #4500,GOOD ;SET UP DATA FOR ERROR MESSAGE
5428 042154 013737 002402 002502 MOV RSAVE,BAD ;
5429 042162 ERRSOF T 1310,E501,ERR501 ;ERROR HANDLER
                    TRAP C$ERSOF T
                    .WORD 1310
                    .WORD E501
                    .WORD ERR501
5430 042172 CKLOOP ;BRANCH TO BGNSEG IF ERLOOP IS SET
                    TRAP C$CLP1
5431 042174 017737 140064 002402 10$: MOV @ICRX,RSAVE ;GET ICR1 CONTENTS
5432 042202 122737 000024 002402 CMPB #24,RSAVE ;ICR1 CONTENTS SHOULD BE 24
5433 042210 001413 BEQ 20$ ;BRANCH IF YES
5434 042212 012737 000024 002500 MOV #24,GOOD ;SET UP DATA FOR ERROR MESSAGE
5435 042220 013737 002402 002502 MOV RSAVE,BAD ;
5436 042226 ERRSOF T 1312,E901,ERR501 ;ERROR HANDLER
                    TRAP C$ERSOF T
                    .WORD 1312
                    .WORD E901
                    .WORD ERR501
5437 042236 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET
                    TRAP C$CLP1
5438 042240 112777 000001 140022 20$: MOVB #1,@ICRHX ;----LOAD NOT DACR INTO ACR1-----
5439 042246 017737 137776 002402 MOV @IIRX,RSAVE ;GET IIR1 CONTENTS
5440 042254 005737 002402 TST RSAVE ;IIR1 CONTENTS SHOULD BE ZERO
5441 042260 001412 BEQ 21$ ;BRANCH IF YES
5442 042262 005037 002500 CLR GOOD ;SET UP DATA FOR ERROR MESSAGE
5443 042266 013737 002402 002502 MOV RSAVE,BAD ;
5444 042274 ERRSOF T 1313,E501,ERR501 ;ERROR HANDLER
                    TRAP C$ERSOF T
                    .WORD 1313
                    .WORD E501
                    .WORD ERR501
                    042274 104457
                    042276 002441
                    042300 005406
                    042302 003702
    
```







HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 55-5  
TEST 13: DEVICE CLEAR INTERFACE FUNCTION TEST

```

042610 005644
042612 003702
5484 042614 104406 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
042614 104406 ;.WORD E901
5485 042616 112777 000001 137444 25$: MOVB #1,@ICRHX ;.WORD ERR501
5486 042624 052777 000010 137446 BIS #10,@CSRX ;TRAP C$CLP1
5487 042632 012737 000002 002374 MOV #2,CHAN ;----LOAD NOT DACR INTO ACR1-----
5488 042640 112777 000077 137430 MOVB #77,@IDRHX ;SELECT CHANNEL 2
5489 042646 004737 011060 JSR PC,LOOP ;LOAD CHANNEL NUMBER
5490 042652 017737 137372 002502 MOV @IIRX,BAD ;----LOAD UNL INTO DOR 2-----
5491 042660 022737 000020 002502 CMP #20,BAD ;WAIT A LITTLE
5492 042666 001410 BEQ 26$ ;GET IIR2 CONTENTS
5493 042670 012737 000020 002500 MOV #20,GOOD ;BO BIT SHOULD BE SET
5494 042676 104457 ERRSOFT 1318,E501,ERR501 ;BRANCH IF YES
042676 104457 ;SET UP DATA FOR ERROR MESSAGE
042700 002446 ;ERROR HANDLER
042702 005406 TRAP C$ERSOFT
042704 003702 .WORD 1318
5495 042706 104406 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
042706 104406 ;.WORD E501
5496 042710 042777 000010 137362 26$: BIC #10,@CSRX ;SELECT CHANNEL 1
5497 042716 012737 000001 002374 MCV #1,CHAN ;LOAD CHANNEL NUMBER
5498 042724 017737 137320 002502 MOV @IIRX,BAD ;GET IIR1 CONTENTS
5499 042732 022737 000001 002502 CMP #1,BAD ;MAC BIT SHOULD BE SET
5500 042740 001410 BEQ 27$ ;BRANCH IF YES
5501 042742 012737 000001 002500 MOV #1,GOOD ;SET UP DATA FOR ERROR MESSAGE
5502 042750 104457 ERRSOFT 1319,E501,ERR501 ;ERROR HANDLER
042750 104457 TRAP C$ERSOFT
042752 002447 .WORD 1319
042754 005406 .WORD E501
042756 003702 .WORD ERR501
5503 042760 104406 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
042760 104406 ;TRAP C$CLP1
5504 042762 27$: ENDSEG ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
042762 104405 10001$: TRAP C$ESEG
5505 042764 052777 000010 137306 BIS #10,@CSRX ;SELECT CHANNEL 1
5506 042772 112777 000200 137270 MOVB #200,@ICRHX ;LOAD SWRST INTO ACR2
5507 043000 112777 000000 137262 MOVB #0,@ICRHX ;LOAD NOT SWRST INTO ACR2
5508 043006 005737 002234 TST QVP ;IS QUICK VERIFY PASS SELECTED
5509 043012 001010 BNE EXQV13 ;IF YES EXIT TEST
5510 043014 005237 002322 INC ITRCNT ;ITERATION COUNTER + 1
5511 043020 023737 002322 002320 CMP ITRCNT,ITRDEF ;DEFAULT ITERATION EXECUTED
5512 043026 001402 BEQ EXQV13 ;IF YES EXIT TEST
5513 043030 000137 041002 JMP ITAC13 ;IF NO TEST ITERATION
5514 043034 104432 EXQV13: EXIT TST ;
043034 104432 TRAP C$EXIT
043036 000056 .WORD L10042-
5515
5516
5517 043040 045 123 062 TSHD13: .NLIST BEX
5518 .ASCIZ /%S2%ADEVICE CLEAR INTERFACE FUNCTION TEST%/
5519 .LIST BEX
5520 043114 .EVEN
043114 .ENDTST
043114 104401 L10042: TRAP C$ETST

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 56  
TEST 14: DEVICE TRIGGER INTERFACE FUNCTION TEST

```

5522 .SBTTL TEST 14: DEVICE TRIGGER INTERFACE FUNCTION TEST
5523 :*****
5524 : IEX - TEST 14
5525 : PART 1 CHECKS THE TRIGGER INTERFACE FUNCTION OF CHANNEL 2 BY MEANS OF
5526 : RECEIVING THE ADDRESS COMMAND GET AS WELL AS THE AUXILIARY COMMAND
5527 : NOT FGET.
5528 : PART 2 CHECKS THE TRIGGER INTERFACE FUNCTION OF CHANNEL 1 BY MEANS OF
5529 : RECEIVING THE ADDRESS COMMAND GET AS WELL AS THE AUXILIARY COMMAND
5530 : NOT FGET.
5531 :*****
5532 043116 BGNTST
5533 043116 005737 002324 TST PNTF ;IS THE PNT FLAG SET
5534 043122 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
5535 043124 PRINTF #TSHD14 ;...
043124 012746 043742 MOV #TSHD14,-(SP)
043130 012746 000001 MOV #1,-(SP)
043134 010600 MOV SP,R0
043136 104417 TRAP C$PNTF
043140 062706 000004 ADD #4,SP
5536 043144 005037 002322 7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
5537 043150 004737 010534 ITAC14: JSR PC,BGIN1 ;SET UP PARAMETER
5538 043154 BGNSEG
043154 104404 TRAP C$BSEG
5539 043156 052777 000010 137114 BIS #10,@CSRX ;SELECT CHANNEL 2
5540 043164 112777 000223 137076 MOVB #223,@ICRHX ;----LOAD DAI INTO ACR 2-----
5541 043172 112777 000200 137062 MOVB #200,@ISRHX ;----LOAD GET BIT INTO ISR2-----
5542 043200 042777 000010 137072 BIC #10,@CSRX ;SELECT CHANNEL 1
5543 043206 013701 002314 MOV DPA2,R1 ;CREATE MLA2
5544 043212 062701 000040 ADD #40,R1
5545 043216 010137 002412 MOV R1,MLA2 ;STORE MLA2
5546 043222 113777 002412 137046 MOVB MLA2,@IDRHX ;----LOAD MLA2 INTO DOR 1-----
5547 043230 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5548 043234 112777 000010 137034 MOVB #10,@IDRHX ;----LOAD GET INTO DOR 1-----
5549 043242 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5550 043246 052777 000010 137024 BIS #10,@CSRX ;SELECT CHANNEL 2
5551 043254 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
5552 043262 017737 136762 002402 MOV @IIRX,RSAVE ;GET IIR 2 CONTENTS
5553 043270 022737 102501 002402 CMP #102501,RSAVE ;GET,MA,INT!,IFC,MAC BIT SHOULD BE SET
5554 043276 001413 BEQ 10$ ;BRANCH IF YES
5555 043300 012737 102501 002500 MOV #102501,GOOD ;SET UP DATA FOR ERROR MESSAGE
5556 043306 013737 002402 002502 MOV RSAVE,BAD
5557 043314 ERRSOFT 1401,E501,ERR501 ;ERROR HANDLER
043314 104457 TRAP C$ERRSOFT
043316 002571 .WORD 1401
043320 005406 .WORD E501
043322 003702 .WORD ERR501
5558 043324 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
043324 104406 TRAP C$CLP1
5559 043326 017737 136732 002402 10$: MOV @ICRX,RSAVE ;GET ICR2 CONTENTS
5560 043334 122737 000010 002402 CMPB #10,RSAVE ;ICR2 CONTENTS SHOULD BE 10
5561 043342 001413 BEQ 20$ ;BRANCH IF YES
5562 043344 012737 000010 002500 MOV #10,GOOD ;SET UP DATA FOR ERROR MESSAGE
5563 043352 013737 002402 002502 MOV RSAVE,BAD
5564 043360 ERRSOFT 1402,E901,ERR501 ;ERROR HANDLER
043360 104457 TRAP C$ERRSOFT
043362 002572 .WORD 1402

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 56-1  
 TEST 14: DEVICE TRIGGER INTERFACE FUNCTION TEST

```

043364 005644
043366 003702
5565 043370 104406 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                .WORD E901
                                .WORD ERR501
                                TRAP C$CLP1
5566 043372 112777 000006 136670 20$: MOV B #6,@ICRHX ;-----LOAD NOT FGET INTO ACR 2-----
5567 043400 112777 000001 136662 MOV B #1,@ICRHX ;-----LOAD NOT DACR INTO ACR2-----
5568 043406 042777 000010 136664 BIC #10,@CSRX ;SELECT CHANNEL 1
5569 043414 112777 000077 136654 MOV B #77,@IDRHX ;-----LOAD UNL INTO DOR 1-----
5570 043422 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5571 043426
                                10000$:
                                TRAP C$ESEG
                                043426 104405
5572
5573
5574
5575
5576 043430 004737 010710 JSR PC,BGIN2 ;SET UP PARAMETER
5577 043434 BGNSEG
                                TRAP C$BSEG
043434 104404
5578 043436 042777 000010 136634 BIC #10,@CSRX ;SELECT CHANNEL 1
5579 043444 112777 000223 136616 MOV B #223,@ICRHX ;-----LOAD DAI INTO ACR 1-----
5580 043452 112777 000200 136602 MOV B #200,@ISRHX ;-----LOAD GET BIT INTO ISR1-----
5581 043460 052777 000010 136612 BIS #10,@CSRX ;SELECT CHANNEL 2
5582 043466 013701 002312 MOV DPA1,R1 ;CREATE MLA1
5583 043472 062701 000040 ADD #40,R1
5584 043476 010137 002410 MOV R1,MLA1 ;STORE MLA1
5585 043502 113777 002410 136566 MOV B MLA1,@IDRHX ;-----LOAD MLA1 INTO DOR 2-----
5586 043510 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5587 043514 112777 000010 136554 MOV B #10,@IDRHX ;-----LOAD GET INTO DOR 2-----
5588 043522 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5589 043526 042777 000010 136544 BIC #10,@CSRX ;SELECT CHANNEL 1
5590 043534 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5591 043542 017737 136502 002402 MOV @IIRX,RSAVE ;GET IIR 1 CONTENTS
5592 043550 022737 102501 002402 CMP #102501,RSAVE ;GET,MA,INT1,IFC,MAC BIT SHUOLD BE SET
5593 043556 001413 BEQ 30$ ;BRANCH IF YES
5594 043560 012737 102501 002500 MOV #102501,GOOD ;SET UP DATA FOR ERROR MESSAGE
5595 043566 013737 002402 002502 MOV RSAVE,BAD
5596 043574 ERRSOFT 1403,E501,ERR501 ;ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 1403
                                .WORD E501
                                .WORD ERR501
043574 104457
043576 002573
043600 005406
043602 003702
5597 043604 104406 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
043604 104406
5598 043606 017737 136452 002402 30$: MOV @ICRX,RSAVE ;GET ICR1 CONTENTS
5599 043614 122737 000010 002402 CMPB #10,RSAVE ;ICR1 CONTENTS SHOULD BE 10
5600 043622 001413 BEQ 40$ ;BRANCH IF YES
5601 043624 012737 000010 002500 MOV #10,GOOD ;SET UP DATA FOR ERROR MESSAGE
5602 043632 013737 002402 002502 MOV RSAVE,BAD
5603 043640 ERRSOFT 1404,E901,ERR501 ;ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 1404
                                .WORD E901
                                .WORD ERR501
043640 104457
043642 002574
043644 005644
043646 003702
5604 043650 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP C$CLP1
043650 104406
5605 043652 112777 000006 136410 40$: MOV B #6,@ICRHX ;-----LOAD NOT FGET INTO ACR 1-----

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 57  
TEST 15: INCOMPLETE SOURCE HANDSHAKE TEST

```

5625 .SBTTL TEST 15: INCOMPLETE SOURCE HANDSHAKE TEST
5626 *****
5627 IEX - TEST 15
5628 :PART 1 CHECKS THE INCOMPLETE SOURCE HANDSHAKE OF CHANNEL 1. SOURCE HANDSHAKE
5629 DOES NOT OCCUR DURING THE DATA TRANSFER, BECAUSE CHANNEL 2 IS NOT
5630 SELECTED AS LISTENER.
5631 :PART 2 CHECKS THE INCOMPLETE SOURCE HANDSHAKE OF CHANNEL 2. SOURCE HANDSHAKE
5632 DOES NOT OCCUR DURING THE DATA TRANSFER, BECAUSE CHANNEL 1 IS NOT
5633 SELECTED AS LISTENER.
5634 *****
5635 BGNTST
                                T15::
5636 044022 005737 002324      TST      PNTF      ;IS THE PNT FLAG SET
5637 044026 001410          BEQ      7$      ;IF YES, PRINT THE TEST HEADER
5638 044030          PRINTF #TSHD15 ;....
                                MOV      #TSHD15,-(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP    C$PNTF
                                ADD     #4,SP
5639 044050 005037 002322      7$: CLR      ITRCNT ;CLEAR ITERATION COUNTER
5640 044054 004737 010220      ITAC15: JSR    PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
5641 044060 004737 010534      JSR    PC,BGIN1 ;SET UP PARAMETER
5642 044064          BGNSEG          TRAP    C$BSEG
5643 044066 042777 000010 136204      BIC     #10,@CSRX ;SELECT CHANNEL 1
5644 044074 012737 000001 002374      MOV     #1,CHAN ;LOAD CHANNEL NUMBER
5645 044102 112777 000212 136160      MOV    #212,@ICRHX ;----LOAD TON INTO ACR 1-----
5646 044110 012703 000002          MOV     #2,R3 ;GET A NONE EXISTENTS MLA
5647 044114 020337 002314          CMP     R3,DPA2 ;....
5648 044120 001410          BEQ     10$ ;....
5649 044122 020337 002312          CMP     R3,DPA1 ;....
5650 044126 001405          BEQ     10$ ;....
5651 044130 062703 000040          ADD     #40,R3 ;CREATE MLA
5652 044134 010337 002412          MOV     R3,MLA2 ;
5653 044140 000402          BR     20$ ;
5654 044142 005203          10$: INC     R3 ;
5655 044144 000763          BR     11$ ;
5656 044146 113777 002412 136122      20$: MOV    MLA2,@IDRHX ;----LOAD MLA2 INTO DOR1-----
5657 044154 004737 011060          JSR    PC,LOOP ;WAIT A LITTLE
5658 044160 112777 000013 136102      MOV    #13,@ICRHX ;----LOAD GTS INTO ACR 1-----
5659 044166 112777 000125 136102      MOV    #125,@IDRHX ;----LOAD DATA PATTERN INTO DOR 1----
5660 044174 004737 011060          JSR    PC,LOOP ;WAIT A LITTLE
5661 044200 017737 136044 002402      MOV    @IIRX,RSAVE ;GET IIR1 CONTENTS
5662 044206 022737 040000 002402      CMP    #40000,RSAVE ;ERR BIT IN IIR REGISTER SHOULD BE SET
5663 044214 001413          BEQ    30$ ;BRANCH IF YES
5664 044216 012737 040000 002500      MOV    #40000,GOOD ;SET UP DATA FOR ERROR MESSAGE
5665 044224 013737 002402 002502      MOV    RSAVE,BAD ;
5666 044232          ERRSOFT 1501,E501,ERR501 ;ERROR HANDLER
                                TRAP    C$ERRSOFT
                                .WORD   1501
                                .WORD   E501
                                .WORD   ERR501
5667 044242          CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP    C$CLP1
5668 044244 017737 136006 002402      30$: MOV    @ISR1,RSAVE ;GET ISR1 CONTENTS
5669 044252 022737 000002 002402      CMP    #2,RSAVE ;TADS BIT SHOULD BE SET

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 57-1  
 TEST 15: INCOMPLETE SOURCE HANDSHAKE TEST

WA  
TE

```

5670 044260 001413          BEQ      40$          ;BRANCH IF YES
5671 044262 012737 000002 002500  MOV      #2,GOOD      ;SET UP DATA FOR ERROR MESSAGE
5672 044270 013737 002402 002502  MOV      RSAVE,BAD    ;
5673 044276          ERRSOFT 1502,E502,ERR501 ;ERROR HANDLER
          044276 104457          TRAP    CSERSOFT
          044300 002736          .WORD  1502
          044302 005447          .WORD  E502
          044304 003702          .WORD  ERR501
5674 044306          CKLOOP          ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
          044306 104406          TRAP    C$CLP1
5675 044310 052777 000010 135762 40$:  BIS      #10,@CSRX    ;SELECT CHANNEL 2
5676 044316 012737 000002 002374  MOV      #2,CHAN      ;LOAD CHANNEL NUMBER
5677 044324 017737 135726 002402  MOV      @ISR,RSAVE   ;GET ISR2 CONTENTS
5678 044332 005737 002402  TST      RSAVE        ;ISR REGISTER SHOULD BE ZERO
5679 044336 001412  BEQ      50$          ;BRANCH IF YES
5680 044340 005037 002500  CLR      GOOD         ;SET UP DATA FOR ERROR MESSAGE
5681 044344 013737 002402 002502  MOV      RSAVE,BAD    ;
5682 044352          ERRSOFT 1503,E502,ERR501 ;ERROR HANDLER
          044352 104457          TRAP    CSERSOFT
          044354 002737          .WORD  1503
          044356 005447          .WORD  E502
          044360 003702          .WORD  ERR501
5683 044362          CKLOOP          ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
          044362 104406          TRAP    C$CLP1
5684 044364          50$:  ENDSEG          10000$: TRAP    C$ESEG
          044364          104405
5685          ;+-----+
5686          ;PART 2 OF THE TEST
5687          ;THIS PART CHECKS THE INCOMPLETE SOURCE HANDSHAKE OF CHANNEL 2
5688          ;+-----+
5689 044366 004737 010220  JSR      PC,CULPA     ;CLEAR ULPA BIT IN ISR 1 AND 2
5690 044372 004737 010710  JSR      PC,BGIN2     ;SET UP PARAMETER
5691 044376          PSEU5:  BGNSEG          TRAP    C$BSEG
          044376 104404
5692 044400 052777 000010 135672  BIS      #10,@CSRX    ;SELECT CHANNEL 2
5693 044406 112777 000212 135654  MOVB     #212,@ICRHX  ;-----LOAD TON INTO ACR 2-----
5694 044414 012703 000002  MOV      #2,R3        ;GET A NONE EXISTENTS MLA
5695 044420 020337 002314 11$:  CMP      R3,DPA2     ;
5696 044424 001410  BEQ      10$          ;
5697 044426 020337 002312  CMP      R3,DPA1     ;
5698 044432 001405  BEQ      10$          ;
5699 044434 062703 000040  ADD      #40,R3       ;CREATE MLA
5700 044440 010337 002410  MOV      R3,MLA1     ;
5701 044444 000402  BR       20$          ;
5702 044446 005203 10$:  INC      R3           ;
5703 044450 000763  BR       11$          ;
5704 044452 113777 002410 135616 20$:  MOVB     MLA1,@IDRHX  ;-----LOAD MLA1 INTO DOR2-----
5705 044460 004737 011060  JSR      PC,LOOP     ;WAIT A LITTLE
5706 044464 112777 000013 135576  MOVB     #13,@ICRHX  ;-----LOAD GTS INTO ACR 2-----
5707 044472 112777 000125 135576  MOVB     #125,@IDRHX ;-----LOAD DATA PATTERN INTO DOR 2-----
5708 044500 004737 011060  JSR      PC,LOOP     ;WAIT A LITTLE
5709 044504 017737 135540 002402  MOV      @IIR,RSAVE   ;GET IIR2 CONTENTS
5710 044512 022737 040000 002402  CMP      #40000,RSAVE ;ERR BIT IN IIR REGISTER SHOULD BE SET
5711 044520 001413  BEQ      30$          ;BRANCH IF YES
5712 044522 012737 040000 002500  MOV      #40000,GOOD  ;SET UP DATA FOR ERROR MESSAGE
5713 044530 013737 002402 002502  MOV      RSAVE,BAD    ;
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 57-2  
 TEST 15: INCOMPLETE SOURCE HANDSHAKE TEST

```

5714 044536 ERRSOFT 1504,E501,ERR501 ;ERROR HANDLER
      044536 104457 TRAP C$ERSOFT
      044540 002740 .WORD 1504
      044542 005406 .WORD E501
      044544 003702 .WORD ERR501
5715 044546 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      044546 104406 TRAP C$CLP1
5716 044550 017737 135502 002402 30$: MOV @ISRX,RSAVE ;GET ISR2 CONTENTS
5717 044556 022737 000002 002402 CMP #2,RSAVE ;TADS BIT SHOULD BE SET
5718 044564 001413 BEQ 40$ ;BRANCH IF YES
5719 044566 012737 000002 002500 MOV #2,GOOD ;SET UP DATA FOR ERROR MESSAGE
5720 044574 013737 002402 002502 MOV RSAVE,BAD
5721 044602 ERRSOFT 1505,E502,ERR501 ;ERROR HANDLER
      044602 104457 TRAP C$ERSOFT
      044604 002741 .WORD 1505
      044606 005447 .WORD E502
      044610 003702 .WORD ERR501
5722 044612 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      044612 104406 TRAP C$CLP1
5723 044614 042777 000010 135456 40$: BIC #10,@CSRX ;SELECT CHANNEL 1
5724 044622 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
5725 044630 017737 135422 002402 MOV @ISRX,RSAVE ;GET ISR1 CONTENTS
5726 044636 005737 002402 TST RSAVE ;ISR REGISTER SHOULD BE ZERO
5727 044642 001412 BEQ 50$ ;BRANCH IF YES
5728 044644 005037 002500 CLR GOOD ;SET UP DATA FOR ERROR MESSAGE
5729 044650 013737 002402 002502 MOV RSAVE,BAD
5730 044656 ERRSOFT 1506,E502,ERR501 ;ERROR HANDLER
      044656 104457 TRAP C$ERSOFT
      044660 002742 .WORD 1506
      044662 005447 .WORD E502
      044664 003702 .WORD ERR501
5731 044666 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      044666 104406 TRAP C$CLP1
5732 044670 50$: ENDSEG
      044670 10001$: TRAP C$ESEG
5733 044672 005737 002234 TST QVP ;IS QUICK VERIFY PASS SELECTED
5734 044676 001010 BNE EXQV15 ;IF YES EXIT TEST
5735 044700 005237 002322 INC ITRCNT ;ITERATION COUNTER +1
5736 044704 023737 002322 002320 CMP ITRCNT,ITRDEF ;DEFAULT ITERATION EXECUTED
5737 044712 001402 BEQ EXQV15 ;IF YES EXIT TEST
5738 044714 000137 044054 JMP ITAC15 ;IF NO TEST ITERATION
5739 044720 EXQV15: EXIT TST
      044720 104432 TRAP C$EXIT
      044722 000052 .WORD L10044-.
5740
5741
5742 044724 045 123 062 TSHD15: .NLIST BEX
      .ASCIZ /%S2%AINCOMPLETE SOURCE HANDSHAKE TEST%/
5743 .LIST BEX
5744 .EVEN
5745 044774 .ENDTST
      044774
      044774 104401 L10044: TRAP C$ETST
    
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 58  
TEST 16: CHANGING OF THE CONTROLLER CONFIGURATION

```

5747 .SBTTL TEST 16: CHANGING OF THE CONTROLLER CONFIGURATION
5748 .....
5749 IEX - TEST 16
5750 :PART 1 CHECKS THE CHANGING OF THE CONTROLLER CONFIGURATION FROM 1 TO 2
5751 BY MEANS OF THE AUXILIARY COMMANDS RQC AND RLC.
5752 :PART 2 CHECKS THE CHANGING OF THE CONTROLLER CONFIGURATION FROM 2 TO 1
5753 BY MEANS OF THE AUXILIARY COMMANDS RQC AND RLC.
5754 .....
5755 BGNTST
                    T16::
5756 044776 005737 002324      TST      PNTF      ;IS THE PNT FLAG SET
5757 045002 001410          BEQ      7$      ;IF YES, PRINT THE TEST HEADER
5758 045004          PRINTF #TSHD16 ;...
                    MOV      #TSHD16,-(SP)
                    MOV      #1,-(SP)
                    MOV      SP,R0
                    TRAP     C$PNTF
                    ADD      #4,SP
5759 045024 005037 002322      7$: CLR      ITRCNT ;CLEAR ITERATION COUNTER
5760 045030          ITAC16: BGNSEG
                    TRAP     C$BSEG
5761 045032 004737 010220      JSR      PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
5762 045036 004737 010534      JSR      PC,BGIN1 ;SET UP PARAMETER
5763 045042 032737 000001 002314  BIT      #1,DPA2 ;IS DPA EVEN
5764 045050 001404          BEQ      3$      ;BRANCH IF YES
5765 045052 052737 000001 002456  BIS      #1,CDAT10 ;SET ULPA BIT IN COMPARE DATA FOR ISR
5766 045060 000403          BR      +10 ;BRANCH OVER NEXT INSTRUCTION
5767 045062 042737 000001 002456  3$: BIC      #1,CDAT10 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
5768 045070 052777 000010 135202  BIS      #10,@CSRX ;SELECT CHANNEL 2
5769 045076 112777 000223 135164  MOVB     #223,@ICRHX ;-----LOAD DAI INTO ACR 2-----
5770 045104 112777 000040 135150  MOVB     #40,@ISRHX ;-----SET UCG BIT IN ISR 2-----
5771 045112 042777 000010 135160  BIC      #10,@CSRX ;SELECT CHANNEL 1
5772 045120 013701 002314          MOV      DPA2,R1 ;CREATE MTA2
5773 045124 062701 000100          ADD      #100,R1
5774 045130 010137 002416          MOV      R1,MTA2 ;STORE MTA2
5775 045134 113777 002416 135134  MOVB     MTA2,@IDRHX ;-----LOAD MTA2 INTO DOR 1-----
5776 045142 004737 011060          JSR      PC,LOOP ;WAIT A LITTLE
5777 045146 112777 000011 135122  MOVB     #11,@IDRHX ;-----LOAD TCT INTO DOR 1-----
5778 045154 004737 011060          JSR      PC,LOOP ;WAIT A LITTLE
5779 045160 052777 000010 135112  BIS      #10,@CSRX ;SELECT CHANNEL 2
5780 045166 012737 000002 002374  MOV      #2,CHAN ;LOAD CHANNEL NUMBER
5781 045174 017737 135050 002402  MOV      @IIRX,RSAVE ;GET IIR2 CONTENTS
5782 045202 022737 022501 002402  CMP      #22501,RSAVE ;UCG,MA,IFC,INT1,MAC BIT SHOULD BE SET
5783 045210 001413          BEQ      10$     ;BRANCH IF YES
5784 045212 013737 002402 002502  MOV      RSAVE,BAD ;SET UP DATA FOR ERROR MESSAGE
5785 045220 012737 022501 002500  MOV      #22501,GOOD
5786 045226          ERRSOFT 1601,E501,ERR501 ;ERROR HANDLER
                    TRAP     C$ERRSOFT
                    .WORD    1601
                    .WORD    E501
                    .WORD    ERR501
5787 045236          CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                    TRAP     C$CLP1
5788 045240 017737 135012 002502  10$: MOV      @ISRX,BAD ;GET ISR2 CONTENTS
5789 045246 023737 002456 002502  CMP      CDAT10,BAD ;ATN,NDAC,DAV,NRFD,ATN,TADS,(ULPA) IS SET
5790 045254 001410          BEQ      20$     ;BRANCH IF YES
5791 045256 013737 002456 002500  MOV      CDAT10,GOOD ;SET UP DATA FOR ERROR MESSAGE

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 58-1  
 TEST 16: CHANGING OF THE CONTROLLER CONFIGURATION

```

5792 045264          ERRSOF T 1602,E502,ERR501          :ERROR HANDLER
      045264 104457
      045266 003102
      045270 005447
      045272 003702
5793 045274          CKLOOP
      045274 104406
5794 045276 017737 134762 002502 20$:  MOV    @ICRX,BAD          :GET ICR2 CONTENTS
5795 045304 122737 000011 002502      CMPB   #11,BAD          :ICR CONTENTS SHOULD BE 11
5796 045312 001407
5797 045314 012737 000011 002500      BEQ   30$              :BRANCH IF YES
5798 045322          ERRSOF T 1603,E901,ERR501          :SET UP DATA FOR ERROR MESSAGE
      045322 104457
      045324 003103
      045326 005644
      045330 003702
5799 045332 112777 000021 134730 30$:  MOVB   #21,@ICRHX        :-----LOAD RGC INTO ACR 2-----
5800 045340 112777 000001 134722      MOVB   #1,@ICRHX        :-----LOAD NOT DACR INTO ACR 2-----
5801 045346 042777 000010 134724      BIC   #10,@CSRX         :SELECT CHANNEL 1
5802 045354 112777 000022 134706      MOVB   #22,@ICRHX        :-----LOAD RLC INTO ACR 1-----
5803 045362 052777 000010 134710      BIS   #10,@CSRX         :SELECT CHANNEL 2
5804 045370 012737 000002 002374      MOV   #2,CHAN           :LOAD CHANNEL NUMBER
5805 045376 017737 134646 002502      MOV   @IIRX,BAD         :GET IIR2 CONTENTS
5806 045404 022737 000020 002502      CMP   #20,BAD           :BO BIT SHOULD BE SET
5807 045412 001410
5808 045414 012737 000020 002500      BEQ   33$              :BRANCH IF YES
5809 045422          ERRSOF T 1604,E501,ERR501          :SET UP DATA FOR ERROR MESSAGE
      045422 104457
      045424 003104
      045426 005406
      045430 003702
5810 045432          CKLOOP
      045432 104406
5811 045434 042777 000010 134636 33$:  BIC   #10,@CSRX         :SELECT CHANNEL 1
5812 045442 012737 000001 002374      MOV   #1,CHAN           :LOAD CHANNEL NUMBER
5813 045450 017737 134574 002502      MOV   @IIRX,BAD         :GET IIR1 CONTENTS
5814 045456 022737 000020 002502      CMP   #20,BAD           :BO BIT SHOULD BE SET
5815 045464 001410
5816 045466 012737 000020 002500      BEQ   40$              :BRANCH IF YES
5817 045474          ERRSOF T 1605,E501,ERR501          :SET UP DATA FOR ERROR MESSAGE
      045474 104457
      045476 003105
      045500 005406
      045502 003702
5818 045504          CKLOOP
      045504 104406
5819 045506 052777 000010 134564 40$:  BIS   #10,@CSRX         :SELECT CHANNEL 2
5820 045514 112777 000137 134554      MOVB   #137,@IDRHX       :-----LOAD UNT INTO DOR 2-----
5821 045522 004737 011060
5822 045526          ENDSEG
      045526 104405
5823
5824
5825
5826 045530          PSEU16: BGNSEG
      045530 104404
                                     10000$: TRAP   C$ESEG
                                     :++++-----
                                     :PART 2 CHECKS THE CHANGING OF THE CONTROLLER CONFIGURATION FROM 2 TO 1
                                     :++++-----
                                     TRAP   C$BSEG

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 58-2  
 TEST 16: CHANGING OF THE CONTROLLER CONFIGURATION

5827	045532	004737	010220			JSR	PC,CULPA		:CLEAR ULPA BIT IN ISR 1 AND 2
5828	045536	004737	010710			JSR	PC,BGIN2		:SET UP PARAMETER
5829	045542	032737	000001	002312		BIT	#1,DPA1		:IS DPA EVEN
5830	045550	001404				BEQ	3%		:BRANCH IF YES
5831	045552	052737	000001	002456		BIS	#1,CDAT10		:SET ULPA BIT IN COMPARE DATA FOR ISR
5832	045560	000403				BR	+10		:BRANCH OVER NEXT INSTRUCTION
5833	045562	042737	000001	002456	3%:	BIC	#1,CDAT10		:CLEAR ULPA BIT IN COMPARE DATA FOR ISR
5834	045570	042777	000010	134502		BIC	#10,@CSRX		:SELECT CHANNEL 1
5835	045576	112777	000223	134464		MOVB	#223,@ICRHX		:-----LOAD DAI INTO ACR 1-----
5836	045604	112777	000040	134450		MOVB	#40,@ISRHX		:-----SET UCG BIT IN ISR 1-----
5837	045612	013701	002312			MOV	DPA1,R1		:CREATE MTA1
5838	045616	062701	000100			ADD	#100,R1		:STORE MTA1
5839	045622	010137	002414			MOV	R1,MTA1		:SELECT CHANNEL 2
5840	045626	052777	000010	134444		BIS	#10,@CSRX		:-----LOAD MTA1 INTO DOR 2-----
5841	045634	113777	002414	134434		MOVB	MTA1,@IDRHX		:WAIT A LITTLE
5842	045642	004737	011060			JSR	PC,LOOP		:-----LOAD TCT INTO DOR 2-----
5843	045646	112777	000011	134422		MOVB	#11,@IDRHX		:WAIT A LITTLE
5844	045654	004737	011060			JSR	PC,LOOP		:SELECT CHANNEL 1
5845	045660	042777	000010	134412		BIC	#10,@CSRX		:LOAD CHANNEL NUMBER
5846	045666	012737	000001	002374		MOV	#1,CHAN		:GET IIR1 CONTENTS
5847	045674	017737	134350	002402		MOV	@IIRX,RSAVE		:UCG,MA,IFC,INT1,MAC BIT SHOULD BE SET
5848	045702	022737	022501	002402		CMP	#22501,RSAVE		:BRANCH IF YES
5849	045710	001413				BEQ	10%		:SET UP DATA FOR ERROR MESSAGE
5850	045712	013737	002402	002502		MOV	RSAVE,BAD		:ERROR HANDLER
5851	045720	012737	022501	002500		MOV	#22501,GOOD		
5852	045726					ERRSOFT	1606,E501,ERR501		
	045726	104457							TRAP C\$ERSOFT
	045730	003106							.WORD 1606
	045732	005406							.WORD E501
	045734	003702							.WORD ERR501
5853	045736					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	045736	104406							TRAP C\$CLP1
5854	045740	017737	134312	002502	10%:	MOV	@ISRX,BAD		:GET ISR1 CONTENTS
5855	045746	023737	002456	002502		CMP	CDAT10,BAD		:ATN,NDAC,DAV,NRFD,ATN,TADS,(ULPA) IS SET
5856	045754	001410				BEQ	20%		:BRANCH IF YES
5857	045756	013737	002456	002500		MOV	CDAT10,GOOD		:SET UP DATA FOR ERROR MESSAGE
5858	045764					ERRSOFT	1607,E502,ERR501		:ERROR HANDLER
	045764	104457							TRAP C\$ERSOFT
	045766	003107							.WORD 1607
	045770	005447							.WORD E502
	045772	003702							.WORD ERR501
5859	045774					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	045774	104406							TRAP C\$CLP1
5860	045776	017737	134262	002502	20%:	MOV	@ICRX,BAD		:GET ICR1 CONTENTS
5861	046004	122737	000011	002502		CMPB	#11,BAD		:ICR CONTENTS SHOULD BE 11
5862	046012	001407				BEQ	30%		:BRANCH IF YES
5863	046014	012737	000011	002500		MOV	#11,GOOD		:SET UP DATA FOR ERROR MESSAGE
5864	046022					ERRSOFT	1608,E901,ERR501		:ERROR HANDLER
	046022	104457							TRAP C\$ERSOFT
	046024	003110							.WORD 1608
	046026	005644							.WORD E901
	046030	003702							.WORD ERR501
5865	046032	112777	000021	134230	30%:	MOVB	#21,@ICRHX		:-----LOAD RGC INTO ACR 1-----
5866	046040	112777	000001	134222		MOVB	#1,@ICRHX		:-----LOAD NOT DACR INTO ACR 1-----
5867	046046	052777	000010	134224		BIS	#10,@CSRX		:SELECT CHANNEL 2
5868	046054	112777	000022	134206		MOVB	#22,@ICRHX		:-----LOAD RLC INTO ACR 2-----
5869	046062	042777	000010	134210		BIC	#10,@CSRX		:SELECT CHANNEL 1

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 58-3  
 TEST 16: CHANGING OF THE CONTROLLER CONFIGURATION

```

5870 046070 012737 000001 002374      MOV      #1,CHAN      ;LOAD CHANNEL NUMBER
5871 046076 017737 134146 002502      MOV      @1IRX,BAD   ;GET IIR1 CONTENTS
5872 046104 022737 000020 002502      CMP      #20,BAD     ;BO BIT SHOULD BE SET
5873 046112 001410                BEQ      33$         ;BRANCH IF YES
5874 046114 012737 000020 002500      MOV      #20,GOOD    ;SET UP DATA FOR ERROR MESSAGE
5875 046122                ERRSOFT 1609,E501,ERR501 ;ERROR HANDLER
                    046122 104457                TRAP      C$ERSOFT
                    046124 003111                .WORD    1609
                    046126 005406                .WORD    E501
                    046130 003702                .WORD    ERR501
5876 046132                CKLOOP          ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                    046132 104406                TRAP      C$CLP1
5877 046134 052777 000010 134136 33$:  BIS      #10,@CSRX   ;SELECT CHANNEL 2
5878 046142 012737 000002 002374      MOV      #2,CHAN     ;LOAD CHANNEL NUMBER
5879 046150 017737 134074 002502      MOV      @1IRX,BAD   ;GET IIR2 CONTENTS
5880 046156 022737 000020 002502      CMP      #20,BAD     ;BO BIT SHOULD BE SET
5881 046164 001410                BEQ      40$         ;BRANCH IF YES
5882 046166 012737 000020 002500      MOV      #20,GOOD    ;SET UP DATA FOR ERROR MESSAGE
5883 046174                ERRSOFT 16010,E501,ERR501 ;ERROR HANDLER
                    046174 104457                TRAP      C$ERSOFT
                    046176 037212                .WORD    16010
                    046200 005406                .WORD    E501
                    046202 003702                .WORD    ERR501
5884 046204                CKLOOP          ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                    046204 104406                TRAP      C$CLP1
5885 046206 042777 000010 134064 40$:  BIC      #10,@CSRX   ;SELECT CHANNEL 1
5886 046214 112777 000137 134054      MOV      #137,@IDRHX ;-----LOAD UNT INTO DOR 1-----
5887 046222 004737 011060                JSR      PC,LOOP     ;WAIT A LITTLE
5888 046226                ENDSEG
                    046226 104405                10001$: TRAP      C$ESEG
5889 046230 005737 002234                TST      QVP         ;IS QUICK VERIFY PASS SELECTED
5890 046234 001010                BNE      EXQV16      ;IF YES EXIT TEST
5891 046236 005237 002322                INC      ITRCNT      ;ITERATION COUNTER +1
5892 046242 023737 002322 002320      CMP      ITRCNT,ITRDEF ;DEFAULT ITERATION EXECUTED
5893 046250 001402                BEQ      EXQV16      ;IF YES EXIT TEST
5894 046252 000137 045030                JMP      ITAC16      ;IF NO TEST ITERATION
5895 046256                EXQV16: EXIT      TST
                    046256 104432                TRAP      C$EXIT
                    046260 000062                .WORD    L10045-
5896
5897
5898 046262 045 123 062 TSHD16: .NLIST BEX
5899 .ASCIZ /%S2%CHANGING OF THE CONTROLLER CONFIGURATION%/
5900 .LIST BEX
5901 046342                .EVEN
                    046342                ENDTST
                    046342 104401                L10045: TRAP      C$ETST

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 59  
TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST

```

5903 .SBTTL TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST
5904 .....
5905 IEX - TEST 17
5906 :PART 1 CHECKS THE REMOTE/LOCAL FUNCTION OF CHANNEL 2 USING THE FOLLOWING
5907 :COMMANDS GTL, LLO, NOT RTL.
5908 :PART 2 CHECKS THE REMOTE/LOCAL FUNCTION OF CHANNEL 1 USING THE FOLLOWING
5909 :COMMANDS GTL, LLO, NOT RTL.
5910 .....
5911 BGNTST
5912 046344 005737 002324 TST PNTF ;IS THE PNT FLAG SET
5913 046350 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
5914 046352 PRINTF #TSHD17 ;....
046352 012746 050644 MOV #TSHD17,-(SP)
046356 012746 000001 MOV #1,-(SP)
046362 010600 MOV SP,R0
046364 104417 TRAP C$PNTF
046366 062706 000004 ADD #4,SP
5915 046372 005037 002322 7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
5916 046376 104404 ITAC17: BGNSEG TRAP C$BSEG
5917 046400 004737 010220 JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
5918 046404 004737 010534 JSR PC,BGIN1 ;SET UP PARAMETER
5919 046410 112777 000220 133652 MOVB #220,@ICRHX ;-----LOAD SRE INTO ACR 1-----
5920 046416 004737 011072 JSR PC,WAIT ;WAIT 100 US
5921 046422 013701 002314 MOV DPA2,R1 ;CREATE MLA2
5922 046426 062701 000040 ADD #40,R1 ;....
5923 046432 010137 002412 MOV R1,MLA2 ;STORE MLA2
5924 046436 032737 000001 002314 BIT #1,DPA2 ;IS DPA EVEN
5925 046444 001415 BEQ 3$ ;BRANCH IF YES
5926 046446 052737 000001 002460 BIS #1,CDAT11 ;SET ULPA BIT IN COMPARE DATA FOR ISR
5927 046454 052737 000001 002462 BIS #1,CDAT12 ;....
5928 046462 052737 000001 002464 BIS #1,CDAT13 ;....
5929 046470 052737 000001 002466 BIS #1,CDAT14 ;....
5930 046476 000414 BR .+32 ;....
5931 046500 042737 000001 002460 3$: BIC #1,CDAT11 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
5932 046506 042737 000001 002462 BIC #1,CDAT12 ;....
5933 046514 042737 000001 002464 BIC #1,CDAT13 ;....
5934 046522 042737 000001 002466 BIC #1,CDAT14 ;....
5935 046530 113777 002412 133540 MOVB MLA2,@IDRHX ;-----LOAD MLA2 INTO DOR 1-----
5936 046536 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
5937 046542 052777 000010 133530 BIS #10,@CSRX ;SELECT CHANNEL 2
5938 046550 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
5939 046556 017737 133466 002502 MOV @IIRX,BAD ;GET IIR2 CONTENTS
5940 046564 022737 002403 002502 CMP #2403,BAD ;MA,IFC,RLC,MAC BIT SHOULD BE SET
5941 046572 001410 BEQ 10$ ;BRANCH IF YES
5942 046574 012737 002403 002500 MOV #2403,GOOD ;SET UP DATA FOR ERROR MESSAGE
5943 046602 ERRSOFT 1701,E501,ERR501 ;ERROR HANDLER
046602 104457 TRAP C$ERRSOFT
046604 003245 .WORD 1701
046606 005406 .WORD E501
046610 003702 .WORD ERR501
5944 046612 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
046612 104406 TRAP C$CLP1
5945 046614 017737 133436 002502 10$: MOV @ISR2,BAD ;GET ISR2 CONTENTS
5946 046622 023737 002460 002502 CMP CDAT11,BAD ;ATN,NDAC,REM,ATN,LAPS,LADS,(ULPA) IS SET
5947 046630 001410 BEQ 20$ ;BRANCH IF YES

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 59-2  
 TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST

5985	047130	042777	000010	133142	40\$:	BIC	#10,@CSRX	:SELECT CHANNEL 1
5986	047136	113777	002412	133132		MOVB	MLA2,@IDRHX	:-----LOAD MLA2 INTO DOR 1-----
5987	047144	004737	011060			JSR	PC,LOOP	:WAIT A LITTLE
5988	047150	052777	000010	133122		BIS	#10,@CSRX	:SELECT CHANNEL 2
5989	047156	017737	133066	002502		MOV	@IIRX,BAD	:GET IIR2 CONTENTS
5990	047164	022737	002002	002502		CMP	#2002,BAD	:MA,RLC BIT SHOULD BE SET
5991	047172	001410				BEQ	43\$	:BRANCH IF YES
5992	047174	012737	002002	002500		MOV	#2002,GOOD	:SET UP DATA FOR ERROR MESSAGE
5993	047202					ERRSOFT	1706,E501,ERR501	:ERROR HANDLER
	047202	104457						TRAP CSERSOFT
	047204	003252						.WORD 1706
	047206	005406						.WORD E501
	047210	003702						.WORD ERR501
5994	047212					CKLOOP		:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	047212	104406						TRAP C\$CLP1
5995	047214	017737	133036	002502	43\$:	MOV	@ISRX,BAD	:GET ISR2 CONTENTS
5996	047222	023737	002460	002502		CMP	CDAT11,BAD	:ATN,NDAC,REN,REM,ATN,LPAS,LADS(ULPA)SET
5997	047230	001410				BEQ	50\$	:BRANCH IF YES
5998	047232	013737	002460	002500		MOV	CDAT11,GOOD	:SET UP DATA FOR ERROR MESSAGE
5999	047240					ERRSOFT	1707,E502,ERR501	:ERROR HANDLER
	047240	104457						TRAP CSERSOFT
	047242	003253						.WORD 1707
	047244	005447						.WORD E502
	047246	003702						.WORD ERR501
6000	047250					CKLOOP		:BRANCH TO BGNSEG IF ERRLOOP IS SET
	047250	104406						TRAP C\$CLP1
6001	047252	042777	000010	133020	50\$:	BIC	#10,@CSRX	:SELECT CHANNEL 1
6002	047260	112777	000021	133010		MOVB	#21,@IDRHX	:-----LOAD LLO INTO DOR 1-----
6003	047266	004737	011060			JSR	PC,LOOP	:WAIT A LITTLE
6004	047272	052777	000010	133000		BIS	#10,@CSRX	:SELECT CHANNEL 2
6005	047300	017737	132744	002502		MOV	@IIRX,BAD	:GET IIR2 CONTENTS
6006	047306	005737	002502			TST	BAD	:CONTENTS SHOULD BE ZERO
6007	047312	001407				BEQ	53\$	:BRANCH IF YES
6008	047314	005037	002500			CLR	GOOD	:SET UP DATA FOR ERROR MESSAGE
6009	047320					ERRSOFT	1708,E501,ERR501	:ERROR HANDLER
	047320	104457						TRAP CSERSOFT
	047322	003254						.WORD 1708
	047324	005406						.WORD E501
	047326	003702						.WORD ERR501
6010	047330					CKLOOP		:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	047330	104406						TRAP C\$CLP1
6011	047332	017737	132720	002502	53\$:	MOV	@ISRX,BAD	:GET ISR2 CONTENTS
6012	047340	023737	002464	002502		CMP	CDAT13,BAD	:ATN,NDAC,REN,REM,LLO,ATN,LADS(ULPA) SET
6013	047346	001410				BEQ	60\$	:BRANCH IF YES
6014	047350	013737	002464	002500		MOV	CDAT13,GOOD	:SET UP DATA FOR ERROR MESSAGE
6015	047356					ERRSOFT	1709,E502,ERR501	:ERROR HANDLER
	047356	104457						TRAP CSERSOFT
	047360	003255						.WORD 1709
	047362	005447						.WORD E502
	047364	003702						.WORD ERR501
6016	047366					CKLOOP		:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	047366	104406						TRAP C\$CLP1
6017	047370	042777	000010	132702	60\$:	BIC	#10,@CSRX	:SELECT CHANNEL 1
6018	047376	112777	000001	132672		MOVB	#1,@IDRHX	:-----LOAD GTL INTO DOR 1-----
6019	047404	004737	011060			JSR	PC,LOOP	:WAIT A LITTLE
6020	047410	112777	000077	132660		MOVB	#77,@IDRHX	:-----LOAD UNL INTO DOR 1-----
6021	047416	004737	011060			JSR	PC,LOOP	:WAIT A LITTLE



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 59-3  
 TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST

```

6022 047422 052777 000010 132650      BIS      #10,@CSRX      ;SELECT CHANNEL 2
6023 047430 017737 132622 002502      MOV      @ISRX,BAD     ;GET ISR2 CONTENTS
6024 047436 023737 002466 002502      CMP      CDAT14,BAD    ;ATN,NDAC,REN,LLO,ATN,(ULPA) IS SET
6025 047444 001410                BEQ      63$           ;BRANCH IF YES
6026 047446 013737 002466 002500      MOV      CDAT14,GOOD   ;SET UP DATA FOR ERROR MESSAGE
6027 047454                ERRSOFT 1710,E502,ERR501 ;ERROR HANDLER
        047454 104457                TRAP    C$ERSOFT
        047456 003256                .WORD  1710
        047460 005447                .WORD  E502
        047462 003702                .WORD  ERR501
6028 047464                CKLOOP                ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        047466 104406                TRAP    C$CLP1
6029 047466 042777 000010 132604 63$:  BIC      #10,@CSRX      ;SELECT CHANNEL 1
6030 047474 112777 000020 132550      MOVB     #20,@IIRLX    ;----LOAD NOT SRE INTO ACR 1-----
6031 047502                ENDSEG
        047502 104405                10000$: TRAP    C$ESEG
6032                ;+-----+
6033                ;PART 2 CHECKS REMOTE/LOCAL INTERFACE FUNCTION OF CHANNEL 1
6034                ;+-----+
6035 047504                PSEU17: BGNSEG
        047504 104404                TRAP    C$BSEG
6036 047506 004737 010220      JSR      PC,CULPA      ;CLEAR ULPA BIT IN ISR 1 AND 2
6037 047512 004737 010710      JSR      PC,BGIN2     ;SET UP PARAMETER
6038 047516 112777 000220 132544      MOVB     #220,@ICRHX   ;----LOAD SRE INTO ACR 2-----
6039 047524 004737 011072      JSR      PC,WAIT      ;WAIT 100 US
6040 047530 032737 000001 002312      BIT      #1,DPA1      ;IS DPA EVEN
6041 047536 001415                BEQ      3$           ;BRANCH IF YES
6042 047540 052737 000001 002460      BIS      #1,CDAT11    ;SET ULPA BIT IN COMPARE DATA FOR ISR
6043 047546 052737 000001 002462      BIS      #1,CDAT12    ;...
6044 047554 052737 000001 002464      BIS      #1,CDAT13    ;...
6045 047562 052737 000001 002466      BIS      #1,CDAT14    ;...
6046 047570 000414                BR      .+32
6047 047572 042737 000001 002460 3$:  BIC      #1,CDAT11    ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
6048 047600 042737 000001 002462      BIC      #1,CDAT12    ;...
6049 047606 042737 000001 002464      BIC      #1,CDAT13    ;...
6050 047614 042737 000001 002466      BIC      #1,CDAT14    ;...
6051 047622 013701 002312      MOV      DPA1,R1      ;CREATE MLA1
6052 047626 062701 000040      ADD      #40,R1
6053 047632 010137 002410      MOV      R1,MLA1     ;STORE MLA1
6054 047636 113777 002410 132432      MOVB     MLA1,@IDRHX   ;----LOAD MLA1 INTO DOR 2-----
6055 047644 004737 011060      JSR      PC,LOOP     ;WAIT A LITTLE
6056 047650 042777 000010 132422      BIC      #10,@CSRX    ;SELECT CHANNEL 1
6057 047656 012737 000001 002374      MOV      #1,CHAN     ;LOAD CHANNEL NUMBER
6058 047664 017737 132360 002502      MOV      @IIRX,BAD   ;GET IIR1 CONTENTS
6059 047672 022737 002403 002502      CMP      #2403,BAD   ;MA,IFC,RLC,MAC BIT SHOULD BE SET
6060 047700 001410                BEQ      10$         ;BRANCH IF YES
6061 047702 012737 002403 002500      MOV      #2403,GOOD   ;SET UP DATA FOR ERROR MESSAGE
6062 047710                ERRSOFT 1711,E501,ERR501 ;ERROR HANDLER
        047710 104457                TRAP    C$ERSOFT
        047712 003257                .WORD  1711
        047714 005406                .WORD  E501
        047716 003702                .WORD  ERR501
6063 047720                CKLOOP                ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        047720 104406                TRAP    C$CLP1
6064 047722 017737 132330 002502 10$:  MOV      @ISRX,BAD   ;GET ISR1 CONTENTS
6065 047730 023737 002460 002502      CMP      CDAT11,BAD   ;ATN,NDAC,REN,ATN,LAPS,LADS,(ULPA) IS SET
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 59-4  
 TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST

6066	047736	001410				BEQ	20\$			:BRANCH IF YES
6067	047740	013737	002460	002500		MOV	CDAT11,GOOD			:SET UP DATA FOR ERROR MESSAGE
6068	047746					ERRSOFT	1712,E502,ERR501			:ERROR HANDLER
	047746	104457								TRAP CSERSOFT
	047750	003260								.WORD 1712
	047752	005447								.WORD E502
	047754	003702								.WORD ERR501
6069	047756					CKLOOP				:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	047756	104406								TRAP C\$CLP1
6070	047760	052777	000010	132312	20\$:	BIS	#10,@CSRX			:SELECT CHANNEL 2
6071	047766	112777	000001	132302		MOVB	#1,@IDRHX			:-----LOAD GTL INTO DOR 2-----
6072	047774	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE
6073	050000	112777	000077	132270		MOVB	#77,@IDRHX			:-----LOAD UNL INTO DOR 2-----
6074	050006	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE
6075	050012	042777	000010	132260		BIC	#10,@CSRX			:SELECT CHANNEL 1
6076	050020	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER
6077	050026	017737	132216	002502		MOV	@IIRX,BAD			:GET IIR1 CONTENTS
6078	050034	022737	000003	002502		CMP	#3,BAD			:RLC,MAC BIT SHOULD BE SET
6079	050042	001410				BEQ	30\$			:BRANCH IF YES
6080	050044	012737	000003	002500		MOV	#3,GOOD			:SET UP DATA FOR ERROR MESSAGE
6081	050052					ERRSOFT	1713,E501,ERR501			:ERROR HANDLER
	050052	104457								TRAP CSERSOFT
	050054	003261								.WORD 1713
	050056	005406								.WORD E501
	050060	003702								.WORD ERR501
6082	050062					CKLOOP				:BRANCH TO BGNSEG IF ERRLOOP IS SET
	050062	104406								TRAP C\$CLP1
6083	050064	052777	000010	132206	30\$:	BIS	#10,@CSRX			:SELECT CHANNEL 2
6084	050072	013701	002312			MOV	DPA1,R1			:CREATE MLA1
6085	050076	062701	000040			ADD	#40,R1			:-----
6086	050102	010137	002410			MOV	R1,MLA1			:STORE MLA1
6087	050106	113777	002410	132162		MOVB	MLA1,@IDRHX			:-----LOAD MLA1 INTO DOR 2-----
6088	050114	004737	011060			JSR	PC,LOOP			:WAIT A LITTLE
6089	050120	042777	000010	132152		BIC	#10,@CSRX			:SELECT CHANNEL 1
6090	050126	012737	000001	002374		MOV	#1,CHAN			:LOAD CHANNEL NUMBER
6091	050134	112777	000007	132126		MOVB	#7,@ICRHX			:-----LOAD NOT RTL INTO DOR 1-----
6092	050142	017737	132102	002502		MOV	@IIRX,BAD			:GET IIR1 CONTENTS
6093	050150	022737	002003	002502		CMP	#2003,BAD			:MA,RLC,MAC BIT SHOULD BE SET
6094	050156	001410				BEQ	33\$			:BRANCH IF YES
6095	050160	012737	002003	002500		MOV	#2003,GOOD			:SET UP DATA FOR ERROR MESSAGE
6096	050166					ERRSOFT	1744,E501,ERR501			:ERROR HANDLER
	050166	104457								TRAP CSERSOFT
	050170	003320								.WORD 1744
	050172	005406								.WORD E501
	050174	003702								.WORD ERR501
6097	050176					CKLOOP				:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	050176	104406								TRAP C\$CLP1
6098	050200	017737	132052	002502	33\$:	MOV	@ISR1,BAD			:GET ISR1 CONTENTS
6099	050206	023737	002462	002502		CMP	CDAT12,BAD			:ATN,NDAC,REN,ATN,LPAS,LADS(ULPA) IS SET
6100	050214	001410				BEQ	40\$			:BRANCH IF YES
6101	050216	013737	002462	002500		MOV	CDAT12,GOOD			:SET UP DATA FOR ERROR MESSAGE
6102	050224					ERRSOFT	1715,E502,ERR501			:ERROR HANDLER
	050224	104457								TRAP CSERSOFT
	050226	003263								.WORD 1715
	050230	005447								.WORD E502
	050232	003702								.WORD ERR501
6103	050234					CKLOOP				:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 59-5  
 TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST

```

050234 104406
6104 050236 052777 000010 132034 40$: BIS #10,@CSRX :SELECT CHANNEL 2 TRAP C$CLP1
6105 050244 113777 002410 132024 MOVB MLA1,@IDRHX :-----LOAD MLA1 INTO DOR 2-----
6106 050252 004737 011060 JSR PC,LOOP :WAIT A LITTLE
6107 050256 042777 000010 132014 BIC #10,@CSRX :SELECT CHANNEL 1
6108 050264 017737 131760 002502 MOV @IIRX,BAD :GET IIR1 CONTENTS
6109 050272 022737 002002 002502 CMP #2002,BAD :MA,RLC, BIT SHOULD BE SET
6110 050300 001410 BEQ 43$ :BRANCH IF YES
6111 050302 012737 002002 002500 MOV #2002,GOOD :SET UP DATA FOR ERROR MESSAGE
6112 050310 ERRSOFT 1716,E501,ERR501 :ERROR HANDLER
050310 104457 TRAP C$ERSOFT
050312 003264 .WORD 1716
050314 005406 .WORD E501
050316 003702 .WORD ERR501
6113 050320 CKLOOP :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET TRAP C$CLP1
050320 104406
6114 050322 017737 131730 002502 43$: MOV @ISRX,BAD :GET ISR1 CONTENTS
6115 050330 023737 002460 002502 CMP CDAT11,BAD :ATN,NDAC,REN,REM,ATN,LPAS,LADS(ULPA)SET
6116 050336 001410 BEQ 50$ :BRANCH IF YES
6117 050340 013737 002460 002500 MOV CDAT11,GOOD :SET UP DATA FOR ERROR MESSAGE
6118 050346 ERRSOFT 1717,E502,ERR501 :ERROR HANDLER
050346 104457 TRAP C$ERSOFT
050350 003265 .WORD 1717
050352 005447 .WORD E502
050354 003702 .WORD ERR501
6119 050356 CKLOOP :BRANCH TO BGNSEG IF ERRLOOP IS SET TRAP C$CLP1
050356 104406
6120 050360 052777 000010 131712 50$: BIS #10,@CSRX :SELECT CHANNEL 2
6121 050366 112777 000021 131702 MOVB #21,@IDRHX :-----LOAD LLO INTO DOR 2-----
6122 050374 004737 011060 JSR PC,LOOP :WAIT A LITTLE
6123 050400 042777 000010 131672 BIC #10,@CSRX :SELECT CHANNEL 1
6124 050406 017737 131636 002502 MOV @IIRX,BAD :GET IIR1 CONTENTS
6125 050414 005737 002502 TST BAD :CONTENTS SHOULD BE ZERO
6126 050420 001407 BEQ 53$ :BRANCH IF YES
6127 050422 005037 002500 CLR GOOD :SET UP DATA FOR ERROR MESSAGE
6128 050426 ERRSOFT 1718,E501,ERR501 :ERROR HANDLER
050426 104457 TRAP C$ERSOFT
050430 003266 .WORD 1718
050432 005406 .WORD E501
050434 003702 .WORD ERR501
6129 050436 CKLOOP :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET TRAP C$CLP1
050436 104406
6130 050440 017737 131612 002502 53$: MOV @ISRX,BAD :GET ISR1 CONTENTS
6131 050446 023737 002464 002502 CMP CDAT13,BAD :ATN,NDAC,REN,REM,LLO,ATN,LADS(ULPA) SET
6132 050454 001410 BEQ 60$ :BRANCH IF YES
6133 050456 013737 002464 002500 MOV CDAT13,GOOD :SET UP DATA FOR ERROR MESSAGE
6134 050464 ERRSOFT 1719,E502,ERR501 :ERROR HANDLER
050464 104457 TRAP C$ERSOFT
050466 003267 .WORD 1719
050470 005447 .WORD E502
050472 003702 .WORD ERR501
6135 050474 CKLOOP :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET TRAP C$CLP1
050474 104406
6136 050476 052777 000010 131574 60$: BIS #10,@CSRX :SELECT CHANNEL 2
6137 050504 112777 000001 131564 MOVB #1,@IDRHX :-----LOAD GTL INTO DOR 2-----
6138 050512 004737 011060 JSR PC,LOOP :WAIT A LITTLE
6139 050516 112777 000077 131552 MOVB #77,@IDRHX :-----LOAD UNL INTO DOR 2-----

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:45 PAGE 59-6  
TEST 17: REMOTE/LOCAL INTERFACE FUNCTION TEST

```

6140 050524 004737 011060          JSR      PC_LOOP          ;WAIT A LITTLE
6141 050530 042777 000010 131542    BIC      #10,@CSRX        ;SELECT CHANNEL 1
6142 050536 017737 131514 002502    MOV      @ISR1,BAD        ;GET ISR1 CONTENTS
6143 050544 023737 002466 002502    CMP      CDAT14,BAD       ;ATN,NDAC,REN,LLO,ATN,(ULPA) IS SET
6144 050552 001410          BEQ      63$              ;BRANCH IF YES
6145 050554 013737 002466 002500    MOV      CDAT14,GOOD      ;SET UP DATA FOR ERROR MESSAGE
6146 050562          ERRSOFT 1720,E502,ERR501 ;ERROR HANDLER
      050562 104457          TRAP    C$ERSOFT
      050564 003270          .WORD  1720
      050566 005447          .WORD  E502
      050570 003702          .WORD  ERR501
6147 050572          CKLOOP          ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      050572 104406          TRAP    C$CLP1
6148 050574 052777 000010 131470 63$:  BIS      #10,@CSRX        ;SELECT CHANNEL 2
6149 050602 112777 000020 131442    MOV      #20,@IIRLX      ;----LOAD NOT SRE INTO ACR 2-----
6150 050610          ENDSEG
      10001$:
6151 050610 104405          TRAP    C$ESEG
      050612 005737 002234          ;IS QUICK VERIFY PASS SELECTED
6152 050616 001010          BNE      EXQV17          ;IF YES EXIT TEST
6153 050620 005237 002322          INC      ITRCNT          ;ITERATION COUNTER +1
6154 050624 023737 002322 002320    CMP      ITRCNT,ITRDEF   ;DEFAULT ITERATION EXECUTED
6155 050632 001402          BEQ      EXQV17          ;IF YES EXIT TEST
6156 050634 000137 046376          JMP      ITAC17          ;IF NO TEST ITERATION
6157 050640          EXQV17: EXIT          ;
      050640 104432          TRAP    C$EXIT
      050642 000056          .WORD  L10046-
6158
6159
6160 050644 045 123 062 TSHD17: .NLIST BEX
      .ASCIZ /%S2%AREMOTE-LOCAL INTERFACE FUNCTION TEST%/
6161 .LIST BEX
6162 .EVEN
6163 050720          ENDTST
      050720          L10046:
      050720 104401          TRAP    C$ETST

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 60  
TEST 18: SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1

HA  
TE

```

6165 .SBTTL TEST 18: SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1
6166 .....
6167 IEX - TEST 18
6168 THIS TEST CHECKS THE SERIAL POLL REGISTER OF CHANNEL 1.
6169 PART 1 SETS AND CLEARS THE RSV BIT IN SPR REGISTER OF CHANNEL 1
6170 AND CHECKS THE SRQ BIT IN ISR2.
6171
6172 PART 2 CHECKS THE SERIAL POLL SEQUENCE OF CHANNEL 1.
6173
6174 IF QUICK VERIFY PASS IS NOT SELECTED, THE SERIAL POLL SEQUENCE IS CARRIED
6175 OUT WITH DIFFERENT DATA.
6176 .....
6177 BGNTST
6178 050722 005737 002324 TST PNTF ;IS THE PNT FLAG SET
6179 050726 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
6180 050730 PRINTF #TSHD18 ;....
        MOV #TSHD18,-(SP)
        MOV #1,-(SP)
        MOV SP,RO
        TRAP C$PNTF
        ADD #4,SP
6181 050750 012737 000101 002402 7$: MOV #101,RSAVE ;DATA STORE FOR SERIAL POLL SEQUENCE
6182 050756 104404 ITAC18: BGNSEG
        TRAP C$BSEG
6183 050760 004737 010220 JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
6184 050764 004737 010710 JSR PC,BGIN2 ;SET UP PARAMETER
6185 050770 032737 000001 002312 BIT #1,DPA1 ;IS DPA EVEN
6186 050776 001404 BEQ 3$ ;BRANCH IF YES
6187 051000 052737 000001 002470 BIS #1,CDAT15 ;SET ULPA BIT IN COMPARE DATA FOR ISR
6188 051006 000403 BR .+10
6189 051010 042737 000001 002470 3$: BIC #1,CDAT15 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
6190 051016 042777 000010 131254 BIC #10,@CSRX ;SELECT CHANNEL 1
6191 051024 112777 000100 131234 MOVB #100,@ICRLX ;-----SET RSV BIT IN SPR 1-----
6192 051032 052777 000010 131240 BIS #10,@CSRX ;SELECT CHANNEL 2
6193 051040 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
6194 051046 017737 131176 002502 MOV @IIRX,BAD ;GET IIR2 CONTENTS
6195 051054 022737 001020 002502 CMP #1020,BAD ;SRQ,BO BIT SHOULD BE SET
6196 051062 001410 BEQ 10$ ;BRANCH IF YES
6197 051064 012737 001020 002500 MOV #1020,GOOD ;SET UP DATA FOR ERROR MESSAGE
6198 051072 ERRSOFT 1801,E501,ERR501 ;ERROR HANDLER
        TRAP C$ERSOFT
        .WORD 1801
        .WORD E501
        .WORD ERR501
6199 051102 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        TRAP C$CLP1
6200 051104 017737 131146 002502 10$: MOV @ISR2,BAD ;GET ISR2 CONTENTS
6201 051112 032737 002000 002502 BIT #2000,BAD ;IS SRQ BIT SET ALSO ATN,NDAC,ATN(ULPA)
6202 051120 001010 BNE 13$ ;BRANCH IF YES
6203 051122 012737 122040 002500 MOV #122040,GOOD ;SET UP DATA FOR ERROR MESSAGE
6204 051130 ERRSOFT 1802,E502,ERR501 ;ERROR HANDLER
        TRAP C$ERSOFT
        .WORD 1802
        .WORD E502
        .WORD ERR501
6205 051140 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 60-1  
 TEST 18: SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1

```

        051140 104406                                TRAP  C$CLP1
6206 051142 042777 000010 131130 13$:  BIC    #10,@CSRX      :SELECT CHANNEL 1
6207 051150 112777 000000 131110      MOVB   #0,@ICRLX     :-----CLEAR SPR 1 REGISTER-----
6208 051156 052777 000010 131114      BIS    #10,@CSRX     :SELECT CHANNEL 2
6209 051164 017737 131066 002502      MOV    @ISR1,BAD     :GET ISR2 CONTENTS
6210 051172 032737 002000 002502      BIT    #2000,BAD    :SRQ BIT SHOULD BE CLEARED
6211 051200 001410                                BEQ    20$           :BRANCH IF YES
6212 051202 012737 120040 002500      MOV    #120040,GOOD  :SET UP DATA FOR COMPARE
6213 051210                                ERRSOFT 1803,E502,ERR501 :ERROR HANDLER
        051210 104457                                TRAP  C$ERSOFT
        051212 003413                                .WORD 1803
        051214 005447                                .WORD E502
        051216 003702                                .WORD ERR501
6214 051220                                CKLOOP              :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        051220 104406                                TRAP  C$CLP1
6215 051222 20$:  ENDSEG                                10000$:  TRAP  C$ESEG
        051222 104405                                :-----
6216 051224 :+++-----
6217 :PART 2 CHECKS THE SERIAL POLL SEQUENCE OF CHANNEL 1
6218 :-----
6219 051224 :+++-----
        051224 104404                                BGNSEG
6220 051226 042777 000010 131044      BIC    #10,@CSRX     :SELECT CHANNEL 1
6221 051234 113777 002402 131024      MOVB   RSAVE,@ICRLX  :-----LOAD DATA INTO SPR 1-----
6222 051242 052777 000010 131030      BIS    #10,@CSRX     :SELECT CHANNEL 2
6223 051250 112777 000077 131020      MOVB   #77,@IDRHX    :-----LOAD UNL INTO DOR 2-----
6224 051256 004737 011060      JSR    PC,LOOP       :WAIT A LITTLE
6225 051262 112777 000211 131000      MOVB   #211,@ICRHX   :-----LOAD LON INTO ACR 2-----
6226 051270 112777 000030 131000      MOVB   #30,@IDRHX    :-----LOAD SPE INTO DOR 2-----
6227 051276 004737 011060      JSR    PC,LOOP       :WAIT A LITTLE
6228 051302 013701 002312      MOV    DPA1,R1       :CREATE MTA1
6229 051306 062701 000100      ADD    #100,R1
6230 051312 010137 002414      MOV    R1,MTA1
6231 051316 113777 002414 130752      MOVB   MTA1,@IDRHX   :-----LOAD MTA1 INTO DOR 2-----
6232 051324 004737 011060      JSR    PC,LOOP       :WAIT A LITTLE
6233 051330 112777 000013 130732      MOVB   #13,@ICRHX   :-----LOAD GTS INTO ACR 2-----
6234 051336 042777 000010 130734      BIC    #10,@CSRX     :SELECT CHANNEL 1
6235 051344 012737 000001 002374      MOV    #1,CHAN       :LOAD CHANNEL NUMBER
6236 051352 017737 130672 002502      MOV    @IIR1,BAD    :GET IIR1 CONTENTS
6237 051360 022737 000405 002502      CMP    #405,BAD     :IFC,SPAS,MAC SHOULD BE SET
6238 051366 001410                                BEQ    30$           :BRANCH IF YES
6239 051370 012737 000405 002500      MOV    #405,GOOD    :SET UP DATA FOR ERROR MESSAGE
6240 051376                                ERRSOFT 1804,E501,ERR501 :ERROR HANDLER
        051376 104457                                TRAP  C$ERSOFT
        051400 003414                                .WORD 1804
        051402 005406                                .WORD E501
        051404 003702                                .WORD ERR501
6241 051406                                CKLOOP              :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        051406 104406                                TRAP  C$CLP1
6242 051410 017737 130642 002502 30$:  MOV    @ISR1,BAD     :GET ISR1 CONTENTS
6243 051416 023737 002470 002502      CMP    CDAT15,BAD   :NDAC,NRFD,TADS,(ULPA) SHOULD BE SET
6244 051424 001410                                BEQ    33$           :BRANCH IF YES
6245 051426 013737 002470 002500      MOV    CDAT15,GOOD  :SET UP DATA FOR ERROR MESSAGE
6246 051434                                ERRSOFT 1805,E502,ERR501 :ERROR HANDLER
        051434 104457                                TRAP  C$ERSOFT
        051436 003415                                .WORD 1805
    
```







HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 60-3  
TEST 18: SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1

```

6284 051724 000137 050756          JMP      ITAC18      ;IF NO TEST ITERATION
6285 051730          EXQV18: EXIT  TST      ;EXIT TEST
        051730 104432
        051732 000076          TRAP    CSEXIT
                                .WORD    L10047-.
6286
6287
6288 051734      045      123      062  TSHD18: .NLIST  BEX
6289              .ASCIZ  /%S2%ASERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 1%N/
6290              .LIST   BEX
6291 052030              .EVEN
        052030 104401          ENDTST
        052030          L10047:  TRAP    CSETST

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 61  
 TEST 19: SERVICE REQUEST INTERFACE FUNTION TEST OF CHANNEL 2

```

6293 .SBTTL TEST 19: SERVICE REQUEST INTERFACE FUNTION TEST OF CHANNEL 2
6294 .....
6295 IEX - TEST 19
6296 :THIS TEST CHECKS THE SERIAL POLL REGISTER OF CHANNEL 2.
6297 :PART 1 SETS AND CLEARS THE RSV BIT IN SPR REGISTER OF CHANNEL 2 AND
6298 :CHECKS THE SRQ BIT IN ISR1.
6299 :
6300 :PART 2 CHECKS THE SERIAL POLL SEQUENCE OF CHANNEL 2.
6301 :
6302 :IF QUICK VERIFY PASS IS NOT SELECTED, THE SERIAL POLL SEQUENCE IS CARRIED
6303 :OUT WITH DIFFERENT DATA.
6304 .....
6305 BGNTST
6306 052032 005737 002324 TST PNTF ;IS THE PNT FLAG SET
6307 052036 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
6308 052040 PRINTF #TSHD19 ;....
        MOV #TSHD19,-(SP)
        MOV #1,-(SP)
        MOV SP,RO
        TRAP C$PNTF
        ADD #4,SP
6309 052060 012737 000101 002402 7$: MOV #101,RSAVE ;DATA STORE FOR SERIAL POLL SEQUENCE
6310 052066 ITAC19: BGNSEG
        TRAP C$BSEG
6311 052070 004737 010220 JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
6312 052074 004737 010534 JSR PC,BGIN1 ;SET UP PARAMETER
6313 052100 032737 000001 002314 BIT #1,DPA2 ;IS DPA EVEN
6314 052106 001404 BEQ 3$ ;BRANCH IF YES
6315 052110 052737 000001 002470 BIS #1,CDAT15 ;SET ULPA BIT IN COMPARE DATA FOR ISR
6316 052116 000403 BR +10
6317 052120 042737 000001 002470 3$: BIC #1,CDAT15 ;CLEAR ULPA BIT IN COMPARE DATA FOR ISR
6318 052126 052777 000010 130144 BIS #10,@CSRX ;SELECT CHANNEL 2
6319 052134 112777 000100 130124 MOVB #100,@ICRLX ;-----SET RSV BIT IN SPR 2-----
6320 052142 042777 000010 130130 BIC #10,@CSRX ;SELECT CHANNEL 1
6321 052150 012737 000001 002374 MOV #1,CHAN ;LOAD CHANNEL NUMBER
6322 052156 017737 130066 002502 MOV @IIRX,BAD ;GET IIR1 CONTENTS
6323 052164 022737 001020 002502 CMP #1020,BAD ;SRQ,BO BIT SHOULD BE SET
6324 052172 001410 BEQ 10$ ;BRANCH IF YES
6325 052174 012737 001020 002500 MOV #1020,GOOD ;SET UP DATA FOR ERROR MESSAGE
6326 052202 ERRSOFT 1901,E501,ERR501 ;ERROR HANDLER
        TRAP C$ERRSOFT
        .WORD 1901
        .WORD E501
        .WORD ERR501
6327 052212 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        TRAP C$CLP1
6328 052214 017737 130036 002502 10$: MOV @ISRX,BAD ;GET ISR1 CONTENTS
6329 052222 032737 002000 002502 BIT #2000,BAD ;IS SRQ BIT SET ALSO ATN,NDAC,ATN(ULPA)
6330 052230 001010 BNE 13$ ;BRANCH IF YES
6331 052232 012737 122040 002500 MOV #122040,GOOD ;SET UP DATA FOR ERROR MESSAGE
6332 052240 ERRSOFT 1902,E502,ERR501 ;ERROR HANDLER
        TRAP C$ERRSOFT
        .WORD 1902
        .WORD E502
        .WORD ERR501
6333 052250 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
    
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 61-1  
 TEST 19: SERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 2

```

052250 104406
6334 052252 052777 000010 130020 13$: BIS #10,@CSRX ;SELECT CHANNEL 2 TRAP C$CLP1
6335 052260 112777 000000 130000 MOVB #0,@ICRLX ;-----CLEAR SPR 2 REGISTER-----
6336 052266 042777 000010 130004 BIC #10,@CSRX ;SELECT CHANNEL 1
6337 052274 017737 127756 002502 MOV @ISR2,BAD ;GET ISR1 CONTENTS
6338 052302 032737 002000 002502 BIT #2000,BAD ;SRQ BIT SHOULD BE CLEARED
6339 052310 001410 BEQ 20$ ;BRANCH IF YES
6340 052312 012737 120040 002500 MOV #120040,GOOD ;SET UP DATA FOR COMPARE
6341 052320 ERRSOFT 1903,E502,ERR501 ;ERROR HANDLER
052320 104457 TRAP C$ERSOFT
052322 003557 .WORD 1903
052324 005447 .WORD E502
052326 003702 .WORD ERR501
6342 052330 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
052330 104406 TRAP C$CLP1
6343 052332 20$: ENDSEG
052332 104405 10000$: TRAP C$ESEG
6344 :+++-----
6345 :PART 2 CHECKS THE SERIAL POLL SEQUENCE OF CHANNEL 2
6346 :+++-----
6347 052334 BGNSEG
052334 104404 TRAP C$BSEG
6348 052336 052777 000010 127734 BIS #10,@CSRX ;SELECT CHANNEL 2
6349 052344 113777 002402 127714 MOVB RSAVE,@ICRLX ;-----LOAD DATA INTO SPR 2-----
6350 052352 042777 000010 127720 BIC #10,@CSRX ;SELECT CHANNEL 1
6351 052360 112777 000077 127710 MOVB #77,@IDRHX ;-----LOAD UNL INTO DOR 1-----
6352 052366 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
6353 052372 112777 000211 127670 MOVB #211,@ICRHX ;-----LOAD LON INTO ACR 1-----
6354 052400 112777 000030 127670 MOVB #30,@IDRHX ;-----LOAD SPE INTO DOR 1-----
6355 052406 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
6356 052412 013701 002314 MOV DPA2,R1 ;CREATE MTA2
6357 052416 062701 000100 ADD #100,R1
6358 052422 010137 002416 MOV R1,MTA2 ;STORE MTA2
6359 052426 113777 002416 127642 MOVB MTA2,@IDRHX ;-----LOAD MTA1 INTO DOR 1-----
6360 052434 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
6361 052440 112777 000013 127622 MOVB #13,@ICRHX ;-----LOAD GTS INTO ACR 1-----
6362 052446 052777 000010 127624 BIS #10,@CSRX ;SELECT CHANNEL 2
6363 052454 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
6364 052462 017737 127562 002502 MOV @IIR2,BAD ;GET IIR2 CONTENTS
6365 052470 022737 000405 002502 CMP #405,BAD ;IFC,SPAS,MAC SHOULD BE SET
6366 052476 001410 BEQ 30$ ;BRANCH IF YES
6367 052500 012737 000405 002500 MOV #405,GOOD ;SET UP DATA FOR ERROR MESSAGE
6368 052506 ERRSOFT 1904,E501,ERR501 ;ERROR HANDLER
052506 104457 TRAP C$ERSOFT
052510 003560 .WORD 1904
052512 005406 .WORD E501
052514 003702 .WORD ERR501
6369 052516 CKLOOP ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
052516 104406 TRAP C$CLP1
6370 052520 017737 127532 002502 30$: MOV @ISR2,BAD ;GET ISR2 CONTENTS
6371 052526 023737 002470 002502 CMP CDAT15,BAD ;NDAC,NRFD,TADS,(ULPA) SHOULD BE SET
6372 052534 001410 BEQ 33$ ;BRANCH IF YES
6373 052536 013737 002470 002500 MOV CDAT15,GOOD ;SET UP DATA FOR ERROR MESSAGE
6374 052544 ERRSOFT 1905,E502,ERR501 ;ERROR HANDLER
052544 104457 TRAP C$ERSOFT
052546 003561 .WORD 1905

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 61-3  
TEST 19: SERVICE REQUEST INTERFACE FUNTION TEST OF CHANNEL 2

```

6412 053034 000137 052066          JMP      ITAC19          ;IF NO TEST ITERATION
6413 053040          EXQV19: EXIT      TST          ;EXIT TEST
      053040 104432
      053042 000076          TRAP      C$EXIT
                                   .WORD      L10050-.
6414
6415
6416 053044      045      123      062  TSHD19: .NLIST  BEX
6417          .ASCIZ  /%S2%ASERVICE REQUEST INTERFACE FUNCTION TEST OF CHANNEL 2%/
6418          .LIST   BEX
6419          .EVEN
      053140          ENDTST
      053140          L10050:
      053140 104401          TRAP      C$ETST

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 62  
TEST 20: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1

```

6421 .SBTTL TEST 20: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1
6422 :*****
6423 : IEX - TEST 20
6424 :PART 1 CHECKS PARALLEL POLL SEQUENCE (LOCAL CONFIGURED).
6425 :
6426 :PART 2 CHECKS PARALLEL POLL SEQUENCE (REMOTE CONFIGURED).
6427 :*****
6428 053142 BGNTST
6429 053142 005737 002324 TST PNTF ;IS THE PNT FLAG SET
6430 053146 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
6431 053150 PRINTF #TSHD20 ;....
        MOV #TSHD20,-(SP)
        MOV #1,-(SP)
        MOV SP,R0
        TRAP C$PNTF
        ADD #4,SP
6432 053170 012737 000001 002420 7$: MOV #1,MSA1 ;LOAD DATA FOR PPR
6433 053176 012737 000140 002402 MOV #140,RSVAE ;LOAD FIRST PPE
6434 053204 004737 010220 ITAC20: JSR PC,CULPA ;CLEAR ULPA BIT IN ISR 1 AND 2
6435 053210 004737 010710 JSR PC,BGIN2 ;SET UP PARAMETER
6436 053214 042777 000010 127056 BIC #10,@CSRX ;SELECT CHANNEL 1
6437 053222 112777 000223 127040 MOVB #223,@ICRHX ;----LOAD DAI INTO ACR 1-----
6438 053230 112777 000040 127024 MOVB #40,@ISRHX ;----SET UCG BIT IN ISR 1-----
6439 053236 BGNSEG
        TRAP C$BSEG
6440 053240 042777 000010 127032 BIC #10,@CSRX ;SELECT CHANNEL 1
6441 053246 112777 000020 127020 MOVB #20,@IDRLX ;----LOAD PP5 INTO PPR 1-----
6442 053254 052777 000010 127016 BIS #10,@CSRX ;SELECT CHANNEL 2
6443 053262 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
6444 053270 112777 000216 126772 MOVB #216,@ICRHX ;----LOAD RPP INTO ACR 2-----
6445 053276 017737 126762 002502 MOV @ICRX,BAD ;GET ICR2 CONTENTS
6446 053304 122737 000020 002502 CMPB #20,BAD ;ICR CONTENTS SHOULD BE 20
6447 053312 001410 BEQ 10$ ;BRANCH IF YES
6448 053314 012737 000020 002500 MOV #20,GOOD ;SET UP DATA FOR ERROR MESSAGE
6449 053322 ERRSOFT 2001,E901,ERR501 ;ERROR HANDLER
        TRAP C$ERRSOFT
        .WORD 2001
        .WORD E901
        .WORD ERR501
6450 053332 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET
        TRAP C$CLP1
6451 053334 017737 126716 002502 10$: MOV @ISRHX,BAD ;GET ISR2 CONTENTS
6452 053342 022737 104040 002502 CMP #104040,BAD ;ATN,E01,ATN SHOULD BE SET
6453 053350 001410 BEQ 20$ ;BRANCH IF YES
6454 053352 012737 104040 002500 MOV #104040,GOOD ;SET UP DATA FOR ERROR MESSAGE
6455 053360 ERRSOFT 2002,E502,ERR501 ;ERROR HANDLER
        TRAP C$ERRSOFT
        .WORD 2002
        .WORD E502
        .WORD ERR501
6456 053370 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET
        TRAP C$CLP1
6457 053372 112777 000016 126670 20$: MOVB #16,@ICRHX ;----LOAD NOT RPP INTO ACR 2-----
6458 053400 ENDSEG
        10000$: TRAP C$ESEG
        053400 104405

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 62-1  
 TEST 20: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1

```

6459
6460
6461
6462 053402
        053402 104404
6463 053404 052777 000010 126666      BIS      #10,@CSRX      ;SELECT CHANNEL 2
6464 053412 112777 000025 126656      MOV      #25,@IDRHX   ;----LOAD PPU INTO DOR 2-----
6465 053420 004737 011060      JSR      PC,LOOP     ;WAIT A LITTLE
6466 053424 042777 000010 126646      BIC      #10,@CSRX   ;SELECT CHANNEL 1
6467 053432 012737 000001 002374      MOV      #1,CHAN     ;LOAD CHANNEL NUMBER
6468 053440 017737 126620 002502      MOV      @ICRX,BAD   ;GET ICR1 CONTENTS
6469 053446 122737 000025 002502      CMP      #25,BAD     ;ICR CONTENTS SHOULD BE 25
6470 053454 001410      BEQ      23$         ;BRANCH IF YES
6471 053456 012737 000025 002500      MOV      #25,GOOD    ;SET UP DATA FOR ERROR MESSAGE
6472 053464      ERRSOFT 2003,E901,ERR501 ;ERROR HANDLER
        053464 104457      TRAP    C$ERSOFT
        053466 003723      .WORD  2003
        053470 005644      .WORD  E901
        053472 003702      .WORD  ERR501
6473 053474      CKLOOP      ;BRANCH TO BGNSEG IF ERRLOOP IS SET
        053474 104406      TRAP    C$CLP1
6474 053476 112777 000000 126570 23$:  MOV      #0,@IDRLX   ;----CLEAR PPR REGISTER-----
6475 053504 112777 000001 126556      MOV      #1,@ICRHX   ;----LOAD NOT DACR INTO ACR 1-----
6476 053512 013701 002312      MOV      DPA1,R1     ;CREATE MLA1
6477 053516 062701 000040      ADD      #40,R1
6478 053522 010137 002410      MOV      R1,MLA1     ;STORE MLA1
6479 053526 052777 000010 126544      BIS      #10,@CSRX   ;SELECT CHANNEL 2
6480 053534 113777 002410 126534      MOV      MLA1,@IDRHX ;----LOAD MLA1 INTO DOR 2-----
6481 053542 004737 011060      JSR      PC,LOOP     ;WAIT A LITTLE
6482 053546      ENDSEG
        053546      10001$:
6483 053550      BGNSEG      TRAP    C$ESEG
        053550 104404      TRAP    C$BSEG
6484 053552 052777 000010 126520      BIS      #10,@CSRX   ;SELECT CHANNEL 2
6485 053560 112777 000005 126510      MOV      #5,@IDRHX   ;----LOAD PPC INTO DOR 2-----
6486 053566 004737 011060      JSR      PC,LOOP     ;WAIT A LITTLE
6487 053572 042777 000010 126500      BIC      #10,@CSRX   ;SELECT CHANNEL 1
6488 053600 017737 126460 002502      MOV      @ICRX,BAD   ;GET ICR1 CONTENTS
6489 053606 122737 000005 002502      CMP      #5,BAD     ;ICR CONTENTS SHOULD BE 5
6490 053614 001410      BEQ      30$         ;BRANCH IF YES
6491 053616 012737 000005 002500      MOV      #5,GOOD    ;SET UP DATA FOR ERROR MESSAGE
6492 053624      ERRSOFT 2004,E901,ERR501 ;ERROR HANDLER
        053624 104457      TRAP    C$ERSOFT
        053626 003724      .WORD  2004
        053630 005644      .WORD  E901
        053632 003702      .WORD  ERR501
6493 053634      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
        053634 104406      TRAP    C$CLP1
6494 053636 112777 000024 126424 30$:  MOV      #24,@ICRHX   ;----LOAD PTS INTO ACR 1-----
6495 053644 112777 000001 126416      MOV      #1,@ICRHX   ;----LOAD NOT DACR INTO ACR 1-----
6496 053652      ENDSEG
        053652      10002$:
6497 053654      BGNSEG      TRAP    C$ESEG
        053654 104404      TRAP    C$BSEG
6498 053656 052777 000010 126414      BIS      #10,@CSRX   ;SELECT CHANNEL 2
    
```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 62-2  
 TEST 20: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1

```

6499 053664 113777 002402 126404      MOVB   RSAVE,@IDRHX      :----LOAD PPE INTO DOR 2-----
6500 053672 004737 011060              JSR    PC,LOOP           :WAIT A LITTLE
6501 053676 042777 000010 126374      BIC    #10,@CSRX        :SELECT CHANNEL 1
6502 053704 012737 000001 002374      MOV    #1,CHAN          :LOAD CHANNEL NUMBER
6503 053712 017737 126346 002502      MOV    @ICRX,BAD        :GET ICR1 CONTENTS
6504 053720 123737 002402 002502      CMPB  RSAVE,BAD         :ICR CONTENTS SHOULD BE PPE
6505 053726 001410              BEQ    40$              :BRANCH IF YES
6506 053730 013737 002402 002500      MOV    RSAVE,GOOD       :SET UP DATA FOR ERROR MESSAGE
6507 053736              ERRSOFT 2005,E901,ERR501 :ERROR HANDLER
                                TRAP   C$ERSOFT
                                .WORD  2005
                                .WORD  E901
                                .WORD  ERR501
6508 053746              CKLOOP                    :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP   C$CLP1
6509 053750 113777 002420 126316 40$:  MOVB   MSA1,@IDRLX      :----LOAD DATA INTO PPR 1-----
6510 053756 112777 000001 126304      MOVB  #1,@ICRHX         :----LOAD NOT DACR INTO ACR 1-----
6511 053764 052777 000010 126306      BIS   #10,@CSRX        :SELECT CHANNEL 2
6512 053772 112777 000077 126276      MOVB  #77,@IDRHX       :----LOAD UNL INTO DOR 2-----
6513 054000 004737 011060              JSR    PC,LOOP           :WAIT A LITTLE
6514 054004 112777 000216 126256      MOVB  #216,@ICRHX      :----LOAD RPP INTO ACR 2-----
6515 054012 017737 126246 002502      MOV    @ICRX,BAD        :GET ICR2 CONTENTS
6516 054020 123737 002420 002502      CMPB  MSA1,BAD         :COMPARE DATA
6517 054026 001410              BEQ    50$              :BRANCH IF YES
6518 054030 013737 002420 002500      MOV    MSA1,GOOD       :SET UP DATA FOR ERROR MESSAGE
6519 054036              ERRSOFT 2006,E901,ERR501 :ERROR HANDLER
                                TRAP   C$ERSOFT
                                .WORD  2006
                                .WORD  E901
                                .WORD  ERR501
6520 054046              CKLOOP                    :BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP   C$CLP1
6521 054050 112777 000016 126212 50$:  MOVB  #16,@ICRHX       :----LOAD NOT RPP INTO ACR 2-----
6522 054056              ENDSEG
                                10003$:
                                TRAP   C$ESEG
6523 054060 005737 002234              TST   QVP               :IS QUICK VERIFY PASS SELECTED
6524 054064 001012              BNE   EXQV20            :IF YES EXIT TEST
6525 054066 005237 002402              INC   RSAVE             :CREATE NEW PPE
6526 054072 106137 002420              ROLB  MSA1              :CHANGE DATA
6527 054076 022737 000157 002402      CMP   #157,RSAVE       :ALL PPE DONE
6528 054104 001402              BEQ   EXQV20            :EXIT TEST IF YES
6529 054106 000137 053204              JMP   ITAC20            :IF NO TEST ITERATION
6530 054112              EXQV20: EXIT          TST   :EXIT TEST
                                TRAP   C$EXIT
                                .WORD  L10051-.
6531 054114 000074
6532
6533 054116 045 123 062 TSHD20: .NLIST BEX
                                .ASCIZ /%S2%APARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 1%/
6534 .LIST BEX
6535 .EVEN
6536 054210 .ENDTST
                                L10051:
                                TRAP   C$ETST
                                054210 104401
                                054210
  
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 63  
TEST 21: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 2

```

6538      .SBTTL TEST 21: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 2
6539      :*****
6540      :               IEX - TEST 21
6541      :PART 1 CHECKS PARALLEL POLL SEQUENCE (LOCAL CONFIGURED).
6542      :
6543      :PART 2 CHECKS PARALLEL POLL SEQUENCE (REMOTE CONFIGURED).
6544      :*****
6545      BGNTST
6546      054212 005737 002324      TST      PNTF      ;IS THE PNT FLAG SET
6547      054216 001410      BEQ      7$      ;IF YES, PRINT THE TEST HEADER
6548      054220      PRINTF  #TSHD21      ;....
6549      054220 012746 055166      MOV      #TSHD21,-(SP)
6550      054224 012746 000001      MOV      #1,-(SP)
6551      054230 010600      MOV      SP,R0
6552      054232 104417      TRAP    C$PNTF
6553      054234 062706 000004      ADD     #4,SP
6554      054240 012737 000001 002420 7$: MOV      #1,MSA1      ;LOAD DATA FOR PPR
6555      054246 012737 000140 002402  MOV      #140,RSAVE  ;LOAD FIRST PPE
6556      054254 004737 010220  ITAC21: JSR     PC,C'ILPA    ;CLEAR ULPA BIT ISR 1 AND 2
6557      054260 004737 010534      JSR     PC,BGIN1     ;SET UP PARAMETER
6558      054264 052777 000010 126006  BIS     #10,@CSRX    ;SELECT CHANNEL 2
6559      054272 112777 000223 125770  MOVB   #223,@ICRHX  ;----LOAD DAI INTO ACR 2-----
6560      054300 112777 000040 125754  MOVB   #40,@ISRHX   ;----SET UCG BIT IN ISR 2-----
6561      054306      BGNSEG      TRAP    C$BSEG
6562      054310 052777 000010 125762  BIS     #10,@CSRX    ;SELECT CHANNEL 2
6563      054316 112777 000020 125750  MOVB   #20,@IDRLX   ;----LOAD PP5 INTO PPR 2-----
6564      054324 042777 000010 125746  BIC     #10,@CSRX    ;SELECT CHANNEL 1
6565      054332 012737 000001 002374  MOV     #1,CHAN      ;LOAD CHANNEL NUMBER
6566      054340 112777 000216 125722  MOVB   #216,@ICRHX  ;----LOAD RPP INTO ACR 1-----
6567      054346 017737 125712 002502  MOV     @ICRX,BAD    ;GET ICR1 CONTENTS
6568      054354 122737 000020 002502  CMPB   #20,BAD      ;ICR CONTENTS SHOULD BE 20
6569      054362 001410      BEQ     10$         ;BRANCH IF YES
6570      054364 012737 000020 002500  MOV     #20,GOOD     ;SET UP DATA FOR ERROR MESSAGE
6571      054372      ERRSOFT 2101,E901,ERR501 ;ERROR HANDLER
6572      054372 104457      TRAP    C$ERRSOFT
6573      054374 004065      .WORD   2101
6574      054376 005644      .WORD   E901
6575      054400 003702      .WORD   ERR501
6576      054402      CKLOOP      ;BRANCH TO BGNSEG IF ERRLOOP IS SET
6577      054402 104406      TRAP    C$CLP1
6578      054404 017737 125646 002502 10$: MOV     @ISRX,BAD    ;GET ISR1 CONTENTS
6579      054412 022737 104040 002502  CMP     #104040,BAD ;ATN,EOI,ATN SHOULD BE SET
6580      054420 001410      BEQ     20$         ;BRANCH IF YES
6581      054422 012737 104040 002500  MOV     #104040,GOOD ;SET UP DATA FOR ERROR MESSAGE
6582      054430      ERRSOFT 2102,E502,ERR501 ;ERROR HANDLER
6583      054430 104457      TRAP    C$ERRSOFT
6584      054432 004066      .WORD   2102
6585      054434 005447      .WORD   E502
6586      054436 003702      .WORD   ERR501
6587      054440      CKLOOP      ;BRANCH TO BGNSEG IF ERRLOOP IS SET
6588      054440 104406      TRAP    C$CLP1
6589      054442 112777 000016 125620 20$: MOVB   #16,@ICRHX   ;----LOAD NOT RPP INTO ACR 1-----
6590      054450      ENDSEG
6591      054450 104405      10000$: TRAP    C$ESEG

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 63-1  
 TEST 21: PARALLEL POLL INTERFACE FUNCTION TEST OF CHANNEL 2

```

6576
6577
6578
6579 054452
054452 104404
6580 054454 042777 000010 125616 BIC #10,@CSRX ;SELECT CHANNEL 1 TRAP C$BSEG
6581 054462 112777 000025 125606 MOVB #25,@IDRHX ;----LOAD PPU INTO DOR 1-----
6582 054470 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
6583 054474 052777 000010 125576 BIS #10,@CSRX ;SELECT CHANNEL 2
6584 054502 012737 000002 002374 MOV #2,CHAN ;LOAD CHANNEL NUMBER
6585 054510 017737 125550 002502 MOV @ICRX,BAD ;GET ICR2 CONTENTS
6586 054516 122737 000025 002502 CMPB #25,BAD ;ICR CONTENTS SHOULD BE 25
6587 054524 001410 BEQ 23$ ;BRANCH IF YES
6588 054526 012737 000025 002500 MOV #25,GOOD ;SET UP DATA FOR ERROR MESSAGE
6589 054534 ERRSOFT 2103,E901,ERR501 ;ERROR HANDLER
054534 104457 TRAP C$ERSOFT
054536 004067 .WORD 2103
054540 005644 .WORD E901
054542 003702 .WORD ERR501
6590 054544 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET TRAP C$CLP1
054544 104406
6591 054546 112777 000000 125520 23$: MOVB #0,@IDRLX ;----CLEAR PPR REGISTER-----
6592 054554 112777 000001 125506 MOVB #1,@ICRHX ;----LOAD NOT DACR INTO ACR 2-----
6593 054562 013701 002314 MOV DPA2,R1 ;CREATE MLA2
6594 054566 062701 000040 ADD #40,R1
6595 054572 010137 002412 MOV R1,MLA2 ;STORE MLA2
6596 054576 042777 000010 125474 BIC #10,@CSRX ;SELECT CHANNEL 1
6597 054604 113777 002412 125464 MOVB MLA2,@IDRHX ;----LOAD MLA2 INTO DOR 1-----
6598 054612 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
6599 054616 ENDSEG
054616 10001$: TRAP C$ESEG
054616 104405 TRAP C$BSEG
6600 054620 BGNSEG
054620 104404
6601 054622 042777 000010 125450 BIC #10,@CSRX ;SELECT CHANNEL 1 TRAP C$BSEG
6602 054630 112777 000005 125440 MOVB #5,@IDRHX ;----LOAD PPC INTO DOR 1-----
6603 054636 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
6604 054642 052777 000010 125430 BIS #10,@CSRX ;SELECT CHANNEL 2
6605 054650 017737 125410 002502 MOV @ICRX,BAD ;GET ICR2 CONTENTS
6606 054656 122737 000005 002502 CMPB #5,BAD ;ICR CONTENTS SHOULD BE 5
6607 054664 001410 BEQ 30$ ;BRANCH IF YES
6608 054666 012737 000005 002500 MOV #5,GOOD ;SET UP DATA FOR ERROR MESSAGE
6609 054674 ERRSOFT 2104,E901,ERR501 ;ERROR HANDLER
054674 104457 TRAP C$ERSOFT
054676 004070 .WORD 2104
054700 005644 .WORD E901
054702 003702 .WORD ERR501
6610 054704 CKLOOP ;BRANCH TO BGNSEG IF ERRLOOP IS SET TRAP C$CLP1
054704 104406
6611 054706 112777 000024 125354 30$: MOVB #24,@ICRHX ;----LOAD PTS INTO ACR 2-----
6612 054714 112777 000001 125346 MOVB #1,@ICRHX ;----LOAD NOT DACR INTO ACR 2-----
6613 054722 ENDSEG
054722 10002$: TRAP C$ESEG
054722 104405 TRAP C$BSEG
6614 054724 BGNSEG
054724 104404
6615 054726 042777 000010 125344 BIC #10,@CSRX ;SELECT CHANNEL 1 TRAP C$BSEG
    
```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 64  
TEST 22: END OF A MESSAGE BLOCK TEST

```

6655      .SBTTL TEST 22: END OF A MESSAGE BLOCK TEST
6656      :*****
6657      :               IEX - TEST 22
6658      :PART 1 CHECKS THE END OF A MESSAGE BLOCK FROM CHANNEL 1. CHANNEL 2 SENDS THE
6659      :      EOI MESSAGE VIA THE IEC/IEEE BUS.
6660      :PART 2 CHECKS THE END OF A MESSAGE BLOCK FROM CHANNEL 2. CHANNEL 1 SENDS THE
6661      :      EOI MESSAGE VIA THE IEC/IEEE BUS.
6662      :*****
6663      BGNSTST
6664      055262 005737 002324      TST      PNTF      :IS THE PNT FLAG SET
6665      055266 001410      BEQ      7$      :IF YES, PRINT THE TEST HEADER
6666      055270      PRINTF  #TSHD22      :....
6667      055270 012746 057202      MOV      #TSHD22,-(SP)
6668      055274 012746 000001      MOV      #1,-(SP)
6669      055300 010600      MOV      SP,R0
6670      055302 104417      TRAP    C$PNTF
6671      055304 062706 000004      ADD      #4,SP
6672      055310 005037 002322      7$:      CLR      ITRCNT      :CLEAR ITERATION COUNTER
6673      055314 004737 010220      JSR      PC,CULPA      :CLEAR ULPA BIT IN ISR 1 AND 2
6674      055320 004737 010534      ITAC22: JSR      PC,BGIN1      :SET UP PARAMETER
6675      055324      BGNSEG
6676      055324 104404      TRAP    C$BSEG
6677      055326 112777 000223 124734      MOVB    #223,@ICRHX      :----LOAD DAI INTO ACR 1-----
6678      055334 052777 000010 124736      BIS     #10,@CSRX      :SELECT CHANNEL 2
6679      055342 112777 000223 124720      MOVB    #223,@ICRHX      :----LOAD DAI INTO ACR 2-----
6680      055350 042777 000010 124722      BIC     #10,@CSRX      :SELECT CHANNEL 1
6681      055356 112777 000203 124704      MOVB    #203,@ICRHX      :----LOAD HDFA INTO ACR 1-----
6682      055364 112777 000211 124676      MOVB    #211,@ICRHX      :----LOAD LON INTO ACR 1-----
6683      055372 013701 002314      MOV     DPA2,R1      :CREATE MTA2
6684      055376 062701 000100      ADD     #100,R1
6685      055402 010137 002416      MOV     R1,MTA2      :STORE MTA2
6686      055406 113777 002416 124662      MOVB    MTA2,@IDRHX      :----LOAD MTA2 INTO DOR 1-----
6687      055414 004737 011060      JSR     PC,LOOP      :WAIT A LITTLE
6688      055420 112777 000013 124642      MOVB    #13,@ICRHX      :----LOAD GTS INTO ACR 1-----
6689      055426      ENDSEG
6690      055426 104405      10000$: TRAP    C$ESEG
6691      055430      BGNSEG      TRAP    C$BSEG
6692      055430 104404      TRAP    C$BSEG
6693      055432 052777 000010 124640      BIS     #10,@CSRX      :SELECT CHANNEL 2
6694      055440 112777 000125 124630      MOVB    #125,@IDRHX      :----LOAD DATA INTO DOR 2-----
6695      055446 042777 000010 124624      BIC     #10,@CSRX      :SELECT CHANNEL 1
6696      055454 012737 000001 002374      MOV     #1,CHAN      :LOAD CHANNEL NUMBER
6697      055462 017737 124562 002502      MOV     @IIRX,BAD      :GET IIR1 CONTENTS
6698      055470 022737 000060 002502      CMP     #60,BAD      :BI,BO BIT SHOULD BE SET
6699      055476 001410      BEQ     10$      :BRANCH IF YES
6700      055500 012737 000060 002500      MOV     #60,GOOD      :SET UP DATA FOR ERROR MESSAGE
6701      055506      ERRSOFT 2201,E501,ERR501 :ERROR HANDLER
6702      055506 104457      TRAP    C$ERSOFT
6703      055510 004231      .WORD  2201
6704      055512 005406      .WORD  E501
6705      055514 003702      .WORD  ERR501
6706      055516      CKLOOP      :BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
6707      055516 104406      TRAP    C$CLP1
6708      055520 017737 124532 002502      10$:  MOV     @ISR1,BAD      :GET ISR1 CONTENTS
6709      055526 022737 030004 002502      CMP     #30004,BAD      :NDAC,NRFD,LADS SHOULD BE SET

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 64-2  
 TEST 22: END OF A MESSAGE BLOCK TEST

```

6733 056006 017737 124236 002502      MOV      @IIRX,BAD      ;GET IIR1 CONTENTS
6734 056014 022737 000050 002502      CMP      #50,BAD      ;BI,END BIT SHOULD BE SET
6735 056022 001410 000050 002500      BEQ      33$          ;BRANCH IF YES
6736 056024 012737 000050 002500      MOV      #50,GOOD     ;SET UP DATA FOR ERROR MESSAGE
6737 056032 104457 000050 002500      ERRSOFT 2205,E501,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    2205
                                .WORD    E501
                                .WORD    ERR501
6738 056042 104406 000050 002500      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
6739 056044 017737 124206 002502 33$:  MOV      @ISR1,BAD     ;GET ISR1 CONTENTS
6740 056052 022737 034004 002502      CMP      #34004,BAD   ;NDAC,NRFD,EOI,LADS, IS SET
6741 056060 001410 000050 002500      BEQ      40$          ;BRANCH IF YES
6742 056062 013737 034004 002500      MOV      34004,GOOD   ;SET UP DATA FOR ERROR MESSAGE
6743 056070 104457 000050 002500      ERRSOFT 2206,E502,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    2206
                                .WORD    E502
                                .WORD    ERR501
6744 056100 104406 000050 002500      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
6745 056102 017737 124164 002502 40$:  MOV      @IDRX,BAD     ;READ DATA FROM DIR1
6746 056110 105037 002502 002502      CLRB     BAD          ;CLEAR LOW BYTE
6747 056114 000337 002502 002502      SWAB    BAD          ;SWAB HIGH WITH LOW BYTE
6748 056120 122737 000127 002502      CMPB    #127,BAD     ;COMPARE LOADED DATA WITH DIR CONTENTS
6749 056126 001410 000127 002502      BEQ      43$          ;BRANCH IF EQUAL
6750 056130 012737 000127 002500      MOV      #127,GOOD   ;SET UP DATA FOR ERROR MESSAGE
6751 056136 104457 000127 002500      ERRSOFT 2207,E222,ERR501 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    2207
                                .WORD    E222
                                .WORD    ERR501
6752 056146 104406 000127 002500      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
6753 056150 104405 000127 002500 43$:  ENDSEG      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
                                10003$:
6754 056152 112777 000002 124110      MOVB     #2,@ICRHX    ;----LOAD RHDF INTO ACR 1-----
6755 056160 112777 000015 124102      MOVB     #15,@ICRHX   ;----LOAD TCS INTO ACR 1-----
6756 056166 112777 000011 124074      MOVB     #11,@ICRHX   ;----LOAD NOT LON INTO ACR 1-----
6757 056174 112777 000137 124074      MOVB     #137,@IDRHX  ;----LOAD UNT INTO DOR 1-----
6758 056202 004737 011060 124054      JSR      PC,LOOP     ;WAIT A LITTLE
6759 056206 112777 000200 124054      MOVB     #200,@ICRHX  ;----LOAD SWRST INTO ACR 1-----
6760 056214 112777 000000 124046      MOVB     #0,@ICRHX   ;----LOAD NOT SWRST INTO ACR 1-----
6761
6762      ;+++-----
6763      ;PART 2 CHECKS THE "END OF A MESSAGE BLOCK" OF CHANNEL 2
6764 056222 004737 010220 124036 PSEU18: JSR      FC,CULPA ;CLEAR ULPA BIT IN ISR 1AND 2
6765 056226 004737 010710 124036      JSR      PC,BGIN2    ;SET UP PARAMETER
6766 056232 104404 000010 124036      BGNSEG
                                TRAP      C$BSEG
6767 056234 052777 000010 124036      BIS      #10,@CSRX   ;SELECT CHANNEL 2
6768 056242 112777 000223 124020      MOVB     #223,@ICRHX ;----LOAD DAI INTO ACR 2-----
6769 056250 042777 000010 124022      BIC      #10,@CSRX   ;SELECT CHANNEL 1
6770 056256 112777 000223 124004      MOVB     #223,@ICRHX ;----LOAD DAI INTO ACR 1-----
6771 056264 052777 000010 124006      BIS      #10,@CSRX   ;SELECT CHANNEL 2

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 64-3  
 TEST 22: END OF A MESSAGE BLOCK TEST

6772	056272	112777	000203	123770		MOVB	#203,@ICRHX		:-----LOAD HDFA INTO ACR 2-----
6773	056300	112777	000211	123762		MOVB	#211,@ICRHX		:-----LOAD LON INTO ACR 2-----
6774	056306	013701	002312			MOV	DPA1,R1		:CREATE MTA1
6775	056312	062701	000100			ADD	#100,R1		:STORE MTA1
6776	056316	010137	002414			MOV	R1,MTA1		:-----LOAD MTA1 INTO DOR 2-----
6777	056322	113777	002414	123746		MOVB	MTA1,@IDRHX		:WAIT A LITTLE
6778	056330	004737	011060			JSR	PC,LOOP		:-----LOAD GTS INTO ACR 2-----
6779	056334	112777	000013	123726		MOVB	#13,@ICRHX		
6780	056342					ENDSEG			
	056342								10004\$:
	056342	104405							TRAP C\$ESEG
6781	056344					BGNSEG			TRAP C\$BSEG
	056344	104404							
6782	056346	042777	000010	123724		BIC	#10,@CSRX		:SELECT CHANNEL 1
6783	056354	112777	000125	123714		MOVB	#125,@IDRHX		:-----LOAD DATA INTO DOR 1-----
6784	056362	052777	000010	123710		BIS	#10,@CSRX		:SELECT CHANNEL 2
6785	056370	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
6786	056376	017737	123646	002502		MOV	@IRX,BAD		:GET IIR2 CONTENTS
6787	056404	022737	000060	002502		CMP	#60,BAD		:BO,BI BIT SHOULD BE SET
6788	056412	001410				BEQ	10\$		:BRANCH IF YES
6789	056414	012737	000060	002500		MOV	#60,GOOD		:SET UP DATA FOR ERROR MESSAGE
6790	056422					ERRSOFT	2208,E501,ERR501		:ERROR HANDLER
	056422	104457							TRAP C\$ERSOFT
	056424	004240							.WORD 2208
	056426	005406							.WORD E501
	056430	003702							.WORD ERR501
6791	056432					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	056432	104406							TRAP C\$CLP1
6792	056434	017737	123616	002502	10\$:	MOV	@ISRX,BAD		:GET ISR2 CONTENTS
6793	056442	022737	030004	002502		CMP	#30004,BAD		:NDAC,NRFD,LADS SHOULD BE SET
6794	056450	001410				BEQ	20\$		:BRANCH IF YES
6795	056452	012737	030004	002500		MOV	#30004,GOOD		:SET UP DATA FOR ERROR MESSAGE
6796	056460					ERRSCFT	2209,E502,ERR501		:ERROR HANDLER
	056460	104457							TRAP C\$ERSOFT
	056462	004241							.WORD 2209
	056464	005447							.WORD E502
	056466	003702							.WORD ERR501
6797	056470					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	056470	104406							TRAP C\$CLP1
6798	056472	017737	123574	002502	20\$:	MOV	@IDRX,BAD		:GET DIR2 CONTENTS
6799	056500	105037	002502			CLRB	BAD		:
6800	056504	000337	002502			SWAB	BAD		:SWAB HIGH AND LOW BYTE
6801	056510	122737	000125	002502		CMPB	#125,BAD		:COMPARE WITH LOADED DATA
6802	056516	001410				BEQ	23\$		:BRANCH IF OK
6803	056520	012737	000125	002500		MOV	#125,GOOD		:SET UP DATA FOR ERROR MESSAGE
6804	056526					ERRSOFT	2210,E222,ERR501		:ERROR HANDLER
	056526	104457							TRAP C\$ERSOFT
	056530	004242							.WORD 2210
	056532	005675							.WORD E222
	056534	003702							.WORD ERR501
6805	056536					CKLOOP			:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	056536	104406							TRAP C\$CLP1
6806	056540				23\$:	ENDSEG			
	056540								10005\$:
	056540	104405							TRAP C\$ESEG
6807	056542	112777	000002	123520		MOVB	#2,@ICRHX		:-----LOAD RHDF INTO ACR 2-----
6808	056550	112777	000003	123512		MOVB	#3,@ICRHX		:-----LOAD NOT HDFA INTO ACR 2-----



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 64-4  
 TEST 22: END OF A MESSAGE BLOCK TEST

6809	056556	112777	000204	123504		MOVB #204,@ICRHX		:-----LOAD HDFE INTO ACR 2-----
6810	056564					BGNSEG		TRAP C\$BSEG
6811	056566	104404				BIC #10,@CSRX		:SELECT CHANNEL 1
6812	056574	042777	000010	123504		MOVB #126,@IDRHX		:-----LOAD DATA INTO DOR 1-----
6813	056602	112777	000126	123474		JSR PC,LOOP		:WAIT A LITTLE
6814	056606	004737	011060			BIS #10,@CSRX		:SELECT CHANNEL 2
6815	056614	052777	000010	123464		MOV @IDRX,BAD		:READ DATA FROM DIR2
6816	056622	017737	123452	002502		CLRB BAD		:CLEAR LOW BYTE OF IDR REGISTER
6817	056626	105037	002502			SWAB BAD		:SWAB HIGH WITH LOW BYTE
6818	056632	000337	002502			CMPB #126,BAD		:COMPARE DIR CONTENTS WITH LOADED DATA
6819	056640	122737	000126	002502		BEQ 30\$		:BRANCH IF EQUAL
6820	056642	001410				MOV #126,GOOD		:SET UP DATA FOR ERROR MESSAGE
6821	056650	012737	000126	002500		ERRSOFT 2211,E222,E501		:ERROR HANDLER
	056650	104457						TRAP C\$ERSOFT
	056652	004243						.WORD 2211
	056654	005675						.WORD E222
	056656	005406						.WORD E501
6822	056660					CKLOOP		:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	056660	104406						TRAP C\$CLP1
6823	056662				30\$:	ENDSEG		
	056662							10006\$:
	056662	104405						TRAP C\$ESEG
6824	056664	042777	000010	123406		BIC #10,@CSRX		:SELECT CHANNEL 1
6825	056672	112777	000010	123370		MOVB #10,@ICRHX		:-----LOAD FE01 INTO ACR 1-----
6826	056700					BGNSEG		TRAP C\$BSEG
	056700	104404						
6827	056702	112777	000127	123366		MOVB #127,@IDRHX		:-----LOAD DATA INTO DOR 1-----
6828	056710	004737	011060			JSR PC,LOOP		:WAIT A LITTLE
6829	056714	052777	000010	123356		BIS #10,@CSRX		:SELECT CHANNEL 2
6830	056722	017737	123322	002502		MOV @IIRX,BAD		:GET IIR2 CONTENTS
6831	056730	022737	000050	002502		CMP #50,BAD		:BI,END BIT SHOULD BE SET
6832	056736	001410				BEQ 33\$		:BRANCH IF YES
6833	056740	012737	000050	002500		MOV #50,GOOD		:SET UP DATA FOR ERROR MESSAGE
6834	056746					ERRSOFT 2212,E501,ERR501		:ERROR HANDLER
	056746	104457						TRAP C\$ERSOFT
	056750	004244						.WORD 2212
	056752	005406						.WORD E501
	056754	003702						.WORD ERR501
6835	056756					CKLOOP		:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	056756	104406						TRAP C\$CLP1
6836	056760	017737	123272	002502	33\$:	MOV @ISR2,BAD		:GET ISR2 CONTENTS
6837	056766	022737	034004	002502		CMP #34004,BAD		:NDAC,NRFD,E01,LADS IS SET
6838	056774	001410				BEQ 40\$		:BRANCH IF YES
6839	056776	012737	034004	002500		MOV #34004,GOOD		:SET UP DATA FOR ERROR MESSAGE
6840	057004					ERRSOFT 2212,E502,ERR501		:ERROR HANDLER
	057004	104457						TRAP C\$ERSOFT
	057006	004244						.WORD 2212
	057010	005447						.WORD E502
	057012	003702						.WORD ERR501
6841	057014					CKLOOP		:BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
	057014	104406						TRAP C\$CLP1
6842	057016	017737	123250	002502	40\$:	MOV @IDRX,BAD		:READ DATA FROM DIR2
6843	057024	105037	002502			CLRB BAD		:CLEAR LOW BYTE
6844	057030	000337	002502			SWAB BAD		:SWAB HIGH WITH LOW BYTE
6845	057034	122737	000127	002502		CMPB #127,BAD		:COMPARE LOADED DATA WITH DIR CONTENTS
6846	057042	001410				BEQ 43\$		:BRANCH IF EQUAL

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 64-5  
TEST 22: END OF A MESSAGE BLOCK TEST

```

6847 057044 012737 000127 002500      MOV      #127,GOOD      ;SET UP DATA FOR ERROR MESSAGE
6848 057052      ERRSOFT 2213,E222,ERR501 ;ERROR HANDLER
      057052 104457      TRAP      C$ERSOFT
      057054 004245      .WORD    2213
      057056 005675      .WORD    E222
      057060 003702      .WORD    ERR501
6849 057062      CKLOOP      ;BRANCH BACK TO BGNSEG IF ERRLOOP IS SET
      057062 104406      TRAP      C$CLP1
6850 057064      43$:      ENDSEG
      057064      10007$:
      057064 104405      TRAP      C$ESEG
6851 057066 112777 000002 123174      MOVB     #2,@ICRHX      ;----LOAD RHDF INTO ACR 2-----
6852 057074 112777 000015 123166      MOVB     #15,@ICRHX     ;----LOAD TCS INTO ACR 2-----
6853 057102 112777 000011 123160      MOVB     #11,@ICRHX     ;----LOAD NOT LON INTO ACR 2-----
6854 057110 112777 000137 123160      MOVB     #137,@IDRHX    ;----LOAD UNT INTO DOR 2-----
6855 057116 004737 011060      JSR      PC,LOOP       ;WAIT A LITTLE
6856 057122 005737 002234      TST      QVP           ;IS QUICK VERIFY PASS SELECTED
6857 057126 001023      BNE      EXQV22        ;IF YES EXIT TEST
6858 057130 005237 002322      INC      ITRCNT        ;INCREMENT COUNTER
6859 057134 023737 002322 002320      CMP      ITRCNT,ITRDEF ;ALL DONE
6860 057142 001415      BEQ      EXQV22        ;IF YES, EXIT TEST
6861 057144 042777 000010 123126      BIC      #10,@CSRX     ;SELECT CHANNEL 1
6862 057152 112777 000217 123110      MOVB     #217,@ICRHX    ;----LOAD SIC INTO ACR 1-----
6863 057160 004737 011072      JSR      PC,WAIT       ;WAIT A LITTLE
6864 057164 112777 000017 123076      MOVB     #17,@ICRHX     ;----LOAD NOT SIC INTO ACR 1-----
6865 057172 000137 055320      JMP      ITAC22        ;IF NO TEST ITERATION
6866 057176      EXQV22: EXIT      TST      ;EXIT TEST
      057176 104432      TRAP      C$EXIT
      057200 000046      .WORD    L10053-
6867
6868
6869 057202      045      123      062      TSHD22: .NLIST BEX
6870      .ASCII /%S2%AEND OF A MESSAGE BLOCK TEST%/
6871      .LIST BEX
6872 057246      .EVEN
      057246      ENDTST
      057246 104401      L10053: TRAP      C$ETST

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

6874  
 6875  
 6876  
 6877  
 6878  
 6879  
 6880  
 6881  
 6882  
 6883  
 6884  
 6885  
 6886  
 6887  
 6888  
 6889  
 6890 057250  
 057250  
 6891 057250 005737 002324  
 6892 057254 001410  
 6893 057256  
 057256 012746 063204  
 057262 012746 000001  
 057266 010600  
 057270 104417  
 057272 062706 000004  
 6894 057276 005037 002322  
 6895 057302 004737 010534  
 6896 057306  
 057306 104431  
 6897 057310 062700 000002  
 6898 057314 010037 002356  
 6899 057320 012701 010000  
 6900 057324 005003  
 6901 057326 110320  
 6902 057330 005203  
 6903 057332 005301  
 6904 057334 001374  
 6905 057336 010037 002360  
 6906 057342  
 057342 012746 000340  
 057346 012746 010152  
 057352 013746 002246  
 057356 012746 000003  
 057362 104437  
 057364 062706 000010  
 6907 057370  
 057370 012746 000340  
 057374 012746 010142  
 057400 013746 002244  
 057404 012746 000003  
 057410 104437  
 057412 062706 000010  
 6908 057416 112777 000212 122644  
 6909 057424 013701 002314  
 6910 057430 062701 000040  
 6911 057434 010137 002412

```

.SBTTL TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2
.....
                IEX - TEST 23
PART 1 SENDS DATA VIA THE IEC/IEEE BUS FROM CHAN. 1 TO 2 BY MEANS OF A DMA
I.E. CHAN. 1 WHICH IS SELECTED AS TALKER PERFORMS A DATI CYCLE,
WHEREAS CHANNEL 2 WHICH IS SELECTED AS A LISTENER PERFORMS A DATOB
CYCLE. THE MAX. SELECTABLE BYTE COUNT FOR THIS DATA TRANSFER IS
2K BYTES AND THE HIGHEST BUS ADDRESS IS BELOW 32K.
PART 2 CHECKS THE NON EXISTENT MEMORY BIT OF CHANNEL 1
THIS IS DONE BY A DMA FROM A NON EXISTING I/O PAGE ADDRESS SELECTED
IN THE BUS ADDRESS REGISTER OF CHAN. 1 (DATI CYCLE).
PART 3 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED
OVER 32K (IF MEMORY MANAGEMENT IS AVAILABLE).
PART 4 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED
OVER 64K (IF MEMORY MANAGEMENT IS AVAILABLE).
.....
                BGNTST
                                T23::
TST      PNTF      ;IS THE PNT FLAG SET
BEQ      7$       ;IF YES, PRINT THE TEST HEADER
PRINTF  #TSHD23  ;...
                                MOV      #TSHD23,-(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP    C$PNTF
                                ADD     #4,SP
7$:      CLR      ITRCNT      ;CLEAR ITERATION COUNTER
        JSR      PC,BGIN1    ;SET UP PARAMETER
ITAC23: MEMORY R0           ;GET THE FIRST FREE MEMORY LOCATION
                                TRAP    C$MEM
        ADD     #2,R0        ;USE THE SEC.FREE LOCATION FOR BUFFER
        MOV     R0,BUFAB     ;LOAD START ADDRESS OF BUFFER A
        MOV     #10000,R1    ;BUILD 2K BUFFER SIZE
        CLR     R3           ;R3 CONTAINS THE LOADED DATA
1$:      MOV     R3,(R0)+    ;LOAD DATA INTO BUFFER A
        INC     R3           ;CREATE NEW DATA
        DEC     R1           ;2K LOADED
        BNE    1$          ;IF NO, LOAD NEXT DATA BYTE
        MOV     R0,BUFBB    ;LOAD START ADDRESS OF BUFFER B
        SETVEC VECC2,#INTSC2,#PRI07 ;SET VECTOR FOR CHANNEL 2
                                MOV     #PRI07,-(SP)
                                MOV     #INTSC2,-(SP)
                                MOV     VECC2,-(SP)
                                MOV     #3,-(SP)
                                TRAP    C$SVEC
                                ADD     #10,SP
        SETVEC VECC1,#INTSC1,#PRI07 ;SET VECTOR FOR CHANNEL 1
                                MOV     #PRI07,-(SP)
                                MOV     #INTSC1,-(SP)
                                MOV     VECC1,-(SP)
                                MOV     #3,-(SP)
                                TRAP    C$SVEC
                                ADD     #10,SP
        MOV     #212,@ICRMX ;----LOAD TON INTO ACR 1-----
        MOV     DPA2,R1     ;CREATE MLA2
        ADD     #40,R1
        MOV     R1,MLA2    ;STORE MLA2
    
```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-2  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

6954	057746	042777	000010	122324		BIC	#10,@CSRX		:SELECT CHANNEL 1	
6955	057754	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER	
6956	057762	013737	002356	002500		MOV	BUFAB,GOOD		:SET UP COMPARE VALUE	
6957	057770	063737	002240	002500		ADD	BCINP,GOOD		:...	
6958	057776	017737	122300	002502		MOV	@BARX,BAD		:GET BAR1 CONTENTS	
6959	060004	023737	002502	002500		CMP	BAD,GOOD		:HAS BAR1 THE CORRECT ADDRESS	
6960	060012	001405				BEQ	23\$		:BRANCH IF YES	
6961	060014					ERRSOFT	2303,E234,ERR501		:ERROR HANDLER	
	060014	104457							TRAP	C\$ERSOFT
	060016	004377							.WORD	2303
	060020	006200							.WORD	E234
	060022	003702							.WORD	ERR501
6962	060024					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS	SET
	060024	104406							TRAP	C\$CLP1
6963	060026	017737	122252	002502	23\$:	MOV	@BCRX,BAD		:GET BCR1 CONTENTS	
6964	060034	005037	002500			CLR	GOOD		:...	
6965	060040	023737	002502	002500		CMP	BAD,GOOD		:IS BCR1 ZERO	
6966	060046	001405				BEQ	30\$		:BRANCH IF YES	
6967	060050					ERRSOFT	2304,E235,ERR501		:ERROR HANDLER	
	060050	104457							TRAP	C\$ERSOFT
	060052	004400							.WORD	2304
	060054	006231							.WORD	E235
	060056	003702							.WORD	ERR501
6968	060060					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS	SET
	060060	104406							TRAP	C\$CLP1
6969	060062	052777	000010	122210	30\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2	
6970	060070	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER	
6971	060076	013737	002360	002500	33\$:	MOV	BUFBB,GOOD		:SET UP COMPARE VALUE	
6972	060104	063737	002240	002500		ADD	BCINP,GOOD		:...	
6973	060112	017737	122164	002502		MOV	@BARX,BAD		:GET BAR2 CONTENTS	
6974	060120	023737	002500	002502		CMP	GOOD,BAD		:HAS BAR2 THE CORRECT ADDRESS	
6975	060126	001405				BEQ	40\$		:BRANCH IF YES	
6976	060130					ERRSOFT	2305,E234,ERR501		:ERROR HANDLER	
	060130	104457							TRAP	C\$ERSOFT
	060132	004401							.WORD	2305
	060134	006200							.WORD	E234
	060136	003702							.WORD	ERR501
6977	060140					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS	SET
	060140	104406							TRAP	C\$CLP1
6978	060142	017737	122136	002502	40\$:	MOV	@BCRX,BAD		:GET BCR2 CONTENTS	
6979	060150	005737	002502			TST	BAD		:BCR2 CONTENTS SHOULD BE ZERO	
6980	060154	001407				BEQ	43\$		:BRANCH IF YES	
6981	060156	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGE	
6982	060162					ERRSOFT	2306,E235,ERR501		:ERROR HANDLER	
	060162	104457							TRAP	C\$ERSOFT
	060164	004402							.WORD	2306
	060166	006231							.WORD	E235
	060170	003702							.WORD	ERR501
6983	060172					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS	SET
	060172	104406							TRAP	C\$CLP1
6984	060174	013701	002356		43\$:	MOV	BUFAB,R1		:PROVIDE FIRST BYTE OF BUFFER A	
6985	060200	013702	002360			MOV	BUFBB,R2		:PROVIDE FIRST BYTE OF BUFFER B	
6986	060204	005037	002404			CLR	CNT1		:CLEAR BUFFER COUNTER	
6987	060210	005237	002404		44\$:	INC	CNT1		:...	
6988	060214	122122				CMPB	(R1)+,(R2)+		:BUFFER A EQUAL BUFFER B	
6989	060216	001433				BEQ	46\$		:IF YES CONTINUE	
6990	060220	005037	002500			CLR	GOOD		:CLEAR GOOD	



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-3  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

```

6991 060224 005037 002502          CLR      BAD          ;CLEAR BAD
6992 060230 005037 002426          CLR      TXADRH       ;SET UP DATA FOR ERROR MESSAGE
6993 060234 010137 002430          MOV      R1,TXADRL    ;...
6994 060240 005337 002430          DEC      TXADRL       ;...
6995 060244 005037 002422          CLR      RXADRH       ;...
6996 060250 010237 002424          MOV      R2,RXADRL    ;...
6997 060254 005337 002424          DEC      RXADRL       ;...
6998 060260 116137 177777 002500  MOVB     -1(R1),GOOD  ;...
6999 060266 116237 177777 002502  MOVB     -1(R2),BAD   ;...
7000 060274          ERRSOFT 2307,E231,ERR231 ;ERROR HANDLER
          060274 104457          TRAP     C$ERSOFT
          060276 004403          .WORD   2307
          060300 006010          .WORD   E231
          060302 003744          .WORD   ERR231
7001 060304          CKLOOP          ;BRANCH TO BGNSEG IF ERRLOOP IS SET
          060304 104406          TRAP     C$CLP1
7002 060306 023737 002404 002240 46$:  CMP      CNT1,BCINP   ;ALL BYTES COMPARED ?
7003 060314 001335          BNE      44$         ;IF NO, GET NEXT ONE
7004 060316          ENDSEG
          060316          10000$:
7005 060320 005077 121754          CLR      @CSRX        ;CLEAR CSR2, SELECT CHANNEL 1
7006 060324 112777 000014 121736  MOVB     #14,@ICRHX   ;---LOAD TCA INTO ACR 1---
7007 060332 017737 121734 002502  MOV      @IDRX,BAD    ;READ DIR1 FOR CLEAR BO BIT IN IIR
7008 060340 005077 121734          CLR      @CSRX        ;CLEAR CSR1
7009
7010          :+++-----
7011          :PART 2 CHECK THE NON EXISTENT MEMORY BIT OF CHAN. 1 (THE I/O PAGE IS USED
7012          :FOR NON EXISTENT MEMORY
7013          :+++-----
7013 060344 012701 002624 002374  PSEU23: MOV      #TABEL,R1 ;CLEAR TWO WORD IN TABEL E
7014 060350 012737 000001          MOV      #1,CHAN     ;LOAD CHANNEL NUMBER
7015 060356 005021          CLR      (R1)+       ;...
7016 060360 005011          CLR      (R1)        ;...
7017 060362          SETVEC  #4,#NXM,#PRI07 ;SET UP VECTOR FOR TRAP TO 4
          060362 012746 000340          MOV      #PRI07,-(SP)
          060366 012746 010132          MOV      #NXM,-(SP)
          060372 012746 000004          MOV      #4,-(SP)
          060376 012746 000003          MOV      #3,-(SP)
          060402 104437          TRAP     C$SVEC
          060404 062706 000010          ADD     #10,SP
7018 060410 005037 002326          CLR      NXMFLG      ;CLEAR FLAG
7019 060414 012701 160000          MOV      #160000,R1  ;LOAD FIRST ADDRESS OF I/O PAGE
7020 060420 005711          TST     (R1)         ;FIND A NON EXISTEND LOCATION
7021 060422 005737 002326          TST     NXMFLG
7022 060426 062701 000002          ADD     #2,R1        ;NEXT I/O PAGE ADD.(NO AUTO INCR.11/44!)
7023 060432 001772          BEQ     1$
7024 060434          CLRVEC #4           ;SET VECTOR 4 TO NORMAL STATE
          060434 012700 000004          MOV      #4,R0
          060440 104436          TRAP     C$CVEC
7025 060442 162701 000002          SUB     #2,R1
7026 060446          BGNSEG          ;...
          060446          TRAP     C$BSEG
7027 060450 005037 002376          CLR      INTFC1      ;CLEAR INTERRUPT FLAG
7028 060454 005037 002400          CLR      INTFC2      ;CLEAR INTERRUPT FLAG
7029 060460 042777 000010 121612  BIC     #10,@CSRX    ;SELECT CHANNEL 1
7030 060466 010177 121610          MOV      R1,@BARX    ;LOAD BAR1 WITH A NON EXISTS I/O ADDR.
7031 060472 012737 000002 002402  MOV      #2,RSAVE    ;LOAD NUMBER OF BYTE COUNTS

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-4  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

```

7032 060500 005437 002402          NEG      RSAVE
7033 060504 013777 002402 121572    MOV      RSAVE,@BCRX
7034 060512 012777 017167 121560    MOV      #17167,@CSRX
7035 060520 052777 000010 121552    BIS      #10,@CSRX
7036 060526 012777 002624 121546    MOV      #TABE,@BARX
7037 060534 013777 002402 121542    MOV      RSAVE,@BCRX
7038 060542 012777 000101 121530    MOV      #101,@CSRX
7039 060550 042777 000010 121522    BIC      #10,@CSRX
7040 060556 012737 000001 002374    MOV      #1,CHAN
7041 060564          SETPRI  #PRI00
      060564 012700 000000
      060570 104441
7042 060572 112777 000013 121470    MOV      #13,@ICRHX
7043 060600 012702 077777          MOV      #77777,R2
7044 060604 005737 002376          TST     INTFC1
7045 060610 091015          BNE     6$
7046 060612 005302          DEC     R2
7047 060614 001373          BNE     3$
7048 060616 017737 121456 002502    MOV      @CSRX,BAD
7049 060624 012737 040066 002500    MOV      #40066,GOOD
7050 060632          ERRSOFT 2308,E233,ERR201
      060632 104457
      060634 004404
      060636 006123
      060640 003500
7051 060642          CKLOOP
      060642 104406
7052 060644          SETPRI  #PRI07
      060644 012700 000340
      060650 104441
7053 060652 112777 000014 121410    MOV      #14,@ICRHX
7054 060660 052777 000010 121412    BIS      #10,@CSRX
7055 060666 012737 000002 002374    MOV      #2,CHAN
7056 060674 017737 121400 002502    MOV      @CSRX,BAD
7057 060702 042737 017000 002502    BIC      #17000,BAD
7058 060710 022737 000111 002502    CMP      #111,BAD
7059 060716 001410          BEQ     12$
7060 060720 012737 000111 002500    MOV      #111,GOOD
7061 060726          ERRSOFT 2309,E401,ERR501
      060726 104457
      060730 004405
      060732 005265
      060734 003702
7062 060736          CKLOOP
      060736 104406
7063 060740          ENDSEG
      060740
      060740 104405
7064 060742 005077 121332          CLR     @CSRX
7065 060746 017737 121320 002502    MOV      @IDRX,BAD
7066 060754 005077 121320          CLR     @CSRX
      060754
7067
7068
7069
7070 060760 005037 002376          CLR     INTFC1
7071 060764 005037 002400          CLR     INTFC2
7072 060770 023727 002342 000001    CMP     PHHSIZ,#1

```

```

;BILD 2'COMPLEMENT
;LOAD BCR1
;DMA ENB,DMA DIR,INT ENB,BA16-21,SYS
;SELECT CHANNEL 2
;LOAD BAR2 WITH STAR ADDRESS OF TABE
;LOAD BCR2
;---SET DMA ENB,INT ENB CSR2-----
;SELECT CHANNEL 1
;LOAD CHANNEL NUMBER
;SET PRIORITY TO ZERO
      MOV      #PRI00,RO
      TRAP     C$SPRI
;---LOAD GTS INTO ACR 1-----
;LOAD LOOP COUNTER
;IS INTERRUPT IN CHANNEL 1 OCCER
;BRANCH IF YES
;DECREMENT COUNTER
;IF NO,TEST AGIN
;GET CSR1 CONTENTS
;NXM,BA 16+17,DMA DIR,SYS CONT
;ERROR HANDLER
      TRAP     C$ERSOFT
      .WORD   2308
      .WORD   E233
      .WORD   ERR201
;BRANCH TO BGNSEG IF ERRLOOP IS SET
      TRAP     C$CLP1
;NO FURTHER INTERRUPT ALLOWED
      MOV      #PRI07,RO
      TRAP     C$SPRI
;----LOAD TCA INTO ACR 1-----
;SELECT CHANNEL 2
;LOAD CHANNEL NUMBER
;GET CSR2 CONTENTS
;IGNORE BIT 9-12
;INT ENB,MUX,DMA ENB SHOULD BE SET
;BRANCH IF YES
;SET UP DATA FOR ERROR MESSAGE
;ERROR HANDLER
      TRAP     C$ERSOFT
      .WORD   2309
      .WORD   E401
      .WORD   ERR501
;BRANCH TO BGNSEG IF ERRLOOP IS SET
      TRAP     C$CLP1
      10001$:
      TRAP     C$ESEG
;CLEAR CSR2 ,SELECT CHANNEL 1
;READ DIR1 FOR CLEAR BO BIT IN IIR
;CLEAR CSR1

```

```

:++
:PART 3 DMA OVER 32K
:++
;CLEAR INTERRUPT FLAG
;CLEAR INTERRUPT FLAG
;IS THERE MORE THAN 32K

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-6  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

7124	061326	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER		
7125	061334	005737	002400		TST	INTFC2	:IS AN INTERRUPT IN CHANNEL 2 OCCURED		
7126	061340	001012			BNE	25\$	:BRANCH IF YES		
7127	061342	017737	120732	002502	MOV	@CSRX,BAD	:GET CSR2 CONTENTS		
7128	061350	012737	100030	002500	MOV	#100030,GOOD	:BC OF,BA 16,MUX SHOULD BE SET		
7129	061356				ERRSOFT	2311,E232,ERR201	:ERROR HANDLER		
	061356	104457						TRAP	C\$ERSOFT
	061360	004407						.WORD	2311
	061362	006072						.WORD	E232
	061364	003500						.WORD	ERR201
7130	061366				25\$:	SETPRI	#PRI07	:NO FURTHER INTERRUPT ALLOWED	
	061366	012700	000340					MOV	#PRI07,R0
	061372	104441						TRAP	C\$SPRI
7131	061374	042777	000010	120676	BIC	#10,@CSRX	:SELECT CHANNEL 1		
7132	061402	012737	000001	002374	MOV	#1,CHAN	:LOAD CHANNEL NUMBER		
7133	061410	112777	000014	120652	MOVB	#14,@ICRHX	:----LOAD TCA INTO ACR 1-----		
7134	061416	013737	002336	002500	MOV	PHLOW,GOOD	:SET UP COMPARE VALUE		
7135	061424	063737	002240	002500	ADD	BCINP,GOOD			
7136	061432	017737	120644	002502	MOV	@BARX,BAD	:GET BAR1 CONTENTS		
7137	061440	023737	002502	002500	CMP	BAD,GOOD	:HAS BAR1 THE CORRECT ADDRESS		
7138	061446	001404			BEQ	30\$	:BRANCH IF YES		
7139	061450				ERRSOFT	2312,E234,ERR501	:ERROR HANDLER		
	061450	104457						TRAP	C\$ERSOFT
	061452	004410						.WORD	2312
	061454	006200						.WORD	E234
	061456	003702						.WORD	ERR501
7140	061460	017737	120620	002502	30\$:	MOV	@BCRX,BAD	:GET BCR1 CONTENTS	
7141	061466	005037	002500			CLR	GOOD		
7142	061472	023737	002502	002500		CMP	BAD,GOOD	:IS BCR1 ZERO	
7143	061500	001404				BEQ	31\$	:BRANCH IF YES	
7144	061502					ERRSOFT	2313,E235,ERR501	:ERROR HANDLER	
	061502	104457						TRAP	C\$ERSOFT
	061504	004411						.WORD	2313
	061506	006231						.WORD	E235
	061510	003702						.WORD	ERR501
7145	061512	052777	000010	120560	31\$:	BIS	#10,@CSRX	:SELECT CHANNEL 2	
7146	061520	012737	000002	002374		MOV	#2,CHAN	:LOAD CHANNEL NUMBER	
7147	061526	013737	002360	002500		MOV	BUFBB,GOOD	:SET UP COMPARE VALUE	
7148	061534	042737	160000	002500		BIC	#160000,GOOD	:CLEAR BIT 13+14+15	
7149	061542	063737	002240	002500		ADD	BCINP,GOOD		
7150	061550	017737	120526	002502		MOV	@BARX,BAD	:GET BAR2 CONTENTS	
7151	061556	023737	002500	002502		CMP	GOOD,BAD	:HAS BAR2 THE CORRECT ADDRESS	
7152	061564	001404				BEQ	40\$	:BRANCH IF YES	
7153	061566					ERRSOFT	2314,E234,ERR501	:ERROR HANDLER	
	061566	104457						TRAP	C\$ERSOFT
	061570	004412						.WORD	2314
	061572	006200						.WORD	E234
	061574	003702						.WORD	ERR501
7154	061576	017737	120502	002502	40\$:	MOV	@BCRX,BAD	:GET BCR2 CONTENTS	
7155	061604	005737	002502			TST	BAD	:BCR2 CONTENTS SHOULD BE ZERO	
7156	061610	001406				BEQ	43\$	:BRANCH IF YES	
7157	061612	005037	002500			CLR	GOOD	:SET UP DATA FOR ERROR MESSAGE	
7158	061616					ERRSOFT	2315,E235,ERR501	:ERROR HANDLER	
	061616	104457						TRAP	C\$ERSOFT
	061620	004413						.WORD	2315
	061622	006231						.WORD	E235
	061624	003702						.WORD	ERR501



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-7  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

```

7159 061626 012737 000001 177572 43$: MOV #1,SRO ;**ENABLE MEMORY MANAGEMENT**
7160 061634 013701 002356 MOV BUFAB,R1 ;PROVIDE FIRST BYTE OF BUFFER A
7161 061640 013702 002360 MOV BUFBB,R2 ;PROVIDE FIRST BYTE OF BUFFER B
7162 061644 005037 002404 CLR CNT1 ;CLEAR BUFFER COUNTER
7163 061650 005237 002404 44$: INC CNT1 ;
7164 061654 122122 CMPB (R1)+,(R2)+ ;BUFFER A EQUAL BUFFER B
7165 061656 001446 BEQ 46$ ;IF YES CONTINUE
7166 061660 005037 002500 CLR GOOD ;CLEAR GOOD
7167 061664 005037 002502 CLR BAD ;CLEAR BAD
7168 061670 012737 000001 002426 MOV #1, TXADRH ;TX ADDRESS IS OVER 32K
7169 061676 012737 000001 002422 MOV #1, RXADRH ;RX ADDRESS IS OVER 32K
7170 061704 010137 002430 MOV R1, TXADRL ;GET ADDRESS OVER 32K
7171 061710 005337 002430 DEC TXADRL ;
7172 061714 042737 160000 002430 BIC #160000, TXADRL ;CLEAR PAR INFORMATION
7173 061722 013737 002430 002424 MOV TXADRL, RXADRL ;GENERATE RX ADDRESS FROM TX ADDRESS
7174 061730 052737 010000 002424 BIS #10000, RXADRL ;RX ADDR IS TX ADDRESS +2K
7175 061736 116137 177777 002500 MOVB -1(R1), GOOD ;
7176 061744 116237 177777 002502 MOVB -1(R2), BAD ;
7177 061752 005037 177572 CLR SRO ;**DISABLE MEMORY MANAGEMENT**
7178 061756 ERRSOFT 2316,E231,ERR231 ;ERROR HANDLER
                                TRAP CSERSOFT
                                .WORD 2316
                                .WORD E231
                                .WORD ERR231
7179 061766 012737 000001 177572 MOV #1,SRO ;**ENABLE MEMORY MANAGEMENT**
7180 061774 023737 002404 002240 46$: CMP CNT1,BCINP ;ALL BYTES COMPARED ?
7181 062002 001322 BNE 44$ ;IF NO, GET NEXT ONE
7182 062004 005037 177572 CLR SRO ;**DISABLE MEMORY MANAGEMENT**
7183 062010 005077 120264 CLR @CSRX ;CLEAR CSR2, SELECT CHANNEL 1
7184 062014 017737 120252 002502 MOV @IDRX, BAD ;READ DIR1 FOR CLEAR BO BIT IN IIR
7185 062022 005077 120252 CLR @CSRX ;CLEAR CSR1
7186 -----
7187 :PART 4 DMA OVER 64K
7188 -----
7189 062026 005037 002376 PSEU33: CLR INTFC1 ;CLEAR INTERRUPT FLAG
7190 062032 005037 002400 CLR INTFC2 ;CLEAR INTERRUPT FLAG
7191 062036 023727 002342 000002 CMP PHHSIZ,#2 ;IS THERE MORE THAN 64K
7192 062044 002404 BLT 11$ ;IF NO SKIP TEST
7193 062046 003005 BGT 13$ ;
7194 062050 005737 002344 TST PHLSIZ ;
7195 062054 001002 BNE 13$ ;
7196 062056 000137 063200 11$: JMP EXQV23 ;JUMP TO TEST END
7197 062062 005037 002336 13$: CLR PHLOW ;LOAD INPUT FOR CONVERSION ROUTINE
7198 062066 012737 000002 002334 MOV #2,PHHIGH ;
7199 062074 004737 007724 JSR PC,PVCON ;CREATE VIRTUAL ADDRESS
7200 062100 013700 002340 MOV VIADD,R0 ;GET START ADDRESS OF BUFFER A
7201 062104 010037 002356 MOV R0,BUFAB ;LOAD START ADDRESS OF BUFFER A
7202 062110 012701 010000 MOV #10000,R1 ;LOAD COUNTER FOR 2K
7203 062114 005003 CLR R3 ;R3 CONTAINS
7204 062116 012737 000001 177572 MOV #1,SRO ;**ENABLE MEMORY MANAGEMENT**
7205 062124 110320 16$: MOVB R3,(R0)+ ;LOAD BUFFER WITH DATA
7206 062126 005203 INC R3 ;CREAT NEXT DATA
7207 062130 005301 DEC R1 ;
7208 062132 001374 BNE 16$ ;
7209 062134 012701 010000 MOV #10000,R1 ;LOAD 2K
7210 062140 010037 002360 MOV R0,BUFBB ;LOAD START ADDRESS OF BUFFER B
7211 062144 105020 20$: CLRB (R0)+ ;CLEAR BUFFER B

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-8  
TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

7212	062146	005301			DEC	R1			:2K CLEARED
7213	062150	001375			BNE	20\$			:IF YES,DO THE TEST
7214	062152	005037	177572		CLR	SRO			:**DISABLE MEMORY MANAGEMENT**
7215	062156	112777	000212	120104	MOVB	#212,@ICRHX			:---LOAD TON INTO ACR 1-----
7216	062164	013701	002314		MOV	DPA2,R1			:CREATE MLA2
7217	062170	062701	000040		ADD	#40,R1			:STORE MLA2
7218	062174	010137	002412		MOV	R1,MLA2			:---LOAD MLA2 INTO DOR 1-----
7219	062200	113777	002412	120070	MOVB	MLA2,@IDRHX			:WAIT A LITTLE
7220	062206	004737	011060		JSR	PC,LOOP			:---LOAD START ADDRESS OF BUFFER A---
7221	062212	005077	120064		CLR	@BARX			:STORE BYTE COUNT INPUT
7222	062216	013737	002240	002402	MOV	BCINP,RSAVE			:BILD 2'COMPL FOR BCR
7223	062224	005437	002402		NEG	RSAVE			:---LOAD INPUT INTO BCR 1-----
7224	062230	013777	002402	120046	MOV	RSAVE,@BCRX			:DMA ENB,DMA DIR,INT ENB,BA 17,SYS C
7225	062236	012777	000147	120034	MOV	#147,@CSRX			:SELECT CHANNEL 2
7226	062244	052777	000010	120026	BIS	#10,@CSRX			:---LOAD START ADDRESS OF BUFFER B---
7227	062252	012777	010000	120022	MOV	#10000,@BARX			:---LOAD INPUT INTO BCR 1---
7228	062260	013777	002402	120016	MOV	RSAVE,@BCRX			:---SET DMA ENB,INT ENB,BIT17 IN CSR2
7229	062266	012777	000141	120004	MOV	#141,@CSRX			:SET PRIORITY TO ZERO
7230	062274				SETPRI	#PRI00			
	062274	012700	000000						MOV #PRI00,RO
	062300	104441							TRAP C\$SPRI
7231	062302	042777	000010	117770	BIC	#10,@CSRX			:SELECT CHANNEL 1
7232	062310	012737	000001	002374	MOV	#1,CHAN			:LOAD CHANNEL NUMBER
7233	062316	112777	000013	117744	MOVB	#13,@ICRHX			:---LOAD GTS INTO ACR 1-----
7234	062324	012701	077777		MOV	#77777,R1			:LOAD LOOP COUNTER
7235	062330	005737	002376		TST	INTFC1			:IS INTERRUPT IN CHANNEL 1 OCCER
7236	062334	001014			BNE	24\$			:BRANCH IF YES
7237	062336	005301			DEC	R1			:DECREMENT COUNTER
7238	062340	001373			BNE	23\$			:IF NO,TEST AGIN
7239	062342	017737	117732	002502	MOV	@CSRX,BAD			:GET CSR1 CONTENTS
7240	062350	012737	100046	002500	MOV	#100046,GOOD			:BC OF,DMA DIR,BA 17,SYS CONT
7241	062356				ERRSOFT	2317,E232,ERR201			:ERROR HANDLER
	062356	104457							TRAP C\$ERSOFT
	062360	004415							.WORD 2317
	062362	006072							.WORD E232
	062364	003500							.WORD ERR201
7242	062366	052777	000010	117704	BIS	#10,@CSRX			:SELECT CHANNEL 2
7243	062374	012737	000002	002374	MOV	#2,CHAN			:LOAD CHANNEL NUMBER
7244	062402	005737	002400		TST	INTFC2			:IS AN INTERRUPT IN CHANNEL 2 OCCURED
7245	062406	001012			BNE	25\$			:BRANCH IF YES
7246	062410	017737	117664	002502	MOV	@CSRX,BAD			:GET CSR2 CONTENTS
7247	062416	012737	100050	002500	MOV	#100050,GOOD			:BC OF,BA 17,MUX SHOULD BE SET
7248	062424				ERRSOFT	2318,E232,ERR201			:ERROR HANDLER
	062424	104457							TRAP C\$ERSOFT
	062426	004416							.WORD 2318
	062430	006072							.WORD E232
	062432	003500							.WORD ERR201
7249	062434				SETPRI	#PRI07			:NO FURTHER INTERRUPT ALLOWED
	062434	012700	000340						MOV #PRI07,RO
	062440	104441							TRAP C\$SPRI
7250	062442	042777	000010	117630	BIC	#10,@CSRX			:SELECT CHANNEL 1
7251	062450	012737	000001	002374	MOV	#1,CHAN			:LOAD CHANNEL NUMBER
7252	062456	112777	000014	117604	MOVB	#14,@ICRHX			:---LOAD TCA INTO ACR 1-----
7253	062464	013737	002336	002500	MOV	PHLOW,GOOD			:SET UP COMPARE VALUE
7254	062472	063737	002240	002500	ADD	BCINP,GOOD			
7255	062500	017737	117576	002502	MOV	@BARX,BAD			:GET BAR1 CONTENTS
7256	062506	023737	002502	002500	CMP	BAD,GOOD			:HAS BAR1 THE CORRECT ADDRESS



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 65-9  
 TEST 23: DMA DATA TRANSFER TEST FROM CHANNEL 1 TO 2

7257	062514	001404				BEQ	30\$		:BRANCH IF YES			
7258	062516					ERRSOFT	2319,E234,ERR501		:ERROR HANDLER			
	062516	104457								TRAP	C\$ERSOFT	
	062520	004417								.WORD	2319	
	062522	006200								.WORD	E234	
	062524	003702								.WORD	ERR501	
7259	062526	017737	117552	002502	30\$:	MOV	@BCRX,BAD		:GET BCR1 CONTENTS			
7260	062534	005037	002500			CLR	GOOD		:IS BCR1 ZERO			
7261	062540	023737	002502	002500		CMP	BAD,GOOD		:BRANCH IF YES			
7262	062546	001404				BEQ	31\$		:ERROR HANDLER			
7263	062550					ERRSOFT	2320,E235,ERR501			TRAP	C\$ERSOFT	
	062550	104457								.WORD	2320	
	062552	004420								.WORD	E235	
	062554	006231								.WORD	ERR501	
	062556	003702										
7264	062560	052777	000010	117512	31\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2			
7265	062566	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER			
7266	062574	013737	002360	002500		MOV	BUFBB,GOOD		:SET UP COMPARE VALUE			
7267	062602	042737	160000	002500		BIC	#160000,GOOD		:CLEAR BIT 13+14+15			
7268	062610	063737	002240	002500		ADD	BCINP,GOOD		:...			
7269	062616	017737	117460	002502		MOV	@BARX,BAD		:GET BAR2 CONTENTS			
7270	062624	023737	002500	002502		CMP	GOOD,BAD		:HAS BAR2 THE CORRECT ADDRESS			
7271	062632	001404				BEQ	40\$		:BRANCH IF YES			
7272	062634					ERRSOFT	2321,E234,ERR501		:ERROR HANDLER			
	062634	104457								TRAP	C\$ERSOFT	
	062636	004421								.WORD	2321	
	062640	006200								.WORD	E234	
	062642	003702								.WORD	ERR501	
7273	062644	017737	117434	002502	40\$:	MOV	@BCRX,BAD		:GET BCR2 CONTENTS			
7274	062652	005737	002502			TST	BAD		:BCR2 CONTENTS SHOULD BE ZERO			
7275	062656	001406				BEQ	43\$		:BRANCH IF YES			
7276	062660	005037	002500			CLR	GOOD		:SET UP DATA FOR ERROR MESSAGE			
7277	062664					ERRSOFT	2322,E235,ERR501		:ERROR HANDLER			
	062664	104457								TRAP	C\$ERSOFT	
	062666	004422								.WORD	2322	
	062670	006231								.WORD	E235	
	062672	003702								.WORD	ERR501	
7278	062674	012737	000001	177572	43\$:	MOV	#1,SRO		:**ENABLE MEMORY MANAGEMENT**			
7279	062702	013701	002356			MOV	BUFAB,R1		:PROVIDE FIRST BYTE OF BUFFER A			
7280	062706	013702	002360			MOV	BUFBB,R2		:PROVIDE FIRST BYTE OF BUFFER B			
7281	062712	005037	002404			CLR	CNT1		:CLEAR BUFFER COUNTER			
7282	062716	005237	002404		44\$:	INC	CNT1		:...			
7283	062722	122122				CMPB	(R1)+,(R2)+		:BUFFER A EQUAL BUFFER B			
7284	062724	001446				BEQ	46\$		:IF YES CONTINUE			
7285	062726	005037	002500			CLR	GOOD		:CLEAR GOOD			
7286	062732	005037	002502			CLR	BAD		:CLEAR BAD			
7287	062736	012737	000002	002426		MOV	#2,TXADRH		:TX ADDRESS IS OVER 64K			
7288	062744	012737	000002	002422		MOV	#2,RXADRH		:RX ADDRESS IS OVER 64K			
7289	062752	010137	002430			MOV	R1,TXADRL		:GET ADDRESS OVER 64K			
7290	062756	005337	002430			DEC	TXADRL		:...			
7291	062762	042737	160000	002430		BIC	#160000,TXADRL		:CLEAR PAR INFORMATION			
7292	062770	013737	002430	002424		MOV	TXADRL,RXADRL		:GENERATE RX ADDRESS FROM TX ADDRESS			
7293	062776	052137	010000	002424		BIS	#10000,RXADRL		:RX ADDRESS IS TX ADDRESS +2K			
7294	063004	116137	177777	002500		MOVB	-1(R1),GOOD		:SET UP DATA FOR ERROR MESSAGE			
7295	063012	116237	177777	002502		MOVB	-1(R2),BAD		:...			
7296	063020	005037	177572			CLR	SRO		:**DISABLE MEMORY MANAGEMENT**			
7297	063024					ERRSOFT	2323,E231,ERR231		:ERROR HANDLER			





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

```

7327 .SBTTL TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1
7328 :*****
7329 : IEX - TEST 24
7330 :PART 1 SENDS DATA VIA THE IEC/IEEE BUS FROM CHAN. 2 TO 1 BY MEANS OF A DMA
7331 : I.E. CHANNEL 2 WHICH IS SELECTED AS TALKER PERFORMS A DATI CYCLE,
7332 : WHEREAS CHANNEL 1 WHICH IS SELECTED AS A LISTENER PERFORMS A DATOB
7333 : CYCLE. THE MAX. SELECTABLE BYTE COUNT FOR THIS DATA TRANSFER IS
7334 : IS 2K BYTES AND THE HIGHEST BUS ADDRESS IS BELOW 32K.
7335 :PART 2 CHECKS THE NON EXISTENT MEMORY BIT OF CHANNEL 1.
7336 : THIS IS DONE BY A DMA FROM A NON EXISTING I/O PAGE ADDRESS SELECTED
7337 : IN THE BUS ADDRESS REGISTER OF CHANNEL 2 (DATOB CYCLE).
7338 :PART 3 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED
7339 : OVER 32K (IF MEMORY MANAGEMENT IS AVAILABLE).
7340 :PART 4 SAME PROCEDURE AS IN PART 1 EXCEPT THE DATA TRANSFER IS EXECUTED
7341 : OVER 64K (IF MEMORY MANAGEMENT IS AVAILABLE).
7342 :*****
7343 BGNSTST
7344 063270 005737 002324 TST PNTF ;IS THE PNT FLAG SET
7345 063274 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
7346 063276 PRINTF #TSHD24 ;...
7347 063316 005037 002322 7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
7348 063322 004737 010710 ITAC24: JSR PC,BGIN2 ;SET UP PARAMETER
7349 063326 MEMORY R0 ;GET THE FIRST FREE MEMORY LOCATION
7350 063330 062700 000002 ADD #2,R0 ;USE THE SEC.FREE LOCATION FOR BUFFER
7351 063334 010037 002356 MOV R0,BUFAB ;LOAD START ADDRESS OF BUFFER A
7352 063340 012701 010000 MOV #10000,R1 ;BUILD 2K BUFFER SIZE
7353 063344 005003 CLR R3 ;R3 CONTAINS THE LOADED DATA
7354 063346 110320 1$: MOVB R3,(R0)+ ;LOAD DATA INTO BUFFER A
7355 063350 005203 INC R3 ;CREATE NEW DATA
7356 063352 005301 DEC R1 ;2K LOADED
7357 063354 001374 BNE 1$ ;IF NO, LOAD NEXT DATA BYTE
7358 063356 010037 002360 MOV R0,BUFBB ;LOAD START ADDRESS OF BUFFER B
7359 063362 SETVEC VECC2,#INTSC2,#PRI07 ;SET VECTOR FOR CHANNEL 2
7360 063362 012746 000340 MOV #PRI07,-(SP)
7361 063366 012746 010152 MOV #INTSC2,-(SP)
7362 063372 013746 002246 MOV VECC2,-(SP)
7363 063376 012746 000003 MOV #3,-(SP)
7364 063402 104437 TRAP C$SVEC
7365 063404 062706 000010 ADD #10,SP
7366 063410 SETVEC VECC1,#INTSC1,#PRI07 ;SET VECTOR FOR CHANNEL 1
7367 063410 012746 000340 MOV #PRI07,-(SP)
7368 063414 012746 010142 MOV #INTSC1,-(SP)
7369 063420 013746 002244 MOV VECC1,-(SP)
7370 063424 012746 000003 MOV #3,-(SP)
7371 063430 104437 TRAP C$SVEC
7372 063432 062706 000010 ADD #10,SP
7373 063436 112777 000212 116624 MOVB #212,@ICRHX ;----LOAD TON INTO ACR 2-----
7374 063444 013701 002312 MOV DPA1,R1 ;CREATE MLA1
7375 063450 062701 000040 ADD #40,R1
7376 063454 010137 002410 MOV R1,MLA1 ;STORE MLA1
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-1  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

7365	063460	113777	002410	116610		MOVB	MLA1,@IDRHX		:----LOAD MLA1 INTO DOR 2-----
7366	063466	004737	011060			JSR	PC,LOOP		:WAIT A LITTLE
7367	063472					BGNSEG			
	063472	104404							TRAP C\$BSEG
7368	063474	005037	002400			CLR	INTFC2		:CLEAR INTERRUPT FLAG
7369	063500	005037	002376			CLR	INTFC1		:CLEAR INTERRUPT FLAG
7370	063504	052777	000010	116566		BIS	#10,@CSRX		:SELECT CHANNEL 2
7371	063512	013777	002356	116562		MOV	BUFAB,@BARX		:---LOAD START ADDRESS OF TABLE A ----
7372	063520	013737	002240	002402		MOV	BCINP,RSAVE		:STORE BYTE COUNT INPUT
7373	063526	005437	002402			NEG	RSAVE		:BILD 2'COMPL FOR BCR
7374	063532	013777	002402	116544		MOV	RSAVE,@BCRX		:---LOAD INPUT INTO BCR 2-----
7375	063540	012777	000107	116532		MOV	#107,@CSRX		:---DMA ENB,DMA DIR,INT ENB,SYS CON.
7376	063546	042777	000010	116524		BIC	#10,@CSRX		:SELECT CHANNEL 1
7377	063554	013700	002360			MOV	BUFBB,RO		:LOAD START ADDRESS OF BUFFER B
7378	063560	012701	010000			MOV	#10000,R1		:BUILD 2K BUFFER SIZE
7379	063564	105020			2\$:	CLRB	(R0)+		:FILL BUFFER B WITH ZERO
7380	063566	005301				DEC	R1		:2K LOADED
7381	063570	001375				BNE	2\$		:IF NO, CLEAR NEXT BUFFER LOCATION
7382	063572	013777	002360	116502		MOV	BUFBB,@BARX		:---LOAD START ADDRESS OF TABLE B----
7383	063600	013777	002402	116476		MOV	RSAVE,@BCRX		:---LOAD INPUT INTO BCR 1---
7384	063606	012777	000101	116464		MOV	#101,@CSRX		:---SET DMA ENB,INT ENB IN CSR1-----
7385	063614					SETPRI	#PRI00		:SET PRIORITY TO ZERO
	063614	012700	000000						MOV #PRI00,RO
	063620	104441							TRAP C\$SPRI
7386	063622	052777	000010	116450		BIS	#10,@CSRX		:SELECT CHANNEL 2
7387	063630	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
7388	063636	112777	000013	116424		MOVB	#13,@ICRHX		:---LOAD GTS INTO ACR 2-----
7389	063644	012701	077777			MOV	#77777,R1		:LOAD LOOP COUNTER
7390	063650	005737	002400		13\$:	TST	INTFC2		:IS INTERRUPT IN CHANNEL 2 OCCER
7391	063654	001015				BNE	10\$		:BRANCH IF YES
7392	063656	005301				DEC	R1		:DECREMENT COUNTER
7393	063660	001373				BNE	13\$		:IF MORE, TEST AGAIN
7394	063662	017737	116412	002502		MOV	@CSRX,BAD		:GET CSR2 CONTENTS
7395	063670	012737	100016	002500		MOV	#100016,GOOD		:BC OF ,MUX,DMA DTR,SYS CONT SHOULD SET
7396	063676					ERRSOFT	2401,E232,ERR201		:ERROR HANDLER
	063676	104457							TRAP C\$ERSOFT
	063700	004541							.WORD 2401
	063702	006072							.WORD E232
	063704	003500							.WORD ERR201
7397	063706					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	063706	104406							TRAP C\$CLP1
7398	063710	042777	000010	116362	10\$:	BIC	#10,@CSRX		:SELECT CHANNEL 2
7399	063716	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
7400	063724	005737	002376			TST	INTFC1		:IS AN INTERRUPT IN CHANNEL 1 OCCURED
7401	063730	001013				BNE	11\$		:BRANCH IF YES
7402	063732	017737	116342	002502		MOV	@CSRX,BAD		:GET CSR1 CONTENTS
7403	063740	012737	100000	002500		MOV	#100000,GOOD		:BC OF SHUOLD BE SET
7404	063746					ERRSOFT	2402,E232,ERR201		:ERROR HANDLER
	063746	104457							TRAP C\$ERSOFT
	063750	004542							.WORD 2402
	063752	006072							.WORD E232
	063754	003500							.WORD ERR201
7405	063756					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	063756	104406							TRAP C\$CLP1
7406	063760	052777	000010	116312	11\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2
7407	063766	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
7408	063774					SETPRI	#PRI07		:NO FURTHER INTERRUPT ALLOWED





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-3  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

```

7444 064246 005037 002500          CLR    GOOD          ;CLEAR GOOD
7445 064252 005037 002502          CLR    BAD           ;CLEAR BAD
7446 064256 005037 002426          CLR    TXADRH        ;SET UP DATA FOR ERROR MESSAGE
7447 064262 010137 002430          MOV    R1,TXADRL     ;...
7448 064266 005337 002430          DEC    TXADRL        ;...
7449 064272 005037 002422          CLR    RXADRH        ;...
7450 064276 010237 002424          MOV    R2,RXADRL    ;...
7451 064302 005337 002424          DEC    RXADRL        ;...
7452 064306 116137 177777 002500  MOVB   -1(R1),GOOD   ;SET UP DATA FOR ERROR MESSAGE
7453 064314 116237 177777 002502  MOVB   -1(R2),BAD    ;...
7454 064322          ERRSOFT 2407,E250,ERR231 ;ERROR HANDLER
          064322 104457          TRAP   C$ERSOFT
          064324 004547          .WORD 2407
          064326 005726          .WORD E250
          064330 003744          .WORD ERR231
7455 064332          CKLOOP          ;BRANCH TO BGNSEG IF ERRLOOP IS SET
          064332 104406          TRAP   C$CLP1
7456 064334 023737 002404 002240 46$:  CMP    CNT1,BCINP    ;ALL BYTES COMPARED ?
7457 064342 001335          BNE    44$          ;IF NO, GET NEXT ONE
7458 064344          ENDSEG
          064344 104405          10000$: TRAP   C$ESEG
7459 064346 005077 115726          CLR    @CSRX        ;CLEAR CSR1,
7460 064352 017737 115714 002502  MOV    @IDRX,BAD     ;READ DIR1 FOR CLEAR BO BIT IN IIR
7461 064360 052777 000010 115712  BIS    #10,@CSRX    ;SELECT CHANNEL 2
7462 064366 005077 115706          CLR    @CSRX        ;CLEAR CSR2,SELECT CHANNEL 1
7463          ;+-----+
7464          ;PART 2 CHECK THE NON EXISTENT MEMORY BIT OF CHAN. 1 (THE I/O PAGE IS USED
7465          ;FOR NON EXISTENT MEMORY)
7466          ;+-----+
7467 064372 012701 002624  PSEU24: MOV    #TABEL,R1 ;CLEAR TWO WORD IN TABEL E
7468 064376 005021          CLR    (R1)+        ;...
7469 064400 005011          CLR    (R1)         ;...
7470 064402          SETVEC #4,#NXM,#PRI07 ;SET UP VECTOR FOR TRAP TO 4
          064402 012746 000340          MOV    #PRI07,-(SP)
          064406 012746 010132          MOV    #NXM,-(SP)
          064412 012746 000004          MOV    #4,-(SP)
          064416 012746 000003          MOV    #3,-(SP)
          064422 104437          TRAP   C$SVEC
          064424 062706 000010          ADD    #10,SP
7471 064430 005037 002326          CLR    NXMFLG       ;CLEAR FLAG
7472 064434 012701 160000          MOV    #160000,R1   ;LOAD FIRST ADDRESS OF I/O PAGE
7473 064440 005711          TST    (R1)         ;FIND A NON EXISTEND LOCATION
7474 064442 005737 002326          TST    NXMFLG       ;...
7475 064446 062701 000002          ADD    #2,R1        ;NEXT I/O PAGE ADD.(NO AUTO INCR.11/44!)
7476 064452 001772          BEQ    1$          ;...
7477 064454          CLRVEC #4          ;SET VECTOR 4 TO NORMAL STATE
          064454 012700 000004          MOV    #4,R0
          064460 104436          TRAP   C$CVEC
7478 064462 162701 000002          SUB    #2,R1        ;...
7479 064466          BGNSEG
          064466 104404          TRAP   C$BSEG
7480 064470 005037 002376          CLR    INTFC1       ;CLEAR INTERRUPT FLAG
7481 064474 005037 002400          CLR    INTFC2       ;CLEAR INTERRUPT FLAG
7482 064500 042777 000010 115572  BIC    #10,@CSRX    ;SELECT CHANNEL 1
7483 064506 010177 115570          MOV    R1,@BARX     ;LOAD BAR1 WITH A NON EXISTS I/O ADDR.
7484 064512 012737 000002 002402  MOV    #2,RSAVE     ;LOAD NUMBER OF BYTE COUNTS
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-4  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

```

7485 064520 005437 002402          NEG      RSAVE
7486 064524 013777 002402 115552    MOV      RSAVE,@BCRX
7487 064532 052777 000010 115540    BIS      #10,@CSRX
7488 064540 012777 000117 115532    MOV      #117,@CSRX
7489 064546 012777 002624 115526    MOV      #TABE,@BARX
7490 064554 013777 002402 115522    MOV      RSAVE,@BCRX
7491 064562 042777 000010 115510    BIC      #10,@CSRX
7492 064570 012777 017161 115502    MOV      #17161,@CSRX
7493 064576 052777 000010 115474    BIS      #10,@CSRX
7494 064604 012737 000002 002374    MOV      #2,CHAN
7495 064612          SETPRI  #PRI00
      064612 012700 000000
      064616 104441
7496 064620 112777 000013 115442    MOV      #13,@ICRH
7497 064626 012702 077777
7498 064632 005737 002400 3$:      TST      INTFC2
7499 064636 001015          BNE      6$
7500 064640 005302          DEC      R2
7501 064642 001373          BNE      3$
7502 064644 017737 115430 002502    MOV      @CSRX,BAD
7503 064652 012737 100016 002500    MOV      #100016,GOOD
7504 064660          ERRSOFT 2408,E232,ERR201
      064660 104457
      064662 004550
      064664 006072
      064666 003500
7505 064670          CKLOOP
      064670 104406
7506 064672          6$:      SETPRI  #PRI07
      064672 012700 000340
      064676 104441
7507 064700 112777 000014 115362    MOV      #14,@ICRH
7508 064706 042777 000010 115364    BIC      #10,@CSRX
7509 064714 012737 000001 002374    MOV      #1,CHAN
7510 064722 017737 115352 002502    MOV      @CSRX,BAD
7511 064730 042737 017000 002502    BIC      #17000,BAD
7512 064736 022737 040060 002502    CMP      #40060,BAD
7513 064744 001410          BEQ      12$
7514 064746 012737 040060 002500    MOV      #40060,GOOD
7515 064754          ERRSOFT 2409,E401,ERR501
      064754 104457
      064756 004551
      064760 005265
      064762 003702
7516 064764          CKLOOP
      064764 104406
7517 064766          12$:     ENDSEG
      064766
      064766 104405
7518 064770 005077 115304          CLR      @CSRX
7519 064774 017737 115272 002502    MOV      @IDRX,BAD
7520 065002 052777 000010 115270    BIS      #10,@CSRX
7521 065010 005077 115264          CLR      @CSRX
7522          :+++-----
7523          :PART 3 DMA OVER 32K
7524          :+++-----
7525 065014 005037 002376          CLR      INTFC1
    
```

```

:BILD 2'COMPLEMENT
:LOAD BCR1
:SELECT CHANNEL 2
:DMA ENB,DMA DIR,INT ENB,SYS
:LOAD BAR2 WITH STAR ADDRESS OF TABE
:LOAD BCR2
:SELECT CHANNEL 1
:---SET DMA ENB,BA 16-21,INT ENB CSR1-
:SELECT CHANNEL 2
:LOAD CHANNEL NUMBER
:SET PRIORITY TO ZERO
      MOV      #PRI00,R0
      TRAP    C$SPRI
:---LOAD GTS INTO ACR 2-----
:LOAD LOOP COUNTER
:IS INTERRUPT IN CHANNEL 2 OCCER
:BRANCH IF YES
:DECREMENT COUNTER
:IF NO,TEST AGIN
:GET CSR2 CONTENTS
:BC OF ,MUX,DMA DIR,SYS CONT SHOULD SET
:ERROR HANDLER
      TRAP    C$ERSOFT
      .WORD  2408
      .WORD  E232
      .WORD  ERR201
:BRANCH TO BGNSEG IF ERRLOOP IS SET
      TRAP    C$CLP1
:NO FURTHER INTERRUPT ALLOWED
      MOV      #PRI07,R0
      TRAP    C$SPRI
:---LOAD TCA INTO ACR 2-----
:SELECT CHANNEL 1
:LOAD CHANNEL NUMBER
:GET CSR1 CONTENTS
:IGNORE BIT 9-12
:NXM,BA 16+17 SHOULD BE SET
:BRANCH IF YES
:SET UP DATA FOR ERROR MESSAGE
:ERROR HANDLER
      TRAP    C$ERSOFT
      .WORD  2409
      .WORD  E401
      .WORD  ERR501
:BRANCH TO BGNSEG IF ERRLOOP IS SET
      TRAP    C$CLP1
      10001$: TRAP    C$ESEG
:CLEAR CSR1
:READ DIR1 FOR CLEAR BO BIT IN IIR
:SELECT CHANNEL 2
:CLEAR CSR2,SELECT CHANNEL 1
:CLEAR INTERRUPT FLAG
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-5  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

7526	065020	005037	002400		CLR	INTFC2	:CLEAR INTERRUPT FLAG
7527	065024	023727	002342	000001	CMP	PHHSIZ,#1	:IS THERE MORE THAN 32K
7528	065032	002404			BLT	11\$	:IF NO SKIP TEST
7529	065034	003005			BGT	13\$	
7530	065036	005737	002344		TST	PHLSIZ	
7531	065042	001002			BNE	13\$	
7532	065044	000137	067200	11\$:	JMP	EXQV24	:JUMP TO TEST END
7533	065050	005037	002336	13\$:	CLR	PHLOW	:LOAD INPUT FOR CONVERSION ROUTINE
7534	065054	012737	000001	002334	MOV	#1,PHHIGH	
7535	065062	004737	007724		JSR	PC,PVCON	:CREATE VIRTUAL ADDRESS
7536	065066	013700	002340		MOV	VIADD,R0	:GET START ADDRESS OF BUFFER A
7537	065072	010037	002356		MOV	R0,BUFAB	:LOAD START ADDRESS OF BUFFER A
7538	065076	012701	010000		MOV	#10000,R1	:LOAD COUNTER FOR 2K
7539	065102	005003			CLR	R3	:R3 CONTAINS
7540	065104	012737	000001	177572	MOV	#1,SRO	:**ENABLE MEMORY MANAGEMENT**
7541	065112	110320		16\$:	MOVB	R3,(R0)+	:LOAD BUFFER WITH DATA
7542	065114	005203			INC	R3	:CREAT NEXT DATA
7543	065116	005301			DEC	R1	
7544	065120	001374			BNE	16\$	
7545	065122	012701	010000		MOV	#10000,R1	:LOAD 2K
7546	065126	010037	002360		MOV	R0,BUFBB	:LOAD START ADDRESS OF BUFFER B
7547	065132	105020		20\$:	CLRB	(R0)+	:CLEAR BUFFER B
7548	065134	005301			DEC	R1	:2K CLEARED
7549	065136	001375			BNE	20\$	:IF YES,DO THE TEST
7550	065140	052777	000010	115132	BIS	#10,@CSRX	:SELECT CHANNEL 2
7551	065146	005037	177572		CLR	SRO	:**DISABLE MEMORY MANAGEMENT**
7552	065152	112777	000212	115110	MOVB	#212,@ICRHX	:-----LOAD TON INTO ACR 2-----
7553	065160	013701	002312		MOV	DPA1,R1	:CREATE MLA1
7554	065164	062701	000040		ADD	#40,R1	
7555	065170	010137	002410		MOV	R1,MLA1	:STORE MLA1
7556	065174	113777	002410	115074	MOVB	MLA1,@IDRHX	:-----LOAD MLA1 INTO DOR 2-----
7557	065202	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
7558	065206				BGNSEG		
	065206	104404					TRAP C\$BSEG
7559	065210	052777	000010	115062	BIS	#10,@CSRX	:SELECT CHANNEL 2
7560	065216	005077	115060		CLR	@BARX	:---LOAD START ADDRESS OF BUFFER A---
7561	065222	013737	002240	002402	MOV	BCINP,RSAVE	:STORE BYTE COUNT INPUT
7562	065230	005437	002402		NEG	RSAVE	:BILD 2'COMPL FOR BCR
7563	065234	013777	002402	115042	MOV	RSAVE,@BCRX	:---LOAD INPUT INTO BCR 2-----
7564	065242	012777	000127	115030	MOV	#127,@CSRX	:DMA ENB,DMA DIR,INT ENB,BIT 16,SYS C
7565	065250	042777	000010	115022	BIC	#10,@CSRX	:SELECT CHANNEL 1
7566	065256	012777	010000	115016	MOV	#10000,@BARX	:---LOAD START ADDRESS OF BUFFER B---
7567	065264	013777	002402	115012	MOV	RSAVE,@BCRX	:---LOAD INPUT INTO BCR 1---
7568	065272	012777	000121	115000	MOV	#121,@CSRX	:---SET DMA ENB,INT ENB,BIT16 IN CSR1
7569	065300				SETPRI	#PRI00	:SET PRIORITY TO ZERO
	065300	012700	000000				MOV #PRI00,R0
	065304	104441					TRAP C\$SPRI
7570	065306	052777	000010	114764	BIS	#10,@CSRX	:SELECT CHANNEL 2
7571	065314	012737	000002	002374	MOV	#2,CHAN	:LOAD CHANNEL NUMBER
7572	065322	112777	000013	114740	MOVB	#13,@ICRHX	:---LOAD GTS INTO ACR 2-----
7573	065330	012701	077777		MOV	#77777,R1	:LOAD LOOP COUNTER
7574	065334	005737	002400	23\$:	TST	INTFC2	:IS INTERRUPT IN CHANNEL 2 OCCER
7575	065340	001015			BNE	24\$	:BRANCH IF YES
7576	065342	005301			DEC	R1	:DECREMENT COUNTER
7577	065344	001373			BNE	23\$	:IF NO,TEST AGIN
7578	065346	017737	114726	002502	MOV	@CSRX,BAD	:GET CSR2 CONTENTS
7579	065354	012737	100036	002500	MOV	#100036,GOOD	:BC OF,MUX,DMA DIR,BA 16,SYS CONT

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-6  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

7580	065362				ERRSOFT 2410,E232,ERR201		:ERROR HANDLER		
	065362	104457						TRAP	C\$ERSOFT
	065364	004552						.WORD	2410
	065366	006072						.WORD	E232
	065370	003500						.WORD	ERR201
7581	065372				CKLOOP		:BRANCH TO BGNSEG IF ERRLOOP IS SET	TRAP	C\$CLP1
	065372	104406							
7582	065374	042777	000010	114676	24\$: BIC #10,@CSRX		:SELECT CHANNEL 1		
7583	065402	012737	000001	002374	MOV #1,CHAN		:LOAD CHANNEL NUMBER		
7584	065410	005737	002376		TST INTFC1		:IS AN INTERRUPT IN CHANNEL 1 OCCURED		
7585	065414	001013			BNE 25\$		:BRANCH IF YES		
7586	065416	017737	114656	002502	MOV @CSRX,BAD		:GET CSR1 CONTENTS		
7587	065424	012737	100020	002500	MOV #100020,GOOD		:BC OF,BA 16 SHOULD BE SET		
7588	065432				ERRSOFT 2411,E232,ERR201		:ERROR HANDLER		
	065432	104457						TRAP	C\$ERSOFT
	065434	004553						.WORD	2411
	065436	006072						.WORD	E232
	065440	003500						.WORD	ERR201
7589	065442				CKLOOP		:BRANCH TO BGNSEG IF ERRLOOP IS SET	TRAP	C\$CLP1
	065442	104406							
7590	065444				25\$: SETPRI #PRI07		:NO FURTHER INTERRUPT ALLOWED		
	065444	012700	000340					MOV	#PRI07,RO
	065450	104441						TRAP	C\$SPRI
7591	065452	052777	000010	114620	BIS #10,@CSRX		:SELECT CHANNEL 2		
7592	065460	012737	000002	002374	MOV #2,CHAN		:LOAD CHANNEL NUMBER		
7593	065466	112777	000014	114574	MOVB #14,@ICRHX		:----LOAD TCA INTO ACR 2-----		
7594	065474	013737	002336	002500	MOV PHLOW,GOOD		:SET UP COMPARE VALUE		
7595	065502	063737	002240	002500	ADD BCINP,GOOD				
7596	065510	017737	114566	002502	MOV @BARX,BAD		:GET BAR2 CONTENTS		
7597	065516	023737	002502	002500	CMP BAD,GOOD		:HAS BAR2 THE CORRECT ADDRESS		
7598	065524	001404			BEQ 30\$		:BRANCH IF YES		
7599	065526				ERRSOFT 2412,E234,ERR501		:ERROR HANDLER		
	065526	104457						TRAP	C\$ERSOFT
	065530	004554						.WORD	2412
	065532	006200						.WORD	E234
	065534	003702						.WORD	ERR501
7600	065536	017737	114542	002502	30\$: MOV @BCRX,BAD		:GET BCR2 CONTENTS		
7601	065544	005037	002500		CLR GOOD				
7602	065550	023737	002502	002500	CMP BAD,GOOD		:IS BCR2 ZERO		
7603	065556	001404			BEQ 31\$		:BRANCH IF YES		
7604	065560				ERRSOFT 2413,E235,ERR501		:ERROR HANDLER		
	065560	104457						TRAP	C\$ERSOFT
	065562	004555						.WORD	2413
	065564	006231						.WORD	E235
	065566	003702						.WORD	ERR501
7605	065570	042777	000010	114502	31\$: BIC #10,@CSRX		:SELECT CHANNEL 1		
7606	065576	012737	000001	002374	MOV #1,CHAN		:LOAD CHANNEL NUMBER		
7607	065604	013737	002360	002500	MOV BUFBB,GOOD		:SET UP COMPARE VALUE		
7608	065612	042737	160000	002500	BIC #160000,GOOD		:CLEAR BIT 13+14+15		
7609	065620	063737	002240	002500	ADD BCINP,GOOD				
7610	065626	017737	114450	002502	MOV @BARX,BAD		:GET BAR1 CONTENTS		
7611	065634	023737	002500	002502	CMP GOOD,BAD		:HAS BAR1 THE CORRECT ADDRESS		
7612	065642	001404			BEQ 40\$		:BRANCH IF YES		
7613	065644				ERRSOFT 2414,E234,ERR501		:ERROR HANDLER		
	065644	104457						TRAP	C\$ERSOFT
	065646	004556						.WORD	2414
	065650	006200						.WORD	E234



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-7  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

```

7614 065652 003702
7614 065654 017737 114424 002502 40$: MOV @BCRX,BAD ;GET BCR1 CONTENTS .WORD ERR501
7615 065662 005737 002502 TST BAD ;BCR1 CONTENTS SHOULD BE ZERO
7616 065666 001406 BEQ 43$ ;BRANCH IF YES
7617 065670 005037 002500 CLR GOOD ;SET UP DATA FOR ERROR MESSAGE
7618 065674 005037 002500 ERRSOFT 2415,E235,ERR501 ;ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 2415
                                .WORD E235
                                .WORD ERR501
7619 065704 012737 000001 177572 43$: MOV #1,SRO ;**ENABLE MEMORY MANAGEMENT**
7620 065712 013701 002356 MOV BUFAB,R1 ;PROVIDE FIRST BYTE OF BUFFER A
7621 065716 013702 002360 MOV BUFBB,R2 ;PROVIDE FIRST BYTE OF BUFFER B
7622 065722 005037 002404 CLR CNT1 ;CLEAR BUFFER COUNTER
7623 065726 005237 002404 44$: INC CNT1 ;
7624 065732 122122 CMPB (R1)+,(R2)+ ;BUFFER A EQUAL BUFFER B
7625 065734 001446 BEQ 46$ ;IF YES CONTINUE
7626 065736 005037 002500 CLR GOOD ;CLEAR GOOD
7627 065742 005037 002502 CLR BAD ;CLEAR BAD
7628 065746 012737 000001 002426 MOV #1, TXADRH ;TX ADDRESS IS OVER 32K
7629 065754 012737 000001 002422 MOV #1, RXADRH ;RX ADDRESS IS OVER 32K
7630 065762 010137 002430 MOV R1, TXADRL ;GET ADDRESS OVER 32K
7631 065766 005337 002430 DEC TXADRL ;
7632 065772 042737 160000 002430 BIC #160000, TXADRL ;CLEAR PAR INFORMATION
7633 066000 013737 002430 002424 MOV TXADRL, RXADRL ;GENERATE RX ADDRESS FROM TX ADDRESS
7634 066006 052737 010000 002424 BIS #10000, RXADRL ;RX ADDRRES IS TX ADDRESS +2K
7635 066014 116137 177777 002500 MOVB -1(R1),GOOD ;SET UP DATA FOR ERROR MESSAGE
7636 066022 116237 177777 002502 MOVB -1(R2),BAD ;
7637 066030 005037 177572 CLR SRO ;**DISABLE MEMORY MANAGEMENT**
7638 066034 005037 177572 ERRSOFT 2416,E250,ERR231 ;ERROR HANDLER
                                TRAP C$ERSOFT
                                .WORD 2416
                                .WORD E250
                                .WORD ERR231
7639 066044 012737 000001 177572 46$: MOV #1,SRO ;**ENABLE MEMORY MANAGEMENT**
7640 066052 023737 002404 002240 CMP CNT1,BCINP ;ALL BYTES COMPARED ?
7641 066060 001322 BNE 44$ ;IF NO, GET NEXT ONE
7642 066062 005037 177572 CLR SRO ;**DISABLE MEMORY MANAGEMENT**
7643 066066 005037 177572 ENDSEG
                                10002$: TRAP C$ESEG
7644 066070 005077 114204 CLR @CSRX ;CLEAR CSR1
7645 066074 017737 114172 002502 MOV @IDRX,BAD ;READ DIR1 FOR CLEAR BO BIT IN IIR
7646 066102 052777 000010 114170 BIS #10,@CSRX ;SELECT CHANNEL 2
7647 066110 005077 114164 CLR @CSRX ;CLEAR CHANNEL 2,SELECT CHANNEL 1
7648
7649
7650
:+++-----:
:PART 4 DMA OVER 64K
:+++-----:
7651 066114 005037 002376 PSEU34: CLR INTFC1 ;CLEAR INTERRUPT FLAG
7652 066120 005037 002400 CLR INTFC2 ;CLEAR INTERRUPT FLAG
7653 066124 023727 002342 000002 CMP PHHSIZ,#2 ;IS THERE MORE THAN 64K
7654 066132 002404 BLT 11$ ;IF NO SKIP TEST
7655 066134 003005 BGT 13$ ;
7656 066136 005737 002344 TST PHLSIZ ;
7657 066142 001002 BNE 13$ ;
7658 066144 000137 067200 11$: JMP EXQV24 ;JUMP TO TEST END
7659 066150 005037 002336 13$: CLR PHLOW ;LOAD INPUT FOR CONVERSION ROUTINE
    
```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-8  
 TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

7660	066154	012737	000002	002334	MOV	#2,PHHIGH				
7661	066162	004737	007724		JSR	PC,PVCON				:CREATE VIRTUAL ADDRESS
7662	066166	013700	002340		MOV	VIADD,R0				:GET START ADDRESS OF BUFFER A
7663	066172	010037	002356		MOV	R0,BUFAB				:LOAD START ADDRESS OF BUFFER A
7664	066176	012701	010000		MOV	#10000,R1				:LOAD COUNTER FOR 2K
7665	066202	005003			CLR	R3				:R3 CONTAINS
7666	066204	012737	000001	177572	MOV	#1,SRO				:**ENABLE MEMORY MANAGEMENT**
7667	066212	110320			MOVB	R3,(R0)+				:LOAD BUFFER WITH DATA
7668	066214	005203			INC	R3				:CREAT NEXT DATA
7669	066216	005301			DEC	R1				
7670	066220	001374			BNE	16\$				
7671	066222	012701	010000		MOV	#10000,R1				:LOAD 2K
7672	066226	010037	002360		MOV	R0,BUFB				:LOAD START ADDRESS OF BUFFER B
7673	066232	105020			CLRB	(R0)+				:CLEAR BUFFER B
7674	066234	005301			DEC	R1				:2K CLEARED
7675	066236	001375			BNE	20\$				:IF YES,DO THE TEST
7676	066240	052777	000010	114032	BIS	#10,@CSRX				:SELECT CHANNEL 2
7677	066246	005037	177572		CLR	SRO				:**DISABLE MEMORY MANAGEMENT**
7678	066252	112777	000212	114010	MOVB	#212,@ICRHX				:----LOAD TON INTO ACR 2-----
7679	066260	013701	002312		MOV	DPA1,R1				:CREATE MLA1
7680	066264	062701	000040		ADD	#40,R1				
7681	066270	010137	002410		MOV	R1,MLA1				:STORE MLA1
7682	066274	113777	002410	113774	MOVB	MLA1,@IDRHX				:----LOAD MLA1 INTO DOR 2-----
7683	066302	004737	011060		JSR	PC,LOOP				:WAIT A LITTLE
7684	066306	005077	113770		CLR	@BARX				:----LOAD START ADDRESS OF BUFFER A----
7685	066312	013737	002240	002402	MOV	BCINP,RSAVE				:STORE BYTE COUNT INPUT
7686	066320	005437	002402		NEG	RSAVE				:BILD 2*COMPL FOR BCR
7687	066324	013777	002402	113752	MOV	RSAVE,@BCRX				:----LOAD INPUT INTO BCR 1-----
7688	066332	012777	000147	113740	MOV	#147,@CSRX				:DMA ENB,DMA DIR,INT ENB,BA 17,SYS C
7689	066340	042777	000010	113732	BIC	#10,@CSRX				:SELECT CHANNEL 1
7690	066346	012777	010000	113726	MOV	#10000,@BARX				:----LOAD START ADDRESS OF BUFFER B---
7691	066354	013777	002402	113722	MOV	RSAVE,@BCRX				:----LOAD INPUT INTO BCR 1---
7692	066362	012777	000141	113710	MOV	#141,@CSRX				:----SET DMA ENB,INT ENB,BIT17 IN CSR1
7693	066370				SETPRI	#PRI00				:SET PRIORITY TO ZERO
	066370	012700	000000							MOV #PR:100,R0
	066374	104441								TRAP C\$SPRI
7694	066376	052777	000010	113674	BIS	#10,@CSRX				:SELECT CHANNEL 2
7695	066404	012737	000002	002374	MOV	#2,CHAN				:LOAD CHANNEL NUMBER
7696	066412	112777	000013	113650	MOVB	#13,@ICRHX				:----LOAD GTS INTO ACR 2-----
7697	066420	012701	077777		MOV	#77777,R1				:LOAD LOOP COUNTER
7698	066424	005737	002400		TST	INTFC2				:IS INTERRUPT IN CHANNEL 2 OCCER
7699	066430	001014			BNE	24\$				:BRANCH IF YES
7700	066432	005301			DEC	R1				:DECREMENT COUNTER
7701	066434	001373			BNE	23\$				:IF NO,TEST AGIN
7702	066436	017737	113636	002502	MOV	@CSRX,BAD				:GET CSR2 CONTENTS
7703	066444	012737	100056	002500	MOV	#100056,GOOD				:BC OF,MUX,DMA DIR,BA 17,SYS CONT
7704	066452				ERRSOFT	2417,E232,ERR201				:ERROR HANDLER
	066452	104457								TRAP C\$ERSOFT
	066454	004561								.WORD 2417
	066456	006072								.WORD E232
	066460	003500								.WORD ERR201
7705	066462	042777	000010	113610	BIC	#10,@CSRX				:SELECT CHANNEL 1
7706	066470	012737	000001	002374	MOV	#1,CHAN				:LOAD CHANNEL NUMBER
7707	066476	005737	002376		TST	INTFC1				:IS AN INTERRUPT IN CHANNEL 1 OCCURED
7708	066502	001012			BNE	25\$				:BRANCH IF YES
7709	066504	017737	113570	002502	MOV	@CSRX,BAD				:GET CSR1 CONTENTS
7710	066512	012737	100040	002500	MOV	#100040,GOOD				:BC OF,BA 17 SHOULD BE SET





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 66-10  
TEST 24: DMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1

```

7746 067016 122122          CMPB   (R1)+,(R2)+      ;BUFFER A EQUAL BUFFER B
7747 067020 001446          BEQ    46$              ;IF YES CONTINUE
7748 067022 005037 002500    CLR    GOOD            ;CLEAR GOOD
7749 067026 005037 002502    CLR    BAD             ;CLEAR BAD
7750 067032 012737 000002 002426  MOV    #2,TXADRH       ;TX ADDRESS IS OVER 64K
7751 067040 012737 000002 002422  MOV    #2,RXADRH       ;RX ADDRESS IS OVER 64K
7752 067046 010137 002430    MOV    R1,TXADRL      ;GET ADDRESS OVER 64K
7753 067052 005337 002430    DEC    TXADRL         ;
7754 067056 042737 160000 002430  BIC    #160000,TXADRL  ;CLEAR PAR INFORMATION
7755 067064 013737 002430 002424  MOV    TXADRL,RXADRL  ;GENERATE RX ADDRESS FROM TX ADDRESS
7756 067072 052737 010000 002424  BIS    #10000,RXADRL  ;RX ADDRESS IS TX ADDRESS +2K
7757 067100 116137 177777 002500  MOVB  -1(R1),GOOD     ;SET UP DATA FOR ERROR MESSAGE
7758 067106 116237 177777 002502  MOVB  -1(R2),BAD     ;
7759 067114 005037 177572    CLR    SRO            ;**DISABLE MEMORY MANAGEMENT**
7760 067120          ERRSOFT 2423,E250,ERR231 ;ERROR HANDLER
          067120 104457          TRAP   C$ERSOFT
          067122 004567          .WORD 2423
          067124 005726          .WORD E250
          067126 003744          .WORD ERR231
7761 067130 012737 000001 177572  MOV    #1,SRO         ;**ENABLE MEMORY MANAGEMENT**
7762 067136 023737 002404 002240 46$:  CMP    CNT1,BCINP     ;ALL BYTES COMPARED ?
7763 067144 001322          BNE    44$            ;IF NO, GET NEXT ONE
7764 067146 005037 177572    CLR    SRO            ;**DISABLE MEMORY MANAGEMENT**
7765 067152 005737 002234    TST   QVP            ;IS QUICK VERIFY PASS SELECTED
7766 067156 001010          BNE    EXQV24        ;IF YES EXIT TEST
7767 067160 005237 002322          INC    ITRCNT        ;INCREMENT COUNTER
7768 067164 023737 002322 002320  CMP    ITRCNT,ITRDEF  ;ALL DONE
7769 067172 001402          BEQ    EXQV24        ;BRANCH IF YES
7770 067174 000137 063322          JMP    ITAC24        ;IF NO TEST ITERATION
7771 067200          EXQV24: EXIT      TST   ;EXIT TEST
          067200 104432          TRAP   C$EXIT
          067202 000064          .WORD L10055-.
7772
7773
7774 067204 045 123 062 TSHD24: .NLIST BEX
          .ASCIZ /%S2%ADMA DATA TRANSFER TEST FROM CHANNEL 2 TO 1%/
7775          .LIST BEX
7776          .EVEN
7777 067266          .ENDTST
          067266          L10055:
          067266 104401          TRAP   C$ETST

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 67  
TEST 25: MCR FUNCTION TEST OF CHANNEL 1

```

7779
7780
7781
7782
7783
7784
7785
7786
7787
7788
7789
7790
7791 067270
067270
7792 067270 005737 002324
7793 067274 001410
7794 067276
067276 012746 070742
067302 012746 000001
067306 010600
067310 104417
067312 062706 000004
7795 067316 005037 002322
7796 067322 004737 010710
7797 067326
067326 012746 000340
067332 012746 010152
067336 013746 002246
067342 012746 000003
067346 104437
067350 062706 000010
7798 067354
067354 012746 000340
067360 012746 010142
067364 013746 002244
067370 012746 000003
067374 104437
067376 062706 000010
7799 067402 112777 000212 112660
7800 067410 013701 002312
7801 067414 062701 000040
7802 067420 010137 002410
7803 067424 113777 002410 112644
7804 067432 004737 011060
7805 067436 042777 000010 112634
7806 067444 005077 112636
7807 067450 113777 002236 112632
7808 067456 052777 100177 112622
7809 067464 052777 000010 112606
7810 067472 012777 002734 112602
7811 067500 013737 002236 002402
7812 067506 062737 000011 002402
7813 067514 005437 002402
7814 067520 013777 002402 112556
7815 067526 005437 002402
7816 067532 012777 000107 112540
7817 067540

.SBTTL TEST 25: MCR FUNCTION TEST OF CHANNEL 1
*****
IEX - TEST 25
PART 1 CHANNEL 2 TRANSMITS 9 DATA BYTES (50) AND THEN
A PREDEFINED QUANTITY (MC INPUT) OF SUCCESSIVE EOS CHARACTERS (177)
VIA THE IEC/IEEE BUS TO CHANNEL 2. AFTER RECEIVING THESE CHARACTERS
THE DMA DATA TRANSFER IS TERMINATED BY CHANNEL 2 (COMP END).
PART 2 SAME AS PART 1 EXCEPT THAT A WRONG QUANTITY OF SUCCESSIVE EOS
CHARACTERS ARE TRANSMITTED BEFORE THE CORRECT QUANTITY OF SUCCESSIVE
EOS CHARACTERS ARE TRANSMITTED.
I.E 2 EOS CHAR.(25), 1 DATA BYTE (50) THAN THE PREDEFINED EOS CHAR.(25)
*****
BGNTST
T25::
TST PNTF ;IS THE PNT FLAG SET
BEQ 7$ ;IF YES, PRINT THE TEST HEADER
PRINTF #TSHD25 ;...
MOV #TSHD25,-(SP)
MOV #1,-(SP)
MOV SP,RO
TRAP C$PNTF
ADD #4,SP
7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
JSR PC,BGIN2 ;SET UP PARAMETER
ITAC25: SETVEC VECC2,#INTSC2,#PRI07 ;SET VECTOR FOR CHANNEL 2
MOV #PRI07,-(SP)
MOV #INTSC2,-(SP)
MOV VECC2,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP
SETVEC VECC1,#INTSC1,#PRI07 ;SET VECTOR FOR CHANNEL 1
MOV #PRI07,-(SP)
MOV #INTSC1,-(SP)
MOV VECC1,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP
7799 067402 112777 000212 112660 MOVB #212,@ICRHX ;----LOAD TON INTO ACR 2-----
MOV DPA1,R1 ;CREATE MLA1
ADD #40,R1
MOV R1,MLA1 ;STORE MLA1
MOVB MLA1,@IDRHX ;----LOAD MLA1 INTO DOR 2-----
JSR PC,LOOP ;WAIT A LITTLE
BIC #10,@CSRX ;SELECT CHANNEL 1
CLR @MCRX ;CLEAR THE MATCH CHARACTER REGISTER
MOVB MCINP,@MCRHX ;---LOAD CNT INPUT INTO MCR HIGH BYTE-
BIS #100177,@MCRX ;--- ENB MATCH BIT +EOS (177) IN MCR1---
BIS #10,@CSRX ;SELECT CHANNEL 2
MOV #TABF,@BARX ;---LOAD START ADDRESS OF TABLE F ---
MOV MCINP,RSVAVE ;STORE MATCH CHARACTER COUNT INPUT
ADD #11,RSVAVE ;ADD 11(OCTAL) DATA BYTES TO INPUT
NEG RSVAVE ;BILD 2'COMPL FOR BCR
MOV RSVAVE,@BCRX ;----LOAD INPUT+DATA BYTES INTO BCR 2-
NEG RSVAVE ;REBUILD INPUT + DATA FOR COUNTER
MOV #107,@CSRX ;----DMA ENB,DMA DIR,INT ENB IN CSR2--
BGNSEG

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 67-2  
TEST 25: MCR FUNCTION TEST OF CHANNEL 1

```

7860 070032 010137 002430      MOV      R1, TXADRL      ;...
7861 070036 005337 002430      DEC      TXADRL         ;...
7862 070042 005037 002422      CLR      RXADRH         ;...
7863 070046 010237 002424      MOV      R2, RXADRL     ;...
7864 070052 005337 002424      DEC      RXADRL         ;...
7865 070056 114137 002500      MOV      -(R1), GOOD    ;SET UP DATA FOR ERROR MESSAGE
7866 070062 114237 002502      MOV      -(R2), BAD    ;...
7867 070066 104457 002502      ERRSOFT 2503, E250, ERR231 ;ERROR HANDLER
                                TRAP      C$ERSOFT
                                .WORD    2503
                                .WORD    E250
                                .WORD    ERR231
                                TRAP      C$CLP1
7868 070076 104406 002404 002402 33$: CKLOOP      ;BRANCH TO BGNSEG IF ERRLOOP IS SET
                                TRAP      C$CLP1
7869 070100 023737 002404 002402 33$: CMP      CNT1, RSAVE    ;ALL BYTES COMPARED ?
7870 070106 001337 002404 002402 33$: BNE      30$          ;IF NOT, GET NEXT ONE
7871 070110 001337 002404 002402 33$: ENDSEG
                                10000$:
7872 070112 012777 000010 112160      MOV      #10, @CSRX     ;SELECT CHA. 2, CLR ALL BIT IN CHA.1
7873 070120 112777 000014 112142      MOV      #14, @ICRHX    ;---LOAD TCA INTO ACR 2---
7874 070126 017737 112140 002502      MOV      @IDRX, BAD     ;READ DIR2 REG. FOR CLEAR ACCRQ SIGNAL
7875
7876
7877
-----
:PART 2 CHECK THE FALSE NUMBER OF EOS CHARACTERS
-----
7878 070134 012777 000002 112136      MOV      #2, @CSRX     ;SELECT CHANNEL 1, SET SYS CON. IN CHA.2
7879 070142 005077 112140 112136      CLR      @MCRX         ;CLEAR MCR1 REGISTER
7880 070146 113777 002236 112134      MOV      MCINP, @MCRHX  ;---LOAD CNT INPUT INTO MCR 1---
7881 070154 052777 100025 112124      BIS      #100025, @MCRX ;---ENB MATCH + EOS CHARACTER (25)
7882 070162 052777 000010 112110      BIS      #10, @CSRX    ;SELECT CHANNEL 2
7883 070170 012777 003250 112104      MOV      #TABK, @BARX   ;---LOAD START ADDRESS OF BUFFER K---
7884 070176 013737 002236 002402      MOV      MCINP, RSAVE   ;STORE MATCH CHARACTER CNT INPUT
7885 070204 062737 000003 002402      ADD      #3, RSAVE     ;ADD 3 DATA BYTES TO CNT INPUT
7886 070212 005437 002402 112060      NEG      RSAVE         ;BILD 2' COMPL FOR BCR
7887 070216 013777 002402 112060      MOV      RSAVE, @BCRX  ;---LOAD BYTE COUNT REGISTER---
7888 070224 005437 002402 112042      NEG      RSAVE         ;REBUILD INPUT+ DATA FOR COUNTER CNT1
7889 070230 012777 000107 112042      MOV      #107, @CSRX   ;---DMA ENB, DMA DIR, INT ENB, SYS CON
7890 070236 104404 002400 112042      BGNSEG
                                TRAP      C$BSEG
7891 070240 005037 002400 112022      CLR      INTFC2        ;CLEAR INTERRUPT FLAG
7892 070244 005037 002376 112022      CLR      INTFC1        ;CLEAR INTERRUPT FLAG
7893 070250 042777 000010 112022      BIC      #10, @CSRX    ;SELECT CHANNEL 1
7894 070256 012700 000102 112022      MOV      #66, R0       ;CLEAR BUFFER TABH
7895 070262 012701 003146 112022      MOV      #TABH, R1     ;...
7896 070266 105021 003146 112022      CLR      (R1)+         ;...
7897 070270 105300 003146 112022      DECB    R0             ;...
7898 070272 001375 003146 112022      BNE      34$          ;...
7899 070274 012777 003146 112000      MOV      #TABH, @BARX  ;---LOAD START ADDRESS OF BUFFER H---
7900 070302 012777 177675 111774      MOV      #177675, @BCRX ;---LOAD MAX. TRANSFER INTO BCR 1---
7901 070310 012777 000101 111762      MOV      #101, @CSRX   ;---SET DMA ENB, INT ENB IN CSR1-
7902 070316 052777 000010 111754      BIS      #10, @CSRX    ;SELECT CHANNEL 2
7903 070324 012737 000002 002374      MOV      #2, CHAN     ;LOAD CHANNEL NUMBER
7904 070332 012700 000000 002374      SETPRI  #PRI00        ;SET PRI. TO ZERO (ALLOW INTERRUPT)
                                MOV      #PRI00, R0
                                TRAP      C$SPRI
7905 070340 112777 000013 111722      MOV      #13, @ICRHX   ;---LOAD GTS INTO ACR 2---
7906 070346 012701 007777 111722      MOV      #7777, R1    ;LOAD COUNTER

```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 67-4  
 TEST 25: MCR FUNCTION TEST OF CHANNEL 1

7945	070614	052777	000010	111456	BIS	#10,@CSRX	:SELECT CHANNEL 2
7946	070622	112777	000014	111440	MOVB	#14,@ICRHX	:---LOAD TCA INTO ACR 2-----
7947	070630	112777	000077	111440	MOVB	#77,@IDRHX	:---LOAD UNL INTO DOR 2-----
7948	070636	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
7949	070642	112777	000200	111420	MOVB	#200,@ICRHX	:---LOAD SWRST INTO ACR 2-----
7950	070650	112777	000000	111412	MOVB	#0,@ICRHX	:---LOAD NOT SWRST INTO ACR 2-----
7951	070656	005737	002234		TST	QVP	:IS QUICK VERIFY PASS SELECTED ?
7952	070662	001025			BNE	EXQV25	:IF YES EXIT TEST
7953	070664	005237	002322		INC	ITRCNT	:ITERATION COUNTER +1
7954	070670	023737	002322	002320	CMP	ITRCNT,ITRDEF	:DEFAULT ITERATION EXECUTED
7955	070676	001417			BEQ	EXQV25	:IF YES EXIT TEST
7956	070700	005077	111374		CLR	@CSRX	:SELECT CHAN.1,CLR ALL BITS IN CHAN.2
7957	070704	012777	000010	111366	MOV	#10,@CSRX	:CLEAR CSR1,SELECT CHANNEL 2
7958	070712	112777	000217	111350	MOVB	#217,@ICRHX	:---LOAD SIC INTO ACR 2-----
7959	070720	004737	011072		JSR	PC,WAIT	:WAIT A LITTLE
7960	070724	112777	000017	111336	MOVB	#17,@ICRHX	:---LOAD NOT SIC INTO ACR 2-----
7961	070732	000137	067326		JMP	ITAC25	:IF NO TEST ITERATION
7962	070736				EXQV25: EXIT	TST	
	070736	104432					TRAP C\$EXIT
	070740	000050					.WORD L10056-
7963							
7964							
7965	070742	045	123	062	TSHD25: .NLIST	BEX	
7966					.ASCIZ	/S2%AMCR FUNCTION TEST OF CHANNEL 1%N/	
7967					.LIST	BEX	
7968	071010				.EVEN		
	071010				ENDTST		
	071010	104401					L10056: TRAP C\$ETST



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 68  
TEST 26: MCR FUNCTION TEST OF CHANNEL 2

```

7970 .SBTTL TEST 26: MCR FUNCTION TEST OF CHANNEL 2
7971 *****
7972 IEX - TEST 26
7973 PART 1 CHANNEL 1 TRANSMITS 9 DATA BYTES (50) AND THEN
7974 A PREDEFINED QUANTITY (MC INPUT) OF SUCCESSIVE EOS CHARACTERS (177)
7975 VIA THE IEC/IEEE BUS TO CHANNEL 2. AFTER RECEIVING THESE CHARACTERS
7976 THE DMA DATA TRANSFER IS TERMINATED BY CHANNEL 2 (COMP END).
7977 PART 2 SAME AS PART 1 EXCEPT THAT A WRONG QUANTITY OF SUCCESSIVE EOS
7978 CHARACTERS ARE TRANSMITTED BEFORE THE CORRECT QUANTITY OF SUCCESSIVE
7979 EOS CHARACTERS ARE TRANSMITTED.
7980 I.E 2 EOS CHAR.(25), 1 DATA BYTE (50) THAN THE PREDEFINED EOS CHAR.(25)
7981 *****
7982 BGNTST
7983 071012 005737 002324 TST PNTF ;IS THE PNT FLAG SET
7984 071016 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
7985 071020 PRINTF #TSHD26 ;....
7986 071012 012746 072474 MOV #TSHD26,-(SP)
7987 071024 012746 000001 MOV #1,-(SP)
7988 071030 010600 MOV SP,R0
7989 071032 104417 TRAP C$PNTF
7990 071034 062706 000004 ADD #4,SP
7991 071040 005037 002322 7$: CLR ITRCNT ;CLEAR ITERATION COUNTER
7992 071044 004737 010534 JSR PC,BGIN1 ;SET UP PARAMETER
7993 071050 ITAC26: SETVEC VECC2,#INTSC2,#PRI07 ;SET VECTOR FOR CHANNEL 2
7994 071050 012746 000340 MOV #PRI07,-(SP)
7995 071054 012746 010152 MOV #INTSC2,-(SP)
7996 071060 013746 002246 MOV VECC2,-(SP)
7997 071064 012746 000003 MOV #3,-(SP)
7998 071070 104437 TRAP C$SVEC
7999 071072 062706 000010 ADD #10,SP
8000 071076 SETVEC VECC1,#INTSC1,#PRI07 ;SET VECTOR FOR CHANNEL 1
8001 071076 012746 000340 MOV #PRI07,-(SP)
8002 071102 012746 010142 MOV #INTSC1,-(SP)
8003 071106 013746 002244 MOV VECC1,-(SP)
8004 071112 012746 000003 MOV #3,-(SP)
8005 071116 104437 TRAP C$SVEC
8006 071120 062706 000010 ADD #10,SP
8007 071124 112777 000212 111136 MOVB #212,@ICRHX ;----LOAD TON INTO ACR 1-----
8008 071132 013701 002314 MOV DPA2,R1 ;CREATE MLA2
8009 071136 062701 000040 ADD #40,R1 ;
8010 071142 010137 002412 MOV R1,MLA2 ;STORE MLA2
8011 071146 113777 002412 111122 MOVB MLA2,@IDRHX ;----LOAD MLA2 INTO DOR 1-----
8012 071154 004737 011060 JSR PC,LOOP ;WAIT A LITTLE
8013 071160 052777 000010 111112 BIS #10,@CSRX ;SELECT CHANNEL 2
8014 071166 005077 111114 CLR @MCRX ;CLEAR MCR2 REGISTER
8015 071172 113777 002236 111110 MOVB MCINP,@MCRHX ;---LOAD CNT INPUT INTO MCR HIGH BYTE-
8016 071200 052777 100125 111100 BIS #100125,@MCRX ;--- ENB MATCH BIT +EOS (125) IN MCR2--
8017 071206 BGNSEG TRAP C$BSEG
8018 071210 005037 002400 CLR INTFC2 ;CLEAR INTERRUPT FLAG
8019 071214 005037 002376 CLR INTFC1 ;CLEAR INTERRUPT FLAG
8020 071220 042777 000010 111052 BIC #10,@CSRX ;SELECT CHANNEL 1
8021 071226 012777 002514 111046 MOV #TABD,@BARX ;---LOAD START ADDRESS OF TABLE D ---
8022 071234 013737 002236 002402 MOV MCINP,RSAVE ;STORE MATCH CHARACTER COUNT INPUT
8023 071242 062737 000011 002402 ADD #11,RSAVE ;ADD 11(OCTAL) DATA BYTES TO INPUT
8024 071250 005437 002402 NEG RSAVE ;BILD 2'COMPL FOR BCR

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 68-1  
 TEST 26: MCR FUNCTION TEST OF CHANNEL 2

8008	071254	013777	002402	111022	MOV	RSAVE,@BCRX	----	LOAD INPUT+DATA BYTES INTO BCR 1-
8009	071262	005437	002402		NEG	RSAVE	----	REBUILD INPUT+DATA FOR COUNTER CNT1
8010	071266	012777	000107	111004	MOV	#107,@CSRX	----	DMA ENB,DMA DIR,INT ENB,SYS CONT
8011	071274	052777	000010	110776	BIS	#10,@CSRX	----	SELECT CHANNEL 2
8012	071302	012700	000110		MOV	#72,,R0	----	CLEAR BUFFER TABE
8013	071306	012701	002624		MOV	#TABE,R1	----	----
8014	071312	105021		2\$:	CLRB	(R1)+	----	----
8015	071314	105300			DECB	R0	----	----
8016	071316	001375			BNE	2\$	----	----
8017	071320	012777	002624	110754	MOV	#TABE,@BARX	----	LOAD START ADDRESS OF TABLE E----
8018	071326	012777	177667	110750	MOV	#177667,@BCRX	----	LOAD INTO BCR 1---
8019	071334	012777	000101	110736	MOV	#101,@CSRX	----	SET DMA ENB,INT ENB IN CSR2
8020	071342				SETPRI	#PRI00	----	SET PRIORITY TO ZERO
	071342	012700	000000				MOV	#PRI00,R0
	071346	104441					TRAP	C\$SPRI
8021	071350	042777	000010	110722	BIC	#10,@CSRX	----	SELECT CHANNEL 1
8022	071356	012737	000001	002374	MOV	#1,CHAN	----	LOAD CHANNEL NUMBER
8023	071364	112777	000013	110676	MOVB	#13,@ICRHX	----	LOAD GTS INTO ACR 1-----
8024	071372	012701	077777		MOV	#77777,R1	----	LOAD LOOP COUNTER
8025	071376	005737	002376	13\$:	TST	INTFC1	----	IS INTERRUPT IN CHANNEL 1 OCCER
8026	071402	001015			BNE	10\$	----	BRANCH IF YES
8027	071404	005301			DEC	R1	----	DECREMENT COUNTER
8028	071406	001373			BNE	13\$	----	IF NO,TEST AGIN
8029	071410	017737	110664	002502	MOV	@CSRX,BAD	----	GET CSR1 CONTENTS
8030	071416	012737	100006	002500	MOV	#100006,GOOD	----	BC OF,DMA DIR,SYS CONT SHOULD BE SET
8031	071424				ERRSOFT	2601,E232,ERR201	----	ERROR HANDLER
	071424	104457					TRAP	C\$ERSOFT
	071426	005051					.WORD	2601
	071430	006072					.WORD	E232
	071432	003500					.WORD	ERR201
8032	071434				CKLOOP		----	BRANCH TO BGNSEG IF ERRLOOP IS SET
	071434	104406					TRAP	C\$CLP1
8033	071436	052777	000010	110634	10\$:	BIS	#10,@CSRX	SELECT CHANNEL 2
8034	071444	012737	000002	002374	MOV	#2,CHAN	----	LOAD CHANNEL NUMBER
8035	071452	005737	002400		TST	INTFC2	----	IS AN INTERRUPT IN CHANNEL 2 OCCURED
8036	071456	001013			BNE	11\$	----	BRANCH IF YES
8037	071460	017737	110614	002502	MOV	@CSRX,BAD	----	GET CSR2 CONTENTS
8038	071466	012737	020010	002500	MOV	#20010,GOOD	----	COMP END,MUX SHOULD BE SET
8039	071474				ERRSOFT	2602,E232,ERR201	----	ERROR HANDLER
	071474	104457					TRAP	C\$ERSOFT
	071476	005052					.WORD	2602
	071500	006072					.WORD	E232
	071502	003500					.WORD	ERR201
8040	071504				CKLOOP		----	BRANCH TO BGNSEG IF ERRLOOP IS SET
	071504	104406					TRAP	C\$CLP1
8041	071506				11\$:	SETPRI	#PRI07	NO FURTHER INTERRUPT ALLOWED
	071506	012700	000340				MOV	#PRI07,R0
	071512	104441					TRAP	C\$SPRI
8042	071514	012701	002514		MOV	#TABD,R1	----	PROVIDE FIRST BYTE OF BUFFER D
8043	071520	012702	002624		MOV	#TABE,R2	----	PROVIDE FIRST BYTE OF BUFFER E
8044	071524	005037	002404		CLR	CNT1	----	CLEAR BUFFER COUNTER
8045	071530	005237	002404	30\$:	INC	CNT1	----	BYTE COUNT
8046	071534	122122			CMPB	(R1)+,(R2)+	----	BUFFER D EQUAL TO BUFFER E
8047	071536	001433			BEQ	33\$	----	IF YES, CONTINUE
8048	071540	005037	002500		CLR	GOOD	----	CLEAR GOOD
8049	071544	005037	002502		CLR	BAD	----	CLEAR BAD
8050	071550	005037	002426		CLR	TXADRH	----	SET UP DATA FOR ERROR MESSAGE

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 68-2  
 TEST 26: MCR FUNCTION TEST OF CHANNEL 2

8051	071554	010137	002430			MOV	R1, TXADRL		:...
8052	071560	005337	002430			DEC	TXADRL		:...
8053	071564	005037	002422			CLR	RXADRH		:...
8054	071570	010237	002424			MOV	R2, RXADRL		:...
8055	071574	005337	002424			DEC	RXADRL		:...
8056	071600	116137	177777	002500		MOVB	-1(R1), GOOD		:SET UP DATA FOR ERROR MESSAGE
8057	071606	116237	177777	002502		MOVB	-1(R2), BAD		:...
8058	071614					ERRSOFT	2603, E231, ERR231		:ERROR HANDLER
	071614	104457							TRAP C\$ERSOFT
	071616	005053							.WORD 2603
	071620	006010							.WORD E231
	071622	003744							.WORD ERR231
8059	071624					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	071624	104406							TRAP C\$CLP1
8060	071626	023737	002404	002402	33\$:	CMP	CNT1, RSAVE		:ALL BYTES COMPARED ?
8061	071634	001335				BNE	30\$		:IF NOT, GET NEXT ONE
8062	071636					ENDSEG			
	071636								10000\$:
	071636	104405							TRAP C\$ESEG
8063	071640	005077	110434			CLR	@CSRX		:SELECT CHAN. 1, CLR ALL BIT IN CHA.2
8064	071644	112777	000014	110416		MOVB	#14, @ICRX		:---LOAD TCA INTO ACR 1---
8065	071652	017737	110414	002502		MOV	@IDRX, BAD		:READ DIR1 TO CLEAR ACCRQ SIGNAL
8066	071660	005077	110414			CLR	@CSRX		:CLEAR CSR1
8067									
8068									
8069									
8070	071664	052777	000012	110406		BIS	#12, @CSRX		:SET SYS CON IN CHA 1, SELECT CHANNEL 2
8071	071672	005077	110410			CLR	@MCRX		:CLEAR MCR2 REGISTER
8072	071676	113777	002236	110404		MCVB	MCINP, @MCRHX		:---LOAD CNT INPUT INTO MCR 2---
8073	071704	052777	100012	110374		BIS	#100012, @MCRX		:--- ENB MATCH + EOS CHARACTER (12)
8074	071712	042777	000010	110360		BIC	#10, @CSRX		:SELECT CHANNEL 1
8075	071720	012737	000001	002374		MOV	#1, CHAN		:LOAD CHANNEL NUMBER
8076	071726	012777	003044	110346		MOV	#TABG, @BARX		:---LOAD START ADDRESS OF BUFFER G---
8077	071734	013737	002236	002402		MOV	MCINP, RSAVE		:STORE MATCH CHARACTER CNT INPUT
8078	071742	062737	000003	002402		ADD	#3, RSAVE		:ADD 3 DATA BYTES TO CNT INPUT
8079	071750	005437	002402			NEG	RSAVE		:BILD 2'COMPL FOR BCR
8080	071754	013777	002402	110322		MOV	RSAVE, @BCRX		:---LOAD BYTE COUNT REGISTER 1---
8081	071762	005437	002402			NEG	RSAVE		:REBUILD INPUT+DATA FOR COUNTER CNT1
8082	071766	012777	000107	110304		MOV	#107, @CSRX		:--- DMA ENB, DMA DIR, INT ENB, SYS CONT
8083	071774					BGNSEG			
	071774	104404							TRAP C\$BSEG
8084	071776	005037	002400			CLR	INTFC2		:CLEAR INTERRUPT FLAG
8085	072002	005037	002376			CLR	INTFC1		:CLEAR INTERRUPT FLAG
8086	072006	052777	000010	110264		BIS	#10, @CSRX		:SELECT CHANNEL 2
8087	072014	012700	000102			MOV	#66, RO		:CLEAR BUFFER TABH
8088	072020	012701	003146			MOV	#TABH, R1		:...
8089	072024	105021			34\$:	CLRB	(R1)+		:...
8090	072026	105300				DECB	RO		:...
8091	072030	001375				BNE	34\$		:...
8092	072032	012777	003146	110242		MOV	#TABH, @BARX		:---LOAD START ADDRESS OF BUFFER H---
8093	072040	012777	177675	110236		MOV	#177675, @BCRX		:---LOAD MAX. TRANSFER INTO BCR 2---
8094	072046	012777	000101	110224		MOV	#101, @CSRX		:---SET DMA ENB, INT ENB IN CSR2---
8095	072054	042777	000010	110216		BIC	#10, @CSRX		:SELECT CHANNEL 1
8096	072062	012737	000001	002374		MOV	#1, CHAN		:LOAD CHANNEL NUMBER
8097	072070					SETPRI	#PRI00		:SET PRI. TO ZERO (ALLOW INTERRUPT)
	072070	012700	000000						MOV #PRI00, RO
	072074	104441							TRAP C\$SPRI



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 68-3  
 TEST 26: MCR FUNCTION TEST OF CHANNEL 2

8098	072076	112777	000013	110164		MOVB	#13,@ICRHX		----
8099	072104	012701	077777			MOV	#77777,R1		LOAD GTS INTO ACR 1-----
8100	072110	005737	002376		40\$:	TST	INTFC1		:LOAD COUNTER
8101	072114	001015				BNE	43\$		:IS INTERRUPT IN CHANNEL 1 OCCUR
8102	072116	005301				DEC	R1		:BRANCH IF YES
8103	072120	001373				BNE	40\$		:IS TIME OVER
8104	072122	017737	110152	002502		MOV	@CSRX,BAD		:IF NO, TEST AGAIN
8105	072130	012737	100006	002500		MOV	#100006,GOOD		:GET CSR1 CONTENTS
8106	072136					ERRSOFT	2604,E232,ERR201		:BC OF,DMA DIR,SYS CONT,SHOULD BE SET
	072136	104457							:ERROR HANDLER
	072140	005054							TRAP CSERSOFT
	072142	006072							.WORD 2604
	072144	003500							.WORD E232
8107	072146					CKLOOP			.WORD ERR201
	072146	104406							:BRANCH TO BGNSEG IF ERRLOOP IS SET
8108	072150	052777	000010	110122	43\$:	BIS	#10,@CSRX		TRAP C\$CLP1
8109	072156	012737	000002	002374		MOV	#2,CHAN		:SELECT CHANNEL 2
8110	072164	005737	002400			TST	INTFC2		:LOAD CHANNEL NUMBER
8111	072170	001013				BNE	41\$		:IS THERE AN INTERRUPT IN CHANNEL 2 -
8112	072172	017737	110102	002502		MOV	@CSRX,BAD		:BRANCH IF YES
8113	072200	012737	020010	002500		MOV	#20010,GOOD		:GET CSR2 CONTENTS
8114	072206					ERRSOFT	2605,E232,ERR201		:COMP END,MUX SHOULD BE SET
	072206	104457							:ERROR HANDLER
	072210	005055							TRAP CSERSOFT
	072212	006072							.WORD 2605
	072214	003500							.WORD E232
8115	072216					CKLOOP			.WORD ERR201
	072216	104406							:BRANCH TO BGNSEG IF ERRLOOP IS SET
8116	072220				41\$:	SETPRI	#PRI07		TRAP C\$CLP1
	072220	012700	000340						:DISABLE INTERRUPTS
	072224	104441							MOV #PRI07,RO
8117	072226	012701	003044			MOV	#TABG,R1		TRAP C\$SPRI
8118	072232	012702	003146			MOV	#TABH,R2		:PROVIDE FIRST BYTE OF BUFFER G
8119	072236	005037	002404			CLR	CNT1		:PROVIDE FIRST BYTE OF BUFFER H
8120	072242	005237	002404		60\$:	INC	CNT1		:CLEAR COUNTER
8121	072246	122122				CMPB	(R1)+,(R2)+		:BYTE COUNT
8122	072250	001431				BEQ	63\$		:BUFFER G EQUAL BUFFER H
8123	072252	005037	002500			CLR	GOOD		:IF YES,CONTINUE
8124	072256	005037	002502			CLR	BAD		:CLEAR GOOD
8125	072262	005037	002426			CLR	TXADRH		:CLEAR BAD
8126	072266	010137	002430			MOV	R1,TXADRL		:SET UP DATA FOR ERROR MESSAGE
8127	072272	005337	002430			DEC	TXADRL		:....
8128	072276	005037	002422			CLR	RXADRH		:....
8129	072302	010237	002424			MOV	R2,RXADRL		:....
8130	072306	005337	002424			DEC	RXADRL		:....
8131	072312	114137	002500			MOVB	-(R1),GOOD		:SET UP DATA FOR ERROR MESSAGE
8132	072316	114237	002502			MOVB	-(R2),BAD		:....
8133	072322					ERRSOFT	2606,E231,ERR231		:ERROR HANDLER
	072322	104457							TRAP CSERSOFT
	072324	005056							.WORD 2606
	072326	006010							.WORD E231
	072330	003744							.WORD ERR231
8134	072332					CKLOOP			:BRANCH TO BGNSEG IF ERRLOOP IS SET
	072332	104406							TRAP C\$CLP1
8135	072334	023737	002404	002402	63\$:	CMP	CNT1,RSAVE		:ALL BYTES COMPARED ?
8136	072342	001337				BNE	60\$		:IF NO,GET NEXT ONE
8137	072344					ENDSEG			



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 68-4  
TEST 26: MCR FUNCTION TEST OF CHANNEL 2

```

072344
072344 104405
8138 072346 042777 000010 107724      BIC      #10,@CSRX      ;SELECT CHANNEL 1
8139 072354 112777 000014 107706      MOVB     #14,@ICRHX    ;---LOAD TCA INTO ACR 1-----
8140 072362 112777 000077 107706      MOVB     #77,@IDRHX    ;---LOAD UNL INTO DOR 1-----
8141 072370 004737 011060      JSR      PC,LOOP      ;WAIT A LITTLE
8142 072374 112777 000200 107666      MOVB     #200,@ICRHX   ;---LOAD SWRST INTO ACR 1-----
8143 072402 112777 000000 107660      MOVB     #0,@ICRHX    ;---LOAD NOT SWRST INTO ACR 1-----
8144 072410 005737 002234      TST      QVP          ;IS QUICK VERIFY PASS SELECTED ?
8145 072414 001025      BNE      EXQV26       ;IF YES EXIT TEST
8146 072416 005237 002322      INC      ITRCNT       ;ITERATION COUNTER +1
8147 072422 023737 002322 002320      CMP      ITRCNT,ITRDEF ;DEFAULT ITERATION EXECUTED
8148 072430 001417      BEQ      EXQV26       ;IF YES EXIT TEST
8149 072432 012777 000010 107640      MCV      #10,@CSRX    ;CLEAR CSR1,SELECT CHANNEL 2
8150 072440 005077 107634      CLR      @CSRX        ;CLEAR CSR2,SELECT CHANNEL 1
8151 072444 112777 000217 107616      MOVB     #217,@ICRHX   ;---LOAD SIC INTO ACR 1-----
8152 072452 004737 011072      JSR      PC,WAIT      ;WAIT A LITTLE
8153 072456 112777 000017 107604      MOVB     #17,@ICRHX   ;---LOAD NOT SIC INTO ACR 1-----
8154 072464 000137 071050      JMP      ITAC26       ;IF NO TEST ITERATION
8155 072470      EXQV26: EXIT        TST
072470 104432      TRAP     C$EXIT
072472 000050      .WORD    L10057-.

8156
8157
8158 072474      045      123      062      TSHD26: .NLIST  BEX
8159      .ASCIZ  /%S2%AMCR FUNCTION TEST OF CHANNEL 2%/
8160      .LIST   BEX
8161 072542      .EVEN
072542      .ENDTST
072542 104401      L10057: TRAP     C$ETST

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 69  
TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST

```

8163 .SBTTL TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST
8164 .....
8165 IEX - TEST 27
8166 THIS TEST IS ONLY BE CARRIED OUT IF A 22-BIT Q-BUS (11/23B) IS USED
8167 AND IF THE AVAILABLE MEMORY IS GREATER THAN 128K.
8168
8169 PART 1 FINDS OUT IF AVAILABLE MEMORY IS GREATER THAN 128K. IF YES, THEN
8170 A DMA IS CARRIED OUT BY SENDING 2K DATA VIA THE IEC/IEEE BUS
8171 FROM CHANNEL 1 TO 2. THE DMA CROSS THE BOUNDARY
8172 THE SEQUENCE DESCRIBED ABOVE IS ALSO EXECUTED WITH 256K (BA 19 SET),
8173 512K (BA 20 SET) AND 1024K (BA 21 SET).
8174 PART 2 SAME AS PART 1 EXCEPT THE CHANNELS. THE DMA DATA TRANSFER
8175 IS CARRIED OUT FROM CHANNEL 2 TO 1.
8176 .....
8177 072544 BGNTST
      072544
8178 072544 READBUS ;DETERMINE BUS TYPE T27::
      072544 104407 ;TRAP CSRDBU
8179 072546 BCOMPLETE 3$ ;BRANCH IF Q-BUS TRAP CSRDBU
      072546 103415 ;BCS 3$
8180 072550 TST PNTF ;IS THE PNT FLAG SET
8181 072554 BEQ 2$ ;IF YES, PRINT MESSAGE
8182 072556 PRINTF #INFO3 ;....
      072556 012746 076502 MOV #INFO3,-(SP)
      072562 012746 000001 MOV #1,-(SP)
      072566 010600 MOV SP,RO
      072570 104417 TRAP C$PNTF
      072572 062706 000004 ADD #4,SP
8183 072576 2$: EXIT TST ;EXIT TEST
      072576 104432 ;TRAP C$EXIT
      072600 002654 .WORD L10060-
8184 072602 012737 000004 002350 3$: MOV #4,MASK ;
8185 072610 005737 002324 TST PNTF ;IS THE PNT FLAG SET
8186 072614 001410 BEQ 7$ ;IF YES, PRINT THE TEST HEADER
8187 072616 PRINTF #TSHD27 ;....
      072616 012746 075406 MOV #TSHD27,-(SP)
      072622 012746 000001 MOV #1,-(SP)
      072626 010600 MOV SP,RO
      072630 104417 TRAP C$PNTF
      072632 062706 000004 ADD #4,SP
8188 072636 7$: SETVEC VECC1,#INTSC1,#PRI07 ;SET VECTOR FOR CHANNEL 1
      072636 012746 000340 MOV #PRI07,-(SP)
      072642 012746 010142 MOV #INTSC1,-(SP)
      072646 013746 002244 MOV VECC1,-(SP)
      072652 012746 000003 MOV #3,-(SP)
      072656 104437 TRAP C$SVEC
      072660 062706 000010 ADD #10,SP
8189 072664 SETVEC VECC2,#INTSC2,#PRI07 ;SET VECTOR FOR CHANNEL 2
      072664 012746 000340 MOV #PRI07,-(SP)
      072670 012746 010152 MOV #INTSC2,-(SP)
      072674 013746 002246 MOV VECC2,-(SP)
      072700 012746 000003 MOV #3,-(SP)
      072704 104437 TRAP C$SVEC
      072706 062706 000010 ADD #10,SP
8190 072712 005037 002364 CLR CSRMS1 ;LOAD MASK FOR CSR1
8191 072716 012737 001000 002366 MOV #1000,CSRMS2 ;LOAD EXTENDED ADDR BIT FOR RX CHANNEL
8192 072724 005037 002376 ITAC27: CLR INTFC1 ;CLEAR INTERRUPT FLAG

```



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 69-1  
TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST

8193	072730	005037	002400		CLR	INTFC2	:CLEAR INTERRUPT FLAG
8194	072734	023737	002342	002350	CMP	PHHSIZ,MASK	:IS THERE MORE THAN 128K
8195	072742	002404			BLT	11\$	:IF NO SKIP TEST
8196	072744	003005			BGT	13\$	:
8197	072746	005737	002344		TST	PHLSIZ	:
8198	072752	001002			BNE	13\$	:
8199	072754			11\$:	EXIT	TST	:EXIT TEST
	072754	104432					
	072756	002476					TRAP C\$EXIT
8200	072760	004737	010534	13\$:	JSR	PC,BGIN1	.WORD L10060-
8201	072764	000240			NOP		:
8202	072766	012737	177777	002336	MOV	#177777,PHLOW	:LOAD INPUT FOR CONVERSION ROUTINE
8203	072774	013704	002350		MOV	MASK,R4	:...
8204	073000	005304			DEC	R4	:...
8205	073002	010437	002334		MOV	R4,PHHIGH	:...
8206	073006	004737	007724		JSR	PC,PVCON	:CREATE VIRTUAL ADDRESS
8207	073012	013700	002340		MOV	VIADD,R0	:GET START ADDRESS OF BUFFER A
8208	073016	010037	002356		MOV	R0,BUFAB	:LOAD START ADDRESS OF BUFFER A
8209	073022	012701	010000		MOV	#10000,R1	:LOAD COUNTER FOR 2K
8210	073026	005003			CLR	R3	:R3 CONTAINS DATA
8211	073030	012737	000020	172516	MOV	#20,SR3	**ENABLE 22-BIT MEMORY MANAGEMENT**
8212	073036	012737	000001	177572	MOV	#1,SR0	**ENABLE MEMORY MANAGEMENT**
8213	073044	110310			MOVB	R3,(R0)	:LOAD FIRST DATA BYTE
8214	073046	012700	120000		MOV	#120000,R0	:SET UP ADDRESS FOR NEXT DATA BYTE
8215	073052	005237	172352		INC	KPAR5	:LOAD NEXT PAGE
8216	073056	005203		16\$:	INC	R3	:CREATE NEXT DATA
8217	073060	110320			MOVB	R3,(R0)+	:LOAD BUFFER WITH DATA
8218	073062	005301			DEC	R1	:
8219	073064	001374			BNE	10\$	:
8220	073066	005300			DEC	R0	:
8221	073070	012701	010000		MOV	#10000,R1	:LOAD 2K
8222	073074	010037	002360		MOV	R0,BUFB	:LOAD START ADDRESS OF BUFFER B
8223	073100	000240		20\$:	NOP		:
8224	073102	105020			CLRB	(R0)+	:CLEAR BUFFER B
8225	073104	005301			DEC	R1	:2K CLEARED
8226	073106	001374			BNE	20\$	:IF YES,DO THE TEST
8227	073110	005037	177572		CLR	SR0	**DISABLE MEMORY MANAGEMENT**
8228	073114	112777	000212	107146	MOVB	#212,@ICRH	-----LOAD TON INTO ACR 1-----
8229	073122	013701	002314		MOV	DPA2,R1	:CREATE MLA2
8230	073126	062701	000040		ADD	#40,R1	:
8231	073132	010137	002412		MOV	R1,MLA2	:STORE MLA2
8232	073136	113777	002412	107132	MOVB	MLA2,@IDRH	-----LOAD MLA2 INTO DOR 1-----
8233	073144	004737	011060		JSR	PC,LOOP	:WAIT A LITTLE
8234	073150	012777	177777	107124	MOV	#177777,@BARX	---LOAD START ADDRESS OF BUFFER A---
8235	073156	013737	002240	002402	MOV	BCINP,RSAVE	:STORE BYTE COUNT INPUT
8236	073164	005437	002402		NEG	RSAVE	:BUILD 2'COMPL FOR BCR
8237	073170	013777	002402	107106	MOV	RSAVE,@BCRX	---LOAD INPUT INTO BCR 1---
8238	073176	012777	000167	107074	MOV	#167,@CSRX	---DMA ENB,DMA DIR,INT ENB,SYS CO,BA16+17
8239	073204	053777	002364	107066	BIS	CSRMS1,@CSRX	---LOAD ADDITIONAL EXTENDED ADDRESS BIT
8240	073212	052777	000010	107060	BIS	#10,@CSRX	:SELECT CHANNEL 2
8241	073220	012777	007777	107054	MOV	#7777,@BARX	---LOAD START ADDRESS OF BUFFER B---
8242	073226	013777	002402	107050	MOV	RSAVE,@BCRX	---LOAD INPUT INTO BCR 2---
8243	073234	012777	000111	107036	MOV	#111,@CSRX	---SET DMA ENB,INT ENB IN CSR2
8244	073242	053777	002366	107030	BIS	CSRMS2,@CSRX	---LOAD EXTENDED ADDRESS BITS-----
8245	073250				SETPRI	#PRIO0	:SET PRIORITY TO ZERO
	073250	012700	000000				MOV #PRIO0,R0
	073254	104441					TRAP C\$SPRI



HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 69-2  
 TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST

8246	073256	042777	000010	107014		BIC	#10,@CSRX		:SELECT CHANNEL 1	
8247	073264	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER	
8248	073272	112777	000013	106770		MOV	#13,@ICRHX		:----LOAD GTS INTO ACR 1-----	
8249	073300	012701	077777			MOV	#77777,R1		:LOAD LOOP COUNTER	
8250	073304	005737	002376		23\$:	TST	INTFC1		:HAS INTERRUPT IN CHANNEL 1 OCCURED	
8251	073310	001017				BNE	24\$		:BRANCH IF YES	
8252	073312	005301				DEC	R1		:DECREMENT COUNTER	
8253	073314	001373				BNE	23\$		:IF NO,TEST AGIN	
8254	073316	017737	106756	002502		MOV	@CSRX,BAD		:GET CSR1 CONTENTS	
8255	073324	012737	100066	002500		MOV	#100066,GOOD		:BC OF,DMA DIR,SYS CONT	
8256	073332	053737	002364	002500		BIS	CSRMS1,GOOD		:SET EXTENDED ADDRESS BIT TO CSR CONT.	
8257	073340					ERRSOFT	2701,E232,ERR201		:ERROR HANDLER	
	073340	104457							TRAP	C\$ERSOFT
	073342	005215							.WORD	2701
	073344	006072							.WORD	E232
	073346	003500							.WORD	ERR201
8258	073350	052777	000010	106722	24\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2	
8259	073356	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER	
8260	073364	005737	002400			TST	INTFC2		:HAS AN INTERRUPT IN CHANNEL 2 OCCURED	
8261	073370	001015				BNE	25\$		:BRANCH IF YES	
8262	073372	017737	106702	002502		MOV	@CSRX,BAD		:GET CSR2 CONTENTS	
8263	073400	012737	100010	002500		MOV	#100010,GOOD		:BC OF,MUX SHOULD BE SET	
8264	073406	053737	002366	002500		BIS	CSRMS2,GOOD		:SET EXTENDED ADDRESS BIT TO CSR CONT.	
8265	073414					ERRSOFT	2702,E232,ERR201		:ERROR HANDLER	
	073414	104457							TRAP	C\$ERSOFT
	073416	005216							.WORD	2702
	073420	006072							.WORD	E232
	073422	003500							.WORD	ERR201
8266	073424				25\$:	SETPRI	#PRI07		:NO FURTHER INTERRUPT ALLOWED	
	073424	012700	000340						MOV	#PRI07,R0
	073430	104441							TRAP	C\$SPRI
8267	073432	042777	000010	106640		BIC	#10,@CSRX		:SELECT CHANNEL 1	
8268	073440	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER	
8269	073446	112777	000014	106614		MOV	#14,@ICRHX		:----LOAD TCA INTO ACR 1-----	
8270	073454	013737	002336	002500		MOV	PHLOW,GOOD		:SET UP COMPARE VALUE	
8271	073462	063737	002240	002500		ADD	BCINP,GOOD		:...	
8272	073470	017737	106606	002502		MOV	@BARX,BAD		:GET BAR1 CONTENTS	
8273	073476	023737	002502	002500		CMP	BAD,GOOD		:HAS BAR1 THE CORRECT ADDRESS	
8274	073504	001404				BEQ	30\$		:BRANCH IF YES	
8275	073506					ERRSOFT	2703,E234,ERR501		:ERROR HANDLER	
	073506	104457							TRAP	C\$ERSOFT
	073510	005217							.WORD	2703
	073512	006200							.WORD	E234
	073514	003702							.WORD	ERR501
8276	073516	017737	106562	002502	30\$:	MOV	@BCRX,BAD		:GET BCR1 CONTENTS	
8277	073524	005037	002500	002500		CLR	GOOD		:	
8278	073530	023737	002502	002500		CMP	BAD,GOOD		:IS BCR1 ZERO	
8279	073536	001404				BEQ	31\$		:BRANCH IF YES	
8280	073540					ERRSOFT	2704,E235,ERR501		:ERROR HANDLER	
	073540	104457							TRAP	C\$ERSOFT
	073542	005220							.WORD	2704
	073544	006231							.WORD	E235
	073546	003702							.WORD	ERR501
8281	073550	052777	000010	106522	31\$:	BIS	#10,@CSRX		:SELECT CHANNEL 2	
8282	073556	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER	
8283	073564	013737	002360	002500		MOV	BUFBB,GOOD		:SET UP COMPARE VALUE	
8284	073572	042737	160000	002500		BIC	#160000,GOOD		:CLEAR BIT 13+14+15	





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 69-4  
 TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST

```

8334 074132 005077 106142          CLR    @CSRX          ;CLEAR CSR1
8335                               :+++-----
8336                               :PART 2 DMA DATA TRANSFER FROM CHANNEL 2 TO 1
8337                               :+++-----
8338 074136 005037 002376          PSEU27: CLR    INTFC1          ;CLEAR INTERRUPT FLAG FOR CHANNEL 1
8339 074142 005037 002400          CLR    INTFC2          ;CLEAR INTERRUPT FLAG FOR CHANNEL 2
8340 074146 004737 010710          JSR    PC,BGIN2        ;SET UP PARAMETER
8341 074152 000240                    NOP
8342 074154 012737 177777 002336    MOV    #177777,PHLOW    ;LOAD INPUT FOR CONVERSION ROUTINE
8343 074162 013704 002350          MOV    MASK,R4         ;...
8344 074166 005304                    DEC    R4              ;...
8345 074170 010437 002334          MOV    R4,PHHIGH
8346 074174 004737 007724          JSR    PC,PVCON
8347 074200 013700 002340          MOV    VIADD,R0        ;CREATE VIRTUAL ADDRESS
8348 074204 010037 002356          MOV    R0,BUFAB        ;GET START ADDRESS OF BUFFER A
8349 074210 012701 010000          MOV    #10000,R1       ;LOAD START ADDRESS OF BUFFER A
8350 074214 005003                    CLR    R3              ;LOAD COUNTER FOR 2K
8351 074216 012737 000020 172516    MOV    #20,SR3         ;R3 CONTAINS DATA
8352 074224 012737 000001 177572    MOV    #1,SR0          ;**ENABLE 22-BIT MEMORY MANAGEMENT**
8353 074232 110310                    MOV    R3,(R0)         ;**ENABLE MEMORY MANAGEMENT**
8354 074234 012700 120000          MOV    #120000,R0      ;LOAD FIRST DATA BYTE
8355 074240 005237 172352          INC    KPAR5           ;SET UP ADDRESS FOR NEXT DATA BYTE
8356 074244 005203                    INC    R3              ;LOAD NEXT PAGE
8357 074246 110320                    MOV    R3,(R0)+        ;CREATE NEXT DATA
8358 074250 005301                    DEC    R1              ;LOAD BUFFER WITH DATA
8359 074252 001374                    BNE    16$
8360 074254 005300                    DEC    R0
8361 074256 012701 010000          MOV    #10000,R1       ;LOAD 2K
8362 074262 010037 002360          MOV    R0,BUFBB        ;LOAD START ADDRESS OF BUFFER B
8363 074266 000240                    NOP
8364 074270 105020                    CLRB   (R0)+           ;CLEAR BUFFER B
8365 074272 005301                    DEC    R1              ;2K CLEARED
8366 074274 001374                    BNE    20$
8367 074276 005037 177572          CLR    SR0             ;IF YES,DO THE TEST
8368 074302 112777 000212 105760    MOV    #212,@ICRHX     ;**DISABLE MEMORY MANAGEMENT**
8369 074310 013701 002312          MOV    DPA1,R1         ;----LOAD TON INTO ACR 2-----
8370 074314 062701 000040          ADD    #40,R1          ;CREATE MLA1
8371 074320 010137 002410          MOV    R1,MLA1         ;...
8372 074324 113777 002410 105744    MOV    MLA1,@IDRHX     ;STORE MLA1
8373 074332 004737 011060          JSR    PC,LOOP         ;----LOAD MLA1 INTO DOR 2-----
8374 074336 012777 177777 105736    MOV    #177777,@BARX   ;WAIT A LITTLE
8375 074344 013737 002240 002402    MOV    BCINP,RSAVE     ;---LOAD START ADDRESS OF BUFFER A---
8376 074352 005437 002402          NEG    RSAVE           ;STORE BYTE COUNT INPUT
8377 074356 013777 002402 105720    MOV    RSAVE,@BCRX     ;BUILD 2'COMPL FOR BCR
8378 074364 012777 000177 105706    MOV    #177,@CSRX      ;----LOAD INPUT INTO BCR 1-----
8379 074372 053777 002364 105700    BIS    CSRMS1,@CSRX    ;-DMA ENB,DMA DIR,INT ENB,SYS CO,BA16+17
8380 074400 042777 000010 105672    BIC    #10,@CSRX      ;--LOAD ADDITIONAL EXTENDED ADDRESS BIT
8381 074406 012777 007777 105666    MOV    #7777,@BARX    ;SELECT CHANNEL 1
8382 074414 013777 002402 105662    MOV    RSAVE,@BCRX    ;----LOAD START ADDRESS OF BUFFER B---
8383 074422 012777 000101 105650    MOV    #101,@CSRX     ;----LOAD INPUT INTO BCR 1---
8384 074430 053777 002366 105642    BIS    CSRMS2,@CSRX   ;---SET DMA ENB,INT ENB IN CSR1
8385 074436                    SETPRI #PR100         ;---LOAD EXTENDED ADDRESS BITS-----
8385 074436 012700 000000          MOV    #PR100,R0      ;SET PRIORITY TO ZERO
8385 074442 104441                    TRAP  C$SPRI
8386 074444 052777 000010 105626    BIS    #10,@CSRX      ;SELECT CHANNEL 2
8387 074452 012737 000002 002374    MOV    #2,CHAN        ;LOAD CHANNEL NUMBER
8388 074460 112777 000013 105602    MOV    #13,@ICRHX     ;----LOAD GTS INTO ACR 2-----
    
```



8389	074466	012701	077777			MOV	#77777,R1		:LOAD LOOP COUNTER
8390	074472	005737	002400	23\$:		TST	INTFC2		:HAS INTERRUPT IN CHANNEL 2 OCCURED
8391	074476	001017				BNE	24\$		:BRANCH IF YES
8392	074500	005301				DEC	R1		:DECREMENT COUNTER
8393	074502	001373				BNE	23\$		:IF NO,TEST AGIN
8394	074504	017737	105570	002502		MOV	@CSRX,BAD		:GET CSR2 CONTENTS
8395	074512	012737	100016	002500		MOV	#100016,GOOD		:BC OF DMA DIR,SYS CONT
8396	074520	053737	002364	002500		BIS	CSRMS1,GOOD		:SET EXTENDED ADDRESS BIT TO CSR CONT.
8397	074526					ERRSOFT	2707,E232,ERR201		:ERROR HANDLER
	074526	104457							TRAP C\$ERSOFT
	074530	005223							.WORD 2707
	074532	006072							.WORD E232
	074534	003500							.WORD ERR201
8398	074536	042777	000010	105534	24\$:	BIC	#10,@CSRX		:SELECT CHANNEL 1
8399	074544	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
8400	074552	005737	002376			TST	INTFC1		:HAS AN INTERRUPT IN CHANNEL 1 OCCURED
8401	074556	001015				BNE	25\$		:BRANCH IF YES
8402	074560	017737	105514	002502		MOV	@CSRX,BAD		:GET CSR1 CONTENTS
8403	074566	012737	100000	002500		MOV	#100000,GOOD		:BC OF SHOULD BE SET
8404	074574	053737	002366	002500		BIS	CSRMS2,GOOD		:SET EXTENDED ADDRESS BIT TO CSR CONT.
8405	074602					ERRSOFT	2708,E232,ERR201		:ERROR HANDLER
	074602	104457							TRAP C\$ERSOFT
	074604	005224							.WORD 2708
	074606	006072							.WORD E232
	074610	003500							.WORD ERR201
8406	074612				25\$:	SETPRI	#PRI07		:NO FURTHER INTERRUPT ALLOWED
	074612	012700	000340						MOV #PRI07,R0
	074616	104441							TRAP C\$SPRI
8407	074620	052777	000010	105452		BIS	#10,@CSRX		:SELECT CHANNEL 2
8408	074626	012737	000002	002374		MOV	#2,CHAN		:LOAD CHANNEL NUMBER
8409	074634	112777	000014	105426		MOVB	#14,@ICRHX		:----LOAD TCA INTO ACR 2-----
8410	074642	013737	002336	002500		MOV	PHLOW,GOOD		:SET UP COMPARE VALUE
8411	074650	063737	002240	002500		ADD	BCINP,GOOD		:. . .
8412	074656	017737	105420	002502		MOV	@BARX,BAD		:GET BAR1 CONTENTS
8413	074664	023737	002502	002500		CMP	BAD,GOOD		:HAS BAR1 THE CORRECT ADDRESS
8414	074672	001404				BEQ	30\$		:BRANCH IF YES
8415	074674					ERRSOFT	2709,E234,ERR501		:ERROR HANDLER
	074674	104457							TRAP C\$ERSOFT
	074676	005225							.WORD 2709
	074700	006200							.WORD E234
	074702	003702							.WORD ERR501
8416	074704	017737	105374	002502	30\$:	MOV	@BCRX,BAD		:GET BCR1 CONTENTS
8417	074712	005037	002500			CLR	GOOD		:. . .
8418	074716	023737	002502	002500		CMP	BAD,GOOD		:IS BCR1 ZERO
8419	074724	001404				BEQ	31\$		:BRANCH IF YES
8420	074726					ERRSOFT	2710,E235,ERR501		:ERROR HANDLER
	074726	104457							TRAP C\$ERSOFT
	074730	005226							.WORD 2710
	074732	006231							.WORD E235
	074734	003702							.WORD ERR501
8421	074736	042777	000010	105334	31\$:	BIC	#10,@CSRX		:SELECT CHANNEL 2
8422	074744	012737	000001	002374		MOV	#1,CHAN		:LOAD CHANNEL NUMBER
8423	074752	013737	002360	002500		MOV	BUFBB,GOOD		:SET UP COMPARE VALUE
8424	074760	042737	160000	002500		BIC	#160000,GOOD		:CLEAR BIT 13+14+15
8425	074766	063737	002240	002500		ADD	BCINP,GOOD		:. . .
8426	074774	017737	105302	002502		MOV	@BARX,BAD		:GET BAR2 CONTENTS
8427	075002	023737	002500	002502		CMP	GOOD,BAD		:HAS BAR2 THE CORRECT ADDRESS

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 69-6  
 TEST 27: EXTENDED ADDRESS BIT (Q22-BUS)TEST

8428	075010	001404				BEQ	43\$		:BRANCH IF YES
8429	075012					ERRSOFT	2711,E234,ERR501		:ERROR HANDLER
	075012	104457							TRAP
	075014	005227							.WORD
	075016	006200							.WORD
	075020	003702							.WORD
									C\$ERSOFT
									2711
									E234
									ERR501
8430	075022	012737	000001	177572	43\$:	MOV	#1,SRO		:**ENABLE MEMORY MANAGEMENT**
8431	075030	005337	172352			DEC	KPAR5		:CROSS THE BOUNDARY FOR FIRST BYTE
8432	075034	013701	002356			MOV	BUFAB,R1		:PROVIDE FIRST BYTE OF BUFFER A
8433	075040	111103				MOVB	(R1),R3		:GET THE FIRST DATA BYTE
8434	075042	010137	002340			MOV	R1,VIADD		:SET UP DATA FOR CONVERSION ROUTINE
8435	075046	004737	010024			JSR	PC,VPCON		:VIRTUAL TO PHYSICAL CONVERSION ROUTINE
8436	075052	012701	120000			MOV	#120000,R1		:POINT TO NEXT LOCATION
8437	075056	005237	172352			INC	KPAR5		:CROSS THE BOUNDARY FOR NEXT DATA BYTE'S
8438	075062	013702	002360			MOV	BUFBB,R2		:PROVIDE FIRST BYTE OF BUFFER B
8439	075066	012737	000001	002404		MOV	#1,CNT1		:SET UP COUNTER FOR DATA COMPARE
8440	075074	120322				CMFB	R3,(R2)+		:COMPARE THE FIRST DATA BYTE
8441	075076	000240				NOP			
8442	075100	001415				BEQ	44\$		:BRANCH IF EQUAL TO NEXT COMPARE
8443	075102	005037	002500			CLR	GOOD		:CLEAR HIGH BYTE OF GOOD
8444	075106	005037	002502			CLR	BAD		:CLEAR HIGH BYTE OF BAD
8445	075112	110337	002500			MOVB	R3,GOOD		:LOAD FIRST TX DATA FOR ERROR MESSAGES
8446	075116	013737	002334	002426		MOV	PHHIGH,TXADRH		:FIRST TX ADDRESS
8447	075124	013737	002336	002430		MOV	PHLOW,IXADRL		:GET ADDRESS OVER 128K
8448	075132	000425				BR	45\$		:BRANCH TO ERROR REPORT
8449	075134	005237	002404		44\$:	INC	CNT1		
8450	075140	122122				CMFB	(R1)+,(R2)+		:BUFFER A EQUAL BUFFER B
8451	075142	001453				BEQ	46\$		:IF YES CONTINUE
8452	075144	005037	002500			CLR	GOOD		:CLEAR GOOD
8453	075150	005037	002502			CLR	BAD		:CLEAR BAD
8454	075154	010137	002340			MOV	R1,VIADD		
8455	075160	004737	010024			JSR	PC,VPCON		:VIRTUAL TO PHYSICAL CONVERSION ROUTINE
8456	075164	013737	002334	002426		MOV	PHHIGH,IXADRH		:TX ADDRESS IS OVER 128K
8457	075172	013737	002336	002430		MOV	PHLOW,IXADRL		
8458	075200	116137	177777	002500		MOVB	-1(R1),GOOD		:LOAD TX DATA FOR ERROR MESSAGE
8459	075206	010237	002340		45\$:	MOV	R2,VIADD		:SET UP DATA FOR CONVERSION ROUTINE
8460	075212	004737	010024			JSR	PC,VPCON		:VIRTUAL TO PHYSICAL CONVERSION ROUTINE
8461	075216	013737	002334	002422		MOV	PHHIGH,RXADRH		:RX ADDRESS IS OVER 128K
8462	075224	013737	002336	002424		MOV	PHLOW,RXADRL		:GET ADDRESS OVER 128K
8463	075232	005337	002430			DEC	TXADRL		:...
8464	075236	005337	002424			DEC	RXADRL		
8465	075242	116237	177777	002502		MOVB	-1(R2),BAD		:LOAD RX DATA FOR ERROR MESSAGE
8466	075250	005037	177572			CLR	SRO		:**DISABLE MEMORY MANAGEMENT**
8467	075254					ERRSOFT	2712,E250,ERR231		:ERROR HANDLER
	075254	104457							TRAP
	075256	005230							.WORD
	075260	005726							.WORD
	075262	003744							.WORD
									C\$ERSOFT
									2712
									E250
									ERR231
8468	075264	012737	000001	177572		MOV	#1,SRO		:**ENABLE MEMORY MANAGEMENT**
8469	075272	023737	002404	002240	46\$:	CMP	CNT1,BCINP		:ALL BYTES COMPARED ?
8470	075300	001315				BNE	44\$		:IF NO, GET NEXT ONE
8471	075302	005037	177572			CLR	SRO		:**DISABLE MEMORY MANAGEMENT**
8472	075306	005077	104766			CLR	@CSRX		:CLEAR CSR2,SELECT CHANNEL 1
8473	075312	017737	104754	002502		MOV	@IDRX,BAD		:READ DIR1 FOR CLEAR BO BIT IN IIR
8474	075320	005077	104754			CLR	@CSRX		:CLEAR CSR1
8475	075324	006337	002350			ASL	MASK		:NEXT MEMORY RANGE
8476	075330	022737	001000	002364		CMP	#1000,CSRMS1		:WAS BA 18 SET



```

8477 075336 003403
8478 075340 012737 000400 002362 BLE 50$
8479 075346 006337 002362 002362 MOV #400,CSRMSK ;BRANCH IF EQUAL OR GREATER
8480 075352 053737 002362 002364 50$: ASL CSRMSK ;SET BA 18
8481 075360 006337 002366 BIS CSRMSK,CSRMS1
8482 075364 000240 ASL CSRMS2
8483 075366 022737 020000 002366 NOP ;SET EXTENDED ADDRESS BIT FOR RX CHANNEL
8484 075374 001402 CMP #20000,CSRMS2 ;ALL EXTENDED ADDRESS BIT TESTED
8485 075376 000137 072724 BEQ EXQV27 ;IF YES, EXIT TEST
8486 075402 EXQV27: JMP ITAC27
      075402 104432 TST
      075404 000050
8487
8488 075406 045 123 062 TSHD27: .NLIST BEX TRAP C$EXIT
8489 .ASCIZ /%S2%AQ-22 EXTENDED ADDRESS BIT TEST%N/ .WORD L10060-
8490 .LIST BEX
8491 075454 .EVEN
      075454 .ENDTST
      075454 104401

```

```

L10060: TRAP C$ETST

```

HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 70  
TEST 28: ADDITIONAL STANDBY TEST

8493  
8494  
8495  
8496  
8497  
8498  
8499  
8500  
8501  
8502  
8503  
8504  
8505  
8506

```
.SBTTL TEST 28: ADDITIONAL STANDBY TEST
:*****
:                               IEX - TEST 28
: THIS TEST MOVES A SLIDING ONE'S BIT PATTERN ACROSS THE
: ADDRESS LINE 16,17,18,19,20,21 IGNORING NXM ERRORS BUT CHECKING
: THE ADDRESS REGISTER LINES TO THE BUS.
:
: THE PATTERN SHOULD BE CHECKED ON A LOGIC ANALYSER .
: THE LOGIC ANALYSER HAS TO BE CONNECT TO ADDRESS LINES 16-21,
: THE TRIGGER HAS TO BE CONNECET TO THE SIGNAL ADREN L (E9,PIN4).
:
: THIS TEST IS ONLY CARRIED OUT IF A Q-BUS IS USED AND IF YOU ANSWER
: THE SOFTWARE QUESTION.
:*****
```

8507 075456  
075456  
8508 075456 005737 002242  
8509 075462 001015  
8510 075464 005737 002324  
8511 075470 001410  
8512 075472  
075472 012746 076374  
075476 012746 000001  
075502 010600  
075504 104417  
075506 062706 000004  
8513 075512  
075512 104432  
075514 001204  
8514 075516  
075516 104407  
8515 075520  
075520 103415  
8516 075522 005737 002324  
8517 075526 001410  
8518 075530  
075530 012746 076502  
075534 012746 000001  
075540 010600  
075542 104417  
075544 062706 000004  
8519 075550  
075550 104432  
075552 001146  
8520 075554 005737 002324  
8521 075560 001410  
8522 075562  
075562 012746 076220  
075566 012746 000001  
075572 010600  
075574 104417  
075576 062706 000004  
8523 075602  
075602 012746 076244  
075606 012746 000001  
075612 010600  
075614 104417

```
      BGNTST
      T28::
      TST     MAINB      ;IS THIS TEST SELECTED
      BNE     3$         ;EXIT IF NO
      TST     PNTF       ;IS THE PNT FLAG SET
      BEQ     2$         ;IF YES ,PRINT INFORMATION
      PRINTF  #INFO2     ;PRINT MESSAGE FOR USER
                        MOV     #INFO2,-(SP)
                        MOV     #1,-(SP)
                        MOV     SP,R0
                        TRAP    C$PNTF
                        ADD     #4,SP
2$:      EXIT     TST     ;EXIT TEST
                        TRAP    C$EXIT
                        .WORD   L10061-.
3$:      READBUS      ;ARE YOU ON A Q-BUS
                        TRAP    C$RDBU
      BCOMPLETE 5$     ;BRANCH IF Q-BUS
                        BCS     5$
      TST     PNTF       ;IS THE PNT FLAG SET
      BEQ     11$        ;IF YES,PRINT INFORMATION
      PRINTF  #INFO3     ;PRINT MESSAGE FOR USER
                        MOV     #INFO3,-(SP)
                        MOV     #1,-(SP)
                        MOV     SP,R0
                        TRAP    C$PNTF
                        ADD     #4,SP
11$:     EXIT     TST     ;IF UNI-BUS EXIT TEST
                        TRAP    C$EXIT
                        .WORD   L10061-.
5$:      TST     PNTF       ;IS THE PNT FLAG SET
      BEQ     7$         ;IF YES, PRINT THE TEST HEADER
      PRINTF  #TSHD28    ;....
                        MOV     #TSHD28,-(SP)
                        MOV     #1,-(SP)
                        MOV     SP,R0
                        TRAP    C$PNTF
                        ADD     #4,SP
7$:      PRINTF  #INFO1     ;PRINT MESSAGE FOR USER
                        MOV     #INFO1,-(SP)
                        MOV     #1,-(SP)
                        MOV     SP,R0
                        TRAP    C$PNTF
```





HARDWARE TESTS MACRO M1113 06-SEP-82 16:46 PAGE 70-2  
 TEST 28: ADDITIONAL STANDBY TEST

```

8555 076104          SETPRI #PRI07          ;DISABLE INTERRUPT
      076104 012700 000340
      076110 104441
8556 076112          EXIT TST
      076112 104432
      076114 000604
8557 076116          SETPRI #PRI07          ;DISABLE INTERRUPT
      076116 012700 000340
      076122 104441
8558 076124 022737 001000 002350
8559 076132 003411
8560 076134 006337 002350
8561 076140 023727 002350 000100
8562 076146 001312
8563 076150 012737 000400 002350
8564 076156 006337 002350
8565 076162 000240
8566 076164 000240
8567 076166 000240
8568 076170 112777 000014 104072
8569 076176 017737 104070 002502
8570 076204 022737 020000 002350
8571 076212 001270
8572 076214
      076214 104432
      076216 000502
      MOV #PRI07,R0
      TRAP C$SPRI
      TRAP C$EXIT
      .WORD L10061-.
      MOV #PRI07,R0
      TRAP C$SPRI
      ; IS BA 19 SET
      ; BRANCH IF YES
      ; WAS BA 17 SET
      ; BRANCH IF NO
      ; LOAD BA 19 INTO MASK
      ; GENERATE NEXT EXTENDED ADDRESS BIT
      ;---LOAD TCA INTO ACR 1-----
      ; READ DIR FOR CLEAR THE BO BIT
      ; ALL EXTENDED ADDRESS BIT SHIFTED
      ; BRANCH IF NO
      ;
      TRAP C$EXIT
      .WORD L10061-.
8573
8574
8575 076220 045 123 062 TSHD28: .NLIST BEX
8576 076244 045 116 045 INFO1: .ASCIZ /%S2%ASTANDBY TEST%/
8577 076323 045 101 101 .ASCIZ /%N%ACONNECT YOUR LOCIG ANALYSER TO BDAL 16-21%/
8578 076374 045 123 062 INFO2: .ASCIZ /%AAND THE TRIGGER TO ADREN L (E9,PIN4)%/
8579 076445 045 123 071 .ASCIZ /%S2%AFOR SELECT THIS TEST PLEASE ANSWER%/
8580 076502 045 123 062 INFO3: .ASCIZ /%S9%ATHE SOFTWARE QUESTION%/
8581 076563 111 123 040 TRIMSG: .ASCIZ /%S2%ATHIS TEST IS ONLY FOR THE Q-BUS INTERFACE%/
8582 076634 116 117 040 E271: .ASCIZ /IS THE LOCIG ANALYSER READY FOR TRIGGER?/
8583 .LIST BEX
8584 .EVEN
8585 076720
      076720
      076720 104401
      L10061: TRAP C$SETST
8586
8587 076722
8588
      ENDMOD
    
```



8591  
8592  
8593  
8594  
8595  
8596  
8597  
8608  
8609  
8638  
8639 076722  
8640  
8641  
8642  
8643  
8644  
8645  
8646  
8647  
8648  
8649

.TITLE PARAMETER CODING  
.SBTTL HARDWARE PARAMETER CODING SECTION  
BGNMOD

:++  
: THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS  
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE  
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE  
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE  
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS  
: WITH THE OPERATOR.  
:--

8650 076722 000032  
076722  
076724

BGNHRD

.WORD L10062-LSHARD/2  
LSHARD::

8651  
8661 076724 000031  
076724 077010  
076726 160000  
076730 177776  
076732

GPRMA HPM1,0,0,160000,177776,YES

.WORD T\$CODE  
.WORD HPM1  
.WORD T\$LOLIM  
.WORD T\$HILIM

8662 076734 001031  
076734 077046  
076736 000200  
076740 000770  
076742

GPRMA HPM2,2,0,200,770,YES

.WORD T\$CODE  
.WORD HPM2  
.WORD T\$LOLIM  
.WORD T\$HILIM

8663 076744 002032  
076744 077104  
076746 000340  
076750 000004  
076752 000006  
076754

GPRMD HPM3,4,0,340,4,6,YES

.WORD T\$CODE  
.WORD HPM3  
.WORD 340  
.WORD T\$LOLIM  
.WORD T\$HILIM

8664 076756 003032  
076756 077164  
076760 177777  
076762 000000  
076764 000036  
076766

GPRMD HPM4,6,0,-1,0,36,YES

.WORD T\$CODE  
.WORD HPM4  
.WORD -1  
.WORD T\$LOLIM  
.WORD T\$HILIM

8665 076770 004032  
076770 077226  
076772 177777  
076774 000000  
076776 000036  
077000

GPRMD HPM5,10,0,-1,0,36,YES

.WORD T\$CODE  
.WORD HPM5  
.WORD -1  
.WORD T\$LOLIM  
.WORD T\$HILIM

8666 077002 005120  
077002 077271  
077004 177777  
077006

GPRML HPM6,12,-1,NO

.WORD T\$CODE  
.WORD HPM6  
.WORD -1

PARAMETER CODING      MACRO M1113 06-SEP-82 16:46 PAGE 71-1  
HARDWARE PARAMETER CODING SECTION

8667  
8668  
8669 077010

ENDHRD

L10062: .EVEN

8670 077010

8677

8678 077010

8679 077046

8680 077104

8681 077164

8682 077226

8683 077271

8684

8685

8686

104

105

126

HPM1:

.NLIST BEX

.ASCIZ /DEVICE ADDRESS

/

126

105

103

HPM2:

.ASCIZ /VECTOR ADDRESS

/

120

122

111

HPM3:

.ASCIZ /PRIORITY LEVEL (FOR LSI WITH FIXED PRI. TYPE 4)/

104

105

126

HPM4:

.ASCIZ /DEVICE PRIMARY ADDRESS CH.1

/

104

105

126

HPM5:

.ASCIZ /DEVICE PRIMARY ADDRESS CH.2

/

111

123

040

HPM6:

.ASCIZ /IS TESTCABLE IN ?

/

.EVEN

.LIST BEX



8689  
8690  
8691  
8692  
8693  
8694  
8695  
8696  
8697  
8698  
8699

.SBTTL SOFTWARE PARAMETER CODING SECTION

:++  
: THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS  
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE  
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE  
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE  
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS  
: WITH THE OPERATOR.  
:--

8700 077334  
077334  
077336

000115

BGNSFT

.WORD L10063-L\$\$SOFT/2  
L\$\$SOFT::

8701  
8710  
8711 077336

000130  
077340 077400  
077342 177777

GPRML PMQVP,0,-1,YES

.WORD T\$CODE  
.WORD PMQVP  
.WORD -1

8712 077344  
077344  
077346  
077350  
077352  
077354

001052  
077427  
177777  
000003  
000077

GPRMD SPRM2,2,D,-1,3,77,YES

.WORD T\$CODE  
.WORD SPRM2  
.WORD -1  
.WORD T\$LOLIM  
.WORD T\$HILIM

8713 077356  
077356  
077360  
077362  
077364  
077366

002052  
077471  
177777  
000001  
003777

GPRMD SPRM4,4,D,-1,1,3777,YES

.WORD T\$CODE  
.WORD SPRM4  
.WORD -1  
.WORD T\$LOLIM  
.WORD T\$HILIM

8714 077370  
077370  
077372  
077374

003130  
077520  
177777

GPRML SPRM5,6,-1,YES

.WORD T\$CODE  
.WORD SPRM5  
.WORD -1

8715 077376  
077376

075004

EXIT SFT

.WORD T\$CODE

8716  
8717 077400  
8718 077427  
8719 077471  
8720 077520

121  
116  
116  
104

125  
125  
125  
117

111  
115  
115  
040

.NLIST BEX  
PMQVP: .ASCIZ /QUICK VERIFY TEST /  
SPRM2: .ASCIZ /NUMBER OF MATCH CHARACTER COUNTS /  
SPRM4: .ASCIZ /NUMBER OF BYTE COUNTS /  
SPRM5: .ASCIZ /DO YOU WANT THE ADDITIONAL STANDBY TEST/  
.LIST BEX  
.EVEN

8721  
8722  
8723  
8724 077570

ENDSFT

L10063: .EVEN

8725  
8726  
8733  
8734 077570  
8735 077570  
8736 077710  
8737  
8738

\$PATCH::

.BLKW 50 ; PATCH AREA  
.BLKB 400-<.8377> ; LASTAD SHIFT FOR LSI BUG

PARAMETER CODING      MACRO M1113 06-SEP-82 16:46 PAGE 73-1  
SOFTWARE PARAMETER CODING SECTION

8745  
8746 100000

LASTAD

100000 100024  
100002 000010

.EVEN  
.WORD T\$FREE  
.WORD T\$SIZE

8747 100004

L\$LAST::  
ENDMOD



PARAMETER CODING      MACRO M1113 06-SEP-82 16:46 PAGE 74  
SOFTWARE PARAMETER CODING SECTION

```

8749
8750
8763
8764 100004      BGNSETUP      1
8765 100004      BGNPTAB
      100004      000000
      100006      000006
      100010
8766 100010      160140      .WORD 160140      L10064:
8767 100012      000420      .WORD 420        :1ST (OF 8) REGISTER ADDRESS
8768 100014      000300      .WORD PRI06      :1ST (OF 2) VECTOR ADDRESS
8769 100016      000000      .WORD 0          : INTERRUPT PRIORITY
8770 100020      000001      .WORD 1          : DEVICE PRIMARY ADDRESS FOR CH.1
8771 100022      000000      .WORD 0          : DEVICE PRIMARY ADDRESS FOR CH.2
8772
8773 100024      ENDPTAB
      100024
8774 100024      ENDSETUP      L10066:
8775      000001      .END

```

PARAMETER CODING  
SYMBOL TABLE

MACRO M1113 06-SEP-82 16:46 PAGE 74-1

ADR = 000020 G	CDAT3 002440 G	C\$GPLO= 000030	EXQV11 036766	F\$HARD= 000004
ANS 002370 G	CDAT4 002442 G	C\$GPRI= 000040	EXQV12 040646	F\$HW = 000013
ASSEMB= 000010	CDAT5 002444 G	C\$INIT= 000011	EXQV13 043034	F\$INIT= 000006
A1 017024	CDAT6 002446 G	C\$INLP= 000020	EXQV14 043736	F\$JMP = 000050
A10 033612	CDAT7 002450 G	C\$MANI= 000050	EXQV15 044720	F\$MOD = 000000
A11 035360	CDAT8 002452 G	C\$MEM = 000031	EXQV16 046256	F\$MSG = 000011
A12 037130	CDAT9 002454 G	C\$MSG = 000023	EXQV17 046640	F\$PROT= 000021
A2 023124	CHAN 002374 G	C\$OPEN= 000034	EXQV18 051730	F\$PWR = 000017
A81 030060	CNT1 002404 G	C\$PNTB= 000014	EXQV19 053040	F\$RPT = 000012
A82 031242	COPA1 020402	C\$PNTF= 000017	EXQV2 013332	F\$SEEG = 000003
A9 032026	COPA10 034546	C\$PNTS= 000016	EXQV20 054112	F\$SOFT= 000005
BAD 002502 G	COPA11 036314	C\$PNTX= 000015	EXQV21 055162	F\$SRV = 000010
BARX 002302 G	COPA12 040152	C\$QIO = 000377	EXQV22 057176	F\$SUB = 000002
BCINP 002240 G	COPA2 024620	C\$RDBU= 000007	EXQV23 063200	F\$SW = 000014
BCRX 002304 G	COPA9 032776	C\$REFG= 000047	EXQV24 067200	F\$TEST= 000001
BGIN1 010534 G	COPB1 021264	C\$RESE= 000033	EXQV25 070736	GETPRM 011244
BGIN2 010710 G	COPB2 025562	C\$REVI= 000003	EXQV26 072470	GOOD 002500 G
BIT0 = 000001 G	COPC1 021616	C\$RFLA= 000021	EXQV27 075402	G\$CNTD= 000200
BIT00 = 000001 G	COPC2 026122	C\$RPT = 000025	EXQV3 014036	G\$DELM= 000372
BIT01 = 000002 G	COPD1 022460	C\$SEFG= 000046	EXQV4 015516	G\$DISP= 000003
BIT02 = 000004 G	COPD2 027000	C\$SPRI= 000041	EXQV5 016726	G\$EXCP= 000400
BIT03 = 000010 G	CSRMSK 002362 G	C\$SVEC= 000037	EXQV6 022774	G\$HILI= 000002
BIT04 = 000020 G	CSRMS1 002364 G	C\$TPRI= 000013	EXQV7 027330	G\$LOLI= 000001
BIT05 = 000040 G	CSRMS2 002366 G	DFPTBL 002216 G	EXQV9 033450	G\$NO = 000000
BIT06 = 000100 G	CSRNAM 014042	DIAGMC= 000000	E\$END = 002100	G\$OFFS= 000400
BIT07 = 000200 G	CSRX 002300 G	DMAHAD 004715	E\$LOAD= 000035	G\$OFFSI= 000376
BIT08 = 000400 G	CULPA 010220 G	DPA1 002312 G	E101 005012 G	G\$PRMA= 000001
BIT09 = 001000 G	C\$AU = 000052	DPA2 002314 G	E200 005057 G	G\$PRMD= 000002
BIT1 = 000002 G	C\$AUTO= 000061	EF .CON= 000036 G	E222 005675 G	G\$PRML= 000000
BIT10 = 002000 G	C\$BRK = 000022	EF .NEW= 000035 G	E231 006010 G	G\$RADA= 000140
BIT11 = 004000 G	C\$BSEG= 000004	EF .PWR= 000034 G	E232 006072 G	G\$RADB= 000000
BIT12 = 010000 G	C\$BSUB= 000002	EF .RES= 000037 G	E233 006123 G	G\$RADD= 000040
BIT13 = 020000 G	C\$CEFG= 000045	EF .STA= 000040 G	E234 006200 G	G\$RADL= 000120
BIT14 = 040000 G	C\$CLCK= 000062	EMG101 004116	E235 006231 G	G\$RADO= 000020
BIT15 = 100000 G	C\$CLEA= 000012	EMG201 004170	E250 005726 G	G\$XFER= 000004
BIT2 = 000004 G	C\$CLOSE= 000035	EMG202 004330	E271 076634	G\$YES = 000010
BIT3 = 000010 G	C\$CLP1= 000006	EMG203 004265	E301 005124 G	HELP = 000000
BIT4 = 000020 G	C\$CVEC= 000036	EMG231 004651	E302 005162 G	HOE = 100000 G
BIT5 = 000040 G	C\$DCLN= 000044	EMG401 004425	E303 005230 G	HPM1 077010
BIT6 = 000100 G	C\$DODU= 000051	EMG402 004521	E400 000172	HPM2 077046
BIT7 = 000200 G	C\$DRPT= 000024	EMG501 004560	E401 005265 G	HPM3 077104
BIT8 = 000400 G	C\$DU = 000053	ERNU 002432 G	E402 005316 G	HPM4 077164
BIT9 = 001000 G	C\$EDIT= 000003	ERRBLK 002512 G	E403 005353 G	HPM5 077226
BOE = 000400 G	C\$ERDF= 000055	ERRMSG 002510 G	E501 005406 G	HPM6 077271
BUFAB 002356 G	C\$ERHR= 000056	ERRNBR 002506 G	E502 005447 G	IBE = 010000 G
BUFBB 002360 G	C\$ERRO= 000060	ERRTYP 002504 G	E801 005510 G	ICRHX 002270 G
CDAT1 002434 G	C\$ERSF= 000054	ERR101 003454 G	E802 005566 G	ICRLX 002266 G
CDAT10 002456 G	C\$ERSO= 000057	ERR201 003500 G	E901 005644 G	ICRNAM 013350
CDAT11 002460 G	C\$ESCA= 000010	ERR202 003556 G	FINIT1 012046	ICRX 002264 G
CDAT12 002462 G	C\$ESEG= 000005	ERR231 003744 G	FINIT2 012136	IDRHX 002276 G
CDAT13 002464 G	C\$ESUB= 000003	ERR401 003616 G	FMDROP 012370	IDRLX 002274 G
CDAT14 002466 G	C\$ETST= 000001	ERR402 003654 G	F\$AU = 000015	IDRNAM 013355
CDAT15 002470 G	C\$EXIT= 000032	ERR501 003702 G	F\$AUTO= 000020	IDRX 002272 G
CDAT16 002472 G	C\$GETB= 000026	EVL = 000004 G	F\$BGN = 000040	IDU = 000040 G
CDAT17 002474 G	C\$GETW= 000027	EXINI 011220	F\$CLEA= 000007	IER = 020000 G
CDAT18 002476 G	C\$GMAN= 000043	EXQV1 012624	F\$DU = 000016	IIRHX 002254 G
CDAT2 002436 G	C\$GPHR= 000042	EXQV10 035220	F\$END = 000041	IIRLX 002252 G



PARAMETER CODING  
SYMBOL TABLE

MACRO M1113 06-SEP-82 16:46 PAGE 74-2

IIRNAM	013336	J\$JMP	= 000167	L\$LOAD	002100 G	L10047	052030	PRI03	= 000140 G
IIRX	002250 G	KPAR0	= 172340	L\$LUN	002074 G	L10050	053140	PRI04	= 000200 G
INFO1	076244	KPAR1	= 172342	L\$MREV	002050 G	L10051	054210	PRI05	= 000240 G
INFO2	076374	KPAR2	= 172344	L\$NAME	002000 G	L10052	055260	PRI06	= 000300 G
INFO3	076502	KPAR3	= 172346	L\$PRIO	002042 G	L10053	057246	PRI07	= 000340 G
INTERR	010162 G	KPAR4	= 172350	L\$PROT	011112 G	L10054	063266	PSEU1	042022
INTFC1	002376 G	KPAR5	= 172352	L\$PRT	002112 G	L10055	067266	PSEU16	045530
INTFC2	002400 G	KPAR6	= 172354	L\$REPP	002062 G	L10056	071010	PSEU17	047504
INTSC1	010142 G	KPAR7	= 172356	L\$REV	002010 G	L10057	072542	PSEU18	056222
INTSC2	010152 G	KPDR0	= 172300	L\$RPT	011104 G	L10060	075454	PSEU23	060344
ISR	= 000100 G	KPDR1	= 172302	L\$SOFT	077336 G	L10061	076720	PSEU24	064372
ISRHX	002262 G	KPDR2	= 172304	L\$SPC	002056 G	L10062	077010	PSEU27	074136
ISRLX	002260 G	KPDR3	= 172306	L\$SPCP	002020 G	L10063	077570	PSEU33	062026
ISRNAM	013343	KPDR4	= 172310	L\$SPTP	002024 G	L10064	100010	PSEU34	066114
ISRX	002256 G	KPDR5	= 172312	L\$STA	002030 G	L10066	100024	PSEU5	044376
ITAC13	041002	KPDR6	= 172314	L\$SW	002234 G	MAINB	002242 G	PVCON	007724 G
ITAC14	043150	KPDR7	= 172316	L\$STEST	002114 G	MASCOM	002352 G	QVP	002234 G
ITAC15	044054	LOCATE	012636 G	L\$TIML	002014 G	MASK	002350 G	QVT3	014010
ITAC16	045030	LOE	= 040000 G	L\$UNIT	002012 G	MCINP	002236 G	REGADD	002354 G
ITAC17	046376	LOGDEV	002372 G	L10000	002232	MCRHX	002310 G	REGERR	006612 G
ITAC18	050756	LOOP	011060	L10001	002244	MCRNAM	013362	REGMSG	007077
ITAC19	052066	LOT	= 000010 G	L10002	003476	MCRX	002306 G	REGTST	006262 G
ITAC20	053204	L\$ACP	002110 G	L10003	003554	MEMINI	007176 G	REGTS1	006320 G
ITAC21	054254	L\$APT	002036 G	L10004	003614	MLA1	002410 G	RERR1	006660
ITAC22	055320	L\$AU	012422 G	L10005	003652	MLA2	002412 G	RERR2	006732
ITAC23	057306	L\$AUT	002070 G	L10006	003700	MMFLG	002332 G	RERR3	007010
ITAC24	063322	L\$AUTO	012230 G	L10007	003742	MM22	002330 G	RSAVE	002402 G
ITAC25	067326	L\$CCP	002106 G	L10010	004110	MSA1	002420 G	RXADRH	002422 G
ITAC26	071050	L\$CLEA	012314 G	L10011	006656	MSIZE	007672	RXADRL	002424 G
ITAC27	072724	L\$CO	002032 G	L10012	010140	MTA1	002414 G	SDPA	002406 G
ITAC28	075736	L\$DEPO	002011 G	L10013	010150	MTA2	002416 G	SFPTBL	002234 G
ITRAC1	012510	L\$DESC	003414 G	L10014	010160	NEWST	011236	SIZEFA	002346 G
ITRAC2	012736	L\$DESP	002076 G	L10015	010216	NEXTBA	075774	SPRM2	077427
ITRAC3	013464	L\$DEVP	002060 G	L10016	011110	NXM	010132 G	SPRM4	077471
ITRAC4	014130	L\$DISP	002124 G	L10020	012226	NXMFLG	002326 G	SPRM5	077520
ITRAC5	015626	L\$DLY	002116 G	L10021	012312	ONEFIL	= 000001	SR0	= 177572
ITRCNT	002322 G	L\$DTP	002040 G	L10022	012336	OSAPTS	= 000000	SR1	= 177574
ITRDEF	002320 G	L\$DTYP	002034 G	L10023	012420	OSAU	= 000001	SR2	= 177576
IXE	= 004000 G	L\$DU	012340 G	L10024	012426	OSBGNR	= 000000	SR3	= 172516
ISAU	= 000041	L\$DUT	002072 G	L10025	012702	OSBGNS	= 000001	STARST	011224
ISAUTO	= 000041	L\$DVTY	003352 G	L10026	012640	OSDU	= 000001	SVCGBL	= 000000
ISCLN	= 000041	L\$EF	002052 G	L10027	013422	OSERRT	= 000001	SVCINS	= 000001
ISDU	= 000041	L\$ENVI	002044 G	L10030	014074	OSGNSW	= 000001	SVCSUB	= 000001
ISHRD	= 000041	L\$ERRT	002504 G	L10031	015572	OSPOIN	= 000001	SVCTAG	= 000001
ISINIT	= 000041	L\$ETP	002102 G	L10032	016760	OSSETU	= 000001	SVCTST	= 000001
ISMOD	= 000041	L\$EXP1	002046 G	L10033	023060	PHHIGH	002334 G	S\$LSYM	= 010000
ISMSG	= 000041	L\$EXP4	002064 G	L10034	027414	PHHSIZ	002342 G	TABD	002514 G
ISPROT	= 000040	L\$EXP5	002066 G	L10035	031762	PHLOW	002336 G	TABE	002624 G
ISPTAB	= 000041	L\$HARD	076724 G	L10036	033546	PHLSIZ	002344 G	TABF	002734 G
ISPWR	= 000041	L\$HIME	002120 G	L10037	035314	PLEV	002316 G	TABG	003044 G
ISRPT	= 000041	L\$HPCP	002016 G	L10040	037064	PMQVP	077400	TABH	003146 G
ISSEG	= 000041	L\$HPTP	002022 G	L10041	040742	PNT	= 001000 G	TABK	003250 G
ISSETU	= 000041	L\$HW	002216 G	L10042	043114	PNTF	002324 G	TQVP6	022636
ISSFT	= 000041	L\$ICP	002104 G	L10043	044020	PRI	= 002000 G	TQVP7	027172
ISSRV	= 000041	L\$INIT	011120 G	L10044	044774	PRI00	= 000000 G	TRIMSG	076563
ISSUB	= 000041	L\$LADP	002026 G	L10045	046342	PRI01	= 000040 G	TSHD1	012642
ISTST	= 000041	L\$LAST	100004 G	L10046	050720	PRI02	= 000100 G	TSHD10	035224

PARAMETER CODING  
SYMBOL TABLE

MACRO M1113 06-SEP-82 16:46 PAGE 74-3

TSHD11	036772	TSHD8	031730	TSSAVL=	177777	TSSSEG=	010001	T26	071012	G
TSHD12	040652	TSHD9	033454	TSSSEGL=	177777	TSSSOF=	010063	T27	072544	G
TSHD13	043040	TXADRH	002426	TSSSEK0=	010001	TSSSRV=	010026	T28	075456	G
TSHD14	043742	TXADRL	002430	TSSSIZE=	000010	TSSSW =	010001	T3	013424	G
TSHD15	044724	T\$ARGC=	000001	TSSUBN=	000000	T\$TES=	010061	T3SEC	013500	
TSHD16	046262	T\$CODE=	075004	T\$TAGL=	177777	T1	012430	T4	014076	G
TSHD17	050644	T\$ERRN=	005361	T\$TAGN=	010067	T10	033550	T5	015574	G
TSHD18	051734	T\$EXCP=	000000	T\$TEMP=	000000	T11	035316	T6	016762	G
TSHD19	053044	T\$FLAG=	000041	T\$TEST=	000034	T12	037066	T7	023062	G
TSHD2	013367	T\$FREE=	100024	T\$TSTM=	177777	T13	040744	T8	027416	G
TSHD20	054116	T\$GMAN=	000000	T\$TSTS=	000001	T14	043116	T9	031764	G
TSHD21	055166	T\$HILI=	003777	T\$SAU =	010024	T15	044022	UAM =	000200	G
TSHD22	057202	T\$LAST=	000001	T\$SAUT=	010021	T16	044776	UNIMSK	013514	
TSHD23	063204	T\$LOLI=	000001	T\$SCLE=	010022	T17	046344	VECC1	002244	G
TSHD24	067204	T\$LSYM=	010000	T\$SDAT=	010066	T18	050722	VECC2	002246	G
TSHD25	070742	T\$LTNO=	000034	T\$SDU =	010023	T19	052032	VIADD	002340	G
TSHD26	072474	T\$NEST=	177777	T\$SHAR=	010062	T2	012704	VPCON	010024	G
TSHD27	075406	T\$NSO =	000000	T\$SHW =	010000	T20	053142	WAIT	011072	G
TSHD28	076220	T\$NS1 =	000005	T\$SINI=	010020	T21	054212	X\$ALWA=	000000	
TSHD3	014047	T\$NS2 =	000003	T\$MSG=	010011	T22	055262	X\$FALS=	000040	G
TSHD4	015522	T\$PCNT=	000000	T\$MPC =	000001	T23	057250	X\$OFFS=	000400	G
TSHD5	016732	T\$PTAB=	010065	T\$PRO=	010017	T24	063270	X\$TRUE=	000020	G
TSHD6	023000	T\$PTHV=	000001	T\$PTA=	010065	T25	067270	\$PATCH	077570	G
TSHD7	027334	T\$PTNU=	000001	T\$RPI=	010016					

. ABS. 100024 000  
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 26992 WORDS ( 106 PAGES)  
DYNAMIC MEMORY: 20774 WORDS ( 79 PAGES)  
ELAPSED TIME: 00:19:40  
ZIEABO.BIN,ZIEABO.SEQ=LIBA/ML,ZIEABO.SRC