

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-T363B-MC  
PRODUCT NAME: CZLAAB0 REPAIR DIAG  
PRODUCT DATE: JANUARY 1983  
MAINTAINER: DISTRIBUTED SYSTEMS DIAGNOSTIC ENGINEERING  
AUTHOR: MICHAEL CINNAMON

COPYRIGHT (C) 1983 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS  
ALL RIGHTS RESERVED

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

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

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DEC  
DECUS

PDP  
DECTAPE

UNIBUS  
VAX

MASSBUS

D I G I T A L

## 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
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
4.0	PERFORMANCE AND PROGRESS REPORTS
5.0	DEVICE INFORMATION TABLES
6.0	TEST SUMMARIES

67USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 4  
 CZUAAB.MAC 07-APR-83 17:03

.PAGE  
 2  
 .REM 8

## 1.0 GENERAL INFORMATION

### 1.1 PROGRAM ABSTRACT

THIS PRODUCT IS THE PDP-11 REPAIR LEVEL DIAGNOSTIC FOR THE UNIBUS TO NI ADAPTER (DEUNA). THIS DIAGNOSTIC WAS DESIGNED TO DETECT STATIC AND DYNAMIC HARDWARE FAILURES IN THE DEUNA BOARDSET. THE DEUNA BOARDSET IS THE TWO MODULES WHICH PLUG INTO THE PDP-11 UNIBUS. THE TWO MODULES ARE THE M7792 PORT MODULE AND THE M7793 LINK MODULE. THIS DIAGNOSTIC IS CAPABLE OF TESTING EIGHT SUCH BOARDSETS ON A SINGLE PDP-11 UNIBUS.

THIS DIAGNOSTIC WILL ONLY RUN IN A STANDALONE, OFFLINE ENVIRONMENT. THE DEUNA IS LOGICALLY REMOVED FROM THE 'WIRE' BY THE DIAGNOSTIC, SO NO MESSAGES FROM OTHER NODES ON THE NETWORK, TO THE DEUNA UNDER TEST, WILL DISRUPT THE TESTING PROCESS. HOWEVER, BECAUSE THIS DIAGNOSTIC RUNS THE DEUNA SELF-TEST IN TEST 9, AND THE SELF-TEST PERFORMS AN EXTERNAL LOOPBACK AS PART OF ITS TESTING PROCEDURE, IT IS RECOMMENDED THAT THE DEUNA TRANSCEIVER CABLE BE REMOVED FROM THE H4000 TRANSCEIVER AND PLUGGED INTO A FIELD SERVICE EXTERNAL LOOPBACK CONNECTOR.

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 AND PAPER TAPE. 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

THE FOLLOWING LIST OF HARDWARE IS REQUIRED TO RUN THIS DIAGNOSTIC:

PDP-11 CPU  
 28K WORDS OF MEMORY  
 CONSOLE TERMINAL  
 DEUNA BOARDSET (M7792, M7793)  
 PLUS, ONE OF THE FOLLOWING:  
 -LINK BOARD TO BULKHEAD CABLE CONNECTED AND BULKHEAD TO  
 TRANSCEIVER TAP CABLE CONNECTED (NORMAL ONLINE CONFIGURATION)  
 OR  
 -LINK BOARD TO BULKHEAD CABLE CONNECTED AND BULKHEAD TO FIELD  
 SERVICE EXTERNAL LOOPBACK CONNECTOR INSTALLED (OFFLINE CONFIGURATION)

### 1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL CHQS  
 XXDP+ PROGRAMMERS MANUAL  
 DEUNA LINK BOARD FUNCTIONAL SPECIFICATION  
 DEUNA PORT BOARD FUNCTIONAL SPECIFICATION  
 DEUNA PROGRAMMING SPECIFICATION

67USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 5  
 CZUAAB.MAC 07-APR-83 17:03

#### 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THIS DIAGNOSTIC ASSUMES THAT THE PDP-11 PROCESSOR AND MEMORY ARE IN WORKING CONDITION AND IS CAPABLE OF EXECUTING PDP-11 INSTRUCTIONS NORMALLY. THE UNIBUS IS EXPECTED TO BE FULLY OPERATIONAL I.E. ANY PROBLEMS REPORTED BY THIS DIAGNOSTIC, ABOUT THE INTEGRITY OF THE UNIBUS, ARE ASSUMED TO BE THE RESULT OF A FAILURE ON THE DEUNA AND NOT THE FAULT OF OTHER DEVICES CONNECTED TO THE UNIBUS.

THIS DIAGNOSTIC DOES NOT REQUIRE ANY PRELIMINARY TESTS BE EXECUTED ON THE DEUNA, NOR DOES RUNNING OF ANY OTHER TESTS PRIOR TO RUNNING THIS DIAGNOSTIC, AFFECT THE OPERATION OF THE TESTS IN THIS DIAGNOSTIC.

FOR A COMPLETE TEST OF THE DEUNA, ALL THE AVAILABLE DIAGNOSTIC SOFTWARE SHOULD BE RUN. THIS WOULD INCLUDE RUNNING THE DEUNA FUNCTIONAL DIAGNOSTIC AND THE DEUX-11 SYSTEM EXERCISOR WITH THE DEUNA MODULE SELECTED.

#### 1.5 ASSUMPTIONS

THIS DIAGNOSTIC ASSUMES THAT THE DEUNA WILL NOT HANG THE UNIBUS WHEN AN ACCESS IS MADE TO ANY ONE OF THE PCSR REGISTERS. THE DEUNA IS CAPABLE OF ASSERTING ACLO ON THE UNIBUS, THIS FEATURE COULD, IF BROKEN, CAUSE THE UNIBUS TO HANG. THIS TYPE OF FAILURE IS NOT DETECTED BY THE DIAGNOSTIC.

PORTIONS OF THIS DIAGNOSTIC USE SPECIAL DIAGNOSTIC MICROCODE THAT IS LOADED INTO THE DEUNA WRITEABLE CONTROL STORE. THIS MICROCODE ALLOWS THE DIAGNOSTIC MORE VISIBILITY INTO THE INTERNALS OF THE DEUNA HARDWARE AS WELL AS NOT RELYING AS HEAVILY ON THE COMPLEX OPERATIONAL MICROCODE IN ROM, HOWEVER, THIS INCREASES THE DIAGNOSTIC'S COMPLEXITY SOMEWHAT. THEREFORE, IT IS ASSUMED THAT THE USER OF THIS DIAGNOSTIC IS FAMILIAR WITH THE DEUNA ENOUGH TO READ DEUNA MICROCODE SHOULD AN ERROR OCCUR.

#### 2.0 OPERATING INSTRUCTIONS

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

#### 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 BTEST 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

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 'DDDD'.

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:DDDD	EXECUTE DDDDD PASSES (DDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDD = 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
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 FLAG 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)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST

\*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

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.

WHAT IS THE PCSRO ADDRESS?

THIS IS THE ADDRESS AT WHICH PCSRO RESIDES ON THE UNIBUS.  
THIS ADDRESS IS SWITCH SELECTABLE ON THE PORT MODULE.  
THE ALLOWABLE RANGE IS 160000-177776.

WHAT IS THE VECTOR ADDRESS?

THIS IS THE INTERRUPT VECTOR ADDRESS. THIS ADDRESS IS ALSO  
SWITCH SELECTABLE ON THE PORT MODULE. THE ALLOWABLE RANGE  
IS 000-776.

SAMPLE DIALOGUE:

DR>START

CHANGE HW (L) ? Y

# UNITS (D) ? 2

UNIT 0

WHAT IS THE PCSRO ADDRESS? (O) ? 170000

WHAT IS THE VECTOR ADDRESS? (O) ? 700

UNIT 1

WHAT IS THE PCSRO ADDRESS? (O) ? 170010

WHAT IS THE VECTOR ADDRESS? (O) ? 710

## 2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER ANY QUESTIONS THE MONITOR ASKS
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

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.

## 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 ON UNIT NUMBER TST NUMBER PC:XXXXXX  
ERROR MESSAGE

.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 "IBR" 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", "IBE" OR "IXE" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

#### 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

AT THE COMPLETION OF THE FIRST PASS OF EACH DEUMA BEING TESTED, INFORMATION FOR THAT DEUMA IS PRINTED. THIS PRINTOUT CONTAINS THE ETHERNET DEFAULT ADDRESS (OBTAINED BY READING THE PHYSICAL ADDRESS ROM), THE OPERATIONAL MICROCODE ROM VERSION NUMBER, AND THE SWITCH PACK SETTINGS FOR SELF TEST LOOPING AND REMOTE BOOTING.

EXAMPLE PRINTOUT:

ETHERNET DEFAULT ADDRESS (HEX): AA-00-03-00-00-02

ROM MICROCODE VERSION (DECIMAL): 1

SWITCH PACK SET FOR :

SELF TEST LOOP DISABLED

REMOTE BOOT ENABLED

#### 6.0 TEST SUMMARIES

TEST 1: PCSRO READ ACCESS TEST

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 0 CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS APPEAR IN THE EXPECTED BIT POSITIONS.



**TEST 2: PCSR1 READ ACCESS TEST**

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS APPEAR IN THE EXPECTED BIT POSITIONS.

**TEST 3: PCSR2 READ ACCESS TEST**

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 2 CAN BE READ FROM THE UNIBUS

**TEST 4: PCSR3 READ ACCESS TEST**

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 3 CAN BE READ FROM THE UNIBUS

**TEST 5: RESET TEST**

THIS TEST WILL VERIFY THE RESET STATE FOR ALL DEUNA UNIBUS REGISTERS

**TEST 6: PCSR2 REGISTER READ/WRITE TEST**

THIS TEST WILL CHECK THE REGISTER FOR ALL SA0 AND SA1 ERRORS (STUCK AT 0 AND STUCK AT 1 ERRORS). THE HOST WILL WRITE PATTERNS TO THE REGISTER AND READ THEM BACK TO VERIFY. THE PATTERNS TO BE USED ARE AT THE LABEL PATERN:: IN THE GLOBAL DATA SECTION OF THIS PROGRAM.

NOTE: SINCE PCSR2 BIT 00 IS ALWAYS PRESET TO LOGIC 0, THE LOWEST ORDER BIT OF THE PATTERN WILL BE MASKED BEFORE DOING THE COMPARISON.

**TEST 7: REGISTER PCSR3 READ/WRITE TEST**

THIS TEST WILL WRITE PATTERNS TO THE WRITEABLE FIELD OF PCSR3 AND WILL READ THESE BACK FOR VERIFICATION.

**TEST 8: NOP TEST**

THIS TEST WILL VERIFY THAT THE DEUNA PROCESSOR IS ALIVE AND CAN RESPOND TO A PORT COMMAND ISSUED. THE NOP PORT COMMAND WILL BE ISSUED TO THE DEUNA IN PCSR0 BITS 3:0 AND WILL WAIT FOR THE 'DNI' BIT TO SET IN PCSR0.

THE NOP PORT COMMAND USES A MINIMUM OF HARDWARE BUT FORCES THE T11 TO EXECUTE THE PORT COMMAND SEQUENCE.

**TEST 9: SELF TEST**

THIS TEST VERIFIES THAT THE ROM BASED SELF TEST CAN BE RUN SUCCESSFULLY WHEN INVOKED VIA THE SELF TEST PORT COMMAND.

#### TEST 10: DEUNA ROM DUMP TEST

THIS TEST WILL VERIFY THAT THE DATA PATH FROM THE T11 PROCESSOR TO THE UNIBUS INTERFACE IS INTACT AND ABLE TO TRANSFER DATA RELIABLY. THIS DATA PATH IS CRUCIAL FOR FURTHER TESTING BECAUSE IT IS NECESSARY FOR LOADING REPAIR-LEVEL DIAGNOSTICS INTO THE WCS.

THE TEST STRATEGY IS TO TRANSFER KNOWN DATA OVER THE DATA PATH AND TO VERIFY THE TRANSFERRED DATA.

THE DATA SOURCE FOR THE DUMP TEST IS THE ROM MICROCODE RESIDENT ON THE DEUNA PORT BOARD. A DUMP OF THE ROM WILL EXERCISE THE DATA PATH NEEDED FOR LOADING WCS AND THE ROM CONTENTS CAN BE VERIFIED. THE ROM MICROCODE WILL BE CHECKED BY VERIFYING THE CRC BYTES. THE CRC BYTES CHARACTERIZE THE DATA CONTENTS OF THE ROM AND ARE BURNED INTO THE ROM AT THE TIME OF MANUFACTURE. A FAILURE TO VERIFY THE CRC CALCULATION ON THE DUMPED ROM DATA DUMP WILL BE INTERPRETED AS AN ERROR IN THE DATA PATH.

#### TEST 11: WCS LOAD/DUMP TEST

THIS TEST WILL USE THE LOAD/DUMP PORT COMMAND TO VERIFY THE DATA PATHWAY TO/FROM THE WCS. PATTERNS WILL BE USED TO CHECK THE DATA PATHWAY FOR ALL SA0 AND SA1 ERRORS.

BECAUSE THE OPERATIONAL MICROCODE NEEDS THE LOWER 2K OF WCS ONLY THE TOP HALF OF WCS WILL BE LOADED WITH A DATA PATTERN THEN DUMPED BACK TO MEMORY FOR VERIFICATION. THIS PROCEDURE WILL BE REPEATED FOR ALL PATTERNS.

#### TEST 12: LOAD AND START FUNCTION TEST

THIS TEST WILL VERIFY THAT THE LOAD AND START MICROADDRESS PORT COMMAND IS OPERATIONAL.

THE PROCESS IS TO LOAD WCS WITH MICROCODE THAT WHEN STARTED WILL WRITE A PATTERN OF DATA TO THE LITE-BYTE FIELD OF PCSR1 REGISTER WHICH CAN BE READ FROM THE UNIBUS AND BE VERIFIED.

NOTE: THIS TEST USES MICROCODE MODULE 'A'

#### TEST 13: COMPREHENSIVE WCS MEMORY TEST

THIS TEST WILL EXHAUSTIVELY TEST THE WCS MEMORY. CUSTOM MICROCODE MODULE B, MICROTEST #1 IS USED TO DO THE ACTUAL TESTING. MICROTEST #1 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE WCS MEMORY CHECKING FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.

PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED

PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR  
PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION  
PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN  
PCBB+6: CONTAINS THE DATA THAT WAS READ

#### TEST 14: INTERRUPT VECTOR TEST

THIS TEST WILL VERIFY THAT THE INTERRUPT INTERFACE LOGIC OF THE DEUNA IS CAPABLE OF GENERATING AN INTERRUPT VECTOR AND ARBITRATING FOR CONTROL OF THE UNIBUS.  
THE DEUNA INTERRUPT ENABLE BIT WILL BE SET AND AN INTERRUPT WILL BE CAUSED BY ISSUING A NOP PORT COMMAND. AN INTERRUPT IS EXPECTED AT THE CORRECT VECTOR AND AT THE CORRECT PRIORITY.

#### TEST 15: PCSRO INTERRUPT BIT TEST

THIS TEST WILL VERIFY THAT EACH OF THE INTERRUPT BITS IN REGISTER PCSRO CAN CAUSE AN INTERRUPT.

EACH OF THE INTERRUPTS OF REGISTER PCSRO IS SET UNDER THE CONTROL OF THE T11 AND NOT DIRECTLY BY HARDWARE. THE T11 THEREFORE CAN INITIATE UNIBUS INTERRUPTS BY SETTING BITS IN REGISTER PCSRO.

THIS TEST USES MICROMODULE C, MICROTEST #1.  
MICROCODE MODULE C IS LOADED IF NOT ALREADY DONE SO BY A PREVIOUS TEST.

THE DEUNA INTERRUPT VECTOR IS SETUP TO STORE THE CONTENTS OF PCSRO WHEN THE INTERRUPT OCCURS. PCBB+0 IS LOADED WITH THE INTERRUPT BIT THAT IS TO BE TESTED THEN PCSRO COMMAND BITS ARE LOADED WITH A 1 TO TELL THE T11 TO EXECUTE MICROTEST #1. WE WAIT FOR THE INTERRUPT TO OCCUR THEN SEE IF THE CONTENTS OF PCSRO AT THE TIME OF THE INTERRUPT CONTAINED THE CORRECT INTERRUPT BIT. THE TEST IS REPEATED FOR ALL THE INTERRUPT BITS.

#### TEST 16: TIMER TEST

THIS TEST WILL USE THE CUSTOM MICROCODE MODULE 'C' TO CHECK THE OPERATION OF THE TIMER.  
THE TIMER IS ACCESSIBLE ONLY TO THE T11 PROCESSOR. THE HOST PROCESSOR CAN START THE TIMER ONLY WITH THE ASSISTANCE OF THE T11 PROCESSOR.

FOR THIS TEST THE MICROCODE WILL BE LOADED ONLY IF IT HAS NOT ALREADY BEEN DONE BY A PREVIOUS TEST.

WHEN THE MICROCODE IS STARTED THE T11 WILL START THE TIMER AND WILL SET 'DNI' WHEN THE TIMING INTERVAL HAS EXPIRED. THE INTERVAL IS 10 SECONDS.

ANY TIME FROM 8 TO 12 SECONDS IS AN ACCEPTABLE RANGE.

#### TEST 17: LINK MEMORY TEST

THIS TEST WILL EXHAUSTIVELY TEST THE LINK MEMORY.

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 13  
 CZUAB.MAC 07-APR-83 17:03

THE LINK MEMORY OCCUPIES THE 16-32K ADDRESS SPACE OF THE T-11. CUSTOM MICROCODE MODULE C MICROTEST #3 IS USED TO DO THE ACTUAL TESTING. MICROTEST #3 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE LINK MEMORY CHECKING FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.

PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED  
 PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR,  
 PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION  
 PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN  
 PCBB+6: CONTAINS THE DATA THAT WAS READ

MICROSUBTEST #	DESCRIPTION
1	ACCESS TEST
2	ADDRESS SHIFT TEST
3	DATA LATCH TEST
4	ADDRESS BIT SHIFT #1
5	ADDRESS BIT SHIFT #2
6	MARCH TEST

#### TEST 18: DMA 'TO' ADDRESS TEST

THIS TEST WILL VERIFY THAT THE INTERNAL REGISTER 'DMATO' CAN BE READ AND WRITTEN. THE T11 WILL BE USED TO WRITE AND READ THIS REGISTER. THIS TEST REQUIRES THE USE OF CUSTOM MICROCODE MODULE C MICROTEST #4. PCBB+0 WILL BE WRITTEN WITH THE DATA PATTERN TO TEST, THE T11 WILL WRITE THIS PATTERN TO THE 'DMATO' REGISTER AND READ IT BACK AND PUT THE DATA READ INTO PCBB+2. THE DATA AT PCBB+2 WILL BE VERIFIED.

#### TEST 19: DMA 'FROM' ADDRESS REGISTER TEST

THIS TEST CHECKS THE OPERATION OF THE REGISTER/COUNTER THAT CONTAINS THE ADDRESS OF THE LINK MEMORY WORD TO BE MOVED TO THE HOST DURING DMA OPERATIONS. THE REGISTER CAN BE WRITTEN BY THE T11 BUT IT CAN NOT BE READ BACK FOR VERIFICATION, THEREFORE IT MUST BE CHECKED INDIRECTLY.

THE METHOD USED IS TO LOAD MICROCODE MODULE C IF IT HAS NOT ALREADY BEEN DONE. THE MICROTEST #5 LOADS EACH LOCATION OF LINK MEMORY WITH ITS ADDRESS THEN IT TAKES THE CONTENTS OF PCBB+0 AND LOADS IT INTO THE DMA 'FROM' ADDRESS REGISTER, THE 'TO' REGISTER IS LOADED WITH THE ADDRESS OF PCBB+2, THE WORD COUNT IS LOADED FOR A ONE WORD TRANSFER AND THE DMA ENGINE IS STARTED. THE HOST VERIFIES PCBB+2 = PCBB+0

#### TEST 20: DMA BLOCK TRANSFER TEST

THIS TEST WILL VERIFY THAT THE DMA ENGINE CAN TRANSFER A MAXIMUM SIZE DATA BLOCK TO HOST MEMORY.

THIS TEST USES CUSTOM MICROCODE MODULE C, MICROTEST #6. THE MICROTEST FILLS EACH LOCATION OF LINK MEMORY WITH ITS ADDRESS AND THEN SETS UP A TRANSFER FROM LINK MEMORY TO THE ADDRESS POINTED TO BY PCBB+0. THE TRANSFER SIZE IS 1776 WORDS. AFTER THE MICROTEST FINISHES THE BUFFER IS CHECKED TO SEE IF IT CONTAINS THE INCREMENTING ADDRESS PATTERN.

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 14  
 CZUAAB.MAC 07-APR-83 17:03

#### TEST 21: TRANSMIT DONE TEST

THE TRANSMIT STATE MACHINE INFORMS THE POPT MODULE PROCESSOR OF A 'TRANSMIT DONE' CONDITION. IT DOES THIS BY GENERATING AN INTERRUPT WHENEVER IT FINISHES TRANSMITTING A DATAGRAM. SINCE THE 'TRANSMIT DONE' INTERRUPT IS A NECESSARY CONDITION OF EVERY DATAGRAM TRANSMISSION, THIS TEST WILL USE THE INTERRUPT TO INDICATE THAT THE TRANSMIT STATE MACHINE IS FUNCTIONING.

MICROCODE MODULE D MICROTEST #1 WILL BE USED FOR THIS TEST. IT SETS UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A TRANSMIT INTERRUPT. THE T-11 WILL BE RELEASED FROM THE LOOP IF THE XMIT DONE INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING THAT THE TEST IS COMPLETE.

#### TEST 22: RECEIVER DONE TEST

THE LINK HARDWARE INCLUDES LOGIC TO TELL THE DEUNA PROCESSOR WHEN A LINK MEMORY BUFFER HAS BEEN FILLED AND DATA IS AVAILABLE FOR PROCESSING. THE HARDWARE INTERRUPTS THE DEUNA PROCESSOR. BECAUSE THE INTERRUPT HAPPENS WHEN A LINK MEMORY BUFFER IS FULL AND THE LINK MEMORY IS FILLED BY THE OPERATION OF THE RECEIVE STATE MACHINE, THE INTERRUPT CAN BE USED TO CHECK IF THE STATE MACHINE WORKS.

MICROCODE MODULE D MICROTEST #2 WILL BE USED FOR THIS TEST. IT SETS UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A RECEIVER INTERRUPT. THE T-11 WILL BE RELEASED FROM THE LOOP IF THE INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING THAT THE TEST IS COMPLETE.

#### TEST 23: DATA BYTE FRAMING TEST

THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR BYTE DATA BOUNDARY CONDITIONS.

THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT BYTES IN THE DATA STREAM (I.E. 111111110000000011...) THE T-11 PROCESSOR WILL VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO THE RECEIVER DATA BUFFER.

THIS TEST WILL USE MICROCODE MODULE 'D' MICROTEST #3. TESTING OF THE DATA FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE, WILL WAIT FOR A 'DNI' IN REGISTER PCSRO. IF 'DNI' APPEARS, THE HOST PROCESSOR WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET, ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVER STATUS WORD  
 PCBB+2: DATA TRANSMITTED  
 PCBB+4: DATA RECEIVED  
 PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA

#### TEST 24: DATA WORD FRAMING TEST

THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR WORD DATA BOUNDARY CONDITIONS.

THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT WORDS IN THE DATA STREAM (I.E. 11111111111111110000000000000011...) THE T-11 PROCESSOR WILL VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO THE RECEIVER DATA BUFFER.

THIS TEST WILL USE MICROCODE MODULE 'D' MICROTEST #4. TESTING OF THE DATA FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE, WILL WAIT FOR A 'DNI' IN REGISTER PCSR0. IF 'DNI' APPEARS, THE HOST PROCESSOR WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET, ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVER STATUS WORD  
PCBB+2: DATA TRANSMITTED  
PCBB+4: DATA RECEIVED  
PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA

#### TEST 25: DATA PATH PATTERN TEST

THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR ALL 'STUCK AT 0' AND 'STUCK AT 1' ERRORS.

THE T-11 PROCESSOR WILL TRANSMIT DATAGRAMS OF MAXIMUM LENGTH IN LOOPBACK MODE. THIS PATTERN LOOPBACK PROCEDURE WILL BE USED FOR ALL PATTERNS OF UP TO WORD WIDTH.

THIS TEST USES MICROMODULE 'D' MICROTEST #5 TO DO THE TESTING. THE HOST PROCESSOR WILL PASS A DATA PATTERN TO THE T-11 PROCESSOR THROUGH THE PCBB. THE T-11 WILL FILL A XMIT BUFFER WITH THE DATA PATTERN AND TRANSMIT THE DATAGRAM OVER THE LOOPBACK. THE T-11 PROCESSOR WILL VERIFY THE PATTERN IN THE RECEIVER BUFFER. IF THE T-11 FINDS AN ERROR, IT WILL WRITE THE FAILING PATTERN TO THE PCBB ALONG WITH THE OFFSET INTO THE RECEIVER BUFFER AT WHICH THE PATTERN WAS FOUND. IT WILL INFORM THE HOST OF THE ERROR BY SETTING PCSR1 TO AN ERROR CONDITION. THE PCBB IS FORMATTED AS FOLLOWS:

PCBB+0: DATA PATTERN  
PCBB+2: RECEIVER STATUS WORD  
PCBB+4: BAD DATA PATTERN  
PCBB+6: OFFSET INTO RECEIVER BUFFER WHERE BAD DATA WAS FOUND

#### TEST 26: STATUS MUX VERIFICATION TEST

THE LINK WRITES STATUS IN LINK MEMORY AFTER EACH TRANSMIT ATTEMPT. THE STATUS GIVES INFORMATION ABOUT THE ATTEMPTED OPERATION. THE STATUS INFORMATION IS WRITTEN INTO THE FIRST TWO LOCATIONS OF THE TRANSMIT BUFFER. THIS INFORMATION IS ACCESSIBLE TO THE T-11 BY SIMPLY READING IT FROM LINK MEMORY.

THIS TEST WILL VERIFY THAT THE STATUS INFORMATION IS WRITTEN INTO THE FIRST LOCATION OF THE TRANSMIT BUFFER. THE TEST WILL ALSO CHECK THE SECOND WORD OF THE TRANSMIT BUFFER.

THIS TEST WILL USE MICROMODULE 'D' MICROTEST #6. WHEN THE TEST IS STARTED, THE T-11 PROCESSOR WILL SET UP THE LINK FOR LOOPBACK OF A DATA PATTERN. A BACKGROUND PATTERN WILL BE WRITTEN INTO THE FIRST WORD OF THE TRANSMIT BUFFER. THIS WORD SHOULD BE OVER-WRITTEN BY THE STATUS WHEN

6800ER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 16  
 CZUAAS.MAC 07-APR-83 17:03

THE BUFFER IS TRANSMITTED. THE SECOND WORD OF THE TRANSMIT BUFFER CAN NOT BE WRITTEN WITH A BACKGROUND BECAUSE IT MUST DESIGNATE THE TRANSMIT BYTE COUNT.

WHEN THE DATAGRAM HAS BEEN LOOPED BACK, THE T-11 PROCESSOR WILL PASS THE FIRST TWO WORDS OF THE TRANSMIT BUFFER TO THE HOS: THRU THE PCBB+0 AND PCBB+2.

PCBB+0: FIRST WORD OF TRANSMIT BUFFER  
 PCBB+2: SECOND WORD OF TRANSMIT BUFFER

THE CORRECT STATUS SHOULD BE:

TRANSMIT STATUS WORD 0 BITS 15,09:00 SHOULD BE ALL 0 AND  
 BIT 13 SHOULD BE A 1

TRANSMIT STATUS WORD 1 BITS 15:13 SHOULD ALL BE 0

#### TEST 27: LINK BYTE COUNTER TEST

BYTE COUNTERS ARE INVOLVED BOTH WITH THE LINK TRANSMIT FUNCTION AND THE LINK RECEIVE FUNCTION. WHEN THE T-11 PREPARES A TRANSMIT BUFFER FOR TRANSMISSION OF A DATAGRAM, IT WRITES THE INTENDED BYTE COUNT IN THE SECOND WORD OF THE TRANSMIT BUFFER. WHEN TRANSMISSION OF THE TRANSMIT BUFFER BEGINS, THE BYTE COUNT VALUE IS USED TO LOAD THE TRANSMIT BYTE COUNTER. THIS COUNTER IS DECREMENTED BY THE TRANSMIT STATE MACHINE AS THE DATAGRAM IS TRANSMITTED. THE DATAGRAM TRANSMISSION WILL CONTINUE UNTIL THE BYTE COUNTER IS DECREMENTED TO ZERO.

THE RECEIVER ALSO HAS A BYTE COUNTER. THIS COUNTER IS CLEARED AT THE START OF A DATAGRAM RECEPTION AND IS INCREMENTED BY THE RECEIVE STATE MACHINE AS THE DATAGRAM IS RECEIVED. THE VALUE IN THIS COUNTER IS WRITTEN INTO WORD TWO OF THE RECEIVE BUFFER AT THE END OF RECEPTION.

THIS TEST WILL USE MICROMODULE 'D' MICROTEST #7. THIS TEST WILL VERIFY THE BYTE COUNT LOGIC BY LOOPING MESSAGES AND VERIFYING THAT THE BYTE COUNT APPEARING IN THE RECEIVE BUFFER CORRESPONDS TO THE BYTE COUNT THAT WAS WRITTEN TO THE TRANSMIT BYTE COUNTER. THE TEST WILL ALSO VERIFY THAT THE ACTUAL NUMBER OF BYTES TRANSFERRED TO THE RECEIVE BUFFER CORRESPONDS TO THE INTENDED BYTE COUNT.

THE TRANSMIT BYTE COUNT IS PASSED TO THE T-11 VIA THE PCBB+0. AFTER THE DATAGRAM LOOPBACK THE RECEIVE BYTE COUNT IS PLACED INTO PCBB+2 BY THE T-11 PROCESSOR. PCBB+4 IS LOADED BY THE T-11 PROCESSOR WITH THE ACTUAL NUMBER OF BYTES THAT WERE TRANSFERRED TO THE RECEIVER BUFFER.

PCBB+0: TRANSMIT BYTE COUNT  
 PCBB+2: RECEIVE BYTE COUNT  
 PCBB+4: ACTUAL NUMBER OF BYTES RECEIVED

#### TEST 28: ODD BYTE TEST

THIS TEST WILL VERIFY THAT THE LINK CAN TRANSMIT AND RECEIVE DATAGRAMS HAVING ONLY ODD BYTE COUNTS.

THIS TEST IS IDENTICAL TO THE PREVIOUS BYTE COUNTER TEST WITH THE ONLY

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 17  
 CZUAAB.MAC 07-APR-83 17:03

EXCEPTION THAT IT PASSES ONLY ODD BYTE COUNTS TO THE MICROCODE. IT ALSO USES MICROMODULE 'D' MICROTEST #7

#### TEST 29: LINK MAXIMUM BYTE COUNTER TEST

THE RECEIVE BYTE COUNTER IS A 12 BIT BINARY COUNTER THAT COUNTS THE NUMBER OF BYTES THAT WERE RECEIVED DURING A DATAGRAM TRANSMISSION. THE BYTE COUNTER IS INCREMENTED AS EACH BYTE IS RECEIVED. THE RECEIVE BYTE COUNTER HAS LOGIC THAT DISABLES THE COUNTER IF THE MAXIMUM VALUE IS REACHED AND PREVENTS THE COUNTER FROM ROLLING OVER TO ZERO.

THIS TEST WILL CHECK THAT THE COUNTER 'TOPS OUT' AT THE MAXIMUM COUNTER VALUE. IT DO THIS MICROMODULE 'D' MICROTEST #9 IS USED. IT WILL TRANSMIT A DATAGRAM OF MAXIMUM COUNTER LENGTH OVER THE LOOPBACK. THE LINK CRC HARDWARE WILL BE ALLOCATED TO THE TRANSMIT SIDE SO THAT CRC BYTES WILL APPENDED TO THE DATAGRAM. THE LENGTH OF THE DATAGRAM WILL THEREFORE EXCEED THE LENGTH OF THE RECEIVE BYTE COUNTER. THE RECEIVE COUNTER WILL BE CHECKED TO INSURE THAT THE COUNTER HAS REMAINED AT THE MAXIMUM VALUE, IF NOT AN ERROR IS PASSED TO THE HOST.

#### TEST 30: FIFO TEST

THERE ARE TWO FIFO'S USED IN THE DEUNA TO KEEP TRACK OF RECEIVER BUFFERS. THE FIRST IS CALLED THE 'RECEIVER BUFFER AVAILABLE FIFO' AND THE SECOND IS CALLED THE 'RECEIVER BUFFER DONE FIFO'.

THE T11 LOADS THE RECEIVER BUFFER AVAILABLE FIFO WITH A LIST OF UNUSED 1K BUFFERS IN LINK MEMORY. WHEN THE DEUNA SENSES THAT A PACKET IS COMING IN IT PULLS AN AVAILABLE BUFFER ADDRESS FROM THE OUTPUT OF THE RECEIVER BUFFER AVAILABLE FIFO AND USES IT TO ADDRESS LINK MEMORY FOR THE STORAGE OF THE RECEIVED DATA. AFTER THE DATA HAS BEEN LOADED THE RECEIVER STATE MACHINE PUTS THE USED BUFFER ADDRESS INTO THE RECEIVER BUFFER DONE FIFO WHERE AN INTERRUPT IS GENERATED TO THE T11 WHEN IT BUBBLES TO THE TOP OF THE FIFO.

THESE FIFO'S ARE 64 DEEP BY 4 BITS WIDE. THE OPERATIONAL MICROCODE ONLY FILLS THE FIFO TO A MAXIMUM OF 16. THE 4 BIT WIDTH REPRESENTS BITS 14-11 OF THE LINK MEMORY ADDRESS. THESE BITS ALLOW THE ADDRESSING OF A 1K BUFFER IN LINK MEMORY.

THIS TEST WILL VERIFY THAT THE RECEIVE BUFFER AVAILABLE FIFO AND THE RECEIVER BUFFER DONE FIFO OPERATE CORRECTLY. THIS WILL BE DONE BY LOADING THE RECEIVER BUFFER AVAILABLE FIFO WITH A 1K BUFFER ADDRESS THEN A PACKET WILL BE TRANSMITTED IN LOOPBACK MODE. AFTER THE RECEIVER INTERRUPT OCCURS THE RECEIVER BUFFER DONE FIFO IS READ AND THE ADDRESS IS COMPARED WITH WHAT WAS GIVEN THE RECEIVER BUFFER AVAILABLE FIFO. THEY SHOULD BE THE SAME IF EVERYTHING IS WORKING CORRECTLY. THE OPERATION IS PERFORMED WITH THE TRANSMITTER BUFFER SET TO 0 AND WILL BE REPEATED WITH RECEIVER BUFFERS 1-15. THIS TEST WILL USE MICROMODULE 'D' MICROTEST #10.

PARAMETERS PASSED TO THE MICROCODE WILL BE FORMATTED IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVE BUFFER ADDRESS  
 PCBB+2: RECEIVE BUFFER COMPLETED ADDRESS



68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 18  
 CZUAAB.MAC 07-APR-83 17:03

### TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

THIS TEST WILL VERIFY THAT BUFFERS 1-15 OF LINK MEMORY CAN BE ADDRESSED CORRECTLY BY THE RECEIVER. THIS WILL BE DONE BY DIRECTING THE MICROCODE TO TRANSMIT A DATA PATTERN FROM BUFFER 0 AND TO RECEIVE THE DATA IN BUFFER X WHERE X = 1-15. THEN A CHECK WILL BE MADE TO SEE IF THE PATTERN NOT ONLY ARRIVED IN THE CORRECT RECEIVER BUFFER BUT THAT THE PATTERN DOES NOT SHOW UP ANYWHERE ELSE IN LINK MEMORY EXCEPT WHERE IT WAS SUPPOSE TO.

THIS TEST WILL USE MICROMODULE 'D' MICROTEST #11. THIS MICROTEST ACCEPTS 2 PARAMETERS: THE TRANSMIT BUFFER AND THE RECEIVER BUFFER. IT WILL SET UP A DATA PATTERN IN THE TRANSMIT BUFFER AND TELL THE LINK TO TRANSMIT, IN LOOPBACK MODE, FROM THE TRANSMIT BUFFER GIVEN TO THE RECEIVER BUFFER GIVEN. AFTER THE RECEIVER INTERRUPT, THE DATA IS CHECKED IN THE EXPECTED RECEIVER BUFFER FOR THE CORRECT DATA PATTERN. THEN ALL OF LINK MEMORY (EXCEPT FOR THE TRANSMIT BUFFER) IS CHECKED TO SEE IF THE PATTERN ENDS UP ELSEWHERE. IF AN ERROR IS FOUND THE MICROCODE PASSES THE ADDRESS OF LINK MEMORY WHERE THE ERROR WAS FOUND, THE DATA THAT WAS FOUND THERE ALONG WITH THE DATA THAT SHOULD HAVE BEEN THERE.

THE PARAMETERS FOR THE MICROCODE ARE FORMATED IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVER BUFFER ADDRESS  
 PCBB+2: TRANSMIT BUFFER ADDRESS  
 PCBB+4: LINK MEMORY ADDRESS (IF ERROR)  
 PCBB+6: GOOD DATA PATTERN (IF ERROR)  
 PCBB+10: BAD DATA PATTERN (IF ERROR)

### TEST 32: TRANSMITTER LINK MEMORY ADDRESS TEST

THIS TEST WILL VERIFY THAT BUFFERS 1-15 CAN BE CORRECTLY ADDRESSED BY THE TRANSMITTER. THIS TEST IS IDENTICAL TO THE RECEIVER ADDRESS TEST IN THAT IT USES THE SAME MICROCODE (MICROMODULE 'D' MICROTEST #11) EXCEPT IT FIXES THE RECEIVER BUFFER AT 0 AND VARIES THE TRANSMIT BUFFER FROM 1-15.

### TEST 33: LINK MEMORY ARBITRATION TEST

THE LINK MEMORY CAN BE ACCESSED BY FOUR PROCESSES; THE T-11 PROCESSOR, THE DMA ENGINE, THE RECEIVE STATE MACHINE AND THE TRANSMIT STATE MACHINE. THE PORT MODULE HAS ARBITRATION CIRCUITRY TO MANAGE LINK MEMORY ACCESSES. THIS CIRCUITRY PREVENTS CONFLICTS BETWEEN PROCESSES AND ASSURES THAT HIGHER PRIORITY PROCESSES GET PRECEDENCE.

THIS TEST WILL VERIFY THE ABILITY OF THE LINK MEMORY ARITRATOR TO HANDLE SIMULTANEOUS REQUESTS BY FOUR PROCESSES. EACH OF THESE PROCESSES WILL INVOLVE TASKS THAT REQUIRE HEAVY ACCESSES OF LINK MEMORY. DATA WILL BE MOVED INTO OR OUT OF LINK MEMORY BY EACH. WHEN THAT TASKS ARE FINISHED THE DATA WILL BE VERIFIED.

THE FOUR PROCESSES ARE:

1-TRANSMIT STATE MACHINE

WILL TRANSMIT A DATAGRAM OF MAXIMUM DATA LENGTH IN LOOPBACK

68158R DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 19  
 CZUAAB.MAC 07-APR-83 17:03

MODE. THE DATA FIELD WILL CONTAIN A BIT PATTERN STRING OF TWO 1'S FOLLOWED BY TWO 0'S I.E. 31463 (OCTAL).

#### 2-RECEIVE STATE MACHINE

WILL RECEIVE A DATAGRAM OF MAXIMUM DATA LENGTH OVER THE LOOPBACK. THE RECEIVE DATA BUFFER WILL BE FILLED WITH 0'S PRIOR TO THE RECEPTION.

#### 3-T-11 MICROPROCESSOR DMA

A 1K BUFFER IN LINK MEMORY WILL BE FILLED WITH AN ALL 1'S DATA PATTERN PRIOR TO THE OPERATION THEN ALTERNATING 1'S AND 0'S DATA PATTERN WILL BE WRITTEN.

#### 4-DMA ENGINE

WILL TRANSFER A 1K BLOCK OF DATA FROM LINK MEMORY TO UNIBUS MEMORY. THE DATA IN LINK MEMORY WILL A BIT PATTERN STRING OF FOUR 1'S FOLLOWED BY A STRING OF FOUR 0'S. THE BUFFER IN UNIBUS MEMORY WILL BE CLEARED PRIOR TO THE OPERATION.

THE FOUR PROCESSES WILL WORK OUT OF FOUR SEPARATE AREAS OF LINK MEMORY.

```

BASE OF LINK MEMORY--> +-----+
                        ! RECEIVE BUFFER !
BASE + 4000---> +-----+
                        ! TRANSMIT BUFFER !
BASE + 10000--> +-----+
                   ! DMA ENGINE BUFFER !
BASE + 14000--> +-----+
                  ! T-11 PROCESSOR BUFFER !
                   +-----+
  
```

THIS WILL ALLOW THE ARITR/TION CIRCUITRY TO BE TESTED AND YET ALLOWS THE DATA TO BE VERIFIED EASILY AND ASSOCIATED WITH A SINGLE PROCESS.

A DATAGRAM WILL BE LOOPED BACK FROM THE TRANSMIT BUFFER TO THE RECEIVE BUFFER. AS THE DATAGRAM IS BEING TRANSFERRED, THE T-11 PROCESSOR WILL FILL IT'S BUFFER AND THE DMA ENGINE WILL TRANSFER IT'S BUFFER FROM LINK MEMORY TO UNIBUS MEMORY.

WHEN THE RECEIVE STATE MACHINE IS DONE, THE T-11 WILL VERIFY THE DATA IN THE RECEIVE BUFFER. IF AN ERROR IS FOUND PCSR1 WILL BE SET TO INDICATE AN ERROR.

THE HOST WILL WAIT FOR THE MICROCODE TO FINISH AND WHEN DONE, IT WILL VERIFY THE DATA TRANSFERRED BY THE DMA ENGINE TO UNIBUS MEMORY.

THE FIRST 3 WORDS OF THE PCBB ARE USED FOR ERROR INFORMATION, THE REST WILL BE THE UNIBUS ADDRESS THAT THE DMA ENGINE WILL TRANSFER TO.

```

PCBB+0: +-----+
         ! EXPECTED PATTERN !
PCBB+2: +-----+
         ! ACTUAL PATTERN !
PCBB+4: +-----+
         ! LINK MEMORY ADDRESS !
  
```

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 20  
 CZUAB.MAC 07-APR-83 17:03



#### TEST 34: STATION ADDRESS PATTERN TEST

WITHOUT EITHER THE PROMISCUOUS MODE OR THE MULTICAST MODE ENABLED, THE LINK LOGIC WILL RECOGNIZE DATAGRAM ADDRESSES ONLY IF THE ADDRESS IS CONTAINED IN THE STATION ADDRESS RAM.

WHEN A DATAGRAM ARRIVES, THE LINK LOGIC COMPARES THE DATAGRAM DESTINATION ADDRESS FIELD TO THE 12 ADDRESSES WRITTEN IN THE STATION ADDRESS RAM. IF THE INCOMING ADDRESS MATCHES ONE OF THESE, THEN THE DATAGRAM WILL BE ACCEPTED BY THE LINK. THE 'MATCH' BIT IS SET IN THE TRANSMIT BUFFER AND THE RECEIVING PROCESS BEGINS.

THIS TEST WILL VERIFY THAT THE LINK CAN RECOGNIZE A DATAGRAM WHEN THE DESTINATION ADDRESS OF THE DATAGRAM MATCHES ONE OF THE ADDRESSES STORED IN THE STATION ADDRESS RAM.

THIS TEST WILL USE MICROMODULE 'E' MICROTEST #1. PATTERNS WILL BE USED FOR ADDRESSES IN CHECKING THE STATION ADDRESS LOGIC. THE PATTERNS WILL BE SUPPLIED TO THE T-11 THROUGH THE PCBB. THE MICROCODE WILL BE RESTARTED FOR EACH DIFFERENT PATTERN TO BE TESTED. UPON START-UP, THE T-11 PROCESSOR WILL PICK UP THE CURRENT PATTERN/ADDRESS, LOAD THE SAME PATTERN INTO ALL 12 LOCATIONS OF THE STATION ADDRESS RAM, FORMAT THE TRANSMIT BUFFER AND LOGIC FOR A LOOPBACK, PRESET PCSR1 TO AN ERROR CONDITION, START THE LINK AND WAIT FOR THE MATCH BIT IN THE TRANSMIT BUFFER. IF THE MATCH BIT SETS THE PCSR1 ERROR CONDITION IS CLEARED AND THE T-11 WAITS FOR BOTH THE TRANSMITTER AND RECEIVER INTERRUPTS BEFORE IT SETS 'DNI' TO INDICATE THE TEST WAS SUCCESSFUL

THE PCBB WILL BE USED TO PASS THE 48 BIT STATION ADDRESS PATTERN:



THE FOLLOWING PATTERNS WILL BE USED:

- ALTERNATING 1'S AND 0'S
- ALTERNATING 0'S AND 1'S
- PAIR OF 0'S FOLLOWED BY PAIR OF 1'S
- FOUR 0'S FOLLOWED BY FOUR 1'S
- EIGHT 0'S FOLLOWED BY EIGHT 1'S
- SIXTEEN 1'S FOLLOWED BY SIXTEEN 0'S FOLLOWED BY 16 1'S

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 21  
 CZUAAB.MAC 07-APR-83 17:03

-TWENTYFOUR 1'S FOLLOWED BY TWENTYFOUR 0'S

**TEST 35: STATION ADDRESS REJECTION TEST**

THIS TEST WILL VERIFY THAT THE STATION ADDRESS DETECTION LOGIC DOES NOT RECOGNIZE A DATAGRAM WHEN THE DATAGRAM ADDRESS IS NOT CONTAINED IN THE STATION ADDRESS RAM.

THE MICROCODE WILL FILL THE STATION ADDRESS RAM WITH 0'S. THE DESTINATION FIELD OF THE TRANSMIT BUFFER IS FILLED WITH 1'S. A TRANSMISSION IS STARTED IN LOOPBACK MODE AND THE T-11 WILL WAIT FOR A RECEIVER INTERRUPT. OF COURSE, THE RECEIVER INTERRUPT SHOULD NEVER HAPPEN BECAUSE THE STATION ADDRESS LOGIC SHOULD NOT GET A SUCCESSFUL COMPARISON BETWEEN 0'S IN THE DESTINATION ADDRESS OF THE INCOMING DATAGRAM AND THE 1'S IN THE STATION ADDRESS RAM. THE T-11 WILL BE PUT INTO A LOOP WAITING FOR A RECEIVER INTERRUPT AND THE DEUNA TIMER IS STARTED. IF THE LOOP IS BROKEN BY THE RECEIVER INTERRUPT AN ERROR WILL BE PRESENTED IN PCSR1 BY THE MICROCODE. IF THE LOOP IS BROKEN BY THE TIMER THEN THE TEST WAS SUCCESSFUL.

**TEST 36: STATION ADDRESS RAM POSITION TEST**

THE STATION ADDRESS RAM CAN HOLD UP TO 12 STATION ADDRESSES. WHEN A DATAGRAM IS RECEIVED THE STATION ADDRESS COMPARISON LOGIC DOES A BIT-WISE COMPARISON OF ALL 12 RAM STATION ADDRESS WITH THE INCOMING DATAGRAM STATION ADDRESS.

THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE A STATION ADDRESS REGARDLESS OF THE LOCATION OF THE ADDRESS IN THE STATION ADDRESS RAM.

THIS TEST WILL USE MICROMODULE 'E' MICROTEST #4. THE MICROCODE WILL WRITE A STATION ADDRESSES OF ALL 1'S INTO A SINGLE LOCATION OF THE STATION ADDRESS RAM. THE OTHER ELEVEN LOCATION WILL BE LOADED WITH 0'S. A DATAGRAM WITH AN ALL 1'S DESTINATION ADDRESS WILL BE TRANSMITTED IN LOOPBACK MODE. THE TEST WILL VERIFY THAT THE DATAGRAM IS RECEIVED. THE TEST WILL BE REPEATED FOR ALL TWELVE LOCATIONS OF THE STATION ADDRESS RAM.

THE MICROTEST WILL BE REPEATED FOR EACH OF THE 12 TEST ITERATIONS. THE PCBB WILL BE USED TO PASS TO THE MICROCODE WHICH POSITION IS TO BE LOADED WITH 1'S. WHEN THE STATION ADDRESS IS LOADED, THE STATION ADDRESSES MUST BE ROTATED ORTHOGONALLY, I.E. BIT 0 OF ALL STATION ADDRESSES LOADED TOGETHER, THEN BIT 1, THEN BIT 2 ETC. THIS MAKES IT DIFFICULT TO DESCRIBE THE POSITION OF ANY SINGLE STATION ADDRESS IN TERMS OF AN OFFSET FROM THE RAM BASE ADDRESS.

THE PCBB IS FORMATTED AS FOLLOWS:

```
PCBB+0:      +-----+
              !  RAM ADDRESS POSITION  !
              +-----+
```

**TEST 37: MULTICAST ADDRESS TEST**

MULTICAST ADDRESSING PERMITS THE DEUNA TO RESPOND TO MESSAGES AIMED AT LOGICALLY RELATED DEVICES ON THE NETWORK. THE MSB OF THE DESTINATION ADDRESS

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 22  
 CZUAAB.MAC 07-APR-83 17:03

OF THESE MESSAGES IS A 1. THIS BIT IS DETECTED BY THE ADDRESS RECOGNITION LOGIC.

THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE AND ACCEPT MESSAGES WITH THE MULTICAST BIT DESIGNATION.

THIS TEST WILL USE MICROMODULE 'E' MICROTEST #4. THE MICROCODE WILL PREPARE A DATAGRAM WITH THE DESTINATION ADDRESS HAVING THE MULTICAST BIT SET. THE DEUNA WILL BE SETUP IN LOOPBACK MODE WITH 'ENABLE ALL MULTICAST'. THE DATAGRAM WILL BE TRANSMITTED AND THE T-11 WILL BE PUT IN A LOOP WAITING FOR A RECEIVER INTERRUPT. THE TIMER WILL INTERRUPT THE LOOP IF THE RECEIVER INTERRUPT DOES NOT OCCUR. IF THIS HAPPENS, PCSR1 WILL INDICATE AN ERROR. OTHERWISE WHEN THE RECEIVER INTERRUPT OCCURS IT WILL BREAK THE LOOP AND PCSR1 WILL INDICATE A SUCCESSFUL COMPLETION OF THE TEST.

#### TEST 38: CRC DATA PATTERN TEST

THE LINK MODULE HAS HARDWARE TO GUARANTEE THAT DATAGRAMS HAVE NOT BEEN CORRUPTED DURING TRANSMISSION AND RECEPTION. THE HARDWARE GENERATES A CRC FOR DATAGRAMS TRANSMITTED AND VERIFIES THE CRC FOR DATAGRAMS RECEIVED. THE CRC IS A 32 BIT NUMBER GENERATED BY DIVIDING THE DATAGRAM BIT STREAM BY A CRC POLYNOMIAL. THE DIVISION RESULTS IN A UNIQUE NUMBER THAT CAN ONLY BE REPRODUCED IN CRC CALCULATIONS IF THE BIT STREAM EXACTLY MATCHES THE ORIGINAL. THE CRC IS CALCULATED DURING DATAGRAM TRANSMISSION AND IS APPENDED TO THE PACKET. THE CRC IS TRANSMITTED AS PART OF THE PACKET. THE CRC IS AGAIN CALCULATED WHEN THE DATAGRAM IS RECEIVED AND THE CALCULATED IS COMPARED TO THE CRC TRANSMITTED. IF THE DATAGRAM HAS BEEN FAITHFULLY TRANSMITTED, THE CRC'S SHOULD MATCH EXACTLY.

THE DEUNA CALCULATES THE CRC WITH DEDICATED CRC LOGIC. THE LOGIC IS EITHER DEDICATED TO THE CALCULATION OF THE OUTGOING DATAGRAM OR THE CALCULATION OF THE INCOMING DATAGRAM, BUT NOT BOTH.

THIS TEST WILL VERIFY THE OPERATION OF THE CRC CALCULATION CIRCUITRY. MICROMODULE 'F' MICROTEST #1 WILL BE USED. THE MICROCODE WILL TRANSMIT DATAGRAMS IN LOOPBACK MODE. THE CRC HARDWARE WILL BE DEDICATED TO THE TRANSMITTER. WHEN THE DATAGRAM IS RECEIVED THE T-11 WILL CALCULATE A CRC ON THE DATA RECEIVED (INCLUDING THE TRANSMITTED CRC). THE RESULT OF THIS CALCULATION WILL BE A 32 BIT CONSTANT. THIS CONSTANT IS THEN COMPARED TO WHAT WAS EXPECTED AND IF THEY DO NOT MATCH. AN ERROR IS PLACED IN PCSR1.

PATTERNS WILL BE PASSED TO THE MICROCODE THROUGH THE PCBB. THE MICROCODE WILL FILL THE TRANSMIT BUFFER WITH THIS PATTERN BEFORE EACH TRANSMISSION TAKES PLACE.

THE PCBB WILL BE FORMATTED AS FOLLOWS:

```
PCBB+0:      +-----+
              ! DATA PATTERN !
              +-----+
```

#### TEST 39: CRC ERROR TEST

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 23  
 CZUAAB.MAC 07-APR-83 17:03

THIS TEST WILL VERIFY THAT THE LINK CRC CIRCUITRY CAN DETECT A BAD CRC.

MICROMODULE 'F' MICROTEST #2 WILL BE USED. THE MICROCODE WILL TRANSMIT DATAGRAMS IN LOOPBACK MODE. EACH DATAGRAM WILL HAVE AN ERRONEOUS CRC APPENDED TO THE DATA FIELD. THE DEUMA CRC LOGIC WILL BE SETUP SUCH THAT THE CRC LOGIC WILL BE DEDICATED TO THE RECEIVER. THIS IS EXPECTED TO CAUSE A CRC ERROR.

THE DATA FIELDS OF EACH DATAGRAM WILL CONSIST OF PATTERNS. THE PATTERNS WILL BE PASSED TO THE MICROCODE VIA THE PCBB.

AFTER THE RECEIVER INTERRUPT THE MICROCODE WILL PASS THE RECEIVER STATUS WORD 0 BACK VIA PCBB+2. THE CRC BIT IN THIS WORD IS CHECKED TO SEE IF IT IS SET.

THE PCBB IS FORMATTED AS FOLLOWS:

```

PCBB+0:      +-----+
              | DATA PATTERN |
              +-----+
PCBB+2:      | RECEIVER STATUS WORD |
              +-----+
  
```

#### TEST 40: CRC PATTERN LENGTH TEST

THIS TEST WILL VERIFY THAT THE RECEIVE CRC HARDWARE CAN CALCULATE CRC FOR DATAGRAMS OF VARYING LENGTHS.

DATAGRAMS WILL BE TRANSMITTED FOR THE TRANSMIT BUFFER TO THE RECEIVE BUFFER IN LOOPBACK MODE. THE TRANSMIT CRC WILL BE DISABLED WHICH WILL ASSIGN THE CRC LOGIC TO CALCULATION OF INCOMING DATAGRAMS. THE CRC FOR TRANSMIT DATAGRAMS WILL BE CALCULATED BY THE MICROCODE. IT IS EXPECTED THAT THE CRC LOGIC WILL VERIFY THE CRC APPENDED TO THE DATAGRAM AS IT IS BEING RECEIVED.

PATTERNS WILL BE USED TO FILL THE DATAGRAM DATA FIELD. THE PATTERNS WILL BE PASSED TO THE MICROCODE THROUGH THE PCBB ALONG WITH THE BYTE COUNT TO BE USED.

AFTER THE RECEPTION OF THE DATAGRAM THE RECEIVER STATUS WORD WILL BE PASSED BACK VIA THE PCBB SO IT CAN BE CHECKED

THE PCBB IS FORMATTED AS FOLLOWS:

```

PCBB+0:      +-----+
              | DATA PATTERN |
              +-----+
PCBB+2:      | BYTE COUNT |
              +-----+
PCBB+4:      | RECEIVE STATUS WORD |
              +-----+
  
```

#### TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

THIS TEST WILL CHECK THE ABILITY OF THE RECEIVE STATE MACHINE TO REJECT A DATAGRAM OF LESS THAN 64 BYTES AND TO RECOVER THE RECEIVER BUFFER.

USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 24  
 CZUAAB.MAC 07-APR-83 17:03

THIS TEST WILL USE MICROMODULE 'F' MICROTEST #4.  
 EACH TRIAL WILL CONSIST OF TWO DATAGRAM TRANSMISSIONS IN LOOPBACK MODE. EACH TRANSMISSION WILL LOOPBACK A DATAGRAM FILLED WITH UNIQUE DATA. THE FIRST DATAGRAM WILL BE A RUNT OF LESS THAN 64 BYTES. THE SECOND WILL BE A DATAGRAM OF LEGAL SIZE.

EACH TRIAL WILL START WITH THE LINK BUFFER POINTER RESET TO THE FIRST LINK BUFFER. THE RUNT WILL BE TRANSMITTED, THEN THE VALID DATAGRAM. IF THE BUFFER RECOVERY IS WORKING CORRECTLY, THE SECOND DATAGRAM IS EXPECTED TO BE WRITTEN INTO THE SAME LINK MEMORY BUFFER AS WAS THE RUNT.

THIS TEST WILL BE REPEATED WITH VARIOUS RUNT PACKET SIZES.

THE BYTE COUNT FOR THE RUNT PACKET TRANSMISSION WILL BE PASSED VIA THE PCBB. AFTER THE TWO TRANSMISSIONS, THE MICROCODE WILL PASS BACK THE CONTENTS OF THE BUFFER DONE FIFO, AND THE CONTENTS OF THE FIRST DATA WORD OF THE RECEIVER BUFFER.

THE PCBB WILL BE FORMATTED AS FOLLOWS:

```

PCBB+0:  +-----+
          !          RUNT BYTE COUNT          !
          +-----+
PCBB+2:  !  BUFFER DONE FIFO CONTENTS  !
          +-----+
PCBB+4:  !  FIRST DATA WORD OF BUFFER  !
          +-----+

```

#### TEST 42: HALF-DUPLEX TEST

THE LINK INCLUDES A 'HALF DUPLEX' MODE OF OPERATION. THIS MODE CAN BE ENABLED OR DISABLED THROUGH THE LINK MODE REGISTER. THE OPERATIONAL MICROCODE NORMALLY USES HALF-DUPLEX MODE.

IN THE HALF-DUPLEX MODE, THE LINK WILL NOT RECEIVE MESSAGES ADDRESSED TO ITSELF. INCOMING MESSAGES LOOPED BACK WILL BE IGNORED BY THE RECEIVE STATE MACHINE. THE STATE MACHINE WILL NOT ISSUE A 'RECEIVER DONE' INTERRUPT AND THE BUFFER CAN BE RECOVERED FOR RECEIVING A LATER DATAGRAM.

THIS TEST USES MICROMODULE 'F' MICROTEST #5.  
 THIS TEST WILL VERIFY THE OPERATION OF HALF-DUPLEX MODE. A DATAGRAM WILL BE TRANSMITTED IN LOOPBACK MODE WITH THE HALF-DUPLEX BIT SET. THE MICROCODE WILL VERIFY THAT THE RECEIVER INTERRUPT DOES NOT OCCUR. THE MICROCODE WILL THEN CLEAR THE HALF-DUPLEX BIT AND LOOP A DATAGRAM AND VERIFY THAT THE ORIGINAL BUFFER WAS RECOVERED.

THIS TEST WILL USE THE PCBB TO PASS INFORMATION. PCBB+0 WILL BE USED TO PASS THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM IS RECEIVED. PCBB+4 WILL BE USED TO PASS THE FIRST WORD OF DATA FROM THE RECEIVER BUFFER AFTER THE SECOND DATAGRAM IS TRANSMITTED.

```

PCBB+0:  +-----+
          !  CONTENTS OF BUFFER DONE FIFO  !
          +-----+
PCBB+2:  !  FIRST DATA WORD OF BUFFER DONE!

```

68USER DOCUMENTATION MACY11 30A(1052) 07-APR-83 17:13 PAGE 25  
 CZUAAB.MAC 07-APR-83 17:03

----->

THE CONTENTS OF THE BUFFER DONE FIFO SHOULD BE 0 AND THE FIRST DATA WORD SHOULD BE AN ALTERNATING 1'S AND 0'S PATTERN.

### TEST 43: COLLISION TEST

THE RECEIVE STATE MACHINE REACTS TO COLLISIONS ON THE WIRE BY ACTIVATING THE RETRY LOGIC. THE RETRY LOGIC WAITS AN INTERVAL OF TIME BEFORE ATTEMPTING TO RETRANSMIT THE DATAGRAM. THE INTERVALS ARE NOT UNIFORM BUT ARE OF GENERALLY INCREASING PSEUDO-RANDOM DURATION. THE RETRY LOGIC WILL ATTEMPT TO RETRANSMIT UP TO 15 ADDITIONAL TIMES BEFORE GIVING UP.

THIS TEST WILL VERIFY THAT THE RECEIVE STATE MACHINE RESPONDS TO A COLLISION AND THAT THE RETRY SEQUENCE IS REPORTED CORRECTLY IN THE TRANSMIT STATUS WORD.

THIS TEST WILL USE MICROMODULE 'G' MICROTEST #1. THE LINK BOARD CONTAINS DIAGNOSTIC LOGIC THAT ALLOWS COLLISIONS TO BE SIMULATED. WITH THE FORCE COLLISIONS LOGIC ACTIVATED, THE RETRY HARDWARE CAN BE STEPPED THROUGH THE RETRY SEQUENCE. THAT IS, EVERY DATAGRAM LOOPED BACK WILL STEP THE RETRY LOGIC THROUGH ONE STEP OF THE RETRY SEQUENCE. THE RETRY SEQUENCE CAN BE VERIFIED BY CHECKING THE TRANSMIT BUFFER STATUS WORDS AFTER EACH RETRY STEP.

THE PCBB WILL BE USED TO PASS PARAMETERS BETWEEN THE MICROCODE AND THE HOST PROCESSOR. PCBB+0 WILL BE USED TO PASS THE DATA TO BE LOADED INTO THE LINK MODE WORD. PCBB+2 WILL BE PASSED BACK BY THE MICROCODE, IT IS THE FIRST WORD OF THE TRANSMIT BUFFER (TRANSMIT STATUS WORD 0). PCBB+4 WILL ALSO BE PASSED BACK, IT IS TRANSMIT STATUS WORD 1.

THE TRANSMIT STATUS WORDS SHOULD SHOW THE FOLLOWING STATUS:

LOOPBACK STEP #	STATUS BITS			
	WORD 0	WORD 0	WORD 1	WORD 1
	ERRS (14)	MORE (12)	ONE (11)	RETRY (10)
1	0	0	1	0
2-15	0	1	0	0
16	1	0	0	1

THE PCBB IS FORMATTED AS FOLLOWS:

PCBB+0:           <----->  
                   !       LINK MODE WORD       !  
                   <----->



```

PCBB+2:      ! TRANSMIT STATUS WORD 0!
              +-----+
PCBB+4:      ! TRANSMIT STATUS WORD 1!
              +-----+

```

#### TEST 44: TDR COUNTER TEST

THE DEUNA HAS A COUNTER DESIGNED TO HELP LOCATE FAULTS IN THE COAXIAL CABLE. THE COUNTER IS INITIALIZED WHEN A MESSAGE IS TRANSMITTED AND INCREMENTS AS THE DATAGRAM IS TRANSMITTED. COUNTING WILL STOP IF A COLLISION OCCURS OR THE CARRIER IS LOST. COUNTING ALSO STOPS IF THE 10 BIT COUNTER REACHES ITS MODULUS.

THIS TEST WILL DETERMINE THAT THE TDR COUNTER VALUE WILL CHANGE AND THAT THE COUNTER IS NOT STUCK.

BECAUSE THE COUNTER COUNTS DURING TRANSMISSION OF A DATAGRAM AND WILL CONTINUE TO COUNT DURING THE TIME THAT THE TRANSMIT STATE MACHINE OPERATES, THE COUNT ACCUMULATED IN THE COUNTER DURING TRANSMISSION IS PROPORTIONAL TO THE LENGTH OF THE DATAGRAM. THIS TEST WILL USE THIS RELATION TO VERIFY THAT THE COUNTER IS NOT STUCK.

THIS TEST USES MICROMODULE 'G' MICROTEST #2. THE TEST WILL SEND DATAGRAMS OVER THE LOOPBACK. THE LENGTH OF THE DATAGRAM WILL BE VARIED BY USING A INCREASING BYTE COUNT IN THE TRANSMIT BUFFER. AFTER EACH DATAGRAM HAS BEEN LOOPED BACK, THE TRANSMIT BUFFER WORD 1 WILL BE PASSED BACK TO THE HOST TO VERIFY THAT IT IS CORRECT. THE CRITERIA FOR CORRECTNESS WILL BE: INCREASING BYTE COUNTS SHOULD RESULT IN INCREASING TDR VALUES IN TRANSMIT STATUS WORD 1.

THE PCBB WILL BE FORMATED AS FOLLOWS:

```

PCBB+0:      +-----+
              !   BYTE COUNT   !
              +-----+
PCBB+2:      ! TRANSMIT STATUS WORD 1!
              +-----+

```

#### TEST 45: RETRY LOGIC TEST

THE RETRY LOGIC IS ACTIVATED WHENEVER A COLLISION IS ENCOUNTERED DURING A TRANSMISSION ATTEMPT. THE LINK STOPS TRANSMISSION AND WAITS FOR A PERIOD OF TIME BEFORE ATTEMPTING TO RETRANSMIT.

THE WAIT TIME IS AN INTEGRAL NUMBER OF 'SLOT TIMES'. THE NUMBER COMES FROM A RANDOM NUMBER GENERATOR. THE NUMBER OF SLOT TIMES IS NOT EXACTLY RANDOM SINCE THE RETRY LOGIC WAITS A GENERALLY INCREASING NUMBER OF SLOT TIMES BEFORE TRYING TO RETRANSMIT. THIS TEST WILL VERIFY THAT THE RETRY LOGIC IS CAPABLE OF GENERATING VARIABILITY IN THE DURATION OF THE RETRY WAIT TIMES.

THIS TEST WILL USE MICROMODULE 'G' MICROTEST #3. THE LINK MODULE HAS A DIAGNOSTIC MAINTENANCE FACILITY MAKING IT POSSIBLE TO SINGLE STEP THE RETRY LOGIC THROUGH THE MAXIMUM SIXTEEN RETRY STEPS. THIS FEATURE WILL ALSO MAKE IT POSSIBLE TO MEASURE THE RETRY WAIT INTERVAL.

THE MICROCODE WILL SET THE COLLISION BIT IN THE LINK MODE REGISTER AND AND TRANSMIT A DATAGRAM IN LOOPBACK MODE. THE T-11 WILL COUNT WHILE WAITING FOR THE TRANSMIT STATE MACHINE TO INTERRUPT. THE ACCUMULATED COUNT SHOULD PROVIDE A MEASURE OF TIME TAKEN FOR THE TRANSMISSION ATTEMPT TO OCCUR. SINCE THE COLLISION BIT IS SET, THIS INTERVAL WILL INCLUDE THE RETRY WAIT INTERVAL. THE ACCUMULATED COUNT WILL BE WRITTEN BY THE MICROCODE TO THE PCBB.

THE MICROTEST WILL BE EXECUTED 16 TIMES. AFTER EACH EXECUTION, THE COUNT WILL BE READ FROM THE PCBB AND STORED IN A TABLE. THE TABLE WILL BE SCANNED TO VERIFY THAT THEY ARE NOT ALL THE SAME.

THE PCBB IS FORMATTED AS FOLLOWS:

```
PCBB+0:  +-----+
          |         |
          |  BYTE COUNT  |
          |         |
          +-----+
PCBB+2:  +-----+
          |         |
          | TRANSMIT WAIT COUNT |
          |         |
          +-----+
```

TEST 46: PRINT DEVICE PARAMETERS TEST

THIS TEST PRINTS THE DEFAULT PHYSICAL ADDRESS, THE MICROCODE REVISION AND THE SWITCH PACK SETTINGS.

74PROGRAM REVISION HISTORY  
CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 29

1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412

.TITLE PROGRAM REVISION HISTORY

: DATE AUTHOR DESCRIPTION OF CHANGE

:3-FEB-83 (MAC001)

ADD THIS SECTION.  
REMOVE REDUNDANT .MCALL STATEMENTS TO SVC.  
CHANGE INIT CODE TO DELAY A PERIOD OF TIME AFTER A  
POWER FAILURE OCCURS TO ALLOW SELF TEST TO FINISH.  
INCREASE AMOUNT OF TIME TO WAIT FOR DNI AFTER ISSUING  
A RESET TO PCSRO.  
ADD PRINTING OF PCSR'S IF ERROR OCCURS.  
UPDATE SELF TEST ERROR CODES AND DESCRIPTIONS.  
CHANGE TEST 9 TO DECODE PCSR1 IF DNI NEVER HAPPENS.  
CHANGE TEST 5 TO NOT CHECK PORT COMMAND FIELD OF PCSRO.  
UPDATE 'HEADER' STATEMENT TO REV A-2.

:24-MAR-83 (RSJ001)

CHANGE ALL WORD ACCESS TO THE UPPER BYTE OF PCSRO TO  
BYTE ACCESS. INTRODUCED NEW VARIABLES TO DESCRIBE THE  
UPPER BYTE, SAME NAMES WITH B ADDED I.E. DNI -> DNIB.  
ADDED ADDRESS STORAGE VARIABLE PCSROUB AS THE ADDRESS OF  
THE UPPER BYTE OF PCSRO.  
CHANGED HEADER TO REV B-0.

1413  
 1414  
 1415  
 1416  
 1417  
 1418  
 1419 000000'  
 1420  
 1421  
 1422  
 1423  
 1424  
 1425  
 1426 000000'  
 1427  
 1428 000000'  
 1429 000000'  
 1430 000000' 103  
 1431 000001' 132  
 1432 000002' 125  
 1433 000003' 101  
 1434 000004' 101  
 1435 000005' 000  
 1436 000006' 000  
 1437 000007' 000  
 1438 000010'  
 1439 000010' 102  
 1440 000011'  
 1441 000011' 060  
 1442 000012'  
 1443 000012' 000000  
 1444 000014'  
 1445 000014' 000000  
 1446 000016'  
 1447 000016' 053256'  
 1448 000020'  
 1449 000020' 000000  
 1450 000022'  
 1451 000022' 000262'  
 1452 000024'  
 1453 000024' 000000  
 1454 000026'  
 1455 000026' 000000G  
 1456 000030'  
 1457 000030' 000000  
 1458 000032'  
 1459 000032' 000000  
 1460 000034'  
 1461 000034' 000000  
 1462 000036'  
 1463 000036' 000000  
 1464 000040'  
 1465 000040' 000124'  
 1466 000042'  
 1467 000042' 000340  
 1468 000044'

.TITLE PROGRAM HEADER AND TABLES  
 .SBTTL PROGRAM HEADER

.ENABL AMA

BGNMOD

:++  
 : THE PROGRAM HEADER IS THE INTERFACE BETWEEN  
 : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.  
 :--

POINTER BGNRPT,BGNAU,BGNDU

HEADER CZUAA,B,0,0,0,340

LSNAME:: :MAC001  
 .ASCII /C/  
 .ASCII /Z/  
 .ASCII /U/  
 .ASCII /A/  
 .ASCII /A/  
 .BYTE 0  
 .BYTE 0  
 .BYTE 0  
 LSREV::  
 .ASCII /B/  
 LSDEPO::  
 .ASCII /O/  
 LSUNIT::  
 .WORD 0  
 LSTIML::  
 .WORD 0  
 LSHPCP::  
 .WORD LSHARD  
 LSSPCP::  
 .WORD 0  
 LSHPTP::  
 .WORD LSHW  
 LSSPTP::  
 .WORD 0  
 LSLADP::  
 .WORD LSLAST  
 LSSTA::  
 .WORD 0  
 LSCO::  
 .WORD 0  
 LSDTYP::  
 .WORD 0  
 LSAPT::  
 .WORD 0  
 LSDTP::  
 .WORD LSDISPATCH  
 LSPRIO::  
 .WORD 340  
 LSENV1::

1469 000044' 000000  
 1470 000046' 000000  
 1471 000046' 000000  
 1472 000050' 000000  
 1473 000050' 003  
 1474 000051' 003  
 1475 000052' 000000  
 1476 000052' 000000  
 1477 000054' 000000  
 1478 000056' 000000  
 1479 000056' 000000  
 1480 000060' 000000  
 1481 000060' 000700'  
 1482 000062' 021176'  
 1483 000062' 021176'  
 1484 000064' 000000  
 1485 000064' 000000  
 1486 000066' 000000  
 1487 000066' 000000  
 1488 000070' 022026'  
 1489 000070' 022026'  
 1490 000072' 022020'  
 1491 000072' 022020'  
 1492 000074' 000000  
 1493 000074' 000000  
 1494 000076' 000706'  
 1495 000076' 000706'  
 1496 000100' 104035  
 1497 000100' 104035  
 1498 000102' 000000  
 1499 000102' 000000  
 1500 000104' 021212'  
 1501 000104' 021212'  
 1502 000106' 021654'  
 1503 000106' 021654'  
 1504 000110' 021652'  
 1505 000110' 021652'  
 1506 000112' 021204'  
 1507 000112' 021204'  
 1508 000114' 000000  
 1509 000114' 000000  
 1510 000116' 000000  
 1511 000116' 000000  
 1512 000120' 000000  
 1513 000120' 000000  
 1514

LSEXP1:: .WORD 0  
 LSEXP1:: .WORD 0  
 LSMREV:: .BYTE CSREVISION  
 LSMREV:: .BYTE CSREDIT  
 LSEF:: .WORD 0  
 LSEF:: .WORD 0  
 LSSPC:: .WORD 0  
 LSDEVP:: .WORD LSDVTYP  
 LSREPP:: .WORD LSRPT  
 LSEXP4:: .WORD 0  
 LSEXP5:: .WORD 0  
 LSAUT:: .WORD LSAU  
 LSDUT:: .WORD LSDU  
 LSLUN:: .WORD 0  
 LSDESP:: .WORD LSDESC  
 LSLOAD:: EMT ESLOAD  
 LSETP:: .WORD 0  
 LSICP:: .WORD LSINIT  
 LSCCP:: .WORD LSCLEAN  
 LSACP:: .WORD LSAUTO  
 LSPRT:: .WORD LSPRCT  
 LSTEST:: .WORD 0  
 LSDLY:: .WORD 0  
 LSHIRE:: .WORD 0  
 LSHIRE:: .WORD 0  
 ;RSJ001

75PROGRAM HEADER AND TABLES  
CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 32  
DISPATCH TABLE

.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 46

1515		
1516		
1517		
1518		
1519		
1520		
1521		
1522	000122'	
1523	000122'	000056
1524	000124'	
1525	000124'	022076'
1526	000126'	022310'
1527	000130'	022642'
1528	000132'	022754'
1529	000134'	023072'
1530	000136'	023560'
1531	000140'	023642'
1532	000142'	023740'
1533	000144'	024162'
1534	000146'	027124'
1535	000150'	027540'
1536	000152'	030326'
1537	000154'	031006'
1538	000156'	031424'
1539	000160'	031714'
1540	000162'	032422'
1541	000164'	032740'
1542	000166'	033406'
1543	000170'	033764'
1544	000172'	034344'
1545	000174'	034672'
1546	000176'	035202'
1547	000200'	035512'
1548	000202'	036072'
1549	000204'	036452'
1550	000206'	037046'
1551	000210'	037610'
1552	000212'	040206'
1553	000214'	040610'
1554	000216'	041130'
1555	000220'	041520'
1556	000222'	042104'
1557	000224'	042470'
1558	000226'	043130'
1559	000230'	043460'
1560	000232'	044000'
1561	000234'	044350'
1562	000236'	044670'
1563	000240'	045230'
1564	000242'	045620'
1565	000244'	046202'
1566	000246'	046634'
1567	000250'	047240'
1568	000252'	050330'
1569	000254'	051052'
1570	000256'	051466'
1571		

LSDISPATCH'::	.WORD	46
	.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
	.WORD	T29
	.WORD	T30
	.WORD	T31
	.WORD	T32
	.WORD	T33
	.WORD	T34
	.WORD	T35
	.WORD	T36
	.WORD	T37
	.WORD	T38
	.WORD	T39
	.WORD	T40
	.WORD	T41
	.WORD	T42
	.WORD	T43
	.WORD	T44
	.WORD	T45
	.WORD	T46

75PROGRAM HEADER AND TABLES  
 CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 34  
 DEFAULT HARDWARE P-TABLE

.SBTTL DEFAULT HARDWARE P-TABLE

;++

: THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF  
 : THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE  
 : IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,  
 : AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.

!--

1572  
 1573  
 1574  
 1575  
 1576  
 1577  
 1578  
 1579  
 1580

1581 000260'  
 1582 000260' 000002  
 1583 000262'  
 1584 000262'

BGNHW DFPTBL

.WORD L10000-LSHW/2  
 LSHW::  
 DFPTBL::

1585  
 1586 000262' 000000  
 1587 000264' 000000  
 1588  
 1589 000266'  
 1590 000266'

.WORD 0 :CSR  
 .WORD 0 :VECTOR

ENDHW

L10000:

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 36  
 CZUAB.MAC 07-APR-83 17:03 DEFAULT HARDWARE P-TABLE

```

1592          .TITLE GLOBAL AREAS
1593          .SBTTL  GLOBAL EQUATES SECTION
1594
1595
1596
1597          :++
1598          : THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
1599          : ARE USED IN MORE THAN ONE TEST.
1600          :--
1601
1602 000266'          EQUALS
1603
1604          :
1605          : BIT DIFINITIONS
1606          :
1606          100000      BIT15== 100000
1607          040000      BIT14== 40000
1608          020000      BIT13== 20000
1609          010000      BIT12== 10000
1610          004000      BIT11== 4000
1611          002000      BIT10== 2000
1612          001000      BIT09== 1000
1613          000400      BIT08== 400
1614          000200      BIT07== 200
1615          000100      BIT06== 100
1616          000040      BIT05== 40
1617          000020      BIT04== 20
1618          000010      BIT03== 10
1619          000004      BIT02== 4
1620          000002      BIT01== 2
1621          000001      BIT00== 1
1622          :
1623          001000      BIT9==  BIT09
1624          000400      BIT8==  BIT08
1625          000200      BIT7==  BIT07
1626          000100      BIT6==  BIT06
1627          000040      BIT5==  BIT05
1628          000020      BIT4==  BIT04
1629          000010      BIT3==  BIT03
1630          000004      BIT2==  BIT02
1631          000002      BIT1==  BIT01
1632          000001      BIT0==  BIT00
1633          :
1634          : EVENT FLAG DEFINITIONS
1635          : EF32-EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
1636          :
1637          000040      EF.START==      32.          ; START COMMAND WAS ISSUED
1638          000037      EF.RESTART==    31.          ; RESTART COMMAND WAS ISSUED
1639          000036      EF.CONTINUE==   30.          ; CONTINUE COMMAND WAS ISSUED
1640          000035      EF.NEW==        29.          ; A NEW PASS HAS BEEN STARTED
1641          000034      EF.PWR==        28.          ; A POWER-FAIL/POWER-UP OCCURRED
1642          :
1643          :
1644          : PRIORITY LEVEL DEFINITIONS
1645          :
1646          000340      PRI07== 340
1647          000300      PRI06== 300

```



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 37  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL EQUATES SECTION

```

1648      000240      PRI05== 240
1649      000200      PRI04== 200
1650      000140      PRI03== 140
1651      000100      PRI02== 100
1652      000040      PRI01== 40
1653      000000      PRI00== 0
1654
1655      ;
1656      ;OPERATOR FLAG BITS
1657      ;
1657      000004      EVL==      4
1658      000010      LOT==      10
1659      000020      ADR==      20
1660      000040      IDU==      40
1661      000100      ISR==     100
1662      000200      UAM==     200
1663      000400      BOE==     400
1664      001000      PNT==    1000
1665      002000      PRI==    2000
1666      004000      IXE==    4000
1667      010000      IBE==   10000
1668      020000      IER==   20000
1669      040000      LOE==   40000
1670      100000      NOE==  100000
1671
1672      000077      SECOND==63.          ;63 LINE CLOCK TICKS = APROX. 1 SECOND
1673      000001      SET==      1
1674      000000      CLEAR== 0
1675      004000      SIZ1K== 4000          ;1K WORDS
1676      010000      SIZ2K== SIZ1K*2      ;2K WORDS
1677      020000      SIZ4K== SIZ2K*2      ;4K WORDS
1678      040000      SIZ8K== SIZ4K*2      ;8K WORDS
1679      020000      WCSSIZ==SIZ4K      ;SIZE OF THE DEUMA WRITEABLE CONTROL STORE
1680      020000      IOSIZ==SIZ4K      ;SIZE OF THE DEUMA I/O PAGE
1681      040000      ROMSIZ==SIZ8K      ;SIZE OF THE DEUMA ROM IN WORDS
1682      077774      LINSIZ==SIZ8K*2-4    ;SIZE OF THE DEUMA LINK MEMORY
1683      000000      WCSADR==0          ;INTERNAL BASE ADDRESS OF WCS
1684      020000      IOADR==WCSADR+WCSSIZ  ;INTERNAL BASE ADDRESS OF THE I/O PAGE FOR THE T11
1685      040000      ROMADR==IOADR+IOSIZ  ;INTERNAL BASE ADDRESS OF THE DEUMA ROM
1686      100000      LINADR==ROMADR+ROMSIZ  ;INTERNAL BASE ADDRESS OF THE DEUMA LINK MEMORY
1687      000100      IE==      BIT6          ;INTERRUPT ENABLE
1688      177777      INITH== -1          ;INITIAL HIGH WORD FOR 32 BIT CRC CALCULATOR
1689      177777      INITL== -1          ;INITIAL LOW WORD FOR 32 BIT CRC CALCULATOR
1690      166670      POLYH== 166670      ;FUNCTION POLYNOMIAL HIGH WORD FOR 32 BIT CRC
1691      101440      POLYL== 101440      ;FUNCTION POLYNOMIAL LOW WORD FOR 32 BIT CRC
1692      120001      POLY16== 120001     ;FUNCTION POLYNOMIAL FOR 16 BIT CRC
1693      000000      DATERR==0          ;DATA ERROR INDICATER FOR LINK MEMORY TESTS
1694      000001      ADRERR==1          ;ADDRESS ERROR INDICATOR FOR LINK MEMORY TESTS
1695      002756      MAXBYT==1518.       ;MAXIMUM NUMBER OF BYTES RECEIVER CAN HANDLE
1696      000100      MINBYT==64.         ;MINIMUM NUMBER OF BYTES RECEIVER CAN HANDLE
1697      000004      CRCSIZ==4.         ;SIZE OF CRC
1698      000001      INMON=1          ;IN MICROMONITOR STATE *** REMOVE COMMENT MARKS AND
1699      000002      INTST=2          ;IN A TEST STATE *** USE THESE THREE VARIABLES
1700      000003      INERR=3          ;IN ERROR STATE *** FOR ASSEMBLY WITH MACY11
1701

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 38  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL EQUATES SECTION

```

1702
1703 ; PCSRO - PORT CONTROL AND STATUS REGISTER 0 (OPERATIONAL MICROCODE DEFINITIONS)
1704
1705 100000 SERI == BIT15 ; STATUS ERROR INTERRUPT
1706 000200 SERIB == BIT07 ; STATUS ERROR INTERRUPT BYTE REFERENCE ;RSJ001
1707 040000 PCEI == BIT14 ; PORT COMMAND ERROR INTERRUPT
1708 000100 PCEIB == BIT06 ; PORT COMMAND ERROR INTERRUPT BYTE REFERENCE ;RSJ001
1709 020000 RXI == BIT13 ; RECEIVE RING INTERRUPT
1710 000040 RXIB == BIT05 ; RECEIVE RING INTERRUPT BYTE REFERENCE ;RSJ001
1711 010000 TXI == BIT12 ; TRANSMIT RING INTERRUPT
1712 000020 TXIB == BIT04 ; TRANSMIT RING INTERRUPT BYTE REFERENCE ;RSJ001
1713 004000 DNI == BIT11 ; DONE INTERRUPT
1714 000010 DNIB == BIT03 ; DONE INTERRUPT BYTE REFERENCE ;RSJ001
1715 002000 RCEI == BIT10 ; RECEIVE BUFFER UNAVAILABLE
1716 000004 RCEIB == BIT02 ; RECEIVE BUFFER UNAVAILABLE BYTE REFERENCE ;RSJ001
1717
1718 000400 FATI == BIT08 ; FATAL ERROR INTERRUPT
1719 000001 FATIB == BIT00 ; FATAL ERROR INTERRUPT BYTE REFERENCE ;RSJ001
1720 000200 INTR == BIT07 ; INTERRUPT SUMMARY <15:08>
1721 000100 INTE == BIT06 ; INTERRUPT ENABLE
1722 000040 RSET == BIT05 ; UNA RESET
1723
1724 ;
1725 ; PCSRO - PORT CONTROL AND STATUS REGISTER 0 (DIAGNOSTIC MICROCODE DEFINITIONS)
1726 ;
1727 100000 NPRERR == BIT15 ;T11 NPR TIMEOUT INTERRUPT OCCURRED
1728 040000 NXMERR == BIT14 ;T11 NON-EXISTANT MEMORY TIMEOUT OCCURRED
1729 020000 UNIERR == BIT13 ;T11 UNEXPECTED INTERRUPT OCCURRED
1730 010000 PARERR == BIT12 ;T11 LINK MEMORY PARITY ERROR OCCURRED
1731 ;
1732 ;PORT COMMANDS <03:00>
1733 000001 GETPCB == BIT00
1734 000002 GETCMD == BIT01
1735 000006 PNOP == BIT01:BIT02
1736 000003 SLFT == BIT00:BIT01
1737 ;
1738 ;PCSR1 - PORT CONTROL AND STATUS REGISTER 1
1739 100000 XPWR == BIT15 ; TRANSCIEVER POWER BAD
1740 040000 ICAB == BIT14 ; PORT/LINK CABLING OK
1741 ;
1742 ;SELF TEST ERROR CODE <13:08>
1743 ;
1744 000200 PCTO == BIT07 ; PORT COMMAND TIMEOUT
1745 ;
1746 000010 RRTC == BIT03 ; REMOTE CONSOLE RESERVED
1747 000400 SFTB0 == BIT8 ;FIRST BIT OF SELF TEST FIELD
1748 001000 SFTB1 == BIT9 ;SECOND
1749 002000 SFTB2 == BIT10 ;THIRD
1750 004000 SFTB3 == BIT11 ;FOURTH
1751 010000 SFTB4 == BIT12 ;FIFTH
1752 020000 SFTB5 == BIT13 ;SIXTH
1753 037400 SFT == BIT8:BIT9:BIT10:BIT11:BIT12:BIT13 ;SELF TEST FIELD
1754 ;
1755 ;PORT STATE <02:00>
1756 ;
1757 000007 PSTATE == BIT00:BIT01:BIT02 ;PORT STATE BITS OF PCSR1

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 39  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL EQUATES SECTION

```

1758 000000 RESET == 0 ;RESET STATE (NOT REALLY A STATE)
1759 000001 PRILD == BIT00 ; PRIMARY LOAD STATE
1760 000002 READY == BIT01 ;READY STATE
1761 000003 RUN == B!T00!BIT01 ;RUNNING STATE
1762 000005 UNIHLT == BIT00!BIT02 ;UNIBUS HALTED STATE
1763 000006 NIHLT == BIT01!BIT02 ;NI HALTED STATE
1764 000007 NIUNI == BIT00!BIT01!BIT02 ;UNIBUS AND NI HALTED STATE
1765
1766 ;
1766 ;DESCRIPTOR RING DEFINITIONS
1767 100000 OWN == BIT15
1768 040000 ERRS == BIT14
1769 001000 STP == BIT09
1770 000400 ENP == BIT08
1771
1772 ;
1772 ;PORT FUNCTION CODES
1773 ;
1774
1775 000000 PFNOP1 == 0 ;NOP PORT FUNCTION #1
1776 000001 LASM == 1 ;LOAD AND START MICROADDRESS
1777 000002 RDPA == 2 ;READ DEFAULT PHYSICAL ADDRESS
1778 000003 PFNOP2 == 3 ;NOP PORT FUNCTION #2
1779 000004 RPA == 4 ;READ PHYSICAL ADDRESS
1780 000005 WPA == 5 ;WRITE PHYSICAL ADDRESS
1781 000006 RMAL == 6 ;READ MULTICAST ADDRESS LIST
1782 000007 WMAL == 7 ;WRITE MULTICAST ADDRESS LIST
1783 000010 RRF == 10 ;READ RING FORMAT
1784 000011 WRF == 11 ;WRITE RING FORMAT
1785 000012 RC == 12 ;READ COUNTERS
1786 000013 RACC == 13 ;READ AND CLEAR COUNTERS
1787 000014 RTM == 14 ;READ THE MODE
1788 000015 WTM == 15 ;WRITE THE MODE
1789 000016 RPS == 16 ;READ PORT STATUS
1790 000017 RACPS == 17 ;READ AND CLEAR PORT STATUS
1791 000020 DIM == 20 ;DUMP INTERNAL MEMORY
1792 000021 LIM == 21 ;LOAD INTERNAL MEMORY
1793 000022 RSIDP == 22 ;READ SYSTEM ID PARAMETERS
1794 000023 WSIDP == 23 ;WRITE SYSTEM ID PARAMETERS
1795 000024 RLSA == 24 ;READ LOAD SERVER ADDRESS
1796 000025 WLSA == 25 ;WRITE LOAD SERVER ADDRESS
1797

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 40  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL EQUATES SECTION

```

1798
1799
1800      .SBTTL  GLOBAL DATA SECTION
1801
1802      :++
1803      : THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1804      : IN MORE THAN ONE TEST.
1805      :--
1806
1807 000266' 000000  UNACSR::WORD 0      ;CSR
1808 000270' 000000  UNAVEC::WORD 0      ;VECTOR
1809 000272' 000000  UNAPRI::WORD 0      ;PRIORITY
1810
1811 000274'
1812 000274' 000000  CLKCSR::WORD 0      ;LINE CLOCK STATUS REGISTER
1813 000276' 000000  CLKBR::WORD 0      ;LINE CLOCK PRIORITY
1814 000300' 000000  CLKVEC::WORD 0      ;LINE CLOCK VECTOR
1815 000302' 000000  CLKFRE::WORD 0      ;LINE CLOCK FREQUENCY
1816
1817 000304' 000000  CSRNUM::WORD 0      ;PCSR NUMBER
1818 000306' 000000  BITNUM::WORD 0      ;BIT NUMBER
1819 000310' 000000  BITNAM::WORD 0      ;POINTER TO BIT NAME ASCII STRING
1820 000312' 000000  BITSTA::WORD 0      ;POINTER TO BIT STATUS ASCII STRING
1821 000314' 000000  PWHEN::WORD 0      ;POINTER TO 'BEFORE' OR 'AFTER' ASCII STRING
1822 000316' 000000  PCOMND::WORD 0      ;POINTER TO PORT COMMAND ASCII STRING
1823 000320' 000000  MICMOD::WORD 0      ;POINTER TO MICROCODE MODULE # LOADED
1824
1825 000322' 000000  FRESIZ::WORD 0      ;POINTER TO WORD CONTAINING SIZE OF FREE MEMORY
1826 000324' 000000  FREMEM::WORD 0      ;POINTER TO FREE MEMORY SPACE
1827
1828 000326' 000000  MICRO::WORD 0      ;CURRENT MICROCODE MODULE LOADED
1829 000330' 000000  UNIT::WORD 0      ;CURRENT UNIT NUMBER BEING TESTED
1830 000332' 000000  METER::WORD 0      ;CLOCK TICKS
1831 000334' 021040  SWADDR::WORD 21040 ;INTERNAL ADDRESS OF SWITCH PACK
1832
1833 000336' 000000  PCSRO::WORD 0      ;PORT CONTROL AND STATUS REGISTER 0
1834 000340' 000000  PCSR1::WORD 0      ;PORT CONTROL AND STATUS REGISTER 1
1835 000342' 000000  PCSR2::WORD 0      ;PORT CONTROL AND STATUS REGISTER 2
1836 000344' 000000  PCSR3::WORD 0      ;PORT CONTROL AND STATUS REGISTER 3
1837 000346' 000000  PCSROUB::WORD 0      ;PORT CONTROL AND STATUS REGISTER 0 UPPER BYTE ;RSJ001
1838
1839 000350' 000000  PCSROC::WORD 0      ;PCSR0 CONTENTS
1840 000352' 000000  PCSR1C::WORD 0      ;PCSR1 CONTENTS
1841 000354' 000000  PCSR2C::WORD 0      ;PCSR2 CONTENTS
1842 000356' 000000  PCSR3C::WORD 0      ;PCSR3 CONTENTS
1843
1844 000360'  BNAMTO::      ;TABLE OF POINTERS TO BIT NAME MNEMONICS FOR PCSRO
1845 000360' 001271'      $BIT0
1846 000362' 001263'      $BIT1
1847 000364' 001255'      $BIT2
1848 000366' 001247'      $BIT3
1849 000370' 001241'      $BIT4
1850 000372' 001054'      $RSET
1851 000374' 001043'      $INTE
1852 000376' 001032'      $INTR
1853 000400' 001021'      $FATI

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 41  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL DATA SECTION

1854 000402' 001203'  
 1855 000404' 001010'  
 1856 000406' 001000'  
 1857 000410' 000770'  
 1858 000412' 000760'  
 1859 000414' 000747'  
 1860 000416' 000736'  
 1861 000420'  
 1862 000420' 000000  
 1863 000422' 000000  
 1864 000424' 000000  
 1865 000426' 001120'  
 1866 000430' 001241'  
 1867 000432' 001233'  
 1868 000434' 001225'  
 1869 000436' 001107'  
 1870 000440' 000000  
 1871 000442' 000000  
 1872 000444' 000000  
 1873 000446' 000000  
 1874 000450' 000000  
 1875 000452' 000000  
 1876 000454' 001076'  
 1877 000456' 001065'  
 1878  
 1879 000460'  
 1880 000460' 001271'  
 1881 000462' 001263'  
 1882 000464' 001255'  
 1883 000466' 001247'  
 1884 000470' 001241'  
 1885 000472' 001233'  
 1886 000474' 001225'  
 1887 000476' 001217'  
 1888 000500' 001211'  
 1889 000502' 001203'  
 1890 000504' 001174'  
 1891 000506' 001165'  
 1892 000510' 001156'  
 1893 000512' 001147'  
 1894 000514' 001140'  
 1895 000516' 001131'  
 1896  
 1897 000520'  
 1898 000520' 125252  
 1899 000522' 052525  
 1900 000524' 031463  
 1901 000526' 007417  
 1902 000530' 000377  
 1903 000532' 177777  
 1904  
 1905 000534' 125252  
 1906 000536' 125252  
 1907 000540' 125252  
 1908 000542' 052525  
 1909 000544' 052525

\$BIT9  
 \$RCEI  
 \$DNI  
 \$TXI  
 \$RXI  
 \$PCEI  
 \$SERI

BNAMT1::

.WORD 0  
 .WORD 0  
 .WORD 0  
 \$AMTC  
 \$BIT4  
 \$BIT5  
 \$BIT6  
 \$PCTO  
 .WORD 0  
 .WORD 0  
 .WORD 0  
 .WORD 0  
 .WORD 0  
 .WORD 0  
 \$ICAB  
 \$XPWR

;TABLE OF POINTERS TO BIT NAME MNEMONICS FOR PCSR1

BNAMT2::

\$BIT0  
 \$BIT1  
 \$BIT2  
 \$BIT3  
 \$BIT4  
 \$BIT5  
 \$BIT6  
 \$BIT7  
 \$BIT8  
 \$BIT9  
 \$BIT10  
 \$BIT11  
 \$BIT12  
 \$BIT13  
 \$BIT14  
 \$BIT15

;TABLE OF GENERIC BIT NAME MNEMONICS

PATERN::

PAT1:: .WORD ^B1010101010101010  
 PAT2:: .WORD ^B0101010101010101  
 PAT3:: .WORD ^B0011001100110011  
 PAT4:: .WORD ^B0000111100001111  
 PAT5:: .WORD ^B0000000011111111  
 PAT6:: .WORD ^B1111111111111111

SPAT1::

.WORD ^B1010101010101010  
 .WORD ^B1010101010101010  
 .WORD ^B1010101010101010  
 SPAT2:: .WORD ^B0101010101010101  
 .WORD ^B0101010101010101

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 42  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL DATA SECTION

1910	000546'	052525		.WORD	^B0101010101010101
1911	000550'	031463	SPAT3::	.WORD	^B0011001100110011
1912	000552'	031463		.WORD	^B0011001100110011
1913	000554'	031463		.WORD	^B0011001100110011
1914	000556'	007417	SPAT4::	.WORD	^B0000111100001111
1915	000560'	007417		.WORD	^B0000111100001111
1916	000562'	007417		.WORD	^B0000111100001111
1917	000564'	000377	SPAT5::	.WORD	^B0000000011111111
1918	000566'	000377		.WORD	^B0000000011111111
1919	000570'	000377		.WORD	^B0000000011111111
1920	000572'	177777	SPAT6::	.WORD	^B1111111111111111
1921	000574'	000000		.WORD	^B0000000000000000
1922	000576'	177777		.WORD	^B1111111111111111
1923	000600'	177777	SPAT7::	.WORD	^B1111111111111111
1924	000602'	000377		.WORD	^B0000000011111111
1925	000604'	000000		.WORD	^B0000000000000000
1926					
1927	000606'	000000	PCBB::	.WORD	0 ;PORT CONTROL BLOCK
1928	000610'	000000		.WORD	0 ;+2
1929	000612'	000000		.WORD	0 ;+4
1930	000614'	000000		.WORD	0 ;+6
1931					
1932	000616'	000000	UDBB::	.WORD	0 ;UNIBUS DATA BLOCK BASE
1933	000620'	000000		.WORD	0 ;+2
1934	000622'	000000		.WORD	0 ;+4
1935	000624'	000000		.WORD	0 ;+6
1936					
1937	000626'	000020	CNTTAB::	.BLKW	16. ;HOLDS THE 16 RETRY WAIT COUNTS FOR TEST 39
1938					
1939	000666'	000000	NEXMEM::	.WORD	0 ;NON-EXISTANT MEMORY FLAG
1940	000670'	000000	ERRINT::	.WORD	0 ;HOLDS PCSRO INTERRUPT FLAG BITS FROM MICROCODE
1941	000672'	000000	UNAIINT::	.WORD	0 ;UNA INTERRUPT FLAG
1942	000674'	000000	FRSTIM::	.WORD	0 ;FLAG INDICATING FIRST TIME THROUGH TESTS FOR ALL UNITS
1943	000676'	000000	CPUPRI::	.WORD	0 ;CPU PRIORITY BEFORE INTERRUPT OCCURRED
1944					
1945					.EVEN

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 44  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL DATA SECTION

```

1946
1947
1948
1949
1950          .SBTTL  GLOBAL TEXT SECTION
1951
1952          :++
1953          : THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
1954          : MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
1955          : MORE THAN ONE TEST.
1956          :--
1957
1958          :
1959          : NAMES OF DEVICES SUPPORTED BY PROGRAM
1960          :
1961          :         DEVTYP  <DEUNA>
1962
1963          :
1964          :
1965          :
1966          :
1967          : TEST DESCRIPTION
1968          :
1969          :         DESCRIPT          <DEUNA REPAIR DIAGNOSTIC>
1970
1971          :
1972          :
1973          :
1974          :
1975          :
1976          :
1977          :         SSERI::  .ASCIZ  /SERI BIT/
1978          :
1979          :         SPCEI::  .ASCIZ  /PCEI BIT/
1980          :
1981          :         SRXI::  .ASCIZ  /RXI BIT/
1982          :
1983          :         $IXI::  .ASCIZ  /IXI BIT/
1984          :
1985          :         $DNI::  .ASCIZ  /DNI BIT/
1986          :
1987          :         $RCEI::  .ASCIZ  /RCEI BIT/
1988          :
1989          :         $FATI::  .ASCIZ  /FATI BIT/
1990          :
1991          :         $INTR::  .ASCIZ  /INTR BIT/
1992          :
1993          :         $INTE::  .ASCIZ  /INTE BIT/
1994          :
1995          :         $RSET::  .ASCIZ  /RSET BIT/
1996          :
1997          :         $XPWR::  .ASCIZ  /XPWR BIT/
1998          :
1999          :         $ICAB::  .ASCIZ  /ICAB BIT/
2000          :
2001          :         $PCTO::  .ASCIZ  /PCTO BIT/

```

```

LSDVTYP::
          .ASCIZ  /DEUNA/
          .EVEN

```

```

LSDESC::
          .ASCIZ  /DEUNA REPAIR DI
          .EVEN

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 45  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2002	001114'	044502	000124				
2003	001120'	046522	041524	041040	\$RMTC::	.ASCIZ	/RMT BIT/
2004	001126'	052111	000				
2005	001131'	102	052111	030440	\$BIT15::	.ASCIZ	/BIT 15/
2006	001136'	000065					
2007	001140'	044502	020124	032061	\$BIT14::	.ASCIZ	/BIT 14/
2008	001146'	000					
2009	001147'	102	052111	030440	\$BIT13::	.ASCIZ	/BIT 13/
2010	001154'	000063					
2011	001156'	044502	020124	031061	\$BIT12::	.ASCIZ	/BIT 12/
2012	001164'	000					
2013	001165'	102	052111	030440	\$BIT11::	.ASCIZ	/BIT 11/
2014	001172'	000061					
2015	001174'	044502	020124	030061	\$BIT10::	.ASCIZ	/BIT 10/
2016	001202'	000					
2017	001203'	102	052111	034440	\$BIT9::	.ASCIZ	/BIT 9/
2018	001210'	000					
2019	001211'	102	052111	034040	\$BIT8::	.ASCIZ	/BIT 8/
2020	001216'	000					
2021	001217'	102	052111	033440	\$BIT7::	.ASCIZ	/BIT 7/
2022	001224'	000					
2023	001225'	102	052111	033040	\$BIT6::	.ASCIZ	/BIT 6/
2024	001232'	000					
2025	001233'	102	052111	032440	\$BIT5::	.ASCIZ	/BIT 5/
2026	001240'	000					
2027	001241'	102	052111	032040	\$BIT4::	.ASCIZ	/BIT 4/
2028	001246'	000					
2029	001247'	102	052111	031440	\$BIT3::	.ASCIZ	/BIT 3/
2030	001254'	000					
2031	001255'	102	052111	031040	\$BIT2::	.ASCIZ	/BIT 2/
2032	001262'	000					
2033	001263'	102	052111	030440	\$BIT1::	.ASCIZ	/BIT 1/
2034	001270'	000					
2035	001271'	102	052111	030040	\$BIT0::	.ASCIZ	/BIT 0/
2036	001276'	000					
2037	001277'	116	052117	051440	\$MSET::	.ASCIZ	/NOT SET/
2038	001304'	052105	000				
2039	001307'	123	052105	000	\$SET::	.ASCIZ	/SET/
2040	001313'	116	052117	041440	\$NCLR::	.ASCIZ	/NOT CLEAR/
2041	001320'	042514	051101	000			
2042	001325'	103	042514	051101	\$CLR::	.ASCIZ	/CLEAR/
2043	001332'	000					
2044	001333'	102	043105	051117	\$BEFOR::	.ASCIZ	/BEFORE/
2045	001340'	000105					
2046	001342'	043101	042524	000122	\$AFTER::	.ASCIZ	/AFTER/
2047	001350'	042507	050124	041103	\$GTPCB::	.ASCIZ	/GETPCB/
2048	001356'	000					
2049	001357'	107	052105	041440	\$GTCMD::	.ASCIZ	/GET COMMAND/
2050	001364'	046517	040515	042116			
2051	001372'	000					
2052	001373'	123	046105	020106	\$SLFT::	.ASCIZ	/SELF TEST/
2053	001400'	042524	052123	000			
2054	001405'	116	050117	000	\$NOP::	.ASCIZ	/NOP/
2055	001411'	123	040524	052122	\$SRT::	.ASCIZ	/START/
2056	001416'	000					
2057	001417'	104	046505	047101	\$PDNDM::	.ASCIZ	/DEMAND POLL/



GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 46  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2058	001424'	020104	047520	046114	
2059	001432'	000			
2060	001433'	123	047524	000120	\$STOP::.ASCIZ /STOP/
2061	001440'	042522	042523	000124	\$RESET::.ASCIZ /RESET/
2062	001446'	040504	040524	042440	\$DATER::.ASCIZ /DATA ERROR/
2063	001454'	051122	051117	000	
2064	001461'	101	042104	042522	\$ADRER::.ASCIZ /ADDRESSING ERROR/
2065	001466'	051523	047111	020107	
2066	001474'	051105	047522	000122	
2067	001502'	040520	044522	054524	\$PARER::.ASCIZ /PARITY ERROR/
2068	001510'	042440	051122	051117	
2069	001516'	000			
2070					
2071					:
2072					: FORMAT STATEMENTS USED IN PRINT CALLS
2073					:
2074					:
2075	001517'	045	022516	052501	UNLOD::.ASCIZ /UNLOADABLE TO LOAD MICROCODE MODULE XTZN/
2076	001524'	040516	046102	020105	
2077	001532'	047524	046040	040517	
2078	001540'	020104	044515	051103	
2079	001546'	041517	042117	020105	
2080	001554'	047515	052504	042514	
2081	001562'	022440	022524	000116	
2082	001570'	042522	044507	052123	RACERR::.ASCIZ /REGISTER ACCESS TEST FAILED/
2083	001576'	051105	040440	041503	
2084	001604'	051505	020123	042524	
2085	001612'	052123	043040	044501	
2086	001620'	042514	000104		
2087	001624'	042522	042523	020124	RSETER::.ASCIZ /RESET TEST FAILED/
2088	001632'	042524	052123	043040	
2089	001640'	044501	042514	000104	
2090	001646'	042522	044507	052123	RRWER::.ASCIZ 'REGISTER READ/WRITE TEST FAILED'
2091	001654'	051105	051040	040505	
2092	001662'	027504	051127	052111	
2093	001670'	020105	042524	052123	
2094	001676'	043040	044501	042514	
2095	001704'	000104			
2096	001706'	047516	020120	042524	NOPERR::.ASCIZ /NOP TEST FAILED/
2097	001714'	052123	043040	044501	
2098	001722'	042514	000104		
2099	001726'	042523	043114	052040	SLFTST::.ASCIZ /SELF TEST FAILED/
2100	001734'	051505	020124	040506	
2101	001742'	046111	042105	000	
2102	001747'	122	046517	042040	ROMDMP::.ASCIZ /ROM DUMP TEST FAILED/
2103	001754'	046525	020120	042524	
2104	001762'	052123	043040	044501	
2105	001770'	042514	000104		
2106	001774'	041527	020123	047514	DATALD::.ASCIZ 'WCS LOAD/DUMP TEST FAILED'
2107	002002'	042101	042057	046525	
2108	002010'	020120	042524	052123	
2109	002016'	043040	044501	042514	
2110	002024'	000104			
2111	002026'	047514	042101	040440	LASFT::.ASCIZ /LOAD AND START FUNCTION TEST FAILED/
2112	002034'	042116	051440	040524	
2113	002042'	052122	043040	047125	

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 47  
 CZUAA3.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2114	002050'	052103	047511	020116	
2115	002056'	042524	052123	043040	
2116	002064'	044501	042514	000104	
2117	002072'	041527	020123	042515	WCSMEM::.ASCIZ /MCS MEMORY TEST FAILED/
2118	002100'	047515	054522	052040	
2119	002106'	051505	020124	040506	
2120	002114'	046111	042105	000	
2121	002121'	111	052116	051105	INTVEC::.ASCIZ /INTERRUPT VECTOR TEST FAILED/
2122	002126'	052522	052120	053040	
2123	002134'	041505	047524	020122	
2124	002142'	042524	052123	043040	
2125	002150'	044501	042514	000104	
2126	002156'	041520	051123	020060	INTBIT::.ASCIZ /PCSR0 INTERRUPT BIT TEST FAILED/
2127	002164'	047111	042524	051122	
2128	002172'	050125	020124	044502	
2129	002200'	020124	042524	052123	
2130	002206'	043040	044501	042514	
2131	002214'	000104			
2132	002216'	044524	042515	020122	TIMTST::.ASCIZ /TIMER TEST FAILED/
2133	002224'	042524	052123	043040	
2134	002232'	044501	042514	000104	
2135	002240'	044514	045516	046440	LNKMEM::.ASCIZ /LINK MEMORY TEST FAILED/
2136	002246'	046505	051117	020131	
2137	002254'	042524	052123	043040	
2138	002262'	044501	042514	000104	
2139	002270'	046504	020101	052047	DMATO::.ASCIZ /DMA 'TO' ADDRESS REGISTER TEST FAILED/
2140	002276'	023517	040440	042104	
2141	002304'	042522	051523	051040	
2142	002312'	043505	051511	042524	
2143	002320'	020122	042524	052123	
2144	002326'	043040	044501	042514	
2145	002334'	000104			
2146	002336'	046504	020101	043047	DMAFRM::.ASCIZ /DMA 'FROM' ADDRESS REGISTER TEST FAILED/
2147	002344'	047522	023515	040440	
2148	002352'	042104	042522	051523	
2149	002360'	051040	043505	051511	
2150	002366'	042524	020122	042524	
2151	002374'	052123	043040	044501	
2152	002402'	042514	000104		
2153	002406'	046504	020101	046102	DMABLK::.ASCIZ /DMA BLOCK TRANSFER TEST FAILED/
2154	002414'	041517	020113	051124	
2155	002422'	047101	043123	051105	
2156	002430'	052040	051505	020124	
2157	002436'	040506	046111	042105	
2158	002444'	000			
2159	002445'	124	040522	051516	TRNDON::.ASCIZ /TRANSMIT DONE TEST FAILED/
2160	002452'	044515	020124	047504	
2161	002460'	042516	052040	051505	
2162	002466'	020124	040506	046111	
2163	002474'	042105	000		
2164	002477'	122	041505	044505	RCVDON::.ASCIZ /RECEIVER DONE TEST FAILED/
2165	002504'	042526	020122	047504	
2166	002512'	042516	052040	051505	
2167	002520'	020124	040506	046111	
2168	002526'	042105	000		
2169	002531'	104	052101	020101	DBFRAM::.ASCIZ /DATA BYTE FRAMING TEST FAILED/

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 48  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2170	002536	054502	042524	043040	
2171	002544	040522	044515	043516	
2172	002552	052040	051505	020124	
2173	002560	040506	046111	042105	
2174	002566	000			
2175	002567	104	052101	020101	DWFRAM::.ASCIZ /DATA WORD FRAMING TEST FAILED/
2176	002574	047527	042122	043040	
2177	002602	040522	044515	043516	
2178	002610	052040	051505	020124	
2179	002616	040506	046111	042105	
2180	002624	000			
2181	002625	104	052101	020101	DPPAT::.ASCIZ /DATA PATH PATTERN TEST FAILED/
2182	002632	040520	044124	050040	
2183	002640	052101	042524	047122	
2184	002646	052040	051505	020124	
2185	002654	040506	046111	042105	
2186	002662	000			
2187	002663	123	040524	052524	STAMUX::.ASCIZ /STATUS MUX VERIFICATION TEST FAILED/
2188	002670	020123	052515	020130	
2189	002676	042524	044522	044506	
2190	002704	040503	044524	047117	
2191	002712	052040	051505	020124	
2192	002720	040506	046111	042105	
2193	002726	000			
2194	002727	114	047111	020113	LNKBYT::.ASCIZ /LINK BYTE COUNTER TEST FAILED/
2195	002734	054502	042524	041440	
2196	002742	052517	052116	051105	
2197	002750	052040	051505	020124	
2198	002756	040506	046111	042105	
2199	002764	000			
2200	002765	114	047111	020113	ODDBYT::.ASCIZ /LINK ODD BYTE COUNTER TEST FAILED/
2201	002772	042117	020104	054502	
2202	003000	042524	041440	052517	
2203	003006	052116	051105	052040	
2204	003014	051505	020124	040506	
2205	003022	046111	042105	000	
2206	003027	114	047111	020113	MAXCNT::.ASCIZ /LINK MAXIMUM BYTE COUNT TEST FAILED/
2207	003034	040515	044530	052515	
2208	003042	020115	054502	042524	
2209	003050	041440	052517	052116	
2210	003056	052040	051505	020124	
2211	003064	040506	046111	042105	
2212	003072	000			
2213	003073	106	043111	020117	FIFTST::.ASCIZ /FIFO TEST FAILED/
2214	003100	042524	052123	043040	
2215	003106	044501	042514	000104	
2216	003114	042522	042503	053111	RLNKAD::.ASCIZ /RECEIVER LINK MEMORY ADDRESS TEST FAILED/
2217	003122	051105	046040	047111	
2218	003130	020113	042515	047515	
2219	003136	054522	040440	042104	
2220	003144	042522	051523	052040	
2221	003152	051505	020124	040506	
2222	003160	046111	042105	000	
2223	003165	124	040522	051516	TLNKAD::.ASCIZ /TRANSMITTER LINK MEMORY ADDRESS TEST FAILED/
2224	003172	044515	052124	051105	
2225	003200	046040	047111	020113	

GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 49  
 CZUAB.MAL 07-APR-83 17:03 GLOBAL TEXT SECTION

2226	003206'	042515	047515	054522	
2227	003214'	040440	042104	042522	
2228	003222'	051523	052040	051505	
2229	003230'	020124	040506	046111	
2230	003236'	042105	000		
2231	003241'	114	047111	020113	LNKARB::ASCIZ /LINK MEMORY ARBITRATION TEST FAILED/
2232	003246'	042515	047515	054522	
2233	003254'	040440	041122	052111	
2234	003262'	040522	044524	047117	
2235	003270'	052040	051505	020124	
2236	003276'	040506	046111	042105	
2237	003304'	000			
2238	003305'	123	040524	044524	STAPAT::ASCIZ /STATION ADDRESS PATTERN TEST FAILED/
2239	003312'	047117	040440	042104	
2240	003320'	042522	051523	050040	
2241	003326'	052101	042524	047122	
2242	003334'	052040	051505	020124	
2243	003342'	040506	046111	042105	
2244	003350'	000			
2245	003351'	123	040524	044524	STAREJ::ASCIZ /STATION ADDRESS REJECTION TEST FAILED/
2246	003356'	047117	040440	042104	
2247	003364'	042522	051523	051040	
2248	003372'	045105	041505	044524	
2249	003400'	047117	052040	051505	
2250	003406'	020124	040506	046111	
2251	003414'	042105	000		
2252	003417'	123	040524	044524	STAPOS::ASCIZ /STATION ADDRESS RAM POSITION TEST FAILED/
2253	003424'	047117	040440	042104	
2254	003432'	042522	051523	051040	
2255	003440'	046501	050040	051517	
2256	003446'	052111	047511	020116	
2257	003454'	042524	052123	043040	
2258	003462'	044501	042514	000104	
2259	003470'	052515	052114	041511	MUCAST::ASCIZ /MULTICAST ADDRESS TEST FAILED/
2260	003476'	051501	020124	042101	
2261	003504'	051104	051505	020123	
2262	003512'	042524	052123	043040	
2263	003520'	044501	042514	000104	
2264	003526'	051103	020103	040504	CRCDAT::ASCIZ /CRC DATA PATTERN TEST FAILED/
2265	003534'	040524	050040	052101	
2266	003542'	042524	047122	052040	
2267	003550'	051505	020124	040506	
2268	003556'	046111	042105	000	
2269	003563'	103	041522	042440	CRCERR::ASCIZ /CRC ERROR TEST FAILED/
2270	003570'	051122	051117	052040	
2271	003576'	051505	020124	040506	
2272	003604'	046111	042105	000	
2273	003611'	103	041522	050040	CRCPAT::ASCIZ /CRC PATTERN LENGTH TEST FAILED/
2274	003616'	052101	042524	047122	
2275	003624'	046040	047105	052107	
2276	003632'	020110	042524	052123	
2277	003640'	043040	044501	042514	
2278	003646'	000104			
2279	003650'	042522	042503	053111	RBRUN::ASCIZ /RECEIVE BUFFER RECOVERY - RUNT TEST FAILED/
2280	003656'	020105	052502	043106	
2281	003664'	051105	051040	041505	

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 50  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2282	003672'	053117	051105	020131	
2283	003700'	020055	052522	052116	
2284	003706'	052040	051505	020124	
2285	003714'	040506	046111	042105	
2286	003722'	000			
2287	003723'	110	046101	026506	HAFDUP::ASCIZ /HALF-DUPLEX TEST FAILED/
2288	003730'	052504	046120	054105	
2289	003736'	052040	051505	020124	
2290	003744'	040506	046111	042105	
2291	003752'	000			
2292	003753'	103	046117	044514	COLTST::ASCIZ /COLLISION TEST FAILED/
2293	003760'	044523	047117	052040	
2294	003766'	051505	020124	040506	
2295	003774'	046111	042105	000	
2296	004001'	124	051104	041440	TDRCNT::ASCIZ /TDR COUNTER TEST FAILED/
2297	004006'	052517	052116	051105	
2298	004014'	052040	051505	020124	
2299	004022'	040506	046111	042105	
2300	004030'	000			
2301	004031'	122	052105	054522	RETLOG::ASCIZ /RETRY LOGIC TEST FAILED/
2302	004036'	046040	043517	041511	
2303	004044'	052040	051505	020124	
2304	004052'	040506	046111	042105	
2305	004060'	000			
2306	004061'	125	040516	046102	PRTPAR::ASCIZ /UNABLE TO PRINT DEVICE PARAMETERS FOR THIS UNIT/
2307	004066'	020105	047524	050040	
2308	004074'	044522	052116	042040	
2309	004102'	053105	041511	020105	
2310	004110'	040520	040522	042515	
2311	004116'	042524	051522	043040	
2312	004124'	051117	052040	044510	
2313	004132'	020123	047125	052111	
2314	004140'	000			
2315	004141'	045	050101	051503	FORM1::ASCIZ /XAPCSRXD1XA DOES NOT EXISTXN/
2316	004146'	022522	030504	040445	
2317	004154'	042040	042517	020123	
2318	004162'	047516	020124	054105	
2319	004170'	051511	022524	000116	
2320	004176'	040445	041520	051123	FORM2::ASCIZ /XAPCSRXD1XA BIT XZ2XA IS XTZN/
2321	004204'	042045	022461	020101	
2322	004212'	044502	020124	055045	
2323	004220'	022462	020101	051511	
2324	004226'	022440	022524	000116	
2325	004234'	040445	041520	051123	FORM3::ASCIZ /XAPCSRXD1XA FAILED DATA PATTERN TESTXN/
2326	004242'	042045	022461	020101	
2327	004250'	040506	046111	042105	
2328	004256'	042040	052101	020101	
2329	004264'	040520	052124	051105	
2330	004272'	020116	042524	052123	
2331	004300'	047045	000		
2332	004303'	045	042101	052101	FORM4::ASCIZ /XADATA WRITTEN: X06XA DATA READ: X06XN/
2333	004310'	020101	051127	052111	
2334	004316'	042524	035116	022440	
2335	004324'	033117	040445	042040	
2336	004332'	052101	020101	042522	
2337	004340'	042101	020072	047445	



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 52  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2394	005044'	042522	051523	020072	
2395	005052'	047445	022466	030523	
2396	005060'	040445	040504	040524	
2397	005066'	051440	035102	022440	
2398	005074'	033117	051445	022461	
2399	005102'	042101	052101	020101	
2400	005110'	040527	035123	022440	
2401	005116'	033117	047045	000	
2402	005123'	045	030501	020066	FORM13::ASCIZ /XA16 BIT CRC SHOULD BE: 0XS2XAWAS: X06XN/
2403	005130'	044502	020124	051103	
2404	005136'	020103	044123	052517	
2405	005144'	042114	041040	035105	
2406	005152'	030040	051445	022462	
2407	005160'	053501	051501	020072	
2408	005166'	047445	022466	000116	
2409	005174'	040445	040504	040524	FORM15::ASCIZ /XADATA WRITTEN TO PCSR1 FROM T11 WAS: X06XN/
2410	005202'	053440	044522	052124	
2411	005210'	047105	052040	020117	
2412	005216'	041520	051123	020061	
2413	005224'	051106	046517	052040	
2414	005232'	030461	053440	051501	
2415	005240'	020072	047445	022466	
2416	005246'	000116			
2417	005250'	040445	040504	040524	FORM16::ASCIZ /XADATA READ FROM PCSR1 FROM UNIBUS WAS: X06XN/
2418	005256'	051040	040505	020104	
2419	005264'	051106	046517	050040	
2420	005272'	051503	030522	043040	
2421	005300'	047522	020115	047125	
2422	005306'	041111	051525	053440	
2423	005314'	051501	020072	047445	
2424	005322'	022466	000116		
2425	005326'	040445	047516	044440	FORM17::ASCIZ /XAND INTERRUPT AFTER NOP PORT COMMAND WAS ISSUEDXN/
2426	005334'	052116	051105	052522	
2427	005342'	052120	040440	052106	
2428	005350'	051105	047040	050117	
2429	005356'	050040	051117	020124	
2430	005364'	047503	046515	047101	
2431	005372'	020104	040527	020123	
2432	005400'	051511	052523	042105	
2433	005406'	047045	000		
2434	005411'	045	052501	040516	FORM18::ASCIZ /XAUNA DID NOT INTERRUPT AT CORRECT PRIORITYXN/
2435	005416'	042040	042111	047040	
2436	005424'	052117	044440	052116	
2437	005432'	051105	052522	052120	
2438	005440'	040440	020124	047503	
2439	005446'	051122	041505	020124	
2440	005454'	051120	047511	044522	
2441	005462'	054524	047045	000	
2442	005467'	045	047101	020117	FORM19::ASCIZ /XAND INTERRUPT AFTER T11 SET XTXA IN PCSROXN/
2443	005474'	047111	042524	051122	
2444	005502'	050125	020124	043101	
2445	005510'	042524	020122	030524	
2446	005516'	020061	042523	020124	
2447	005524'	052045	040445	044440	
2448	005532'	020116	041520	051123	
2449	005540'	022460	000116		

GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 53  
 CZUAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2450	005544'	040445	044524	042515	FORM20:::ASCIZ	/XATIMER DID NOT INTERRUPT T11XN/
2451	005552'	020122	044504	020104		
2452	005560'	047516	020124	047111		
2453	005566'	042524	051122	050125		
2454	005574'	020124	030524	022461		
2455	005602'	000116				
2456	005604'	040445	044524	042515	FORM21:::ASCIZ	/XATIMER DID NOT INTERRUPT WHEN EXPECTEDXN/
2457	005612'	020122	044504	020104		
2458	005620'	047516	020124	047111		
2459	005626'	042524	051122	050125		
2460	005634'	020124	044127	047105		
2461	005642'	042440	050130	041505		
2462	005650'	042524	022504	000116		
2463	005656'	040445	054105	042520	FORM22:::ASCIZ	/XAEEXPECTED INTERRUPT BETWEEN 8 AND 12 SECONDSXN/
2464	005664'	052103	052105	044440		
2465	005672'	052116	051105	052522		
2466	005700'	052120	041040	052105		
2467	005706'	042527	047105	034040		
2468	005714'	040440	042116	030440		
2469	005722'	020062	042523	047503		
2470	005730'	042116	022523	000116		
2471	005736'	040445	042522	044503	FORM23:::ASCIZ	/XARECIEVED INTERRUPT AT XD1XA SECONDSXN/
2472	005744'	053105	042105	044440		
2473	005752'	052116	051105	052522		
2474	005760'	052120	040440	020124		
2475	005766'	042045	022461	020101		
2476	005774'	042523	047503	042116		
2477	006002'	022523	000116			
2478	006006'	040445	044515	051103	FORM24:::ASCIZ	/XAMICRO TEST XD2XA HUNGXN/
2479	006014'	020117	042524	052123		
2480	006022'	022440	031104	040445		
2481	006030'	044040	047125	022507		
2482	006036'	000116				
2483	006040'	040445	040504	040524	FORM25:::ASCIZ	/XADATA WRITTEN TO 'DMATO' REGISTER = X06XN/
2484	006046'	053440	044522	052124		
2485	006054'	047105	052040	020117		
2486	006062'	042047	040515	047524		
2487	006070'	020047	042522	044507		
2488	006076'	052123	051105	036440		
2489	006104'	022440	033117	047045		
2490	006112'	000				
2491	006113'	045	042101	052101	FORM26:::ASCIZ	/XADATA READ FROM 'DMATO' REGISTER = X06XN/
2492	006120'	020101	042522	042101		
2493	006126'	043040	047522	020115		
2494	006134'	042047	040515	047524		
2495	006142'	020047	042522	044507		
2496	006150'	052123	051105	020040		
2497	006156'	020075	047445	022466		
2498	006164'	000116				
2499	006166'	040445	040504	040524	FORM27:::ASCIZ	/XADATA SHOULD BE: X06XA DATA WAS: X06XN/
2500	006174'	051440	047510	046125		
2501	006202'	020104	042502	020072		
2502	006210'	047445	022466	020101		
2503	006216'	040504	040524	053440		
2504	006224'	051501	020072	047445		
2505	006232'	022466	000116			



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 54  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2506	006236'	052045	047045	000	FORM28:::ASCIZ	/XTXN/
2507	006243'	045	040501	042104	FORM29:::ASCIZ	/XAADDRESS = X06XS2/
2508	006250'	042522	051523	036440		
2509	006256'	022440	033117	051445		
2510	006264'	000062				
2511	006266'	040445	042522	042503	FORM30:::ASCIZ	/XARECEIVER STATUS:X06XN/
2512	006274'	053111	051105	051440		
2513	006302'	040524	052524	035123		
2514	006310'	047445	022466	000116		
2515	006316'	040445	052502	043106	FORM31:::ASCIZ	/XABUFFER OFFSET:X06XN/
2516	006324'	051105	047440	043106		
2517	006332'	042523	035124	047445		
2518	006340'	022466	000116			
2519	006344'	040445	051124	047101	FORM32:::ASCIZ	/XATRANSMIT STATUS WORD XD1XS1XTXA NOT CLEAR AFTER TRANSMITXN/
2520	006352'	046523	052111	051440		
2521	006360'	040524	052524	020123		
2522	006366'	047527	042122	022440		
2523	006374'	030504	051445	022461		
2524	006402'	022524	020101	047516		
2525	006410'	020124	046103	040505		
2526	006416'	020122	043101	042524		
2527	006424'	020122	051124	047101		
2528	006432'	046523	052111	047045		
2529	006440'	000				
2530	006441'	045	052101	040522	FORM33:::ASCIZ	/XATRANSMIT BYTE COUNT = XD4XN/
2531	006446'	051516	044515	020124		
2532	006454'	054502	042524	041440		
2533	006462'	052517	052116	036440		
2534	006470'	022440	032104	047045		
2535	006476'	000				
2536	006477'	045	051101	041505	FORM34:::ASCIZ	/XARECEIVE BYTE COUNT SHOULD BE: XD4XS3XAWAS: XD4XN/
2537	006504'	044505	042526	041040		
2538	006512'	052131	020105	047503		
2539	006520'	047125	020124	044123		
2540	006526'	052517	042114	041040		
2541	006534'	035105	022440	032104		
2542	006542'	051445	022463	053501		
2543	006550'	051501	020072	042045		
2544	006556'	022464	000116			
2545	006562'	040445	0415C1	052524	FORM35:::ASCIZ	/XA ACTUAL NUMBER OF BYTES RECEIVED SHOULD BE: XD4XA WAS: XD4XN/
2546	006570'	046101	047040	046525		
2547	006576'	042502	020122	043117		
2548	006604'	041040	052131	051505		
2549	006612'	051040	041505	044505		
2550	006620'	042526	020104	044123		
2551	006626'	052517	042114	041040		
2552	006634'	035105	022440	032104		
2553	006642'	040445	020040	053440		
2554	006650'	051501	020072	042045		
2555	006656'	022464	000116			
2556	006662'	040445	044514	045516	FORM36:::ASCIZ	/XALINK MEMORY ADDRESS = X06XN/
2557	006670'	046440	046505	051117		
2558	006676'	020131	042101	051104		
2559	006704'	051505	020123	020075		
2560	006712'	047445	022466	000116		
2561	006720'	040445	051124	047101	FORM37:::ASCII	/XATRANSMIT STATUS INFORMATION INCORRECT AFTER LOOPBACK STEP /

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 55  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2562	006726'	046523	052111	051440	
2563	006734'	040524	052524	020123	
2564	006742'	047111	047506	046522	
2565	006750'	052101	047511	020116	
2566	006756'	047111	047503	051122	
2567	006764'	041505	020124	043101	
2568	006772'	042524	020122	047514	
2569	007000'	050117	040502	045503	
2570	007006'	051440	042524	020120	
2571	007014'	052516	041115	051105	
2572	007022'	022440	031104	047045	.ASCIZ /NUMBER %D2%N/
2573	007030'	000			
2574	007031'	045	052101	040522	FORM38::ASCIZ /%ATRANSMIT STATUS WORD %D1% SHOULD BE: %06% WAS: %06%N/
2575	007036'	051516	044515	020124	
2576	007044'	052123	052101	051525	
2577	007052'	053440	051117	020104	
2578	007060'	042045	022461	020101	
2579	007066'	044123	052517	042114	
2580	007074'	041040	035105	022440	
2581	007102'	033117	040445	053440	
2582	007110'	051501	020072	047445	
2583	007116'	022466	000116		
2584	C 7122'	040445	042124	020122	FORM39::ASCIZ /%ATDR COUNTER NOT INCREMENTING%N/
2585	007130'	047503	047125	042524	
2586	007136'	020122	047516	020124	
2587	007144'	047111	051103	046505	
2588	007152'	047105	044524	043516	
2589	007160'	047045	000		
2590	007163'	045	051101	052105	FORM40::ASCIZ /%ARETRY BACKOFF WAIT TIME INTERVAL NOT VARYING%N%N/
2591	007170'	054522	041040	041501	
2592	007176'	047513	043106	053440	
2593	007204'	044501	020124	044524	
2594	007212'	042515	044440	052116	
2595	007220'	051105	040526	020114	
2596	007226'	047516	020124	040526	
2597	007234'	054522	047111	022507	
2598	007242'	022516	000116		
2599	007246'	040445	042522	051124	FORM41::ASCIZ /%ARETRY #%S4%AWAIT INTERVAL COUNT%N%N/
2600	007254'	020131	022443	032123	
2601	007262'	040445	040527	052111	
2602	007270'	044440	052116	051105	
2603	007276'	040526	020114	047503	
2604	007304'	047125	022524	022516	
2605	007312'	000116			
2606	007314'	042045	022465	033523	FORM42::ASCIZ /%D5%S7/
2607	007322'	000			
2608	007323'	045	032504	047045	FORM43::ASCIZ /%D5%N/
2609	007330'	000			
2610	007331'	045	052101	042510	FORM44::ASCIZ /%ATHE 48 BIT DESTINATION ADDRESS PATTERN IS:%N/
2611	007336'	032040	020070	044502	
2612	007344'	020124	042504	052123	
2613	007352'	047111	052101	047511	
2614	007360'	020116	042101	051104	
2615	007366'	051505	020123	040520	
2616	007374'	052124	051105	020116	
2617	007402'	051511	022472	000116	

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 56  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2618	007410'	040445	047514	042527	FORM45::ASCIZ	/%LOWER ORDER = %06XN/
2619	007416'	020122	051117	042504		
2620	007424'	020122	020075	047445		
2621	007432'	022466	000116			
2622	007436'	040445	044515	042104	FORM46::ASCIZ	/%AMIDDLE ORDER = %06XN/
2623	007444'	042514	047440	042122		
2624	007452'	051105	036440	022440		
2625	007460'	033117	047045	000		
2626	007465'	045	052501	050120	FORM47::ASCIZ	/%AUPPER ORDER = %06XN/
2627	007472'	051105	047440	042122		
2628	007500'	051105	020040	020075		
2629	007506'	047445	022466	000116		
2630	007514'	040445	042504	047125	FORM48::ASCIZ	/%ADEUNA FAILED TO RECOGNIZE A STATION ADDRESS THAT MATCHESXN/
2631	007522'	020101	040506	046111		
2632	007530'	042105	052040	020117		
2633	007536'	042522	047503	047107		
2634	007544'	055111	020105	020101		
2635	007552'	052123	052101	04751'		
2636	007560'	020116	042101	051104		
2637	007566'	051505	020123	044124		
2638	007574'	052101	046440	052101		
2639	007602'	044103	051505	047045		
2640	007610'	000				
2641	007611'	045	052101	042510	FORM49::ASCIZ	/%ATHE STATION ADDRESS IN RAM POSITION %D2XN/
2642	007616'	051440	040524	044524		
2643	007624'	047117	040440	042104		
2644	007632'	042522	051523	044440		
2645	007640'	020116	040522	020115		
2646	007646'	047520	044523	044524		
2647	007654'	047117	022440	031104		
2648	007662'	047045	000			
2649	007665'	045	051101	041505	FORM50::ASCIZ	/%ARECEIVER STATUS WORD 0 SHOULD BE: %06XS3XAWAS: %06XN/
2650	007672'	044505	042526	020122		
2651	007700'	052123	052101	051525		
2652	007706'	053440	051117	020104		
2653	007714'	020060	044123	052517		
2654	007722'	042114	041040	035105		
2655	007730'	022440	033117	051445		
2656	007736'	022463	053501	051501		
2657	007744'	020072	047445	022466		
2658	007752'	000116				
2659	007754'	040445	051103	020103	FORM51::ASCIZ	/%ACRC ERROR BIT NOT SETXN/
2660	007762'	051105	047522	020122		
2661	007770'	044502	020124	047516		
2662	007776'	020124	042523	022524		
2663	010004'	000116				
2664	010006'	040445	051105	047522	FORM52::ASCIZ	/%AERROR SUMMARY BIT NOT SETXN/
2665	010014'	020122	052523	046515		
2666	010022'	051101	020131	044502		
2667	010030'	020124	047516	020124		
2668	010036'	042523	022524	000116		
2669	010044'	040445	052516	041115	FORM53::ASCIZ	/%ANUMBER OF DATA BYTES TRANSMITTED: %D4XN/
2670	010052'	051105	047440	020106		
2671	010060'	040504	040524	041040		
2672	010066'	052131	051505	052040		
2673	010074'	040522	051516	044515		

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 57  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2674	010102'	052124	042105	020072	
2675	010110'	042045	022464	000116	
2676	010116'	040445	040504	040524	FORM54::ASCIZ /XADATA PATTERN: X06XN/
2677	010124'	050040	052101	042524	
2678	010132'	047122	020072	047445	
2679	010140'	022466	000116		
2680	010144'	040445	042504	047125	FORM55::ASCIZ /XADEUNA FAILED TO REJECT A RUNT PACKETXN/
2681	010152'	020101	040506	046111	
2682	010160'	042105	052040	020117	
2683	010166'	042522	042512	052103	
2684	010174'	040440	051040	047125	
2685	010202'	020124	040520	045503	
2686	010210'	052105	047045	000	
2687	010215'	045	042101	052505	FORM56::ASCIZ /XADEUNA FAILED TO RECOVER RECEIVE BUFFERXN/
2688	010222'	040516	043040	044501	
2689	010230'	042514	020104	047524	
2690	010236'	051040	041505	053117	
2691	010244'	051105	051040	041505	
2692	010252'	044505	042526	041040	
2693	010260'	043125	042506	022522	
2694	010266'	000116			
2695	010270'	040445	052516	041115	FORM57::ASCIZ /XANUMBER OF BYTES IN RUNT PACKET: XD2XN/
2696	010276'	051105	047440	020106	
2697	010304'	054502	042524	020123	
2698	010312'	047111	051040	047125	
2699	010320'	020124	040520	045503	
2700	010326'	052105	020072	042045	
2701	010334'	022462	000116		
2702	010340'	040445	052516	041115	FORM58::ASCIZ /XANUMBER OF BYTES IN LEGITIMATE PACKET: XD4XN/
2703	010346'	051105	047440	020106	
2704	010354'	054502	042524	020123	
2705	010362'	047111	046040	043505	
2706	010370'	052111	046511	052101	
2707	010376'	020105	040520	045503	
2708	010404'	052105	020072	042045	
2709	010412'	022464	000116		
2710	010416'	040445	042522	042503	FORM59::ASCIZ /XARECEIVE BUFFER ADDRESS AVAILABLE BEFORE RECEPTION: X06XN/
2711	010424'	053111	020105	052502	
2712	010432'	043106	051105	040440	
2713	010440'	042104	042522	051523	
2714	010446'	040440	040526	046111	
2715	010454'	041101	042514	041040	
2716	010462'	043105	051117	020105	
2717	010470'	042522	042503	052120	
2718	010476'	047511	035116	022440	
2719	010504'	033117	047045	000	
2720	010511'	045	051101	041505	FORM60::ASCIZ /XARECEIVE BUFFER ADDRESS COMPLETED AFTER RECEPTIONXN/
2721	010516'	044505	042526	041040	
2722	010524'	043125	042506	020122	
2723	010532'	042101	051104	051505	
2724	010540'	020123	047503	050115	
2725	010546'	042514	042524	020104	
2726	010554'	043101	042524	020122	
2727	010562'	042522	042503	052120	
2728	010570'	047511	022516	000116	
2729	010576'	040445	040504	040524	FORM61::ASCIZ /XADATA COMPARE ERROR IN RECOVERED RECEIVE BUFFERXN/

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 58  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2730	010604'	041440	046517	040520	
2731	010612'	042522	042440	051122	
2732	010620'	051117	044440	020116	
2733	010626'	042522	047503	042526	
2734	010634'	042522	020104	042522	
2735	010642'	042503	053111	020105	
2736	010650'	052502	043106	051105	
2737	010656'	047045	000		
2738	010661'	045	022524	020101	FORM62::ASCIZ /XTXA DID NOT CLEAR AFTER WRITING 1 TO ITXN/
2739	010666'	044504	020104	047516	
2740	010674'	020124	046103	040505	
2741	010702'	020122	043101	042524	
2742	010710'	020122	051127	052111	
2743	010716'	047111	020107	020061	
2744	010724'	047524	044440	022524	
2745	010732'	000116			
2746	010734'	040445	042523	043114	FORM63::ASCIZ /%ASELF TEST ERROR CODE = %02XN/
2747	010742'	052040	051505	020124	
2748	010750'	051105	047522	020122	
2749	010756'	047503	042504	036440	
2750	010764'	022440	031117	047045	
2751	010772'	000			
2752	010773'	045	022516	051101	FORM64::ASCIZ /%NXAROM MICROCODE VERSION (DECIMAL): %D2/
2753	011000'	046517	046440	041511	
2754	011006'	047522	047503	042504	
2755	011014'	053040	051105	044523	
2756	011022'	047117	024040	042504	
2757	011030'	044503	040515	024514	
2758	011036'	020072	042045	000062	
2759	011044'	047045	040445	053523	FORM65::ASCIZ /%NXASWITCH PACK = %06/
2760	011052'	052111	044103	050040	
2761	011060'	041501	020113	020075	
2762	011066'	047445	000066		
2763	011072'	040445	047516	044440	FORM66::ASCIZ /%XANO INTERRUPT FROM TRANSMIT STATE MACHINE TO T-11XN/
2764	011100'	052116	051105	052522	
2765	011106'	052120	043040	047522	
2766	011114'	020115	051124	047101	
2767	011122'	046523	052111	051440	
2768	011130'	040524	042524	046440	
2769	011136'	041501	044510	042516	
2770	011144'	052040	020117	026524	
2771	011152'	030461	047045	000	
2772	011157'	045	047101	020117	FORM67::ASCIZ /%XANO INTERRUPT FROM RECEIVE STATE MACHINE TO T-11XN/
2773	011164'	047111	042524	051122	
2774	011172'	050125	020124	051106	
2775	011200'	046517	051040	041505	
2776	011206'	044505	042526	051440	
2777	011214'	040524	042524	046440	
2778	011222'	041501	044510	042516	
2779	011230'	052040	020117	026524	
2780	011236'	030461	047045	000	
2781	011243'	045	051101	041505	FORM68::ASCIZ /%XARECEIVER BYTE COUNTER FAILED TO LOCK UP AT MAXIMUM VALUEXN/
2782	011250'	044505	042526	020122	
2783	011256'	054502	042524	041440	
2784	011264'	052517	052116	051105	
2785	011272'	043040	044501	042514	

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 59  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2786	011300'	020104	047524	046040	
2787	011306'	041517	020113	050125	
2788	011314'	040440	020124	040515	
2789	011322'	044530	052515	020115	
2790	011330'	040526	052514	022505	
2791	011336'	000116			
2792	011340'	040445	042504	047125	FORM69::.ASCII /%ADEUNA DID NOT REJECT A PACKET TRANSMITTED TO ITSELF/
2793	011346'	020101	044504	020104	
2794	011354'	047516	020124	042522	
2795	011362'	042512	052103	040440	
2796	011370'	050040	041501	042513	
2797	011376'	020124	051124	047101	
2798	011404'	046523	052111	042524	
2799	011412'	020104	047524	044440	
2800	011420'	051524	046105	106	
2801	011425'	045	020101	047111	.ASCIZ /%A IN HALF-DUPLEX MODE%N/
2802	011432'	044040	046101	026506	
2803	011440'	052504	046120	054105	
2804	011446'	046440	042117	022505	
2805	011454'	000116			
2806	011456'	040445	051124	047101	FORM70::.ASCIZ /%ATRANSMIT BUFFER ADDRESS = %06%N/
2807	011464'	046523	052111	041040	
2808	011472'	043125	042506	020122	
2809	011500'	042101	051104	051505	
2810	011506'	020123	020075	047445	
2811	011514'	022466	000116		
2812	011520'	040445	042522	042503	FORM71::.ASCIZ /%ARECEIVE BUFFER ADDRESS = %06%N/
2813	011526'	053111	020105	052502	
2814	011534'	043106	051105	040440	
2815	011542'	042104	042522	051523	
2816	011550'	020040	020075	047445	
2817	011556'	022466	000116		
2818	011562'	040445	030524	020061	FORM72::.ASCIZ /%AT11 LINK MEMORY PARITY ERROR OCCURRED%N/
2819	011570'	044514	045516	046440	
2820	011576'	046505	051117	020131	
2821	011604'	040520	044522	054524	
2822	011612'	042440	051122	051117	
2823	011620'	047440	041503	051125	
2824	011626'	042522	022504	000116	
2825	011634'	040445	030524	020061	FORM73::.ASCIZ /%AT11 NPR TIMEOUT ERROR OCCURRED%N/
2826	011642'	050116	020122	044524	
2827	011650'	042515	052517	020124	
2828	011656'	051105	047522	020122	
2829	011664'	041517	052503	051122	
2830	011672'	042105	047045	000	
2831	011677'	045	052101	030461	FORM74::.ASCIZ /%AT11 NON-EXISTANT MEMORY TIMEOUT OCCURRED%N/
2832	011704'	047040	047117	042455	
2833	011712'	044530	052123	047101	
2834	011720'	020124	042515	047515	
2835	011726'	054522	052040	046511	
2836	011734'	047505	052125	047440	
2837	011742'	041503	051125	042522	
2838	011750'	022504	000116		
2839	011754'	040445	030524	020061	FORM75::.ASCIZ /%AT11 UNEXPECTED INTERRUPT OCCURRED%N/
2840	011762'	047125	054105	042520	
2841	011770'	052103	042105	044440	

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 60  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2842	011776'	052116	051105	052522	
2843	012004'	052120	047440	041503	
2844	012012'	051125	042522	022504	
2845	012020'	000116			
2846	012022'	040445	040515	041524	FORM76:::ASCIZ /%AMATCH BIT FAILED TO SET%N/
2847	012030'	020110	044502	020124	
2848	012036'	040506	046111	042105	
2849	012044'	052040	020117	042523	
2850	012052'	022524	000116		
2851	012056'	040445	040515	041524	FORM77:::ASCIZ /%AMATCH BIT SET BUT NO RECEIVER INTERRUPT%N/
2852	012064'	020110	044502	020124	
2853	012072'	042523	020124	052502	
2854	012100'	020124	047516	051040	
2855	012106'	041505	044505	042526	
2856	012114'	020122	047111	042524	
2857	012122'	051122	050125	022524	
2858	012130'	000116			
2859	012132'	040445	044123	052517	FORM78:::ASCIZ /%ASHOULD BE: %06XS2%AWAS: %06%N/
2860	012140'	042114	041040	035105	
2861	012146'	022440	033117	051445	
2862	012154'	022462	053501	051501	
2863	012162'	020072	047445	022466	
2864	012170'	000116			
2865	012172'	040445	042504	047125	FORM79:::ASCIZ /%ADEUNA FAILED TO REJECT A DATAGRAM.%N/
2866	012200'	020101	040506	046111	
2867	012206'	042105	052040	020117	
2868	012214'	042522	042512	052103	
2869	012222'	040440	042040	052101	
2870	012230'	043501	040522	027115	
2871	012236'	047045	000		
2872	012241'	045	042101	051505	FORM80:::ASCIZ /%ADESTINATION ADDRESS OF DATAGRAM IS ALL 1'S%N/
2873	012246'	044524	040516	044524	
2874	012254'	047117	040440	042104	
2875	012262'	042522	051523	047440	
2876	012270'	020106	040504	040524	
2877	012276'	051107	046501	044440	
2878	012304'	020123	046101	020114	
2879	012312'	023461	022523	000116	
2880	012320'	040445	052123	052101	FORM81:::ASCIZ /%ASTATION ADDRESS RAM IS ALL 0'S%N/
2881	012326'	047511	020116	042101	
2882	012334'	051104	051505	020123	
2883	012342'	040522	020115	051511	
2884	012350'	040440	046114	030040	
2885	012356'	051447	047045	000	
2886	012363'	045	052101	046511	FORM82:::ASCIZ /%ATIMEOUT WAITING FOR MICROCODE TO ENTER MICROMONITOR%N/
2887	012370'	047505	052125	053440	
2888	012376'	044501	044524	043516	
2889	012404'	043040	051117	046440	
2890	012412'	041511	047522	047503	
2891	012420'	042504	052040	020117	
2892	012426'	047105	042524	020122	
2893	012434'	044515	051103	046517	
2894	012442'	047117	052111	051117	
2895	012450'	047045	000		
2896	012453'	045	050101	051503	FORM83:::ASCIZ /%APCSR%D1% = %06%N/
2897	012460'	022522	030504	040445	

:MAC001

GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 61  
CZUAAB.MAC 07-APR-83 17:03 GLOBAL TEXT SECTION

2898 012466' 036440 022440 033117  
2899 012474' 047045 000  
2900 012500'  
2901  
2902

.EVEN



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 62  
 CZUAAB.MAC 07-APR-83 17:03

GLOBAL ERROR REPORT SECTION

```

2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914 012500'          BGNMSG RACM61
2915 012500'
2916 012500'          PRINTB #FORM1,CSRNUM          RACM61::
2917 012500' 013746 000304'          MOV CSRNUM,-(SP)
2918 012504' 012746 004141'          MOV #FORM1,-(SP)
2919 012510' 012746 000002          MOV #2,-(SP)
2920 012514' 010600          MOV SP,RO
2921 012516' 104414          TRAP C$PNTB
2922 012520' 062706 000006          ADD #6,SP
2923 012524'          ENDMMSG
2924 012524'
2925 012524' 104423          L10001: TRAP C$MSG
2926
2927 012526'          BGNMSG RACM62
2928 012526'
2929 012526'          PRINTB #FORM2,CSRNUM,BITNUM,BITSTA          RACM62::
2930 012526' 013746 000312'          MOV BITSTA,-(SP)
2931 012532' 013746 000306'          MOV BITNUM,-(SP)
2932 012536' 013746 000304'          MOV CSRNUM,-(SP)
2933 012542' 012746 004176'          MOV #FORM2,-(SP)
2934 012546' 012746 000004          MOV #4,-(SP)
2935 012552' 010600          MOV SP,RO
2936 012554' 104414          TRAP C$PNTB
2937 012556' 062706 000012          ADD #12,SP
2938 012562' 004737 017764'          JSR PC,PRNPCR          ;PRINT PCSR'S          :MAC001
2939 012566'          ENDMMSG
2940 012566'
2941 012566' 104423          L10002: TRAP C$MSG
2942
2943 012570'          BGNMSG RACM63
2944 012570'
2945 012570'          PRINTB #FORM3,CSRNUM          RACM63::
2946 012570' 013746 000304'          MOV CSRNUM,-(SP)
2947 012574' 012746 004234'          MOV #FORM3,-(SP)
2948 012600' 012746 000002          MOV #2,-(SP)
2949 012604' 010600          MOV SP,RO
2950 012606' 104414          TRAP C$PNTB
2951 012610' 062706 000006          ADD #6,SP
2952 012614'          PRINTB #FORM4,R3,R4          MOV R4,-(SP)
2953 012614' 010446          MOV R3,-(SP)
2954 012616' 010346          MOV #FORM4,-(SP)
2955 012620' 012746 004303'          MOV #3,-(SP)
2956 012624' 012746 000003          MOV #3,-(SP)
2957 012630' 010600          MOV SP,RO
2958 012632' 104414          TRAP C$PNTB

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 63  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

2959	012634	062706	000010						
2960	012640			ENDMSG				ADD	#10,SP
2961	012640								
2962	012640	104423					L10003:	TRAP	C8MSG
2963									
2964	012642			BGNMSG	RACMG4				
2965	012642								
2966	012642				PRINTB	#FORM5			RACMG4::
2967	012642	012746	004352					MOV	#FORM5,-(SP)
2968	012646	012746	000001					MOV	#1,-(SP)
2969	012652	010600						MOV	SP,RO
2970	012654	104414						TRAP	C\$PNTB
2971	012656	062706	000004					ADD	#4,SP
2972	012662	004737	017764						:MAC001
2973	012666			ENDMSG	JSR	PC,PRNPCR			:PRINT PCSR'S
2974	012666								L10004:
2975	012666	104423						TRAP	C8MSG
2976									
2977	012670			BGNMSG	RACMG7				
2978	012670								
2979	012670				PRINTB	#FORM8			RACMG7::
2980	012670	012746	004555					MOV	#FORM8,-(SP)
2981	012674	012746	000001					MOV	#1,-(SP)
2982	012700	010600						MOV	SP,RO
2983	012702	104414						TRAP	C\$PNTB
2984	012704	062706	000004					ADD	#4,SP
2985	012710	004737	017764						:MAC001
2986	012714			ENDMSG	JSR	PC,PRNPCR			:PRINT PCSR'S
2987	012714								L10005:
2988	012714	104423						TRAP	C8MSG
2989									
2990	012716			BGNMSG	MSG1				
2991	012716								
2992	012716				PRINTB	#FORM6,BITNAM,BITSTA,PWHEN,PCOMND			MSG1::
2993	012716	013746	000316					MOV	PCOMND,-(SP)
2994	012722	013746	000314					MOV	PWHEN,-(SP)
2995	012726	013746	000312					MOV	BITSTA,-(SP)
2996	012732	013746	000310					MOV	BITNAM,-(SP)
2997	012736	012746	004415					MOV	#FORM6,-(SP)
2998	012742	012746	000005					MOV	#5,-(SP)
2999	012746	010600						MOV	SP,RO
3000	012750	104414						TRAP	C\$PNTB
3001	012752	062706	000014					ADD	#14,SP
3002	012756	004737	017764						:MAC001
3003	012762			ENDMSG	JSR	PC,PRNPCR			:PRINT PCSR'S
3004	012762								L10006:
3005	012762	104423						TRAP	C8MSG
3006									
3007	012764			BGNMSG	MSG2				
3008	012764								
3009	012764				PRINTB	#FORM7			MSG2::
3010	012764	012746	004473					MOV	#FORM7,-(SP)
3011	012770	012746	000001					MOV	#1,-(SP)
3012	012774	010600						MOV	SP,RO
3013	012776	104414						TRAP	C\$PNTB
3014	013000	062706	000004					ADD	#4,SP

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 64  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3015	013004'								
3016	013004'	010446							
3017	013006'	012746	005123'					MOV	R4,-(SP)
3018	013012'	012746	000002					MOV	#FORM13,-(SP)
3019	013016'	010600						MOV	#2,-(SP)
3020	013020'	104414						MOV	SP,RO
3021	013022'	062706	000006					TRAP	CSPNTB
3022	013026'			ENDMSG				ADD	#6,SP
3023	013026'								
3024	013026'	104423					L10007:	TRAP	CMSG
3025									
3026	013030'			BGNMSG	MSG3				
3027	013030'								
3028	013030'								MSG3::
3029	013030'	013746	000314'						
3030	013034'	013746	000312'					MOV	PWHEN,-(SP)
3031	013040'	013746	000310'					MOV	BITSTA,-(SP)
3032	013044'	013746	000304'					MOV	BITNAM,-(SP)
3033	013050'	012746	004637'					MOV	CSRNUM,-(SP)
3034	013054'	012746	000005					MOV	#FORM9,-(SP)
3035	013060'	010600						MOV	#5,-(SP)
3036	013062'	104414						MOV	SP,RO
3037	013064'	062706	000014					TRAP	CSPNTB
3038	013070'	004737	017764'					ADD	#14,SP
3039	013074'			ENDMSG	JSR	PC,PRNPCR			:PRINT PCSR'S
3040	013074'								
3041	013074'	104423					L10010:	TRAP	CMSG
3042									
3043	013076'			BGNMSG	MSG4				
3044	013076'								
3045	013076'								MSG4::
3046	013076'	012746	004715'						
3047	013102'	012746	000001					MOV	#FORM10,-(SP)
3048	013106'	010600						MOV	#1,-(SP)
3049	013110'	104414						MOV	SP,RO
3050	013112'	062706	000004					TRAP	CSPNTB
3051	013116'	004737	017764'					ADD	#4,SP
3052	013122'			ENDMSG	JSR	PC,PRNPCR			:PRINT PCSR'S
3053	013122'								
3054	013122'	104423					L10011:	TRAP	CMSG
3055									
3056	013124'			BGNMSG	MSG5				
3057	013124'								
3058	013124'								MSG5::
3059	013124'	012746	005001'						
3060	013130'	012746	000001					MOV	#FORM11,-(SP)
3061	013134'	010600						MOV	#1,-(SP)
3062	013136'	104414						MOV	SP,RO
3063	013140'	062706	000004					TRAP	CSPNTB
3064	013144'							ADD	#4,SP
3065	013144'	011146							
3066	013146'	011446						MOV	(R1),-(SP)
3067	013150'	010146						MOV	(R4),-(SP)
3068	013152'	012746	005030'					MOV	R1,-(SP)
3069	013156'	012746	000004					MOV	#FORM12,-(SP)
3070	013162'	010600						MOV	#4,-(SP)
								MOV	SP,RO

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 65  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3071	013164'	104414							
3072	013166'	062706	000012					TRAP	CSPNTB
3073	013172'			ENDMSG				ADD	#12,SP
3074	013172'							L10012:	
3075	013172'	104423						TRAP	C\$MSG
3076									
3077	013174'			BGNMSG	MSG6				
3078	013174'							MSG6::	
3079	013174'				PRINTB	#FORM11			
3080	013174'	012746	005001'					MOV	#FORM11,-(SP)
3081	013200'	012746	000001					MOV	#1,-(SP)
3082	013204'	010600						MOV	SP,RO
3083	013206'	104414						TRAP	CSPNTB
3084	013210'	062706	000004					ADD	#4,SP
3085	013214'				PRINTB	#FORM15,R1			
3086	013214'	010146						MOV	R1,-(SP)
3087	013216'	012746	005174'					MOV	#FORM15,-(SP)
3088	013222'	012746	000002					MOV	#2,-(SP)
3089	013226'	010600						MOV	SP,RO
3090	013230'	104414						TRAP	CSPNTB
3091	013232'	062706	000006					ADD	#6,SP
3092	013236'				PRINTB	#FORM16,R2			
3093	013236'	010246						MOV	R2,-(SP)
3094	013240'	012746	005250'					MOV	#FORM16,-(SP)
3095	013244'	012746	000002					MOV	#2,-(SP)
3096	013250'	010600						MOV	SP,RO
3097	013252'	104414						TRAP	CSPNTB
3098	013254'	062706	000006					ADD	#6,SP
3099	013260'			ENDMSG					
3100	013260'							L10013:	
3101	013260'	104423						TRAP	C\$MSG
3102									
3103	013262'			BGNMSG	MSG7				
3104	013262'							MSG7::	
3105	013262'				PRINTB	#FORM17			
3106	013262'	012746	005326'					MOV	#FORM17,-(SP)
3107	013266'	012746	000001					MOV	#1,-(SP)
3108	013272'	010600						MOV	SP,RO
3109	013274'	104414						TRAP	CSPNTB
3110	013276'	062706	000004					ADD	#4,SP
3111	013302'	004737	017764'		JSR	PC,PRNPCR		:PRINT	PCSR'S
3112	013306'			ENDMSG					:MAC001
3113	013306'							L10014:	
3114	013306'	104423						TRAP	C\$MSG
3115									
3116	013310'			BGNMSG	MSG8				
3117	013310'							MSG8::	
3118	013310'				PRINTB	#FORM18			
3119	013310'	012746	005411'					MOV	#FORM18,-(SP)
3120	013314'	012746	000001					MOV	#1,-(SP)
3121	013320'	010600						MOV	SP,RO
3122	013322'	104414						TRAP	CSPNTB
3123	013324'	062706	000004					ADD	#4,SP
3124	013330'			ENDMSG					
3125	013330'							L10015:	
3126	013330'	104423						TRAP	C\$MSG

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 66  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

SEQ 61

3127							
3128	013332'	BGNMSG	MSG9				
3129	013332'					MSG9::	
3130	013332'		PRINTB	#FORM19, @BITNAM			
3131	013332'	017746	164752				MOV @BITNAM, -(SP)
3132	013336'	012746	005467'				MOV #FORM19, -(SP)
3133	013342'	012746	000002				MOV #2, -(SP)
3134	013346'	010600					MOV SP, R0
3135	013350'	104414					TRAP C\$PNTB
3136	013352'	062706	000006				ADD #6, SP
3137	013356'	ENDMSG					
3138	013356'					L10016:	
3139	013356'	104423					TRAP C\$MSG
3140							
3141	013360'	BGNMSG	MSG10				
3142	013360'					MSG10::	
3143	013360'		PRINTB	#FORM20			
3144	013360'	012746	005544'				MOV #FORM20, -(SP)
3145	013364'	012746	000001				MOV #1, -(SP)
3146	013370'	010600					MOV SP, R0
3147	013372'	104414					TRAP C\$PNTB
3148	013374'	062706	000004				ADD #4, SP
3149	013400'	ENDMSG					
3150	013400'					L10017:	
3151	013400'	104423					TRAP C\$MSG
3152							
3153	013402'	BGNMSG	MSG11				
3154	013402'					MSG11::	
3155	013402'		PRINTB	#FORM21			
3156	013402'	012746	005604'				MOV #FORM21, -(SP)
3157	013406'	012746	000001				MOV #1, -(SP)
3158	013412'	010600					MOV SP, R0
3159	013414'	104414					TRAP C\$PNTB
3160	013416'	062706	000004				ADD #4, SP
3161	013422'		PRINTB	#FORM22			
3162	013422'	012746	005656'				MOV #FORM22, -(SP)
3163	013426'	012746	000001				MOV #1, -(SP)
3164	013432'	010600					MOV SP, R0
3165	013434'	104414					TRAP C\$PNTB
3166	013436'	062706	000004				ADD #4, SP
3167	013442'		PRINTB	#FORM23, R1			
3168	013442'	010146					MOV R1, -(SP)
3169	013444'	012746	005736'				MOV #FORM23, -(SP)
3170	013450'	012746	000002				MOV #2, -(SP)
3171	013454'	010600					MOV SP, R0
3172	013456'	104414					TRAP C\$PNTB
3173	013460'	062706	000006				ADD #6, SP
3174	013464'	ENDMSG					
3175	013464'					L10020:	
3176	013464'	104423					TRAP C\$MSG
3177							
3178	013466'	BGNMSG	MSG12				
3179	013466'					MSG12::	
3180	013466'		PRINTB	#FORM24, R2			
3181	013466'	010246					MOV R2, -(SP)
3182	01347C'	012746	006006'				MOV #FORM24, -(SP)

62GLOBAL AREAS MAC11 30A(1052) 07-APR-83 17:13 PAGE 67  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3183	013474'	012746	000002		
3184	013500'	010600			MOV #2,-(SP)
3185	013502'	104414			MOV SP,R0
3186	013504'	062706	000006		TRAP C\$PNTB
3187	013510'			ENDMSG	ADD #6,SP
3188	013510'				
3189	013510'	104423			L10021: TRAP C\$MSG
3190					
3191	013512'			BGNMSG MSG13	
3192	013512'				MSG13::
3193	013512'			PRINTB #FORM25,R3	
3194	013512'	010346			
3195	013514'	012746	006040'		MOV R3,-(SP)
3196	013520'	012746	000002		MOV #FORM25,-(SP)
3197	013524'	010600			MOV #2,-(SP)
3198	013526'	104414			MOV SP,R0
3199	013530'	062706	000006		TRAP C\$PNTB
3200	013534'			PRINTB #FORM26,R4	ADD #6,SP
3201	013534'	010446			
3202	013536'	012746	006113'		MOV R4,-(SP)
3203	013542'	012746	000002		MOV #FORM26,-(SP)
3204	013546'	010600			MOV #2,-(SP)
3205	013550'	104414			MOV SP,R0
3206	013552'	062706	000006		TRAP C\$PNTB
3207	013556'			ENDMSG	ADD #6,SP
3208	013556'				
3209	013556'	104423			L10022: TRAP C\$MSG
3210					
3211	013560'			BGNMSG MSG14	
3212	013560'				MSG14::
3213	013560'			PRINTB #FORM27,R3,R4	
3214	013560'	010446			
3215	013562'	010346			MOV R4,-(SP)
3216	013564'	012746	006166'		MOV R3,-(SP)
3217	013570'	012746	000003		MOV #FORM27,-(SP)
3218	013574'	010600			MOV #3,-(SP)
3219	013576'	104414			MOV SP,R0
3220	013600'	062706	000010		TRAP C\$PNTB
3221	013604'			ENDMSG	ADD #10,SP
3222	013604'				
3223	013604'	104423			L10023: TRAP C\$MSG
3224					
3225	013606'			BGNMSG MSG15	
3226	013606'				MSG15::
3227	013606'			PRINTB #FORM12,R2,(R1),(R2)	
3228	013606'	011246			
3229	013610'	011146			MOV (R2),-(SP)
3230	013612'	010246			MOV (R1),-(SP)
3231	013614'	012746	005030'		MOV R2,-(SP)
3232	013620'	012746	000004		MOV #FORM12,-(SP)
3233	013624'	010600			MOV #4,-(SP)
3234	013626'	104414			MOV SP,R0
3235	013630'	062706	000012		TRAP C\$PNTB
3236	013634'			ENDMSG	ADD #12,SP
3237	013634'				
3238	013634'	104423			L10024: TRAP C\$MSG

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 68  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3239						
3240	013636'	BGNMSG	MSG16			
3241	013636'					
3242	013636'				MSG16::	
3243	013636' 010246			PRINTB	#FORM28,R2	
3244	013640' 012746 006236'					MOV R2,-(SP)
3245	013644' 012746 000002					MOV #FORM28,-(SP)
3246	013650' 010600					MOV #2,-(SP)
3247	013652' 104414					MOV SP,R0
3248	013654' 062706 000006					TRAP C\$PNTB
3249	013660'			PRINTB	#FORM29,R1	ADD #6,SP
3250	013660' 010146					
3251	013662' 012746 006243'					MOV R1,-(SP)
3252	013666' 012746 000002					MOV #FORM29,-(SP)
3253	013672' 010600					MOV #2,-(SP)
3254	013674' 104414					MOV SP,R0
3255	013676' 062706 000006					TRAP C\$PNTB
3256	013702'			PRINTB	#FORM27,R3,R4	ADD #6,SP
3257	013702' 010446					
3258	013704' 010346					MOV R4,-(SP)
3259	013706' 012746 006166'					MOV R3,-(SP)
3260	013712' 012746 000003					MOV #FORM27,-(SP)
3261	013716' 010600					MOV #3,-(SP)
3262	013720' 104414					MOV SP,R0
3263	013722' 062706 000010					TRAP C\$PNTB
3264	013726'	ENDMSG				ADD #10,SP
3265	013726'					
3266	013726' 104423				L10025:	TRAP C\$MSG
3267						
3268	013730'	BGNMSG	MSG17			
3269	013730'					
3270	013730'					
3271	013730' 010446			PRINTB	#FORM30,R4	
3272	013732' 012746 006266'					MOV R4,-(SP)
3273	013736' 012746 000002					MOV #FORM30,-(SP)
3274	013742' 010600					MOV #2,-(SP)
3275	013744' 104414					MOV SP,R0
3276	013746' 062706 000006					TRAP C\$PNTB
3277	013752'			PRINTB	#FORM31,R1	ADD #6,SP
3278	013752' 010146					
3279	013754' 012746 006316'					MOV R1,-(SP)
3280	013760' 012746 000002					MOV #FORM31,-(SP)
3281	013764' 010600					MOV #2,-(SP)
3282	013766' 104414					MOV SP,R0
3283	013770' 062706 000006					TRAP C\$PNTB
3284	013774'			PRINTB	#FORM27,R2,R3	ADD #6,SP
3285	013774' 010346					
3286	013776' 010246					MOV R3,-(SP)
3287	014000' 012746 006166'					MOV R2,-(SP)
3288	014004' 012746 000003					MOV #FORM27,-(SP)
3289	014010' 010600					MOV #3,-(SP)
3290	014012' 104414					MOV SP,R0
3291	014014' 062706 000010					TRAP C\$PNTB
3292	014020'	ENDMSG				ADD #10,SP
3293	014020'					
3294	014020' 104423				L10026:	TRAP C\$MSG

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 69  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3295							
3296	014022'			BGNMSG	MSG18		
3297	014022'					MSG18::	
3298	014022'				PRINTB	#FORM32,R1,BITNAM	
3299	014022'	013746	000310'				MOV BITNAM,-(SP)
3300	014026'	010146					MOV R1,-(SP)
3301	014030'	012746	006344'				MOV #FORM32,-(SP)
3302	014034'	012746	000003				MOV #3,-(SP)
3303	014040'	010600					MOV SP,R0
3304	014042'	104414					TRAP CSPNTB
3305	014044'	062706	000010				ADD #10,SP
3306	014050'			ENDMSG			
3307	014050'					L10027:	
3308	014050'	104423				TRAP	C\$MSG
3309							
3310	014052'			BGNMSG	MSG19		
3311	014052'					MSG19::	
3312	014052'				PRINTB	#FORM33,R1	
3313	014052'	010146					MOV R1,-(SP)
3314	014054'	012746	006441'				MOV #FORM33,-(SP)
3315	014060'	012746	000002				MOV #2,-(SP)
3316	014064'	010600					MOV SP,R0
3317	014066'	104414					TRAP CSPNTB
3318	014070'	062706	000006				ADD #6,SP
3319	014074'				PRINTB	#FORM34,R2,R3	
3320	014074'	010346					MOV R3,-(SP)
3321	014076'	010246					MOV R2,-(SP)
3322	014100'	012746	006477'				MOV #FORM34,-(SP)
3323	014104'	012746	000003				MOV #3,-(SP)
3324	014110'	010600					MOV SP,R0
3325	014112'	104414					TRAP CSPNTB
3326	014114'	062706	000010				ADD #10,SP
3327	014120'				PRINTB	#FORM35,R1,R4	
3328	014120'	010446					MOV R4,-(SP)
3329	014122'	010146					MOV R1,-(SP)
3330	014124'	012746	006562'				MOV #FORM35,-(SP)
3331	014130'	012746	000003				MOV #3,-(SP)
3332	014134'	010600					MOV SP,R0
3333	014136'	104414					TRAP CSPNTB
3334	014140'	062706	000010				ADD #10,SP
3335	014144'			ENDMSG			
3336	014144'					L10030:	
3337	014144'	104423				TRAP	C\$MSG
3338							
3339	014146'			BGNMSG	MSG20		
3340	014146'					MSG20::	
3341	014146'				PRINTB	#FORM11	
3342	014146'	012746	005001'				MOV #FORM11,-(SP)
3343	014152'	012746	000001				MOV #1,-(SP)
3344	014156'	010600					MOV SP,R0
3345	014160'	104414					TRAP CSPNTB
3346	014162'	062706	000004				ADD #4,SP
3347	014166'				PRINTB	#FORM36,R3	
3348	014166'	010346					MOV R3,-(SP)
3349	014170'	012746	006662'				MOV #FORM36,-(SP)
3350	014174'	012746	000002				MOV #2,-(SP)



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 70  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3351	014200	010600				MOV	SP,RO
3352	014202	104414				TRAP	C\$PNTB
3353	014204	062706	000006			ADD	#6,SP
3354	014210			PRINTB	#FORM27,R1,R2		
3355	014210	010246				MOV	R2,-(SP)
3356	014212	010146				MOV	R1,-(SP)
3357	014214	012746	006166			MOV	#FORM27,-(SP)
3358	014220	012746	000003			MOV	#3,-(SP)
3359	014224	010600				MOV	SP,RO
3360	014226	104414				TRAP	C\$PNTB
3361	014230	062706	000010			ADD	#10,SP
3362	014234			ENDMSG			
3363	014234						
3364	014234	104423				L10031:	TRAP
3365							C\$MSG
3366	014236			BGNMSG	MSG21		
3367	014236					MSG21::	
3368	014236			PRINTB	#FORM11		
3369	014236	012746	005001			MOV	#FORM11,-(SP)
3370	014242	012746	000001			MOV	#1,-(SP)
3371	014246	010600				MOV	SP,RO
3372	014250	104414				TRAP	C\$PNTB
3373	014252	062706	000004			ADD	#4,SP
3374	014256			PRINTB	#FORM29,R1		
3375	014256	010146				MOV	R1,-(SP)
3376	014260	012746	006243			MOV	#FORM29,-(SP)
3377	014264	012746	000002			MOV	#2,-(SP)
3378	014270	010600				MOV	SP,RO
3379	014272	104414				TRAP	C\$PNTB
3380	014274	062706	000006			ADD	#6,SP
3381	014300			PRINTB	#FORM27,R2,R3		
3382	014300	010346				MOV	R3,-(SP)
3383	014302	010246				MOV	R2,-(SP)
3384	014304	012746	006166			MOV	#FORM27,-(SP)
3385	014310	012746	000003			MOV	#3,-(SP)
3386	014314	010600				MOV	SP,RO
3387	014316	104414				TRAP	C\$PNTB
3388	014320	062706	000010			ADD	#10,SP
3389	014324			ENDMSG			
3390	014324					L10032:	
3391	014324	104423				TRAP	C\$MSG
3392							
3393	014326			BGNMSG	MSG22		
3394	014326					MSG22::	
3395	014326	010146		MOV	R1,-(SP)		
3396	014330			PRINTB	#FORM37,R2		
3397	014330	010246				MOV	R2,-(SP)
3398	014332	012746	006720			MOV	#FORM37,-(SP)
3399	014336	012746	000002			MOV	#2,-(SP)
3400	014342	010600				MOV	SP,RO
3401	014344	104414				TRAP	C\$PNTB
3402	014346	062706	000006			ADD	#6,SP
3403	014352	005001		CLR	R1		
3404	014354			PRINTB	#FORM38,R1,R3,PCBB+2		
3405	014354	013746	000610			MOV	PCBB+2,-(SP)
3406	014360	010346				MOV	R3,-(SP)





62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 73  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3519	014770'	010600						
3520	014772'	104414					MOV	SP,RO
3521	014774'	062706	000006				TRAP	C\$PNTB
3522	015000'			ENDMSG			ADD	#6,SP
3523	015000'							
3524	015000'	104423				L10036:	TRAP	C\$MSG
3525								
3526	015002'			BGNMSG	MSG26			
3527	015002'					MSG26::		
3528	015002'				PRINTB	#FORM48		
3529	015002'	012746	007514'				MOV	#FORM48,-(SP)
3530	015006'	012746	000001				MOV	#1,-(SP)
3531	015012'	010600					MOV	SP,RO
3532	015014'	104414					TRAP	C\$PNTB
3533	015016'	062706	000004				ADD	#4,SP
3534	015022'				PRINTB	#FORM49,R1		
3535	015022'	010146					MOV	R1,-(SP)
3536	015024'	012746	007611'				MOV	#FORM49,-(SP)
3537	015030'	012746	000002				MOV	#2,-(SP)
3538	015034'	010600					MOV	SP,RO
3539	015036'	104414					TRAP	C\$PNTB
3540	015040'	062706	000006				ADD	#6,SP
3541	015044'			ENDMSG				
3542	015044'					L10037:	TRAP	C\$MSG
3543	015044'	104423						
3544								
3545	015046'			BGNMSG	MSG27			
3546	015046'					MSG27::		
3547	015046'				PRINTB	#FORM51		
3548	015046'	012746	007754'				MOV	#FORM51,-(SP)
3549	015052'	012746	000001				MOV	#1,-(SP)
3550	015056'	010600					MOV	SP,RO
3551	015060'	104414					TRAP	C\$PNTB
3552	015062'	062706	000004				ADD	#4,SP
3553	015066'				PRINTB	#FORM50,R4,R3		
3554	015066'	010346					MOV	R3,-(SP)
3555	015070'	010446					MOV	R4,-(SP)
3556	015072'	012746	007665'				MOV	#FORM50,-(SP)
3557	015076'	012746	000003				MOV	#3,-(SP)
3558	015102'	010600					MOV	SP,RO
3559	015104'	104414					TRAP	C\$PNTB
3560	015106'	062706	000010				ADD	#10,SP
3561	015112'			ENDMSG				
3562	015112'					L10040:	TRAP	C\$MSG
3563	015112'	104423						
3564								
3565	015114'			BGNMSG	MSG28			
3566	015114'					MSG28::		
3567	015114'				PRINTB	#FORM52		
3568	015114'	012746	010006'				MOV	#FORM52,-(SP)
3569	015120'	012746	000001				MOV	#1,-(SP)
3570	015124'	010600					MOV	SP,RO
3571	015126'	104414					TRAP	C\$PNTB
3572	015130'	062706	000004				ADD	#4,SP
3573	015134'				PR.NTB	#FORM50,R4,R3		
3574	015134'	010346					MOV	R3,-(SP)

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 74  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3575	015136'	010446			MOV	R4,-(SP)
3576	015140'	012746	007665'		MOV	#FORM50,-(SP)
3577	015144'	012746	000003		MOV	#3,-(SP)
3578	015150'	010600			MOV	SP,RO
3579	015152'	104414			TRAP	C\$PNTB
3580	015154'	062706	000010		ADD	#10,SP
3581	015160'			ENDMSG		
3582	015160'				L10041:	
3583	015160'	104423			TRAP	C\$MSG
3584						
3585	015162'			BGNMSG	MSG29	
3586	015162'				MSG29::	
3587	015162'			PRINTB	#FORM50,#0,PCBB+4	
3588	015162'	013746	000612'		MOV	PCBB+4,-(SP)
3589	015166'	012746	000000		MOV	#0,-(SP)
3590	015172'	012746	007665'		MOV	#FORM50,-(SP)
3591	015176'	012746	000003		MOV	#3,-(SP)
3592	015202'	010600			MOV	SP,RO
3593	015204'	104414			TRAP	C\$PNTB
3594	015206'	062706	000010		ADD	#10,SP
3595	015212'			PRINTB	#FORM53,PCBB+2	
3596	015212'	013746	000610'		MOV	PCBB+2,-(SP)
3597	015216'	012746	010044'		MOV	#FORM53,-(SP)
3598	015222'	012746	000002		MOV	#2,-(SP)
3599	015226'	010600			MOV	SP,RO
3600	015230'	104414			TRAP	C\$PNTB
3601	015232'	062706	000006		ADD	#6,SP
3602	015236'			PRINTB	#FORM54,PCBB	
3603	015236'	013746	000606'		MOV	PCBB,-(SP)
3604	015242'	012746	010116'		MOV	#FORM54,-(SP)
3605	015246'	012746	000002		MOV	#2,-(SP)
3606	015252'	010600			MOV	SP,RO
3607	015254'	104414			TRAP	C\$PNTB
3608	015256'	062706	000006		ADD	#6,SP
3609	015262'			ENDMSG		
3610	015262'				L10042:	
3611	015262'	104423			TRAP	C\$MSG
3612						
3613	015264'			BGNMSG	MSG30	
3614	015264'				MSG30::	
3615	015264'			PRINTB	#FORM55	
3616	015264'	012746	010144'		MOV	#FORM55,-(SP)
3617	015270'	012746	000001		MOV	#1,-(SP)
3618	015274'	010600			MOV	SP,RO
3619	015276'	104414			TRAP	C\$PNTB
3620	015300'	062706	000004		ADD	#4,SP
3621	015304'			PRINTB	#FORM57,PCBB	
3622	015304'	013746	000606'		MOV	PCBB,-(SP)
3623	015310'	012746	010270'		MOV	#FORM57,-(SP)
3624	015314'	012746	000002		MOV	#2,-(SP)
3625	015320'	010600			MOV	SP,RO
3626	015322'	104414			TRAP	C\$PNTB
3627	015324'	062706	000006		ADD	#6,SP
3628	015330'			ENDMSG		
3629	015330'				L10043:	
3630	015330'	104423			TRAP	C\$MSG

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 75  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3631							
3632	015332'	BGNMSG	MSG31				
3633	015332'					MSG31::	
3634	015332'		PRINTB	#FORM56			
3635	015332'	012746	010215'			MOV	#FORM56,-(SP)
3636	015336'	012746	000001			MOV	#1,-(SP)
3637	015342'	010600				MOV	SP,RO
3638	015344'	104414				TRAP	CSPNTB
3639	015346'	062706	000004			ADD	#4,SP
3640	015352'						
3641	015352'	010146				MOV	R1,-(SP)
3642	015354'	012746	010270'			MOV	#FORM57,-(SP)
3643	015360'	012746	000002			MOV	#2,-(SP)
3644	015364'	010600				MOV	SP,RO
3645	015366'	104414				TRAP	CSPNTB
3646	015370'	062706	000006			ADD	#6,SP
3647	015374'						
3648	015374'	012746	000150			MOV	#104,-(SP)
3649	015400'	012746	010340'			MOV	#FORM58,-(SP)
3650	015404'	012746	000002			MOV	#2,-(SP)
3651	015410'	010600				MOV	SP,RO
3652	015412'	104414				TRAP	CSPNTB
3653	015414'	062706	000006			ADD	#6,SP
3654	015420'						
3655	015420'	012746	100000			MOV	#100000,-(SP)
3656	015424'	012746	010416'			MOV	#FORM59,-(SP)
3657	015430'	012746	000002			MOV	#2,-(SP)
3658	015434'	010600				MOV	SP,RO
3659	015436'	104414				TRAP	CSPNTB
3660	015440'	062706	000006			ADD	#6,SP
3661	015444'						
3662	015444'	013746	000610'			MOV	PCBB+2,-(SP)
3663	015450'	012746	100000			MOV	#100000,-(SP)
3664	015454'	012746	010511'			MOV	#FORM60,-(SP)
3665	015460'	012746	000003			MOV	#3,-(SP)
3666	015464'	010600				MOV	SP,RO
3667	015466'	104414				TRAP	CSPNTB
3668	015470'	062706	000010			ADD	#10,SP
3669	015474'						
3670	015474'						
3671	015474'	104423				L10044:	TRAP
3672							CMSG
3673	015476'	BGNMSG	MSG32				
3674	015476'					MSG32::	
3675	015476'		PRINTB	#FORM61			
3676	015476'	012746	010576'			MOV	#FORM61,-(SP)
3677	015502'	012746	000001			MOV	#1,-(SP)
3678	015506'	010600				MOV	SP,RO
3679	015510'	104414				TRAP	CSPNTB
3680	015512'	062706	000004			ADD	#4,SP
3681	015516'						
3682	015516'	013746	000612'			MOV	PCBB+4,-(SP)
3683	015522'	012746	052525			MOV	#52525,-(SP)
3684	015526'	012746	006166'			MOV	#FORM27,-(SP)
3685	015532'	012746	000003			MOV	#3,-(SP)
3686	015536'	010600				MOV	SP,RO

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 76  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3687	015540'	104414			TRAP	CSPNTB
3688	015542'	062706	000010		ADD	#10,SP
3689	015546'			PRINTB	#FORM57,PCBB	
3690	015546'	013746	000606'		MOV	PCBB,-(SP)
3691	015552'	012746	010270'		MOV	#FORM57,-(SP)
3692	015556'	012746	000002		MOV	#2,-(SP)
3693	015562'	010600			MOV	SP,RO
3694	015564'	104414			TRAP	CSPNTB
3695	015566'	062706	000006		ADD	#6,SP
3696	015572'			ENDMSG		
3697	015572'				L10045:	
3698	015572'	104423			TRAP	CMSG
3699						
3700	015574'			BGNMSG	MSG33	
3701	015574'				MSG33::	
3702	015574'			PRINTB	#FORM62,2BITNAM	
3703	015574'	017746	162510		MOV	2BITNAM,-(SP)
3704	015600'	012746	010661'		MOV	#FORM62,-(SP)
3705	015604'	012746	000002		MOV	#2,-(SP)
3706	015610'	010600			MOV	SP,RO
3707	015612'	104414			TRAP	CSPNTB
3708	015614'	062706	000006		ADD	#6,SP
3709	015620'			ENDMSG		
3710	015620'				L10046:	
3711	015620'	104423			TRAP	CMSG
3712						
3713	015622'			BGNMSG	MSG34	
3714	015622'				MSG34::	
3715	015622'			PRINTB	#FORM63,R4	
3716	015622'	010446			MOV	R4,-(SP)
3717	015624'	012746	010734'		MOV	#FORM63,-(SP)
3718	015630'	012746	000002		MOV	#2,-(SP)
3719	015634'	010600			MOV	SP,RO
3720	015636'	104414			TRAP	CSPNTB
3721	015640'	062706	000006		ADD	#6,SP
3722	015644'			PRINTB	#FORM28,STMSG	
3723	015644'	013746	024406'		MOV	STMSG,-(SP)
3724	015650'	012746	006236'		MOV	#FORM28,-(SP)
3725	015654'	012746	000002		MOV	#2,-(SP)
3726	015660'	010600			MOV	SP,RO
3727	015662'	104414			TRAP	CSPNTB
3728	015664'	062706	000006		ADD	#6,SP
3729	015670'			ENDMSG		
3730	015670'				L10047:	
3731	015670'	104423			TRAP	CMSG
3732						
3733	015672'			BGNMSG	MSG35	
3734	015672'				MSG35::	
3735	015672'			PRINTB	#FORM59,PCBB	
3736	015672'	013746	000606'		MOV	PCBB,-(SP)
3737	015676'	012746	010416'		MOV	#FORM59,-(SP)
3738	015702'	012746	000002		MOV	#2,-(SP)
3739	015706'	010600			MOV	SP,RO
3740	015710'	104414			TRAP	CSPNTB
3741	015712'	062706	000006		ADD	#6,SP
3742	015716'			PRINTB	#FORM60	

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 77  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3743	015716'	012746	010511'		MOV	#FORM60,-(SP)
3744	015722'	012746	000001		MOV	#1,-(SP)
3745	015726'	010600			MOV	SP,RO
3746	015730'	104414			TRAP	C\$PNTB
3747	015732'	062706	000004		ADD	#4,SP
3748	015736'			PRINTB	#FORM78,PCBB,PCBB+2	
3749	015736'	013746	000610'		MOV	PCBB+2,-(SP)
3750	015742'	013746	000606'		MOV	PCBB,-(SP)
3751	015746'	012746	012132'		MOV	#FORM78,-(SP)
3752	015752'	012746	000003		MOV	#3,-(SP)
3753	015756'	010600			MOV	SP,RO
3754	015760'	104414			TRAP	C\$PNTB
3755	015762'	062706	000010		ADD	#10,SP
3756	015766'			ENDMSG		
3757	015766'					
3758	015766'	104423			L10050:	TRAP
3759						C\$MSG
3760	015770'			BGNMSG	MSG36	
3761	015770'					
3762	015770'			PRINTB	#FORM66	MSG36::
3763	015770'	012746	011072'			
3764	015774'	012746	000001		MOV	#FORM66,-(SP)
3765	016000'	010600			MOV	#1,-(SP)
3766	016002'	104414			MOV	SP,RO
3767	016004'	062706	000004		TRAP	C\$PNTB
3768	016010'			ENDMSG	ADD	#4,SP
3769	016010'					
3770	016010'	104423			L10051:	TRAP
3771						C\$MSG
3772	016012'			BGNMSG	MSG37	
3773	016012'					
3774	016012'			PRINTB	#FORM67	MSG37::
3775	016012'	012746	011157'			
3776	016016'	012746	000001		MOV	#FORM67,-(SP)
3777	016022'	010600			MOV	#1,-(SP)
3778	016024'	104414			MOV	SP,RO
3779	016026'	062706	000004		TRAP	C\$PNTB
3780	016032'			ENDMSG	ADD	#4,SP
3781	016032'					
3782	016032'	104423			L10052:	TRAP
3783						C\$MSG
3784	016034'			BGNMSG	MSG38	
3785	016034'					
3786	016034'			PRINTB	#FORM68	MSG38::
3787	016034'	012746	011243'			
3788	016040'	012746	000001		MOV	#FORM68,-(SP)
3789	016044'	010600			MOV	#1,-(SP)
3790	016046'	104414			MOV	SP,RO
3791	016050'	062706	000004		TRAP	C\$PNTB
3792	016054'			ENDMSG	ADD	#4,SP
3793	016054'					
3794	016054'	104423			L10053:	TRAP
3795						C\$MSG
3796	016056'			BGNMSG	MSG39	
3797	016056'					
3798	016056'			PRINTB	#FORM69	MSG39::



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 78  
 CZUAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3799 016056' 012746 011340'  
 3800 016062' 012746 000001  
 3801 016066' 010600  
 3802 016070' 104414  
 3803 016072' 062706 000004  
 3804 016076'  
 3805 016076'  
 3806 016076' 104423  
 3807  
 3808 016100'  
 3809 016100'  
 3810 016100'  
 3811 016100' 012746 010215'  
 3812 016104' 012746 000001  
 3813 016110' 010600  
 3814 016112' 104414  
 3815 016114' 062706 000004  
 3816 016120'  
 3817 016120' 012746 100000  
 3818 016124' 012746 010416'  
 3819 016130' 012746 000002  
 3820 016134' 010600  
 3821 016136' 104414  
 3822 016140' 062706 000006  
 3823 016144'  
 3824 016144' 013746 000606'  
 3825 016150' 012746 100000  
 3826 016154' 012746 010511'  
 3827 016160' 012746 000003  
 3828 016164' 010600  
 3829 016166' 104414  
 3830 016170' 062706 000010  
 3831 016174'  
 3832 016174'  
 3833 016174' 104423  
 3834  
 3835 016176'  
 3836 016176'  
 3837 016176'  
 3838 016176' 012746 010576'  
 3839 016202' 012746 000001  
 3840 016206' 010600  
 3841 016210' 104414  
 3842 016212' 062706 000004  
 3843 016216'  
 3844 016216' 013746 000610'  
 3845 016222' 012746 052525  
 3846 016226' 012746 006166'  
 3847 016232' 012746 000003  
 3848 016236' 010600  
 3849 016240' 104414  
 3850 016242' 062706 000010  
 3851 016246'  
 3852 016246'  
 3853 016246' 104423  
 3854

ENDMSG

BGNMSG MSG40

PRINTB #FORM56

PRINTB #FORM59,#100000

PRINTB #FORM60,#100000,PCBB

ENDMSG

BGNMSG MSG41

PRINTB #FORM61

PRINTB #FORM27,#52525,PCBB+2

ENDMSG

MOV #FORM69,-(SP)  
 MOV #1,-(SP)  
 MOV SP,R0  
 TRAP C\$PNTB  
 ADD #4,SP

L10054: TRAP C\$MSG

MSG40::

MOV #FORM56,-(SP)  
 MOV #1,-(SP)  
 MOV SP,R0  
 TRAP C\$PNTB  
 ADD #4,SP

MOV #100000,-(SP)  
 MOV #FORM59,-(SP)  
 MOV #2,-(SP)  
 MOV SP,R0  
 TRAP C\$PNTB  
 ADD #6,SP

MOV PCBB,-(SP)  
 MOV #100000,-(SP)  
 MOV #FORM60,-(SP)  
 MOV #3,-(SP)  
 MOV SP,R0  
 TRAP C\$PNTB  
 ADD #10,SP

L10055: TRAP C\$MSG

MSG41::

MOV #FORM61,-(SP)  
 MOV #1,-(SP)  
 MOV SP,R0  
 TRAP C\$PNTB  
 ADD #4,SP

MOV PCBB+2,-(SP)  
 MOV #52525,-(SP)  
 MOV #FORM27,-(SP)  
 MOV #3,-(SP)  
 MOV SP,R0  
 TRAP C\$PNTB  
 ADD #10,SP

L10056: TRAP C\$MSG

62GLOBAL AREAS MACY11 30A(1052)  
CZUAAB.MAC 07-APR-83 17:03

07-APR-83 17:13 PAGE 79  
GLOBAL ERROR REPORT SECTION

3855 016250'  
3856 016250'  
3857 016250'  
3858 016250' 013746 000610'  
3859 016254' 012746 011456'  
3860 016260' 012746 000002  
3861 016264' 010600  
3862 016266' 104414  
3863 016270' 062706 000006  
3864 016274'  
3865 016274' 013746 000606'  
3866 016300' 012746 011520'  
3867 016304' 012746 000002  
3868 016310' 010600  
3869 016312' 104414  
3870 016314' 062706 000006  
3871 016320'  
3872 016320' 010146  
3873 016322' 012746 006441'  
3874 016326' 012746 000002  
3875 016332' 010600  
3876 016334' 104414  
3877 016336' 062706 000006  
3878 016342'  
3879 016342' 013746 000612'  
3880 016346' 012746 006662'  
3881 016352' 012746 000002  
3882 016356' 010600  
3883 016360' 104414  
3884 016362' 062706 000006  
3885 016366'  
3886 016366' 013746 000616'  
3887 016372' 013746 000614'  
3888 016376' 012746 006166'  
3889 016402' 012746 000003  
3890 016406' 010600  
3891 016410' 104414  
3892 016412' 062706 000010  
3893 016416'  
3894 016416'  
3895 016416' 104423  
3896  
3897 016420'  
3898 016420'  
3899 016420'  
3900 016420' 012746 011562'  
3901 016424' 012746 000001  
3902 016430' 010600  
3903 016432' 104414  
3904 016434' 062706 000004  
3905 016440'  
3906 016440'  
3907 016440' 104423  
3908  
3909 016442'  
3910 016442'

BGNMSG MSG42  
PRINTB #FORM70,PCBB+2  
PRINTB #FORM71,PCBB  
PRINTB #FORM33,R1  
PRINTB #FORM36,PCBB+4  
PRINTB #FORM27,PCBB+6,PCBB+10  
ENDMSG  
BGNMSG MSG43  
PRINTB #FORM72  
ENDMSG  
BGNMSG MSG44

MSG42::

MOV PCBB+2,-(SP)  
MOV #FORM70,-(SP)  
MOV #2,-(SP)  
MOV SP,R0  
TRAP C\$PNTB  
ADD #6,SP  
MOV PCBB,-(SP)  
MOV #FORM71,-(SP)  
MOV #2,-(SP)  
MOV SP,R0  
TRAP C\$PNTB  
ADD #6,SP  
MOV R1,-(SP)  
MOV #FORM33,-(SP)  
MOV #2,-(SP)  
MOV SP,R0  
TRAP C\$PNTB  
ADD #6,SP  
MOV PCBB+4,-(SP)  
MOV #FORM36,-(SP)  
MOV #2,-(SP)  
MOV SP,R0  
TRAP C\$PNTB  
ADD #6,SP  
MOV PCBB+10,-(SP)  
MOV PCBB+6,-(SP)  
MOV #FORM27,-(SP)  
MOV #3,-(SP)  
MOV SP,R0  
TRAP C\$PNTB  
ADD #10,SP

L10057:

TRAP C\$MSG

MSG43::

MOV #FORM72,-(SP)  
MOV #1,-(SP)  
MOV SP,R0  
TRAP C\$PNTB  
ADD #4,SP

L10060:

TRAP C\$MSG

MSG44::

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 80  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

```

3911 016442' 032737 100000 000670' BIT #NPRERR,ERRINT ;NPR ERROR OCCUR?
3912 016450' 001410 BEQ 10$ ;NO
3913 016452' PRINTB #FORM73 ;YES, PRINT NPR ERROR MESSAGE
3914 016452' 012746 011634' MOV #FORM73,-(SP)
3915 016456' 012746 000001 MOV #1,-(SP)
3916 016462' 010600 MOV SP,RO
3917 016464' 104414 TRAP C$PNTB
3918 016466' 062706 000004 ADD #4,SP
3919 016472' 032737 040000 000670' 10$: BIT #NXMERR,ERRINT ;NON-EXISTANT MEMORY OCCUR?
3920 016500' 001410 BEQ 20$ ;NO
3921 016502' PRINTB #FORM74 ;YES, PRINT NON-EXISTANT MEMORY MESSAGE
3922 016502' 012746 011677' MOV #FORM74,-(SP)
3923 016506' 012746 000001 MOV #1,-(SP)
3924 016512' 010600 MOV SP,RO
3925 016514' 104414 TRAP C$PNTB
3926 016516' 062706 000004 ADD #4,SP
3927 016522' 032737 020000 000670' 20$: BIT #UNIERR,ERRINT ;UNEXPECTED INTERRUPT OCCUR?
3928 016530' 001410 BEQ 30$ ;NO
3929 016532' PRINTB #FORM75 ;YES, PRINT UNEXPECTED INTERRUPT MESSAGE
3930 016532' 012746 011754' MOV #FORM75,-(SP)
3931 016536' 012746 000001 MOV #1,-(SP)
3932 016542' 010600 MOV SP,RO
3933 016544' 104414 TRAP C$PNTB
3934 016546' 062706 000004 ADD #4,SP
3935 016552' 032737 010000 000670' 30$: BIT #PARERR,ERRINT ;PARITY ERROR OCCUR?
3936 016560' 001410 BEQ 40$ ;NO
3937 016562' PRINTB #FORM72 ;YES, PRINT PARITY ERROR MESSAGE
3938 016562' 012746 011562' MOV #FORM72,-(SP)
3939 016566' 012746 000001 MOV #1,-(SP)
3940 016572' 010600 MOV SP,RO
3941 016574' 104414 TRAP C$PNTB
3942 016576' 062706 000004 ADD #4,SP
3943 016602' 40$:
3944 016602' ENDMMSG
3945 016602'
3946 016602' 104423 L10061: TRAP C$MSG
3947
3948 016604' BGNMSG MSG45
3949 016604'
3950 016604' PRINTB #FORM79 MSG45::
3951 016604' 012746 012172' MOV #FORM79,-(SP)
3952 016610' 012746 000001 MOV #1,-(SP)
3953 016614' 010600 MOV SP,RO
3954 016616' 104414 TRAP C$PNTB
3955 016620' 062706 000004 ADD #4,SP
3956 016624' PRINTB #FORM80
3957 016624' 012746 012241' MOV #FORM80,-(SP)
3958 016630' 012746 000001 MOV #1,-(SP)
3959 016634' 010600 MOV SP,RO
3960 016636' 104414 TRAP C$PNTB
3961 016640' 062706 000004 ADD #4,SP
3962 016644' PRINTB #FORM81
3963 016644' 012746 012320' MOV #FORM81,-(SP)
3964 016650' 012746 000001 MOV #1,-(SP)
3965 016654' 010600 MOV SP,RO
3966 016656' 104414 TRAP C$PNTB

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 81  
 CZUAAB.MAC 07-APR-83 17:03 GLOBAL ERROR REPORT SECTION

3967	016660'	062706	000004				
3968	016664'			ENDMSG		ADD	#4,SP
3969	016664'						
3970	016664'	104423			L10062:	TRAP	C8MSG
3971							
3972	016666'			BGNMSG	MSG46		
3973	016666'					MSG46::	
3974	016666'				PRINTB	#FORMB2	
3975	016666'	012746	012363'			MOV	#FORMB2,-(SP)
3976	016672'	012746	000001			MOV	#1,-(SP)
3977	016676'	010600				MOV	SP,R0
3978	016700'	104414				TRAP	C8PNTB
3979	016702'	062706	000004			ADD	#4,SP
3980	016706'			ENDMSG			
3981	016706'						
3982	016706'	104423			L10063:	TRAP	C8MSG

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 82  
 CZUAAAB.MAC 07-APR-83 17:03 GLOBAL SUBROUTINES SECTION

```

3983      .SBTTL GLOBAL SUBROUTINES SECTION
3984
3985      :++
3986      : THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
3987      : THAT ARE USED IN MORE THAN ONE TEST.
3988      :--
3989      .SBTTL 32 BIT CRC CALCULATOR
3990
3991      :++
3992      : FUNCTIONAL DESCRIPTION:
3993      : SUBROUTINE TO CALCULATE A 32 BIT CRC ON A BLOCK OF DATA
3994
3995      : INPUTS:
3996      : R0 = ADDRESS OF DATA BLOCK
3997      : R2 = BYTE COUNT
3998
3999      : IMPLICIT INPUTS: NONE
4000
4001      : OUTPUTS:
4002      : R4 = CRC HIGH WORD
4003      : R5 = CRC LOW WORD
4004
4005      : SUBORDINATE ROUTINES USED: GETCRC
4006
4007      : FUNCTIONAL SIDE EFFECTS: NONE
4008
4009      : CALLING SEQUENCE: PUT ADDRESS OF DATA TO PERFORM CRC ON IN R0
4010      : PUT NUMBER OF BYTES TO CHECK IN R2
4011      : JSR PC,BLKCRC
4012
4013      :--
4014
4015      016710'
4016      016710' 010046
4017      016712' 010146
4018      016714' 010246
4019      016716' 010346
4020      016720' 012704 177777
4021      016724' 012705 177777
4022      016730' 112001
4023      016732' 004737 016752'
4024      016736' 077204
4025      016740' 012603
4026      016742' 012602
4027      016744' 012601
4028      016746' 012600
4029      016750' 000207
4030
4031
  
```

```

CRC32::
      MOV     R0,-(SP)      ;SAVE REGISTERS 0-3
      MOV     R1,-(SP)
      MOV     R2,-(SP)
      MOV     R3,-(SP)
      MOV     #INITH,R4    ;INITIAL CRC HIGH WORD
      MOV     #INITL,R5    ;INITIAL CRC LOW WORD
10$:   MOVB   (R0)+,R1      ;GET NEXT BYTE OF DATA
      JSR    PC,GETCRC     ;CALCULATE THE CRC
      SOB   R2,10$        ;LOOP TILL DONE
      MOV   (SP)+,R3      ;RESTORE REGISTERS
      MOV   (SP)+,R2
      MOV   (SP)+,R1
      MOV   (SP)+,R0
      RTS    PC           ;RETURN
  
```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 83  
 CZUAAB.MAC 07-APR-83 17:03 32 BIT CRC CALCULATOR

4032  
 4033  
 4034  
 4035  
 4036  
 4037  
 4038  
 4039  
 4040  
 4041  
 4042  
 4043  
 4044  
 4045  
 4046  
 4047  
 4048  
 4049  
 4050  
 4051  
 4052  
 4053  
 4054  
 4055  
 4056  
 4057  
 4058  
 4059  
 4060  
 4061  
 4062  
 4063  
 4064  
 4065

016752'  
 016752' 010146  
 016754' 010246  
 016756' 010346  
 016760' 042701 177400  
 016764' 074105  
 016766' 012702 166670  
 016772' 012703 101440  
 016776' 012701 000010  
 017002' 000241  
 017004' 006004  
 017006' 006005  
 017010' 103002  
 017012' 074204  
 017014' 074305  
 017016' 077107  
 017020' 012603  
 017022' 012602  
 017024' 012601  
 017026' 000207

```

:++
:BYTE WISE 32-BIT CRC CALCULATOR
:
:INPUTS:
:       R1 = NEW BYTE TO ADD TO CRC
:       R4,R5 = CURRENT PARTIAL CRC CODE
:
:OUTPUTS:
:       R4,R5 = UPDATED CRC
:
:NOTE: THIS ROUTINE IS ONLY USED BY BLKCRC
:
:--
  
```

```

GETCRC:
      MOV     R1,-(SP)      ;SAVE R1-3
      MOV     R2,-(SP)
      MOV     R3,-(SP)
      BIC     #^C377,R1    ;CLEAR HIGH BYTE
      XOR     R1,R5        ;MERGE NEW BYTE WITH OLD CRC
      MOV     #POLYH,R2    ;GET CRC POLYNOMIAL HIGH WORD
      MOV     #POLYL,R3    ;GET CRC POLYNOMIAL LOW WORD
      MOV     #8,R1        ;LOOP COUNT
1$:   CLC                ;CLEAR THE CARRY
      ROR     R4            ;SHIFT RIGHT THE CRC
      ROR     R5            ;32 BITS WORTH
      BCC     2$           ;SKIP IF BIT 0 NOT SET
      XOR     R2,R4        ;EXCLUSIVE OR IN THE POLY
      XOR     R3,R5        ;BOTH HIGH AND LOW WORDS
2$:   SOB     R1,1$        ;AND LOOP ON ALL 8 BITS
      MOV     (SP)+,R3     ;RESTORE REGISTERS
      MOV     (SP)+,R2
      MOV     (SP)+,R1
      RTS     PC           ;RETURN
  
```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 84  
CZJAAB.MAC 07-APR-83 17:03 32 BIT CRC CALCULATOR

4066  
4067  
4068  
4069  
4070  
4071  
4072  
4073  
4074  
4075  
4076  
4077  
4078  
4079  
4080  
4081  
4082  
4083  
4084  
4085  
4086  
4087  
4088  
4089  
4090  
4091  
4092  
4093  
4094  
4095  
4096  
4097  
4098  
4099  
4100  
4101  
4102  
4103  
4104  
4105  
4106  
4107  
4108  
4109  
4110  
4111  
4112

017030'  
017030' 010046  
017032' 010146  
017034' 010246  
017036' 010346  
017040' 010546  
017042' 010203  
017044' 010002  
017046' 012705 120001  
017052' 112200  
017054' 042700 177400  
017060' 074004  
017062' 012701 000010  
017066' 000241  
017070' 006004  
017072' 103001  
017074' 074504  
017076' 077105  
017100' 077314  
017102' 012605  
017104' 012603  
017106' 012602  
017110' 012601  
017112' 012600  
017114' 000207

.SBTTL 16 BIT CRC CALCULATOR

```
.....
SUBROUTINE - CRC16

THIS SUBROUTINE CALCULATES A 16 BIT CRC
ON A BLOCK OF DATA.

INPUTS:      R0 = ADDRESS OF DATA BLOCK
              R2 = BYTE COUNT
              R4 = INITIAL CRC VALUE

OUTPUTS:     R4 = CRC

CALLING SEQUENCE:
              JSR   PC,CRC16
.....
```

```
CRC16::
MOV   R0,-(SP)      ; SAVE R0
MOV   R1,-(SP)      ; SAVE R1
MOV   R2,-(SP)      ; SAVE R2
MOV   R3,-(SP)      ; SAVE R3
MOV   R5,-(SP)      ; SAVE R5
MOV   R2,R3         ; R3 = BYTE COUNT
MOV   R0,R2         ; R2 = ADDRESS OF DATA BLOCK
MOV   #POLY16,R5    ; CRC POLYNOMIAL
1$:   MOVB  (R2)+,R0 ; GET NEXT BYTE
      BIC  #^C377,R0 ; CLEAR HIGH BYTE
      XOR  R0,R4     ; MERGE BYTE WITH OLD CRC
      MOV  #8.,R1    ; LOOP COUNT
2$:   CLC           ; CLEAR CARRY
      ROR  R4       ; SHIFT RIGHT THE CRC
      BCC  3$       ; SKIP IF BIT ZERO NOT SET
      XOR  R5,R4    ; EXCLUSIVE OR IN THE POLY
3$:   SOB  R1,2$    ; AND LOOP ON ALL 8 BITS
      SOB  R3,1$
      MOV  (SP)+,R5 ; RESTORE R5
      MOV  (SP)+,R3 ; RESTORE R3
      MOV  (SP)+,R2 ; RESTORE R2
      MOV  (SP)+,R1 ; RESTORE R1
      MOV  (SP)+,R0 ; RESTORE R0
      RTS  PC       ; AND RETURN
```

62GLOBAL APEAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 85  
 CZUAAB.MAC 07-APR-83 17:03 16 BIT CRC CALCULATOR

4113  
 4114  
 4115  
 4116  
 4117  
 4118  
 4119  
 4120  
 4121  
 4122  
 4123  
 4124  
 4125  
 4126  
 4127  
 4128  
 4129  
 4130  
 4131  
 4132  
 4133  
 4134  
 4135  
 4136  
 4137  
 4138  
 4139  
 4140  
 4141  
 4142  
 4143  
 4144  
 4145  
 4146  
 4147  
 4148  
 4149  
 4150  
 4151  
 4152  
 4153  
 4154  
 4155

017116'  
 017116' 010046  
 017120' 010346  
 017122' 010546  
 017124' 012700 000006  
 017130' 012703 052625'  
 017134' 012705 052542'  
 017140' 112537 052560'  
 017144' 004737 017200'  
 017150' 113723 052561'  
 017154' 004737 017236'  
 017160' 113723 052561'  
 017164' 105723  
 017166' 077014  
 017170' 012605  
 017172' 012603  
 017174' 012600  
 017176' 00020'

.SBTTL HEXIDECIMAL CONVERTER FOR DEFAULT PHYSICAL ADDRESS

```

*****
SUBROUTINE - HEXDPA

THIS SUBROUTINE LOADS DEFADR WITH THE ASCII HEX VALUE
FOR THE DEFAULT PHYSICAL ADDRESS DPA.

INPUTS:          NONE

IMPLICIT
INPUTS:          DPA = DEFAULT PHYSICAL ADDRESS

OUTPUTS:         DEFADR = ASCII HEX VALUE FOR DPA

CALLING SEQUENCE:
                JSR    PC,HEXDPA
*****
    
```

```

HEXDPA::
        MOV    R0,-(SP)          ; SAVE R0
        MOV    R3,-(SP)          ; SAVE R3
        MOV    R5,-(SP)          ; SAVE R5
;
        MOV    #6,R0             ; DO LOOP = 6 BYTES
        MOV    #DEFADR,R3        ; POINT TO ASCII MESSAGE
        MOV    #DPA,R5           ; POINT TO DEFAULT PHYSICAL ADDR
;
10$:    MOVB   (R5)+,HEXDAT       ; LOAD BYTE FOR CONVERSION
        JSR    PC,HEXH           ; CONVERT HIGH NIBBLE
        MOVB   HEXVAL,(R3)+      ; LOAD INTO ASCII MESSAGE
        JSR    PC,HEXL           ; CONVERT LOW NIBBLE
        MOVB   HEXVAL,(R3)+      ; LOAD INTO ASCII MESSAGE
        TSTB   (R3)+             ; SKIP OVER HYPHEN IN MESSAGE
        SOB    R0,10$           ; LOOP TILL ALL 6 BYTES ARE DONE
;
        MOV    (SP)+,R5          ; RESTORE R5
        MOV    (SP)+,R3          ; RESTORE R3
        MOV    (SP)+,R0          ; RESTORE R0
        RTS    PC                ; AND RETURN
    
```



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 86  
CZUAAB.MAC 07-APR-83 17:03 HEXIDECIMAL CONVERTER FOR DEFAULT PHYSICAL ADDRESS

4156  
4157  
4158  
4159  
4160  
4161  
4162  
4163  
4164  
4165  
4166  
4167  
4168  
4169  
4170  
4171  
4172  
4173  
4174  
4175  
4176  
4177  
4178  
4179  
4180  
4181  
4182  
4183  
4184  
4185  
4186  
4187  
4188  
4189  
4190  
4191

017200'  
017200' 010146  
017202' 013701 052560'  
017206' 042701 177417  
017212' 006201  
017214' 006201  
017216' 006201  
017220' 006201  
017222' 062701 052651'  
017226' 111137 052561'  
017232' 012601  
017234' 000207

.....  
: SUBROUTINE - HEXH  
: THIS SUBROUTINE LOADS HEXVAL WITH THE ASCII HEX VALUE  
: FOR THE HIGH NIBBLE IN HEXDAT  
: INPUTS: NONE  
: IMPLICIT  
: INPUTS: HEXDAT = BYTE TO BE CONVERTED  
: OUTPUTS: HEXVAL = ASCII HEX VALUE FOR THE HIGH NIBBLE  
: CALLING SEQUENCE:  
: JSR PC,HEXH  
: .....

HEXH::  
: MOV R1,-(SP) ; SAVE R1  
: MOV HEXDAT,R1 ; LOAD DATA FOR CONVERSION  
: BIC #177417,R1 ; MASK HIGH NIBBLE  
: ASR R1 ; SHIFT RIGHT  
: ASR R1  
: ASR R1  
: ASR R1  
: ADD #HEXTBL,R1 ; GET INDEX INTO HEXTBL  
: MOVB (R1),HEXVAL ; AND LOAD HEXVAL  
: MOV (SP)+,R1 ; RESTORE R1  
: RTS PC ; AND RETURN

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 87  
CZUAAB.MAC 07-APR-83 17:03

HEXIDECIMAL CONVERTER FOR DEFAULT PHYSICAL ADDRESS

4192  
4193  
4194  
4195  
4196  
4197  
4198  
4199  
4200  
4201  
4202  
4203  
4204  
4205  
4206  
4207  
4208  
4209  
4210  
4211  
4212 017236'  
4213 017236' 010146  
4214  
4215 017240' 013701 052560'  
4216 017244' 042701 177760  
4217  
4218 017250' 062701 052651'  
4219 017254' 111137 052561'  
4220  
4221 017260' 012601  
4222 017262' 000207

```

*****
SUBROUTINE - HEXL
THIS SUBROUTINE LOADS HEXVAL WITH THE ASCII HEX VALUE
FOR THE LOW NIBBLE IN HEXDAT
INPUTS:          NONE
IMPLICIT
INPUTS:          HEXDAT = BYTE TO BE CONVERTED
OUTPUTS:         HEXVAL = ASCII HEX VALUE FOR THE LOW NIBBLE
CALLING SEQUENCE:
                JSR      PC,HEXL
*****

```

```

HEXL::
MOV      R1,-(SP)          ; SAVE R1
:
MOV      HEXDAT,R1        ; LOAD DATA FOR CONVERSION
BIC      #177760,R1       ; MASK LOW NIBBLE
:
ADD      #HEXTBL,R1       ; GET INDEX INTO HEXTBL
MOVB    (R1),HEXVAL      ; AND LOAD HEXVAL
:
MOV      (SP)+,R1         ; RESTORE R1
RTS      PC               ; AND RETURN

```

HEXIDECIMAL CONVERTER FOR DEFAULT PHYSICAL ADDRESS

```

4223
4224 .SBTTL TURN ON THE CLOCK
4225
4226 :*****
4227 :
4228 :THIS ROUTINE TURNS ON THE CLOCK
4229 :
4230 :*****
4231
4232 017264' TIMON:: SETPRI #PRI05 ;SET PROCESSOR PRIORITY TO 5
4233 017264' 012700 000240 ;
4234 017270' 104441 ;
4235 017272' 012777 000100 160774 MOV #IE,@CLKCSR ;ENABLE CLOCK INTERRUPTS
4236 017300' 000207 RTS PC ;
4237
4238
4239 :*****
4240 :
4241 :THIS ROUTINE TURNS THE CLOCK OFF
4242 :
4243 :*****
4244
4245 017302' 005077 160766 TIMOFF:: CLR @CLKCSR ;CLEAR INTERRUPT ENABLE
4246 017306' SETPRI #PRI07 ;PUT PRIORITY BACK UP
4247 017306' 012700 000340 ;
4248 017312' 104441 ;
4249 017314' 000207 RTS PC ;

```

```

4250 .SBTTL CHECK FOR DONE INTERRUPT
4251
4252 :*****
4253 :
4254 :FUNCTIONAL DESCRIPTION:
4255 :ROUTINE TO WAIT FOR THE 'DONE INTERRUPT' BIT TO SET IN PCSRO
4256 :
4257 :INPUTS: NONE
4258 :
4259 :IMPLICIT INPUTS: METER
4260 :
4261 :OUTPUTS: C BIT SET IN PSW IF 'DNI' NOT SET
4262 :          C BIT CLEAR IN PSW IF 'DNI' SET
4263 :
4264 :SUBORDINATE ROUTINES USED: TIMON, TIMOFF
4265 :
4266 :FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK INTERRUPTS ARE ENABLED
4267 :
4268 :CALLING SEQUENCE: PUT NUMBER OF CLOCK TICKS TO WAIT FOR IN--->METER
4269 :                   JSR PC,CHKDNI
4270 :
4271 :*****
4272

```

```

4273 017316' 004737 017264' CHKDNI::JSR PC,TIMON :TURN ON THE LINE CLOCK
4274 017322' 032777 004000 161006 10$: BIT #DNI,BPCSRO :IS 'DNI' SET?
4275 017330' 001010 :BNE 20$ :YES
4276 017332' :BREAK :RETURN TO THE DRS FOR A MOMENT
4277 017332' 104422 :TRAP CSBRK
4278 017334' 005737 000332' TST METER :HAS THE TIME EXPIRED?
4279 017340' 001370 :BNE 10$ :NO, KEEP WAITING
4280 017342' 004737 017302' JSR PC,TIMOFF :TURN THE CLOCK OFF
4281 017346' 000261 SEC :INDICATE 'DNI' DID NOT SET
4282 017350' 000403 BR 30$ :LEAVE
4283 017352' 004737 017302' 20$: JSR PC,TIMOFF :STOP THE CLOCK
4284 017356' 000241 CLC :INDICATE 'DNI' SET
4285 017360' 000207 30$: RTS PC
4286

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 90  
CZUAAB.MAC 07-APR-83 17:03 CLEAR DONE INTERRUPT

4287  
4288  
4289  
4290  
4291  
4292  
4293  
4294  
4295  
4296  
4297  
4298  
4299  
4300  
4301  
4302  
4303  
4304  
4305  
4306  
4307  
4308  
4309  
4310  
4311  
4312  
4313  
4314

.SBTTL CLEAR DONE INTERRUPT  
:.....  
:FUNCTIONAL DESCRIPTION:  
:ROUTINE TO CLEAR THE 'DNI' BIT IN PCSRO  
:INPUTS: NONE  
:IMPLICIT INPUTS: PCSRO  
:OUTPUTS: C BIT SET IN PCSW IF 'DNI' WILL NOT CLEAR  
:C BIT CLEAR IN PSW IF 'DNI' CLEARED SUCCESSFULLY  
:SUBORDINATE ROUTINES CALLED: NONE  
:FUNCTIONAL SIDE EFFECTS: PSW CHANGED  
:CALLING SEQUENCE: JSR PC,CLRDNI  
:.....

017362 112777 000010 160756  
017370 032777 004000 160740  
017376 001402  
017400 000261  
017402 000401  
017404 000241  
017406 000207

CLRDNI::MOVB #DNI,B,#PCSROUB ;CLEAR 'DNI' BIT ;RSJ001  
BIT #DNI,#PCSRO ;DID IS CLEAR?  
BEQ 10\$ ;YES  
SEC ;NO, INDICATE ERROR  
BR 20\$  
10\$: CLC ;YES, INDICATE SUCCESS  
20\$: RTS PC

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 91  
CZUAAB.MAC 07-APR-83 17:03 CLEAR OUTSTANDING INTERRUPT BITS

```

4315 .SBTTL CLEAR OUTSTANDING INTERRUPT BITS
4316
4317 :*****8
4318
4319 :FUNCTIONAL DESCRIPTION:
4320 :ROUTINE TO CLEAR ALL INTERRUPT BITS IN PCSRO
4321
4322 :INPUTS: NONE
4323
4324 :IMPLICIT INPUTS: NONE
4325
4326 :OUTPUTS: C BIT SET IF UNABLE TO CLEAR A INTERRUPT BIT
4327 :C BIT CLEARED IF SUCCESSFUL
4328
4329 :SUBORDINATE ROUTINES USED: NONE
4330
4331 :FUNCTIONAL SIDE EFFECTS: ANY OUTSTANDING INTERRUPT IS CLEARED
4332
4333 :CALLING SEQUENCE: JSR PC,CLRINT
4334
4335 :*****
4336

```

```

4337 017410' 032777 100000 160720 CLRINT: BIT #SERI,OPCSRO ;IS 'SERI' BIT SET?
4338 017416' 001413 BEQ 10$ ;NO
4339 017420' 112777 000200 160720 MOVB #SERIB,OPCSROUB ;WRITE ONE TO CLEAR 'SERI' ;RSJ001
4340 017426' 032777 100000 160702 BIT #SERI,OPCSRO ;DID IT CLEAR?
4341 017434' 001404 BEQ 10$ ;YES
4342 017436' 012737 000736' 000310' MOV #SERI,BITNAM ;NO, GET POINTER TO BIT NAME STRING
4343 017444' 000531 BR 70$ ;LEAVE
4344 017446' 032777 040000 160662 10$: BIT #PCEI,OPCSRO ;IS 'PCEI' BIT SET?
4345 017454' 001413 BEQ 20$ ;NO
4346 017456' 112777 000100 160662 MOVB #PCEIB,OPCSROUB ;YES, WRITE ONE TO CLEAR 'PCEI' ;RSJ001
4347 017464' 032777 040000 160644 BIT #PCEI,OPCSRO ;DID IT CLEAR?
4348 017472' 001404 BEQ 20$ ;YES
4349 017474' 012737 000747' 000310' MOV #PCEI,BITNAM ;NO, GET POINTER TO BIT NAME STRING
4350 017502' 000512 BR 70$ ;LEAVE
4351 017504' 032777 020000 160624 20$: BIT #RXI,OPCSRO ;IS 'RXI' BIT SET?
4352 017512' 001413 BEQ 30$ ;NO
4353 017514' 112777 000040 160624 MOVB #RXIB,OPCSROUB ;YES, WRITE ONE TO CLEAR 'RXI' ;RSJ001
4354 017522' 032777 020000 160606 BIT #RXI,OPCSRO ;DID IT CLEAR?
4355 017530' 001404 BEQ 30$ ;YES
4356 017532' 012737 000760' 000310' MOV #RXI,BITNAM ;NO, GET POINTER TO BIT NAME STRING
4357 017540' 000473 BR 70$ ;LEAVE
4358 017542' 032777 010000 160566 30$: BIT #TXI,OPCSRO ;IS 'TXI' BIT SET?
4359 017550' 001413 BEQ 40$ ;NO
4360 017552' 112777 000020 160566 MOVB #TXIB,OPCSROUB ;YES, WRITE ONE TO CLEAR 'TXI' ;RSJ001
4361 017560' 032777 010000 160550 BIT #TXI,OPCSRO ;DID IT CLEAR?
4362 017566' 001404 BEQ 40$ ;YES
4363 017570' 012737 000770' 000310' MOV #TXI,BITNAM ;NO, GET POINTER TO BIT NAME STRING
4364 017576' 000454 BR 70$ ;LEAVE
4365 017600' 032777 004000 160530 40$: BIT #DNI,OPCSRO ;IS 'DNI' BIT SET?
4366 017606' 001413 BEQ 50$ ;NO
4367 017610' 112777 000010 160530 MOVB #DNI,OPCSROUB ;YES, WRITE ONE TO CLEAR 'DNI' ;RSJ001
4368 017616' 032777 004000 160512 BIT #DNI,OPCSRO ;DID IT CLEAR?
4369 017624' 001404 BEQ 50$ ;YES
4370 017626' 012737 001000' 000310' MOV #DNI,BITNAM ;NO, GET POINTER TO BIT NAME STRING

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 92  
CZUAAB.MAC 07-APR-83 17:03 CLEAR OUTSTANDING INTERRUPT BITS

4371	017634'	000435				BR	708		:LEAVE
4372	017636'	032777	002000	160472	508:	BIT	#RCEI,BPCSR0		:IS 'RCEI' BIT SET?
4373	017644'	001413				BEG	608		:NO
4374	017646'	112777	000004	160472		MOVB	#RCEIB,BPCSR0UB		:WRITE ONE TO CLEAR 'RCEI'
4375	017654'	032777	002000	160454		BIT	#RCEI,BPCSR0		:DID IT CLEAR? ;RSJ001
4376	017662'	001404				BEG	608		:YES
4377	017664'	012737	001010'	000310'		MOV	#RCEI,BITNAM		:NO, GET POINTER TO BIT NAME STRING
4378	017672'	000416				BR	708		:LEAVE
4379	017674'	032777	000400	160434	608:	BIT	#FATI,BPCSR0		:IS 'FATI' BIT SET?
4380	017702'	001426				BEG	808		:NO
4381	017704'	112777	000001	160434		MOVB	#FATIB,BPCSR0UB		:WRITE ONE TO CLEAR 'FATI'
4382	017712'	032777	000400	160416		BIT	#FATI,BPCSR0		:DID IS CLEAR? ;RSJ001
4383	017720'	001417				BEG	808		:YES
4384	017722'	012737	001021'	000310'		MOV	#SFATI,BITNAM		:NO, GET POINTER TO BIT NAME STRING
4385	017730'				708:	PRINTF	#FORM62,BITNAM		:PRINT ERROR MESSAGE
4386	017730'	013746	000310'					MOV	BITNAM,-(SP)
4387	017734'	012746	010661'					MOV	#FORM62,-(SP)
4388	017740'	012746	000002					MOV	#2,-(SP)
4389	017744'	010600						MOV	SP,R0
4390	017746'	104417						TRAP	CSPNTF
4391	017750'	062706	000006					ADD	#6,SP
4392	017754'	000261				SEC			:INDICATE ERROR TO CALLER
4393	017756'	000401				BR	1008		:LEAVE
4394	017760'	000241			808:	CLC			:INDICATE SUCCESS
4395	017762'	000207			1008:	RTS	PC		

;MAC001

```

4396 .SBTTL PRINT PCSR'S ;MAC001
4397 ;MAC001
4398 :.....;MAC001
4399 :FUNCTIONAL DESCRIPTION: ;MAC001
4400 :ROUTINE TO PRINT THE CONTENTS OF THE PCSR'S ;MAC001
4401 : ;MAC001
4402 :ROUTINE TO PRINT THE CONTENTS OF THE PCSR'S ;MAC001
4403 : ;MAC001
4404 :INPUTS: NONE ;MAC001
4405 : ;MAC001
4406 :OUTPUTS: CONTENTS OF PCSR'S 0,1,2,3 ARE PLACED IN LOCATIONS PCSROC, ;MAC001
4407 : PCSR1C,PCSR2C,PCSR3C RESPECTIVELY AND PRINTED ;MAC001
4408 : ;MAC001
4409 :CALLING SEQUENCE: JSR PC,PRNPCR ;MAC001
4410 : ;MAC001
4411 :.....;MAC001
4412 : ;MAC001
4413 PRNPCR: MOV R1,-(SP) ;SAVE R1 AND R2 ;MAC001
4414 MOV R2,-(SP) ;MAC001
4415 MOV #PCSR3C+2,R1 ;POINT TO BOTTOM OF STORAGE ;MAC001
4416 MOV @PCSR3,-(R1) ;SAVE PCSR3 IN TABLE ;MAC001
4417 MOV @PCSR2,-(R1) ;SAVE PCSR2 IN TABLE ;MAC001
4418 MOV @PCSR1,-(R1) ;SAVE PCSR1 IN TABLE ;MAC001
4419 MOV @PCSR0,-(R1) ;SAVE PCSR0 IN TABLE ;MAC001
4420 CLR R2 ;WHICH PCSR TO PRINT ;MAC001
4421 10$: PRINTX #FORMB3,R2,(R1)+ ;PRINT CONTENTS OF A PCSR ;MAC001
4422 ;MAC001
4423 MOV (R1)+,-(SP) ;MAC001
4424 MOV R2,-(SP) ;MAC001
4425 MOV #FORMB3,-(SP) ;MAC001
4426 MOV #3,-(SP) ;MAC001
4427 MOV SP,R0 ;MAC001
4428 TRAP CSPTX ;MAC001
4429 ADD #10,SP ;MAC001
4430 INC R2 ;NEXT PCSR ;MAC001
4431 CMP R2,#4 ;DONE LAST PCSR? ;MAC001
4432 BNE 10$ ;NOT YET ;MAC001
4433 MOV (SP)+,R2 ;RESTORE R2 ;MAC001
4434 MOV (SP)+,R1 ;RESTORE R1 ;MAC001
RTS PC ;RETURN ;MAC001

```



62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 94  
CZUAAB.MAC 07-APR-83 17:03 CHECK MICROMONITOR

```

4435 .SBTTL CHECK MICROMONITOR
4436
4437 :*****
4438 :
4439 :FUNCTIONAL DESCRIPTION:
4440 :ROUTINE TO WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR
4441 :
4442 :INPUTS: NONE
4443 :
4444 :IMPLICIT INPUTS: PCSR1
4445 :
4446 :OUTPUTS: C BIT SET IN PSW IF TIMEOUT WAITING FOR MICROMONITOR
4447 :          C BIT CLEAR IN PSW IF MICROCODE IN MICROMONITOR
4448 :
4449 :SUBORDINATE ROUTINES CALLED: TIMON, TIMOFF
4450 :
4451 :FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK INTERRUPTS ARE ENABLED
4452 :
4453 :CALLING SEQUENCE: JSR PC,CHKMON
4454 :
4455 :*****

```

```

4456
4457 020060' 012737 000077 000332' CHKMON::MOV #1*SECOND,METER ;TIMEOUT PERIOD IS 1 SECOND
4458 020066' 004737 017264' JSR PC,TIMON ;TURN ON THE CLOCK
4459 020072' 122777 000001 160240 10$: CMPB #TIMON,BPCSR1 ;IS THE MICROCODE IN THE MICROMONITOR?
4460 020100' 001410 BEQ 20$ ;YES
4461 020102' BREAK ;RETURN TO DRS FOR A MOMENT
4462 020102' 104422 TRAP CSBRK
4463 020104' 005737 000332' TST METER ;HAS THE TIMER EXPIRED?
4464 020110' 001370 BNE 10$ ;NOT YET, KEEP CHECKING MICROCODE
4465 020112' 004737 017302' JSR PC,TIMOFF ;TIMER HAS EXPIRED TURN OFF THE TIMER
4466 020116' 000261 SEC ;INDICATE ERROR TO CALLER
4467 020120' 000403 BR 30$ ;LEAVE
4468 020122' 004737 017302' 20$: JSR PC,TIMOFF ;STOP THE CLOCK
4469 020126' 000241 CLC ;INDICATE TO CALLER MICROCODE IS IN
4470 ;MICROMONITOR
4471 020130' 000207 30$: RTS PC

```

GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 95  
CZUAAB.MAC 07-APR-83 17:03 CHECK INTERRUPT ERROR BITS

4472  
4473  
4474  
4475  
4476  
4477  
4478  
4479  
4480  
4481  
4482  
4483  
4484  
4485  
4486  
4487  
4488  
4489  
4490  
4491  
4492  
4493  
4494

.SBTTL CHECK INTERRUPT ERROR BITS  
:\*\*\*\*\*  
:FUNCTIONAL DESCRIPTION:  
:ROUTINE TO CHECK FOR ANY T11 INTERRUPT ERROR BITS IN PCSRO  
:INPUTS: NONE  
:IMPLICIT INPUTS: PCSRO  
:OUTPUTS: C BIT SET IN PSW IF AN INTERRUPT BIT IS SET  
:C BIT CLEAR IN NO INTERRUPT BITS ARE SET  
:'ERRINT' CONTAINS COPY OF PCSRO  
:SUBORDINATE ROUTINES CALLED: NONE  
:FUNCTIONAL SIDE EFFECTS: ANY INTERRUPT BIT SET IS CLEARED  
:CALLING SEQUENCE: JSR PC,CHKINT  
:\*\*\*\*\*

4495 020132° 017737 160200 000670° CHKINT::MOV @PCSRO,ERRINT ;GET PCSRO CONTENTS  
4496 020140° 032737 170000 000670° BIT #NPRERR!NXMERR!UNIERR!PARERR,ERRINT ;ANY INTERRUPT ERRORS SET?  
4497 020146° 001405 BEQ 10\$ ;NO  
4498 020150° 012777 170000 160160 MOV #NPRERR!NXMERR!UNIERR!PARERR,@PCSRO ;CLEAR ANY ERROR INTERRUPT  
4499 020156° 000261 SEC ;INDICATE ERROR  
4500 020160° 000401 BR 20\$  
4501 020162° 000241 10\$: CLC ;INDICATE NO ERRORS  
4502 020164° 000207 20\$: RTS PC

4503  
4504  
4505  
4506  
4507  
4508  
4509  
4510  
4511  
4512  
4513  
4514  
4515  
4516  
4517  
4518  
4519  
4520  
4521  
4522  
4523  
4524  
4525  
4526  
4527  
4528  
4529  
4530  
4531  
4532  
4533  
4534  
4535  
4536  
4537  
4538  
4539  
4540  
4541  
4542  
4543  
4544  
4545  
4546  
4547  
4548  
4549  
4550  
4551  
4552  
4553  
4554  
4555  
4556  
4557  
4558

.SBTTL RESET UNA

\*\*\*\*\*  
:FUNCTIONAL DESCRIPTION:  
:ROUTINE TO RESET DEUNA  
:INPUTS: NONE  
:IMPLICIT INPUTS: NONE  
:OUTPUTS: C BIT SET IF ERROR OCCURRED  
:C BIT CLEARED IF SUCCESS  
:SUBORDINATE ROUTINES USED: CHKDNI, CLRDNI  
:FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK IS TURNED ON, OPERATIONAL  
:MICROCODE IS STARTED.  
:CALLING SEQUENCE: JSR PC,REUNA  
:\*\*\*\*\*8

```
REUNA:: MOV #RSET,BPCSR0 ;RESET DEUNA BACK TO OPERATIONAL MICRO
        MOV #12,*SECOND,METER ;PUT SOME TIME ON THE METER
        JSR PC,CHKDNI ;WAIT FOR 'DNI'
        BCC 10$ ;OK
        ;ERROR DNI NOT SET AFTER RESET!
        ;SETUP ERROR MESSAGE
        MOV #SDNI,BITNAM
        MOV #SNSET,BITSTA
        MOV #SAFTER,PWHEN
        CLR CSRNUM
        PRINTF #FORM9,CSRNUM,BITNAM,BITSTA,PWHEN
        MOV PWHEN,-(SP)
        MOV BITSTA,-(SP)
        MOV BITNAM,-(SP)
        MOV CSRNUM,-(SP)
        MOV #FORM9,-(SP)
        MOV #5,-(SP)
        MOV SP,RO
        TRAP CSPNTF
        ADD #14,SP
        ;MAC001
        JSR PC,PRNPCR ;PRINT PCRS'S
        SEC ;INDICATE ERROR
        BR 20$
10$: JSR PC,CLRDNI ;GO CLEAR DNI
     BCC 20$ ;OK
     PRINTF #FORM8
     MOV #FORM8,-(SP)
     MOV #1,-(SP)
     MOV SP,RO
     TRAP CSPNTF
     ADD #4,SP
```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 97  
CZUAAB.MAC 07-APR-83 17:03 RESET UNA

4559 020334' 000261  
4560 020336' 000207

20%: SEC  
RTS PC

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 98  
CZUAAB.MAC 07-APR-83 17:03 RESET UNA

```

4561
4562
4563 .SBTTL LOAD MICROMODULE
4564
4565
4566
4567
4568
4569
4570
4571 020340' 012737 000326' 000320' LODMIC: MOV #MICRO,MICMOD ;POINT TO MICRO MODULE NAME
4572 020346' 004737 020166' JSR PC,REUNA ;GO START OPERATION MICROCODE
4573 020352' 103015 BCC 5$ ;OK
4574 020354' PRINTF #UNLOD,MICMOD ;PRINT ERROR MESSAGE
4575 020354' 013746 000320' MOV MICMOD,-(SP)
4576 020360' 012746 001517' MOV #UNLOD,-(SP)
4577 020364' 012746 000002 MOV #2,-(SP)
4578 020370' 010600 MOV SP,RO
4579 020372' 104417 TRAP CSPNTF
4580 020374' 062706 000006 ADD #6,SP
4581 020400' 000261 SEC ;INDICATE ERROR OCCURRED
4582 020402' 000137 021174' JMP 20$ ;LEAVE
4583 020406' 023727 000326' 000101 5$: CMP MICRO,#'A ;LOAD MICROCODE MODULE A?
4584 020414' 001007 BNE 10$ ;NO
4585 ;YES
4586 020416' 013737 000000G 000616' MOV MICASZ,UDBB ;SIZE OF MODULE A
4587 020424' 012737 000000G 000620' MOV #MICROA,UDBB+2 ;BASE ADDRESS OF MODULE A
4588 020432' 000475 BR 70$
4589
4590 020434' 023727 000326' 000102 10$: CMP MICRO,#'B ;LOAD MICROCODE MODULE B?
4591 020442' 001007 BNE 20$ ;NO
4592 ;YES
4593 020444' 013737 000000G 000616' MOV MICBSZ,UDBB ;SIZE OF MODULE B
4594 020452' 012737 000000G 000620' MOV #MICROB,UDBB+2 ;BASE ADDRESS OF MODULE B
4595 020460' 000462 BR 70$
4596
4597 020462' 023727 000326' 000103 20$: CMP MICRO,#'C ;LOAD MICROCODE MODULE C?
4598 020470' 001007 BNE 30$ ;NO
4599 ;YES
4600 020472' 013737 000000G 000616' MOV MICCSZ,UDBB ;SIZE OF MODULE C
4601 020500' 012737 000000G 000620' MOV #MICROC,UDBB+2 ;BASE ADDRESS OF MODULE C
4602 020506' 000447 BR 70$
4603
4604 020510' 023727 000326' 000104 30$: CMP MICRO,#'D ;LOAD MICROCODE MODULE D?
4605 020516' 001007 BNE 40$ ;NO
4606 ;YES
4607 020520' 013737 000000G 000616' MOV MICDSZ,UDBB ;SIZE OF MODULE D
4608 020526' 012737 000000G 000620' MOV #MICROD,UDBB+2 ;BASE ADDRESS OF MODULE D
4609 020534' 000434 BR 70$
4610
4611 020536' 023727 000326' 000105 40$: CMP MICRO,#'E ;LOAD MICROCODE MODULE E?
4612 020544' 001007 BNE 50$ ;NO
4613 ;YES
4614 020546' 013737 000000G 000616' MOV MICESZ,UDBB ;SIZE OF MODULE E
4615 020554' 012737 000000G 000620' MOV #MICROE,UDBB+2 ;BASE ADDRESS OF MODULE E
4616 020562' 000421 BR 70$

```

62GLOBAL AREAS MACY11 30A(1052) 07-APR-83 17:13 PAGE 99  
 CZUAAB.MAC 07-APR-83 17:03 LOAD MICROMODULE

```

4617
4618 020564' 023727 000326' 000106 50$: CMP MICRO,#'F ;LOAD MICROCODE MODULE F?
4619 020572' 001007 BNE 60$ ;NO
4620 ;YES
4621 020574' 013737 000000G 000616' MOV MIC:SZ,UDBB ;SIZE OF MODULE F
4622 020602' 012737 000000G 000620' MOV #MICROF,UDBB+2 ;BASE ADDRESS OF MODULE F
4623 020610' 000406 BR 70$
4624
4625 020612' 013737 000000G 000616' 60$: MOV MICGSZ,UDBB ;SIZE OF MODULE G
4626 020620' 012737 000000G 000620' MOV #MICROG,UDBB+2 ;BASE ADDRESS OF MODULE G
4627
4628 020626' 005037 000622' 70$: CLR UDBB+4
4629 020632' 012737 010000 000624' MOV #WCSADR+<WCSSIZ/2>,UDBB+6 ;LOAD INTO TOP HALF OF WCS
4630
4631 ;SETUP PCB
4632 020640' 012737 000021 000606' MOV #LIM,PCBB ;'LOAD INTERNAL MEMORY' FUNCTION
4633 020646' 012737 000616' 000610' MOV #UDBB,PCBB+2 ;SET ADDRESS OF UDBB
4634 020654' 005037 000612' CLR PCBB+4
4635 020660' 012777 000606' 157454 MOV #PCBB,#PCSR2 ;TELL DEUNA WHERE PCBB IS
4636 020666' 005077 157452 CLR #PCSR3
4637
4638 020672' 004737 017410' JSR PC,CLRINT ;CLEAR ANY OUTSTANDING INTERRUPT BITS
4639 020676' 103014 BCC 75$ ;OK
4640 020700' PRINTF #UNLOD,MICMOD ;CAN'T CONTINUE WITH INTERRUPT BITS SET
4641 020700' 013746 000320' MOV MICMOD,-(SP)
4642 020704' 012746 001517' MOV #UNLOD,-(SP)
4643 020710' 012746 000002 MOV #2,-(SP)
4644 020714' 010600 MOV SP,RO
4645 020716' 104417 TRAP C$PNTF
4646 020720' 062706 000006 ADD #6,SP
4647 020724' 000261
4648 020726' 000522 SEC
4649 BR 200$
4650 020730' 012777 000001 157400 75$: MOV #GETPCB,#PCSR0 ;ISSUE 'GET PCB' PORT COMMAND
4651 020736' 012737 001364 000332' MOV #12.*SECOND,METER ;SETUP TIMER
4652 020744' 004737 017316' JSR PC,CHKDNI ;WAIT FOR 'DNI' TO SET
4653 020750' 103014 BCC 80$ ;OK
4654 ;ERROR DNI NOT SET!
4655 PRINTF #UNLOD,MICMOD ;PRINT MESSAGE
4656 020752' 013746 000320' MOV MICMOD,-(SP)
4657 020756' 012746 001517' MOV #UNLOD,-(SP)
4658 020762' 012746 000002 MOV #2,-(SP)
4659 020766' 010600 MOV SP,RO
4660 020770' 104417 TRAP C$PNTF
4661 020772' 062706 000006 ADD #6,SP
4662 020776' 000261
4663 021000' 000475 SEC ;INDICATE ERROR
4664 021002' 80$: BR 200$
4665 021002' 004737 017362' JSR PC,CLR DNI ;GO CLEAR 'DNI'
4666 021006' 103014 BCC 90$ ;OK
4667 ;ERROR 'DNI' NOT CLEAR!
4668 PRINTF #UNLOD,MICMOD ;PRINT MESSAGE
4669 021010' 013746 000320' MOV MICMOD,-(SP)
4670 021014' 012746 001517' MOV #UNLOD,-(SP)
4671 021020' 012746 000002 MOV #2,-(SP)
4672 021024' 010600 MOV SP,RO
    
```



73 MISCELLANEOUS SECTIONS  
 CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 101  
 LOAD MICROMODULE

```

4713
4714
4715 021176'
4716 021176'
4717 021176'
4718 021176' 000167
4719 021200' 000000
4720 021202'
4721 021202'
4722 021202' 104425
4723
4724
4725
4726
4727
4728
4729
4730
4731
4732 021204'
4733 021204'
4734
4735 021204' 177777
4736 021206' 177777
4737 021210' 177777
4738
4739 021212'
4740

.TITLE MISCELLANEOUS SECTIONS
.SBTTL REPORT CODING SECTION
      BGNRPT
      EXIT RPT
      ENDRPT

      LSRPT::
            .WORD JSJMP
            .WORD L10064-2-.

L10064: TRAP CSRPT

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

      BGNPROT
            LSPROT::
-1          :OFFSET INTO P-TABLE FOR CSR ADDRESS
-1          :OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
-1          :OFFSET INTO P-TABLE FOR DRIVE NUMBER

      ENDPROT

```



73MISCELLANEOUS SECTIONS  
CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 102  
INITIALIZE SECTION

```

4741
4742
4743
4744
4745
4746
4747
4748 021212'          BGNINIT
4749 021212'
4750
4751
4752
4753 021212'          READEF #EF.CONTINUE          ;WAS A CONTINUE COMMAND ENTERED
4754 021212' 012700 000036          ;MOV #EF.CONTINUE,RO
4755 021216' 104447          ;TRAP CSREFG
4756 021220'          BCOMPLETE          30$          ;YES, LEAVE INIT CODE
4757 021220' 103572          READEF #EF.PWR          ;WAS THERE A POWER FAILURE?
4758 021222'          ;BCS 30$
4759 021222' 012700 000034          ;MOV #EF.PWR,RO
4760 021226' 104447          ;TRAP CSREFG
4761
4762 021230'          ;MAC001 BCOMPLETE          30$          ;YES, LEAVE INIT CODE
4763 021230' 103007          ;BNCOMPLETE          2$          ;NO
4764
4765          ;          ;          ;          ;MAC001
4766          ;          ;          ;          ;MAC001
4767          ;          ;          ;          ;MAC001
4768          ;          ;          ;          ;MAC001
4769          ;          ;          ;          ;MAC001
4770          ;          ;          ;          ;MAC001
4771          ;          ;          ;          ;MAC001
4772          ;          ;          ;          ;MAC001
4773          ;          ;          ;          ;MAC001
4774 021232' 012701 000150          ;          ;          ;          ;MAC001
4775 021234' 104447          ;          ;          ;          ;MAC001
4776 021236'          ;          ;          ;          ;MAC001
4777 021236' 103072          ;          ;          ;          ;MAC001
4778 021260'          ;          ;          ;          ;MAC001
4779 021260' 012700 000040          ;          ;          ;          ;MAC001
4780 021264' 104447          ;          ;          ;          ;MAC001
4781 021266'          ;          ;          ;          ;MAC001
4782 021266' 103061          ;          ;          ;          ;MAC001
4783 021270' 012737 000001 000674'          ;          ;          ;          ;MAC001
4784 021276'          ;          ;          ;          ;MAC001
4785 021276' 104431          ;          ;          ;          ;MAC001
4786 021300' 010037 000322'          ;          ;          ;          ;MAC001
4787 021304' 013737 000322' 000324'          ;          ;          ;          ;MAC001
4788 021312' 062737 000002 000324'          ;          ;          ;          ;MAC001
4789 021320'          ;          ;          ;          ;MAC001
4790 021320' 012700 000114          ;          ;          ;          ;MAC001
4791 021324' 104462          ;          ;          ;          ;MAC001
4792 021326' 010001          ;          ;          ;          ;MAC001
4793 021330'          ;          ;          ;          ;MAC001
4794 021330' 103411          ;          ;          ;          ;MAC001
4795 021332'          ;          ;          ;          ;MAC001
4796 021332' 012746 021610'          ;          ;          ;          ;MAC001

```

73MISCELLANEOUS SECTIONS  
CZUAAB.MAC 07-APR-83 17:03

MACY1: 30A(1052) 07-APR-83 17:13 PAGE 103  
INITIALIZE SECTION

4797	021336'	012746	000001						
4798	021342'	010600						MOV	#1,-(SP)
4799	021344'	104417						MOV	SP,RO
4800	021346'	062706	000004					TRAP	CSPNIF
4801	021352'	000512						ADD	#4,SP
4802	021354'	012137	000274'	18:	BR	20s			:CANNOT CONTINUE
4803	021360'	012102			MOV	(R1)+,CLKCSR			:LINE CLOCK CSR
4804	021362'	072227	000005		MOV	(R1)+,R2			:GET CLOCK PRIORITY
4805	021366'	010237	000276'		ASH	#5,R2			
4806	021372'	012137	000300'		MOV	R2,CLKBR			
4807	021376'	012137	000302'		MOV	(R1)+,CLKVEC			:VECTOR
4808	021402'				MOV	(R1)+,CLKFRE			:FREQUENCY
4809	021402'	013746	000276'		SETVEC	CLKVEC,#CLKSRV,CLKBR			:SETUP CLOCK INTERRUPT VECTOR
4810	021406'	012746	022062'					MOV	CLKBR,-(SP)
4811	021412'	013746	000300'					MOV	#CLKSRV,-(SP)
4812	021416'	012746	000003					MOV	CLKVEC,-(SP)
4813	021422'	104437						MOV	#3,-(SP)
4814	021424'	062706	000010					TRAP	CSSVEC
4815	021430'	000402						ADD	#10,SP
4816	021432'	005037	000674'	68:	BR	5s			
4817	021436'	012737	177777	000330'	CLR	FRSTIM			:INDICATE NOT THE FIRST TIME THROUGH
4818	021444'	005237	000330'	58:	MOV	#-1,UNIT			:YES, INITIALIZE UNIT #
4819	021450'	023737	000330'	108:	INC	UNIT			:SETUP FOR NEXT UNIT
4820	021456'	003050	000012'		CFP	UNIT,LSUNIT			:WE TESTED ALL AVAILABLE UNITS?
4821	021460'				BGT	20s			:YES, LEAVE
4822	021460'	013700	000330'		GPHARD	UNIT,R1			:GET P-TAB POINTER FOR THIS UNIT
4823	021464'	104442						MOV	UNIT,RO
4824	021466'	010001						TRAP	CSGPHRD
4825	021470'							MOV	RO,R1
4826	021470'	103365			BNCOMPLETE	10s			:THIS ONE IS NOT AVAILABLE
4827	021472'	012137	000266'					BCC	10s
4828	021476'	012137	000270'		MOV	(R1)+,UNACSR			:SAVE CSR
4829	021502'	012737	000240	000272'	MOV	(R1)+,UNAVEC			:SAVE VECTOR
4830	021510'	013737	000266'	000336'	MOV	#PRI05,UNAPRI			:SAVE PRIORITY
4831	021516'	013737	000336'	000346'	MOV	UNACSR,PCSR0			:PCSR0
4832	021524'	062737	000001	000346'	MOV	PCSR0,PCSR0UB			:PCSR0 UPPER BYTE
4833	021532'	013737	000336'	000340'	ADD	#1,PCSR0UB			:RSJ001
4834	021540'	062737	000002	000340'	MOV	PCSR0,PCSR1			:RSJ001
4835	021546'	013737	000340'	000342'	ADD	#2,PCSR1			:PCSR1
4836	021554'	062737	000002	000342'	MOV	PCSR1,PCSR2			:PCSR2
4837	021562'	013737	000342'	000344'	ADD	#2,PCSR2			:PCSR2
4838	021570'	062737	000002	000344'	MOV	PCSR2,PCSR3			:PCSR3
4839	021576'	000403			ADD	#2,PCSR3			:LEAVE
4840	021600'	005037	000674'	20s:	BR	30s			:CLEAR FIRST TIME THROUGH FLAG
4841	021604'				CLR	FRSTIM			:ABORT PASS
4842	021604'	104444			DOCLN				
4843	021606'			30s:				TRAP	CSOCLN
4844	021606'			ENDINIT					
4845	021606'								
4846	021606'	104411						L10066:	TRAP
4847									CSINIT
4848									
4849	021610'	040503	047116	052117	NOCLK::	.ASCIZ /CANNOT CONTINUE - NEED LINE CLOCK/			:MAC001
4850	021616'	041440	047117	044524					
4851	021624'	052516	020105	020055					
4852	021632'	042516	042105	046040					

73 MISCELLANEOUS SECTIONS  
CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 104  
INITIALIZE SECTION

4853 021640' 047111 020105 046103  
4854 021646' 041517 000113  
4855  
4856

.EVEN

73MISCELLANEOUS SECTIONS  
C7UAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 105  
AUTODROP SECTION

```

4857
4858
4859 021652'
4860 021652'
4861 021652'
4862 021652'
4863 021652' 104461
4864
4865
4866
4867
4868
4869
4870
4871 021654'
4872 021654'
4873 021654' 005737 000666'
4874 021660' 001025
4875 021662' 012777 000040 156446
4876
4877
4878 021670' 012737 001364 000332'
4879 021676' 004737 017316'
4880 021702' 103012
4881 021704'
4882 021704' 012746 021744'
4883 021710' 012746 000001
4884 021714' 010600
4885 021716' 104417
4886 021720' 062706 000004
4887 021724' 004737 017764'
4888 021730' 004737 017362'
4889 021734' 004737 017302'
4890 021740'
4891 021740' 104432
4892 021742' 000054
4893
4894 021744' 047045 040445 051105
4895 021752' 047522 020122 041517
4896 021760' 052503 051122 042105
4897 021766' 042040 051125 047111
4898 021774' 020107 042504 044526
4899 022002' 042503 051125 051505
4900 022010' 052105 047045 000
4901
4902
4903 022016'
4904 022016'
4905 022016' 104412

```

```

.SBTTL AUTODROP SECTION
BGNAUTO
ENDAUTO
LBAUTO::
L10067: TRAP CSAUTO

.SBTTL CLEANUP CODING SECTION
:++
: THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
: AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
:--
BGNCLN
LSCLEAN::
TST NEXMEM ;DOES PCSRO EXIST?
BNE 208 ;NO SKIP RESET
MOV #RSET,BPCSR0 ;RESTORE DEUNA TO OPERATIONAL
;STATE
;MAC001 MOV #10*SECOND,METER ;SETUP TIMER
MOV #12*SECOND,METER ;SET UP TIMER ;MAC001
JSR PC,CHKDNI ;DMI SET WHEN DONE
BCC 108
PRINTF #CLNERR ;ERROR
MOV #CLNERR,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP CSPTF
ADD #4,SP
;MAC001
108: JSR PC,PRMPCR ;PRINT PCSR'S
208: JSR PC,CLRDMI ;CLEAR DMI BIT
EXIT CLN ;TURN OFF THE LINE CLOCK ;CC
TRAP CSEXIT
.MWORD L10070-.

CLNERR: .ASCIZ /%N%AERROR OCCURRED DURING DEVILE RESET%N/

.EVEN
ENDCLN
L10070: TRAP CSCLEAN

```

73 MISCELLANEOUS SECTIONS  
CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 106  
DROP UNIT SECTION

```

4906      .SBTTL  DROP UNIT SECTION
4907
4908      BGNDU
4909      022020'
4910      022020'
4911      022020' 000167
4912      022022' 000000
4913      022024'
4914      022024'
4915      022024' 104453
4916
4917      .SBTTL  ADD UNIT SECTION
4918
4919      BGNAU
4920      022026'
4921      022026'
4922      022026' 000167
4923      022030' 000000
4924      022032'
4925      022032'
4926      022032' 104452
4927
4928      .TITLE  GLOBAL INTERRUPT SERVICE ROUTINES
4929
4930      .SBTTL  NON-EXISTANT MEMORY INTERRUPT SERVICE ROUTINE
4931
4932      :*****
4933      :
4934      :FUNCTIONAL DESCRIPTION:
4935      :
4936      :           THIS ROUTINE IS ASSIGNED TO VECTOR 4 BY THE ACCESS TESTS
4937      :           IT SETS THE NEXMEM FLAG SIGNALING THAT AN ACCESS WAS
4938      :           ATTEMPTED ON NON-EXISTANT MEMORY.
4939      :
4940      :*****
4941
4942      BGNSRV TRAP4
4943      022034'
4944      022034' 012737 000001 000666'
4945      022042'
4946      022042'
4947      022042' 000002

                                LSDU::
                                .WORD  JSJMP
                                .WORD  L10071-2-.
L10071: TRAP  CSU

                                LSAU::
                                .WORD  JSJMP
                                .WORD  L10072-2-.
L10072: TRAP  CSAU

TRAP4::
L10073: RTI

```

84GLOBAL INTERRUPT SERVICE ROUTINES  
 CZUAAB.MAC 07-APR-83 17:03

MACY11 30A(1052) 07-APR-83 17:13 PAGE 107  
 NON-EXISTANT MEMORY INTERRUPT SERVICE ROUTINE

4948  
 4949  
 4950  
 4951  
 4952  
 4953  
 4954  
 4955  
 4956  
 4957  
 4958  
 4959  
 4960  
 4961  
 4962  
 4963  
 4964  
 4965  
 4966  
 4967  
 4968  
 4969  
 4970  
 4971  
 4972  
 4973  
 4974  
 4975  
 4976  
 4977  
 4978  
 4979  
 4980  
 4981  
 4982  
 4983  
 4984  
 4985  
 4986  
 4987  
 4988  
 4989  
 4990  
 4991  
 4992  
 4993  
 4994  
 4995  
 4996  
 4997

.SBTTL UNA INTERRUPT SERVICE ROUTINE

```

:*****
:
:FUNCTIONAL DESCRIPTION:
:   CONTROL GOES HERE WHEN THE INTERRUPT ENABLE BIT IS SET
:   AND ANY OF THE INTERRUPT BITS SET.
:   A COPY OF PCSRO IS STORED AT 'UNMAINT:' AND A COPY OF THE
:   PSW AT THE TIME OF THE INTERRUPT IS STORED AT 'CPUPRI:'.
:
:*****
    
```

```

BGNSRV UNASRV
UNASRV::
MOV @PCSRO,UNMAINT ;INDICATE UNA INTERRUPTED
MOV 2(SP),CPUPRI ;GET CPU PRIORITY AT TIME OF INTERRUPT
ENDSRV
L10074:
RTI
    
```

.SBTTL CLOCK INTERRUPT SERVICE ROUTINE

```

:*****
:
:FUNCTIONAL DESCRIPTION:
:   THIS ROUTINE COUNTS A PRESET NUMBER OF CLOCK TICKS THEN IT
:   TURNS THE CLOCK OFF
:
:INPUTS: METER
:OUTPUTS:METER
:ROUTINES CALLED: NONE
:*****
    
```

```

BGNSRV CLKSRV
CLKSRV::
TST METER ;HAS THE METER EXPIRED?
BEQ 20$ ;YES, STOP COUNTING
DEC METER ;COUNT TICKS
20$:
ENDSRV
L10075:
RTI
    
```

```

022044'
022044'
022044' 017737 156266 000672'
022052' 016637 000002 000676'
022060'
022060'
022060' 000002
022062'
022062'
022062' 005737 000332'
022066' 001402
022070' 005337 000332'
022074'
022074'
022074'
022074' 000002
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 108  
CZUAAB.MAC 07-APR-83 17:03 CLOCK INTERRUPT SERVICE ROUTINE

.TITLE HARDWARE TESTS

.SBTTL TEST 1: PCSKO READ ACCESS TEST

\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 0  
:CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS APPEAR  
:IN THE EXPECTED BIT POSITIONS.  
:TEST SEQUENCE: 1-READ PCSRO  
:                  2-VERIFY BITS 9 AND 4 = 0  
\*\*\*\*\*

4998  
4999  
5000  
5001  
5002  
5003  
5004  
5005  
5006  
5007  
5008  
5009  
5010  
5011  
5012

5013 022076'  
5014 022076'  
5015 022076'  
5016 022076' 012746 000340  
5017 022102' 012746 022034'  
5018 022106' 012746 000004  
5019 022112' 012746 000003  
5020 022116' 104437  
5021 022120' 062706 000010  
5022 022124'  
5023 022124'  
5024 022124' 104402

BGNTST

SETVEC #4,#TRAP4,#PRI07

T1::  
;SETUP TIME-OUT TRAP  
MOV #PRI07,-(SP)  
MOV #TRAP4,-(SP)  
MOV #4,-(SP)  
MOV #3,-(SP)  
TRAP CSVEC  
ADD #10,SP

BGNSUB ;#1

T1.1: TRAP CSBSUB

:CHECK TO SEE IF PCSRO EXISTS

5025  
5026  
5027  
5028 022126' 005037 000666'  
5029 022132' 005037 000304'  
5030 022136' 005777 156174  
5031 022142' 005737 000666'  
5032 022146' 001414  
5033 022150'  
5034 022150' 104455  
5035 022152' 000001  
5036 022154' 001570'  
5037 022156' 012500'  
5038 022160'  
5039 022160' 104406  
5040 022162'  
5041 022162' 012700 000004  
5042 022166' 104436  
5043 022170'  
5044 022170' 013700 000330'  
5045 022174' 104451  
5046 022176'  
5047 022176' 104444  
5048 022200'  
5049 022200'  
5050 022200'  
5051 022200' 104403  
5052 022202'  
5053 022202'

CLR NEXMEM ;CLEAR NON-EXISTANT MEMORY FLAG  
CLR CSRNUM ;HOLDS WHICH PCSR WE ARE DOING  
TST @PCSRO ;DOES PCSRO EXIST?  
TST NEXMEM  
BEQ 10\$ ;YES  
ERRDF 001,RACERR,RACMG1 ;NO, PRINT FATAL ERROR MESSAGE

TRAP CSERDF  
.WORD 1  
.WORD RACERR  
.WORD RACMG1

CKLJOP ;LOOP BACK FROM HERE IF ERROR  
TRAP CSCLP1

CLRVEC #4 ;RELEASE TRAP 4 VECTOR  
MOV #4,R0  
TRAP CSCVEC

DODU UNIT ;DROP UNIT  
MOV UNIT,R0  
TRAP CSDDODU

DOCLN ;ABORT SUB-PASS  
TRAP CSDECLN

10\$:  
ENDSUB ;#1

L10077: TRAP CSSESUB

BGNSUB ;#2

T1.2:





5102  
5103  
5104  
5105  
5106  
5107  
5108  
5109  
5110  
5111  
5112  
5113  
5114  
5115  
5116  
5117  
5118  
5119  
5120  
5121  
5122  
5123  
5124  
5125  
5126  
5127  
5128  
5129  
5130  
5131  
5132  
5133  
5134  
5135  
5136  
5137  
5138  
5139  
5140  
5141  
5142  
5143  
5144  
5145  
5146  
5147  
5148  
5149  
5150  
5151  
5152  
5153  
5154  
5155  
5156  
5157

022310'  
022310'  
022310' 012746 000340  
022314' 012746 022034'  
022320' 012746 000004  
022324' 012746 000003  
022330' 104437  
022332' 062706 000010  
022336'  
022336'  
022336' 104402  
022340' 005037 000666'  
022344' 012737 000001 000304'  
022352' 005777 155762  
022356' 005737 000666'  
022362' 001414  
022364'  
022364' 104455  
022366' 000004  
022370' 001570'  
022372' 012500'  
022374'  
022374' 104406  
022376'  
022376' 012700 000004  
022402' 104436  
022404'  
022404' 013700 000330'  
022410' 104451  
022412'  
022412' 104444  
022414'  
022414'  
022414'  
022414' 104403  
022416'  
022416'  
022416' 104402

.SBTTL TEST 2: PCSR1 READ ACCESS TEST

:\*\*\*\*\*  
: THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER  
: CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS  
: APPEAR IN THE EXPECTED BIT POSITIONS.  
: TEST SEQUENCE: 1-READ PCSR1  
: 2-VERIFY BITS 4,5,6,14,15 = 0  
:\*\*\*\*\*

BGNTST

SETVEC #4,#TRAP4,#PRI07

T2::  
:SETUP TIMEOUT TRAP

MOV #PRI07,-(SP)  
MOV #TRAP4,-(SP)  
MOV #4,-(SP)  
MOV #3,-(SP)  
TRAP C\$SVEC  
ADD #10,SP

BGNSUB ;#1

T2.1:

TRAP C\$BSUB

:CHECK TO SEE IF PCSR1 EXISTS

CLR NEXMEM  
MOV #1,CSRMUM  
TST @PCSR1  
TST NEXMEM  
BEQ 10\$  
ERRDF 004,RACERR,RACMG1

:CLEAR NON-EXISTANT MEMORY FLAG  
:TESTING PCSR1  
:DOES PCSR1 EXIST?

:YES  
:NO,PRINT FATAL ERROR MESSAGE

TRAP C\$ERDF  
.WORD 4  
.WORD RACERR  
.WORD RACMG1

CKLOOP

:LOOP FROM HERE IF ERROR

TRAP C\$CLP1

CLRVEC #4

:RELEASE TRAP 4 VECTOR

MOV #4,R0  
TRAP C\$CVEC

DODU UNIT

:DROP THE UNIT

MOV UNIT,R0  
TRAP C\$DODU

DOCLN

:ABORT SUB-PASS

TRAP C\$DCLN

10\$:

ENDSUB ;#1

L10102:

TRAP C\$ESUB

BGNSUB ;#2

T2.2:

TRAP C\$BSUB

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 111  
CZUAAB.MAC 07-APR-83 17:03 TEST 2: PCSR1 READ ACCESS TEST

```

5158
5159      ;OK, PCSR1 EXISTS NOW CHECK SOME BITS
5160
5161      022420'
5162      022420' 104404
5163
5164      ;CHECK BIT 4 = 0
5165
5166      022422' 012737 001313' 000312'      MOV      #SNCLR,BITSTA      ;TESTING CLEARED BITS
5167      022430' 012737 000004 000306'      MOV      #4,BITNUM        ;TESTING BIT 4
5168      022436' 032777 000020 155674      BIT      #BIT4,@PCSR1     ;IS BIT 4=0?
5169      022444' 001404      BEQ      20$              ;YES
5170      022446'      ERRHRD 005,RACERR,RACMG2    ;NO, PRINT HARD ERROR MESSAGE
5171      022446' 104456      TRAP
5172      022450' 000005      .WORD   5                CSERHRD
5173      022452' 001570'      .WORD   RACERR
5174      022454' 012526'      .WORD   RACMG2
5175      022456'
5176      022456'      20$:
5177      022456'      ENDSEG
5178      022456' 104405      10000$:
5179      022460'      BGNSEG      TRAP      C$ESEG
5180      022460' 104404      TRAP      C$BSEG
5181
5182      ;CHECK BIT 5 = 0
5183
5184      022462' 012737 000005 000306'      MOV      #5,BITNUM        ;TESTING BIT 5
5185      022470' 032777 000040 155642      BIT      #BIT5,@PCSR1     ;IS BIT 5=0?
5186      022476' 001404      BEQ      30$              ;YES
5187      022500'      ERRHRD 006,RACERR,RACMG2    ;NO,PRINT HARD ERROR MESSAGE
5188      022500' 104456      TRAP
5189      022502' 000006      .WORD   6                CSERHRD
5190      022504' 001570'      .WORD   RACERR
5191      022506' 012526'      .WORD   RACMG2
5192      022510'
5193      022510'      30$:
5194      022510'      ENDSEG
5195      022510' 104405      10001$:
5196      022512'      BGNSEG      TRAP      C$ESEG
5197      022512' 104404      TRAP      C$BSEG
5198
5199      ;CHECK BIT 6 = 0
5200
5201      022514' 012737 000006 000306'      MOV      #6,BITNUM        ;TESTING BIT 6
5202      022522' 032777 000100 155610      BIT      #BIT6,@PCSR1     ;IS BIT 6=0?
5203      022530' 001404      BEQ      40$              ;YES
5204      022532'      ERRHRD 007,RACERR,RACMG2    ;NO,PRINT HARD ERROR MESSAGE
5205      022532' 104456      TRAP
5206      022534' 000007      .WORD   7                CSERHRD
5207      022536' 001570'      .WORD   RACERR
5208      022540' 012526'      .WORD   RACMG2
5209      022542'
5210      022542'      40$:
5211      022542'      ENDSEG
5212      022542' 104405      10002$:
5213      022544'      BGNSEG      TRAP      C$ESEG

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 112  
 CZUAAB.RAC 07-APR-83 17:03 TEST 2: PCSR1 READ ACCESS TEST

```

5214 022544' 104404                                TRAP  C$BSEG
5215
5216                                ;CHECK BIT 14 = 0
5217                                ;
5218 022546' 012737 000016 000306'                MOV    #14.,BITNUM                ;TESTING BIT 14
5219 022554' 032777 040000 155556                BIT    #BIT14,@PCSR1            ;IS BIT 14 =0?
5220 022562' 001404                                BEQ    50$                       ;YES
5221 022564'                                ERRHRD 008,RACERR,RACMG2        ;NO, PRINT ERROR MESSAGE
5222 022564' 104456                                TRAP  C$ERHRD
5223 022566' 000010                                .WORD 8
5224 022570' 001570'                                .WORD RACERR
5225 022572' 012526'                                .WORD RACMG2
5226 022574'                                50$:
5227 022574'                                ENDSEG
5228 022574'                                10003$:
5229 022574' 104405                                TRAP  C$ESEG
5230 022576'                                BGNSEG
5231 022576' 104404                                TRAP  C$BSEG
5232
5233                                ;CHECK BIT 15 = 0
5234                                ;
5235 022600' 012737 000017 000306'                MOV    #15.,BITNUM                ;TESTING BIT 15
5236 022606' 032777 100000 155524                BIT    #BIT15,@PCSR1            ;IS BIT 15 = 0?
5237 022614' 001404                                BEQ    60$                       ;YES
5238 022616'                                ERRHRD 009,RACERR,RACMG2        ;NO, PRINT ERROR MESSAGE
5239 022616' 104456                                TRAP  C$ERHRD
5240 022620' 000011                                .WORD 9
5241 022622' 001570'                                .WORD RACERR
5242 022624' 012526'                                .WORD RACMG2
5243 022626'                                60$:
5244 022626'                                ENDSEG
5245 022626'                                10004$:
5246 022626' 104405                                TRAP  C$ESEG
5247 022630'                                ENDSUB ;#2
5248 022630'                                L10103:
5249 022630' 104403                                TRAP  C$ESUB
5250 022632'                                CLRVEC #4                        ;FREE VECTOR 4
5251 022632' 012700 000004                                MOV    #4,RO
5252 022636' 104436                                TRAP  C$CVEC
5253 022640'                                ENDTST
5254 022640'                                L10101:
5255 022640' 104401                                TRAP  C$ETST
5256

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 113  
CZUAAB.MAC 07-APR-83 17:03 TEST 3: PCRS2 READ ACCESS TEST

5257  
5258  
5259  
5260  
5261  
5262  
5263  
5264  
5265  
5266  
5267  
5268  
5269  
5270  
5271  
5272  
5273  
5274  
5275  
5276  
5277  
5278  
5279  
5280  
5281  
5282  
5283  
5284  
5285  
5286  
5287  
5288  
5289  
5290  
5291  
5292  
5293  
5294  
5295  
5296  
5297  
5298  
5299  
5300  
5301  
5302  
5303  
5304

.SBTTL TEST 3: PCRS2 READ ACCESS TEST

\*\*\*\*\*  
: THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 2  
: CAN BE READ FROM THE UNIBUS  
: TEST SEQUENCE: 1-READ PCRS2  
\*\*\*\*\*

BGNTST

T3::  
SETVEC #4,#TRAP4,PRI07 ;SETUP TIMEOUT TRAP VECTOR  
MOV PRI07,-(SP)  
MOV #TRAP4,-(SP)  
MOV #4,-(SP)  
MOV #3,-(SP)  
TRAP CSSVEC  
ADD #10,SP  
CLR NEXMEM ;CLEAR NON-EXISTANT MEMORY FLAG  
MOV #2,CSNUM ;TESTING PCRS2  
TST @PCSR2 ;DOES PCRS2 EXIST?  
TST NEXMEM  
BEQ 10\$ ;YES  
ERRDF 010,RACERR,RACMG1 ;NO,PRINT FATAL ERROR MESSAGE  
TRAP CSERDF  
.WORD 10  
.WORD RACERR  
.WORD RACMG1  
CKLOOP ;LOOP BACK FROM HERE IF ERROR  
TRAP CSCLP1  
CLRVEC #4 ;RELEASE TRAP 4 VECTOR  
MOV #4,RO  
TRAP CSCVEC  
DODU UNIT ;DROP THE UNIT  
MOV UNIT,RO  
TRAP CSDDDU  
DOCLN ;ABORT SUBPASS  
TRAP CSDDLN  
10\$: CLRVEC #4 ;RELEASE TRAP 4 VECTOR  
MOV #4,RO  
TRAP CSCVEC  
L10104:  
TRAP CSETST

ENDTST

65HARDWARE TESTS MACV11 30A(1052) 07-APR-83 17:13 PAGE 114  
CZUAAB.MAC 07-APR-83 17:03 TEST 4: PCRS3 READ ACCESS TEST

5305  
5306  
5307  
5308  
5309  
5310  
5311  
5312  
5313  
5314  
5315  
5316  
5317  
5318  
5319  
5320  
5321  
5322  
5323  
5324  
5325  
5326  
5327  
5328  
5329  
5330  
5331  
5332  
5333  
5334  
5335  
5336  
5337  
5338  
5339  
5340  
5341  
5342  
5343  
5344  
5345  
5346  
5347  
5348  
5349  
5350  
5351  
5352  
5353  
5354  
5355  
5356

022754'  
022754'  
022754'  
022754' 012746 000340  
022760' 012746 022034'  
022764' 012746 000004  
022770' 012746 000003  
022774' 104437  
022776' 062706 000010  
023002'  
023002' 104404  
023004' 005037 000666'  
023010' 012737 000003 000304'  
023016' 005777 155322  
023022' 005737 000666'  
023026' 001414  
023030'  
023030' 104455  
023032' 000013  
023034' 001570'  
023036' 012500'  
023040'  
023040' 104406  
023042'  
023042' 012700 000004  
023046' 104436  
023050'  
023050' 013700 000330'  
023054' 104451  
023056'  
023056' 104444  
023060'  
023060'  
023060'  
023060' 104405  
023062'  
023062' 012700 000004  
023066' 104436  
023070'  
023070'  
023070' 104401

.SBTTL TEST 4: PCRS3 READ ACCESS TEST

:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 3  
:CAN BE READ FROM THE UNIBUS  
:TEST SEQUENCE: 1-READ PCRS3  
:\*\*\*\*\*

BGNTST

T4::  
SETVEC #4,#TRAP4,#PRI07' ;SETUP VECTOR FOR TIME-OUT  
MOV #PRI07,-(SP)  
MOV #TRAP4,-(SP)  
MOV #4,-(SP)  
MOV #3,-(SP)  
TRAP C\$SVEC  
ADD #10,SP

BGNSEG

TRAP C\$BSEG  
CLR NEXMEM ;CLEAR NON-EXISTANT MEMORY FLAG  
MOV #3,CSRNUM ;TESTING PCRS3  
TST @PCRS3 ;DOES PCRS3 EXIST?  
TST NEXMEM  
BEQ 10\$ ;YES  
ERRDF 011,RACERR,RACMG1 ;NO,PRINT FATAL ERROR MESSAGE  
TRAP C\$ERDF  
.WORD 11  
.WORD RACERR  
.WORD RACMG1  
CKLOOP ;LOOP BACK FROM HERE IF ERROR  
TRAP C\$CLP1  
CLRVEC #4 ;RELEASE TRAP 4 VECTOR  
MOV #4,R0  
TRAP C\$CVEC  
DODU UNIT ;DROP UNIT  
MOV UNIT,R0  
TRAP C\$DODU  
DOCLN ;ABORT SUB-PASS  
TRAP C\$DCLN

10\$:  
ENDSEG

10000\$:  
TRAP C\$ESEG  
CLRVEC #4 ;RETURN TIME-OUT TRAP  
MOV #4,R0  
TRAP C\$CVEC

ENDTST

L10105:  
TRAP C\$ETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 115  
CZUAAB.MAC 07-APR-83 17:03 TEST 5: RESET TEST

5357  
5358  
5359  
5360  
5361  
5362  
5363  
5364  
5365  
5366  
5367  
5368  
5369  
5370  
5371  
5372  
5373  
5374  
5375  
5376  
5377  
5378  
5379  
5380  
5381  
5382  
5383  
5384  
5385  
5386  
5387  
5388  
5389  
5390  
5391  
5392  
5393  
5394  
5395  
5396  
5397  
5398  
5399  
5400  
5401  
5402  
5403  
5404  
5405  
5406  
5407  
5408  
5409  
5410  
5411  
5412

.SBTTL TEST 5: RESET TEST

.....  
: THIS TEST WILL VERIFY THE RESET STATE FOR ALL DEUMA UNIBUS REGISTERS

: TEST SEQUENCE:  
1-WRITE A 1 TO PCSRO BIT 5  
2-READ PCSRO  
-VERIFY DNI SET  
-VERIFY INTR SET  
3-VERIFY PCSRO BITS 15:12 AND 10:08 AND 06:04 FOR LOGICAL 0 ;MAC001  
4-READ PCSR1  
5-VERIFY PCSR1 BITS 07:03 AND 14 AND 15 FOR LOGICAL 0 AND  
PORT STATE FIELD BITS 02:00 FOR READY STATE

.....

BGNTST

TS::  
: TESTING PCSRO FIRST  
: SETUP PART OF ERROR MESSAGE  
: FOR THE FOLLOWING TEST SEGMENTS

BGNSUB :#1

TS.1: TRAP CSBSUB

: SET THE RESET BIT IN PCSRO AND WAIT FOR 'DNI' TO SET

:MAC001 MOV #RSET,BPCSRO ;RESET DEUMA  
MOV #10\*SECOND,METER ;PUT SOME TIME ON THE METER  
MOV #12.\*SECOND,METER ;PUT PLENTY OF TIME ON THE METER;MAC001  
JSR PC,CHKDNI ;WAIT FOR DNI TO SET  
BCC 10\$ ;OK

:ERROR 'DNI' DID NOT SET  
:SETUP ERROR MESSAGE

ERRHRD 012,RSETER,MSG3 ;PRINT ERROR MESSAGE

TRAP CSERHRD  
.WORD 12  
.WORD RSETER  
.WORD MSG3

ESCAPE TST ;NO POINT IN CONTINUING TEST  
TRAP CSERHRD  
.WORD L10106-

10\$:  
ENDSUB :#1

L10107. TRAP CSBSUB

BGNSUB :#2

TS.2: TRAP CSBSUB

: CHECK ALL THE BITS AFFECTED BY RESET

```

5413 023164'          BGNSEG
5414 023164' 104404          TRAP      CSBSEG
5415
5416          :CHECK THE INTERRUPT SUMMARY BIT = 0
5417          :
5418 023166' 032777 000200 155142      BIT      #INTR,BPCSR0      :IS THE INTERRUPT SUMMARY BIT SET?
5419 023174' 001012          BNE      11$              :YES
5420
5421 023176' 012737 001032' 000310'      MOV      #SINTR,BITNAM      :ERROR, INTERRUPT SUMMARY NOT SET!
5422 023204' 012737 001277' 000312'      MOV      #SNSET,BITSTA      :PREPARE ERROR MESSAGE
5423 023212'          ERRHRD 013,RSETER,MSG3      :PRINT ERROR MESSAGE
5424 023212' 104456          TRAP      CSERHRD
5425 023214' 000015          .WORD 13
5426 023216' 001624'          .WORD RSETER
5427 023220' 013030'          .WORD MSG3
5428 023222'          11$:
5429 023222'          ENDSEG
5430 023222'          10000$:
5431 023222' 104405          TRAP      CSESEG
5432
5433          :CHECK BITS 15:12 AND 10:08 AND 06:04 OF PCSR0 FOR 0
5434          :
5435 023224' 012701 000360'      MOV      #BNART0,R1          :POINT TO BIT NAME MNEMONICS
5436 023230' 062701 000010      ADD      #4,R1              :INDEX DOWN TO BIT 4 MNEMONIC
5437          :MAC001 MOV      #BIT0,R2          :START TESTING AT LSB
5438 023234' 012702 000020      MOV      #BIT4,R2          :START TESTING AT BIT 4
5439 023240' 012737 001313' 000312'      MOV      #SNCLR,BITSTA      :TESTING FOR CLEARED BITS
5440 023246'          15$:
5441 023246'          BGNSEG
5442 023246' 104404          TRAP      CSBSEG
5443 023250' 030277 155062      BIT      R2,BPCSR0          :IS THIS BIT CLEARED?
5444 023254' 001406          BEQ      20$              :YES
5445
5446 023256' 011137 000310'      MOV      (R1),BITNAM        :ERROR BIT FAILED TO CLEAR
5447 023262'          ERRHRD 014,RSETER,MSG3      :GET MNEMONIC FOR THIS BIT
5448 023262' 104456          :PRINT ERROR MESSAGE
5449 023264' 000016          TRAP      CSERHRD
5450 023266' 001624'          .WORD 14
5451 023270' 013030'          .WORD RSETER
5452 023272'          .WORD MSG3
5453 023272'          20$:
5454 023272'          ENDSEG
5455 023272' 104405          10001$:
5456 023274' 005702          TRAP      CSESEG
5457 023276' 100411          21$:  TST      R2          :WE TESTED ALL BITS IN PCSR0?
5458 023300' 006302          BMI      25$              :YES
5459 023302' 062701 000002      ASL      R2              :POINT TO NEXT BIT TO TEST
5460 023306' 105702          ADD      #2,R1          :POINT TO NEXT MNEMONIC
5461 023310' 100771          TSTB    R2              :ARE WE POINTING TO 'INTR' BIT?
5462 023312' 022702 004000      BMI      21$              :YES, SKIP IT
5463 023316' 001766          CMP     #DNI,R2          :ARE WE POINTING TO 'DNI' BIT?
5464 023320' 000752          BEQ     21$              :YES SKIP IT
5465          BR      15$          :CONTINUE TESTING
5466 023322'          25$:
5467 023322'          BGNSEG
5468 023322' 104404          TRAP      CSBSEG

```

```

5469
5470      ;CHECK PCSR1 BITS 02:00 FOR THE READY STATE
5471
5472 023324' 012737 000001 000304'      MOV      #1,CSRNUM      :TESTING PCSR1
5473 023332' 017701 155002      MOV      @PCSR1,R1      :GET PCSR1 CONTENTS
5474 023336' 042701 177770      BIC      @CPSTATE,R1    :MASK ALL BUT PORT STATUS BITS
5475 023342' 022701 000002      CMP      @READY,R1      :IS THE DEUMA IN THE READY STATE?
5476 023346' 001404      BEQ      30$            :YES
5477
5478 023350'      ERRHRD 015,RSETER,MSG4 :ERROR, DEUMA NOT IN READY STATE!
5479 023350' 104456      :PRINT ERROR MESSAGE
5480 023352' 000017      TRAP     CSERHRD
5481 023354' 001624'      .WORD   15
5482 023356' 013076'      .WORD   RSETER
5483 023360'      .WORD   MSG4
5484 023360'
5485 023360'
5486 023360' 104405      10002$: TRAP     C$ESEG
5487
5488      ;CHECK PCSR1 BITS 07:03 FOR 0
5489
5490 023362' 012701 000420'      MOV      #BNAMT1,R1     :POINT TO MNEMONIC TABLE FOR PCSR1
5491 023366' 062701 000006      ADD      #3*2,R1        :INDEX PAST STATE BITS
5492 023372' 012737 001313' 000312'      MOV      #SNCLR,BITSTA  :PREPARE ERROR MESSAGE
5493 023400' 012737 001342' 000314'      MOV      #SAFTER,PWEN
5494 023406' 012702 000010      MOV      #BIT3,R2       :POINT TO BIT 3 TO START TESTING
5495 023412'
5496 023412'      40$:   BGNSEG
5497 023412' 104404      TRAP     CSBSEG
5498 023414' 030277 154720      BIT      R2,@PCSR1      :IS THIS BIT A 0?
5499 023420' 001406      BEQ      45$            :YES
5500
5501 023422' 011137 000310'      MOV      (R1),BITNAM    :ERROR, BIT IS NOT A 0 AFTER RESET!
5502 023426'      ERRHRD 016,RSETER,MSG3 :GET MNEMONIC FOR THIS BIT
5503 023426' 104456      :PRINT ERROR MESSAGE
5504 023430' 000020      TRAP     CSERHRD
5505 023432' 001624'      .WORD   16
5506 023434' 013030'      .WORD   RSETER
5507 023436'      .WORD   MSG3
5508 023436'
5509 023436'      45$:   ENDSEG
5510 023436' 104405      10003$: TRAP     C$ESEG
5511 023440' 105702      TRAP     CSBSEG
5512 023442' 100404      TSTB    R2              :WE TESTED BITS 07:03?
5513 023444' 006302      BMI     50$            :YES
5514 023446' 062701 000002      ASL     R2              :NO, POINT TO NEXT BIT
5515 023452' 000757      ADD     #2,R1           :POINT TO MNEMONIC FOR NEXT BIT
5516 023454'      BR      40$           :CONTINUE TESTING
5517 023454'
5518 023454' 104404      50$:   BGNSEG
5519
5520      ;CHECK PCSR1 BIT 14 = 0
5521
5522 023456' 032777 040000 154654      BIT      #14,@PCSR1    :IS PORT/CABLING BIT CLEAR?
5523 023464' 001412      BEQ     60$            :YES
5524
5524      :ERROR, PORT/CABLING BIT NOT CLEAR!

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 118  
CZUAAB.MAC 07-APR-83 17:03 TEST 5: RESET TEST

```

5525 023466' 012737 001313' 000312'      MOV      #SNCLR,BITSTA      :PREPARE ERROR MESSAGE
5526 023474' 012737 001076' 000310'      MOV      #SICAB,BITNAM     :GET MNEMONIC
5527 023502'          ERRHRD 017,RSETER,MSG3  :PRINT ERROR MESSAGE
5528 023502' 104456
5529 023504' 000021          TRAP      C$ERHRD
5530 023506' 001624'          .WORD    17
5531 023510' 013030'          .WORD    RSETER
5532 023512'          .WORD    MSG3
5533 023512'          60$:
5534 023512'          ENDSEG
5535 023512' 104405          10004$:
5536 023514'          BGNSEG          TRAP      C$ESEG
5537 023514' 104404          TRAP      C$BSEG
5538
5539          :CHECK PCSR1 BIT 15 = 0
5540
5541 023516' 032777 100000 154614      BIT      #XPWR,PCSR1      :IS TRANSCEIVER POWER BIT CLEAR?
5542 023524' 001412          BEQ      70$             :YES
5543
5544 023526' 012737 001313' 000312'      MOV      #SNCLR,BITSTA     :ERROR, TRANSCEIVER POWER BIT NOT CLEAR!
5545 023534' 012737 001065' 000310'      MOV      #SXPR,BITNAM     :PREPARE ERROR MESSAGE
5546 023542'          ERRHRD 018,RSETER,MSG3  :GET MNEMONIC
5547 023542' 104456          :PRINT ERROR MESSAGE
5548 023544' 000022          TRAP      C$ERHRD
5549 023546' 001624'          .WORD    18
5550 023550' 013030'          .WORD    RSETER
5551 023552'          .WORD    MSG3
5552 023552'          70$:
5553 023552'          ENDSEG
5554 023552' 104405          10005$:
5555 023554'          ENDSUB :#2          TRAP      C$ESEG
5556 023554'          L10110:
5557 023554' 104403          TRAP      C$ESUB
5558 023556'          ENDTST
5559 023556'          L10106:
5560 023556' 104401          TRAP      C$ETST

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 119  
CZUAAB.MAC 07-APR-83 17:03 TEST 6: PCSR2 REGISTER READ/WRITE TEST

5561  
5562  
5563  
5564  
5565  
5566  
5567  
5568  
5569  
5570  
5571  
5572  
5573  
5574  
5575  
5576  
5577  
5578  
5579  
5580  
5581  
5582  
5583  
5584  
5585  
5586  
5587  
5588  
5589  
5590  
5591  
5592  
5593  
5594  
5595  
5596  
5597  
5598  
5599  
5600  
5601  
5602  
5603  
5604  
5605  
5606  
5607  
5608

.SBTTL TEST 6: PCSR2 REGISTER READ/WRITE TEST  
:\*\*\*\*\*  
:THIS TEST WILL CHECK THE REGISTER FOR ALL SA0 AND SA1 ERRORS (STUCK AT  
:0 AND STUCK AT 1 ERRORS). THE HOST WILL WRITE PATTERNS TO THE REGISTER  
:AND READ THEM BACK TO VERIFY. THE PATTERNS TO BE USED ARE AT THE LABEL  
:PATERN:: IN THE GLOBAL DATA SECTION OF THIS PROGRAM.  
:NOTE: SINCE PCSR2 BIT 00 IS ALWAYS PRESET TO LOGIC 0, THE LOWEST ORDER  
:BIT OF THE PATTERN WILL BE MASKED BEFORE DOING THE COMPARISON.  
:TEST SEQUENCE:  
: 1-WRITE PATTERN TO REGISTER PCSR2  
: 2-COMPARE MASKED PATTERN WITH REGISTER PCSR2 CONTENTS  
: 3-REPEAT STEPS 1 TO 2 FOR ALL PATTERNS  
:\*\*\*\*\*

```
BGNTST
                                T6::
5580 023560'                   MOV    #2,CSRNUM           ;TESTING PCSR2
5581 023560'                   MOV    #PATERN,R1         ;GET ADDRESS OF DATA PATTERNS
5582 023560' 012737 000002 000304'  MOV    #5,R5           ;5 DATA PATTERNS
5583 023566' 012701 000520'       MOV    #5,R5           ;GET DATA PATTERN
5584 023572' 012705 000005       10$: MOV    (R1)+,R3
5585 023576' 012103              BGNSEG
5586 023600'                   TRAP   CSBSEG
5587 023600' 104404              MOV    R3,@PCSR2       ;WRITE PATTERN TO PCSR2
5588 023602' 010377 154534       MOV    @PCSR2,R4       ;READ PCSR2
5589 023606' 017704 154530       BIC    #BIT0,R3        ;MASK BIT 00
5590 023612' 042703 000001       CMP    R3,R4          ;COMPARE WHAT WAS WRITTEN TO...
5591 023616' 020304              ;WHAT WAS READ
5592                               ;COMPARED OK
5593 023620' 001404              BEQ    20$            ;PRINT ERROR MESSAGE
5594 023622'                   ERRHRD 019,RRWER,RACMG3
5595 023622' 104456              TRAP   CSERHRD
5596 023624' 000023              .WORD 19
5597 023626' 001646'           .WORD RRWER
5598 023630' 012570'           .WORD RACMG3
5599 023632'                   20$:
5600 023632'                   ENDSEG
5601 023632'                   10000$:
5602 023632' 104405              TRAP   CSESEG
5603 023634' 005305              DEC    R5              ;DONE ALL DATA PATTERNS?
5604 023636' 001357              BNE    10$            ;NO, CONTINUE TESTING
5605 023640'                   ENDTST
5606 023640'                   L10111:
5607 023640' 104401              TRAP   CSETST
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 120  
CZUAAB.MAC 07-APR-83 17:03 TEST 7: REGISTER PCRS3 READ/WRITE TEST

5609  
5610  
5611  
5612  
5613  
5614  
5615  
5616  
5617  
5618  
5619  
5620  
5621  
5622  
5623  
5624  
5625  
5626  
5627  
5628  
5629  
5630  
5631  
5632  
5633  
5634  
5635  
5636  
5637  
5638  
5639  
5640  
5641  
5642  
5643  
5644  
5645  
5646  
5647  
5648  
5649  
5650  
5651  
5652  
5653  
5654  
5655  
5656  
5657  
5658  
5659  
5660  
5661

023642'  
023642'  
023642' 012737 000003 000304'  
023650' 012703 000001  
023654'  
023654' 104404  
023656' 010377 154462  
023662' 017704 154456  
023666' 020304  
023670' 001404  
023672'  
023672' 104456  
023674' 000024  
023676' 001646'  
023700' 012570'  
023702'  
023702'  
023702' 104405  
023704' 006303  
023706'  
023706' 104404  
023710' 010377 154430  
023714' 017704 154424  
023720' 020304  
023722' 001404  
023724'  
023724' 104456  
023726' 000025  
023730' 001646'  
023732' 012570'  
023734'  
023734'  
023734'  
023734' 104405  
023736'  
023736'  
023736' 104401

.SBTTL TEST 7: REGISTER PCRS3 READ/WRITE TEST

.....  
:THIS TEST WILL WRITE PATTERNS TO THE WRITEABLE FIELD OF PCRS3  
:AND WILL READ THESE BACK FOR VERIFICATION.  
:TEST SEQUENCE:  
: 1-WRITE PATTERN 000001 TO PCRS3  
: 2-VERIFY PATTERN IN PCRS3  
: 3-WRITE PATTERN 000002 TO PCRS3  
: 4-VERIFY PATTERN IN PCRS3  
:.....

BGNTST

T7::  
MOV #3,CSRMUM :TESTING PCRS3  
MOV #BIT00,R3 :DATA PATTERN =1  
BGNSEG  
MOV R3,@PCRS3 :WRITE PATTERN TO PCRS3 TRAP CSBSEG  
MOV @PCRS3,R4 :READ PCRS3  
CMP R3,R4 :COMPARE  
BEQ 10\$ :IF == GOOD  
ERRHRD 020,RRWER,RACMG3 :ELSE PRINT ERROR MESSAGE  
TRAP CSERHRD  
.WORD 20  
.WORD RRWER  
.WORD RACMG3  
10\$:  
ENDSEG  
10000\$:  
ASL R3 :DATA PATTERN =2 TRAP CSESEG  
BGNSEG  
MOV R3,@PCRS3 :WRITE PATTERN TO PCRS3 TRAP CSBSEG  
MOV @PCRS3,R4 :READ PCRS3  
CMP R3,R4 :COMPARE  
BEQ 20\$ :IF = GOOD  
ERRHRD 021,RRWER,RACMG3 :ELSE PRINT ERROR MESSAGE  
TRAP CSERHRD  
.WORD 21  
.WORD RRWER  
.WORD RACMG3  
20\$:  
ENDSEG  
10001\$:  
ENDTST TRAP CSESEG  
L10112:  
TRAP CSETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 121  
CZUAAB.MAC 07-APR-83 17:03 TEST 8: NOP TEST

5662  
5663  
5664  
5665  
5666  
5667  
5668  
5669  
5670  
5671  
5672  
5673  
5674  
5675  
5676  
5677  
5678  
5679  
5680  
5681  
5682  
5683  
5684  
5685  
5686  
5687  
5688  
5689  
5690  
5691  
5692  
5693  
5694  
5695  
5696  
5697  
5698  
5699  
5700  
5701  
5702  
5703  
5704  
5705  
5706  
5707  
5708  
5709  
5710  
5711  
5712  
5713  
5714  
5715  
5716  
5717

023740'  
023740'  
023740'  
023740'  
023740' 104402  
  
023742' 013701 000336'  
023746' 011102  
023750' 032702 000100  
023754' 001412  
023756' 112711 000000  
023762' 011102  
023764' 032702 000100  
023770' 001404  
023772' 104456  
023774' 000026  
023776' 001706'  
024000' 012642'  
  
024002'  
024002'  
024002' 104403  
024004'  
024004' 104402

.SBTTL TEST 8: NOP TEST

\*\*\*\*\*  
: THIS TEST WILL VERIFY THAT THE DEUNA PROCESSOR IS ALIVE AND CAN  
: RESPOND TO A PORT COMMAND ISSUED. THE NOP PORT COMMAND WILL BE ISSUED  
: TO THE DEUNA IN PCSRO BITS 3:0 AND WILL WAIT FOR THE 'DNI' BIT TO  
: SET IN PCSRO.  
: THE NOP PORT COMMAND USES A MINIMUM OF HARDWARE BUT FORCES THE T11  
: TO EXECUTE THE PORT COMMAND SEQUENCE.  
: TEST SEQUENCE:  
: 1-READ REGISTER PCSRO  
: 2-VERIFY INTERRUPT ENABLE BIT 06 = LOGIC 0  
: 3-VERIFY DONE INTERRUPT BIT 11 = LOGIC 0  
: 4-WRITE PORT COMMAND NOP TO PCSRO BITS 3:0  
: 5-READ REGISTER PCSRO  
: 6-VERIFY DONE INTERRUPT BIT 11 = LOGIC 1  
: 7-WRITE REGISTER PCSRO BIT 11 WITH LOGIC 1  
: 8-READ REGISTER PCSRO  
: 9-VERIFY PCSRO BIT 11 = 0  
\*\*\*\*\*

BGNTST

BGNSUB :#1

T8::

T8.1:

TRAP CSBSUB

:CHECK THE INTERUPT ENABLE BIT; IF SET CLEAR IT

MOV PCSRO,R1 :GET PCSRO ADDRESS  
MOV (R1),R2 :GET CONTENTS  
BIT #INTE,R2 :IS INTERRUPT ENABLE SET?  
BEQ 10\$ :NO  
MOVB #0,(R1) :CLEAR INTERRUPT ENABLE  
MOV (R1),R2 :GET PCSRO CONTENTS  
BIT #INTE,R2 :IS INTERRUPT ENABLE CLEAR NOW?  
BEQ 10\$ :YES  
ERRHRD 022,NOPERR,RACMG4 :NO,PRINT ERROR MESSAGE

TRAP CSERHRD  
.WORD 22  
.WORD NOPERR  
.WORD RACMG4

10\$:  
ENDSUB :#2

L10114:

TRAP CSesub

BGNSUB :#2

T8.2:

TRAP CSBSUB

:CHECK THE DONE INTERRUPT BIT; IF SET WRITE ONE TO CLEAR IT

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 122  
 CZUAAB.MAC 07-APR-83 17:03 TEST 8: NOP TEST

5718	024006'	032702	004000		BIT	#DNI,R2		:IS DONE INTERRUPT CLEAR?			
5719	024012'	001413			BEQ	20\$		:YES			
5720	024014'	000302			SWAB	R2		:NO,ORIENT UPPER & LOWER BYTES			
5721	024016'	110261	000001		MOVB	R2.1(R1)		:SO WE CAN CLEAR UPPER BITS			
5722	024022'	011102			MOV	(R1),R2		:GET PCSRO CONTENTS			
5723	024024'	032702	004000		BIT	#DNI,R2		:IS DNI CLEAR NOW?			
5724	024030'	001404			BEQ	20\$		:YES			
5725	024032'				ERRHRD	023,NOPERR,RACMG7		:NO,PRINT ERROR MESSAGE			
5726	024032'	104456							TRAP	C\$ERHRD	
5727	024034'	000027							.WORD	23	
5728	024036'	001706'							.WORD	NOPERR	
5729	024040'	012670'							.WORD	RACMG7	
5730	024042'			20\$:							
5731	024042'			ENDSUB	:	#2					
5732	024042'										
5733	024042'	104403							L10115:		
5734	024044'			BGNSUB	:	#3			TRAP	C\$ESUB	
5735	024044'										
5736	024044'	104402							T8.3:		
5737									TRAP	C\$BSUB	
5738				:	:ISSUE NOP PORT COMMAND AND CHECK FOR DNI						
5739				:							
5740	024046'	012777	000006	154262	MOV	#PNOP,@PCSRO		:ISSUE NOP PORT COMMAND			
5741	024054'	012737	000176	000332'	MOV	#2*SECOND,METER		:SETUP TIMER			
5742	024062'	004737	017316'		JSR	PC,CHKDNI		:WAIT FOR DNI TO SET			
5743	024066'	103022			BCC	30\$		:OK			
5744								:ERROR DNI DID NOT SET!			
5745								:SETUP TO PRINT ERROR MESSAGE			
5746	024070'	012737	001000'	000310'	MOV	#SDNI,BITNAM		:POINT TO 'DNI' ASCII STRING			
5747	024076'	012737	001277'	000312'	MOV	#SNSET,BITSTA		:POINT TO 'NOT SET' ASCII STRING			
5748	024104'	012737	001342'	000314'	MOV	#SAFTER,PWHEN		:POINT TO 'AFTER' ASCII STRING			
5749	024112'	012737	001405'	000316'	MOV	#SNOP,PCOMND		:POINT TO 'NOP' ASCII STRING			
5750	024120'				ERRHRD	024,NOPERR,MSG1		:PRINT ERROR MESSAGE			
5751	024120'	104456							TRAP	C\$ERHRD	
5752	024122'	000030							.WORD	24	
5753	024124'	001706'							.WORD	NOPERR	
5754	024126'	012716'							.WORD	MSG1	
5755	024130'				ESCAPE	TST		:DON'T CONTINUE TEST IF ERROR OCCURRED	TRAP	C\$ESCAPE	
5756	024130'	104410							.WORD	L10113-	
5757	024132'	000026									
5758	024134'			30\$:							
5759	024134'			BGNSEG							
5760	024134'	104404							TRAP	C\$BSEG	
5761				:	:WRITE ONE TO CLEAR 'DNI'						
5762				:							
5763				:							
5764	024136'	004737	017362'		JSR	PC,CLRDN1		:GO CLEAR DNI BIT			
5765	024142'	103004			BCC	40\$		:IT CLEARED OK			
5766								:ERROR DNI DID NOT CLEAR!			
5767	024144'				ERRHRD	025,NOPERR,RACMG7		:NO,PRINT ERROR MESSAGE			
5768	024144'	104456							TRAP	C\$ERHRD	
5769	024146'	000031							.WORD	25	
5770	024150'	001706'							.WORD	NOPERR	
5771	024152'	012670'							.WORD	RACMG7	
5772	024154'			40\$:							
5773	024154'			ENDSEG							

5774 024154'  
5775 024154' 104405  
5776 024156'  
5777 024156'  
5778 024156' 104403  
5779 024160'  
5780 024160'  
5781 024160' 104401  
5782

ENDSUB ;#3

ENDTST

10008: TRAP CSESEG  
L10116: TRAP CSESUB  
L10113: TRAP CSETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 124  
 CZUAAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

5783  
5784  
5785  
5786  
5787  
5788  
5789  
5790  
5791  
5792  
5793  
5794  
5795  
5796  
5797  
5798  
5799  
5800  
5801  
5802  
5803  
5804  
5805  
5806  
5807  
5808  
5809  
5810  
5811  
5812  
5813  
5814  
5815  
5816  
5817  
5818  
5819  
5820  
5821  
5822  
5823  
5824  
5825  
5826  
5827  
5828  
5829  
5830  
5831  
5832  
5833  
5834  
5835  
5836  
5837  
5838

.SBTTL TEST 9: SELF TEST

\*\*\*\*\*  
 THIS TEST VERIFIES THAT THE ROM BASED SELF TEST  
 CAN BE RUN SUCCESSFULLY WHEN INVOKED VIA  
 THE SELF TEST PORT COMMAND.

TEST SEQUENCE:

1. CLEAR OUTSTANDING INTERRUPTS
  2. ISSUE THE SELF TEST PORT COMMAND
  3. WAIT FOR DNI
  4. CHECK LITE BITE REGISTER FOR SUCCESSFUL SELF TEST
  5. WRITE ONE TO CLEAR DNI
- \*\*\*\*\*

BGNTST

```

T9::
JSR PC,CLRINT      ;CLEAR ANY OUTSTANDING INTERRUPTS
BCC 10$           ;OK
ERRHRD 026,SLFTST ;ERROR OCCURRED TRYING TO CLEAR
                                TRAP C$ERRHRD
                                .WORD 26
                                .WORD SLFTST
                                .WORD 0
ESCAPE TST        ;LEAVE THIS TEST
                                TRAP C$ESCAPE
                                .WORD L10117-.
30$: MOV #12,SECOND,METER ;PUT SOME TIME ON THE METER
MOV #SLFT,@PCSR0  ; RUN SELF TEST
JSR PC,CHKDNI    ; DNI ?
BCC 30$         ; YES
                ;ERROR DNI FAILED TO SET!
MOV #SDNI,BITNAM
MOV #SNSET,BITSTA
MOV #SAFTER,PWHEN
MOV #SSLFT,PCOMND
ERRHRD 027,SLFTST,MSG1 ;PRINT ERROR MESSAGE
                                TRAP C$ERRHRD
                                .WORD 27
                                .WORD SLFTST
                                .WORD MSG1
;MAC001 ESCAPE TST ; AND ABORT TEST
30$: MOV @PCSR1,R3 ;GET PCSR1 CONTENTS
BIT #SFT,R3      ;WAS SELF TEST SUCCESSFUL?
BEQ 40$         ;YES
COM R3          ;GET TEST NUMBER CORRECT POLARITY
BIC #^CSFT,R3   ;MASK ALL BUT SELF TEST FIELD
MOV #8.,R1      ;SHIFT RESULT OVER 8 BIT POSTIONS
35$: ASR R3
DEC R1
BNE 35$
MOV R3,R4      ;SAVE ERROR NUMBER
ASL R3         ;MAKE NUMBER A BYTE OFFSET
    
```

6SHARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 125  
CZUAAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

514

5839	024324'	062703	024410'		ADD	#STTBL,R3	:	INDEX INTO SELF TEST TABLE	
5840	024330'	105773	000000		TSTB	B(R3)	:	IS ERROR CODE UNUSED?	:MAC001
5841	024334'	001004			BNE	368	:	NO, CODE IS OK	:MAC001
5842	024336'	012737	027112'	024406'	MOV	#UNUSED,STMSG	:	BOGUS ERROR CODE MESSAGE	:MAC001
5843	024344'	000402			BR	378	:		:MAC001
5844	024346'	011337	024406'		MOV	(R3),STMSG	:	LOAD INTO SELF TEST MESSAGE	
5845	024352'			368:	ERRHRD	028,SLFTST,MSG34	:	REPORT SELF TEST FAILURE	
5846	024352'	104456						TRAP	C\$ERHRD
5847	024354'	000034						.WORD	28
5848	024356'	001726'						.WORD	SLFTST
5849	024360'	015622'						.WORD	MSG34
5850	024362'				ESCAPE	TST	:	AND ABORT TEST	
5851	024362'	104410						TRAP	C\$ESCAPE
5852	024364'	000020						.WORD	L10117-
5853									
5854	024366'	004737	017362'		JSR	PC,CLRDNI	:	WRITE ONE TO CLEAR DNI	
5855				408:			:	ERROR?	
5856	024372'	103004			BCC	508	:	NO	
5857							:	ERROR DNI FAILED TO CLEAR!	
5858	024374'				ERRHRD	029,SLFTST,RACMG7	:	PRINT ERROR MESSAGE	
5859	024374'	104456						TRAP	C\$ERHRD
5860	024376'	000035						.WORD	29
5861	024400'	001726'						.WORD	SLFTST
5862	024402'	012670'						.WORD	RACMG7
5863	024404'			508:					
5864	024404'				ENDTST				
5865	024404'							L10117:	
5866	024404'	104401						TRAP	C\$ETST



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 126  
CZUAAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

5867  
5868  
5869  
5870 024406' 000000  
5871  
5872  
5873  
5874 024410' 024610'  
5875 024412' 024632'  
5876 024414' 024652'  
5877 024416' 024656'  
5878 024420' 024706'  
5879 024422' 024742'  
5880 024424' 024766'  
5881 024426' 025030'  
5882 024430' 025073'  
5883 024432' 025140'  
5884 024434' 025160'  
5885 024436' 025166'  
5886 024440' 025213'  
5887 024442' 025214'  
5888 024444' 025215'  
5889 024446' 025216'  
5890 024450' 025217'  
5891 024452' 025233'  
5892 024454' 025234'  
5893 024456' 025235'  
5894 024460' 025236'  
5895 024462' 025237'  
5896 024464' 025240'  
5897 024466' 025241'  
5898 024470' 025242'  
5899 024472' 025304'  
5900 024474' 025347'  
5901 024476' 025413'  
5902 024500' 025450'  
5903 024502' 025512'  
5904 024504' 025546'  
5905 024506' 025610'  
5906 024510' 025644'  
5907 024512' 025721'  
5908 024514' 025773'  
5909 024516' 026046'  
5910 024520' 026112'  
5911 024522' 026163'  
5912 024524' 026226'  
5913 024526' 026227'  
5914 024530' 026230'  
5915 024532' 026307'  
5916 024534' 026363'  
5917 024536' 026440'  
5918 024540' 026506'  
5919 024542' 026561'  
5920 024544' 026626'  
5921 024546' 026627'  
5922 024550' 026630'

: LOCAL STORAGE FOR TEST 9  
: STMSG:: .WORD 0 ; SELF TEST MESSAGE ADDRESS  
: SELF TEST MESSAGE TABLE  
: STTBL:: .WORD SMSG00  
.WORD SMSG01  
.WORD SMSG02  
.WORD SMSG03  
.WORD SMSG04  
.WORD SMSG05  
.WORD SMSG06  
.WORD SMSG07  
.WORD SMSG10  
.WORD SMSG11  
.WORD SMSG12  
.WORD SMSG13  
.WORD SMSG14  
.WORD SMSG15  
.WORD SMSG16  
.WORD SMSG17  
.WORD SMSG20  
.WORD SMSG21  
.WORD SMSG22  
.WORD SMSG23  
.WORD SMSG24  
.WORD SMSG25  
.WORD SMSG26  
.WORD SMSG27  
.WORD SMSG30  
.WORD SMSG31  
.WORD SMSG32  
.WORD SMSG33  
.WORD SMSG34  
.WORD SMSG35  
.WORD SMSG36  
.WORD SMSG37  
.WORD SMSG40  
.WORD SMSG41  
.WORD SMSG42  
.WORD SMSG43  
.WORD SMSG44  
.WORD SMSG45  
.WORD SMSG46  
.WORD SMSG47  
.WORD SMSG50  
.WORD SMSG51  
.WORD SMSG52  
.WORD SMSG53  
.WORD SMSG54  
.WORD SMSG55  
.WORD SMSG56  
.WORD SMSG57  
.WORD SMSG60

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 127  
 CZUAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

5923	024552'	026644'				.WORD	SMSG61	
5924	024554'	026670'				.WORD	SMSG62	
5925	024556'	026714'				.WORD	SMSG63	
5926	024560'	026734'				.WORD	SMSG64	
5927	024562'	026740'				.WORD	SMSG65	
5928	024564'	026752'				.WORD	SMSG66	
5929	024566'	026764'				.WORD	SMSG67	
5930	024570'	027000'				.WORD	SMSG70	
5931	024572'	027012'				.WORD	SMSG71	
5932	024574'	027036'				.WORD	SMSG72	
5933	024576'	027060'				.WORD	SMSG73	
5934	024600'	027061'				.WORD	SMSG74	
5935	024602'	027062'				.WORD	SMSG75	
5936	024604'	027063'				.WORD	SMSG76	
5937	024606'	027064'				.WORD	SMSG77	
5938								
5939	024610'	042516	042526	020122		:ASCII MESSAGES		
5940	024616'	047507	020124	052123		SMSG00::	.ASCIZ /NEVER GOT STARTED/	
5941	024624'	051101	042524	000104				
5942	024632'	050103	020125	047111		SMSG01::	.ASCIZ /CPU INSTRUCTION/	:MAC001
5943	024640'	052123	052522	052103				
5944	024646'	047511	000116					
5945	024652'	047522	000115			SMSG02::	.ASCIZ /ROM/	:MAC001
5946	024656'	051127	052111	040505		SMSG03::	.ASCIZ /WRITEABLE CONTROL STORE/	:MAC001
5947	024664'	046102	020105	047503				
5948	024672'	052116	047522	020114				
5949	024700'	052123	051117	000105				
5950						:MAC001SMSG04::	.ASCIZ /PHYSICAL ADDRESS ROM/	
5951	024706'	030524	020061	047125		SMSG04::	.ASCIZ /T11 UNIBUS ADDRESS REGISTER/	:MAC001
5952	024714'	041111	051525	040440				
5953	024722'	042104	042522	051523				
5954	024730'	051040	043505	051511				
5955	024736'	042524	000122					
5956	024742'	042522	042503	053111		SMSG05::	.ASCIZ /RECEIVER UNIBUS DMA/	:MAC001
5957	024750'	051105	052440	044516				
5958	024756'	052502	020123	046504				
5959	024764'	000101						
5960	024766'	041520	051123	020061		SMSG06::	.ASCIZ /PCSR1 LOWER BYTE AND T11 DMA READ/	:MAC001
5961	024774'	047514	042527	020122				
5962	025002'	054502	042524	040440				
5963	025010'	042116	052040	030461				
5964	025016'	042040	040515	051040				
5965	025024'	040505	000104					
5966	025030'	041520	051123	020060		SMSG07::	.ASCIZ /PCSR0 UPPER BYTE AND T11 DMA WRITE/	:MAC001
5967	025036'	050125	042520	020122				
5968	025044'	054502	042524	040440				
5969	025052'	042116	052040	030461				
5970	025060'	042040	040515	053440				
5971	025066'	044522	042524	000				
5972	025073'	120	051503	030122		SMSG10::	.ASCIZ /PCSR0 LOWER BYTE AND LINK MEMORY DMA/	:MAC001
5973	025100'	046040	053517	051105				
5974	025106'	041040	052131	020105				
5975	025114'	047101	020104	044514				
5976	025122'	045516	046440	046505				
5977	025130'	051117	020131	046504				
5978	025136'	000101						

65 HARDWARE TESTS		MACY11	30A(1052)	07-APR-83	17:13	PAGE 128		
CZUAAB.MAC		07-APR-83	17:03	TEST 9: SELF TEST				
5979	025140'	041520	051123	020062	SMSG11::	.ASCIZ	/PCSR2 AND PCSR3/	:MAC001
5980	025146'	047101	020104	041520				
5981	025154'	051123	000063					
5982	025160'	044524	042515	000122	SMSG12::	.ASCIZ	/TIMER/	:MAC001
5983	025166'	044120	051531	041511	SMSG13::	.ASCIZ	/PHYSICAL ADDRESS ROM/	:MAC001
5984	025174'	046101	040440	042104				
5985	025202'	042522	051523	051040				
5986	025210'	046517	000					
5987	025213'	000			SMSG14::	.BYTE	0	:UNUSED
5988	025214'	000			SMSG15::	.BYTE	0	:UNUSED
5989	025215'	000			SMSG16::	.BYTE	0	:UNUSED
5990	025216'	000			SMSG17::	.BYTE	0	:UNUSED
5991	025217'	114	047111	020113	SMSG20::	.ASCIZ	/LINK MEMORY/	:MAC001
5992	025224'	042515	047515	054522				
5993	025232'	000						
5994	025233'	000			SMSG21::	.BYTE	0	:UNUSED
5995	025234'	000			SMSG22::	.BYTE	0	:UNUSED
5996	025235'	000			SMSG23::	.BYTE	0	:UNUSED
5997	025236'	000			SMSG24::	.BYTE	0	:UNUSED
5998	025237'	000			SMSG25::	.BYTE	0	:UNUSED
5999	025240'	000			SMSG26::	.BYTE	0	:UNUSED
6000	025241'	000			SMSG27::	.BYTE	0	:UNUSED
6001	025242'	047514	040503	020114	SMSG30::	.ASCIZ	/LOCAL LOOPBACK - XMITTER TIMEOUT/	:MAC001
6002	025250'	047514	050117	040502				
6003	025256'	045503	026440	020040				
6004	025264'	046530	052111	042524				
6005	025272'	020122	044524	042515				
6006	025300'	052517	000124					
6007	025304'	047514	040503	020114	SMSG31::	.ASCIZ	/LOCAL LOOPBACK - RECEIVER TIMEOUT/	:MAC001
6008	025312'	047514	050117	040502				
6009	025320'	045503	026440	020040				
6010	025326'	042522	042503	053111				
6011	025334'	051105	052040	046511				
6012	025342'	047505	052125	000				
6013	025347'	114	041517	046101	SMSG32::	.ASCIZ	/LOCAL LOOPBACK - BUFFER COMPARSION/	:MAC001
6014	025354'	046040	047517	041120				
6015	025362'	041501	020113	020055				
6016	025370'	041040	043125	042506				
6017	025376'	020122	047503	050115				
6018	025404'	051101	044523	047117				
6019	025412'	000						
6020	025413'	114	041517	046101	SMSG33::	.ASCIZ	/LOCAL LOOPBACK - BYTE COUNT/	:MAC001
6021	025420'	046040	047517	041120				
6022	025426'	041501	020113	020055				
6023	025434'	041040	052131	020105				
6024	025442'	047503	047125	000124				
6025	025450'	047514	040503	020114	SMSG34::	.ASCIZ	/LOCAL LOOPBACK - RECEIVER STATUS/	:MAC001
6026	025456'	047514	050117	040502				
6027	025464'	045503	026440	020040				
6028	025472'	042522	042503	053111				
6029	025500'	051105	051440	040524				
6030	025506'	052524	000123					
6031	025512'	047514	040503	020114	SMSG35::	.ASCIZ	/LOCAL LOOPBACK - CRC ERROR/	:MAC001
6032	025520'	047514	050117	040502				
6033	025526'	045503	026440	020040				
6034	025534'	051103	020103	051105				

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 129  
 CZUAAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

6035	025542'	047522	000122				
6036	025546'	047514	040503	020114	SMSG36::	.ASCIZ /LOCAL LOOPBACK - MATCH BIT ERROR/	;MAC001
6037	025554'	047514	050117	040502			
6038	025562'	045503	026440	020040			
6039	025570'	040515	041524	020110			
6040	025576'	044502	020124	051105			
6041	025604'	047522	000122				
6042	025610'	047514	040503	020114	SMSG37::	.ASCIZ /LOCAL LOOPBACK - TDR ERROR/	;MAC001
6043	025616'	047514	050117	040502			
6044	025624'	045503	026440	020040			
6045	025632'	042124	020122	051105			
6046	025640'	047522	000122				
6047	025644'	046530	052111	042524	SMSG40::	.ASCIZ /XMITTER BUFFER ADDRESS - TRANSMITTER TIMEOUT/	;MAC001
6048	025652'	020122	052502	043106			
6049	025660'	051105	040440	042104			
6050	025666'	042522	051523	026440			
6051	025674'	052040	040522	051516			
6052	025702'	044515	052124	051105			
6053	025710'	052040	046511	047505			
6054	025716'	052125	000				
6055	025721'	130	044515	052124	SMSG41::	.ASCIZ /XMITTER BUFFER ADDRESS - RECEIVER TIMEOUT/	;MAC001
6056	025726'	051105	041040	043125			
6057	025734'	042506	020122	042101			
6058	025742'	051104	051505	020123			
6059	025750'	020055	042522	042503			
6060	025756'	053111	051105	052040			
6061	025764'	046511	047505	052125			
6062	025772'	000					
6063	025773'	130	044515	052124	SMSG42::	.ASCIZ /XMITTER BUFFER ADDRESS - BUFFER COMPARISON/	;MAC001
6064	026000'	051105	041040	043125			
6065	026006'	042506	020122	042101			
6066	026014'	051104	051505	020123			
6067	026022'	020055	052502	043106			
6068	026030'	051105	041440	046517			
6069	026036'	040520	051522	047511			
6070	026044'	000116					
6071	026046'	046530	052111	042524	SMSG43::	.ASCIZ /XMITTER BUFFER ADDRESS - BYTE COUNT/	;MAC001
6072	026054'	020122	052502	043106			
6073	026062'	051105	040440	042104			
6074	026070'	042522	051523	026440			
6075	026076'	041040	052131	020105			
6076	026104'	047503	047125	000124			
6077	026112'	046530	052111	042524	SMSG44::	.ASCIZ /XMITTER BUFFER ADDRESS - RECEIVER STATUS/	;MAC001
6078	026120'	020122	052502	043106			
6079	026126'	051105	040440	042104			
6080	026134'	042522	051523	026440			
6081	026142'	051040	041505	044505			
6082	026150'	042526	020122	052123			
6083	026156'	052101	051525	000			
6084	026163'	130	044515	052124	SMSG45::	.ASCIZ /XMITTER BUFFER ADDRESS - CRC ERROR/	;MAC001
6085	026170'	051105	041040	043125			
6086	026176'	042506	020122	042101			
6087	026204'	051104	051505	020123			
6088	026212'	020055	051103	020103			
6089	026220'	051105	047522	000122			
6090	026226'	000			SMSG46::	.BYTE 0 ;UNUSED	;MAC001

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 130  
 CZUAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

6091	026227'	000			SMSG47::	.BYTE 0	:UNUSED	:MAC001
6092	026230'	042522	042503	053111	SMSG50::	.ASCIZ /RECEIVER BUFFER ADDRESS -	TRANSMITTER TIMEOUT/	:MAC001
6093	026236'	051105	041040	043125				
6094	026244'	042506	020122	042101				
6095	026252'	051104	051505	020123				
6096	026260'	020055	052040	040522				
6097	026266'	051516	044515	052124				
6098	026274'	051105	052040	046511				
6099	026302'	047505	052125	000				
6100	026307'	122	041505	044505	SMSG51::	.ASCIZ /RECEIVER BUFFER ADDRESS -	RECEIVER TIMEOUT/	:MAC001
6101	026314'	042526	020122	052502				
6102	026322'	043106	051105	040440				
6103	026330'	042104	042522	051523				
6104	026336'	026440	020040	042522				
6105	026344'	042503	053111	051105				
6106	026352'	052040	046511	047505				
6107	026360'	052125	000					
6108	026363'	122	041505	044505	SMSG52::	.ASCIZ /RECEIVER BUFFER ADDRESS -	BUFFER COMPARSION/	:MAC001
6109	026370'	042526	020122	052502				
6110	026376'	043106	051105	040440				
6111	026404'	042104	042522	051523				
6112	026412'	026440	020040	052502				
6113	026420'	043106	051105	041440				
6114	026426'	046517	040520	051522				
6115	026434'	047511	000116					
6116	026440'	042522	042503	053111	SMSG53::	.ASCIZ /RECEIVER BUFFER ADDRESS -	BYTE COUNT/	:MAC001
6117	026446'	051105	041040	043125				
6118	026454'	042506	020122	042101				
6119	026462'	051104	051505	020123				
6120	026470'	020055	041040	052131				
6121	026476'	020105	047503	047125				
6122	026504'	000124						
6123	026506'	042522	042503	053111	SMSG54::	.ASCIZ /RECEIVER BUFFER ADDRESS -	RECEIVER STATUS/	:MAC001
6124	026514'	051105	041040	043125				
6125	026522'	042506	020122	042101				
6126	026530'	051104	051505	020123				
6127	026536'	020055	051040	041505				
6128	026544'	044505	042526	020122				
6129	026552'	052123	052101	051525				
6130	026560'	000						
6131	026561'	122	041505	044505	SMSG55::	.ASCIZ /RECEIVER BUFFER ADDRESS -	CRC ERROR/	:MAC001
6132	026566'	042526	020122	052502				
6133	026574'	043106	051105	040440				
6134	026602'	042104	042522	051523				
6135	026610'	026440	020040	051103				
6136	026616'	020103	051105	047522				
6137	026624'	000122						
6138	026626'	000			SMSG56::	.BYTE 0	:UNUSED	:MAC001
6139	026627'	000			SMSG57::	.BYTE 0	:UNUSED	:MAC001
6140	026630'	052522	052116	050040	SMSG60::	.ASCIZ /RUNT PACKET/		:MAC001
6141	026636'	041501	042513	000124				
6142	026644'	044515	044516	052515	SMSG61::	.ASCIZ /MINIMUM PACKET SIZE/		:MAC001
6143	026652'	020115	040520	045503				
6144	026660'	052105	051440	055111				
6145	026666'	000105						
6146	026670'	040515	044530	052515	SMSG62::	.ASCIZ /MAXIMUM PACKET SIZE/		:MAC001

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 131  
 CZUAAB.MAC 07-APR-83 17:03 TEST 9: SELF TEST

6147	026676'	020115	040520	045503				
6148	026704'	052105	051440	055111				
6149	026712'	000105						
6150	026714'	053117	051105	044523	SMSG63::	.ASCIZ	/OVERSIZE PACKET/	:MAC001
6151	026722'	042532	050040	041501				
6152	026730'	042513	000124					
6153	026734'	051103	000103		SMSG64::	.ASCIZ	/CRC/	:MAC001
6154	026740'	047503	046114	051511	SMSG65::	.ASCIZ	/COLLISION/	:MAC001
6155	026746'	047511	000116					
6156	026752'	042510	051101	041124	SMSG66::	.ASCIZ	/HEARTBEAT/	:MAC001
6157	026760'	040505	000124					
6158	026764'	040510	043114	042040	SMSG67::	.ASCIZ	/HALF DUPLEX/	:MAC001
6159	026772'	050125	042514	000130				
6160	027000'	052515	052114	041511	SMSG70::	.ASCIZ	/MULTICAST/	:MAC001
6161	027006'	051501	000124					
6162	027012'	042101	051104	051505	SMSG71::	.ASCIZ	/ADDRESS RECOGNITION/	:MAC001
6163	027020'	020123	042522	047503				
6164	027026'	047107	052111	047511				
6165	027034'	000116						
6166	027036'	054105	042524	047122	SMSG72::	.ASCIZ	/EXTERNAL LOOPBACK/	:MAC001
6167	027044'	046101	046040	047517				
6168	027052'	041120	041501	000113				
6169	027060'	000			SMSG73::	.BYTE	0	:UNUSED
6170	027061'	000			SMSG74::	.BYTE	0	:UNUSED
6171	027062'	000			SMSG75::	.BYTE	0	:UNUSED
6172	027063'	000			SMSG76::	.BYTE	0	:UNUSED
6173	027064'	047503	050115	042514	SMSG77::	.ASCIZ	/COMPLETED - NO ERRORS/	:MAC001
6174	027072'	042524	020104	020055				
6175	027100'	047516	042440	051122				
6176	027106'	051117	000123					
6177	027112'	047125	042504	044506	UNUSED::	.ASCIZ	/UNDEFINED/	:MAC001
6178	027120'	042516	000104					
6179						.EVEN		

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 132  
CZUAAB.MAC 07-APR-83 17:03 TEST 10: DEUNA ROM DUMP TEST

6180  
6181  
6182  
6183  
6184  
6185  
6186  
6187  
6188  
6189  
6190  
6191  
6192  
6193  
6194  
6195  
6196  
6197  
6198  
6199  
6200  
6201  
6202  
6203  
6204  
6205  
6206  
6207  
6208  
6209  
6210  
6211  
6212  
6213  
6214  
6215  
6216  
6217  
6218  
6219  
6220  
6221  
6222  
6223  
6224  
6225  
6226  
6227  
6228  
6229  
6230  
6231  
6232  
6233  
6234  
6235

027124'  
027124'  
027124'  
027124'  
027124' 104402

000606' 151204  
151202  
000001 151166  
000176 000332'  
004737 017316'  
103022

```

.SBTTL TEST 10: DEUNA ROM DUMP TEST
:*****
:THIS TEST WILL VERIFY THAT THE DATA PATH FROM THE T11 PROCESSOR
:TO THE UNIBUS INTERFACE IS INTACT AND ABLE TO TRANSFER DATA RELIABLY.
:THIS DATA PATH IS CRUCIAL FOR FURTHER TESTING BECAUSE IT IS NECESSARY
:FOR LOADING REPAIR-LEVEL DIAGNOSTICS INTO THE WCS.
:
:THE TEST STRATEGY IS TO TRANSFER KNOWN DATA OVER THE DATA PATH AND TO
:VERIFY THE TRANSFERRED DATA.
:
:THE DATA SOURCE FOR THE DUMP TEST IS THE ROM MICROCODE RESIDENT ON THE
:DEUNA PORT BOARD. A DUMP OF THE ROM WILL EXERCISE THE DATA PATH NEEDED
:FOR LOADING WCS AND THE ROM CONTENTS CAN BE VERIFIED. THE ROM MICROCODE
:WILL BE CHECKED BY VERIFYING THE CRC BYTES. THE CRC BYTES CHARACTERIZE
:THE DATA CONTENTS OF THE ROM AND ARE BURNED INTO THE ROM AT THE TIME OF
:MANUFACTURE. A FAILURE TO VERIFY THE CRC CALCULATION ON THE DUMPED ROM
:DATA DUMP WILL BE INTERPRETED AS AN ERROR IN THE DATA PATH.
:
:TEST SEQUENCE:
:
:1-WRITE PCSR2 AND PCSR3 WITH THE ADDRESS OF THE PORT CONTROL BLOCK
:2-WRITE <GET PCBB> PORT COMMAND TO PCSRO
:3-READ PCSRO AND VERIFY DNI BIT SET
:4-WRITE PCSRO DNI BIT TO RESET IT
:5-FILL MEMORY BUFFER WITH A BACKGROUND PATTERN
:6-WRITE PORT CONTROL BLOCK WITH 'DUMP INTERNAL MEMORY' FUNCTION CODE
:7-WRITE PORT CONTROL BLOCK WITH UNIBUS DATA BLOCK BASE ADDRESS
:8-WRITE DATA BLOCK LENGTH TO UNIBUS DATA BLOCK
:9-WRITE MEMORY BUFFER ADDRESS TO UNIBUS DATA BLOCK
:10-WRITE INTERNAL DATA BLOCK ADDRESS TO UNIBUS DATA BLOCK
:11-WRITE <GET CMD> PORT COMMAND TO PCSRO
:12-READ PCSRO AND VERIFY DNI SET
:13-CALCULATE CRC ON DUMPED DATA
:14-REPEAT STEPS 5-13 ON EACH 1K OF ROM
:15-VERIFY CRC =0
:*****
BGMTST
BGNSUB ;#1
T10::
T10.1: TRAP CSBSUB
:
:LOAD PCSR2+3; ISSUE GET PORT CONTROL BLOCK PORT COMMAND; AND WAIT FOR 'DNI'
:
BGNSEG
TRAP CSB EG
MOV #PCBB,PCSR2 ;LOAD PCSR2+3 WITH PORT CONTROL BLOCK..
CLR #PCSR3 ;ADDRESS
MOV #GETPCB,PCSR0 ;ISSUE GET PORT CONTROL BLOCK COMMAND
MOV #2*SECOND,METER ;PUT SOME TIME IN THE METER
JSR PC,CHKDNI ;WAIT FOR DNI TO SET
BCC 20$ ;OK

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 133  
 CZUAAB.MAC 07-APR-83 17:03 TEST 10: DEUNA ROM DUMP TEST

```

6236
6237 027164' 012737 001000' 000310'          MOV    #SDNI,BITNAM          ;ERROR DNI DID NOT SET!
6238 027172' 012737 001277' 000312'          MOV    #SNSET,BITSTA        ;NO, FORMAT ERROR MESSAGE
6239 027200' 012737 001342' 000314'          MOV    #AFTER,PWHEN
6240 027206' 012737 001350' 000316'          MOV    #SGIPCB,PCOMND
6241 027214'          ERRHRD  030,ROMDMP,MSG1 ;PRINT ERROR MESSAGE
6242 027214' 104456
6243 027216' 000036          TRAP   CSERHRD
6244 027220' 001747'          .WORD 30
6245 027222' 012716'          .WORD ROMDMP
6246 027224'          .WORD MSG1
6247 027224' 104410          ESCAPE TST          ;CAN NOT CONTINUE TESTING
6248 027226' 000310          TRAP   C$ESCAPE
6249 027230'          .WORD L10120-.
6250 027230'          20$:
6251 027230'          ENDSEG
6252 027230' 104405          10000$:
6253 027232'          BGNSEG          TRAP   C$ESEG
6254 027232' 104404          TRAP   C$BSEG
6255
6256          ;WRITE ONE TO CLEAR 'DNI'
6257          ;
6258 027234' 004737 017362'          JSR    PC,CLRDN1          ;GO CLEAR DNI
6259 027240' 103004          BCC    25$              ;OK
6260
6261 027242'          ERRHRD  031,ROMDMP,RACMG7 ;ERROR DNI FAILED TO CLEAR!
6262 027242' 104456          ;PRINT ERROR MESSAGE
6263 027244' 000037          TRAP   CSERHRD
6264 027246' 001747'          .WORD 31
6265 027250' 012670'          .WORD ROMDMP
6266 027252'          .WORD RACMG7
6267 027252'          25$:
6268 027252'          ENDSEG
6269 027252' 104405          10001$:
6270 027254'          ENDSUB ;#1          TRAP   C$ESEG
6271 027254'          L10121:
6272 027254' 104403          TRAP   C$ESUB
6273 027256'          BGNSUB ;#2
6274 027256'
6275 027256' 104402          T10.2:
6276          TRAP   C$BSUB
6277          ;FILL BUFFER WITH PATTERN; FORMAT PCBB AND UDBB; ISSUE GET COMMAND PORT COMMAND
6278          ;AND WAIT FOR 'DNI'
6279          ;
6280 027260' 012737 000020 000606'          MOV    #DIM,PCBB          ;LOAD PCB WITH DUMP INTERNAL MEMORY FUNCTION
6281 027266' 012737 000616' 000610'          MOV    #UDBB,PCBB+2      ;LOAD UNIBUS DATA BLOCK BASE ADDRESS...
6282 027274' 005037 000612'          CLR    PCBB+4            ;INTO PCB
6283 027300' 012737 040000 000624'          MOV    #ROMADR,UDBB+6    ;INTERNAL BASE ADDRESS OF ROM
6284 027306' 012705 000010          MOV    #8.,R5            ;DUMP ROM IN 8. 1K CHUNKS
6285 027312' 005004          CLR    R4                ;INITIAL CRC VALUE
6286 027314'          26$:
6287 027314'          BGNSEG
6288 027314' 104404          TRAP   C$BSEG
6289 027316' 013701 000324'          MOV    FREMEM,R1         ;GET POINTER TO BUFFER
6290 027322' 012702 002000          MOV    #SIZ1K/2,R2      ;GET 1K WORD COUNT
6291 027326' 012703 000520'          MOV    #PAT1,R3         ;GET A BACKGROUND PATTERN

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 134  
 CZUAAB.MAC 07-APR-83 17:03 TEST 10: DEUNA ROM DUMP TEST

```

6292 027332' 011321          27$:  MOV      (R3),(R1)+      ;FILL BUFFER WITH BACKGROUND PATTERN
6293 027334' 005302          DEC      R2
6294 027336' 001375          BNE     27$
6295 027340' 012737 004000 000616'  MOV     #POMSIZ/8.,UDBB      ;1K BYTE COUNT
6296 027346' 013737 000324' 000620'  MOV     FREMEM,UDBB+2      ;AND BUFFER ADDRESS
6297 027354' 005037 000622'      CLR     UDBB+4
6298 027360' 012737 000176 000332'  MOV     #2*SECOND,METER    ;PUT SOME TIME ON THE METER
6299 027366' 012777 000002 150742  MOV     #GETCMD,BPCSR0     ;LOAD 'GET COMMAND' PORT COMMAND
6300 027374' 004737 017316'      JSR     PC,CHKDNI          ;WAIT FOR DNI TO SET
6301 027400' 103022          BCC     30$                ;OK
6302                                     ;ERROR, DNI DID NOT SET!
6303 027402' 012737 001000' 000310'  MOV     #$DNI,BITNAM      ;FORMAT ERROR MESSAGE
6304 027410' 012737 001277' 000312'  MOV     #$NSET,BITSTA
6305 027416' 012737 001342' 000314'  MOV     #$AFTER,PWHEM
6306 027424' 012737 001357' 000316'  MOV     #SGTCMD,PCOMND
6307 027432'          ERRHRD  C32,ROMDMP,MSG1
6308 027432' 104456
6309 027434' 000040          TRAP   C$ERHRD
6310 027436' 001747'          .WORD  32
6311 027440' 012716'          .WORD  ROMDMP
6312 027442'          .WORD  MSG1
6313 027442' 104410          TRAP   C$ESCAPE
6314 027444' 000072          .WORD  L10120-.
6315 027446'
6316 027446'
6317 027446'
6318 027446' 104405          10000$: TRAP   C$ESEG
6319 027450'          BGNSEG
6320 027450' 104404          TRAP   C$BSEG
6321
6322 ;WRITE ONE TO CLEAR 'DNI'
6323 ;
6324 027452' 004737 017362'      JSR     PC,CLRDN1          ;GO CLEAR DNI BIT
6325 027456' 103004          BCC     33$                ;OK
6326                                     ;ERROR DNI FAILED TO CLEAR
6327 027460'          ERRHRD  033,ROMDMP,RACMG7 ;PRINT ERROR MESSAGE
6328 027460' 104456          TRAP   C$ERHRD
6329 027462' 000041          .WORD  33
6330 027464' 001747'          .WORD  ROMDMP
6331 027466' 012670'          .WORD  RACMG7
6332 027470'
6333 027470'          33$:  ENDSEG
6334 027470'
6335 027470' 104405          10001$: TRAP   C$ESEG
6336
6337 ;CALCULATE CRC ON 1K OF DATA DUMPED FROM ROM
6338 ;
6339 027472' 013700 000324'      MOV     FREMEM,R0          ;GET BUFFER ADDRESS
6340 027476' 012702 004000      MOV     #SIZ1K,R2          ;GET BYTE COUNT
6341 027502' 004737 017030'      JSR     PC,CRC16          ;CALC CRC ON 1K BUFFER
6342 027506' 062737 004000 000624'  ADD     #PROMSIZ/8.,UDBB+6 ;POINT TO NEXT 1K OF ROM
6343 027514' 005305          DEC     R5                 ;HAVE WE DONE ALL 8K?
6344 027516' 001276          BNE     26$                ;NO
6345 027520' 005704          TST     R4                 ;YES, IS CRC = 0?
6346 027522' 001404          BEQ     40$                ;YES, OK
6347 027524'          ERRHRD  034,ROMDMP,MSG2 ;PRINT ERROR MESSAGE

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 135  
CZUAAB.MAC 07-APR-83 17:03 TEST 10: DEUNA ROM DUMP TEST

6348 027524' 104456  
6349 027526' 000042  
6350 027530' 001747'  
6351 027532' 012764'  
6352 027534'  
6353 027534'  
6354 027534'  
6355 027534' 104403  
6356 027536'  
6357 027536'  
6358 027536' 104401  
6359  
6360  
6361

40\$:  
ENDSUB ;#2

ENDTST

TRAP C\$ERHD  
.WORD 34  
.WORD ROMDMP  
.WORD MSG2

L10122:

TRAP C\$ESUB

L10120:

TRAP C\$ETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 136  
CZUAAB.MAC 07-APR-83 17:03 TEST 10: DEUNA ROM DUMP TEST

6362  
6363  
6364  
6365  
6366  
6367  
6368  
6369  
6370  
6371  
6372  
6373  
6374  
6375  
6376  
6377  
6378  
6379  
6380  
6381  
6382  
6383  
6384  
6385  
6386  
6387  
6388  
6389  
6390  
6391  
6392  
6393  
6394  
6395  
6396  
6397  
6398  
6399  
6400  
6401  
6402  
6403  
6404  
6405  
6406  
6407  
6408  
6409  
6410  
6411  
6412  
6413  
6414  
6415  
6416  
6417

.SBTTL TEST 11: WCS LOAD/DUMP TEST

\*\*\*\*\*  
: THIS TEST WILL USE THE LOAD/DUMP PORT COMMAND TO VERIFY THE DATA  
: PATHWAY TO/FROM THE WCS. PATTERNS WILL BE USED TO CHECK THE DATA PATHWAY  
: FOR ALL SA0 AND SA1 ERRORS.  
:  
: BECAUSE THE OPERATIONAL MICROCODE NEEDS THE LOWER 2K OF WCS ONLY THE TOP HALF  
: OF WCS WILL BE LOADED WITH A DATA PATTERN THEN DUMPED BACK  
: TO MEMORY FOR VERIFICATION. THIS PROCEDURE WILL BE REPEATED FOR ALL PATTERNS.  
:  
: TEST SEQUENCE:  
: 1-FORMAT UNIBUS DATA BLOCK WITH NUMBER OF WORDS, WCS DESTINATION  
: ADDRESS, AND SOURCE BUFFER ADDRESS.  
: 2-FILL SOURCE BUFFER WITH DATA PATTERN  
: 3-WRITE PORT CONTROL BLOCK WITH ADDRESS OF UNIBUS DATA BLOCK  
: AND WITH THE 'LOAD INTERNAL MEMORY' FUNCTION CODE  
: 4-WRITE PCSR2 AND PCSR3 WITH PORT CONTROL BLOCK ADDRESS  
: 5-WRITE PCSR0 WITH <GET CMD> PORT COMMAND  
: 6-READ PCSR0 TO VERIFY DNI SET  
: 7-WRITE ONE TO CLEAR DNI  
: 8-FILL DESTINATION BUFFER WITH ZEROS  
: 9-WRITE PORT CONTROL BLOCK WITH 'DUMP INTERNAL MEMORY'  
: FUNCTION CODE.  
: 10-WRITE PCSR0 WITH <GET CMD> PORT COMMAND  
: 11-VERIFY DNI SET  
: 12-WRITE ONE TO CLEAR DNI  
: 13-COMPARE DESTINATION BUFFER WITH DATA PATTERN  
: 14-REPEAT STEPS 1-13 FOR ALL DATA PATTERNS  
: \*\*\*\*\*

BGNTST

027540'  
027540'  
027540' 012704 000520'  
027544'  
027544'  
027544' 104402  
027546'  
027546' 104404  
027550' 013702 000324'  
027554' 012703 004000  
027560' 011422  
027562' 005303  
027564' 001375

MOV #PATERN,R4 ;POINT TO LIST OF DATA PATTERNS T11::  
BGNSUB ;#1 ; T11.1: TRAP C\$BSUB  
:  
:LOAD WCS  
:  
BGNSEG ; TRAP C\$BSEG  
:  
:FILL SOURCE BUFFER; FORMAT UDBB AND PCBB; ISSUE GET PORT CONTROL BLOCK PORT  
:COMMAND AND WAIT FOR 'DNI'  
:  
MOV FREMEM,R2 ;GET UNIBUS BUFFER ADDRESS  
MOV #WCSSIZ/4,R3 ;GET HALF SIZE OF WCS IN WORDS  
10S: MOV (R4),(R2)+ ;FILL BUFFER WITH DATA PATTERN  
DEC R3  
BNE 10S  
:  
:SETUP UNIBUS DATA BLOCK

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 137  
 CZUAAB.MAC 07-APR-83 17:03 TEST 11: WCS LOAD/DUMP TEST

6418	027566'	012737	010000	000616'	MOV	#WCSSIZ/2,UDBB	:BYTE COUNT	
6419	027574'	013737	000324'	000620'	MOV	FREMEM,UDBB+2	:BUFFER ADDRESS	
6420	027602'	005037	000622'		CLR	UDBB+4		
6421	027606'	012737	010000	000624'	MOV	#WCSADR+<WCSSIZ/2>,UDBB+6	:BASE ADDRESS OF TOP HALF OF WCS	
6422							:SETUP PORT CONTROL BLOCK	
6423	027614'	012737	000021	000606'	MOV	#LIM,PCBB	: 'LOAD INTERNAL MEMORY' FUNCTION	
6424	027622'	012737	000616'	000610'	MOV	#UDBB,PCBB+2	:ADDRESS OF UNIBUS DATA BLOCK	
6425	027630'	005037	000612'		CLR	PCBB+4		
6426	027634'	012777	000606'	150500	MOV	#PCBB,@PCSR2	:LOAD PCSR2+3 WITH PCB ADDRESS	
6427	027642'	005077	150476		CLR	@PCSR3		
6428								
6429	027646'	012777	000001	150462	MOV	#GETPCB,@PCSR0	:ISSUE 'GET PORT CONTROL BLOCK' PORT COMMAND	
6430	027654'	012737	000176	000332'	MOV	#2*SECOND,METER	:PUT SOME TIME ON THE METER	
6431	027662'	004737	017316'		JSR	PC,CHKDNI	:WAIT FOR DNI TO SET	
6432	027666'	103022			BCC	20\$	:OK	
6433							:ERROR DNI DID NOT SET!	
6434	027670'	012737	001000'	000310'	MOV	#\$DNI,BITNAM	:POINT TO 'DNI' ASCII STRING	
6435	027676'	012737	001277'	000312'	MOV	#\$NSET,BITSTA	:POINT TO 'NOT SET' ASCII STRING	
6436	027704'	012737	001342'	000314'	MOV	#\$AFTER,PWHEN	:POINT TO 'AFTER' ASCII STRING	
6437	027712'	012737	001350'	000316'	MOV	#\$GTPCB,PCOMND	:POINT TO 'GET PORT CONTROL BLOCK' ASCII STRING	
6438	027720'				ERRHRD	035,DATAID,MSG1	:PRINT ERROR MESSAGE	
6439	027720'	104456					TRAP	C\$ERHRD
6440	027722'	000043					.WORD	35
6441	027724'	001774'					.WORD	DATAID
6442	027726'	012716'					.WORD	MSG1
6443	027730'				ESCAPE	TST		
6444	027730'	104410					TRAP	C\$ESCAPE
6445	027732'	000372					.WORD	L10123-
6446	027734'				20\$:			
6447	027734'				ENDSEG			
6448	027734'						10000\$:	
6449	027734'	104405					TRAP	C\$ESEG
6450	027736'				BGNSEG			
6451	027736'	104404					TRAP	C\$BSEG
6452								
6453					:	WRITE ONE TO CLEAR 'DNI'		
6454					:			
6455	027740'	004737	017362'		JSR	PC,CLRDN1	:GO CLEAR DNI	
6456	027744'	103004			BCC	25\$	:OK	
6457							:ERROR DNI FAILED TO CLEAR	
6458	027746'				ERRHRD	036,DATAID,RACMG7	:PRINT ERROR MESSAGE	
6459	027746'	104456					TRAP	C\$ERHRD
6460	027750'	000044					.WORD	36
6461	027752'	001774'					.WORD	DATAID
6462	027754'	012670'					.WORD	RACMG7
6463	027756'				25\$:			
6464	027756'				ENDSEG			
6465	027756'						10001\$:	
6466	027756'	104405					TRAP	C\$ESEG
6467	027760'				BGNSEG			
6468	027760'	104404					TRAP	C\$BSEG
6469					:			
6470					:	ISSUE GET COMMAND PORT COMMAND AND WAIT FOR 'DNI'		
6471					:			
6472	027762'	012777	000002	150346	MOV	#GETCMD,@PCSR0	:ISSUE GET COMMAND PORT COMMAND	
6473	027770'	012737	000275	000332'	MOV	#3*SECOND,METER	:PUT SOME TIME ON THE METER	

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 138  
CZUAAB.MAC 07-APR-83 17:03 TEST 11: WCS LOAD/DUMP TEST

```

6474 027776' 004737 017316'      JSR    PC,CHKDNI      ;GO WAIT FOR DNI
6475 030002' 103022      BCC    30$           ;OK
6476                                ;ERROR DNI DID NOT SET!
6477 030004' 012737 001000' 000310'  MOV    #'DNI,BITNAM
6478 030012' 012737 001277' 000312'  MOV    #SNSET,BITSTA
6479 030020' 012737 001342' 000314'  MOV    #SAFTER,PWHEN
6480 030026' 012737 001357' 000316'  MOV    #SGTCMD,PCOMND
6481 030034'                                ERRHRD 037,DATAID,MSG1      ;PRINT ERROR MESSAGE
6482 030034' 104456      TRAP   C$ERHRD
6483 030036' 000045      .WORD 37
6484 030040' 001774'      .WORD DATAID
6485 030042' 012716'      .WORD MSG1
6486 030044'                                ESCAPE TST
6487 030044' 104410      TRAP   C$ESCAPE
6488 030046' 000256      .WORD L10123-.
6489 030050'                                30$:
6490 030050'      ENDSEG
6491 030050'                                10002$:
6492 030050' 104405      TRAP   C$ESEG
6493 030052'                                BGNSEG
6494 030052' 104404      TRAP   C$BSEG
6495                                ;
6496                                ;WRITE ONE TO CLEAR 'DNI'
6497                                ;
6498 030054' 004737 017362'      JSR    PC,CLRDN1      ;GO CLEAR DNI
6499 030060' 103004      BCC    35$           ;OK
6500                                ;ERROR DNI FAILED TO CLEAR
6501                                ;PRINT ERROR MESSAGE
6502 030062' 104456      TRAP   C$ERHRD
6503 030064' 000046      .WORD 38
6504 030066' 001774'      .WORD DATAID
6505 030070' 012670'      .WORD RACMG7
6506 030072'                                35$:
6507 030072'      ENDSEG
6508 030072'                                10003$:
6509 030072' 104405      TRAP   C$ESEG
6510 030074'                                ENDSUB ;#1
6511 030074'                                L10124:
6512 030074' 104403      TRAP   C$ESUB
6513 030076'                                BGNSUB ;#2
6514 030076'                                T11.2:
6515 030076' 104402      TRAP   C$BSUB
6516                                ;
6517                                ;DUMP WCS
6518                                ;
6519 030100'                                BGNSEG
6520 030100' 104404      TRAP   C$BSEG
6521                                ;
6522                                ;CLEAR DESTINATION BUFFER; FORMAT PCBB AND UDBB; ISSUE GET COMMAND PORT COMMAND
6523                                ;AND WAIT FOR 'DNI'
6524                                ;
6525 030102' 013702 000324'      MOV    FREMEM,R2      ;GET ADDRESS OF FREE MEMORY
6526 030106' 012703 004000      MOV    #WCSSIZ/4,R3  ;SIZE IN WORDS OF BUFFER
6527 030112' 005022      40$: CLR    (R2)+        ;FILL DESTINATION BUFFER WITH ZEROS
6528 030114' 005303      DEC    R3
6529 030116' 001375      BNE   40$

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 139  
CZUAB.MAC 07-APR-83 17:03 TEST 11: WCS LOAD/DUMP TEST

```

6530
6531 030120' 013737 000324' 000620'      MOV    FREMEM,UDBB+2
6532 030126' 005037 000622'      CLR    UDBB+4
6533 030132' 012737 000020 000606'      MOV    #DIM,PCBB           ;LOAD 'DUMP INTERNAL MEMORY' FUNCTION
6534 030140' 012777 000002 150170      MOV    #GE:CMD,BPCSR0     ;ISSUE GET COMMAND PORT COMMAND
6535 030146' 012737 000275 000332'      MOV    #3*SECOND,METER   ;PUT SOME TIME ON THE METER
6536 030154' 004737 017316'      JSR    PC,CHKDNI         ;WAIT FOR DNI TO SET
6537 030160' 103022      BCC    50$              ;OK
6538 030162' 012737 001000' 000310'      MOV    #SDNI,BITNAM
6539 030170' 012737 001277' 000312'      MOV    #SNSET,BITSTA
6540 030176' 012737 001342' 000314'      MOV    #SAFTER,PWHEM
6541 030204' 012737 001357' 000316'      MOV    #SGTCMD,PCOMND
6542 030212'      ERRHRD 039,DATAID,MSG1
6543 030212' 104456      TRAP  CSERHRD
6544 030214' 000047      .WORD 39
6545 030216' 001774'      .WORD DATAID
6546 030220' 012716'      .WORD MSG1
6547 030222'      ESCAPE TST
6548 030222' 104410      TRAP  CSESCAPE
6549 030224' 000100      .WORD L10123-.
6550 030226'      50$: ;
6551 030226'      ENDSEG
6552 030226'      10000$:
6553 030226' 104405      TRAP  CSESEG
6554 030230'      BGNSEG
6555 030230' 104404      TRAP  CSBSEG
6556 030232' 004737 017362'      JSR    PC,CLRDNI         ;GU CLEAR DNI
6557 030236' 103004      BCC    60$              ;OK
6558      ;ERROR DNI FAILED TO CLEAR
6559 030240'      ERRHRD 040,DATAID,RACMG7 ;PRINT ERROR MESSAGE
6560 030240' 104456      TRAP  CSERHRD
6561 030242' 000050      .WORD 40
6562 030244' 001774'      .WORD DATAID
6563 030246' 012670'      .WORD RACMG7
6564 030250'      60$:
6565 030250'      ENDSEG
6566 030250'      10001$:
6567 030250' 104405      TRAP  CSESEG
6568      ;
6569      ;COMPARE DUMPED DATA TO WRITTEN PATTERN
6570      ;
6571 030252' 013701 000324'      MOV    FREMEM,R1         ;SOURCE BUFFER ADDRESS
6572 030256' 012703 004000      MOV    #<WCS1Z/2>/2,R3  ;# OF WORDS TO COMPARE
6573 030262' 021114      CMP    (R1),(R4)        ;IS WHAT WAS LOADED SAME AS...
6574      ;WHAT WAS DUMPED?
6575 030264' 001404      BEQ    80$              ;YES
6576 030266'      ERRHRD 041,DATAID,MSG5 ;ERROR DATA COMPARE
6577 030266' 104456      TRAP  CSERHRD
6578 030270' 000051      .WORD 41
6579 030272' 001774'      .WORD DATAID
6580 030274' 013124'      .WORD MSG5
6581 030276' 005721      80$: TST    (R1)+        ;POINT TO NEXT LOCATION
6582 030300' 005303      DEC    R3               ;WE DONE ALL WORDS?
6583 030302' 001367      BNE    70$              ;NOT YET
6584 030304'
6585 030304'      ENDSUB ;#2

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 140  
CZUAAB.MAC 07-APR-83 17:03 TEST 11: WCS LOAD/DUMP TEST

6586 030304' 104403

TRAP CSESUB

6587

6588

6589

:CKECK TO SEE IF ALL PATTERNS HAVE BEEN RUN THROUGH

6590 030306' 062704 000002

ADD #2,K4  
CMP R4,#PAT6  
BEQ 908  
JMP T11.1

:POINT TO NEXT DATA PATTERN  
:WE DONE ALL DATA PATTERNS?  
:YES END OF TEST  
:NO CONTINUE WITH NEW DATA PATTERN

6591 030312' 020427 000532'

6592 030316' 001402

6593 030320' 000137 027544'

908:  
ENDTST

6594 030324'

6595 030324'

6596 030324'  
6597 030324' 104401

L10123:  
TRAP CSETST

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 141  
CZUAAB.MAC 07-APR-83 17:03 TEST 11: WCS LOAD/DUMP TEST

6598  
6599  
6600  
6601  
6602  
6603  
6604  
6605  
6606  
6607  
6608  
6609  
6610  
6611  
6612  
6613  
6614  
6615  
6616  
6617  
6618  
6619  
6620  
6621  
6622  
6623  
6624  
6625  
6626  
6627  
6628  
6629  
6630  
6631  
6632  
6633  
6634  
6635  
6636  
6637  
6638  
6639  
6640  
6641  
6642  
6643  
6644  
6645  
6646  
6647  
6648  
6649  
6650  
6651  
6652  
6653

.SBTTL TEST 12: LOAD AND START FUNCTION TEST  
:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE LOAD AND START MICROADDRESS PORT COMMAND  
:IS OPERATIONAL.  
:THE PROCESS IS TO LOAD WCS WITH MICROCODE THAT WHEN STARTED WILL WRITE  
:A PATTERN OF DATA TO THE LITE-BYTE FIELD OF PCSR1 REGISTER WHICH CAN BE READ  
:FROM THE UNIBUS AND BE VERIFIED.  
:NOTE: THIS TEST USES MICROCODE MODULE 'A'  
:TEST SEQUENCE:  
: 1-FORMAT UNIBUS DATA BLOCK WITH BYTE COUNT, WCS DESTINATION ADDRESS  
: AND UNIBUS SOURCE ADDRESS OF THE MICROCODE.  
: 2-WRITE PORT CONTROL BLOCK WITH ADDRESS OF UNIBUS DATA BLOCK AND  
: LOAD INTERNAL MEMORY FUNCTION CODE.  
: 3-WRITE PCSR2 AND PCSR3 WITH ADDRESS OF THE PORT CONTROL BLOCK.  
: 4-WRITE PCSRO WITH <GET PCB> PORT COMMAND  
: 5-READ PCSRO AND VERIFY 'DNI'  
: 6-WRITE PCSRO 'DNI' TO CLEAR  
: 7-WRITE PCSRO WITH <GET COMMAND> PORT COMMAND  
: 8-READ PCSRO AND VERIFY 'DNI'  
: 9-WRITE PCSRO 'DNI' TO CLEAR  
: 10-WRITE PORT CONTROL BLOCK WITH START ADDRESS OF WCS MICROCODE  
: AND WITH LOAD AND START FUNCTION CODE  
: 11-WRITE PCSRO WITH <GET COMMAND> PORT COMMAND  
: 12-READ PCSRO AND VERIFY 'DNI' SET  
: 13-READ PCSR1 AND VERIFY BITS 13:8 ARE PATTERN WRITTEN BY MICROCODE  
: 14-ISSUE RESET TO PCSRO TO RESTART OPERATIONAL MICROCODE  
: 15-READ PCSRO AND VERIFY 'DNI' SET  
: 16-WRITE PCSRO WITH 'DNI' TO CLEAR  
:\*\*\*\*\*

030326'  
030326'  
030326'  
030326' 104404

BGNTST  
T12::  
:LOAD MICROCODE MODULE 'A' INTO THE TOP HALF OF WCS  
:BGNSEG TRAP CSBSEG  
:FORMAT THE UNIBUS DATA BLOCK AND THE PORT CONTROL BLOCK FOR THE LOAD  
:INTERNAL MEMORY FUNCTION  
:SETUP UDBB  
MOV MICASZ,UDBB :SIZE OF MICROCODE MODULE TO LOAD  
MOV #MICROA,UDBB+2 :BASE ADDRESS OF MICROCODE MODULE  
CLR UDBB+4  
MOV #WCSADR+<WCSSIZ/2>,UDBB+6 :LOAD INTO TOP HALF OF WCS  
:SETUP PCB  
MOV #LIM,PCBB :'LOAD INTERNAL MEMORY' FUNCTION



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 142  
 CZUAAB.MAC 07-APR-83 17:03 TEST 12: LOAD AND START FUNCTION TEST

```

6654 030364' 012737 000616' 000610'      MOV    #UDBB,PCBB+2      ;SET ADDRESS OF UDBB
6655 030372' 005037 000612'          CLR    PCBB+4
6656 030376' 012777 000606' 147736      MOV    #PCBB,BPCSR2     ;TELL DEUNA WHERE PCBB IS
6657 030404' 005077 147734          CLR    BPCSR3
6658
6659 030410' 012777 000001 147720      MOV    #GETPCB,BPCSR0   ;ISSUE 'GET PCB' PORT COMMAND
6660 030416' 012737 000176 000332'      MOV    #2*SECOND,METER ;SETUP TIMER
6661 030424' 004737 017316'          JSR    PC,CHKDNI        ;WAIT FOR 'DNI' TO SET
6662 030430' 103022          BCC    10$              ;OK
6663
6664 030432' 012737 001000' 000310'      MOV    #SDNI,BITNAM     ;ERROR DNI NOT SET!
6665 030440' 012737 001277' 000312'      MOV    #SNSET,BITSTA    ;SETUP ERROR MESSAGE
6666 030446' 012737 001342' 000314'      MOV    #SAFTER,PWHEN
6667 030454' 012737 001350' 000316'      MOV    #SGTPCB,PCOMND
6668 030462'          ERRHRD 042,LASFT,MSG1      ;PRINT ERROR MESSAGE
6669 030462' 104456
6670 030464' 000052          TRAP  C$ERHRD
6671 030466' 002026'          .WORD 42
6672 030470' 012716'          .WORD  LASFT
6673 030472'          .WORD  MSG1
6674 030472' 104410          ESCAPE TST              ;NO POINT IN CONTINUING TEST
6675 030474' 000310          TRAP  C$ESCAPE
6676 030476'          .WORD  L10126-.
6677 030476'
6678 030476'
6679 030476' 104405          10000$: TRAP  C$ESEG
6680 030500'          BGNSEG                  TRAP  C$BSEG
6681 030500' 104404          TRAP  C$BSEG
6682
6683
6684
6685 030502' 004737 017362'          ;WRITE ONE TO CLEAR 'DNI'
6686 030506' 103006          ;
6687
6688
6689 030510'          JSR    PC,CLRDKI        ;GO CLEAR 'DNI'
6690 030512' 000053          BCC    20$              ;OK
6691 030514' 002026'          ;ERROR 'DNI' NOT CLEAR!
6692 030516' 012670'          ERRHRD 043,LASFT,RACMG7 ;PRINT ERROR MESSAGE
6693 030520'          TRAP  C$ERHRD
6694 030520' 104410          .WORD 43
6695 030522' 000262          .WORD  LASFT
6696 030524'          .WORD  RACMG7
6697 030524'          ESCAPE TST              ;DO NOT CONTINUE TEST
6698 030524'          TRAP  C$ESCAPE
6699 030524' 104405          .WORD  L10126-.
6700 030526'          10001$: TRAP  C$ESEG
6701 030526' 104404          TRAP  C$BSEG
6702
6703
6704
6705
6706 030530' 012777 000002 147600      ;NOW THAT THE UNA KNOWS WHERE THE MICROCODE IS, ISSUE THE GET COMMAND PORT
6707 030536' 012737 000176 000332'      ;COMMAND SO THE MICROCODE CAN GET LOADED INTO WCS
6708 030544' 004737 017316'          ;
6709 030550' 103022          MOV    #GETCMD,BPCSR0   ;ISSUE <GET COMMAND> PORT COMMAND
6709 030550' 103022          MOV    #2*SECOND,METER ;SETUP TIMER
6709 030550' 103022          JSR    PC,CHKDNI        ;WAIT FOR 'DNI'
6709 030550' 103022          BCC    30$              ;OK
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 143  
CZUAAB.MAC 07-APR-83 17:03 TEST 12: LOAD AND START FUNCTION TEST

```

6710                                     :ERROR 'DNI' NOT SET!
6711 030552' 012737 001000' 000310'   MOV   #SDNI,BITNAM   :SETUP ERROR MESSAGE
6712 030560' 012737 001277' 000312'   MOV   #SNSET,BITSTA
6713 030566' 012737 001342' 000314'   MOV   #AFTER,PWHEM
6714 030574' 012737 001357' 000316'   MOV   #SGICMD,PCOMND
6715 030602'                                     ERRHRD 044,LASFT,MSG1   :PRINT ERROR MESSAGE
6716 030602' 104456                                     TRAP   CSERHRD
6717 030604' 000054                                     .WORD 44
6718 030606' 002026'                                     .WORD LASFT
6719 030610' 012716'                                     .WORD MSG1
6720 030612'                                     ESCAPE TST   :LEAVE TEST
6721 030612' 104410                                     TRAP   CSESCAPE
6722 030614' 000170                                     .WORD L10126-
6723 030616'   30$:
6724 030616'   ENDSEG
6725 030616'                                     10002$:
6726 030616' 104405                                     TRAP   CSESEG
6727 030620'   BGNSEG
6728 030620' 104404                                     TRAP   CSBSEG
6729                                     :
6730                                     :WRITE ONE TO CLEAR 'DNI'
6731                                     :
6732 030622' 004737 017362'   JSR   PC,CLRDMI   :GO CLEAR 'DNI'
6733 030626' 103006   BCC   408   :OK
6734                                     :ERROR 'DNI' NOT CLEAR
6735 030630'   ERRHRD 045,LASFT,RACMG7   :PRINT ERROR MESSAGE
6736 030630' 104456                                     TRAP   CSERHRD
6737 030632' 000055                                     .WORD 45
6738 030634' 002026'                                     .WORD LASFT
6739 030636' 012670'                                     .WORD RACMG7
6740 030640'                                     ESCAPE TST   :DO NOT CONTINUE TEST
6741 030640' 104410                                     TRAP   CSESCAPE
6742 030642' 000142                                     .WORD L10126-
6743 030644'   40$:
6744 030644'   ENDSEG
6745 030644'                                     10003$:
6746 030644' 104405                                     TRAP   CSESEG
6747                                     :
6748                                     :OK, MICROCODE MODULE 'A' IS LOADED INTO WCS. NOW START IT AND CHECK PCSR1.
6749                                     :
6750                                     :BGNSEG
6751 030646' 104404                                     TRAP   CSBSEG
6752 030650' 012737 000001 000606'   MOV   #LASH,PCBB   :LOAD PCBB WITH 'LOAD AND START' FUNCTION CODE
6753 030656' 012737 010000 000610'   MOV   #WCSADR+<WCSSIZ/2>,PCBB+2 :STARTING MICROADDRESS
6754 030664' 012777 000002 147444   MOV   #GETCMD,BPCSR0 :ISSUE <GET COMMAND> PORT COMMAND
6755 030672' 012737 000176 000332'   MOV   #2*SECOND,METER :SETUP TIMER
6756 030700' 004737 017316'   JSR   PC,CHKDNI   :GO WAIT FOR 'DNI' TO SET
6757 030704' 103020   BCC   50$   :OK
6758                                     :ERROR 'DNI' NOT SET!
6759 030706' 012737 001000' 000310'   MOV   #SDNI,BITNAM   :SETUP ERROR MESSAGE
6760 030714' 012737 001277' 000312'   MOV   #SNSET,BITSTA
6761 030722' 012737 001342' 000314'   MOV   #AFTER,PWHEM
6762 030730' 012737 001357' 000316'   MOV   #SGICMD,PCOMND
6763 030736'                                     ERRHRD 046,LASFT,MSG1   :PRINT ERROR MESSAGE
6764 030736' 104456                                     TRAP   CSERHRD
6765 030740' 000056                                     .WORD 46

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 144  
 CZUAB.MAC 07-APR-83 17:03 TEST 12: LOAD AND START FUNCTION TEST

6766	030742'	002026'					
6767	030744'	012716'				.WORD	LASFT
6768	030746'	017702	147366	508:	MOV	BPCSR1,R2	:GET PCRS1 CONTENTS
6769	030752'	042702	140370		BIC	#*C<SFT!PSTATE>,R2	:CLEAR ALL BUT SELF TEST AND STATE BITS
6770	030756'	012701	012402		MOV	#SF1B0!SFTB2!SFTB4!INTST,R1	:PATTERN THAT SHOULD BE IN PCRS1
6771	030762'	020102			MOV	R1,R2	:IS PCRS1 PATTERN CORRECT?
6772	030764'	001404			BEQ	608	:YES
6773	030766'				ERRHRD	047,LASFT,MSG6	:NO, PRINT ERROR MESSAGE
6774	030766'	104456					TRAP
6775	030770'	000057					.WORD
6776	030772'	002026'					.WORD
6777	030774'	013174'					.WORD
6778	030776'			608:			MSG6
6779	030776'			ENDSEG			
6780	030776'						
6781	030776'	104405					100048:
6782							TRAP
6783							CSESEG
6784							
6785							
6786	031000'	004737	020166'				
6787					JSR	PC,REUNA	:RESET DEUNA TO RESTORE OPERATIONAL
6788	031004'						:MICROCODE
6789	031004'			ENDTST			
6790	031004'	104401					L10126:
							TRAP
							CSETST

:EVERYTHING WORKED JUST FINE, NOW WE HAVE TO GET THE OPERATIONAL MICROCODE  
 :GOING AGAIN BEFORE WE LEAVE OTHERWISE EVERYTHING WILL BE SCREWED UP.

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 145  
CZUAAB.MAC 07-APR-83 17:03 TEST 13: COMPREHENSIVE WCS MEMORY TEST

6791  
6792  
6793  
6794  
6795  
6796  
6797  
6798  
6799  
6800  
6801  
6802  
6803  
6804  
6805  
6806  
6807  
6808  
6809  
6810  
6811  
6812  
6813  
6814  
6815  
6816  
6817  
6818

.SBTTL TEST 13: COMPREHENSIVE WCS MEMORY TEST  
:\*\*\*\*\*8  
:THIS TEST WILL EXHAUSTIVELY TEST THE WCS MEMORY.  
:CUSTOM MICROCODE MODULE B, MICROTEST #1 IS USED TO DO THE ACTUAL TESTING.  
:MICROTEST #1 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE WCS MEMORY CHECKING  
:FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL  
:BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.  
:PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED  
:PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR  
:PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION  
:PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN  
:PCBB+6: CONTAINS THE DATA THAT WAS READ  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'B' INTO THE TOP HALF OF WCS IF NOT ALREADY DONE SO  
: 2-WAIT FOR THE MICROMONITOR TO BECOME ACTIVE  
: 3-CLEAR PCBB LOCATIONS 0-7  
: 4-TELL MICROMONITOR TO EXECUTE MICROTEST #1  
: 5-WAIT FOR 'DNI'  
: 6-IF ERROR PRINT PCBB CONTENTS  
: 7-WRITE ONE TO CLEAR 'DNI'  
: 8-RESTORE OPERATIONAL MICROCODE  
:\*\*\*\*\*

6819 031006'  
6820 031006'  
6821  
6822  
6823  
6824  
6825

BGNTST  
T13::  
:CHECK TO SEE IF MICROCODE MODULE 'B' HAS BEEN LOADED. IF NOT, LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

6826 031006'  
6827 031006' 104404  
6828 031010' 022737 000102 000326'  
6829 031016' 001004  
6830 031020' 122777 000001 147312  
6831 031026' 001435  
6832 031030' 012737 000102 000326' 58:  
6833 031036' 004737 020340'  
6834 031042' 103564  
6835 031044' 012737 000176 000332' 108:  
6836 031052' 004737 017316'  
6837 031056' 103021  
6838 031060' 012737 001000' 000310'  
6839 031066' 012737 001277' 000312'  
6840 031074' 012737 001342' 000314'  
6841 031102' 012737 001357' 000316'  
6842 031110'  
6843 031110' 104456  
6844 031112' 000060  
6845 031114' 002072'  
6846 031116' 012716'

BGNSEG  
TRAP CSBSEG  
:HAS MICROCODE MODULE 'B' BEEN LOADED?  
:NO  
:YES, IS THE MICROMONITOR ACTIVE?  
:YES SKIP LOADING THE MICROMODULE  
:GO LOAD MICROCODE MODULE B  
:ERROR  
:WAIT FOR THE MICROMONITOR  
:OK  
:PRINT ERROR MESSAGE  
TRAP CSERHRD  
.WORD 48  
.WORD WCSMEM  
.WORD MSG1

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 146  
 CZUAAB.MAC 07-APR-83 17:03 TEST 13: COMPREHENSIVE WCS MEMORY TEST

```

6847 031120' 000535
6848 031122' 004737 017362'
6849 031126' 103005
6850 031130'
6851 031130' 104456
6852 031132' 000061
6853 031134' 002072'
6854 031136' 012670'
6855 031140' 000525
6856 031142'
6857 031142'
6858 031142'
6859 031142' 104405
6860
6861
6862
6863
6864
6865
6866 031144'
6867 031144' 104404
6868 031146' 004737 020060'
6869 031152' 103006
6870 031154'
6871 031154' 104456
6872 031156' 000062
6873 031160' 002072'
6874 031162' 016666'
6875 031164'
6876 031164' 104410
6877 031166' 000234
6878 031170' 012777 000606' 147144 308:
6879 031176' 005077 147142
6880 031202' 005037 000606'
6881 031206' 005037 000610'
6882 031212' 005037 000612'
6883 031216' 005037 000614'
6884 031222' 012777 000001 147106
6885
6886 031230' 012737 000770 000332'
6887 031236' 004737 017316'
6888 031242' 103017
6889 031244' 004737 020132'
6890 031250' 103005
6891 031252'
6892 031252' 104456
6893 031254' 000063
6894 031256' 002072'
6895 031260' 016442'
6896 031262' 000454
6897 031264' 012702 000003
6898 031270'
6899 031270' 104456
6900 031272' 000064
6901 031274' 002072'
6902 031276' 013466'

      BR      558
208:     JSR      PC,CLRDN1      ;CLEAR DNI
         BCC      258
         ERRHRD   0'9,WCSMEM,RACMG7 ;DNI DID NOT CLEAR!
                                TRAP   CSERHRD
                                .WORD  49
                                .WORD  WCSMEM
                                .WORD  RACMG7

      BR      558
258:
ENDSEG

                                100008:
                                TRAP   CSESEG

;
;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE THEN START MICROTST #1.
;WAIT FOR 'DNI'. CHECK FOR ILLEGAL INTERRUPTS CHECK THE STATE BITS FOR AN
;ERROR CONDITION. IF ERROR REPORT IT. WRITE '1' TO CLEAR 'DNI' AND RESTORE
;OPERATIONAL MICROCODE.
;
      BGNSEG

                                TRAP   CSBSEG
                                .WORD  50
                                .WORD  WCSMEM
                                .WORD  MSG46
                                TRAP   CSERHRD
                                .WORD  50
                                .WORD  WCSMEM
                                .WORD  MSG46
                                TRAP   CSESCAPE
                                .WORD  L10127-

308:     MOV      #PCBB,@PCSR2      ;TELL MICROCODE TEST WHERE PCBB IS
         CLR      @PCSR3
         CLR      PCBB+0
         CLR      PCBB+2
         CLR      PCBB+4
         CLR      PCBB+6
         MOV      #1,@PCSR0
                                ;TELL MICROMONITOR TO EXECUTE...
                                ;MICROTST #1
                                ;PUT SOME TIME ON THE METER
                                ;WAIT FOR MICROTST TO FINISH
                                ;OK, IT FINISHED
                                ;SEE IF ANY ERROR INTERRUPTS OCCURRED
                                ;NO, OK
                                ;PRINT ERROR MESSAGE
                                TRAP   CSERHRD
                                .WORD  51
                                .WORD  WCSMEM
                                .WORD  MSG44

358:     BR      558
         MOV      #3,R2
         ERRHRD   052,WCSMEM,MSG12
                                ;LEAVE
                                ;MICROTST # THAT IS HUNG
                                ;TELL MICROTST HUNG
                                TRAP   CSERHRD
                                .WORD  52
                                .WORD  WCSMEM
                                .WORD  MSG12

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 148  
CZUAAB.MAC 07-APR-83 17:03 TEST 14: INTERRUPT VECTOR TEST

6938  
6939  
6940  
6941  
6942  
6943  
6944  
6945  
6946  
6947  
6948  
6949  
6950  
6951  
6952  
6953  
6954  
6955  
6956  
6957  
6958  
6959  
6960  
6961  
6962  
6963  
6964  
6965  
6966  
6967  
6968  
6969  
6970  
6971  
6972  
6973  
6974  
6975  
6976  
6977  
6978  
6979  
6980  
6981  
6982  
6983  
6984  
6985  
6986  
6987  
6988  
6989  
6990  
6991  
6992  
6993

031424'  
031424'  
  
  
  
  
  
031424'  
031424' 104404  
031426' 013746 000272'  
031432' 012746 022044'  
031436' 013746 000270'  
031442' 012746 000003  
031446' 104437  
031450' 062706 000010  
031454' 005037 000672'  
031460' 012777 000100 146650  
031466' 012777 000106 146642  
031474' 012737 000176 000332'  
031502' 004737 017316'  
031506' 103020  
  
031510' 012737 001000' 000310'  
031516' 012737 001277' 000312'  
031524' 012737 001342' 000314'  
031532' 012737 001405' 000316'  
031540'  
031540' 104456  
031542' 000067  
031544' 002121'  
031546' 012716'

.SBTTL TEST 14: INTERRUPT VECTOR TEST  
:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE INTERRUPT INTERFACE LOGIC OF THE DEUMA  
:IS CAPABLE OF GENERATING AN INTERRUPT VECTOR AND ARBITRATING FOR CONTROL  
:OF THE UNIBUS.  
:THE DEUMA INTERRUPT ENABLE BIT WILL BE SET AND AN INTERRUPT WILL BE  
:CAUSED BY ISSUING A NOP PORT COMMAND. AN INTERRUPT IS EXPECTED AT THE  
:CORRECT VECTOR AND AT THE CORRECT PRIORITY.  
:TEST SEQUENCE:  
: 1-SETUP INTERRUPT VECTOR  
: 2-CLEAR INTERRUPT FLAG  
: 3-SET INTERRUPT ENABLE IN PCSRO  
: 4-ISSUE NOP PORT COMMAND  
: 5-WAIT FOR DNI SET IN PCSRO  
: 6-VERIFY INTERRUPT FLAG SET  
: 7-VERIFY INTERRUPT AT CORRECT PRIORITY  
: 8-RELEASE INTERRUPT VECTOR  
: 9-CLEAR INTERRUPT ENABLE IN PCSRO  
: 10-WRITE DNI TO CLEAR  
:\*\*\*\*\*

BGNTST  
T14::  
:SETUP INTERRUPT VECTOR, CLEAR INTERRUPT FLAG, SET INTERRUPT ENABLE,  
:ISSUE NOP PORT COMMAND AND WAIT FOR 'DNI'  
:BGNSEG  
SETVEC UNAVEC,#UNASRV,UNAPRI ;SETUP DEUMA INTERRUPT VECTOR TRAP CSBSEG  
MOV UNAPRI,-(SP) TRAP  
MOV #UNASRV,-(SP) TRAP  
MOV UNAVEC,-(SP) TRAP  
MOV #3,-(SP) TRAP  
ADD C\$SVEC TRAP  
ADD #10,SP TRAP  
CLR UNAINTR ;CLEAR UNA INTERRUPTED FLAG  
MOV #IE,@PCSRO ;SET INTERRUPT ENABLE  
MOV #IE!PNOP,@PCSRO ;ISSUE NOP PORT COMMAND  
MOV #2\*SECOND,METER ;SETUP TIMER  
JSR PC,CHKDNI ;GO WAIT FOR DNI  
BCC 10\$ ;OK  
MOV #SDNI,BITNAM ;ERROR DNI NOT SET!  
MOV #SNSSET,BITSTA ;SETUP ERROR MESSAGE  
MOV #SAFTER,PWHEN  
MOV #SNOP,PCOMND  
ERRHRD 055,INTVEC,MSG1 ;PRINT ERROR MESSAGE  
TRAP CSERHRD  
.WORD 55  
.WORD INTVEC  
.WORD MSG1

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 149  
CZUAAB.MAC 07-APR-83 17:03 TEST 14: INTERRUPT VECTOR TEST

```

6994 031550'
6995 031550'
6996 031550'
6997 031550' 104405
6998
6999
7000
7001 031552'
7002 031552' 104404
7003 031554' 012701 000006
7004
7005 031560' 010102
7006 031562' 072227 000005
7007 031566'
7008 031566' 010200
7009 031570' 104441
7010 031572' 000240
7011 031574' 005301
7012 031576' 100370
7013
7014 031600' 005737 000672'
7015 031604' 001006
7016
7017 031606'
7018 031606' 104456
7019 031610' 000070
7020 031612' 002121'
7021 031614' 013262'
7022 031616'
7023 031616' 104410
7024 031620' 000072
7025 031622'
7026 031622' 013701 000272'
7027 031626' 072127 177773
7028 031632' 005301
7029 031634' 072127 000005
7030 031640' 020137 000676'
7031 031644' 001404
7032
7033 031646'
7034 031646' 104456
7035 031650' 000071
7036 031652' 002121'
7037 031654' 013310'
7038 031656'
7039 031656'
7040 031656'
7041 031656' 104405
7042 031660'
7043 031660' 013700 000270'
7044 031664' 104436
7045 031666' 012777 000000 146442
7046 031674' 004737 017362'
7047 031700' 103004
7048 031702'
7049 031702' 104456

```

```

10$:
ENDSEG

10000$:
TRAP C$ESEG

:
:VERIFY THAT INTERRUPT OCCURRED AT CORRECT PRIORITY
:
BGNSEG

TRAP C$BSEG
MOV #6,R1 ;START CPU PRIORITY LOWERING
;FROM PRIORITY 7
20$: MOV R1,R2 ;GET INTEGER PRIORITY
ASH #5,R2 ;PUT PRIORITY IN CORRECT POSITION
SETPRI R2 ;SET NEW PRIORITY

MOV R2,R0
TRAP C$SPRI
NOP ;LET INTERRUPT OCCUR HERE IF PENDING
DEC R1 ;LOWER PRIORITY
BPL 20$ ;IF DONE FROM 6-->0 THEN DONE

TST UNAIINT ;UNA SHOULD HAVE INTERRUPTED BY NOW
BNE 30$ ;OK, GO CHECK THE PRIORITY OF THE INTERUPT
ERRHRD 056,INTVEC,MSG7 ;ERROR! UNA DID NOT INTERRUPT
;PRINT ERROR MESSAGE

TRAP C$ERHRD
.WORD 56
.WORD INTVEC
.WORD MSG7

ESCAPE TST ;LEAVE TEST

TRAP C$ESCAPE
.WORD L10130-.

30$:
MOV UNAPRI,R1 ;GET UNA PRIORITY
ASH #-5,R1 ;MAKE IT AN INTEGER
DEC R1 ;THIS IS THE CPU PRIORITY WHEN THE...
ASH #5,R1 ;UNA SHOULD HAVE INTERRUPTED
CMP R1,CPUPRI ;DID UNA INTERRUPT AT CORRECT PRIORITY?
BEQ 40$ ;YES.
ERRHRD 057,INTVEC,MSG8 ;ERROR! UNA INTERRUPT PRIORITY INCORRECT
;PRINT ERROR MESSAGE

TRAP C$ERHRD
.WORD 57
.WORD INTVEC
.WORD MSG8

40$:
ENDSEG

10001$:
TRAP C$ESEG
CLRVEC UNAVEC ;RELEASE INTERRUPT VECTOR
MOV UNAVEC,R0
TRAP C1,VEC

MOV #0,BPCSRO ;CLEAR INTERRUPT ENABLE
JSR PC,CLRDN1 ;CLEAR DNI
BCC 50$
ERRHRD 058,INTVEC,RACMG7 ;ERROR! DNI DID NOT CLEAR

TRAP C$ERHRD

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 150  
CZUAAB.MAC 07-APR-83 17:03 TEST 14: INTERRUPT VECTOR TEST

7050 031704' 000072  
7051 031706' 002121'  
7052 031710' 012670'  
7053 031712'  
7054 031712'  
7055 031712'  
7056 031712' 104401

S0\$:  
ENDTST

.WORD 58  
.WORD INTVEC  
.WORD RACMG7

L10130:  
TRAP CSETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 151  
CZUAAB.MAC 07-APR-83 17:03 TEST 15: PCSRO INTERRUPT BIT TEST

7057  
7058  
7059  
7060  
7061  
7062  
7063  
7064  
7065  
7066  
7067  
7068  
7069  
7070  
7071  
7072  
7073  
7074  
7075  
7076  
7077  
7078  
7079  
7080  
7081  
7082  
7083  
7084  
7085  
7086  
7087  
7088  
7089  
7090  
7091

.SBTTL TEST 15: PCSRO INTERRUPT BIT TEST  
:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT EACH OF THE INTERRUPT BITS IN REGISTER PCSRO  
:CAN CAUSE AN INTERRUPT.  
:EACH OF THE INTERRUPTS OF REGISTER PCSRO IS SET UNDER THE CONTROL OF THE  
:T11 AND NOT DIRECTLY BY HARDWARE. THE T11 THEREFORE CAN INITIATE UNIBUS  
:INTERRUPTS BY SETTING BITS IN REGISTER PCSRO.  
:THIS TEST USES MICROMODULE C, MICROTEST #1.  
:MICROCODE MODULE C IS LOADED IF NOT ALREADY DONE SO BY A PREVIOUS TEST.  
:THE DEUNA INTERRUPT VECTOR IS SETUP TO STORE THE CONTENTS OF PCSRO WHEN THE  
:INTERRUPT OCCURS. PCBB+0 IS LOADED WITH THE INTERRUPT BIT THAT IS TO BE TESTED  
:THEN PCSRO COMMAND BITS ARE LOADED WITH A 1 TO TELL THE T11 TO EXECUTE  
:MICROTEST #1. WE WAIT FOR THE INTERRUPT TO OCCUR THEN SEE IF THE CONTENTS  
:OF PCSRO AT THE TIME OF THE INTERRUPT CONTAINED THE CORRECT INTERRUPT BIT.  
:THE TEST IS REPEATED FOR ALL THE INTERRUPT BITS.  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE C INTO THE TOP HALF OF WCS IF NOT ALREADY DONE SO  
: 2-SETUP DEUNA INTERRUPT VECTOR  
: 3-WAIT FOR THE MICROMONITOR TO BECOME ACTIVE  
: 4-SET A BIT IN PCBB+0 THAT CORRESPONDS TO THE INTERRUPT BIT TO TEST  
: 5-SET INTERRUPT ENABLE  
: 6-TELL MICROMONITOR TO EXECUTE MICROTEST #1  
: 7-VERIFY INTERRUPT OCCURRED  
: 8-VERIFY CORRECT BIT CAUSED INTERRUPT  
: 9-WRITE ONE TO CLEAR INTERRUPT BIT  
: 10-REPEAT STEPS 3-9 FOR ALL THE INTERRUPT BITS

7092 031714'  
7093 031714'  
7094  
7095  
7096  
7097  
7098

BGNTST  
T15::  
:CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

7099 031714'  
7100 031714' 104404  
7101 031716' 022737 000103 000326'  
7102 031724' 001004  
7103 031726' 122777 000001 146404  
7104 031734' 001440  
7105 031736' 012737 000103 000326' 58:  
7106 031744' 004737 020340'  
7107 031750' 103002  
7108 031752'  
7109 031752' 104410  
7110 031754' 000444  
7111 031756' 012737 000176 000332' 108:  
7112 031764' 004737 017316'

BGNSEG  
TRAP CSBSEG  
:HAS MICROCODE MODULE C BEEN LOADED?  
:NO  
:YES, IS THE MICROMONITOR ACTIVE?  
:YES SKIP LOADING THE MICROMODULE  
:SETUP TO LOAD MODULE C  
:GO LOAD MICROMODULE C  
:SUCCESS  
:ERROR OCCURRED LEAVE  
TRAP CSESCAPE  
:WORD L10131-  
:PUT SOME TIME ON THE METER  
:GO WAIT FOR DNI TO SET AFTER THE

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 152  
 'ZUAAB.MAC 07-APR-83 17:03 TEST 15: PCSRO INTERRUPT BIT TEST

```

7113                                     ;LOAD AND START FUNCTION
7114 031770' 103022                     BCC      20$
7115 031772' 012737 001000' 000310'    MOV      #S0NI,BITNAM
7116 032000' 012737 001277' 000312'    MOV      #S1SET,BITSTA
7117 032006' 012737 001342' 000314'    MOV      #S1TER,PWHEN
7118 032014' 012737 001357' 000316'    MOV      #SGTCMD,PCOMND
7119 032022'                                     ERRHRD  059,INTBIT,MSG1
7120 032022' 104456                                     ;PRINT ERROR MESSAGE
7121 032024' 000073                                     TRAP   C$ERHRD
7122 032026' 002156'                                     .WORD  59
7123 032030' 012716'                                     .WORD  INTBIT
7124 032032'                                     .WORD  MSG1
7125 032032' 104410                                     ESCAPE TST
7126 032034' 000364                                     TRAP   C$ESCAPE
7127 032036' 004737 017362' 20$:      JSR      PC,CLRDMI
7128 032042' 103006                                     .WORD  L10131-.
7129 032044'                                     BCC      25$
7130 032044' 104456                                     ERRHRD  060,INTBIT,RACMG7
7131 032046' 000074                                     ;CLEAR DMI
7132 032050' 002156'                                     ;ERROR! DMI DID NOT CLEAR
7133 032052' 012670'                                     TRAP   C$ERHRD
7134 032054'                                     .WORD  60
7135 032054' 104410                                     .WORD  INTBIT
7136 032056' 000342                                     .WORD  RACMG7
7137 032060' 25$:      SETVEC UNAVEC,#UNASRV,UNAPRI
7138 032060' 013746 000272'
7139 032064' 012746 022044'
7140 032070' 013746 000270'
7141 032074' 012746 000003
7142 032100' 104437
7143 032102' 062706 000010
7144 032106'                                     ESCAPE TST
7145 032106'                                     TRAP   C$ESCAPE
7146 032106' 104405                                     .WORD  L10131-.
7147                                     MOV      UNAPRI,-(SP)
7148                                     MOV      #UNASRV,-(SP)
7149                                     MOV      UNAVEC,-(SP)
7150                                     MOV      #3,-(SP)
7151                                     TRAP   C$SVEC
7152                                     ADD      #10,SP
7153                                     ;SETUP DEUNA INTERRUPT VECTOR
7154                                     10000$: TRAP   C$ESEG
7155                                     ;
7156                                     ;THE FOLLOWING LOOP WILL BE EXECUTED 6 TIMES- ONCE FOR EACH BIT 10 THRU 15
7157                                     ;OF PCSRO. IT WAITS FOR THE MICROMONITOR TO BECOME ACTIVE THEN CALLS
7158                                     ;MICROTEST #1 TO SET A BIT IN PCSRO AS DEFINED BY THE PARAMETER IN PCBB+0.
7159                                     ;THIS OPERATION SHOULD CAUSE AN INTERRUPT WHICH WILL BE REFLECTED BY THE
7160                                     ;VARIABLE 'UNAINI' WHICH IS SET BY THE UNA INTERRUPT SERVICE ROUTINE. UNAINI
7161                                     ;IS LOADED WITH THE VALUE OF PCSRO AT THE TIME OF THE INTERRUPT. THIS WAY
7162                                     ;THE BIT THAT CAUSED THE INTERRUPT CAN BE CHECKED.
7163                                     ;
7164                                     ;
7165                                     ;
7166                                     ;
7167                                     ;
7168 032110' 005037 000304'
7169 032114' 012737 001277' 000312'
7170 032122' 012701 002000
7171 032126' 012737 000012 000306'
7172 032134'
7173 032134'
7174 032134' 104404
7175 032136' 004737 020060'
7176 032142' 103006
7177 032144'
7178 032144' 104456
7179 032146' 000075
7180 032150' 002156'
7181                                     CLR      CSRMUR
7182                                     MOV      #S1SET,BITSTA
7183                                     MOV      #BIT10,R1
7184                                     MOV      #10.,BITMUR
7185                                     ;CHECKING PCSRO
7186                                     ;CHECKING FOR SET BITS
7187                                     ;START WITH BIT 10
7188 26$:      BGNSEG
7189                                     JSR      PC,CHKMON
7190                                     BCC      30$
7191                                     ERRHRD  061,INTBIT,MSG46
7192                                     ;WAIT FOR MICROMONITOR
7193                                     ;OK
7194                                     ;PRINT ERROR
7195                                     TRAP   C$ERHRD
7196                                     .WORD  61
7197                                     .WORD  INTBIT

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 153  
 CZUAB.MAC 07-APR-83 17:03 TEST 15: PCSRO INTERRUPT BIT TEST

```

7169 032152' 016666'
7170 032154'          ESCAPE TST          ;LEAVE TEST          .WORD  MSG46
7171 032154' 104410
7172 032156' 000242          TRAP      C8ESCAPE
7173 032160' 005037 000672'          .WORD  L10131-.
308: CLR  UNAINI          ;CLEAR UNA INTERRUPTED FLAG
      MOV  R1,PCBB+0      ;TELL MICROMONITOR WHICH BIT TO SET
7174 032164' 010137 000606'          ;PUT SOME TIME ON THE METER
7175 032170' 012737 000077 000332'        ;SET INTERRUPT ENABLE AND...
7176 032176' 012777 000100 146132        ;CLEAR SELF TEST BITS IN PCSR1
7177
7178 032204' 012777 000101 146124        ;INVOKC INTERRUPT BIT MICROTEST
7179 032212' 004737 017264'          ;TURN ON THE TIMER
7180 032216'
7181 032216' 012700 000200          ;LOWER PRIORITY TO ALLOW UNA INTERRUPT
7182 032222' 104441          MOV      #PRI04,RO
7183 032224' 005737 000672'          TRAP    C8SPRI
358: TST  UNAINI          ;HAS INTERRUPT OCCURRED YET?
      BNE  408            ;YES
7184 032230' 001022          ;NO, HAS METER EXPIRED?
7185 032232' 005737 000332'        ;NOT YET
7186 032236' 001372          ;TIMER HAS EXPIRED, SHUT IT OFF
7187 032240' 004737 017302'        ;ERROR, NO UNA INTERRUPT!
7188
7189 032244' 013703 000306'        ;GET WHICH BIT
7190 032250' 006303          ;MAKE IT A BYTE OFFSET
7191 032252' 012737 000360' 000310'      ;POINT TO TABLE OF BIT PNEMONICS
7192 032260' 060337 000310'        ;INDEX INTO TABLE OF BIT PNEMONICS
7193 032264'          ;PRINT ERROR MESSAGE
7194 032264' 104456          TRAP    C8ERHRD
7195 032266' 000076          .WORD  62
7196 032270' 002156'        .WORD  INTBIT
7197 032272' 013332'        .WORD  MSG9
7198 032274' 000402
7199 032276' 004737 017302'        408: BR      458
7200 032302' 013703 000672'        458: JSR    PC,TIMOFF ;INTERRUPT OCCURRED, SHUT OFF THE TIMER
7201 032306' 042703 000377        MOV    UNAINI,R3 ;GET SAVED PCSRO CONTENTS
7202 032312' 020103          BIC    #377,R3 ;CLEAR UNWANTED BITS
7203 032314' 001404          CMP    R1,R3 ;DID CORRECT BIT CAUSE INTERRUPT?
7204
7205          BEQ    508 ;YES, OK
7206 032316'          ERRHRD 063,INTBIT,RACMG2 ;ERROR, INCORRECT BIT CAUSED INTERRUPT!
7207 032320' 000077          ;PRINT ERROR MESSAGE
7208 032322' 002156'        TRAP    C8ERHRD
7209 032324' 012526'        .WORD  63
7210 032326'          .WORD  INTBIT
7211 032326'          .WORD  RACMG2
308: ENDSEG
7212 032326'
7213 032326' 104405          100018: TRAP  C8SEEG
7214
7215          ;
7216          ;NOW WRITE '1' TO CLEAR THE BIT THAT CAUSED THE INTERRUPT
7217          ;
7217 032330'          ;
7218 032330' 104404          ;
7219 032332' 010377 146000        ;WRITE '1' TO CLEAR INTERRUPT BIT
7220 032336' 017704 145774        MOV    R3,@PCSRO ;READ IT BACK
7221 032342' 030403          MOV    @PCSRO,R4
7222 032344' 001420          BIT    R4,R3 ;IS BIT CLEARED?
7223
7224 032346' 012705 000017        BEQ    558 ;YES
7225
7226          ;ERROR, BIT DID NOT CLEAR!
7227
7228          ;
7229          ;
7230          ;
7231          ;
7232          ;
7233          ;
7234          ;
7235          ;
7236          ;
7237          ;
7238          ;
7239          ;
7240          ;
7241          ;
7242          ;
7243          ;
7244          ;
7245          ;
7246          ;
7247          ;
7248          ;
7249          ;
7250          ;
7251          ;
7252          ;
7253          ;
7254          ;
7255          ;
7256          ;
7257          ;
7258          ;
7259          ;
7260          ;
7261          ;
7262          ;
7263          ;
7264          ;
7265          ;
7266          ;
7267          ;
7268          ;
7269          ;
7270          ;
7271          ;
7272          ;
7273          ;
7274          ;
7275          ;
7276          ;
7277          ;
7278          ;
7279          ;
7280          ;
7281          ;
7282          ;
7283          ;
7284          ;
7285          ;
7286          ;
7287          ;
7288          ;
7289          ;
7290          ;
7291          ;
7292          ;
7293          ;
7294          ;
7295          ;
7296          ;
7297          ;
7298          ;
7299          ;
7300          ;
7301          ;
7302          ;
7303          ;
7304          ;
7305          ;
7306          ;
7307          ;
7308          ;
7309          ;
7310          ;
7311          ;
7312          ;
7313          ;
7314          ;
7315          ;
7316          ;
7317          ;
7318          ;
7319          ;
7320          ;
7321          ;
7322          ;
7323          ;
7324          ;
7325          ;
7326          ;
7327          ;
7328          ;
7329          ;
7330          ;
7331          ;
7332          ;
7333          ;
7334          ;
7335          ;
7336          ;
7337          ;
7338          ;
7339          ;
7340          ;
7341          ;
7342          ;
7343          ;
7344          ;
7345          ;
7346          ;
7347          ;
7348          ;
7349          ;
7350          ;
7351          ;
7352          ;
7353          ;
7354          ;
7355          ;
7356          ;
7357          ;
7358          ;
7359          ;
7360          ;
7361          ;
7362          ;
7363          ;
7364          ;
7365          ;
7366          ;
7367          ;
7368          ;
7369          ;
7370          ;
7371          ;
7372          ;
7373          ;
7374          ;
7375          ;
7376          ;
7377          ;
7378          ;
7379          ;
7380          ;
7381          ;
7382          ;
7383          ;
7384          ;
7385          ;
7386          ;
7387          ;
7388          ;
7389          ;
7390          ;
7391          ;
7392          ;
7393          ;
7394          ;
7395          ;
7396          ;
7397          ;
7398          ;
7399          ;
7400          ;
7401          ;
7402          ;
7403          ;
7404          ;
7405          ;
7406          ;
7407          ;
7408          ;
7409          ;
7410          ;
7411          ;
7412          ;
7413          ;
7414          ;
7415          ;
7416          ;
7417          ;
7418          ;
7419          ;
7420          ;
7421          ;
7422          ;
7423          ;
7424          ;
7425          ;
7426          ;
7427          ;
7428          ;
7429          ;
7430          ;
7431          ;
7432          ;
7433          ;
7434          ;
7435          ;
7436          ;
7437          ;
7438          ;
7439          ;
7440          ;
7441          ;
7442          ;
7443          ;
7444          ;
7445          ;
7446          ;
7447          ;
7448          ;
7449          ;
7450          ;
7451          ;
7452          ;
7453          ;
7454          ;
7455          ;
7456          ;
7457          ;
7458          ;
7459          ;
7460          ;
7461          ;
7462          ;
7463          ;
7464          ;
7465          ;
7466          ;
7467          ;
7468          ;
7469          ;
7470          ;
7471          ;
7472          ;
7473          ;
7474          ;
7475          ;
7476          ;
7477          ;
7478          ;
7479          ;
7480          ;
7481          ;
7482          ;
7483          ;
7484          ;
7485          ;
7486          ;
7487          ;
7488          ;
7489          ;
7490          ;
7491          ;
7492          ;
7493          ;
7494          ;
7495          ;
7496          ;
7497          ;
7498          ;
7499          ;
7500          ;
7501          ;
7502          ;
7503          ;
7504          ;
7505          ;
7506          ;
7507          ;
7508          ;
7509          ;
7510          ;
7511          ;
7512          ;
7513          ;
7514          ;
7515          ;
7516          ;
7517          ;
7518          ;
7519          ;
7520          ;
7521          ;
7522          ;
7523          ;
7524          ;
7525          ;
7526          ;
7527          ;
7528          ;
7529          ;
7530          ;
7531          ;
7532          ;
7533          ;
7534          ;
7535          ;
7536          ;
7537          ;
7538          ;
7539          ;
7540          ;
7541          ;
7542          ;
7543          ;
7544          ;
7545          ;
7546          ;
7547          ;
7548          ;
7549          ;
7550          ;
7551          ;
7552          ;
7553          ;
7554          ;
7555          ;
7556          ;
7557          ;
7558          ;
7559          ;
7560          ;
7561          ;
7562          ;
7563          ;
7564          ;
7565          ;
7566          ;
7567          ;
7568          ;
7569          ;
7570          ;
7571          ;
7572          ;
7573          ;
7574          ;
7575          ;
7576          ;
7577          ;
7578          ;
7579          ;
7580          ;
7581          ;
7582          ;
7583          ;
7584          ;
7585          ;
7586          ;
7587          ;
7588          ;
7589          ;
7590          ;
7591          ;
7592          ;
7593          ;
7594          ;
7595          ;
7596          ;
7597          ;
7598          ;
7599          ;
7600          ;
7601          ;
7602          ;
7603          ;
7604          ;
7605          ;
7606          ;
7607          ;
7608          ;
7609          ;
7610          ;
7611          ;
7612          ;
7613          ;
7614          ;
7615          ;
7616          ;
7617          ;
7618          ;
7619          ;
7620          ;
7621          ;
7622          ;
7623          ;
7624          ;
7625          ;
7626          ;
7627          ;
7628          ;
7629          ;
7630          ;
7631          ;
7632          ;
7633          ;
7634          ;
7635          ;
7636          ;
7637          ;
7638          ;
7639          ;
7640          ;
7641          ;
7642          ;
7643          ;
7644          ;
7645          ;
7646          ;
7647          ;
7648          ;
7649          ;
7650          ;
7651          ;
7652          ;
7653          ;
7654          ;
7655          ;
7656          ;
7657          ;
7658          ;
7659          ;
7660          ;
7661          ;
7662          ;
7663          ;
7664          ;
7665          ;
7666          ;
7667          ;
7668          ;
7669          ;
7670          ;
7671          ;
7672          ;
7673          ;
7674          ;
7675          ;
7676          ;
7677          ;
7678          ;
7679          ;
7680          ;
7681          ;
7682          ;
7683          ;
7684          ;
7685          ;
7686          ;
7687          ;
7688          ;
7689          ;
7690          ;
7691          ;
7692          ;
7693          ;
7694          ;
7695          ;
7696          ;
7697          ;
7698          ;
7699          ;
7700          ;
7701          ;
7702          ;
7703          ;
7704          ;
7705          ;
7706          ;
7707          ;
7708          ;
7709          ;
7710          ;
7711          ;
7712          ;
7713          ;
7714          ;
7715          ;
7716          ;
7717          ;
7718          ;
7719          ;
7720          ;
7721          ;
7722          ;
7723          ;
7724          ;
7725          ;
7726          ;
7727          ;
7728          ;
7729          ;
7730          ;
7731          ;
7732          ;
7733          ;
7734          ;
7735          ;
7736          ;
7737          ;
7738          ;
7739          ;
7740          ;
7741          ;
7742          ;
7743          ;
7744          ;
7745          ;
7746          ;
7747          ;
7748          ;
7749          ;
7750          ;
7751          ;
7752          ;
7753          ;
7754          ;
7755          ;
7756          ;
7757          ;
7758          ;
7759          ;
7760          ;
7761          ;
7762          ;
7763          ;
7764          ;
7765          ;
7766          ;
7767          ;
7768          ;
7769          ;
7770          ;
7771          ;
7772          ;
7773          ;
7774          ;
7775          ;
7776          ;
7777          ;
7778          ;
7779          ;
7780          ;
7781          ;
7782          ;
7783          ;
7784          ;
7785          ;
7786          ;
7787          ;
7788          ;
7789          ;
7790          ;
7791          ;
7792          ;
7793          ;
7794          ;
7795          ;
7796          ;
7797          ;
7798          ;
7799          ;
7800          ;
7801          ;
7802          ;
7803          ;
7804          ;
7805          ;
7806          ;
7807          ;
7808          ;
7809          ;
7810          ;
7811          ;
7812          ;
7813          ;
7814          ;
7815          ;
7816          ;
7817          ;
7818          ;
7819          ;
7820          ;
7821          ;
7822          ;
7823          ;
7824          ;
7825          ;
7826          ;
7827          ;
7828          ;
7829          ;
7830          ;
7831          ;
7832          ;
7833          ;
7834          ;
7835          ;
7836          ;
7837          ;
7838          ;
7839          ;
7840          ;
7841          ;
7842          ;
7843          ;
7844          ;
7845          ;
7846          ;
7847          ;
7848          ;
7849          ;
7850          ;
7851          ;
7852          ;
7853          ;
7854          ;
7855          ;
7856          ;
7857          ;
7858          ;
7859          ;
7860          ;
7861          ;
7862          ;
7863          ;
7864          ;
7865          ;
7866          ;
7867          ;
7868          ;
7869          ;
7870          ;
7871          ;
7872          ;
7873          ;
7874          ;
7875          ;
7876          ;
7877          ;
7878          ;
7879          ;
7880          ;
7881          ;
7882          ;
7883          ;
7884          ;
7885          ;
7886          ;
7887          ;
7888          ;
7889          ;
7890          ;
7891          ;
7892          ;
7893          ;
7894          ;
7895          ;
7896          ;
7897          ;
7898          ;
7899          ;
7900          ;
7901          ;
7902          ;
7903          ;
7904          ;
7905          ;
7906          ;
7907          ;
7908          ;
7909          ;
7910          ;
7911          ;
7912          ;
7913          ;
7914          ;
7915          ;
7916          ;
7917          ;
7918          ;
7919          ;
7920          ;
7921          ;
7922          ;
7923          ;
7924          ;
7925          ;
7926          ;
7927          ;
7928          ;
7929          ;
7930          ;
7931          ;
7932          ;
7933          ;
7934          ;
7935          ;
7936          ;
7937          ;
7938          ;
7939          ;
7940          ;
7941          ;
7942          ;
7943          ;
7944          ;
7945          ;
7946          ;
7947          ;
7948          ;
7949          ;
7950          ;
7951          ;
7952          ;
7953          ;
7954          ;
7955          ;
7956          ;
7957          ;
7958          ;
7959          ;
7960          ;
7961          ;
7962          ;
7963          ;
7964          ;
7965          ;
7966          ;
7967          ;
7968          ;
7969          ;
7970          ;
7971          ;
7972          ;
7973          ;
7974          ;
7975          ;
7976          ;
7977          ;
7978          ;
7979          ;
7980          ;
7981          ;
7982          ;
7983          ;
7984          ;
7985          ;
7986          ;
7987          ;
7988          ;
7989          ;
7990          ;
7991          ;
7992          ;
7993          ;
7994          ;
7995          ;
7996          ;
7997          ;
7998          ;
7999          ;
8000          ;

```

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 154  
 CZUAB.MAC 07-APR-83 17:03 TEST 15: PCSRD INTERRUPT BIT TEST

```

7225 032352' 006303          528:  ASL      R3          ;IS THIS THE BIT WE SET?
7226 032354' 103402          BCS      538          ;YES
7227 032356' 005305          DEC      R5          ;NO
7228 032360' 000774          BR       528
7229 032362' 006305          538:  ASL      R5          ;MAKE IT A BYTE OFFSET
7230 032364' 012737 000360' 000310'  MOV     #BNAMTO,BITNAM ;GET POINTER TO BIT NAME MNEMONICS
7231 032372' 060537 000310'  ADD     R5,BITNAM     ;INDEX INTO BIT NAME TABLE
7232 032376'          ERRHRD  064,INTBIT,MSG33 ;PRINT ERROR MESSAGE
7233 032376' 104456          TRAP   CSEHRD
7234 032400' 000100          .WORD  64
7235 032402' 002156'          .WORD  INTBIT
7236 032404' 015574'          .WORD  MSG33
7237 032406'          558:
7238 032406'          ENDSEG
7239 032406'          100028:
7240 032406' 104405          TRAP   CSESEG
7241 032410' 005237 000306'  INC     BITNUM
7242 032414' 006301          ASL     R1
7243 032416' 103246          BCC    268
7244          ;SETUP FOR NEXT BIT
7245 032420'          ;POINT TO NEXT HIGHER BIT
7246 032420'          ;IF BIT 15 NOT DONE YET GO ON
7247 032420' 104401          ;ELSE ALL DONE
          L10131:
          TRAP   CSETST
  
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 155  
CZUAAB.MAC 07-APR-83 17:03 TEST 16: TIMER TEST

7248  
7249  
7250  
7251  
7252  
7253  
7254  
7255  
7256  
7257  
7258  
7259  
7260  
7261  
7262  
7263  
7264  
7265  
7266  
7267  
7268  
7269  
7270  
7271  
7272  
7273  
7274  
7275  
7276  
7277  
7278  
7279  
7280  
7281  
7282  
7283  
7284  
7285  
7286  
7287  
7288  
7289  
7290  
7291  
7292  
7293  
7294  
7295  
7296  
7297  
7298  
7299  
7300  
7301  
7302  
7303

032422'  
032422'  
  
032422'  
032422' 104404  
032424' 022737 000103 000326'  
032432' 001004  
032434' 122777 000001 145676  
032442' 001440  
032444' 012737 000103 000326' 58:  
032452' 004737 020340'  
032456' 103002  
032460'  
032460' 104410  
032462' 000254  
032464' 012737 000176 000332' 108:  
032472' 004737 017316'  
032476' 103022  
032500' 012737 001000' 000310'  
032506' 012737 001277' 000312'  
032514' 012737 001342' 000314'  
032522' 012737 001357' 000316'  
032530'  
032530' 104456  
032532' 000101  
032534' 002216'

.SBTTL TEST 16: TIMER TEST  
:\*\*\*\*\*  
: THIS TEST WILL USE THE CUSTOM MICROCODE MODULE 'C' TO CHECK THE OPERATION  
: OF THE TIMER.  
: THE TIMER IS ACCESSIBLE ONLY TO THE T11 PROCESSOR. THE HOST PROCESSOR  
: CAN START THE TIMER ONLY WITH THE ASSISTANCE OF THE T11 PROCESSOR.  
: FOR THIS TEST THE MICROCODE WILL BE LOADED ONLY IF IT HAS NOT ALREADY  
: BEEN DONE BY A PREVIOUS TEST.  
: WHEN THE MICROCODE IS STARTED THE T11 WILL START THE TIMER AND WILL  
: SET 'DNI' WHEN THE TIMING INTERVAL HAS EXPIRED. THE INTERVAL IS 10 SECONDS.  
: ANY TIME FROM 8 TO 12 SECONDS IS AN ACCEPTABLE RANGE.  
: TEST SEQUENCE:  
: 1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO  
: 2-START MICROCODE  
: 3-START TIMER  
: 4-WAIT FOR 'DNI'  
: 5-CHECK TIME INTERVAL  
:\*\*\*\*\*

BGNTST

T16::

: CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO  
: THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
: AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG

TRAP CSBSEG  
:HAS MICROCODE MODULE C BEEN LOADED?  
:NO  
:YES, IS THE MICROMONITOR ACTIVE?  
:YES SKIP LOADING THE MICROMODULE  
:NO  
:GO LOAD MICROCODE MODULE C  
:OK  
:ERROR OCCURRED LOADING MICROCODE  
TRAP C\$ESCAPE  
.WORD L10132-  
:WAIT FOR MICROMONITOR TO TAKE OVER  
MOV #2\*SECOND,METER  
JSR PC,CHKDNI  
BCC 208  
MOV #SDNI,BITMAN  
MOV #SNSET,BITSTA  
MOV #SAFTER,PWEN  
MOV #SGTCMD,PCORND  
ERRMRD 065,YINTST,MS61  
TRAP CSERRRD  
.WORD 65  
.WORD TINTST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 156  
 CZUAAB.MAC 07-APR-83 17:03 TEST 16: TIMER TEST

```

7304 032536' 012716'
7305 032540'
7306 032540' 104410
7307 032542' 000174
7308 032544' 004737 017362'
7309 032550' 103006
7310 032552'
7311 032552' 104456
7312 032554' 000102
7313 032556' 002216'
7314 032560' 012670'
7315 032562'
7316 032562' 104410
7317 032564' 000152
7318 032566'
7319 032566'
7320 032566'
7321 032566' 104405
7322
7323
7324
7325
7326
7327
7328 032570'
7329 032570' 104404
7330 032572' 004737 020060'
7331 032576' 103006
7332 032600'
7333 032600' 104456
7334 032602' 000104
7335 032604' 002216'
7336 032606' 016666'
7337 032610'
7338 032610' 104410
7339 032612' 000124
7340 032614' 012737 001356 000332' 308:
7341 032622' 012777 000002 145506
7342 032630' 004737 017316'
7343 032634' 103006
7344 032636'
7345 032636' 104456
7346 032640' 000105
7347 032642' 002216'
7348 032644' 013360'
7349 032646'
7350 032646' 104410
7351 032650' 000066
7352
7353
7354
7355
7356 032652' 023727 000332' 000764 408:
7357 032660' 003416
7358 032662' 012700 001161
7359 032666' 163700 000332'

```

ESCAPE TST

208: JSR PC,LLRDNI ;CLEAR DNI  
 BCC 258  
 ERRHRD 066,TIMTST,RACMG7 ;ERROR DNI DID NOT CLEAR!

ESCAPE TST

258: ENDSEG

100008: TRAP C8ESEG

:WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR, SETUP OUR TIMEOUT TO BE  
 :12 SECONDS (THIS GIVES A BETTER RESOLUTION THAN 1 SECOND), START THE MICROTEST  
 :BY LOADING THE COMMAND FIELD OF PCSRO WITH THE MICROTEST # TO EXECUTE.  
 :CHECK FOR 'DNI' TO BE SET IN LESS THAN 12 SECONDS.

BGNSEG

TRAP C8BSEG

308: JSR PC,CHKMON ;WAIT FOR MICROMONITOR  
 BCC 308 ;OK  
 ERRHRD 068,TIMTST,MSG46 ;PRINT ERROR

TRAP C8ERHRD  
 .WORD 68  
 .WORD TIMTST  
 .WORD MSG46

ESCAPE TST ;LEAVE TEST

TRAP C8ESCAPE  
 .WORD L10132-

308: MOV #750.,METER ;TIMEOUT = 12 SECONDS  
 MOV #2,PCSRO ;START MICROTEST #2  
 JSR PC,CHKDNI ;WAIT FOR DNI  
 BCC 408 ;OK IT FINISHED IN TIME  
 ERRHRD 069,TIMTST,MSG10 ;NO TIMER INTERRUPT

TRAP C8ERHRD  
 .WORD 69  
 .WORD TIMTST  
 .WORD MSG10

ESCAPE TST

TRAP C8ESCAPE  
 .WORD L10132-

:OK THE TIMER INTERRUPTED IN LESS THAN 12 SECONDS, SO NOW CHECK TO SEE IF IT  
 :HAPPENED IN LESS THAN 8.

408: CMP METER,#500. ;DID INTERRUPT OCCUR BEFORE 8 SECS.?  
 BLE 608 ;NO, OK  
 MOV #625.,RO  
 SUB METER,RO ;CALC HOW MUCH TIME





65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 158  
CZUAAB.MAC 07-APR-83 17:03 TEST 17: LINK MEMORY TEST

7384  
7385  
7386  
7387  
7388  
7389  
7390  
7391  
7392  
7393  
7394  
7395  
7396  
7397  
7398  
7399  
7400  
7401  
7402  
7403  
7404  
7405  
7406  
7407  
7408  
7409  
7410  
7411  
7412  
7413  
7414  
7415  
7416  
7417  
7418  
7419  
7420  
7421  
7422  
7423  
7424  
7425  
7426  
7427  
7428  
7429  
7430  
7431  
7432  
7433  
7434  
7435  
7436  
7437  
7438  
7439

.SBTTL TEST 17: LINK MEMORY TEST

\*\*\*\*\*  
:THIS TEST WILL EXHAUSTIVELY TEST THE LINK MEMORY.  
:THE LINK MEMORY OCCUPIES THE 16-32K ADDRESS SPACE OF THE T-11.  
:CUSTOM MICROCODE MODULE C MICROTEST #3 IS USED TO DO THE ACTUAL TESTING.  
:MICROTEST #3 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE LINK MEMORY CHECKING  
:FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL  
:BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.  
:PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED  
:PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR,  
:PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION  
:PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN  
:PCBB+6: CONTAINS THE DATA THAT WAS READ  
:MICROSUBTEST # DESCRIPTION  
: 1 ACCESS TEST  
: 2 ADDRESS SHIFT TEST  
: 3 DATA LATCH TEST  
: 4 ADDRESS BIT SHIFT #1  
: 5 ADDRESS BIT SHIFT #2  
: 6 MARCH TEST  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'C' INTO THE TOP HALF OF WCS IF NOT ALREADY DONE SO  
: 2-WAIT FOR THE MICROMONITOR TO BECOME ACTIVE  
: 3-TELL MICROMONITOR TO EXECUTE MICROTEST #3  
: 4-VERIFY DNI SET INDICATING TEST COMPLETE  
: 5-CHECK STATE FIELD OF PCSR1 FOR ERROR CONDITION  
\*\*\*\*\*

BGNTST

T17::

:CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

BGNSEG

TRAP CSBSEG  
;HAS MICROCODE MODULE 'C' BEEN LOADED?  
;NO  
;YES. IS THE MICROMONITOR ACTIVE?  
;YES SKIP LOADING THE MICROMODULE  
;GO LOAD MICROCODE MODULE C  
;OK  
TRAP CSESCAPE  
.WORD L10133-  
;WAIT FOR THE MICROMONITOR

032740'  
032740'  
032740' 104404  
032742' 022737 000103 000326'  
032750' 001004  
032752' 122777 000001 145360  
032760' 001440  
032762' 012737 000103 000326' 58:  
032770' 004737 020340'  
032774' 103002  
032776'  
032776' 104410  
033000' 000404  
033002' 012737 000176 000332' 108:

CMF #'C,MICRO  
BNE 58  
CMPB #INMON,PCSR1  
BEQ 208  
MOV #'C,MICRO  
JSR PC,LODMIC  
BCC 108  
ESCAPE TST  
MOV #2\*SECOND,METER

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 159  
 CZUAAB.MAC 07-APR-83 17:03 TEST 17: LINK MEMORY TEST

```

7440 033010' 004737 017316'      JSR    PC,CHKDNI
7441 033014' 103022      BCC    20$                :OK
7442 033016' 012737 001000' 000310'  MOV    #SDNI,BITNAM
7443 033024' 012737 001277' 000312'  MOV    #CNSET,BITSTA
7444 033032' 012737 001342' 000314'  MOV    #SA+TER,PWHEN
7445 033040' 012737 001357' 000316'  MOV    #SGTCMD,PCOMND
7446 033046'      ERRHRD 072,LNKMEM,MSG1 ;PRINT ERROR MESSAGE
7447 033046' 104456      TRAP  C$ERHRD
7448 033050' 000110      .WORD 72
7449 033052' 002240'      .WORD LNKMEM
7450 033054' 012716'      .WORD MSG1
7451 033056'      ESCAPE TST
7452 033056' 104410      TRAP  C$ESCAPE
7453 033060' 000324      .WORD L10133-.
7454 033062' 004737 017362' 20$:  JSR    PC,CLRDNI        :CLEAR DNI
7455 033066' 103006      BCC    25$
7456 033070'      ERRHRD 073,LNKMEM,RACMG7
7457 033070' 104456      TRAP  C$ERHRD
7458 033072' 000111      .WORD 73
7459 033074' 002240'      .WORD LNKMEM
7460 033076' 012670'      .WORD RACMG7
7461 033100'      ESCAPE TST
7462 033100' 104410      TRAP  C$ESCAPE
7463 033102' 000302      .WORD L10133-.
7464 033104'      25$:
7465 033104'      ENDSEG
7466 033104'      10000$:
7467 033104' 104405      TRAP  C$ESEG
7468
7469      ;
7470      ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE, CLEAR THE
7471      ;LOCATIONS FOR ERROR INFORMATION, LOAD COMMAND FIELD OF PCSRO WITH A 3
7472      ;CAUSING THE MICROMONITOR TO EXECUTE MICROTEST #3, THIS WILL START THE
7473      ;EXECUTION OF THE MICROSUBTEST SEQUENCE OF MEMORY TESTS. DNI WILL SET
7474      ;WHEN THE TEST IS COMPLETE
7475      ;
7476      ;
7477      ;
7478      ;
7479      ;
7480      ;
7481      ;
7482      ;
7483      ;
7484      ;
7485      ;
7486      ;
7487      ;
7488      ;
7489      ;
7490      ;
7491      ;
7492      ;
7493      ;
7494      ;
7495      ;
7475 033106'      BGNSEG
7476 033106' 104404      TRAP  C$BSEG
7477 033110' 004737 020060'      JSR    PC,CHKMON        ;WAIT FOR MICROMONITOR
7478 033114' 103006      BCC    30$                :OK
7479 033116'      ERRHRD 074,LNKMEM,MSG46      ;PRINT ERROR
7480 033116' 104456      TRAP  C$ERHRD
7481 033120' 000112      .WORD 74
7482 033122' 002240'      .WORD LNKMEM
7483 033124' 016666'      .WORD MSG46
7484 033126'      ESCAPE TST                :LEAVE TEST
7485 033126' 104410      TRAP  C$ESCAPE
7486 033130' 000254      .WORD L10133-.
7487 033132' 012777 000606' 145202 30$:  MOV    #PCBB,@PCSR2        ;TELL MICROCODE TEST WHERE PCBB IS
7488 033140' 005077 145200      CLR    @PCSR3
7489 033144' 005037 000606'      CLR    PCBB+0            ;CLEAR OUT THE PCBB
7490 033150' 005037 000610'      CLR    PCBB+2
7491 033154' 005037 000612'      CLR    PCBB+4
7492 033160' 005037 000614'      CLR    PCBB+6
7493 033164' 012777 000003 145144  MOV    #3,@PCSR0        ;TELL MICROMONITOR TO EXECUTE...
7494      ;MICROTEST #3
7495 033172' 012737 000770 000332'  MOV    #10*SECOND,METER ;PUT SOME TIME ON THE METER

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 160  
 CZUAAB.MAC 07-APR-83 17:03 TEST 17: LINK MEMORY TEST

```

7496 033200' 004737 017316'      JSR    PC,CHKDNI      :WAIT FOR MICROTEST TO FINISH
7497 033204' 103021                BCC    40$           :OK, IT FINISHED
7498 033206' 004737 020132'      JSR    PC,CHKINT     :SEE IF ANY ERROR INTERRUPTS OCCURRED
7499 033212' 103006                BCC    35$           :NO, OK
7500 033214'                        ERRHRD 075,LNKMEM,MSG44 :PRINT ERROR MESSAGE
7501 033214' 104456                TRAP   CSERHRD
7502 033216' 000113                .WORD 75
7503 033220' 002240'                .WORD LNKMEM
7504 033222' 016442'                .WORD MSG44
7505 033224'                        ESCAPE TST           :LEAVE TEST
7506 033224' 104410                TRAP   CSERHRD
7507 033226' 000156                .WORD L10133-
7508 033230' 012702 000003      35$: MOV    #3,R2       :MICROTEST #
7509 033234'                        ERRHRD 076,LNKMEM,MSG12 :TELL MICROTEST HUNG
7510 033234' 104456                TRAP   CSERHRD
7511 033236' 000114                .WORD 76
7512 033240' 002240'                .WORD LNKMEM
7513 033242' 013466'                .WORD MSG12
7514 033244'                        ESCAPE TST
7515 033244' 104410                TRAP   CSERHRD
7516 033246' 000136                .WORD L10133-
7517
7518                               :MICROTEST IS COMPLETE, NOW CHECK FOR AN ERROR CONDITION
7519
7520 033250' 122777 000003 145062 40$: CMPB   #INERR,@PCSR1   :DID AN ERROR OCCUR?
7521 033256' 001027                BNE    47$           :NO
7522 033260' 122737 000000 000607' CMPB   #DATERR,PCBB+1 :YES, WAS IT A DATA ERROR?
7523 033266' 001003                BNE    45$           :NO
7524 033270' 012702 001446'      MOV    #SDATER,R2   :YES, POINT TO DATA ERROR STRING
7525 033274' 000406                BR     46$
7526 033276' 122737 000001 000607' 45$: CMPB   #ADRERR,PCBB+1 :WAS IT AN ADDRESS ERROR
7527 033304' 001014                BNE    47$           :NO
7528 033306' 012702 001461'      MOV    #SADRER,R2   :POINT TO ADDRESS ERROR STRING
7529 033312' 013701 000610'      46$: MOV    PCBB+2,R1   :GET FAILING ADDRESS
7530 033316' 013703 000612'      MOV    PCBB+4,R3   :GET GOOD DATA
7531 033322' 013704 000614'      MOV    PCBB+6,R4   :GET BAD DATA
7532 033326'                        ERRHRD 077,LNKMEM,MSG16 :PRINT ERROR MESSAGE
7533 033326' 104456                TRAP   CSERHRD
7534 033330' 000115                .WORD 77
7535 033332' 002240'                .WORD LNKMEM
7536 033334' 013636'                .WORD MSG16
7537 033336' 032777 010000 144772 47$: BIT    #PARERR,@PCSR0 :DID A PARITY ERROR OCCUR?
7538 033344' 001407                BEQ    50$           :NO
7539 033346'                        ERRHRD 078,LNKMEM,MSG43 :YES PRINT ERROR
7540 033346' 104456                TRAP   CSERHRD
7541 033350' 000116                .WORD 78
7542 033352' 002240'                .WORD LNKMEM
7543 033354' 016420'                .WORD MSG43
7544 033356' 012777 010000 144752 MOV    #PARERR,@PCSR0 :CLEAR PARITY ERROR FLAG
7545
7546                               :WRITE ONE TO CLEAR THE DNI BIT
7547
7548
7549 033364' 004737 017362'      50$: JSR    PC,CLRDN1   :CLEAR DNI
7550 033370' 103004                BCC    55$
7551 033372'                        ERRHRD 079,LNKMEM,RACMG7

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 161  
CZUAAB.MAC 07-APR-83 17:03 TEST 17: LINK MEMORY TEST

7552 033372' 104456  
7553 033374' 000117  
7554 033376' 002240'  
7555 033400' 012670'  
7556 033402'  
7557 033402'  
7558 033402'  
7559 033402' 104405  
7560 033404'  
7561 033404'  
7562 033404' 104401  
7563

558:  
ENDSEG

ENDTST

TRAP CSEHRD  
.WORD 79  
.WORD LNKMEM  
.WORD RACMG7

100018: TRAP CSESEG

L10133: TRAP CSETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 162  
 CZUAAB.MAC 07-APR-83 17:03 TEST 18: DMA 'TO' ADDRESS TEST

7564  
7565  
7566  
7567  
7568  
7569  
7570  
7571  
7572  
7573  
7574  
7575  
7576  
7577  
7578  
7579  
7580  
7581  
7582  
7583  
7584  
7585  
7586  
7587  
7588  
7589  
7590  
7591  
7592  
7593  
7594  
7595  
7596  
7597  
7598  
7599  
7600  
7601  
7602  
7603  
7604  
7605  
7606  
7607  
7608  
7609  
7610  
7611  
7612  
7613  
7614  
7615  
7616  
7617  
7618  
7619

033406'  
033406'  
  
  
  
  
  
033406'  
033406' 104404  
033410' 022737 000103 000326'  
033416' 001004  
033420' 122777 000001 144712  
033426' 001440  
033430' 012737 000103 000326' 5\$:  
033436' 004737 020340'  
033442' 103002  
033444'  
033444' 104410  
033446' 000314  
033450' 012737 000176 000332' 10\$:  
033456' 004737 017316'  
033462' 103022  
033464' 012737 001000' 000310'  
033472' 012737 001277' 000312'  
033500' 012737 001342' 000314'  
033506' 012737 001357' 000316'  
033514'  
033514' 104456  
033516' 000120  
033520' 002270'  
033522' 012716'  
033524'  
033524' 104410  
033526' 000234  
033530' 004737 017362' 20\$:

.SBTTL TEST 18: DMA 'TO' ADDRESS TEST  
 ;\*\*\*\*\*  
 ; THIS TEST WILL VERIFY THAT THE INTERNAL REGISTER 'DMATO' CAN BE READ  
 ; AND WRITTEN. THE T11 WILL BE USED TO WRITE AND READ THIS REGISTER.  
 ; THIS TEST REQUIRES THE USE OF CUSTOM MICROCODE MODULE C MICROTST #4.  
 ; PCBB+0 WILL BE WRITTEN WITH THE DATA PATTERN TO TEST, THE T11 WILL  
 ; WRITE THIS PATTERN TO THE 'DMATO' REGISTER AND READ IT BACK AND PUT  
 ; THE DATA READ INTO PCBB+2. THE DATA AT PCBB+2 WILL BE VERIFIED.  
 ;  
 ; TEST SEQUENCE:  
 ; 1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO  
 ; 2-LOAD PCBB+0 WITH DATA PATTERN  
 ; 3-START MICROCODE  
 ; 4-WAIT FOR 'DNI'  
 ; 5-VERIFY PCBB+2 FOR CORRECT PATTERN  
 ; 6-REPEAT STEPS 2-6 FOR ALL DATA PATTERNS  
 ;\*\*\*\*\*

BGNTST  
 T18::  
 ;  
 ; CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO  
 ; THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
 ; AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
 ;

BGNSEG  
 TRAP CSBSEG  
 CMP #'C,MICRO ;HAS MICROCODE MODULE 'C' BEEN LOADED?  
 BNE 5\$ ;NO  
 CMPB #INMON,@PCSR1 ;YES, IS THE MICROMONITOR ACTIVE?  
 BEQ 20\$ ;YES SKIP LOADING THE MICROMODULE  
 MOV #'C,MICRO ;NO  
 JSR PC,LODMIC ;GO LOAD MICROCODE MODULE C  
 BCC 10\$ ;OK  
 ESCAPE TST  
 TRAP C\$ESCAPE  
 .WORD L10134-.  
 MOV #2\*SECOND,METER  
 JSR PC,CHKDNI ;WAIT FOR MICROMONITOR TO TAKE OVER  
 BCC 20\$ ;OK  
 MOV #SDNI,BITNAM  
 MOV #SNSET,BISTA  
 MOV #SAFTER,PWHEN  
 MOV #SGTCMD,PCOMND  
 ERRHRD OBO,DMATO,MSG1  
 TRAP C\$ERHRD  
 .WORD 30  
 .WORD DMATO  
 .WORD MSG1  
 ESCAPE TST  
 TRAP C\$ESCAPE  
 .WORD L10134-.  
 JSR PC,CLRDN1 ;CLEAR DNI

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 163  
 CZUAAB.MAC 07-APR-83 17:03 TEST 18: DMA 'TO' ADDRESS TEST

```

7620 033534' 103006          BCC      25$
7621 033536'                ERRHRD   081,DMATO,RACMG7
7622 033536' 104456
7623 033540' 000121          TRAP     CSERHRD
7624 033542' 002270'        .WORD   81
7625 033544' 012670'        .WORD   DMATO
7626 033546'                .WORD   RACMG7
7627 033546' 104410          ESCAPE   TST
7628 033550' 000212          TRAP     CS$ESCAPE
7629 033552'                .WORD   L10134-.
7630 033552'                25$:
7631 033552'                ENDSEG
7632 033552' 104405          10000$:
7633                                TRAP     C$ESEG
7634                                ;
7635                                ;POINT TO LIST OF DATA PATTERNS TO USE, THERE ARE FIVE ENTRIES IN THE LIST
7636                                ;SO THE LOOP WILL BE EXECUTED 5 TIMES ONCE FOR EACH DATA PATTERN.
7637 033554' 012701 000520'   ;
7638 033560' 012705 000005'   ;      MOV     #PATERN,R1      ;GET ADDRESS OF DATA PATTERNS
7639 033564'                ;      MOV     #5,R5        ;NUMBER OF DATA PATTERNS
7640                                30$:
7641                                ;
7642                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE, LOAD PCBB+0 WITH
7643                                ;A DATA PATTERN, LOAD THE COMMAND FIELD OF PCSRO WITH 4 TO START THE EXECUTION
7644                                ;OF MICROTEST #4, DNI SETS WHEN IT IS COMPLETE
7645                                ;
7646                                ;      BGNSEG
7647 033564' 104404          ;
7648 033566' 004737 020060'   ;      JSR     PC,CHKMON      ;WAIT FOR MICROMONITOR
7649 033572' 103006          ;      BCC     35$          ;OK
7650 033574' 104456          ;      ERRHRD  082,DMATO,MSG46 ;PRINT ERROR
7651 033576' 000122          ;
7652 033600' 002270'        TRAP     CSERHRD
7653 033602' 016666'        .WORD   82
7654 033604'                .WORD   DMATO
7655 033604' 104410          .WORD   MSG46
7656 033606' 000154          ESCAPE   TST          ;LEAVE TEST
7657 033610' 012137 000606'   ;
7658 033614' 012777 000004 144514 ;
7659 033622' 012737 000176 000332' ;      MOV     (R1)+,PCBB+0 ;GET A DATA PATTERN
7660 033630' 004737 017316'   ;      MOV     #4,@PCSRO   ;START MICROTEST #4
7661 033634' 103021          ;      MOV     #2*SECOND,METER ;PUT SOME TIME ON THE METER
7662 033636' 004737 020132'   ;      JSR     PC,CHKDNI   ;WAIT FOR DNI
7663 033642' 103006          ;      BCC     40$          ;OK
7664 033644'                ;      JSR     PC,CHKINT   ;SEE IF ANY ERROR INTERRUPTS OCCURRED
7665 033644' 104456          ;      BCC     36$          ;NO, OK
7666 033646' 000123          ;      ERRHRD  083,DMATO,MSG44 ;PRINT ERROR MESSAGE
7667 033650' 002270'        TRAP     CSERHRD
7668 033652' 016442'        .WORD   83
7669 033654'                .WORD   DMATO
7670 033654' 104410          .WORD   MSG44
7671 033656' 000104          ESCAPE   TST          ;LEAVE TEST
7672 033660' 012702 000004          TRAP     CS$ESCAPE
7673 033664'                .WORD   L10134-.
7674 033664' 104456          36$:
7675 033666' 000124          ;      MOV     #4,R2      ;MICROTEST #
7676                                ;      ERRHRD  084,DMATO,MSG12 ;TELL MICROTEST HUNG
7677                                TRAP     CSERHRD
7678                                .WORD   84

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 164  
 CZJAAB.MAC 07-APR-83 17:03 TEST 18: DMA 'TO' ADDRESS TEST

```

7676 033670' 002270'
7677 033672' 013466'
7678 033674'
7679 033674' 104410
7680 033676' 000064
7681
7682
7683
7684
7685 033700' 013703 000606'
7686 033704' 042703 000001'
7687 033710' 020337 000610'
7688 033714' 001407
7689 033716' 013704 000610'
7690 033722'
7691 033722' 104456
7692 033724' 000125
7693 033726' 002270'
7694 033730' 013512'
7695 033732'
7696 033732'
7697 033732' 104405
7698
7699
7700
7701 033734' 004737 017362'
7702 033740' 103006
7703 033742'
7704 033742' 104456
7705 033744' 000126
7706 033746' 002270'
7707 033750' 012670'
7708 033752'
7709 033752' 104410
7710 033754' 000006
7711 033756' 005305
7712 033760' 001301
7713 033762'
7714 033762'
7715 033762' 104401

```

```

          ESCAPE TST
          TRAP C$ESLAPE
          .WORD L10134-.
          .WORD DMATO
          .WORD MSG12
          .WORD C$SERHRD
          .WORD B5
          .WORD DMATO
          .WORD MSG13
          .WORD C$ESEG
          .WORD C$SERHRD
          .WORD B6
          .WORD DMATO
          .WORD RACMG7
          .WORD C$ESCAPE
          .WORD L10134-.
          .WORD C$SETST
          .WORD L10134-.

```

```

:
:OK NOW CHECK TO SEE IF DATA READ IS SAME AS THE DATA WRITTEN
:REMEMBER BIT 0 OF DMATO IS NOT USED
:
40$:  MOV    PCBB+0,R3      ;GET ORIGINAL DATA PATTERN
      BIC    #BIT0,R3      ;STRIP LSB
      CMP    R3,PCBB+2     ;SEE IF DATA WRITTEN = DATA READ
      BEQ    S0$           ;YES
      MOV    PCBB+2,R4     ;NO, ERROR
      ERRHRD 0B5,DMATO,MSG13 ;PRINT ERROR

```

```

          ENDSEG
          TRAP C$SERHRD
          .WORD B5
          .WORD DMATO
          .WORD MSG13
          .WORD C$ESEG
          .WORD C$SERHRD
          .WORD B6
          .WORD DMATO
          .WORD RACMG7
          .WORD C$ESCAPE
          .WORD L10134-.
          .WORD C$SETST
          .WORD L10134-.

```

```

:
:WRITE ONE TO CLEAR DNI BIT
:
50$:  JSR    PC,CLRDNI     ;CLEAR DNI
      BCC    S5$
      ERRHRD 0B6,DMATO,RACMG7;ERROR DNI DID NOT CLEAR

```

```

          ESCAPE TST
          TRAP C$ESCAPE
          .WORD L10134-.
          .WORD C$SETST
          .WORD L10134-.

```

```

55$:  DEC    R5           ;ANY MORE DATA PATTERNS?
      BNE    S0$         ;YES

```

```

          ENDTST
          TRAP C$SETST
          .WORD L10134-.

```





65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 166  
CZUAAB.MAC 07-APR-83 17:03 TEST 19: DMA 'FROM' ADDRESS REGISTER TEST

7772	034074'	000127						.WORD	87
7773	034076'	002336'						.WORD	DMAFRM
7774	034100'	012716'						.WORD	MSG1
7775	034102'				ESCAPE	TST			
7776	034102'	104410						TRAP	C\$ESCAPE
7777	034104'	000236						.WORD	L10135-
7778	034106'	004737	017362'	208:	JSR	PC,CLRDNI			:CLEAR DNI BIT
7779	034112'	103906			BCC	258			
7780	034114'				ERRHRD	088,DMAFRM,RACMG7			:DNI BIT DID NOT CLEAR!
7781	034114'	104456						TRAP	C\$ERHRD
7782	034116'	000130						.WORD	88
7783	034120'	002336'						.WORD	DMAFRM
7784	034122'	012670'						.WORD	RACMG7
7785	034124'				ESCAPE	TST			
7786	034124'	104410						TRAP	C\$ESCAPE
7787	034126'	000214						.WORD	L10135-
7788	034130'			258:					
7789	034130'				ENDSEG				
7790	034130'								100008:
7791	034130'	104405						TRAP	C\$ESEG
7792	034132'	005037	000606'		CLR	PCBB+0			:TELL T11 TO START LOAD AT BASE
7793									:OF LINK MEMORY
7794	034136'			308:					
7795					:				:WAIT FOR THE MICROMONITOR TO ENTER 'IN MONITOR' STATE, LOAD COMMAND FIELD
7796					:				:OF PCSRO WITH 5 TO START THE EXECUTION OF MICROTEST #5, WAIT FOR DNI
7797					:				
7798					:				
7799	034136'				BGNSEG				
7800	034136'	104404						TRAP	C\$BSEG
7801	034140'	004737	020060'		JSR	PC,CHKNOI			:WAIT FOR MICROMONITOR
7802	034144'	103006			BCC	408			:OK
7803	034146'				ERRHRD	089,DMAFRM,R.646			:PRINT ERROR
7804	034146'	104456						TRAP	C\$ERHRD
7805	034150'	000131						.WORD	89
7806	034152'	002336'						.WORD	DMAFRM
7807	034154'	016666'						.WORD	MSG46
7808	034156'				ESCAPE	TST			:LEAVE TEST
7809	034156'	104410						TRAP	C\$ESCAPE
7810	034160'	000162						.WORD	L10135-
7811	034162'	012777	000005	144146	408:	MOV	#5,BPCSRO		:TELL T11 TO START MICROTEST #5
7812	034170'	012737	000176	000332'		MOV	#2*SECOND,METER		:WAIT A WHILE FOR DNI
7813	034176'	004737	017316'		JSR	PC,CHKDNI			
7814	034202'	103021			BCC	508			
7815	034204'	004737	020132'		JSR	PC,CHKINT			:SEE IF ANY ERROR INTERRUPTS OCCURRED
7816	034210'	103006			BCC	458			:NO, OK
7817	034212'				ERRHRD	090,DMAFRM,MSG44			:PRINT ERROR MESSAGE
7818	034212'	104456						TRAP	C\$ERHRD
7819	034214'	000132						.WORD	90
7820	034216'	002336'						.WORD	DMAFRM
7821	034220'	016442'						.WORD	MSG44
7822	034222'				ESCAPE	TST			:LEAVE TEST
7823	034222'	104410						TRAP	C\$ESCAPE
7824	034224'	000116						.WORD	L10135-
7825	034226'	012702	000005		458:	MOV	#5,R2		:WE WERE EXECUTING TEST 5
7826	034232'				ERRHRD	091,DMAFRM,MSG12			:TE'. HIM IT HUNG
7827	034232'	104456						TRAP	C\$ERHRD

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 167  
 CZUAAB.MAC 07-APR-83 17:03 TEST 19: DMA 'FROM' ADDRESS REGISTER TEST

```

7828 034234' 000133
7829 034236' 002336' .WORD 91
7830 034240' 013466' .WORD DMAFRM
7831 034242' .WORD MSG12
7832 034242' 104410 ESCAPE TST
7833 034244' 000076 TRAP C$ESCAPE
7834 .WORD L10135-
7835 :
7836 :OK GET THE 'FROM ADDRESS' THAT I PASSED TO THE MICROCODE AND MAKE IT AN
7837 :ACTUAL LINK MEMORY ADDRESS BY FORCING BIT 15 THIS MAPS IT TO THE 16-32K RANGE
7838 :ALSO FORCE BIT 2 BECAUSE THE FIRST 2 WORDS OF EACH BUFFER CAN NOT BE DMA'ED
7839 :FROM LINK MEMORY TO THE UNIBUS. CHECK THAT THE DATA TRANSFERRED IS GOOD
7840 034246' 013703 000606' 508: MOV PCBB+0,R3 :GET ORIGINAL 'FROM' ADDRESS
7841 034252' 052703 100004 BIS #BIT15!BIT2,R3 :MAKE IT ACTUAL LINK MEMORY ADDRESS
7842 034256' 013704 000610' MOV PCBB+2,R4 :GET WHAT WAS READ FROM LINK MEMORY
7843 034262' 020304 CMP R3,R4 :IS DATA CORRECT?
7844 034264' 001404 BEQ 558 :YES
7845 034266' ERRHRD 092,DMAFRM,MSG14 :NO
7846 034266' 104456 TRAP C$ERHRD
7847 034270' 000134 .WORD 92
7848 034272' 002336' .WORD DMAFRM
7849 034274' 013560' .WORD MSG14
7850 034276' 558:
7851 034276' ENDSEG
7852 034276'
7853 034276' 104405 100018: TRAP C$ESEG
7854 :
7855 :WRITE ONE TO CLEAR DNI
7856 :
7857 034300' 004737 017362' JSR PC,CLRDNI
7858 034304' 103006 BCC 578
7859 034306' ERRHRD 093,DMAFRM,RACMG7 :ERROR DNI DID NOT CLEAR!
7860 034306' 104456 TRAP C$ERHRD
7861 034310' 000135 .WORD 93
7862 034312' 002336' .WORD DMAFRM
7863 034314' 012670' .WORD RACMG7
7864 034316' ESCAPE TST
7865 034316' 104410 TRAP C$ESCAPE
7866 034320' 000022 .WORD L10135-
7867 034322' 022737 074000 000606' 578: CMP #74000,PCBB+0 :HAVE WE CHECKED ALL 1K CHUNKS?
7868 034330' 001404 BEQ 608 :YES
7869 034332' 062737 004000 000606' ADD #4000,PCBB+0 :NO, CHECK NEXT 1K
7870 034340' 000676 BR 308 :DO AGAIN
7871 034342' 608:
7872 034342' ENDTST
7873 034342'
7874 034342' 104401 L10135: TRAP C$ETST

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 168  
CZUAAB.MAC 07-APR-83 17:03 TEST 20: DMA BLOCK TRANSFER TEST

7875  
7876  
7877  
7878  
7879  
7880  
7881  
7882  
7883  
7884  
7885  
7886  
7887  
7888  
7889  
7890  
7891  
7892  
7893  
7894  
7895  
7896

.SBTTL TEST 20: DMA BLOCK TRANSFER TEST  
:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE DMA ENGINE CAN TRANSFER A MAXIMUM SIZE DATA  
:BLOCK TO HOST MEMORY.  
:THIS TEST USES CUSTOM MICROCODE MODULE C, MICROTEST #6. THE MICROTEST  
:FILLS EACH LOCATION OF LINK MEMORY WITH ITS ADDRESS AND THEN SETS  
:UP A TRANSFER FROM LINK MEMORY TO THE ADDRESS POINTED TO BY PCBB+0.  
:THE TRANSFER SIZE IS 1776 WORDS. AFTER THE MICROTEST FINISHES THE  
:BUFFER IS CHECKED TO SEE IF IT CONTAINS THE INCREMENTING ADDRESS PATTERN.  
:TEST SEQUENCE:  
: 1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO  
: 2-LOAD PCBB+0 WITH IIBUFFER ADDRESS  
: 3-START MICROCODE  
: 4-WAIT FOR DNI  
: 5-VERIFY ALL 1776 WORDS STARTING AT BUFFER ADDRESS BASE  
:\*\*\*\*\*

7897 034344'  
7898 034344'  
7899  
7900  
7901  
7902  
7903

BGNTST  
T20::  
:CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

7904 034344'  
7905 034344' 104404  
7906 034346' 022737  
7907 034354' 001004  
7908 034356' 122777  
7909 034364' 001435  
7910 034366' 012737  
7911 034374' 004737  
7912 034400' 103530  
7913 034402' 012737  
7914 034410' 004737  
7915 034414' 103021  
7916 034416' 012737  
7917 034424' 012737  
7918 034432' 012737  
7919 034440' 012737  
7920 034446'  
7921 034446' 104456  
7922 034450' 000136  
7923 034452' 002406'  
7924 034454' 012716'  
7925 034456' 000501  
7926 034460' 004737  
7927 034464' 103005  
7928 034466'  
7929 034466' 104456  
7930 034470' 000137

BGNSEG  
TRAP CSBSEG  
CMP #'C,MICRO ;HAS MICRO MODULE C BEEN LOADED?  
S8 ;NO  
BNE 58  
CMPB #INMON,APCSR1 ;YES, IS THE MICROMONITOR ACTIVE?  
208 ;YES SKIP LOADING THE MICROMODULE  
BEQ 208 ;NO, LOAD MICRO MODULE C  
MOV #'C,MICRO  
JSR PC,LODMIC  
BCS 708 ;ERROR  
MOV #2\*SECOND,METER ;PUT SOME TIME ON THE METER  
JSR PC,CHKDNI ;WAIT FOR THE MICROMONITOR  
BCC 208 ;OK  
MOV #SDNI,BITNAM  
MOV #SNSET,BITSTA  
MOV #AFTER,PWHEN  
MOV #SGTCHD,PCOMND  
ERRHRD 094,DHABLK,MSG1  
TRAP CSERHRD  
.WORD 94  
.WORD DHABLK  
.WORD MSG1  
BR 708  
JSR PC,CLRDMI ;GO CLEAR THE DNI BIT  
BCC 258  
ERRHRD 095,DHABLK,RACMG7 ;DNI DID NOT CLEAR!  
TRAP CSERHRD  
.WORD 95

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 169  
 CZUAAB.MAC 07-APR-83 17:03 TEST 20: DMA BLOCK TRANSFER TEST

```

7931 034472' 002406' .WORD DMABLK
7932 034474' 012670' .WORD RACMG7
7933 034476' 000471 BR 708
7934 034500' 258:
7935 034500' ENDSEG
7936 034500' 100008:
7937 034500' 104405 TRAP CSESEG
7938
7939 ;
7940 ; TELL MICROCODE TO DMA TO A BUFFER IN FREE MEMORY LOCATED ABOVE THIS DIAGNOSTIC
7941 ; BY LOADING PCBB+0 WITH THE ADDRESS TO DMA TO. WAIT FOR THE MICROMONITOR TO
7942 ; ENTER THE 'IN MONITOR' STATE AND LOAD THE COMMAND FIELD OF PCSRO WITH /, 6
7943 ; TO START THE EXECUTION OF MICROTEST #6. WAIT FOR DMI TO SET INDICATING IT IS
7944 ; COMPLETE
7945 ;
7946 ; BGNSEG
7947 034502' 104404 TRAP C8BSEG
7948 034504' 013737 000324' 000606' MOV FREEM,PCBB+0 ;GET ADDRESS OF FREE MEMORY
7949 034512' 004737 020060' JSR PC,CHKMON ;WAIT FOR MICROMONITOR
7950 034516' 103006 BCC 308 ;OK
7951 034520' 104456 ERRHRD 096,DMABLK,MSG46 ;PRINT ERROR
7952 034522' 000140 TRAP CSERHRD
7953 034524' 002406' .WORD 96
7954 034526' 016666' .WORD DMABLK
7955 034530' ESCAPE TST ;LEAVE TEST .WORD MSG46
7956 034530' 104410 TRAP C8SEscape
7957 034532' 000136 .WORD L10136-
7958 034534' 012777 000006 143574 308: MOV #6,BPCSRO ;START MICROTEST #6
7959 034542' 012737 000473 000332' MOV #5*SECOND,METER ;PUT SOME TIME ON THE METER
7960 034550' 004737 017316' JSR PC,CHKDMI ;WAIT FOR MICROTEST FINISH
7961 034554' 103020 BCC 408 ;OK
7962 034556' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
7963 034562' 103006 BCC 358 ;NO, OK
7964 034564' ER2HRD 097,DMABLK,MSG44 ;PRINT ERROR MESSAGE
7965 034564' 104456 TRAP CSERHRD
7966 034566' 000141 .WORD 97
7967 034570' 002406' .WORD DMABLK
7968 034572' 016442' .WORD MSG44
7969 034574' ESCAPE TST ;LEAVE TEST
7970 034574' 104410 TRAP C8SEscape
7971 034576' 000072 .WORD L10136-
7972 034600' 012702 000006 358: MOV #6,R2 ;MICROTEST NEVER FINISHED!
7973 034604' ER2HRD 098,DMABLK,MSG12
7974 034604' 104456 TRAP CSERHRD
7975 034606' 000142 .WORD 98
7976 034610' 002406' .WORD DMABLK
7977 034612' 013466' .WORD MSG12
7978 034614' 000422 BR 708
7979 ;
7980 ; OK NOW CHECK THE DATA TRANSFERRED FROM LINK MEMORY TO UNIBUS MEMORY.
7981 ; THE DATA PATTERN IS ACTUALLY THE ADDRESS FROM WHICH THE DATA CAME FROM.
7982 ; THE FIRST LOCATION WILL BE 4 BYTES FROM THE BEGINING OF LINK MEMORY BECAUSE
7983 ; THE DMA FROM ADDRESS REGISTER DOES NOT USE BITS 0 AND 1.
7984 ;
7985 034616' 013702 000606' 408: MOV PCBB+0,R2 ;GET ADDRESS OF BUFFER
7986 034622' 012746 100004 MOV #100004,-(SP) ;STARTING DATA PATTERN

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 170  
CZUAAB.MAC 07-APR-83 17:33 TEST 20: DMA BLOCK TRANSFER TEST

7987 034626' 021216  
7988 034630' 001405  
7989 034632' 010601  
7990 034634' 104456  
7991 034634' 000143  
7992 034640' 002406'  
7993 034642' 013606'  
7994 034644' 021627 103776  
7996 034650' 002004  
7997 034652' 062716 000002  
7998 034656' 005722  
7999 034660' 000762  
8000 034662' 004737 020166'  
8001 034666'  
8002 034666'  
8003 034666' 104405  
8004 034670'  
8005 034670'  
8006 034670' 104401

508: CMP (R2),(SP)  
BEQ 608  
MOV SP,R1  
ERRHRD 009,DNABLK,MSG15

:IS DATA PATTERN CORRECT?  
:YES  
:NO GET GOOD DATA  
:PRINT ERROR MESSAGE

TRAP CSEHRD  
.WORD 99  
.WORD DNABLK  
.WORD MSG15

608: CMP (SP),#103776  
BGE 708  
ADD #2,(SP)  
TST (R2)+  
BR 508  
708: JSR PC,REUNA  
ENDSEG

:DONE ALL DATA?  
:YES  
:NEXT GOOD DATA PATTERN  
:NEXT BUFFER ADDRESS  
:CONTINUE CHECKING  
:RESTORE OPERATIONAL MICROCODE

10G018: TRAP CSESEG

ENDTST

L10136: TRAP CSETST

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 171  
CZUAAB.MAC 07-APR-83 17:03 TEST 21: TRANSMIT DONE TEST

8007  
8008  
8009  
8010  
8011  
8012  
8013  
8014  
8015  
8016  
8017  
8018  
8019  
8020  
8021  
8022  
8023  
8024  
8025  
8026  
8027  
8028  
8029  
8030  
8031  
8032 034672'  
8033 034672'  
8034  
8035  
8036  
8037  
8038  
8039 034672'  
8040 034672' 104404  
8041 034674' 022737 000104 000326'  
8042 034702' 001004  
8043 034704' 122777 000001 143426  
8044 034712' 001440  
8045 034714' 012737 000104 000326' 5%:  
8046 034722' 004737 020340'  
8047 034726' 103002  
8048 034730'  
8049 034730' 104410  
8050 034732' 000246  
8051 034734' 012737 000176 000332' 10%:  
8052 034742' 004737 017316'  
8053 034746' 103022  
8054 034750' 012737 001000' 000310'  
8055 034756' 012737 001277' 000312'  
8056 034764' 012737 001342' 000314'  
8057 034772' 012737 001357' 000316'  
8058 035000'  
8059 035000' 104456  
8060 035002' 000144  
8061 035004' 002445'  
8062 035006' 012716'

.SBTTL TEST 21: TRANSMIT DONE TEST

:\*\*\*\*\*  
:THE TRANSMIT STATE MACHINE INFORMS THE PORT MODULE PROCESSOR OF A  
: 'TRANSMIT DONE' CONDITION. IT DOES THIS BY GENERATING AN INTERRUPT WHENEVER  
: IT FINISHES TRANSMITTING A DATAGRAM. SINCE THE 'TRANSMIT DONE' INTERRUPT IS  
: A NECESSARY CONDITION OF EVERY DATAGRAM TRANSMISSION, THIS TEST WILL USE THE  
: INTERRUPT TO INDICATE THAT THE TRANSMIT STATE MACHINE IS FUNCTIONING.  
:  
: MICROCODE MODULE D MICROTTEST #1 WILL BE USED FOR THIS TEST. IT SETS  
: UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A  
: TRANSMIT INTERRUPT. THE T-11 WILL BE RELEASED FROM THE LOOP IF THE XMIT DONE  
: INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING  
: THAT THE TEST IS COMPLETE.  
:  
: TEST SEQUENCE:  
: 1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO  
: 2-VERIFY THE MICROMONITOR IN THE 'IN MONITOR' STATE  
: 3-SELECT MICROTTEST #1  
: 4-VERIFY 'DNI' BIT IN PCSRO AFTER A REASONABLE PERIOD OF TIME  
: 5-WRITE ONE TO CLEAR DNI  
:\*\*\*\*\*

BGNTST

T21::

:CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
: THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'  
: AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG

TRAP CSBSEG  
;HAS MICROCODE MODULE 'D' BEEN LOADED  
;NO  
;YES, IS THE MICROMONITOR ACTIVE?  
;YES SKIP LOADING THE MICROMODULE  
;GO LOAD MICRO MODULE 'D'  
;OK  
TRAP CSESCAPE  
.WORD L10137-.  
;WAIT FOR THE MICROMONITOR  
;OK  
TRAP CSERHRD  
.WORD 100  
.WORD TRNDON  
.WORD MSG1

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 172  
 CZUAAB.MAC 07-APR-83 17:03 TEST 21: TRANSMIT DONE TEST

```

8063 035010'          ESCAPE TST
8064 035010' 104410
8065 035012' 000166          TRAP  C$ESCAPE
8066 035014' 004737 017362' 20$: JSR  PC,CLRDMI          ;CLEAR DNI BIT      .WORD  L10137-.
8067 035020' 103006          BCC  25$
8068 035022'          ERRHRD 101,TRNDON,RACMG7 ;DNI DID NOT CLEAR!
8069 035022' 104456          .WORD  C$ERHRD
8070 035024' 000145          .WORD  101
8071 035026' 002445'        .WORD  TRNDON
8072 035030' 012670'        .WORD  RACMG7
8073 035032'          ESCAPE TST
8074 035032' 104410          TRAP  C$ESCAPE
8075 035034' 000144          .WORD  L10137-.
8076 035036'          25$:
8077 035036'          ENDSEG
8078 035036'          10000$:
8079 035036' 104405          TRAP  C$ESEG
8080
8081 ;
8082 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. LOAD THE COMMAND
8083 ;FIELD BITS OF PC$RO WITH A 1 TO START THE EXECUTION OF MICROTEST #1.
8084 ;WAIT ABOUT 1 SECOND FOR IT TO FINISH. IF NO 'DNI' SET PRINT ERROR.
8085 ;
8085 035040'          BGNSEG
8086 035040' 104404          TRAP  C$BSEG
8087 035042' 004737 020060' JSR  PC,CHKMON          ;WAIT FOR MICROMONITOR
8088 035046' 103006          BCC  30$              ;OK
8089 035050'          ERRHRD 102,TRNDON,MSG46 ;PRINT ERROR
8090 035050' 104456          .WORD  C$ERHRD
8091 035052' 000146          .WORD  102
8092 035054' 002445'        .WORD  TRNDON
8093 035056' 016666'        .WORD  MSG46
8094 035060'          ESCAPE TST          ;LEAVE TEST
8095 035060' 104410          TRAP  C$ESCAPE
8096 035062' 000116          .WORD  L10137-.
8097 035064' 012777 000001 143244 30$: MOV  #1,@PC$RO          ;TELL T11 TO EXECUTE FIRST MICROTEST
8098 035072' 012737 000077 000332' MOV  #1*SECOND,METER ;WAIT FOR DNI
8099 035100' 004737 017316' JSR  PC,CHKDMI
8100 035104' 103025          BCC  40$              ;OK-'DNI' SET
8101          ;ERROR 'DNI' DID NOT SET! NO TRANSMIT
8102          ;INTERRUPT!
8103          ERRHRD 103,TRNDON,MSG36 ;PRINT ERROR MESSAGE
8104 035106' 104456          TRAP  C$ERHRD
8105 035110' 000147          .WORD  103
8106 035112' 002445'        .WORD  TRNDON
8107 035114' 015770'        .WORD  MSG36
8108 035116' 004737 020132' JSR  PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
8109 035122' 103006          BCC  35$              ;NO, OK
8110 035124'          ERRHRD 103,TRNDON,MSG44 ;PRINT ERROR MESSAGE
8111 035124' 104456          TRAP  C$ERHRD
8112 035126' 000147          .WORD  103
8113 035130' 002445'        .WORD  TRNDON
8114 035132' 016442'        .WORD  MSG44
8115 035134'          ESCAPE TST          ;LEAVE TEST
8116 035134' 104410          TRAP  C$ESCAPE
8117 035136' 000042          .WORD  L10137-.
8118 035140' 012702 000001 35$: MOV  #1,R2          ;MICROTEST #1 IS HUNG

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 173  
CZUAAB.MAC 07-APR-83 17:03 TEST 21: TRANSMIT DONE TEST

8119	035144'		ERRHRD 104,TRNDON,MSG12		;PRINT MESSAGE ABOUT HUNG MICROTST
8120	035144'	104456			TRAP C\$ERHRD
8121	035146'	000150			.WORD 104
8122	035150'	002445'			.WORD TRNDON
8123	035152'	013466'			.WORD MSG12
8124	035154'		ESCAPE TST		
8125	035154'	104410			TRAP C\$ESCAPE
8126	035156'	000022			.WORD L10137-
8127					
8128			;WRITE 1 TO CLEAR DNI BIT		
8129					
8130	035160'	004737 017362'	40\$: JSR PC,CLRDNI		;CLEAR DNI BIT
8131	035164'	103004	BCC 55\$		
8132	035166'		ERRHRD 105,TRNDON,RACMG7		;ERROR DNI DID NOT CLEAR!
8133	035166'	104456			TRAP C\$ERHRD
8134	035170'	000151			.WORD 105
8135	035172'	002445'			.WORD TRNDON
8136	035174'	012670'			.WORD RACMG7
8137	035176'		55\$:		
8138	035176'		ENDSEG		
8139	035176'				
8140	035176'	104405			100018: TRAP C\$ESEG
8141	035200'		ENDTST		
8142	035200'				L10137: TRAP C\$ESET
8143	035200'	104401			



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 174  
CZUAAB.MAC 07-APR-83 17:03 TEST 22: RECEIVER DONE TEST

8144  
8145  
8146  
8147  
8148  
8149  
8150  
8151  
8152  
8153  
8154  
8155  
8156  
8157  
8158  
8159  
8160  
8161  
8162  
8163  
8164  
8165  
8166  
8167  
8168  
8169  
8170  
8171  
8172  
8173  
8174  
8175  
8176  
8177  
8178  
8179  
8180  
8181  
8182  
8183  
8184  
8185  
8186  
8187  
8188  
8189  
8190  
8191  
8192  
8193  
8194  
8195  
8196  
8197  
8198  
8199

.SBTTL TEST 22: RECEIVER DONE TEST

\*\*\*\*\*  
:THE LINK HARDWARE INCLUDES LOGIC TO TELL THE DEUNA PROCESSOR WHEN  
:A LINK MEMORY BUFFER HAS BEEN FILLED AND DATA IS AVAILABLE FOR PROCESSING.  
:THE HARDWARE INTERRUPTS THE DEUNA PROCESSOR, BECAUSE THE INTERRUPT HAPPENS  
:WHEN A LINK MEMORY BUFFER IS FULL AND THE LINK MEMORY IS FILLED BY THE  
:OPERATION OF THE RECEIVE STATE MACHINE, THE INTERRUPT CAN BE USED TO CHECK  
:IF THE STATE MACHINE WORKS.  
:MICROCODE MODULE D MICROTEST #2 WILL BE USED FOR THIS TEST. IT SETS  
:UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A  
:RECEIVER INTERRUPT. THE T-11 WILL BE RELEASED FROM THE LOOP IF THE  
:INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING  
:THAT THE TEST IS COMPLETE.  
:TEST SEQUENCE:  
: 1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO  
: 2-VERIFY THE MICROMONITOR IN THE 'IN MONITOR' STATE  
: 3-SELECT MICROTEST #2  
: 4-VERIFY 'DNI' BIT IN PCSRO AFTER A REASONABLE PERIOD OF TIME  
: 5-WRITE ONE TO CLEAR DNI  
\*\*\*\*\*

035202'  
035202'

BGNTST

T22::

:CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG

035202'  
035202' 104404  
035204' 022737  
035212' 001004  
035214' 122777  
035222' 001440  
035224' 012737  
035232' 004737  
035236' 103002  
035240'  
035240' 104410  
035242' 000246  
035244' 012737  
035252' 004737  
035256' 103022  
035260' 012737  
035266' 012737  
035274' 012737  
035302' 012737  
035310'  
035312' 104456  
035312' 000152  
035314' 002477'

000104 000326'  
000001 143116  
000104 000326' 5\$:  
020340'  
000176 000332' 10\$:  
017316'  
001000' 000310'  
001277' 000312'  
001342' 000314'  
001357' 000316'

CMP #D,MICRO  
BNE 5\$  
CMPB #INMON,@PCSR1  
BEQ 20\$  
MOV #D,MICRO  
JSR PC,LODMIC  
BCC 10\$  
ESCAPE TST  
MOV #2\*SECOND,METER  
JSR PC,CHKDNI  
BCC 20\$  
MOV #SDNI,BITNAM  
MOV #SNSET,BITSTA  
MOV #SAFTER,PWHEN  
MOV #SGTCMD,PCOMND  
ERRHRD 106,RCVDON,MSG1

TRAP CSBSEG  
:HAS MICROCODE MODULE 'D' BEEN LOADED  
:NO  
:YES, IS THE MICROMONITOR ACTIVE?  
:YES SKIP LOADING THE MICROMODULE  
:GO LOAD MICRO MODULE 'D'

:OK

TRAP C\$ESCAPE  
.WORD L10140-

:WAIT FOR THE MICROMONITOR

:OK

TRAP C\$ERHRD  
.WORD 106  
.WORD RCVDON

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 175  
 CZUAAB.MAC 07-APR-83 17:03 TEST 22: RECEIVER DONE TEST

```

8200 035316' 012716'
8201 035320'          ESCAPE TST          .WORD  MSG1
8202 035320' 104410
8203 035322' 000166          TRAP    C$ESCAPE
8204 035324' 004737 017362'          .WORD  L10140-.
8205 035330' 103006          20$:   JSR    PC,LLRDN1          ;CLEAR DNI BIT
8206 035332'          BCC    25$
8207 035332' 104456          ERRHRD 107,RCVDON,RACMG7          ;DNI DID NOT CLEAR!
8208 035334' 000153          TRAP    C$ERHRD
8209 035336' 002477'          .WORD  107
8210 035340' 012670'          .WORD  RCVDON
8211 035342'          .WORD  RACMG7
8212 035342' 104410          ESCAPE TST
8213 035344' 000144          TRAP    C$ESCAPE
8214 035346'          .WORD  L10140-.
8215 035346'          25$:
8216 035346'          ENDSEG
8217 035346' 104405          10000$:
8218
8219          ;
8220          ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. LOAD THE COMMAND
8221          ;FIELD BITS OF PCSRO WITH A 2 TO START THE EXECUTION OF MICROTEST #2.
8222          ;WAIT ABOUT 1 SECOND FOR IT TO FINISH. IF NO 'DNI' SET PRINT ERROR.
8223 035350'          ;
8224 035350' 104404          BGNSEG
8225 035352' 004737 020060'          JSR    PC,CHKMON          ;WAIT FOR MICROMONITOR
8226 035356' 103006          BCC    30$
8227 035360'          ERRHRD 108,RCVDON,MSG46          ;OK
8228 035360' 104456          ;PRINT ERROR
8229 035362' 000154          TRAP    C$ERHRD
8230 035364' 002477'          .WORD  108
8231 035366' 016666'          .WORD  RCVDON
8232 035370'          .WORD  MSG46
8233 035370' 104410          ESCAPE TST          ;LEAVE TEST
8234 035372' 000116          TRAP    C$ESCAPE
8235 035374' 012777 000002 142734 30$:   MOV    #2,@PCSRO          .WORD  L10140-.
8236 035402' 012737 000077 000332'   MOV    #1*SECOND,METER          ;TELL T11 TO EXECUTE MICROTEST #2
8237 035410' 004737 017316'   JSR    PC,CHKDNI          ;WAIT FOR DNI
8238 035414' 103025          BCC    40$
8239          ;OK DNI SET
8240          ;ERROR-DNI NOT SET, NO RECEIVER
8241 035416'          ERRHRD 109,RCVDON,MSG37          ;INTERRUPT
8242 035416' 104456          ;PRINT ERROR MESSAGE
8243 035420' 000155          TRAP    C$ERHRD
8244 035422' 002477'          .WORD  109
8245 035424' 016012'          .WORD  RCVDON
8246 035426' 004737 020132'          .WORD  MSG37
8247 035432' 103006          JSR    PC,CHKINT          ;SEE IF ANY ERRGR INTERRUPTS OCCURRED
8248 035434'          BCC    35$
8249 035434' 104456          ERRHRD 110,RCVDON,MSG44          ;NO, OK
8250 035436' 000156          ;PRINT ERROR MESSAGE
8251 035440' 002477'          TRAP    C$ERHRD
8252 035442' 016442'          .WORD  110
8253 035444'          .WORD  RCVDON
8254 035444' 104410          ESCAPE TST          .WORD  MSG44
8255 035446' 000042          TRAP    C$ESCAPE
          .WORD  L10140-.

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 176  
CZUAAB.MAC 07-APR-83 17:03 TEST 22: RECEIVER DONE TEST

8256 035450' 012702 000002  
8257 035454'  
8258 035454' 104456  
8259 035456' 000157  
8260 035460' 002477'  
8261 035462' 013466'  
8262 035464'  
8263 035464' 104410  
8264 035466' 000022  
8265  
8266  
8267  
8268 035470' 004737 017362'  
8269 035474' 103004  
8270 035476'  
8271 035476' 104456  
8272 035500' 000160  
8273 035502' 002477'  
8274 035504' 012670'  
8275 035506'  
8276 035506'  
8277 035506'  
8278 035506' 104405  
8279 035510'  
8280 035510'  
8281 035510' 104401

35\$: MOV #2,R2  
ERRHRD 111,RCVDON,MSG12

:MICROTEST #2 IS HUNG  
:PRINT ERPR MESSAGE

TRAP C\$ERHRD  
.WORD 111  
.WORD RCVDON  
.WORD MSG12

ESCAPE TST

TRAP C\$ESCAPE  
.WORD L10140~.

:WRITE 1 TO CLEAR DNI BIT

40\$: JSR PC,CLRDNI  
BCC 55\$  
ERRHRD 112,RCVDON,RACMG7

:CLEAR DNI BIT  
:ERROR DNI DID NOT CLEAR!

TRAP C\$ERHRD  
.WORD 112  
.WORD RCVDON  
.WORD RACMG7

55\$:  
ENDSEG

ENDTST

10001\$: TRAP C\$ESEG

L10140: TRAP C\$ETST



65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 178  
CZUAAB.MAC 07-APR-83 17:03 TEST 23: DATA BYTE FRAMING TEST

```

8338 035562' 004737 017316' JSR PC,CHKDNI
8339 035566' 103022 BCC 208 ;OK
8340 035570' 012737 001000' 000310' MOV #SDNI,BITNAM ;ERROR 'DNI' BIT NEVER SET
8341 035576' 012737 001277' 000312' MOV #NSET,BITSTA
8342 035604' 012737 001342' 000314' MOV #SALTER,PWHEN
8343 035612' 012737 001357' 000316' MOV #SGTCMD,PCOMND
8344 035620' ERRHRD 113,DBFRAM,MSG1
8345 035620' 104456 TRAP C$ERHRD
8346 035622' 000161 .WORD 113
8347 035624' 002531' .WORD DBFRAM
8348 035626' 012716' .WORD MSG1
8349 035630' ESCAPE TST
8350 035630' 104410 TRAP C$ESCAPE
8351 035632' 000236 .WORD L10141-.
8352 035634' 004737 017362' 208: JSR PC,CLRDN1 ;CLEAR DNI BIT
8353 035640' 103006 BCC 258
8354 035642' ERRHRD 114,DBFRAM,RACMG7 ;DNI DID NOT CLEAR!
8355 035642' 104456 TRAP C$ERHRD
8356 035644' 000162 .WORD 114
8357 035646' 002531' .WORD DBFRAM
8358 035650' 012670' .WORD RACMG7
8359 035652' ESCAPE TST
8360 035652' 104410 TRAP C$ESCAPE
8361 035654' 000214 .WORD L10141-.
8362 035656' 258:
8363 035656' ENDSEG
8364 035656' 100008: TRAP C$ESEG
8365 035656' 104405
8366
8367 ;
8368 ;WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE. CLEAR THE PCBB. LOAD THE
8369 ;COMMAND FIELD OF PCSRO WITH A 3 TO START THE EXECUTION OF MICROTEST #3.
8370 ;WAIT FOR THE 'DNI' BIT. IF NO 'DNI' OR IF ANY EXTRANEIOUS INTERRUPTS HAPPEN
8371 ;PRINT ERROR
8372 ;
8373 BGNSEG
8374 035660' 104404 TRAP C$BSEG
8375 035662' 004737 020060' JSR PC,CHKMON ;WAIT FOR MICROMONITOR
8376 035666' 103022 BCC 308 ;OK
8377 035670' ERRHRD 115,DBFRAM,MSG46 ;PRINT ERROR
8378 035670' 104456 TRAP C$ERHRD
8379 035672' 000163 .WORD 115
8380 035674' 002531' .WORD DBFRAM
8381 035676' 016666' .WORD MSG46
8382 035700' ESCAPE TST ;LEAVE TEST
8383 035700' 104410 TRAP C$ESCAPE
8384 035702' 000166 .WORD L10141-.
8385 035704' 005037 000606' 308: CLR PCBB ;THIS IS WHERE MICROCODE WILL PUT...
8386 ;RECEIVE BUFFER STATUS WORD
8387 035710' 005037 000610' CLR PCBB+2 ;HERE IS WHERE THE GOOD DATA GOES
8388 035714' 005037 000612' CLR PCBB+4 ;HERE IS WHERE THE BAD DATA GOES
8389 035720' 005037 000614' CLR PCBB+6 ;HERE IS WHERE THE BUFFER OFFSET GOES
8390 035724' 012777 000003 142404 MOV #3,PCSRO ;TELL T11 TO EXECUTE MICROTEST #3
8391 035732' 012737 000176 000332' MOV #2*SECOND,M.TER ;WAIT FOR DNI
8392 035740' 004737 017316' JSR PC,CHKDNI
8393 035744' 103021 BCC 408

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 179  
 CZUAAB.MAC 07-APR-83 17:03 TEST 23: DATA BYTE FRAMING TEST

```

8394 035746' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
8395 035752' 103006 BCC 558 ;NO, OK
8396 035754' ERRHRD 116,DBFRAM,MSG44 ;PRINT ERROR MESSAGE
8397 035754' 104456 TRAP CSERHRD
8398 035756' 000164 .WORD 116
8399 035760' 002531' .WORD DBFRAM
8400 035762' 016442' .WORD MSG44
8401 035764' ESCAPE TST
8402 035764' 104410 TRAP CSESCAPE
8403 035766' 000102 .WORD L10141-.
8404 035770' 012702 000003 358: MOV #3,R2 ;MICROTEST #3 IS HUNG
8405 035774' ERRHRD 117,DBFRAM,MSG12
8406 035774' 104456 TRAP CSERHRD
8407 035776' 000165 .WORD 117
8408 036000' 002531' .WORD DBFRAM
8409 036002' 013466' .WORD MSG12
8410 036004' ESCAPE TST
8411 036004' 104410 TRAP CSESCAPE
8412 036006' 000062 .WORD L10141-.
8413 ;
8414 ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF THE T-11 FOUND A BOGUS BYTE IN THE
8415 ;RECEIVED DATA. IF SO, GET THE INFORMATION FROM THE PCBB AND PRINT ERROR
8416 ;
8417 036010' 122777 000003 142322 408: CMPB #INERR,PCSR1 ;DID AN ERROR OCCUR?
8418 036016' 001014 BNE 508 ;NO
8419 036020' 013701 000614' MOV PCBB+6,R1 ;GET BUFFER OFFSET FOR ERROR REPORT
8420 036024' 013702 000610' MOV PCBB+2,R2 ;THIS IS THE GOOD DATA FOR ERROR REPORT
8421 036030' 013703 000612' MOV PCBB+4,R3 ;THIS IS THE BAD DATA FOR ERROR REPORT
8422 036034' 013704 000606' MOV PCBB,R4 ;THIS IS THE RECEIVER STATUS WORD
8423 036040' ERRHRD 118,DBFRAM,MSG17 ;PRINT ERROR MESSAGE
8424 036040' 104456 TRAP CSERHRD
8425 036042' 000166 .WORD 118
8426 036044' 002531' .WORD DBFRAM
8427 036046' 013730' .WORD MSG17
8428 ;
8429 ;WRITE '1' TO CLEAR 'DNI' BIT
8430 ;
8431 036050' 004737 017362' 508: JSR PC,CLRDN1 ;CLEAR DNI BIT
8432 036054' 103004 BCC 558
8433 036056' ERRHRD 119,DBFRAM,RACMG7 ;ERROR DNI DID NOT CLEAR!
8434 036056' 104456 TRAP CSERHRD
8435 036060' 000167 .WORD 119
8436 036062' 002531' .WORD DBFRAM
8437 036064' 012670' .WORD RACMG7
8438 036066' 558:
8439 036066' ENDSEG
8440 036066' 100018:
8441 036066' 104405 TRAP CSESEG
8442 ^36070' ENDTST
8443 ^6070'
8444 ^6070' 104401 TRAP CSETST

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 180  
CZUAAB.MAC 07-APR-83 17:03 TEST 24: DATA WORD FRAMING TEST

8445  
8446  
8447  
8448  
8449  
8450  
8451  
8452  
8453  
8454  
8455  
8456  
8457  
8458  
8459  
8460  
8461  
8462  
8463  
8464  
8465  
8466  
8467  
8468  
8469  
8470  
8471  
8472  
8473  
8474  
8475  
8476  
8477  
8478  
8479  
8480  
8481  
8482  
8483  
8484  
8485  
8486  
8487  
8488  
8489  
8490  
8491  
8492  
8493  
8494  
8495  
8496  
8497  
8498  
8499  
8500

036072'  
036072'  
  
  
  
  
  
  
  
036072'  
036072' 104404  
036074' 022737 000104 000326'  
036102' 001004  
036104' 122777 000001 142226  
036112' 001440  
036114' 012737 000104 000326' 58:  
036122' 004737 020340'  
036126' 103002  
036130'  
036130' 104410  
036132' 000316  
036134' 012737 000176 000332' 108:  
036142' 004737 017316'

.SBTTL TEST 24: DATA WORD FRAMING TEST  
.....  
: THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR WORD DATA BOUNDARY  
: CONDITIONS.  
: THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE  
: ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT WORDS  
: IN THE DATA STREAM (I.E. 11111111111111110000000000000011...) THE T-11  
: PROCESSOR WILL VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO  
: THE RECEIVER DATA BUFFER.  
: THIS TEST WILL USE MICROCODE MODULE 'D' MICROTST #4. TESTING OF THE DATA  
: FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE,  
: WILL WAIT FOR A 'DNI' IN REGISTER PCSRO. IF 'DNI' APPEARS, THE HOST PROCESSOR  
: WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET,  
: ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:  
: PCBB+0: RECEIVER STATUS WORD  
: PCBB+2: DATA TRANSMITTED  
: PCBB+4: DATA RECEIVED  
: PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA  
: TEST SEQUENCE:  
: 1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO  
: 2-VERIFY MICROMONITOR IS IN THE 'INNON' STATE  
: 3-CLEAR OUT THE PCBB+0,+2,+4,+6  
: 4-SELECT MICROTST #4  
: 5-VERIFY 'DNI' BIT SET IN PCSRO  
: 6-CHECK FOR AN ERROR CONDITION IN PCSR1  
: 7-WRITE ONE TO CLEAR THE DNI BIT  
.....  
BGNTST  
T24::  
: CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
: THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'  
: AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
: BGNSEG  
: TRAP CSBSEG  
: HAS MICROCODE MODULE 'D' BEEN LOADED  
: NO  
: YES, IS THE MICROMONITOR ACTIVE?  
: YES SKIP LOADING THE MICROMODULE  
: GO LOAD MICRO MODULE 'D'  
: OK  
: TRAP CSBSEG  
: WORD L10142-  
: WAIT FOR THE MICROMONITOR

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 181  
 CZUAAB.MAC 07-APR-83 17:03 TEST 24: DATA WORD FRAMING TEST

```

8501 036146' 103022          BCC      20$          :OK
8502 036150' 012737 001000' 000310'      MOV      #SDNI,BITNAM
8503 036156' 012737 001277' 000312'      MOV      #SNSET,BITSTA
8504 036164' 012737 001342' 000314'      MOV      #AFTER,PWHEN
8505 036172' 012737 001357' 000316'      MOV      #SGICMD,PCOMND
8506 036200'          ERRHRD  120,DWFRAM,MSG1
8507 036200' 104456
8508 036202' 000170          TRAP     CSERHRD
8509 036204' 002567'          .WORD   120
8510 036206' 012716'          .WORD   DWFRAM
8511 036210'          .WORD   MSG1
8512 036210' 104410          ESCAPE  TST
8513 036212' 000236          TRAP     CSERHRD
8514 036214' 004737 017362'          .WORD   L10142-
8515 036220' 103006          20$:    JSR      PC,CLRDNI          ;CLEAR DNI BIT
8516 036222'          BCC      25$
8517 036222' 104456          ERRHRD  121,DWFRAM,RACMG7      ;DNI DID NOT CLEAR!
8518 036224' 000171          TRAP     CSERHRD
8519 036226' 002567'          .WORD   121
8520 036230' 012670'          .WORD   DWFRAM
8521 036232'          .WORD   RACMG7
8522 036232' 104410          ESCAPE  TST
8523 036234' 000214          TRAP     CSERHRD
8524 036236'          .WORD   L10142-
8525 036236'          25$:
8526 036236'          ENDSEG
8527 036236' 104405          10000$: TRAP     CSERHRD
8528
8529          ;
8530          ;WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE. CLEAR THE PCBB. LOAD THE
8531          ;COMMAND FIELD OF PCSRO WITH A 4 TO START THE EXECUTION OF MICROTEST #4.
8532          ;WAIT FOR THE 'DNI' BIT. IF NO 'DNI' OR IF ANY EXTRANEIOUS INTERRUPTS HAPPEN
8533          ;PRINT ERROR
8534          ;
8535          ;
8536          ;
8537          ;
8538          ;
8539          ;
8540          ;
8541          ;
8542          ;
8543          ;
8544          ;
8545          ;
8546          ;
8547          ;
8548          ;
8549          ;
8550          ;
8551          ;
8552          ;
8553          ;
8554          ;
8555          ;
8556          ;
    
```

;WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE. CLEAR THE PCBB. LOAD THE  
 ;COMMAND FIELD OF PCSRO WITH A 4 TO START THE EXECUTION OF MICROTEST #4.  
 ;WAIT FOR THE 'DNI' BIT. IF NO 'DNI' OR IF ANY EXTRANEIOUS INTERRUPTS HAPPEN  
 ;PRINT ERROR

;THIS IS WHERE MICROCODE WILL PUT...  
 ;RECEIVE BUFFER STATUS WORD  
 ;HERE IS WHERE THE GOOD DATA GOES  
 ;HERE IS WHERE THE BAD DATA GOES  
 ;HERE IS WHERE THE BUFFER OFFSET GOES  
 ;TELL T11 TO EXECUTE MICROTEST #4  
 ;WAIT FOR DNI

;SEE IF ANY ERROR INTERRUPTS OCCURRED  
 ;NO, OK



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 182  
 CZUAAB.MAC 07-APR-83 17:03 TEST 24: DATA WORD FRAMING TEST

```

8557 036334' ERRHRD 123,DWFRAM,MSG44 ;PRINT ERROR MESSAGE
8558 036334' 104436 TRAP CSERHRD
8559 036336' 000173 .WORD 123
8560 036340' 002567' .WORD DWFRAM
8561 036342' 016442' .WORD MSG44
8562 036344' ESCAPE TST
8563 036344' 104410 TRAP CSERHRD
8564 036346' 000102 .WORD L10142-.
8565 036350' 012702 000004 35$: MOV #4,R2 ;MICROTEST #4 IS HUNG
8566 036354' ERRHRD 124,DWFRAM,MSG12
8567 036354' 104456 TRAP CSERHRD
8568 036356' 000174 .WORD 124
8569 036360' 002567' .WORD DWFRAM
8570 036362' 013466' .WORD MSG12
8571 036364' ESCAPE TST
8572 036364' 104410 TRAP CSERHRD
8573 036366' 000062 .WORD L10142-.
8574
8575 ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF THE T-11 DETECTED A BOGUS WORD
8576 ;IN THE RECEIVER BUFFER. IF SO, PRINT ERROR.
8577
8578 036370' 122777 000003 141742 40$: CMPB #INERR,PCSR1 ;DID AN ERROR OCCUR?
8579 036376' 001014 BNE 50$ ;NO
8580 036400' 013701 000614' MOV PCBB+6,R1 ;GET BUFFER OFFSET FOR ERROR REPORT
8581 036404' 013702 000610' MOV PCBB+2,R2 ;THIS IS THE GOOD DATA FOR ERROR REPORT
8582 036410' 013703 000612' MOV PCBB+4,R3 ;THIS IS THE BAD DATA FOR ERROR REPORT
8583 036414' 013704 000606' MOV PCBB,R4 ;THIS IS THE RECEIVER STATUS WORD
8584 036420' ERRHRD 125,DWFRAM,MSG17 ;PRINT ERROR MESSAGE
8585 036420' 104456 TRAP CSERHRD
8586 036422' 000175 .WORD 125
8587 036424' 002567' .WORD DWFRAM
8588 036426' 013730' .WORD MSG17
8589
8590 ;WRITE '1' TO CLEAR 'DNI' BIT
8591
8592 036430' 004737 017362' 50$: JSR PC,CLRDN1 ;CLEAR DNI BIT
8593 036434' 103004 BCC 55$
8594 036436' ERRHRD 126,DWFRAM,RACMG7 ;ERROR DNI DID NOT CLEAR!
8595 036436' 104456 TRAP CSERHRD
8596 036440' 000176 .WORD 126
8597 036442' 002567' .WORD DWFRAM
8598 036444' 012670' .WORD RACMG7
8599 036446' 55$:
8600 036446' ENDSEG
8601 036446' 10001$:
8602 036446' 104405 TRAP CSESEG
8603 036450' ENDTST
8604 036450' L10142:
8605 036450' 104401 TRAP CSETST
    
```

8606  
8607  
8608  
8609  
8610  
8611  
8612  
8613  
8614  
8615  
8616  
8617  
8618  
8619  
8620  
8621  
8622  
8623  
8624  
8625  
8626  
8627  
8628  
8629  
8630  
8631  
8632  
8633 036452'  
8634 036452'  
8635  
8636  
8637  
8638  
8639  
8640 036452'  
8641 036452' 104404  
8642 036454' 022737 000104 000326'  
8643 036462' 001004  
8644 036464' 122777 000001 141646  
8645 036472' 001440  
8646 036474' 012737 000104 000326' 58:  
8647 036502' 004737 020340'  
8648 036506' 103002  
8649 036510'  
8650 036510' 104410  
8651 036512' 000332  
8652 036514' 012737 000176 000332' 108:  
8653 036522' 004737 017316'  
8654 036526' 103022  
8655 036530' 012737 001000' 000310'  
8656 036536' 012737 001277' 000312'  
8657 036544' 012737 001342' 000314'  
8658 036552' 012737 001357' 000316'  
8659 036560'  
8660 036560' 104456  
8661 036562' 000177

.SBTTL TEST 25: DATA PATH PATTERN TEST

:\*\*\*\*\*  
: THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR ALL 'STUCK AT 0' AND  
: 'STUCK AT 1' ERRORS.  
: THE T-11 PROCESSOR WILL TRANSMIT DATAGRAMS OF MAXIMUM LENGTH IN LOOPBACK MODE.  
: THIS PATTERN LOOPBACK PROCEDURE WILL BE USED FOR ALL PATTERNS OF UP TO WORD  
: WIDTH.  
: THIS TEST USES MICROMODULE 'D' MICROTEST #5 TO DO THE TESTING. THE HOST  
: PROCESSOR WILL PASS A DATA PATTERN TO THE T-11 PROCESSOR THROUGH THE PCBB.  
: THE T-11 WILL FILL A XMIT BUFFER WITH THE DATA PATTERN AND TRANSMIT THE  
: DATAGRAM OVER THE LOOPBACK. THE T-11 PROCESSOR WILL VERIFY THE PATTERN  
: IN THE RECEIVER BUFFER. IF THE T-11 FINDS AN ERROR, IT WILL WRITE THE FAILING  
: PATTERN TO THE PCBB ALONG WITH THE OFFSE: INTO THE RECEIVER BUFFER AT WHICH  
: THE PATTERN WAS FOUND. IT WILL INFORM THE HOST OF THE ERROR BY SETTING PCSR1  
: TO AN ERROR CONDITION. THE PCBB IS FORMATTED AS FOLLOWS:  
: PCBB+0: DATA PATTERN  
: PCBB+2: RECEIVER STATUS WORD  
: PCBB+4: BAD DATA PATTERN  
: PCBB+6: OFFSET INTO RECEIVER BUFFER WHERE BAD DATA WAS FOUND  
:\*\*\*\*\*

BGNTST

T25::

: CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
: THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'  
: AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

BGNSEG

							TRAP	C\$BSEG
8642	036454'	022737	000104	000326'	CMP	#'D,MICRO	:	HAS MICROCODE MODULE 'D' BEEN LOADED
					BNE	58	:	NO
8644	036464'	122777	000001	141646	CMPEB	#INMON,BPCSR1	:	YES, IS THE MICROMONITOR ACTIVE?
8645	036472'	001440			BEQ	208	:	YES SKIP LOADING THE MICROMODULE
8646	036474'	012737	000104	000326'	MOV	#'D,MICRO	:	GO LOAD MICRO MODULE 'D'
8647	036502'	004737	020340'		JSR	PC,LODMIC		
8648	036506'	103002			BCC	108	:	OK
8649	036510'				ESCAPE	TST		
8651	036512'	000332					TRAP	C\$ESCAPE
8652	036514'	012737	000176	000332'	MOV	#2*SECOND,METER	.WORD	L10143 .
8653	036522'	004737	017316'		JSR	PC,CHKDNI	:	WAIT FOR THE MICROMONITOR
8654	036526'	103022			BCC	208	:	OK
8655	036530'	012737	001000'	000310'	MOV	#SDNI,BITNAM		
8656	036536'	012737	001277'	000312'	MOV	#SNSET,BITSTA		
8657	036544'	012737	001342'	000314'	MOV	#SAFTER,PWHEN		
8658	036552'	012737	001357'	000316'	MOV	#SGTCMD,PCOMND		
8659	036560'				ERRHRD	127,DPPAT,MSG1		
8660	036560'	104456					TRAP	C\$ERHRD
8661	036562'	000177					.WORD	127

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 184  
 CZUAAB.MAC 07-APR-83 17:03 TEST 25: DATA PATH PATTERN TEST

```

8662 036564' 002625' .WORD DPPAT
8663 036566' 012716' .WORD MSG1
8664 036570'          ESCAPE TST
8665 036570' 104410          TRAP C$ESCAPE
8666 036572' 000252          .WORD L10143-
8667 036574' 004737 017362' 20$: JSR    PC,CLRDMI      ;CLEAR DMI BIT
8668 036600' 103006          BCC    25$
8669 036602'          ERRHRD 128,DPPAT,RACMG7 ;DMI DID NOT CLEAR!
8670 036602' 104456          TRAP C$ERHRD
8671 036604' 000200          .WORD 128
8672 036606' 002625'          .WORD DPPAT
8673 036610' 012670'          .WORD RACMG7
8674 036612'          ESCAPE TST
8675 036612' 104410          TRAP C$ESCAPE
8676 036614' 000230          .WORD L10143-
8677 036616'          25$:
8678 036616'          ENDSEG
8679 036616'          10000$:
8680 036616' 104405          TRAP C$ESEG
8681
8682 ;CLEAR OUT THE PCBB+2,+4,+6 AND LOAD PCBB+0 WITH A DATA PATTERN SELECTED
8683 ;FROM A LIST OF STANDARD DATA PATTERNS
8684
8685 036620' 005037 000610' ;
8686          CLR    PCBB+2          ;HERE IS WHERE MICROCODE WILL PUT...
8687          CLR    PCBB+4          ;STATUS WORD
8688 036630' 005037 000614' ;HERE IS WHERE BAD DATA GOES
8689 036634' 012701 000520' ;HERE IS WHERE BUFFER OFFSET GOES
8690 036640' 012705 000005' ;GET ADDRESS OF DATA PATTERN TABLE
8691 036644' 012137 000606' ;# OF DATA PATTERNS
8692 036650'          27$: MOV    (R1)+,PCBB          ;LOAD PCBB WITH A DATA PATTERN
8693 036650' 104404          ;
8694          BGNSEG
8695          TRAP C$BSEG
8696 ;WAIT FOR THE T-11 TO ENTER THE 'INMON' STATE AT WHICH TIME SELECT MICROTEST #5
8697 ;BY LOADING PCSRO COMMAND FIELD BITS WITH A 5. WAIT FOR 'DMI'.
8698 036652' 004737 020060' ;
8699 036656' 103006          JSR    PC,CHKMON          ;WAIT FOR MICROMONITOR
8700 036660'          BCC    30$
8701 036660' 104456          ERRHRD 129,DPPAT,MSG46 ;OK
8702 036662' 000201          ;PRINT ERROR
8703 036664' 002625'          TRAP C$ERHRD
8704 036666' 016666'          .WORD 129
8705 036670'          ESCAPE TST          ;LEAVE TEST
8706 036670' 104410          .WORD DPPAT
8707 036672' 000152          .WORD MSG46
8708 036674' 012777 000005 141434 30$: MOV    #5,@PCSRO
8709 036702' 012737 000176 000332' MOV    #2*SECOND,METER ;TELL T11 TO EXECUTE MICROTEST #5
8710 036710' 004737 017316' JSR    PC,CHKDMI          ;WAIT FOR DMI
8711 036714' 103021          BCC    40$
8712 036716' 004737 020132' JSR    PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
8713 036722' 103006          BCC    35$
8714 036724'          ERRHRD 130,DPPAT,MSG44 ;NO, OK
8715 036724' 104456          ;PRINT ERROR MESSAGE
8716 036726' 000202          TRAP C$ERHRD
8717 036730' 002625'          .WORD 130
          .WORD DPPAT
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 185  
 CZUAAB.MAC 07-APR-83 17:03 TEST 25: DATA PATH PATTERN TEST

```

8718 036732' 016442' .WORD MSG44
8719 036734' ESCAPE TST
8720 036734' 104410 TRAP C$ESCAPE
8721 036736' 000106 .WORD L10143-
8722 036740' 012702 000005 358: MOV #5,R2 ;MICROTEST #5 IS HUNG
8723 036744' ERRHRD 131,DPPAT,MSG12
8724 036744' 104456 TRAP C$ERHRD
8725 036746' 000203 .WORD 131
8726 036750' 002625' .WORD DPPAT
8727 036752' 013466' .WORD MSG12
8728 036754' ESCAPE TST
8729 036754' 104410 TRAP C$ESCAPE
8730 036756' 000066 .WORD L10143-
8731
8732 ;OK WE GOT 'DNI' NOW CHECK PC$R1 TO SEE IF AN ERROR HAPPENED, IF SO, PRINT
8733 ;PERTINENT INFORMATION
8734
8735 036760' 122777 000003 141352 408: CMPB #INERR,@PC$R1 ;DID AN ERROR OCCUR?
8736 036766' 001014 BNE 508 ;NO
8737 036770' 013701 000614' MOV PCBB+6,R1 ;GET BUFFER OFFSET FOR ERROR REPORT
8738 036774' 013702 000606' MOV PCBB,R2 ;THIS IS THE GOOD DATA FOR ERROR REPORT
8739 037000' 013703 000612' MOV PCBB+4,R3 ;THIS IS THE BAD DATA FOR ERROR REPORT
8740 037004' 013704 000610' MOV PCBB+2,R4 ;THIS IS THE RECEIVER STATUS WORD
8741 037010' ERRHRD 132,DPPAT,MSG17 ;PRINT ERROR MESSAGE
8742 037010' 104456 TRAP C$ERHRD
8743 037012' 000204 .WORD 132
8744 037014' 002625' .WORD DPPAT
8745 037016' 013730' .WORD MSG17
8746
8747 ;WRITE '1' TO CLEAR 'DNI' BIT
8748
8749 037020' 004737 017362' 508: JSR PC,CLRDN1 ;CLEAR DNI BIT
8750 037024' 103004 BCC 558
8751 037026' ERRHRD 133,DPPAT,RACMG7 ;ERROR DNI DID NOT CLEAR!
8752 037026' 104456 TRAP C$ERHRD
8753 037030' 000205 .WORD 133
8754 037032' 002625' .WORD DPPAT
8755 037034' 012670' .WORD RACMG7
8756 037036'
8757 037036' 558: ENDSEG
8758 037036'
8759 037036' 104405 100018: TRAP C$ESEG
8760 037040' 005305 .WORD 278 ;HAVE WE TESTED WITH ALL DATA PATTERNS?
8761 037042' 001300 DEC R5 ;NOT YET
8762 037044' ENDTST
8763 037044'
8764 037044' 104401 L10143: TRAP C$ETST

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 186  
CZUAAB.MAC 07-APR-83 17:03 TEST 26: STATUS MUX VERIFICATION TEST

8765  
8766  
8767  
8768  
8769  
8770  
8771  
8772  
8773  
8774  
8775  
8776  
8777  
8778  
8779  
8780  
8781  
8782  
8783  
8784  
8785  
8786  
8787  
8788  
8789  
8790  
8791  
8792  
8793  
8794  
8795  
8796  
8797  
8798  
8799  
8800  
8801  
8802  
8803  
8804  
8805  
8806  
8807  
8808  
8809  
8810  
8811  
8812  
8813  
8814  
8815  
8816  
8817  
8818  
8819  
8820

037046'  
037046'  
  
  
  
  
  
  
  
  
037046'  
037046' 104404  
037050' 022737 000104 000326'  
037056' 001004

.SBTTL TEST 26: STATUS MUX VERIFICATION TEST  
:\*\*\*\*\*  
:THE LINK WRITES STATUS IN LINK MEMORY AFTER EACH TRANSMIT ATTEMPT. THE STATUS  
:GIVES INFORMATION ABOUT THE ATTEMPTED OPERATION. THE STATUS INFORMATION IS  
:WRITTEN INTO THE FIRST TWO LOCATIONS OF THE TRANSMIT BUFFER. THIS INFORMATION  
:IS ACCESSIBLE TO THE T-11 BY SIMPLY READING IT FROM LINK MEMORY.  
:THIS TEST WILL VERIFY THAT THE STATUS INFORMATION IS WRITTEN INTO THE FIRST  
:LOCATION OF THE TRANSMIT BUFFER. THE TEST WILL ALSO CHECK THE SECOND WORD  
:OF THE TRANSMIT BUFFER.  
:THIS TEST WILL USE MICROMODULE 'D' MICROTEST #6.  
:WHEN THE TEST IS STARTED, THE T-11 PROCESSOR WILL SET UP THE LINK FOR LOOPBACK  
:OF A DATA PATTERN. A BACKGROUND PATTERN WILL BE WRITTEN INTO THE FIRST WORD  
:OF THE TRANSMIT BUFFER. THIS WORD SHOULD BE OVER-WRITTEN BY THE STATUS WHEN  
:THE BUFFER IS TRANSMITTED. THE SECOND WORD OF THE TRANSMIT BUFFER CAN NOT BE  
:WRITTEN WITH A BACKGROUND BECAUSE IT MUST DESIGNATE THE TRANSMIT BYTE COUNT.  
:WHEN THE DATAGRAM HAS BEEN LOOPED BACK, THE T-11 PROCESSOR WILL PASS THE FIRST  
:TWO WORDS OF THE TRANSMIT BUFFER TO THE HOST THRU THE PCBB+0 AND PCBB+2.  
:PCBB+0: FIRST WORD OF TRANSMIT BUFFER  
:PCBB+2: SECOND WORD OF TRANSMIT BUFFER  
:THE CORRECT STATUS SHOULD BE:  
:TRANSMIT STATUS WORD 0 BITS 15,09:00 SHOULD BE ALL 0 AND  
:BIT 13 SHOULD BE A 1  
:TRANSMIT STATUS WORD 1 BITS 15:13 SHOULD ALL BE 0  
:TEST SEQUENCE:  
:1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO  
:2-VERIFY MICROMONITOR IS IN THE 'INMON' STATE  
:3-CLEAR PCBB+0 AND PCBB+2  
:4-SELECT MICROTEST #6  
:5-VERIFY 'DNI' SET  
:6-VERIFY PCBB+0 BITS 15,09:00 = 0 AND BIT 13 = 1  
:AND PCBB+2 BITS 15:13 = 0  
:7-WRITE ONE TO CLEAR 'DNI'  
:\*\*\*\*\*  
BGNTST  
T26::  
:CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS. START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:BGNSEG  
CMP #D.MICRO ;HAS MICROCODE MODULE 'D' BEEN LOADED  
BNE 55 ;NO  
TRAP CSBSEG

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 187  
 CZUAB.MAC 07-APR-83 17:03 TEST 26: STATUS MUX VERIFICATION TEST

```

8821 037060' 122777 000001 141252          CMPB  #INMON, @PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
8822 037066' 001440          BEQ   20$                    ;YES SKIP LOADING THE MICROMODULE
8823 037070' 012737 000104 000326' 5$:      MOV   #'D, MICRO           ;GO LOAD MICRO MODULE 'D'
8824 037076' 004737 020340'          JSR   PC, LODMIC
8825 037102' 103002          BCC   10$                    ;OK
8826 037104'          ESCAPE TST
8827 037104' 104410          TRAP .WORD C$ESCAPE
8828 037106' 000500          .WORD L10144-.
8829 037110' 012737 000176 000332' 10$:     MOV   #2*SECOND, METER     ;WAIT FOR THE MICROMONITOR
8830 037116' 004737 017316'          JSR   PC, CHKDNI
8831 037122' 103022          BCC   20$                    ;OK
8832 037124' 012737 001000' 000310'        MOV   #SDNI, BITNAM
8833 037132' 012737 001277' 000312'        MOV   #SNSET, BITSTA
8834 037140' 012737 001342' 000314'        MOV   #SAFTER, PWHEN
8835 037146' 012737 001357' 000316'        MOV   #SGTCMD, PCOMND
8836 037154'          ERRHRD 134, STAMUX, MSG1
8837 037154' 104456          TRAP .WORD C$ERHRD
8838 037156' 000206          .WORD 134
8839 037160' 002663'          .WORD STAMUX
8840 037162' 012716'          .WORD MSG1
8841 037164'          ESCAPE TST
8842 037164' 104410          TRAP .WORD C$ESCAPE
8843 037166' 000420          .WORD L10144-.
8844 037170' 004737 017362' 20$:          JSR   PC, CLRDNI          ;CLEAR DNI BIT
8845 037174' 103006          BCC   25$
8846 037176'          ERRHRD 135, STAMUX, RACMG7 ;DNI DID NOT CLEAR!
8847 037176' 104456          TRAP .WORD C$ERHRD
8848 037200' 000207          .WORD 135
8849 037202' 002663'          .WORD STAMUX
8850 037204' 012670'          .WORD RACMG7
8851 037206'          ESCAPE TST
8852 037206' 104410          TRAP .WORD C$ESCAPE
8853 037210' 000376          .WORD L10144-.
8854 037212'          25$:
8855 037212'          ENDSEG
8856 037212'          10000$:
8857 037212' 104405          TRAP .WORD C$ESEG
8858          ;
8859          ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, THEN CLEAR OUT THE
8860          ;FIRST TWO WORDS OF THE PCBB. EXECUTE MICROTEST #6 BY LOADING THE COMMAND
8861          ;FIELD BITS OF PCSRO WITH A 6, WAIT FOR 'DNI'.
8862          ;
8863          BGNSEG
8864 037214' 104404          TRAP .WORD C$BSEG
8865 037216' 004737 020060'          JSR   PC, CHKMON          ;WAIT FOR MICROMONITOR
8866 037222' 103006          BCC   30$                    ;OK
8867 037224'          ERRHRD 136, STAMUX, MSG46 ;PRINT ERROR
8868 037224' 104456          TRAP .WORD C$ERHRD
8869 037226' 000210          .WORD 136
8870 037230' 002663'          .WORD STAMUX
8871 037232' 016666'          .WORD MSG46
8872 037234'          ESCAPE TST          ;LEAVE TEST
8873 037234' 104410          TRAP .WORD C$ESCAPE
8874 037236' 000350          .WORD L10144-.
8875 037240' 005037 000606' 30$:          CLR   PCBB                ;THIS IS WHERE THE MICROCODE WILL...
8876          ;PUT THE CONTENTS OF THE FIRST WORD
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 188  
 CZUAAB.MAC 07-APR-83 17:03 TEST 26: STATUS MUX VERIFICATION TEST

```

8877
8878 037244' 005037 000610' CLR PCBB+2 ;OF THE TRANSMIT BUFFER
8879 037250' 012777 000006 141060 MOV #6,BPCSR0 ;HERE IS WHERE THE SECOND WORD GOES
8880 037256' 012737 000176 0G0332' MOV #2*SECOND,METER ;TELL T11 TO EXECUTE MICROTEST #6
8881 037264' 004737 017316' JSR PC,LHKDNI ;WAIT FOR DNI
8882 037270' 103021 BCC 45$
8883 037272' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
8884 037276' 103006 BCC 35$ ;NO, OK
8885 037300' ERRHRD 137,STAMUX,MSG44 ;PRINT ERROR MESSAGE
8886 037300' 104456 TRAP CSERHRD
8887 037302' 000211 .WORD 137
8888 037304' 002663' .WORD STAMUX
8889 037306' 016442' .WORD MSG44
8890 037310' ESCAPE TST
8891 037310' 104410 TRAP C$ESCAPE
8892 037312' 000274 .WORD L10144-
8893 037314' 012702 000006 35$: MOV #6,R2 ;MICROTEST #6 IS HUNG
8894 037320' ERRHRD 138,STAMUX,MSG12
8895 037320' 104456 TRAP CSERHRD
8896 037322' 000212 .WORD 138
8897 037324' 002663' .WORD STAMUX
8898 037326' 013466' .WORD MSG12
8899 037330' ESCAPE TST
8900 037330' 104410 TRAP C$ESCAPE
8901 037332' 000254 .WORD L10144-
8902
8903 ;
8904 ;OK, NOW PCBB+0 SHOULD CONTAIN THE TRANSMIT STATUS WORD 0 AND PCBB+2 SHOULD
8905 ;CONTAIN TRANSMIT STATUS WORD 1. CHECK THAT PCBB+0 CONTAINS STATUS BITS
8906 ;FOR A GOOD TRANSMIT I.E. BITS 09:00 SHOULD ALL BE 0 BIT 13 SHOULD BE A 1
8907 ;AND BIT 15 SHOULD BE A 0
8908 037334' 005001 45$: CLR R1 ;WE ARE GOING TO CHECK TRANSMIT WORD 0
8909 037336' 012703 000460' MOV #BNAMT2,R3 ;POINT TO A TABLE OF BIT MNEMONICS
8910 037342' 012704 000012 MOV #10,R4 ;FIRST 10 BITS SHOULD BE 0
8911 037346' 012702 000001 MOV #1,R2 ;R2 POINTS TO THE BIT WE ARE TESTING
8912 037352' 030237 000606' 50$: BIT R2,PCBB ;IS THIS BIT A 0?
8913 037356' 001406 BEQ 55$ ;YES
8914 037360' 011337 000310' MOV (R3),BITNAM ;NO, GET POINTER TO BIT NAME ASCII STRING
8915 037364' ERRHRD 139,STAMUX,MSG18 ;PRINT ERROR MESSAGE
8916 037364' 104456 TRAP CSERHRD
8917 037366' 000213 .WORD 139
8918 037370' 002663' .WORD STAMUX
8919 037372' 014022' .WORD MSG18
8920 037374' 062703 000002 55$: ADD #2,R3 ;POINT TO NEXT BIT MNEMONIC
8921 037400' 006302 ASL R2 ;POINT TO NEXT BIT
8922 037402' 005304 DEC R4 ;HAVE WE DONE ALL 10 BITS?
8923 037404' 001362 BNE 50$ ;NO
8924
8925 037406' 032737 020000 000606' BIT #BIT13,PCBB ;IS BIT 13 A 1?
8926 037414' 001007 BNE 60$ ;YES
8927 037416' 012737 001147' 000310' MOV #S$BIT13,BITNAM ;NO, POINT TO ASCII STRING
8928 037424' ERRHRD 140,STAMUX,MSG18 ;PRINT ERROR MESSAGE
8929 037424' 104456 TRAP CSERHRD
8930 037426' 000214 .WORD 140
8931 037430' 002663' .WORD STAMUX
8932 037432' 014022' .WORD MSG18

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 189  
 CZUAAB.MAC 07-APR-83 17:03 TEST 26: STATUS MUX VERIFICATION TEST

Srw

```

8933
8934 037434' 032737 1G0000 000606' 60$: BIT #BIT15,PCBB :IS BIT 15 A 0?
8935 037442' 001407 BEQ 65$ :YES
8936 037444' 012737 001131' 000310' MOV #BIT15,BITNAM :NO, POINT TO ASCII STRING
8937 037452' ERRHRD 141,STAMUX,MSG18 :PRINT ERROR MESSAGE
8938 037452' 74456 TRAP CSERHRD
8939 037454' 000215 .WORD 141
8940 037456' 002663' .WORD STAMUX
8941 037460' 014022' .WORD MSG18
8942
8943 ;NOW CHECK TRANSMIT WORD 1 BITS 15,14 AND 13 TO BE ALL 0
8944
8945 037462' 005201 65$: INC R1 :WE ARE CHECKING TRNASMIT WORD 1 NOW
8946 037464' 032737 020000 000610' BIT #BIT13,PCBB+2 :IS BIT 13 A 0?
8947 037472' 001407 BEQ 70$ :YES
8948 037474' 012737 001147' 000310' MOV #BIT13,BITNAM :NO, POINT TO ASCII STRING
8949 037502' ERRHRD 142,STAMUX,MSG18 :PRINT ERROR MESSAGE
8950 037502' 104456 TRAP CSERHRD
8951 037504' 000216 .WORD 142
8952 037506' 002663' .WORD STAMUX
8953 037510' 014022' .WORD MSG18
8954
8955 037512' 032737 040000 000610' 70$: BIT #BIT14,PCBB+2 :IS BIT 14 A 0?
8956 037520' 001407 BEQ 75$ :YES
8957 037522' 012737 001140' 000310' MOV #BIT14,BITNAM :NO, POINT TO ASCII STRING
8958 037530' ERRHRD 143,STAMUX,MSG18 :PRINT ERROR MESSAGE
8959 037530' 104456 TRAP CSERHRD
8960 037532' 000217 .WORD 143
8961 037534' 002663' .WORD STAMUX
8962 037536' 014022' .WORD MSG18
8963
8964 037540' 032737 100000 000610' 75$: BIT #BIT15,PCBB+2 :IS BIT 15 A 0?
8965 037546' 001407 BEQ 80$ :YES
8966 037550' 012737 001131' 000310' MOV #BIT15,BITNAM :NO, POINT TO ASCII STRING
8967 037556' ERRHRD 144,STAMUX,MSG18 :PRINT ERROR MESSAGE
8968 037556' 104456 TRAP CSERHRD
8969 037560' 000220 .WORD 144
8970 037562' 002663' .WORD STAMUX
8971 037564' 014022' .WORD MSG18
8972
8973 ;WRITE '1' TO CLEAR 'DNI' BIT
8974
8975 037566' 004737 017362' 80$: JSR PC,CLRDN1 :CLEAR DNI BIT
8976 037572' 103004 BCC 85$
8977 037574' ERRHRD 145,STAMUX,RACMG7 :ERROR DNI DID NOT CLEAR!
8978 037574' 104456 TRAP CSERHRD
8979 037576' 000221 .WORD 145
8980 037600' 002663' .WORD STAMUX
8981 037602' 012670' .WORD RACMG7
8982 037604' 85$:
8983 037604' ENDSEG
8984 037604' 100018: TRAP CSESEG
8985 037604' 104405 .WORD
8986 037606' ENDTST
8987 037606' L10144: TRAP
8988 037606' 104401 .WORD CSETST

```





65HARDWARE TESTS MACY11 37A(1052) 07-APR-83 17:13 PAGE 191  
 CZUAAB.MAL 07-APR-83 17:03 TEST 27: LINK BYTE COUNTER TEST

```

9045 037610' 104404
9046 037612' 022737 000104 000326' CMP #D,MICRO ;HAS MICROCODE MODULE 'D' BEEN LOADED
9047 037620' 001004 BNE 5$ ;NO
9048 037622' 122777 000001 140510 CMPB #INMON,BPCSR1 ;YES, IS THE MICROMONITOR ACTIVE?
9049 037630' 001440 BEQ 20$ ;YES SKIP LOADING THE MICROMODULE
9050 037632' 012737 000104 000326' 5$: MOV #D,MICRO ;GO LOAD MICRO MODULE 'D'
9051 037640' 004737 020340' JSR PC,LODMIC
9052 037644' 103002 BCC 10$ ;OK
9053 037646' ESCAPE TST
9054 037646' 104410
9055 037650' 000334 TRAP C$BSEG
9056 037652' 012737 000176 000332' 10$: MOV #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
9057 037660' 004737 017316' JSR PC,CHKDNI
9058 037664' 103022 BCC 20$ ;OK
9059 037666' 012737 001000' 000310' MOV #SDNI,BITNAM
9060 037674' 012737 001277' 000312' MOV #SNSET,BITSTA
9061 037702' 012737 001342' 000314' MOV #AFTER,PWHEN
9062 037710' 012737 001357' 000316' MOV #SGTCMD,PCOMND
9063 037716' ERRHRD 146,LNKBYT,MSG1
9064 037716' 104456 TRAP C$ERHRD
9065 037720' 000222 .WORD 146
9066 037722' 002727' .WORD LNKBYT
9067 037724' 012716' .WORD MSG1
9068 037726' ESCAPE TST
9069 037726' 104410 TRAP C$ESCAPE
9070 037730' 000254 .WORD L10145-
9071 037732' 004737 017362' 20$: JSR PC,CLRDMI ;CLEAR DMI BIT
9072 037736' 103006 BCC 25$
9073 037740' ERRHRD 147,LNKBYT,RACMG7 ;DMI DID NOT CLEAR!
9074 037740' 104456 TRAP C$ERHRD
9075 037742' 000223 .WORD 147
9076 037744' 002727' .WORD LNKBYT
9077 037746' 012670' .WORD RACMG7
9078 037750' ESCAPE TST
9079 037750' 104410 TRAP C$ESCAPE
9080 037752' 000232 .WORD L10145-
9081 037754' 25$:
9082 037754' ENDSEG
9083 037754' 10000$:
9084 037754' 104405 TRAP C$ESEG
9085 :
9086 :LOAD PCBB+0 WITH THE MINIMUM BYTE COUNT
9087 :
9088 037756' 012737 000100 000606' MOV #MINBYT,PCBB ;BEGIN WITH MINIMUM BYTE COUNT
9089 :
9090 :CLEAR PCBB+2 AND PCBB+4, AIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
9091 :EXECUTE MICROTST #7 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 7.
9092 :WAIT FOR 'DMI'
9093 :
9094 037764' 30$:
9095 037764' BGNSEG
9096 037764' 104404 TRAP C$BSEG
9097 037766' 005037 000610' CLR PCBB+2 ;THIS IS WHERE MICROCODE WILL PUT...
9098 :RECEIVER BYTE COUNT
9099 037772' 005037 000612' CLR PCBB+4 ;HERE IS WHERE MICROCODE WILL PUT...
9100 :ACTUAL NUMBER OF BYTES RECEIVED

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 192  
 CZUAAB.MAC 07-APR-83 17:03 TEST 27: LINK BYTE COUNTER TEST

```

9101 037776' 004737 020060'      JSR      PC,CHKMON      ;WAIT FOR MICROMONITOR
9102 040002' 103006      BCC      35$           ;OK
9103 040004'           ERRHRD  148,LNKBYT,MSG46 ;PRINT ERROR
9104 040004' 104456           TRAP     CSERHRD
9105 040006' 000224           .WORD   148
9106 040010' 002727'           .WORD   LNKBYT
9107 040012' 016666'           .WORD   MSG46
9108 040014'           ESCAPE  TST           ;LEAVE TEST
9109 040014' 104410           TRAP     CSERHRD
9110 040016' 000166           .WORD   L10145-.
9111 040020' 012777 000007 140310 35$:  MOV      #7,@PCSR0      ;TELL T11 TO EXECUTE MICROTEST #7
9112 040026' 012737 000176 000332'  MOV      #2*SECOND,METER ;WAIT FOR DNI
9113 040034' 004737 017316'  JSR      PC,CHKDNI
9114 040040' 103021      BCC      40$
9115 040042' 004737 020132'  JSR      PC,CHKINT      ;SEE IF ANY ERROR INTERRUPTS OCCURRED
9116 040046' 103006      BCC      36$           ;NO, OK
9117 040050'           ERRHRD  149,LNKBYT,MSG44 ;PRINT ERROR MESSAGE
9118 040050' 104456           TRAP     CSERHRD
9119 040052' 000225           .WORD   149
9120 040054' 002727'           .WORD   LNKBYT
9121 040056' 016442'           .WORD   MSG44
9122 040060'           ESCAPE  TST
9123 040060' 104410           TRAP     CSERHRD
9124 040062' 000122           .WORD   L10145-.
9125 040064' 012702 000007 36$:  MOV      #7,R2           ;MICROTEST #7 IS HUNG
9126 040070'           ERRHRD  150,LNKBYT,MSG12
9127 040070' 104456           TRAP     CSERHRD
9128 040072' 000226           .WORD   150
9129 040074' 002727'           .WORD   LNKBYT
9130 040076' 013466'           .WORD   MSG12
9131 040100'           ESCAPE  TST
9132 040100' 104410           TRAP     CSERHRD
9133 040102' 000102           .WORD   L10145-.
9134
9135 ;OK, CHECK PCBB+2, WHICH CONTAINS THE BYTE COUNT WRITTEN INTO THE RECEIVER
9136 ;BUFFER BY THE LINK, TO SEE IF IT IS THE SAME AS THE TRANSMIT BYTE COUNT. ALSO
9137 ;CHECK PCBB+4, WHICH IS THE ACTUAL NUMBER OF BYTES RECEIVED IN THE RECEIVER
9138 ;BUFFER, TO SEE IF IT IS THE SAME AS THE RECEIVE BYTE COUNT
9139 ;IF NOT PRINT AN ERROR
9140
9141 040104' 013701 000606' 40$:  MOV      PCBB,R1        ;GET TRANSMIT BYTE COUNT
9142 040110' 010102      MOV      R1,R2
9143 040112' 013703 000610'  MOV      PCBB+2,R3      ;GET RECEIVE BYTE COUNT
9144 040116' 013704 000612'  MOV      PCBB+4,R4      ;GET ACTUAL NUMBER OF BYTES TRANSMITTED
9145 040122' 020203      CMP      R2,R3          ;IS THE RECEIVE BYTE COUNT CORRECT?
9146 040124' 001002      BNE     45$           ;NO, ERROR
9147 040126' 020104      CMP      R1,R4          ;IS THE ACTUAL NUMBER OF BYTES CORRECT?
9148 040130' 001404      BEQ     50$           ;YES
9149 040132'           ERRHRD  151,LNKBYT,MSG19 ;PRINT ERROR MESSAGE
9150 040132' 104456           TRAP     CSERHRD
9151 040134' 000227           .WORD   151
9152 040136' 002727'           .WORD   LNKBYT
9153 040140' 014052'           .WORD   MSG19
9154 040142'
9155 040142' 50$:
9156 040142' ENDSEG

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 194  
CZUAAB.MAC 07-APR-83 17:03 TEST 28: ODD BYTE TEST

9181  
9182  
9183  
9184  
9185  
9186  
9187  
9188  
9189  
9190  
9191  
9192  
9193  
9194  
9195  
9196  
9197  
9198  
9199  
9200  
9201  
9202  
9203  
9204  
9205  
9206  
9207  
9208  
9209  
9210  
9211  
9212  
9213  
9214  
9215  
9216  
9217  
9218  
9219  
9220  
9221  
9222  
9223  
9224  
9225  
9226  
9227  
9228  
9229  
9230  
9231  
9232  
9233  
9234  
9235  
9236

040206'  
040206'  
  
  
  
  
  
040206'  
040206' 104404  
040210' 022737 000104 000326'  
040216' 001004  
040220' 122777 000001 140112  
040226' 001440  
040230' 012737 000104 000326' 5\$:  
040236' 004737 020340'  
040242' 103002  
040244'  
040244' 104410  
040246' 000340  
040250' 012737 000176 000332' 10\$:  
040256' 004737 017316'  
040262' 103022  
040264' 012737 001000' 000310'  
040272' 012737 001277' 000312'  
040300' 012737 001342' 000314'  
040306' 012737 001357' 000316'  
040314'  
040314' 104456  
040316' 000231  
040320' 002765'  
040322' 012716'

.SBTTL TEST 28: ODD BYTE TEST  
:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE LINK CAN TRANSMIT AND RECEIVE DATAGRAMS  
:HAVING ONLY ODD BYTE COUNTS.  
:THIS TEST IS IDENTICAL TO THE PREVIOUS BYTE COUNTER TEST WITH THE ONLY  
:EXCEPTION THAT IT PASSES ONLY ODD BYTE COUNTS TO THE MICROCODE. IT ALSO  
:USES MICROMODULE 'D' MICROTEST #7  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO  
: 2-WRITE MINIMUM ODD BYTE COUNT IN PCBB+0  
: 3-CLEAR PCBB+2 AND PCBB+4  
: 4-VERIFY MICROMONITOR IS IN 'INMON' STATE  
: 5-SELECT MICROTEST #7  
: 6-VERIFY TRANSMIT BYTE COUNT SAME AS RECEIVE BYTE COUNT  
: 7-VERIFY ACTUAL NUMBER OF BYTES RECEIVED IS CORRECT  
: 8-WRITE '1' TO CLEAR 'DNI'  
: 9-ADD 2 TO BYTE COUNT IN PCBB+0  
: 10-REPEAT STEPS 3-9 UNTIL PCBB+0 REACHES MAXIMUM BYTE COUNT  
:\*\*\*\*\*

BGNTST  
T28::  
:CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG  
TRAP CSBSEG  
;HAS MICROCODE MODULE 'D' BEEN LOADED  
;NO  
;YES, IS THE MICROMONITOR ACTIVE?  
;YES SKIP LOADING THE MICROMODULE  
;GO LOAD MICRO MODULE 'D'  
;OK  
TRAP C\$ESCAPE  
.WORD L10146-.  
;WAIT FOR THE MICROMONITOR  
;OK  
TRAP C\$ERHRD  
.WORD 153  
.WORD ODDBYT  
.WORD MSG1

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 195  
C7UAAB.MAC 07-APR-83 17:03 TEST 28: ODD BYTE TEST

```

9237 040324'          ESCAPE TST
9238 040324' 104410
9239 040326' 000260          TRAP  C$ESCAPE
9240 040330' 004737 017362' 208: JSR  PC,CLRDN1          :CLEAR DNI BIT
9241 040334' 103006          BCC  25$
9242 040336'          ERRHRD 154,ODDBYT,RACMG7 :DNI DID NOT CLEAR!
9243 040336' 104456          TRAP  C$ERHRD
9244 040340' 000232          .WORD 154
9245 040342' 002765'          .WORD ODDBYT
9246 040344' 012670'          .WORD RACMG7
9247 040346'          ESCAPE TST
9248 040346' 104410          TRAP  C$ESCAPE
9249 040350' 000236          .WORD L10146-.
9250 040352'
9251 040352'
9252 040352'
9253 040352' 104405          1000US: TRAP  C$ESEG
9254
9255 :LOAD PCBB+0 WITH THE MINIMUM ODD BYTE COUNT
9256 :
9257 040354' 012737 000101 000606' 308: MOV  #MINBYT+1,PCBB          :BEGIN WITH MINIMUM ODD BYTE COUNT
9258 040362'
9259 :
9260 :CLEAR PCBB+2 AND PCBB+4. WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON'
9261 :STATE. EXECUTE MICROTEST #7 BY LOADING THE COMMAND FIELD OF PCSRO WITH A
9262 :'7'. WAIT FOR 'DNI'
9263 :
9264 040362'          BGNSEG
9265 040362' 104404
9266 040364' 005037 000610'          CLR  PCBB+2          TRAP  C$BSEG
9267 :THIS IS WHERE MICROCODE WILL PUT...
9268 040370' 005037 000612'          CLR  PCBB+4          :RECEIVER BYTE COUNT
9269 :HERE IS WHERE MICROCODE WILL PUT...
9270 040374' 004737 020060'          JSR  PC,CHKMON          :ACTUAL NUMBER OF BYTES RECEIVED
9271 040400' 103006          BCC  35$          :WAIT FOR MICROMONITOR
9272 040402'          ERRHRD 156,ODDBYT,MSG46 :OK
9273 040402' 104456          :PRINT ERROR
9274 040404' 000234          TRAP  C$ERHRD
9275 040406' 002765'          .WORD 156
9276 040410' 016666'          .WORD ODDBYT
9277 040412'          .WORD MSG46
9278 040412' 104410          ESCAPE TST          :LEAVE TEST
9279 040414' 000172          TRAP  C$ESCAPE
9280 040416' 012777 000007 137712 358: MOV  #7,APCSRO          .WORD L10146-.
9281 040424' 012737 000176 000332' MOV  #2*SECOND,METER :TELL T11 TO EXECUTE MICROTEST #7
9282 040432' 004737 017316' JSR  PC,CHKDNI          :WAIT FOR DNI
9283 040436' 103021          BCC  40$
9284 040440' 004737 020132' JSR  PC,CHKINT          :SEE IF ANY ERROR INTERRUPTS OCCURRED
9285 040444' 103006          BCC  36$          :NO, OK
9286 040446'          ERRHRD 157,ODDBYT,MSG44 :PRINT ERROR MESSAGE
9287 040446' 104456          TRAP  C$ERHRD
9288 040450' 000235          .WORD 157
9289 040452' 002765'          .WORD ODDBYT
9290 040454' 016442'          .WORD MSG44
9291 040456'          ESCAPE TST
9292 040456' 104410          TRAP  C$ESCAPE

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 196  
 CZUAAB.MAC 07-APR-83 17:03 TEST 28: ODD BYTE TEST

```

9293 040460' 000126
9294 040462' 012702 000007      36$:  MOV    #7,R2                ;MICROTEST #7 IS MUNG      .WORD  L10146-.
9295 040466'
9296 040466' 104456
9297 040470' 000236
9298 040472' 002765'
9299 040474' 013466'
9300 040476'
9301 040476' 104410
9302 040500' 000106
9303
9304
9305
9306
9307
9308
9309
9310 040502' 013701 000606'      40$:  MOV    PCBB,R1                ;GET TRANSMIT BYTE COUNT
9311 040506' 010102
9312 040510' 013703 000610'
9313 040514' 013704 000612'
9314 040520' 020203
9315 040522' 001004
9316 040524' 162704 000001
9317 040530' 020104
9318 040532' 001404
9319 040534'
9320 040534' 104456
9321 040536' 000237
9322 040540' 002765'
9323 040542' 014052'
9324 040544'
9325 040544'
9326 040544'
9327 040544' 104405
9328
9329
9330
9331 040546' 004737 017362'
9332 040552' 103006
9333 040554'
9334 040554' 104456
9335 040556' 000240
9336 040560' 002765'
9337 040562' 012670'
9338 040564'
9339 040564' 104410
9340 040566' 000020
9341
9342
9343
9344
9345 040570' 062737 000002 000606'      60$:  ADD    #2,PCBB                ;UP THE BYTE COUNT
9346 040576' 023727 000606' 002756
9347 040604' 003666
9348 040606'
    ENDTST
    ;CHECK PCBB+2 TO SEE IF IT IS THE SAME AS PCBB+0, THIS WILL VERIFY THE
    ;THE TRANSMIT AND RECEIVE BYTE COUNTS ARE THE SAME. THEN CHECK PCBB+4
    ;TO SEE IF IT IS ONE LARGER THAN THE RECEIVE BYTE COUNT. THIS IS BECAUSE
    ;THE RECEIVE CAN ONLY ADDRESS LINK MEMORY IN WORD BOUNDARIES SO THE ACTUAL
    ;NUMBER OF BYTES THAT ARE CHANGED IS ONE MORE THAN WHAT WAS TRANSMITTED
    ;GET RECEIVE BYTE COUNT
    ;GET ACTUAL NUMBER OF BYTES TRANSMITTED
    ;IS THE RECEIVE BYTE COUNT CORRECT?
    ;NO, ERROR
    ;ACCOUNT FOR THE GARBAGE BYTE
    ;IS THE ACTUAL NUMBER OF BYTES CORRECT?
    ;YES
    ;PRINT ERROR MESSAGE
    ;WRITE '1' TO CLEAR 'DNI'
    ;CLEAR DNI BIT
    ;ERROR DNI DID NOT CLEAR!
    ;REPEAT MICROTEST #7 EACH TIME GIVING IT A LARGER ODD BYTE COUNT. CONTINUE
    ;UNTIL THE MAXIMUM BYTE COUNT IS REACHED
    ;UP THE BYTE COUNT
    ;HAVE WE DONE ALL ODD BYTE COUNTS?
    ;NO, CONTINUE TEST WITH NEW BYTE COUNT
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 197  
CZUAAB.MAC 07-APR-83 17:03 TEST 28: ODD BYTE TEST

9349 040606'  
9350 040606' 104401

L10146: TRAP CSETST



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 198  
CZUAAB.MAC 07-APR-83 17:03 TEST 29: LINK MAXIMUM BYTE COUNTER TEST

9351  
9352  
9353  
9354  
9355  
9356  
9357  
9358  
9359  
9360  
9361  
9362  
9363  
9364  
9365  
9366  
9367  
9368  
9369  
9370  
9371  
9372  
9373  
9374  
9375  
9376  
9377  
9378  
9379  
9380  
9381  
9382  
9383  
9384  
9385  
9386  
9387  
9388  
9389  
9390  
9391  
9392  
9393  
9394  
9395  
9396  
9397  
9398  
9399  
9400  
9401  
9402  
9403  
9404  
9405  
9406

040610'  
040610'  
  
  
  
  
  
040610'  
040610' 104404  
040612' 022737 000104 000326'  
040620' 001004  
040622' 122777 000001 137510  
040630' 001440  
040632' 012737 000104 000326' 5\$:  
040640' 004737 020340'  
040644' 103002  
040646'  
040646' 104410  
040650' 000256  
040652' 012737 000176 000332' 10\$:  
040660' 004737 017316'  
040664' 103022  
040666' 012737 001000' 000310'  
040674' 012737 001277' 000312'  
040702' 012737 001342' 000314'  
040710' 012737 001357' 000316'  
040716'  
040716' 104456

.SBTTL TEST 29: LINK MAXIMUM BYTE COUNTER TEST

\*\*\*\*\*  
:THE RECEIVE BYTE COUNTER IS A 12 BIT BINARY COUNTER THAT COUNTS THE NUMBER OF  
:BYTES THAT WERE RECEIVED DURING A DATAGRAM TRANSMISSION. THE BYTE COUNTER IS  
:INCREMENTED AS EACH BYTE IS RECEIVED. THE RECEIVE BYTE COUNTER HAS LOGIC THAT  
:DISABLES THE COUNTER IF THE MAXIMUM VALUE IS REACHED AND PREVENTS THE COUNTER  
:FROM ROLLING OVER TO ZERO.  
:THIS TEST WILL CHECK THAT THE COUNTER 'TOPS OUT' AT THE MAXIMUM COUNTER VALUE.  
:IT DO THIS MICROMODULE 'D' MICROTEST #9 IS USED. IT WILL TRANSMIT A DATAGRAM  
:OF MAXIMUM COUNTER LENGTH OVER THE LOOPBACK. THE LINK CRC HARDWARE WILL BE  
:ALLOCATED TO THE TRANSMIT SIDE SO THAT CRC BYTES WILL APPENDED TO THE DATAGRAM.  
:THE LENGTH OF THE DATAGRAM WILL THEREFORE EXCEED THE LENGTH OF THE RECEIVE  
:BYTE COUNTER. THE RECEIVE COUNTER WILL BE CHECKED TO INSURE THAT THE COUNTER  
:HAS REMAINED AT THE MAXIMUM VALUE, IF NOT AN ERROR IS PASSED TO THE HOST.  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO  
: 2-VERIFY MICROMONITOR IS IN THE 'INMON' STATE  
: 3-SELECT MICROTEST #9  
: 4-WAIT FOR 'DNI'  
: 5-CHECK FOR AN ERROR IN PCRSR1  
: 6-WRITE '1' TO CLEAR 'DNI'  
\*\*\*\*\*

BGNTST

T29::

:CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG

TRAP CSBSEG  
;HAS MICROCODE MODULE 'D' BEEN LOADED?  
;NO  
;YES, IS THE MICROMONITOR ACTIVE?  
;YES SKIP LOADING THE MICROMODULE  
;GO LOAD MICROCODE MODULE D  
;OK  
TRAP C\$ESCAPE  
.WORD L10147-.  
;WAIT FOR THE MICROMONITOR  
;OK  
;PRINT ERROR MESSAGE  
TRAP C\$ERRHD

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 199  
 CZUAAB.MAC 07-APR-83 17:03 TEST 29: LINK MAXIMUM BYTE COUNTER TEST

9407	040720'	000241							.WORD	161
9408	040722'	003027'							.WORD	MAXCNT
9409	040724'	012716'							.WORD	MSG1
9410	040726'				ESCAPE	TST				
9411	040726'	104410							TRAP	C\$ESCAPE
9412	040730'	000176							.WORD	L10147-
9413	040732'	004737	017362'	20\$:	JSR	PC,CLRDNI		:CLEAR DNI		
9414	040736'	103006			BCC	25\$				
9415	040740'				ERRHRD	162,MAXCNT,RACMG7				
9416	040740'	104456							TRAP	C\$ERHRD
9417	040742'	000242							.WORD	162
9418	040744'	003027'							.WORD	MAXCNT
9419	040746'	012670'							.WORD	RACMG7
9420	040750'				ESCAPE	TST				
9421	040750'	104410							TRAP	C\$ESCAPE
9422	040752'	000154							.WORD	L10147-
9423	040754'			25\$:						
9424	040754'				ENDSEG					
9425	040754'							10000\$:		
9426	040754'	104405							TRAP	C\$ESEG
9427										
9428					:					
9429					:	WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE, LOAD COMMAND				
9430					:	FIELD OF PCSRO WITH A 9 CAUSING THE MICROMONITOR TO EXECUTE MICROTEST #9.				
9431					:	WAIT FOR DNI				
9432	040756'				:					
9433	040756'	104404			BGNSEG					
9434	040760'	004737	020060'						TRAP	C\$BSEG
9435	040764'	103006			JSR	PC,CHKMON		:WAIT FOR MICROMONITOR		
9436	040766'				BCC	30\$		:OK		
9437	040766'	104456			ERRHRD	163,MAXCNT,MSG46		:PRINT ERROR		
9438	040770'	000243							TRAP	C\$ERHRD
9439	040772'	003027'							.WORD	163
9440	040774'	016666'							.WORD	MAXCNT
9441	040776'								.WORD	MSG46
9442	040776'	104410			ESCAPE	TST		:LEAVE TEST		
9443	041000'	000126							TRAP	C\$ESCAPE
9444	041002'	012777	000011	137326	30\$:	MOV	#9.,@PCSRO		.WORD	L10147-
9445								:TELL MICROMONITOR TO EXECUTE...		
9446	041010'	012737	000176	000332'		MOV	#2*SECOND,METER	:MICROTEST #9		
9447	041016'	004737	017316'			JSR	PC,CHKDNI	:PUT SOME TIME ON THE METER		
9448	041022'	103021				BCC	40\$	:WAIT FOR MICROTEST TO FINISH		
9449	041024'	004737	020132'			JSR	PC,CHKINT	:OK, IT FINISHED		
9450	041030'	103006				BCC	35\$	:SEE IF ANY ERROR INTERRUPTS OCCURRED		
9451	041032'					ERRHRD	164,MAXCNT,MSG44	:NO, OK		
9452	041032'	104456						:PRINT ERROR MESSAGE		
9453	041034'	000244							TRAP	C\$ERHRD
9454	041036'	003027'							.WORD	164
9455	041040'	016442'							.WORD	MAXCNT
9456	041042'				ESCAPE	TST			.WORD	MSG44
9457	041042'	104410							TRAP	C\$ESCAPE
9458	041044'	000062							.WORD	L10147-
9459	041046'	012702	000011		35\$:	MOV	#9.,R2	:MICROTEST #		
9460	041052'					ERRHRD	165,MAXCNT,MSG12	:TELL MICROTEST HUNG		
9461	041052'	104456							TRAP	C\$ERHRD
9462	041054'	000245							.WORD	165

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 200  
CZUAAB.MAC 07-APR-83 17:03 TEST 29: LINK MAXIMUM BYTE COUNTER TEST

9463 041056' 003027'  
9464 041060' 013466'  
9465 041062' 104410  
9466 041062' 104410  
9467 041064' 000042  
9468  
9469  
9470  
9471 041066' 122777 000003 137244  
9472 041074' 001004  
9473 041076'  
9474 041076' 104456  
9475 041100' 000246  
9476 041102' 003027'  
9477 041104' 016034'  
9478  
9479  
9480  
9481 041106' 004737 017362'  
9482 041112' 103004  
9483 041114'  
9484 041114' 104456  
9485 041116' 000247  
9486 041120' 003027'  
9487 041122' 012670'  
9488 041124'  
9489 041124'  
9490 041124'  
9491 041124' 104405  
9492 041126'  
9493 041126'  
9494 041126' 104401

ESCAPE TST

.WORD MAXCNT  
.WORD MSG12  
TRAP C\$ESCAPE  
.WORD L10147-

: MICROTEST IS COMPLETE, NOW CHECK TO SEE IF IT WAS SUCCESSFULL

40\$: CMPB #INERR,BPCSR1 ;DID AN ERROR OCCUR?  
BNE 50\$ ;NO  
ERRHRD 166,MAXCNT,MSG38 ;PRINT ERROR MESSAGE

TRAP C\$ERHRD  
.WORD 166  
.WORD MAXCNT  
.WORD MSG38

: WRITE ONE TO CLEAR THE DNI BIT

50\$: JSR PC,CLRDNI ;CLEAR DNI  
BCC 55\$  
ERRHRD 167,MAXCNT,RACMG7

TRAP C\$ERHRD  
.WORD 167  
.WORD MAXCNT  
.WORD RACMG7

55\$:  
ENDSEG

10001\$:  
TRAP C\$ESEG

ENDTST

L10147:  
TRAP C\$ETST

55HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 201  
CZUAAB.MAC 07-APR-83 17:03 TEST 30: FIFO TEST

9495  
9496  
9497  
9498  
9499  
9500  
9501  
9502  
9503  
9504  
9505  
9506  
9507  
9508  
9509  
9510  
9511  
9512  
9513  
9514  
9515  
9516  
9517  
9518  
9519  
9520  
9521  
9522  
9523  
9524  
9525  
9526  
9527  
9528  
9529  
9530  
9531  
9532  
9533  
9534  
9535  
9536  
9537  
9538  
9539  
9540  
9541  
9542  
9543  
9544  
9545  
9546  
9547  
9548  
9549  
9550

.SBTTL TEST 30: FIFO TEST  
:\*\*\*\*\*  
:THERE ARE TWO FIFO'S USED IN THE DEUNA TO KEEP TRACK OF RECEIVER BUFFERS.  
:THE FIRST IS CALLED THE 'RECEIVER BUFFER AVAILABLE FIFO' AND THE SECOND IS  
:CALLED THE 'RECEIVER BUFFER DONE FIFO'.  
:THE T11 LOADS THE RECEIVER BUFFER AVAILABLE FIFO WITH A LIST OF UNUSED 1K  
:BUFFERS IN LINK MEMORY. WHEN THE DEUNA SENSES THAT A PACKET IS COMING IN IT  
:PULLS AN AVAILABLE BUFFER ADDRESS FROM THE OUTPUT OF THE RECEIVER BUFFER  
:AVAILABLE FIFO AND USES IT TO ADDRESS LINK MEMORY FOR THE STORAGE OF THE  
:RECEIVED DATA. AFTER THE DATA HAS BEEN LOADED THE RECEIVER STATE MACHINE  
:PUTS THE USED BUFFER ADDRESS INTO THE RECEIVER BUFFER DONE FIFO WHERE AN  
:INTERRUPT IS GENERATED TO THE T11 WHEN IT BUBBLES TO THE TOP OF THE FIFO.  
:THESE FIFO'S ARE 64 DEEP BY 4 BITS WIDE. THE OPERATIONAL MICROCODE ONLY FILLS  
:THE FIFO TO A MAXIMUM OF 16. THE 4 BIT WIDTH REPRESENTS BITS 14-11 OF THE  
:LINK MEMORY ADDRESS. THESE BITS ALLOW THE ADDRESSING OF A 1K BUFFER IN LINK  
:MEMORY.  
:THIS TEST WILL VERIFY THAT THE RECEIVE BUFFER AVAILABLE FIFO AND THE RECEIVER  
:BUFFER DONE FIFO OPERATE CORRECTLY. THIS WILL BE DONE BY LOADING THE RECEIVER  
:BUFFER AVAILABLE FIFO WITH A 1K BUFFER ADDRESS THEN A PACKET WILL BE  
:TRANSMITTED IN LOOPBACK MODE. AFTER THE RECEIVER INTERRUPT OCCURS THE RECEIVER  
:BUFFER DONE FIFO IS READ AND THE ADDRESS IS COMPARED WITH WHAT WAS GIVEN  
:THE RECEIVER BUFFER AVAILABLE FIFO. THEY SHOULD BE THE SAME IF EVERYTHING IS  
:WORKING CORRECTLY. THE OPERATION IS PERFORMED WITH THE TRANSMITTER BUFFER  
:SET TO 0 AND WILL BE REPEATED WITH RECEIVER BUFFERS 1-15.  
:THIS TEST WILL USE MICROMODULE 'D' MICROTEST #10.  
:PARAMETERS PASSED TO THE MICROCODE WILL BE FORMATTED IN THE PCBB AS FOLLOWS:  
:PCBB+0: RECEIVE BUFFER ADDRESS  
:PCBB+2: RECEIVE BUFFER COMPLETED ADDRESS  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO  
: 2-LOAD PCBB+0 WITH A RECEIVER BUFFER  
: 3-CLEAR PCBB+2  
: 4-VERIFY MICROMONITOR IS IN THE 'INMON' STATE  
: 5-SELECT MICROTEST #10  
: 6-WAIT FOR 'DNI' BIT  
: 7-VERIFY PCBB+2 SAME AS PCBB+0  
: 8-WRITE '1' TO CLEAR 'DNI'  
: 9-REPEAT STEPS 2-8 FOR RECEIVER BUFFERS 1-15  
:\*\*\*\*\*  
BGNTST  
T30::  
:CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT. S

041130'  
041130'

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 202  
 CZUAAB.MAC 07-APR-83 17:03 TEST 30: FIFO TEST

```

9551 041130'          BGNSEG
9552 041130' 104404
9553 041132' 022737 000104 000326'      CMP    #'D,MICRO          :HAS MICROCODE MODULE 'D' BEEN LOADED
9554 041140' 001004          BNE    5%                :NO
9555 041142' 122777 000001 137170      CMPB   #INMON,@PCSR1     :YES, IS THE MICROMONITOR ACTIVE?
9556 041150' 001440          BEQ    20%                :YES SKIP LOADING THE MICROMODULE
9557 041152' 012737 000104 000326' 58:  MOV    #'D,MICRO          :GO LOAD MICRO MODULE 'D'
9558 041160' 004737 020340'          JSR    PC,LODMIC
9559 041164' 103002          BCC   10%                :OK
9560 041166'
9561 041166' 104410          ESCAPE TST
9562 041170' 000326          TRAP   C$BSEG
9563 041172' 012737 000176 000332' 10$: .WORD L10150-.
9564 041200' 004737 017316'          JSR    PC,CHKDNI
9565 041204' 103022          BCC   20%                :OK
9566 041206' 012737 001000' 000310'      MOV    #SDNI,BITNAM
9567 041214' 012737 001277' 000312'      MOV    #SNSET,BITSTA
9568 041222' 012737 001342' 000314'      MOV    #SAFTER,PWHEN
9569 041230' 012737 001357' 000316'      MOV    #SGTCMD,PCOMND
9570 041236'          ERRHRD 168,FIFTST,MSG1
9571 041236' 104456          TRAP   C$ERHRD
9572 041240' 000250          .WORD 168
9573 041242' 003073'          .WORD FIFTST
9574 041244' 012716'          .WORD MSG1
9575 041246'          ESCAPE TST
9576 041246' 104410          TRAP   C$ESCAPE
9577 041250' 000246          .WORD L10150-.
9578 041252' 004737 017362'          20$: JSR    PC,CLRDN1
9579 041256' 103006          BCC   25%                :CLEAR DNI BIT
9580 041260'          ERRHRD 169,FIFTST,RACMG7 :DNI DID NOT CLEAR!
9581 041260' 104456          TRAP   C$ERHRD
9582 041262' 000251          .WORD 169
9583 041264' 003073'          .WORD FIFTST
9584 041266' 012670'          .WORD RACMG7
9585 041270'          ESCAPE TST
9586 041270' 104410          TRAP   C$ESCAPE
9587 041272' 000224          .WORD L10150-.
9588 041274'          25$:
9589 041274'          ENDSEG
9590 041274'          10000$:
9591 041274' 104405          TRAP   C$ESEG
9592
9593          :
9594          :LINK MEMORY STARTS AT MICROMEMORY ADDRESS 100000, THE MICROTEST WILL USE
9595          :100000 AS THE TRANSMIT BUFFER ADDRESS SO THE FIRST RECEIVE BUFFER ADDRESS
9596          :WE WILL PASS TO THE MICROCODE WILL BE 104000
9597 041276' 012701 104000          :
9598          :MOV    #LINADR+SIZ1K,R1      :FIRST RECEIVE BUFFER STARTS 1K FROM
9599 041302'          30$:          :BASE OF LINK MEMORY
9600          :
9601          :PUT THE BUFFER ADDRESS WE WANT THE MICROCODE TO LOAD INTO THE AVAILABLE BUFFER
9602          :INTO PCBB AND CLEAR PCBB+2. WAIT FOR THE MICROMONITOR TO BECOME READY, THEN
9603          :SELECT MICROTEST #10 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 10. WAIT FOR
9604          :'DNI'
9605          :
9606 041302'          BGNSEG

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 203  
CZUAAB.MAC 07-APR-83 17:03 TEST 30: FIFO TEST

```

9607 041302' 104404
9608 041304' 010137 000606' MOV R1,PCBB ;PASS AVAILABLE RECEIVER BUFFER TO DEUNA
9609 041310' 005037 000610' CLR PCBB+2 ;HERE IS WHERE THE MICRO WILL PUT THE
9610 ; 'DONE' RECEIVE BUFFER FIFO ADDRESS
9611 041314' 004737 020060' JSR PC,LHKMON ;WAIT FOR MICROMONITOR
9612 041320' 103006 BCC 35$ ;OK
9613 041322' ;ERRHRD 170,FIFTST,MSG46 ;PRINT ERROR
9614 041322' 104456
9615 041324' 000252 TRAP C$BSEG
9616 041326' 003073' .WORD 170
9617 041330' 016666' .WORD FIFTST
9618 041332' ESCAPE TST ;LEAVE TEST .WORD MSG46
9619 041332' 104410 TRAP C$ESCAPE
9620 041334' 000162 .WORD L10150-
9621 041336' 012777 000012 136772 35$: MOV #10.,@PCSR0 ;TELL T11 TO EXECUTE MICROTEST #10
9622 041344' 012737 000176 000332' MOV #2*SECOND,METER ;WAIT FOR DNI
9623 041352' 004737 017316' JSR PC,CHKDNI
9624 041356' 103021 BCC 45$ ;OK
9625 041360' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
9626 041364' 103006 BCC 40$ ;NO, OK
9627 041366' ;ERRHRD 171,FIFTST,MSG44 ;PRINT ERROR MESSAGE
9628 041366' 104456 TRAP C$ERHRD
9629 041370' 000253 .WORD 171
9630 041372' 003073' .WORD FIFTST
9631 041374' 016442' .WORD MSG44
9632 041376' ESCAPE TST
9633 041376' 104410 TRAP C$ESCAPE
9634 041400' 000116 .WORD L10150-
9635 041402' 012702 000012 40$: MOV #10.,R2 ;MICROTEST #10 IS HUNG
9636 041406' ;ERRHRD 172,FIFTST,MSG12 ;PRINT ERROR MESSAGE
9637 041406' 104456 TRAP C$ERHRD
9638 041410' 000254 .WORD 172
9639 041412' 003073' .WORD FIFTST
9640 041414' 013466' .WORD MSG12
9641 041416' ESCAPE TST
9642 041416' 104410 TRAP C$ESCAPE
9643 041420' 000076 .WORD L10150-
9644 ;
9645 ;OK, THE MICROCODE HAS FILLED PCBB+2 WITH THE OUTPUT OF THE RECEIVER BUFFER
9646 ;DONE FIFO HOWEVER, THIS VALUE IS NOT IN ADDRESS FORM BECAUSE THERE ARE SOME
9647 ;UNUSED BITS (15,10:00) THAT COULD BE FLOATING SO, STRIP THESE BITS FIRST.
9648 ;THEN SET BIT 15 TO FORCE THE ADDRESS TO BE A LINK MEMORY ADDRESS. COMPARE
9649 ;THE ADDRESS GIVEN TO THE MICROCODE WITH THE ONE JUST GENERATED, THEY SHOULD
9650 ;BE IDENTICAL, IF NOT, PRINT ERROR.
9651 ;
9652 041422' 042737 103777 000610' 45$: BIC #103777,PCBB+2 ;STRIP OFF THE FLOATING BITS
9653 041430' 052737 100000 000610' BIS #100000,PCBB+2 ;MAKE IT A LINK MEMORY ADDRESS
9654 041436' 023737 000606' 000610' CMP PCBB,PCBB+2 ;DID THE DEUNA USE THE BUFFER AVAILABLE
9655 041444' 001404 BEQ 50$ ;YES
9656 041446' ;ERRHRD 173,FIFTST,MSG35 ;NO, PRINT ERROR MESSAGE
9657 041446' 104456 TRAP C$ERHRD
9658 041450' 000255 .WORD 173
9659 041452' 003073' .WORD FIFTST
9660 041454' 015672' .WORD MSG35
9661 041456' 50$:
9662 041456' ENDSEG

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 204  
 CZUAAB.MAC 07-APR-83 17:03 TEST 30: FIFO TEST

SEQ 199

```

9663 041456'
9664 041456' 104405
9665
9666
9667
9668 041460' 004737 017362'
9669 041464' 103006
9670 041466'
9671 041466' 104456
9672 041470' 000256
9673 041472' 003073'
9674 041474' 012670'
9675 041476'
9676 041476' 104410
9677 041500' 000016
9678
9679
9680
9681
9682 041502' 020127 174000
9683 041506' 001403
9684 041510' 062701 004000
9685 041514' 000672
9686 041516'
9687 041516'
9688 041516'
9689 041516' 104401

:WRITE '1' TO CLEAR 'DNI' BIT
:
JSR PC,CLRDNI :GO CLEAR DNI
BCC 60$ :OK
ERRHRD 174,FIFTST,RACMG7 :ERROR DNI DID NOT CLEAR!
TRAP CSEHRD
.WORD 174
.WORD FIFTST
.WORD RACMG7
ESCAPE TST
TRAP CSESCAPE
.WORD L10150-

:CHECK TO SEE IF WE HAVE TRIED ALL 15 RECEIVER BUFFER ADDRESSES, IF NOT,
:GENERATE A NEW ADDRESS TO PASS TO THE MICROCODE AND RUN THRU THE TEST AGAIN
:
60$: CMP R1,#LINADR+<SIZ1K+15.> :HAVE WE TRIED ALL 15 BUFFERS
BEQ 70$ :YES ALL DONE
ADD #SIZ1K,R1 :NO POINT TO NEXT RECEIVER BUFFER
BR 30$ :DO AGAIN

70$:
ENDTST

10001$: TRAP CSESEG
L10150: TRAP CSETST

```

6SHARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 205  
CZUAAB.MAC 07-APR-83 17:03

TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

.SBTTL TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

9690  
9691  
9692  
9693  
9694  
9695  
9696  
9697  
9698  
9699  
9700  
9701  
9702  
9703  
9704  
9705  
9706  
9707  
9708  
9709  
9710  
9711  
9712  
9713  
9714  
9715  
9716  
9717  
9718  
9719  
9720  
9721  
9722  
9723  
9724  
9725  
9726  
9727  
9728  
9729  
9730  
9731  
9732

\*\*\*\*\*  
: THIS TEST WILL VERIFY THAT BUFFERS 1-15 OF LINK MEMORY CAN BE ADDRESSED  
: CORRECTLY BY THE RECEIVER. THIS WILL BE DONE BY DIRECTING THE MICROCODE TO  
: TRANSMIT A DATA PATTERN FROM BUFFER 0 AND TO RECEIVE THE DATA IN BUFFER X  
: WHERE X = 1-15. THEN A CHECK WILL BE MADE TO SEE IF THE PATTERN NOT ONLY  
: ARRIVED IN THE CORRECT RECEIVER BUFFER BUT THAT THE PATTERN DOES NOT SHOW UP  
: ANYWHERE ELSE IN LINK MEMORY EXCEPT WHERE IT WAS SUPPOSE TO.  
:  
: THIS TEST WILL USE MICROMODULE 'D' MICROTEST #11. THIS MICROTEST ACCEPTS 2  
: PARAMETERS: THE TRANSMIT BUFFER AND THE RECEIVER BUFFER. IT WILL SET UP  
: A DATA PATTERN IN THE TRANSMIT BUFFER AND TELL THE LINK TO TRANSMIT, IN  
: LOOPBACK MODE, FROM THE TRANSMIT BUFFER GIVEN TO THE RECEIVER BUFFER GIVEN.  
: AFTER THE RECEIVER INTERRUPT, THE DATA IS CHECKED IN THE EXPECTED RECEIVER  
: BUFFER FOR THE CORRECT DATA PATTERN. THEN ALL OF LINK MEMORY (EXCEPT FOR THE  
: TRANSMIT BUFFER) IS CHECKED TO SEE IF THE PATTERN ENDS UP ELSEWHERE. IF  
: AN ERROR IS FOUND THE MICROCODE PASSES THE ADDRESS OF LINK MEMORY WHERE THE  
: ERROR WAS FOUND, THE DATA THAT WAS FOUND THERE ALONG WITH THE DATA THAT SHOULD  
: HAVE BEEN THERE.  
:  
: THE PARAMETERS FOR THE MICROCODE ARE FORMATED IN THE PCBB AS FOLLOWS:  
:  
: PCBB+0: RECEIVER BUFFER ADDRESS  
: PCBB+2: TRANSMIT BUFFER ADDRESS  
: PCBB+4: LINK MEMORY ADDRESS (IF ERROR)  
: PCBB+6: GOOD DATA PATTERN (IF ERROR)  
: PCBB+10: BAD DATA PATTERN (IF ERROR)  
:  
: TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO  
: 2-SET PCBB+0 WITH TRANSMIT BUFFER 0 ADDRESS  
: 3-SET PCBB+2 WITH RECEIVER BUFFER ADDRESS  
: 4-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE  
: 5-SELECT MICROTEST #11  
: 6-WAIT FOR 'DNI'  
: 7-CHECK FOR ERROR IN PCSR1 IF SO PRINT ERROR MESSAGE  
: 8-WRITE '1' TO CLEAR 'DNI'  
: 9-REPEAT STEPS 2-8 WITH RECEIVER BUFFERS 1-15  
:\*\*\*\*\*

9733 041520'  
9734 041520'  
9735  
9736  
9737  
9738  
9739

BGNTST

T31::

: CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO  
: THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
: AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

9740 041520'  
9741 041520' 104404  
9742 041522' 022737 000104 000326'  
9743 041530' 001004  
9744 041532' 122777 000001 136600  
9745 041540' 001440

BGNSEG

CMP #D,MICRO ;HAS MICROCODE MODULE 'D' BEEN LOADED  
BNE 5\$ ;NO  
CMPB #INMON,APCSR1 ;YES, IS THE MICROMONITOR ACTIVE?  
BEQ 20\$ ;YES SKIP LOADING THE MICROMODULE  
TRAP CSBSEG



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 206  
 CZUAAB.MAC 07-APR-83 17:03 TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

```

9746 041542' 012737 000104 000326' 58:  MOV  #D,MICRO           :GO LOAD MICRO MODULE 'D'
9747 041550' 004737 020340'  JSR  PC,LODMIC
9748 041554' 103002  BCC  10$           :OK
9749 041556'  ESCAPE TST
9750 041556' 104410  TRAP  C$ESCAPE
9751 041560' 000322  .WORD L10151-.
9752 041562' 012737 000176 000332' 10$:  MOV  #2*SECOND,METER  :WAIT FOR THE MICROMONITOR
9753 041570' 004737 017316'  JSR  PC,CHKDNI
9754 041574' 103022  BCC  20$           :OK
9755 041576' 012737 001000' 000310'  MOV  #SDNI,BITNAM
9756 041604' 012737 001277' 000312'  MOV  #SNSET,BITSTA
9757 041612' 012737 001342' 000314'  MOV  #SAFTER,PWHEN
9758 041620' 012737 001357' 000316'  MOV  #SGTCMD,PCOMND
9759 041626'  ERRHRD 175,RLNKAD,MSG1
9760 041626' 104456  TRAP  C$ERHRD
9761 041630' 000257  .WORD 175
9762 041632' 003114'  .WORD RLNKAD
9763 041634' 012716'  .WORD MSG1
9764 041636'  ESCAPE TST
9765 041636' 104410  TRAP  C$ESCAPE
9766 041640' 000242  .WORD L10151-.
9767 041642' 004737 017362' 20$:  JSR  PC,CLRDN1      :CLEAR DNI BIT
9768 041646' 103006  BCC  25$
9769 041650'  ERRHRD 176,RLNKAD,RACMG7  :DNI DID NOT CLEAR!
9770 041650' 104456  TRAP  C$ERHRD
9771 041652' 000260  .WORD 176
9772 041654' 003114'  .WORD RLNKAD
9773 041656' 012670'  .WORD RACMG7
9774 041660'  ESCAPE TST
9775 041660' 104410  TRAP  C$ESCAPE
9776 041662' 000220  .WORD L10151-.
9777 041664' 25$:
9778 041664'  ENDSEG
9779 041664' 10000$:
9780 041664' 104405  TRAP  C$SEEG
9781  :
9782  :LOAD PCBB+2 WITH THE ADDRESS OF THE FIRST TRANSMIT BUFFER
9783  :LOAD PCBB+0 WITH THE FIRST RECEIVER BUFFER ADDRESS
9784  :
9785 041666' 012701 002756  MOV  #MAXBYT,R1      :TRANSMIT BYTE COUNT
9786 041672' 012737 100000 000610'  MOV  #LINADR,PCBB+2  :SET TRANSMIT BUFFER AT BASE OF LINK
9787  :MEMORY
9788 041700' 012737 104000 000606'  MOV  #LINADR+SIZEK,PCBB :FIRST RECEIVE BUFFER STARTS 1K FROM
9789  :BASE OF LINK MEMORY
9790 041706' 30$:
9791 041706'  BGNSEG
9792 041706' 104404  TRAP  C$BSEG
9793  :
9794  :WAIT FOR THE MICROMONITOR, THEN SELECT MICROTEST #11 BY LOADING THE COMMAND
9795  :FIELD OF PCSRO WITH 11 , WAIT FOR 'DNI'
9796  :
9797 041710' 004737 020060'  JSR  PC,CHKMON      :WAIT FOR MICROMONITOR
9798 041714' 103006  BCC  35$           :OK
9799 041716'  ERRHRD 177,RLNKAD,MSG46 :PRINT ERROR
9800 041716' 104456  TRAP  C$ERHRD
9801 041720' 000261  .WORD 177

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 207  
 CZUAA8.IIAC 07-APR-83 17:03 TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

```

9802 041722' 003114'
9803 041724' 016666'
9804 041726'          ESCAPE TST          ;LEAVE TEST          .WORD  RLNKAD
9805 041726' 104410          ;                               .WORD  MSG46
9806 041730' 000152          TRAP    C$ESCAPE
9807 041732' 012777 000013 136376 35$: MOV    #11,@PCSR0          ;TELL T11 TO EXECUTE MICROTEST #11
9808 041740' 012737 000176 000332' MOV    #2*SECOND,METER          ;WAIT FOR DNI
9809 041746' 004737 017316' JSR    PC,CHKDNI
9810 041752' 103021          BCC    45$
9811 041754' 004737 020132' JSR    PC,CHKINT          ;OK
9812 041760' 103006          BCC    40$
9813 041762'          ERRHRD 178,RLNKAD,MSG44          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
9814 041762' 104456          ;NO, OK
9815 041764' 000262          ;PRINT ERROR MESSAGE
9816 041766' 003114'          TRAP    C$ERHRD
9817 041770' 016442'          .WORD  178
9818 041772'          ESCAPE TST          .WORD  RLNKAD
9819 041772' 104410          ;                               .WORD  MSG44
9820 041774' 000106          TRAP    C$ESCAPE
9821 041776' 012702 000013 40$: MOV    #11,R2          ;MICROTEST #11 IS HUNG
9822 042002'          ERRHRD 179,RLNKAD,MSG12          ;PRINT ERROR MESSAGE
9823 042002' 104456          TRAP    C$ERHRD
9824 042004' 000263          .WORD  179
9825 042006' 003114'          .WORD  RLNKAD
9826 042010' 013466'          ;OR      MSG12
9827 042012'          ESCAPE TST
9828 042012' 104410          TRAP    C$ESCAPE
9829 042014' 000066          .WORD  L10151-.
9830
9831 ;IF THE MICROCODE FOUND AN ERROR IT SETS PCRS1 TO THE ERROR STATE SO. CHECK
9832 ;PCRS1 TO SEE IF AN ERROR OCCURRED
9833 ;IF SO PRINT ERROR MESSAGE
9834
9835 042016' 122777 000003 136314 45$: CMPB  #INERR,@PCSR1          ;DID AN ERROR OCCUR?
9836 042024' 001004          BNE    50$
9837 042026'          ERRHRD 180,RLNKAD,MSG42          ;NO
9838 042026' 104456          ;YES, PRINT ERROR MESSAGE
9839 042030' 000264          TRAP    C$ERHRD
9840 042032' 003114'          .WORD  180
9841 042034' 016250'          .WORD  RLNKAD
9842 042036'          ;OR      MSG42
9843 042036'          50$:
9844 042036'          ENDSEG
9845 042036' 104405          10001$: TRAP    C$ESEG
9846
9847 ;WRITE '1' TO CLEAR 'DNI'
9848
9849 042040' 004737 017362' JSR    PC,CLEAR DNI          ;GO CLEAR DNI
9850 042044' 103006          BCC    60$
9851 042046'          ERRHRD 181,RLNKAD,RACMG7          ;OK
9852 042046' 104456          ;ERROR DNI DID NOT CLEAR!
9853 042050' 000265          TRAP    C$ERHRD
9854 042052' 003114'          .WORD  181
9855 042054' 012670'          .WORD  RLNKAD
9856 042056'          .WORD  RACMG7
9857 042056' 104410          ESCAPE TST
9858
9859
9860
9861
9862
9863
9864
9865
9866
9867
9868
9869
9870
9871
9872
9873
9874
9875
9876
9877
9878
9879
9880
9881
9882
9883
9884
9885
9886
9887
9888
9889
9890
9891
9892
9893
9894
9895
9896
9897
9898
9899
9900
9901
9902
9903
9904
9905
9906
9907
9908
9909
9910
9911
9912
9913
9914
9915
9916
9917
9918
9919
9920
9921
9922
9923
9924
9925
9926
9927
9928
9929
9930
9931
9932
9933
9934
9935
9936
9937
9938
9939
9940
9941
9942
9943
9944
9945
9946
9947
9948
9949
9950
9951
9952
9953
9954
9955
9956
9957
9958
9959
9960
9961
9962
9963
9964
9965
9966
9967
9968
9969
9970
9971
9972
9973
9974
9975
9976
9977
9978
9979
9980
9981
9982
9983
9984
9985
9986
9987
9988
9989
9990
9991
9992
9993
9994
9995
9996
9997
9998
9999

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 208  
CZUAAB.MAC 07-APR-83 17:03 TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

9858 042060' 000022

.WORD L10151-

9859

9860

9861

9862

9863

9864

9865 042062' 023727 000606' 174000

9866 042070' 001404

9867 042072' 062737 004000 000606'

9868 042100' 000702

9869 042102'

9870 042102'

9871 042102'

9872 042102' 104401

9873

; THIS TEST IS TO BE REPEATED 15 TIMES ONCE FOR EACH RECEIVER BUFFER (1-15).  
; CHECK TO SEE IF WE HAVE PASSED THE LAST RECEIVER BUFFER TO THE MICROCODE.  
; IF SO STOP, ELSE ADD 1K TO THE CONTENTS OF PCBB TO MAKE THE NEW RECEIVER  
; BUFFER AND REPEAT THE TEST AGAIN

60\$: CMP PCBB,#LINADR+<SIZ1K\*15.> ;HAVE WE TRIED ALL 15 BUFFERS  
BEQ 70\$ ;YES ALL DONE  
ADD #SIZ1K,PCBB ;NO POINT TO NEXT RECEIVER BUFFER  
BR 30\$ ;DO AGAIN

70\$:  
ENDTST

L10151:  
TRAP CSETST



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 210  
 CZUAAB.MAC 07-APR-83 17:03 TEST 32: TRANSMITTER LINK MEMORY ADDRESS TEST

```

9930 042226' 004737 017362' 20$: JSR PC,CLRDNI ;CLEAR DNI BIT
9931 042232' 103006 BCC 25$
9932 042234' ERRHRD 183,TLNKAD,RACMG7 ;DNI DID NOT CLEAR!
9933 042234' 104456 TRAP C$ERHRD
9934 042236' 000267 .WORD 183
9935 042240' 003165' .WORD TLNKAD
9936 042242' 012670' .WORD RACMG7
9937 042244' ESCAPE TST
9938 042244' 104410 TRAP C$ESCAPE
9939 042246' 000220 .WORD L10152-.
9940 042250' 25$:
9941 042250' ENDSEG
9942 042250'
9943 042250' 104405 10000$: TRAP C$ESEG
9944
9945 ;
9946 ;LOAD PCBB+0 WITH THE BASE ADDRESS OF LINK MEMORY. THIS IS WHERE THE RECEIVER
9947 ;BUFFER WILL BE FIXED AT. LOAD PCBB+2 WITH THE ADDRESS OF BUFFER #1 (104000)
9948 042252' 012701 002756 MOV #MAXBYT,R1 ;TRANSMIT BYTE COUNT
9949 042256' 012737 100000 000606' MOV #LINADR,PCBB ;SET RECEIVE BUFFER AT BASE OF LINK
9950 ;MEMORY
9951 042264' 012737 104000 000610' MOV #LINADR+SIZ1K,PCBB+2 ;TRANSMIT BUFFER STARTS 1K FROM
9952 ;BASE OF LINK MEMORY
9953 042272' 30$:
9954 ;
9955 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE. THEN EXECUTE MICROTEST
9956 ;#11 BY LOADING PCSRO COMMAND FIELD BITS WITH A 11. WAIT FOR 'DNI'
9957 ;
9958 ; BGNSEG
9959 042272' 104404 TRAP C$BSEG
9960 042274' 004737 020060' JSR PC,CHKMON ;WAIT FOR MICROMONITOR
9961 042300' 103006 BCC 35$ ;OK
9962 042302' ERRHRD 184,TLNKAD,MSG46 ;PRINT ERROR
9963 042302' 104456 TRAP C$ERHRD
9964 042304' 000270 .WORD 184
9965 042306' 003165' .WORD TLNKAD
9966 042310' 016666' .WORD MSG46
9967 042312' ESCAPE TST ;LEAVE TEST
9968 042312' 104410 TRAP C$ESCAPE
9969 042314' 000152 .WORD L10152-.
9970 042316' 012777 000013 136012 35$: MOV #11, @PCSRO ;TELL T11 TO EXECUTE MICROTEST #11
9971 042324' 012737 000176 000332' MOV #2*SECOND,METER ;WAIT FOR DNI
9972 042332' 004737 017316' JSR PC,CHKDNI
9973 042336' 103021 BCC 45$ ;OK
9974 042340' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
9975 042344' 103006 BCC 40$ ;NO, OK
9976 042346' ERRHRD 185,TLNKAD,MSG44 ;PRINT ERROR MESSAGE
9977 042346' 104456 TRAP C$ERHRD
9978 042350' 000271 .WORD 185
9979 042352' 003165' .WORD TLNKAD
9980 042354' 016442' .WORD MSG44
9981 042356' ESCAPE TST
9982 042356' 104410 TRAP C$ESCAPE
9983 042360' 000106 .WORD L10152-.
9984 042362' 012702 000013 40$: MOV #11, R2 ;MICROTEST #11 IS HUNG
9985 042366' ERRHRD 186,TLNKAD,MSG12 ;PRINT ERROR MESSAGE

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 211  
CZUAAB.MAC 07-APR-83 17:03 TEST 32: TRANSMITTER LINK MEMORY ADDRESS TEST

```

9986 042366' 104456
9987 042370' 000272
9988 042372' 003165'
9989 042374' 013466'
9990 042376'
9991 042376' 104410
9992 042400' 000066
9993
9994
9995
9996
9997 042402' 122777 000003 135730
9998 042410' 001004
9999 042412'
10000 042412' 104456
10001 042414' 000273
10002 042416' 003165'
10003 042420' 016250'
10004 042422'
10005 042422'
10006 042422'
10007 042422' 104405
10008
10009
10010
10011 042424' 004737 017362'
10012 042430' 103006
10013 042432'
10014 042432' 104456
10015 042434' 000274
10016 042436' 003165'
10017 042440' 012670'
10018 042442'
10019 042442' 104410
10020 042444' 000022
10021
10022
10023
10024
10025
10026 042446' 023727 000610' 174000
10027 042454' 001404
10028 042456' 062737 004000 000610'
10029 042464' 000702
10030 042466'
10031 042466'
10032 042466'
10033 042466' 104401
10034

```

```

          TRAP C$ERHRD
          .WORD 186
          .WORD TLNKAD
          .WORD MSG12
          ESCAPE TST
          TRAP C$ESCAPE
          .WORD L10152-.
          :THE MICROCODE WILL SET PCSR1 TO AN ERROR CONDITION IF IT DETECTED AN ERROR.
          :SO CHECK PCSR1 IF ERROR, PRINT ERROR MESSAGE
45$: CMPB #INERR,PCSR1 ;DID AN ERROR OCCUR?
      BNE 50$ ;NO
      ERHRD 187,TLNKAD,MSG42 ;YES, PRINT ERROR MESSAGE
          TRAP C$ERHRD
          .WORD 187
          .WORD TLNKAD
          .WORD MSG42
50$:
      ENDSEG
          10001$: TRAP C$ESEG
          :WRITE '1' TO CLEAR 'DNI'
          :
          JSR PC,CLRDMI ;GO CLEAR DNI
          BCC 60$ ;OK
          ERHRD 188,TLNKAD,RACMG7 ;ERROR DNI DID NOT CLEAR!
          TRAP C$ERHRD
          .WORD 188
          .WORD TLNKAD
          .WORD RACMG7
          ESCAPE TST
          TRAP C$ESCAPE
          .WORD L10152-.
          :THIS TEST IS TO BE REPEATED FOR EACH TRANSMIT BUFFER SO CHECK IF THE MICROCODE
          :HAS BEEN PASSED THE LAST TRANSMIT BUFFER, IF NOT, CHANGE PCBB+2 TO POINT
          :TO THE NEXT TRANSMIT BUFFER BY ADDING 1K TO IT
          :
          60$: CMP PCBB+2,#LINADR+<SIZ1K*15.> ;TRIED BUFFERS 1-15 FOR TRANSMITTER?
              BEQ 70$ ;YES ALL DONE
              ADD #SIZ1K,PCBB+2 ;NO POINT TO NEXT TRANSMIT BUFFER
              BR 30$ ;DO AGAIN
          70$:
              ENDTST
          L10152: TRAP C$ETST

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 212  
CZUAAB.MAC 07-APR-83 17:03

TEST 33: LINK MEMORY ARBITRATION TEST

10035  
10036  
10037  
10038  
10039  
10040  
10041  
10042  
10043  
10044  
10045  
10046  
10047  
10048  
10049  
10050  
10051  
10052  
10053  
10054  
10055  
10056  
10057  
10058  
10059  
10060  
10061  
10062  
10063  
10064  
10065  
10066  
10067  
10068  
10069  
10070  
10071  
10072  
10073  
10074  
10075  
10076  
10077  
10078  
10079  
10080  
10081  
10082  
10083  
10084  
10085  
10086  
10087  
10088  
10089  
10090

.SBTTL TEST 33: LINK MEMORY ARBITRATION TEST

\*\*\*\*\*

:THE LINK MEMORY CAN BE ACCESSED BY FOUR PROCESSES; THE T-11 PROCESSOR, THE  
:DMA ENGINE, THE RECEIVE STATE MACHINE AND THE TRANSMIT STATE MACHINE. THE  
:PORT MODULE HAS ARBITRATION CIRCUITRY TO MANAGE LINK MEMORY ACCESSES. THIS  
:CIRCUITRY PREVENTS CONFLICTS BETWEEN PROCESSES AND ASSURES THAT HIGHER  
:PRIORITY PROCESSES GET PRECEDENCE.

:THIS TEST WILL VERIFY THE ABILITY OF THE LINK MEMORY ARITRATOR TO HANDLE  
:SIMULTANEOUS REQUESTS BY FOUR PROCESSES. EACH OF THESE PROCESSES WILL INVOLVE  
:TASKS THAT REQUIRE HEAVY ACCESSES OF LINK MEMORY. DATA WILL BE MOVED INTO OR  
:OUT OF LINK MEMORY BY EACH. WHEN THAT TASKS ARE FINISHED THE DATA WILL BE  
:VERIFIED.

:THE FOUR PROCESSES ARE:

1-TRANSMIT STATE MACHINE

WILL TRANSMIT A DATAGRAM OF MAXIMUM DATA LENGTH IN LOOPBACK  
MODE. THE DATA FIELD WILL CONTAIN A BIT PATTERN STRING OF  
TWO 1'S FOLLOWED BY TWO 0'S I.E. 31463 (OCTAL).

2-RECEIVE STATE MACHINE

WILL RECEIVE A DATAGRAM OF MAXIMUM DATA LENGTH OVER THE  
LOOPBACK. THE RECEIVE DATA BUFFER WILL BE FILLED WITH 0'S  
PRIOR TO THE RECEPTION.

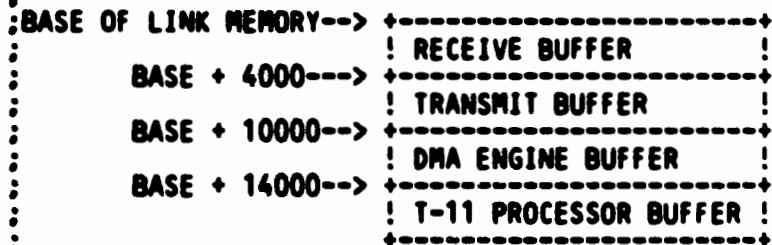
3-T-11 MICROPROCESSOR DMA

A 1K BUFFER IN LINK MEMORY WILL BE FILLED WITH AN ALL 1'S DATA  
PATTERN PRIOR TO THE OPERATION THEN ALTERNATING 1'S AND 0'S  
DATA PATTERN WILL BE WRITTEN.

4-DMA ENGINE

WILL TRANSFER A 1K BLOCK OF DATA FROM LINK MEMORY TO UNIBUS  
MEMORY. THE DATA IN LINK MEMORY WILL A BIT PATTERN STRING  
OF FOUR 1'S FOLLOWED BY A STRING OF FOUR 0'S. THE BUFFER  
IN UNIBUS MEMORY WILL BE CLEARED PRIOR TO THE OPERATION.

:THE FOUR PROCESSES WILL WORK OUT OF FOUR SEPARATE AREAS OF LINK MEMORY.



:THIS WILL ALLOW THE ARITRATION CIRCUITRY TO BE TESTED AND YET ALLOWS THE DATA





65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 214  
 CZUAAB.MAC 07-APR-83 17:03 TEST 33: LINK MEMORY ARBITRATION TEST

```

10147 042510' 001440          BEQ      20$          ;YES SKIP LOADING THE MICROMODULE
10148 042512' 012737 000104 000326' 5$:  MOV      #'D,MICRO  ;GO LOAD MICRO MODULE 'D'
10149 042520' 004737 020340'          JSR      PC,LODMIC
10150 042524' 103002          BCC      10$          ;OK
10151 042526'          ESCAPE  TST
10152 042526' 104410          TRAP    C$ESCAPE
10153 042530' 000376          .WORD  L10153-.
10154 042532' 012737 000176 000332' 10$:  MOV      #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
10155 042540' 004737 017316'          JSR      PC,CHKDNI
10156 042544' 103022          BCC      20$          ;OK
10157 042546' 012737 001000' 000310'  MOV      #SDNI,BITNAM
10158 042554' 012737 001277' 000312'  MOV      #SNSET,BITSTA
10159 042562' 012737 001342' 000314'  MOV      #SAFTER,PWHEN
10160 042570' 012737 001357' 000316'  MOV      #SGTCMD,PCOMND
10161 042576'          ERRHRD  189,LNKARB,MSG1
10162 042576' 104456          TRAP    C$ERHRD
10163 042600' 000275          .WORD  189
10164 042602' 003241'          .WORD  LNKARB
10165 042604' 012716'          .WORD  MSG1
10166 042606'          ESCAPE  TST
10167 042606' 104410          TRAP    C$ESCAPE
10168 042610' 000316          .WORD  L10153-.
10169 042612' 004737 017362'          20$:  JSR      PC,CLRDNI  ;CLEAR DNI BIT
10170 042616' 103006          BCC      25$
10171 042620'          ERRHRD  190,LNKARB,RACMG7 ;DNI DID NOT CLEAR!
10172 042620' 104456          TRAP    C$ERHRD
10173 042622' 000276          .WORD  190
10174 042624' 003241'          .WORD  LNKARB
10175 042626' 012670'          .WORD  RACMG7
10176 042630'          ESCAPE  TST
10177 042630' 104410          TRAP    C$ESCAPE
10178 042632' 000274          .WORD  L10153-.
10179 042634'          25$:
10180 042634'          ENDSEG
10181 042634'          10000$:
10182 042634' 104405          TRAP    C$ESEG
10183          ;
10184          ;SETUP PCBB TO BE JUST AFTER LAST LOCATION USED BY THIS DIAGNOSTIC
10185          ;TELL MICROCODE WHERE PCBB IS BY LOADING PCRS2 AND PCRS3 WITH THE PCBB ADDRESS
10186          ;
10187 042636' 013777 000324' 135476  MOV      FREMEM,@PCRS2 ;TELL MICROCODE WHERE PCBB IS
10188 042644' 005077 135474  CLR      @PCRS3
10189          ;
10190          ;FIRST CLEAR THE 3 WORDS THAT WILL BE USED BY THE MICROCODE IF AN ERROR OCCURS.
10191          ;THEN CLEAR OUT THE BUFFER TO BE USED AS THE DMA ENGINE'S 'TO' BUFFER.
10192          ;
10193 042650' 013700 000324'  MOV      FREMEM,RO ;GET A POINTER TO PCBB
10194 042654' 005020          CLR      (RO)+ ;HERE IS WHERE MICROCODE WILL PUT
10195          ;EXPECTED DATA PATTERN IF ERROR
10196 042656' 005020          CLR      (RO)+ ;HERE, THE BAD DATA
10197 042660' 005020          CLR      (RO)+ ;AND HERE THE LINK MEMORY ADDRESS
10198          ;
10199 042662' 012701 002000  30$:  MOV      #SZ1K/2,R1 ;SIZE OF DMA ENGINE 'TO' BUFFER
10200 042666' 005020          CLR      (RO)+ ;CLEAR THE AREA WHERE THE DMA ENGI...
10201 042670' 005301          DEC      R1 ;WILL LOAD DATA
10202 042672' 001375          BNE     30$

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 215  
 CZUAB.MAC 07-APR-83 17:03 TEST 33: LINK MEMORY ARBITRATION TEST

```

10203 042674'
10204 042674'
10205 042674' 104404
10206
10207
10208
10209
10210
10211 042676' 004737 020060'
10212 042702' 103006
10213 042704'
10214 042704' 104456
10215 042706' 000277
10216 042710' 003241'
10217 042712' 016666'
10218 042714'
10219 042714' 104410
10220 042716' 000210
10221 042720' 012777 000010 135410 368:
10222 042726' 012737 000176 000332'
10223 042734' 004737 017316'
10224 042740' 103021
10225 042742' 004737 020132'
10226 042746' 103006
10227 042750'
10228 042750' 104456
10229 042752' 000300
10230 042754' 003241'
10231 042756' 016442'
10232 042760'
10233 042760' 104410
10234 042762' 000144
10235 042764' 012702 000010 378:
10236 042770'
10237 042770' 104456
10238 042772' 000301
10239 042774' 003241'
10240 042776' 013466'
10241 043000'
10242 043000' 104410
10243 043002' 000124
10244
10245
10246
10247
10248 043004' 122777 000003 135326 408:
10249 043012' 001013
10250 043014' 013701 000324'
10251 043020' 013702 000326'
10252 043024' 013703 000330'
10253 043030'
10254 043030' 104456
10255 043032' 000302
10256 043034' 003241'
10257 043036' 014146'
10258 043040' 000410

358:
  BGNSEG
  TRAP C$BSEG
  :
  :WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
  :SELECT MICROTEST #8 BY LOADING PCSRO COMMAND FIELD BITS WITH A #8.
  :WAIT FOR 'DNI'
  :
  JSR PC,CHKMON ;WAIT FOR MICROMONITOR
  BCC 36$ ;OK
  ERRHRD 191,LNKARB,MSG46 ;PRINT ERROR
  TRAP C$ERHRD
  .WORD 191
  .WORD LNKARB
  .WORD MSG46
  ESCAPE TST ;LEAVE TEST
  TRAP C$ESCAPE
  .WORD L10153-
  MOV #8.,DPCSR0 ;TELL T11 TO EXECUTE MICROTEST #8
  MOV #2*SECOND,METER ;WAIT FOR DNI
  JSR PC,CHKDNI
  BCC 40$
  JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
  BCC 37$ ;NO, OK
  ERRHRD 192,LNKARB,MSG44 ;PRINT ERROR MESSAGE
  TRAP C$ERHRD
  .WORD 192
  .WORD LNKARB
  .WORD MSG44
  ESCAPE TST
  TRAP C$ESCAPE
  .WORD L10153-
  MOV #8.,R2 ;MICROTEST #8 IS HUNG
  ERRHRD 193,LNKARB,MSG12
  TRAP C$ERHRD
  .WORD 193
  .WORD LNKARB
  .WORD MSG12
  ESCAPE TST
  TRAP C$ESCAPE
  .WORD L10153-
  :
  :CHECK MICROCODE FINISH STATUS BY CHECKING PCSR1 STATE BITS FOR AN ERROR
  :IF ERROR, PRINT ERROR MESSAGE
  :
  CMPN #INERR,@PCSR1 ;DID AND ERROR OCCUR?
  BNE 50$ ;NO
  MOV FREM,R1 ;YES, GET EXPECTED DATA PATTERN
  MOV FREM+2,R2 ;GET BAD DATA
  MOV FREM+4,R3 ;GET LINK MEMORY ADDRESS
  ERRHRD 194,LNKARB,MSG20 ;PRINT ERROR MESSAGE
  TRAP C$ERHRD
  .WORD 194
  .WORD LNKARB
  .WORD MSG20
  BR 60$

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 216  
 CZUAAB.MAC 07-APR-83 17:03 TEST 33: LINK MEMORY ARBITRATION TEST

```

10259
10260
10261
10262 043042' 013701 000324'
10263 043046' 062701 000006
10264 043052' 012704 003774
10265 043056' 012702 000017
10266 043062' 111103
10267 043064' 120211
10268 043066' 001404
10269 043070'
10270 043070' 104456
10271 043072' 000303
10272 043074' 003241'
10273 043076' 014236'
10274 043100' 005201
10275 043102' 005304
10276 043104' 001366
10277
10278
10279
10280
10281 043106' 004737 017362'
10282 043112' 103004
10283 043114'
10284 043114' 104456
10285 043116' 000304
10286 043120' 003241'
10287 043122' 012670'
10288 043124'
10289 043124'
10290 043124'
10291 043124' 104405
10292 043126'
10293 043126'
10294 043126' 104401

:CHECK DATA THAT THE DMA ENGINE TRANSFERRED TO UNIBUS MEMORY
50$: MOV FPOMEM,R1 ;GET POINTER TO DMA ENGINE BUFFER
ADD #6,R1
MOV #3774,R4 ;NUMBER OF BYTES
MOV #17,R2 ;DATA THAT SHOULD BE IN BUFFER
60$: MOVB (R1),R3 ;GET DATA FROM BUFFER
CMPB R2,(R1) ;IS DATA CORRECT?
BEQ 70$ ;YES
ERRHRD 195,LNKARB,MSG21 ;NO, PRINT ERROR
TRAP C$ERHRD
.WORD 195
.WORD LNKARB
.WORD MSG21
70$: INC R1
DEC R4
BNE 60$ ;CONTINUE CHECKING
;ALL DONE
:WRITE '1' TO CLEAR 'DNI'
80$: JSR PC,CLRDN1 ;CLEAR DNI BIT
BCC 90$
ERRHRD 196,LNKARB,RACMG7 ;ERROR DNI DID NOT CLEAR!
TRAP C$ERHRD
.WORD 196
.WORD LNKARB
.WORD RACMG7
90$: ENDSEG
10001$: TRAP C$ESEG
ENDTST
L10153: TRAP C$ETST

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 217  
CZUAAB.MAC 07-APR-83 17:03 TEST 34: STATION ADDRESS PATTERN TEST

10295  
10296  
10297  
10298  
10299  
10300  
10301  
10302  
10303  
10304  
10305  
10306  
10307  
10308  
10309  
10310  
10311  
10312  
10313  
10314  
10315  
10316  
10317  
10318  
10319  
10320  
10321  
10322  
10323  
10324  
10325  
10326  
10327  
10328  
10329  
10330  
10331  
10332  
10333  
10334  
10335  
10336  
10337  
10338  
10339  
10340  
10341  
10342  
10343  
10344  
10345  
10346  
10347  
10348  
10349  
10350

.SBTTL TEST 34: STATION ADDRESS PATTERN TEST

:\*\*\*\*\*8

:WITHOUT EITHER THE PROMISCUIOUS MODE OR THE MULTICAST MODE ENABLED, THE LINK  
:LOGIC WILL RECOGNIZE DATAGRAM ADDRESSES ONLY IF THE ADDRESS IS CONTAINED IN  
:THE STATION ADDRESS RAM.

:WHEN A DATAGRAM ARRIVES, THE LINK LOGIC COMPARES THE DATAGRAM DESTINATION  
:ADDRESS FIELD TO THE 12 ADDRESSES WRITTEN IN THE STATION ADDRESS RAM. IF THE  
:INCOMING ADDRESS MATCHES ONE OF THESE, THEN THE DATAGRAM WILL BE ACCEPTED BY  
:THE LINK. THE 'MATCH' BIT IS SET IN THE TRANSMIT BUFFER AND THE RECEIVING  
:PROCESS BEGINS.

:THIS TEST WILL VERIFY THAT THE LINK CAN RECOGNIZE A DATAGRAM WHEN THE  
:DESTINATION ADDRESS OF THE DATAGRAM MATCHES ONE OF THE ADDRESSES STORED IN  
:THE STATION ADDRESS RAM.

:THIS TEST WILL USE MICROMODULE 'E' MICROTEST #1.  
:PATTERNS WILL BE USED FOR ADDRESSES IN CHECKING THE STATION ADDRESS LOGIC.  
:THE PATTERNS WILL BE SUPPLIED TO THE T-11 THROUGH THE PCBB. THE MICROCODE  
:WILL BE RESTARTED FOR EACH DIFFERENT PATTERN TO BE TESTED. UPON START-UP, THE  
:T-11 PROCESSOR WILL PICK UP THE CURRENT PATTERN/ADDRESS, LOAD THE SAME PATTERN  
:INTO ALL 12 LOCATIONS OF THE STATION ADDRESS RAM, FORMAT THE TRANSMIT BUFFER  
:AND LOGIC FOR A LOOPBACK, PRESET PCSR1 TO AN ERROR CONDITION, START THE LINK  
:AND WAIT FOR THE MATCH BIT IN THE TRANSMIT BUFFER. IF THE MATCH BIT SETS  
:THE PCSR1 ERROR CONDITION IS CLEARED AND THE T-11 WAITS FOR BOTH THE  
:TRANSMITTER AND RECEIVER INTERRUPTS BEFORE IT SETS 'DNI' TO INDICATE T-  
:TEST WAS SUCCESSFUL

:THE PCBB WILL BE USED TO PASS THE 48 BIT STATION ADDRESS PATTERN:



:THE FOLLOWING PATTERNS WILL BE USED:  
:-ALTERNATING 1'S AND 0'S  
:-ALTERNATING 0'S AND 1'S  
:-PAIR OF 0'S FOLLOWED BY PAIR OF 1'S  
:-FOUR 0'S FOLLOWED BY FOUR 1'S  
:-EIGHT 0'S FOLLOWED BY EIGHT 1'S  
:-SIXTEEN 1'S FOLLOWED BY SIXTEEN 0'S FOLLOWED BY 16 1'S  
:-TWENTYFOUR 1'S FOLLOWED BY TWENTYFOUR 0'S

:TEST SEQUENCE:  
:1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO  
:2-LOAD PCBB+0,+2,+4 WITH A PATTERN  
:3-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE  
:4-SELECT MICROTEST #1  
:5-WAIT FOR 'DNI'  
:6-IF NO DNI AND PCSR1 INDICATES ERROR THEN PRINT NO MATCH BIT SET



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 219  
CZUAAB.MAC 07-APR-83 17:03 TEST 34: STATION ADDRESS PATTERN TEST

```

10407 ;SELECT ONE OF THE PATTERNS FROM THE STATION ADDRESS PATTERN TABLE AND LOAD
10408 ;IT INTO THE PCBB
10409 ;
10410 043276' 012701 000534'      MOV      #SPAT1,R1      ;POINT TO STATION ADDRESS PATTERN TABLE
10411 043302' 012705 000007'      MOV      #7,R5         ;# OF ADDRESS PATTERNS
10412 043306' 011137 000606'      27$:    MOV      (R1),PCBB ;LOAD PCBB WITH AN ADDRESS PATTERN
10413 043312' 016137 000002' 000610'  MOV      2(R1),PCBB+2 ;MIDDLE ORDER
10414 043320' 016137 000004' 000612'  MOV      4(R1),PCBB+4 ;UPPER ORDER
10415 043326'
10416 043326' 104404      BGNSEG
10417
10418 ;
10419 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE. SELECT MICROTEST #1
10420 ;BY LOADING PCSRO COMMAND FIELD BITS WITH A 1. WAIT FOR DNI.
10421 043330' 004737 020060'      JSR      PC,CHKMON    ;WAIT FOR MICROMONITOR
10422 043334' 103006      BCC     30$          ;OK
10423 043336'      ERRHRD 199,STAPAT,MSG46 ;PRINT ERROR
10424 043336' 104456
10425 043340' 000307      TRAP    C$ERHRD
10426 043342' 003305'      .WORD  199
10427 043344' 016666'      .WORD  STAPAT
10428 043346'      .WORD  MSG46
10429 043346' 104410      ESCAPE TST          ;LEAVE TEST
10430 043350' 000106      TRAP    C$ESCAPE
10431 043352' 012777 000001' 134756 30$:    MOV      #1,@PCSRO    ;TELL T11 TO EXECUTE MICROTEST #1
10432 043360' 012737 000176' 000332'  MOV      #2+SECOND,METER ;WAIT FOR DNI
10433 043366' 004737 017316'      JSR      PC,CHKDNI
10434 043372' 103015      BCC     40$
10435 ;
10436 ;DNI DID NOT SET WHICH INDICATES THAT THE RECEIVER INTERRUPT NEVER HAPPENED.
10437 ;THE REASON MIGHT BE BECAUSE THE MATCH BIT NEVER SET IN THE TRANSMIT
10438 ;STATUS REGISTER INDICATING THAT THE STATION ADDRESS COMPARATOR FAILED
10439 ;TO RECOGNIZE THE DATAGRAM DESTINATION ADDRESS. THE MICROCODE CHECKS
10440 ;FOR THE MATCH BIT AND IF IT IS NOT SET IT PUTS AN ERROR STATUS IN PCSR1.
10441 ;IF PCSR1 DOES NOT HAVE AN ERROR STATE THEN THE RECEIVER INTERRUPT FAILED
10442 ;FOR SOME OTHER REASON AND ITS NOT THE STATION ADDRESS RECOGNITION'S FAULT.
10443 ;
10444 043374' 004737 020132'      JSR      PC,CHKINT    ;SEE IF ANY ERROR INTERRUPTS OCCURRED
10445 043400' 103006      BCC     35$          ;NO
10446 043402'      ERRHRD 200,STAPAT,MSG44 ;PRINT ERROR MESSAGE
10447 043402' 104456
10448 043404' 000310      TRAP    C$ERHRD
10449 043406' 003305'      .WORD  200
10450 043410' 016442'      .WORD  STAPAT
10451 043412'      .WORD  MSG44
10452 043412' 104410      ESCAPE TST          ;LEAVE TST
10453 043414' 000042      TRAP    C$ESCAPE
10454 043416'      .WORD  L10154-.
10455 043416' 104456      35$:    ERRHRD 201,STAPAT,MSG25 ;STATION ADDRESS PATTERN WAS NOT
10456 043420' 000311      TRAP    C$ERHRD
10457 043422' 003305'      .WORD  201
10458 043424' 014614'      .WORD  STAPAT
10459      .WORD  MSG25
10460 ;
10461 ;WRITE '1' TO CLEAR 'DNI'
10462 ;

```

65 HARDWARE TESTS  
CZUAAB.MAC 07-APR-83

MACY11 30A(1052) 07-APR-83 17:13 PAGE 220  
TEST 34: STATION ADDRESS PATTERN TEST

```

10463 043426' 004737 017362'      40$: JSR PC,CLRDNI      ;CLEAR DNI BIT
10464 043432' 103004                BCC 50$
10465 043434'                        ERRHRD 202,STAPAT,RACMG7 ;ERROR DNI DID NOT CLEAR'
10466 043434' 104456                TRAP CSEHRD
10467 043436' 000312                .WORD 202
10468 043440' 003305'                .WORD STAPAT
10469 043442' 012670'                .WORD RACMG7
10470 043444'
10471 043444'
10472 043444'
10473 043444' 104405                10001$: TRAP CSESEG
10474
10475
10476 ;REPEAT THE TEST WITH ALL SEVEN PATTERNS
10477 043446' 062701 000006        ADD #6,R1      ;POINT TO NEXT PATTERN
10478 043452' 005305                DEC R5         ;TESTED WITH ALL ADDRESS PATTERNS?
10479 043454' 001314                BNE 27$       ;NOT YET
10480
10481 043456'
10482 043456'
10483 043456' 104401                ENDTST        L10154: TRAP CSETST

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 221  
CZUAAB.MAC 07-APR-83 17:03 TEST 35: STATION ADDRESS REJECTION TEST

10484  
10485  
10486  
10487  
10488  
10489  
10490  
10491  
10492  
10493  
10494  
10495  
10496  
10497  
10498  
10499  
10500  
10501  
10502  
10503  
10504  
10505  
10506  
10507  
10508  
10509  
10510  
10511  
10512  
10513 043460'  
10514 043460'  
10515  
10516  
10517  
10518  
10519  
10520 043460'  
10521 043460' 104404  
10522 043462' 022737 000105 000326'  
10523 043470' 001004  
10524 043472' 122777 000001 134640  
10525 043500' 001440  
10526 043502' 012737 000105 000326' 58:  
10527 043510' 004737 020340'  
10528 043514' 103002  
10529 043516'  
10530 043516' 104410  
10531 043520' 000256  
10532 043522' 012737 000176 000332' 108:  
10533 043530' 004737 017316'  
10534 043534' 103022  
10535 043536' 012737 001000' 000310'  
10536 043544' 012737 001277' 000312'  
10537 043552' 012737 001342' 000314'  
10538 043560' 012737 001357' 000316'  
10539 043566'

.SBTTL TEST 35: STATION ADDRESS REJECTION TEST

:\*\*\*\*\*

:THIS TEST WILL VERIFY THAT THE STATION ADDRESS DETECTION LOGIC DOES NOT  
:RECOGNIZE A DATAGRAM WHEN THE DATAGRAM ADDRESS IS NOT CONTAINED IN THE  
:STATION ADDRESS RAM.

:THE MICROCODE WILL FILL THE STATION ADDRESS RAM WITH 0'S. THE DESTINATION  
:FIELD OF THE TRANSMIT BUFFER IS FILLED WITH 1'S . A TRANSMISSION IS STARTED  
:IN LOOPBACK MODE AND THE T-11 WILL WAIT FOR A RECEIVER INTERRUPT. OF COURSE,  
:THE RECEIVER INTERRUPT SHOULD NEVER HAPPEN BECAUSE THE STATION ADDRESS  
:LOGIC SHOULD NOT GET A SUCCESSFUL COMPARISON BETWEEN 0'S IN THE DESTINATION  
:ADDRESS OF THE INCOMING DATAGRAM AND THE 1'S IN THE STATION ADDRESS RAM.  
:THE T-11 WILL BE PUT INTO A LOOP WAITING FOR A RECEIVER INTERRUPT AND THE  
:DEUNA TIMER IS STARTED. IF THE LOOP IS BROKEN BY THE RECEIVER INTERRUPT  
:AN ERROR WILL BE PRESENTED IN PCSR1 BY THE MICROCODE. IF THE LOOP IS BROKEN  
:BY THE TIMER THEN THE TEST WAS SUCCESSFUL.

:TEST SEQUENCE:

- 1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO
- 2-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
- 3-SELECT MICROTTEST #2
- 4-WAIT FOR 'DNI'
- 5-CHECK PCSR1 FOR AN ERROR CONDITION
- 6-WRITE '1' TO CLEAR 'DNI'

:\*\*\*\*\*

BGNTST

T35::

:CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG

						TRAP	C\$BSEG
				CMP	#'E,MICRO		:HAS MICROCODE MODULE 'E' BEEN LOADED
				BNE	58		:NO
				CMPB	#INMON,PCSR1		:YES, IS THE MICROMONITOR ACTIVE?
				BEQ	208		:YES SKIP LOADING THE MICROMODULE
				MOV	#'E,MICRO		:GO LOAD MICRO MODULE 'E'
				JSR	PC,LODMIC		
				BCC	108		:OK
				ESCAPE	TST		
						TRAP	C\$ESCAPE
						.WORD	L10155-
				MOV	#2*SECOND,METER		:WAIT FOR THE MICROMONITOR
				JSR	PC,CHKDNI		
				BCC	208		:OK
				MOV	#\$DNI,BITNAM		
				MOV	#\$NSET,BITSTA		
				MOV	#\$AFTER,PWHEN		
				MOV	#\$GTCMD,PCOMND		
				ERRHRD	203,STAREJ,MSG1		





65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 223  
 CZUAB.MAC 07-APR-83 17:03 TEST 35: STATION ADDRESS REJECTION TEST

```

10596 043724' 000317
10597 043726' 003351'
10598 043730' 013466'
10599 043732'
10600 043732' 104410
10601 043734' 000042
10602
10603
10604
10605
10606 043736' 122777 000003 134374
10607 043744' 001004
10608 043746'
10609 043746' 104456
10610 043750' 000320
10611 043752' 003351'
10612 043754' 000000
10613
10614
10615
10616 043756' 004737 017362'
10617 043762' 103004
10618 043764'
10619 043764' 104456
10620 043766' 000321
10621 043770' 003351'
10622 043772' 012670'
10623 043774'
10624 043774'
10625 043774'
10626 043774' 104405
10627 043776'
10628 043776'
10629 043776' 104401

          ESCAPE TST

          ;
          ;OK 'DNI' SET INDICATING EITHER THE RECEIVER INTERRUPT HAPPENED OR THE TIMER
          ;BROKE THE LOOP, CHECK PCSR1 TO FIND OUT WHICH
          ;
          40$:  CMPB  #INERR,@PCSR1      ;DID AN ERROR OCCUR?
                BNE   50$              ;NO
                ERRHRD 208,STAREJ      ;YES, PRINT ERROR MESSAGE
          TRAP  C$ERHRD
          .WORD 208
          .WORD STAREJ
          .WORD 0

          ;WRITE '1' TO CLEAR 'DNI'
          ;
          50$:  JSR   PC,CLRDNI          ;CLEAR DNI BIT
                BCC  60$
                ERRHRD 209,STAREJ,RACMG7 ;ERROR DNI DID NOT CLEAR!
          TRAP  C$ERHRD
          .WORD 209
          .WORD STAREJ
          .WORD RACMG7

          60$:  ENDSEG

          10001$: TRAP  C$ESEG

          L10155: TRAP  C$ETST

          ENDTST

```

65HARDWARE TESTS  
CZUAAB.MAC

MACY11 30A(1052)  
07-APR-83 17:03

07-APR-83 17:13 PAGE 224  
TEST 36: STATION ADDRESS RAM POSITION TEST

10630  
10631  
10632  
10633  
10634  
10635  
10636  
10637  
10638  
10639  
10640  
10641  
10642  
10643  
10644  
10645  
10646  
10647  
10648  
10649  
10650  
10651  
10652  
10653  
10654  
10655  
10656  
10657  
10658  
10659  
10660  
10661  
10662  
10663  
10664  
10665  
10666  
10667  
10668  
10669  
10670  
10671  
10572  
10673  
10674  
10675  
10676  
10677  
10678  
10679  
10680  
10681  
10682  
10683  
10684  
10685

```

.SBTTL TEST 36. STATION ADDRESS RAM POSITION TEST
:*****
:THE STATION ADDRESS RAM CAN HOLD UP TO 12 STATION ADDRESSES. WHEN A DATAGRAM
:IS RECEIVED THE STATION ADDRESS COMPARISON LOGIC DOES A BIT-WISE COMPARISON
:OF ALL 12 RAM STATION ADDRESS WITH THE INCOMING DATAGRAM STATION ADDRESS.
:
:THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE A STATION ADDRESS
:REGARDLESS OF THE LOCATION OF THE ADDRESS IN THE STATION ADDRESS RAM.
:
:THIS TEST WILL USE MICROMODULE 'E' MICROTEST #3. THE MICROCODE WILL WRITE
:A STATION ADDRESSES OF ALL 1'S INTO A SINGLE LOCATION OF THE STATION ADDRESS
:RAM. THE OTHER ELEVEN LOCATION WILL BE LOADED WITH 0'S. A DATAGRAM WITH
:AN ALL 1'S DESTINATION ADDRESS WILL BE TRANSMITTED IN LOOPBACK MODE. THE TEST
:WILL VERIFY THAT THE DATAGRAM IS RECEIVED. THE TEST WILL BE REPEATED FOR ALL
:TWELVE LOCATIONS OF THE STATION ADDRESS RAM.
:
:THE MICROTEST WILL BE REPEATED FOR EACH OF THE 12 TEST ITERATIONS. THE PCBB
:WILL BE USED TO PASS TO THE MICROCODE WHICH POSITION IS TO BE LOADED WITH
:1'S. WHEN THE STATION ADDRESS IS LOADED, THE STATION ADDRESSES MUST BE
:ROTATED ORTHOGONALLY, I.E. BIT 0 OF ALL STATION ADDRESSES LOADED TOGETHER,
:THEN BIT 1, THEN BIT 2 ETC. THIS MAKES IT DIFFICULT TO DESCRIBE THE POSITION
:OF ANY SINGLE STATION ADDRESS IN TERMS OF AN OFFSET FROM THE RAM BASE ADDRESS.
:
:THE PCBB IS FORMATTED AS FOLLOWS:
:
:PCBB+0:      +-----+
:              | RAM ADDRESS POSTION |
:              +-----+
:
:TEST SEQUENCE:
: 1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO.
: 2-LOAD PCBB+0 WITH THE RAM POSTION TO BE TESTED
: 3-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
: 4-SELECT MICROTEST #3
: 5-WAIT FOR 'DNI'
: 6-CHECK PCSR1 FOR AN ERROR CONDITION
: 7-WRITE '1' TO CLEAR 'DNI'
: 8-REPEAT STEPS 2-7 FOR ALL 12 RAM POSTIONS
:*****

```

044000'  
044000'

BGNTST

T36::

```

:CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
:

```

BGNSEG

044000' 104404  
044000' 022737 000105 000326'  
044010' 001004  
044012' 122777 000001 134320  
044020' 001440

```

CMP #'E,MICRO
BNE 58
CMPB #INMON,@PCSR1
BEQ 208

```

```

TRAP CSBSEG
:HAS MICROCODE MODULE 'E' BEEN LOADED
:NO
:YES, IS THE MICROMONITOR ACTIVE?
:YES SKIP LOADING THE MICROMODULE

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 225  
 CZUAAB.MAC 07-APR-83 17:03 TEST 36: STATION ADDRESS RAM POSITION TEST

```

10686 044022' 012737 000105 000326' 58:  MOV    #'E,MICRO           ;GO LOAD MICRO MODULE 'E'
10687 044030' 004737 020340'                JSR    PC,LODMIC
10688 044034' 103002                BCC    10$                ;OK
10689 044036'                ESCAPE TST
10690 044036' 104410                TRAP   C$ESCAPE
10691 044040' 000306                .WORD L10156-.
10692 044042' 012737 000176 000332' 10$:  MOV    #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
10693 044050' 004737 017316'                JSR    PC,CHKDNI
10694 044054' 103022                BCC    20$                ;OK
10695 044056' 012737 001000' 000310'  MOV    #SDNI,BITNAM
10696 044064' 012737 001277' 000312'  MOV    #SNSET,BITSTA
10697 044072' 012737 001342' 000314'  MOV    #SAFTER,PWHEN
10698 044100' 012737 001357' 000316'  MOV    #SGTCMD,PCOMND
10699 044106'                ERRHRD 210,STAPOS,MSG1
10700 044106' 104456                TRAP   C$ERHRD
10701 044110' 000322                .WORD 210
10702 044112' 003417'                .WORD STAPOS
10703 044114' 012716'                .WORD MSG1
10704 044116'                ESCAPE TST
10705 044116' 104410                TRAP   C$ESCAPE
10706 044120' 000226                .WORD L10156-.
10707 044122' 004737 017362' 20$:  JSR    PC,CLR,DNI       ;CLEAR DNI BIT
10708 044126' 103006                BCC    25$
10709 044130'                ERRHRD 211,STAPOS,RACMG7 ;DNI DID NOT CLEAR!
10710 044130' 104456                TRAP   C$ERHRD
10711 044132' 000323                .WORD 211
10712 044134' 003417'                .WORD STAPOS
10713 044136' 012670'                .WORD RACMG7
10714 044140'                ESCAPE TST
10715 044140' 104410                TRAP   C$ESCAPE
10716 044142' 000204                .WORD L10156-.
10717 044144'                25$:
10718 044144'                ENDSEG
10719 044144'                10000$:
10720 044144' 104405                TRAP   C$ESEG
10721                ;
10722                ;R1 WILL CONTAIN THE RAM POSTION TO BE TESTED
10723                ;
10724 044146' 012701 000001                MOV    #1,R1             ;BEGIN WITH RAM POSTION #1
10725 044152'                30$:
10726                ;
10727                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, LOAD PCBB WITH
10728                ;THE RAM POSTION TO BE TESTED, SELECT MICROTEST #3 BY LOADING PCSRO COMMAND
10729                ;FIELD WITH A 3, AND WAIT FOR 'DNI' TO SET.
10730                ;
10731                ;BGNSEG
10732 044152' 104404                TRAP   C$BSEG
10733 044154' 004737 020060'                JSR    PC,CHKMON        ;WAIT FOR MICROMONITOR
10734 044160' 103006                BCC    35$                ;OK
10735 044162'                ERRHRD 212,STAPOS,MSG46 ;PRINT ERROR
10736 044162' 104456                TRAP   C$ERHRD
10737 044164' 000324                .WORD 212
10738 044166' 003417'                .WORD STAPOS
10739 044170' 016666'                .WORD MSG46
10740 044172'                ESCAPE TST             ;LEAVE TEST
10741 044172' 104410                TRAP   C$ESCAPE

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 226  
CZUAAB.MAC 07-APR-83 17:03 TEST 36: STATION ADDRESS RAM POSITION TEST

```

10742 044174' 000152
10743 044176' 010137 000606'
10744 044202' 012777 000003 134126
10745 044210' 012737 000275 000332'
10746 044216' 004737 017316'
10747 044222' 103021
10748 044224' 004737 020132'
10749 044230' 103006
10750 044232'
10751 044232' 104456
10752 044234' 000325
10753 044236' 003417'
10754 044240' 016442'
10755 044242'
10756 044242' 104410
10757 044244' 000102
10758 044246' 012702 000003
10759 044252'
10760 044252' 104456
10761 044254' 000326
10762 044256' 003417'
10763 044260' 013466'
10764 044262'
10765 044262' 104410
10766 044264' 000062
10767
10768
10769
10770 044266' 122777 000003 134044
10771 044274' 001004
10772 044276'
10773 044276' 104456
10774 044300' 000327
10775 044302' 003417'
10776 044304' 015002'
10777 044306'
10778 044306'
10779 044306'
10780 044306' 104405
10781
10782
10783
10784 044310'
10785 044310' 104404
10786 044312' 004737 017362'
10787 044316' 103006
10788 044320'
10789 044320' 104456
10790 044322' 000330
10791 044324' 003417'
10792 044326' 012670'
10793 044330'
10794 044330' 104410
10795 044332' 000014
10796 044334'
10797 044334'

```

```

35$: MOV R1,PCBB ;TELL MICROCODE WHICH RAM POSTION TO USE
      MOV #3,BPCSR0 ;TELL T11 TO EXECUTE MICROTEST #3
      MOV #3*SECOND,METER ;WAIT FOR DNI
      JSR PC,LHKDNI
      BCC 40$
      JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
      BCC 36$ ;NO, OK
      ERRHRD 213,STAPOS,MSG44 ;PRINT ERROR MESSAGE

```

```

TRAP C$ERHRD
.WORD 213
.WORD STAPOS
.WORD MSG44

```

```

TRAP C$ESCAPE
.WORD L10156-

```

```

36$: MOV #3,R2 ;MICROTEST #3 IS HUNG
      ERRHRD 214,STAPOS,MSG12

```

```

TRAP C$ERHRD
.WORD 214
.WORD STAPOS
.WORD MSG12

```

```

TRAP C$ESCAPE
.WORD L10156-

```

```

;CHECK PCSR1 TO SEE IF THE DATAGRAM WAS RECEIVED
40$: CMPB #INERR,BPCSR1 ;DID AN ERROR OCCUR?
      BNE 45$ ;NO
      ERRHRD 215,STAPOS,MSG26 ;PRINT ERROR MESSAGE

```

```

TRAP C$ERHRD
.WORD 215
.WORD STAPOS
.WORD MSG26

```

```

45$:
      ENDSEG

```

```

10001$:
      TRAP C$ESEG

```

```

;WRITE '1' TO CLEAR 'DNI'
;
      BGNSEG

```

```

TRAP C$BSEG

```

```

      JSR PC,CLRDN1 ;GO CLEAR DNI
      BCC 50$ ;OK
      ERRHRD 216,STAPOS,RACMG7 ;ERROR OCCURRED PRINT MESSAGE

```

```

TRAP C$ERHRD
.WORD 216
.WORD STAPOS
.WORD RACMG7

```

```

      ESCAPE TST

```

```

TRAP C$ESCAPE
.WORD L10156-

```

```

50$:
      ENDSEG

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 227  
CZUAAB.MAC 07-APR-83 17:03 TEST 36: STATION ADDRESS RAM POSITION TEST

10798 044334'  
10799 044334' 104405  
10800  
10801  
10802  
10803 044336' 005201  
10804 044340' 020127 000014  
10805 044344' 103702  
10806 044346'  
10807 044346'  
10808 044346' 104401

:  
:REPEAT TEST FOR ALL 12 RAM POSTIONS  
:

INC R1  
CMP R1,#12.  
BLO 30\$  
ENDTST

:DO NEXT RAM POSTION  
:HAVE WE DONE ALL 12 RAM POSTIONS?  
:NO KEEP GOING

10002\$: TRAP C\$ESEG

L10156: TRAP C\$ETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 228  
CZUAAB.MAC 07-APR-83 17:03 TEST 37: MULTICAST ADDRESS TEST

10809  
10810  
10811  
10812  
10813  
10814  
10815  
10816  
10817  
10818  
10819  
10820  
10821  
10822  
10823  
10824  
10825  
10826  
10827  
10828  
10829  
10830  
10831  
10832  
10833  
10834  
10835  
10836  
10837  
10838  
10839  
10840  
10841  
10842  
10843  
10844  
10845  
10846  
10847  
10848  
10849  
10850  
10851  
10852  
10853  
10854  
10855  
10856  
10857  
10858  
10859  
10860  
10861  
10862  
10863  
10864

044350\*  
044350\*  
  
  
  
  
  
  
  
  
044350\*  
044350\* 104404  
044352\* 022737  
044360\* 001004  
044362\* 122777  
044370\* 001440  
044372\* 012737  
044400\* 004737  
044404\* 103002  
044406\*  
044406\* 104410  
044410\* 000256  
044412\* 012737  
044420\* 004737  
044424\* 103022  
044426\* 012737  
044434\* 012737  
044442\* 012737

.SBTTL TEST 37: MULTICAST ADDRESS TEST  
:.....  
:MULTICAST ADDRESSING PERMITS THE DEUNA TO RESPOND TO MESSAGES AIMED AT  
:LOGICALLY RELATED DEVICES ON THE NETWORK. THE MSB OF THE DESTINATION ADDRESS  
:OF THESE MESSAGES IS A 1. THIS BIT IS DETECTED BY THE ADDRESS RECOGNITION  
:LOGIC.  
:THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE AND ACCEPT MESSAGES WITH  
:THE MULTICAST BIT DESIGNATION.  
:THIS TEST WILL USE MICROMODULE 'E' MICROTEST #4.  
:THE MICROCODE WILL PREPARE A DATAGRAM WITH THE DESTINATION ADDRESS HAVING  
:THE MULTICAST BIT SET. THE DEUNA WILL BE SETUP IN LOOPBACK MODE WITH 'ENABLE  
:ALL MULTICAST'. THE DATAGRAM WILL BE TRANSMITTED AND THE T-11 WILL BE PUT IN  
:A LOOP WAITING FOR A RECEIVER INTERRUPT. THE TIMER WILL INTERRUPT THE LOOP  
:IF THE RECEIVER INTERRUPT DOES NOT OCCUR. IF THIS HAPPENS, PCSR1 WILL INDICATE  
:AN ERROR, OTHERWISE WHEN THE RECEIVER INTERRUPT OCCURS IT WILL BREAK THE LOOP  
:AND PCSR1 WILL INDICATE A SUCCESSFULL COMPLETION OF THE TEST.  
:TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO  
: 2-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE  
: 3-SELECT MICROTEST #4  
: 4-WAIT FOR 'DNI'  
: 5-CHECK PCSR1 FOR AN ERROR CONDITION  
: 6-WRITE '1' TO CLEAR 'DNI'  
:.....

BGNTST  
T37::  
:CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TC CLEAR IT.  
: BGNSEG  
TRAP CSBSEG  
:HAS MICROCODE MODULE 'E' BEEN LOADED  
:NO  
:YES, IS THE MICROMONITOR ACTIVE?  
:YES SKIP LOADING THE MICROMODULE  
:GO LOAD MICRO MODULE 'E'  
:OK  
TRAP CSBSEG  
:WAIT FOR THE MICROMONITOR  
:OK  
L10157-.

000105 000326\*  
000001 133750  
000105 000326\* 5\$:  
020340\*  
000176 000332\* 10\$:  
001000\* 000310\*  
001277\* 000312\*  
001342\* 000314\*

CMR #E,MICRO  
BNE 5\$  
CMPB #INMON,BPCSR1  
BEQ 20\$  
MOV #E,MICRO  
JSR PC,LODMIC  
BCC 10\$  
ESCAPE TST  
  
MOV #2\*SECOND,METER  
JSR PC,CHKDNI  
BCC 20\$  
MOV #SDNI,BITNAM  
MOV #SNSET,BITSTA  
MOV #SAFTER,PMEN

```

65HARDWARE TESTS      MACY11 30A(1052) 07-APR-83 17:13 PAGE 229
CZUAAB.MAC           07-APR-83 17:03      TEST 37: MULTICAST ADDRESS TEST
10865 044450' 012737 001357' 000316'      MOV      #SGTCMD,PCOMND
10866 044456'      ERRHRD  217,MUCAST,MSG1
10867 044456' 104456
10868 044460' 000331
10869 044462' 003470'
10870 044464' 012716'
10871 044466'      ESCAPE  TST
10872 044466' 104410
10873 044470' 000176
10874 044472' 004737 017362'      20$:  JSR      PC,CLRDNI      :CLEAR DNI BIT
10875 044476' 103006      BCC      25$
10876 044500'      ERRHRD  218,MUCAST,RACMG7      :DNI DID NOT CLEAR!
10877 044500' 104456
10878 044502' 000332
10879 044504' 003470'
10880 044506' 012670'
10881 044510'      ESCAPE  TST
10882 044510' 104410
10883 044512' 000154
10884 044514'      25$:
10885 044514'      ENDSEG
10886 044514'
10887 044514' 104405      10000$:
10888
10889      :WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
10890
10891 044516'      BGNSEG
10892 044516' 104404
10893 044520' 004737 020060'      JSR      PC,CHKMON      :WAIT FOR MICROMONITOR
10894 044524' 103006      BCC      30$
10895 044526'      ERRHRD  219,MUCAST,MSG46      :OK
10896 044526' 104456      :PRINT ERROR
10897 044530' 000333
10898 044532' 003470'
10899 044534' 016666'
10900 044536'      ESCAPE  TST
10901 044536' 104410
10902 044540' 000126
10903 044542'      30$:
10904
10905      :EXECUTE MICROTEST #4 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 4
10906
10907 044542' 012777 000004 133566      MOV      #4,BPCSRO      :TELL T11 TO EXECUTE MICROTEST #4
10908
10909      :WAIT AT LEAST 3 SECONDS FOR TEST TO FINISH BECAUSE MICROCODE WAITS 2 SECONDS
10910      :FOR THE RECEIVER INTERRUPT
10911
10912 044550' 012737 000275 000332'      MOV      #3*SECOND,METER      :WAIT FOR DNI
10913 044556' 004737 017316'      JSR      PC,CHKDNI
10914 044562' 103021      BCC      40$
10915 044564' 004737 020132'      JSR      PC,CHKINT      :SEE IF ANY ERROR INTERRUPTS OCCURRED
10916 044570' 103006      BCC      35$
10917 044572'      ERRHRD  220,MUCAST,MSG44      :NO, OK
10918 044572' 104456      :PRINT ERROR MESSAGE
10919 044574' 000334
10920 044576' 003470'

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 230  
 CZUAAB.MAC 07-APR-83 17:03 TEST 37: MULTICAST ADDRESS TEST

```

10921 044600' 016442' .WORD MSG44
10922 044602' ESCAPE TST
10923 044602' 104410 TRAP C$ESCAPE
10924 044604' 000062 .WORD L10157-.
10925 044606' 012702 000004 35$: MOV #4,M2 ;MICROTEST #4 IS HUNG
10926 044612' ERRHRD 221,MUCAST,MSG12
10927 044612' 104456 TRAP C$ERRHRD
10928 044614' 000335 .WORD 221
10929 044616' 003470' .WORD MUCAST
10930 044620' 013466' .WORD MSG12
10931 044622' ESCAPE TST
10932 044622' 104410 TRAP C$ESCAPE
10933 044624' 000042 .WORD L10157-.
10934
10935 ;OK, EITHER THE TIMER BROKE THE LOOP OR THE RECEIVER INTERRUPT DID.
10936 ;WHICH WAS IT?
10937
10938 044626' 122777 000003 133504 40$: CMPB #INERR,@PCSR1 ;DID AN ERROR OCCUR?
10939 044634' 001004 BNE 50$ ;NO
10940 044636' ERRHRD 222,MUCAST ;YES, PRINT ERROR MESSAGE
10941 044636' 104456 TRAP C$ERRHRD
10942 044640' 000336 .WORD 222
10943 044642' 003470' .WORD MUCAST
10944 044644' 000000 .WORD 0
10945
10946 ;WRITE '1' TO CLEAR 'DNI'
10947
10948 044646' 004737 017362' 50$: JSR PC,CLRDN1 ;CLEAR DNI BIT
10949 044652' 103004 BCC 60$
10950 044654' ERRHRD 223,MUCAST,RACMG7 ;ERROR DNI DID NOT CLEAR!
10951 044654' 104456 TRAP C$ERRHRD
10952 044656' 000337 .WORD 223
10953 044660' 003470' .WORD MUCAST
10954 044662' 012670' .WORD RACMG7
10955 044664' 60$:
10956 044664' ENDSEG
10957 044664' 10001$:
10958 044664' 104405 TRAP C$ESEG
10959 044666'
10960 044666' L10157:
10961 044666' 104401 TRAP C$ETST
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 231  
CZUAAB.MAC 07-APR-83 17:03 TEST 38: CRC DATA PATTERN TEST

10962  
10963  
10964  
10965  
10966  
10967  
10968  
10969  
10970  
10971  
10972  
10973  
10974  
10975  
10976  
10977  
10978  
10979  
10980  
10981  
10982  
10983  
10984  
10985  
10986  
10987  
10988  
10989  
10990  
10991  
10992  
10993  
10994  
10995  
10996  
10997  
10998  
10999  
11000  
11001  
11002  
11003  
11004  
11005  
11006  
11007  
11008  
11009  
11010  
11011  
11012  
11013  
11014  
11015  
11016  
11017

044670'  
044670'

.SBTTL TEST 38: CRC DATA PATTERN TEST  
:\*\*\*\*\*8  
:THE LINK MODULE HAS HARDWARE TO GUARANTEE THAT DATAGRAMS HAVE NOT BEEN  
:CORRUPTED DURING TRANSMISSION AND RECEPTION. THE HARDWARE GENERATES A CRC  
:FOR DATAGRAMS TRANSMITTED AND VERIFIES THE CRC FOR DATAGRAMS RECEIVED.  
:THE CRC IS A 32 BIT NUMBER GENERATED BY DIVIDING THE DATAGRAM BIT STREAM BY A  
:CRC POLYNOMIAL. THE DIVISION RESULTS IN A UNIQUE NUMBER THAT CAN ONLY BE  
:REPRODUCED IN CRC CALCULATIONS IF THE BIT STREAM EXACTLY MATCHES THE ORIGINAL.  
:THE CRC IS CALCULATED DURING DATAGRAM TRANSMISSION AND IS APPENDED TO THE  
:PACKET. THE CRC IS TRANSMITTED AS PART OF THE PACKET. THE CRC IS AGAIN  
:CALCULATED WHEN THE DATAGRAM IS RECEIVED AND THE CALCULATED IS COMPARED TO  
:THE CRC TRANSMITTED. IF THE DATAGRAM HAS BEEN FAITHFULLY TRANSMITTED, THE  
:CRC'S SHOULD MATCH EXACTLY.  
:THE DEUNA CALCULATES THE CRC WITH DEDICATED CRC LOGIC. THE LOGIC IS EITHER  
:DEDICATED TO THE CALCULATION OF THE OUTGOING DATAGRAM OR THE CALCULATION OF  
:THE INCOMING DATAGRAM, BUT NOT BOTH.  
:THIS TEST WILL VERIFY THE OPERATION OF THE CRC CALCULATION CIRCUITRY.  
:MICROMODULE 'F' MICROTEST #1 WILL BE USED.  
:THE MICROCODE WILL TRANSMIT DATAGRAMS IN LOOPBACK MODE. THE CRC HARDWARE WILL  
:BE DEDICATED TO THE TRANSMITTER. WHEN THE DATAGRAM IS RECEIVED THE T-11 WILL  
:CALCULATE A CRC ON THE DATA RECEIVED (INCLUDING THE TRANSMITTED CRC).  
:THE RESULT OF THIS CALCULATION WILL BE A 32 BIT CONSTANT. THIS CONSTANT IS  
:THEN COMPARED TO WHAT WAS EXPECTED AND IF THEY DO NOT MATCH. AN ERROR IS  
:PLACED IN PCSR1.  
:PATTERNS WILL BE PASSED TO THE MICROCODE THROUGH THE PCBB. THE MICROCODE WILL  
:FILL THE TRANSMIT BUFFER WITH THIS PATTERN BEFORE EACH TRANSMISSION TAKES  
:PLACE.  
:THE PCBB WILL BE FORMATTED AS FOLLOWS:  
:PCBB+0:           +-----+  
:                  ! DATA PATTERN !  
:                  +-----+  
:TEST SEQUENCE:  
:          1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO  
:          2-PLACE A DATA PATTERN IN PCBB+0  
:          3-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE  
:          4-SELECT MICROTEST #1  
:          5-WAIT FOR 'DNI'  
:          6-CHECK PCSR1 FOR AN ERROR CONDITION  
:          7-WRITE '1' TO CLEAR 'DNI'  
:\*\*\*\*\*8  
BGNTST  
T38::  
:CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 232  
 CZUAAB.MAC 07-APR-83 17:03 TEST 38: CRC DATA PATTERN TEST

```

11018
11019 044670'          ; BGNSEG
11020 044670' 104404
11021 044672' 022737 000106 000326'      CMP    #'F,MICRO      ;HAS MICROCODE MODULE 'F' BEEN LOADED
11022 044700' 001004          BNE    5$              ;NO
11023 044702' 122777 000001 133430      CMPB   #INMON,@PCSR1  ;YES, IS THE MICROMONITOR ACTIVE?
11024 044710' 001440          BEQ    20$              ;YES SKIP LOADING THE MICROMODULE
11025 044712' 012737 000106 000326' 5$:  MOV    #'F,MICRO      ;GO LOAD MICRO MODULE 'F'
11026 044720' 004737 020340'          JSR    PC,LODMIC
11027 044724' 103002          BCC   10$              ;OK
11028 044726'          ESCAPE TST
11029 044726' 104410
11030 044730' 000276          TRAP   C$BSEG
11031 044732' 012737 000176 000332' 10$:  MOV    #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
11032 044740' 004737 017316'          JSR    PC,CHKDNI
11033 044744' 103022          BCC   20$              ;OK
11034 044746' 012737 001000' 000310'      MOV    #SDNI,BITNAM
11035 044754' 012737 001277' 000312'      MOV    #SSET,BITSTA
11036 044762' 012737 001342' 000314'      MOV    #SAFTER,PWHEN
11037 044770' 012737 001357' 000316'      MOV    #SGTCMD,PCOMND
11038 044776'          ERRHRD 224,CRCDAT,MSG1
11039 044776' 104456          TRAP   C$ERHRD
11040 045000' 000340          .WORD 224
11041 045002' 003526'          .WORD CRCDAT
11042 045004' 012716'          .WORD MSG1
11043 045006'          ESCAPE TST
11044 045006' 104410          TRAP   C$ESCAPE
11045 045010' 000216          .WORD L10160-.
11046 045012' 004737 017362'          20$:  JSR    PC,CLR DNI      ;CLEAR DNI BIT
11047 045016' 103006          BCC   25$
11048 045020'          ERRHRD 225,CRCDAT,RACMG7 ;DNI DID NOT CLEAR!
11049 045020' 104456          TRAP   C$ERHRD
11050 045022' 000341          .WORD 225
11051 045024' 003526'          .WORD CRCDAT
11052 045026' 012670'          .WORD RACMG7
11053 045030'          ESCAPE TST
11054 045030' 104410          TRAP   C$ESCAPE
11055 045032' 000174          .WORD L10160-.
11056 045034'          25$:
11057 045034'          ENDSEG
11058 045034'          10000$:
11059 045034' 104405          TRAP   C$ESEG
11060
11061          ;SELECT A DATA PATTERN FROM A TABLE OF PREDEFINED DATA PATTERNS
11062          ;AND LOAD IT INTO PCBB+0 FOR THE MICROCODE
11063
11064 045036' 012701 000520'      MOV    #PATERN,R1      ;GET ADDRESS OF DATA PATTERN TABLE
11065 045042' 012705 000005      MOV    #5,R5            ;# OF DATA PATTERNS
11066 045046' 012137 000606'      27$:  MOV    (R1)+,PCBB    ;LOAD PCBB WITH A DATA PATTERN
11067 045052'          BGNSEG
11068 045052' 104404          TRAP   C$BSEG
11069
11070          ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
11071
11072 045054' 004737 020060'      JSR    PC,CHKMON      ;WAIT FOR MICROMONITOR
11073 045060' 103006          BCC   30$              ;OK

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 233  
 CZUAAB.MAC 07-APR-83 17:03 TEST 38: CRC DATA PATTERN TEST

```

11074 045062'                ERRHRD  227,CRCDAT,MSG46                ;PRINT ERROR
11075 045062' 104456
11076 045064' 000343                TRAP  CSERHRD
11077 045066' 003526'                .WORD  227
11078 045070' 016666'                .WORD  CRCDAT
11079 045072'                .WORD  MSG46
11080 045072' 104410                ESCAPE  TST                ;LEAVE TEST
11081 045074' 000132                TRAP  CSERHRD
11082 045076'                .WORD  L10160-
11083
11084                ;
11085                ;EXECUTE MICROTEST #1 BY LOADING PCSRO COMMAND FIELD WITH A 1
11086                ;WAIT FOR DNI
11087 045076' 012777 000001 133232  MOV  #1,@PCSRO                ;TELL T11 TO EXECUTE MICROTEST #1
11088 045104' 012737 000176 000332'  MOV  #2*SECOND.METER        ;WAIT FOR DNI
11089 045112' 004737 017316'  JSR  PC,CHKDNI
11090 045116' 103021                BCC  40$
11091 045120' 004737 020132'  JSR  PC,CHKINT                ;SEE IF ANY ERROR INTERRUPTS OCCURRED
11092 045124' 103006                BCC  35$                      ;NO, OK
11093 045126'                ERRHRD  228,CRCDAT,MSG44        ;PRINT ERROR MESSAGE
11094 045126' 104456                TRAP  CSERHRD
11095 045130' 000344                .WORD  228
11096 045132' 003526'                .WORD  CRCDAT
11097 045134' 016442'                .WORD  MSG44
11098 045136'                ESCAPE  TST
11099 045136' 104410                TRAP  CSERHRD
11100 045140' 000066                .WORD  L10160-
11101 045142' 012702 000005 35$:  MOV  #5,R2                ;MICROTEST #1 IS HUNG
11102 045146'                ERRHRD  229,CRCDAT,MSG12
11103 045146' 104456                TRAP  CSERHRD
11104 045150' 000345                .WORD  229
11105 045152' 003526'                .WORD  CRCDAT
11106 045154' 013466'                .WORD  MSG12
11107 045156'                ESCAPE  TST
11108 045156' 104410                TRAP  CSERHRD
11109 045160' 000046                .WORD  L10160-
11110
11111                ;
11112                ;CHECK PCSR1 TO SEE IF THE MICROCODE DETECTED AN ERROR IN THE CRC CALCULATION
11113 045162' 122777 000003 133150 40$:  CMPB  #INERR,@PCSR1          ;DID AN ERROR OCCUR?
11114 045170' 001004                BNE  50$                      ;NO
11115 045172'                ERRHRD  230,CRCDAT          ;PRINT ERROR MESSAGE
11116 045172' 104456                TRAP  CSERHRD
11117 045174' 000346                .WORD  230
11118 045176' 003526'                .WORD  CRCDAT
11119 045200' 000000                .WORD  0
11120
11121                ;
11122                ;WRITE '1' TO CLEAR 'DNI'
11123 045202' 004737 017362' 50$:  JSR  PC,CLRDNI                ;CLEAR DNI BIT
11124 045206' 103004                BCC  55$
11125 045210'                ERRHRD  231,CRCDAT,RACMG7    ;ERROR DNI DID NOT CLEAR!
11126 045210' 104456                TRAP  CSERHRD
11127 045212' 000347                .WORD  231
11128 045214' 003526'                .WORD  CRCDAT
11129 045216' 012670'                .WORD  RACMG7

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 234  
CZUAAB.MAC 07-APR-83 17:03 TEST 38: CRC DATA PATTERN TEST

11130  
11131 045220'  
11132 045220'  
11133 045220'  
11134 045220' 104405  
11135  
11136  
11137  
11138 045222' 005305  
11139 045224' 001310  
11140 045226'  
11141 045226'  
11142 045226' 104401

55\$:  
ENDSEG

10001\$:  
TRAP C\$ESEG

:  
:REPEAT THE TEST EACH TIME WITH A NEW DATA PATTERN  
:

DEC R5  
BNE 27\$

:HAVE WE TESTED WITH ALL DATA PATTERNS?  
:NOT YET

ENDTST

L10160:  
TRAP C\$ETST

11143  
11144  
11145  
11146  
11147  
11148  
11149  
11150  
11151  
11152  
11153  
11154  
11155  
11156  
11157  
11158  
11159  
11160  
11161  
11162  
11163  
11164  
11165  
11166  
11167  
11168  
11169  
11170  
11171  
11172  
11173  
11174  
11175  
11176  
11177  
11178  
11179  
11180  
11181  
11182  
11183  
11184  
11185  
11186  
11187  
11188  
11189  
11190  
11191  
11192  
11193  
11194  
11195  
11196  
11197  
11198

.SBTTL TEST 39: CRC ERROR TEST  
:\*\*\*\*\*  
:THIS TEST WILL VERIFY THAT THE LINK CRC CIRCUITRY CAN DETECT A BAD CRC.  
:MICROMODULE 'F' MICROTEST #2 WILL BE USED. THE MICROCODE WILL TRANSMIT  
:DATAGRAMS IN LOOPBACK MODE. EACH DATAGRAM WILL HAVE AN ERRONEOUS CRC  
:APPENDED TO THE DATA FIELD. THE DELMA CRC LOGIC WILL BE SETUP SUCH THAT  
:THE CRC LOGIC WILL BE DEDICATED TO THE RECEIVER. THIS IS EXPECTED TO CAUSE  
:A CRC ERROR.  
:THE DATA FIELDS OF EACH DATAGRAM WILL CONSIST OF PATTERNS. THE PATTERNS  
:WILL BE PASSED TO THE MICROCODE VIA THE PCBB.  
:AFTER THE RECEIVER INTERRUPT THE MICROCODE WILL PASS THE RECEIVER STATUS WORD  
:0 BACK VIA PCBB+2. THE CRC BIT IN THIS WORD IS CHECKED TO SEE IF IT IS SET.  
:THE PCBB IS FORMATTED AS FOLLOWS:  
:PCBB+0:           ↑-----↑  
:                  ! DATA PATTERN !  
:                  ↑-----↑  
:PCBB+2:           ! RECEIVER STATUS WORD !  
:                  ↑-----↑  
:TEST SEQUENCE:  
:      1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO  
:      2-PLACE A DATA PATTERN IN PCBB+0  
:      3-WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE  
:      4-SELECT MICROTEST #2  
:      5-WAIT FOR 'DNI'  
:      6-VERIFY CRC BIT AND ERROR SUMMARY BIT SET IN PCBB+2  
:      7-WRITE '1' TO CLEAR 'DNI'  
:\*\*\*\*\*

045230'  
045230'

BGNTST  
T39::  
:CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

045230'  
045230' 104404  
045232' 022737  
045240' 001004  
045242' 122777  
045250' 001440  
045252' 012737  
045260' 004737  
045264' 103002  
045266'             
045266' 104410  
045270' 000326

```
BGNSEG
TRAP CSBSEG
;HAS MICROCODE MODULE 'F' BEEN LOADED
;NO
;YES. IS THE MICROMONITOR ACTIVE?
;YES SKIP LOADING THE MICROMODULE
;GO LOAD MICRO MODULE 'F'
;OK
TRAP CSBSEG
.WORD L10161-
```

000106 000326'   CMP   #F,MICRO  
                  BNE   5\$  
000001 133070   CMPB  #INMON,@PCSR1  
                  BEQ   20\$  
000106 000326' 5\$:  MOV   #F,MICRO  
                  JSR   PC,LODMIC  
                  BCC   10\$  
                  ESCAPE TST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 236  
 CZUAAB.MAC 07-APR-83 17:03 TEST 39: CRC ERROR TEST

```

11199 045272' 012737 000176 000332' 10$: MOV #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
11200 045300' 004737 017316' JSR PC,CHKDNI
11201 045304' 103022 BCC 20$ ;OK
11202 045306' 012737 001000' 000310' MOV #DNI,BITNAM
11203 045314' 012737 001277' 000312' MOV #SNSET,BITSTA
11204 045322' 012737 001342' 000314' MOV #SAFTER,PWHEN
11205 045330' 012737 001357' 000316' MOV #SGTCMD,PCOMND
11206 045336' ERRHRD 232,CRCERR,MSG1
11207 045336' 104456 TRAP CSERHRD
11208 045340' 000350 .WORD 232
11209 045342' 003563' .WORD CRCERR
11210 045344' 012716' .WORD MSG1
11211 045346' ESCAPE TST
11212 045346' 104410 TRAP C$ESCAPE
11213 045350' 000246 .WORD L10161-.
11214 045352' 004737 017362' 20$: JSR PC,CLRDNI ;CLEAR DNI BIT
11215 045356' 103006 BCC 25$
11216 045360' ERRHRD 233,CRCERR,RACMG7 ;DNI DID NOT CLEAR!
11217 045360' 104456 TRAP CSERHRD
11218 045362' 000351 .WORD 233
11219 045364' 003563' .WORD CRCERR
11220 045366' 012670' .WORD RACMG7
11221 045370' ESCAPE TST
11222 045370' 104410 TRAP C$ESCAPE
11223 045372' 000224 .WORD L10161-.
11224 045374' 25$:
11225 045374' ENDSEG
11226 045374'
11227 045374' 104405 10000$: TRAP C$SEEG
11228 :
11229 :CLEAR PCBB+2, GET A DATA PATTERN FROM A LIST OF PRESELECTED DATA PATTERNS
11230 :AND PLACE IT IN PCBB+0 FOR THE MICROCODE
11231 :
11232 045376' 005037 000610' CLR PCBB+2 ;HERE IS WHERE MICROCODE WILL PUT...
11233 :STATUS WORD
11234 045402' 012701 000520' MOV #PATERN,R1 ;GET ADDRESS OF DATA PATTERN TABLE
11235 045406' 012705 000005 MOV #5,R5 ;# OF DATA PATTERNS
11236 045412' 012137 000606' 27$: MOV (R1)+,PCBB ;LOAD PCBB WITH A DATA PATTERN
11237 :
11238 :WAIT FOR THE MICROMONITOR TO BECOME READY
11239 :
11240 : BGNSEG
11241 045416' 104404 TRAP C$BSEG
11242 045420' 004737 020060' JSR PC,CHKMON ;WAIT FOR MICROMONITOR
11243 045424' 103006 BCC 30$ ;OK
11244 045426' ERRHRD 234,CRCERR,MSG46 ;PRINT ERROR
11245 045426' 104456 TRAP CSERHRD
11246 045430' 000352 .WORD 234
11247 045432' 003563' .WORD CRCERR
11248 045434' 016666' .WORD MSG46
11249 045436' ESCAPE TST ;LEAVE TEST
11250 045436' 104410 TRAP C$ESCAPE
11251 045440' 000156 .WORD L10161-.
11252 045442' 30$:
11253 :
11254 :EXECUTE MICROTEST #2 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 2

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 237  
 CZUAAB.MAC 07-APR-83 17:03 TEST 39: CRC ERROR TEST

```

11255          ;WAIT FOR 'DNI'
11256          ;
11257 045442' 012777 000002 132666      MOV      #2,@PCSR0      ;TELL T11 TO EXECUTE MICROTEST #2
11258 045450' 012737 000176 000332'    MOV      *?+SECOND,METER ;WAIT FOR DNI
11259 045456' 004737 017316'    JSR      PC,LHKDNI
11260 045462' 103021          BCC      40$
11261 045464' 004737 020132'    JSR      PC,CHI.INT      ;SEE IF ANY ERROR INTERRUPTS OCCURRED
11262 045470' 103006          BCC      35$            ;NO, OK
11263 045472'          ERRHRD 235,CRCERR,MSG44 ;PRINT ERROR MESSAGE
11264 045472' 104456          TRAP      C$ERHRD
11265 045474' 000353          .WORD    235
11266 045476' 003563'        .WORD    CRCERR
11267 045500' 016442'        .WORD    MSG44
11268 045502'          ESCAPE  TST
11269 045502' 104410          TRAP      C$ESCAPE
11270 045504' 000112          .WORD    L10161-
11271 045506' 012702 000002    35$:  MOV      #2,R2      ;MICROTEST #2 IS HUNG
11272 045512'          ERRHRD 236,CRCERR,MSG12
11273 045512' 104456          TRAP      C$ERHRD
11274 045514' 000354          .WORD    236
11275 045516' 003563'        .WORD    CRCERR
11276 045520' 013466'        .WORD    MSG12
11277 045522'          ESCAPE  TST
11278 045522' 104410          TRAP      C$ESCAPE
11279 045524' 000072          .WORD    L10161-
11280          ;
11281          ;OK, TEST IS COMPLETED, NOW CHECK PCBB+2. PCBB+2 CONTAINS THE RECEIVER STATUS
11282          ;WORD, IT SHOULD HAVE THE CRC ERROR BIT SET AND THE ERROR SUMMARY BIT SET
11283          ;
11284 045526' 013704 000610'    40$:  MOV      PCBB+2,R4   ;THIS IS THE RECEIVER STATUS WORD
11285 045532' 012703 004000'    MOV      #BIT11,R3      ;BIT CRC SHOULD BE SET
11286 045536' 030304          BIT      R3,R4          ;IS CRC BIT SET?
11287 045540' 001004          BNE      45$            ;YES, OK
11288 045542'          ERRHRD 237,CRCERR,MSG27 ;NO, PRINT ERROR MESSAGE
11289 045542' 104456          TRAP      C$ERHRD
11290 045544' 000355          .WORD    237
11291 045546' 003563'        .WORD    CRCERR
11292 045550' 015046'        .WORD    MSG27
11293 045552' 012703 040000'    45$:  MOV      #BIT14,R3   ;ERROR SUMMARY BIT SHOULD BE SET
11294 045556' 030304          BIT      R3,R4          ;IS ERROR SUMMARY BIT SET?
11295 045560' 001004          BNE      50$            ;YES, GOOD
11296 045562'          ERRHRD 238,CRCERR,MSG28 ;NO, PRINT ERROR MESSAGE
11297 045562' 104456          TRAP      C$ERHRD
11298 045564' 000356          .WORD    238
11299 045566' 003563'        .WORD    CRCERR
11300 045570' 015114'        .WORD    MSG28
11301          ;
11302          ;WRITE '1' TO CLEAR 'DNI'
11303          ;
11304 045572' 004737 017362'    50$:  JSR      PC,CLRDN1    ;CLEAR DNI BIT
11305 045576' 103004          BCC      55$
11306 045600'          ERRHRD 239,CRCERR,RACMG7 ;ERROR DNI DID NOT CLEAR!
11307 045600' 104456          TRAP      C$ERHRD
11308 045602' 000357          .WORD    239
11309 045604' 003563'        .WORD    CRCERR
11310 045606' 012670'        .WORD    RACMG7

```



11311 045610'  
11312 045610'  
11313 045610'  
11314 045610' 104405  
11315  
11316  
11317  
11318 045612' 005305  
11319 045614' 001276  
11320  
11321 045616'  
11322 045616'  
11323 045616' 104401

558:  
ENDSEG

:  
:REPEAT TEST WITH ALL DATA PATTERNS  
:

DEC R5  
BNE 278

ENDTST

100018: TRAP CSESEG

:HAVE WE TESTED WITH ALL DATA PATTERNS?  
:NOT YET

L10161: TRAP CSETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 239  
 CZUAAB.MAC 07-APR-83 17:03 TEST 40: CRC PATTERN LENGTH TEST

11324  
11325  
11326  
11327  
11328  
11329  
11330  
11331  
11332  
11333  
11334  
11335  
11336  
11337  
11338  
11339  
11340  
11341  
11342  
11343  
11344  
11345  
11346  
11347  
11348  
11349  
11350  
11351  
11352  
11353  
11354  
11355  
11356  
11357  
11358  
11359  
11360  
11361  
11362  
11363  
11364  
11365  
11366  
11367  
11368  
11369  
11370  
11371  
11372  
11373  
11374  
11375  
11376  
11377  
11378  
11379

045620'  
045620'  
  
  
  
  
045620'  
045620' 104404  
045622' 022737  
045630' 001004

000106 000326'

```

.SBTTL TEST 40: CRC PATTERN LENGTH TEST
:*****
:THIS TEST WILL VERIFY THAT THE RECEIVE CRC HARDWARE CAN CALCULATE CRC FOR
:DATAGRAMS OF VARYING LENGTHS.
:
:DATAGRAMS WILL BE TRANSMITTED FOR THE TRANSMIT BUFFER TO THE RECEIVE BUFFER
:IN LOOPBACK MODE. THE TRANSMIT CRC WILL BE DISABLED WHICH WILL ASSIGN THE
:CRC LOGIC TO CALCULATION OF INCOMING DATAGRAMS. THE CRC FOR TRANSMIT DATAGRAMS
:WILL BE CALCULATED BY THE MICROCODE. IT IS EXPECTED THAT THE CRC LOGIC WILL
:VERIFY THE CRC APPENDED TO THE DATAGRAM AS IT IS BEING RECEIVED.
:
:PATTERNS WILL BE USED TO FILL THE DATAGRAM DATA FIELD. THE PATTERNS WILL BE
:PASSED TO THE MICROCODE THROUGH THE PCBB ALONG WITH THE BYTE COUNT TO BE USED.
:
:AFTER THE RECEPTION OF THE DATAGRAM THE RECEIVER STATUS WORD WILL BE PASSED
:BACK VIA THE PCBB SO IT CAN BE CHECKED
:
:THE PCBB IS FORMATTED AS FOLLOWS:
:
:PCBB+0:          +-----+
:                | DATA PATTERN |
:                +-----+
:PCBB+2:          |   BYTE COUNT   |
:                +-----+
:PCBB+4:          | RECEIVE STATUS WORD |
:                +-----+
:
:TEST SEQUENCE:
:  1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO
:  2-PLACE A PATTERN IN PCBB+0
:  3-PREPARE A MINIMUM BYTE COUNT
:  4-PLACE BYTE COUNT IN PCBB+4
:  5-WAIT FOR MICROMONITOR TO ENTER 'INMON' STATE
:  6-SELECT MICROTTEST #3
:  7-WAIT FOR 'DNI'
:  8-VERIFY NO ERRORS IN PCBB+4
:  9-WRITE '1' TO CLEAR 'DA:'
: 10-MULTIPLY BYTE COUNT BY 2
: 11-REPEAT STEPS 4-10 UNTIL MAXIMUM BYTE COUNT IS REACHED
: 12-REPEAT STEPS 2-11 WITH ALL DATA PATTERNS
:*****
BGNTST
T40::
:
:CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
:
BGNSEG
          TRAP  CSBSEG
CMP      #'F,MICRO  :HAS MICROCODE MODULE 'F' BEEN LOADED
BNE      58        :NO
  
```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 240  
 CZUAAB.MAC 07-APR-83 17:03 TEST 40: CRC PATTERN LENGTH TEST

```

11380 045632' 122777 000001 132500      CMPB   #INMON,BPCSR1      ;YES, IS THE MICROMONITOR ACTIVE?
11381 045640' 001440                      BEQ    20$                ;YES SKIP LOADING THE MICROMODULE
11382 045642' 012737 000106 000326' 5$:   MOV    #'F,MICRO        ;GO LOAD MICRO MODULE 'F'
11383 045650' 004737 020340'                      JSR    PC,LODMIC
11384 045654' 103002                      BCC    10$                ;OK
11385 045656'                      ESCAPE TST
11386 045656' 104410                      TRAP   CS_ESCAPE         ;TRAP
11387 045660' 000320                      .WORD L10162-           ;.WORD
11388 045662' 012737 000176 000332' 10$:  MOV    #2*SECOND,METER  ;WAIT FOR THE MICROMONITOR
11389 045670' 004737 017316'                      JSR    PC,CHKDNI
11390 045674' 103022                      BCC    20$                ;OK
11391 045676' 012737 001000' 000310'      MOV    #SDNI,BITNAM
11392 045704' 012737 001277' 000312'      MOV    #SNSET,BITSTA
11393 045712' 012737 001342' 000314'      MOV    #SAFTER,PWEN
11394 045720' 012737 001357' 000316'      MOV    #SGTCMD,PCOMND
11395 045726'                      ERRHRD 240,CRCPAT,MSG1
11396 045726' 104456                      TRAP   CS_ERRHRD        ;TRAP
11397 045730' 000360                      .WORD 240                ;.WORD
11398 045732' 003611'                      .WORD CRCPAT             ;.WORD
11399 045734' 012716'                      .WORD MSG1               ;.WORD
11400 045736'                      ESCAPE TST
11401 045736' 104410                      TRAP   CS_ESCAPE         ;TRAP
11402 045740' 000240                      .WORD L10162-           ;.WORD
11403 045742' 004737 017362' 20$:   JSR    PC,CLRDN1        ;CLEAR DNI BIT
11404 045746' 103006                      BCC    25$
11405 045750'                      ERRHRD 241,CRCPAT,RACMG7 ;DNI DID NOT CLEAR!
11406 045750' 104456                      TRAP   CS_ERRHRD        ;TRAP
11407 045752' 000361                      .WORD 241                ;.WORD
11408 045754' 003611'                      .WORD CRCPAT             ;.WORD
11409 045756' 012670'                      .WORD RACMG7            ;.WORD
11410 045760'                      ESCAPE TST
11411 045760' 104410                      TRAP   CS_ESCAPE         ;TRAP
11412 045762' 000216                      .WORD L10162-           ;.WORD
11413 045764'                      25$:
11414 045764'                      ENDSEG
11415 045764'                      10000$:
11416 045764' 104405                      TRAP   CS_SEG           ;TRAP
11417
11418 ;POINT TO LIST OF DATA PATTERNS
11419 ;
11420 045766' 012701 000520'      MOV    #PATERN,R1      ;GET ADDRESS OF DATA PATTERN TABLE
11421 045772' 012705 000005      MOV    #5,R5           ;# OF DATA PATTERNS
11422 ;
11423 ;GET A DATA PATTERN FROM THE LIST AND PLACE IT IN PCBB+0
11424 ;
11425 045776'      27$:
11426 045776' 012137 000606'      MOV    (R1)+,PCBB      ;LOAD PCBB WITH A DATA PATTERN
11427 046002'      BGNSEG
11428 046002' 104404                      TRAP   CS_BSEG         ;TRAP
11429 ;
11430 ;GENERATE A MINIMUM BYTE COUNT
11431 ;
11432 046004' 012703 000100      MOV    #BIT6,R3        ;R3 WILL BE BYTE COUNT
11433 046010'      28$:
11434 ;
11435 ;LOAD BYTE COUNT INTO THE PCBB+4

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 241  
 CZUAAB.MAC 07-APR-83 17:03 TEST 40: CRC PATTERN LENGTH TEST

```

11436
11437 046010'          : BGNSEG
11438 046010' 104404
11439 046012' 010337 000610'          TRAP      C8BSEG
11440          MOV      R3,PCBB+2          ;LOAD BYTE COUNT INTO PCBB
11441          :
11442          :WAIT FOR THE MICROCODE TO ENTER THE 'INPMON' STATE
11443          :EXECUTE MICROTEST #3 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 3
11444          :WAIT FOR 'DNI'
11445          :
11445 046016' 004737 020060'          :
11446 046022' 103006          JSR      PC,CHKMON          ;WAIT FOR MICROMONITOR
11447 046024'          BCC      30$          ;OK
11448 046024' 104456          ERRHRD   242,CRCPAT,MSG46      ;PRINT ERROR
11449 046026' 000362          TRAP      C8ERHRD
11450 046030' 003611'          .WORD    242
11451 046032' 016666'          .WORD    CRCPAT
11452 046034'          .WORD    MSG46
11453 046034' 104410          ESCAPE   TST          ;LEAVE TEST
11454 046036' 000142          TRAP      C8ESCAPE
11455 046040' 012777 000003 132270 30$: MOV      #3,@PCSRO          ;TELL T11 TO EXECUTE MICROTEST #3
11456 046046' 012737 000176 000332' MOV      #2+SECOND,METER      ;WAIT FOR DNI
11457 046054' 004737 017316' JSR      PC,CHKDNI
11458 046060' 103021          BCC      40$
11459 046062' 004737 020132' JSR      PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
11460 046066' 103006          BCC      35$          ;NO, OK
11461 046070'          ERRHRD   243,CRCPAT,MSG44      ;PRINT ERROR MESSAGE
11462 046070' 104456          TRAP      C8ERHRD
11463 046072' 000363          .WORD    243
11464 046074' 003611'          .WORD    CRCPAT
11465 046076' 016442'          .WORD    MSG44
11466 046100'          ESCAPE   TST
11467 046100' 104410          TRAP      C8ESCAPE
11468 046102' 000076          .WORD    L10162-
11469 046104' 012702 000003 35$: MOV      #3,R2          ;MICROTEST #3 IS HUNG
11470 046110'          ERRHRD   244,CRCPAT,MSG12
11471 046110' 104456          TRAP      C8ERHRD
11472 046112' 000364          .WORD    244
11473 046114' 003611'          .WORD    CRCPAT
11474 046116' 013466'          .WORD    MSG12
11475 046120'          ESCAPE   TST
11476 046120' 104410          TRAP      C8ESCAPE
11477 046122' 000056          .WORD    L10162-
11478          :
11479          :CHECK THE RECEIVER STATUS WORD, WHICH THE MICROCODE PLACED IN PCBB+4, FOR ANY
11480          :ERRORS
11481          :
11482 046124' 005737 000612' 40$: TST      PCBB+4          ;ANYTHING SET IN RECEIVER STATUS WORD 0
11483 046130' 001404          BEQ      50$          ;NO, OK
11484 046132'          ERRHRD   245,CRCPAT,MSG29      ;PRINT ERROR MESSAGE
11485 046132' 104456          TRAP      C8ERHRD
11486 046134' 000365          .WORD    245
11487 046136' 003611'          .WORD    CRCPAT
11488 046140' 015162'          .WORD    MSG29
11489          :
11490          :WRITE '1' TO CLEAR 'DNI'
11491          :

```

6SHARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 242  
 CZUAAB.MAC 07-APR-83 17:03 TEST 40: CRC PATTERN LENGTH TEST

```

11492 046142' 004737 017362'      508:   JSR   PC,CLRDNI      ;CLEAR DNI BIT
11493 046146' 103004                BCC   55$
11494 046150'                        ERRHRD 246,CRCPAT,RACMG7 ;ERROR DNI DID NOT CLEAR!
11495 046150' 104456                TRAP  CSEHRD
11496 046152' 000366                .WORD 246
11497 046154' 003611'                .WORD CRCPAT
11498 046156' 012670'                .WORD RACMG7
11499 046160'
11500 046160'
11501 046160'
11502 046160' 104405                10002$: TRAP  CSESEG
11503
11504 ;GENERATE A NEW BYTE COUNT BY SHIFTING IT OVER ONE PLACE TO THE LEFT.
11505 ;THIS EFFECTIVELY MULTIPLIES THE NUMBER BY TWO.
11506 ;CHECK TO SEE IF THE NEW NUMBER IS NOT TOO LARGE.
11507 ;REPEAT THE TEST WITH THE NEW BYTE COUNT
11508
11509 046162' 006303
11510 046164' 020327 002000          ASL   R3                ;UP THE BYTE COUNT TO NEXT BIT POSITION
11511                                     CMP   R3,#BIT10        ;HAVE WE CHECKED ALL BIT POSITIONS IN
11512 046170' 001307                                     BNE   28$              ;THE BYTE COUNT REGISTER?
11513 046172'                                     ENDSEG                ;NOT YET
11514 046172'
11515 046172' 104405                10001$: TRAP  CSESEG
11516
11517 ;GET A NEW DATA PATTERN FROM THE LIST OF PRESELECTED DATA PATTERNS
11518 ;REPEAT TEST WITH NEW DATA PATTERN
11519
11520 046174' 005305
11521 046176' 001277                DEC   R5                ;HAVE WE TESTED WITH ALL DATA PATTERNS?
11522                                     BNE   27$              ;NOT YET
11523 046200'
11524 046200'
11525 046200' 104401                ENDTST                L10162: TRAP  CSETST
    
```

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 243  
CZUAAB.MAC 07-APR-83 17:03

TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

.SBTTL TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

11526  
11527  
11528  
11529  
11530  
11531  
11532  
11533  
11534  
11535  
11536  
11537  
11538  
11539  
11540  
11541  
11542  
11543  
11544  
11545  
11546  
11547  
11548  
11549  
11550  
11551  
11552  
11553  
11554  
11555  
11556  
11557  
11558  
11559  
11560  
11561  
11562  
11563  
11564  
11565  
11566  
11567  
11568  
11569  
11570  
11571  
11572  
11573  
11574  
11575  
11576  
11577  
11578  
11579  
11580  
11581

\*\*\*\*\*

: THIS TEST WILL CHECK THE ABILITY OF THE RECEIVE STATE MACHINE TO REJECT A DATAGRAM OF LESS THAN 64 BYTES AND TO RECOVER THE RECEIVER BUFFER.

: THIS TEST WILL USE MICROMODULE 'F' MICROTEST #4.  
: EACH TRIAL WILL CONSIST OF TWO DATAGRAM TRANSMISSIONS IN LOOPBACK MODE. EACH TRANSMISSION WILL LOOPBACK A DATAGRAM FILLED WITH UNIQUE DATA. THE FIRST DATAGRAM WILL BE A RUNT OF LESS THAN 64 BYTES. THE SECOND WILL BE A DATAGRAM OF LEGAL SIZE.

: EACH TRIAL WILL START WITH THE LINK BUFFER POINTER RESET TO THE FIRST LINK BUFFER. THE RUNT WILL BE TRANSMITTED, THEN THE VALID DATAGRAM. IF THE BUFFER RECOVERY IS WORKING CORRECTLY, THE SECOND DATAGRAM IS EXPECTED TO BE WRITTEN INTO THE SAME LINK MEMORY BUFFER AS WAS THE RUNT.

: THIS TEST WILL BE REPEATED WITH VARIOUS RUNT PACKET SIZES.

: THE BYTE COUNT FOR THE RUNT PACKET TRANSMISSION WILL BE PASSED VIA THE PCBB. AFTER THE TWO TRANSMISSIONS, THE MICROCODE WILL PASS BACK THE CONTENTS OF THE BUFFER DONE FIFO, AND THE CONTENTS OF THE FIRST DATA WORD OF THE RECEIVER BUFFER.

: THE PCBB WILL BE FORMATTED AS FOLLOWS:

```

:PCBB+0:  +-----+
:          | RUNT BYTE COUNT |
:          +-----+
:PCBB+2:  +-----+
:          | BUFFER DONE FIFO CONTENTS |
:          +-----+
:PCBB+4:  +-----+
:          | FIRST DATA WORD OF BUFFER |
:          +-----+

```

: TEST SEQUENCE:

- 1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO
- 2-PLACE A RUNT BYTE COUNT IN PCBB+0
- 3-CLEAR PCBB+2,+4
- 4-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
- 5-SELECT MICROTEST #4
- 6-WAIT FOR 'DNI'
- 7-CHECK PCSR1 FOR AN ERROR CONDITION (RECEIVER INTERRUPT OCCURRED ON RUNT PACKET RECEPTION)
- 8-CHECK PCBB+2 FOR CORRECT BUFFER DONE ADDRESS
- 9-CHECK PCBB+4 FOR CORRECT DATA PATTERN
- 10-WRITE '1' TO CLEAR 'DNI'
- 11-REPEAT STEPS 2-10 WITH A NEW RUNT BYTE COUNT

\*\*\*\*\*

BGNTST

T41::

046202'  
046202'

: CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.

65HARDWARE TESTS MACY1: 30A(1052) 07-APR-83 17:13 PAGE 244  
 CZUAAB.MAC 07-APR-83 17:03 TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

```

11582 ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
11583 ;
11584 046202' BGNSEG
11585 046202' 104404
11586 046204' 022737 000106 000326' CMP #'F,MICRO ;HAS MICROCODE MODULE 'F' BEEN LOADED
11587 046212' 001004 BNE 5$ ;NO
11588 046214' 122777 000001 132116 CMPB #INMON,@PCSR1 ;YES, IS THE MICROMONITOR ACTIVE?
11589 046222' 001440 BEQ 20$ ;YES SKIP LOADING THE MICROMODULE
11590 046224' 012737 000106 000326' 5$: MOV #'F,MICRO ;GO LOAD MICRO MODULE 'F'
11591 046232' 004737 020340' JSR PC,LODMIC
11592 046236' 103002 BCC 10$ ;OK
11593 046240'
11594 046240' 104410
11595 046242' 000370 TRAP C$ESCAPE
11596 046244' 012737 000176 000332' 10$: MOV #2*SECOND,METER .WORD L10163-.
11597 046252' 004737 017316' JSR PC,CHKDNI ;WAIT FOR THE MICROMONITOR
11598 046256' 103022 BCC 20$ ;OK
11599 046260' 012737 001000' 000310' MOV #SDNI,BITNAM
11600 046266' 012737 001277' 000312' MOV #SNSET,BITSTA
11601 046274' 012737 001342' 000314' MOV #SAFTER,PWHEN
11602 046302' 012737 001357' 000316' MOV #SGTCMD,PCOMND
11603 046310' ERRHRD 247,RBRRUN,MSG1
11604 046310' 104456 TRAP C$ERHRD
11605 046312' 000367 .WORD 247
11606 046314' 003650' .WORD RBRRUN
11607 046316' 012716' .WORD MSG1
11608 046320'
11609 046320' 104410
11610 046322' 000310 TRAP C$ESCAPE
11611 046324' 004737 017362' 20$: JSR PC,CLRDN1 .WORD L10163-.
11612 046330' 103006 BCC 25$ ;CLEAR DNI BIT
11613 046332' ERRHRD 248,RBRRUN,RACMG7 ;DNI DID NOT CLEAR!
11614 046332' 104456 TRAP C$ERHRD
11615 046334' 000370 .WORD 248
11616 046336' 003650' .WORD RBRRUN
11617 046340' 012670' .WORD RACMG7
11618 046342'
11619 046342' 104410
11620 046344' 000266 ESCAPE TST TRAP C$ESCAPE
11621 046346' 25$: .WORD L10163-.
11622 046346' ENDSEG
11623 046346'
11624 046346' 104405 10000$: TRAP C$SEEG
11625 ;
11626 ;R1 WILL CONTAIN THE RUNT BYTE COUNT TO BE USED BY THE MICROCODE
11627 ;
11628 046350' 012701 000001 MOV #1,R1 ;BEGIN WITH BYTE COUNT OF 1
11629 046354' 30$:
11630 ;
11631 ;PASS THE BYTE COUNT TO MICROCODE THROUGH PCBB+, CLEAR PCBB+2 AND PCBB+4
11632 ;
11633 046354' BGNSEG
11634 046354' 104404 TRAP C$BSEEG
11635 046356' 010137 000606' MOV R1,PCBB ;PASS BYTE COUNT TO MICROCODE
11636 046362' 005037 000610' CLR PCBB+2 ;HERE IS WHERE THE MICRO WILL PUT THE
11637 ;'DONE' RECEIVE BUFFER FIFO ADDRESS

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 245  
 CZUAAB.MAC 07-APR-83 17:03 TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

```

11638 046366' 005037 000612' CLR PCBB+4 ;HERE IS WHERE THE MICRO WILL PUT THE
11639 ;FIRST WORD OF DATA FROM THE RECEIVE BUFFER
11640
11641 ;WAIT FOR THE MICROCODE TO ENTER THE 'IMON' STATE, START MICROTEST #4 BY
11642 ;LOADING THE COMMAND FIELD OF PCSRO WITH A 4, WAIT FOR 'DNI'
11643
11644 046372' 004737 020060' JSR PC,CHKMON ;WAIT FOR MICROMONITOR
11645 046376' 103006 BCC 35$ ;OK
11646 046400' ERRHRD 249,RBRRUN,MSG46 ;PRINT ERROR
11647 046400' 104456 TRAP C$ERHPD
11648 046402' 000371 .WORD 249
11649 046404' 003650' .WORD RBRRUN
11650 046406' 016666' .WORD MSG46
11651 046410' ESCAPE TST ;LEAVE TEST
11652 046410' 104410 TRAP C$ESCAPE
11653 046412' 000220 .WORD L10163-
11654 046414' 012777 000004 131714 35$: MOV #4,@PCSRO ;TELL T11 TO EXECUTE MICROTEST #4
11655 046422' 012737 000275 000332' MOV #3*SECOND,METER ;WAIT FOR DNI
11656 046430' 004737 017316' JSR PC,CHKDNI
11657 046434' 103021 BCC 40$ ;OK
11658 046436' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
11659 046442' 103006 BCC 36$ ;NO, OK
11660 046444' ERRHRD 250,RBRRUN,MSG44 ;PRINT ERROR MESSAGE
11661 046444' 104456 TRAP C$ERHRD
11662 046446' 000372 .WORD 250
11663 046450' 003650' .WORD RBRRUN
11664 046452' 016442' .WORD MSG44
11665 046454' ESCAPE TST
11666 046454' 104410 TRAP C$ESCAPE
11667 046456' 000154 .WORD L10163-
11668 046460' 012702 000004 36$: MOV #4,R2 ;MICROTEST #4 IS HUNG
11669 046464' ERRHRD 251,RBRRUN,MSG12 ;PRINT ERROR MESSAGE
11670 046464' 104456 TRAP C$ERHRD
11671 046466' 000373 .WORD 251
11672 046470' 003650' .WORD RBRRUN
11673 046472' 013466' .WORD MSG12
11674 046474' ESCAPE TST
11675 046474' 104410 TRAP C$ESCAPE
11676 046476' 000134 .WORD L10163-
11677
11678 ;'DNI' SET INDICATING THE TEST IS FINISHED, NOW CHECK PCSR1 FOR AN ERROR
11679 ;CONDITION. THIS CONDITION WILL BE SET IF, AFTER THE MICROCODE TRANSMITTED THE
11680 ;RUNT PACKET, A RECEIVER INTERRUPT OCCURRED. THIS SHOULD NOT HAPPEN BECAUSE
11681 ;THE RECEIVER STATE MACHINE SHOULD GO THROUGH A BAD PACKET STATE AND NOT
11682 ;CAUSE A RECEIVER INTERRUPT
11683
11684 046500' 122777 000003 131632 40$: CMPB #INERR,@PCSR1 ;DID AN ERROR OCCUR?
11685 046506' 001005 BNE 45$ ;NO
11686 046510' ERRHRD 252,RBRRUN,MSG30 ;YES, PRINT ERROR MESSAGE
11687 046510' 104456 TRAP C$ERHRD
11688 046512' 000374 .WORD 252
11689 046514' 003650' .WORD RBRRUN
11690 046516' 015264' .WORD MSG30
11691 046520' 000425 BR 55$
11692
11693 ;OK, NO RECEIVER INTERRUPT. NOW CHECK PCBB+2, WHICH IS THE CONTENTS OF THE
  
```



65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 246  
 CZUAAB.MAC 07-APR-83 17:03 TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

```

11694 ;BUFFER DONE FIFO AFTER THE SECOND PACKET WAS RECEIVED. BITS 14:11 SHOULD
11695 ;ALL BE ZERO INDICATING BUFFER 0 WAS RECOVERED BY THE RECEIVER STATE MACHINE.
11696 ;
11697 046522' 042737 103777 000610' 45$: BIC #103777,PCBB+2 ;STRIP OFF THE FLOATING BITS
11698 046530' 005737 000610' TST PCBB+2 ;DID DEUNA RECOVER RECEIVE BUFFER?
11699 046534' 001407 BEQ 508 ;YES
11700 046536' 052737 100000 000610' BIS #100000,PCBB+2 ;NO, MAKE IT A LINK MEMORY ADDRESS
11701 046544' ERRHRD 253,RBRRUN,MSG31 ;PRINT ERROR MESSAGE
11702 046544' 104456 TRAP C$ERHRD
11703 046546' 000375 .WORD 253
11704 046550' 003650' .WORD RBRRUN
11705 046552' 015332' .WORD MSG31
11706 ;
11707 ;NOW CHECK THE DATA THAT WAS RECEIVED INTO THE RECEIVER BUFFER. IT SHOULD
11708 ;BE AN ALTERNATING 1'S AND 0'S PATTERN.
11709 ;
11710 046554' 022737 052525 000612' 50$: CMP #52525,PCBB+4 ;IS DATA GOOD IN LEGIT RECEIVE BUFFER?
11711 046562' 001404 BEQ 558 ;YES
11712 046564' ERRHRD 254,RBRRUN,MSG32 ;NO,PRINT ERROR MESSAGE
11713 046564' 104456 TRAP C$ERHRD
11714 046566' 000376 .WORD 254
11715 046570' 003650' .WORD RBRRUN
11716 046572' 015476' .WORD MSG32
11717 046574' 55$:
11718 046574' ENDSEG
11719 046574'
11720 046574' 100018: TRAP C$ESEG
11721 ;
11722 ;WRITE '1' TO CLEAR 'DNI'
11723 ;
11724 046576' 004737 017362' JSR PC,CLRDN1 ;GO CLEAR DNI
11725 046602' 103006 BCC 608 ;OK
11726 046604' ERRHRD 255,RBRRUN,RACMG7 ;ERROR DNI DID NOT CLEAR!
11727 046604' 104456 TRAP C$ERHRD
11728 046606' 000377 .WORD 255
11729 046610' 003650' .WORD RBRRUN
11730 046612' 012670' .WORD RACMG7
11731 046614' ESCAPE TST
11732 046614' 104410 TRAP C$ESCAPE
11733 046616' 000014 .WORD L10163-.
11734 ;
11735 ;GENERATE A NEW RUNT PACKET BYTE COUNT. WE WILL JUST SLIDE A BIT THROUGH
11736 ;THE COUNTER, UP TO THE LAST RUNT PACKET SIZE OF 63
11737 ;
11738 046620' 006301 60$: ASL R1 ;MOVE OVER THE ALREADY SET BITS
11739 046622' 005201 INC R1 ;SET LSB
11740 046624' 020127 000100 CMP R1,#MINBYT ;STOP WITH A BYTE COUNT GREATER THAN
11741 ; ;MINIMUM SIZE
11742 046630' 002651 BLT 308
11743 046632' ENDTST
11744 046632'
11745 046632' 104401 L10163: TRAP C$ETST
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 247  
CZUAAB.MAC 07-APR-83 17:03 TEST 42: HALF-DUPLEX TEST

11746  
11747  
11748  
11749  
11750  
11751  
11752  
11753  
11754  
11755  
11756  
11757  
11758  
11759  
11760  
11761  
11762  
11763  
11764  
11765  
11766  
11767  
11768  
11769  
11770  
11771  
11772  
11773  
11774  
11775  
11776  
11777  
11778  
11779  
11780  
11781  
11782  
11783  
11784  
11785  
11786  
11787  
11788  
11789  
11790  
11791  
11792  
11793  
11794  
11795  
11796  
11797  
11798  
11799  
11800  
11801

046634'  
046634'

046634'  
046634' 104404  
046634' 022737 600106 000326'

.SBTTL TEST 42: HALF-DUPLEX TEST

.....

:THE LINK INCLUDES A 'HALF DUPLEX' MODE OF OPERATION. THIS MODE CAN BE ENABLED  
:OR DISABLED THROUGH THE LINK MODE REGISTER. THE OPERATIONAL MICROCODE NORMALLY  
:USES HALF-DUPLEX MODE.

:IN THE HALF-DUPLEX MODE, THE LINK WILL NOT RECEIVE MESSAGES ADDRESSED TO  
:ITSELF. INCOMING MESSAGES LOOPED BACK WILL BE IGNORED BY THE RECEIVE STATE  
:MACHINE. THE STATE MACHINE WILL NOT ISSUE A 'RECEIVER DONE' INTERRUPT AND THE  
:BUFFER CAN BE RECOVERED FOR RECEIVING A LATER DATAGRAM.

:THIS TEST USES MICROMODULE 'F' MICROTTEST #5.  
:THIS TEST WILL VERIFY THE OPERATION OF HALF-DUPLEX MODE. A DATAGRAM WILL BE  
:TRANSMITTED IN LOOPBACK MODE WITH THE HALF-DUPLEX BIT SET. THE MICROCODE  
:WILL VERIFY THAT THE RECEIVER INTERRUPT DOES NOT OCCUR. THE MICROCODE WILL  
:THEN CLEAR THE HALF-DUPLEX BIT AND LOOP A DATAGRAM AND VERIFY THAT THE  
:ORIGINAL BUFFER WAS RECOVERED.

:THIS TEST WILL USE THE PCBB TO PASS INFORMATION. PCBB+0 WILL BE USED TO PASS  
:THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM IS RECEIVED.  
:PCBB+4 WILL BE USED TO PASS THE FIRST WORD OF DATA FROM THE RECEIVER BUFFER  
:AFTER THE SECOND DATAGRAM IS TRANSMITTED.

:PCBB+0:            +-----+  
                  ! CONTENTS OF BUFFER DONE FIFO !  
                  +-----+  
:PCBB+2:            +-----+  
                  ! FIRST DATA WORD OF BUFFER DONE!  
                  +-----+

:THE CONTENTS OF THE BUFFER DONE FIFO SHOULD BE 0 AND THE FIRST DATA WORD  
:SHOULD BE AN ALTERNATING 1'S AND 0'S PATTERN.

:TEST SEQUENCE:  
:  1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO.  
:  2-CLEAR PCBB+0 AND PCBB+2  
:  3-WAIT FOR THE MICROMONITOR TO ENTER THE 'IMMON' STATE  
:  4-WAIT FOR 'DNI'  
:  5-CHECK PCSR1 FOR AND ERROR CONDITION (THIS SIGNIFIES THAT THE DATAGRAM  
:  SENT IN HALF-DUPLEX MODE CAUSED AN INTERRUPT)  
:  6-VERIFY PCBB+0 IS LINK BUFFER 0  
:  7-VERIFY PCBB+2 HAS ALTERNATING 1'S AND 0'S

.....

BGNTST

Y42::

:CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS. START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.

BGNSEG

CMP #'F.MICRO

TRAP C8BSEG  
:HAS MICROCODE MODULE 'F' BEEN LOADED

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 248  
 CZUAAB.MAC 07-APR-83 17:03 TEST 42: HALF-DUPLEX TEST

```

11802 046644' 001004      BNE      5$      ;NO
11803 046646' 122777      CMPB     #INMON,&PCSR1 ;YES, IS THE MICROMONITOR ACTIVE?
11804 046654' 001440      BEQ      20$     ;YES SKIP LOADING THE MICROMODULE
11805 046656' 012737      000106 000326' 5$:  MOV      #'F,MICRO ;GO LOAD MICRO MODULE 'F'
11806 046664' 004737      020340'  JSR      PC,L0DMIC
11807 046670' 103002      BCC      10$     ;OK
11808 046672'          ESCAPE   TST
11809 046672' 104410          TRAP     C$ESCAPE
11810 046674' 000342          .WORD   L10164-.
11811 046676' 012737      000176 000332' 10$:  MOV      #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
11812 046704' 004737      017316'  JSR      PC,CHKDNI
11813 046710' 103022      BCC      20$     ;OK
11814 046712' 012737      001000' 000310'  MOV      #SDNI,BITNAM
11815 046720' 012737      001277' 000312'  MOV      #SNSET,BITSTA
11816 046726' 012737      001342' 000314'  MOV      #SAFTER,PWHEN
11817 046734' 012737      001357' 000316'  MOV      #SGTCMD,PCOMND
11818 046742'          ERRHRD  256,HAFDUP,MSG1
11819 046742' 104456          TRAP     C$ERRHRD
11820 046744' 000400          .WORD   256
11821 046746' 003723'          .WORD   HAFDUP
11822 046750' 012716'          .WORD   MSG1
11823 046752'          ESCAPE   TST
11824 046752' 104410          TRAP     C$ESCAPE
11825 046754' 000262          .WORD   L10164-.
11826 046756' 004737      017362'  20$:  JSR      PC,CLRDN1 ;CLEAR DNI BIT
11827 046762' 103006      BCC      25$     ;DNI DID NOT CLEAR!
11828 046764'          ERRHRD  257,HAFDUP,RACMG7
11829 046764' 104456          TRAP     C$ERRHRD
11830 046766' 000401          .WORD   257
11831 046770' 003723'          .WORD   HAFDUP
11832 046772' 012670'          .WORD   RACMG7
11833 046774'          ESCAPE   TST
11834 046774' 104410          TRAP     C$ESCAPE
11835 046776' 000240          .WORD   L10164-.
11836 047000'          25$:
11837 047000'          ENDSEG
11838 047000'          10000$:
11839 047000' 104405          TRAP     C$ESEG
11840          :
11841          :CLEAR PCBB+0 AND PCBB+2 THESE LOCATIONS ARE WHERE THE MICROCODE WILL PLACE
11842          :INFORMATION ABOUT THE TEST SUCCESS
11843          :
11844          :BGNSEG
11845          :
11846          :CLEAR PCBB+0
11847          :
11848          :CLEAR PCBB+2
11849          :
11850          :
11851          :WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE
11852          :THEN EXECUTE MICROTEST #5 BY LOADING PCSRO WITH A 5
11853          :WAIT FOR 'DNI'
11854          :
11855 047014' 004737      020060'  JSR      PC,CHKMON ;WAIT FOR MICROMONITOR
11856 047020' 103006      BCC      35$     ;OK
11857 047022'          ERRHRD  258,HAFDUP,MSG46 ;PRINT ERROR

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 249  
 CZUAAB.MAC 07-APR-83 17:03 TEST 42: HALF-DUPLEX TEST

```

11858 047022' 104456
11859 047024' 000402 TRAP CSERHRD
11860 047026' 003723' .WORD 258
11861 047030' 016666' .WORD HAFDUP
11862 047032' ESCAPE TST :LEAVE TEST .WORD MSG46
11863 047032' 104410 TRAP CSERHRD
11864 047034' 000202 .WORD 259
11865 047036' 012777 000005 131272 35$: MOV #5, @PCSR0 ;TELL T11 TO EXECUTE MICROTEST #5
11866 047044' 012737 000473 000332' MOV #5*SECOND, METER ;WAIT FOR DNI
11867 047052' 004737 017316' JSR PC, CHK DNI
11868 047056' 103021 BCC 40$ ;OK
11869 047060' 004737 020132' JSR PC, CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
11870 047064' 103006 BCC 36$ ;NO, OK
11871 047066' ERRHRD 259, HAFDUP, MSG44 ;PRINT ERROR MESSAGE
11872 047066' 104456 TRAP CSERHRD
11873 047070' 000403 .WORD 259
11874 047072' 003723' .WORD HAFDUP
11875 047074' 016442' .WORD MSG44
11876 047076' ESCAPE TST
11877 047076' 104410 TRAP CSERHRD
11878 047100' 000136 .WORD 259
11879 047102' 012702 000005 36$: MOV #5, R2 ;MICROTEST #5 IS HUNG
11880 047106' ERRHRD 260, HAFDUP, MSG12 ;PRINT ERROR MESSAGE
11881 047106' 104456 TRAP CSERHRD
11882 047110' 000404 .WORD 260
11883 047112' 003723' .WORD HAFDUP
11884 047114' 013466' .WORD MSG12
11885 047116' ESCAPE TST
11886 047116' 104410 TRAP CSERHRD
11887 047120' 000116 .WORD 261
11888
11889 ;AN ERROR IN PCSR1 MEANS THAT THE FIRST DATAGRAM SENT, WHICH IS THE ONE SENT
11890 ;IN HALF-DUPLEX MODE, CAUSED A RECEIVER INTERRUPT.
11891
11892 047122' 122777 000003 131210 40$: CMPB #INERR, @PCSR1 ;DID AN ERROR OCCUR?
11893 047130' 001005 BNE 45$ ;NO
11894 047132' ERRHRD 261, HAFDUP, MSG39 ;YES, PRINT ERROR MESSAGE
11895 047132' 104456 TRAP CSERHRD
11896 047134' 000405 .WORD 261
11897 047136' 003723' .WORD HAFDUP
11898 047140' 016056' .WORD MSG39
11899 047142' 000425 BR 55$
11900
11901 ;PCBB+0 CONTAINS THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM
11902 ;WAS RECEIVED IT SHOULD CONTAIN THE BUFFER ADDRESS OF THE FIRST BUFFER IN LINK
11903 ;MEMORY
11904
11905 047144' 042737 103777 000606' 45$: BIC #103777, PCBB ;STRIP OFF THE FLOATING BITS
11906 047152' 005737 000606' TST PCBB ;DID DEUNA RECOVER RECEIVE BUFFER?
11907 047156' 001407 BEQ 50$ ;YES
11908 047160' 052737 100000 000606' BIS #100000, PCBB ;MAKE IT A LINK MEMORY ADDRESS
11909 047166' ERRHRD 262, HAFDUP, MSG40 ;NO, PRINT ERROR MESSAGE
11910 047166' 104456 TRAP CSERHRD
11911 047170' 000406 .WORD 262
11912 047172' 003723' .WORD HAFDUP
11913 047174' 016100' .WORD MSG40

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 250  
 CZUAAB.MAC 07-APR-83 17:03 TEST 42: HALF-DUPLEX TEST

```

11914
11915 ;PCBB+2 CONTAINS THE FIRST DATA WORD FROM THE FIRST BUFFER IN LINK MEMORY
11916 ;IT SHOULD BE ALTERNATING 1'S AND 0'S IF THE DATA WAS PLACED IN THIS BUFFER
11917 ;
11918 047176' 022737 052525 000610' 50$: CMP #52>25,PCBB+2 ;IS DATA GOOD IN LEGIT RECEIVE BUFFER?
11919 047204' 001404 BEQ 55$ ;YES
11920 047206' ERRHRD 263,HAFDUP,MSG41 ;NC.PRINT ERROR MESSAGE
11921 047206' 104456 TRAP C$ERHRD
11922 047210' 000407 .WORD 263
11923 047212' 003723' .WORD HAFDUP
11924 047214' 016176' .WORD MSG41
11925 047216' 55$:
11926 047216' ENDSEG
11927 047216'
11928 047216' 104405 10001$: TRAP C$ESEG
11929 ;
11930 ;WRITE '1' TO CLEAR 'DNI'
11931 ;
11932 047220' 004737 017362' JSR PC,CLRDNI ;GO CLEAR DNI
11933 047224' 103004 BCC 60$ ;OK
11934 047226' ERRHRD 264,HAFDUP,RACMG7 ;ERROR DNI DID NOT CLEAR!
11935 047226' 104456 TRAP C$ERHRD
11936 047230' 000410 .WORD 264
11937 047232' 003723' .WORD HAFDUP
11938 047234' 012670' .WORD RACMG7
11939 047236' 60$:
11940 047236' ENDTST
11941 047236'
11942 047236' 104401 L10164: TRAP C$ETST
    
```

.SBTTL TEST 43: COLLISION TEST

11943  
11944  
11945  
11946  
11947  
11948  
11949  
11950  
11951  
11952  
11953  
11954  
11955  
11956  
11957  
11958  
11959  
11960  
11961  
11962  
11963  
11964  
11965  
11966  
11967  
11968  
11969  
11970  
11971  
11972  
11973  
11974  
11975  
11976  
11977  
11978  
11979  
11980  
11981  
11982  
11983  
11984  
11985  
11986  
11987  
11988  
11989  
11990  
11991  
11992  
11993  
11994  
11995  
11996  
11997  
11998

\*\*\*\*\*  
:THE RECEIVE STATE MACHINE REACTS TO COLLISIONS ON THE WIRE BY ACTIVATING  
:THE RETRY LOGIC. THE RETRY LOGIC WAITS AN INTERVAL OF TIME BEFORE ATTEMPTING  
:TO RETRANSMIT THE DATAGRAM. THE INTERVALS ARE NOT UNIFORM BUT ARE OF  
:GENERALLY INCREASING PSEUDO-RANDOM DURATION. THE RETRY LOGIC WILL ATTEMPT  
:TO RETRANSMIT UP TO 15 ADDITIONAL TIMES BEFORE GIVING UP.  
:THIS TEST WILL VERIFY THAT THE RECEIVE STATE MACHINE RESPONDS TO A COLLISION  
:AND THAT THE RETRY SEQUENCE IS REPORTED CORRECTLY IN THE TRANSMIT STATUS WORD.  
:THIS TEST WILL USE MICROMODULE 'G' MICROTEST #1.  
:THE LINK BOARD CONTAINS DIAGNOSTIC LOGIC THAT ALLOWS COLLISIONS TO BE SIMULATED.  
:WITH THE FORCE COLLISIONS LOGIC ACTIVATED, THE RETRY HARDWARE CAN BE STEPPED  
:THROUGH THE RETRY SEQUENCE. THAT IS, EVERY DATAGRAM LOOPED BACK WILL STEP  
:THE RETRY LOGIC THROUGH ONE STEP OF THE RETRY SEQUENCE. THE RETRY SEQUENCE  
:CAN BE VERIFIED BY CHECKING THE TRANSMIT BUFFER STATUS WORDS AFTER EACH RETRY  
:STEP.  
:THE PCBB WILL BE USED TO PASS PARAMETERS BETWEEN THE MICROCODE AND THE HOST  
:PROCESSOR. PCBB+0 WILL BE USED TO PASS THE DATA TO BE LOADED INTO THE LINK  
:MODE WORD. PCBB+2 WILL BE PASSED BACK BY THE MICROCODE, IT IS THE FIRST WORD  
:OF THE TRANSMIT BUFFER (TRANSMIT STATUS WORD 0). PCBB+4 WILL ALSO BE PASSED  
:BACK, IT IS TRANSMIT STATUS WORD 1.  
:THE TRANSMIT STATUS WORDS SHOULD SHOW THE FOLLOWING STATUS:

STATUS BITS

	WORD 0			WORD 1
	ERRS	MORE	ONE	RETRY
	(14)	(12)	(11)	(10)
LOOPBACK STEP #				
1	0	0	1	0
2-15	0	1	0	0
16	1	0	0	1

:THE PCBB IS FORMATTED AS FOLGWS:  
:PCBB+0: LINK MODE WORD  
:PCBB+2: TRANSMIT STATUS WORD 0!  
:PCBB+4: TRANSMIT STATUS WORD 1!

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 252  
CZUAAB.MAC 07-APR-83 17:03 TEST 43: COLLISION TEST

11999  
12000  
12001  
12002  
12003  
12004  
12005  
12006  
12007  
12008  
12009  
12010  
12011  
12012  
12013  
12014  
12015  
12016  
12017  
12018  
12019  
12020  
12021  
12022  
12023  
12024  
12025  
12026  
12027

: TEST SEQUENCE:  
: 1-LOAD MICROMODULE 'G' IF NOT ALREADY SOME SO  
: 2-LOAD PCBB+0 WITH PROMISCUOUS MODE, INTERNAL LOOPBACK AND FORCE  
: COLLISIONS.  
: 3-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE  
: 4-EXECUTE MICROTEST #1  
: 5-WAIT FOR 'DNI'  
: 6-CHECK FOR 'ONE' BIT IN PCBB+0  
: 7-WRITE '1' TO CLEAR 'DNI'  
: 8-LOAD PCBB+0 WITH PROMISCUOUS MODE, INTERNAL LOOPBACK AND FORCE  
: COLLISIONS.  
: 9-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE  
: 10-EXECUTE MICROTEST #1  
: 11-WAIT FOR 'DNI'  
: 12-CHECK PCBB+2 FOR 'MORE' BIT  
: 13-WRITE '1' TO CLEAR 'DNI'  
: 14-REPEAT STEPS 8-13 15 TIMES  
: 15-LOAD PCBB+0 WITH PROMISCUOUS MODE, INTERNAL LOOPBACK AND FORCE  
: COLLISIONS  
: 16-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE  
: 17-EXECUTE MICROTEST #1  
: 18-WAIT FOR 'DNI'  
: 19-CHECK PCBB+4 FOR ERROR SUMMARY BIT IN PCBB+2 AND RETRY BIT IN  
: PCBB+4  
: 20-WRITE '1' TO CLEAR 'DNI'

.....8

12028 047240'  
12029 047240'  
12030  
12031  
12032  
12033  
12034

BGNTST  
T43::  
: CHECK TO SEE IF MODULE 'G' HAS BEEN LOADED. IF NOT LOAD IT INTO  
: THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
: AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

12035 047240'  
12036 047240' 104404  
12037 047242' 022737 000107 000326'  
12038 047250' 001004  
12039 047252' 122777 000001 131060  
12040 047260' 001440  
12041 047262' 012737 000107 000326' 5S:  
12042 047270' 004737 020340'  
12043 047274' 103002  
12044 047276'  
12045 047276' 104410  
12046 047300' 001026  
12047 047302' 012737 000176 000332' 10S:  
12048 047310' 004737 017316'  
12049 047314' 103022  
12050 047316' 012737 001000' 000310'  
12051 047324' 012737 001277' 000312'  
12052 047332' 012737 001342' 000314'  
12053 047340' 012737 001357' 000316'  
12054 047346'

BGNSEG  
: TRAP CSBSEG  
: HAS MICROCODE MODULE 'G' BEEN LOADED  
: NO  
: YES, IS THE MICROMONITOR ACTIVE?  
: YES SKIP LOADING THE MICROMODULE  
: GO LOAD MICRO MODULE 'G'  
: OK  
: TRAP CSEscape  
: WORD L10165-  
: MOV #2\*SECOND,METER :WAIT FOR THE MICROMONITOR  
: JSR PC,CHKDNI  
: BCC 20\$ :OK  
: MOV #SDNI,BITNAM  
: MOV #SNSET,BITSTA  
: MOV #SAFTER,PWEN  
: MOV #SGTCMD,PCOMND  
ERRHRD 265,COLTST,MSG1

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 253  
 CZUAAB.MAC 07-APR-83 17:03 TEST 43: COLLISION TEST

```

12055 047346' 104456 TRAP CSERHRD
12056 047350' 000411 .WORD 265
12057 047352' 003753' .WORD COLTST
12058 047354' 012716' .WORD MSG1
12059 047356' ESCAPE TST
12060 047356' 104410 TRAP CSERHRD
12061 047360' 000746 .WORD 266
12062 047362' 004737 017362' 20$: JSR PC,CLRDNI ;CLEAR DNI BIT
12063 047366' 103006 BCC 25$
12064 047370' ERRHRD 266,COLTST,RACMG7 ;DNI DID NOT CLEAR!
12065 047370' 104456 TRAP CSERHRD
12066 047372' 000412 .WORD 266
12067 047374' 003753' .WORD COLTST
12068 047376' 012670' .WORD RACMG7
12069 047400' ESCAPE TST
12070 047400' 104410 TRAP CSERHRD
12071 047402' 000724 .WORD L10165-
12072 047404' 25$:
12073 047404' ENDSEG
12074 047404'
12075 047404' 104405 10000$: TRAP CSESEG
12076
12077 ;
12078 ;LOAD PCSR2 WITH ADDRESS OF PORT CONTROL BLOCK
12079 ;LOAD REGISTER 5 WITH BITS TO BE LOADED INTO THE LINK MODE REGISTER BY THE
12080 ;MICROCODE
12081 047406' 012777 000606' 130726 MOV #PCBB,@PCSR2 ;TELL DEUMA WHERE PCBB IS
12082 047414' 005077 130724 CLR @PCSR5
12083 047420' 012705 100024 MOV #BIT15!BIT4!BIT2,R5 ;TELL MICROCODE TO LOAD THE FOLLOWING
12084 ;INTO THE LINK MODE REGISTER:
12085 ;PROMISCUOUS MODE, INTERNAL LOOPBACK,
12086 ;AND FORCE COLLISIONS
12087 ;
12088 ;BEGIN FIRST LOOPBACK
12089 ;
12090 ; BGNSEG
12091 047424' TRAP CSBSEG
12092 047424' 104404
12093 047426' 012702 000001 MOV #1,R2 ;START LOOPBACK STEP #1
12094 047432' 010537 000606' MOV R5,PCBB ;LOAD PCBB WITH LINK MODE REGISTER DATA
12095 047436' 005037 000610' CLR PCBB+2 ;MICROCODE WILL PUT XMIT STAT WORD 0
12096 047442' 005037 000612' CLR PCBB+4 ;MICROCODE WILL PUT XMIT STAT WORD 1
12097 047446' 004737 020060' JSR PC,CHKMON ;WAIT FOR MICROMONITOR
12098 047452' 103006 BCC 30$ ;OK
12099 047454' ERRHRD 267,COLTST,MSG46 ;PRINT ERROR
12100 047454' 104456 TRAP CSERHRD
12101 047456' 000413 .WORD 267
12102 047460' 003753' .WORD COLTST
12103 047462' 016666' .WORD MSG46
12104 047464' ESCAPE TST ;LEAVE TEST
12105 047466' 104410 TRAP CSERHRD
12106 047470' 000640 .WORD L10165-
12107 047470' 012777 000001 130640 30$: MOV #1,@PCSR0 ;EXECUTE MICROTST #1
12108 047476' 012737 000176 000332' MOV #2*SECOND,METER ;WAIT FOR DNI
12109 047504' 004737 017316' JSR PC,CHKDNI
12110 047510' 103021 BCC 40$
12111 047512' 004737 020132' JSR PC,CHKINT ;SEE IF ANY ERROR INTERRUPTS OCCURRED
    
```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 254  
CZUAAB.MAC 07-APR-83 17:03 TEST 43: COLLISION TEST

```

12111 047516' 103006          BCC      358          ;NO, OK
12112 047520'                ERRHRD   268,COLTST,MSG44 ;PRINT ERROR MESSAGE
12113 047520' 104456                TRAP      C$ERHRD
12114 047522' 000414                .WORD    268
12115 047524' 003753'                .WORD    COLTST
12116 047526' 016442'                .WORD    MSG44
12117 047530'                ESCAPE   TST
12118 047530' 104410                TRAP      C$ESCAPE
12119 047532' 000574                .WORD    L10165-.
12120 047534' 012702 000001      358:   MOV      #1,R2          ;MICROTEST #1 IS MUNG
12121 047540'                ERRHRD   269,COLTST,MSG12
12122 047540' 104456                TRAP      C$ERHRD
12123 047542' 000415                .WORD    269
12124 047544' 003753'                .WORD    COLTST
12125 047546' 013466'                .WORD    MSG12
12126 047550'                ESCAPE   TST
12127 047550' 104410                TRAP      C$ESCAPE
12128 047552' 000554                .WORD    L10165-.
12129
12130      ;THE RESULT OF THE FIRST LOOPBACK SHOULD BE TX WORD 0 'ONE' BIT SET AND NO
12131      ;OTHERS
12132
12133 047554' 012703 004000      408:   MOV      #BIT11,R3          ;'ONE' BIT SHOULD BE SET IN TX WORD 0
12134 047560' 005004                CLR      R4                ;NO BITS SHOULD BE SET IN TX WORD 1
12135 047562' 032737 040000 000610'  BIT      #BIT14,PCBB+2      ;IS 'ERROR SUMMARY' SET IN WORD 0?
12136 047570' 001014                BNE      458                ;YES, ERROR
12137 047572' 032737 010000 000610'  BIT      #BIT12,PCBB+2      ;IS 'MORE' BIT SET IN WORD 0?
12138 047600' 001010                BNE      458                ;YES, ERROR
12139 047602' 032737 004000 000610'  BIT      #BIT11,PCBB+2      ;IS 'ONE' BIT SET IN WORD 0?
12140 047610' 001404                BEQ      458                ;NO, ERROR
12141 047612' 032737 002000 000612'  BIT      #BIT10,PCBB+4      ;IS 'RETRY' BIT SET IN WORD 1?
12142 047620' 001404                BEQ      508                ;NO
12143 047622'                458:   ERRHRD   270,COLTST,MSG22 ;PRINT ERROR MESSAGE
12144 047622' 104456                TRAP      C$ERHRD
12145 047624' 000416                .WORD    270
12146 047626' 003753'                .WORD    COLTST
12147 047630' 014326'                .WORD    MSG22
12148
12149
12150      ;WRITE 'ONE' TO CLEAR 'DNI'
12151
12152 047632' 004737 017362'      508:   JSR      PC,CLRDNI          ;GO CLEAR DNI
12153 047636' 103004                BCC      558                ;OK
12154 047640'                ERRHRD   271,COLTST,RACMG7 ;PRINT ERROR MESSAGE
12155 047640' 104456                TRAP      C$ERHRD
12156 047642' 000417                .WORD    271
12157 047644' 003753'                .WORD    COLTST
12158 047646' 012670'                .WORD    RACMG7
12159 047650'                558:
12160 047650'                ENDSEG
12161 047650'
12162 047650' 104405                100018: TRAP      C$ESEG
12163
12164      ;BEGIN LOOPBACKS 2-15
12165
12166 047652'                BGNSEG

```

65 HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 255  
 CZUAMB.MAC 07-APR-83 17:03 TEST 43: COLLISION TEST

```

12167 047652' 104404
12168 047654' 012702 000002          MOV      #2,R2          :START LOOPS 2-15          TRAP      CSBSEG
12169
12170 047660' 010537 000606'          60S:    MOV      R5,PCBB          :LOAD PCBB WITH LINK MODE REGISTER DATA
12171 047664' 005037 000610'          CLR      PCBB+2          :MICROCODE WILL PUT XMIT STAT WORD 0
12172 047670' 005037 000612'          CLR      PCBB+4          :MICROCODE WILL PUT XMIT STAT WORD 1
12173 047674' 004737 020060'          JSR      PC,CHKMON       :WAIT FOR MICROMONITOR
12174 047700' 103006          BCC      70S            :OK
12175 047702'          ERRHRD  272,COLTST,MSG46 :PRINT ERROR
12176 047702' 104456          TRAP      CSERHRD
12177 047704' 000420          .WORD    272
12178 047706' 003753'          .WORD    COLTST
12179 047710' 016666'          .WORD    MSG46
12180 047712'          ESCAPE  TST            :LEAVE TEST
12181 047712' 104410          TRAP      CSERHRD
12182 047714' 000412          .WORD    L10165-.
12183 047716' 012777 000001 130412 70S:    MOV      #1,BPCSR0       :EXECUTE MICROTEST #1
12184 047724' 012737 000176 000332'    MOV      #2*SECOND,METER :WAIT FOR DMI
12185 047732' 004737 017316'    JSR      PC,CHKDMI
12186 047736' 103017          BCC      80S
12187 047740' 004737 020132'    JSR      PC,CHKINT       :SEE IF ANY ERROR INTERRUPTS OCCURRED
12188 047744' 103004          BCC      75S            :NO, OK
12189 047746'          ERRHRD  273,COLTST,MSG44 :PRINT ERROR MESSAGE
12190 047746' 104456          TRAP      CSERHRD
12191 047750' 000421          .WORD    273
12192 047752' 003753'          .WORD    COLTST
12193 047754' 016442'          .WORD    MSG44
12194 047756' 012702 000001          75S:    MOV      #1,R2
12195 047762'          ERRHRD  274,COLTST,MSG12 :MICROTEST #1 IS HUNG
12196 047762' 104456          TRAP      CSERHRD
12197 047764' 000422          .WORD    274
12198 047766' 003753'          .WORD    COLTST
12199 047770' 013466'          .WORD    MSG12
12200 047772'          ESCAPE  TST
12201 047772' 104410          TRAP      CSERHRD
12202 047774' 000332          .WORD    L10165-.
12203
12204          :THE RESULT OF LOOPBACKS 2-15 SHOULD BE THE 'MORE' BIT IN TX 0 AND NO OTHERS
12205
12206 047776' 012703 010000          80S:    MOV      #BIT12,R3       :'MORE' BIT SHOULD BE SET IN TX0
12207 050002' 032737 040000 000610'    BIT     #BIT14,PCBB+2    :IS 'ERROR SUMMARY' SET IN TX 0?
12208 050010' 001014          BNE      90S            :YES, ERROR
12209 050012' 032737 010000 000610'    BIT     #BIT12,PCBB+2    :IS 'MORE' BIT SET IN TX 0?
12210 050020' 001410          BEQ      90S            :NO, ERROR
12211 050022' 032737 004000 000610'    BIT     #BIT11,PCBB+2    :IS 'ONE' BIT SET IN TX 0?
12212 050030' 001004          BNE      90S            :YES, ERROR
12213 050032' 032737 002000 000612'    BIT     #BIT10,PCBB+4    :IS 'RETRY' BIT SET IN TX 1?
12214 050040' 001404          BEQ      100S           :NO
12215 050042'          90S:    ERRHRD  275,COLTST,MSG22 :PRINT ERROR MESSAGE
12216 050042' 104456          TRAP      CSERHRD
12217 050044' 000423          .WORD    275
12218 050046' 003753'          .WORD    COLTST
12219 050050' 014326'          .WORD    MSG22
12220
12221
12222          :WRITE '1' TO CLEAR 'DMI'
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 256  
 CZUAAB.MAC 07-APR-83 17:03 TEST 43: COLLISION TEST

```

12223
12224 050052' 004737 017362'      1008: JSR      PC,CLRDNI      :GO CLEAR DNI
12225 050056' 103004                BCC      1058             :OK
12226 050060'                   ERRHRD  276,COLTST,RACMG7 :PRINT ERROR MESSAGE
12227 050060' 104456                TRAP     CSERHRD
12228 050062' 000424                .WORD   276
12229 050064' 003753'                .WORD   COLTST
12230 050066' 012670'                .WORD   RACMG7
12231 050070' 005202                1058:  INC      R2             :LOOP COUNT
12232 050072' 022702 000020        CMP      #16.,R2         :HAVE WE DONE LOOP STEPS 2-15?
12233 050076' 001270                BNE     608             :NO CONTINUE
12234 050100'                   ENDSEG
12235 050100'
12236 050100' 104405                100028: TRAP     CSESEG
12237
12238 :BEGIN LOOPBACK #16
12239 :
12240 :BGNSEG
12241 050102' 104404                TRAP     CSBSEG
12242 050104' 010537 000606'        MOV      R5,PCBB         :LOAD PCBB WITH LINK MODE REGISTER DATA
12243 050110' 005037 000610'        CLR     PCBB+2          :MICROCODE WILL PUT XMIT STAT WORD 0
12244 050114' 005037 000612'        CLR     PCBB+4          :MICROCODE WILL PUT XMIT STAT WORD 1
12245 050120' 004737 020060'        JSR     PC,CHKMON       :WAIT FOR MICROMONITOR
12246 050124' 103006                BCC     1108            :OK
12247 050126'                   ERRHRD  277,COLTST,MSG46 :PRINT ERROR
12248 050126' 104456                TRAP     CSERHRD
12249 050130' 000425                .WORD   277
12250 050132' 003753'                .WORD   COLTST
12251 050134' 016666'                .WORD   MSG46
12252 050136'                   ESCAPE  TST             :LEAVE TEST
12253 050136' 104410                TRAP     CSESCAPE
12254 050140' 000166                .WORD   L10165-.
12255 050142' 012777 000001 130166 1108: MOV      #1,BPCSR0       :EXECUTE MICROTEST #1
12256 050150' 012737 000176 000332' MOV      #2+SECOND,METER :WAIT FOR DNI
12257 050156' 004737 017316'        JSR     PC,CHKDNI
12258 050162' 103021                BCC     1208
12259 050164' 004737 020132'        JSR     PC,CHKINT
12260 050170' 103006                BCC     1158            :SEE IF ANY ERROR INTERRUPTS OCCURRED
12261 050172'                   ERRHRD  278,COLTST,MSG44 :NO, OK
12262 050172' 104456                :PRINT ERROR MESSAGE
12263 050174' 000426                TRAP     CSERHRD
12264 050176' 003753'                .WORD   278
12265 050200' 016442'                .WORD   COLTST
12266 050202'                   ESCAPE  TST             :PRINT ERROR MESSAGE
12267 050202' 104410                TRAP     CSESCAPE
12268 050204' 000122                .WORD   L10165-.
12269 050206' 012702 000001 1158: MOV      #1,R2           :MICROTEST #1 IS HUNG
12270 050212'                   ERRHRD  279,COLTST,MSG12
12271 050212' 104456                TRAP     CSERHRD
12272 050214' 000427                .WORD   279
12273 050216' 003753'                .WORD   COLTST
12274 050220' 013466'                .WORD   MSG12
12275 050222'                   ESCAPE  TST
12276 050222' 104410                TRAP     CSESCAPE
12277 050224' 000102                .WORD   L10165-.
12278

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 258  
CZUAAB.MAC 07-APR-83 17:03 TEST 44: TDR COUNTER TEST

12316  
12317  
12318  
12319  
12320  
12321  
12322  
12323  
12324  
12325  
12326  
12327  
12328  
12329  
12330  
12331  
12332  
12333  
12334  
12335  
12336  
12337  
12338  
12339  
12340  
12341  
12342  
12343  
12344  
12345  
12346  
12347  
12348  
12349  
12350  
12351  
12352  
12353  
12354  
12355  
12356  
12357  
12358  
12359  
12360  
12361  
12362  
12363  
12364  
12365  
12366  
12367  
12368  
12369  
12370  
12371

.SBTTL TEST 44: TDR COUNTER TEST

:\*\*\*\*\*

:THE DEUNA HAS A COUNTER DESIGNED TO HELP LOCATE FAULTS IN THE COAXIAL CABLE.  
:THE COUNTER IS INITIALIZED WHEN A MESSAGE IS TRANSMITTED AND INCREMENTS AS  
:THE DATAGRAM IS TRANSMITTED. COUNTING WILL STOP IF A COLLISION OCCURS OR THE  
:CARRIER IS LOST. COUNTING ALSO STOPS IF THE 10 BIT COUNTER REACHES ITS  
:MODULUS.

:THIS TEST WILL DETERMINE THAT THE TDR COUNTER VALUE WILL CHANGE AND THAT THE  
:COUNTER IS NOT STUCK.

:BECAUSE THE COUNTER COUNTS DURING TRANSMISSION OF A DATAGRAM AND WILL CONTINUE  
:TO COUNT DURING THE TIME THAT THE TRANSMIT STATE MACHINE OPERATES, THE COUNT  
:ACCUMULATED IN THE COUNTER DURING TRANSMISSION IS PROPORTIONAL TO THE LENGTH  
:OF THE DATAGRAM. THIS TEST WILL USE THIS RELATION TO VERIFY THAT THE COUNTER  
:IS NOT STUCK.

:THIS TEST USES MICROMODULE 'G' MICROTEST #2.  
:THE TEST WILL SEND DATAGRAMS OVER THE LOOPBACK. THE LENGTH OF THE DATAGRAM  
:WILL BE VARIED BY USING AN INCREASING BYTE COUNT IN THE TRANSMIT BUFFER.  
:AFTER EACH DATAGRAM HAS BEEN LOOPED BACK, THE TRANSMIT BUFFER WORD 1 WILL BE  
:PASSED BACK TO THE HOST TO VERIFY THAT IT IS CORRECT. THE CRITERIA FOR  
:CORRECTNESS WILL BE: INCREASING BYTE COUNTS SHOULD RESULT IN INCREASING TDR  
:VALUES IN TRANSMIT STATUS WORD 1.

:THE PCBB WILL BE FORMATED AS FOLLOWS:

:PCBB+0:            +-----+  
                  !        BYTE COUNT        !  
                  +-----+  
:PCBB+2:            ! TRANSMIT STATUS WORD 1 !  
                  +-----+

:TEST SEQUENCE:

- 1-LOAD MICROMODULE 'G' IF NOT ALREADY DONE SO
- 2-LOAD MINIMUM BYTE COUNT INTO PCBB+0
- 3-CLEAR PCBB+2
- 4-WAIT FOR THE MICROMONITOR TO ENTER THE 'INPMON' STATE
- 5-SELECT MICROTEST #2
- 6-WAIT FOR 'DNI'
- 7-VERIFY PCBB+2 HAS NON-ZERO VALUE (TDR COUNTER NOT ZERO)
- 8-WRITE '1' TO CLEAR 'DNI'
- 9-INCREASE BYTE COUNT IN PCBB+0 BY 1
- 10-WAIT FOR MICROMONITOR TO ENTER 'INPMON' STATE
- 11-SELECT MICROTEST #2
- 12-WAIT FOR 'DNI'
- 13-VERIFY VALUE IN PCBB+2 IS GREATER THAN PREVIOUS VALUE IN PCBB+2  
(VERIFY THAT TDR VALUE IS GETTING LARGER WITH LARGER BYTE COUNTS)
- 14-WRITE '1' TO CLEAR 'DNI'
- 15-REPEAT STEPS 9-14 UNTIL BYTE REACHES MINIMUM SIZE +64

:\*\*\*\*\*

050330'

DNSTST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 259  
CZUAAB.MAC 07-APR-83 17:03 TEST 44: TDR COUNTER TEST

T44::

12372 050330'  
12373  
12374  
12375  
12376  
12377

:  
:CHECK TO SEE IF MODULE 'G' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF 'CS', START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.  
:AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.  
:

12378 050330'  
12379 050330' 104404  
12380 050332' 022737 000107 000326'  
12381 050340' 001004  
12382 050342' 122777 000001 127770  
12383 050350' 001440  
12384 050352' 012737 000107 000326' 58:  
12385 050360' 004737 020340'  
12386 050364' 103002  
12387 050366'  
12388 050366' 104410  
12389 050370' 000460  
12390 050372' 012737 000176 000332' 108:  
12391 050400' 004737 017316'  
12392 050404' 103022  
12393 050406' 012737 001000' 000310'  
12394 050414' 012737 001277' 000312'  
12395 050422' 012737 001342' 000314'  
12396 050430' 012737 001357' 000316'  
12397 050436'  
12398 050436' 104456  
12399 050440' 000432  
12400 050442' 004001'  
12401 050444' 012716'  
12402 050446'  
12403 050446' 104410  
12404 050450' 000400  
12405 050452' 004737 017362'  
12406 050456' 103006  
12407 050460'  
12408 050460' 104456  
12409 050462' 000433  
12410 050464' 004001'  
12411 050466' 012670'  
12412 050470'  
12413 050470' 104410  
12414 050472' 000356  
12415 050474'  
12416 050474'  
12417 050474'  
12418 050474' 104405

BGNSEG

TRAP CSBSEG  
:HAS MICROCODE MODULE 'G' BEEN LOADED  
58  
:NO  
:YES, IS THE MICROMONITOR ACTIVE?  
208  
:YES SKIP LOADING THE MICROMODULE  
:GO LOAD MICRO MODULE 'G'  
108  
:OK

TRAP CSBSEG  
.WORD L10166-

MOV #2\*SECOND,METER ;WAIT FOR THE MICROMONITOR

JSR PC,CNKDNI ;OK  
BCC 208

MOV #SDNI,BITNAM  
MOV #SNSET,BITSTA  
MOV #SAFTER,PLWEN  
MOV #SGTCMD,PCOMND  
ERRHRD 282,TDRCNT,MSG1

TRAP CSERHRD  
.WORD 282  
.WORD TDRCNT  
.WORD MSG1

ESCAPE TST

TRAP CSBSEG  
.WORD L10166-

JSR PC,CLRDN1 ;CLEAR DNI BIT  
BCC 258

ERRHRD 283,TDRCNT,RACMG7 ;DNI DID NOT CLEAR!

TRAP CSERHRD  
.WORD 283  
.WORD TDRCNT  
.WORD RACMG7

ESCAPE TST

TRAP CSBSEG  
.WORD L10166-

258:  
ENDSEG

100008:

TRAP CSBSEG

:  
:LOAD MINIMUM BYTE COUNT INTO PCBB+0, CLEAR PCBB+2, WAIT FOR MICROMONITOR  
:EXECUTE MICROTEST #2 BY LOADING PCSRO WITH A 2  
:WAIT FOR 'DNI'  
:

BGNSEG

12424 050476'  
12425 050476' 104404  
12426 050500' 012737 000100 000606'  
12427 050506' 005037 000610'

TRAP CSBSEG  
MOV #MINBYT,PCBB ;BEGIN WITH MINIMUM BYTE COUNT  
CLR PCBB+2 ;THIS IS WHERE MICROCODE WILL PUT...

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 260  
 CZUAAB.MAC 07-APR-83 17:03 TEST 44: TDR COUNTER TEST

```

12428
12429 050512' 004737 020060'          JSR    PC,CHKMON          ;THE TDR COUNTER VALUE
12430 050516' 103006          BCC    35$                ;WAIT FOR MICROMONITOR
12431 050520'          ERRHRD  284,TDRCNT,MSG46  ;OK
12432 050520' 104456          ;PRINT ERROR
12433 050522' 000434          TRAP   CSERHRD
12434 050524' 004001'          .WORD  284
12435 050526' 016666'          .WORD  TDRCNT
12436 050530'          .WORD  MSG46
12437 050530' 104410          ESCAPE TST                ;LEAVE TEST
12438 050532' 000316          TRAP   CSERHRD
12439 050534' 012777 000002 127574 35$:  MOV    #2,BPCSR0          ;TELL T11 TO EXECUTE MICROTEST #2
12440 050542' 012737 000176 000332'  MOV    #2*SECOND,METER  ;WAIT FOR DNI
12441 050550' 004737 017316'          JSR    PC,CHKDNI
12442 050554' 103021          BCC    40$
12443 050556' 004737 020132'          JSR    PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
12444 050562' 103006          BCC    36$                ;NO, OK
12445 050564'          ERRHRD  285,TDRCNT,MSG44  ;PRINT ERROR MESSAGE
12446 050564' 104456          TRAP   CSERHRD
12447 050566' 000435          .WORD  285
12448 050570' 004001'          .WORD  TDRCNT
12449 050572' 016442'          .WORD  MSG44
12450 050574'          ESCAPE TST
12451 050574' 104410          TRAP   CSERHRD
12452 050576' 000252          .WORD  L10166-.
12453 050600' 012702 000002          36$:  MOV    #2,R2                ;MICROTEST #2 IS HUNG
12454 050604'          ERRHRD  286,TDRCNT,MSG12
12455 050604' 104456          TRAP   CSERHRD
12456 050606' 000436          .WORD  286
12457 050610' 004001'          .WORD  TDRCNT
12458 050612' 013466'          .WORD  MSG12
12459 050614'          ESCAPE TST
12460 050614' 104410          TRAP   CSERHRD
12461 050616' 000232          .WORD  L10166-.
12462
12463
12464 ;CHECK THAT THE TRANSMIT STATUS WORD 1 HAS A NON-ZERO TDR VALUE IN IT
12465
12466 050620' 013701 000610'          40$:  MOV    PCBB+2,R1          ;GET MINIMUM TDR VALUE
12467 050624' 001004          BNE    45$                ;SHOULD BE A NON-ZERO VALUE
12468 050626'          ERRHRD  287,TDRCNT,MSG23  ;ERROR TDR COUNTER NOT COUNTING
12469 050626' 104456          TRAP   CSERHRD
12470 050630' 000437          .WORD  287
12471 050632' 004001'          .WORD  TDRCNT
12472 050634' 014442'          .WORD  MSG23
12473
12474 ;WRITE '1' TO CLEAR 'DNI'
12475
12476 050636' 004737 017362'          45$:  JSR    PC,CLRDN1          ;GO CLEAR DNI
12477 050642' 103004          BCC    46$                ;OK
12478 050644'          ERRHRD  288,TDRCNT,RACMG7  ;ERROR OCCURRED PRINT ERROR MESSAGE
12479 050644' 104456          TRAP   CSERHRD
12480 050646' 000440          .WORD  288
12481 050650' 004001'          .WORD  TDRCNT
12482 050652' 012670'          .WORD  RACMG7
12483 050654'          46$:

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 261  
CZUAAB.MAC 07-APR-83 17:03 TEST 44: TDR COUNTER TEST

```

12484 050654'          ENDSEG
12485 050654'
12486 050654' 104405          100018: TRAP C8ESEG
12487
12488      : INCREASE THE BYTE COUNT
12489
12490 050656' 005237 000606' 478: INC PCBB          : INCREASE BYTE COUNT BY 1
12491 050662'          BGNSEG
12492 050662' 104404          TRAP C8BSEG
12493
12494      : WAIT FOR THE MICROMONITOR, EXECUTE MICROTEST #2, AND WAIT FOR 'DNI'
12495
12496 050664' 004737 020060'      JSR PC,CHKMON          :WAIT FOR MICROMONITOR
12497 050670' 103006          BCC 50$              :OK
12498 050672'          ERRHRD 289,TDRCNT,MSG46      :PRINT ERROR
12499 050672' 104456          TRAP C8ERHRD
12500 050674' 000441          .WORD 289
12501 050676' 004001'        .WORD TDRCNT
12502 050700' 016666'        .WORD MSG46
12503 050702'          ESCAPE TST          :LEAVE TEST
12504 050702' 104410          TRAP C8ESCAPE
12505 050704' 000144          .WORD L10166-
12506 050706' 012777 000002 127422 508: MOV #2,BPCSRO          :TELL T11 TO EXECUTE MICROTEST #2
12507 050714' 012737 000176 000332' MOV #2+SECOND,METER      :WAIT FOR DNI
12508 050722' 004737 017316'      JSR PC,CHKDNI
12509 050726' 103021          BCC 55$
12510 050730' 004737 020132'      JSR PC,CHKINT          :SEE IF ANY ERROR INTERRUPTS OCCURRED
12511 050734' 103006          BCC 51$              :NO, OK
12512 050736'          ERRHRD 290,TDRCNT,MSG44      :PRINT ERROR MESSAGE
12513 050736' 104456          TRAP C8ERHRD
12514 050740' 000442          .WORD 290
12515 050742' 004001'        .WORD TDRCNT
12516 050744' 016442'        .WORD MSG44
12517 050746'          ESCAPE TST
12518 050746' 104410          TRAP C8ESCAPE
12519 050750' 000100          .WORD L10166-
12520 050752' 012702 000002 518: MOV #2,R2              :MICROTEST #2 IS HUNG
12521 050756'          ERRHRD 291,TDRCNT,MSG12
12522 050756' 104456          TRAP C8ERHRD
12523 050760' 000443          .WORD 291
12524 050762' 004001'        .WORD TDRCNT
12525 050764' 013466'        .WORD MSG12
12526 050766'          ESCAPE TST
12527 050766' 104410          TRAP C8ESCAPE
12528 050770' 000060          .WORD L10166-
12529
12530      : VERIFY THAT THE TRANSMIT STATUS WORD 1 TDR VALUE IS LARGER THAN THE PREVIOUS
12531      : VALUE
12532
12533 050772' 013702 000610' 558: MOV PCBB+2,R2          :GET NEW TDR VALUE
12534 050776' 020201          CMP R2,R1            :IS TDR GETTING LARGER?
12535 051000' 101006          BHI 60$              :YES
12536 051002'          ERRHRD 292,TDRCNT,MSG23      :NO ERROR
12537 051002' 104456          TRAP C8ERHRD
12538 051004' 000444          .WORD 292
12539 051006' 004001'        .WORD TDRCNT

```





65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 263  
CZUAAB.MAC 07-APR-83 17:03 TEST 45: RETRY LOGIC TEST

12567  
12568  
12569  
12570  
12571  
12572  
12573  
12574  
12575  
12576  
12577  
12578  
12579  
12580  
12581  
12582  
12583  
12584  
12585  
12586  
12587  
12588  
12589  
12590  
12591  
12592  
12593  
12594  
12595  
12596  
12597  
12598  
12599  
12600  
12601  
12602  
12603  
12604  
12605  
12606  
12607  
12608  
12609  
12610  
12611  
12612  
12613  
12614  
12615  
12616  
12617  
12618  
12619  
12620  
12621  
12622

051052'  
051052'

.SBTTL TEST 45: RETRY LOGIC TEST

```

:*****
:THE RETRY LOGIC IS ACTIVATED WHENEVER A COLLISION IS ENCOUNTERED DURING A
:TRANSMISSION ATTEMPT. THE LINK STOPS TRANSMISSION AND WAITS FOR A PERIOD OF
:TIME BEFORE ATTEMPTING TO RETRANSMIT.
:
:THE WAIT TIME IS AN INTEGRAL NUMBER OF 'SLOT TIMES'. THE NUMBER COMES FROM
:A RANDOM NUMBER GENERATOR. THE NUMBER OF SLOT TIMES IS NOT EXACTLY RANDOM
:SINCE THE RETRY LOGIC WAITS A GENERALLY INCREASING NUMBER OF SLOT TIMES BEFORE
:TRYING TO RETRANSMIT. THIS TEST WILL VERIFY THAT THE RETRY LOGIC IS CAPABLE OF
:GENERATING VARIABILITY IN THE DURATION OF THE RETRY WAIT TIMES.
:
:THIS TEST WILL USE MICROMODULE 'G' MICROTEST #3
:THE LINK MODULE HAS A DIAGNOSTIC MAINTENANCE FACILITY MAKING IT POSSIBLE TO
:SINGLE STEP THE RETRY LOGIC THROUGH THE MAXIMUM SIXTEEN RETRY STEPS. THIS
:FEATURE WILL ALSO MAKE IT POSSIBLE TO MEASURE THE RETRY WAIT INTERVAL.
:
:THE MICROCODE WILL SET THE COLLISION BIT IN THE LINK MODE REGISTER AND
:AND TRANSMIT A DATAGRAM IN LOOPBACK MODE. THE T-11 WILL COUNT WHILE WAITING
:FOR THE TRANSMIT STATE MACHINE TO INTERRUPT. THE ACCUMULATED COUNT SHOULD
:PROVIDE A MEASURE OF TIME TAKEN FOR THE TRANSMISSION ATTEMPT TO OCCUR. SINCE
:THE COLLISION BIT IS SET, THIS INTERVAL WILL INCLUDE THE RETRY WAIT INTERVAL.
:THE ACCUMULATED COUNT WILL BE WRITTEN BY THE MICROCODE TO THE PCBB.
:
:THE MICROTEST WILL BE EXECUTED 16 TIMES. AFTER EACH EXECUTION, THE COUNT WILL
:BE READ FROM THE PCBB AND STORED IN A TABLE. THE TABLE WILL BE SCANNED TO
:VERIFY THAT THEY ARE NOT ALL THE SAME.
:
:THE PCBB IS FORMATTED AS FOLLOWS:
:
:PCBB+0:      +-----+
:              |         BYTE COUNT         |
:              +-----+
:PCBB+2:      +-----+
:              |   TRANSMIT WAIT COUNT   |
:              +-----+
:
:TEST SEQUENCE:
: 1-LOAD MICROMODULE 'G' IF NOT ALREADY DONE SO
: 2-PLACE A MINIMUM BYTE COUNT IN PCBB+0
: 3-CLEAR PCBB+2
: 4-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
: 5-SELECT MICROTEST #3
: 6-WAIT FOR 'DNI'
: 7-WRITE '1' TO CLEAR 'DNI'
: 8-READ PCBB+2 AND PUT COUNT IN TABLE
: 9-REPEAT STEPS 2-8 15 TIMES
:10-VERIFY NO 10 CONSECUTIVE ENTRIES IN THE TABLE ARE THE SAME.
:*****

```

BGNTST

T45::

:CHECK TO SEE IF MODULE 'G' HAS BEEN LOADED. IF NOT LOAD IT INTO  
:THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.

```

12623      :AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
12624      :
12625      BGNSEG
12626      051052' 104404      TRAP      CSBSEG
12627      051054' 022737 000107 000326'      CMP      #'G,MICRO      ;HAS MICROCODE MODULE 'G' BEEN LOADED
12628      051062' 001004      BNE      SS      ;NO
12629      051064' 122777 000001 127246      CMPB     #INMON,BPCSR1 ;YES, IS THE MICROMONITOR ACTIVE?
12630      051072' 001440      BEQ      20$      ;YES SKIP LOADING THE MICROMODULE
12631      051074' 012737 000107 000326' 5$:      MOV      #'G,MICRO      ;GO LOAD MICRO MODULE 'G'
12632      051102' 004737 020340'      JSR      PC,LODMIC
12633      051106' 103002      BCC      10$      ;OK
12634      051110'      ESCAPE   TST
12635      051110' 104410      TRAP      CS$ESCAPE
12636      051112' 000352      .WORD    L10167-.
12637      051114' 012737 000176 000332' 10$:      MOV      #2*SECOND,METER ;WAIT FOR THE MICROMONITOR
12638      051122' 004737 017316'      JSR      PC,CHKDNI
12639      051126' 103022      BCC      20$      ;OK
12640      051130' 012737 001000' 000310'      MOV      #$DNI,BITNAM
12641      051136' 012737 001277' 000312'      MOV      #$SNSET,BITSTA
12642      051144' 012737 001342' 000314'      MOV      #$AFTER,PWHEN
12643      051152' 012737 001357' 000316'      MOV      #$GTCMD,PCOMND
12644      051160'      ERRHRD   294,RETLOG,MSG1
12645      051160' 104456      TRAP      CSERHRD
12646      051162' 000446      .WORD    294
12647      051164' 004031'      .WORD    RETLOG
12648      051166' 012716'      .WORD    MSG1
12649      051170'      ESCAPE   TST
12650      051170' 104410      TRAP      CS$ESCAPE
12651      051172' 000272      .WORD    L10167-.
12652      051174' 004737 017362'      20$:      JSR      PC,CLRDNI      ;CLEAR DNI BIT
12653      051200' 103006      BCC      25$
12654      051202'      ERRHRD   295,RETLOG,RACMG7 ;DNI DID NOT CLEAR!
12655      051202' 104456      TRAP      CSERHRD
12656      051204' 000447      .WORD    295
12657      051206' 004031'      .WORD    RETLOG
12658      051210' 012670'      .WORD    RACMG7
12659      051212'      ESCAPE   TST
12660      051212' 104410      TRAP      CS$ESCAPE
12661      051214' 000250      .WORD    L10167-.
12662      051216'      25$:
12663      051216'      ENDSEG
12664      051216'      10000$:
12665      051216' 104405      TRAP      CS$ESEG
12666      :
12667      :WRITE A BYTE COUNT TO PCBB+0, CLEAR PCBB+2, GET POINTER TO TOP OF TABLE
12668      :INITALIZE A COUNTER TO 16
12669      :
12670      051220' 012737 000100 000606'      MOV      #MINBYT,PCBB ;BEGIN WITH MINIMUM BYTE COUNT
12671      051226' 005037 000610'      CLR      PCBB+2 ;THIS IS WHERE MICROCODE WILL PUT...
12672      :
12673      051232' 012702 000626'      MOV      #CNTTAB,R2 ;THE RETRY WAIT INTERVAL
12674      051236' 012701 000020      MOV      #16,,R1 ;GET POINTER TO WAIT INTERVAL STORAGE AREA
12675      :
12676      :WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR. EXECUTE MICROTEST #3 BY
12677      :LOADING THE COMMAND FIELD OF PCSRO WITH A 3. WAIT FOR 'DNI'
12678      :
    
```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 265  
 CZUAAB.MAC 07-APR-83 17:03 TEST 45: RETRY LOGIC TEST

```

12679 051242'          BGNSEG
12680 051242' 104404
12681 051244' 004737 020060'          JSR    PC,CHKMON          ;WAIT FOR MICROMONITOR
12682 051250' 103006          BCC    358                ;OK
12683 051252'          ERRHRD 296,RETLOG,MSG46 ;PRINT ERROR
12684 051252' 104456
12685 051254' 000450          TRAP   CSBSEG
12686 051256' 004031'          .WORD 296
12687 051260' 016666'          .WORD RETLOG
12688 051262'          ESCAPE TST          ;LEAVE TEST
12689 051262' 104410          TRAP   CSERHRD
12690 051264' 000200          .WORD 296
12691 051266' 012777 000003 127042 358:  MOV    #3,@PCSR0          ;TELL T11 TO EXECUTE MICROTEST #3
12692 051274' 012737 000770 000332'  MOV    #10*SECOND,METER ;WAIT FOR DNI
12693 051302' 004737 017316'          JSR    PC,CHKDNI
12694 051306' 103021          BCC    408
12695 051310' 004737 020132'          JSR    PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
12696 051314' 103006          BCC    368                ;NO, OK
12697 051316'          ERRHRD 297,RETLOG,MSG44 ;PRINT ERROR MESSAGE
12698 051316' 104456          TRAP   CSERHRD
12699 051320' 000451          .WORD 297
12700 051322' 004031'          .WORD RETLOG
12701 051324' 016442'          .WORD MSG44
12702 051326'          ESCAPE TST
12703 051326' 104410          TRAP   CSERHRD
12704 051330' 000134          .WORD 297
12705 051332' 012702 000003          368:  MOV    #3,R2                ;MICROTEST #3 IS HUNG
12706 051336'          ERRHRD 298,RETLOG,MSG12
12707 051336' 104456          TRAP   CSERHRD
12708 051340' 000452          .WORD 298
12709 051342' 004031'          .WORD RETLOG
12710 051344' 013466'          .WORD MSG12
12711 051346'          ESCAPE TST
12712 051346' 104410          TRAP   CSERHRD
12713 051350' 000114          .WORD 298
12714          ;
12715          ;WRITE '1' TO CLEAR 'DNI'
12716          ;
12717 051352' 004737 017362'          408:  JSR    PC,CLRDN1          ;GO CLEAR DNI
12718 051356' 103006          BCC    458                ;OK
12719 051360'          ERRHRD 299,RETLOG,RACMG7 ;ERROR
12720 051360' 104456          TRAP   CSERHRD
12721 051362' 000453          .WORD 299
12722 051364' 004031'          .WORD RETLOG
12723 051366' 012670'          .WORD RACMG7
12724 051370'          ESCAPE TST          ;LEAVE
12725 051370' 104410          TRAP   CSERHRD
12726 051372' 000072          .WORD 299
12727          ;
12728          ;STORE THE VALUE FORM PCBB+2 IN THE TABLE AND BUMP THE POINTER TO NEXT ENTRY
12729          ;REPEAT THE TEST UNTIL ALL 16 ENTRIES ARE OBTAINED
12730          ;
12731 051374' 013722 000610'          458:  MOV    PCBB+2,(R2)+        ;STORE COUNTER VALUE IN THE TABLE
12732 051400' 005301          DEC    R1                  ;HAVE WE STORED 16 VALUES?
12733 051402' 001331          BNE    358                ;NOT YET
12734          ;

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 266  
 CZUAAB.MAC 07-APR-83 17:03 TEST 45: RETRY LOGIC TEST

```

12735                                     :TREAT THE TABLE AS 7 GROUPS OF 10 ENTRIES. VERIFY THAT ALL 10 ENTRIES OF EACH
12736                                     :GROUP ARE NOT THE SAME.
12737                                     .
12738 051404' 012701 000626'                MOV    #CNTTAB,R1                :POINT TO TOP OF TABLE
12739 051410' 010102                SOS:  MOV    R1,R2                :GET POINTER TO FIRST ELEMENT IN THIS
12740                                     :GROUP
12741 051412' 010204                MOV    R2,R4                :GET POINTER TO SECOND ELEMENT IN THIS
12742                                     :GROUP
12743 051414' 062704 000002                ADD    #2,R4
12744 051420' 012703 000007                MOV    #7.,R3                :NUMBER OF GROUPS OF 10 TO COMPARE
12745 051424' 022224                55$:  CMP    (R2)+,(R4)+        :IS THIS PAIR THE SAME?
12746 051426' 001010                BNE    60$                    :NO, START A NEW GROUP TO CHECK
12747 051430' 005303                DEC    R3                    :HAVE WE CHECKED ALL 10 VALUES IN THIS
12748                                     :GROUP?
12749 051432' 001374                BNE    55$                    :NOT YET
12750 051434'                                ERRHRD 300,RETLOG,MSG24        :YES, PRINT ERROR MESSAGE AND DUMP TABLE
12751 051434' 104456                                TRAP   CSERHRD
12752 051436' 000454                                .WORD 300
12753 051440' 004031'                                .WORD RETLOG
12754 051442' 014464'                                .WORD MSG24
12755 051444'                                ESCAPE TST
12756 051444' 104410                                TRAP   CSESCAPE
12757 051446' 000016                                .WORD L10167-.
12758 051450' 062701 000002                60$:  ADD    #2,R1                :POINT TO NEXT GROUP OF 10 ELEMENTS
12759 051454' 020127 000644'                CMP    R1,#CNTTAB+16        :CHECKED ALL GROUPS OF 10?
12760 051460' 001353                BNE    50$                    :NOT YET
12761 051462'                                ENDSEG
12762 051462'                                10001$: TRAP   CSESEG
12763 051462' 104405                                L10167: TRAP   CSETST
12764 051464'                                ENDTST
12765 051464'
12766 051464' 104401

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 267  
CZUAAB.MAC 07-APR-83 17:03 TEST 45: RETRY LOGIC TEST

12767  
12768  
12769  
12770  
12771  
12772  
12773  
12774  
12775  
12776  
12777  
12778  
12779  
12780  
12781  
12782  
12783  
12784  
12785  
12786  
12787  
12788  
12789  
12790  
12791  
12792  
12793  
12794  
12795  
12796  
12797  
12798  
12799  
12800  
12801  
12802  
12803  
12804  
12805  
12806  
12807  
12808  
12809  
12810  
12811  
12812  
12813  
12814  
12815  
12816  
12817  
12818  
12819  
12820  
12821  
12822

051466'  
051466'  
051466' 005737 000674'  
051472' 001002  
051474'  
051474' 104432  
051476' 001042  
051500' 004737 020166'  
051504' 103002  
051506'  
051506' 104410  
051510' 001030  
051512' 012777 000606' 126622  
051520' 005077 126620  
051524' 012777 000001 126604  
051532' 012737 000077 000332'  
051540' 004737 017316'  
051544' 103022  
051546' 012737 001000' 000310'  
051554' 012737 001277' 000312'  
051562' 012737 001342' 000314'  
051570' 012737 001350' 000316'  
051576'  
051576' 104456  
051600' 000455  
051602' 004061'  
051604' 012716'  
051606'  
051606' 104410  
051610' 000730  
051612' 004737 017362'  
051616' 103006

.SBTTL TEST 46: PRINT DEVICE PARAMETERS TEST

.....

THIS TEST PRINTS THE DEFAULT PHYSICAL ADDRESS, THE MICROCODE REVISION AND THE SWITCH PACK SETTINGS.

TEST SEQUENCE:

1. READ DEFAULT PHYSICAL ADDRESS
2. READ MICROCODE REVISION
3. READ SWITCH PACK SETTINGS
4. PRINT

NOTE:

THIS TEST IS ONLY EXECUTED ONCE FOR EACH UNIT REGARDLESS OF THE PASS #

.....

BGNTST

```

T46::
TST FRSTIM ; RUN THIS TEST ?
BNE S8 ; YES
EXIT TST ; NO, EXIT

S8: JSR PC,REUNA ;GU RESET UNA
BCC 208 ;OK
ESCAPE TST ; ABORT TEST

TRAP .WORD CSEXIT
L10170-.

208: MOV #PCBB,@PCSR2 ;LOAD PCSR2 WITH PORT CONTROL BLOCK ADR
CLR @PCSR3 ;LOAD PCSR3 WITH 0
MOV #GETPCB,@PCSR0 ; ISSUE GET PCBB PORT COMMAND
MOV #1*SECOND,METER ;PUT SOME TIME ON THE METER
JSR PC,CHKDNI ; DNI?
BCC 408 ; YES
;ERROR DNI FAILED TO SET!
;FORMAT ERROR MESSAGE

MOV #SDNI,BITNAM
MOV #SNSET,BITSTA
MOV #AFTER,PWHEN
MOV #SGTPCB,PCOMND
ERRHRD 301,PRTPAR,MSG1

TRAP .WORD CSEHRD
WORD 301
WORD PRTPAR
WORD MSG1

ESCAPE TST ; AND ABORT TEST

TRAP .WORD C$ESCAPE
WORD L10170-.

408: JSR PC,CLR DNI ; WRITE ONE TO CLEAR DNI
; ERROR ?
BCC 508 ; NO

```

HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 268  
 CZUAAB.MAC 07-APR-83 17:03 TEST 46: PRINT DEVICE PARAMETERS TEST

```

12823                                     ;ERROR DNI FAILED TO CLEAR!
12824 051620'                               ERRHRD 302,PRTPAR,RACMG7
12825 051620' 104456                         TRAP      CSERHRD
12826 051622' 000456                         .WORD    302
12827 051624' 004061'                       .WORD    PRTPAR
12828 051626' 012670'                       .WORD    RACMG7
12829 051630'                               ESCAPE TST                               ; AND ABORT TEST
12830 051630' 104410                         TRAP      CSERHRD
12831 051632' 000706                         .WORD    L10170-.
12832                                     ;
12833                                     ;:READ DEFAULT PHYSICAL ADDRESS
12834                                     ;
12835 051634' 012737 000002 000606' 50$:  MOV    #R0PA,PCBB                ;LOAD PCBB WITH READ DEFAULT PHY ADR
12836 051642' 012737 000176 000332'        MOV    #2*SECOND,METER            ;PUT SOME TIME ON THE METER
12837 051650' 012777 000002 126460        MOV    #GETCMD,BPCSRO            ; ISSUE GET_CMD PORT COMMAND
12838 051656' 004737 017316'              JSR    PC,CHKDNI                 ; DNI ?
12839 051662' 103022                       BCC    60$                       ; YES
12840                                     ;:ERROR DNI FAILED TO SET!
12841 051664' 012737 001000' 000310'      MOV    #SDNI,BITNAM
12842 051672' 012737 001277' 000312'      MOV    #SNSET,BITSTA
12843 051700' 012737 001342' 000314'      MOV    #SAFTER,PWEN
12844 051706' 012737 001357' 000316'      MOV    #SGTCMD,PCOMND
12845 051714'                               ERRHRD 303,PRTPAR,MSG1 ;PRINT ERROR MESSAGE
12846 051714' 104456                         TRAP      CSERHRD
12847 051716' 000457                         .WORD    303
12848 051720' 004061'                       .WORD    PRTPAR
12849 051722' 012716'                       .WORD    MSG1
12850 051724'                               ESCAPE TST                               ; AND ABORT TEST
12851 051724' 104410                         TRAP      CSERHRD
12852 051726' 000612                         .WORD    L10170-.
12853                                     ;
12854 051730' 004737 017362'              60$:  JSR    PC,CLRDN1            ; WRITE ONE TO CLEAR DNI
12855                                     ;: ERROR ?
12856 051734' 103006                       BCC    70$                       ; NO
12857                                     ;:ERROR DNI FAILED TO CLEAR!
12858 051736'                               ERRHRD 304,PRTPAR,RACMG7 ;PRINT ERROR MESSAGE
12859 051736' 104456                         TRAP      CSERHRD
12860 051740' 000460                         .WORD    304
12861 051742' 004061'                       .WORD    PRTPAR
12862 051744' 012670'                       .WORD    RACMG7
12863 051746'                               ESCAPE TST                               ; AND ABORT TEST
12864 051746' 104410                         TRAP      CSERHRD
12865 051750' 000570                         .WORD    L10170-.
12866                                     ;
12867                                     ;:MOVE DEFAULT PHYSICAL ADDRESS FROM PCBB -> DPA
12868                                     ;
12869 051752' 013737 000610' 052542' 70$:  MOV    PCBB+2,DPA
12870 051760' 013737 000612' 052544'      MOV    PCBB+4,DPA+2
12871 051766' 013737 000614' 052546'      MOV    PCBB+6,DPA+4
12872                                     ;
12873                                     ;:LOAD ASCII MESSAGE (DEFADR)
12874                                     ;
12875 051774' 004737 017116'              JSR    PC,HEXDPA                ; CONVERT TO ASCII HEX
12876                                     ;
12877                                     ;:READ MICROCODE REVISION
12878                                     ;

```

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 269  
 CZUAAB.MAC 07-APR-83 17:03 TEST 46: PRINT DEVICE PARAMETERS TEST

```

12879 052000' 012737 000016 000606' 100$: MOV #RPS,PCBB ;LOAD PCBB WITH READ PORT STATUS FUNCTION
12880 052006' 012737 000176 000332' MOV #2*SECOND,METER ;PUT SOME TIME ON THE METER
12881 052014' 012777 000002 126314 MOV #GETCMD,@PCRSO ;ISSUE GET_CMD PORT COMMAND
12882 052022' 004737 017316' JSR PC,CHKDNI ;DNI ?
12883 052026' 103022 BCC 110$ ;YES
12884 ;ERROR DNI FAILED TO SET
12885 052030' 012737 001000' 000310' MOV #SDNI,BITNAM
12886 052036' 012737 001277' 000312' MOV #SNSET,BITSTA
12887 052044' 012737 001342' 000314' MOV #SAFTER,PWMEN
12888 052052' 012737 001357' 000316' MOV #SGTCMD,PCOMND
12889 052060' ERRHRD 305,PRTPAR,MSG1 ;PRINT ERROR MESSAGE
12890 052060' 104456 TRAP CSERHRD
12891 052062' 000461 .WORD 305
12892 052064' 004061' .WORD PRTPAR
12893 052066' 012716' .WORD MSG1
12894 052070' ESCAPE TST ;AND ABORT TEST
12895 052070' 104410 TRAP C$ESCAPE
12896 052072' 000446 .WORD L10170-.
12897
12898 052074' 004737 017362' 110$: JSR PC,CLRDN1 ;WRITE ONE TO CLEAR DNI
12899 BCC 120$ ;ERROR ?
12900 052100' 103006 ;NO
12901 ERRHRD 306,PRTPAR,RACMG7 ;ERROR DNI FAILED TO SET
12902 ;PRINT ERROR MESSAGE
12903 052102' 104456 TRAP CSERHRD
12904 052104' 000462 .WORD 306
12905 052106' 004061' .WORD PRTPAR
12906 052110' 012670' .WORD RACMG7
12907 052112' ESCAPE TST ;AND ABORT TEST
12908 052112' 104410 TRAP C$ESCAPE
12909 052114' 000424 .WORD L10170-.
12910
12911 ;MOVE MICROCODE REVISION FROM PCBB -> RREV
12912
12913 052116' 013737 000610' 052550' 120$: MOV PCBB+2,RREV
12914 052124' 042737 177700 052550' BIC #177700,RREV ;MASK RREV
12915
12916 ;READ SWITCH PACK
12917
12918 052132' 012737 000020 000606' 130$: MOV #DIM,PCBB ;LOAD DUMP INTERNAL MEMORY FUNCTION
12919 052140' 012737 000616' 000610' MOV #UDBB,PCBB+2 ;GIVE THE UNIBUS DATA BLOCK BASE ADR
12920 052146' 005037 000612' CLR PCBB+4
12921 052152' 012737 000002 000616' MOV #2,UDBB ;SETUP TO LOAD 2 BYTES
12922 052160' 012737 052552' 000620' MOV #SUPACK,UDBB+2 ;LOAD ADDRESS
12923 052166' 005037 000622' CLR UDBB+4
12924 052172' 013737 000334' 000624' MOV SWADDR,UDBB+6 ;LOAD INTERNAL ADDRESS
12925 052200' 012737 000176 000332' MOV #2*SECOND,METER ;PUT SOME TIME ON THE METER
12926 052206' 012777 000002 126122 MOV #GETCMD,@PCRSO ;ISSUE GET COMMAND PORT COMMAND
12927 052214' 004737 017316' JSR PC,CHKDNI ;DNI ?
12928 052220' 103022 BCC 140$ ;YES
12929 ;ERROR DNI FAILED TO SET
12930 052222' 012737 001000' 000310' MOV #SDNI,BITNAM
12931 052230' 012737 001277' 000312' MOV #SNSET,BITSTA
12932 052236' 012737 001342' 000314' MOV #SAFTER,PWMEN
12933 052244' 012737 001357' 000316' MOV #SGTCMD,PCOMND
12934 052252' ERRHRD 307,PRTPAR,MSG1

```



65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 270  
 CZUAAB.MAC 07-APR-83 17:03 TEST 46: PRINT DEVICE PARAMETERS TEST

12935	052252'	104456				TRAP	C\$ERHRD
12936	052254'	000463				.WORD	307
12937	052256'	004061'				.WORD	PRTPAR
12938	052260'	012716'				.WORD	MSG1
12939	052262'			ESCAPE TST			; AND ABORT TEST
12940	052262'	104410				TRAP	C\$ESCAPE
12941	052264'	000254				.WORD	L10170-
12942				:			
12943	052266'	004737	017362'	140:	JSR	PC,CLRDMI	; WRITE ONE TO CLEAR DMI
12944							; ERROR ?
12945	052272'	103006			BCC	150\$	; NO
12946	052274'				ERRHRD	308,PRTPAR,RACMG7	
12947	052274'	104456					TRAP
12948	052276'	000464				.WORD	C\$ERHRD
12949	052300'	004061'				.WORD	308
12950	052302'	012670'				.WORD	PRTPAR
12951	052304'			ESCAPE TST			RACMG7
12952	052304'	104410					; AND ABORT TEST
12953	052306'	000232				TRAP	C\$ESCAPE
12954						.WORD	L10170-
12955				:			
12956				:			; GET SWITCH PACK INFO READY TO PRINT
12957	052310'	013704	052552'	150:	MOV	SWPACK,R4	; SWITCH PACK -> R4
12958	052314'	042704	167777		BIC	#167777,R4	; MASK BIT 12
12959	052320'	012700	000013		MOV	#11,,R0	; SHIFT BIT FOR INDEX
12960	052324'	006204		160:	ASR	R4	
12961	052326'	005300			DEC	R0	
12962	052330'	001375			BNE	160\$	
12963	052332'	062704	052672'		ADD	#LPTBL,R4	; INDEX INTO LOOP TABLE
12964	052336'	011437	052554'		MOV	(R4),LPMSG	; LOAD INTO LOOP MESSAGE
12965	052342'	013704	052552'	170:	MOV	SWPACK,R4	; SWITCH PACK -> R4
12966	052346'	042704	171777		BIC	#171777,R4	; MASK BITS 10 AND 11
12967	052352'	012700	000011		MOV	#9,,R0	; SHIFT BITS FOR INDEX
12968	052356'	006204		180:	ASR	R4	
12969	052360'	005300			DEC	R0	
12970	052362'	001375			BNE	180\$	
12971	052364'	062704	052676'		ADD	#BTBL,R4	; INDEX INTO BOOT TABLE
12972	052370'	011437	052556'		MOV	(R4),BTMSG	; LOAD INTO BOOT MESSAGE
12973				:			
12974				:			; PRINT
12975				:			
12976	052374'				PRINTB	#FORM28,#DEFHDR	; PRINT DEFAULT PHYSICAL ADDRESS
12977	052374'	012746	052562'				MOV
12978	052400'	012746	006236'				MOV
12979	052404'	012746	000002				MOV
12980	052410'	010600					MOV
12981	052412'	104414					TRAP
12982	052414'	062706	000006				ADD
12983	052420'				PRINTB	#FORM64,RREV	; PRINT MICROCODE REV
12984	052420'	013746	052550'				MOV
12985	052424'	012746	010773'				MOV
12986	052430'	012746	000002				MOV
12987	052434'	010600					MOV
12988	052436'	104414					TRAP
12989	052440'	062706	000006				ADD
12990	052444'				PRINTB	#FORM28,#SWHDR	; PRINT SWITCH PACK HEADER

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 271  
 CZUAAB.MAC 07-APR-83 17:03 TEST 46: PRINT DEVICE PARAMETERS TEST

12991	052444'	012746	052706'				
12992	052450'	012746	006236'			MOV	#SUMDR,-(SP)
12993	052454'	012746	000002			MOV	#FORM28,-(SP)
12994	052460'	010600				MOV	#2,-(SP)
12995	052462'	104414				MOV	SP,R0
12996	052464'	062706	000006			TRAP	CSPNTB
12997	052470'			PRINTB	#FORM28.LPMSG	ADD	#6,SP
12998	052470'	013746	052554'				
12999	052474'	012746	006236'			MOV	LPMSG,-(SP)
13000	052500'	012746	000002			MOV	#FORM28,-(SP)
13001	052504'	010600				MOV	#2,-(SP)
13002	052506'	104414				MOV	SP,R0
13003	052510'	062706	000006			TRAP	CSPNTB
13004	052514'			PRINTB	#FORM28.BTMSG	ADD	#6,SP
13005	052514'	013746	052556'				
13006	052520'	012746	006236'			MOV	BTMSG,-(SP)
13007	052524'	012746	000002			MOV	#FORM28,-(SP)
13008	052530'	010600				MOV	#2,-(SP)
13009	052532'	104414				MOV	SP,R0
13010	052534'	062706	000006			TRAP	CSPNTB
13011						ADD	#6,SP
13012	052540'			2508:			
13013	052540'				ENDTST		
13014	052540'					L10170:	
13015	052540'	104401				TRAP	CSETST

65HARDWARE TESTS MACY11 30A(1052) 07-APR-83 17:13 PAGE 272  
CZUAAB.MAC 07-APR-83 17:03 TEST 46: PRINT DEVICE PARAMETERS TEST

```

13016          :LOCAL STORAGE FOR TEST 41
13017 052542' 000000 DPA::          .WORD 0          : DEFAULT PHYSICAL ADDRESS (15:00)
13018 052544' 000000          .WORD 0          : DEFAULT PHYSICAL ADDRESS (31:16)
13019 052546' 000000          .WORD 0          : DEFAULT PHYSICAL ADDRESS (47:32)
13020          :
13021 052550' 000000 RREV::        .WORD 0          : MICROCODE REVISION
13022          :
13023 052552' 000000 SWPACK::      .WORD 0          : SWITCH PACK CONTENTS
13024 052554' 000000 LPMSG::      .WORD 0          : LOOPBACK MESSAGE ADDRESS
13025 052556' 000000 BTMSG::      .WORD 0          : BOOT MESSAGE ADDRESS
13026          :
13027 052560' 000      HEXDAT::      .BYTE 0          : HEX DATA FOR CONVERSION
13028 052561' 000      HEXVAL::      .BYTE 0          : ASCII HEX VALUE
13029          :
13030 052562' 005015 052105 042510 DEFHDR::      .ASCII <15><12>/ETHERNET DEFAULT ADDRESS (HEX): /
13031 052570' 047122 052105 042040
13032 052576' 043105 052501 052114
13033 052604' 040440 042104 042522
13034 052612' 051523 024040 042510
13035 052620' 024530 020072 040
13036 052625' 040      040      DEFADR::      .ASCII / /
13037 052627' 055      .ASCII /- /
13038 052630' 020040      .ASCII / /
13039 052632' 055      .ASCII /- /
13040 052633' 040      040      .ASCII / /
13041 052635' 055      .ASCII /- /
13042 052636' 020040      .ASCII / /
13043 052640' 055      .ASCII /- /
13044 052641' 040      040      .ASCII / /
13045 052643' 055      .ASCII /- /
13046 052644' 020040      .ASCII / /
13047 052646' 005015 000      .ASCIZ <15><12>
13048          :
13049 052651' 060      HEXTBL::      .ASCII /0/
13050 052652' 061      .ASCII /1/
13051 052653' 062      .ASCII /2/
13052 052654' 063      .ASCII /3/
13053 052655' 064      .ASCII /4/
13054 052656' 065      .ASCII /5/
13055 052657' 066      .ASCII /6/
13056 052660' 067      .ASCII /7/
13057 052661' 070      .ASCII /8/
13058 052662' 071      .ASCII /9/
13059 052663' 101      .ASCII /A/
13060 052664' 102      .ASCII /B/
13061 052665' 103      .ASCII /C/
13062 052666' 104      .ASCII /D/
13063 052667' 105      .ASCII /E/
13064 052670' 106      .ASCII /F/
13065          .EVEN
13066          :
13067          :
13068          :LOOP MESSAGE TABLE
13069 052672' 052740' LPTBL::      .WORD LPMSG0
13070 052674' 052777'      .WORD LPMSG1
13071          :BOOT MESSAGE TABLE

```

13072	052676'	053035'			BTTBL::	.WORD	BTMSG0	
13073	052700'	053073'				.WORD	BTMSG1	
13074	052702'	053126'				.WORD	BTMSG2	
13075	052704'	053172'				.WORD	BTMSG3	
13076					:ASCII MESSAGES			
13077	052706'	005015	053523	052111	SUMDR::	.ASCII	<15><12>/SWITCH PACK SET FOR :/	
13078	052714'	044103	050040	041501				
13079	052722'	020113	042523	020124				
13080	052730'	047506	020122	072				
13081	052735'	015	000012					
13082	052740'	020040	020040	051440	LPMSG0::	.ASCIZ	<15><12>	
13083	052746'	046105	020106	042524		.ASCII	/ SELF TEST LOOP DISABLED/	
13084	052754'	052123	046040	047517				
13085	052762'	020120	044504	040523				
13086	052770'	046102	042105					
13087	052774'	005015	000					
13088	052777'	040	020040	020040	LPMSG1::	.ASCIZ	<15><12>	
13089	053004'	042523	043114	052040		.ASCII	/ SELF TEST LOOP ENABLED/	
13090	053012'	051505	020124	047514				
13091	053020'	050117	042440	040516				
13092	053026'	046102	042105					
13093	053032'	005015	000					
13094	053035'	040	020040	020040	BTMSG0::	.ASCIZ	<15><12>	
13095	053042'	047516	051040	046505		.ASCII	/ NO REMOTE BOOT ENABLED/	
13096	053050'	052117	020105	047502				
13097	053056'	052117	042440	040516				
13098	053064'	046102	042105					
13099	053070'	005015	000					
13100	053073'	040	020040	020040	BTMSG1::	.ASCIZ	<15><12>	
13101	053100'	042522	047515	042524		.ASCII	/ REMOTE BOOT ENABLED/	
13102	053106'	041040	047517	020124				
13103	053114'	047105	041101	042514				
13104	053122'	104						
13105	053123'	015	000012					
13106	053126'	020040	020040	051040	BTMSG2::	.ASCIZ	<15><12>	
13107	053134'	046505	052117	020105		.ASCII	/ REMOTE BOOT ENABLED WITH ROM/	
13108	053142'	047502	052117	042440				
13109	053150'	040516	046102	042105				
13110	053156'	053440	052111	020110				
13111	053164'	047522	115					
13112	053167'	015	000012					
13113	053172'	020040	020040	051040	BTMSG3::	.ASCIZ	<15><12>	
13114	053200'	046505	052117	020105		.ASCII	/ REMOTE BOOT AND POWER UP BOOT BOTH ENABLED/	
13115	053206'	047502	052117	040440				
13116	053214'	042116	050040	053517				
13117	053222'	051105	052440	020120				
13118	053230'	047502	052117	041040				
13119	053236'	052117	020110	047105				
13120	053244'	041101	042514	104				
13121	053251'	015	000012			.ASCIZ	<15><12>	
13122						.EVEN		



67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 275  
CZUAAB.MAC 07-APR-83 17:03 HARDWARE PARAMETER CODING SECTION

```
13167 053366' 000100          SPATCH::.BLKW 100
13168
13169
13170          000000'          .CSECT M!CRA
13171          000000'          .CSECT MICKB
13172          000000'          .CSECT MICRC
13173          000000'          .CSECT MICRD
13174          000000'          .CSECT MICRE
13175          000000'          .CSECT MICRF
13176          000000'          .CSECT MICRG
13177          000000'          .CSECT NOMORE
13178 000000'
13179          000001          .END
          ENDMOD
```

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 277  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

ADR = 000020 G		1659#																	
ADRERR= 000001 G		1694#	6910	7526															
ASKCSR 053276R	002	13143	13156#																
ASKVEC 053331R	002	13148	13161#																
ASSEMB= 000010		1416																	
BITNAM 000310RG	002	1819#	2996	3031	3131	3299	3703	4342*	4349*	4356*	4363*	4370*	4377*	4384*					
		4386	4534*	4541	5392*	5421*	5446*	5501*	5526*	5545*	5746*	5817*	6237*	6303*					
		6434*	6477*	6538*	6664*	6711*	6759*	6838*	6985*	7115*	7191*	7192*	7230*	7231*					
		7296*	7442*	7607*	7766*	7916*	8054*	8192*	8340*	8502*	8655*	8832*	8914*	8927*					
		8936*	8948*	8957*	8966*	9059*	9228*	9401*	9566*	9755*	9918*	10157*	10380*	10535*					
		10695*	10862*	11034*	11202*	11391*	11599*	11814*	12050*	12393*	12640*	12807*	12841*	12885*					
		12930*																	
BITNUM 000306RG	002	1818#	2931	5064*	5081*	5167*	5184*	5201*	5218*	5235*	7159*	7189	7241*						
BITSTA 000312RG	002	1820#	2930	2995	3030	4535*	4540	5063*	5166*	5393*	5422*	5439*	5492*	5525*					
		5544*	5747*	5818*	6238*	6304*	6435*	6478*	6539*	6665*	6712*	6760*	6839*	6986*					
		7116*	7157*	7297*	7443*	7608*	7767*	7917*	8055*	8193*	8341*	8503*	8656*	8833*					
		9060*	9229*	9402*	9567*	9756*	9919*	10158*	10381*	10536*	10696*	10863*	11035*	11203*					
		11392*	11600*	11815*	12051*	12394*	12641*	12808*	12842*	12886*	12931*								
BIT0 = 000001 G		1632#	5590	7686															
BIT00 = 000001 G		1621#	1632	1719	1733	1736	1757	1759	1761	1762	1764	5627							
BIT01 = 000002 G		1620#	1631	1734	1735	1736	1757	1760	1761	1763	1764								
BIT02 = 000004 G		1619#	1630	1716	1735	1757	1762	1763	1764										
BIT03 = 000010 G		1618#	1629	1714	1746														
BIT04 = 000020 G		1617#	1628	1712															
BIT05 = 000040 G		1616#	1627	1710	1722														
BIT06 = 000100 G		1615#	1626	1708	1721														
BIT07 = 000200 G		1614#	1625	1706	1720	1744													
BIT08 = 000400 G		1613#	1624	1718	1770														
BIT09 = 001000 G		1612#	1623	1769															
BIT1 = 000002 G		1631#																	
BIT10 = 002000 G		1611#	1715	1749	1753	7158	11510	12141	12213	12284	12292								
BIT11 = 004000 G		1610#	1713	1750	1753	11285	12133	12139	12211	12290									
BIT12 = 010000 G		1609#	1711	1730	1751	1753	12137	12206	12209	12288									
BIT13 = 020000 G		1608#	1709	1729	1752	1753	8925	8946											
BIT14 = 040000 G		1607#	1707	1728	1740	1768	5219	8955	11293	12135	12207	12283	12286						
BIT15 = 100000 G		1606#	1705	1727	1739	1767	5236	7841	8934	8964	12083								
BIT2 = 000004 G		1630#	7841	12083															
BIT3 = 000010 G		1629#	5494																
BIT4 = 000020 G		1628#	5065	5168	5438	12083													
BIT5 = 000040 G		1627#	5185																
BIT6 = 000100 G		1626#	1687	5202	11432														
BIT7 = 000200 G		1625#																	
BIT8 = 000400 G		1624#	1747	1753															
BIT9 = 001000 G		1623#	1748	1753	5082														
BNAMT0 000360RG	002	1844#	5435	7191	7230														
BNAMT1 000420RG	002	1861#	5490																
BNAMT2 000460RG	002	1879#	8909																
BOE = 000400 G		1663#																	
BTMSG 052556RG	002	12972*	13005	13025#															
BTMSG0 053035RG	002	13072	13094#																
BTMSG1 053073RG	002	13073	13100#																
BTMSG2 053126RG	002	13074	13106#																
BTMSG3 053172RG	002	13075	13113#																
BTTBL 052676RG	002	12971	13072#																
CHKDNI 017316RG	002	4273#	4531	4652	4680	4879	5389	5742	5814	6234	6300	6431	6474	6536					
		6661	6708	6756	6836	6887	6982	7112	7294	7342	7440	7496	7605	7660					





67PARAMETER CODING MACY11 30A(1052) C7-APR-83 17:13 PAGE 279  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

CFCLOS= 000035	1416#													
C\$CLP1= 000006	1416#	5039	5142	5288	5338									
C\$CVEC= 000036	1416#	5042	5098	5145	5252	5291	5299	5341	5353	7044				
C\$DCLN= 000044	1416#	4842	5047	5150	5296	5346								
C\$DODU= 000051	1416#	5045	5148	5294	5344									
C\$DRPT= 000024	1416#													
C\$DU = 000053	1416#	4915												
C\$EDIT= 000003	1416#	1474												
C\$ERDF= 000055	1416#	5034	5137	5283	5333									
C\$ERHR= 000056	1416#	5068	5085	5171	5188	5205	5222	5239	5395	5424	5448	5479	5503	
	5528	5547	5595	5635	5651	5704	5726	5751	5768	5805	5822	5846	5859	
	6242	6262	6308	6328	6348	6439	6459	6482	6502	6543	6560	6577	6669	
	6689	6716	6736	6764	6774	6843	6851	6871	6892	6899	6919	6926	6990	
	7018	7034	7049	7120	7130	7166	7194	7206	7233	7301	7311	7333	7345	
	7366	7373	7447	7457	7480	7501	7510	7533	7540	7552	7612	7622	7650	
	7665	7674	7691	7704	7771	7781	7804	7818	7827	7846	7860	7921	7929	
	7951	7965	7974	7991	8059	8069	8090	8104	8111	8120	8133	8197	8207	
	8228	8242	8249	8258	8271	8345	8355	8378	8397	8406	8424	8434	8507	
	8517	8539	8558	8567	8585	8595	8660	8670	8701	8715	8724	8742	8752	
	8837	8847	8868	8886	8895	8916	8929	8938	8950	8959	8968	8978	9064	
	9074	9104	9118	9127	9150	9164	9233	9243	9273	9287	9296	9320	9334	
	9406	9416	9437	9452	9461	9474	9484	9571	9581	9614	9628	9637	9657	
	9671	9760	9770	9800	9814	9823	9838	9852	9923	9933	9963	9977	9986	
	10000	10014	10162	10172	10214	10228	10237	10254	10270	10284	10385	10395	10424	
	10447	10455	10466	10540	10550	10570	10586	10595	10609	10619	10700	10710	10736	
	10751	10760	10773	10789	10867	10877	10896	10918	10927	10941	10951	11039	11049	
	11075	11094	11103	11116	11126	11207	11217	11245	11264	11273	11289	11297	11307	
	11396	11406	11448	11462	11471	11485	11495	11604	11614	11647	11661	11670	11687	
	11702	11713	11727	11819	11829	11858	11872	11881	11895	11910	11921	11935	12055	
	12065	12099	12113	12122	12144	12155	12176	12190	12196	12216	12227	12248	12262	
	12271	12295	12305	12398	12408	12432	12446	12455	12469	12479	12499	12513	12522	
	12537	12550	12645	12655	12684	12698	12707	12720	12751	12812	12825	12846	12859	
	12890	12903	12935	12947										
C\$ERRO= 000060	1416#													
C\$ERSF= 000054	1416#													
C\$ERSO= 000057	1416#													
C\$ESCA= 000010	1416#	5400	5756	5810	5851	6247	6313	6444	6487	6548	6674	6694	6721	
	6741	6876	7023	7109	7125	7135	7171	7291	7306	7316	7338	7350	7437	
	7452	7462	7485	7506	7515	7602	7617	7627	7655	7670	7679	7709	7761	
	7776	7786	7809	7823	7832	7865	7956	7970	8049	8064	8074	8095	8116	
	8125	8187	8202	8212	8233	8254	8263	8335	8350	8360	8383	8402	8411	
	8497	8512	8522	8544	8563	8572	8650	8665	8675	8706	8720	8729	8827	
	8842	8852	8873	8891	8900	9054	9069	9079	9109	9123	9132	9169	9223	
	9238	9248	9278	9292	9301	9339	9396	9411	9421	9442	9457	9466	9561	
	9576	9586	9619	9633	9642	9676	9750	9765	9775	9805	9819	9828	9857	
	9913	9928	9938	9968	9982	9991	10019	10152	10167	10177	10219	10233	10242	
	10375	10390	10400	10429	10452	10530	10545	10555	10575	10591	10600	10690	10705	
	10715	10741	10756	10765	10794	10857	10872	10882	10901	10923	10932	11029	11044	
	11054	11080	11099	11108	11197	11212	11222	11250	11269	11278	11386	11401	11411	
	11453	11467	11476	11594	11609	11619	11652	11666	11675	11732	11809	11824	11834	
	11863	11877	11886	12045	12060	12070	12104	12118	12127	12181	12201	12253	12267	
	12276	12388	12403	12413	12437	12451	12460	12504	12518	12527	12542	12635	12650	
	12660	12689	12703	12712	12725	12756	12796	12817	12830	12851	12864	12895	12903	
	12940	12952												
C\$ESEG= 000005	1416#	5075	5092	5178	5195	5212	5229	5246	5350	5431	5455	5486	5510	
	5535	5554	5602	5642	5658	5775	6252	6269	6318	6335	6449	6466	6492	









67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 284  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

6359	6397	6400	6444	6450	6467	6487	6493	6510	6511	6513	6514	6548
6554	6568	6585	6587	6596	6598	6636	6674	6680	6694	6700	6721	6727
6741	6747	6782	6789	6791	6820	6860	6876	6934	6935	6937	6964	6998
7023	7042	7055	7057	7093	7109	7125	7135	7147	7171	7214	7241	7246
7248	7275	7291	7306	7316	7322	7338	7350	7381	7382	7384	7421	7437
7452	7462	7468	7485	7506	7515	7560	7561	7563	7586	7602	7617	7627
7633	7655	7670	7679	7698	7709	7714	7716	7745	7761	7776	7786	7792
7809	7823	7832	7854	7865	7873	7875	7898	7938	7956	7970	8004	8005
8007	8033	8049	8064	8074	8080	8095	8116	8125	8141	8142	8144	8171
8187	8202	8212	8218	8233	8254	8263	8279	8280	8282	8319	8335	8350
8360	8366	8383	8402	8411	8442	8443	8445	8481	8497	8512	8522	8528
8544	8563	8572	8603	8604	8606	8634	8650	8665	8675	8681	8706	8720
8729	8760	8763	8765	8811	8827	8842	8852	8858	8873	8891	8900	8986
8987	8989	9038	9054	9069	9079	9085	9109	9123	9132	9158	9169	9179
9181	9207	9223	9238	9248	9254	9278	9292	9301	9328	9339	9349	9351
9380	9396	9411	9421	9427	9442	9457	9466	9492	9493	9495	9545	9561
9576	9586	9592	9619	9633	9642	9665	9676	9688	9690	9734	9750	9765
9775	9781	9805	9819	9828	9846	9857	9871	9873	9897	9913	9928	9938
9944	9968	9982	9991	10008	10019	10032	10034	10136	10152	10167	10177	10183
10219	10233	10242	10292	10293	10295	10359	10375	10390	10400	10406	10429	10452
10474	10482	10484	10514	10530	10545	10555	10561	10575	10591	10600	10627	10628
10630	10674	10690	10705	10715	10721	10741	10756	10765	10781	10794	10800	10807
10809	10841	10857	10872	10882	10888	10901	10923	10932	10959	10960	10962	11013
11029	11044	11054	11060	11080	11099	11108	11135	11141	11143	11181	11197	11212
11222	11228	11250	11269	11278	11315	11322	11324	11370	11386	11401	11411	11417
11453	11467	11476	11516	11524	11526	11578	11594	11609	11619	11625	11652	11666
11675	11721	11732	11744	11746	11793	11809	11824	11834	11840	11863	11877	11886
11929	11941	11943	12029	12045	12060	12070	12076	12104	12118	12127	12163	12181
12201	12237	12253	12267	12276	12313	12314	12316	12372	12388	12403	12413	12419
12437	12451	12460	12487	12504	12518	12527	12542	12558	12565	12567	12619	12635
12650	12660	12666	12689	12703	12712	12725	12756	12764	12765	12767	12786	12790
12796	12817	12830	12851	12864	12895	12908	12940	12952	13014	13016	13154	13179
FSHARD= 000004	1416#	13139	13152									
FSHW = 000013	1416#	1582	1590									
FSINIT= 000006	1416#	4749	4845									
FSJMP = 000050	1416#	4718	4891	4911	4922	12790						
FSMOD = 000000	1416#	1420	13179									
FSMSG = 000011	1416#	2915	2924	2928	2940	2944	2961	2965	2974	2978	2987	2991
	3008	3023	3027	3040	3044	3053	3057	3074	3078	3100	3104	3113
	3125	3129	3138	3142	3150	3154	3175	3179	3188	3192	3208	3212
	3226	3237	3241	3265	3269	3293	3297	3307	3311	3336	3340	3363
	3390	3394	3425	3429	3437	3441	3475	3479	3523	3527	3542	3546
	3566	3582	3586	3610	3614	3629	3633	3670	3674	3697	3701	3710
	3730	3734	3757	3761	3769	3773	3781	3785	3793	3797	3805	3809
	3836	3852	3856	3894	3898	3906	3910	3945	3949	3969	3973	3981
FSPROT= 000021	1416#	4733	4740									
FSPWR = 000017	1416#											
FSRPT = 000012	1416#											
FSSEG = 000003	1416#	4716	4721									
	5231	5245	5326	5349	5414	5430	5442	5454	5468	5485	5497	5509
	5534	5537	5553	5587	5601	5629	5641	5645	5657	5760	5774	6229
	6254	6268	6288	6317	6320	6334	6406	6448	6451	6465	6468	6491
	6508	6520	6552	6555	6566	6641	6678	6681	6698	6701	6725	6728
	6751	6780	6827	6858	6867	6932	6970	6996	7002	7040	7100	7145
	7212	7218	7239	7282	7320	7329	7379	7428	7466	7476	7558	7593
	7646	7696	7752	7790	7800	7852	7905	7936	7946	8002	8040	8078
												8086

FSHARD= 000004  
 FSHW = 000013  
 FSINIT= 000006  
 FSJMP = 000050  
 FSMOD = 000000  
 FSMSG = 000011

FSPROT= 000021  
 FSPWR = 000017  
 FSRPT = 000012  
 FSSEG = 000003



67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 286  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

ICAB = 040000 G		1740#	5522											
IDU = 000040 G		1660#												
IE = 000100 G		1687#	4235	6979	6980	7176	7178							
IER = 020000 G		1668#												
INERR = 000003		1700#	3480	6904	7520	8417	8578	8735	9471	9835	9997	10248	10606	10770
		10938	11113	11684	11892									
IMITH = 177777 G		1688#	4020											
INITL = 177777 G		1689#	4021											
INPRN = 000001		1698#	4459	6830	7103	7285	7431	7596	7755	7908	8043	8181	8329	8491
		8644	8821	9048	9217	9390	9555	9744	9907	10146	10369	10524	10684	10851
		11023	11191	11380	11588	11803	12039	12382	12629					
INTBIT 002156RG	002	2126#	7122	7132	7168	7196	7208	7235						
INTE = 000100 G		1721#	5697	5701										
INTR = 000200 G		1720#	5418											
INTST = 000002		1699#	6770											
INTVEC 002121RG	002	2121#	6992	7020	7036	7051								
IOADR = 020000 G		1684#	1685											
IOSIZ = 020000 G		1680#	1685											
ISR = 000100 G		1661#												
IXE = 004000 G		1666#												
ISAU = 000041		1416#	4920#	4927#										
ISAUTO= 000041		1416#	4860#	4864#										
ISCLN = 000041		1416#	4872#	4891	4906#									
ISDU = 000041		1416#	4909#	4916#										
ISHRD = 000041		13139#	13154#											
ISINIT= 000041		1416#	4749#	4847#										
ISMOD = 000041		1416#	1420#	13179#										
ISMSG = 000041		1416#	2915#	2926#	2928#	2942#	2944#	2963#	2965#	2976#	2978#	2989#	2991#	3006#
		3008#	3025#	3027#	3042#	3044#	3055#	3057#	3076#	3078#	3102#	3104#	3115#	3117#
		3127#	3129#	3140#	3142#	3152#	3154#	3177#	3179#	3190#	3192#	3210#	3212#	3224#
		3226#	3239#	3241#	3267#	3269#	3295#	3297#	3309#	3311#	3338#	3340#	3365#	3367#
		3392#	3394#	3427#	3429#	3439#	3441#	3477#	3479#	3525#	3527#	3544#	3546#	3564#
		3566#	3584#	3586#	3612#	3614#	3631#	3633#	3672#	3674#	3699#	3701#	3712#	3714#
		3732#	3734#	3759#	3761#	3771#	3773#	3783#	3785#	3795#	3797#	3807#	3809#	3834#
		3836#	3854#	3856#	3896#	3898#	3908#	3910#	3947#	3949#	397#	3973#	3983#	
ISPROT= 000040		1416#	4733#											
ISPTAB= 000041		1416#												
ISPUR = 000041		1416#												
ISRPT = 000041		1416#	4716#	4723#										
ISSEG = 000041		1416#	5014	5023	5053	5059#	5076#	5077#	5093#	5117	5126	5156	5162#	5179#
		5180#	5196#	5197#	5213#	5214#	5230#	5231#	5247#	5269	5317	5326#	5351#	5376
		5381	5408	5414#	5432#	5442#	5456#	5468#	5487#	5497#	5511#	5518#	5536#	5537#
		5555#	5581	5587#	5603#	5625	5629#	5643#	5645#	5659#	5688	5690	5713	5735
		5760#	5776#	5801	6221	6223	6229#	6253#	6254#	6270#	6274	6288#	6319#	6320#
		6336#	6397	6400	6406#	6450#	6451#	6467#	6468#	6493#	6494#	6510#	6514	6520#
		6554#	6555#	6568#	6636	6641#	6680#	6681#	6700#	6701#	6727#	6728#	6747#	6751#
		6782#	6820	6827#	6860#	6867#	6934#	6964	6970#	6998#	7002#	7042#	7093	7100#
		7147#	7162#	7214#	7218#	7219#	7275	7282#	7322#	7329#	7381#	7421	7428#	7468#
		7476#	7560#	7586	7593#	7633#	7646#	7698#	7745	7752#	7792#	7800#	7854#	7898
		7905#	7938#	7946#	8004#	8033	8040#	8080#	8086#	8141#	8171	8178#	8218#	8224#
		8279#	8319	8326#	8366#	8374#	8442#	8481	8488#	8528#	8535#	8603#	8634	8641#
		8681#	8693#	8760#	8811	8818#	8858#	8864#	8986#	9038	9045#	9085#	9096#	9158#
		9207	9214#	9254#	9265#	9328#	9380	9387#	9427#	9433#	9492#	9545	9552#	9592#
		9607#	9665#	9734	9741#	9781#	9792#	9846#	9897	9904#	9944#	9959#	10008#	10136
		10143#	10183#	10205#	10292#	10359	10366#	10406#	10416#	10474#	10514	10521#	10561#	10566#
		10627#	10674	10681#	10721#	10732#	10781#	10785#	10800#	10841	10848#	10888#	10892#	10959#





67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 288  
 CZUAA9.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

LPM5G	052554RG	002	12964*	12998	13024#
LPM5G0	052740RG	002	13069	13082#	
LPM5G1	052777RG	002	13070	13088#	
LPTBL	052672RG	002	12963	13069#	
LSACP	000110RG	002	1504#		
LSAPT	000036RG	002	1462#		
LSAU	022026RG	002	1489	4920#	
LSAUT	000070RG	002	1488#		
LSAUTO	021652RG	002	1505	4860#	
LSCCP	000106RG	002	1502#		
LSCLEA	021654RG	002	1503	4872#	
LSCO	000032RG	002	1458#		
LSDEPO	000011RG	002	1440#		
LSDESC	000706RG	002	1495	1970#	
LSDESP	000076RG	002	1494#		
LSDEVP	000060RG	002	1480#		
LSDISP	000124RG	002	1465	1524#	
LSDLY	000116RG	002	1510#		
LSDTP	000040RG	002	1464#		
LSDTYP	000034RG	002	1460#		
LSDU	022020RG	002	1491	4909#	
LSDUT	022072RG	002	1490#		
LSDVTY	000700RG	002	1481	1962#	
LSEF	000052RG	002	1475#		
LSENV1	000044RG	002	1468#		
LSETP	000102RG	002	1498#		
LSEXP1	000046RG	002	1470#		
LSEXP4	000064RG	002	1484#		
LSEXP5	000066RG	002	1486#		
LSHARD	053256RG	002	1447	13139	13140#
LSHIME	000120RG	002	1512#		
LSHPCP	000016RG	002	1446#		
LSHPTP	000022RG	002	1450#		
LSHW	000262RG	002	1451	1582	1583#
LSICP	000104RG	002	1500#		
LSINIT	021212RG	002	1501	4749#	
LSLADP	000026RG	002	1454#		
LSLAST=	***** GX		1455		
LSLOAD	000100RG	002	1496#		
LSLUN	000074RG	002	1492#		
LSMREV	000050RG	002	1472#		
LSNAME	000000RG	002	1429#		
LSPRIO	000042RG	002	1466#		
LSPROT	021204RG	002	1507	4733#	
LSPRT	000112RG	002	1506#		
LSREPP	000062RG	002	1482#		
LSREV	000010RG	002	1438#		
LSRPT	021176RG	002	1483	4716#	
LSSPC	000036RG	002	1478#		
LSSPCP	000020RG	002	1448#		
LSSTP	000024RG	002	1452#		
LSSTA	000030RG	002	1456#		
LSTEST	000114RG	002	1508#		
LSTIML	000014RG	002	1444#		
LSUNIT	000012RG	002	1442#	4819	
L1000	000266R	002	1582	1590#	

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 289  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

L10001	012524R	002	2924#	
L10002	012566R	002	2940#	
L10003	012640R	002	2961#	
L10004	012666R	002	2974#	
L10005	012714R	002	2987#	
L10006	012762R	002	3004#	
L10007	013026R	002	3023#	
L10010	013074R	002	3040#	
L10011	013122R	002	3053#	
L10012	013172R	002	3074#	
L10013	013260R	002	3100#	
L10014	013306R	002	3113#	
L10015	013330R	002	3125#	
L10016	013356R	002	3138#	
L10017	013400R	002	3150#	
L10020	013464R	002	3175#	
L10021	013510R	002	3188#	
L10022	013556R	002	3208#	
L10023	013604R	002	3222#	
L10024	013634R	002	3237#	
L10025	013726R	002	3265#	
L10026	014020R	002	3293#	
L10027	014050R	002	3307#	
L10030	014144R	002	3336#	
L10031	014234R	002	3363#	
L10032	014324R	002	3390#	
L10033	014440R	002	3425#	
L10034	014462R	002	3437#	
L10035	014612R	002	3475#	
L10036	015000R	002	3523#	
L10037	015044R	002	3542#	
L10040	015112R	002	3562#	
L10041	015160R	002	3582#	
L10042	015262R	002	3610#	
L10043	015330R	002	3629#	
L10044	015474R	002	3670#	
L10045	015572R	002	3697#	
L10046	015620R	002	3710#	
L10047	015670R	002	3730#	
L10050	015766R	002	3757#	
L10051	016010R	002	3769#	
L10052	016032R	002	3781#	
L10053	016054R	002	3793#	
L10054	016076R	002	3805#	
L10055	016174R	002	3832#	
L10056	016246R	002	3852#	
L10057	016416R	002	3894#	
L10060	016440R	002	3906#	
L10061	016602R	002	3945#	
L10062	016664R	002	3969#	
L10063	016706R	002	3981#	
L10064	021202R	002	4719	4721#
L10066	021606R	002	4845#	
L10067	021652R	002	4862#	
L10070	022016R	002	4892	4904#
L10071	022024R	002	4912	4914#



















67PARAMETER CODING MACY11 30A(1652) 07-APR-83 17:13 PAGE 298  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

3656	3657	3658	3659	3660	3662	3663	3664	3665	3666	3667	3668	3671
3676	3677	3678	3679	3680	3682	3683	3684	3685	3686	3687	3688	3690
3691	3692	3693	3694	3695	3698	3703	3704	3705	3706	3707	3708	3711
3716	3717	3718	3719	3720	3721	3723	3724	3725	3726	3727	3728	3731
3736	3737	3738	3739	3740	3741	3743	3744	3745	3746	3747	3749	3750
3751	3752	3753	3754	3755	3758	3763	3764	3765	3766	3767	3770	3775
3776	3777	3778	3779	3782	3787	3788	3789	3790	3791	3794	3799	3800
3801	3802	3803	3806	3811	3812	3813	3814	3815	3817	3818	3819	3820
3821	3822	3824	3825	3826	3827	3828	3829	3830	3833	3838	3839	3840
3841	3842	3844	3845	3846	3847	3848	3849	3850	3853	3858	3859	3860
3861	3862	3863	3865	3866	3867	3868	3869	3870	3872	3873	3874	3875
3876	3877	3879	3880	3881	3882	3883	3884	3886	3887	3888	3889	3890
3891	3892	3895	3900	3901	3902	3903	3904	3907	3914	3915	3916	3917
3918	3922	3923	3924	3925	3926	3930	3931	3932	3933	3934	3938	3939
3940	3941	3942	3946	3951	3952	3953	3954	3955	3957	3958	3959	3960
3961	3963	3964	3965	3966	3967	3970	3975	3976	3977	3978	3979	3982
4233	4234	4247	4248	4277	4386	4387	4388	4389	4390	4391	4422	4423
4424	4425	4426	4427	4428	4462	4539	4540	4541	4542	4543	4544	4545
4546	4547	4554	4555	4556	4557	4558	4575	4576	4577	4578	4579	4580
4641	4642	4643	4644	4645	4646	4656	4657	4658	4659	4660	4661	4669
4670	4671	4672	4673	4674	4684	4685	4686	4687	4688	4689	4697	4698
4699	4700	4701	4702	4718	4719	4722	4754	4755	4757	4759	4760	4763
4774	4775	4777	4779	4780	4782	4785	4786	4790	4791	4792	4794	4796
4797	4798	4799	4800	4809	4810	4811	4812	4813	4814	4822	4823	4824
4826	4842	4846	4863	4882	4883	4884	4885	4886	4891	4892	4905	4911
4912	4915	4922	4923	4926	4947	4967	4995	5016	5017	5018	5019	5020
5021	5024	5034	5035	5036	5037	5039	5041	5042	5044	5045	5047	5051
5054	5059	5068	5069	5070	5071	5075	5077	5085	5086	5087	5088	5092
5095	5097	5098	5101	5119	5120	5121	5122	5123	5124	5127	5137	5138
5139	5140	5142	5144	5145	5147	5148	5150	5154	5157	5162	5171	5172
5173	5174	5178	5180	5188	5189	5190	5191	5195	5197	5205	5206	5207
5208	5212	5214	5222	5223	5224	5225	5229	5231	5239	5240	5241	5242
5246	5249	5251	5252	5255	5271	5272	5273	5274	5275	5276	5283	5284
5285	5286	5288	5290	5291	5293	5294	5296	5298	5299	5303	5319	5320
5321	5322	5323	5324	5326	5333	5334	5335	5336	5338	5340	5341	5343
5344	5346	5350	5352	5353	5356	5382	5395	5396	5397	5398	5400	5401
5405	5409	5414	5424	5425	5426	5427	5431	5442	5448	5449	5450	5451
5455	5468	5479	5489	5481	5482	5486	5497	5503	5504	5505	5506	5510
5518	5528	5529	5530	5531	5535	5537	5547	5548	5549	5550	5554	5557
5560	5587	5595	5596	5597	5598	5602	5607	5629	5635	5636	5637	5638
5642	5645	5651	5652	5653	5654	5658	5661	5691	5704	5705	5706	5707
5711	5714	5726	5727	5728	5729	5733	5736	5751	5752	5753	5754	5756
5757	5760	5768	5769	5770	5771	5775	5778	5781	5805	5806	5807	5808
5810	5811	5822	5823	5824	5825	5846	5847	5848	5849	5851	5852	5859
5860	5861	5862	5866	6224	6229	6242	6243	6244	6245	6247	6248	6252
6254	6262	6263	6264	6265	6269	6272	6275	6288	6308	6309	6310	6311
6313	6314	6318	6320	6328	6329	6330	6331	6335	6348	6349	6350	6351
6355	6358	6401	6406	6439	6440	6441	6442	6444	6445	6449	6451	6459
6460	6461	6462	6466	6468	6482	6483	6484	6485	6487	6488	6492	6494
6502	6503	6504	6505	6509	6512	6515	6520	6543	6544	6545	6546	6548
6549	6553	6555	6560	6561	6562	6563	6567	6577	6578	6579	6580	6586
6597	6641	6669	6670	6671	6672	6674	6675	6679	6681	6689	6690	6691
6692	6694	6695	6699	6701	6716	6717	6718	6719	6721	6722	6726	6728
6736	6737	6738	6739	6741	6742	6746	6751	6764	6765	6766	6767	6774
6775	6776	6777	6781	6790	6827	6843	6844	6845	6846	6851	6852	6853
6854	6859	6867	6871	6872	6873	6874	6876	6877	6892	6893	6894	6895

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 299  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

6899	6900	6901	6902	6919	6920	6921	6922	6926	6927	6928	6929	6933
6936	6970	6972	6973	6974	6975	6976	6977	6990	6991	6992	6993	6997
7002	7008	7009	7018	7019	7020	7021	7023	7024	7034	7035	7036	7037
7041	7043	7044	7049	7050	7051	7052	7056	7100	7109	7110	7120	7121
7122	7123	7125	7126	7150	7131	7132	7133	7135	7136	7138	7139	7140
7141	7142	7143	7146	7162	7166	7167	7168	7169	7171	7172	7181	7182
7194	7195	7196	7197	7206	7207	7208	7209	7213	7218	7233	7234	7235
7236	7240	7247	7282	7291	7292	7301	7302	7303	7304	7306	7307	7311
7312	7313	7314	7316	7317	7321	7329	7333	7334	7335	7336	7338	7339
7345	7346	7347	7348	7350	7351	7366	7367	7368	7369	7373	7374	7375
7376	7380	7383	7428	7437	7438	7447	7448	7449	7450	7452	7453	7457
7458	7459	7460	7462	7463	7467	7476	7480	7481	7482	7483	7485	7486
7501	7502	7503	7504	7506	7507	7510	7511	7512	7513	7515	7516	7533
7534	7535	7536	7540	7541	7542	7543	7552	7553	7554	7555	7559	7562
7593	7602	7603	7612	7613	7614	7615	7617	7618	7622	7623	7624	7625
7627	7628	7632	7646	7650	7651	7652	7653	7655	7656	7665	7666	7667
7668	7670	7671	7674	7675	7676	7677	7679	7680	7691	7692	7693	7694
7697	7704	7705	7706	7707	7709	7710	7715	7752	7761	7762	7771	7772
7773	7774	7776	7777	7781	7782	7783	7784	7786	7787	7791	7800	7804
7805	7806	7807	7809	7810	7818	7819	7820	7821	7823	7824	7827	7828
7829	7830	7832	7833	7846	7847	7848	7849	7853	7860	7861	7862	7863
7865	7866	7874	7905	7921	7922	7923	7924	7929	7930	7931	7932	7937
7946	7951	7952	7953	7954	7956	7957	7965	7966	7967	7968	7970	7971
7974	7975	7976	7977	7991	7992	7993	7994	8003	8006	8040	8049	8050
8059	8060	8061	8062	8064	8065	8069	8070	8071	8072	8074	8075	8079
8086	8090	8091	8092	8093	8095	8096	8104	8105	8106	8107	8111	8112
8113	8114	8116	8117	8120	8121	8122	8123	8125	8126	8133	8134	8135
8136	8140	8143	8178	8187	8188	8197	8198	8199	8200	8202	8203	8207
8208	8209	8210	8212	8213	8217	8224	8228	8229	8230	8231	8233	8234
8242	8243	8244	8245	8249	8250	8251	8252	8254	8255	8258	8259	8260
8261	8263	8264	8271	8272	8273	8274	8278	8281	8326	8335	8336	8345
8346	8347	8348	8350	8351	8355	8356	8357	8358	8360	8361	8365	8374
8378	8379	8380	8381	8383	8384	8397	8398	8399	8400	8402	8403	8406
8407	8408	8409	8411	8412	8424	8425	8426	8427	8434	8435	8436	8437
8441	8444	8488	8497	8498	8507	8508	8509	8510	8512	8513	8517	8518
8519	8520	8522	8523	8527	8535	8539	8540	8541	8542	8544	8545	8558
8559	8560	8561	8563	8564	8567	8568	8569	8570	8572	8573	8585	8586
8587	8588	8595	8596	8597	8598	8602	8605	8641	8650	8651	8660	8661
8662	8663	8665	8666	8670	8671	8672	8673	8675	8676	8680	8693	8701
8702	8703	8704	8706	8707	8715	8716	8717	8718	8720	8721	8724	8725
8726	8727	8729	8730	8742	8743	8744	8745	8752	8753	8754	8755	8759
8764	8818	8827	8828	8837	8838	8839	8840	8842	8843	8847	8848	8849
8850	8852	8853	8857	8864	8868	8869	8870	8871	8873	8874	8886	8887
8888	8889	8891	8892	8895	8896	8897	8898	8900	8901	8916	8917	8918
8919	8929	8930	8931	8932	8938	8939	8940	8941	8950	8951	8952	8953
8959	8960	8961	8962	8968	8969	8970	8971	8978	8979	8980	8981	8985
8988	9045	9054	9055	9064	9065	9066	9067	9069	9070	9074	9075	9076
9077	9079	9080	9084	9096	9104	9105	9106	9107	9109	9110	9118	9119
9120	9121	9123	9124	9127	9128	9129	9130	9132	9133	9150	9151	9152
9153	9157	9164	9165	9166	9167	9169	9170	9180	9214	9223	9224	9233
9234	9235	9236	9238	9239	9243	9244	9245	9246	9248	9249	9253	9265
9273	9274	9275	9276	9278	9279	9287	9288	9289	9290	9292	9293	9296
9297	9298	9299	9301	9302	9320	9321	9322	9323	9327	9334	9335	9336
9337	9339	9340	9350	9387	9396	9397	9406	9407	9408	9409	9411	9412
9416	9417	9418	9419	9421	9422	9426	9433	9437	9438	9439	9440	9442
9443	9452	9453	9454	9455	9457	9458	9461	9462	9463	9464	9466	9467

9474	9475	9476	9477	9484	9485	9486	9487	9491	9494	9552	9561	9562
9571	9572	9573	9574	9576	9577	9581	9582	9583	9584	9586	9587	9591
9607	9614	9615	9616	9617	9619	9620	9628	9629	9630	9631	9633	9634
9637	9638	9639	9640	9642	9643	9657	9658	9659	9660	9664	9671	9672
9673	9674	9676	9677	9689	9741	9750	9751	9760	9761	9762	9763	9765
9766	9770	9771	9772	9773	9775	9776	9780	9792	9800	9801	9802	9803
9805	9806	9814	9815	9816	9817	9819	9820	9823	9824	9825	9826	9828
9829	9838	9839	9840	9841	9845	9852	9853	9854	9855	9857	9858	9872
9904	9913	9914	9923	9924	9925	9926	9928	9929	9933	9934	9935	9936
9938	9939	9943	9959	9963	9964	9965	9966	9968	9969	9977	9978	9979
9980	9982	9983	9986	9987	9988	9989	9991	9992	10000	10001	10002	10003
10007	10014	10015	10016	10017	10019	10020	10033	10143	10152	10153	10162	10163
10164	10165	10167	10168	10172	10173	10174	10175	10177	10178	10182	10205	10214
10215	10216	10217	10219	10220	10228	10229	10230	10231	10233	10234	10237	10238
10239	10240	10242	10243	10254	10255	10256	10257	10270	10271	10272	10273	10284
10285	10286	10287	10291	10294	10366	10375	10376	10385	10386	10387	10388	10390
10391	10395	10396	10397	10398	10400	10401	10405	10416	10424	10425	10426	10427
10429	10430	10447	10448	10449	10450	10452	10453	10455	10456	10457	10458	10466
10467	10468	10469	10473	10483	10521	10530	10531	10540	10541	10542	10543	10545
10546	10550	10551	10552	10553	10555	10556	10560	10566	10570	10571	10572	10573
10575	10576	10586	10587	10588	10589	10591	10592	10595	10596	10597	10598	10600
10601	10609	10610	10611	10612	10619	10620	10621	10622	10626	10629	10681	10690
10691	10700	10701	10702	10703	10705	10706	10710	10711	10712	10713	10715	10716
10720	10732	10736	10737	10738	10739	10741	10742	10751	10752	10753	10754	10756
10757	10760	10761	10762	10763	10765	10766	10773	10774	10775	10776	10780	10785
10789	10790	10791	10792	10794	10795	10799	10808	10848	10857	10858	10867	10868
10869	10870	10872	10873	10877	10878	10879	10880	10882	10883	10887	10892	10896
10897	10898	10899	10901	10902	10918	10919	10920	10921	10923	10924	10927	10928
10929	10930	10932	10933	10941	10942	10943	10944	10951	10952	10953	10954	10958
10961	11020	11029	11030	11039	11040	11041	11042	11044	11045	11049	11050	11051
11052	11054	11055	11059	11068	11075	11076	11077	11078	11080	11081	11094	11095
11096	11097	11099	11100	11103	11104	11105	11106	11108	11109	11116	11117	11118
11119	11126	11127	11128	11129	11134	11142	11188	11197	11198	11207	11208	11209
11210	11212	11213	11217	11218	11219	11220	11222	11223	11227	11241	11245	11246
11247	11248	11250	11251	11264	11265	11266	11267	11269	11270	11273	11274	11275
11276	11278	11279	11289	11290	11291	11292	11297	11298	11299	11300	11307	11308
11309	11310	11314	11323	11377	11386	11387	11396	11397	11398	11399	11401	11402
11406	11407	11408	11409	11411	11412	11416	11428	11438	11448	11449	11450	11451
11453	11454	11462	11463	11464	11465	11467	11468	11471	11472	11473	11474	11476
11477	11485	11486	11487	11488	11495	11496	11497	11498	11502	11515	11525	11585
11594	11595	11604	11605	11606	11607	11609	11610	11614	11615	11616	11617	11619
11620	11624	11634	11647	11648	11649	11650	11652	11653	11661	11662	11663	11664
11666	11667	11670	11671	11672	11673	11675	11676	11687	11688	11689	11690	11702
11703	11704	11705	11713	11714	11715	11716	11720	11727	11728	11729	11730	11732
11733	11745	11800	11809	11810	11819	11820	11821	11822	11824	11825	11829	11830
11831	11832	11834	11835	11839	11845	11858	11859	11860	11861	11863	11864	11872
11873	11874	11875	11877	11878	11881	11882	11883	11884	11886	11887	11895	11896
11897	11898	11910	11911	11912	11913	11921	11922	11923	11924	11928	11935	11936
11937	11938	11942	12036	12045	12046	12055	12056	12057	12058	12060	12061	12065
12066	12067	12068	12070	12071	12075	12091	12099	12100	12101	12102	12104	12105
12113	12114	12115	12116	12118	12119	12122	12123	12124	12125	12127	12128	12144
12145	12146	12147	12155	12156	12157	12158	12162	12167	12176	12177	12178	12179
12181	12182	12190	12191	12192	12193	12196	12197	12198	12199	12201	12202	12216
12217	12218	12219	12227	12228	12229	12230	12236	12241	12248	12249	12250	12251
12253	12254	12262	12263	12264	12265	12267	12268	12271	12272	12273	12274	12276
12277	12295	12296	12297	12298	12305	12306	12307	12308	12312	12315	12379	12388

57PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 301  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

12389	12398	12399	12400	12401	12403	12404	12408	12409	12410	12411	12413	12414
12418	12425	12432	12433	12434	12435	12437	12438	12446	12447	12448	12449	12451
12452	12455	12456	12457	12458	12460	12461	12469	12470	12471	12472	12479	12480
12481	12482	12486	12492	12499	12500	12501	12502	12504	12505	12513	12514	12515
12516	12518	12519	12522	12523	12524	12525	12527	12528	12537	12538	12539	12540
12542	12543	12550	12551	12552	12553	12557	12566	12626	12635	12636	12645	12646
12647	12648	12650	12651	12655	12656	12657	12658	12660	12661	12665	12680	12684
12685	12686	12687	12689	12690	12698	12699	12700	12701	12703	12704	12707	12708
12709	12710	12712	12713	12720	12721	12722	12723	12725	12726	12751	12752	12753
12754	12756	12757	12763	12766	12790	12791	12796	12797	12812	12813	12814	12815
12817	12818	12825	12826	12827	12828	12830	12831	12846	12847	12848	12849	12851
12852	12859	12860	12861	12862	12864	12865	12890	12891	12892	12893	12895	12896
12903	12904	12905	12906	12908	12909	12935	12936	12937	12938	12940	12941	12947
12948	12949	12950	12952	12953	12977	12978	12979	12980	12981	12982	12984	12985
12986	12987	12988	12989	12991	12992	12993	12994	12995	12996	12998	12999	13000
13001	13002	13003	13005	13006	13007	13008	13009	13010	13015	13139	13142	13143
13144	13145	13147	13148	13149	13150	13152						

SVCSUB= 000001

1416#	5023	5053	5126	5156	5381	5408	5690	5713	5735	6223	6274	6400
6514												

SVCTAG= 000001

1416#	1590	2924	2940	2961	2974	2987	3004	3023	3040	3053	3074	3100
3113	3125	3138	3150	3175	3188	3208	3222	3237	3265	3293	3307	3336
3363	3390	3425	3437	3475	3523	3542	3562	3582	3610	3629	3670	3697
3710	3730	3757	3769	3781	3793	3805	3832	3852	3894	3906	3945	3969
3981	4721	4845	4862	4904	4914	4925	4946	4966	4994	5050	5074	5091
5094	5100	5153	5177	5194	5211	5228	5245	5248	5254	5302	5349	5355
5404	5430	5454	5485	5509	5534	5553	5556	5559	5601	5606	5641	5657
5660	5710	5732	5774	5777	5780	5865	6251	6268	6271	6317	6334	6354
6357	6448	6465	6491	6508	6511	6552	6566	6585	6596	6678	6698	6725
6745	6780	6789	6858	6932	6935	6996	7040	7055	7145	7212	7239	7246
7320	7379	7382	7466	7558	7561	7631	7696	7714	7790	7852	7873	7936
8002	8005	8078	8139	8142	8216	8277	8280	8364	8440	8443	8526	8601
8604	8679	8758	8763	8856	8984	8987	9083	9156	9179	9252	9326	9349
9425	9490	9493	9590	9663	9688	9779	9844	9871	9942	10006	10032	10181
10290	10293	10404	10472	10482	10559	10625	10628	10719	10779	10798	10807	10886
10957	10960	11058	11133	11141	11226	11313	11322	11415	11501	11514	11524	11623
11719	11744	11838	11927	11941	12074	12161	12235	12311	12314	12417	12485	12556
12565	12664	12762	12765	13014	13153							

SVCTST= 000001

1416#	5014	5117	5269	5317	5376	5581	5625	5688	5801	6221	6397	6636
6820	6964	7093	7275	7421	7586	7745	7898	8033	8171	8319	8481	8634
8811	9038	9207	9380	9545	9734	9897	10136	10359	10514	10674	10841	11013
11181	11370	11578	11793	12029	12372	12619	12786					

SWADDR 000334RG 002  
SUNDR 052706RG 002  
SWPACK 052552RG 002  
SBL SYM= 010000

1831#	12924											
12991	13077#											
12922	12957	12965	13023#									
1416#	1591#	2925#	2941#	2962#	2975#	2988#	3005#	3024#	3041#	3054#	3075#	3101#
3114#	3126#	3139#	3151#	3176#	3189#	3209#	3223#	3238#	3266#	3294#	3308#	3337#
3364#	3391#	3426#	3438#	3476#	3524#	3543#	3563#	3583#	3611#	3630#	3671#	3698#
3711#	3731#	3758#	3770#	3782#	3794#	3806#	3833#	3853#	3895#	3907#	3946#	3970#
3982#	4722#	4846#	4863#	4905#	4915#	4926#	4947#	4967#	4995#	5051#	5059#	5077#
5095#	5101#	5154#	5162#	5180#	5197#	5214#	5231#	5249#	5255#	5303#	5326#	5356#
5405#	5414#	5442#	5468#	5497#	5518#	5537#	5557#	5560#	5587#	5607#	5629#	5645#
5661#	5711#	5733#	5760#	5778#	5781#	5866#	6229#	6254#	6272#	6288#	6320#	6355#
6358#	6406#	6451#	6468#	6494#	6512#	6520#	6555#	6586#	6597#	6641#	6681#	6701#
6728#	6751#	6790#	6827#	6867#	6936#	6970#	7002#	7056#	7100#	7162#	7218#	7247#
7282#	7329#	7383#	7428#	7476#	7562#	7593#	7646#	7715#	7752#	7800#	7874#	7905#
7946#	8006#	8040#	8086#	8143#	8178#	8224#	8281#	8326#	8374#	8444#	8488#	8535#





67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 303  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

TSEXCP= 000000  
TSFLAG= 000040

11911#	11922#	11936#	12056#	12066#	2100#	12114#	12123#	12145#	12156#	12177#	12191#	12197#
12217#	12228#	12249#	12263#	12272#	12296#	12306#	12399#	12409#	12433#	12447#	12456#	12470#
12480#	12500#	12514#	12523#	12538#	12551#	12646#	12656#	12685#	12699#	12708#	12721#	12752#
12813#	12826#	12847#	12860#	12891#	12904#	12936#	12948#					
13142#	13146	13147#	13151									
4718#	4720	4891#	4911#	4913	4922#	4924	5400#	5756#	5810#	5851#	6247#	6313#
6444#	6487#	6548#	6674#	6694#	6721#	6741#	6876#	7023#	7109#	7125#	7135#	7171#
7291#	7306#	7316#	7338#	7350#	7437#	7452#	7462#	7485#	7506#	7515#	7602#	7617#
7627#	7655#	7670#	7679#	7709#	7761#	7776#	7786#	7809#	7823#	7832#	7865#	7956#
7970#	8049#	8064#	8074#	8095#	8116#	8125#	8187#	8202#	8212#	8233#	8254#	8263#
8335#	8350#	8360#	8383#	8402#	8411#	8497#	8512#	8522#	8544#	8563#	8572#	8650#
8665#	8675#	8706#	8720#	8729#	8827#	8842#	8852#	8873#	8891#	8900#	9054#	9069#
9079#	9109#	9123#	9132#	9169#	9223#	9238#	9248#	9278#	9292#	9301#	9339#	9396#
9411#	9421#	9442#	9457#	9466#	9561#	9576#	9586#	9619#	9633#	9642#	9676#	9750#
9765#	9775#	9805#	9819#	9828#	9857#	9913#	9928#	9938#	9968#	9982#	9991#	10019#
10152#	10167#	10177#	10219#	10233#	10242#	10375#	10390#	10400#	10429#	10452#	10530#	10545#
10555#	10575#	10591#	10600#	10690#	10705#	10715#	10741#	10756#	10765#	10794#	10857#	10872#
10882#	10901#	10923#	10932#	11029#	11044#	11054#	11080#	11099#	11108#	11197#	11212#	11222#
11250#	11269#	11278#	11386#	11401#	11411#	11453#	11467#	11476#	11594#	11609#	11619#	11652#
11666#	11675#	11732#	11809#	11824#	11834#	11863#	11877#	11886#	12045#	12060#	12070#	12104#
12118#	12127#	12181#	12201#	12253#	12267#	12276#	12388#	12403#	12413#	12437#	12451#	12460#
12504#	12518#	12527#	12542#	12635#	12650#	12660#	12689#	12703#	12712#	12725#	12756#	12790#
12796#	12817#	12830#	12851#	12864#	12895#	12908#	12940#	12952#				

TSGMAN= 000000  
TSHILI= 000776  
TSLAST= 000000  
TSLOLI= 000000  
TSLSYM= 010000

1416#												
13142#	13145	13147#	13150									
1416#												
13142#	13144	13147#	13149									
1416#	1591	2925	2941	2962	2975	2988	3005	3024	3041	3054	3075	3101
3114	3126	3139	3151	3176	3189	3209	3223	3238	3266	3294	3308	3337
3364	3391	3426	3438	3476	3524	3543	3563	3583	3611	3630	3671	3698
3711	3731	3758	3770	3782	3794	3806	3833	3853	3895	3907	3946	3970
3982	4722	4846	4863	4905	4915	4926	4947	4967	4995	5051	5095	5101
5154	5249	5255	5303	5356	5405	5557	5560	5607	5661	5711	5733	5778
5781	5866	6272	6355	6358	6512	6586	6597	6790	6936	7056	7247	7383
7562	7715	7874	8006	8143	8281	8444	8605	8764	8988	9180	9350	9494
9689	9872	10033	10294	10483	10629	10808	10961	11142	11323	11525	11745	11942

T\$NEST= 177777

12315	12566	12766	13015	13154								
1416#	1420#	1582#	1590#	2915#	2924#	2928#	2940#	2944#	2961#	2965#	2974#	2978#
2987#	2991#	3004#	3008#	3023#	3027#	3040#	3044#	3053#	3057#	3074#	3078#	3100#
3104#	3113#	3117#	3125#	3129#	3138#	3142#	3150#	3154#	3175#	3179#	3188#	3192#
3208#	3212#	3222#	3226#	3237#	3241#	3265#	3269#	3293#	3297#	3307#	3311#	3336#
3340#	3363#	3367#	3390#	3394#	3425#	3429#	3437#	3441#	3475#	3479#	3523#	3527#
3542#	3546#	3562#	3566#	3582#	3586#	3610#	3614#	3629#	3633#	3670#	3674#	3697#
3701#	3710#	3714#	3730#	3734#	3757#	3761#	3769#	3773#	3781#	3785#	3793#	3797#
3805#	3809#	3832#	3836#	3852#	3856#	3894#	3898#	3906#	3910#	3945#	3949#	3969#
3973#	3981#	4716#	4721#	4733#	4740#	4749#	4845#	4860#	4862#	4872#	4904#	4909#
4914#	4920#	4925#	4943#	4946#	4962#	4966#	4988#	4994#	5015#	5024#	5050#	5054#
5059#	5074#	5077#	5091#	5094#	5100#	5118#	5127#	5153#	5157#	5162#	5177#	5180#
5194#	5197#	5211#	5214#	5228#	5231#	5245#	5248#	5254#	5270#	5302#	5318#	5326#
5349#	5355#	5377#	5382#	5404#	5409#	5414#	5430#	5442#	5454#	5468#	5485#	5497#
5509#	5518#	5534#	5537#	5553#	5556#	5559#	5582#	5587#	5601#	5606#	5626#	5629#
5641#	5645#	5657#	5660#	5689#	5691#	5710#	5714#	5732#	5736#	5760#	5774#	5777#
5780#	5802#	5865#	6222#	6224#	6229#	6251#	6254#	6268#	6271#	6275#	6288#	6317#
6320#	6334#	6354#	6357#	6398#	6401#	6406#	6448#	6451#	6465#	6468#	6491#	6494#
6508#	6511#	6515#	6520#	6552#	6555#	6566#	6585#	6596#	6637#	6641#	6678#	6681#
6698#	6701#	6725#	6728#	6745#	6751#	6780#	6789#	6821#	6827#	6858#	6867#	6932#

6935#	6965#	6970#	6996#	7002#	7040#	7055#	7094#	7100#	7145#	7162#	7212#	7218#
7239#	7246#	7276#	7282#	7320#	7329#	7379#	7382#	7422#	7428#	7466#	7476#	7558#
7561#	7587#	7593#	7631#	7646#	7696#	7714#	7746#	7752#	7790#	7800#	7852#	7873#
7899#	7905#	7936#	7946#	8002#	8005#	8034#	8040#	8078#	8086#	8139#	8142#	8172#
8178#	8216#	8224#	8277#	8280#	8320#	8326#	8364#	8374#	8408#	8443#	8482#	8488#
8526#	8535#	8601#	8604#	8635#	8641#	8679#	8693#	8758#	8763#	8812#	8818#	8856#
8864#	8984#	8987#	9039#	9045#	9083#	9096#	9156#	9179#	9208#	9214#	9252#	9265#
9326#	9349#	9381#	9387#	9425#	9433#	9490#	9493#	9546#	9552#	9590#	9607#	9663#
9688#	9735#	9741#	9779#	9792#	9844#	9871#	9898#	9904#	9942#	9959#	10006#	10032#
10137#	10143#	10181#	10205#	10290#	10293#	10360#	10366#	10404#	10416#	10472#	10482#	10515#
10521#	10559#	10566#	10625#	10628#	10675#	10681#	10719#	10732#	10779#	10785#	10798#	10807#
10842#	10848#	10886#	10892#	10957#	10960#	11014#	11020#	11058#	11068#	11133#	11141#	11182#
11188#	11226#	11241#	11313#	11322#	11371#	11377#	11415#	11428#	11438#	11501#	11514#	11524#
11579#	11585#	11623#	11634#	11719#	11744#	11794#	11800#	11836#	11845#	11927#	11941#	12030#
12036#	12074#	12091#	12161#	12167#	12235#	12241#	12311#	12314#	12373#	12379#	12417#	12425#
12485#	12492#	12556#	12565#	12620#	12626#	12664#	12680#	12762#	12765#	12787#	13014#	13139#
13152#	13179#											
1420#	15179											
1582#	1590	2915#	2924	2928#	2940	2944#	2961	2965#	2974	2978#	2987	2991#
3004	3008#	3023	3027#	3040	3044#	3053	3057#	3074	3078#	3100	3104#	3113
3117#	3125	3129#	3138	3142#	3150	3154#	3175	3179#	3188	3192#	3208	3212#
3222	3226#	3237	3241#	3265	3269#	3293	3297#	3307	3311#	3336	3340#	3363
3367#	3390	3394#	3425	3429#	3437	3441#	3475	3479#	3523	3527#	3542	3546#
3562	3566#	3582	3586#	3610	3614#	3629	3633#	3670	3674#	3697	3701#	3710
3714#	3730	3734#	3757	3761#	3769	3773#	3781	3785#	3793	3797#	3805	3809#
3832	3836#	3852	3856#	3894	3898#	3906	3910#	3945	3949#	3969	3973#	3981
4716#	4721	4733#	4740	4749#	4845	4860#	4862	4872#	4904	4909#	4914	4920#
4925	4943#	4946	4962#	4966	4988#	4994	5015#	5100	5118#	5254	5270#	5302
5318#	5355	5377#	5559	5582#	5606	5626#	5660	5689#	5780	5802#	5865	6222#
6357	6398#	6596	6637#	6789	6821#	6935	6965#	7055	7094#	7246	7276#	7382
7422#	7561	7587#	7714	7746#	7873	7899#	8005	8034#	8142	8172#	8280	8320#
8443	8482#	8604	8635#	8763	8812#	8987	9039#	9179	9208#	9349	9381#	9493
9546#	9688	9735#	9871	9898#	10032	10137#	10293	10360#	10482	10515#	10628	10675#
10807	10842#	10960	11014#	11141	11182#	11322	11371#	11524	11579#	11744	11794#	11941
12030#	12314	12373#	12565	12620#	12765	12787#	13014	13139#	13152			
5024#	5050	5054#	5094	5127#	5153	5157#	5248	5326#	5349	5382#	5404	5409#
5556	5587#	5601	5629#	5641	5645#	5657	5691#	5710	5714#	5732	5736#	5777
6224#	6271	6275#	6354	6401#	6511	6515#	6585	6641#	6678	6681#	6698	6701#
6725	6728#	6745	6751#	6780	6827#	6858	6867#	6932	6970#	6996	7002#	7040
7100#	7145	7162#	7212	7218#	7239	7282#	7320	7329#	7379	7428#	7466	7476#
7558	7593#	7631	7646#	7696	7752#	7790	7800#	7852	7905#	7936	7946#	8002
8040#	8078	8086#	8139	8178#	8216	8224#	8277	8326#	8364	8374#	8440	8488#
8526	8535#	8601	8641#	8679	8693#	8758	8818#	8856	8864#	8984	9045#	9083
9096#	9156	9214#	9252	9265#	9326	9387#	9425	9433#	9490	9552#	9590	9607#
9663	9741#	9779	9792#	9844	9904#	9942	9959#	10006	10143#	10181	10205#	10290
10366#	10404	10416#	10472	10521#	10559	10566#	10625	10681#	10719	10732#	10779	10785#
10798	10848#	10886	10892#	10957	11020#	11058	11068#	11133	11188#	11226	11241#	11313
11377#	11415	11428#	11514	11585#	11623	11634#	11719	11800#	11838	11845#	11927	12036#
12074	12091#	12161	12167#	12235	12241#	12311	12379#	12417	12425#	12485	12492#	12556
12626#	12664	12680#	12762									
5059#	5074	5077#	5091	5162#	5177	5180#	5194	5197#	5211	5214#	5228	5231#
5245	5414#	5430	5442#	5454	5468#	5485	5497#	5509	5518#	5534	5537#	5553
5760#	5774	6229#	6251	6254#	6268	6288#	6317	6320#	6334	6406#	6448	6451#
6465	6468#	6491	6494#	6508	6520#	6552	6555#	6566	11438#	11501		
TSPTMU= 000000	1416#											
TSSAVL= 177777	1416#											

T&NSO = 000000  
T&NS1 = 000004

T&NS2 = 000003

T&NS3 = 000003

T&PTMU= 000000  
T&SAVL= 177777

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 305  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

TSSEGL= 177777

1416#	5059#	5074#	5076	5077#	5091#	5093	5162#	5177#	5179	5180#	5194#	5196
5197#	5211#	5213	5214#	5228#	5230	5231#	5245#	5247	5326#	5349#	5351	5414#
5430#	5432	5442#	5454#	5456	5468#	5485#	5487	5497#	5509#	5511	5518#	5534#
5536	5537#	5553#	5555	5587#	5601#	5603	5629#	5641#	5643	5645#	5657#	5659
5760#	5774#	5776	6229#	6251#	6253	6254#	6268#	6270	6288#	6317#	6319	6320#
6334#	6336	6406#	6448#	6450	6451#	6465#	6467	6468#	6491#	6493	6494#	6508#
6510	6520#	6552#	6554	6555#	6566#	6568	6641#	6678#	6680	6681#	6698#	6700
6701#	6725#	6727	6728#	6745#	6747	6751#	6780#	6782	6827#	6858#	6860	6867#
6932#	6934	6970#	6996#	6998	7002#	7040#	7042	7100#	7145#	7147	7162#	7212#
7214	7218#	7239#	7241	7282#	7320#	7322	7329#	7379#	7381	7428#	7466#	7468
7476#	7558#	7560	7593#	7631#	7633	7646#	7696#	7698	7752#	7790#	7792	7800#
7852#	7854	7905#	7936#	7938	7946#	8002#	8004	8040#	8078#	8080	8086#	8139#
8141	8178#	8216#	8218	8224#	8277#	8279	8326#	8364#	8366	8374#	8440#	8442
8488#	8526#	8528	8535#	8601#	8603	8641#	8679#	8681	8693#	8758#	8760	8818#
8856#	8858	8864#	8984#	8986	9045#	9083#	9085	9096#	9156#	9158	9214#	9252#
9254	9265#	9326#	9328	9387#	9425#	9427	9433#	9490#	9492	9552#	9590#	9592
9607#	9663#	9665	9741#	9779#	9781	9792#	9844#	9846	9904#	9942#	9944	9959#
10006#	10008	10143#	10181#	10183	10205#	10290#	10292	10366#	10404#	10406	10416#	10472#
10474	10521#	10559#	10561	10566#	10625#	10627	10681#	10719#	10721	10732#	10779#	10781
10785#	10798#	10800	10848#	10886#	10888	10892#	10957#	10959	11020#	11058#	11060	11068#
11133#	11135	11188#	11226#	11228	11241#	11313#	11315	11377#	11415#	11417	11428#	11438#
11501#	11503	11514#	11516	11585#	11623#	11625	11634#	11719#	11721	11800#	11838#	11840
11845#	11927#	11929	12036#	12074#	12076	12091#	12161#	12163	12167#	12235#	12237	12241#
12311#	12313	12379#	12417#	12419	12425#	12485#	12487	12492#	12556#	12558	12626#	12664#
12666	12680#	12762#	12764									

TSSEK0= 010001

5059#	5074	5077#	5091	5162#	5177	5180#	5194	5197#	5211	5214#	5228	5231#
5245	5326#	5349	5414#	5430	5442#	5454	5468#	5485	5497#	5509	5518#	5534
5537#	5553	5587#	5601	5629#	5641	5645#	5657	5760#	5774	6229#	6251	6254#
5268	6288#	6317	6320#	6334	6406#	6448	6451#	6465	6468#	6491	6494#	6508
6520#	6552	6555#	6566	6641#	6678	6681#	6698	6701#	6725	6728#	6745	6751#
6780	6827#	6858	6867#	6932	6970#	6996	7002#	7040	7100#	7145	7162#	7212
7218#	7239	7282#	7320	7329#	7379	7428#	7466	7476#	7558	7593#	7631	7646#
7696	7752#	7790	7800#	7852	7905#	7936	7946#	8002	8040#	8078	8086#	8139
8178#	8216	8224#	8277	8326#	8364	8374#	8440	8488#	8526	8535#	8601	8641#
8679	8693#	8758	8818#	8856	8864#	8984	9045#	9083	9096#	9156	9214#	9252
9265#	9326	9387#	9425	9433#	9490	9552#	9590	9607#	9663	9741#	9779	9792#
9844	9904#	9942	9959#	10026	10143#	10181	10205#	10290	10366#	10404	10416#	10472
10521#	10559	10566#	10625	10681#	10719	10732#	10779	10785#	10798	10848#	10886	10892#
10957	11020#	11058	11068#	11133	11188#	11226	11241#	11313	11377#	11415	11428#	11514
11585#	11623	11634#	11719	11800#	11838	11845#	11927	12036#	12074	12091#	12161	12167#
12235	12241#	12311	12379#	12417	12425#	12485	12492#	12556	12626#	12664	12680#	12762

TSSEK1= 010002  
TSSUBN= 000000

11438#	11501											
1416#	5014#	5023#	5053#	5117#	5126#	5156#	5269#	5317#	5376#	5381#	5408#	5581#
5625#	5688#	5690#	5713#	5735#	5801#	6221#	6223#	6274#	6397#	6400#	6514#	6636#
6820#	6964#	7093#	7275#	7421#	7586#	7745#	7898#	8033#	8171#	8319#	8481#	8634#
8811#	9038#	9207#	9380#	9545#	9734#	9897#	10136#	10359#	10514#	10674#	10841#	11013#
11181#	11370#	11578#	11793#	12029#	12372#	12619#	12786#					

TSTAGL= 177777  
TSTAGN= 010172

1416#	1582#	2915#	2928#	2944#	2965#	2978#	2991#	3008#	3027#	3044#	3057#	3078#
3104#	3117#	3129#	3142#	3154#	3179#	3192#	3212#	3226#	3241#	3269#	3297#	3311#
3340#	3367#	3394#	3429#	3441#	3479#	3527#	3546#	3566#	3586#	3614#	3633#	3674#
3701#	3714#	3734#	3761#	3773#	3785#	3797#	3809#	3836#	3856#	3898#	3910#	3949#
3973#	4716#	4733#	4749#	4860#	4872#	4909#	4920#	4943#	4962#	4988#	5015#	5024#
5054#	5118#	5127#	5157#	5270#	5318#	5377#	5382#	5409#	5582#	5626#	5689#	5691#
5714#	5736#	5802#	6222#	6224#	6275#	6398#	6401#	6515#	6637#	6821#	6965#	7094#
7276#	7422#	7587#	7746#	7899#	8034#	8172#	8320#	8482#	8635#	8812#	9039#	9208#

PARAMETER CODING MACY11 30A('052) 07-APR-83 17:13 PAGE 306  
 CZUAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

T\$TEMP= 000000

9381#	9546#	9735#	9898#	10137#	10360#	10515#	10675#	10842#	11014#	11182#	11371#	11579#
11794#	12030#	12373#	12620#	12787#	13139#							
1525#	1526#	1527#	1528#	1529#	1530#	1531#	1532#	1533#	1534#	1535#	1536#	1537#
1538#	1539#	1540#	1541#	1542#	1543#	1544#	1545#	1546#	1547#	1548#	1549#	1550#
1551#	1552#	1553#	1554#	1555#	1556#	1557#	1558#	1559#	1560#	1561#	1562#	1563#
1564#	1565#	1566#	1567#	1568#	1569#	1570#	1571#	1590#	2924#	2940#	2961#	2974#
2987#	3004#	3023#	3040#	3053#	3074#	3100#	3113#	3125#	3138#	3150#	3175#	3188#
3208#	3222#	3237#	3265#	3293#	3307#	3336#	3363#	3390#	3425#	3437#	3475#	3523#
3542#	3562#	3582#	3610#	3629#	3670#	3697#	3710#	3730#	3757#	3769#	3781#	3793#
3805#	3832#	3852#	3894#	3906#	3945#	3969#	3981#	4718#	4719#	4721#	4740#	4845#
4862#	4891#	4892#	4904#	4911#	4912#	4914#	4922#	4923#	4925#	4946#	4966#	4994#
5050#	5074#	5091#	5094#	5100#	5153#	5177#	5194#	5211#	5228#	5245#	5248#	5254#
5302#	5349#	5355#	5400#	5401#	5404#	5430#	5454#	5485#	5509#	5534#	5553#	5556#
5559#	5601#	5606#	5641#	5657#	5660#	5710#	5732#	5756#	5757#	5774#	5777#	5780#
5810#	5811#	5851#	5852#	5865#	6247#	6248#	6251#	6268#	6271#	6313#	6314#	6317#
6334#	6354#	6357#	6444#	6445#	6448#	6465#	6487#	6488#	6491#	6508#	6511#	6548#
6549#	6552#	6566#	6585#	6596#	6674#	6675#	6678#	6694#	6695#	6698#	6721#	6722#
6725#	6741#	6742#	6745#	6780#	6789#	6858#	6876#	6877#	6932#	6935#	6996#	7023#
7024#	7040#	7055#	7109#	7110#	7125#	7126#	7135#	7136#	7145#	7171#	7172#	7212#
7239#	7246#	7291#	7292#	7306#	7307#	7316#	7317#	7320#	7338#	7339#	7350#	7351#
7379#	7382#	7437#	7438#	7452#	7453#	7462#	7463#	7466#	7485#	7486#	7506#	7507#
7515#	7516#	7558#	7561#	7602#	7603#	7617#	7618#	7627#	7628#	7631#	7655#	7656#
7670#	7671#	7679#	7680#	7696#	7709#	7710#	7714#	7761#	7762#	7776#	7777#	7786#
7787#	7790#	7809#	7810#	7823#	7824#	7832#	7833#	7852#	7865#	7866#	7873#	7936#
7956#	7957#	7970#	7971#	8002#	8005#	8049#	8050#	8064#	8065#	8074#	8075#	8078#
8095#	8096#	8116#	8117#	8125#	8126#	8139#	8142#	8187#	8188#	8202#	8203#	8212#
8213#	8216#	8233#	8234#	8254#	8255#	8263#	8264#	8277#	8280#	8335#	8336#	8350#
8351#	8360#	8361#	8364#	8383#	8384#	8402#	8403#	8411#	8412#	8440#	8443#	8497#
8498#	8512#	8513#	8522#	8523#	8526#	8544#	8545#	8563#	8564#	8572#	8573#	8601#
8604#	8650#	8651#	8665#	8666#	8675#	8676#	8679#	8706#	8707#	8720#	8721#	8729#
8730#	8758#	8763#	8827#	8828#	8842#	8843#	8852#	8853#	8856#	8873#	8874#	8891#
8892#	8900#	8901#	8984#	8987#	9054#	9055#	9069#	9070#	9079#	9080#	9083#	9109#
9110#	9123#	9124#	9132#	9133#	9156#	9169#	9170#	9179#	9223#	9224#	9238#	9239#
9248#	9249#	9252#	9278#	9279#	9292#	9293#	9301#	9302#	9326#	9339#	9340#	9349#
9396#	9397#	9411#	9412#	9421#	9422#	9425#	9442#	9443#	9457#	9458#	9466#	9467#
9490#	9493#	9561#	9562#	9576#	9577#	9586#	9587#	9590#	9619#	9620#	9633#	9634#
9642#	9643#	9663#	9676#	9677#	9688#	9750#	9751#	9765#	9766#	9775#	9776#	9779#
9805#	9806#	9819#	9820#	9828#	9829#	9844#	9857#	9858#	9871#	9913#	9914#	9928#
9929#	9938#	9939#	9942#	9968#	9969#	9982#	9983#	9991#	9992#	10006#	10019#	10020#
10032#	10152#	10153#	10167#	10168#	10177#	10178#	10181#	10219#	10220#	10233#	10234#	10242#
10243#	10290#	10293#	10375#	10376#	10390#	10391#	10400#	10401#	10404#	10429#	10430#	10452#
10453#	10472#	10482#	10530#	10531#	10545#	10546#	10555#	10556#	10559#	10575#	10576#	10591#
10592#	10600#	10601#	10625#	10628#	10690#	10691#	10705#	10706#	10715#	10716#	10719#	10741#
10742#	10756#	10757#	10765#	10766#	10779#	10794#	10795#	10798#	10807#	10857#	10858#	10872#
10873#	10882#	10883#	10886#	10901#	10902#	10923#	10924#	10932#	10933#	10957#	10960#	11029#
11030#	11044#	11045#	11054#	11055#	11058#	11080#	11081#	11099#	11100#	11108#	11109#	11133#
11141#	11197#	11198#	11212#	11213#	11222#	11223#	11226#	11250#	11251#	11269#	11270#	11278#
11279#	11313#	11322#	11386#	11387#	11401#	11402#	11411#	11412#	11415#	11453#	11454#	11467#
11468#	11476#	11477#	11501#	11514#	11524#	11594#	11595#	11609#	11610#	11619#	11620#	11623#
11652#	11653#	11666#	11667#	11675#	11676#	11719#	11732#	11733#	11744#	11809#	11810#	11824#
11825#	11834#	11835#	11838#	11863#	11864#	11877#	11878#	11886#	11887#	11927#	11941#	12045#
12046#	12060#	12061#	12070#	12071#	12074#	12104#	12105#	12118#	12119#	12127#	12128#	12161#
12181#	12182#	12201#	12202#	12235#	12236#	12254#	12267#	12268#	12276#	12277#	12311#	12314#
12388#	12389#	12403#	12404#	12413#	12414#	12417#	12437#	12438#	12451#	12452#	12460#	12461#
12485#	12504#	12505#	12518#	12519#	12527#	12528#	12542#	12543#	12556#	12565#	12635#	12636#
12650#	12651#	12660#	12661#	12664#	12689#	12690#	12703#	12704#	12712#	12713#	12725#	12726#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 307  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

	12756#	12757	12762#	12765#	12790#	12791	12796#	12797	12817#	12818	12830#	12831	12851#
	12852	12864#	12865	12895#	12896	12908#	12909	12940#	12941	12952#	12953	13014#	13142#
	13147#	13152#	13179#										
TSTEST= 000056	1416#	5014#	5023	5053	5117#	5126	5156	5269#	5317#	5376#	5381	5408	5581#
	5625#	5688#	5690	5713	5755	5801#	6221#	6223	6274	6397#	6400	6514	6636#
	6820#	6964#	7093#	7275#	7421#	7586#	7745#	7898#	8033#	8171#	8319#	8481#	8634#
	8811#	9038#	9207#	9380#	9545#	9734#	9897#	10136#	10359#	10514#	10674#	10841#	11013#
TSTSTM= 177777	11181#	11370#	11578#	11793#	12029#	12372#	12619#	12786#					
	1416#	2921	2925	2936	2941	2950	2958	2962	2970	2975	2983	2988	3000
	3005	3013	3020	3024	3036	3041	3049	3054	3062	3071	3075	3083	3090
	3097	3101	3109	3114	3122	3126	3135	3139	3147	3151	3159	3165	3172
	3176	3185	3189	3198	3205	3209	3219	3223	3234	3238	3247	3254	3262
	3266	3275	3282	3290	3294	3304	3308	3317	3325	3333	3337	3345	3352
	3360	3364	3372	3379	3387	3391	3401	3411	3421	3426	3434	3438	3446
	3452	3462	3469	3476	3486	3493	3499	3506	3513	3520	3524	3532	3539
	3543	3551	3559	3563	3571	3579	3583	3593	3600	3607	3611	3619	3626
	3630	3638	3645	3652	3659	3667	3671	3679	3687	3694	3698	3707	3711
	3720	3727	3731	3740	3746	3754	3758	3766	3770	3778	3782	3790	3794
	3802	3806	3814	3821	3829	3833	3841	3849	3853	3862	3869	3876	3883
	3891	3895	3903	3907	3917	3925	3933	3941	3946	3954	3960	3966	3970
	3978	3982	4234	4248	4277	4390	4427	4462	4546	4557	4579	4645	4660
	4673	4688	4701	4722	4755	4760	4775	4780	4785	4791	4799	4813	4823
	4842	4846	4863	4885	4891	4905	4915	4926	5020	5024	5034	5039	5042
	5045	5047	5051	5054	5059	5068	5075	5077	5085	5092	5095	5098	5101
	5123	5127	5137	5142	5145	5148	5150	5154	5157	5162	5171	5178	5180
	5188	5195	5197	5205	5212	5214	5222	5229	5231	5239	5246	5249	5252
	5255	5275	5283	5288	5291	5294	5296	5299	5303	5323	5326	5333	5338
	5341	5344	5346	5350	5353	5356	5382	5395	5400	5405	5409	5414	5424
	5431	5442	5448	5455	5468	5479	5486	5497	5503	5510	5518	5528	5535
	5537	5547	5554	5557	5560	5587	5595	5602	5607	5629	5635	5642	5645
	5651	5658	5661	5691	5704	5711	5714	5726	5733	5736	5751	5756	5760
	5768	5775	5778	5781	5805	5810	5822	5846	5851	5859	5866	6224	6229
	6242	6247	6252	6254	6262	6269	6272	6275	6288	6308	6313	6318	6320
	6328	6335	6348	6355	6358	6401	6406	6439	6444	6449	6451	6459	6466
	6468	6482	6487	6492	6494	6502	6509	6512	6515	6520	6543	6548	6553
	6555	6560	6567	6577	6586	6597	6641	6669	6674	6679	6681	6689	6694
	6699	6701	6716	6721	6726	6728	6736	6741	6746	6751	6764	6774	6781
	6790	6827	6843	6851	6859	6867	6871	6876	6892	6899	6919	6926	6933
	6936	6970	6976	6990	6997	7002	7009	7018	7023	7034	7041	7044	7049
	7056	7100	7109	7120	7125	7130	7135	7142	7146	7162	7165	7171	7182
	7194	7206	7213	7218	7233	7240	7247	7282	7291	7301	7306	7311	7316
	7321	7329	7333	7338	7345	7350	7366	7373	7380	7383	7428	7437	7447
	7452	7457	7462	7467	7476	7480	7485	7501	7506	7510	7515	7533	7540
	7552	7559	7562	7593	7602	7612	7617	7622	7627	7632	7646	7650	7655
	7665	7670	7674	7679	7691	7697	7704	7709	7715	7752	7761	7771	7776
	7781	7786	7791	7800	7804	7809	7818	7823	7827	7832	7846	7853	7860
	7865	7874	7905	7921	7929	7937	7946	7951	7956	7965	7970	7974	7991
	8003	8006	8040	8049	8059	8064	8069	8074	8079	8086	8090	8095	8104
	8111	8116	8120	8125	8133	8140	8143	8178	8187	8197	8202	8207	8212
	8217	8224	8228	8233	8242	8249	8254	8258	8263	8271	8278	8281	8326
	8335	8345	8350	8355	8360	8365	8374	8378	8383	8397	8402	8406	8411
	8424	8434	8441	8444	8488	8497	8507	8512	8517	8522	8527	8535	8539
	8544	8558	8563	8567	8572	8585	8595	8602	8605	8641	8650	8660	8665
	8670	8675	8680	8693	8701	8706	8715	8720	8724	8729	8742	8752	8759
	8764	8818	8827	8837	8842	8847	8852	8857	8864	8868	8873	8886	8891
	8895	8900	8916	8929	8938	8950	8959	8968	8978	8985	8988	9045	9054

9064	9069	9074	9079	9084	9096	9104	9109	9118	9123	9127	9132	9150	
9157	9164	9169	9180	9214	9223	9233	9238	9243	9248	9253	9265	9273	
9278	9287	9292	9296	9301	9320	9327	9334	9339	9350	9387	9396	9406	
9411	9416	9421	9426	9433	9437	9442	9452	9457	9461	9466	9474	9484	
9491	9494	9552	9561	9571	9576	9581	9586	9591	9607	9614	9619	9628	
9633	9637	9642	9657	9664	9671	9676	9689	9741	9750	9760	9765	9770	
9775	9780	9792	9800	9805	9814	9819	9823	9828	9838	9845	9852	9857	
9872	9904	9913	9923	9928	9933	9938	9943	9959	9963	9968	9977	9982	
9986	9991	10000	10007	10014	10019	10033	10143	10152	10162	10167	10172	10177	
10182	10205	10214	10219	10228	10233	10237	10242	10254	10270	10284	10291	10294	
10366	10375	10385	10390	10395	10400	10405	10416	10424	10429	10447	10452	10455	
10466	10473	10483	10521	10530	10540	10545	10550	10555	10560	10566	10570	10575	
10586	10591	10595	10600	10609	10619	10626	10629	10681	10690	10700	10705	10710	
10715	10720	10732	10736	10741	10751	10756	10760	10765	10773	10780	10785	10789	
10794	10799	10808	10848	10857	10867	10872	10877	10882	10887	10892	10896	10901	
10918	10923	10927	10932	10941	10951	10958	10961	11020	11029	11039	11044	11049	
11054	11059	11068	11075	11080	11094	11099	11103	11108	11116	11126	11134	11142	
11188	11197	11207	11212	11217	11222	11227	11241	11245	11250	11264	11269	11273	
11278	11289	11297	11307	11314	11323	11377	11386	11396	11401	11406	11411	11416	
11428	11438	11448	11453	11462	11467	11471	11476	11485	11495	11502	11515	11525	
11585	11594	11604	11609	11614	11619	11624	11634	11647	11652	11661	11666	11670	
11675	11687	11702	11713	11720	11727	11732	11745	11800	11809	11819	11824	11829	
11834	11839	11845	11858	11863	11872	11877	11881	11886	11895	11910	11921	11928	
11935	11942	12036	12045	12055	12060	12065	12070	12075	12091	12099	12104	12113	
12118	12122	12127	12144	12155	12162	12167	12176	12181	12190	12196	12201	12216	
12227	12236	12241	12248	12253	12262	12267	12271	12276	12295	12305	12312	12315	
12379	12388	12398	12403	12408	12413	12418	12425	12432	12437	12446	12451	12455	
12460	12469	12479	12486	12492	12499	12504	12513	12518	12522	12527	12537	12542	
12550	12557	12566	12626	12635	12645	12650	12655	12660	12665	12680	12684	12689	
12698	12703	12707	12712	12720	12725	12751	12756	12763	12766	12790	12796	12812	
12817	12825	12830	12846	12851	12859	12864	12890	12895	12903	12908	12935	12940	
12947	12952	12981	12988	12995	13002	13009	13015						
TSTSTS= 000001	1416#	5015#	5118#	5270#	5318#	5377#	5582#	5626#	5689#	5802#	6222#	6398#	6637#
	6821#	6965#	7094#	7276#	7422#	7587#	7746#	7899#	8034#	8172#	8320#	8482#	8635#
	8812#	9039#	9208#	9381#	9546#	9735#	9898#	10137#	10360#	10515#	10675#	10842#	11014#
	11182#	11371#	11579#	11794#	12030#	12373#	12620#	12787#					
TSSAU = 010072	4920#	4922	4925										
TSSAUT= 010067	4860#	4862											
TSSCLE= 010070	4872#	4891	4904										
TSSDU = 010071	4909#	4911	4914										
TSSHAR= 010171	13139#	13153											
TSSHW = 010000	1582#	1590											
TSSINI= 010066	4749#	4845											
TSSMSG= 010063	2915#	2924	2928#	2940	2944#	2961	2965#	2974	2978#	2987	2991#	3004	3008#
	3023	3027#	3040	3044#	3053	3057#	3074	3078#	3100	3104#	3113	3117#	3125
	3129#	3138	3142#	3150	3154#	3175	3179#	3188	3192#	3208	3212#	3222	3226#
	3237	3241#	3265	3269#	3293	3297#	3307	3311#	3336	3340#	3363	3367#	3390
	3394#	3425	3429#	3437	3441#	3475	3479#	3523	3527#	3542	3546#	3562	3566#
	3582	3586#	3610	3614#	3629	3633#	3670	3674#	3697	3701#	3710	3714#	3730
	3734#	3757	3761#	3769	3773#	3781	3785#	3793	3797#	3805	3809#	3832	3836#
	3852	3856#	3894	3898#	3906	3910#	3945	3949#	3969	3973#	3981		
TSSPRO= 010065	4733#												
TSSRPT= 010064	4716#	4718	4721										
TSSSEG= 010001	5059#	5074#	5077#	5091#	5162#	5177#	5180#	5194#	5197#	5211#	5214#	5228#	5231#
	5245#	5326#	5349#	5414#	5430#	5442#	5454#	5468#	5485#	5497#	5509#	5518#	5534#
	5537#	5553#	5587#	5601#	5629#	5641#	5645#	5657#	5760#	5774#	6229#	6251#	6254#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 309  
 CZUAAB.MAC 07-APP-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

6268#	6288#	6317#	6320#	6334#	6406#	6448#	6451#	6465#	6468#	6491#	6494#	6508#
6520#	6552#	6555#	6566#	6641#	6678#	6681#	6698#	6701#	6725#	6728#	6745#	6751#
6780#	6827#	6858#	6867#	6932#	6970#	6996#	7002#	7040#	7100#	7145#	7162#	7212#
7218#	7239#	7282#	7320#	7329#	7379#	7428#	7466#	7476#	7558#	7593#	7631#	7646#
7696#	7752#	7790#	7800#	7852#	7905#	7936#	7946#	8002#	8040#	8078#	8086#	8139#
8178#	8216#	8224#	8277#	8326#	8364#	8374#	8440#	8488#	8526#	8535#	8601#	8641#
8679#	8693#	8758#	8818#	8856#	8864#	8984#	9045#	9083#	9096#	9156#	9214#	9252#
9265#	9326#	9387#	9425#	9433#	9490#	9552#	9590#	9607#	9663#	9711#	9779#	9792#
9844#	9904#	9942#	9959#	10006#	10143#	10181#	10205#	10290#	10366#	10404#	10416#	10472#
10521#	10559#	10566#	10625#	10681#	10719#	10732#	10779#	10785#	10798#	10848#	10886#	10892#
10957#	11020#	11058#	11068#	11133#	11188#	11226#	11241#	11313#	11377#	11415#	11428#	11438#
11501#	11514#	11585#	11623#	11634#	11719#	11800#	11838#	11845#	11927#	12036#	12074#	12091#
12161#	12167#	12235#	12241#	12311#	12379#	12417#	12425#	12485#	12492#	12556#	12626#	12664#
12680#	12762#											
4943#	4946	4962#	4966	4988#	4994							
5024#	5050	5054#	5094	5127#	5153	5157#	5248	5382#	5404	5409#	5556	5691#
5710	5714#	5732	5736#	5777	6224#	6271	6275#	6354	6401#	6511	6515#	6585
5015#	5100	5118#	5254	5270#	5302	5318#	5355	5377#	5400	5559	5582#	5606
5626#	5660	5689#	5756	5780	5802#	5810	5851	5865	6222#	6247	6313	6357
6398#	6444	6487	6548	6596	6637#	6674	6694	6721	6741	6789	6821#	6876
6935	6965#	7023	7055	7094#	7109	7125	7135	7171	7246	7276#	7291	7306
7316	7338	7350	7382	7422#	7437	7452	7462	7485	7506	7515	7561	7587#
7602	7617	7627	7655	7670	7679	7709	7714	7746#	7761	7776	7786	7809
7823	7832	7865	7873	7899#	7956	7970	8005	8034#	8049	8064	8074	8095
8116	8125	8142	8172#	8187	8202	8212	8233	8254	8263	8280	8320#	8335
8350	8360	8383	8402	8411	8443	8482#	8497	8512	8522	8544	8563	8572
8604	8635#	8650	8665	8675	8706	8720	8729	8763	8812#	8827	8842	8852
8873	8891	8900	8987	9039#	9054	9069	9079	9109	9123	9132	9169	9179
9208#	9223	9238	9248	9278	9292	9301	9339	9349	9381#	9396	9411	9421
9442	9457	9466	9493	9546#	9561	9576	9586	9619	9633	9642	9676	9688
9735#	9750	9765	9775	9805	9819	9828	9857	9871	9898#	9913	9928	9938
9968	9982	9991	10019	10032	10137#	10152	10167	10177	10219	10233	10242	10293
10360#	10375	10390	10400	10429	10452	10482	10515#	10530	10545	10555	10575	10591
10600	10628	10675#	10690	10705	10715	10741	10756	10765	10794	10807	10842#	10857
10872	10882	10901	10923	10932	10960	11014#	11029	11044	11054	11080	11099	11108
11141	11182#	11197	11212	11222	11250	11269	11278	11322	11371#	11386	11401	11411
11453	11467	11476	11524	11579#	11594	11609	11619	11652	11666	11675	11732	11744
11794#	11809	11824	11834	11863	11877	11886	11941	12030#	12045	12060	12070	12104
12118	12127	12181	12201	12253	12267	12276	12314	12335#	12388	12403	12413	12437
12451	12460	12504	12518	12527	12542	12565	12620#	12635	12650	12660	12689	12703
12712	12725	12756	12765	12787#	12790	12796	12817	12830	12851	12864	12895	12908
12940	12952	13014										

T1	022076RG	002	1525	5014#
T1.1	022124R	002	5023#	
T1.2	022202R	002	5053#	
T10	027124RG	002	1534	6221#
T10.1	027124R	002	6223#	
T10.2	027256R	002	6274#	
T11	027540RG	002	1535	6397#
T11.1	027544R	002	6400#	6593
T11.2	030076R	002	6514#	
T12	030326RG	002	1536	6636#
T13	031006RG	002	1537	6820#
T14	031424RG	002	1538	6964#
T15	031714RG	002	1539	7093#
T16	032422RG	002	1540	7275#







67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 312  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- USER SYMBOLS

SNOP	001405RG	002	2054#	5749	6988															
SNSSET	001277RG	002	2037#	4535	5393	5422	5747	5818	6238	6304	6435	6478	6539	6665	6712					
			6760	6839	6986	7116	7157	7297	7443	7608	7767	7917	8055	8193	8341					
			8503	8656	8833	9060	9229	9402	9567	9756	9919	10158	10381	10536	10696					
			10863	11035	11203	11392	11600	11815	12051	12394	12641	12808	12842	12886	12931					
SPARER	001502RG	002	2067#	6914																
SPATCH	053366RG	002	13167#																	
SPCEI	000747RG	002	1859	1979#	4349															
SPCTO	001107RG	002	1869	2001#																
SPDNDM	001417RG	002	2057#																	
SRCEI	001010RG	002	1855	1987#	4377															
SRESET	001440RG	002	2061#																	
SRMTC	001120RG	002	1865	2003#																
SASET	001054RG	002	1850	1995#																
SRXI	000760RG	002	1858	1981#	4356															
SSERI	000736RG	002	1860	1977#	4342															
SSET	001307RG	002	2039#																	
SSLFT	001373RG	002	2052#	5820																
SSTOP	001433RG	002	2060#																	
SSTRY	001411RG	002	2055#																	
STXI	000770RG	002	1857	1983#	4363															
EXPWR	001065RG	002	1877	1997#	5545															
.	= 000000R	012	1937#	2900#	4719	4892	4901#	4912	4923	5401	5757	5811	5852	6248	6314					
			6445	6488	6549	6675	6695	6722	6742	6877	7024	7110	7126	7136	7172					
			7292	7307	7317	7339	7351	7438	7453	7463	7486	7507	7516	7603	7618					
			7628	7656	7671	7680	7710	7762	7777	7787	7810	7824	7833	7866	7957					
			7971	8050	8065	8075	8096	8117	8126	8188	8203	8213	8234	8255	8264					
			8336	8351	8361	8384	8403	8412	8498	8513	8523	8545	8564	8573	8651					
			8666	8676	8707	8721	8730	8828	8843	8853	8874	8892	8901	9055	9070					
			9080	9110	9124	9133	9170	9224	9239	9249	9279	9293	9302	9340	9397					
			9412	9422	9443	9458	9467	9562	9577	9587	9620	9634	9643	9677	9751					
			9766	7776	9806	9820	9829	9858	9914	9929	9939	9969	9983	9992	10020					
			10153	10168	10178	10220	10234	10243	10376	10391	10401	10430	10453	10531	10546					
			10556	10576	10592	10601	10691	10706	10716	10742	10757	10766	10795	10858	10873					
			10883	10902	10924	10933	11030	11045	11055	11081	11100	11109	11198	11213	11223					
			11251	11270	11279	11387	11402	11412	11454	11468	11477	11595	11610	11620	11653					
			11667	11676	11733	11810	11825	11835	11864	11878	11887	12046	12061	12071	12105					
			12119	12128	12182	12202	12254	12268	12277	12389	12404	12414	12438	12452	12461					
			12505	12519	12528	12543	12636	12651	12661	12690	12704	12713	12726	12757	12791					
			12797	12818	12831	12852	12865	12896	12909	12941	12953	13065#	13166#	13167#						



67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 315  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

ENDHRD	1#	1416#	13151												
ENDHW	1#	1416#	1589												
ENDINI	1#	1416#	4844												
ENDROD	1#	1416#	13178												
ENDMSG	1#	1416#	2923	2939	2960	2973	2986	3003	3022	3039	3052	3073	3099	3112	3124
		3137	3149	3174	3187	3207	3221	3236	3264	3292	3306	3335	3362	3389	3424
		3474	3522	3541	3561	3581	3609	3628	3669	3696	3709	3729	3756	3768	3780
		3804	3831	3851	3893	3905	3944	3968	3980						3792
ENDPRO	1#	1416#	4739												
ENDPTA	1#	1416#													
ENDRPT	1#	1416#	4720												
ENDSEG	1#	1416#	5073	5090	5176	5193	5210	5227	5244	5348	5429	5453	5484	5508	5533
		5552	5600	5640	5656	5773	6250	6267	6316	6333	6447	6464	6490	6507	6551
		6677	6697	6724	6744	6779	6857	6931	6995	7039	7144	7211	7238	7319	7378
		7557	7630	7695	7789	7851	7935	8001	8077	8138	8215	8276	8363	8439	8525
		8678	8757	8855	8983	9082	9155	9251	9325	9424	9489	9589	9662	9778	9843
		10005	10180	10289	10403	10471	10558	10624	10718	10778	10797	10885	10956	11057	11132
		11312	11414	11500	11513	11622	11718	11837	11926	12073	12160	12234	12310	12416	12484
		12663	12761												12555
ENDSET	1#	1416#													
ENDSFT	1#	1416#													
ENDSRV	1#	1416#	4945	4965	4993										
ENDSUB	1#	1416#	5049	5093	5152	5247	5403	5555	5709	5731	5776	6270	6353	6510	6584
ENDSW	1#	1416#													
ENDTST	1#	1416#	5099	5253	5301	5354	5558	5605	5659	5779	5864	6356	6595	6788	6934
		7054	7245	7381	7560	7713	7872	8004	8141	8279	8442	8603	8762	8986	9178
		9492	9687	9870	10031	10292	10481	10627	10806	10959	11140	11321	11523	11743	11940
		12564	12764	13013											12313
EQUALS	1#	1416#	1602												
ERRDF	1#	1416#	5033	5136	5282	5332									
ERRHRD	1#	1416#	5067	5084	5170	5187	5204	5221	5238	5394	5423	5447	5478	5502	5527
		5546	5594	5634	5650	5703	5750	5767	5804	5821	5845	5858	6241	6261	6307
		6327	6347	6438	6458	6481	6501	6542	6559	6576	6668	6688	6715	6735	6773
		6842	6850	6870	6891	6898	6918	6925	6989	7017	7033	7048	7119	7129	7193
		7205	7232	7300	7310	7332	7344	7365	7372	7446	7456	7479	7500	7509	7532
		7551	7611	7621	7649	7664	7673	7690	7703	7770	7780	7803	7817	7826	7845
		7920	7928	7950	7964	7973	7990	8058	8068	8089	8103	8110	8119	8132	8196
		8227	8241	8248	8257	8270	8344	8354	8377	8396	8405	8423	8433	8506	8516
		8557	8566	8584	8594	8659	8669	8700	8714	8723	8741	8751	8836	8846	8867
		8894	8915	8928	8937	8949	8958	8967	8977	9063	9073	9103	9117	9126	9149
		9232	9242	9272	9286	9295	9319	9333	9405	9415	9436	9451	9460	9473	9483
		9580	9613	9627	9636	9656	9670	9759	9769	9799	9813	9822	9837	9851	9922
		9962	9976	9985	9999	10013	10161	10171	10213	10227	10236	10253	10269	10283	10384
		10423	10446	10454	10465	10539	10549	10569	10585	10594	10608	10618	10699	10709	10735
		10759	10772	10788	10866	10876	10895	10917	10926	10940	10950	11038	11048	11074	11093
		11115	11125	11206	11216	11244	11263	11272	11288	11296	11306	11395	11405	11447	11461
		11484	11494	11603	11613	11646	11660	11669	11686	11701	11712	11726	11818	11828	11857
		11880	11894	11909	11920	11934	12054	12064	12098	12112	12121	12143	12154	12175	12189
		12215	12226	12247	12261	12270	12294	12304	12397	12407	12431	12445	12454	12468	12478
		12512	12521	12536	12549	12644	12654	12683	12697	12706	12719	12750	12811	12824	12845
		12889	12902	12934	12946										12858
ERROR	1#	1416#													
ERRSF	1#	1416#													
ERRSOF	1#	1416#													
ERRTBL	1#	1416#													
ESCAPE	1#	1416#	5399	5755	5809	5850	6246	6312	6443	6486	6547	6673	6693	6720	6740

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 316  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

	6875	7022	7108	7124	7134	7170	7290	7305	7315	7337	7349	7436	7451	7461	7484
	7505	7514	7601	7616	7626	7654	7669	7678	7708	7760	7775	7785	7808	7822	7831
	7864	7955	7969	8048	8063	8073	8094	8115	8124	8186	8201	8211	8232	8253	8262
	8334	8349	8359	8382	8401	8410	8496	8511	8521	8543	8562	8571	8649	8664	8674
	8705	8719	8728	8826	8841	8851	8872	8890	8899	9053	9068	9078	9108	9122	9131
	9168	9222	9237	9247	9277	9291	9300	9338	9395	9410	9420	9441	9456	9465	9560
	9575	9585	9618	9632	9641	9675	9749	9764	9774	9804	9818	9827	9856	9912	9927
	9937	9967	9981	9990	10018	10151	10166	10176	10218	10232	10241	10374	10389	10399	10428
	10451	10529	10544	10554	10574	10590	10599	10689	10704	10714	10740	10755	10764	10793	10856
	10871	10881	10900	10922	10931	11028	11043	11053	11079	11098	11107	11196	11211	11221	11249
	11268	11277	11385	11400	11410	11452	11466	11475	11593	11608	11618	11651	11665	11674	11731
	11808	11823	11833	11862	11876	11885	12044	12059	12069	12103	12117	12126	12180	12200	12252
	12266	12275	12387	12402	12412	12436	12450	12459	12503	12517	12526	12541	12634	12649	12659
	12688	12702	12711	12724	12755	12795	12816	12829	12850	12863	12894	12907	12939	12951	
EXIT	1#	1416#	4717	4890	4910	4921	12789								
FEQUAL	1#	1416#													
GETBYT	1#	1416#													
GETPRI	1#	1416#													
GETWOR	1#	1416#													
GMANIA	1#	1416#													
GMANID	1#	1416#													
GMANIL	1#	1416#													
GPHARD	1#	1416#	4821												
GPRMA	1#	1416#	13141	13146											
GPRMD	1#	1416#													
GPRML	1#	1416#													
HEADER	1#	1416#	1428												
INLOOP	1#	1416#													
IOSETU	1#	1416#													
IOSTAR	1#	1416#													
KT11	1#	1416#													
LASTAD	1#	1416#													
MANUAL	1#	1416#													
MEMORY	1#	1416#	4784												
MSBYTE	1#	1416#	1429#	1435	1436	1437									
MSCHEC	1#	1416#	4718#	4891#	4911#	4922#	12790#								
MSCNTO	1#	1416#	13142#	13147#											
MSCOUN	1#	1416#	2917#	2930#	2946#	2953#	2967#	2980#	2993#	3010#	3016#	3029#	3046#	3059#	3065#
	3080#	3086#	3093#	3106#	3119#	3131#	3144#	3156#	3162#	3168#	3181#	3194#	3201#	3214#	3228#
	3243#	3250#	3257#	3271#	3278#	3285#	3299#	3313#	3320#	3328#	3342#	3348#	3355#	3369#	3375#
	3382#	3397#	3405#	3415#	3431#	3443#	3449#	3458#	3465#	3483#	3490#	3496#	3502#	3509#	3516#
	3529#	3535#	3548#	3554#	3568#	3574#	3588#	3596#	3603#	3616#	3622#	3635#	3641#	3648#	3655#
	3662#	3676#	3682#	3690#	3703#	3716#	3723#	3736#	3743#	3749#	3763#	3775#	3787#	3799#	3811#
	3817#	3824#	3838#	3844#	3858#	3865#	3872#	3879#	3886#	3900#	3914#	3922#	3930#	3938#	3951#
	3957#	3963#	3975#	4386#	4422#	4539#	4554#	4575#	4641#	4656#	4669#	4684#	4697#	4796#	4882#
	12977#	12984#	12991#	12998#	13005#										
MSDATA	1#	1416#	1429#	1438	1440	1442	1444	1446	1448	1450	1452	1454	1456	1458	1460
	1462	1464	1466	1468#	1470	1472	1475	1478	1480	1482	1484	1486	1488	1490	1492
	1494	1496	1498	1500	1502	1504	1506	1508	1510	1512	1962#	1970#			
MSDECR	1#	1416#	1590#	2924#	2940#	2961#	2974#	2987#	3004#	3023#	3040#	3053#	3074#	3100#	3113#
	3125#	3138#	3150#	3175#	3188#	3208#	3222#	3237#	3265#	3293#	3307#	3336#	3363#	3390#	3425#
	3437#	3475#	3523#	3542#	3562#	3582#	3610#	3629#	3670#	3697#	3710#	3730#	3757#	3769#	3781#
	3793#	3805#	3832#	3852#	3894#	3906#	3945#	3969#	3981#	4721#	4740#	4845#	4862#	4904#	4914#
	4925#	4946#	4966#	4994#	5050#	5074#	5091#	5094#	5100#	5153#	5177#	5194#	5211#	5228#	5245#
	5248#	5254#	5302#	5349#	5355#	5404#	5430#	5454#	5485#	5509#	5534#	5553#	5556#	5559#	5601#
	5606#	5641#	5657#	5660#	5710#	5732#	5774#	5777#	5780#	5865#	6251#	6268#	6271#	6317#	6334#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 317  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

	6354#	6357#	6448#	6465#	6491#	6508#	6511#	6552#	6566#	6585#	6596#	6678#	6698#	6725#	6745#
	6780#	6789#	6858#	6932#	6935#	6996#	7040#	7055#	7145#	7212#	7239#	7246#	7320#	7379#	7382#
	7466#	7558#	7561#	7631#	7696#	7714#	7790#	7852#	7873#	7936#	8002#	8005#	8078#	8139#	8142#
	8216#	8277#	8280#	8364#	8440#	8443#	8526#	8601#	8604#	8679#	8758#	8763#	8856#	8984#	8987#
	9083#	9156#	9179#	9252#	9326#	9349#	9425#	9490#	9493#	9590#	9663#	9688#	9779#	9844#	9871#
	9942#	10006#	10032#	10181#	10290#	10293#	10404#	10472#	10482#	10559#	10625#	10628#	10719#	10779#	10798#
	10807#	10886#	10957#	10960#	11058#	11133#	11141#	11226#	11313#	11322#	11415#	11501#	11514#	11524#	11623#
	11719#	11744#	11838#	11927#	11941#	12074#	12161#	12235#	12311#	12314#	12417#	12485#	12556#	12565#	12664#
	12762#	12765#	13014#	13152#	13179#										
MSDEFA	1#	1416#	13142#	13147#											
MSENDE	1#	1416#	1590#	2924#	2940#	2961#	2974#	2987#	3004#	3023#	3040#	3053#	3074#	3100#	3113#
	3125#	3138#	3150#	3175#	3188#	3208#	3222#	3237#	3265#	3293#	3307#	3336#	3363#	3390#	3425#
	3437#	3475#	3523#	3542#	3562#	3582#	3610#	3629#	3670#	3697#	3710#	3730#	3757#	3769#	3781#
	3793#	3805#	3832#	3852#	3894#	3906#	3945#	3969#	3981#	4721#	4845#	4862#	4904#	4914#	4925#
	4946#	4966#	4994#	5050#	5074#	5091#	5094#	5100#	5153#	5177#	5194#	5211#	5228#	5245#	5248#
	5254#	5302#	5349#	5355#	5404#	5430#	5454#	5485#	5509#	5534#	5553#	5556#	5559#	5601#	5606#
	5641#	5657#	5660#	5710#	5732#	5774#	5777#	5780#	5865#	6251#	6268#	6271#	6317#	6334#	6354#
	6357#	6448#	6465#	6491#	6508#	6511#	6552#	6566#	6585#	6596#	6678#	6698#	6725#	6745#	6780#
	6789#	6858#	6932#	6935#	6996#	7040#	7055#	7145#	7212#	7239#	7246#	7320#	7379#	7382#	7466#
	7558#	7561#	7631#	7696#	7714#	7790#	7852#	7873#	7936#	8002#	8005#	8078#	8139#	8142#	8216#
	8277#	8280#	8364#	8440#	8443#	8526#	8601#	8604#	8679#	8758#	8763#	8856#	8984#	8987#	9083#
	9156#	9179#	9252#	9326#	9349#	9425#	9490#	9493#	9590#	9663#	9688#	9779#	9844#	9871#	9942#
	10006#	10032#	10181#	10290#	10293#	10404#	10472#	10482#	10559#	10625#	10628#	10719#	10779#	10798#	10807#
	10886#	10957#	10960#	11058#	11133#	11141#	11226#	11313#	11322#	11415#	11501#	11514#	11524#	11623#	11719#
	11744#	11838#	11927#	11941#	12074#	12161#	12235#	12311#	12314#	12417#	12485#	12556#	12565#	12664#	12762#
	12765#	13014#	13152#	13179#											
MSERRI	1#	1416#	5034#	5068#	5085#	5137#	5171#	5188#	5205#	5222#	5239#	5283#	5333#	5395#	5424#
	5448#	5479#	5503#	5528#	5547#	5595#	5635#	5651#	5704#	5726#	5751#	5768#	5805#	5822#	5846#
	5859#	6242#	6262#	6308#	6328#	6348#	6439#	6459#	6482#	6502#	6543#	6560#	6577#	6669#	6689#
	6716#	6736#	6764#	6774#	6843#	6851#	6871#	6892#	6899#	6919#	6926#	6990#	7018#	7034#	7049#
	7120#	7130#	7166#	7194#	7206#	7233#	7301#	7311#	7333#	7345#	7366#	7373#	7447#	7457#	7480#
	7501#	7510#	7533#	7540#	7552#	7612#	7622#	7650#	7665#	7674#	7691#	7704#	7771#	7781#	7804#
	7818#	7827#	7846#	7860#	7921#	7929#	7951#	7965#	7974#	7991#	8059#	8069#	8090#	8104#	8111#
	8120#	8133#	8197#	8207#	8228#	8242#	8249#	8258#	8271#	8345#	8355#	8378#	8397#	8406#	8424#
	8434#	8507#	8517#	8539#	8558#	8567#	8585#	8595#	8660#	8670#	8701#	8715#	8724#	8742#	8752#
	8837#	8847#	8868#	8886#	8895#	8916#	8929#	8938#	8950#	8959#	8968#	8978#	9064#	9074#	9104#
	9118#	9127#	9150#	9164#	9233#	9243#	9273#	9287#	9296#	9320#	9334#	9406#	9416#	9437#	9452#
	9441#	9474#	9484#	9571#	9581#	9614#	9628#	9637#	9657#	9671#	9760#	9770#	9800#	9814#	9823#
	9852#	9923#	9933#	9963#	9977#	9986#	10000#	10014#	10162#	10172#	10172#	10214#	10228#	10237#	10254#
	10270#	10284#	10385#	10395#	10424#	10447#	10455#	10466#	10540#	10550#	10570#	10586#	10595#	10609#	10619#
	10700#	10710#	10736#	10751#	10760#	10773#	10789#	10867#	10877#	10896#	10918#	10927#	10941#	10951#	11039#
	11049#	11075#	11094#	11103#	11116#	11126#	11207#	11217#	11245#	11264#	11273#	11289#	11297#	11307#	11396#
	11406#	11448#	11462#	11471#	11485#	11495#	11604#	11614#	11647#	11661#	11670#	11687#	11702#	11713#	11727#
	11819#	11829#	11858#	11872#	11881#	11895#	11910#	11921#	11935#	12055#	12065#	12099#	12113#	12122#	12144#
	12155#	12176#	12190#	12196#	12216#	12227#	12248#	12262#	12271#	12295#	12305#	12398#	12408#	12432#	12446#
	12455#	12469#	12479#	12499#	12513#	12522#	12537#	12550#	12645#	12655#	12684#	12698#	12707#	12720#	12751#
	12812#	12825#	12846#	12859#	12890#	12903#	12935#	12947#							
MSERCA	1#	1416#	5400#	5401	5756#	5757	5810#	5811	5851#	5852	6247#	6248	6313#	6314	6444#
	6445	6487#	6488	6548#	6549	6674#	6675	6694#	6695	6721#	6722	6741#	6742	6876#	6877
	7023#	7024	7109#	7110	7125#	7126	7135#	7136	7171#	7172	7291#	7292	7306#	7307	7316#
	7317	7338#	7339	7350#	7351	7437#	7438	7452#	7453	7462#	7463	7485#	7486	7506#	7507
	7515#	7516	7602#	7603	7617#	7618	7627#	7628	7655#	7656	7670#	7671	7679#	7680	7709#
	7710	7761#	7762	7776#	7777	7786#	7787	7809#	7810	7823#	7824	7832#	7833	7865#	7866
	7956#	7957	7970#	7971	8049#	8050	8064#	8065	8074#	8075	8095#	8096	8116#	8117	8125#
	8126	8187#	8188	8202#	8203	8212#	8213	8233#	8234	8254#	8255	8263#	8264	8335#	8336
	8350#	8351	8360#	8361	8383#	8384	8402#	8403	8411#	8412	8497#	8498	8512#	8513	8522#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 318  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

	8523	8544#	8545	8563#	8564	8572#	8573	8650#	8651	8665#	8666	8675#	8676	8706#	8707
	8720#	8721	8729#	8730	8827#	8828	8842#	8843	8852#	8853	8873#	8874	8891#	8892	8900#
	8901	9054#	9055	9069#	9070	9079#	9080	9109#	9110	9123#	9124	9132#	9133	9169#	9170
	9223#	9224	9238#	9239	9248#	9249	9278#	9279	9292#	9293	9301#	9302	9339#	9340	9396#
	9397	9411#	9412	9421#	9422	9442#	9443	9457#	9458	9466#	9467	9561#	9562	9576#	9577
	9586#	9587	9619#	9620	9633#	9634	9642#	9643	9676#	9677	9750#	9751	9765#	9766	9775#
	9776	9805#	9806	9819#	9820	9828#	9829	9857#	9858	9913#	9914	9928#	9929	9938#	9939
	9968#	9969	9982#	9983	9991#	9992	10019#	10020	10152#	10153	10167#	10168	10177#	10178	10219#
	10220	10233#	10234	10242#	10243	10375#	10376	10390#	10391	10400#	10401	10429#	10430	10452#	10453
	10530#	10531	10545#	10546	10555#	10556	10575#	10576	10591#	10592	10600#	10601	10690#	10691	10705#
	10706	10715#	10716	10741#	10742	10756#	10757	10765#	10766	10794#	10795	10857#	10858	10872#	10873
	10882#	10883	10901#	10902	10923#	10924	10932#	10933	11029#	11030	11044#	11045	11054#	11055	11080#
	110J1	11099#	11100	11108#	11109	11197#	11198	11212#	11213	11222#	11223	11250#	11251	11269#	11270
	11278#	11279	11386#	11387	11401#	11402	11411#	11412	11453#	11454	11467#	11468	11476#	11477	11594#
	11595	11609#	11610	11619#	11620	11652#	11653	11666#	11667	11675#	11676	11732#	11733	11809#	11810
	11824#	11825	11834#	11835	11863#	11864	11877#	11878	11886#	11887	12045#	12046	12060#	12061	12070#
	12071	12104#	12105	12118#	12119	12127#	12128	12181#	12182	12201#	12202	12253#	12254	12267#	12268
	12276#	12277	12388#	12389	12403#	12404	12413#	12414	12437#	12438	12451#	12452	12460#	12461	12504#
	12505	12518#	12519	12527#	12528	12542#	12543	12635#	12636	12650#	12651	12660#	12661	12689#	12690
	12703#	12704	12712#	12713	12725#	12726	12756#	12757	12796#	12797	12817#	12818	12830#	12831	12851#
	12852	12864#	12865	12895#	12896	12908#	12909	12940#	12941	12952#	12953				
MSECCS	1#	1416#	5400#	5756#	5810#	5851#	6247#	6313#	6444#	6487#	6548#	6674#	6694#	6721#	6741#
	6876#	7023#	7109#	7125#	7135#	7171#	7291#	7306#	7316#	7338#	7350#	7437#	7452#	7462#	7485#
	7506#	7515#	7602#	7617#	7627#	7655#	7670#	7679#	7709#	7761#	7776#	7786#	7809#	7823#	7832#
	7865#	7956#	7970#	8049#	8064#	8074#	8095#	8116#	8125#	8187#	8202#	8212#	8233#	8254#	8263#
	8335#	8350#	8360#	8383#	8402#	8411#	8497#	8512#	8522#	8544#	8563#	8572#	8650#	8665#	8675#
	8706#	8720#	8729#	8827#	8842#	8852#	8873#	8891#	8900#	9054#	9069#	9079#	9109#	9123#	9132#
	9169#	9223#	9238#	9248#	9278#	9292#	9301#	9339#	9396#	9411#	9421#	9442#	9457#	9466#	9561#
	9576#	9586#	9619#	9633#	9642#	9676#	9750#	9765#	9775#	9805#	9819#	9828#	9857#	9913#	9928#
	9938#	9968#	9982#	9991#	10019#	10152#	10167#	10177#	10219#	10233#	10242#	10375#	10390#	10400#	10429#
	10452#	10530#	10545#	10555#	10575#	10591#	10600#	10690#	10705#	10715#	10741#	10756#	10765#	10794#	10857#
	10872#	10882#	10901#	10923#	10932#	11029#	11044#	11054#	11080#	11099#	11108#	11197#	11212#	11222#	11250#
	11269#	11278#	11386#	11401#	11411#	11453#	11467#	11476#	11594#	11609#	11619#	11652#	11666#	11675#	11732#
	11809#	11824#	11834#	11863#	11877#	11886#	12045#	12060#	12070#	12104#	12118#	12127#	12181#	12201#	12253#
	12267#	12276#	12388#	12403#	12413#	12437#	12451#	12460#	12504#	12518#	12527#	12542#	12635#	12650#	12660#
	12689#	12703#	12712#	12725#	12756#	12796#	12817#	12830#	12851#	12864#	12895#	12908#	12940#	12952#	
MSEXCP	1#	1416#	13142#	13147#											
MSEXIT	1#	1416#	4718#	4891#	4892	4911#	4922#	12790#	12791						
MSEXSE	1#	1416#	4718#	4891#	4911#	4922#	12790#								
MSEXTJ	1#	1416#	4718#	4719	4891#	4911#	4912	4922#	4923	12790#					
MSEGEN	1#	1416#	1429#	1438#	1440#	1442#	1444#	1446#	1448#	1450#	1452#	1454#	1456#	1458#	1460#
	1462#	1464#	1466#	1468#	1470#	1472#	1475#	1478#	1480#	1482#	1484#	1486#	1488#	1490#	1492#
	1494#	1496#	1498#	1500#	1502#	1504#	1506#	1508#	1510#	1512#	1524#	1583#	1584#	1590#	1962#
	1970#	2915#	2924#	2928#	2940#	2944#	2961#	2965#	2974#	2978#	2987#	2991#	3004#	3008#	3023#
	3027#	3040#	3044#	3053#	3057#	3074#	3078#	3100#	3104#	3113#	3117#	3125#	3129#	3138#	3142#
	3150#	3154#	3175#	3179#	3188#	3192#	3208#	3212#	3222#	3226#	3237#	3241#	3265#	3269#	3293#
	3297#	3307#	3311#	3336#	3340#	3363#	3367#	3390#	3394#	3425#	3429#	3437#	3441#	3475#	3479#
	3523#	3527#	3542#	3546#	3562#	3566#	3582#	3586#	3610#	3614#	3629#	3633#	3670#	3674#	3697#
	3701#	3710#	3714#	3730#	3734#	3757#	3761#	3769#	3773#	3781#	3785#	3793#	3797#	3805#	3809#
	3832#	3836#	3852#	3856#	3894#	3898#	3906#	3910#	3945#	3949#	3969#	3973#	3981#	4716#	4721#
	4733#	4749#	4845#	4860#	4862#	4872#	4904#	4909#	4914#	4920#	4925#	4943#	4946#	4962#	4966#
	4988#	4994#	5014#	5023#	5050#	5053#	5074#	5091#	5094#	5100#	5117#	5126#	5153#	5156#	5177#
	5194#	5211#	5228#	5245#	5248#	5254#	5269#	5302#	5317#	5349#	5355#	5376#	5381#	5404#	5408#
	5430#	5454#	5485#	5509#	5534#	5553#	5556#	5559#	5581#	5601#	5606#	5625#	5641#	5657#	5660#
	5688#	5690#	5710#	5713#	5732#	5735#	5774#	5777#	5780#	5801#	5865#	6221#	6223#	6251#	6268#
	6271#	6274#	6317#	6334#	6354#	6357#	6397#	6400#	6448#	6465#	6491#	6508#	6511#	6514#	6552#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 319  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

6566#	6585#	6596#	6636#	6678#	6698#	6725#	6745#	6780#	6789#	6820#	6858#	6932#	6935#	6964#	
6996#	7040#	7055#	7093#	7145#	7212#	7239#	7246#	7275#	7320#	7379#	7382#	7421#	7466#	7558#	
7561#	7586#	7631#	7696#	7714#	7745#	7790#	7852#	7873#	7898#	7936#	8002#	8005#	8033#	8078#	
8139#	8142#	8171#	8216#	8277#	8280#	8319#	8364#	8440#	8443#	8481#	8526#	8601#	8604#	8534#	
8679#	8758#	8763#	8811#	8856#	8984#	8987#	9038#	9083#	9156#	9179#	9207#	9252#	9326#	9349#	
9380#	9425#	9490#	9493#	9545#	9590#	9663#	9688#	9734#	9779#	9844#	9871#	9897#	9942#	10006#	
10032#	10136#	10181#	10290#	10293#	10359#	10404#	10472#	10482#	10514#	10559#	10625#	10628#	10674#	10719#	
10779#	10798#	10807#	10841#	10886#	10957#	10960#	11013#	11058#	11133#	11141#	11181#	11226#	11313#	11322#	
11370#	11415#	11501#	11514#	11524#	11578#	11623#	11719#	11744#	11793#	11838#	11927#	11941#	12029#	12074#	
12161#	12235#	12311#	12314#	12372#	12417#	12485#	12556#	12565#	12619#	12664#	12762#	12765#	12786#	13014#	
13140#	13153#														
MSGEMB	1#														
MSGETS	1#	1416#	1590#	2924#	2940#	2961#	2974#	2987#	3004#	3023#	3040#	3053#	3074#	3100#	3113#
	3125#	3138#	3150#	3175#	3188#	3208#	3222#	3237#	3265#	3293#	3307#	3336#	3363#	3390#	3425#
	3437#	3475#	3523#	3542#	3562#	3582#	3610#	3629#	3670#	3697#	3710#	3730#	3757#	3769#	3781#
	3793#	3805#	3832#	3852#	3894#	3906#	3945#	3969#	3981#	4721#	4740#	4845#	4862#	4904#	4914#
	4925#	4946#	4966#	4994#	5050#	5074#	5091#	5094#	5100#	5153#	5177#	5194#	5211#	5228#	5245#
	5248#	5254#	5302#	5349#	5355#	5404#	5430#	5454#	5485#	5509#	5534#	5553#	5556#	5559#	5601#
	5606#	5641#	5657#	5660#	5710#	5732#	5774#	5777#	5780#	5865#	6251#	6268#	6271#	6317#	6334#
	6354#	6357#	6448#	6465#	6491#	6508#	6511#	6552#	6566#	6585#	6596#	6678#	6698#	6725#	6745#
	6780#	6789#	6858#	6932#	6935#	6996#	7040#	7055#	7145#	7212#	7239#	7246#	7320#	7379#	7382#
	7466#	7558#	7561#	7631#	7696#	7714#	7790#	7852#	7873#	7936#	8002#	8005#	8078#	8139#	8142#
	8216#	8277#	8280#	8364#	8440#	8443#	8526#	8601#	8604#	8679#	8758#	8763#	8856#	8984#	8987#
	9083#	9156#	9179#	9252#	9326#	9349#	9425#	9490#	9493#	9590#	9663#	9688#	9779#	9844#	9871#
	9942#	10006#	10032#	10181#	10290#	10293#	10404#	10472#	10482#	10559#	10625#	10628#	10719#	10779#	10798#
	10807#	10886#	10957#	10960#	11058#	11133#	11141#	11226#	11313#	11322#	11415#	11501#	11514#	11524#	11623#
	11719#	11744#	11838#	11927#	11941#	12074#	12161#	12235#	12311#	12314#	12417#	12485#	12556#	12565#	12664#
	12762#	12765#	13014#	13152#	13179#										
MSGETT	1#	1416#	4718#	4891#	4911#	4922#	5400#	5756#	5810#	5851#	6247#	6313#	6444#	6487#	6548#
	6674#	6694#	6721#	6741#	6876#	7023#	7109#	7125#	7135#	7171#	7291#	7306#	7316#	7338#	7350#
	7437#	7452#	7462#	7485#	7506#	7515#	7602#	7617#	7627#	7655#	7670#	7679#	7709#	7761#	7776#
	7786#	7809#	7823#	7832#	7865#	7956#	7970#	8049#	8064#	8074#	8095#	8116#	8125#	8187#	8202#
	8212#	8233#	8254#	8263#	8335#	8350#	8360#	8383#	8402#	8411#	8497#	8512#	8522#	8544#	8563#
	8572#	8650#	8665#	8675#	8706#	8720#	8729#	8827#	8842#	8852#	8873#	8891#	8900#	9054#	9069#
	9079#	9109#	9123#	9132#	9169#	9223#	9238#	9248#	9278#	9292#	9301#	9339#	9396#	9411#	9421#
	9442#	9457#	9466#	9561#	9576#	9586#	9619#	9633#	9642#	9676#	9750#	9765#	9775#	9805#	9819#
	9828#	9857#	9913#	9928#	9938#	9968#	9982#	9991#	10019#	10152#	10167#	10177#	10219#	10233#	10242#
	10375#	10390#	10400#	10429#	10452#	10530#	10545#	10555#	10575#	10591#	10600#	10690#	10705#	10715#	10741#
	10756#	10765#	10794#	10857#	10872#	10882#	10901#	10923#	10932#	11029#	11044#	11054#	11080#	11099#	11108#
	11197#	11212#	11222#	11250#	11269#	11278#	11386#	11401#	11411#	11453#	11467#	11476#	11594#	11609#	11619#
	11652#	11666#	11675#	11732#	11809#	11824#	11834#	11863#	11877#	11886#	12045#	12060#	12070#	12104#	12118#
	12127#	12181#	12201#	12253#	12267#	12276#	12388#	12403#	12413#	12437#	12451#	12460#	12504#	12518#	12527#
	12542#	12635#	12650#	12660#	12689#	12703#	12712#	12725#	12756#	12790#	12796#	12817#	12830#	12851#	12864#
	12895#	12908#	12940#	12952#											
MSGNGB	1#	1416#	1420#	1429#	1438#	1440#	1442#	1444#	1446#	1448#	1450#	1452#	1454#	1456#	1458#
	1460#	1462#	1464#	1466#	1468#	1470#	1472#	1475#	1478#	1480#	1482#	1484#	1486#	1488#	1490#
	1492#	1494#	1496#	1498#	1500#	1502#	1504#	1506#	1508#	1510#	1512#	1523#	1524	1582#	1583
	1584	1962#	1970#	2915#	2928#	2944#	2965#	2978#	2991#	3008#	3027#	3044#	3057#	3078#	3104#
	3117#	3129#	3142#	3154#	3179#	3192#	3212#	3226#	3241#	3269#	3297#	3311#	3340#	3367#	3394#
	3429#	3441#	3479#	3527#	3546#	3566#	3586#	3614#	3633#	3674#	3701#	3714#	3733#	3761#	3773#
	3785#	3797#	3809#	3836#	3856#	3898#	3910#	3949#	3973#	4716#	4733#	4749#	4860#	4872#	4909#
	4920#	4943#	4962#	4988#	13139#	13140									
MSGNIN	1#	1416#	1429#	1430	1431	1432	1433	1434	1435#	1436#	1437#	1438#	1439	1440#	1441
	1442#	1443	1444#	1445	1446#	1447	1448#	1449	1450#	1451	1452#	1453	1454#	1455	1456#
	1457	1458#	1459	1460#	1461	1462#	1463	1464#	1465	1466#	1467	1468#	1469	1470#	1471
	1472#	1473	1474	1475#	1476	1477#	1478#	1479	1480#	1481	1482#	1483	1484#	1485	1486#



67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 320  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

1487	1488#	1489	1490#	1491	1492#	1493	1494#	1495	1496#	1497	1498#	1499	1500#	1501
1502#	1503	1504#	1505	1506#	1507	1508#	1509	1510#	1511	1512#	1513	1523#	1525#	1526#
1527#	1528#	1529#	1530#	1531#	1532#	1533#	1534#	1535#	1536#	1537#	1538#	1539#	1540#	1541#
1542#	1543#	1544#	1545#	1546#	1547#	1548#	1549#	1550#	1551#	1552#	1553#	1554#	1555#	1556#
1557#	1558#	1559#	1560#	1561#	1562#	1563#	1564#	1565#	1566#	1567#	1568#	1569#	1570#	1582#
1962#	1963	1964	1970#	1971	1975	2917#	2918#	2919#	2920	2921#	2922	2925#	2930#	2931#
2932#	2933#	2934#	2935	2936#	2937	2941#	2946#	2947#	2948#	2949	2950#	2951	2953#	2954#
2955#	2956#	2957	2958#	2959	2952#	2967#	2968#	2969	2970#	2971	2975#	2980#	2981#	2982
2983#	2984	2988#	2993#	2994#	2995#	2996#	2997#	2998#	2999	3000#	3001	3005#	3010#	3011#
3012	3013#	3014	3016#	3017#	3018#	3019	3020#	3021	3024#	3029#	3030#	3031#	3032#	3033#
3034#	3035	3036#	3037	3041#	3046#	3047#	3048	3049#	3050	3054#	3059#	3060#	3061	3062#
3063	3065#	3066#	3067#	3068#	3069#	3070	3071#	3072	3075#	3080#	3081#	3082	3083#	3084
3086#	3087#	3088#	3089	3090#	3091	3093#	3094#	3095#	3096	3097#	3098	3101#	3106#	3107#
3108	3109#	3110	3114#	3119#	3120#	3121	3122#	3123	3126#	3131#	3132#	3133#	3134	3135#
3136	3139#	3144#	3145#	3146	3147#	3148	3151#	3156#	3157#	3158	3159#	3160	3162#	3163#
3164	3165#	3166	3168#	3169#	3170#	3171	3172#	3173	3176#	3181#	3182#	3183#	3184	3185#
3186	3189#	3194#	3195#	3196#	3197	3198#	3199	3201#	3202#	3203#	3204	3205#	3206	3209#
3214#	3215#	3216#	3217#	3218	3219#	3220	3223#	3228#	3229#	3230#	3231#	3232#	3233	3234#
3235	3238#	3243#	3244#	3245#	3246	3247#	3248	3250#	3251#	3252#	3253	3254#	3255	3257#
3258#	3259#	3260#	3261	3262#	3263	3266#	3271#	3272#	3273#	3274	3275#	3276	3278#	3279#
3280#	3281	3282#	3283	3285#	3286#	3287#	3288#	3289	3290#	3291	3294#	3299#	3300#	3301#
3302#	3303	3304#	3305	3308#	3313#	3314#	3315#	3316	3317#	3318	3320#	3321#	3322#	3323#
3324	3325#	3326	3328#	3329#	3330#	3331#	3332	3333#	3334	3337#	3342#	3343#	3344	3345#
3346	3348#	3349#	3350#	3351	3352#	3353	3355#	3356#	3357#	3358#	3359	3360#	3361	3364#
3369#	3370#	3371	3372#	3373	3375#	3376#	3377#	3378	3379#	3380	3382#	3383#	3384#	3385#
3386	3387#	3388	3391#	3397#	3398#	3399#	3400	3401#	3402	3405#	3406#	3407#	3408#	3409#
3410	3411#	3412	3415#	3416#	3417#	3418#	3419#	3420	3421#	3422	3426#	3431#	3432#	3433
3434#	3435	3438#	3443#	3444#	3445	3446#	3447	3449#	3450#	3451	3452#	3453	3458#	3459#
3460#	3461	3462#	3463	3465#	3466#	3467#	3468	3469#	3470	3476#	3483#	3484#	3485	3486#
3487	3490#	3491#	3492	3493#	3494	3496#	3497#	3498	3499#	3500	3502#	3503#	3504#	3505
3506#	3507	3509#	3510#	3511#	3512	3513#	3514	3516#	3517#	3518#	3519	3520#	3521	3524#
3529#	3530#	3531	3532#	3533	3535#	3536#	3537#	3538	3539#	3540	3543#	3548#	3549#	3550
3551#	3552	3554#	3555#	3556#	3557#	3558	3559#	3560	3563#	3568#	3569#	3570	3571#	3572
3574#	3575#	3576#	3577#	3578	3579#	3580	3583#	3588#	3589#	3590#	3591#	3592	3593#	3594
3596#	3597#	3598#	3599	3600#	3601	3603#	3604#	3605#	3606	3607#	3608	3611#	3616#	3617#
3618	3619#	3620	3622#	3623#	3624#	3625	3626#	3627	3630#	3635#	3636#	3637	3638#	3639
3641#	3642#	3643#	3644	3645#	3646	3648#	3649#	3650#	3651	3652#	3653	3655#	3656#	3657#
3658	3659#	3660	3662#	3663#	3664#	3665#	3666	3667#	3668	3671#	3676#	3677#	3678	3679#
3680	3682#	3683#	3684#	3685#	3686	3687#	3688	3690#	3691#	3692#	3693	3694#	3695	3698#
3703#	3704#	3705#	3706	3707#	3708	3711#	3716#	3717#	3718#	3719	3720#	3721	3723#	3724#
3725#	3726	3727#	3728	3731#	3736#	3737#	3738#	3739	3740#	3741	3743#	3744#	3745	3746#
3747	3749#	3750#	3751#	3752#	3753	3754#	3755	3758#	3763#	3764#	3765	3766#	3767	3770#
3775#	3776#	3777	3778#	3779	3782#	3787#	3788#	3789	3790#	3791	3794#	3799#	3800#	3801
3802#	3803	3806#	3811#	3812#	3813	3814#	3815	3817#	3818#	3819#	3820	3821#	3822	3824#
3825#	3826#	3827#	3828	3829#	3830	3833#	3838#	3839#	3840	3841#	3842	3844#	3845#	3846#
3847#	3848	3849#	3850	3853#	3858#	3859#	3860#	3861	3862#	3863	3865#	3866#	3867#	3868
3869#	3870	3872#	3873#	3874#	3875	3876#	3877	3879#	3880#	3881#	3882	3883#	3884	3886#
3887#	3888#	3889#	3890	3891#	3892	3895#	3900#	3901#	3902	3903#	3904	3907#	3914#	3915#
3916	3917#	3918	3922#	3923#	3924	3925#	3926	3930#	3931#	3932	3933#	3934	3938#	3939#
3940	3941#	3942	3946#	3951#	3952#	3953	3954#	3955	3957#	3958#	3959	3960#	3961	3963#
3964#	3965	3966#	3967	3970#	3975#	3976#	3977	3978#	3979	3982#	4233#	4234#	4247#	4248#
4277#	4386#	4387#	4388#	4389	4390#	4391	4422#	4423#	4424#	4425#	4426	4427#	4428	4462#
4539#	4540#	4541#	4542#	4543#	4544#	4545	4546#	4547	4554#	4555#	4556	4557#	4558	4575#
4576#	4577#	4578	4579#	4580	4641#	4642#	4643#	4644	4645#	4646	4656#	4657#	4658#	4659
4660#	4661	4669#	4670#	4671#	4672	4673#	4674	4684#	4685#	4686#	4687	4688#	4689	4697#
4698#	4699#	4700	4701#	4702	4718#	4719#	4722#	4754#	4755#	4757#	4759#	4760#	4763#	4774#

4775#	4777#	4779#	4780#	4782#	4785#	4786#	4790#	4791#	4792#	4794#	4796#	4797#	4798	4799#
4800	4809#	4810#	4811#	4812#	4813#	4814	4822#	4823#	4824#	4826#	4842#	4846#	4863#	4882#
4883#	4884	4885#	4886	4891#	4892#	4905#	4911#	4912#	4915#	4922#	4923#	4926#	4946#	4947
4966#	4967	4994#	4995	5016#	5017#	5018#	5019#	5020#	5021	5024#	5034#	5035#	5036#	5037#
5039#	5041#	5042#	5044#	5045#	5047#	5051#	5054#	5059#	5068#	5069#	5070#	5071#	5075#	5077#
5085#	5086#	5087#	5088#	5092#	5095#	5097#	5098#	5101#	5119#	5120#	5121#	5122#	5123#	5124
5127#	5137#	5138#	5139#	5140#	5142#	5144#	5145#	5147#	5148#	5150#	5154#	5157#	5162#	5171#
5172#	5173#	5174#	5178#	5180#	5188#	5189#	5190#	5191#	5195#	5197#	5205#	5206#	5207#	5208#
5212#	5214#	5222#	5223#	5224#	5225#	5229#	5231#	5239#	5240#	5241#	5242#	5246#	5249#	5251#
5252#	5255#	5271#	5272#	5273#	5274#	5275#	5276	5283#	5284#	5285#	5286#	5288#	5290#	5291#
5293#	5294#	5296#	5298#	5299#	5303#	5319#	5320#	5321#	5322#	5323#	5324	5326#	5333#	5334#
5335#	5336#	5338#	5340#	5341#	5343#	5344#	5346#	5350#	5352#	5353#	5356#	5382#	5395#	5396#
5397#	5398#	5400#	5401#	5405#	5409#	5414#	5424#	5425#	5426#	5427#	5431#	5442#	5448#	5449#
5450#	5451#	5455#	5468#	5479#	5480#	5481#	5482#	5486#	5497#	5503#	5504#	5505#	5506#	5510#
5518#	5528#	5529#	5530#	5531#	5535#	5537#	5547#	5548#	5549#	5550#	5554#	5557#	5560#	5587#
5595#	5596#	5597#	5598#	5602#	5607#	5629#	5635#	5636#	5637#	5638#	5642#	5645#	5651#	5652#
5653#	5654#	5658#	5661#	5691#	5704#	5705#	5706#	5707#	5711#	5714#	5726#	5727#	5728#	5729#
5733#	5736#	5751#	5752#	5753#	5754#	5756#	5757#	5760#	5768#	5769#	5770#	5771#	5775#	5778#
5781#	5805#	5806#	5807#	5808#	5810#	5811#	5822#	5823#	5824#	5825#	5846#	5847#	5848#	5849#
5851#	5852#	5859#	5860#	5861#	5862#	5866#	6224#	6229#	6242#	6243#	6244#	6245#	6247#	6248#
6252#	6254#	6262#	6263#	6264#	6265#	6269#	6272#	6275#	6288#	6308#	6309#	6310#	6311#	6313#
6314#	6318#	6320#	6328#	6329#	6330#	6331#	6335#	6348#	6349#	6350#	6351#	6355#	6358#	6401#
6406#	6439#	6440#	6441#	6442#	6444#	6445#	6449#	6451#	6459#	6460#	6461#	6462#	6466#	6468#
6482#	6483#	6484#	6485#	6487#	6488#	6492#	6494#	6502#	6503#	6504#	6505#	6509#	6512#	6515#
6520#	6543#	6544#	6545#	6546#	6548#	6549#	6553#	6555#	6560#	6561#	6562#	6563#	6567#	6577#
6578#	6579#	6580#	6586#	6597#	6641#	6669#	6670#	6671#	6672#	6674#	6675#	6679#	6681#	6689#
6690#	6691#	6692#	6694#	6695#	6699#	6701#	6716#	6717#	6718#	6719#	6721#	6722#	6726#	6728#
6736#	6737#	6738#	6739#	6741#	6742#	6746#	6751#	6764#	6765#	6766#	6767#	6774#	6775#	6776#
6777#	6781#	6790#	6827#	6843#	6844#	6845#	6846#	6851#	6852#	6853#	6854#	6859#	6867#	6871#
6872#	6873#	6874#	6876#	6877#	6892#	6893#	6894#	6895#	6899#	6900#	6901#	6902#	6919#	6920#
6921#	6922#	6926#	6927#	6928#	6929#	6933#	6936#	6970#	6972#	6973#	6974#	6975#	6976#	6977
6990#	6991#	6992#	6993#	6997#	7002#	7008#	7009#	7018#	7019#	7020#	7021#	7023#	7024#	7034#
7035#	7036#	7037#	7041#	7043#	7044#	7049#	7050#	7051#	7052#	7056#	7100#	7109#	7110#	7120#
7121#	7122#	7123#	7125#	7126#	7130#	7131#	7132#	7133#	7135#	7136#	7138#	7139#	7140#	7141#
7142#	7143	7146#	7162#	7166#	7167#	7168#	7169#	7171#	7172#	7181#	7182#	7194#	7195#	7196#
7197#	7206#	7207#	7208#	7209#	7213#	7218#	7233#	7234#	7235#	7236#	7240#	7247#	7282#	7291#
7292#	7301#	7302#	7303#	7304#	7306#	7307#	7311#	7312#	7313#	7314#	7316#	7317#	7321#	7329#
7333#	7334#	7335#	7336#	7338#	7339#	7345#	7346#	7347#	7348#	7350#	7351#	7366#	7367#	7368#
7369#	7373#	7374#	7375#	7376#	7380#	7383#	7428#	7437#	7438#	7447#	7448#	7449#	7450#	7452#
7453#	7457#	7458#	7459#	7460#	7462#	7463#	7467#	7476#	7480#	7481#	7482#	7483#	7485#	7486#
7501#	7502#	7503#	7504#	7506#	7507#	7510#	7511#	7512#	7513#	7515#	7516#	7533#	7534#	7535#
7536#	7540#	7541#	7542#	7543#	7552#	7553#	7554#	7555#	7559#	7562#	7593#	7602#	7603#	7612#
7613#	7614#	7615#	7617#	7618#	7622#	7623#	7624#	7625#	7627#	7628#	7632#	7646#	7650#	7651#
7652#	7653#	7655#	7656#	7665#	7666#	7667#	7668#	7670#	7671#	7674#	7675#	7676#	7677#	7679#
7680#	7691#	7692#	7693#	7694#	7697#	7704#	7705#	7706#	7707#	7709#	7710#	7715#	7752#	7761#
7762#	7771#	7772#	7773#	7774#	7776#	7777#	7781#	7782#	7783#	7784#	7786#	7787#	7791#	7800#
7804#	7805#	7806#	7807#	7809#	7810#	7818#	7819#	7820#	7821#	7823#	7824#	7827#	7828#	7829#
7830#	7832#	7833#	7846#	7847#	7848#	7849#	7853#	7860#	7861#	7862#	7863#	7865#	7866#	7874#
7905#	7921#	7922#	7923#	7924#	7929#	7930#	7931#	7932#	7937#	7946#	7951#	7952#	7953#	7954#
7956#	7957#	7965#	7966#	7967#	7968#	7970#	7971#	7974#	7975#	7976#	7977#	7991#	7992#	7993#
7994#	8003#	8006#	8040#	8049#	8050#	8059#	8060#	8061#	8062#	8064#	8065#	8069#	8070#	8071#
8072#	8074#	8075#	8079#	8086#	8090#	8091#	8092#	8093#	8095#	8096#	8104#	8105#	8106#	8107#
8111#	8112#	8113#	8114#	8116#	8117#	8120#	8121#	8122#	8123#	8125#	8126#	8133#	8134#	8135#
8136#	8140#	8143#	8178#	8187#	8188#	8197#	8198#	8199#	8200#	8202#	8203#	8207#	8208#	8209#
8210#	8212#	8213#	8217#	8224#	8228#	8229#	8230#	8231#	8233#	8234#	8242#	8243#	8244#	8245#
8249#	8250#	8251#	8252#	8254#	8255#	8258#	8259#	8260#	8261#	8263#	8264#	8271#	8272#	8273#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 322  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

8274#	8278#	8281#	8326#	8335#	8336#	8345#	8346#	8347#	8348#	8350#	8351#	8355#	8356#	8357#
8358#	8360#	8361#	8365#	8374#	8378#	8379#	8380#	8381#	8383#	8384#	8397#	8398#	8399#	8400#
8402#	8403#	8406#	8407#	8408#	8409#	8411#	8412#	8424#	8425#	8426#	8427#	8434#	8435#	8436#
8437#	8441#	8444#	8488#	8497#	8498#	8507#	8508#	8509#	8510#	8512#	8513#	8517#	8518#	8519#
8520#	8522#	8523#	8527#	8535#	8539#	8540#	8541#	8542#	8544#	8545#	8558#	8559#	8560#	8561#
8563#	8564#	8567#	8568#	8569#	8570#	8572#	8573#	8585#	8586#	8587#	8588#	8595#	8596#	8597#
8598#	8602#	8605#	8641#	8650#	8651#	8660#	8661#	8662#	8663#	8665#	8666#	8670#	8671#	8672#
8673#	8675#	8676#	8680#	8693#	8701#	8702#	8703#	8704#	8706#	8707#	8715#	8716#	8717#	8718#
8720#	8721#	8724#	8725#	8726#	8727#	8729#	8730#	8742#	8743#	8744#	8745#	8752#	8753#	8754#
8755#	8759#	8764#	8818#	8827#	8828#	8837#	8838#	8839#	8840#	8842#	8843#	8847#	8848#	8849#
8850#	8852#	8853#	8857#	8864#	8868#	8869#	8870#	8871#	8873#	8874#	8886#	8887#	8888#	8889#
8891#	8892#	8895#	8896#	8897#	8898#	8900#	8901#	8916#	8917#	8918#	8919#	8929#	8930#	8931#
8932#	8938#	8939#	8940#	8941#	8950#	8951#	8952#	8953#	8959#	8960#	8961#	8962#	8968#	8969#
8970#	8971#	8978#	8979#	8980#	8981#	8985#	8988#	9045#	9054#	9055#	9064#	9065#	9066#	9067#
9069#	9070#	9074#	9075#	9076#	9077#	9079#	9080#	9084#	9096#	9104#	9105#	9106#	9107#	9109#
9110#	9118#	9119#	9120#	9121#	9123#	9124#	9127#	9128#	9129#	9130#	9132#	9133#	9150#	9151#
9152#	9153#	9157#	9164#	9165#	9166#	9167#	9169#	9170#	9180#	9214#	9223#	9224#	9233#	9234#
9235#	9236#	9238#	9239#	9243#	9244#	9245#	9246#	9248#	9249#	9253#	9265#	9273#	9274#	9275#
9276#	9278#	9279#	9287#	9288#	9289#	9290#	9292#	9293#	9296#	9297#	9298#	9299#	9301#	9302#
9320#	9321#	9322#	9323#	9327#	9334#	9335#	9336#	9337#	9339#	9340#	9350#	9387#	9396#	9397#
9406#	9407#	9408#	9409#	9411#	9412#	9416#	9417#	9418#	9419#	9421#	9422#	9426#	9433#	9437#
9438#	9439#	9440#	9442#	9443#	9452#	9453#	9454#	9455#	9457#	9458#	9461#	9462#	9463#	9464#
9466#	9467#	9474#	9475#	9476#	9477#	9484#	9485#	9486#	9487#	9491#	9494#	9552#	9561#	9562#
9571#	9572#	9573#	9574#	9576#	9577#	9581#	9582#	9583#	9584#	9586#	9587#	9591#	9607#	9614#
9615#	9616#	9617#	9619#	9620#	9628#	9629#	9630#	9631#	9633#	9634#	9637#	9638#	9639#	9640#
9642#	9643#	9657#	9658#	9659#	9660#	9664#	9671#	9672#	9673#	9674#	9676#	9677#	9689#	9741#
9750#	9751#	9760#	9761#	9762#	9763#	9765#	9766#	9770#	9771#	9772#	9773#	9775#	9776#	9780#
9792#	9800#	9801#	9802#	9803#	9805#	9806#	9814#	9815#	9816#	9817#	9819#	9820#	9823#	9824#
9825#	9826#	9828#	9829#	9838#	9839#	9840#	9841#	9845#	9852#	9853#	9854#	9855#	9857#	9858#
9872#	9904#	9913#	9914#	9923#	9924#	9925#	9926#	9928#	9929#	9933#	9934#	9935#	9936#	9938#
9939#	9943#	9959#	9963#	9964#	9965#	9966#	9968#	9969#	9977#	9978#	9979#	9980#	9982#	9983#
9986#	9987#	9988#	9989#	9991#	9992#	10000#	10001#	10002#	10003#	10007#	10014#	10015#	10016#	10017#
10019#	10020#	10033#	10143#	10152#	10153#	10162#	10163#	10164#	10165#	10167#	10168#	10172#	10173#	10174#
10175#	10177#	10178#	10182#	10205#	10214#	10215#	10216#	10217#	10219#	10220#	10228#	10229#	10230#	10231#
10233#	10234#	10237#	10238#	10239#	10240#	10242#	10243#	10254#	10255#	10256#	10257#	10270#	10271#	10272#
10273#	10284#	10285#	10286#	10287#	10291#	10294#	10366#	10375#	10376#	10385#	10386#	10387#	10388#	10390#
10391#	10395#	10396#	10397#	10398#	10400#	10401#	10405#	10416#	10424#	10425#	10426#	10427#	10429#	10430#
10447#	10448#	10449#	10450#	10452#	10453#	10455#	10456#	10457#	10458#	10466#	10467#	10468#	10469#	10473#
10483#	10521#	10530#	10531#	10540#	10541#	10542#	10543#	10545#	10546#	10550#	10551#	10552#	10553#	10555#
10556#	10560#	10566#	10570#	10571#	10572#	10573#	10575#	10576#	10586#	10587#	10588#	10589#	10591#	10592#
10595#	10596#	10597#	10598#	10600#	10601#	10609#	10610#	10611#	10612#	10619#	10620#	10621#	10622#	10626#
10629#	10681#	10690#	10691#	10700#	10701#	10702#	10703#	10705#	10706#	10710#	10711#	10712#	10713#	10715#
10716#	10720#	10732#	10736#	10737#	10738#	10739#	10741#	10742#	10751#	10752#	10753#	10754#	10756#	10757#
10760#	10761#	10762#	10763#	10765#	10766#	10773#	10774#	10775#	10776#	10780#	10785#	10789#	10790#	10791#
10792#	10794#	10795#	10799#	10808#	10848#	10857#	10858#	10867#	10868#	10869#	10870#	10872#	10873#	10877#
10878#	10879#	10880#	10882#	10883#	10887#	10892#	10896#	10897#	10898#	10899#	10901#	10902#	10918#	10919#
10920#	10921#	10923#	10924#	10927#	10928#	10929#	10930#	10932#	10933#	10941#	10942#	10943#	10944#	10951#
10952#	10953#	10954#	10958#	10961#	11020#	11029#	11030#	11039#	11040#	11041#	11042#	11044#	11045#	11049#
11050#	11051#	11052#	11054#	11055#	11059#	11068#	11075#	11076#	11077#	11078#	11080#	11081#	11094#	11095#
11096#	11097#	11099#	11100#	11103#	11104#	11105#	11106#	11108#	11109#	11116#	11117#	11118#	11119#	11126#
11127#	11128#	11129#	11134#	11142#	11188#	11197#	11198#	11207#	11208#	11209#	11210#	11212#	11213#	11217#
11218#	11219#	11220#	11222#	11223#	11227#	11241#	11245#	11246#	11247#	11248#	11250#	11251#	11264#	11265#
11266#	11267#	11269#	11270#	11273#	11274#	11275#	11276#	11278#	11279#	11289#	11290#	11291#	11292#	11297#
11298#	11299#	11300#	11307#	11308#	11309#	11310#	11314#	11323#	11377#	11386#	11387#	11396#	11397#	11398#
11399#	11401#	11402#	11406#	11407#	11408#	11409#	11411#	11412#	11416#	11428#	11438#	11448#	11449#	11450#
11451#	11453#	11454#	11462#	11463#	11464#	11465#	11467#	11468#	11471#	11472#	11473#	11474#	11476#	11477#



67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 324  
CZUAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

3117#	3122#	3126#	3129#	3135#	3139#	3142#	3147#	3151#	3154#	3159#	3165#	3172#	3176#	3179#
3185#	3189#	3192#	3198#	3205#	3209#	3212#	3219#	3223#	3226#	3234#	3238#	3241#	3247#	3254#
3262#	3266#	3269#	3275#	3282#	3290#	3294#	3297#	3304#	3308#	3311#	3317#	3325#	3333#	3337#
3340#	3345#	3352#	3360#	3364#	3367#	3372#	3379#	3387#	3391#	3394#	3401#	3411#	3421#	3426#
3429#	3434#	3438#	3441#	3446#	3452#	3462#	3469#	3476#	3479#	3486#	3493#	3499#	3506#	3513#
3520#	3524#	3527#	3532#	3539#	3543#	3546#	3551#	3559#	3563#	3566#	3571#	3579#	3583#	3586#
3593#	3600#	3607#	3611#	3614#	3619#	3626#	3630#	3633#	3638#	3645#	3652#	3659#	3667#	3671#
3674#	3679#	3687#	3694#	3698#	3701#	3707#	3711#	3714#	3720#	3727#	3731#	3734#	3740#	3746#
3754#	3758#	3761#	3766#	3770#	3773#	3778#	3782#	3785#	3790#	3794#	3797#	3802#	3806#	3809#
3814#	3821#	3829#	3833#	3836#	3841#	3849#	3853#	3856#	3862#	3869#	3876#	3883#	3891#	3895#
3898#	3903#	3907#	3910#	3917#	3925#	3933#	3941#	3946#	3949#	3954#	3960#	3966#	3970#	3973#
3978#	3982#	4234#	4248#	4277#	4390#	4427#	4462#	4546#	4557#	4579#	4645#	4660#	4673#	4688#
4701#	4716#	4722#	4733#	4749#	4755#	4760#	4775#	4780#	4785#	4791#	4799#	4813#	4823#	4842#
4846#	4860#	4863#	4872#	4885#	4891#	4905#	4909#	4915#	4920#	4926#	4943#	4962#	4988#	5014#
5015#	5020#	5023#	5024#	5034#	5039#	5042#	5045#	5047#	5051#	5053#	5054#	5059#	5068#	5075#
5077#	5085#	5092#	5095#	5098#	5101#	5117#	5118#	5123#	5126#	5127#	5137#	5142#	5145#	5148#
5150#	5154#	5156#	5157#	5162#	5171#	5178#	5180#	5188#	5195#	5197#	5205#	5212#	5214#	5222#
5229#	5231#	5239#	5246#	5249#	5252#	5255#	5269#	5270#	5275#	5283#	5288#	5291#	5294#	5296#
5299#	5303#	5317#	5318#	5323#	5326#	5333#	5338#	5341#	5344#	5346#	5350#	5353#	5356#	5376#
5377#	5381#	5382#	5395#	5400#	5405#	5408#	5409#	5414#	5424#	5431#	5442#	5448#	5455#	5468#
5479#	5486#	5497#	5503#	5510#	5518#	5528#	5535#	5537#	5547#	5554#	5557#	5560#	5581#	5582#
5587#	5595#	5602#	5607#	5625#	5626#	5629#	5635#	5642#	5645#	5651#	5658#	5661#	5680#	5689#
5690#	5691#	5704#	5711#	5713#	5714#	5726#	5733#	5735#	5736#	5751#	5756#	5760#	5768#	5775#
5778#	5781#	5801#	5802#	5805#	5810#	5822#	5846#	5851#	5859#	5866#	6221#	6222#	6223#	6224#
6229#	6242#	6247#	6252#	6254#	6262#	6269#	6272#	6274#	6275#	6288#	6308#	6313#	6318#	6320#
6328#	6335#	6348#	6355#	6358#	6397#	6398#	6400#	6401#	6406#	6439#	6444#	6449#	6451#	6459#
6466#	6468#	6482#	6487#	6492#	6494#	6502#	6509#	6512#	6514#	6515#	6520#	6543#	6548#	6553#
6555#	6560#	6567#	6577#	6586#	6597#	6636#	6637#	6641#	6669#	6674#	6679#	6681#	6689#	6694#
6699#	6701#	6716#	6721#	6726#	6728#	6736#	6741#	6746#	6751#	6764#	6774#	6781#	6790#	6820#
6821#	6827#	6843#	6851#	6859#	6867#	6871#	6876#	6892#	6899#	6919#	6926#	6933#	6936#	6964#
6965#	6970#	6976#	6990#	6997#	7002#	7009#	7018#	7023#	7034#	7041#	7044#	7049#	7056#	7093#
7094#	7100#	7109#	7120#	7125#	7130#	7135#	7142#	7146#	7162#	7166#	7171#	7182#	7194#	7206#
7213#	7218#	7233#	7240#	7247#	7275#	7276#	7282#	7291#	7301#	7306#	7311#	7316#	7321#	7329#
7333#	7338#	7345#	7350#	7366#	7373#	7380#	7383#	7421#	7422#	7428#	7437#	7447#	7452#	7457#
7462#	7467#	7476#	7480#	7485#	7501#	7506#	7510#	7515#	7533#	7540#	7552#	7559#	7562#	7586#
7587#	7593#	7602#	7612#	7617#	7622#	7627#	7632#	7646#	7650#	7655#	7665#	7670#	7674#	7679#
7691#	7697#	7704#	7709#	7715#	7745#	7746#	7752#	7761#	7771#	7776#	7781#	7786#	7791#	7800#
7804#	7809#	7818#	7823#	7827#	7832#	7846#	7853#	7860#	7865#	7874#	7898#	7899#	7905#	7921#
7929#	7937#	7946#	7951#	7956#	7965#	7970#	7974#	7991#	8003#	8006#	8033#	8034#	8040#	8049#
8059#	8064#	8069#	8074#	8079#	8086#	8090#	8095#	8104#	8111#	8116#	8120#	8125#	8133#	8140#
8143#	8171#	8172#	8178#	8187#	8197#	8202#	8207#	8212#	8217#	8224#	8228#	8233#	8242#	8249#
8254#	8258#	8263#	8271#	8278#	8281#	8319#	8320#	8326#	8335#	8345#	8350#	8355#	8360#	8365#
8374#	8378#	8383#	8397#	8402#	8406#	8411#	8424#	8434#	8441#	8444#	8481#	8482#	8488#	8497#
8507#	8512#	8517#	8522#	8527#	8535#	8539#	8544#	8558#	8563#	8567#	8572#	8585#	8595#	8602#
8605#	8634#	8635#	8641#	8650#	8660#	8665#	8670#	8675#	8680#	8693#	8701#	8706#	8715#	8720#
8724#	8729#	8742#	8752#	8759#	8764#	8811#	8812#	8818#	8827#	8837#	8842#	8847#	8852#	8857#
8864#	8868#	8873#	8886#	8891#	8895#	8900#	8916#	8929#	8938#	8950#	8959#	8968#	8978#	8985#
8988#	9038#	9039#	9045#	9054#	9064#	9069#	9074#	9079#	9084#	9096#	9104#	9109#	9118#	9123#
9127#	9132#	9150#	9157#	9164#	9169#	9180#	9207#	9208#	9214#	9223#	9233#	9238#	9243#	9248#
9253#	9265#	9273#	9278#	9287#	9292#	9296#	9301#	9320#	9327#	9334#	9339#	9350#	9380#	9381#
9387#	9396#	9406#	9411#	9416#	9421#	9426#	9433#	9437#	9442#	9452#	9457#	9461#	9466#	9474#
9484#	9491#	9494#	9545#	9546#	9552#	9561#	9571#	9576#	9581#	9586#	9591#	9607#	9614#	9619#
9628#	9633#	9637#	9642#	9657#	9664#	9671#	9676#	9689#	9734#	9735#	9741#	9750#	9760#	9765#
9770#	9775#	9780#	9792#	9800#	9805#	9814#	9819#	9823#	9828#	9838#	9845#	9852#	9857#	9872#
9897#	9898#	9904#	9913#	9923#	9928#	9933#	9938#	9943#	9959#	9963#	9968#	9977#	9982#	9986#
9991#	10000#	10007#	10014#	10019#	10033#	10136#	10137#	10143#	10152#	10162#	10167#	10172#	10177#	10182#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 325  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

10205#	10214#	10219#	10228#	10233#	10237#	10242#	10254#	10270#	10284#	10291#	10294#	10359#	10360#	10366#
10375#	10385#	10390#	10395#	10400#	10405#	10416#	10424#	10429#	10447#	10452#	10455#	10466#	10473#	10483#
10514#	10515#	10521#	10530#	10540#	10545#	10550#	10555#	10560#	10566#	10570#	10575#	10586#	10591#	10595#
10600#	10609#	10619#	10626#	10629#	10674#	10675#	10681#	10690#	10700#	10705#	10710#	10715#	10720#	10732#
10736#	10741#	10751#	10756#	10760#	10765#	10773#	10780#	10785#	10789#	10794#	10799#	10808#	10841#	10842#
10848#	10857#	10867#	10872#	10877#	10882#	10887#	10892#	10896#	10901#	10918#	10923#	10927#	10932#	10941#
10951#	10958#	10961#	11013#	11014#	11020#	11029#	11039#	11044#	11049#	11054#	11059#	11068#	11075#	11080#
11094#	11099#	11103#	11108#	11116#	11126#	11134#	11142#	11181#	11182#	11188#	11197#	11207#	11212#	11217#
11222#	11227#	11241#	11245#	11250#	11264#	11269#	11273#	11278#	11289#	11297#	11307#	11314#	11323#	11370#
11371#	11377#	11386#	11396#	11401#	11406#	11411#	11416#	11428#	11438#	11448#	11453#	11462#	11467#	11471#
11476#	11485#	11495#	11502#	11515#	11525#	11578#	11579#	11585#	11594#	11604#	11609#	11614#	11619#	11624#
11634#	11647#	11652#	11661#	11666#	11670#	11675#	11687#	11702#	11713#	11720#	11727#	11732#	11745#	11793#
11794#	11800#	11809#	11819#	11824#	11829#	11834#	11839#	11845#	11858#	11863#	11872#	11877#	11881#	11886#
11895#	11910#	11921#	11928#	11935#	11942#	12029#	12030#	12036#	12045#	12055#	12060#	12065#	12070#	12075#
12091#	12099#	12104#	12113#	12118#	12122#	12127#	12144#	12155#	12162#	12167#	12176#	12181#	12190#	12196#
12201#	12216#	12227#	12236#	12241#	12248#	12253#	12262#	12267#	12271#	12276#	12295#	12305#	12312#	12315#
12372#	12373#	12379#	12388#	12398#	12403#	12408#	12413#	12418#	12425#	12432#	12437#	12446#	12451#	12455#
12460#	12469#	12479#	12486#	12492#	12499#	12504#	12513#	12518#	12522#	12527#	12537#	12542#	12550#	12557#
12566#	12619#	12620#	12626#	12635#	12645#	12650#	12655#	12660#	12665#	12680#	12684#	12689#	12698#	12703#
12707#	12712#	12720#	12725#	12751#	12756#	12763#	12766#	12786#	12787#	12790#	12796#	12812#	12817#	12825#
12830#	12846#	12851#	12859#	12864#	12890#	12895#	12903#	12908#	12935#	12940#	12947#	12952#	12981#	12988#
12995#	13002#	13009#	13015#	13139#										

MSIOSE	1#	1416#													
MSLDRO	1#	1416#	4233#	4247#	4754#	4759#	4774#	4779#	4790#	4822#	5041#	5044#	5097#	5144#	5147#
	5251#	5290#	5293#	5298#	5340#	5343#	5352#	7008#	7043#	7181#					
MSMASK	1#	1416#													
MSM(HI	1#	1416#													
MSM(CLO	1#	1416#													
MSMSK1	1#	1416#													
MSPOP	1#	1416#	1590#	2924#	2940#	2961#	2974#	2987#	3004#	3023#	3040#	3053#	3074#	3100#	3113#
	3125#	3138#	3150#	3175#	3188#	3208#	3222#	3237#	3265#	3293#	3307#	3336#	3363#	3390#	3425#
	3437#	3475#	3523#	3542#	3562#	3582#	3610#	3629#	3670#	3697#	3710#	3730#	3757#	3769#	3781#
	3793#	3805#	3832#	3852#	3894#	3906#	3945#	3969#	3981#	4721#	4740#	4845#	4862#	4904#	4914#
	4925#	4946#	4966#	4994#	5050#	5074#	5091#	5094#	5100#	5153#	5177#	5194#	5211#	5228#	5245#
	5248#	5254#	5302#	5349#	5355#	5404#	5430#	5454#	5485#	5509#	5534#	5553#	5556#	5559#	5601#
	5606#	5641#	5657#	5660#	5710#	5732#	5774#	5777#	5780#	5865#	6251#	6268#	6271#	6317#	6334#
	6354#	6357#	6448#	6465#	6491#	6508#	6511#	6552#	6566#	6585#	6596#	6678#	6698#	6725#	6745#
	6780#	6789#	6858#	6932#	6935#	6996#	7040#	7055#	7145#	7212#	7239#	7246#	7320#	7379#	7382#
	7466#	7558#	7561#	7631#	7696#	7714#	7790#	7852#	7873#	7936#	8002#	8005#	8078#	8139#	8142#
	8216#	8277#	8280#	8364#	8440#	8443#	8526#	8601#	8604#	8679#	8758#	8763#	8856#	8984#	8987#
	9083#	9156#	9179#	9252#	9326#	9349#	9425#	9490#	9493#	9590#	9663#	9688#	9779#	9844#	9871#
	9942#	10006#	10032#	10181#	10290#	10293#	10404#	10472#	10482#	10559#	10625#	10628#	10719#	10779#	10798#
	10807#	10886#	10957#	10960#	11058#	11133#	11141#	11226#	11313#	11322#	11415#	11501#	11514#	11524#	11623#
	11719#	11744#	11838#	11927#	11941#	12074#	12161#	12235#	12311#	12314#	12417#	12485#	12556#	12565#	12664#
	12762#	12765#	13014#	13152#	13179#										
MSPRIN	1#	1416#	2917#	2930#	2946#	2953#	2967#	2980#	2993#	3010#	3016#	3029#	3046#	3059#	3065#
	3080#	3086#	3093#	3106#	3119#	3131#	3144#	3156#	3162#	3168#	3181#	3194#	3201#	3214#	3228#
	3243#	3250#	3257#	3271#	3278#	3285#	3299#	3313#	3320#	3328#	3342#	3348#	3355#	3369#	3375#
	3382#	3397#	3405#	3415#	3431#	3443#	3449#	3458#	3465#	3483#	3490#	3496#	3502#	3509#	3516#
	3529#	3535#	3548#	3554#	3568#	3574#	3588#	3596#	3603#	3616#	3622#	3635#	3641#	3648#	3655#
	3662#	3676#	3682#	3690#	3703#	3716#	3723#	3736#	3743#	3749#	3763#	3775#	3787#	3799#	3811#
	3817#	3824#	3838#	3844#	3858#	3865#	3872#	3879#	3886#	3900#	3914#	3922#	3930#	3938#	3951#
	3957#	3963#	3975#	4386#	4422#	4539#	4554#	4575#	4641#	4656#	4669#	4684#	4697#	4796#	4882#
	12977#	12984#	12991#	12998#	13005#										
MSPUSH	1#	1416#	1420#	1582#	2915#	2928#	2944#	2965#	2978#	2991#	3008#	3027#	3044#	3057#	3078#
	3104#	3117#	3129#	3142#	3154#	3179#	3192#	3212#	3226#	3241#	3269#	3297#	3311#	3340#	3367#



67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 327  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

MSRNR0	1#	1416#	4785#	4786	4790#	4792	4822#	4824							
MSSETS	1#	1416#	1420#	1582#	2915#	2928#	2944#	2965#	2978#	2991#	3008#	3027#	3044#	3057#	3078#
	3104#	3117#	3129#	3142#	3154#	3179#	3192#	3212#	3226#	3241#	3269#	3297#	3311#	3340#	3367#
	3394#	3429#	3441#	3479#	3527#	3546#	3566#	3586#	3614#	3633#	3674#	3701#	3714#	3734#	3761#
	3773#	3785#	3797#	3809#	3836#	3856#	3878#	3910#	3949#	3973#	4716#	4733#	4749#	4860#	4872#
	4909#	4920#	4943#	4962#	4988#	5015#	5024#	5054#	5059#	5077#	5118#	5127#	5157#	5162#	5180#
	5197#	5214#	5231#	5270#	5318#	5326#	5377#	5382#	5409#	5414#	5442#	5468#	5497#	5518#	5537#
	5582#	5587#	5626#	5629#	5645#	5689#	5691#	5714#	5736#	5760#	5802#	6222#	6224#	6229#	6254#
	6275#	6288#	6320#	6398#	6401#	6406#	6451#	6468#	6494#	6515#	6520#	6555#	6637#	6641#	6681#
	6701#	6728#	6751#	6821#	6827#	6867#	6965#	6970#	7002#	7094#	7100#	7162#	7218#	7276#	7282#
	7329#	7422#	7428#	7476#	7587#	7593#	7646#	7746#	7752#	7800#	7899#	7905#	7946#	8034#	8040#
	8086#	8172#	8178#	8224#	8320#	8326#	8374#	8482#	8488#	8535#	8635#	8641#	8693#	8812#	8818#
	8864#	9039#	9045#	9096#	9208#	9214#	9265#	9381#	9387#	9433#	9546#	9552#	9607#	9735#	9741#
	9792#	9898#	9904#	9959#	10137#	10143#	10205#	10360#	10366#	10416#	10515#	10521#	10566#	10675#	10681#
	10732#	10785#	10842#	10848#	10892#	11014#	11020#	11068#	11182#	11188#	11241#	11371#	11377#	11428#	11438#
	11579#	11585#	11634#	11794#	11800#	11845#	12030#	12036#	12091#	12167#	12241#	12373#	12379#	12425#	12492#
	12620#	12626#	12680#	12787#	13139#										
MSSTAR	1#	1416#													
MSVC	1#	1416#	2917#	2921	2924#	2925	2930#	2936	2940#	2941	2946#	2950	2953#	2958	2961#
	2962	2967#	2970	2974#	2975	2980#	2983	2987#	2988	2993#	3000	3004#	3005	3010#	3013
	3016#	3020	3023#	3024	3029#	3036	3040#	3041	3046#	3049	3053#	3054	3059#	3062	3065#
	3071	3074#	3075	3080#	3083	3086#	3090	3093#	3097	3100#	3101	3106#	3109	3113#	3114
	3119#	3122	3125#	3126	3131#	3135	3138#	3139	3144#	3147	3150#	3151	3156#	3159	3162#
	3165	3168#	3172	3175#	3176	3181#	3185	3188#	3189	3194#	3198	3201#	3205	3208#	3209
	3214#	3219	3222#	3223	3228#	3234	3237#	3238	3243#	3247	3250#	3254	3257#	3262	3265#
	3266	3271#	3275	3278#	3282	3285#	3290	3293#	3294	3299#	3304	3307#	3308	3313#	3317
	3320#	3325	3328#	3333	3336#	3337	3342#	3345	3348#	3352	3355#	3360	3363#	3364	3369#
	3372	3375#	3379	3382#	3387	3390#	3391	3397#	3401	3405#	3411	3415#	3421	3425#	3426
	3431#	3434	3437#	3436	3443#	3446	3449#	3452	3458#	3462	3465#	3469	3475#	3476	3483#
	3486	3490#	3493	3496#	3499	3502#	3506	3509#	3513	3516#	3520	3523#	3524	3529#	3532
	3535#	3539	3542#	3543	3548#	3551	3554#	3559	3562#	3563	3568#	3571	3574#	3579	3582#
	3583	3588#	3593	3596#	3600	3603#	3607	3610#	3611	3616#	3619	3622#	3626	3629#	3630
	3635#	3638	3641#	3645	3648#	3652	3655#	3659	3662#	3667	3670#	3671	3676#	3679	3682#
	3687	3690#	3694	3697#	3698	3703#	3707	3710#	3711	3716#	3720	3723#	3727	3730#	3731
	3736#	3740	3743#	3746	3749#	3754	3757#	3758	3763#	3766	3769#	3770	3775#	3778	3781#
	3782	3787#	3790	3793#	3794	3799#	3802	3805#	3806	3811#	3814	3817#	3821	3824#	3829
	3832#	3833	3838#	3841	3844#	3849	3852#	3853	3858#	3862	3865#	3869	3872#	3876	3879#
	3883	3886#	3891	3894#	3895	3900#	3903	3906#	3907	3914#	3917	3922#	3925	3930#	3933
	3938#	3941	3945#	3946	3951#	3954	3957#	3960	3963#	3966	3969#	3970	3975#	3978	3981#
	3982	4233#	4234	4247#	4248	4277#	4386#	4390	4422#	4427	4462#	4539#	4546	4554#	4557
	4575#	4579	4641#	4645	4656#	4660	4669#	4673	4684#	4688	4697#	4701	4718#	4721#	4722
	4754#	4755	4759#	4760	4774#	4775	4779#	4780	4785#	4790#	4791	4796#	4799	4809#	4813
	4822#	4823	4842#	4845#	4846	4862#	4863	4882#	4885	4891#	4904#	4905	4911#	4914#	4915
	4922#	4925#	4926	5016#	5020	5023#	5024	5034	5039#	5041#	5042	5044#	5045	5047#	5050#
	5051	5053#	5054	5059#	5068	5074#	5075	5077#	5085	5091#	5092	5094#	5095	5097#	5098
	5100#	5101	5119#	5123	5126#	5127	5137	5142#	5144#	5145	5147#	5148	5150#	5153#	5154
	5156#	5157	5162#	5171	5177#	5178	5180#	5188	5194#	5195	5197#	5205	5211#	5212	5214#
	5222	5228#	5229	5231#	5239	5245#	5246	5248#	5249	5251#	5252	5254#	5255	5271#	5275
	5283	5288#	5290#	5291	5293#	5294	5296#	5298#	5299	5302#	5303	5319#	5323	5326#	5333
	5338#	5340#	5341	5343#	5344	5346#	5349#	5350	5352#	5353	5355#	5356	5381#	5382	5395
	5400#	5404#	5405	5408#	5409	5414#	5424	5430#	5431	5442#	5448	5454#	5455	5468#	5479
	5485#	5486	5497#	5503	5509#	5510	5518#	5528	5534#	5535	5537#	5547	5553#	5554	5556#
	5557	5559#	5560	5587#	5595	5601#	5602	5606#	5607	5629#	5635	5641#	5642	5645#	5651
	5657#	5658	5660#	5661	5690#	5691	5704	5710#	5711	5713#	5714	5726	5732#	5733	5735#
	5736	5751	5756#	5760#	5768	5774#	5775	5777#	5778	5780#	5781	5805	5810#	5822	5846
	5851#	5859	5865#	5866	6223#	6224	6229#	6242	6247#	6251#	6252	6254#	6262	6268#	6269



PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 328  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

6271#	6272	6274#	6275	6288#	6308	6313#	6317#	6318	6320#	6328	6334#	6335	6348	6354#
6355	6357#	6358	6400#	6401	6406#	6439	6444#	6448#	6449	6451#	6459	6465#	6466	6468#
6482	6487#	6491#	6492	6494#	6502	6508#	6509	6511#	6512	6514#	6515	6520#	6543	6548#
6552#	6553	6555#	6560	6566#	6567	6577	6585#	6586	6596#	6597	6641#	6669	6674#	6678#
6679	6681#	6689	6694#	6698#	6699	6701#	6716	6721#	6725#	6726	6728#	6736	6741#	6745#
6746	6751#	6764	6774	6780#	6781	6789#	6790	6827#	6843	6851	6858#	6859	6867#	6871
6876#	6892	6899	6919	6926	6932#	6933	6935#	6936	6970#	6972#	6976	6990	6996#	6997
7002#	7008#	7009	7018	7023#	7034	7040#	7041	7043#	7044	7049	7055#	7056	7100#	7109#
7120	7125#	7130	7135#	7138#	7142	7145#	7146	7162#	7166	7171#	7181#	7182	7194	7206
7212#	7213	7218#	7233	7239#	7240	7246#	7247	7282#	7291#	7301	7306#	7311	7316#	7320#
7321	7329#	7333	7338#	7345	7350#	7366	7373	7379#	7380	7382#	7383	7428#	7437#	7447
7452#	7457	7462#	7466#	7467	7476#	7480	7485#	7501	7506#	7510	7515#	7533	7540	7552
7558#	7559	7561#	7562	7593#	7602#	7612	7617#	7622	7627#	7631#	7632	7646#	7650	7655#
7665	7670#	7674	7679#	7691	7696#	7697	7704	7709#	7714#	7715	7752#	7761#	7771	7776#
7781	7786#	7790#	7791	7800#	7804	7809#	7818	7823#	7827	7832#	7846	7852#	7853	7860
7865#	7873#	7874	7905#	7921	7929	7936#	7937	7946#	7951	7956#	7965	7970#	7974	7991
8002#	8003	8005#	8006	8040#	8049#	8059	8064#	8069	8074#	8078#	8079	8086#	8090	8095#
8104	8111	8116#	8120	8125#	8133	8139#	8140	8142#	8143	8178#	8187#	8197	8202#	8207
8212#	8216#	8217	8224#	8228	8233#	8242	8249	8254#	8258	8263#	8271	8277#	8278	8280#
8281	8326#	8335#	8345	8350#	8355	8360#	8364#	8365	8374#	8378	8383#	8397	8402#	8406
8411#	8424	8434	8440#	8441	8443#	8444	8488#	8497#	8507	8512#	8517	8522#	8526#	8527
8535#	8539	8544#	8558	8563#	8567	8572#	8585	8595	8601#	8602	8604#	8605	8641#	8650#
8660	8665#	8670	8675#	8679#	8680	8693#	8701	8706#	8715	8720#	8724	8729#	8742	8752
8758#	8759	8763#	8764	8818#	8827#	8837	8842#	8847	8852#	8856#	8857	8864#	8868	8873#
8886	8891#	8895	8900#	8916	8929	8938	8950	8959	8968	8978	8984#	8985	8987#	8988
9045#	9054#	9064	9069#	9074	9079#	9083#	9084	9096#	9104	9109#	9118	9123#	9127	9132#
9150	9156#	9157	9164	9169#	9179#	9180	9214#	9223#	9233	9238#	9243	9248#	9252#	9253
9265#	9273	9278#	9287	9292#	9296	9301#	9320	9326#	9327	9334	9339#	9349#	9350	9387#
9396#	9406	9411#	9416	9421#	9425#	9426	9433#	9437	9442#	9452	9457#	9461	9466#	9474
9484	9490#	9491	9493#	9494	9552#	9561#	9571	9576#	9581	9586#	9590#	9591	9607#	9614
9619#	9628	9633#	9637	9642#	9657	9663#	9664	9671	9676#	9688#	9689	9741#	9750#	9760
9765#	9770	9775#	9779#	9780	9792#	9800	9805#	9814	9819#	9823	9828#	9838	9844#	9845
9852	9857#	9871#	9879#	9904#	9913#	9923	9928#	9933	9938#	9942#	9943	9959#	9963	9968#
9977	9982#	9986	9991#	10000	10006#	10007	10014	10019#	10032#	10033	10143#	10152#	10162	10167#
10172	10177#	10181#	10182	10205#	10214	10219#	10228	10233#	10237	10242#	10254	10270	10284	10290#
10291	10293#	10294	10366#	10375#	10385	10390#	10395	10400#	10404#	10405	10416#	10424	10429#	10447
10452#	10455	10466	10472#	10473	10482#	10483	10521#	10530#	10540	10545#	10550	10555#	10559#	10560
10566#	10570	10575#	10586	10591#	10595	10600#	10609	10619	10625#	10626	10628#	10629	10681#	10690#
10700	10705#	10710	10715#	10719#	10720	10732#	10736	10741#	10751	10756#	10760	10765#	10773	10779#
10780	10785#	10789	10794#	10798#	10799	10807#	10808	10848#	10857#	10867	10872#	10877	10882#	10886#
10887	10892#	10896	10901#	10918	10923#	10927	10932#	10941	10951	10957#	10958	10960#	10961	11020#
11029#	11039	11044#	11049	11054#	11058#	11059	11068#	11075	11080#	11094	11099#	11103	11108#	11116
11126	11133#	11134	11141#	11142	11188#	11197#	11207	11212#	11217	11222#	11226#	11227	11241#	11245
11250#	11264	11269#	11273	11278#	11289	11297	11307	11313#	11314	11322#	11323	11377#	11386#	11396
11401#	11406	11411#	11415#	11416	11428#	11438#	11448	11453#	11462	11467#	11471	11476#	11485	11495
11501#	11502	11514#	11515	11524#	11525	11585#	11594#	11604	11609#	11614	11619#	11623#	11624	11634#
11647	11652#	11661	11666#	11670	11675#	11687	11702	11713	11719#	11720	11732#	11744#	11745	11745
11800#	11809#	11819	11824#	11829	11834#	11838#	11839	11845#	11858	11863#	11872	11877#	11881	11886#
11895	11910	11921	11927#	11928	11935	11941#	11942	12036#	12045#	12055	12060#	12065	12070#	12074#
12075	12091#	12099	12104#	12113	12118#	12122	12127#	12144	12155	12161#	12162	12167#	12176	12181#
12190	12196	12201#	12216	12227	12235#	12236	12241#	12248	12253#	12262	12267#	12271	12276#	12295
12305	12311#	12312	12314#	12315	12379#	12388#	12398	12403#	12408	12413#	12417#	12418	12425#	12432
12437#	12446	12451#	12455	12460#	12469	12479	12485#	12486	12492#	12499	12504#	12513	12518#	12522
12527#	12537	12542#	12550	12556#	12557	12565#	12566	12626#	12635#	12645	12650#	12655	12660#	12664#
12665	12680#	12684	12689#	12698	12703#	12707	12712#	12720	12725#	12751	12756#	12762#	12763	12765#
12766	12790#	12796#	12812	12817#	12825	12830#	12846	12851#	12859	12864#	12890	12895#	12903	12908#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 329  
 CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

	12935	12940#	12947	12952#	12977#	12981	12984#	12988	12991#	12995	12998#	13002	13005#	13009	13014#
MSTLAB	1#	1416#	2921#	2925#	2936#	2941#	2950#	2958#	2962#	2970#	2975#	2983#	2988#	3000#	3005#
	13015														
	3013#	3020#	3024#	3036#	3041#	3049#	3054#	3062#	3071#	3075#	3083#	3090#	3097#	3101#	3109#
	3114#	3122#	3126#	3135#	3139#	3147#	3151#	3159#	3165#	3172#	3176#	3185#	3189#	3198#	3205#
	3209#	3219#	3223#	3234#	3238#	3247#	3254#	3262#	3266#	3275#	3282#	3290#	3294#	3304#	3308#
	3317#	3325#	3333#	3337#	3345#	3352#	3360#	3364#	3372#	3379#	3387#	3391#	3401#	3411#	3421#
	3426#	3434#	3438#	3446#	3452#	3462#	3469#	3476#	3486#	3493#	3499#	3506#	3513#	3520#	3524#
	3532#	3539#	3543#	3551#	3559#	3563#	3571#	3579#	3583#	3593#	3600#	3607#	3611#	3619#	3626#
	3630#	3638#	3645#	3652#	3659#	3667#	3671#	3679#	3687#	3694#	3698#	3707#	3711#	3720#	3727#
	3731#	3740#	3746#	3754#	3758#	3766#	3770#	3778#	3782#	3790#	3794#	3802#	3806#	3814#	3821#
	3829#	3833#	3841#	3849#	3853#	3862#	3869#	3876#	3883#	3891#	3895#	3903#	3907#	3917#	3925#
	3933#	3941#	3946#	3954#	3960#	3966#	3970#	3978#	3982#	4234#	4248#	4277#	4390#	4427#	4462#
	4546#	4557#	4579#	4645#	4660#	4673#	4688#	4701#	4722#	4755#	4760#	4775#	4780#	4785#	4791#
	4799#	4813#	4823#	4842#	4846#	4863#	4885#	4891#	4905#	4915#	4926#	5020#	5024#	5034#	5039#
	5042#	5045#	5047#	5051#	5054#	5059#	5068#	5075#	5077#	5085#	5092#	5095#	5098#	5101#	5123#
	5127#	5137#	5142#	5145#	5148#	5150#	5154#	5157#	5162#	5171#	5178#	5180#	5188#	5195#	5197#
	5205#	5212#	5214#	5222#	5229#	5231#	5239#	5246#	5249#	5252#	5255#	5275#	5283#	5288#	5291#
	5294#	5296#	5299#	5303#	5323#	5326#	5333#	5338#	5341#	5344#	5346#	5350#	5353#	5356#	5382#
	5395#	5400#	5405#	5409#	5414#	5424#	5431#	5442#	5448#	5455#	5468#	5479#	5486#	5497#	5503#
	5510#	5518#	5528#	5535#	5537#	5547#	5554#	5557#	5560#	5587#	5595#	5602#	5607#	5629#	5635#
	5642#	5645#	5651#	5658#	5661#	5691#	5704#	5711#	5714#	5726#	5733#	5736#	5751#	5756#	5760#
	5768#	5775#	5778#	5781#	5805#	5810#	5822#	5846#	5851#	5859#	5866#	6224#	6229#	6242#	6247#
	6252#	6254#	6262#	6269#	6272#	6275#	6288#	6308#	6313#	6318#	6320#	6328#	6335#	6348#	6355#
	6358#	6401#	6406#	6439#	6444#	6449#	6451#	6459#	6466#	6468#	6482#	6487#	6492#	6494#	6502#
	6509#	6512#	6515#	6520#	6543#	6548#	6553#	6555#	6560#	6567#	6577#	6586#	6597#	6641#	6669#
	6674#	6679#	6681#	6689#	6694#	6699#	6701#	6716#	6721#	6726#	6728#	6736#	6741#	6746#	6751#
	6764#	6774#	6781#	6790#	6827#	6843#	6851#	6859#	6867#	6871#	6876#	6892#	6899#	6919#	6926#
	6933#	6936#	6970#	6976#	6990#	6997#	7002#	7009#	7018#	7023#	7034#	7041#	7044#	7049#	7056#
	7100#	7109#	7120#	7125#	7130#	7135#	7142#	7146#	7162#	7166#	7171#	7182#	7194#	7206#	7213#
	7218#	7233#	7240#	7247#	7282#	7291#	7301#	7306#	7311#	7316#	7321#	7329#	7333#	7338#	7345#
	7350#	7366#	7373#	7380#	7383#	7428#	7437#	7447#	7452#	7457#	7462#	7467#	7476#	7480#	7485#
	7501#	7506#	7510#	7515#	7533#	7540#	7552#	7559#	7562#	7593#	7602#	7612#	7617#	7622#	7627#
	7632#	7646#	7650#	7655#	7665#	7670#	7674#	7679#	7691#	7697#	7704#	7709#	7715#	7722#	7761#
	7771#	7776#	7781#	7786#	7791#	7800#	7804#	7809#	7818#	7823#	7827#	7832#	7846#	7853#	7860#
	7865#	7874#	7905#	7921#	7929#	7937#	7946#	7951#	7956#	7965#	7970#	7974#	7991#	8003#	8006#
	8040#	8049#	8059#	8064#	8069#	8074#	8079#	8086#	8090#	8095#	8104#	8111#	8116#	8120#	8125#
	8133#	8140#	8143#	8178#	8187#	8197#	8202#	8207#	8212#	8217#	8224#	8228#	8233#	8242#	8249#
	8254#	8258#	8263#	8271#	8278#	8281#	8326#	8335#	8345#	8350#	8355#	8360#	8365#	8374#	8378#
	8383#	8397#	8402#	8406#	8411#	8424#	8434#	8441#	8444#	8488#	8497#	8507#	8512#	8517#	8522#
	8527#	8535#	8539#	8544#	8558#	8563#	8567#	8572#	8585#	8595#	8602#	8605#	8641#	8650#	8660#
	8665#	8670#	8675#	8680#	8693#	8701#	8706#	8715#	8720#	8724#	8729#	8742#	8752#	8759#	8764#
	8818#	8827#	8837#	8842#	8847#	8852#	8857#	8864#	8868#	8873#	8886#	8891#	8895#	8900#	8916#
	8929#	8938#	8950#	8959#	8968#	8978#	8985#	8988#	9045#	9054#	9064#	9069#	9074#	9079#	9084#
	9096#	9104#	9109#	9118#	9123#	9127#	9132#	9150#	9157#	9164#	9169#	9180#	9214#	9223#	9233#
	9238#	9243#	9248#	9253#	9265#	9273#	9278#	9287#	9292#	9296#	9301#	9320#	9327#	9334#	9339#
	9350#	9387#	9396#	9406#	9411#	9416#	9421#	9426#	9433#	9437#	9442#	9452#	9457#	9461#	9466#
	9474#	9484#	9491#	9494#	9552#	9561#	9571#	9576#	9581#	9586#	9591#	9607#	9614#	9619#	9628#
	9633#	9637#	9642#	9657#	9664#	9671#	9676#	9689#	9741#	9750#	9760#	9765#	9770#	9775#	9780#
	9792#	9800#	9805#	9814#	9819#	9823#	9828#	9838#	9845#	9852#	9857#	9872#	9904#	9913#	9923#
	9928#	9933#	9938#	9943#	9959#	9963#	9968#	9977#	9982#	9986#	9991#	10000#	10007#	10014#	10019#
	10033#	10143#	10152#	10162#	10167#	10172#	10177#	10182#	10205#	10214#	10219#	10228#	10233#	10237#	10242#
	10254#	10270#	10284#	10291#	10294#	10366#	10375#	10385#	10390#	10395#	10400#	10405#	10416#	10424#	10429#
	10447#	10452#	10455#	10466#	10473#	10483#	10521#	10530#	10540#	10545#	10550#	10555#	10560#	10566#	10570#
	10575#	10586#	10591#	10595#	10600#	10609#	10619#	10626#	10629#	10681#	10690#	10700#	10705#	10710#	10715#
	10720#	10732#	10736#	10741#	10751#	10756#	10760#	10765#	10773#	10780#	10785#	10789#	10794#	10799#	10808#

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 330  
CZUAA.B.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

10848#	10857#	10867#	10872#	10877#	10882#	10887#	10892#	10896#	10901#	10918#	10923#	10927#	10932#	10941#	
10951#	10958#	10961#	11020#	11029#	11039#	11044#	11049#	11054#	11059#	11068#	11075#	11080#	11094#	11099#	
11103#	11108#	11116#	11126#	11134#	11142#	11188#	11197#	11207#	11212#	11217#	11222#	11227#	11241#	11245#	
11250#	11264#	11269#	11273#	11278#	11289#	11297#	11307#	11314#	11323#	11377#	11386#	11396#	11401#	11406#	
11411#	11416#	11428#	11438#	11448#	11453#	11462#	11467#	11471#	11476#	11485#	11495#	11502#	11515#	11525#	
11585#	11594#	11604#	11609#	11614#	11619#	11624#	11634#	11647#	11652#	11661#	11666#	11670#	11675#	11687#	
11702#	11713#	11720#	11727#	11732#	11745#	11800#	11809#	11819#	11824#	11829#	11834#	11839#	11845#	11858#	
11863#	11872#	11877#	11881#	11886#	11895#	11910#	11921#	11928#	11935#	11942#	12036#	12045#	12055#	12060#	
12065#	12070#	12075#	12091#	12099#	12104#	12113#	12118#	12122#	12127#	12144#	12155#	12162#	12167#	12176#	
12181#	12190#	12196#	12201#	12216#	12227#	12236#	12241#	12248#	12253#	12262#	12267#	12271#	12276#	12295#	
12305#	12312#	12315#	12379#	12388#	12398#	12403#	12408#	12413#	12418#	12425#	12432#	12437#	12446#	12451#	
12455#	12460#	12469#	12479#	12486#	12492#	12499#	12504#	12513#	12518#	12522#	12527#	12537#	12542#	12550#	
12557#	12566#	12626#	12635#	12645#	12650#	12655#	12660#	12665#	12680#	12694#	12689#	12698#	12703#	12707#	
12712#	12720#	12725#	12751#	12756#	12763#	12766#	12790#	12796#	12812#	12817#	12825#	12830#	12846#	12851#	
12859#	12864#	12890#	12895#	12903#	12908#	12935#	12940#	12947#	12952#	12981#	12988#	12995#	13002#	13009#	
13015#															
MSTSTL	1#	1416#	2921#	2925#	2936#	2941#	2950#	2958#	2962#	2970#	2975#	2983#	2988#	3000#	3005#
	3013#	3020#	3024#	3036#	3041#	3049#	3054#	3062#	3071#	3075#	3083#	3090#	3097#	3101#	3109#
	3114#	3122#	3126#	3135#	3139#	3147#	3151#	3159#	3165#	3172#	3176#	3185#	3189#	3198#	3205#
	3209#	3219#	3223#	3234#	3238#	3247#	3254#	3262#	3266#	3275#	3282#	3290#	3294#	3304#	3308#
	3317#	3325#	3333#	3337#	3345#	3352#	3360#	3364#	3372#	3379#	3387#	3391#	3401#	3411#	3421#
	3426#	3434#	3438#	3446#	3452#	3462#	3469#	3476#	3486#	3493#	3499#	3506#	3513#	3520#	3524#
	3532#	3539#	3543#	3551#	3559#	3563#	3571#	3579#	3583#	3593#	3600#	3607#	3611#	3619#	3626#
	3630#	3638#	3645#	3652#	3659#	3667#	3671#	3679#	3687#	3694#	3698#	3707#	3711#	3720#	3727#
	3731#	3740#	3746#	3754#	3758#	3766#	3770#	3778#	3782#	3790#	3794#	3802#	3806#	3814#	3821#
	3829#	3833#	3841#	3849#	3853#	3862#	3869#	3876#	3883#	3891#	3895#	3903#	3907#	3917#	3925#
	3933#	3941#	3946#	3954#	3960#	3966#	3970#	3978#	3982#	4234#	4248#	4277#	4390#	4427#	4462#
	4546#	4557#	4579#	4645#	4660#	4673#	4688#	4701#	4722#	4755#	4760#	4775#	4780#	4785#	4791#
	4799#	4813#	4823#	4842#	4846#	4863#	4885#	4891#	4905#	4915#	4926#	5020#	5024#	5034#	5039#
	5042#	5045#	5047#	5051#	5054#	5059#	5068#	5075#	5077#	5085#	5092#	5095#	5098#	5101#	5123#
	5127#	5137#	5142#	5145#	5148#	5150#	5154#	5157#	5162#	5171#	5178#	5180#	5188#	5195#	5197#
	5205#	5212#	5214#	5222#	5229#	5231#	5239#	5246#	5249#	5252#	5255#	5275#	5283#	5288#	5291#
	5294#	5296#	5299#	5303#	5323#	5326#	5333#	5338#	5341#	5344#	5346#	5350#	5353#	5356#	5382#
	5395#	5400#	5405#	5409#	5414#	5424#	5431#	5442#	5448#	5455#	5468#	5479#	5486#	5497#	5503#
	5510#	5518#	5528#	5535#	5537#	5547#	5554#	5557#	5560#	5587#	5595#	5602#	5607#	5629#	5635#
	5642#	5645#	5651#	5658#	5661#	5691#	5704#	5711#	5714#	5726#	5733#	5736#	5751#	5756#	5760#
	5768#	5775#	5778#	5781#	5805#	5810#	5822#	5846#	5851#	5859#	5866#	6224#	6229#	6242#	6247#
	6252#	6254#	6262#	6269#	6272#	6275#	6288#	6308#	6313#	6318#	6320#	6328#	6335#	6348#	6355#
	6358#	6401#	6406#	6439#	6444#	6449#	6451#	6459#	6466#	6468#	6482#	6487#	6492#	6494#	6502#
	6509#	6512#	6515#	6520#	6543#	6548#	6553#	6555#	6560#	6567#	6577#	6586#	6597#	6641#	6669#
	6674#	6679#	6681#	6689#	6694#	6699#	6701#	6716#	6721#	6726#	6728#	6736#	6741#	6746#	6751#
	6764#	6774#	6781#	6790#	6827#	6843#	6851#	6859#	6867#	6871#	6876#	6892#	6899#	6919#	6926#
	6933#	6936#	6970#	6976#	6990#	6997#	7002#	7009#	7018#	7023#	7034#	7041#	7044#	7049#	7056#
	7100#	7109#	7120#	7125#	7130#	7135#	7142#	7146#	7162#	7166#	7171#	7182#	7194#	7206#	7213#
	7218#	7233#	7240#	7247#	7282#	7291#	7301#	7306#	7311#	7316#	7321#	7329#	7333#	7338#	7345#
	7350#	7366#	7373#	7380#	7383#	7428#	7437#	7447#	7452#	7457#	7462#	7467#	7476#	7480#	7485#
	7501#	7506#	7510#	7515#	7533#	7540#	7552#	7559#	7562#	7593#	7602#	7612#	7617#	7622#	7627#
	7632#	7646#	7650#	7655#	7665#	7670#	7674#	7679#	7691#	7697#	7704#	7709#	7715#	7722#	7761#
	7771#	7776#	7781#	7786#	7791#	7800#	7804#	7809#	7818#	7823#	7827#	7832#	7846#	7853#	7860#
	7865#	7874#	7905#	7921#	7929#	7937#	7946#	7951#	7956#	7965#	7970#	7974#	7991#	8003#	8006#
	8040#	8049#	8059#	8064#	8069#	8074#	8079#	8086#	8090#	8095#	8104#	8111#	8116#	8120#	8125#
	8133#	8140#	8143#	8178#	8187#	8197#	8202#	8207#	8212#	8217#	8224#	8228#	8233#	8242#	8249#
	8254#	8258#	8263#	8271#	8278#	8281#	8326#	8335#	8345#	8350#	8355#	8360#	8365#	8374#	8378#
	8383#	8397#	8402#	8406#	8411#	8424#	8434#	8441#	8444#	8488#	8497#	8507#	8512#	8517#	8522#
	8527#	8535#	8539#	8544#	8558#	8563#	8567#	8572#	8585#	8595#	8602#	8605#	8641#	8650#	8660#
	8665#	8670#	8675#	8680#	8693#	8701#	8706#	8715#	8720#	8724#	8729#	8742#	8752#	8759#	8764#

57PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 331  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

8818#	8827#	8837#	8842#	8847#	8852#	8857#	8864#	8868#	8873#	8886#	8891#	8895#	8900#	8916#
8929#	8938#	8950#	8959#	8968#	8978#	8985#	8988#	9045#	9054#	9064#	9069#	9074#	9079#	9084#
9096#	9104#	9109#	9118#	9123#	9127#	9132#	9150#	9157#	9164#	9169#	9180#	9214#	9223#	9233#
9238#	9243#	9248#	9253#	9265#	9273#	9278#	9287#	9292#	9296#	9301#	9320#	9327#	9334#	9339#
9350#	9387#	9396#	9406#	9411#	9416#	9421#	9426#	9433#	9437#	9442#	9452#	9457#	9461#	9466#
9474#	9484#	9491#	9494#	9552#	9561#	9571#	9576#	9581#	9586#	9591#	9607#	9614#	9619#	9628#
9633#	9637#	9642#	9657#	9664#	9671#	9676#	9689#	9741#	9750#	9760#	9765#	9770#	9775#	9780#
9792#	9800#	9805#	9814#	9819#	9823#	9828#	9838#	9845#	9852#	9857#	9872#	9904#	9913#	9923#
9928#	9933#	9938#	9943#	9959#	9963#	9968#	9977#	9982#	9986#	9991#	10000#	10007#	10014#	10019#
10033#	10143#	10152#	10162#	10167#	10172#	10177#	10182#	10205#	10214#	10219#	10228#	10233#	10237#	10242#
10254#	10270#	10284#	10291#	10294#	10366#	10375#	10385#	10390#	10395#	10400#	10405#	10416#	10424#	10429#
10447#	10452#	10455#	10466#	10473#	10483#	10521#	10530#	10540#	10545#	10550#	10555#	10560#	10566#	10570#
10575#	10586#	10591#	10595#	10600#	10609#	10619#	10626#	10629#	10681#	10690#	10700#	10705#	10710#	10715#
10720#	10732#	10736#	10741#	10751#	10756#	10760#	10765#	10773#	10780#	10785#	10789#	10794#	10799#	10808#
10848#	10857#	10867#	10872#	10877#	10882#	10887#	10892#	10896#	10901#	10918#	10923#	10927#	10932#	10941#
10951#	10958#	10961#	11020#	11029#	11039#	11044#	11049#	11054#	11059#	11068#	11075#	11080#	11094#	11099#
11103#	11108#	11116#	11126#	11134#	11142#	11188#	11197#	11207#	11212#	11217#	11222#	11227#	11241#	11245#
11250#	11264#	11269#	11273#	11278#	11289#	11297#	11307#	11314#	11323#	11377#	11386#	11396#	11401#	11406#
11411#	11416#	11428#	11438#	11448#	11453#	11462#	11467#	11471#	11476#	11485#	11495#	11502#	11515#	11525#
11585#	11594#	11604#	11609#	11614#	11619#	11624#	11634#	11647#	11652#	11661#	11666#	11670#	11675#	11687#
11702#	11713#	11720#	11727#	11732#	11745#	11800#	11809#	11819#	11824#	11829#	11834#	11839#	11845#	11858#
11863#	11872#	11877#	11881#	11886#	11895#	11910#	11921#	11928#	11935#	11942#	12036#	12045#	12055#	12060#
12065#	12070#	12075#	12091#	12099#	12104#	12113#	12118#	12122#	12127#	12144#	12155#	12162#	12167#	12176#
12181#	12190#	12196#	12201#	12216#	12227#	12236#	12241#	12248#	12253#	12262#	12267#	12271#	12276#	12295#
12305#	12312#	12315#	12379#	12388#	12398#	12403#	12408#	12413#	12418#	12425#	12432#	12437#	12446#	12451#
12455#	12460#	12469#	12479#	12486#	12492#	12499#	12504#	12513#	12518#	12522#	12527#	12537#	12542#	12550#
12557#	12566#	12626#	12635#	12645#	12650#	12655#	12660#	12665#	12680#	12684#	12689#	12698#	12703#	12707#
12712#	12720#	12725#	12751#	12756#	12763#	12766#	12790#	12796#	12812#	12817#	12825#	12830#	12846#	12851#
12859#	12864#	12890#	12895#	12903#	12908#	12935#	12940#	12947#	12952#	12981#	12988#	12995#	13002#	13009#
13015#														

MSWORD

1#	1416#	1468#	1477	1523#	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534
1535	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549
1550	1551	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564
1565	1566	1567	1568	1569	1570	4718#	4891#	4911#	4922#	5034#	5035	5036	5037	5068#
5069	5070	5071	5085#	5086	5087	5088	5137#	5138	5139	5140	5171#	5172	5173	5174
5188#	5189	5190	5191	5205#	5206	5207	5208	5222#	5223	5224	5225	5239#	5240	5241
5242	5283#	5284	5285	5286	5333#	5334	5335	5336	5395#	5396	5397	5398	5424#	5425
5426	5427	5448#	5449	5450	5451	5479#	5480	5481	5482	5503#	5504	5505	5506	5526#
5529	5530	5531	5547#	5548	5549	5550	5595#	5596	5597	5598	5635#	5636	5637	5638
5651#	5652	5653	5654	5704#	5705	5706	5707	5726#	5727	5728	5729	5751#	5752	5753
5754	5768#	5769	5770	5771	5805#	5806	5807	5808	5822#	5823	5824	5825	5846#	5847
5848	5849	5859#	5860	5861	5862	6242#	6243	6244	6245	6262#	6263	6264	6265	6308#
6309	6310	6311	6328#	6329	6330	6331	6348#	6349	6350	6351	6439#	6440	6441	6442
6459#	6460	6461	6462	6482#	6483	6484	6485	6502#	6503	6504	6505	6543#	6544	6545
6546	6560#	6561	6562	6563	6577#	6578	6579	6580	6669#	6670	6671	6672	6689#	6690
6691	6692	6716#	6717	6718	6719	6736#	6737	6738	6739	6764#	6765	6766	6767	6774#
6775	6776	6777	6843#	6844	6845	6846	6851#	6852	6853	6854	6871#	6872	6873	6874
6892#	6893	6894	6895	6899#	6900	6901	6902	6919#	6920	6921	6922	6926#	6927	6928
6929	6990#	6991	6992	6993	7018#	7019	7020	7021	7034#	7035	7036	7037	7049#	7050
7051	7052	7120#	7121	7122	7123	7130#	7131	7132	7133	7166#	7167	7168	7169	7194#
7195	7196	7197	7206#	7207	7208	7209	7233#	7234	7235	7236	7301#	7302	7303	7304
7311#	7312	7313	7314	7333#	7334	7335	7336	7345#	7346	7347	7348	7366#	7367	7368
7369	7373#	7374	7375	7376	7447#	7448	7449	7450	7457#	7458	7459	7460	7480#	7481
7482	7483	7501#	7502	7503	7504	7510#	7511	7512	7513	7533#	7534	7535	7536	7540#
7541	7542	7543	7552#	7553	7554	7555	7612#	7613	7614	7615	7622#	7623	7624	7625
7650#	7651	7652	7653	7665#	7666	7667	7668	7674#	7675	7676	7677	7691#	7692	7693

67PARAMETER CODING KACY11 30A(1052) 07-APR-83 17:13 PAGE 332  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

7694	7704#	7705	7706	7707	7771#	7772	7773	7774	7781#	7782	7783	7784	7804#	7805
7806	7807	7818#	7819	7820	7821	7827#	7828	7829	7830	7846#	7847	7848	7849	7860#
7861	7862	7863	7921#	7922	7923	7924	7929#	7930	7931	7932	7951#	7952	7953	7954
7965#	7966	7967	7968	7974#	7975	7976	7977	7991#	7992	7993	7994	8059#	8060	8061
8062	8069#	8070	8071	8072	8090#	8091	8092	8093	8104#	8105	8106	8107	8111#	8112
8113	8114	8120#	8121	8122	8123	8133#	8134	8135	8136	8197#	8198	8199	8200	8207#
8208	8209	8210	8228#	8229	8230	8231	8242#	8243	8244	8245	8249#	8250	8251	8252
8258#	8259	8260	8261	8271#	8272	8273	8274	8345#	8346	8347	8348	8355#	8356	8357
8358	8378#	8379	8380	8381	8397#	8398	8399	8400	8406#	8407	8408	8409	8424#	8425
8426	8427	8434#	8435	8436	8437	8507#	8508	8509	8510	8517#	8518	8519	8520	8539#
8540	8541	8542	8558#	8559	8560	8561	8567#	8568	8569	8570	8585#	8586	8587	8588
8595#	8596	8597	8598	8660#	8661	8662	8663	8670#	8671	8672	8673	8701#	8702	8703
8704	8715#	8716	8717	8718	8724#	8725	8726	8727	8742#	8743	8744	8745	8752#	8753
8754	8755	8837#	8838	8839	8840	8847#	8848	8849	8850	8868#	8869	8870	8871	8886#
8887	8888	8889	8895#	8896	8897	8898	8916#	8917	8918	8919	8929#	8930	8931	8932
8938#	8939	8940	8941	8950#	8951	8952	8953	8959#	8960	8961	8962	8966#	8969	8970
8971	8978#	8979	8980	8981	9064#	9065	9066	9067	9074#	9075	9076	9077	9104#	9105
9106	9107	9118#	9119	9120	9121	9127#	9128	9129	9130	9150#	9151	9152	9153	9164#
9165	9166	9167	9233#	9234	9235	9236	9243#	9244	9245	9246	9273#	9274	9275	9276
9287#	9288	9289	9290	9296#	9297	9298	9299	9320#	9321	9322	9323	9334#	9335	9336
9337	9406#	9407	9408	9409	9416#	9417	9418	9419	9437#	9438	9439	9440	9452#	9453
9454	9455	9461#	9462	9463	9464	9474#	9475	9476	9477	9484#	9485	9486	9487	9571#
9572	9573	9574	9581#	9582	9583	9584	9614#	9615	9616	9617	9628#	9629	9630	9631
9637#	9638	9639	9640	9657#	9658	9659	9660	9671#	9672	9673	9674	9760#	9761	9762
9763	9770#	9771	9772	9773	9800#	9801	9802	9803	9814#	9815	9816	9817	9823#	9824
9825	9826	9838#	9839	9840	9841	9852#	9853	9854	9855	9923#	9924	9925	9926	9933#
9934	9935	9936	9963#	9964	9965	9966	9977#	9978	9979	9980	9986#	9987	9988	9989
10000#	10001	10002	10003	10014#	10015	10016	10017	10162#	10163	10164	10165	10172#	10173	10174
10175	10214#	10215	10216	10217	10228#	10229	10230	10231	10237#	10238	10239	10240	10254#	10255
10256	10257	10270#	10271	10272	10273	10284#	10285	10286	10287	10385#	10386	10387	10388	10395#
10396	10397	10398	10424#	10425	10426	10427	10447#	10448	10449	10450	10455#	10456	10457	10458
10466#	10467	10468	10469	10540#	10541	10542	10543	10550#	10551	10552	10553	10570#	10571	10572
10573	10586#	10587	10588	10589	10595#	10596	10597	10598	10609#	10610	10611	10612	10619#	10620
10621	10622	10700#	10701	10702	10703	10710#	10711	10712	10713	10736#	10737	10738	10739	10751#
10752	10753	10754	10760#	10761	10762	10763	10773#	10774	10775	10776	10789#	10790	10791	10792
10867#	10868	10869	10870	10877#	10878	10879	10880	10896#	10897	10898	10899	10918#	10919	10920
10921	10927#	10928	10929	10930	10941#	10942	10943	10944	10951#	10952	10953	10954	11039#	11040
11041	11042	11049#	11050	11051	11052	11075#	11076	11077	11078	11094#	11095	11096	11097	11103#
11104	11105	11106	11116#	11117	11118	11119	11126#	11127	11128	11129	11207#	11208	11209	11210
11217#	11218	11219	11220	11245#	11246	11247	11248	11264#	11265	11266	11267	11273#	11274	11275
11276	11289#	11290	11291	11292	11297#	11298	11299	11300	11307#	11308	11309	11310	11396#	11397
11398	11399	11406#	11407	11408	11409	11448#	11449	11450	11451	11462#	11463	11464	11465	11471#
11472	11473	11474	11485#	11486	11487	11488	11495#	11496	11497	11498	11604#	11605	11606	11607
11614#	11615	11616	11617	11647#	11648	11649	11650	11661#	11662	11663	11664	11670#	11671	11672
11673	11687#	11688	11689	11690	11702#	11703	11704	11705	11713#	11714	11715	11716	11727#	11728
11729	11730	11819#	11820	11821	11822	11829#	11830	11831	11832	11858#	11859	11860	11861	11872#
11873	11874	11875	11881#	11882	11883	11884	11895#	11896	11897	11899	11910#	11911	11912	11913
11921#	11922	11923	11924	11935#	11936	11937	11938	12055#	12056	12057	12058	12065#	12066	12067
12068	12099#	12100	12101	12102	12113#	12114	12115	12116	12122#	12123	12124	12125	12144#	12145
12146	12147	12155#	12156	12157	12158	12176#	12177	12178	12179	12190#	12191	12192	12193	12196#
12197	12198	12199	12216#	12217	12218	12219	12227#	12228	12229	12230	12248#	12249	12250	12251
12262#	12263	12264	12265	12271#	12272	12273	12274	12295#	12296	12297	12298	12305#	12306	12307
12308	12398#	12399	12400	12401	12408#	12409	12410	12411	12432#	12433	12434	12435	12446#	12447
12448	12449	12455#	12456	12457	12458	12469#	12470	12471	12472	12479#	12480	12481	12482	12499#
12500	12501	12502	12513#	12514	12515	12516	12522#	12523	12524	12525	12537#	12538	12539	12540
12550#	12551	12552	12553	12645#	12646	12647	12648	12655#	12656	12657	12658	12684#	12685	12686

67PARAMETER CODING MACY11 30A(1052) 07-APR-83 17:13 PAGE 333  
CZUAAB.MAC 07-APR-83 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

	12687	12698#	12699	12700	12701	12707#	12708	12709	12710	12720#	12721	12722	12723	12751#	12752
	12753	12754	12790#	12812#	12813	12814	12815	12825#	12826	12827	12828	12846#	12847	12848	12849
	12859#	12860	12861	12862	12890#	12891	12892	12893	12903#	12904	12905	12906	12935#	12936	12937
	12938	12947#	12948	12949	12950	13142#	13147#								
MSXFER	1#	1416#													
OPEN	1#	1416#													
POINTE	1#	1416#	1426												
PRINTR	1#	1416#	2916	2929	2945	2952	2966	2979	2992	3009	3015	3028	3045	3058	3064
	3079	3085	3092	3105	3118	3130	3143	3155	3161	3167	3180	3193	3200	3213	3227
	3242	3249	3256	3270	3277	3284	3298	3312	3319	3327	3341	3347	3354	3368	3374
	3381	3396	3404	3414	3430	3442	3448	3457	3464	3482	3489	3495	3501	3508	3515
	3528	3534	3547	3553	3567	3573	3587	3595	3602	3615	3621	3634	3640	3647	3654
	3661	3675	3681	3689	3702	3715	3722	3735	3742	3748	3762	3774	3786	3798	3810
	3816	3823	3837	3843	3857	3864	3871	3878	3885	3899	3913	3921	3929	3937	3950
	3956	3962	3974	12976	12983	12990	12997	13004							
PRINTF	1#	1416#	4385	4538	4553	4574	4640	4655	4668	4683	4696	4795	4881		
PRINTS	1#	1416#													
PRINTX	1#	1416#	4421												
READBU	1#	1416#													
READEF	1#	1416#	4753	4758	4773	4778									
RFLAGS	1#	1416#													
SETPRI	1#	1416#	4232	4246	7007	7180									
SETVEC	1#	1416#	4808	5015	5118	5270	5318	6971	7137						
SLASH	1#	1416#													
STARS	1#	1416#													
SVC	1#	1416#													
XFER	1#	1416#	4718#	4891#	4911#	4922#	12790#								
XFERF	1#	1416#													
XFERT	1#	1416#													

. ABS.	000000	000
	000000	001
UNAREP	053566	002
MICRA	000000	003
MICRB	000000	004
MICRC	000000	005
MICRD	000000	006
MICRE	000000	007
MICRF	000000	010
MICRG	000000	011
NOMORE	000000	012

ERRORS DETECTED: 0  
% DEFAULT GLOBALS GENERATED: 15

CZUAAB.OBJ,CZUAAB.LST/CR/SOL/ML:TOC=SVC34R.P11,CZUAAB.MAC  
RUN-TIME: 72 86 10 SECONDS  
RUN-TIME RATIO: 215/168=1.2  
CORE USED: 31K (61 PAGES)

68SVC.MLB SOURCE FILE MACY11 30A(1052) 07-APR-83 16:48 PAGE 2  
 MICROA.MAC 07-APR-83 16:06

```

1
2
3      .TITLE  MICROA - MICROCODE MODULE A
4      .CSECT  MICRA
5
6      .SBTTL  REGISTER DEFINITIONS USED BY THE T11
7
8      02100C      IPCSR0 =      21000      ;INTERNAL PCSRO ADDRESS
9      021002      DMACSR =      21002      ;DMA ENGINE CONTROL STATUS REGISTER
10     021004      DMATO  =      21004      ;DMA ENGINE TO ADDRESS REGISTER #0
11     021006      DMAT1  =      21006      ;DMA ENGINE TO ADDRESS REGISTER #1
12     021010      MDMA0  =      21010      ;MICROCPU DMA TO ADDRESS REGISTER #0
13     021012      MDMA1  =      21012      ;MICROCPU DMA TO ADDRESS REGISTER #1
14     021014      MDMPRO =      21014      ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15     021016      MDMAR1 =      21016      ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16     021020      IPCSR1 =      21020      ;INTERNAL PCSR1 ADDRESS
17     021022      DMAF   =      21022      ;DMA ENGINE FROM ADDRESS REGISTER
18     021024      DMAWC  =      21024      ;DMA ENGINE WORD COUNT REGISTER
19     021026      MDMAW0 =      21026      ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20     021030      LTAC   =      21030      ;LINK TRANSMIT ADDRESS COUNTER REGISTER
21     021032      LFRBUF =      21032      ;LINK RECIEVE BUFFER ADDRESS FIFO
22     021034      CLRFIG =      21034      ;CLEAR FIFO
23     021036      MDMAW1 =      21036      ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24     021040      PCSRSW =      21040      ;SWITCH PACK REGISTER
25     021042      MDMSR  =      21042      ;MICROCPU DMA STATUS REGISTER
26     021044      LRBUF  =      21044      ;LINK RECIEVE BUFFER COMPLETED
27     021060      PHYAD0 =      21060      ;PHYSICAL ADDRESS ROM BYTE 0
28     021062      PHYAD1 =      21062      ;PHYSICAL ADDRESS ROM BYTE 1
29     021064      PHYAD2 =      21064      ;PHYSICAL ADDRESS ROM BYTE 2
30     021066      PHYAD3 =      21066      ;PHYSICAL ADDRESS ROM BYTE 3
31     021070      PHYAD4 =      21070      ;PHYSICAL ADDRESS ROM BYTE 4
32     021072      PHYAD5 =      21072      ;PHYSICAL ADDRESS ROM BYTE 5
33
34     .SBTTL  OTHER DEFINITIONS USED BY THE MICROCODE
35
36     100000      BIT15  =      100000
37     040000      BIT14  =      40000
38     020000      BIT13  =      20000
39     010000      BIT12  =      10000
40     004000      BIT11  =      4000
41     002000      BIT10  =      2000
42     001000      BIT9   =      1000
43     000400      BIT8   =      400
44     000200      BIT7   =      200
45     000100      BIT6   =      100
46     000040      BIT5   =      40
47     000020      BIT4   =      20
48     000010      BIT3   =      10
49     000004      BIT2   =      4
50     000002      BIT1   =      2
51     000001      BIT0   =      1
52     ;
53     012400      LASFTP =      BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
54     000340      PRI07  =      340
55     000300      PRI06  =      300
56     000240      PRI05  =      240

```

76MICROA - MICROCODE MODULE A  
MICROA.MAC 07-APR-83 16:06

MACY11 3GA(1052) 07-APR-83 16:48 PAGE 3  
OTHER DEFINITIONS USED BY THE MICROCODE

57	000200	PRI04 =	200	
58	000140	PRI03 =	140	
59	000100	PRI02 =	100	
60	000040	PRI01 =	40	
61	000000	PRI00 =	0	
62		:		
63		:PCSR0 - PORT CONTROL STATUS REGISTER 0		
64		:		
65	100000	SERI =	BIT15	
66	040000	PCEI =	BIT14	
67	020000	RXI =	BIT13	
68	010000	TXI =	BIT12	
69	004000	DNI =	BIT11	
70	002000	RCEI =	BIT10	
71	000400	FATI =	BIT8	
72		:		
73	000134	SANVEC=	134	:VECTOR ADDRESS FOR THE SANITY TIMER
74	000150	NXMVEC=	150	:VECTOR ADDRESS FOR THE NON-EXISTANT MEMORY TIMEOUT
75	000064	CSRVEC=	64	:VECTOR ADDRESS FOR CSR WRITE INTERRUPT
76	000114	DMAVEC=	114	:VECTOR ADDRESS FOR DMA DONE INTERRUPT
77	000140	PARVEC=	140	:VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
78	000150	NXMVEC=	150	:VECTOR ADDRESS FOR NON-EXISTANT UNIBUS ADDRESS
79	001000	STACK=	1000	:STACK LOCATION
80	000001	INMON=	1	:IN MICROMONITOR STATE *** FOR ASSEMBLY WITH MACY11
81	000002	INTST=	2	:IN A TEST STATE *** THESE THREE VARIABLES MUST
82	000003	INERR=	3	:IN ERROR STATE *** BE CHANGED TO LOCAL(I.E.=)
83	000001	CSRFLG=	BIT0	:CSR WRITE INTERRUPT OCCURED
84	000002	ERRFLG=	BIT1	:UNEXPECTED ERROR OCCURED
85	000004	PARFLG=	BIT2	:PARITY ERROR OCCURED
86	000010	NXMFLG=	BIT3	:NON-EXISTANT MEMORY ERROR OCCURED
87	020000	SIZ4K=	20000	:4K WORDS
88	040000	SIZ8K=	SIZ4K*2	:8K WORDS
89	020000	WCSSIZ=	SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
90	020000	IOSIZ=	SIZ4K	:SIZE OF I/O PAGE
91	040000	ROMSIZ=	SIZ8K	:SIZE OF ROM
92	077774	LINSIZ=	SIZ8K*2-4	:SIZE OF LINK MEMORY
93	000000	WCSADR=	0	:BASE ADDRESS OF WCS
94	020000	IQADR=	WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
95	040000	ROMADR=	IQADR+IOSIZ	:BASE ADDRESS OF ROM
96	100000	LINADR=	ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
97	000000	DATERR=	0	:FLAG INDICATING DATA ERROR OCCURRED
98	000001	ADRERR=	1	:FLAG INDICATING ADDRESS ERROR OCCURRED
99	000002	PARERR=	2	:FLAG INDICATING PARITY ERROR OCCURRED
100	177774	MODREG=	LINADR+LINSIZ	:LINK MODE REGISTER
101	177774	ADDRG=	MODREG	:LINK STATION ADDRESS RAM REGISTER
102	177776	CMDRG=	MODREG+2	:LINK COMMAND REGISTER
103				



76MICROA - MICROCODE MODULE A  
MICROA.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:48 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122

.SBTTL A\_MODULE MICROCODE

:\*\*\*\*\*  
:  
:THIS MODULE CONTAINS MICROCODE THAT IS USED FOR THE LOAD AND START  
:FUNCTION TEST  
:  
:\*\*\*\*\*

000000' 106427 000340  
000004' 012737 012402 021020  
000012' 012737 004000 021000  
000020' 000777  
000022' 000024  
000001

MICROA::MTPS #PRI07 ;DISABLE INTERRUPTS  
MOV #LASFTP!INTST,@#IPCSR1 ;SET TEST PATTERN AND IN TEST STATE BITS  
MOV #DNI,@#IPCSRO ;SET DONE BIT  
BR . ;HANG HERE UNTIL HOST RESETS US  
  
MICASZ::MICASZ-MICROA+2 ;SIZE OF MICROCODE MODULE A  
.END

76MICROA - MICROCODE MODULE A  
MICROA.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:48 PAGE 6  
CROSS REFERENCE TABLE -- USER SYMBOLS

ADDREG=	177774	101#		
ADRERR=	000001	98#		
BIT0 =	000001	51#	83	
BIT1 =	000002	50#	84	
BIT10 =	002000	41#	53	70
BIT11 =	004000	40#	69	
BIT12 =	010000	39#	53	68
BIT13 =	020000	38#	67	
BIT14 =	040000	37#	66	
BIT15 =	100000	36#	65	
BIT2 =	000004	49#	85	
BIT3 =	000010	48#	86	
BIT4 =	000020	47#		
BIT5 =	000040	46#		
BIT6 =	000100	45#		
BIT7 =	000200	44#		
BIT8 =	000400	43#	53	71
BIT9 =	001000	42#		
CLRFIF=	021034	22#		
CMREG=	177776	102#		
CSRFLG=	000001	83#		
CSRVEC=	000064	75#		
DATERR=	000000	97#		
DMACSR=	021002	9#		
DMAF =	021022	17#		
DMATO =	021004	10#		
DMAT1 =	021006	11#		
DMAVEC=	000114	76#		
DMAVC =	021024	18#		
DNI =	004000	69#	118	
ERRFLG=	000002	84#		
FATI =	000400	71#		
INERR =	000003	82#		
INRON =	000001	80#		
INTST =	000002	81#	117	
IQADR =	020000	94#	95	
IOSIZ =	020000	90#	95	
IPCSRO=	021000	8#	118*	
IPCSR1=	021020	16#	117*	
LASFTP=	012400	53#	117	
LFRBUF=	021032	21#		
LINADR=	100000	96#	100	
LINSIZ=	077774	92#	100	
LRFUF =	021044	26#		
LTAC =	021030	20#		
MDMARD=	021014	14#		
MDMAR1=	021016	15#		
MDMARD=	021026	19#		
MDMAR1=	021036	23#		
MDMARD =	021010	12#		
MDMAR1 =	021012	13#		
MDMSR =	021042	25#		
MICASZ	000022RG	002	121#	
MICROA	000000RG	002	116#	121
MODREG=	177774	100#	101	102
MODWLG=	000010	86#		

76MICROA - MICROCODE MODULE A  
 MICROA.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:48 PAGE 7  
 CROSS REFERENCE TABLE -- USER SYMBOLS

NXWVEC= 000150	74#	78#		
PARERR= 000002	99#			
PARFLG= 000004	85#			
PARVEC= 000140	77#			
PCEI = 040000	66#			
PCSRW= 021040	24#			
PHYAD0= 021060	27#			
PHYAD1= 021062	28#			
PHYAD2= 021064	29#			
PHYAD3= 021066	30#			
PHYAD4= 021070	31#			
PHYAD5= 021072	32#			
PRI00 = 000000	61#			
PRI01 = 000040	60#			
PRI02 = 000100	59#			
PRI03 = 000140	58#			
PRI04 = 000200	57#			
PRI05 = 000240	56#			
PRI06 = 000300	55#			
PRI07 = 000340	54#	116*		
RCEI = 002000	70#			
ROMADR= 040000	95#	96		
ROMSIZ= 040000	91#	96		
RXI = 020000	67#			
SANVEC= 000134	73#			
SERI = 100000	65#			
SIZ4K = 020000	87#	88	89	90
SIZBK = 040000	88#	91	92	
STACK = 001000	79#			
TXI = 010000	68#			
WCSADR= 000000	93#	94		
WCSSIZ= 020000	89#	94		
. = 000024R	002	119		

76MICROA - MICROCODE MODULE A  
MICROA.MAC 07-APR-83 16:06MACY11 30A(1052) 07-APR-83 16:48 PAGE 9  
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 332

BCOMPL	10
BERROR	10
BGNAU	10
BGNAUT	10
BCACLN	10
BGNDU	10
BGNHRD	10
BGNHW	10
BGNINI	10
BGNPOD	10
BGNPSG	10
BGNPRO	10
BGNPTA	10
BGNRPT	10
BGNSEG	10
BGNSET	10
BGNSFT	10
BGNSRV	10
BGNSUB	10
BGNSU	10
BGNTST	10
BNCORP	10
BNERRO	10
BREAK	10
BRESET	10
CKLOOP	10
CLOCK	10
CLOSE	10
CLVEC	10
COMEN	10
DELAY	10
DESCR1	10
DEVTYP	10
DISPAT	10
DISPLA	10
DOCLN	10
DODU	10
DORPT	10
ENDAU	10
ENDAUT	10
ENDCLN	10
ENDCOR	10
ENDDU	10
ENDHRD	10
ENDHW	10
ENDINI	10
ENDPOD	10
ENDPSG	10
ENDPRO	10
ENDPTA	10
ENDRPT	10
ENDSEG	10
ENDSET	10
ENDSFT	10
ENDSRV	10
ENDSUB	10

77MICROA - MICROCODE MODULE A  
MICROA.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:48 PAGE 10  
CROSS REFERENCE TABLE -- MACRO NAMES

ENDSY	10
ENDTST	10
EQUALS	10
ERRDF	10
ERRHRD	10
ERROR	10
ERRSF	10
ERRSOF	10
ERRTBL	10
ESCAPE	10
EXIT	10
FEQUAL	10
GETBYT	10
GETPRI	10
GETWOR	10
GMANIA	10
GMANID	10
GMANIL	10
GPHARD	10
GPRMA	10
GPRMD	10
GPRML	10
HEADER	10
INLOOP	10
IOSETU	10
IOSTAR	10
KT11	10
LASTAD	10
MANUAL	10
MEMORY	10
MSBYTE	10
MSCHEC	10
MSCNTO	10
MSCOUN	10
MSDATA	10
MSDECR	10
MSDEFA	10
MSENDE	10
MSERRI	10
MSESCA	10
MSESCS	10
MSXCP	10
MSEXIT	10
MSXSE	10
MSXTJ	10
MSGEN	10
MSGENB	10
MSGETS	10
MSGETT	10
MSGNGB	10
MSGNIN	10
MSGNLS	10
MSGNSU	10
MSGNTA	10
MSGNTE	10
MSHAPT	10

77MICROA - MICROCODE MODULE A MACY11 30A(1052) 07-APR-83 16:48 PAGE 11  
 MICROA.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

MSHMAP 1#  
 MSINCR 1#  
 MSIOSE 1#  
 MSLDRO 1#  
 MSMASK 1#  
 MSACHI 1#  
 MSACLO 1#  
 MSASK1 1#  
 MSPOP 1#  
 MSPRIN 1#  
 MSPUSH 1#  
 MSPUT 1#  
 MSPUT1 1#  
 MSRADI 1#  
 MSRBRO 1#  
 MSRNRO 1#  
 MSSETS 1#  
 MSSTAR 1#  
 MSSVC 1#  
 MSTLAB 1#  
 MSTSTL 1#  
 MSWORD 1#  
 MSXFER 1#  
 OPEN 1#  
 POINTE 1#  
 PRINTB 1#  
 PRINTF 1#  
 PRINTS 1#  
 PRINTX 1#  
 READBU 1#  
 READEF 1#  
 RFLAGS 1#  
 SETPRI 1#  
 SETVEC 1#  
 SLASH 1#  
 STARS 1#  
 SVC 1#  
 XFER 1#  
 XFERT 1#  
 XFERT 1#

. ABS. 000000 000  
 000000 001  
 MICRA 000024 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MICROA.OBJ,MICROA.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROA.MAC  
 RUN-TIME: 2 1 .2 SECONDS  
 RUN-TIME RATIO: 39/4=8.8  
 CORE USED: 31K (61 PAGES)

68SVC.MLB SOURCE FILE MACY11 50A(1052) 07-APR-83 16:49 PAGE 2  
 MICROB.MAC 07-APR-83 16:06

```

1
2          .TITLE  MICROB - MICROCODE MODULE B
3
4          000000'      .CSECT  MICRB
5
6          .SBTTL  REGISTER DEFINITIONS USED BY THE T11
7
8          021000      IPCSRO  =      21000      ;INTERNAL PCSRO ADDRESS
9          021002      DMACSR  =      21002      ;DMA ENGINE CONTROL STATUS REGISTER
10         021004      DNATO   =      21004      ;DMA ENGINE TO ADDRESS REGISTER #0
11         021006      DMAT1   =      21006      ;DMA ENGINE TO ADDRESS REGISTER #1
12         021010      MDMA0   =      21010      ;MICROCPU DMA TO ADDRESS REGISTER #0
13         021012      MDMA1   =      21012      ;MICROCPU DMA TO ADDRESS REGISTER #1
14         021014      MDMARO  =      21014      ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15         021016      MDMAR1  =      21016      ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16         021020      IPCSR1  =      21020      ;INTERNAL PCSR1 ADDRESS
17         021022      DMAF    =      21022      ;DMA ENGINE FROM ADDRESS REGISTER
18         021024      DMAWC   =      21024      ;DMA ENGINE WORD COUNT REGISTER
19         021026      MDMAWO  =      21026      ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20         021030      LTAC    =      21030      ;LINK TRANSMIT ADDRESS COUNTER REGISTER
21         021032      LFRBUF  =      21032      ;LINK RECIEVE BUFFER ADDRESS FIFO
22         021034      CLRFIF  =      21034      ;CLEAR FIFO
23         021036      MDMAW1  =      21036      ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24         021040      PCSRSW  =      21040      ;SWITCH PACK REGISTER
25         021042      MDMSR   =      21042      ;MICROCPU DMA STATUS REGISTER
26         021044      LRBUF   =      21044      ;LINK RECIEVE BUFFER COMPLETED
27         021060      PHYAD0  =      21060      ;PHYSICAL ADDRESS ROM BYTE 0
28         021062      PHYAD1  =      21062      ;PHYSICAL ADDRESS ROM BYTE 1
29         021064      PHYAD2  =      21064      ;PHYSICAL ADDRESS ROM BYTE 2
30         021066      PHYAD3  =      21066      ;PHYSICAL ADDRESS ROM BYTE 3
31         021070      PHYAD4  =      21070      ;PHYSICAL ADDRESS ROM BYTE 4
32         021072      PHYAD5  =      21072      ;PHYSICAL ADDRESS ROM BYTE 5
33
34         .SBTTL  OTHER DEFINITIONS USED BY THE MICROCODE
35
36         100000      BIT15   =      100000
37         040000      BIT14   =      40000
38         020000      BIT13   =      20000
39         010000      BIT12   =      10000
40         004000      BIT11   =      4000
41         002000      BIT10   =      2000
42         001000      BIT9    =      1000
43         000400      BIT8    =      400
44         000200      BIT7    =      200
45         000100      BIT6    =      100
46         000040      BIT5    =      40
47         000020      BIT4    =      20
48         000010      BIT3    =      10
49         000004      BIT2    =      4
50         000002      BIT1    =      2
51         000001      BIT0    =      1
52         ;
53         012400      LASFTP  =      BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
54         000340      PRI07   =      340
55         000300      PRI06   =      300
56         000240      PRI05   =      240

```

76MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 3  
OTHER DEFINITIONS USED BY THE MICROCODE

57	000200	PRI04 =	200	
58	000140	PRI03 =	140	
59	000100	PRI02 =	100	
60	000040	PRI01 =	40	
61	000000	PRI00 =	0	
62		:		
63		:PCSR0 - PORT CONTROL STATUS REGISTER 0		
64		:		
65	100000	SERI =	BIT15	
66	040000	PCEI =	BIT14	
67	020000	RXI =	BIT13	
68	010000	TXI =	BIT12	
69	004000	DNI =	BIT11	
70	002000	RCEI =	BIT10	
71	000400	FATI =	BIT8	
72		:		
73	000134	SANVEC=	134	:VECTOR ADDRESS FOR THE SANITY TIMER
74	000064	CSRVEC=	64	:VECTOR ADDRESS FOR CSR WRITE INTERRUPT
75	000114	DMAVEC=	114	:VECTOR ADDRESS FOR DMA DONE INTERRUPT
76	000140	PARVEC=	140	:VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
77	001000	STACK=	1000	:STACK LOCATION
78	000001	INMON=	1	:IN MICROMONITOR STATE
79	000002	INTST=	2	:IN A TEST STATE
80	000003	INERR=	3	:IN ERROR STATE
81	000001	CSRFLG=	BIT0	:CSR WRITE INTERRUPT OCCURED
82	000002	ERRFLG=	BIT1	:UNEXPECTED ERROR OCCURED
83	000004	PARFLG=	BIT2	:PARITY ERROR OCCURED
84	000010	NXMFLG=	BIT3	:NON-EXISTANT MEMORY ERROR OCCURRED
85	000020	NPRFLG=	BIT4	:NPR TIMEOUT OCCURRED
86	100000	NPRERR=	BIT15	:PCSR0 FLAG INDICATING NPR ERROR OCCURRED
87	040000	NXMERR=	BIT14	:PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
88	020000	UNIERR=	BIT13	:PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
89	010000	PARERR=	BIT12	:PCSR0 FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
90	020000	SIZ4K=	20000	:4K WORDS
91	040000	SIZ8K=	SIZ4K*2	:8K WORDS
92	020000	WCSSIZ=	SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
93	020000	IOSIZ=	SIZ4K	:SIZE OF I/O PAGE
94	040000	ROMSIZ=	SIZ8K	:SIZE OF ROM
95	077774	LINSIZ=	SIZ8K*2-4	:SIZE OF LINK MEMORY
96	000000	WCSADR=	0	:BASE ADDRESS OF WCS
97	020000	IOADR=	WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
98	040000	ROMADR=	IOADR+IOSIZ	:BASE ADDRESS OF ROM
99	100000	LINADR=	ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
100	000000	DATERR=	0	:FLAG INDICATING DATA ERROR OCCURRED
101	000001	ADRERR=	1	:FLAG INDICATING ADDRESS ERROR OCCURRED
102	177774	MODREG=	LINADR+LINSIZ	:LINK MODE REGISTER
103	177774	ADDRREG=	MODREG	:LINK STATION ADDRESS RAM REGISTER
104	177776	CMDREG=	MODREG+2	:LINK COMMAND REGISTER
105				



76MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161

.SBTTL B\_MODULE MICROCODE

\*\*\*\*\*  
:THIS MODULE CONTAINS A MICROMONITOR AND MICROCODE FOR:  
: 1-WCS MEMORY TEST  
\*\*\*\*\*

MICROB::MTPS #PRI07 ;DISABLE INTERRUPTS  
MOV #0,#CMDREG ;TURN OFF THE LINK  
MOV #STACK,SP ;SETUP STACK  
MOV #INMON,PCSR1 ;TELL HOST WE ARE IN MICROMONITOR  
MOV PCSR1,#IPCSR1  
MOV #DNI,#IPCSRO ;TELL HOST THE LOAD AND START FINISHED  
MOV PC,R0 ;GET ADDRESS OF UNEXPECTED ERROR...  
ADD #ERRINT-.,R0 ;HANDLER  
CLR R1 ;FILL ALL UNUSED VECTORS WITH TRAP...  
10S: MOV R0,(R1)+ ;HANDLER  
MOV #PRI07,(R1)+  
CMP R1,#1000  
BLT 10S  
  
MOV PC,R0 ;SETUP PARITY TRAP VECTOR  
ADD #PARINT-.,R0  
MOV R0,#PARVEC  
MOV #PRI07,#PARVEC+2  
  
MOV PC,R0 ;SETUP DMA INTERRUPT VECTOR  
ADD #DMAINT-.,R0  
MOV R0,#DMAVEC  
MOV #PRI07,#DMAVEC+2  
  
MOV PC,R0 ;SETUP CSR WRITE VECTOR  
ADD #CSRWRT-.,R0  
MOV R0,#CSRVEC  
MOV #PRI04,#CSRVEC+2  
  
MOV PC,R0 ;SETUP SANITY TIMER VECTOR  
ADD #TIMINT-.,R0  
MOV R0,#SANVEC  
MOV #PRI05,#SANVEC+2  
  
MOV #PCSRW,R0 ;GET SWITCH PACK BITS  
BIS #176000,R0 ;MAP THEM INTO HOST I/O PAGE  
ASL R0 ;SHIFT OVER TO POSITION CORRECTLY  
ASL R0  
ASL R0  
ADD #4,R0 ;PCSR2 IS PCSRO+4  
MOV R0,IPCSR2 ;SAVE PCSR2 ADDRESS  
MOV #3,IPCSR2+2 ;HIGH ORDER BITS 17:16  
CLR FLG2 ;INITIALIZE FLAG WORD  
15S: MTPS #PRI00 ;ALLOW INTERRUPTS

76MICROB - MICROCODE MODULE B MACY11 30A(1052) 07-APR-83 16:49 PAGE 5  
 MICROB.MAC 07-APR-83 16:06 B\_MODULE MICROCODE

162										
163	000230'	005767	000274		20\$:	TST	FLG2			:WAIT FOR A COMMAND FROM HOST
164	000234'	001775				BEQ	20\$			
165										
166	000236'	106427	000340			MTPS	#PRI07			:RAISE CPU PRIORITY TO SERVICE COMMAND
167	000242'	032767	000001	000260		BIT	#CSRFLG,FLG2			:DID HOST GIVE US A COMMAND?
168	000250'	001001				BNE	30\$			:YES
169	000252'	000777				BR	.			:NO, ERROR SO JUST SIT HERE...
170										:FOR LACK OF ANYTHING BETTER TO DO
171										
172	000254'	013700	021000		30\$:	MOV	@#IPCSRO,R0			:GET WHAT HOST WROTE TO PCSRO
173	000260'	042700	177760			BIC	#177760,R0			:STRIP ALL BUT COMMAND BITS
174	000264'	001004				BNE	35\$			:WAS IT THE CLEAR FUNCTION?
175	000266'	012737	000001	021020		MOV	#INMON,@#IPCSR1			:YES, CLEAR SELF TEST BITS
176	000274'	000432				BR	50\$			
177	000276'	022700	000017		35\$:	CMP	#17,R0			:RETURN TO OPERATIONAL MICROCODE?
178	000302'	001432				BEQ	60\$			:YES
179	000304'	162700	000001			SUB	#1,R0			:WILL FORM A TABLE INDEX WITH THIS
180	000310'	010701				MOV	PC,R1			:GET ADDRESS OF OUR COMMAND TABLE
181	000312'	062701	000214			ADD	#TBLB-.,R1			
182	000316'	006300				ASL	R0			:MAKE COMMAND A BYTE OFFSET
183	000320'	060001				ADD	R0,R1			:USE IT TO INDEX INTO COMMAND TABLE
184	000322'	061101				ADD	(R1),R1			:R1 NOW HAS COMMAND ROUTINE ADDRESS
185	000324'	004711				JSR	PC,(R1)			:EXECUTE AS COMMANDED FROM HOST
186	000326'	103404				BCS	40\$			:ERROR OCCURRED
187	000330'	112767	000001	000176		MOVB	#INMON,PCSR1			:INDICATE TO HOST WE ARE BACK IN...
188	000336'	000403				BR	45\$			:MICROMONITR
189	000340'	112767	000003	000166	40\$:	MOVB	#INERR,PCSR1			:INDICATE TO HOST ERROR OCCURRED
190	000346'	016737	000162	021020	45\$:	MOV	PCSR1,@#IPCSR1			
191	000354'	012737	004000	021000		MOV	#DNI,@#IPCSRO			:TELL HOST THIS MICROTEST FINISHED
192	000362'	005067	000142		50\$:	CLR	FLG2			:RESET FLAG WORD
193	000366'	000716				BR	15\$			:GO WAIT FOR ANOTHER COMMAND
194										
195	000370'	005000			60\$:	CLR	R0			:FAKE SUCCESSFULL SELF TEST DONE
196	000372'	000137	040006			JMP	@#40006			:START OPERATIONAL MICROCODE
197										
198	000376'	052767	000001	000124	CSRWRT:	BIS	#CSRFLG,FLG2			:INDICATE A CSR WRITE INTERRUPT OCCURED
199	000404'	000002				RTI				
200										
201	000406'	052767	000002	000114	ERRINT:	BIS	#ERRFLG,FLG2			:INDICATE A UNEXPECTED INTERRUPT OCCURED
202	000414'	012737	020000	021000		MOV	#UNIERR,@#IPCSRO			:TELL HOST AN UNEXPECTED INTERRUPT
203										:HAPPENED
204	000422'	000777				BR	.			:JUST SIT HERE AND SPIN WHEELS
205										
206	000424'	005267	000102		TIMINT:	INC	SANTIM			:COUNT TICKS AS THEY OCCUR
207	000430'	000002				RTI				
208										
209	000432'	013767	021002	000106	DMAINT:	MOV	@#DMACSR,DMDONE			:GET DMA STATUS
210	000440'	032767	040000	000100		BIT	#BIT14,DMDONE			:DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
211	000446'	001404				BEQ	10\$			:NO
212	000450'	012737	040000	021000		MOV	#NXMERR,@#IPCSRO			:YES, TELL HOST A NON-EXISTANT MEMORY
213										:LOCATION WAS ADDRESSED
214	000456'	000407				BR	20\$			
215	000460'	032767	100000	000060	10\$:	BIT	#BIT15,DMDONE			:DID A NPR TIMEOUT OCCUR?
216	000466'	001407				BEQ	30\$			:NO
217	000470'	012737	100000	021000		MOV	#NPRERR,@#IPCSRO			:TELL HOST NPR TIMEOUT HAPPENED



76MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 7  
B\_MODULE MICROCODE

```

237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256 000552' 112767 000002 177754 MICB1: MOVB #INTST,PCSR1 ;TELL HOST WE ARE TESTING
257 000560' 016737 177750 021020 MOV PCSR1,&#IPCSR1
258 000566' 005067 000320 CLR BOTH ;START TESTING BOTTOM HALF FIRST
259 000572' 012767 001000 000304 MOV #1000,LOWLIM ;SET LOW LIMIT JUST ABOVE VECTORS
260 000600' 012767 010000 000300 MOV #WCSSIZ/2,HILIM ;HIGH LIMIT IS HALF WAY INTO WCS
261
262 000606' 010700 58: MOV PC,R0 ;GET ADDRESS OF SUBTEST LIST
263 000610' 062700 000256 ADD #B1STBL-.,R0
264
265 000614' 005067 177710 108: CLR FLG2 ;CLEAR THE FLAG WORD
266 000620' 011001 MOV (R0),R1 ;GET OFFSET FROM ENTRY TO SUBTEST
267 000622' 001437 BEQ 308 ;NO MORE ENTRIES
268 000624' 060001 ADD R0,R1 ;CALC SUBTEST ADDRESS
269 000626' 004711 JSR PC,(R1) ;GO EXECUTE SUBTEST
270 000630' 103403 BCS 208 ;ERROR OCCURRED IN SUBTEST
271 000632' 062700 000002 ADD #2,R0 ;POINT TO NEXT SUBTEST IN LIST
272 000636' 000766 BR 108
273
274 000640' 016737 177672 021010 208: MOV IPCSR2,&#NDMA0 ;GET CONTENTS OF HOST'S PCRS2 AND PCRS3
275 000646' 016737 177666 021012 MOV IPCSR2+2,&#NDMA1
276 000654' 013700 021014 MOV &#NDMA0,R0 ;R0=CONTENTS OF PCRS2
277 000660' 013702 021014 MOV &#NDMA0,R2 ;R2=CONTENTS OF PCRS3
278 000664' 010037 021010 MOV R0,&#NDMA0 ;POINT TO PCBB+0
279 000670' 010237 021012 MOV R2,&#NDMA1
280 000674' 016737 000210 021026 MOV SUBNUM,&#NDMA0 ;LOAD PCBB+0 WITH SUBTEST #...
281 ;AND PCBB+1 WITH ERROR TYPE
282 000702' 010137 021026 MOV R1,&#NDMA0 ;LOAD PCBB+2 WITH FAILING ADDRESS
283 000706' 010337 021026 MOV R3,&#NDMA0 ;LOAD PCBB+4 WITH GOOD DATA
284 000712' 010437 021026 MOV R4,&#NDMA0 ;LOAD PCBB+6 WITH BAD DATA
285 000716' 000261 SEC ;INDICATE THIS MICROTEST FAILED
286 000720' 000434 BR 458
287
288 000722' 005767 000164 308: TST BOTH ;HAVE WE DONE BOTH HALVES YET?
289 000726' 001030 BNE 408 ;YES
290 000730' 005267 000156 INC BOTH ;NO, WE ARE ABOUT TO DO OTHER HALF
291 000734' 012767 010000 000142 MOV #WCSSIZ/2,LOWLIM ;SET LOW LIMIT AT BOTTOM OF UPPER HALF
292 000742' 012767 020000 000136 MOV #WCSSIZ,HILIM ;HIGH LIMIT IS TOP OF UPPER HALF

```



76MICRMS - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 9  
MODULE B, MICROTEST #1, MICROSUBTEST A

334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380

.SBTTL MODULE B, MICROTEST #1, MICROSUBTEST A

```

:*****
:
:THIS IS AN ACCESS TEST OF WCS MEMORY. IT WRITES ONES
:IN MEMORY BETWEEN LOWLIM AND HILIM AND VERIFIES SAME. IT THEN WRITES
:ZEROS AND VERIFIES SAME. IT ALSO CHECKS FOR BOGUS PARITY ERRORS.
:
:*****
    
```

```

MICB1A: MOV     R0,-(SP)           ;SAVE R0
        MOVB    #1,SUBNUM        ;TELL WE ARE IN SUBTEST A
        MTPS    #PRI06          ;ALLOW PARITY ERRORS
        MOV     #177777,R3       ;GOOD DATA = ALL ONES
5$:     MOV     LOWLIM,R0        ;GET BASE ADDRESS OF MEMORY
10$:    MOV     R3,(R0)+         ;WRITE ALL OVER MEMORY
        CMP     R0,HILIM
        BLO    10$

        MOV     LOWLIM,R1       ;POINT BACK TO BASE
12$:    MOV     (R1),R4          ;READ DATA
        BIT     #PARFLG,FLG2    ;DID A PARITY ERROR OCCUR?
        BEQ    15$             ;NO
        CMP     R3,R4           ;YES, WAS DATA READ GOOD?
        BNE    20$             ;NO, DATA ERROR
        MOVB    #PARERR,ERRTYP  ;TELL MICROMONITOR TYPE OF ERROR
        SEC    BR               ;TELL MICROMONITOR ERROR OCCURRED
        BR     40$             ;LEAVE

15$:    CMP     R3,R4           ;WAS DATA GOOD?
        BEQ    30$             ;YES
20$:    MOVB    #DATERR,ERRTYP  ;TELL MICROMONITOR TYPE OF ERROR
        SEC    BR               ;TELL MICROMONITOR ERROR OCCURRED
        BR     40$

30$:    ADD     #2,R1           ;POINT TO NEXT MEMORY ADDRESS
        CMP     R1,HILIM       ;DONE ALL MEMORY?
        BNE    12$            ;NOT YET
        TST    R3              ;DONE BOTH DATA TYPES?
        BEQ    35$             ;YES
        COM    R3              ;NO, WRITE ONES NOW
        BR     5$

35$:    CLC                    ;INDICATE SUCCESS
40$:    MOV     (SP)+,R0        ;RESTORE R0
        RTS     PC
    
```

77MICROB - MICROCODE MODULE B  
 MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 10  
 MODULE B, MICROTEST #1, MICROSUBTEST B

.SBTTL MODULE B, MICROTEST #1, MICROSUBTEST B

381  
 382  
 383  
 384  
 385  
 386  
 387  
 388  
 389  
 390  
 391  
 392  
 393  
 394  
 395  
 396  
 397  
 398  
 399  
 400  
 401  
 402  
 403  
 404  
 405  
 406  
 407  
 408  
 409  
 410  
 411  
 412  
 413  
 414  
 415  
 416  
 417  
 418  
 419  
 420  
 421  
 422  
 423  
 424  
 425  
 426  
 427  
 428  
 429  
 430  
 431  
 432  
 433  
 434  
 435

```

:*****
:ADDRESS SHIFT TEST
:
:THIS TEST ASSUMES ALL MEMORY BETWEEN LOWLIM AND HILIM IS ZEROS
:IT CHECKS FOR PROPER BANK SELECTION BY WRITING 1'S IN A LOCATION
:AND CHECKING FOR 1'S IN THE SAME LOCATION OF OTHER 4K BANKS (ADDRESS ERROR).
:IT ALSO CHECKS FOR DATA ERRORS I.E. NON-ZERO DATA IN LOCATIONS NOT
:WRITTEN WITH 1'S AND MAKES SURE LOCATIONS WRITTEN WITH 1'S HAVE 1'S IN THEM
:*****
    
```

```

MICB1B: MOV    RO,-(SP)           ;SAVE RO
          MOVB   #2,SUBNUM       ;TELL HIM WE ARE IN THE SECOND SUBTEST
10S:     MOV    LOWLIM,RO        ;SET 'WRITTEN TO' ADDRESS
          MOV    LOWLIM,R1       ;SET 'READ FROM' ADDRESS
          MOV    #177777,(R0)    ;WRITE DATA INTO LOCATION AT BASE
20S:     MOV    (R1),R4         ;READ DATA FROM MEMORY WHICH IS AN EVEN
          ;INCREMENT AWAY
          CMP    RO,R1          ;IS 'READ FROM AND 'WRITTEN TO'
          ;ADDRESSES THE SAME?
          BEQ   70S             ;YES, SO DATA IN BOTH SHOULD BE THE SAME
          CLR   R3              ;GOOD DATA IS ZEROS
          TST  R4              ;DATA IN 'READ FROM' MUST BE 0'S
          BEQ   50S             ;OK GO CHANGE 'READ FROM' ADDRESS
          ;ERROR OCCURRED BUT WE DON'T KNOW IF...
          ;IT IS A DATA ERROR OR AN ADDRESS ERROR
          CMP   #177777,R4      ;WAS DATA READ ALL 1'S?
          BNE  40S             ;NO, SO IT WAS A DATA ERROR
          MOVB  #ADRERR,ERRTYP  ;YES, IT WAS AN ADDRESS ERROR
          SEC   ;INDICATE FAILURE
          BR    60S            ;LEAVE THIS SUBTEST
70S:     CMP   #177777,R4      ;DATA READ MUST BE ALL 1'S
          BEQ   50S            ;IT IS GOOD
          MOV   #177777,R3      ;GOOD DATA IS ONES
          MOVB  #DATERR,ERRTYP  ;IT WAS A DATA ERROR
          SEC   ;INDICATE FAILURE
          BR    60S            ;LEAVE THIS SUBTEST
40S:     MOV   #1000,R1        ;CHANGE 'READ FROM' ADDRESS BY .25K
          CMP   R1,HILIM       ;WE GO PAST BOUNDARY?
          BLO  20S            ;NO CONTINUE WITH SAME 'WRITTEN TO'
          CLR   (R0)           ;CLEAR OLD 'WRITTEN TO' ADDRESS
          ADD   #1000,R0        ;CHANGE 'WRITTEN TO' ADDRESS BY .25K
          CMP   RO,HILIM       ;WE PAST BOUNDARY?
          BLO  10S            ;NO CONTINUE THIS SUBTEST
          CLC   ;INDICATE SUCCESS OF THIS SUBTEST
60S:     MOV   (SP)+,RO        ;RESTORE RO
          RTS   PC
    
```

77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 11  
MODULE B, MICROTEST #1, MICROSUBTEST C

.SBTTL MODULE B, MICROTEST #1, MICROSUBTEST C

436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479

001412' 010046  
001414' 112767 000003 177466  
001422' 016701 177456  
001426' 012700 000001  
001432' 010002  
001434' 010011  
001436' 011104  
001440' 020004  
001442' 001406  
001444' 112767 000000 177437  
001452' 010003  
001454' 000261  
001456' 000421  
001460' 005702  
001462' 001406  
001464' 006300  
001466' 103362  
001470' 005002  
001472' 012700 177776  
001476' 000756  
001500' 000261  
001502' 006100  
001504' 103753  
001506' 062701 001000  
001512' 020167 177370  
001516' 103743  
001520' 000241  
001522' 012600  
001524' 000207

.....  
:DATA LATCH TEST  
:  
:AT THE FIRST LOCATION OF EACH BANK A '1' IS SHIFTED THROUGH EACH  
:BIT POSITION AND CHECKED FROM LSB TO MSB.  
:THEN IN THE SAME LOCATION A '0' IS SHIFTED THROUGH EACH BIT POSITION  
:AND CHECKED.  
:.....

MICB1C: MOV RO,-(SP) :SAVE RO  
MOVB #3,SUBNUM :TELL WEA ARE IN SUBTEST 'C'  
MOV LOWLIM,R1 :GET BASE ADDRESS OF MEMORY  
1S: MOV #1,R0 :DATA = 1 IN LEAST SIGNIFICANT BIT  
MOV RO,R2 :INDICATE WE ARE SHIFTING A '1'  
2S: MOV RO,(R1) :WRITE LOCATION WITH GOOD DATA  
MOV (R1),R4 :READ DATA FROM SAME LOCATION  
3S: CMP RO,R4 :IS DATA THE SAME AS WRITTEN?  
BEQ 4S :YES, OK  
MOVB #DATERR,ERRTP :ERROR IS DATA ERROR  
MOV RO,R3 :GOOD DATA  
SEC :INDICATE THIS SUBTEST FAILED  
BR 6S :LEAVE THIS SUBTEST  
4S: TST R2 :ARE WE SHIFTING A 1 OR A 0?  
BEQ 5S :ZERO  
ASL RO :SHIFT THE ONE OVER  
BCC 2S :IF THE '1' HAS NOT BEEN SHIFTED...  
CLR R2 :THRU THE 16 POSITIONS CONTINUE WITH 1  
MOV #177776,R0 :ELSE START SHIFTING A '0'  
BR 2S :START WITH LSB = 0 ALL OTHERS 1'S  
5S: SEC :CONTINUE WITH SHIFTING A 0  
ROL RO :MOVE '0' OVER ONE BIT POSITION  
BCS 2S :HAS '0' BEEN IN ALL POSITIONS?  
ADD #1000,R1 :NO CONTINUE WITH SHIFTING A 0  
CMP R1,HILIM :CONTINUE TEST AT NEXT BOUNDARY  
BLO 1S :DONE ALL OF MEMORY?  
CLC :NO  
6S: MOV (SP)+,RO :INDICATE THIS SUBTEST SUCCESSFUL  
RTS PC :RESTORE RO



77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 12  
MODULE B, MICROTEST #1, MICROSUBTEST D

```

480 .SBTTL MODULE B, MICROTEST #1, MICROSUBTEST D
481
482 ;*****
483 ;
484 ;ADDRESS BIT SHIFT #1
485 ;
486 ;THIS TEST CHECKS FOR DUAL ADDRESS PROBLEMS BY FIRST WRITING A BACKGROUND
487 ;PATTERN THROUGHOUT MEMORY.
488 ;THEN STARTING AT THE LOWEST LOCATION IN A BANK IT WRITES THE COMPLEMENT
489 ;OF THE BACKGROUND PATTERN. THEN READS THE MEMORY FOR CORRECT CONTENTS
490 ;IT THEN SHIFTS A '1' THROUGH THE ADDRESS POINTER AND REPEATS THE ABOVE.
491 ;IT DOES THIS FOR EACH BANK.
492 ;
493 ;*****
494
495 001526' 010046 MICB1D: MOV RO,-(SP) ;SAVE R0
496 001530' 112767 000004 177352 MOVB #4,SUBNUM ;TELL WE ARE IN SUBTEST 4
497 001536' 012757 000377 177004 MOV #377,BAKPAT ;LOAD BAKPAT CONSTANT
498 001544' 005003 1$: CLR R3 ;CONTAINS BANK ADDRESS
499 001546' 016700 177332 2$: MOV LOWLIM,R0 ;WRITE LINK MEMORY WITH BACKGROUND...
500 001552' 016720 176772 3$: MOV BAKPAT,(R0)+ ;PATTERN
501 001556' 020067 177324 CMP RO,HILIM
502 001562' 103773 BLO 3$
503 001564' 016700 176760 MOV BAKPAT,R0 ;R0 CONTAINS GOOD DATA
504
505 001570' 005002 4$: CLR R2 ;R2 WILL BE OUR 'WRITTEN TO' ADDRESS
506 001572' 050302 6$: BIS R3,R2 ;INDEX INTO THIS BANK
507 001574' 020267 177304 CMP R2,LOWLIM ;IS RESULT LESS THAN MEM BASE?
508 001600' 103450 BLO 16$ ;YES, DON'T USE THIS ADDRESS
509 001602' 020267 177300 CMP R2,HILIM ;IS RESULT LARGER THAN MEM TOP?
510 001606' 103061 BHIS 20$ ;YES, DON'T USE THIS ADDRESS EITHER
511 001610' 000312 SWAB (R2) ;WRITE COMPLEMENT OF PATTERN
512 001612' 005001 CLR R1 ;R1 WILL BE OUR 'READ FROM' ADDRESS
513 001614' 050301 7$: BIS R3,R1 ;INDEX INTO THIS BANK
514 001616' 020167 177262 CMP R1,LOWLIM ;IS RESULT LESS THAN MEM BASE?
515 001622' 103426 BLO 12$ ;YES, DON'T USE THIS ADDRESS
516 001624' 020167 177256 CMP R1,HILIM ;IS RESULT LARGER THAN MEM TOP?
517 001630' 103333 BHIS 15$ ;YES, DON'T USE THIS ADDRESS EITHER
518 001632' 011104 MOV (R1),R4 ;READ DATA
519 001634' 020102 CMP R1,R2 ;IS 'READ FROM' AND 'WRITTEN TO' SAME?
520 001636' 001414 BEQ 10$ ;YES, GO CHECK DATA
521 001640' 020004 CMP R0,R4 ;NO, DATA READ SHOULD BE SAME AS BAKPAT
522 001642' 001416 BEQ 12$ ;IF SO CONTINUE WITH NEW INDEX
523
524 001644' 112767 000001 177237 8$: MOVB #ADRERR,ERRTYP ;INDICATE ADDRESS ERROR
525 001652' 010003 MOV RO,R3 ;GET GOOD DATA
526 001654' 000261 SEC ;INDICATE THIS SUBTEST FAILED
527 001656' 000441 BR 25$ ;LEAVE THIS SUBTEST
528 001660' 020067 176664 CMP RO,BAKPAT ;DOES R0 CONTAIN SWAPPED DATA?
529 001664' 001405 BEQ 12$ ;NO
530 001666' 000403 BR 11$ ;YES
531
532 001670' 000300 10$: SWAB RO ;MAKE GOOD DATA LIKE SWAPPED BAKPAT
533 001672' 020004 CMP RO,R4 ;IS DATA READ SAME AS DATA WRITTEN?
534 001674' 001363 BNE 8$ ;NO, ERROR
535 001676' 000300 11$: SWAB RO ;RESTORE GOOD DATA TO BAKPAT

```



77MICROB - MICROCODE MODULE B  
 MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 14  
 MODULE B, MICROTEST #1, MICROSUBTEST E

```

561 .SBTTL MODULE B, MICROTEST #1, MICROSUBTEST E
562
563 ;*****
564 ;
565 ;ADDRESS BIT SHIFT #2
566 ;
567 ;THIS TEST CHECKS FOR DUAL ADDRESSING PROBLEMS BY WRITING EACH LOCATION
568 ;OF MEMORY WITH ITS ADDRESS AND THEN VERIFYING EACH LOCATION.
569 ;IT THEN DOES THE SAME THING BUT USES THE COMPLEMENT OF THE ADDRESS
570 ;AS DATA.
571 ;
572 ;*****
573
574 001766' 010046 MICB1E: MOV RO,-(SP) ;SAVE RO
575 001770' 112767 000005 177112 MOVB #5,SUBNUM ;TELL WHICH SUBTEST WE ARE IN
576 001776' 005003 CLR R3 ;FLAG INDICATING ADDRESS IS COMPLEMENTED
577 002000' 016701 177100 MOV LOWLIM,R1 ;GET STARTING ADDRESS OF MEMORY
578 002004' 010100 1$: MOV R1,RO ;GET ADDRESS TO WORK WITH
579 002006' 005703 TST R3 ;SHOULD WE COMPLEMENT THE DATA TO STORE?
580 002010' 001401 BEQ 2$ ;NO STORE AS IS
581 002012' 005100 COM RO ;COMPLEMENT DATA
582 002014' 010021 2$: MOV RO,(R1)+ ;WRITE DATA
583 002016' 020167 177064 CMP R1,HILIM ;IS NEW ADDRESS LARGER THAN MEM?
584 002022' 103770 BLO 1$ ;NO, KEEP FILLING MEMORY
585
586 002024' 014104 3$: MOV -(R1),R4 ;READ DATA STORED
587 002026' 020004 CMP RO,R4 ;IS DATA READ SAME AS WRITTEN?
588 002030' 001406 BEQ 4$ ;YES
589 002032' 112767 000001 177051 MOVB #ADRERR,ERRTYP ;INDICATE ADDRESS ERROR
590 002040' 010003 MOV RO,R3 ;GET GOOD DATA
591 002042' 000261 SEC ;INDICATE FAILURE
592 002044' 000414 BR 10$ ;LEAVE THIS SUBTEST
593 002046' 010100 4$: MOV R1,RO ;CALC GOOD DATA FOR NEXT LOCATION
594 002050' 162700 000002 SUB #2,RO
595 002054' 005703 TST R3
596 002056' 001401 BEQ 5$
597 002060' 005100 COM RO
598 002062' 020167 177016 5$: CMP R1,LOWLIM ;HAVE WE CHECKED ALL LOCATIONS
599 002066' 101356 BHI 3$ ;NOT YET
600 002070' 005103 COM R3 ;HAVE WE DONE IT COMPLEMENTED?
601 002072' 001344 BNE 1$ ;NO, REPEAT WITH COMPLEMENT
602 002074' 000241 CLC ;INDICATE SUCCESS
603 002076' 012600 10$: MOV (SP)+,RO ;RESTORE RO
604 002100' 000207 RTS PC
605

```

77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 15  
MODULE B, MICROTEST #1, MICROSUBTEST F

```

606 .SBTTL MODULE B, MICROTEST #1, MICROSUBTEST F
607
608 :*****
609 :
610 :MARCH TEST
611 :
612 :THIS TEST WRITES A BACKGROUND PATTERN IN ALL OF MEMORY
613 :
614 :1-READ EVERY LOCATION FOR CORRECT DATA, SWAPS BYTES AT EACH LOCATION
615 :AND PROCEED IN MAX TO MIN DIRECTION
616 :2-READ EVERY LOCATION FOR SWAPPED DATA, WRITES ORIGINAL PATTERN IN EACH
617 :LOCATION AND PROCEEDS IN MIN TO MAX DIRECTION
618 :3-REPEAT 1 GOING IN MIN TO MAX DIRECTION
619 :4-REPEAT 2 GOING IN MAX TO MIN DIRECTION
620 :
621 :*****
622
623 002102' 010046 MICB1F: MOV RO,-(SP) ;SAVE RO
624 002104' 112767 000006 176776 MOVB #6,SUBNUM ;TELL WHICH SUBTEST WE ARE IN
625 002112' 005003 CLR R3 ;ADDRESS DIRECTION FLAG 0 = MIN.->MAX
626 002114' 016701 176764 10S: MOV LOWLIM,R1 ;FILL MEMORY WITH BACKGROUND PATTERN
627 002120' 012700 000377 MOV #377,R0 ;BACKGROUND PATTERN=LOW BYTE ALL 1'S
628 002124' 010021 12S: MOV RO,(R1)+
629 002126' 020167 176754 CMP R1,HILIM
630 002132' 103774 BLO 12S
631
632 002134' 014104 20S: MOV -(R1),R4 ;STARTING FROM THE TOP, READ DATA
633 002136' 020004 CMP RO,R4 ;RO = GOOD DATA, R4 = DATA READ
634 002140' 001406 BEQ 30S ;IF SAME OK
635 002142' 112767 000000 176741 MOVB #DATERR,ERRTYP ;INDICATE DATA ERROR
636 002150' 010003 MOV RO,R3 ;GET GOOD DATA
637 002152' 000261 SEC ;INDICATE FAILURE
638 002154' 000454 BR 200S ;LEAVE THIS SUBTEST
639 002156' 000300 30S: SWAB RO ;NEW GOOD DATA PATTERN
640 002160' 010011 MOV RO,(R1) ;STORE AT SAME PLACE
641 002162' 011104 MOV (R1),R4 ;READ IT BACK
642 002164' 020400 CMP R4,RO ;RO=GOOD DATA, R4=DATA READ
643 002166' 001406 BEQ 40S ;IF SAME OK
644 002170' 112767 000000 176713 MOVB #DATERR,ERRTYP ;INDICATE DATA ERROR
645 002176' 010003 MOV RO,R3 ;GET GOOD DATA
646 002200' 000261 SEC ;FAILURE
647 002202' 000441 BR 200S ;LEAVE THIS SUBTEST
648 002204' 000300 40S: SWAB RO ;SWITCH GOOD DATA AGAIN
649 002206' 001027 BNE 90S ;IF ORIGINAL PATTERN THEN WE ARE...
650 ;READING THE MEMORY TO CONTAIN A...
651 ;BACKGROUND OF LOW BYTE = ALL 1'S...
652 ;AND WRITING IT BACK SWAPPED AND...
653 ;VERIFYING SWAPPED DATA
654 002210' 005703 50S: TST R3
655 002212' 001027 BNE 100S
656 002214' 062701 000002 ADD #2,R1 ;WE ARE GOING MIN->MAX SO ADJUST POINTER
657 002220' 020167 176662 60S: CMP R1,HILIM ;WE AT MAX?
658 002224' 103012 BHIS B0S ;YES
659 002226' 011104 70S: MOV (R1),R4 ;READ DATA
660 002230' 020004 CMP RO,R4 ;RO=GOOD DATA, R4=DATA READ
661 002232' 001751 BEQ 30S ;OK IF SAME

```

MICROB - MICROCODE MODULE B  
 MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 16  
 MODULE B, MICROTEST #1, MICROSUBTEST F

```

662 002234' 112767 000000 176647      MOVB  #DATERR,ERRTYP      ;INDICATE DATA ERROR
663 002242' 010003                MOV  RO,R3                ;GET GOOD DATA
664 002244' 000261                SEC                        ;INDICATE FAILURE
665 002246' 000417                BR   200$                 ;LEAVE THIS SUBTEST
666 002250' 000742                BR   30$
667 002252' 000300                80$: SWAB RO                ;SWITCH GOOD DATA
668 002254' 001727                BEQ  20$                 ;IF LOW BYTE = ALL 0'S GO IN MAX->MIN
669 002256' 005103                COM  R3                  ;ELSE GO IN MIN->MAX DIRECTION
670 002260' 016701 176620        MOV  LOWLIM,R1            ;SET ADDRESS POINTER TO MIN
671 002264' 000760                BR   70$
672
673 002266' 005703                90$: TST R3                ;ARE WE GOING MIN->MAX?
674 002270' 001347                BNE  50$                 ;YES
675
676 002272' 020167 176606        100$: CMP R1,LOWLIM        ;NO, CHECK TO SEE IF WE ARE AT MIN
677 002276' 101316                BHI  20$                 ;NOT YET
678 002300' 000300                SWAB RO                    ;WE ARE AT MIN SO SWITCH GOOD DATA
679 002302' 001751                BEQ  70$                 ;IF LOW BYTE = ALL 0'S ELSE WE ARE DONE
680 002304' 000241                CLC                        ;INDICATE SUCCESS
681 002306' 012600                200$: MOV (SP)+,RO        ;RESTORE RO
682 002310' 000207                RTS PC
683
684 002312' 002314                MICBSZ::MICBSZ-MICROB+2   ;SIZE OF MICROCODE MODULE B
685 000001                .END

```

77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 18  
CROSS REFERENCE TABLE -- USER SYMBOLS

ADDREG= 177774		103#																	
ADRERR= 000001		101#	414	524	589														
BAKPAT 000550R	002	235#	497*	500	503	528	555*												
BIT0 = 000001		51#	81																
BIT1 = 000002		50#	82																
BIT10 = 002000		41#	53	70															
BIT11 = 004000		40#	69																
BIT12 = 010000		39#	53	68	89														
BIT13 = 020000		38#	67	88															
BIT14 = 040000		37#	66	87	210														
BIT15 = 100000		36#	65	86	215	218													
BIT2 = 000004		49#	83																
BIT3 = 000010		48#	84																
BIT4 = 000020		47#	85																
BIT5 = 000040		46#																	
BIT6 = 000100		45#																	
BIT7 = 000200		44#																	
BIT8 = 000400		43#	53	71															
BIT9 = 001000		42#																	
BOTH 001112R	002	258*	288	290*	306	332#													
B1STBL 001066R	002	263	320#																
CLRFIF= 021034		22#																	
CPDREG= 177776		104#	119*																
CSRFLG= 000001		81#	167	198															
CSRVEC= 000064		74#	144*	145*															
CSRVRT 000376R	002	143	198#																
DATERR= 000000		100#	365	420	459	635	644	662											
DMACSR= 021002		9#	209	218*															
DMAF = 021022		17#																	
DMAINT 000432R	002	138	209#																
DMATO = 021004		10#																	
DMAT1 = 021006		11#																	
DMAVEC= 000114		75#	139*	140*															
DMAVC = 021024		18#																	
DMIDONE 000546R	002	209*	210	215	234#														
DNI = 004000		69#	123	191															
ERRFLG= 000002		82#	201																
ERRINT 000406R	002	125	201#																
ERRTYP 001111R	002	331#	359*	365*	414*	420*	459*	524*	589*	635*	644*	662*							
FAT1 = 000400		71#																	
FLG2 000530R	002	160*	163	167	192*	198*	201*	222*	229#	265*	355								
HILIM 001106R	002	260*	292*	329#	350	370	425	429	475	501	509	516	553	583					
		629	657																
INERR = 000003		80#	189																
INPDM = 000001		78#	121	175	187														
INTST = 000002		79#	256																
IOADR = 020000		97#	98																
IOSIZ = 020000		93#	98																
IPCSR0= 021000		8#	123*	172	191*	202*	212*	217*	223*										
IPCSR1= 021020		16#	122*	175*	190*	257*	317*												
IPCSR2 000536R	002	158*	159*	232#	274	275													
LASFTP= 012400		53#																	
LFRBUF= 021032		21#																	
LINADR= 100000		99#	102																
LINSIZ= 077774		95#	102																
LOWLIM 001104R	002	259*	291*	328#	348	353	398	399	452	499	507	514	577	598					



77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 20  
CROSS REFERENCE TABLE -- USER SYMBOLS

SUBMUM	001110R	002	280	330#	345*	396*	451*	496*	575*	624*							
TBLB	000526R	002	181	227#													
TIMINT	000424R	002	148	206#													
TXI	= 010000		68#														
UNIERR=	020060		88#	202													
WCSADR=	000000		96#	97													
WCSSIZ=	020000		92#	97	260	291	292	302	311								
.	= 002314R	002	125	133	138	143	148	169	181	204	219	225	227	263	294		
			301	309	320	321	322	323	324	325							



77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 72  
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 353

BCOMPL	1#
BERROR	1#
BGAU	1#
BGAUT	1#
BGNCLN	1#
BGNDU	1#
BGNHRD	1#
BGNHW	1#
BGINI	1#
BGNMOD	1#
BGNMSG	1#
BGNPRO	1#
BGNPTA	1#
BGNRPT	1#
BGNSEG	1#
BGNSET	1#
BGNSFT	1#
BGNSRV	1#
BGNSUB	1#
BGNSW	1#
BGNTST	1#
BNCOMP	1#
BNERRO	1#
BREAK	1#
BRESET	1#
CRLOOP	1#
CLOCK	1#
CLOSE	1#
CLRVEC	1#
COMMEN	1#
DELAY	1#
DESCRI	1#
DEVTYP	1#
DISPAT	1#
DISPLA	1#
DOCLN	1#
DODU	1#
DORPT	1#
ENDAU	1#
ENDAUT	1#
ENDCLN	1#
ENDCOM	1#
ENDDU	1#
ENDHRD	1#
ENDHW	1#
ENDINI	1#
ENDMOD	1#
ENDMSG	1#
ENDPRO	1#
ENDPTA	1#
ENDRPT	1#
ENDSEG	1#
ENDSET	1#
ENDSFT	1#
ENDSRV	1#
ENDSUB	1#

77MICROB - MICROCODE MODULE B  
MICROB.MAC 07-APR-83 16:06MACY11 30A(1052) 07-APR-83 16:49 PAGE 23  
CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	10
ENDTST	10
EQUALS	10
ERRDF	10
ERRHRD	10
ERROR	10
ERRSF	10
ERRSOF	10
ERRTBL	10
ESCAPE	10
EXIT	10
FEQUAL	10
GETBYT	10
GETPRI	10
GETWOR	10
GMAIA	10
GMAID	10
GMAIIL	10
GPHARD	10
GPRMA	10
GPRMD	10
GPRML	10
HEADER	10
INLOOP	10
IOSETU	10
IOSTAR	10
KT11	10
LASTAD	10
MANUAL	10
MEMORY	10
MSBYTE	10
MSCHEC	10
MSCNTO	10
MSCOUN	10
MSDATA	10
MSDECR	10
MSDEFA	10
MSENDE	10
MSERRI	10
MSESCA	10
MSESCS	10
MSEXCP	10
MSEXIT	10
MSXSE	10
MSXTJ	10
MSGEN	10
MSGEND	10
MSGETS	10
MSGETT	10
MSGNGB	10
MSGNIN	10
MSGNLS	10
MSGNSU	10
MSGNTA	10
MSGNTE	10
MSHAPT	10

77MICROB - MICROCODE MODULE B  
 MICROB.MAC 07-APR-83 16:06

MACV11 30A(1052) 07-APR-83 16:49 PAGE 24  
 CROSS REFERENCE TABLE -- MACRO NAMES

MSHMAP 1#  
 MSINCR 1#  
 MSIOSE 1#  
 MBLDRO 1#  
 MSMASK 1#  
 MSACHI 1#  
 MSACLO 1#  
 MSASK1 1#  
 MSPOP 1#  
 MSPRIN 1#  
 MSPUSH 1#  
 MSPUT 1#  
 MSPUT1 1#  
 MSRAD! 1#  
 MSRDRO 1#  
 MSRNRO 1#  
 MSSETS 1#  
 MSSTAR 1#  
 MSVC 1#  
 MSTLAB 1#  
 MSTSL 1#  
 MSWORD 1#  
 MSXFER 1#  
 OPEN 1#  
 POINTE 1#  
 PRINTB 1#  
 PRINTF 1#  
 PRINTS 1#  
 PRINTX 1#  
 READBU 1#  
 READEF 1#  
 RFLAGS 1#  
 SETPRI 1#  
 SETVEC 1#  
 SLASH 1#  
 STARS 1#  
 SVC 1#  
 XFER 1#  
 XFERF 1#  
 XFERT 1#

. ABS. 000000 000  
 000000 001  
 MICROB 002314 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MICROB.OBJ,MICROB.LST/CR/SOL/ML:TOC=SVC34R.P11,MICROB.MAC  
 RUN-TIME: 2 3 .3 SECONDS  
 RUN-TIME RATIO: 36/6=6.0  
 CORE USED: 31K (61 PAGES)

68SVC.MLB SOURCE FILE MACY11 30A(1052) 07-APR-83 16:49 PAGE 2  
 MICROC.MAC 07-APR-83 16:06

```

1
2
3      .TITLE  MICROC - MICROCODE MODULE C
4      000000'
5      .CSECT MICRC
6
7      .SBTTL  REGISTER DEFINITIONS USED BY THE T11
8      021000  IPCSRO = 21000  :INTERNAL PCSRO ADDRESS
9      021002  DMACSR = 21002  :DMA ENGINE CONTROL STATUS REGISTER
10     021004  DMATO  = 21004  :DMA ENGINE TO ADDRESS REGISTER #0
11     021006  DMAT1  = 21006  :DMA ENGINE TO ADDRESS REGISTER #1
12     021010  MDMA0  = 21010  :MICROCPU DMA TO ADDRESS REGISTER #0
13     021012  MDMA1  = 21012  :MICROCPU DMA TO ADDRESS REGISTER #1
14     021014  MDMA0  = 21014  :MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15     021016  MDMA1  = 21016  :MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16     021020  IPCSR1 = 21020  :INTERNAL PCSR1 ADDRESS
17     021022  DMAF   = 21022  :DMA ENGINE FROM ADDRESS REGISTER
18     021024  DMAWC  = 21024  :DMA ENGINE WORD COUNT REGISTER
19     021026  MDMAW0 = 21026  :MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20     021030  LTAC   = 21030  :LINK TRANSMIT ADDRESS COUNTER REGISTER
21     021032  LFRBUF = 21032  :LINK RECIEVE BUFFER ADDRESS FIFO
22     021034  CLRFIF = 21034  :CLEAR FIFO
23     021036  MDMAW1 = 21036  :MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24     021040  PCSRSW = 21040  :SWITCH PACK REGISTER
25     021042  MDMSR  = 21042  :MICROCPU DMA STATUS REGISTER
26     021044  LRBUF  = 21044  :LINK RECIEVE BUFFER COMPLETED
27     021060  PHYAD0 = 21060  :PHYSICAL ADDRESS ROM BYTE 0
28     021062  PHYAD1 = 21062  :PHYSICAL ADDRESS ROM BYTE 1
29     021064  PHYAD2 = 21064  :PHYSICAL ADDRESS ROM BYTE 2
30     021066  PHYAD3 = 21066  :PHYSICAL ADDRESS ROM BYTE 3
31     021070  PHYAD4 = 21070  :PHYSICAL ADDRESS ROM BYTE 4
32     021072  PHYAD5 = 21072  :PHYSICAL ADDRESS ROM BYTE 5
33
34     .SBTTL  OTHER DEFINITIONS USED BY THE MICROCODE
35
36     100000  BIT15  = 100000
37     040000  BIT14  = 40000
38     020000  BIT13  = 20000
39     010000  BIT12  = 10000
40     004000  BIT11  = 4000
41     002000  BIT10  = 2000
42     001000  BIT9   = 1000
43     000400  BIT8   = 400
44     000200  BIT7   = 200
45     000100  BIT6   = 100
46     000040  BIT5   = 40
47     000020  BIT4   = 20
48     000010  BIT3   = 10
49     000004  BIT2   = 4
50     000002  BIT1   = 2
51     000001  BIT0   = 1
52
53     012400  LASFTP = BIT8:BIT10:BIT12 ;LOAD AND START FUNCTION TEST PATTERN
54     000340  PRI07  = 340
55     000300  PRI06  = 300
56     000240  PRI05  = 240
    
```

77 MICROCODE - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 3  
 MICROCODE.MAC 07-APR-83 10:06 OTHER DEFINITIONS USED BY THE MICROCODE

57	000200	PRI04 =	200	
58	000140	PRI03 =	140	
59	000100	PRI02 =	100	
60	000040	PRI01 =	40	
61	000000	PRI00 =	0	
62		:		
63		:PCSRO - PORT CONTROL STATUS REGISTER 0		
64		:		
65	100000	SERI =	BIT15	
66	040000	PCEI =	BIT14	
67	020000	RXI =	BIT13	
68	010000	TXI =	BIT12	
69	004000	DMI =	BIT11	
70	002000	RCEI =	BIT10	
71	000400	FATI =	BIT8	
72		:		
73	000134	SANVEC=	134	:VECTOR ADDRESS FOR THE SANITY TIMER
74	000064	CSRVEC=	64	:VECTOR ADDRESS FOR CSR WRITE INTERRUPT
75	000114	DMAVEC=	114	:VECTOR ADDRESS FOR DMA DONE INTERRUPT
76	000140	PARVEC=	140	:VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
77	001000	STACK=	1000	:STACK LOCATION
78	000001	INMON=	1	:IN MICROMONITOR STATE
79	000002	INTST=	2	:IN A TEST STATE
80	000003	INERR=	3	:IN ERROR STATE
81	000001	CSRFLG=	BIT0	:CSR WRITE INTERRUPT OCCURED
82	000002	ERRFLG=	BIT1	:UNEXPECTED ERROR OCCURED
83	000004	PARFLG=	BIT2	:PARITY ERROR OCCURED
84	000010	NXPFLG=	BIT3	:NON-EXISTANT MEMORY ERROR OCCURRED
85	000020	NPRFLG=	BIT4	:NPR TIMEOUT ERROR OCCURRED
86	100000	NPRERR=	BIT15	:PCSRO FLAG INDICATING NPR ERROR OCCURRED
87	040000	NXPERR=	BIT14	:PCSRO FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
88	020000	UNIERR=	BIT13	:PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
89	010000	PARERR=	BIT12	:PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
90	020000	SIZ4K=	20000	:4K WORDS
91	040000	SIZ8K=	SIZ4K*2	:8K WORDS
92	020000	WCSSIZ=	SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
93	020000	IOSIZ=	SIZ4K	:SIZE OF I/O PAGE
94	040000	ROMSIZ=	SIZ8K	:SIZE OF ROM
95	077774	LINSIZ=	SIZ8K*2-4	:SIZE OF LINK MEMORY
96	000000	WCSADR=	0	:BASE ADDRESS OF WCS
97	020000	IOADR=	WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
98	040000	ROMADR=	IOADR+IOSIZ	:BASE ADDRESS OF ROM
99	100000	LINADR=	ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
100	000000	DATERR=	0	:FLAG INDICATING DATA ERROR OCCURRED
101	000001	ADRERR=	1	:FLAG INDICATING ADDRESS ERROR OCCURRED
102	177774	MODREG=	LINADR+LINSIZ	:LINK MODE REGISTER
103	177774	ADDREG=	MODREG	:LINK STATION ADDRESS RAM REGISTER
104	177776	CMODREG=	MODREG+2	:LINK COMMAND REGISTER
105				

77 MICROCODE MODULE C  
MICROC.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161

.SBTTL C\_MODULE MICROCODE

\*\*\*\*\*  
:THIS MODULE CONTAINS A MICROMONITOR AND MICROCODE FOR:  
:1-CLEAR INSTRUCTION  
:2-PCSR0 INTERRUPT BIT TEST  
:3-SANITY TIMER TEST  
:4-COMPREHENSIVE LINK MEMORY TEST  
:5-DMA 'TO' ADDRESS REGISTER TEST  
:6-DMA 'FROM' ADDRESS REGISTER TEST  
:7-DMA BLOCK TRANSFER TEST  
\*\*\*\*\*

MICROC::MTPS #PRI07 ;DISABLE INTERRUPTS  
MOV #0,#PCMDREG ;TURN OFF THE LINK  
MOV #STACK,SP ;SETUP STACK  
MOV #INMON,PCSR1 ;TELL HOST WE ARE IN MICROMONITOR  
MOV PCSR1,#IPCSR1  
MOV PC,R0 ;GET ADDRESS OF UNEXPECTED ERROR...  
ADD #ERRINT-,R0 ;HANDLER  
CLR R1 ;FILL ALL UNUSED VECTORS WITH TRAP...  
10S: MOV R0,(R1)+ ;HANDLER  
MOV #PRI07,(R1)+  
CMP R1,#1000  
BLT 10S  
  
MOV PC,R0 ;SETUP PARITY TRAP VECTOR  
ADD #PARINT-,R0  
MOV R0,#PARVEC  
MOV #PRI07,#PARVEC+2  
  
MOV PC,R0 ;SETUP DMA INTERRUPT VECTOR  
ADD #DMAINT-,R0  
MOV R0,#DMAVEC  
MOV #PRI06,#DMAVEC+2  
  
MOV PC,R0 ;SETUP CSR WRITE VECTOR  
ADD #CSRWRT-,R0  
MOV R0,#CSRVEC  
MOV #PRI04,#CSRVEC+2  
  
MOV PC,R0 ;SETUP SANTITY TIMER VECTOR  
ADD #T,MINT-,R0  
MOV R0,#SANVEC  
MOV #PRI05,#SANVEC+2  
  
MOV #PPCSR0,R0 ;GET SWITCH PACK BITS  
BIS #176000,R0 ;MAP THEM INTO HOST I/O PAGE  
ASL R0 ;SHIFT OVER TO POSITION CORRECTLY  
ASL R0  
ASL R0

77 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 5  
 MICROC.MAC 07-APR-83 16:06 C\_MODULE MICROCODE

162	000174'	062700	000004			ADD	#4,R0		:PCSR2 IS PCSR0+4
163	000200'	010067	000324			MOV	R0,IPCSR2		:SAVE PCSR2 ADDRESS
164	000204'	012767	000003	000320		MOV	#3,IPCSR2+2		:HIGH ORDER BITS 17:16
165	000212'	005067	000304			CLR	FLG3		:INITIALIZE FLAG WORD
166	000216'	012737	004000	021000		MOV	#DN1,#IPCSRO		:TELL HOST THE LOAD AND START FINISHED
167	000224'	106427	000000		158:	MTPS	#PRI00		:ALLOW INTERRUPTS
168									
169	000230'	005767	000266		208:	TST	FLG3		:WAIT FOR AN INTERRUPT
170	000234'	001775				BEQ	208		
171									
172	000236'	106427	000340			MTPS	#PRI07		:RAISE CPU PRIORITY
173	000242'	032767	000001	000252		BIT	#CSRFLG,FLG3		:DID HOST GIVE US A COMMAND?
174	000250'	001001				BNE	308		:YES
175	000252'	000777				BR	.		:NO, ERROR SO JUST SIT HERE...
176									:FOR LACK OF ANYTHING BETTER TO DO
177									
178	000254'	113700	021000		308:	MOVB	#IPCSRO,R0		:GET WHAT HOST WROTE TO PCSRO
179	000260'	042700	177760			BIC	#177760,R0		:STRIP ALL BUT COMMAND BITS
180	000264'	022700	000017			CMP	#17,R0		:RETURN TO OPERATIONAL MICROCODE?
181	000270'	001425				BEQ	608		:YES
182	000272'	010701				MOV	PC,R1		:GET ADDRESS OF OUR COMMAND TABLE
183	000274'	062701	000206			ADD	#TBLC-,R1		
184	000300'	006300				ASL	R0		:MAKE COMMAND A BYTE OFFSET
185	000302'	060001				ADD	R0,R1		:USE IT TO INDEX INTO COMMAND TABLE
186	000304'	061101				ADD	(R1),R1		:R1 NOW HAS COMMAND ROUTINE ADDRESS
187	000306'	004711				JSR	PC,(R1)		:EXECUTE AS COMMANDED FROM HOST
188	000310'	103404				BCC	408		:ERROR OCCURRED
189	000312'	112767	000001	000206		MOVB	#INMON,PCSR1		:INDICATE TO HOST WE ARE BACK IN...
190	000320'	000403				BR	458		:MICROMONITR
191	000322'	112767	000003	000176	408:	MOVB	#INERR,PCSR1		:INDICATE TO HOST ERROR OCCURRED
192	000330'	016737	000172	021020	458:	MOV	PCSR1,#IPCSR1		
193	000336'	005067	000160			CLR	FLG3		:RESET FLAG WORD
194	000342'	000730				BR	158		:GO WAIT FOR ANOTHER COMMAND
195									
196	000344'	005000			608:	CLR	R0		:FAKE SUCCESSFUL SELF TEST RESULTS
197	000346'	000137	040006			JMP	#40006		:START OPERATIONAL MICROCODE
198									
199	000352'	052767	000001	000142	CSRWRT:	BIS	#CSRFLG,FLG3		:INDICATE A CSR WRITE INTERRUPT OCCURED
200	000360'	000002				RTI			
201									
202	000362'	052767	000002	000132	ERRINT:	BIS	#ERRFLG,FLG3		:INDICATE A UNEXPECTED INTERRUPT OCCURED
203	000370'	012737	020000	021000		MOV	#UNIERR,#IPCSRO		:TELL HOST AN UNEXPECTED INTERRUPT
204									:HAPPENED
205	000376'	000777				BR	.		:JUST SIT HERE AND SPIN WHEELS
206									:COUNT ON THE HOST TO TIMEOUT WAITING
207									
208	000400'	005267	000:20		TIMINT:	INC	SANTIM		:COUNT TICKS AS THEY OCCUR
209	000404'	000002				RTI			
210									
211	000406'	013767	021002	000124	DMAINT:	MOV	#DMACSR,DMDONE		:GET DMA STATUS
212	000414'	032767	040000	000116		BIT	#BIT14,DMDONE		:DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
213	000422'	001404				BEQ	108		:NO
214	000424'	012737	040000	021000		MOV	#NXPERR,#IPCSRO		:YES, TELL HOST A NON-EXISTANT MEMORY
215									:LOCATION WAS ADDRESSED
216	000432'	000407				BR	208		
217	000434'	032767	100000	000076	108:	BIT	#BIT15,DMDONE		:DID A NPR TIMEOUT OCCUR?





77 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 7  
MICROC.MAC 07-APR-83 16:06 C\_MODULE MICROCODE

247  
248  
249  
250

.SBTTL MODULE C, MICRO CLEAR FUNCTION

251 000544' 105067 177757  
252 000550' 000241  
253 000552' 000207

CLRC: CLRB PCSNi+1  
CLC  
RTS PC

;CLEAR THE SELF TEST FIELD

77 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 8  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #1

```

254
255
256 000554' 112767 000002 177744 MICC1: MOVB #INTST,PCSR1
257 000562' 016737 177740 021020 MOV PCSR1,@#IPCSR1
258 000570' 016737 177734 021010 MOV IPCSR2,@#MDMA0 ;PICK UP ADDRESS OF PCBB
259 000576' 016737 177730 021012 MOV IPCSR2+2,@#MDMA1
260 000604' 013700 021014 MOV @#MDMAR0,R0 ;R0 HAS CONTENTS OF HOST'S PCSR2
261 000610' 013701 021014 MOV @#MDMAR0,R1 ;R1 HAS CONTENTS OF HOST'S PCSR3
262 000614' 010037 021010 MOV R0,@#MDMA0 ;FETCH CONTENTS OF PCBB+0
263 000620' 010137 021012 MOV R1,@#MDMA1
264
265 ;AT THIS POINT MDMAR0 WILL CONTAIN THE INTERRUPT BIT THAT WE ARE TO SET
266
267 000624' 013737 021014 021000 MOV @#MDMAR0,@#IPCSR0 ;SET CORRESPONDING INTERRUPT BIT
268 000632' 112767 000001 177667 MOVB #1,PCSR1+1 ;TELL HOST WHAT TEST WE JUST FINISHED
269 000640' 000241 CLC ;SUCCESS
270 000642' 000207 RTS PC

```

77 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 9  
MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #2

.SBTTL MODULE C, MICROTEST #2

271										
272										
273										
274	000644'	112767	000002	177654	MICC2:	MOVB	#!NTST,PCSR1			:TELL HOST WE ARE TESTING
275	000652'	016737	177650	021020		MOV	PCSR1,@#IPCSR1			
276	000660'	005067	177640			CLR	SANTIM			:CLEAR TIMER COUNT
277	000664'	106427	000200			MTPS	#PRIO4			:LOWER PRIORITY TO ALLOW TIMER INTERRUPT
278	000670'	026727	177630	000012	10\$:	CMP	SANTIM,#10.			:WAIT FOR TEN TICKS OF THE TIMER
279	000676'	002774				BLT	10\$			
280	000700'	106427	000340			MTPS	#PRIO7			:RAISE PRIORITY TO STOP TIMER
281	000704'	112767	000002	177615		MOVB	#2,PCSR1+1			:TELL HOST WHAT TEST WE JUST FINISHED
282	000712'	016737	177610	021020		MOV	PCSR1,@#IPCSR1			
283	000720'	012737	004000	021000		MOV	#DNI,@#IPCSRO			:TELL HOST WE ARE DONE
284	000726'	000241				CLC				
285	000730'	000207				RTS	PC			

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 10  
MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #3

.SBTTL MODULE C, MICROTEST #3

\*\*\*\*\*8  
: THIS MICROTEST CHECKS THE LINK MEMORY. IT DOES SO BY RUNNING A SERIES  
: OF MICROSUBTESTS ON THE MEMORY BETWEEN 16K AND 32K (100000-177776).  
: THE SUBTESTS ARE:  
: A-ACCESS TEST  
: B-ADDRESS SHIFT TEST  
: C-DATA LATCH TEST  
: D-ADDRESS BIT SHIFT #1  
: E-ADDRESS BIT SHIFT #2  
: F-MARCH TEST  
\*\*\*\*\*8

286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341

000732° 112767 000002 177566  
000740° 016737 177562 021020  
000746° 010700  
000750° 062700 000134  
000754° 011001  
000756° 001437  
000760° 060001  
000762° 004711  
000764° 103403  
000766° 062700 000002  
000772° 000770  
000774° 016737 177530 021010  
001002° 016737 177524 021012  
001010° 013700 021014  
001014° 013702 021014  
001020° 010037 021010  
001024° 010237 021012  
001030° 016737 000066 021026  
001036° 010137 021026  
001042° 010337 021026  
001046° 010437 021026  
001052° 000261  
071054° 000401  
001056° 000241  
001050° 112767 000003 177441  
001066° 016737 177434 021020  
001074° 012737 004000 021000  
001102° 000207  
001104° 000020  
001106° 000160  
001110° 000322  
001112° 000436  
001114° 000704  
001116° 001024  
001120° 000000

MICC3: MOVB #INTST,PCSR1 :TELL HOST WE ARE TESTING  
MOV PCSR1,@#IPCSR1  
MOV PC,R0 :GET ADDRESS OF SUBTEST LIST  
ADD #C3STBL-.,R0  
10\$: MOV (R0),R1 :GET OFFSET FROM ENTRY TO SUBTEST  
BEQ 30\$ :NO MORE ENTRIES  
ADD R0,R1 :CALC ACTUAL SUBTEST ADDRESS  
JSR PC,(R1) :GO EXECUTE SUBTEST  
BCS 20\$ :ERROR OCCURRED IN SUBTEST  
ADD #2,R0 :POINT TO NEXT SUBTEST IN LIST  
BR 10\$  
20\$: MOV IPCSR2,@#MDMA0 :GET CONTENTS OF HOST'S PCRS2 AND PCRS3  
MOV IPCSR2+2,@#MDMA1  
MOV @#MDMAR0,R0 :R0=CONTENTS OF PCRS2  
MOV @#MDMAR0,R2 :R2=CONTENTS OF PCRS3  
MOV R0,@#MDMA0 :POINT TO PCBB+0  
MOV R2,@#MDMA1  
MOV SUBNUM,@#MDMAW0 :LOAD PCBB+0 WITH SUBTEST #...  
:AND PCBB+1 WITH ERROR TYPE  
MOV R1,@#MDMAW0 :LOAD PCBB+2 WITH FAILING ADDRESS  
MOV R3,@#MDMAW0 :LOAD PCBB+4 WITH GOOD DATA  
MOV R4,@#MDMAW0 :LOAD PCBB+6 WITH BAD DATA  
SEC :TELL MICROMONITOR THIS TEST FAILED  
BR 40\$  
30\$: CLC :TELL MICROMONITOR THIS TEST WAS SUCCESSFUL  
40\$: MOVB #3,PCSR1+1 :TELL HOST WHICH TEST THIS WAS  
MOV PCSR1,@#IPCSR1  
MOV #DNI,@#IPCSRO :TELL HOST THIS MICROTEST IS FINISHED  
RTS PC  
C3STBL: MICC3A-.  
MICC3B-.  
MICC3C-.  
MICC3D-.  
MICC3E-.  
MICC3F-.  
.WCRD 0 :END OF LIST

MICROC - MICROCODE MODULE C  
MICROC.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 11  
MODULE C, MICROTEST #3

342  
343 001122' 000  
344 001123' 000  
345

SUBNUM: .BYTE 0  
ERRTYP: .BYTE 0

:CURRENT SUBTEST # BEING EXECUTED  
:TYPE OF ERROR, DATA =0, ADDRESS=1

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 12  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #3, MICROSUBTEST A

.SBTTL MODULE C, MICROTEST #3, MICROSUBTEST A

```

346
347
348
349
350
351
352
353
354
355
356 001124 010046
357 001126 112767 000001 177766
358 001134 106427 000300
359
360 001140 010600
361 001142 012706 177774
362 001146 012705 037776
363 001152 004767 000000
364 001156 077503
365
366 001160 010006
367 001162 010703
368 001164 062703 177772
369
370 001170 012701 100000
371 001174 011104
372 001176 032767 000004 177316
373 001204 001002
374
375 001206 020304
376 001210 001410
377 001212 112767 000003 177306
378 001220 112767 000000 177675
379 001226 000261
380 001230 000414
381
382 001232 062701 000002
383 001236 020127 177774
384 001242 001354
385
386 001244 012701 100000
387 001250 005021
388 001252 020127 177774
389 001256 001374
390
391 001260 000241
392 001262 012600
393 001264 000207
394

```

```

:*****
:THIS IS AN ACCESS TEST OF LINK MEMORY. IT WRITES DATA
:ALL THROUGH 16K TO 32K(100000,177772) AND VERIFIES SAME. IT THEN WRITES
:ZEROS . IT ALSO CHECKS FOR PARITY ERRORS.
:*****
MICC3A: MOV      RO,-(SP)          :SAVE RO
        MOVB     #1,SUBNUM       :TELL WE ARE IN SUBTEST A
        MTPS     #PRI06          :ALLOW PARITY ERRORS

        MOV      SP,RO           :SAVE THE STACK POINTER FOR A MOMENT
        MOV      #LINADR+LINSIZ,SP :POINT STACK TO TOP OF LINK MEMORY
        MOV      #LINSIZ/2,R5     :NUMBER OF WORDS IN LINK MEMORY
58:     JSR      PC,108          :WRITE ADDRESS OF NEXT INSTRUCTION
108:    SOB      R5,58           :ALL OVER LINK MEMORY

        MOV      RO,SP           :RESTORE STACK
        MOV      PC,R3           :GET DATA THAT WAS WRITTEN TO LINK MEMORY
        ADD      #108-.,R3

        MOV      #LINADR,R1      :POINT TO BASE OF LINK MEMORY
128:    MOV      (R1),R4          :READ DATA
        BIT      #PARFLG,FLG3     :DID A PARITY ERROR OCCUR?
        BNE     208              :YES

158:    CMP      R3,R4           :NO, WAS DATA GOOD?
        BEQ     308              :YES
208:    MOVB     #INERR,PCSR1     :TELL HOST THIS SUBTEST FAILED
        MOVB     #DATERR,ERRTYP  :TELL MICROMONITOR TYPE OF ERROR
        SEC     BR               :TELL MICROMONITOR ERROR OCCURRED
        BR      408

308:    ADD      #2,R1           :POINT TO NEXT ADDRESS
        CMP     R1,#LINADR+LINSIZ :DONE ALL OF LINK MEMORY?
        BNE     128              :NOT YET

328:    MOV      #LINADR,R1      :OK NOW FILL WITH ZEROES
        CLR     (R1)+
        CMP     R1,#LINADR+LINSIZ
        BNE     328

358:    CLC                    :INDICATE SUCCESS
408:    MOV      (SP)+,RO        :RESTORE RO
        RTS     PC

```

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 13  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #3, MICROSUBTEST B

.SBTTL MODULE C, MICROTEST #3, MICROSUBTEST B

395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449

```

:*****
:ADDRESS SHIFT TEST
:THIS TEST ASSUMES ALL MEMORY BETWEEN 10000 AND 177776 IS ZEROS
:IT CHECKS FOR PROPER BANK SELECTION BY WRITING 1'S IN A LOCATION
:AND CHECKING FOR 1'S IN THE SAME LOCATION OF OTHER 4K BANKS (ADDRESS ERROR).
:IT ALSO CHECKS FOR DATA ERRORS I.E. NON-ZERO DATA IN LOCATIONS NOT
:WRITTEN WITH 1'S AND MAKES SURE LOCATIONS WRITTEN WITH 1'S HAVE 1'S IN THEM
:*****
    
```

```

001266' 010046      MICCC38: MOV    R0,-(SP)      ;SAVE R0
001270' 112767 000002 177624  MOVB   #2,SUBNUM    ;TELL HIM WE ARE IN THE SECOND SUBTFST
                                10S:  MOV    #LINADR,R0      ;SET 'WRITTEN TO' ADDRESS
                                MOV    #LINADR,R1      ;SET 'READ FROM' ADDRESS
                                MOV    #177777,(R0)    ;WRITE DATA INTO LOCATION AT BASE...
                                20S:  MOV    (R1),R4      ;OF A 4K BOUNDARY
                                20S:  MOV    (R1),R4      ;READ DATA FROM MEMORY ADDRESS WHICH...
                                CMP    R0,R1           ;IS AN EVEN 4K INCREMENT AWAY
                                20S:  CMP    R0,R1           ;IS 'READ FROM' AND 'WRITTEN TO'
                                BEQ    70S            ;ADDRESSES THE SAME?
                                CLR    R3              ;YES, SO DATA IN BOTH SHOULD BE THE SAME
                                TST    R4              ;GOOD DATA IS ZEROS
                                BEQ    50S            ;DATA IN 'READ FROM' MUST BE 0'S
                                40S:  MOV    #177777,R3    ;OK GO CHANGE 'READ FROM' ADDRESS
                                40S:  MOVB   #DATERR,ERRTYP ;ERROR OCCURRED BUT WE DON'T KNOW IF...
                                40S:  MOVB   #INERR,PCSR1  ;IT IS A DATA ERROR OR AN ADDRESS ERROR
                                SEC                ;WAS DATA READ ALL 1'S?
                                BR         60S          ;NO, SO IT WAS A DATA ERROR
                                70S:  CMP    #177777,R4    ;YES, IT WAS AN ADDRESS ERROR
                                BEQ    50S            ;TELL HOST THIS TEST FAILED
                                70S:  MOV    #177777,R3    ;INDICATE FAILURE
                                70S:  MOV    #177777,R3    ;LEAVE THIS SUBTEST
                                70S:  MOVB   #INERR,PCSR1  ;DATA READ MUST BE ALL 1'S
                                70S:  MOVB   #INERR,PCSR1  ;IT IS GOOD
                                70S:  SEC                ;GOOD DATA IS ONES
                                70S:  BR         60S          ;IT WAS A DATA ERROR
                                40S:  MOV    #177777,R3    ;TELL HOST THIS TEST FAILED
                                40S:  MOVB   #INERR,PCSR1  ;INDICATE FAILURE
                                40S:  MOVB   #INERR,PCSR1  ;LEAVE THIS SUBTEST
                                40S:  SEC                ;CHANGE 'READ FROM' ADDRESS BY 4K
                                40S:  BR         60S          ;CONTINUE SAME 'WRITTEN TO' IF NOT PAST 32K
                                50S:  ADD    #20000,R1     ;CLEAR OLD 'WRITTEN TO' ADDRESS
                                50S:  BCC    20S          ;CHANGE 'WRITTEN TO' ADDRESS BY 4K
                                50S:  CLR    (R0)          ;CONTINUE IF NOT PAST 32K
                                50S:  ADD    #20000,R0     ;INDICATE SUCCESS OF THIS SUBTEST
                                50S:  BCC    10S          ;RESTORE R0
                                60S:  CLC                ;RESTORE R0
                                60S:  MOV    (SP)+,R0
                                60S:  RTS    PC
    
```

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 14  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #3, MICROSUBTEST C

```

450 .SBTTL MODULE C, MICROTEST #3, MICROSUBTEST C
451
452 :*****
453 :
454 :DATA LATCH TEST
455 :
456 :AT THE FIRST LOCATION OF EACH 4K BANK A '1' IS SHIFTED THROUGH EACH
457 :BIT POSITION AND CHECKED FROM LSB TO MSB.
458 :THEN IN THE SAME LOCATION A '0' IS SHIFTED THROUGH EACH BIT POSITION
459 :AND CHECKED.
460 :
461 :*****
462
463
464 001432' 010046 MICCC3C: MOV R0,-(SP) ;SAVE R0
465 001434' 112767 000003 177460 MOVB #3,SUBNUM ;TELL WE ARE IN SUBTEST 'C'
466 001442' 012701 100000 MOV #LINADR,R1 ;GET BASE ADDRESS OF LINK MEMORY
467 001446' 012700 000001 1S: MOV #1,R0 ;DATA = 1 IN LEAST SIGNIFICANT BIT
468 001452' 010002 MOV R0,R2 ;INDICATE WE ARE SHIFTING A '1'
469 001454' 010011 2S: MOV R0,(R1) ;WRITE LOCATION WITH GOOD DATA
470 001456' 011104 MOV (R1),R4 ;READ DATA FROM SAME LOCATION
471 001460' 020004 3S: CMP R0,R4 ;IS DATA THE SAME AS WRITTEN?
472 001462' 001411 BEQ 4S ;YES, OK
473 001464' 112767 000000 177431 MOVB #DATERR,ERRTP ;ERROR IS DATA ERROR
474 001472' 010003 MOV R0,R3 ;GOOD DATA
475 001474' 112767 000003 177024 MOVB #INERR,PCSR1 ;TELL HOST THIS TEST FAILED
476 001502' 000261 SEC ;INDICATE THIS SUBTEST FAILED
477 001504' 000417 BR 6S ;LEAVE THIS SUBTEST
478 001506' 005702 4S: TST R2 ;ARE WE SHIFTING A 1 OR A 0?
479 001510' 001406 BEQ 5S ;ZERO
480 001512' 006300 ASL R0 ;SHIFT THE ONE OVER
481 001514' 103357 BCC 2S ;IF THE '1' HAS NOT BEEN SHIFTED...
482
483 001516' 005002 CLR R2 ;THRU THE 16 POSITIONS CONTINUE WITH 1
484 001520' 012700 177776 MOVB #177776,R0 ;ELSE START SHIFTING A '0'
485 001524' 000753 BR 2S ;START WITH LSB = 0 ALL OTHERS 1'S
486 001526' 000261 5S: SEC ;CONTINUE WITH SHIFTING A 0
487 001530' 006100 ROL R0 ;MOVE '0' OVER ONE BIT POSITION
488 001532' 103750 BCS 2S ;HAS '0' BEEN IN ALL POSITIONS?
489 001534' 062701 020000 ADD #SIZ4K,R1 ;NO CONTINUE WITH SHIFTING A 0
490 001540' 103342 BCC 1S ;CONTINUE TEST AT NEXT 4K BOUNDARY...
491 001542' 000241 CLC ;IF NOT PAST 32K
492 001544' 012600 6S: MOV (SP)+,R0 ;INDICATE THIS SUBTEST SUCCESSFUL
493 001546' 000207 RTS PC ;RESTORE R0

```



78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 15  
MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #2, MICROSUBTEST D

494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549

.SBTTL MODULE C, MICROTEST #2, MICROSUBTEST D  
:\*\*\*\*\*  
:ADDRESS BIT SHIFT #1  
:THIS TEST CHECKS FOR DUAL ADDRESS PROBLEMS BY FIRST WRITING A BACKGROUND  
:PATTERN THROUGHOUT LINK MEMORY.  
:THEN STARTING AT THE LOWEST LOCATION IN A BANK IT WRITES THE COMPLEMENT  
:OF THE BACKGROUND PATTERN. THEN READS THE MEMORY FOR CORRECT CONTENTS  
:IT THEN SHIFTS A '1' THROUGH THE ADDRESS POINTER AND REPEATS THE ABOVE.  
:IT DOES THIS FOR EACH 4K BANK.  
:\*\*\*\*\*  
001550' 010046  
001552' 112767 000004 177342  
001560' 012767 000377 176'54  
001566' 005003  
001570' 012700 100000  
001574' 016720 176742  
001600' 020027 177774  
001604' 103773  
001606' 016700 176730  
  
001612' 005002  
001614' 050302  
001616' 020227 100000  
001622' 103453  
001624' 020227 177774  
001630' 103065  
001632' 000312  
001634' 005001  
001636' 050301  
001640' 020127 100000  
001644' 103431  
001646' 020127 177774  
001652' 103036  
001654' 011104  
001656' 020102  
001660' 001417  
001662' 020004  
001664' 001421  
  
001666' 112767 000001 177227  
001674' 010003  
001676' 000261  
001700' 112767 000003 176620  
001706' 000442  
001710' 020067 176626  
001714' 001405  
001716' 000403  
  
001720' 000300  
001722' 020004  
001724' 001360

MICC3D: MOV R0, -(SP) ;SAVE R0  
MOV #4, SUBNUM ;TELL WE ARE IN SUBTEST 4  
MOV #377, BAKPAT ;LOAD BAKPAT CONSTANT  
1\$: CLR R3 ;CONTAINS 4K BANK ADDRESS  
2\$: MOV #LINADR, R0 ;WRITE LINK MEMORY WITH BACKGROUND...  
3\$: MOV BAKPAT, (R0)+ ;PATTERN  
CMP R0, #LINADR+LINSIZ  
BLO 3\$  
MOV BAKPAT, R0 ;R0 CONTAINS GOOD DATA  
  
4\$: CLR R2 ;R2 WILL BE OUR 'WRITTEN TO' ADDRESS  
6\$: BIS R3, R2 ;INDEX INTO THIS 4K BANK  
CMP R2, #LINADR ;IS RESULT LESS THAN LINK MEM BASE?  
BLO 16\$ ;YES, DON'T USE THIS ADDRESS  
CMP R2, #LINADR+LINSIZ ;IS RESULT LARGER THAN LINK MEM TOP?  
BHIS 20\$ ;YES, DON'T USE THIS ADDRESS EITHER  
SWAB (R2) ;WRITE COMPLEMENT OF PATTERN  
CLR R1 ;R1 WILL BE OUR 'READ FROM' ADDRESS  
7\$: BIS R3, R1 ;INDEX INTO THIS 4K BANK  
CMP R1, #LINADR ;IS RESULT LESS THAN LINK MEM BASE?  
BLO 12\$ ;YES, DON'T USE THIS ADDRESS  
CMP R1, #LINADR+LINSIZ ;IS RESULT LARGER THAN LINK MEM TOP?  
BHIS 15\$ ;YES, DON'T USE THIS ADDRESS EITHER  
MOV (R1), R4 ;READ DATA  
CMP R1, R2 ;IS 'READ FROM' AND 'WRITTEN TO' SAME?  
BEQ 10\$ ;YES, GO CHECK DATA  
CMP R0, R4 ;NO, DATA READ SHOULD BE SAME AS BAKPAT  
BEQ 12\$ ;IF SO CONTINUE WITH NEW INDEX  
  
8\$: MOV #ADRERR, ERRYP ;INDICATE ADDRESS ERROR  
MOV R0, R3 ;GET GOOD DATA  
SEC ;INDICATE THIS SUBTEST FAILED  
MOV #INERR, PCSR1 ;TELL HOST THIS TEST FAILED  
BR 25\$ ;LEAVE THIS SUBTEST  
CMP R0, BAKPAT ;DOES R0 CONTAIN SWAPPED DATA?  
BEQ 12\$ ;NO  
BR 11\$ ;YES  
  
10\$: SWAB R0 ;MAKE GOOD DATA LIKE SWAPPED BAKPAT  
CMP R0, R4 ;IS DATA READ SAME AS DATA WRITTEN?  
BNE 8\$ ;NO, ERROR



78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 17  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #3, MICROSUBTEST E

```

577 .SBTTL MODULE C, MICROTEST #3, MICROSUBTEST E
578
579 :*****
580 :
581 :ADDRESS BIT SHIFT #2
582 :
583 :THIS TEST CHECKS FOR DUAL ADDRESSING PROBLEMS BY WRITING EACH LOCATION
584 :OF LINK MEMORY WITH ITS ADDRESS AND THEN VERIFYING EACH LOCATION.
585 :IT THEN DOES THE SAME THING BUT USES THE COMPLEMENT OF THE ADDRESS
586 :AS DATA.
587 :
588 :*****
589
590 MICC3E: MOV RO,-(SP) ;SAVE RO
591 002020' 010046 000005 177072 MOVB #5,SUBNUM ;TELL WHICH SUBTEST WE ARE IN
592 002030' 005003 CLR R3 ;FLAG INDICATING ADDRESS IS COMPLEMENTED
593 002032' 012701 100000 MOV #LINADR,R1 ;GET STARTING ADDRESS OF LINK MEMORY
594 002036' 010100 1$: MOV R1,RO ;GET ADDRESS TO WORK WITH
595 002040' 005703 TST R3 ;SHOULD WE COMPLEMENT THE DATA TO STORE?
596 002042' 001401 BEQ 2$ ;NO STORE AS IS
597 002044' 005100 COM RO ;COMPLEMENT DATA
598 002046' 010021 2$: MOV RO,(R1)+ ;WRITE DATA
599 002050' 020127 177774 CMP R1,#LINADR+LINSIZ ;IS NEW ADDRESS LARGER THAN LINK MEM?
600 002054' 103770 BLO 1$ ;NO, KEEP FILLING LINK MEMORY
601
602 002056' 014104 3$: MOV -(R1),R4 ;READ DATA STORED
603 002060' 020004 CMP RO,R4 ;IS DATA READ SAME AS WRITTEN?
604 002062' 001411 BEQ 4$ ;YES
605 002064' 112767 000001 177031 MOVB #ADRERR,ERRTYP ;INDICATE ADDRESS ERROR
606 002072' 010003 MOV RO,R3 ;GET GOOD DATA
607 002074' 000261 SEC ;INDICATE FAILURE
608 002076' 112767 000003 176422 MOVB #INERR,PCSR1 ;TELL HOST THIS TEST FAILED
609 002104' 000414 BR 10$ ;LEAVE THIS SUBTEST
610 002106' 010100 4$: MOV R1,RO ;CALC GOOD DATA FOR NEXT LOCATION
611 002110' 162700 000002 SUB #2,RO
612 002114' 005703 TST R3
613 002116' 001401 BEQ 5$
614 002120' 005100 COM RO
615 002122' 020127 100000 5$: CMP R1,#LINADR ;HAVE WE CHECKED ALL LOCATIONS
616 002126' 101353 BHI 3$ ;NOT YET
617 002130' 005103 COM R3 ;HAVE WE DONE IT COMPLEMENTED?
618 002132' 001341 BNE 1$ ;NO, REPEAT WITH COMPLEMENT
619 002134' 000241 CLC ;INDICATE SUCCESS
620 002136' 012600 10$: MOV (SP)+,RO ;RESTORE RO
621 002140' 000207 RTS PC
622
    
```

MICROC - MICROCODE MODULE C  
MICROC.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 18  
MODULE C, MICROTEST #3, MICROSUBTEST F

```

623 .SBTTL MODULE C, MICROTEST #3, MICROSUBTEST F
624
625 :*****
626 :
627 :MARCH TEST
628 :
629 :THIS TEST WRITES A BACKGROUND PATTERN IN ALL OF LINK MEMORY
630 :
631 :1-READ EVERY LOCATION FOR CORRECT DATA, SWAPS BYTES AT EACH LOCATION
632 :AND PROCEED IN MAX TO MIN DIRECTION
633 :2-READ EVERY LOCATION FOR SWAPPED DATA, WRITES ORIGINAL PATTERN IN EACH
634 :LOCATION AND PROCEEDS IN MIN TO MAX DIRECTION
635 :3-REPEAT 1 GOING IN MIN TO MAX DIRECTION
636 :4-REPEAT 2 GOING IN MAX TO MIN DIRECTION
637 :
638 :*****
639
640 002142' 010046 MICCF: MOV R0,-(SP) ;SAVE R0
641 002144' 112767 000006 176750 MOVB #6,SUBNUM ;TELL WHICH SUBTEST WE ARE IN
642 002152' 005003 CLR R3 ;ADDRESS DIRECTION FLAG 0 = MIN.->MAX
643 002154' 012701 100000 10$: MOV #LINADR,R1 ;FILL LINK MEMORY WITH BACKGROUND PATTERN
644 002160' 012700 000377 MOV #377,R0 ;BACKGROUND PATTERN=LOW BYTE ALL 1'S
645 002164' 010021 12$: MOV R0,(R1)+
646 002166' 020127 177774 CMP R1,#LINADR+LINSIZ
647 002172' 103774 BLO 12$
648
649 002174' 014104 20$: MOV -(R1),R4 ;STARTING FROM THE TOP, READ DATA
650 002176' 020004 CMP R0,R4 ;R0 = GOOD DATA, R4 = DATA READ
651 002200' 001411 BEQ 30$ ;IF SAME OK
652 002202' 112767 000000 176713 MOVB #DATERR,ERRTYP ;INDICATE DATA ERROR
653 002210' 010003 MOV R0,R3 ;GET GOOD DATA
654 002212' 000261 SEC ;INDICATE FAILURE
655 002214' 112767 000003 176304 MOVB #INERR,PCSR1 ;TELL HOST THIS TEST FAILED
656 002222' 000462 BR 200$ ;LEAVE THIS SUBTEST
657 002224' 000300 30$: SWAB R0 ;NEW GOOD DATA PATTERN
658 002226' 010011 MOV R0,(R1) ;STORE AT SAME PLACE
659 002230' 011104 MOV (R1),R4 ;READ IT BACK
660 002232' 020400 CMP R4,R0 ;R0=GOOD DATA, R4=DATA READ
661 002234' 001411 BEQ 40$ ;IF SAME OK
662 002236' 112767 000000 176657 MOVB #DATERR,ERRTYP ;INDICATE DATA ERROR
663 002244' 010003 MOV R0,R3 ;GET GOOD DATA
664 002246' 000261 SEC ;FAILURE
665 002250' 112767 000003 176250 MOVB #INERR,PCSR1 ;TELL HOST THIS TEST FAILED
666 002256' 000444 BR 200$ ;LEAVE THIS SUBTEST
667 002260' 000300 40$: SWAB R0 ;SWITCH GOOD DATA AGAIN
668 002262' 001032 BNE 90$ ;IF ORIGINAL PATTERN THEN WE ARE...
669 ;READING THE MEMORY TO CONTAIN A...
670 ;BACKGROUND OF LOW BYTE = ALL 1'S...
671 ;AND WRITING IT BACK SWAPPED AND...
672 ;VERIFING SWAPPED DATA
673 002264' 005703 50$: TST R3
674 002266' 001032 BNE 100$
675 002270' 062701 000002 60$: ADD #2,R1 ;WE ARE GOING MIN->MAX SO ADJUST POINTER
676 002274' 020127 177774 CMP R1,#LINADR+LINSIZ ;WE AT MAX?
677 002300' 103015 BHS 80$ ;YES
678 002302' 011104 70$: MOV (R1),R4 ;READ DATA

```



79 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 20  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #4

```

704                                     .SBTTL  MODULE C, MICROTEST #4
705
706 002374' 112767 000002 176124  MICC4:  MOVB  #INTST,PCSR1      ;TELL HOST WE ARE TESTING
707 002402' 016737 176120 021020      MOV  PCSR1,#IPCSR1
708 002410' 016737 176114 021010      MOV  IPCSR2,#DMA0      ;GET PCBB ADDRESS THROUGH HOST'S PCSR2
709 002416' 016737 176110 021012      MOV  IPCSR2+2,#DMA1
710 002424' 013700 021014      MOV  @#DMAR0,R0      ;R0 HAS CONTENTS OF HOST'S PCSR2
711 002430' 013701 021014      MOV  @#DMAR0,R1      ;R1 HAS CONTENTS OF HOST'S PCSR3
712 002434' 010037 021010      MOV  R0,#DMA0      ;SETUP TO READ PCBB+0
713 002440' 010137 021012      MOV  R1,#DMA1
714 002444' 013737 021014 021004      MOV  @#DMAR0,@#DMATO
715                                     ;GET DATA PATTERN AND WRITE IT TO...
716 002452' 013737 021004 021026      MOV  @#DMATO,@#DMAW0  ;DMA 'TO' ADDRESS REGISTER
717                                     ;READ DATA PATTERN BACK AND WRITE...
718 002460' 112767 000004 176041      MOVB  #4,PCSR1+1     ;BACK INTO HOST MEMORY
719 002466' 016737 176034 021020      MOV  PCSR1,@#IPCSR1  ;TELL HOST WHAT TEST WE JUST FINISHED
720 002474' 012737 004000 021000      MOV  #DNI,@#IPCSRO
721 002502' 000241      CLC
722 002504' 000207      RTS
723                                     PC

```

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 21  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #5

.SBTTL MODULE C, MICROTEST #5

724									
725									
726									
727	002506'	112767	000002	176012	MICCS:	MOVB	#'NTST,PCSR1		:TELL HOST WE ARE TESTING
728	002514'	016737	176006	021020		MOV	PCSR1,@#IPCSR1		
729	002522'	012700	100000			MOV	#LINADR,RO		:BASE ADDRESS OF LINK MEMORY
730	002526'	012701	037776			MOV	#LINSIZ/2,R1		:NUMBER OF WORDS IN LINK MEMORY
731	002532'	010010			10\$:	MOV	RO,(RO)		:FILL LINK MEMORY WITH THE ADDRESS
732	002534'	005720				TST	(RO)+		
733	002536'	005301				DEC	R1		:OF EACH LOCATION
734	002540'	001374				BNE	10\$		
735	002542'	016737	175762	021010		MOV	IPCSR2,@#DMAO		:GET PCBB THROUGH HOST'S PCSR2
736	002550'	016737	175756	021012		MOV	IPCSR2+2,@#DMA1		
737	002556'	013700	021014			MOV	@#DMARO,RO		:RO HAS CONTENTS OF HOST'S PCSR2
738	002562'	013701	021014			MOV	@#DMARO,R1		:R1 HAS CONTENTS OF HOST'S PCSR3
739	002566'	010037	021010			MOV	RO,@#DMAO		:RO HAS ADDRESS OF PCBB+0
740	002572'	010137	021012			MOV	R1,@#DMA1		:R1 HAS ITS HIGH ORDER BITS
741	002576'	013737	021014	021022		MOV	@#DMARO,@#DMAF		:LOAD DMA 'FROM' ADDRESS REGISTER
742									:WITH ADDRESS SUPPLIED FROM HOST
743	002604'	062700	000002			ADD	#2,RO		:LOAD DMA 'TO' ADDRESS REGISTER
744	002610'	005501				ADC	R1		:WITH PCBB+2 ADDRESS
745	002612'	010037	021004			MOV	RO,@#DMATO		
746	002616'	010137	021006			MOV	R1,@#DMAT1		
747	002622'	012737	000002	021024		MOV	#2,@#DMAWC		:LOAD WORD COUNT TO TRANSFER 1 WORD
748	002630'	005067	175704			CLR	DMDONE		:CLEAR DMA DONE FLAG
749	002634'	005237	021002			INC	@#DMACSR		:START DMA ENGINE
750	002640'	106427	000240			MTPS	#PRI05		:LOWER CPU PRIORITY
751	002644'	005767	175670		20\$:	TST	DMDONE		:IS THE TRANSFER COMPLETE?
752	002650'	001775				BEQ	20\$		:NO, WAIT FOR THE INTERRUPT
753	002652'	106427	000340			MTPS	#PRI07		
754	002656'	112767	000005	175643		MOVB	#5,PCSR1+1		:TELL HOST WHAT TEST WE JUST FINISHED
755	002664'	016737	175636	021020		MOV	PCSR1,@#IPCSR1		
756	002672'	012737	004000	021000		MOV	#DWI,@#IPCSRO		:TELL HOST WE ARE DONE
757	002700'	000241				CLC			
758	002702'	000207				RTS	PC		

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 22  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #6

SEQ 376

```

759
760
761
762 002704' 112767 000002 175614 MICC6: MOVB #!NTST,PCSR1 ;TELL HOST WE ARE TESTING
763 002712' 016737 175610 021020 MOV PCSR1,#IPCSR1
764 002720' 012700 100000 MOV #LINADR,R0 ;GET BASE ADDRESS OF LINK MEMORY
765 002724' 012701 037776 MOV #LINSIZ/2,R1 ;SIZE OF LINK MEMORY IN WORDS
766 002730' 010010 10S: MOV R0,(R0) ;FILL LINK MEMORY WITH ADDRESS OF
767 002732' 005720 TST (R0)+
768 002734' 005301 DEC R1 ;EACH LOCATION
769 002736' 001374 BNE 10S
770 002740' 016737 175564 021010 MOV IPCSR2,#DMA0 ;GET CONTENTS OF HOST'S PCSR2
771 002746' 016737 175560 021012 MOV IPCSR2+2,#DMA1
772 002754' 013700 021014 MOV #DMAR0,R0 ;R0 HAS CONTENTS OF HOST'S PCSR2
773 002760' 013701 021014 MOV #DMAR0,R1 ;R1 HAS CONTENTS OF HOST'S PCSR3
774 002764' 010037 021010 MOV R0,#DMA0 ;SETUP TO READ CONTENTS OF PCBB+0
775 002770' 010137 021012 MOV R1,#DMA1
776 002774' 013737 021014 021004 MOV #DMAR0,#DMATO ;SETUP DMA 'TO' REGISTER FROM PCBB+0
777 003002' 012737 100000 021022 MOV #LINADR,#DMAF ;SETUP DMA 'FROM' WITH LINK MEM BASE
778 003010' 012737 003774 021024 MOV #3774,#DMAWC ;SETUP MAXIMUM BYTE COUNT
779 003016' 005067 175516 CLR DMADONE
780 003022' 005237 021002 INC #DMACSR ;TURN ON DMA ENGINE
781 003026' 106427 000240 MTPS #PRIOS ;ALLOW A DMA DONE INTERRUPT
782 003032' 005747 175502 20S: TST DMADONE ;WAIT FOR THE INTERRUPT FROM THE DMA
783 003036' 001775 BEQ 20S ;ENGINE
784 003040' 112767 000006 175461 MOVB #6,PCSR1+1 ;TELL HOST WHAT TEST WE JUST FINISHED
785 003046' 016737 175454 021020 MOV PCSR1,#IPCSR1
786 003054' 012737 004000 021000 MOV #DNI,#IPCSRO ;TELL HOST THAT THIS TEST IS DONE
787 003062' 000241 CLC
788 003064' 000207 RTS PC

```



78 MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 23  
 MICROC.MAC 07-APR-83 16:06 MODULE C, MICROTEST #6

```

789
790      :SBTTL <DMA 'TO' REGISTER RIPPLE TEST>
791      :
792      :
793 003066' 112767 000002 175432 MICCSZ:: MOV  #INIST,PCSR1
794 003074' 016737 175426 021020      MOV  PCSR1,@#IPCSR1
795      :
796      :
797 003102' 012700 100004      MOV  #LINADR+4,R0
798 003106' 012720 052525      MOV  #052525,(R0)+
799 003112' 012710 031463      MOV  #031463,(R0)
800      :
801      :
802 003116' 016737 175406 021010      MOV  IPCSR2,@#DMA0
803 003124' 016737 175402 021012      MOV  IPCSR2+2,@#DMA1
804 003132' 013700 021014      MOV  @#DMAR0,R0
805 003136' 013701 021014      MOV  @#DMAR0,R1
806      :
807      :
808 003142' 010037 021004      MOV  R0,@#DMAT0
809 003146' 012737 100004 021022      MOV  #LINADR+4,@#DMAF
810 003154' 012737 000004 021024      MOV  #4,@#DMAWC
811 003162' 010700      MOV  PC,R0
812 003164' 062700 175222      ADD  #DMAINT-.,R0
813 003170' 010037 000114      MOV  R0,@#DMAVEC
814 003174' 012737 000300 000116      MOV  #PRIO6,@#DMAVEC+2
815 003202' 005067 175332      CLR  DMDCNE
816 003206' 005237 021002      INC  @#DMACSR
817 003212' 106427 000240      MTPS #PRIO5
818 003216' 005767 175316      20$: TST  DMDCNE
819 003222' 001775      BEQ  20$
820      :
821      :
822 003224' 112767 000007 175275      MOV  #7,PCSR1+1
823 003232' 016737 175270 021020      MOV  PCSR1,@#IPCSR1
824 003240' 012737 004000 021000      MOV  #DNI,@#IPCSRO
825 003246' 000241      CLC
826 003250' 000207      RTS  PC
827
828 003252' 003254      MICCSZ::MICCSZ-MICROC+2      :SIZE OF MICROCODE MODULE C
829
830      000001      .END

```

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 25  
 MICROC.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- USER SYMBOLS

ADDREG= 177774		103#																		
ADRERR= 000001		101#	428	538	605															
BAKPAT 000542R	002	245#	511*	514	517	543	571*													
BIT0 = 000001		51#	81																	
BIT1 = 000002		50#	82																	
BIT10 = 002000		41#	53	70																
BIT11 = 004000		40#	69																	
BIT12 = 010000		39#	53	68	89															
BIT13 = 020000		38#	67	88																
BIT14 = 040000		37#	66	87	212															
BIT15 = 100000		36#	65	86	217	220														
BIT2 = 000004		49#	83																	
BIT3 = 000010		48#	84																	
BIT4 = 000020		47#	85																	
BIT5 = 000040		46#																		
BIT6 = 000100		45#																		
BIT7 = 000200		44#																		
BIT8 = 000400		43#	53	71																
BIT9 = 001000		42#																		
CLRC 000544R	002	230	251#																	
CLRFIF= 021034		22#																		
CMDREG= 177776		104#	125*																	
CSRFLG= 000001		81#	173	199																
CSRVEC= 000064		74#	149*	150*																
CSRWRT 000352R	002	148	199#																	
C3STBL 001104R	002	305	335#																	
DATERR= 000000		100#	378	435	473	652	662	681												
DMACSR= 021002		9#	211	220*	749*	780*	816*													
DMAF = 021022		17#	741*	777*	809*															
DMAINT 000406R	002	143	211#	812																
DMATO = 021004		10#	714*	716	745*	776*	808*													
DMAT1 = 021006		11#	746*																	
DMAVEC= 000114		75#	144*	145*	813*	814*														
DMAWC = 021024		18#	747*	778*	810*															
DMADONE 000540R	002	211*	212	217	244#	748*	751	779*	782	815*	818									
DNI = 004000		69#	166	283	332	720	756	786	824											
ERRFLG= 000002		82#	202																	
ERRINT 000362R	002	130	202#																	
ERRTYP 001123R	002	344#	378*	428*	435*	473*	538*	605*	652*	662*	681*									
FAT1 = 000400		71#																		
FLG3 000522R	002	165*	169	173	193*	199*	202*	225*	239#	372										
INERR = 000003		80#	191	377	429	436	475	541	608	655	665	684								
INPDM = 000001		78#	127	189																
INTST = 000002		79#	256	274	302	706	727	762	793											
IOADR = 020000		97#	98																	
IOSIZ = 020000		93#	98																	
IPCSRO= 021000		8#	166*	178	203*	214*	219*	226*	267*	283*	332*	720*	756*	786*						
		824*																		
IPCSR1= 021020		16#	128*	192*	257*	275*	282*	303*	331*	707*	719*	728*	755*	763*						
		785*	794*	823*																
IPCSR2 000530R	002	163*	164*	242#	258	259	315	316	708	709	735	736	770	771						
		802	803																	
LASFTP= 012400		53#																		
LFRBUF= 021032		21#																		
LINADR= 100000		99#	102	361	370	383	386	388	412	413	466	513	515	521						
		523	528	530	569	593	599	615	643	646	676	690	696	729						



78 MICROC - MICROCODE MODULE C  
MICROC.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:49 PAGE 27  
CROSS REFERENCE TABLE -- USER SYMBOLS

PRI06 = 000300		55#	145	358*	814															
PRI07 = 000340		54#	124*	133	140	172*	280*	753*												
RCE1 = 002000		70#																		
ROMADR = 040000		98#	99																	
ROMSIZ = 040000		94#	99																	
RX1 = 020000		67#																		
SANTIM 000524R	002	208*	240#	276*	278															
SAWVEC = 000134		73#	154*	155*																
SER1 = 100000		65#																		
SIZ4K = 020000		90#	91	92	93	489	556	565												
SIZ8K = 040000		91#	94	95																
STACK = 001000		77#	126																	
SUBMUM 001122R	002	321	343#	357*	410*	465*	510*	591*	641*											
TBLC 000502R	002	183	230#																	
TIMINT 000400R	002	153	208#																	
TXI = 010000		68#																		
UNIERR = 020000		88#	203																	
WCSADR = 000000		96#	97																	
WCSSIZ = 020000		92#	97																	
. = 003254R	002	130	138	143	148	153	175	183	205	221	230	231	232	233						
		234	235	236	237	305	335	336	337	338	339	340	368	812						

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 29  
MICROC.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL	1#
BERRON	1#
BGNAU	1#
BGNAUT	1#
BGNCLN	1#
BGNDU	1#
BGNHRD	1#
BGNHW	1#
BGNINI	1#
BGNMOD	1#
BGNMSG	1#
BGNPRO	1#
BGNPTA	1#
BGNRPT	1#
BGNSEG	1#
BGNSET	1#
BGNSFT	1#
BGNSRV	1#
BGNSUB	1#
BGNSW	1#
BGNTST	1#
BNCOMP	1#
BNERRO	1#
BREAK	1#
BRESET	1#
CKLOOP	1#
CLOCK	1#
CLOSE	1#
CLRVEC	1#
COMREN	1#
DELAY	1#
DESCRI	1#
DEVTYP	1#
DISPAT	1#
DISPLA	1#
DOCLN	1#
DODU	1#
DORPT	1#
ENDAU	1#
ENDAUT	1#
ENDCLN	1#
ENDCOM	1#
ENDDU	1#
ENDHRD	1#
ENDHW	1#
ENDINI	1#
ENDMOD	1#
ENDMSG	1#
ENDPRO	1#
ENDPTA	1#
ENDRPT	1#
ENDSEG	1#
ENDSET	1#
ENDSFT	1#
ENDSRV	1#
ENDSUB	1#

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 30  
MICROC.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	1#
ENDTST	1#
EQUALS	1#
ERRDF	1#
ERRHRD	1#
ERROR	1#
ERRSF	1#
ERRSOF	1#
ERRTBL	1#
ESCAPE	1#
EXIT	1#
FEQUAL	1#
GETBYT	1#
GETPRI	1#
GETWOR	1#
GMANIA	1#
GMANID	1#
GMANIL	1#
GPHARD	1#
GPRMA	1#
GPRMD	1#
GPRML	1#
HEADER	1#
INLOOP	1#
IOSETU	1#
IOSTAR	1#
KT11	1#
LASTAD	1#
MANUAL	1#
MEMORY	1#
MSBYTE	1#
MSCHEC	1#
MSCNTO	1#
MSCOUN	1#
MSDATA	1#
MSDECR	1#
MSDEFA	1#
MSENDE	1#
MSERRI	1#
MSESCA	1#
MSESCS	1#
MSXCP	1#
MSEXIT	1#
MSXSE	1#
MSXTJ	1#
MSGEN	1#
MSGENB	1#
MSGETS	1#
MSGETT	1#
MSGNGB	1#
MSGNIN	1#
MSGNLS	1#
MSGNSU	1#
MSGNTA	1#
MSGNTE	1#
MSHAPT	1#

78 MICROC - MICROCODE MODULE C MACY11 30A(1052) 07-APR-83 16:49 PAGE 31  
 MICROC.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

MSHMAP 1#  
 MSINCR 1#  
 MSIOSE 1#  
 MSLDRO 1#  
 MSMASK 1#  
 MSACHI 1#  
 MSACLO 1#  
 MSASK1 1#  
 MSPOP 1#  
 MSPRIN 1#  
 MSPUSH 1#  
 MSPUT 1#  
 MSPUT1 1#  
 MSRADI 1#  
 MSABRO 1#  
 MSANRO 1#  
 MSSETS 1#  
 MSSTAR 1#  
 MSSVC 1#  
 MSTLAB 1#  
 MSTSTL 1#  
 MSWORD 1#  
 MSXFER 1#  
 OPEN 1#  
 POINTE 1#  
 PRINTB 1#  
 PRINTF 1#  
 PRINTS 1#  
 PRINTX 1#  
 READBU 1#  
 READEF 1#  
 RFLAGS 1#  
 SETPRI 1#  
 SETVEC 1#  
 SLASH 1#  
 STARS 1#  
 SVC 1#  
 XFER 1#  
 XFERF 1#  
 XFERT 1#

. ABS. 000000 000  
 000000 001  
 MICRC 003254 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MICROC.OBJ,MICROC.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROC.MAC  
 RUN-TIME: 23.4 SECONDS  
 RUN-TIME RATIO: 39/6=5.9  
 CORE USED: 31K (61 PAGES)

68SVC.MLB SOURCE FILE MACY11 30A(1052) 07-APR-83 16:50 PAGE 2  
 MICROD.MAC 07-APR-83 16:06

```

1
2
3
4      000000'
5
6
7
8      021000      IPCSRO =      21000      :INTERNAL PCSRO ADDRESS
9      021002      DMACSR =      21002      :DMA ENGINE CONTROL STATUS REGISTER
10     021004      DMATO  =      21004      :DMA ENGINE TO ADDRESS REGISTER #0
11     021006      DMAT1  =      21006      :DMA ENGINE TO ADDRESS REGISTER #1
12     021010      MDMA0  =      21010      :MICROCPU DMA TO ADDRESS REGISTER #0
13     021012      MDMA1  =      21012      :MICROCPU DMA TO ADDRESS REGISTER #1
14     021014      MDMAR0 =      21014      :MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15     021016      MDMAR1 =      21016      :MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16     021020      IPCSR1 =      21020      :INTERNAL PCSR1 ADDRESS
17     021022      DMAF   =      21022      :DMA ENGINE FROM ADDRESS REGISTER
18     021024      DMAWC  =      21024      :DMA ENGINE WORD COUNT REGISTER
19     021026      MDMAW0 =      21026      :MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20     021030      LTAC   =      21030      :LINK TRANSMIT ADDRESS COUNTER REGISTER
21     021032      LFRBUF =      21032      :LINK RECIEVE BUFFER ADDRESS FIFO
22     021034      CLRFIF =      21034      :CLEAR FIFO
23     021036      MDMAW1 =      21036      :MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24     021040      PCSRSW =      21040      :SWITCH PACK REGISTER
25     021042      MDMSR  =      21042      :MICROCPU DMA STATUS REGISTER
26     021044      LRBUF  =      21044      :LINK RECIEVE BUFFER COMPLETED
27     021060      PHYAD0 =      21060      :PHYSICAL ADDRESS ROM BYTE 0
28     021062      PHYAD1 =      21062      :PHYSICAL ADDRESS ROM BYTE 1
29     021064      PHYAD2 =      21064      :PHYSICAL ADDRESS ROM BYTE 2
30     021066      PHYAD3 =      21066      :PHYSICAL ADDRESS ROM BYTE 3
31     021070      PHYAD4 =      21070      :PHYSICAL ADDRESS ROM BYTE 4
32     021072      PHYAD5 =      21072      :PHYSICAL ADDRESS ROM BYTE 5
33     177774      MODREG =      177774     :LINK MODE REGISTER
34     177774      ADDRREG =      177774     :LINK STATION ADDRESS RAM REGISTER
35     177776      CMDREG =      177776     :LINK COMMAND REGISTER
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56     012400      LASFTP =      BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN

```



76MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 3  
OTHER DEFINITIONS USED BY THE MICROCODE

57	000340	PRI07 =	340	
58	000300	PRI06 =	300	
59	000240	PRI05 =	240	
60	000200	PRI04 =	200	
61	000140	PRI03 =	140	
62	000100	PRI02 =	100	
63	000040	PRI01 =	40	
64	000000	PRI00 =	0	
65		:		
66		:PCSR0 - PORT CONTROL STATUS REGISTER 0		
67		:		
68	100000	SERI =	BIT15	
69	040000	PCEI =	BIT14	
70	020000	RXI =	BIT13	
71	010000	TXI =	BIT12	
72	004000	DNI =	BIT11	
73	002000	RCEI =	BIT10	
74	000400	FATI =	BIT8	
75		:		
76		:LINK COMMAND REGISTER		
77		:		
78	100000	ENABLE =	BIT15	:ENABLE LINK MODULE
79	000200	MODE =	BIT7	:ENABLE MODE REGISTER
80	000100	ARAM =	BIT6	:ENABLE STATION ADDRESS RAM
81		:		
82		:LINK MODE REGISTER		
83		:		
84	100000	PROM =	BIT15	:PROMISCUOUS MODE
85	040000	ENAL =	BIT14	:ENABLE MULTICAST
86	004000	ENCR =	BIT11	:ENABLE COLLISION TEST
87	002000	ACLO =	BIT10	:ENABLE ACLO
88	000040	DRTY =	BIT5	:DISABLE RETRY LOGIC
89	000020	COLL =	BIT4	:SIMULATE A COLLISION
90	000010	DTCR =	BIT3	:DISABLE TRANSMIT CRC LOGIC
91	000004	LOOP =	BIT2	:ENABLE LOOPBACK
92				
93	000070	TRNVEC=	70	:VECTOR ADDRESS FOR THE TRANSMITTER
94	000120	RCVVEC=	120	:VECTOR ADDRESS FOR THE RECEIVER
95	000134	SANVEC=	134	:VECTOR ADDRESS FOR THE SANITY TIMER
96	000064	CSRVEC=	64	:VECTOR ADDRESS FOR CSR WRITE INTERRUPT
97	000114	DMAVEC=	114	:VECTOR ADDRESS FOR DMA DONE INTERRUPT
98	000140	PARVEC=	140	:VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
99	001000	STACK=	1000	:STACK LOCATION
100	000001	INMON=	1	:IN MICROMONITOR STATE
101	000002	INTST=	2	:IN A TEST STATE
102	000003	INERR=	3	:IN ERROR STATE
103	000001	CSRFLG=	BIT0	:CSR WRITE INTERRUPT OCCURED
104	000002	ERRFLG=	BIT1	:UNEXPECTED ERROR OCCURED
105	000004	PARFLG=	BIT2	:PARITY ERROR OCCURED
106	000010	NXMFLG=	BIT3	:NON-EXISTANT MEMORY ERROR OCCURRED
107	000020	NPRFLG=	BIT4	:NPR TIMEOUT OCCURRED
108	000040	TRNFLG=	BIT5	:TRANSMITTER INTERRUPT OCCURRED
109	000100	RCVFLG=	BIT6	:RECEIVER INTERRUPT OCCURRED
110	100000	NPRERR=	BIT15	:PCSR0 FLAG INDICATING NPR ERROR ERROR OCCURRED
111	040000	NXMERR=	BIT14	:PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
112	020000	UNIERR=	BIT13	:PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED

76MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

113	010000	PARERR= BIT12	:PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
114	004000	SIZ1K= 4000	:1K WORDS
115	010000	SIZ2K= SIZ1K*2	:2K WORDS
116	014000	SIZ3K= SIZ1K*3	:3K WORDS
117	020000	SIZ4K= SIZ2K*2	:4K WORDS
118	040000	SIZ8K= SIZ4K*2	:8K WORDS
119	020000	WCSSIZ= SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
120	020000	IOSIZ= SIZ4K	:SIZE OF I/O PAGE
121	040000	ROMSIZ= SIZ8K	:SIZE OF ROM
122	077774	LINSIZ= SIZ8K*2-4	:SIZE OF LINK MEMORY
123	000000	WCSADR= 0	:BASE ADDRESS OF WCS
124	020000	IOADR= WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
125	040000	ROMADR= IOADR+IOSIZ	:BASE ADDRESS OF ROM
126	100000	LINADR= ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
127	002756	MAXBC= 1518.	:MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
128	007777	MXMTBC= 7777	: MAXIMUM COUNT IN LINK XMIT BYTE COUNT FIELD
129	007777	MRECBC= 7777	: MAXIMUM COUNT IN LINK RECEIVE 'MLEN' FIELD
130	000004	CRCSIZ= 4	:NUMBER OF BYTES IN THE CRC
131	000000	DATERR= 0	:FLAG INDICATING DATA ERROR OCCURRED
132	000001	ADRERR= 1	:FLAG INDICATING ADDRESS ERROR OCCURRED
133			

76MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 5  
OTHER DEFINITIONS USED BY THE MICROCODE

```

134
135
136
137 000000' 106427 000340 MICROD::MTPS #PRI07 ;DISABLE INTERRUPTS
138 000004' 012737 000000 177776 MOV #0,#PCMDREG ;TURN OFF THE LINK
139 000012' 012706 001000 MOV #STACK,SP ;SETUP STACK
140 000016' 112767 000001 000620 MOVB #INMON,PCSR1 ;TELL HOST WE ARE IN MICROMONITOR
141 000024' 016737 000614 021020 MOV PCSR1,#IPCSR1
142 000032' 012737 004000 021000 MOV #DN1,#IPCSRO
143 000040' 010700 MOV PC,RO ;TELL HOST THE LOAD AND START FINISHED
144 000042' 062700 000404 ADD #ERRINT-.,RO ;GET ADDRESS OF UNEXPECTED ERROR...
145 000046' 005001 CLR R1 ;HANDLER
146 000050' 010021 10$: MOV RO,(R1)+ ;FILL ALL UNUSED VECTORS WITH TRAP...
147 000052' 012721 000340 MOV #PRI07,(R1)+ ;HANDLER
148 000056' 020127 001000 CMP R1,#1000
149 000062' 002772 BLT 10$
150
151 000064' 010700 MOV PC,RO ;SETUP PARITY TRAP VECTOR
152 000066' 062700 000462 ADD #PARINT-.,RO
153 000072' 010037 000140 MOV RO,#PARVEC
154 000076' 012737 000340 000142 MOV #PRI07,#PARVEC+2
155
156 000104' 010700 MOV PC,RO ;SETUP DMA INTERRUPT VECTOR
157 000106' 062700 000364 ADD #DMAINT-.,RO
158 000112' 010037 000114 MOV RO,#DMAVEC
159 000116' 012737 000340 000116 MOV #PRI07,#DMAVEC+2
160
161 000124' 010700 MOV PC,RO ;SETUP CSR WRITE VECTOR
162 000126' 062700 000310 ADD #CSRWRT-.,RO
163 000132' 010037 000064 MOV RO,#CSRVEC
164 000136' 012737 000200 000066 MOV #PRI04,#CSRVEC+2
165
166 000144' 010700 MOV PC,RO ;SETUP SANTITY TIMER VECTOR
167 000146' 062700 000316 ADD #TININT-.,RO
168 000152' 010037 000134 MOV RO,#SANVEC
169 000156' 012737 000240 000136 MOV #PRI05,#SANVEC+2
170
171 000164' 010700 MOV PC,RO ;SETUP TRANSMITTER VECTOR
172 000166' 062700 000414 ADD #TRNINT-.,RO
173 000172' 010037 000070 MOV RO,#TRNVEC
174 000176' 012737 000200 000072 MOV #PRI04,#TRNVEC+2
175
176 000204' 010700 MOV PC,RO ;SETUP RECEIVER VECTOR
177 000206' 062700 000360 ADD #RCVINT-.,RO
178 000212' 010037 000120 MOV RO,#RCVVEC
179 000216' 012737 000240 000122 MOV #PRI05,#RCVVEC+2
180
181 000224' 013700 021040 MOV #PCSR5,RO ;GET SWITCH PACK BITS
182 000230' 052700 176000 BIS #176000,RO ;MAP THEM INTO HOST I/O PAGE
183 000234' 006300 ASL RO ;SHIFT OVER TO POSITION CORRECTLY
184 000236' 006300 ASL RO
185 000240' 006300 ASL RO
186 000242' 062700 000004 ADD #4,RO ;PCSR2 IS PCSR0+4
187 000246' 010067 000374 MOV RO,IPCSR2 ;SAVE PCSR2 ADDRESS
188 000252' 012767 000003 000370 MOV #3,IPCSR2+2 ;HIGH ORDER BITS 17:16
189 000260' 005067 000354 CLR FLG4 ;INITIALIZE FLAG WORD
    
```

76MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 6  
 MICROD.MAC 07-APR-83 16:06 D\_MODULE MICROCODE

```

190 000264' 106427 000000      15$:  MTPS      #PRI00      ;ALLOW INTERRUPTS
191
192 000270' 005767 000344      20$:  TST        FLG4      ;WAIT FOR A COMMAND FROM HOST
193 000274' 001775      BEQ        20$
194
195 000276' 106427 000340      MTPS      #PRI07      ;RAISE CPU PRIORITY TO SERVICE COMMAND
196 000302' 032767 000001 000330  BIT        #CSRFLG,FLG4 ;DID HOST GIVE US A COMMAND?
197 000310' 001001      BNE        30$      ;YES
198 000312' 000777      BR         .         ;NO, ERROR SO JUST SIT HERE...
199                                ;FOR LACK OF ANYTHING BETTER TO DO
200
201 000314' 113700 021000      30$:  MOVB       @#IPCSRO,R0 ;GET WHAT HOST WROTE TO PCSRO
202 000320' 042700 177760      BIC       #177760,R0 ;STRIP ALL BUT COMMAND BITS
203 000324' 001004      BNE        35$      ;WAS IT THE CLEAR FUNCTION?
204 000326' 012737 000001 021020  MOV        #INMON,@#IPCSR1 ;YES, CLEAR OUT THE TEST # BITS
205 000334' 000432      BR         50$
206 000336' 022700 000017      35$:  CMP        #17,R0      ;START OPERATIONAL MICROCODE?
207 000342' 001432      BEQ        60$
208 000344' 162700 000001      SUB        #1,R0
209 000350' 010701      MOV        PC,R1      ;GET ADDRESS OF OUR COMMAND TABLE
210 000352' 062701 000240      ADD        #TBLD-.,R1
211 000356' 006300      ASL        R0
212 000360' 060001      ADD        R0,R1      ;MAKE COMMAND A BYTE OFFSET
213 000362' 061101      ADD        (R1),R1    ;USE IT TO INDEX INTO COMMAND TABLE
214 000364' 004711      JSR        PC,(R1)    ;R1 NOW HAS COMMAND ROUTINE ADDRESS
215 000366' 103404      BCS       40$      ;EXECUTE AS COMMANDED FROM HOST
216 000370' 112767 000001 000246  MOVB       #INMON,PCSR1 ;ERROR OCCURRED
217 000376' 000403      BR         45$      ;INDICATE TO HOST WE ARE BACK IN...
218 000400' 112767 000003 000236 40$:  MOVB       #INERR,PCSR1 ;MICROMONITR
219 000406' 016737 000232 021020 45$:  MOV        PCSR1,@#IPCSR1 ;INDICATE TO HOST ERROR OCCURRED
220 000414' 012737 004000 021000  MOV        #DNI,@#IPCSRO
221 000422' 005067 000212      50$:  CLR        FLG4      ;TELL HOST THIS MICROTEST FINISHED
222 000426' 000716      BR         15$      ;RESET FLAG WORD
223                                ;GO WAIT FOR ANOTHER COMMAND
224 000430' 005000      60$:  CLR        R0
225 000432' 000137 040006      JMP        @#40006    ;FAKE SELF TEST RESULTS
226                                ;START OPERATIONAL MICROCODE
227 000436' 052767 000001 000174  CSRWRT:  BIS        #CSRFLG,FLG4 ;INDICATE A CSR WRITE INTERRUPT OCCURED
228 000444' 000002      RTI
229
230 000446' 052767 000002 000164  ERRINT:  BIS        #ERRFLG,FLG4 ;INDICATE A UNEXPECTED INTERRUPT OCCURED
231 000454' 012737 020000 021000  MOV        #UNIERR,@#IPCSRO ;TELL HOST AN UNEXPECTED INTERRUPT
232                                ;HAPPENED
233 000462' 000777      BR         .         ;JUST SIT HERE AND SPIN WHEELS
234                                ;COUNT ON HOST TO TIME OUT
235
236 000464' 005267 000152      TIMINT:  INC        SANTIM ;COUNT TICKS AS THEY OCCUR
237 000470' 000002      RTI
238
239 000472' 013767 021002 000156  DMAINT:  MOV        @#DMACSR,DMDONE ;GET DMA STATUS
240 000500' 032767 040000 000150  BIT        #BIT14,DMDONE ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
241 000506' 001404      BEQ        10$      ;NO
242 000510' 012737 040000 021000  MOV        #NXMERR,@#IPCSRO ;YES, TELL HOST A NON-EXISTANT MEMORY
243                                ;LOCATION WAS ADDRESSED
244 000516' 000407      BR         20$
245 000520' 032767 100000 000130 10$:  BIT        #BIT15,DMDONE ;DID A NPR TIMEOUT OCCUR?
    
```

76MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 7  
MICROD.MAC 07-APR-83 16:06 D\_MODULE MICROCODE

```

246 000526' 001407          BEQ      3CS          :NO
247 000530' 012737 100000 021000      MOV      #NPRERR,#IPCSRO :TELL HOST NPR TIMEOUT HAPPENED
248 000536' 012737 100000 021002 208:  MOV      #BIT15,#DMACSR :CLEAR THE INTERRUPT IN THE DMA ENGINE
249 000544' 000777          BR          .          :SIT HERE AND SPIN WHEELS
250 000546' 0000C2          308:  RTI
251
252
253 000550' 052767 000004 000062 PARINT: BIS      #PARFLG,FLG4      :SET PARITY ERROR OCCURRED
254 000556' 012737 010000 021000      MOV      #PARERR,#IPCSRO :TELL HOST A LINK MEMORY PARITY ERROR
255                                     BR          .          :OCCURRED
256 000564' 000777          BR          .          :SIT HERE AND SPIN WHEELS
257
258 000566' 005737 021044          RCVINT: TST     @#LRBUF      :READ BUFFER DONE REGISTER...
259                                     :WHICH CLEARS THE INTERRUPT
260 000572' 052767 000100 00J040          BIS      #RCVFLG,FLG4      :SET RECEIVER INTERRUPT OCCURRED
261 000600' 000002          RTI
262
263 000602' 052767 000040 000030 TRNINT: BIS      #TRNFLG,FLG4      :SET TRANSMITTER INTERRUPT OCCURRED
264 000610' 000002          RTI
265
266 000612' 000056          TBLD:  .WORD  MICD1-.      :TRANSMITTER DONE TEST
267 000614' 000236          .WORD  MICD2-.      :RECEIVER DONE TESTS
268 000616' 000432          .WORD  MICD3-.      : DATA BYTE FRAMING TEST
269 000620' 000734          .WORD  MICD4-.      : DATA WORD FRAMING TEST
270 000622' 001244          .WORD  MICD5-.      : DATA PATH PATTERN
271 000624' 001604          .WORD  MICD6-.      : STATUS MUX TEST
272 000626' 002046          .WORD  MICD7-.      : LINK BYTE COUNT TEST
273 000630' 002354          .WORD  MICD8-.      :LINK MEMORY ARBITRATION TEST
274 000632' 003014          .WORD  MICD9-.      :LINK BYTE COUNTER MAXIMUM TEST
275 000634' 003206          .WORD  MICD10-.     :FIFO TEST
276 000636' 003460          .WORD  MICD11-.     :LINK MEMORY ADDRESS TEST
277
278 000640' 000000          FLG4:  .WORD  0          :FLAG WORD
279 000642' 000000          SANTIM: .WORD  0        :COUNT FOR SANITY TIMER
280 000644' 000000          PCSR1: .WORD  0        :COPY OF WHAT GOES TO PCSR1
281 000646' 000000 000000          IPCSR2: .WORD  0,0      :ADDRESS IN HOST MEMORY FOR PCSR2
282 000652' 000000 000000          PCBADR: .WORD  0,0      :ADDRESS IN HOST MEMORY FOR PCB
283 000656' 000000          DM DONE: .WORD  0      :
284 000660' 000000          RBUF:  .WORD  0        :POINTER TO RECIEVE BUFFER
285 000662' 000000          TBUF:  .WORD  0        :POINTER TO XMIT BUFFER
286 000664' 000000          DBUF:  .WORD  0        :POINTER TO DMA ENGINE BUFFER
287 000666' 000000          MBUF:  .WORD  0        :POINTER TO MICROCPU DMA BUFFER
288

```

76MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 8  
D\_MODULE MICROCODE

```

289 000670' 112767 000002 177746 MICD1: MOVB #INTST,PCSR1 ;TELL HOST WE ARE TESTING
290 000676' 016737 177742 021020 MOV PCSR1,@#IPCSR1
291 000704' 012703 177777 MOV #177777,R3 ;FILL RECIEVE BUFFER WITH 1'S
292 000710' 012700 100000 MOV #LINADR,RO ;RECIEVE BUFFER STARTS HERE
293 000714' 010067 177740 MOV RO,#RBUF
294 000720' 010320 10S: MOV R3,(RO)+
295 000722' 020027 104000 CMP RO,#LINADR+SIZ1K ;FILL ENTIRE BUFFER
296 000726' 103774 BLO 10S
297 000730' 005003 CLR R3 ;FILL XMIT BUFFER WITH 0'S
298 000732' 010067 177724 MOV RO,TBUPF ;XMIT BUFFER STARTS 1K AWAY FROM RECIEVE
299 000736' 010320 20S: MOV R3,(RO)+
300 000740' 020027 110000 CMP RO,#LINADR+SIZ2K
301 000744' 103774 BLO 20S
302
303 000746' 012737 100200 177776 MOV #MODE!ENABLE,@#CMDREG ;ENABLE LINK MODULE AND SELECT MODE REG
304 000754' 012737 100004 177774 MOV #PROM!LOOP,@#MODREG ;SET PROMISCUIOUS MODE AND ENABLE LOOPBACK
305
306 000762' 016701 177674 MOV TBUPF,R1 ;POINT TO XMIT BUFFER
307 000766' 005021 CLR (R1)+ ;CLEAR OUT STATUS WORD
308 000770' 012721 002752 MOV #MAXBC-CRCSIZ,(R1)+ ;SET BYTE COUNT TO MAXIMUM ALLOWED
309 000774' 005037 021034 CLR @#CLRIFIF ;CLEAR THE FIFO
310 001000' 005067 177634 CLR FLG4 ;CLEAR THE INTERRUPT FLAG
311 001004' 016737 177650 021032 MOV RBUF,@#LFRBUF ;TELL LINK WHERE RECIEVE BUFF IS
312 001012' 016737 177644 021030 MOV TBUPF,@#LTAC ;TELL LINK WHERE TRANSMIT BUFF IS...
313 ;WHICH WILL START A XMIT OPERATION
314 001020' 106427 000140 MTPS #PRI03 ;ALLOW XMITTER TO INTERRUPT
315 001024' 032767 000040 177606 30S: BIT #TRNFLG,FLG4 ;WAIT FOR XMIT INTERRUPT
316 001032' 001774 BEQ 30S
317 001034' 106427 000340 MTPS #PRI07 ;DISABLE INTERRUPTS
318 001040' 112767 000001 177577 MOVB #1,PCSR1+1 ;TELL HOST THIS IS TEST #1
319 001046' 000241 CLC
320 001050' 000207 RTS PC
321

```

76MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 9  
 MICROD.MAC 07-APR-83 16:06 D\_MODULE MICROCODE

```

322 001052' 112767 000002 177564 MICD2: MOVB #INTST,PCSR1 ;TELL HOST WE ARE TEST #1
323 001060' 016737 177560 021020 MOV PCSR1,@IPCSR1
324 001066' 012703 177777 MOV #177777,R3 ;FILL RECIEVE BUFFER WITH 1'S
325 001072' 012700 100000 MOV #LINADR,R0 ;RECIEVE BUFFER STARTS HERE
326 001076' 010067 177556 MOV R0,R2BUF
327 001102' 010320 108: MOV R3,(R0)+
328 001104' 020027 104000 CMP R0,#LINADR+SIZ1K ;FILL ENTIRE BUFFER
329 001110' 103774 BLO 108
330 001112' 005003 CLR R3 ;FILL XMIT BUFFER WITH 0'S
331 001114' 010067 177542 MOV R0,TBUF ;XMIT BUFFER STARTS 1K AWAY FROM RECIEVE
332 001120' 010320 208: MOV R3,(R0)+
333 001122' 020027 110000 CMP R0,#LINADR+SIZ2K
334 001126' 103774 BLO 208
335
336 001130' 012737 100200 177776 MOV #MODE!ENABLE,@#CMDREG ;ENABLE LINK MODULE AND SELECT MODE REG
337 001136' 012737 100004 177774 MOV #PROM!LOOP,@#MODREG ;SET PROMISCUIOUS MODE AND ENABLE LOOPBACK
338
339 001144' 016701 177512 MOV TBUF,R1 ;POINT TO XMIT BUFFER
340 001150' 005021 CLR (R1)+ ;CLEAR OUT STATUS WORD
341 001152' 012721 002752 MOV #MAXBC-CRCSIZ,(R1)+ ;SET BYTE COUNT TO MAXIMUM ALLOWED
342 001156' 005037 021034 CLR @#CLRFIF ;CLEAR THE FIFO
343 001162' 005067 177452 CLR FLG4 ;CLEAR THE INTERRUPT FLAG
344 001166' 016737 177466 021032 MOV RBUF,@#LFRBUF ;TELL LINK WHERE RECIEVE BUFF IS
345 001174' 016737 177462 021030 MOV TBUF,@#LTAC ;TELL LINK WHERE TRANSMIT BUFF IS...
346 ;WHICH WILL START A XMIT OPERATION
347 001202' 106427 000200 MTPS #PRI04 ;ALLOW RECEIVER TO INTERRUPT
348 001206' 032767 000100 177424 308: BIT #RCVFLG,FLG4 ;WAIT FOR RECEIVE INTERRUPT
349 001214' 001774 BEQ 308
350
351 001216' 106427 000140 MTPS #PRI03 ; ALLOW XMTR TO INTERRUPT
352 001222' 032767 000040 177410 408: BIT #TRNFLG,FLG4 ; WAIT FOR XMIT INTERRUPT
353 001230' 001774 BEQ 408
354
355 001232' 106427 000340 MTPS #PRI07 ;DISABLE INTERRUPTS
356 001236' 112767 000002 177401 MOVB #2,PCSR1+1 ;TELL HOST THIS IS TEST #2
357 001244' 000241 CLC
358 001246' 000207 RTS PC

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 10  
D\_MODULE MICROCODE

```

359
360      : DATA BYTE FRAMING TEST
361      :
362      : THIS IS 'MICROCODE' FOR DATA BYTE FRAMING TEST. FILLS XMIT BUFFER
363      : WITH PATTERN 00000001111111 (BINARY) AND TRANSMITS OVER LOOPBACK.
364      : CHECKS RECEIVE BUFFER FOR SAME PATTERN. REPORTS ERRENT PATTERN,
365      : OFFSET FROM FRONT OF BUFFER, AND RECEIVE BUFFER STATUS WORD TO
366      : HOST
367      :
368      : SBTTL  MODULE D, MICROTEST #3
369
370 001250' 112767 000002 177366 MICD3: MOVB  #INTST,PCSR1      ; TELL HOST WE ARE TESTING
371 001256' 016737 177362 021020      MOV   PCSR1,@#IPCSR1
372
373      : *****
374      : ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
375      : *****
376
377 001264' 012700 100000      MOV   #LINADR,R0      ; RECEIVE BUFFER STARTS HERE
378 001270' 010067 177364      MOV   R0,RBUF
379 001274' 005020 10S:      CLR   (R0)+          ; FILL RECEIVE BUFFER WITH ZEROS
380 001276' 020027 104000      CMP   R0,#LINADR+SIZ1K ; FILL ENTIRE BUFFER
381 001302' 103774      BLO  10S
382
383      : *****
384      : ***** FILL TRANSMIT BUFFER WITH TEST PATIERN *****
385      : *****
386
387
388 001304' 012703 000377      MOV   #0377,R3      ; WORST CASE FOR CLOCKING
389 001310' 010067 177346      MOV   R0,TBUF
390 001314' 010320 20S:      MOV   R3,(R0)+      ; FILL XMIT BUFFER WITH PATTERN
391 001316' 020027 110000      CMP   R0,#LINADR+SIZ2K ; STOP AT TOP
392 001322' 103774      BLO  20S
393
394
395      : *****
396      : ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
397      : *****
398
399 001324' 012737 100200 177776      MOV   #MODE!ENABLE,@#CMDREG ; ENABLE LINK, SELECT MODE REG
400 001332' 012737 100004 177774      MOV   #PROM!LOOP,@#MODREG  ; PROM MODE AND LOOPBACK
401
402 001340' 016701 177316      MOV   TBUF,R1      ; POINT AT XMIT BUFFER
403 001344' 005021      CLR   (R1)+        ; CLEAR OUT STATUS WORD
404 001346' 012721 002752      MOV   #MAXBC-CRCSIZ,(R1)+ ; SET BYTE COUNT TO MAX ALLOWED
405 001352' 005037 021034      CLR   @#CLRFIF     ; CLEAR THE FIFO
406 001356' 005067 177256      CLR   FLG4         ; CLEAR INTERRUPT FLAG
407 001362' 016737 177272 021032      MOV   RBUF,@#LFRBUF  ; TELL UNA WHERE RECEIVE BUFF IS
408 001370' 016737 177266 021030      MOV   TBUF,@#LTAC   ; TELL UNA WHERE XMIT BUFF IS
409
410 001376' 106427 000140      MTPS  #PRI03        ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
411 001402' 032767 000100 177230 30S:  BIT   #RCVFLG,FLG4  ; WAIT FOR RECEIVER INTERRUPT
412 001410' 001774      BEQ   30S
413
414 001412' 032767 000040 177220 35S:  BIT   #TRMFLG,FLG4  ;WAIT FOR TRANSMIT INTERRUPT TOO

```



77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 11  
MODULE D, MICROTEST #3

```

415 001420' 001774          BEQ      35$
416
417 001422' 106427 000340    MTPS    #PRI07          : DISABLE INTERRUPTS
418
419
420
421      : *****
422      : ***** VERIFY PATTERN IN RECEIVE BUFFER *****
423      : *****
424 001426' 012700 100000      MOV      #LINADR,R0      : VERIFY RECEIVE BUFFER CONTENT
425 001432' 012005          MOV      (R0)+,R5        : SAVE STATUS IN CASE ERROR
426 001434' 005004          CLR      R4              : TRACK OFFSET IN CASE ERROR
427 001436' 062700 000002      ADD      #2,R0           : DON'T NEED 'LENGTH' IN BUFFER
428
429 001442' 011001          40$:  MOV      (R0),R1      : READ THE BUFFER
430 001444' 020103          CMP      R1,R3          : R3 CONTAINS ORIGINAL PATTERN
431 001446' 001012          BNE     70$            : GO TO ERROR EXIT
432 001450' 005200          INC     R0              : COULDN'T BUMP TILL AFTER TEST
433 001452' 005204          INC     R4              : TRACK OFFSET FROM BUFFER START
434 001454' 022704 002752      CMP      #MAXBC-CRCSIZ,R4 : COMPARE ALL BUFFER ENTRIES
435 001460' 001370          BNE     40$
436 001462' 000241          CLC                      : TELL MICROMONITOR SUCCESS
437
438      : *****
439      : ***** EXIT POINT FOR NO ERROR CONDITION *****
440      : *****
441
442 001464' 112767 000003 177153 50$:  MOVB    #3,PCSR1+1      : TELL HOST TEST 3 DONE
443 001472' 000207          RTS      PC              : RETURN TO SENDER
444
445
446      : *****
447      : ***** EXIT POINT FOR ERROR CONDITION *****
448      : *****
449
450 001474' 016737 177146 021010 70$:  MOV      IPCSR2,&#MDMA0      : PICK UP ADDRESS OF PCBB
451 001502' 016737 177142 021012      MOV      IPCSR2+2,&#MDMA1
452 001510' 013700 021014          MOV      &#MDMAR0,R0      : R0=CONTENTS OF PCSR2
453 001514' 013702 021014          MOV      &#MDMAR0,R2      : R2=CONTENTS OF PCSR3
454 001520' 010037 021010          MOV      R0,&#MDMA0        : POINT TO PCBB+0
455 001524' 010237 021012          MOV      R2,&#MDMA1
456
457 001530' 010537 021026          MOV      R5,&#MDMAW0      : WRITE STATUS WORD TO HOST
458 001534' 010337 021026          MOV      R3,&#MDMAW0      : WRITE ORIGINAL PATTERN BACK
459 001540' 010137 021026          MOV      R1,&#MDMAW0      : WRITE ERRENT PATTERN TO HOST
460 001544' 010437 021026          MOV      R4,&#MDMAW0      : WRITE ERROR OFFSET TO HOST
461
462 001550' 000261          SEC                      : TELL MICROMONITOR ERROR OCCURRED
463 001552' 000744          BR      50$             : GO EXIT THROUGH NORMAL
464

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 12  
MODULE D, MICROTEST #3

```

465
466      ; DATA WORD FRAMING TEST
467      ;
468      ; THIS IS 'MICROCODE' FOR DATA WORD FRAMING TEST. FILLS XMIT BUFFER
469      ; WITH PATTERN 00000000000000011111111111111111 (BINARY) AND TRANSMITS
470      ; OVER THE LOOPBACK.
471      ;
472      ; CHECKS RECEIVE BUFFER FOR SAME PATTERN. REPORTS ERRENT PATTERN,
473      ; OFFSET FROM FRONT OF BUFFER, AND RECEIVE BUFFER STATUS WORD TO
474      ; HOST
475      ;
476      .SBTTL  MODULE D,MICROTEST #4
477      ;
478      ; *****
479      ; ***** TELL HOST WE ARE BUSY *****
480      ; *****
481
482 001554' 112767 000002 177062 MICD4:  MOV    #INTST,PCSR1      ; TELL HOST WE ARE TESTING
483 001562' 016737 177056 021020      MOV    PCSR1,#IPCSR1
484
485
486      ; *****
487      ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
488      ; *****
489
490 001570' 012700 100000      MOV    #LINADR,R0      ; RECEIVE BUFFER STARTS HERE
491 001574' 010067 177060      MOV    R0,RBUF
492 001600' 005020      10$:  CLR    (R0)+      ; FILL RECEIVE BUFFER WITH ZEROS
493 001602' 020027 104000      CMP    R0,#LINADR+SIZ1K ; FILL ENTIRE BUFFER
494 001606' 103774      BLO   10$
495
496
497      ; *****
498      ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
499      ; *****
500
501 001610' 012703 177777      MOV    #177777,R3      ; WORST CASE FOR CLOCKING
502 001614' 010067 177042      MOV    R0,TBUF        ; SAVE COPY OF ADDRESS
503 001620' 010320      20$:  MOV    R3,(R0)+      ; FILL XMIT BUFFER WITH PATTERN
504 001622' 005103      COM    R3              ; FLIP IT OVER
505 001624' 020027 110000      CMP    R0,#LINADR+SIZ2K ; STOP AT TOP
506 001630' 103773      BLO   20$
507
508
509      ; *****
510      ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
511      ; *****
512
513 001632' 012737 100200 177776      MOV    #MODE!ENABLE,#CMDREG ; ENABLE LINK, SELECT MODE REG
514 001640' 012737 100004 177774      MOV    #PROM!LOOP,#MODREG  ; PROM MODE AND LOOPBACK
515
516 001646' 016701 177010      MOV    TBUF,R1        ; POINT AT XMIT BUFFER
517 001652' 005021      CLR    (R1)+          ; CLEAR OUT STATUS WORD
518 001654' 012721 002752      MOV    #MAXBC-CRCSIZ,(R1)+ ; SET BYTE COUNT TO MAX ALLOWED
519 001660' 005037 021034      CLR    @CLRFIF        ; CLEAR THE FIFO
520 001664' 005067 176750      CLR    FLG4          ; CLEAR INTERRUPT FLAG

```

77MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 13  
MICROD.MAC 07-APR-83 16:06 MODULE D,MICROTEST #4

```

521 001670' 016737 176764 021032      MOV      RBUF, @LFRBUF      ; TELL UNA WHERE RECEIVE BUFF IS
522 001676' 016737 176760 021030      MOV      TBUF, @LTAC       ; TELL UNA WHERE XMIT BUFF IS
523
524 001704' 106427 000140                MTPS     #PRI03            ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
525 001710' 032767 000100 176722 30$:  BIT      #RCVFLG, FLG4     ; WAIT FOR RECEIVER INTERRUPT
526 001716' 001774                        BEQ
527
528 001720' 032767 000040 176712 35$:  BIT      #TRNFLG, FLG4     ; WAIT FOR XMIT INTERRUPT TOO
529 001726' 001774                        BEQ      35$
530
531 001730' 106427 000340                MTPS     #PRI07            ; DISABLE INTERRUPTS
532
533
534 ; *****
535 ; ***** VERIFY THE CONTENTS OF RECEIVE BUFFER *****
536 ; *****
537
538 001734' 012700 100000                MOV      #LINADR, R0       ; VERIFY RECEIVE BUFFER CONTENT
539 001740' 012005                        MOV      (R0)+, R5        ; SAVE STATUS IN CASE ERROR
540 001742' 005004                        CLR      R4               ; TRACK OFFSET IN CASE ERROR
541 001744' 062700 000002                ADD      #2, R0           ; DON'T NEED 'LENGTH' IN BUFFER
542
543 001750' 005003                        CLR      R3               ; NEED A ZERO
544 001752' 005103 40$:  COM      R3               ; FLIP IT OVER
545 001754' 011001                        MOV      (R0), R1        ; READ DATA BACK
546 001756' 020103                        CMP      R1, R3
547 001760' 001012                        BNE     70$              ; GO TO ERROR EXIT
548 001762' 005200                        INC      R0               ; NEW ADDRESS AFTER TESTING
549 001764' 005204                        INC      R4               ; NEXT OFFSET
550 001766' 022704 002752                CMP      #MAXBC-CRCSIZ, R4
551 001772' 103767                        BLO     40$              ; END OF LOOP
552 001774' 000241                        CLC
553 ; TELL MICROMONITOR SUCCESS
554 ; *****
555 ; ***** FALLTHROUGH EXIT IF NO ERROR *****
556 ; *****
557
558 001776' 112767 000004 176641 50$:  MOVB    #4, PCSR1+1     ; TELL HIM TEST FINISHED
559 002004' 000207                        RTS      PC               ; RETURN TO SENDER
560
561 ; *****
562 ; ***** ERROR EXIT *****
563 ; *****
564
565 002006' 016737 176634 021010 70$:  MOV      IPCSR2, @#DMA0     ; PICK UP ADDRESS OF PCBB
566 002014' 016737 176630 021012      MOV      IPCSR2+2, @#DMA1
567 002022' 013700 021014                MOV      @#DMA0, R0       ; R0=CONTENTS OF PCSR2
568 002026' 013702 021014                MOV      @#DMA0, R2       ; R2=CONTENTS OF PCSR3
569 002032' 010037 021010                MOV      R0, @#DMA0       ; POINT TO PCBB+0
570 002036' 010237 021012                MOV      R2, @#DMA1
571
572 002042' 010537 021026                MOV      R5, @#DMA0       ; WRITE STATUS WORD TO HOST
573 002046' 010337 021026                MOV      R3, @#DMA0       ; WRITE ORIGNAL TO HOST
574 002052' 010137 021026                MOV      R1, @#DMA0       ; WRITE ERRENT PATTERN TO HOST
575 002056' 010437 021026                MOV      R4, @#DMA0       ; WRITE ERROR OFFSET TO HOST
576

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 14  
MODULE D,MICROTEST #4

577 002062' 000261  
578 002064' 000744  
579

SEC  
BR 508

:TELL MICROMONITOR ERROR OCCURRED  
: GO EXIT THROUGH NORMAL

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 15  
MODULE D,MICROTEST #4

580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635

```

: DATA PATH PATTERN TEST
:
: THIS IS 'MICROCODE' FOR DATA PATH PATTERN TEST. RETRIEVES DATA PATTERN
: FROM HOST MEMORY. FILLS XMIT BUFFER WITH PATTERN AND SENDS DATAGRAM
: OVER THE LOOPBACK.
:
: CHECKS RECEIVE BUFFER FOR SAME PATTERN. REPORTS ERRENT PATTERN,
: OFFSET FROM FRONT OF BUFFER, AND RECEIVE BUFFER STATUS WORD TO
: HOST
.SBTTL MODULE D,MICROTEST #5
: *****
: ***** TELL HOST WE ARE BUSY *****
: *****
597 002066' 112767 000002 176550 MICD5: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
598 002074' 016737 176544 021020 MOV PCSR1,#IPCSR1
: *****
: ***** RETRIEVE PATTERN FROM HOST MEMORY *****
: *****
604 002102' 016737 176540 021010 MOV IPCSR2,#MDMA0 ; SET TO GET HOST PCBB ADDRESS
605 002110' 016737 176534 021012 MOV IPCSR2+2,#MDMA1
606 002116' 013700 021014 MOV #MDMAR0,R0 ; R0 NOW CONTAINS PCBB LOW
607 002122' 013701 021014 MOV #MDMAR0,R1 ; R1 NOW CONTAINS PCBB HIGH
608 002126' 010037 021010 MOV R0,#MDMA0 ; POINT AT PCBB
609 002132' 010137 021012 MOV R1,#MDMA1
610 002136' 013703 021014 MOV #MDMAR0,R3 ; R3 NOW HOLDS DATA PATTERN
: *****
: ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
: *****
617 002142' 012700 100000 MOV #LINADR,R0 ; RECEIVE BUFFER STARTS HERE
618 002146' 010067 176506 MOV R0,RBUF
619 002152' 005020 104000 10$: CLR (R0)+ ; FILL RECEIVE BUFFER WITH ZEROS
620 002154' 020027 104000 CMP R0,#LINADR+SIZ1K ; FILL ENTIRE BUFFER
621 002160' 103774 BLO 10$
: *****
: ***** FILL XMIT BUFFER WITH TEST PATTERN *****
: *****
628 002162' 010067 176474 20$: MOV R0,TBUF ; SAVE COPY OF ADDRESS
629 002166' 010320 MOV R3,(R0)+ ; FILL XMIT BUFFER WITH PATTERN
630 002170' 020027 110000 CMP R0,#LINADR+SIZ2K ; STOP AT TOP
631 002174' 103774 BLO 20$
: *****
: ***** SET UP LINK FOR DATAGRAM LOOPBACK *****

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 16  
MODULE D,MICROTEST #5

```

636 ; *****
637
638 002176' 012737 100200 177776      MOV      #MODE!ENABLE,@#CMDREG      ; ENABLE LINK, SELECT MODE REG
639 002204' 012737 100004 177774      MOV      #PROM!LOOP,@#MODREG       ; PROM MODE AND LOOPBACK
640
641 002212' 016701 176444              MOV      TBUF,R1                    ; POINT AT XMIT BUFFER
642 002216' 005021              CLR      (R1)+                      ; CLEAR OUT STATUS WORD
643 002220' 012721 002752              MOV      #MAXBC-CRCSIZ,(R1)+       ; SET BYTE COUNT TO MAX ALLOWED
644 002224' 005037 021034              CLR      @#CLRFIF                  ; CLEAR THE FIFO
645 002230' 005067 176404              CLR      FLG4                      ; CLEAR INTERRUPT FLAG
646 002234' 016737 176420 021032      MOV      RBUF,@#LFRBUF              ; TELL UNA WHERE RECEIVE BUFF IS
647 002242' 016737 176414 021030      MOV      TBUF,@#LTAC                ; TELL UNA WHERE XMIT BUFF IS
648
649 002250' 106427 000140              MTPS    #PRI03                     ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
650 002254' 032767 000100 176356 30%:  BIT      #RCVFLG,FLG4              ; WAIT FOR RECEIVER INTERRUPT
651 002262' 001774                      BEQ
652
653 002264' 032767 000040 176346 35%:  BIT      #TRNFLG,FLG4              ;WAIT FOR XMIT INTERRUPT TOO
654 002272' 001774                      BEQ      35%
655
656 002274' 106427 000340              MTPS    #PRI07                     ; DISABLE INTERRUPTS
657
658
659 ; *****
660 ; ***** VERIFY THE CONTENTS OF RECEIVE BUFFER *****
661 ; *****
662
663 002300' 012700 100000              MOV      #LINADR,R0                 ; VERIFY RECEIVE BUFFER CONTENT
664 002304' 012002              MOV      (R0)+,R2                   ; SAVE STATUS IN CASE ERROR
665 002306' 005004              CLR      R4                         ; TRACK OFFSET IN CASE ERROR
666 002310' 062700 000002              ADD      #2,R0                      ; DON'T NEED 'LENGTH' IN BUFFER
667
668 002314' 011005              40%:  MOV      (R0),R5                 ; READ DATA BACK
669 002316' 020503              CMP      R5,R3                      ; R3 HOLDS ORIGINAL PATTERN
670 002320' 001012              BNE      70%                        ; GO TO ERROR EXIT
671 002322' 005200              INC      R0                          ; NEW ADDRESS AFTER TESTING
672 002324' 005204              INC      R4                          ; NEXT OFFSET
673 002326' 022704 002752              CMP      #MAXBC-CRCSIZ,R4
674 002332' 103770              BLO      40%                        ; END OF LOOP
675 002334' 000241              CLC                                  ;TELL MICROMONITOR SUCCESS
676
677 ; *****
678 ; ***** FALLTHROUGH EXIT IF NO ERROR *****
679 ; *****
680
681 002336' 112767 000005 176301 50%:  MOVB    #5,PCSR1+1                 ; EXIT HERE IF NO ERROR
682 002344' 000207                      RTS      PC                          ; RETURN TO SENDER
683
684
685 ; *****
686 ; ***** ERROR EXIT *****
687 ; *****
688
689 002346' 016737 176274 021010 70%:  MOV      IPCSR2,@#MDMA0              ; SET TO GET HOST PCBB ADDRESS
690 002354' 016737 176270 021012      MOV      IPCSR2+2,@#MDMA1
691 002362' 013700 021014      MOV      @#MDMAR0,R0                ; R0 NOW CONTAINS PCBB LOW

```

77MICROD - MICROCODE MODULE D    MACY11 30A(1052) 07-APR-83 16:50 PAGE 17  
 MICROD.MAC    07-APR-83 16:06    MODULE D,MICROTEST #5

692	002366'	013701	021014	MOV	@#MDMAR0,R1	: R1 NOW CONTAINS PCBB HIGH
693						
694	002372'	062700	000002	ADD	#2,R0	: INDEX DOWN TO PCBB+2
695	002376'	005501		ADC	R1	
696						
697	002400'	010037	021010	MOV	R0,@#MDMA0	: POINT TO PCBB+0
698	002404'	010137	021012	MOV	R1,@#MDMA1	
699						
700	002410'	010237	021026	MOV	R2,@#MDMA0	: WRITE STATUS WORD TO HOST
701	002414'	010537	021026	MOV	R5,@#MDMA0	: WRITE ERRENT PATTERN TO HOST
702	002420'	010437	021026	MOV	R4,@#MDMA0	: WRITE ERROR OFFSET TO HOST
703						
704	002424'	000261		SEC		: TELL MICROMONITOR ERROR OCCURRED
705	002426'	000743		BR	508	: GO EXIT THROUGH NORMAL
706						

```

707
708      ; STATUS MUX VERIFICATION TEST
709      :
710      : THIS TEST INSURES THAT THE STATUS MULTIPLEXER WRITES INTO THE FIRST
711      : TWO LOCATIONS TO THE LINK TRANSMIT BUFFER. THE DEUNA PROCESSOR WILL
712      : FILL A TRANSMIT BUFFER AND A RECEIVE BUFFER AND WILL TRANSMIT A
713      : DATAGRAM OVER THE LOOPBACK. THE FIRST TWO WORDS OF THE RECEIVE BUFFER
714      : WILL BE COPIED TO HOST MEMORY FOR VERIFICATION.
715      :
716      : HOST WILL VERIFY THAT PREDETERMINED BITS OF THE STATUS WORD ARE ZERO.
717      :
718      :
719      : *****
720      : ***** TELL HOST WE ARE BUSY *****
721      : *****
722
723 002430' 112767 000002 176206 MICD6:  MOVB  #INTST,PCSR1  ; TELL HOST WE ARE TESTING
724 002436' 016737 176202 021020      MOV   PCSR1,@#IPCSR1
725
726
727      : *****
728      : ***** FILL RECEIVE BUFFER *****
729      : *****
730
731 002444' 012703 177777      MOV   #177777,R3      ; FILL RECEIVE BUFFER WITH ONES
732 002450' 012700 100000      MOV   #LINADR,R0     ; RECEIVE BUFFER STARTS HERE
733 002454' 010067 176200      MOV   R0,RBUF        ; SAVE A COPY
734
735 002460' 010320 104000 10$:  MOV   R3,(R0)+        ; FILL THE BUFFER
736 002462' 020027 104000      CMP   R0,#LINADR+SIZ1K
737 002466' 103774      BLO  10$
738
739      : *****
740      : ***** FILL TRANSMIT BUFFER WITH ZERO *****
741      : *****
742
743 002470' 010067 176166      MOV   R0,TBUF        ; SAVE A COPY OF TRANSMIT ADDRESS
744 002474' 005003      CLR   R3              ; NEED A ZERO
745 002476' 010320 110000 20$:  MOV   R3,(R0)+        ; XMIT BUFFER STARTS 1K FROM RECEIVE
746 002500' 020027      CMP   R0,#LINADR+SIZ2K
747 002504' 103774      BLO  20$
748
749
750      : *****
751      : ***** FILL DATAGRAM FOR DATAGRAM LOOPBACK *****
752      : *****
753
754 002506' 012737 100200 177776  MOV   #MODE!ENABLE,@#CMDREG  ; ENABLE LINK MODE, SEL MODE REG
755 002514' 012737 100004 177774  MOV   #PROM!LOOP,@#MODREG   ; PROMIS, ENABLE LOOPBACK
756
757 002522' 016701 176134      MOV   TBUF,R1        ; POINT TO XMIT BUFFER
758 002526' 005103      COM   R3              ; NEED SOME 1'S
759 002530' 010321      MOV   R3,(R1)+        ; BACKGROUND- SHOW STATUS OVLAY
760 002532' 012721 002752      MOV   #MAXBC-CRCSIZ,(R1)+  ; SET BYTE COUNT TO MAX ALLOWED
761 002536' 005037 021034      CLR   @#CLRFIF       ; CLEAR THE INTERRUPT FLAG
762 002542' 005067 176072      CLR   FLG4           ; CLEAR THE INTERRUPT FLAG

```



77MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 19  
MICROD.MAC 07-APR-83 16:06 MODULE D.MICROTEST #5

```

763 002546' 016737 176106 021032      MOV      RBUF, @#LFRBUF      ; TELL LINK WHERE RECEIVE IS
764 002554' 016737 176102 021030      MOV      TBUF, @#LTAC       ; TELL LINK WHERE XMIT IS
765 002562' 106427 000140                MTPS     #PRI03             ; WAIT FOR AN INTERRUPT
766 002566' 032767 000G40 176044 30$:   BIT      #TRNFLG, FLG4      ; WAIT FOR XMIT INTERRUPT
767 002574' 001774                BEQ      30$
768 002576' 032767 000100 176034 35$:   BIT      #RCVFLG, FLG4      ;WAIT FOR RECEIVER BEFORE GOING ANY FURTHER
769 002604' 001774                BEQ      35$

```

770

771

772

773

774

775

776

777 002606' 016700 176050

778 002612' 012003

779 002614' 011004

780

781

782

783

784

785

786 002616' 016737 176024 021010

787 002624' 016737 176020 021012

788 002632' 013700 021014

789 002636' 013702 021014

790 002642' 010037 021010

791 002646' 010237 021012

792

793 002652' 010337 021026

794 002656' 010437 021026

795

796 002662' 112767 000006 175755

797 002670' 000241

798 002672' 000207

799

800

```

: *****
: ***** GET STATUS WORDS FROM TRANSMIT BUFFER *****
: *****

```

```

MOV      TBUF, R0          ; POINT AT XMIT BUFFER
MOV      (R0)+, R3         ; R5 NOW HOLDS TX0 STATUS WORD
MOV      (R0), R4          ; R6 NOW HOLDS TX1 STATUS WORD

```

```

: *****
: ***** GET HOST MEMORY ADDRESS AND WRITE STATUS WORDS *****
: *****

```

```

MOV      IPCSR2, @#MDMA0   ; PICK UP ADDRESS OF PCBB
MOV      IPCSR2+2, @#MDMA1
MOV      @#MDMAR0, R0      ; R0=CONTENTS OF PCSR2
MOV      @#MDMAR0, R2      ; R2=CONTENTS OF PCSR3
MOV      R0, @#MDMA0       ; POINT AT PCBB+0
MOV      R2, @#MDMA1
MOV      R3, @#MDMAW0     ; WRITE TX0 TO HOST MEMORY
MOV      R4, @#MDMAW0     ; WRITE TX1 TO HOST MEMORY
MOVB     #6, PCSR1+1
CLC
RTS      PC

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 20  
MODULE D,MICROTEST #5

SEQ 402

```

801
802      : LINK BYTE COUNTER TEST
803      :
804      : THIS IS 'MICROCODE' FOR THE LINK BYTE COUNTER TEST.
805      : GETS A BYTE COUNT FROM THE HOST MEMORY. LOOPS BACK A DATAGRAM
806      : WITH THE BYTE COUNT TO THE RECEIVE BUFFER. VERIFIES THE RECEIVE
807      : BUFFER TO:
808      :
809      :     1. MAKE SURE RECEIVE BYTE COUNT IS THE SAME AS THAT DESIGNATED
810      :        BY THE TRANSMIT BUFFER BYTE COUNT ENTRY
811      :
812      :     2. MAKE SURE RECEIVE BUFFER WAS ACTUALLY OVERWRITTEN WITH THE
813      :        BYTE COUNT WRITTEN IN THE TRANSMIT BYTE COUNT BUFFER
814      :        ENTRY.
815      :
816      : WRITES THE VALUE OF THE RECEIVE BUFFER BYTE COUNT TO HOST MEMORY.
817      :
818      .SBTTL LINK BYTE COUNTER TEST
819      :
820      : *****
821      : ***** TELL HOST WE ARE BUSY *****
822      : *****
823
824 002674' 112767 000002 175742 MICD7:  MOVB  #INTST,PCSR1      ; TELL HOST WE ARE TESTING
825 002702' 016737 175736 021020      MOV   PCSR1,#IPCSR1
826
827
828      : *****
829      : ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
830      : *****
831
832 002710' 012700 100000      MOV   #LINADR,R0      ; RECEIVE BUFFER STARTS HERE
833 002714' 012703 177777      MOV   #177777,R3     ; GET ALL ONES
834 002720' 010067 175734      MOV   R0,RBUF        ; SAVE ADDRESS OF RECV BUFFER
835
836 002724' 010320      10$:  MOV   R3,(R0)+      ; FILL UP THE BUFFER
837 002726' 020027 104000      CMP   R0,#LINADR+SIZ!K ; OVERFILL- SHOW RECV OVERLAY
838 002732' 103774      BLO   10$
839
840
841      : *****
842      : ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
843      : *****
844
845 002734' 010067 175722      20$:  MOV   R0,TBUF        ; SAVE ADDRESS XMIT BUFFER
846 002740' 005020      CLR   (R0)+          ; ZEROS FOR OVERLAY
847 002742' 020027 110000      CMP   R0,#LINADR+SIZ2K ; FILL XMIT BUFFER WITH PATTERN
848 002746' 103774      BLO   20$           ; STOP AT THE TOP
849
850
851      : *****
852      : ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
853      : *****
854
855 002750' 016737 175672 021010      MOV   IPCSR2,#MDRA0  ; SET TO GET HOST PCBB ADDRESS
856 002756' 016737 175666 021012      MOV   IPCSR2+2,#MDRA1

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 21  
LINK BYTE COUNTER TEST

```

857 002764' 013700 021014      MOV    @#MDHAR0,R0      ; R0 NOW CONTAINS PCBB LO
858 002770' 013701 021014      MOV    @#MDHAR0,R1      ; R1 NOW CONTAINS PCBB HI
859 002774' 010037 021010      MOV    R0,@#MDHAR0      ; POINT AT PCBB
860 003000' 010137 021012      MOV    R1,@#MDHAR1
861 003004' 013703 021014      MOV    @#MDHAR0,R3      ; R3 NOW CONTAINS BYTE COUNT
862
863
864                               ; *****
865                               ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
866                               ; *****
867
868 003010' 012737 100200 177776  MOV    #MODE!ENABLE,@#CMDREG ; ENABLE LINK, SELECT MODE REG
869 003016' 012737 100014 177774  MOV    #PROM!LOOP!DTCR,@#MODREG ; PROM MODE AND LOOPBACK
870
871 003024' 016704 175632      MOV    TBUF,R4          ; POINT AT XMIT BUFFER
872 003030' 005024      CLR    (R4)+            ; CLEAR OUT STATUS WORD
873 003032' 010324      MOV    R3,(R4)+        ; WRITE PASSED BYTE COUNT
874 003034' 005037 021034      CLR    @#CLRFIF        ; CLEAR THE FIFO
875 003040' 005067 175574      CLR    FLG4            ; CLEAR INTERRUPT FLAG
876 003044' 016737 175610 021032  MOV    RBUF,@#LFRBUF    ; TELL UNA WHERE RECIEVE BUFF IS
877 003052' 016737 175604 021030  MOV    TBUF,@#LTAC      ; TELL UNA WHERE XMIT BUFF IS
878
879 003060' 106427 000140      MTPS   #PRI03          ; ALLOW XMITTER TO INTERRUPT
880 003064' 032767 000040 175546 30$:  BIT    #TRNFLG,FLG4    ; WAIT FOR INTERRUPT
881 003072' 001774      BEQ    30$
882 003074' 032767 000100 175536 35$:  BIT    #RCVFLG,FLG4    ; WAIT FOR RECEIVER INTERRUPT TOO
883 003102' 001774      BEQ    35$
884
885
886 003104' 106427 000340      MTPS   #PRI07          ; DISABLE INTERRUPTS
887
888
889                               ; *****
890                               ; ***** COUNT ZEROS IN RECEIVE BUFFER *****
891                               ; *****
892
893 003110' 016704 175544      MOV    RBUF,R4          ; VERIFY RECEIVE BUFFER CONTENTS
894 003114' 062704 000002      DD    #2,R4            ; BB COULD REPORT STATUS 66
895 003120' 012402      MOV    (R4)+,R2        ; SAVE 'LEN' BUFFER LENGTH
896 003122' 005005      CLR    R5              ; COUNT DATA WORDS TRANSFERRED
897
898 003124' 112403 40$:  MOV    (R4)+,R3        ; READ DATA BACK
899 003126' 122703 000000      CMPB   #0,R3          ; IS IT ZERO?
900 003132' 001004      BNE    45$            ; EXIT IF AT THE EDGE
901 003134' 005205      INC    R5              ; BUMP THE TALLY
902 003136' 020527 002756      CMP    R5,#MAXBC      ; DONE YET?
903 003142' 103770      BLO    40$            ; IF NOT, KEEP GOING
904
905 003144' 062700 000002 45$:  ADD    #2,R0            ; INDEX DOWN TO PCBB+2
906 003150' 005501      ADC    R1
907
908 003152' 010037 021010      MOV    R0,@#MDHAR0      ; POINT TO PCBB+2
909 003156' 010137 021012      MOV    R1,@#MDHAR1
910
911 003162' 010237 021026      MOV    R2,@#MDHAR0      ; WRITE RECEIVE BYTE COUNT
912 003166' 010537 021026      MOV    R5,@#MDHAR0      ; WRITE BUFFER BYTE COUNT

```

77MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 22  
MICROD.MAC 07-APR-83 16:06 LINK BYTE COUNTER TEST

913							
914	003172'	112767	000007	175445	MOVB	#7,PCSR1+1	: TELL HIM WHAT TEST IT IS
915	003200'	000241			CLC		
916	003202'	000207			RTS	P'	

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 23  
LINK BYTE COUNTER TEST

```

917
918
919
920
921      ; LINK MEMORY ARBITRATION TEST
922
923      .SBTTL LINK MEMORY ARBITRATION TEST
924
925      ; THIS IS MICROCODE FOR THE LINK ARBITRATION TEST.
926      ; THIS MICROCODE WILL HAVE THE DMA ENGINE, THE T-11, AND BOTH STATE
927      ; MACHINES ATTEMPTING TO ACCESS LINK MEMORY AT THE SAME TIME.
928      ;
929      ; EACH PROCESS OPERATES ON DATA THAT IS UNIQUE FOR IDENTIFICATION.
930      ;
931      ; *****
932      ; ***** TELL HOST WE ARE TESTING *****
933      ; *****
934
935      003204' 112737 000002 021020 MICDB:  MOVB  #INTST,@#IPCSR1      ; TELL HOST WE ARE TESTING
936      003212' 016737 175426 021020      MOV   PCSR1,@#IPCSR1
937
938
939      ; *****
940      ; ***** RETRIEVE HOST MEMORY FROM PCBB *****
941      ; *****
942
943      003220' 016737 175422 021010      MOV   IPCSR2,@#MDMA0      ; SET TO GET HOST PCBB ADDRESS
944      003226' 016737 175416 021012      MOV   IPCSR2+2,@#MDMA1
945      003234' 013704 021014      MOV   @#MDMA0,R4      ; R4 NOW HOLDS PCBB LOW
946      003240' 013705 021014      MOV   @#MDMA0,R5      ; R5 NOW HOLDS PCBB HIGH
947
948
949      ; *****
950      ; ***** FILL RECEIVE BUFFER WITH ZEROS *****
951      ; *****
952
953      003244' 012700 100000      MOV   #LINADR,R0      ; FILL RECEIVE BUFFER
954      003250' 010067 175404      MOV   R0,RBUF        ; SAVE A COPY OF POINTER
955      003254' 005020      10$:  CLR   (R0)+          ; CLEAR IT OUT
956      003256' 020027 104000      CMP   R0,#LINADR+SIZ1K
957      003262' 103774      BLO  10$
958
959
960      ; *****
961      ; ***** FILL TRANSMIT BUFFER WITH 33 HEX **
962      ; ***** THATS 31463 OCTAL *****
963      ; *****
964
965      003264' 010067 175372      20$:  MOV   R0,TBUF        ; FILL TRANSMIT BUFFER
966      003270' 012720 031463      MOV   #31463,(R0)+   ; 31463 = 3333 HEX (THE REAL THING)
967      003274' 020027 110000      CMP   R0,#LINADR+SIZ2K
968      003300' 103773      BLO  20$
969
970
971      ; *****
972      ; ***** FILL DMA BUFFER WITH OF HEX ***

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 24  
LINK MEMORY ARBITRATION TEST

```

973          : ***** THATS 7417 OCTAL *****
974          : *****
975
976 003302' 010067 175356          MOV      R0,DBUFF          : SAVE COPY OF POINTER
977 003306' 012720 007417 30$:   MOV      #7417,(R0)+
978 003312' 020027 114000          CMP      R0,#LINADR+SIZ3K
979 003316' 103773          BLO     30$
980
981          : *****
982          : ***** FILL BUFFER WITH FF HEX ***
983          : ***** THATS 177777 OCTAL *****
984          : *****
985
986
987 003320' 010067 175342          MOV      R0,RBUF          : T-11 BUFFER POINTER
988 003324' 012720 177777 40$:   MOV      #177777,(R0)+
989 003330' 020027 120000          CMP      R0,#LINADR+SIZ4K
990 003334' 103773          BLO     40$
991
992          : *****
993          : ***** GET DMA ENGINE READY TO GO *****
994          : *****
995
996
997 003336' 010402          MOV      R4,R2          : GET HOST ADDRESS OF PCBB
998 003340' 010503          MOV      R5,R3
999
1000 003342' 062702 000006          ADD      #6,R2          : DUMP BUFFER AFTER ERROR CODES
1001 003346' 005503          ADC      R3
1002
1003 003350' 010237 021004          MOV      R2,&#DMATO      : 'TO' REGISTERS INCREMENTS
1004 003354' 010337 021006          MOV      R3,&#DMAT1
1005 003360' 016737 175300 021022  MOV      DBUFF,&#DMAF    : WHERE 'FROM' REG IS LOCATED
1006 003366' 012737 003774 021024  MOV      #3774,&#DMAWC   : WILL XFER 1K-2 WORDS
1007
1008 003374' 010700          MOV      PC,R0          : CALCULATE THE INTERRUPT VECTOR
1009 003376' 062700 175074          ADD      #DMAINT-.,R0   : THROW IN THE OFFSET
1010 003402' 010037 000114          MOV      R0,&#DMAVEC
1011 003406' 012737 000300 000116  MOV      #PRIORITY,&#DMAVEC+2 : PRIORITY OF INTERRUPT SERVICE ROUTINE
1012 003414' 005067 175236          CLR      DMONE         : FLAG
1013
1014          : *****
1015          : ***** SET UP THE STATE MACHINES *****
1016          : *****
1017
1018 003420' 012737 100200 177776  MOV      #MODE!ENABLE,&#CMDREG
1019 003426' 012737 100004 177774  MOV      #PROM!LOOP,&#MODREG
1020
1021 003434' 016701 175222          MOV      TBUF,R1       : GET ADDRESS OF XMIT BUFFER
1022 003440' 062701 000002          ADD      #2,R1         : SKIP TO BYTE COUNT
1023 003444' 012721 002752          MOV      #MAXBC-CRCSIZ,(R1)+ : SET BYTE COUNT TO MAX ALLOWED
1024 003450' 005037 021034          CLR      &#CLRFIF     : CLEAR INTERRUPT FLAG
1025 003454' 005067 175160          CLR      FLG4         : CLEAR INTERRUPT FLAG
1026 003460' 016737 175174 021032  MOV      RBUF,&#LFRBUF  : TELL LINK WHERE RECEIVE BUF IS
1027 003466' 016700 175174          MOV      RBUF,R0       : POINT TO MICROCPU BUFFER
1028 003472' 012702 002000          MOV      #SIZ1K/2,R2  : WORD COUNT FOR MICROCPU LOOP

```

77MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 25  
 MICROD.MAC 07-APR-83 16:06 LINK MEMORY ARBITRATION TEST

```

1029 003476' 012701 125252          MOV      #125252,R1          ;DATA FOR MICROCPU LOOP
1030
1031          ; *****
1032          ; ***** STATE THE MACHINES *****
1033          ; *****
1034
1035          ; IS EVERYBODY READY?
1036 003502' 016737 175154 021030    MOV      TBUF, @#LTAC      ; THIS STARTS THE STATE MACHINES
1037
1038 003510' 005237 021002          INC      @#DMACSR          ; THIS STARTS DMA ENGINE
1039
1040 003514' 106427 000140          MTPS    #PRI03            ; ALLOW INTERRUPTS
1041
1042 003520' 010120          50$:    MOV      R1, (R0)+      ;START FILLING LINK MEMORY...
1043 003522' 077202          SOB      R2, 50$          ;WITH MICROCPU
1044
1045          ;THE MICROCPU IS DONE
1046
1047 003524' 032767 000100 175106 60$:  BIT      #RCVFLG, FLG4      ; IS THE RECEIVE DONE?
1048 003532' 001774          BEQ      60$              ;NOT YET
1049
1050 003534' 032767 000040 175076 70$:  BIT      #TRNFLG, FLG4      ; IS THE TRANSMIT DONE?
1051 003542' 001774          BEQ      70$              ;NOT YET
1052
1053 003544' 005767 175106          80$:    TST      DMDONE        ; IS THE DMA ENGINE DONE?
1054 003550' 001775          BEQ      80$              ;NOT YET
1055
1056          ;EVERYBODY IS DONE, SO NOW CHECK THE...
1057          ;DATA
1058
1059 003552' 016700 175102          MOV      RBUF, R0          ;POINT TO RECEIVE BUFFER
1060 003556' 062700 000004          ADD      #4, R0            ;INDEX DOWN TO DATA PART
1061 003562' 012702 001365          MOV      #<MAXBC-CRCSIZ>/2, R2 ;AMOUNT OF DATA TO CHECK
1062 003566' 022710 031463          90$:    CMP      #31463, (R0)      ; IS THE DATA CORRECT?
1063 003572' 001005          BNE      100$              ;NO
1064 003574' 062700 000002          ADD      #2, R0            ;YES, POINT TO NEXT WORD OF DATA
1065 003600' 077206          SOB      R2, 90$          ;CONTINUE CHECKING DATA UNTIL DONE
1066 003602' 0002'1          CLC                          ;TELL HOST THIS TEST SUCCESS
1067 003604' 000414          BR      110$
1068
1069 003606' 010437 021010          100$:   MOV      R4, @#DMA0         ;POINT TO HOST PCBB
1070 003612' 010537 021012          MOV      R5, @#DMA1
1071 003616' 012737 031463 021026    MOV      #31463, @#DMA0     ;GIVE HOST EXPECTED PATTERN IN PCBB
1072 003624' 011037 021026          MOV      (R0), @#DMA0      ;GIVE HOST ACTUAL PATTERN IN PCBB+2
1073 003630' 010037 021026          MOV      R0, @#DMA0        ;GIVE HOST LINK MEMORY ADDRESS
1074 003634' 000261          SEC                          ;TELL MICROMONITOR ERROR OCCURRED
1075
1076 003636' 112767 000010 175001 110$:  MOVB    #8., PCSR1+1        ;TELL HOST WHAT TEST THIS IS
1077 003644' 000207          RTS      PC                ;RETURN TO MICROMONITOR
1078

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 26  
LINK MEMORY ARBITRATION TEST

```

1079
1080
1081
1082
1083
1084
1085
1086 003646' 012767 000002 174770
1087 003654' 016737 174764 021020
1088
1089
1090
1091
1092
1093
1094 003662' 012700 100000
1095 003666' 010067 174766
1096 003672' 012700 004000
1097 003676' 010067 174760
1098
1099
1100
1101
1102
1103
1104 003702' 012737 100200 177776
1105 003710' 012737 100004 177774
1106
1107 003716' 016704 174740
1108 003722' 005024
1109 003724' 012724 007777
1110 003730' 005037 021034
1111 003734' 005067 174700
1112 003740' 016737 174714 021032
1113 003746' 016737 174710 021030
1114
1115
1116
1117
1118
1119
1120 003754' 106427 000100
1121 003760' 032767 000040 174652
1122 003766' 001774
1123
1124 003770' 032767 000100 174642
1125 003776' 001774
1126
1127 004000' 106427 000340
1128
1129
1130
1131
1132
1133
1134 004004' 016704 174650

```

```

.SBTTL LINK BYTE COUNTER MAXIMUM TEST
:
: *****
: ***** TELL THE HOST WE ARE TESTING *****
: *****
PICD9: MOV #INTST,PCSR1 ; TELL HOST WE ARE TESTING
      MOV PCSR1,#IPCSR1
:
: *****
: ***** CALCULATE BUFFER ADDRESS POINTERS *****
: *****
      MOV #LINADR,RO ; RELATIVE TO LINK MEMORY
      MOV RO,RBUF
      ADD #SIZ1K,RO
      MOV RO,TBUF
:
: *****
: ***** SET UP LINK FOR A LOOPBACK *****
: *****
      MOV #MODE!ENABLE,#CMDREG
      MOV #PROM!LOOP,#MODREG
:
      MOV TBUF,R4 ; SET UP XMIT BUFFER
      CLR (R4)+
      MOV #XMTBC,(R4)+
      CLR #CLRFIF
      CLR FLG4
      MOV RBUF,#LFRBUF
      MOV TBUF,#LTAC
:
: *****
: ***** WAIT FOR INTERRUPTS *****
: *****
      MTPS #PRI02
10$: BIT #TRNFLG,FLG4
      BEQ 10$
:
      MTPS #PRI07
20$: BIT #RCVFLG,FLG4
      BEQ 20$
:
      MTPS #PRI07
:
: *****
: ***** CHECK RECEIVE BUFFER BYTE COUNT *****
: *****
      MOV RBUF,R4

```



MICROD - MICROCODE MODULE D      MACY11 30A(1052) 07-APR-83 16:50 PAGE 27  
 MICROD.MAC    07-APR-83 16:06      LINK BYTE COUNTER MAXIMUM TEST

1135	004010'	062704	000002			ADD	#2,R4		: POINT AT MLEN
1136	004014'	012402				MOV	(R4)+,R2		
1137	004016'	022702	007777			CMP	#MRECBC,R2		
1138	004022'	001005				BNE	40\$		
1139	004024'	000241				CLC			
1140	004026'	112767	000011	174611	30\$:	MOVB	#9.,PCSR1+1		
1141	004034'	000207				RTS	PC		
1142									
1143	004036'	000261			40\$:	SEC			: ERROR EXIT
1144	004040'	000772				BR	30\$		

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 28  
MODULE D, MICROTEST #10 - FIFO TEST

```

1145 .SBTTL MODULE D, MICROTEST #10 - FIFO TEST
1146
1147 ;*****
1148 ;
1149 ;THIS IS THE MICROCODE FOR THE FIFO TEST.
1150 ;IT TRANSMITS A PACKET FROM BUFFER 0, SETS UP THE RECEIVER TO RECEIVE TO
1151 ;THE BUFFER SPECIFIED BY THE HOST IN PCBB+0. AFTER THE INTERRUPT THE BUFFER
1152 ;DONE FIFO CONTENTS ARE READ AND PASSED TO THE HOST IN PCBB+2.
1153 ;
1154 ;*****
1155
1156 004042' 112767 000002 174574 MICD10: MOVB #INTST,PCSR1 ;TELL HOST WE ARE TESTING
1157 004050' 016737 174570 021020 MOV PCSR1,@#IPCSR1
1158 ;
1159 ;GET RECEIVE BUFFER ADDRESS FROM HOST MEMORY
1160 ;
1161 004056' 016737 174564 021010 MOV IPCSR2,@#MDMA0 ;GET CONTENTS OF HOST'S PCSR2+3
1162 004064' 016737 174560 021012 MOV IPCSR2+2,@#MDMA1
1163 004072' 013700 021014 MOV @#MDMAR0,R0 ;R0 = CONTENTS OF HOST'S PCSR2
1164 004076' 013701 021014 MOV @#MDMAR0,R1 ;R1 = CONTENTS OF HOST'S PCSR3
1165 004102' 010037 021010 MOV R0,@#MDMA0 ;POINT TO PCBB+0
1166 004106' 010137 021012 MOV R1,@#MDMA1
1167 004112' 013767 021014 174540 MOV @#MDMAR0,RBUFP ;GET RECEIVE BUFFER ADDRESS
1168 ;
1169 ;CLEAR ALL OF LINK MEMORY
1170 ;
1171 004120' 012702 100000 MOV #LINADR,R2
1172 004124' 005022 105: CLR (R2)+
1173 004126' 020227 177774 CMP R2,#LINADR+LINSIZ
1174 004132' 103774 BLO 105
1175 ;
1176 ;FILL THE TRANSMIT BUFFER WITH 1'S
1177 ;
1178 004134' 012702 100000 MOV #LINADR,R2 ;USE BUFFER 0 FOR TRANSMIT
1179 004140' 010267 174516 MOV R2,TBUFP ;SAVE IT
1180 004144' 012722 177777 205: MOV #177777,(R2)+
1181 004150' 020227 104000 CMP R2,#LINADR+SIZ1K
1182 004154' 103773 BLO 205
1183 ;
1184 ;SET UP LINK FOR PROMISCUOUS MODE AND INTERNAL LOOPBACK. TRANSMIT THE MAX
1185 ;SIZE PACKET. CLEAR THE FIFO AND GIVE THE LINK A RECEIVER BUFFER AND A TRANSMIT
1186 ;BUFFER TO START THE OPERATION.
1187 ;
1188 004156' 012737 100200 177776 MOV #MODE!ENABLE,@#CMDREG ;TURN ON THE LINK AND SELECT MODE REG
1189 004164' 012737 100004 177774 MOV #PROM!LOOP,@#MODREG ;SET PROMISCUOUS MODE AND LOOPBACK
1190 004172' 016702 174464 MOV TBUFP,R2 ;GET TRANSMIT BUFFER
1191 004176' 005022 CLR (R2)+ ;CLEAR THE STATUS WORD
1192 004200' 012712 002752 MOV #MAXBC-CRCSIZ,(R2) ;SET TO MAX BYTE COUNT
1193 004204' 005037 021034 CLR @#CLRIFIF ;CLEAR THE FIFO
1194 004210' 005067 174424 CLR FLG4 ;CLEAR THE INTERRUPT FLAG
1195 004214' 016737 174440 021032 MOV RBUFP,@#LFRBUF ;LOAD THE FIFO WITH A RECEIVE BUFFER
1196 004222' 016737 174434 021030 MOV TBUFP,@#LTAC ;GIVE TRANSMIT BUFFER TO START TRANSMIT
1197 ;
1198 ;LOWER THE PROCESSOR PRIORITY AND WAIT FOR THE INTERRUPTS
1199 ;
1200 004230' 106427 000140 MTPS #PRI03 ;ALLOW BOTH TRANSMIT AND RECEIVER INTERRUPT

```

MICROD - MICROCODE MODULE D  
 MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 29  
 MODULE D, MICROTEST #10 - FIFO TEST

```

1201 004234' 032767 000100 174376 30$: BIT #RCVFLG,FLG4 ;WAIT FOR RECEIVER FIRST
1202 004242' 001774 BEQ 30$
1203
1204 004244' 032767 000040 174366 40$: BIT #TRNFLG,FLG4 ;THEN THE TRANSMITTER
1205 004252' 001774 BEQ 40$
1206
1207 004254' 106427 000340 MTPS #PRI07 ;DISABLE FURTHER INTERRUPTS
1208
1209 ;READ RECEEVER BUFFER DONE FIFO AND PASS BACK TO HOST IN PCBB+2
1210 ;
1211 004260' 062700 000002 ADD #2,R0 ;POINT TO PCBB+2
1212 004264' 005501 ADC R1
1213 004266' 010037 021010 MOV R0,@#MDMA0
1214 004272' 010137 021012 MOV R1,@#MDMA1
1215 004276' 013737 021044 021026 MOV @#LRBUF,@#MDMAW0 ;PASS BUFFER DONE DATA TO HOST
1216
1217 004304' 112767 000012 174333 MOVB #10.,PCSR1+1 ;INDICATE WHAT TEST WE JUST FINISHED
1218 004312' 000241 CLC ;INDICATE SUCCESS
1219 004314' 000207 RTS
1220 PC

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 30  
MODULE D, MICROTST #11 - LINK ADDRESS TEST

.SBTTL MODULE D, MICROTST #11 - LINK ADDRESS TEST

1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236

\*\*\*\*\*  
: THIS THE MICROCODE FOR THE RECEIVER AND TRANSMITTER LINK MEMORY ADDRESS TESTS.  
: IT FILLS ALL OF LINK MEMORY WITH 0'S THEN FILLS THE TRANSMIT BUFFER OBTAINED  
: FROM THE HOST'S PCBB+2 WITH 1'S. IT SETS UP THE RECEIVER TO RECEIVE A PACKET  
: INTO THE BUFFER OBTAINED FROM THE HOST'S PCBB+0. THE PACKET OF 1'S IS  
: TRANSMITTED WITH LOOPBACK ENABLED IN PROMISCUIOUS MODE. AFTER THE INTERRUPT  
: ALL LINK MEMORY OUTSIDE OF THE TRANSMIT AND RECEIVE BUFFERS IS CHECKED  
: TO SEE IF AND DATA WAS WRITTEN TO AN INCORRECT ADDRESS. IF SO THE FAULTY  
: ADDRESS IS PASSED BACK TO THE HOST IN PCBB+4 AND THE GOOD DATA IN PCBB+6 AND  
: THE BAD DATA IN PCBB+10.  
\*\*\*\*\*

1237 004316' 112767 000002 174320  
1238 004324' 016737 174314 021020

MICD11: MOVB #INTST,PCSR1 ;TELL HOST WE ARE TESTING  
MOV PCSR1,@#IPCSR1

1240  
1241  
1242  
1243

: GET THE RECEIVER BUFFER FROM THE HOST'S PCBB+0 AND THE TRANSMIT BUFFER FROM  
: THE HOST'S PCBB+2

1244 004332' 016737 174310 021010  
1245 004340' 016737 174304 021012  
1246 004346' 013700 021014  
1247 004352' 013701 021014  
1248 004356' 010037 021010  
1249 004362' 010137 021012  
1250 004366' 013767 021014 174264  
1251 004374' 013767 021014 174260

: MOV IPCSR2,@#MDMA0 ;GET CONTENTS OF HOST'S PCSR2+3  
: MOV IPCSR2+2,@#MDMA1  
: MOV @#MDMAR0,R0 ;R0 = CONTENTS OF HOST'S PCSR2  
: MOV @#MDMAR0,R1 ;R1 = CONTENTS OF HOST'S PCSR3  
: MOV R0,@#MDMA0 ;POINT TO PCBB+0  
: MOV R1,@#MDMA1  
: MOV @#MDMAR0,RBUF ;GET RECEIVER BUFFER FROM PCBB+0  
: MOV @#MDMAR0,TBUF ;GET TRANSMIT BUFFER FROM PCBB+2

1252  
1253  
1254

: FILL ALL OF LINK MEMORY WITH 0'S

1255 004402' 012702 100000  
1256 004406' 005022  
1257 004410' 020227 177774  
1258 004414' 103774

10\$: MOV #LINADR,R2  
CLR (R2)+  
CMP R2,#LINADR+LINSIZ  
BLO 10\$

1260 004416' 016702 174240  
1261 004422' 005022  
1262 004424' 012722 002756

: MOV TBUF,R2 ;POINT TO BASE OF TRANSMIT BUFFER  
: CLR (R2)+ ;CLEAR STATUS WORD  
: MOV #MAXBC,(R2)+ ;PUT IN BYTE COUNT

1263  
1264  
1265

: FILL TRANSMIT BUFFER WITH 1'S

1266 004430' 012701 002756  
1267 004434' 006201  
1268 004436' 012722 177777  
1269 004442' 005301  
1270 004444' 001374

20\$: MOV #MAXBC,R1  
ASR R1  
MOV #177777,(R2)+  
DEC R1  
BNE 20\$

1271  
1272  
1273  
1274

: SET UP LINK FOR PROMISCUIOUS MODE AND INTERNAL LOOPBACK. TRANSMIT THE PACKET

1275 004446' 012737 000200 177776  
1276 004454' 012737 100014 177774

: MOV #MODE,@#CMDREG ;SELECT THE MODE REGISTER  
: MOV #PRON!LOOP!DTCR,@#MODREG ;ENABLE PROMISCUIOUS MODE AND LOOPBACK

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 31  
MODULE D, MICROTEST #11 - LINK ADDRESS TEST

```

1277
1278 004462' 012737 100000 177776      MOV      #ENABLE,#CMDREG      :AND DISABLE TRANSMIT CRC
1279 004470' 005037 021034              CLR      @#CLRIF              :TURN ON THE LINK
1280 004474' 005067 174140              CLR      FLG4                 :CLEAR THE FIFO
1281 004500' 016737 174154 021032      MOV      RBUF,#LFRBUF         :CLEAR THE INTERRUPT WORD
1282 004506' 016737 174150 021030      MOV      TBUF,#LTAC           :GIVE THE LINK A RECEIVE BUFFER
1283                                         :GIVE THE LINK A TRANSMIT BUFFER
1284                                         :TO START THE OPERATION
1285
1286      ;LOWER THE PROCESSOR PRIORITY AND WAIT FOR THE INTERRUPT
1287 004514' 106427 000140              MTPS     #PRI03               :ALLOW BOTH TRANSMIT AND RECEIVE INTERRUPT
1288 004520' 032767 000100 174112 30$:   BIT      #RCVFLG,FLG4         :WAIT FOR THE RECEIVE INTERRUPT
1289 004526' 001774              BEQ
1290 004530' 032767 000040 174102 40$:   BIT      #TRNFLG,FLG4
1291 004536' 001774              BEQ      40$
1292
1293 004540' 106427 000340              MTPS     #PRI07               :DISABLE FURTHER INTERRUPTS
1294
1295      ;CHECK RECEIVE BUFFER TO MAKE SURE IT RECEIVED NON-ZERO DATA
1296
1297 004544' 012703 177777              MOV      #177777,R3           :PASS THIS NON-ZERO DATA IF FAILURE
1298 004550' 016702 174104              MOV      RBUF,R2              :GET POINTER TO RECEIVER BUFFER
1299 004554' 062702 000004              ADD      #4,R2                 :SKIP STATUS AND BYTE COUNT
1300 004560' 012701 002756              MOV      #MAXBC,R1            :NUMBER OF BYTES WE SENT
1301 004564' 006201              ASR      R1                     :MAKE IT WORDS
1302 004566' 005712 45$:   TST      (R2)                 :IS THERE NON-ZERO DATA HERE?
1303 004570' 001432              BEQ      90$                   :NO, ERROR!
1304 004572' 062702 000002              ADD      #2,R2                 :POINT TO NEXT WORD IN RECEIVER BUFFER
1305 004576' 005301              DEC      R1
1306 004600' 001372              BNE      45$
1307
1308      ;CHECK ALL LINK MEMORY EXCEPT THE TRANSMIT AND RECEIVE BUFFERS FOR ANY NON-ZERO
1309      ;DATA
1310
1311 004602' 005003              CLR      R3                     :PASS THIS ZERO DATA IF FAILURE
1312 004604' 012702 100000              MOV      #LINADR,R2           :START AT BASE OF LINK MEMORY
1313 004610' 020267 174046 50$:   CMP      R2,TBUF              :ARE WE AT THE TRANSMIT BUFFER?
1314 004614' 001002              BNE      60$                   :NO
1315 004616' 062702 002762              ADD      #MAXBC+4,R2          :YES, SKIP OVER THE HEADER AND THE DATA
1316 004622' 020267 174032 60$:   CMP      R2,RBUF              :ARE WE AT THE RECEIVE BUFFER?
1317 004626' 001002              BNE      70$                   :NO
1318 004630' 062702 002762              ADD      #MAXBC+4,R2          :YES, SKIP OVER THE HEADER AND THE DATA
1319 004634' 005712 70$:   TST      (R2)                 :IS ANY NON-ZERO DATA IN HERE?
1320 004636' 001007              BNE      90$                   :YES, ERROR!
1321 004640' 062702 000002              ADD      #2,R2                 :POINT TO NEXT WORD IN LINK MEMORY
1322 004644' 020227 177774              CMP      R2,#LINADR+LINSIZ     :HAVE WE CHECKED ALL OF LINK MEMORY?
1323 004650' 103757              BLO      50$                   :NOT YET
1324
1325      ;TEST WAS SUCESSFULL
1326
1327 004652' 000241              CLC
1328 004654' 000407              BR       100$                 :INDICATE SUCCESS
1329
1330      ;PASS THE ADDRESS BACK TO THE HOST IN PCBB+4 AND THE DATA IN PCBB+6
1331
1332 004656' 010237 021026 90$:   MOV      R2,#MDMAWO           :PASS ADDRESS TO PCBB+4

```

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 32  
MODULE D, MICROTTEST #11 - LINK ADDRESS TEST

1333	004662'	010337	021026					
1334	004666'	011237	021026		MOV	R3,@#MDMAWO		:PASS GOOD DATA TO PCBB+6
1335	004672'	000261			MOV	(R2),@#MDMAWO		:PASS DATA TO PCBB+10
1336					SEC			:INDICATE FAILURE
1337	004674'	112767	000013	173743	100%:	MOVB	#11.,PCSR1+1	
1338	004702'	000207				RTS	PC	:TELL HOST WHAT TEST JUST FINISHED
1339								
1340								
1341	004704'	004706						
1342		000001						

MICDSZ::MICDSZ-MICROD+2  
.END







77MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 36  
 MICROD.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- USER SYMBOLS

PCSR1	000644R	002	140*	141	216*	218*	219	280#	289*	290	318*	322*	323	356*	370*
			371	442*	482*	483	558*	597*	598	681*	723*	724	796*	824*	825
			914*	936	1076*	1086*	1087	1140*	1156*	1157	1217*	1237*	1238	1337*	
PHYAD0=	021060		27#												
PHYAD1=	021062		28#												
PHYAD2=	021064		29#												
PHYAD3=	021066		30#												
PHYAD4=	021070		31#												
PHYAD5=	021072		32#												
PRI00 =	000000		64#	190*											
PRI01 =	000040		63#												
PRI02 =	000100		62#	1120*											
PRI03 =	000140		61#	314*	351*	410*	524*	649*	765*	879*	1040*	1200*	1287*		
PRI04 =	000200		60#	164	174	347*									
PRI05 =	000240		59#	169	179										
PRI06 =	000300		58#	1011											
PRI07 =	000340		57#	137*	147	154	159	195*	317*	355*	417*	531*	656*	886*	1127*
			1207*	1293*											
PROM =	100000		84#	304	337	400	514	639	755	869	1019	1105	1189	1276	
RBUF	000660R	002	284#	293*	311	326*	344	378*	407	491*	521	618*	646	733*	763
			834*	876	893	954*	1026	1059	1095*	1112	1134	1167*	1195	1250*	1281
			1298	1316											
RCEI =	002000		73#												
RCVFLG=	000100		109#	260	348	411	525	650	768	882	1047	1124	1201	1288	
RCVINT	000566R	002	177	258#											
RCVVEC=	000120		94#	178*	179*										
ROMADR=	040000		125#	126											
ROMSIZ=	040000		121#	126											
RXI =	020000		70#												
SANTIM	000642R	002	236*	279#											
SANVEC=	000134		95#	168*	169*										
SERI =	100000		68#												
SIZ1K =	004000		114#	115	116	295	328	380	493	620	736	837	956	1028	1096
			1181												
SIZ2K =	010000		115#	117	300	333	391	505	630	746	847	967			
SIZ3K =	014000		116#	978											
SIZ4K =	020000		117#	118	119	120	989								
SIZ8K =	040000		118#	121	122										
STACK =	001000		99#	139											
TBLD	000612R	002	210	266#											
TBUF	000662R	002	285#	298*	306	312	331*	339	345	389*	402	408	502*	516	522
			628*	641	647	743*	757	764	777	845*	871	877	965*	1021	1036
			1097*	1107	1113	1179*	1190	1196	1251*	1260	1282	1313			
TIMINT	000464R	002	167	236#											
TRNFLG=	000040		108#	263	315	352	414	528	653	766	880	1050	1121	1204	1290
TRNINT	000602R	002	172	263#											
TRNVEC=	000070		93#	173*	174*										
TXI =	010000		71#												
UNIERR=	020000		112#	231											
WCSADR=	000000		123#	124											
WCSSIZ=	020000		119#	124											
.	004706R	002	144	152	157	162	167	172	177	198	210	233	249	256	266
			267	268	269	270	271	272	273	274	275	276	1009		

77MICROD - MICROCODE MODULE D MACY11 30A(1052) 07-APR-83 16:50 PAGE 38  
MICROD.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL	1#
BERROR	1#
BGNAU	1#
BGNAUT	1#
BGNCLN	1#
BGNDU	1#
BGNHRD	1#
BGNHW	1#
BGNINI	1#
BGNMOD	1#
BGNMSG	1#
BGNPRO	1#
BGNPTA	1#
BGNRPT	1#
BGNSEG	1#
BGNSET	1#
BGNSFT	1#
BGNSRV	1#
BGNSUB	1#
BGNSW	1#
BGNTST	1#
BNCOMP	1#
BNERRO	1#
BREAK	1#
BRESET	1#
CKLOOP	1#
CLOCK	1#
CLOSE	1#
CLRVEC	1#
COMMEN	1#
DELAY	1#
DESCRI	1#
DEVTYP	1#
DISPAT	1#
DISPLA	1#
DOCLN	1#
DODU	1#
DORPT	1#
ENDAU	1#
ENDAUT	1#
ENDCLN	1#
ENDCOM	1#
ENDDU	1#
ENDHRD	1#
ENDHW	1#
ENDINI	1#
ENDMOD	1#
ENDMSG	1#
ENDPRO	1#
ENDPTA	1#
ENDRPT	1#
ENDSEG	1#
ENDSET	1#
ENDSFT	1#
ENDSRV	1#
ENDSUB	1#

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 39  
CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	1#
ENDTST	1#
EQUALS	1#
ERRDF	1#
ERRHRD	1#
ERROR	1#
ERRSF	1#
ERRSOF	1#
ERRTBL	1#
ESCAPE	1#
EXIT	1#
FEQUAL	1#
GETBYT	1#
GETPRI	1#
GETWOR	1#
GMANIA	1#
GMANID	1#
GMANIL	1#
GPHARD	1#
GPRMA	1#
GPRMD	1#
GPRML	1#
HEADER	1#
INLOOP	1#
IOSETU	1#
IOSTAR	1#
KT11	1#
LASTAD	1#
MANUAL	1#
MEMORY	1#
MSBYTE	1#
MSCHEC	1#
MSCNTO	1#
MSCOUN	1#
MSDATA	1#
MSDECR	1#
MSDEFA	1#
MSENDE	1#
MSERRI	1#
MSESCA	1#
MSESCS	1#
MSEXCP	1#
MSEXIT	1#
MSXSE	1#
MSXTJ	1#
MSGEN	1#
MSGENB	1#
MSGETS	1#
MSGETT	1#
MSGNGB	1#
MSGNIN	1#
MSGNLS	1#
MSGNSU	1#
MSGHTA	1#
MSGNTE	1#
MSHAPT	1#

77MICROD - MICROCODE MODULE D  
MICROD.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:50 PAGE 40  
CROSS REFERENCE TABLE -- MACRO NAMES

MSHAP 1#  
MSINCR 1#  
MSIOSE 1#  
MSLDRO 1#  
MSMASK 1#  
MSMCHI 1#  
MSMCLO 1#  
MSMSK1 1#  
MSPOP 1#  
MSPRIN 1#  
MSPUSH 1#  
MSPUT 1#  
MSPUT1 1#  
MSRADI 1#  
MSRBRO 1#  
MSRNRO 1#  
MSSETS 1#  
MSSTAR 1#  
MSVC 1#  
MSTLAB 1#  
MSTSTL 1#  
MSWORD 1#  
MSXFER 1#  
OPEN 1#  
POINTE 1#  
PRINTB 1#  
PRINTF 1#  
PRINTS 1#  
PRINTX 1#  
READBU 1#  
READEF 1#  
RFLAGS 1#  
SETPRI 1#  
SETVEC 1#  
SLASH 1#  
STARS 1#  
SVC 1#  
XFER 1#  
XFERF 1#  
XFERT 1#

. ABS. 000000 000  
000000 001  
MICRD 004706 002

ERRORS DETECTED: 0  
DEFAULT GLOBALS GENERATED: 0

MICROD.OBJ,MICROD.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROD.MAC  
RUN-TIME: 3 4 .4 SECONDS  
RUN-TIME RATIO: 39/7=5.0  
CORE USED: 31K (61 PAGES)

```

1      .TITLE MICROE - MICROCODE MODULE E
2      ; 88 DEDICATED THE CRC CIRCUITRY TO THE RECEIVE SIDE OF THE LINK
3
4      000000' .CSECT MICRE
5
6      .SBTTL REGISTER DEFINITIONS USED BY THE T11
7
8      021060 IPCSRO = 21000 :INTERNAL PCSRO ADDRESS
9      021002 DMACSR = 21002 :DMA ENGINE CONTROL STATUS REGISTER
10     021004 DMATO = 21004 :DMA ENGINE TO ADDRESS REGISTER #0
11     021006 DMAT1 = 21006 :DMA ENGINE TO ADDRESS REGISTER #1
12     021010 MDMA0 = 21010 :MICROCPU DMA TO ADDRESS REGISTER #0
13     021012 MDMA1 = 21012 :MICROCPU DMA TO ADDRESS REGISTER #1
14     021014 MDMAR0 = 21014 :MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15     021016 MDMAR1 = 21016 :MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16     021020 IPCSR1 = 21020 :INTERNAL PCSR1 ADDRESS
17     021022 DMAF = 21022 :DMA ENGINE FROM ADDRESS REGISTER
18     021024 DMAWC = 21024 :DMA ENGINE WORD COUNT REGISTER
19     021026 MDMAW0 = 21026 :MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20     021030 LTAC = 21030 :LINK TRANSMIT ADDRESS COUNTER REGISTER
21     021032 LFRBUF = 21032 :LINK RECIEVE BUFFER ADDRESS FIFO
22     021034 CLRFIF = 21034 :CLEAR FIFO
23     021036 MDMAW1 = 21036 :MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24     021040 PCSRSW = 21040 :SWITCH PACK REGISTER
25     021042 MDMSR = 21042 :MICROCPU DMA STATUS REGISTER
26     021044 LRBUF = 21044 :LINK RECIEVE BUFFER COMPLETED
27     021060 PHYAD0 = 21060 :PHYSICAL ADDRESS ROM BYTE 0
28     021062 PHYAD1 = 21062 :PHYSICAL ADDRESS ROM BYTE 1
29     021064 PHYAD2 = 21064 :PHYSICAL ADDRESS ROM BYTE 2
30     021066 PHYAD3 = 21066 :PHYSICAL ADDRESS ROM BYTE 3
31     021070 PHYAD4 = 21070 :PHYSICAL ADDRESS ROM BYTE 4
32     021072 PHYAD5 = 21072 :PHYSICAL ADDRESS ROM BYTE 5
33     177774 MODREG = 177774 :LINK MODE REGISTER
34     177774 ADDRREG = 177774 :LINK STATION ADDRESS RAM REGISTER
35     177776 CMDREG = 177776 :LINK COMMAND REGISTER
36
37     .SBTTL OTHER DEFINITIONS USED BY THE MICROCODE
38
39     100000 BIT15 = 100000
40     040000 BIT14 = 40000
41     020000 BIT13 = 20000
42     010000 BIT12 = 10000
43     004000 BIT11 = 4000
44     002000 BIT10 = 2000
45     001000 BIT9 = 1000
46     000400 BIT8 = 400
47     000200 BIT7 = 200
48     000100 BIT6 = 100
49     000040 BIT5 = 40
50     000020 BIT4 = 20
51     000010 BIT3 = 10
52     000004 BIT2 = 4
53     000002 BIT1 = 2
54     000001 BIT0 = 1
55
56     012400 :LASFTP = BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN

```

76MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 3  
OTHER DEFINITIONS USED BY THE MICROCODE

```

57      000340      PRI07 =      340
58      000300      PRI06 =      300
59      000240      PRI05 =      240
60      000200      PRI04 =      200
61      000140      PRI03 =      140
62      000100      PRI02 =      100
63      000040      PRI01 =       40
64      000000      PRI00 =       0
65      :
66      :PCSRO - PORT CONTROL STATUS REGISTER 0
67      :
68      100000      SERI =      BIT15
69      040000      PCEI =      BIT14
70      020000      RXI =      BIT13
71      010000      TXI =      BIT12
72      004000      DNI =      BIT11
73      002000      RCEI =      BIT10
74      000400      FATI =      BIT8
75      :
76      :LINK COMMAND REGISTER
77      :
78      100000      ENABLE =      BIT15 ;ENABLE LINK MODULE
79      000200      MODE =      BIT7  ;ENABLE MODE REGISTER
80      000100      ARAM =      BIT6  ;ENABLE STATION ADDRESS RAM
81      :
82      :LINK MODE REGISTER
83      :
84      100000      PROM =      BIT15 ;PROMISCUOUS MODE
85      040000      ENAL =      BIT14 ;ENABLE MULTICAST
86      004000      ENCR =      BIT11 ;ENABLE COLLISION TEST
87      002000      ACLO =      BIT10 ;ENABLE ACLO
88      000040      DRTY =      BIT5  ;DISABLE RETRY LOGIC
89      000020      COLL =      BIT4  ;SIMULATE A COLLISION
90      000010      DTCR =      BIT3  ;DISABLE TRANSMIT CRC LOGIC
91      000004      LOOP =      BIT2  ;ENABLE LOOPBACK
92      :
93      000070      TRNVEC= 70      ;VECTOR ADDRESS FOR THE TRANSMITTER
94      000120      RCVVEC= 120     ;VECTOR ADDRESS FOR THE RECEIVER
95      000134      SANVEC= 134     ;VECTOR ADDRESS FOR THE SANITY TIMER
96      000064      CSRVEC= 64      ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
97      000114      DMAVEC= 114     ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
98      000140      PARVEC= 140     ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
99      001000      STACK= 1000    ;STACK LOCATION
100     000001      INMON= 1        ;IN MICROMONITOR STATE
101     000002      INTST= 2        ;IN A TEST STATE
102     000003      INERR= 3        ;IN ERROR STATE
103     000001      CSRFLG= BIT0     ;CSR WRITE INTERRUPT OCCURED
104     000002      ERRFLG= BIT1     ;UNEXPECTED ERROR OCCURED
105     000004      PARFLG= BIT2     ;PARITY ERROR OCCURED
106     000010      NXMFLG= BIT3     ;NON-EXISTANT MEMORY ERROR OCCURRED
107     000020      NPRFLG= BIT4     ;NPR TIMEOUT OCCURRED
108     000040      TRNFLG= BIT5     ;TRANSMITTER INTERRUPT OCCURRED
109     000100      RCVFLG= BIT6     ;RECEIVER INTERRUPT OCCURRED
110     100000      NPRERR= BIT15    ;PCSRO FLAG INDICATING NPR ERROR OCCURRED
111     040000      NXMERR= BIT14    ;PCSRO FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
112     020000      UNIERR= BIT13    ;PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED

```

76MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

SEQ 423

113	010000	PARERR= BIT12	:PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
114	020000	MTCH= BIT13	:MATCH BIT
115	004000	SIZ1K= 4000	:1K WORDS
116	010000	SIZ2K= SIZ1K*2	:2K WORDS
117	020000	SIZ4K= SIZ2K*2	:4K WORDS
118	040000	SIZ8K= SIZ4K*2	:8K WORDS
119	020000	WCSSIZ= SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
120	020000	IOSIZ= SIZ4K	:SIZE OF I/O PAGE
121	040000	ROMSIZ= SIZ8K	:SIZE OF ROM
122	077774	LINSIZ= SIZ8K*2-4	:SIZE OF LINK MEMORY
123	000000	WCSADR= 0	:BASE ADDRESS OF WCS
124	020000	IOADR= WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
125	040000	ROMADR= IOADR+IOSIZ	:BASE ADDRESS OF ROM
126	100000	LINADR= ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
127	000100	MINBC= 64.	: 64 BYTES
128	002752	MAXBC= 1518.-4.	:MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
129	000000	DATERR= 0	:FLAG INDICATING DATA ERROR OCCURRED
130	000001	ADRERR= 1	:FLAG INDICATING ADDRESS ERROR OCCURRED
131			

76MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 5  
OTHER DEFINITIONS USED BY THE MICROCODE

```

132
133
134
135 000000' 106427 000340 MICROE::MTPS #PRI07 ;DISABLE INTERRUPTS
136 000004' 012737 000000 177776 MOV #0,#CMDREG ;TURN OFF THE LINK
137 000012' 012706 001000 MOV #STACK,SP ;SETUP STACK
138 000016' 112767 000001 000602 MOVB #INMON,PCSR1 ;TELL HOST WE ARE IN MICROMONITOR
139 000024' 016737 000576 021020 MOV PCSR1,#IPCSR1
140 000032' 012737 004000 021000 MOV #DNI,#IPCSRO
141 000040' 010700 MOV PC,RO ;TELL HOST THE LOAD AND START FINISHED
142 000042' 062700 000404 ADD #ERRINT-.,RO ;GET ADDRESS OF UNEXPECTED ERROR...
143 000046' 005001 CLR R1 ;HANDLER
144 000050' 010021 10$: MOV RO,(R1)+ ;FILL ALL UNUSED VECTORS WITH TRAP...
145 000052' 012721 000340 MOV #PRI07,(R1)+ ;HANDLER
146 000056' 020127 001000 CMP R1,#1000
147 000062' 002772 BLT 10$
148
149 000064' 010700 MOV PC,RO ;SETUP PARITY TRAP VECTOR
150 000066' 062700 000462 ADD #PARINT-.,RO
151 000072' 010037 000140 MOV RO,#PARVEC
152 000076' 012737 000340 000142 MOV #PRI07,#PARVEC+2
153
154 000104' 010700 MOV PC,RO ;SETUP DMA INTERRUPT VECTOR
155 000106' 062700 000364 ADD #DMAINT-.,RO
156 000112' 010037 000114 MOV RO,#DMAVEC
157 000116' 012737 000340 000116 MOV #PRI07,#DMAVEC+2
158
159 000124' 010700 MOV PC,RO ;SETUP CSR WRITE VECTOR
160 000126' 062700 000310 ADD #CSRWRT-.,RO
161 000132' 010037 000064 MOV RO,#CSRVEC
162 000136' 012737 000200 000066 MOV #PRI04,#CSRVEC+2
163
164 000144' 010700 MOV PC,RO ;SETUP SANTITY TIMER VECTOR
165 000146' 062700 000316 ADD #TIMINT-.,RO
166 000152' 010037 000134 MOV RO,#SANVEC
167 000156' 012737 000240 000136 MOV #PRI05,#SANVEC+2
168
169 000164' 010700 MOV PC,RO ;SETUP TRANSMITTER VECTOR
170 000166' 062700 000414 ADD #TRNINT-.,RO
171 000172' 010037 000070 MOV RO,#TRNVEC
172 000176' 012737 000200 000072 MOV #PRI04,#TRNVEC+2
173
174 000204' 010700 MOV PC,RO ;SETUP RECEIVER VECTOR
175 000206' 062700 000360 ADD #RCVINT-.,RO
176 000212' 010037 000120 MOV RO,#RCVVEC
177 000216' 012737 000240 000122 MOV #PRI05,#RCVVEC+2
178
179 000224' 013700 021040 MOV #PCSRSW,RO ;GET SWITCH PACK BITS
180 000230' 052700 176000 BIS #176000,RO ;MAP THEM INTO HOST I/O PAGE
181 000234' 006300 ASL RO ;SHIFT OVER TO POSITION CORRECTLY
182 000236' 006300 ASL RO
183 000240' 006300 ASL RO
184 000242' 062700 000004 ADD #4,RO ;PCSR2 IS PCSR0+4
185 000246' 010067 000356 MOV RO,IPCSR2 ;SAVE PCSR2 ADDRESS
186 000252' 012767 000003 000352 MOV #3,IPCSR2+2 ;HIGH ORDER BITS 17:16
187 000260' 005067 000336 CLR FLG5 ;INITIALIZE FLAG WORD

```



76MICROE - MICROCODE MODULE E MACY11 30A(1052) 07-APR-83 16:51 PAGE 6  
 MICROE.MAC 07-APR-83 16:06 E\_MODULE MICROCODE

```

188 000264' 106427 000000      15$:  MTPS    #PRI00      :ALLOW INTERRUPTS
189
190 000270' 005767 000326      20$:  TST     FLG5      :WAIT FOR A COMMAND FROM HOST
191 000274' 001775      BEQ     20$
192
193 000276' 106427 000340      MTPS    #PRI07      :RAISE CPU PRIORITY TO SERVICE COMMAND
194 000302' 032767 000001 000312  BIT     #CSRFLG,FLG5 :DID HOST GIVE US A COMMAND?
195 000310' 001001      BNE     30$          :YES
196 000312' 000777      BR      .            :NO, ERROR SO JUST SIT HERE...
197
198
199 000314' 113700 021000      30$:  MOVB    @#IPCSRO,RO :GET WHAT HOST WROTE TO PCSRO
200 000320' 042700 177760      BIC     #177760,RO  :STRIP ALL BUT COMMAND BITS
201 000324' 001004      BNE     35$          :WAS IT THE CLEAR FUNCTION?
202 000326' 012737 000001 021020  MOV     #INMON,@#IPCSR1 :YES, CLEAR OUT THE TEST # BITS
203 000334' 000432      BR      50$
204 000336' 022700 000017      35$:  CMP     #17,RO      :RESTART OPERATIONAL MICROCODE?
205 000342' 001432      BEQ     60$          :YES
206 000344' 162700 000001      SUB     #1,RO
207 000350' 010701      MOV     PC,R1        :GET ADDRESS OF OUR COMMAND TABLE
208 000352' 062701 000240      ADD     #TBLD-.,R1
209 000356' 006300      ASL     RO
210 000360' 060001      ADD     RO,R1        :MAKE COMMAND A BYTE OFFSET
211 000362' 061101      ADD     (R1),R1      :USE IT TO INDEX INTO COMMAND TABLE
212 000364' 004711      JSR     PC,(R1)      :R1 NOW HAS COMMAND ROUTINE ADDRESS
213 000366' 103404      BCS     40$          :EXECUTE AS COMMANDED FROM HOST
214 000370' 112767 000001 000230  MOVB    #INMON,PCSR1 :ERROR OCCURRED
215 000376' 000403      BR      45$          :INDICATE TO HOST WE ARE BACK IN...
216 000400' 112767 000003 000220  40$:  MOVB    #INERR,PCSR1 :MICROMONITR
217 000406' 016737 000214 021020  45$:  MOV     PCSR1,@#IPCSR1 :INDICATE TO HOST ERROR OCCURRED
218 000414' 012737 004000 021000  MOV     #DNI,@#IPCSRO :TELL HOST THIS MICROTEST FINISHED
219 000422' 005067 000174      50$:  CLR     FLG5        :RESET FLAG WORD
220 000426' 000716      BR      15$          :GO WAIT FOR ANOTHER COMMAND
221
222 000430' 005000      60$:  CLR     RO          :FAKE SUCCESSFUL SELF TEST RESULTS
223 000432' 000137 040006      JMP     @#40006      :START OPERATIONAL MICROCODE
224
225 000436' 052767 000001 000156  CSRWRT: BIS     #CSRFLG,FLG5 :INDICATE A CSR WRITE INTERRUPT OCCURED
226 000444' 000002      RTI
227
228 000446' 052767 000002 000146  ERRINT: BIS     #ERRFLG,FLG5 :INDICATE A UNEXPECTED INTERRUPT OCCURED
229 000454' 012737 020000 021000  MOV     #UNIERR,@#IPCSRO :TELL HOST AN UNEXPECTED INTERRUPT
230
231 000462' 000777      BR      .            :HAPPENED
232
233
234 000464' 005267 000134      TIMINT: INC     SANTIM    :JUST SIT HERE AND SPIN WHEELS
235 000470' 000002      RTI                  :COUNT TICKS AS THEY OCCUR
236
237 000472' 013767 021002 000140  DMAINT: MOV     @#DMACSR,DMDONE :GET DMA STATUS
238 000500' 032767 040000 000132  BIT     #BIT14,DMDONE :DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
239 000506' 001404      BEQ     10$          :NO
240 000510' 012737 040000 021000  MOV     #NXMERR,@#IPCSRO :YES, TELL HOST A NON-EXISTANT MEMORY
241
242 000516' 000407      BR      20$          :LOCATION WAS ADDRESSED
243 000520' 032767 100000 000112  10$:  BIT     #BIT15,DMDONE :DID A NPR TIMEOUT OCCUR?
    
```

76MICROE - MICROCODE MODULE E MACY11 30A(1052) 07-APR-83 16:51 PAGE 7  
MICROE.MAC 07-APR-83 16:06 E\_MODULE MICROCODE

```

244 000526' 001407          BEQ      30$          ;NO
245 000530' 012737 100000 021000      MOV      #NPRERR,@#IPCSRO ;TELL HOST NPR TIMEOUT HAPPENED
246 000536' 012737 100000 021002 20$:  MOV      #BIT15,@#DMACSR ;CLEAR THE INTERRUPT IN THE DMA ENGINE
247 000544' 000777          BR           .          ;SIT HERE AND SPIN WHEELS
248 000546' 000002          30$:  RTI
249
250 000550' 052767 000004 000044  PARINT: BIS      #PARFLG,FLG5 ;SET PARITY ERROR OCCURRED
251 000556' 012737 010000 021000      MOV      #PARERR,@#IPCSRO ;TELL HOST A LINK MEMORY PARITY ERROR
252                                     .          OCCURRED
253 000564' 000777          BR           .          ;SIT HERE AND SPIN WHEELS
254
255 000566' 005737 021044          RCVINT: TST    @#LRBUF ;READ BUFFER DONE REGISTER...
256                                     .          WHICH CLEARS THE INTERRUPT
257 000572' 052767 000100 000022      BIS      #RCVFLG,FLG5 ;SET RECEIVER INTERRUPT OCCURRED
258 000600' 000002          RTI
259
260 000602' 052767 000040 000012  TRNINT: BIS      #TRNFLG,FLG5 ;SET TRANSMITTER INTERRUPT OCCURRED
261 000610' 000002          RTI
262
263 000612' 000034          TBLD:  .WORD  MICE1-. ; STATION ADDRESS PATTERN
264 000614' 000614          .WORD  MICE2-. ; STATION ADDRESS REJECTION
265 000616' 001064          .WORD  MICE3-. ; STATION ADDRESS POSITION
266 000620' 001620          .WORD  MICE4-. ; MULTICAST ADDRESS TEST
267
268 000622' 000000          FLG5:  .WORD  0 ;FLAG WORD
269 000624' 000000          SANTIM: .WORD 0 ;COUNT FOR SANITY TIMER
270 000626' 000000          PCSR1: .WORD 0 ;COPY OF WHAT GOES TO PCSR1
271 000630' 000000 000000 000000  IPCSR2: .WORD 0,0 ;ADDRESS IN HOST MEMORY FOR PCSR2
272 000634' 000000 000000 000000  PCBADR: .WORD 0,0 ;ADDRESS IN HOST MEMORY FOR PCB
273 000640' 000000          DMDONE: .WORD 0
274 000642' 000000          RBUF:  .WORD 0 ;POINTER TO RECIEVE BUFFER
275 000644' 000000          TBUF:  .WORD 0 ;POINTER TO XMIT BUFFER
276

```

MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 8  
E\_MODULE MICROCODE

SEQ 427

```

277
278
279
280
281
282
283
284
285
286
287
288
289
290
291 000646' 112767 000002 177752 MICE1: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
292 000654' 016737 177746 021020 MOV PCSR1,@#IPCSR1
293
294
295
296
297
298
299 000662' 016737 177742 021010 MOV IPCSR2,@#MDMA0 ; PICK UP ADDRESS OF PCBB
300 000670' 016737 177736 021012 MOV IPCSR2+2,@#MDMA1
301 000676' 013700 021014 MOV @#MDMAR0,R0 ; R0=CONTENTS OF PCSR2
302 000702' 013701 021014 MOV @#MDMAR0,R1 ; R1=CONTENTS OF PCSR3
303
304 000706' 010037 021010 MOV R0,@#MDMA0 ; POINT TO HOST PCBB
305 000712' 010137 021012 MOV R1,@#MDMA1
306
307 000716' 013702 021014 MOV @#MDMAR0,R2 ; R2 NOW HOLDS LS ADDRESS WORD
308
309 000722' 062700 000002 ADD #2,R0 ; INDEX TO NEXT HOST WORD
310 000726' 005501 ADC R1
311
312 000730' 013703 021014 MOV @#MDMAR0,R3 ; R3 NOW HOLDS MIDDLE PATTERN
313
314 000734' 062700 000002 ADD #2,R0 ; INDEX TO NEXT HOST WORD
315 000740' 005501 ADC R1
316
317 000742' 013704 021014 MOV @#MDMAR0,R4 ; R4 NOW HOLDS MS PATTERN WORD
318
319 000746' 010446 MOV R4,-(SP) ; SAVE THEM FOR LATER
320 000750' 010344 MOV R3,-(SP)
321 000752' 010246 MOV R2,-(SP)
322
323
324
325
326
327
328 000754' 010700 MOV PC,R0 ; POINT AT SA FILE
329 000756' 062700 000342 ADD #AFIL-.,R0
330
331 000762' 012705 000014 MOV #12.,R5 ; NEED TOTAL OF 12 ENTRIES
332 000766' 010220 108: MOV R2,(R0)+ ; LS WORD OF PATTERN

```

```

:SBTTL STATION ADDRESS PATTERN TEST
:
: MICROCODE FOR STATION ADDRESS PATTERN TEST. GETS A 'PATTERN' TO BE
: USED AS A STATION ADDRESS FOR TESTING THE STATION ADDRESS RAM AND
: DETECTION CIRCUITRY. FILLS THE STATION ADDRESS RAM. LOOPS A DATAGRAM
: WITH A STATION ADDRESS IDENTICAL TO THE ADDRESS FILLING STATION
: ADDRESS RAM. IF ALL IS WORKING, DATAGRAM SHOULD BE RECEIVED OK.

```

```

*****
***** TELL HIM WE ARE TESTING *****
*****

```

```

*****
***** PICK UP HOST ADDRESS OF PCBB *****
*****

```

```

*****
***** MAKE A STATION ADDRESS FILE FROM THE PASSED PATTERN *****
*****

```

76MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 9  
STATION ADDRESS PATTERN TEST

```

333 000770' 010320          MOV      R3,(R0)+      ; MIDDLE WORD OF PATTERN
334 000772' 010420          MOV      R4,(R0)+      ; MS WORD OF PATTERN
335 000774' 005305          DEC      R5              ; END OF LOOP
336 000776' 001373          BNE     10$
337
338
339
340
341
342
343 001000' 012737 000200 177776  MOV      #MODE,B#CMDREG      ; SET LOOPBACK BEFORE LOADING
344 001006' 012737 000004 177774  MOV      #LOOP,B#MODREG
345
346 001014' 010701          MOV      PC,R1          ; FORM POS/IND POINTER
347 001016' 062701 000302          ADD     #AFIL-.,R1
348
349 001022' 005000          CLR     R0              ; THIS WILL CLEAR HIGH BITS
350
351 001024' 012746 000120          MOV     #ARAM+20,-(SP)     ; SA RAM STARTS AT +20 LOCATIONS
352
353 001030' 012702 000003          MOV     #3,R2           ; 3 WORDS PER ADDRESS/PATTERN
354 001034' 012704 000020 20$:   MOV     #16.,R4          ; SIXTEEN BITS PER WORD
355 001040' 012705 000014 30$:   MOV     #12.,R5          ; 12 POSITIONS IN SA RAM
356
357 001044' 010103          MOV     R1,R3           ; COPY THE POINTER
358
359 001046' 006013 40$:   ROR     (R3)            ; GET LSB OF ALL 12 ADDRESSES
360 001050' 006100          ROL     R0              ; R0 WILL HOLD ORTHOGONAL WORD
361 001052' 062703 000006          ADD     #6,R3           ; 6 BYTES PER ADDRESS/PATTERN
362 001056' 077505          SOB     R5,40$         ; GO TILL DONE
363
364 001060' 011637 177776          MOV     (SP),B#CMDREG     ; SET MODE TO WRITE SA RAM
365 001064' 010037 177774          MOV     R0,B#ADRREG     ; ORTHOGONAL WORD TO SA RAM
366 001070' 005216          INC     (SP)            ; BUMP STATION ADDRESS
367 001072' 077416          SOB     R4,30$         ; DO ANOTHER ONE
368
369 001074' 062701 000002          ADD     #2,R1           ; ADVANCE TO NEXT WORD
370 001100' 077223          SOB     R2,20$         ; LOOP TILL DONE
371
372 001102' 012600          MOV     (SP)+,R0        ; R0 IS A JUNK REGISTER
373
374
375
376
377
378
379 001104' 012700 100000          MOV     #LINADR,R0      ; RECEIVE BUFFER STARTS HERE
380 001110' 010067 177526          MOV     R0,RBUF
381 001114' 005020 50$:   CLR     (R0)+          ; FILL RECEIVE BUFFER WITH ZEROS
382 001116' 020027 104000          CMP     R0,#LINADR+SIZ1K ; FILL THE BUFFER
383 001122' 103774          BLO    50$             ; FILL ENTIRE BUFFER
384
385
386
387
388

```

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 10  
STATION ADDRESS PATTERN TEST

```

389
390 001124' 012703 000377      ;
391 001130' 010067 177510      ;      MOV      #0377,R3      ; WORST CASE FOR CLOCKING
392 001134' 010320      ;      MOV      R0,TBUF      ;
393 001136' 020027 110000      60$:      MOV      R3,(R0)+      ; FILL XMIT BUFFER WITH PATTERN
394 001142' 103774      ;      CMP      R0,#LINADR+SIZ2K      ; STOP AT TOP
395      ;      BLO      60$
396      ;
397      ;
398      ; *****
399      ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
400      ; *****
401 001144' 012737 100020 177776      ;      MOV      #ENABLE+20,@#CMDREG      ; LEAVE 20 IN COMMAND REGISTER
402      ;
403 001152' 016701 177466      ;      MOV      TBUF,R1      ; POINT AT XMIT BUFFER
404 001156' 005021      ;      CLR      (R1)+      ; CLEAR OUT STATUS WORD
405 001160' 012721 000100      ;      MOV      #MINBC,(R1)+      ; SET BYTE COUNT TO MIN ALLOWED
406      ;
407 001164' 012621      ;      MOV      (SP)+,(R1)+      ; GET BACK HIGH ADDRESS PATTERN
408 001166' 012621      ;      MOV      (SP)+,(R1)+
409 001170' 012621      ;      MOV      (SP)+,(R1)+      ; GET BACK LOW ADDRESS PATTERN
410      ;
411 001172' 005037 021034      ;      CLR      @#CLRFIF      ; CLEAR THE FIFO
412 001176' 005067 177420      ;      CLR      FLG5      ; CLEAR INTERRUPT FLAG
413 001202' 016737 177434 021032      ;      MOV      RBUF,@#LFRBUF      ; TELL UNA WHERE RECEIVE BUF IS
414 001210' 016737 177430 021030      ;      MOV      TBUF,@#LTAC      ; TELL UNA WHERE XMIT BUF IS
415      ;
416 001216' 106427 000140      ;      MTPS     #PRIOS      ; ALLOW XMIT AND REC TO INTER
417 001222' 112767 000003 177376      ;      MOV      #INERR,PCSR1      ; TELL HOST IN CASE OF FAILURE
418 001230' 016737 177372 021020      ;      MOV      PCSR1,@#IPCSR1
419      ;
420 001236' 016701 177402      ;      MOV      TBUF,R1      ; WAIT FOR MATCH BIT FIRST
421 001242' 011102      65$:      MOV      (R1),R2
422 001244' 032702 020000      ;      BIT      #MATCH,R2
423 001250' 001774      ;      BEQ      65$
424      ;
425 001252' 112767 000002 177346      ;      MOV      #INTST,PCSR1      ; TELL HOST WE GOT BY IT
426 001260' 016737 177342 021020      ;      MOV      PCSR1,@#IPCSR1
427      ;
428 001266' 032767 000100 177326      70$:      BIT      #RCVFLG,FLG5      ; WAIT FOR RECEIVE INTER
429 001274' 001774      ;      BEQ      70$
430      ;
431 001276' 032767 000040 177316      80$:      BIT      #TRNFLG,FLG5      ; WAIT FOR XMIT INT TOO!
432 001304' 001774      ;      BEQ      80$
433      ;
434 001306' 112767 000001 177313      ;      MOV      #1,PCSR1+1      ; TELL HOST WE ARE DONE
435 001314' 000241      ;      CLC
436 001316' 000207      ;      RTS      PC
437      ;
438      ;
439 001320' 000000 000000 000000      AF FILE: .WORD 0,0,0      ; WORD #1
440 001326' 000000 000000 000000      .WORD 0,0,0
441 001334' 000000 000000 000000      .WORD 0,0,0
442 001342' 000000 000000 000000      .WORD 0,0,0
443 001350' 000000 000000 000000      .WORD 0,0,0
444 001356' 000000 000000 000000      .WORD 0,0,0

```

77MICROE - MICROCODE MODULE E MACY11 30A(1052) 07-APR-83 16:51 PAGE 11  
MICROE.MAC 07-APR-83 16:06 STATION ADDRESS PATTERN TEST

445	001364'	000000	000000	000000	.WORD	0,0,0
446	001372'	000000	000000	000000	.WORD	0,0,0
447	001400'	000000	000000	000000	.WORD	0,0,0
448	001406'	000000	000000	000000	.WORD	0,0,0
449	001414'	000000	000000	000000	.WORD	0,0,0
450	001422'	000000	000000	000000	.WORD	0,0,0

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 12  
STATION ADDRESS PATTERN TEST

```

451
452
453
454
455
456
457
458
459
460
461
462
463
464 001430' 112767 000002 177170 MICE2: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
465 001436' 016737 177164 021020 MOV PCSR1,&#IPCSR1
466
467 001444' 005067 177154 : CLR SANTIM ; CLEAR FLAG FOR TIMER
468
469
470
471
472
473
474 001450' 012737 000200 177776 : MOV #MODE,&#CMDREG ; SET LOOP TO LOAD SA RAM
475 001456' 012737 000004 177774 : MOV #LOOP,&#MODREG
476
477 001464' 012704 000060 : MOV #48.,R4 ; COUNTER
478 001470' 005000 : CLR R0
479
480 001472' 012701 000120 : MOV #ARAM+20,R1 ; STATION ADDRESS STARTS AT +20
481 001476' 010137 177776 10$: MOV R1,&#CMDREG ; SELECT STATION ADDRESS RAM
482 001502' 010037 177774 : MOV R0,&#ADRREG ; PARK IN RAM
483 001506' 005201 : INC R1 ; NEXT LOCATION
484 001510' 077406 : SOB R4,10$ ; DO THEM ALL
485
486
487
488
489
490
491 001512' 012700 100000 : MOV #LINADR,R0 ; RECEIVE BUFFER STARTS HERE
492 001516' 010067 177120 : MOV R0,RBUF
493 001522' 005020 20$: CLR (R0)+ ; FILL RECEIVE BUFFER WITH ZEROS
494 001524' 020027 104000 : CMP R0,#LINADR+SIZ1K ; FILL THE BUFFER
495 001530' 103774 : BLO 20$ ; FILL ENTIRE BUFFER
496
497
498
499
500
501
502 001532' 012703 000377 : MOV #0377,R3 ; WORST CASE FOR CLOCKING
503 001536' 010067 177102 : MOV R3,TBUF
504 001542' 010320 30$: MOV R3,(R0)+ ; FILL XMIT BUFFER WITH PATTERN
505 001544' 020027 110000 : CMP R0,#LINADR+SIZ2K ; STOP AT TOP
506 001550' 103774 : BLO 30$

```

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 13  
STATION ADDRESS REJECTION TEST

SEQ 432

```

507
508
509
510
511
512
513 001552' 012737 100020 177776      MOV    #ENABLE+20,0#CMDREG    ; POINTER TO FRONT OF RAM
514
515 001560' 016701 177060      MOV    TBUF, R1              ; POINT AT XMIT BUFFER
516 001564' 005021              CLR    (R1)+                 ; CLEAR OUT STATUS WORD
517 001566' 012721 000100      MOV    #MINBC, (R1)+        ; SET BYTE COUNT TO MIN ALLOWED
518
519
520
521
522
523
524 001572' 012700 177777      MOV    #177777, R0          ; && STAND-IN FOR PHYSICAL ADDR
525 001576' 010021              MOV    R0, (R1)+
526 001600' 010021              MOV    R0, (R1)+
527 001602' 010021              MOV    R0, (R1)+
528
529
530 001604' 005037 021034      CLR    0#CLRFIF             ; CLEAR THE FIFO
531 001610' 005067 177006      CLR    FLG5                 ; CLEAR INTERRUPT FLAG
532 001614' 016737 177022 021032  MOV    RBUF, 0#LFRBUF       ; TELL UNA WHERE RECEIVE BUF IS
533 001622' 016737 177016 021030  MOV    TBUF, 0#LTAC         ; TELL UNA WHERE XMIT BUF IS
534
535 001630' 106427 000140      MTPS   #PRI03               ; ALLOW XMIT AND REC TO INTER
536 001634' 026727 176764 000002 40S:  CMP    SANTIM, #2           ; WAIT FOR 2 SECONDS
537 001642' 002012              BGE    50S                  ; EXIT NORMALLY IF TIMER DONE
538 001644' 032767 000100 176750  BIT    #RCVFLG, FLG5       ; WAIT FOR RECEIVER INTERRUPT
539 001652' 001770              BEQ    40S                  ; IF NONE, WE'RE OK
540
541
542
543
544
545
546 001654' 032767 000040 176740 45S:  BIT    #TRMFLG, FLG5       ; WAIT FOR XMIT INT TOO!
547 001662' 001774              BEQ    45S
548
549 001664' 000261              SEC                               ; INDICATE ERROR
550 001666' 000401              BR     60S
551
552
553
554
555
556
557 001670' 000241 50S:  CLC                               ; INDICATE SUCCESS
558 001672' 112767 000002 176727 60S:  MOVB   #2, PCSR1+1        ; TELL HOST WE ARE DONE
559 001700' 000207              RTS    PC
560
561

```



77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 14  
STATION ADDRESS REJECTION TEST

562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617

: SBTTL STATION ADDRESS RAM POSITION TEST

THIS TEST WILL CHECK THAT A STATION ADDRESS IS RECOGNIZED  
REGARDLESS OF WHICH OF THE 12 RAM POSITIONS THE ADDRESS  
RESIDES. THE PHYSICAL ADDRESS WILL BE WRITTEN TO EACH OF THE 12  
STATION ADDRESS RAM POSITIONS WITH THE REST OF THE POSITIONS  
FILLED WITH KNOWN DATA. A DATAGRAM WITH THE PHYSICAL ADDRESS  
WILL BE LOOPED AROUND. THE TEST WILL VERIFY THE DATAGRAM IS  
RECEIVED.

\*\*\*\*\*  
\*\*\*\*\* TELL HIM WE ARE TESTING \*\*\*\*\*  
\*\*\*\*\*

```
MICE3: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
        MOV PCSR1,@#IPCSR1
        CLR SANTIM ; CLEAR THE FLAG
```

\*\*\*\*\*  
\*\*\*\*\* PICK UP HOST ADDRESS OF PCBB \*\*\*\*\*  
\*\*\*\*\*

```
        MOV IPCSR2,@#NDMA0 ; PICK UP ADDRESS OF PCBB
        MOV IPCSR2+2,@#NDMA1
        MOV @#NDMA0,R0 ; R0=CONTENTS OF PCSR2
        MOV @#NDMA0,R1 ; R1=CONTENTS OF PCSR3
        MOV R0,@#NDMA0 ; POINT TO HOST PCBB
        MOV R1,@#NDMA1
        MOV @#NDMA0,R2 ; R2 NOW HOLDS SA RAM POSITION
```

\*\*\*\*\*  
\*\*\*\*\* FILL THE BUFFER WITH PHONEY STATION ADDRESSES \*\*\*\*\*  
\*\*\*\*\*

```
        MOV PC,R0 ; CALCULATE POS/IND ADDRESS
        ADD #BFILE-.,R0
        MOV #36.,R5 ; THERE ARE 12 ADDRESSES
10$: CLR (R0)+ ; FILL WITH ZERO (ADDRESS)
        DEC R5
        BNE 10$
```

\*\*\*\*\*  
\*\*\*\*\* PUT PHYSICAL ADDRESS IN RAM POSITION \*\*\*\*\*  
\*\*\*\*\*

```
        MOV PC,R0 ; CALCULATE POS/IND ADDRESS
        ADD #BFILE-.,R0
```

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 15  
STATION ADDRESS RAM POSITION TEST

```

618
619 002010' 005302      20$: DEC      R2          ; PARAMTER IS NEVER > 1
620 002012' 001403      BEQ      30$          ; GO TO WORK
621 002014' 062700 000006 ADD      #5,R0
622 002020' 000773      BR       20$          ; GO ROUND
623
624 002022' 012701 177777 30$: MOV      #177777,R1    ; ## STAND-IN FOR PHSICAL ADDR
625 002026' 010120      MOV      R1,(R0)+
626 002030' 010120      MOV      R1,(R0)+
627 002032' 010120      MOV      R1,(R0)+
628
629
630
631
632
633
634 002034' 012737 000200 177776 MOV      #MODE,#CMDREG    ; SET LOOP TO LOAD SA RAM
635 002042' 012737 000004 177774 MOV      #LOOP,#MODREG
636
637 002050' 010701      MOV      PC,R1          ; FORM A POS/IND ADDRES
638 002052' 062701 000256 ADD      #BFILE-.,R1    ; ADD THE GFFSET
639
640 002056' 005000      CLR      R0            ; THIS CLEARS HIGH BITS
641
642 002060' 012746 000120 MOV      #ARAM+20,-(SP)  ; FOR MODE REG/NEED A REGISTER
643
644 002064' 012702 000003 MOV      #3,R2          ; 3 WORDS PER ADDRESS/PATTERN
645 002070' 012704 000020 40$: MOV      #16.,R4      ; SIXTEEN BITS PER WORD
646 002074' 012705 000014 50$: MOV      #12.,R5      ; 12 POSITIONS IN SA RAM
647
648 002100' 010103      MOV      R1,R3          ; COPY THE POINTER
649
650 002102' 006013      60$: ROR      (R3)        ; GET LSD OF ALL 12 ADDRESSES
651 002104' 006100      ROL      R0            ; R0 WILL HOLD ORTHOGONAL WORD
652 002106' 062703 000006 ADD      #6,R3          ; 6 BYTES PER ADDRES/PATTERN
653 002112' 077505      SOB      R5,60$        ; GO TILL DONE
654
655 002114' 011637 177776 MOV      (SP),#CMDREG    ; SET MODE TO WRITE SA RAM
656 002120' 010037 177774 MOV      R0,#ADRREG     ; ORTHOGONAL WORD TO SA RAM
657 002124' 005216      INC      (SP)          ; BUMP STATION ADDRESS
658 002126' 077416      SOB      R4,50$        ; DO ANOTHER ONE
659
660 002130' 062701 000002 ADD      #2,R1          ; ADVANCE TO NEXT WORD
661 002134' 077223      SOB      R2,40$        ; LOOP TILL DONE
662
663 002136' 012600      MOV      (SP)+,R0      ; POP TO BYTE BUCKET
664
665
666
667
668
669
670 002140' 012700 100000 MOV      #LINADR,R0     ; RECEIVE BUFFER STARTS HERE
671 002144' 010067 176472 MOV      R0,RBUFF
672 002150' 005020      70$: CLR      (R0)+        ; FILL RECEIVE BUFFER WITH ZEROS
673 002152' 020027 104000 CMP      R0,#LINADR+SIZ1K ; FILL THE BUFFER

```

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 16  
STATION ADDRESS RAM POSITION TEST

```

674 002156' 103774          BLO      70$          ; FILL ENTIRE BUFFER
675
676
677
678
679
680
681 002160' 012703 000377   MOV      #0377,R3          ; WORST CASE FOR CLOCKING
682 002164' 010067 176454   MOV      R0,TBUFF        ;
683 002170' 010320          80$:  MOV      R3,(R0)+        ; FILL XMIT BUFFER WITH PATTERN
684 002172' 020027 110000   CMP      R0,#LINADR+SIZ2K ; STOP AT TOP
685 002176' 103774          BLO      80$
686
687
688
689
690
691
692 002200' 012737 100020 177776  MOV      #ENABLE+20,&#CMDREG ; ENABLE LINK
693
694 002206' 016701 176432   MOV      TBUFF,R1        ; POINT AT XMIT BUFFER
695 002212' 005021          CLR      (R1)+           ; CLEAR OUT STATUS WORD
696 002214' 012721 000100   MOV      #MINBC,(R1)+    ; SET BYTE COUNT TO MIN ALLOWED
697
698 002220' 012700 177777   MOV      #177777,R0      ; && STAND IN PHYSICAL ADDRESS
699 002224' 010021          MOV      R0,(R1)+
700 002226' 010021          MOV      R0,(R1)+
701 002230' 010021          MOV      R0,(R1)+
702
703 002232' 005037 021034   CLR      &#CLRIFIF      ; CLEAR THE FIFO
704 002236' 005067 176360   CLR      FLG5           ; CLEAR INTERRUPT FLAG
705 002242' 016737 176374 021032  MOV      RBUF,&#LFRBUF   ; TELL UNA WHERE RECEIVE BUF IS
706 002250' 016737 176370 021030  MOV      TBUFF,&#LTAC    ; TELL UNA WHERE XMIT BUF IS
707
708 002256' 106427 000140   MTPS    #PRI03          ; ALLOW XMIT AND REC TO INTER
709 002262' 026727 176336 000002 90$:  CMP      SANTI,#2        ; WAIT FOR COUNTS TO ACCUMULATE
710 002270' 002015          BGE     120$
711 002272' 032767 000100 176322  BIT      #RCVFLG,FLG5    ; THIS SETS IF DATAGRAM LOOPS
712 002300' 001770          BEQ     90$
713
714 002302' 032767 000040 176312 100$: BIT      #TRNFLG,FLG5    ; WAIT FOR XMIT TOO
715 002310' 001774          BEQ     100$
716
717 002312' 000241          CLC
718 002314' 112767 000003 176305 110$: MOVB    #3,PCSR1+1      ; TELL HIM WHAT TEST WE'RE IN
719 002322' 000207          RTS      PC
720
721
722 002324' 000261          120$:  SEC                ; INDICATE ERROR
723 002326' 000772          BR      110$          ; GO BACK TO MICROMONITOR
724
725
726 002330' 000000 000000 000000 000000 BFILE:: .WORD 0,0,0
727 002336' 000000 000000 000000 000000   .WORD 0,0,0
728 002344' 000000 000000 000000 000000   .WORD 0,0,0
729 002352' 000000 000000 000000 000000   .WORD 0,0,0

```

77MICROE - MICROCODE MODULE E    MACY11 30A(1052) 07-APR-83 16:51 PAGE 17  
MICROE.MAC    07-APR-83 16:06    STATION ADDRESS RAM POSITION TEST

SEQ 436

730	002360'	000000	000000	000000	.WORD	0,0,0
731	002366'	000000	000000	000000	.WORD	0,0,0
732	002374'	000000	000000	000000	.WORD	0,0,0
733	002402'	000000	000000	000000	.WORD	0,0,0
734	002410'	000000	000000	000000	.WORD	0,0,0
735	002416'	000000	000000	000000	.WORD	0,0,0
736	002424'	000000	000000	000000	.WORD	0,0,0
737	002432'	000000	000000	000000	.WORD	0,0,0

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 18  
STATION ADDRESS RAM POSITION TEST

```

738
739
740
741
742
743
744
745 002440' 112767 000002 176160 MICE4: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
746 002446' 016737 176154 021020 MOV PCSR1,#IPCSR1
747
748 002454' 005067 176144 CLR SANTIM ; CLEAR FLAG FOR TIMER
749
750
751
752
753
754
755 002460' 012737 000200 177776 MOV #MODE,#CMDREG ; SET LOOP TO LOAD SA RAM
756 002466' 012737 000004 177774 MOV #LOOP,#MODREG
757
758 002474' 012704 000060 MOV #48.,R4 ; COUNTER
759 002500' 005000 CLR R0
760
761 002502' 012701 000120
762 002506' 010137 177776 10$: MOV #ARAM+20,R1 ; STATION ADDRESS STARTS AT +20
763 002512' 010037 177774 MOV R1,#CMDREG ; SELECT STATION ADDRESS RAM
764 002516' 005201 MOV R0,#ADRREG ; PARK IN RAM
765 002520' 077406 INC R1 ; NEXT LOCATION
SOB R4,10$ ; DO THEM ALL
766
767
768
769
770
771
772 002522' 012700 100000 MOV #LINADR,R0 ; RECEIVE BUFFER STARTS HERE
773 002526' 010067 176110 MOV R0,RBUF
774 002532' 005020 20$: CLR (R0)+ ; FILL RECEIVE BUFFER WITH ZEROS
775 002534' 020027 104000 CMP R0,#LINADR+SIZ1K ; FILL THE BUFFER
776 002540' 103774 BLO 20$ ; FILL ENTIRE BUFFER
777
778
779
780
781
782
783 002542' 012703 000377 MOV #0377,R3 ; WORST CASE FOR CLOCKING
784 002546' 010067 176072 MOV R0,TBUF
785 002552' 010320 30$: MOV R3,(R0)+ ; FILL XMIT BUFFER WITH PATTERN
786 002554' 020027 110000 CMP R0,#LINADR+SIZ2K ; STOP AT TOP
787 002560' 103774 BLO 30$
788
789
790
791
792
793

```

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 19  
MULTICAST ADDRESS TEST

```

794 002562' 012737 000200 177776      MOV      #MODE,@#CMDREG      ; ENABLE LINK, SELECT MODE
795 002570' 012737 040004 177774      MOV      #ENAL!LOOP,@#MODREG ; ENABLE LOOPBACK
796 002576' 012737 100020 177776      MOV      #ENABLE+20,@#CMDREG ; POINTER TO FRONT OF RAM
797                                     ;
798 002604' 016701 176034      MOV      TBUF,R1             ; POINT AT XMIT BUFFER
799 002610' 005021                                     CLR      (R1)+               ; CLEAR OUT STATUS WORD
800 002612' 012721 000100      MOV      #MINBC,(R1)+       ; SET BYTE COUNT TO MIN ALLOWED
801                                     ;
802                                     ;
803                                     ;
804                                     ; *****
805                                     ; ***** GET PHYSICAL ADDRESS INTO DATAGRAM *****
806                                     ; *****
807 002616' 012700 177777      MOV      #177777,R0         ; && STAND-IN FOR PHYSICAL ADDR
808 002622' 010011                                     MOV      R0,(R1)
809 002624' 052721 000001      BIS      #01,(R1)+         ; SET MULTICAST ADDRESS BIT
810 002630' 010021                                     MOV      R0,(R1)+
811 002632' 010021                                     MOV      R0,(R1)+
812                                     ;
813                                     ;
814 002634' 005037 021034      CLR      @#CLRFIF          ; CLEAR THE FIFO
815 002640' 005067 175756      CLR      FLG5              ; CLEAR INTERRUPT FLAG
816 002644' 016737 175772 021032      MOV      RBUF,@#LFRBUF     ; TELL UNA WHERE RECEIVE BUF IS
817 002652' 016737 175766 021030      MOV      TBUF,@#LTAC       ; TELL UNA WHERE XMIT BUF IS
818                                     ;
819 002660' 106427 000140      MTPS     #PRI03            ; ALLOW XMIT AND REC TO INTER
820 002664' 026727 175734 000002 40$:   CMP      SANTIM,#2         ; && LEAVE IT 2 SECONDS
821 002672' 002015                                     BGE      60$
822 002674' 032767 000100 175720      BIT      #RCVFLG,FLG5     ; EXIT IF TIMER DONE
823 002702' 001770      BEQ      40$              ; WAIT FOR RECEIVER INTERRUPT
824                                     ;
825 002704' 032767 000040 175710 45$:   BIT      #TRMFLG,FLG5     ; WAIT FOR XMIT INT TOO!
826 002712' 001774      BEQ      45$
827                                     ;
828 002714' 000241                                     CLC
829 002716' 112767 000004 175703 50$:   MOVB     #4,PCSR1+1       ; INDICATE SUCCESS
830 002724' 000207      RTS      PC               ; TELL HOST WE ARE DONE
831                                     ;
832                                     ;
833 002726' 000261 60$:   SEC
834 002730' 000772      BR      50$              ; INDICATE ERROR
835                                     ;
836                                     ;

```

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 20  
MULTICAST ADDRESS TEST

SEQ 439

837  
838 002732' 002734  
839  
840 000001

⋮  
MICESZ::MICESZ-MICROE+2  
⋮  
.END







77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 24  
CROSS REFERENCE TABLE -- USER SYMBOLS

RCVFLG= 000100		109#	257	428	538	711	822							
RCVINT 000566R	002	175	255#											
RCVVEC= 000120		94#	176*	177*										
ROMADR= 040000		125#	126											
ROMSIZ= 040000		121#	126											
RXI = 020000		70#												
SANTIM 000624R	002	234*	269#	467*	536	580*	709	748*	820					
SANVEC= 000134		95#	166*	167*										
SERI = 100000		68#												
SIZ1K = 004000		115#	116	382	494	673	775							
SIZ2K = 010000		116#	117	393	505	684	786							
SIZ4K = 020000		117#	118	119	120									
SIZ8K = 040000		118#	121	122										
STACK = 001000		99#	137											
TBLD 000612R	002	208	263#											
TBUF 000644R	002	275#	391*	403	414	420	503*	515	533	682*	694	706	784*	798
		817												
TIMINT 000464R	002	165	234#											
TRNFLG= 000040		108#	260	431	546	714	825							
TRNINT 000602R	002	170	260#											
TRNVEC= 000070		93#	171*	172*										
Txi = 010000		71#												
UNIERR= 020000		112#	229											
WCSADR= 000000		123#	124											
WCSIZ= 020000		119#	124											
.	002	142	150	155	160	165	170	175	196	208	231	247	253	263
		264	265	266	329	347	603	616	638					

77MICROE - MICROCODE MODULE E  
MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 26  
CROSS REFERENCE TABLE -- MACRO NAMES

BAMPL	1#
BERROR	1#
BGAU	1#
BGAUT	1#
BGNCLN	1#
BGNDU	1#
BGNHRD	1#
BGNHJ	1#
BGNINI	1#
BGNMOD	1#
BGNMSG	1#
BGNPRO	1#
BGNPTA	1#
BGNRPT	1#
BGNSEG	1#
BGNSET	1#
BGNSFT	1#
BGNSRV	1#
BGNSUB	1#
BGNSU	1#
BGNTST	1#
BNCOMP	1#
BNERRO	1#
BREAK	1#
BRESET	1#
CKLOOP	1#
CLOCK	1#
CLOSE	1#
CLVEC	1#
COMMEN	1#
DELAY	1#
DESCRI	1#
DEVTYP	1#
DISPAT	1#
DISPLA	1#
DOCLN	1#
DODU	1#
DORPT	1#
ENDAU	1#
ENDAUT	1#
ENDCLN	1#
ENDCOM	1#
ENDDU	1#
ENDHRD	1#
ENDHJ	1#
ENDINI	1#
ENDMOD	1#
ENDMSG	1#
ENDPRO	1#
ENDPTA	1#
ENDRPT	1#
ENDSEG	1#
ENDSET	1#
ENDSFT	1#
ENDSRV	1#
ENDSUB	1#

77MICROE - MICROCODE MODULE E  
 MICROE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 27  
 CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	1#
ENDTST	1#
EQUALS	1#
ERRDF	1#
ERRHRD	1#
ERROR	1#
ERRSF	1#
ERRSOF	1#
ERRTBL	1#
ESCAPE	1#
EXIT	1#
FEQUAL	1#
GETBYT	1#
GETPRI	1#
GETWOR	1#
GMANIA	1#
GMANID	1#
GMANIL	1#
GPHARD	1#
GPRMA	1#
GPRMD	1#
GPRML	1#
HEADER	1#
INLOOP	1#
IOSETU	1#
IOSTAR	1#
KT11	1#
LASTAD	1#
MANUAL	1#
MEMORY	1#
MSBYTE	1#
MSCHEC	1#
MSCNTO	1#
MSCOUN	1#
MSDATA	1#
MSDEC	1#
MSDEFA	1#
MSENDE	1#
MSERRI	1#
MSESCA	1#
MSESCS	1#
MSEXCP	1#
MSEXIT	1#
MSXSE	1#
MSXTJ	1#
MSGEN	1#
MSGEND	1#
MSGETS	1#
MSGETT	1#
MSGNGD	1#
MSGNIN	1#
MSGNLS	1#
MSGNSU	1#
MSGNTA	1#
MSGNTE	1#
MSHAPT	1#

77MICROE - MICROCODE MODULE E MACY11 30A(1052) 07-APR-83 16:51 PAGE 28  
 MICROE.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

MSHNP 1#  
 MSINCR 1#  
 MSIOSE 1#  
 MSLDRO 1#  
 MSMASK 1#  
 MSACHI 1#  
 MSACLO 1#  
 MSASK! 1#  
 MSPOP 1#  
 MSPRIN 1#  
 MSPUSH 1#  
 MSPUT 1#  
 MSPUT1 1#  
 MSRADI 1#  
 MSRBRO 1#  
 MSRNRO 1#  
 MSSETS 1#  
 MSSTAR 1#  
 MSSVC 1#  
 MSTLAB 1#  
 MSTSTL 1#  
 MSWORD 1#  
 MSXFER 1#  
 OPEN 1#  
 POINTE 1#  
 PRINTB 1#  
 PRINTF 1#  
 PRINTS 1#  
 PRINTX 1#  
 READBU 1#  
 READEF 1#  
 RFLAGS 1#  
 SETPRI 1#  
 SETVEC 1#  
 SLASH 1#  
 STARS 1#  
 SVC 1#  
 XFER 1#  
 XFERF 1#  
 XFERT 1#

. ABS. 000000 000  
 000000 001  
 MICRE 002734 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MICROE.OBJ,MICROE.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROE.MAC  
 RUN-TIME: 2 3 .4 SECONDS  
 RUN-TIME RATIO: 33/6=5.1  
 CORE USED: 31K (61 PAGES)

76MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 2

```

1      .TITLE MICROF - MICROCODE MODULE F
2      ; 88 DEDICATED THE CRC CIRCUITRY TO THE RECEIVE SIDE OF THE LINK
3
4      .CSECT MICRF
5      000000'
6
7      .SBTTL REGISTER DEFINITIONS USED BY THE T11
8      021000 IPCSRO = 21000 ;INTERNAL PCSRO ADDRESS
9      021002 DMACSR = 21002 ;DMA ENGINE CONTROL STATUS REGISTER
10     021004 DMATO = 21004 ;DMA ENGINE TO ADDRESS REGISTER #0
11     021006 DMAT1 = 21006 ;DMA ENGINE TO ADDRESS REGISTER #1
12     021010 MDMA0 = 21010 ;MICROCPU DMA TO ADDRESS REGISTER #0
13     021012 MDMA1 = 21012 ;MICROCPU DMA TO ADDRESS REGISTER #1
14     021014 MDMA0 = 21014 ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15     021016 MDMA1 = 21016 ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16     021020 IPCSR1 = 21020 ;INTERNAL PCSR1 ADDRESS
17     021022 DMAF = 21022 ;DMA ENGINE FROM ADDRESS REGISTER
18     021024 DMAWC = 21024 ;DMA ENGINE WORD COUNT REGISTER
19     021026 MDMAW0 = 21026 ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20     021030 LTAC = 21030 ;LINK TRANSMIT ADDRESS COUNTER REGISTER
21     021032 LFRBUF = 21032 ;LINK RECIEVE BUFFER ADDRESS FIFO
22     021034 CLRFIF = 21034 ;CLEAR FIFO
23     021036 MDMAW1 = 21036 ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24     021040 PCSRSW = 21040 ;SWITCH PACK REGISTER
25     021042 MDMSR = 21042 ;MICROCPU DMA STATUS REGISTER
26     021044 LRBUF = 21044 ;LINK RECIEVE BUFFER COMPLETED
27     021060 PHYAD0 = 21060 ;PHYSICAL ADDRESS ROM BYTE 0
28     021062 PHYAD1 = 21062 ;PHYSICAL ADDRESS ROM BYTE 1
29     021064 PHYAD2 = 21064 ;PHYSICAL ADDRESS ROM BYTE 2
30     021066 PHYAD3 = 21066 ;PHYSICAL ADDRESS ROM BYTE 3
31     021070 PHYAD4 = 21070 ;PHYSICAL ADDRESS ROM BYTE 4
32     021072 PHYAD5 = 21072 ;PHYSICAL ADDRESS ROM BYTE 5
33     177774 MDREG = 177774 ;LINK MODE REGISTER
34     177774 ADRREG = 177774 ;LINK STATION ADDRESS RAM REGISTER
35     177776 CMDREG = 177776 ;LINK COMMAND REGISTER
36
37     .SBTTL OTHER DEFINITIONS USED BY THE MICROCODE
38
39     100000 BIT15 = 100000
40     040000 BIT14 = 40000
41     020000 BIT13 = 20000
42     010000 BIT12 = 10000
43     004000 BIT11 = 4000
44     002000 BIT10 = 2000
45     001000 BIT9 = 1000
46     000400 BIT8 = 400
47     000200 BIT7 = 200
48     000100 BIT6 = 100
49     000040 BIT5 = 40
50     000020 BIT4 = 20
51     000010 BIT3 = 10
52     000004 BIT2 = 4
53     000002 BIT1 = 2
54     000001 BIT0 = 1
55
56     012400 LASFTP = BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN

```

76MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 3  
OTHER DEFINITIONS USED BY THE MICROCODE

57	000340	PRI07 =	340	
58	000300	PRI06 =	300	
59	000240	PRI05 =	240	
60	000200	PRI04 =	200	
61	000140	PRI03 =	140	
62	000100	PRI02 =	100	
63	000040	PRI01 =	40	
64	000000	PRI00 =	0	
65		:		
66		;PCSR0 - PORT CONTROL STATUS REGISTER 0		
67		:		
68	100000	SERI =	BIT15	
69	040000	PCEI =	BIT14	
70	020000	RXI =	BIT13	
71	010000	TXI =	BIT12	
72	004000	DNI =	BIT11	
73	002000	RCEI =	BIT10	
74	000400	FATI =	BIT8	
75		:		
76		;LINK COMMAND REGISTER		
77		:		
78	100000	ENABLE =	BIT15	;ENABLE LINK MODULE
79	000200	MODE =	BIT7	;ENABLE MODE REGISTER
80	000100	ARAM =	BIT6	;ENABLE STATION ADDRESS RAM
81		:		
82		;LINK MODE REGISTER		
83		:		
84	100000	PROM =	BIT15	;PROMISCUIOUS MODE
85	040000	ENAL =	BIT14	;ENABLE MULTICAST
86	004000	ENCR =	BIT11	;ENABLE COLLISION TEST
87	002000	ACLO =	BIT10	;ENABLE ACLO
88	000040	DRTY =	BIT5	;DISABLE RETRY LOGIC
89	000020	COLL =	BIT4	;SIMULATE A COLLISION
90	000010	DTCR =	BIT3	;DISABLE TRANSMIT CRC LOGIC
91	000004	LOOP =	BIT2	;ENABLE LOOPBACK
92	000001	HDX =	BIT0	;HALF DUPLEX BIT
93		:		
94	000070	TRNVEC =	70	;VECTOR ADDRESS FOR THE TRANSMITTER
95	000120	RCVVEC =	120	;VECTOR ADDRESS FOR THE RECEIVER
96	000134	SANVEC =	134	;VECTOR ADDRESS FOR THE SANITY TIMER
97	000064	CSRVEC =	64	;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
98	000114	DMAVEC =	114	;VECTOR ADDRESS FOR DMA DONE INTERRUPT
99	000140	PARVEC =	140	;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
100	001000	STACK =	1000	;STACK LOCATION
101	000001	INMON =	1	;IN MICROMONITOR STATE
102	000002	INTST =	2	;IN A TEST STATE
103	000003	INERR =	3	;IN ERROR STATE
104	000001	CSRFLG =	BIT0	;CSR WRITE INTERRUPT OCCURED
105	000002	ERRFLG =	BIT1	;UNEXPECTED ERROR OCCURED
106	000004	PARFLG =	BIT2	;PARITY ERROR OCCURED
107	000010	NXMFLG =	BIT3	;NON-EXISTANT MEMORY ERROR OCCURED
108	000020	NPRFLG =	BIT4	;NPR TIMEOUT OCCURED
109	000040	TRNFLG =	BIT5	;TRANSMITTER INTERRUPT OCCURED
110	000100	RCVFLG =	BIT6	;RECEIVER INTERRUPT OCCURED
111	100000	NPRERR =	BIT15	;PCSR0 FLAG INDICATING NPR ERROR OCCURED
112	040000	NXMERR =	BIT14	;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURED

76MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

113	020000	UNIERR= BIT13	:PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
114	010000	PARERR= BIT12	:PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
115	004000	SIZ1K= 4000	:1K WORDS
116	010000	SIZ2K= SIZ1K*2	:2K WORDS
117	020000	SIZ4K= SIZ2K*2	:4K WORDS
118	040000	SIZ8K= SIZ4K*2	:8K WORDS
119	020000	WCSSIZ= SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
120	020000	IOSIZ= SIZ4K	:SIZE OF I/O PAGE
121	040000	ROMSIZ= SIZ8K	:SIZE OF ROM
122	077774	LINSIZ= SIZ8K*2-4	:SIZE OF LINK MEMORY
123	000000	WCSADR= 0	:BASE ADDRESS OF WCS
124	020000	IOADR= WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
125	040000	ROMADR= IOADR+IOSIZ	:BASE ADDRESS OF ROM
126	100000	LINADR= ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
127	000100	MINBC= 64.	: 64 BYTES
128	002752	MAXBC= 1518.-4.	:MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
129	000000	DATERR= 0	:FLAG INDICATING DATA ERROR OCCURRED
130	000001	ADRERR= 1	:FLAG INDICATING ADDRESS ERROR OCCURRED
131		.	
132	177777	INITH = -1	: MINUS ONE ( INITIAL CRC VALUE)
133	177777	INITL = -1	
134	166670	POLYH = 166670	: FUNCTION POLYNOMIAL HIGH WORD
135	101440	POLYL = 101440	: FUNCTION POLYNOMIAL LOW WORD
136	157273	EXCRCH = 157273	: EXPECTED CRC HIGH
137	020343	EXCRCL = 020343	: EXPECTED CRC LOW



76MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 5  
OTHER DEFINITIONS USED BY THE MICROCODE

```

138
139
140      .SBTTL  F_MODULE MICROCODE
141 000000' 106427 000340      MICROF::MTPS      #PRI07      ;DISABLE INTERRUPTS
142 000004' 012737 000000 177776      MOV      #0,#PCMDREG      ;TURN OFF THE LINK
143 000012' 012706 001000      MOV      #STACK,SP      ;SETUP STACK
144 000016' 112767 000001 000604      MOV      #INMON,PCSR1    ;TELL HOST WE ARE IN MICROMONITOR
145 000024' 016737 000600 021020      MOV      PCSR1,#IPCSR1
146 000032' 012737 004000 021000      MOV      #DNI,#IPCSR0    ;TELL HOST THE LOAD AND START FINISHED
147 000040' 010700      MOV      PC,RO      ;GET ADDRESS OF UNEXPECTED ERROR...
148 000042' 062700 000404      ADD      #ERRINT-.,RO    ;HANDLER
149 000046' 005001      CLR      R1      ;FILL ALL UNUSED VECTORS WITH TRAP...
150 000050' 010021      10$: MOV      RO,(R1)+      ;HANDLER
151 000052' 012721 000340      MOV      #PRI07,(R1)+
152 000056' 020127 001000      CMP      R1,#1000
153 000062' 002772      BLT
154
155 000064' 010700      MOV      PC,RO      ;SETUP PARITY TRAP VECTOR
156 000066' 062700 000462      ADD      #PARINT-.,RO
157 000072' 010037 000140      MOV      RO,#PARVEC
158 000076' 012737 000340 000142      MOV      #PRI07,#PARVEC+2
159
160 000104' 010700      MOV      PC,RO      ;SETUP DMA INTERRUPT VECTOR
161 000106' 062700 000364      ADD      #DMAINT-.,RO
162 000112' 010037 000114      MOV      RO,#DMAVEC
163 000116' 012737 000340 000116      MOV      #PRI07,#DMAVEC+2
164
165 000124' 010700      MOV      PC,RO      ;SETUP CSR WRITE VECTOR
166 000126' 062700 000310      ADD      #CSRWRT-.,RO
167 000132' 010037 000064      MOV      RO,#CSRVEC
168 000136' 012737 000200 000066      MOV      #PRI04,#CSRVEC+2
169
170 000144' 010700      MOV      PC,RO      ;SETUP SANITY TIMER VECTOR
171 000146' 062700 000316      ADD      #TIMINT-.,RO
172 000152' 010037 000134      MOV      RO,#SANVEC
173 000156' 012737 000240 000136      MOV      #PRI05,#SANVEC+2
174
175 000164' 010700      MOV      PC,RO      ;SETUP TRANSMITTER VECTOR
176 000166' 062700 000414      ADD      #TRNINT-.,RO
177 000172' 010037 000070      MOV      RO,#TRNVEC
178 000176' 012737 000200 000072      MOV      #PRI04,#TRNVEC+2
179
180 000204' 010700      MOV      PC,RO      ;SETUP RECEIVER VECTOR
181 000206' 062700 000360      ADD      #RCVINT-.,RO
182 000212' 010037 000120      MOV      RO,#RCVVEC
183 000216' 012737 000240 000122      MOV      #PRI05,#RCVVEC+2
184
185 000224' 013700 021040      MOV      #PCRSW,RO      ;GET SWITCH PACK BITS
186 000230' 052700 176000      BIS      #176000,RO      ;MAP THEM INTO HOST I/O PAGE
187 000234' 006300      ASL      RO      ;SHIFT OVER TO POSITION CORRECTLY
188 000236' 006300      ASL      RO
189 000240' 006300      ASL      RO
190 000242' 062700 000004      ADD      #4,RO      ;PCSR2 IS PCSR0+4
191 000246' 010067 000360      MOV      RO,IPCSR2      ;SAVE PCSR2 ADDRESS
192 000252' 012767 000003 000354      MOV      #3,IPCSR2+2    ;HIGH ORDER BITS 17:16
193 000260' 005067 000340      CLR      FLG6      ;INITIALIZE FLAG WORD

```

76MICROF - MICROCODE MODULE F MACY11 30A(1052) 07-APR-83 16:51 PAGE 6  
 MICROF.MAC 07-APR-83 16:06 F\_MODULE MICROCODE

194	000264'	106427	000000		15%:	MTPS	#PRI00		:ALLOW INTERRUPTS
195									
196	000270'	005767	000330		20%:	TST	FLG6		:WAIT FOR A COMMAND FROM HOST
197	000274'	001775				BEQ	20\$		
198									
199	000276'	106427	000340			MTPS	#PRI07		:RAISE CPU PRIORITY TO SERVICE COMMAND
200	000302'	032767	000001	000314		BIT	#CSRFLG,FLG6		:DID HOST GIVE US A COMMAND?
201	000310'	001001				BNE	30\$		:YES
202	000312'	000777				BR	.		:NO, ERROR SO JUST SIT HERE...
203									:FOR LACK OF ANYTHING BETTER TO DO
204									
205	000314'	113700	021000		30%:	MOVB	#IPCSRO,RO		:GET WHAT HOST WROTE TO PCSRO
206	000320'	042700	177760			BIC	#177760,RO		:STRIP ALL BUT COMMAND BITS
207	000324'	001004				BNE	35\$		:WAS IT THE CLEAR FUNCTION?
208	000326'	012737	000001	021020		MOV	#INMON,#IPCSR1		:YES, CLEAR OUT THE TEST # BITS
209	000334'	000432				BR	50\$		
210	000336'	022700	000017		35%:	CMP	#17,RO		:RESTART OPERATIONAL MICROCODE?
211	000342'	001432				BEQ	60\$		:YES
212	000344'	162700	000001			SUB	#1,RO		
213	000350'	010701				MOV	PC,R1		:GET ADDRESS OF OUR COMMAND TABLE
214	000352'	062701	000240			ADD	#TBLD-.,R1		
215	000356'	006300				ASL	RO		:MAKE COMMAND A BYTE OFFSET
216	000360'	060001				ADD	RO,R1		:USE IT TO INDEX INTO COMMAND TABLE
217	000362'	061101				ADD	(R1),R1		:R1 NOW HAS COMMAND ROUTINE ADDRESS
218	000364'	004711				JSR	PC,(R1)		:EXECUTE AS COMMANDED FROM HOST
219	000366'	103404				BCS	40\$		:ERROR OCCURRED
220	000370'	112767	000001	000232		MOVB	#INMON,PCSR1		:INDICATE TO HOST WE ARE BACK IN...
221	000376'	000403				BR	45\$		:MICROMONITR
222	000400'	112767	000003	000222	40%:	MOVB	#INERR,PCSR1		:INDICATE TO HOST ERROR OCCURRED
223	000406'	016737	000216	021020	45%:	MOV	PCSR1,#IPCSR1		
224	000414'	012737	004000	021000		MOV	#DN1,#IPCSRO		:TELL HOST THIS MICROTEST FINISHED
225	000422'	005067	000176		50%:	CLR	FLG6		:RESET FLAG WORD
226	000426'	000716				BR	15\$		:GO WAIT FOR ANOTHER COMMAND
227									
228	000430'	005000			60%:	CLR	RO		:FAKE SUCCESSFUL SELF TEST RESULTS
229	000432'	000137	040006			JMP	#40006		:START OPERATIONAL MICROCODE
230									
231	000436'	052767	000001	000160	CSRWRT:	BIS	#CSRFLG,FLG6		:INDICATE A CSR WRITE INTERRUPT OCCURED
232	000444'	000002				RTI			
233									
234	000446'	052767	000002	000150	ERRINT:	BIS	#ERRFLG,FLG6		:INDICATE A UNEXPECTED INTERRUPT OCCURED
235	000454'	012737	020000	021000		MOV	#UNIERR,#IPCSRO		:TELL HOST AN UNEXPECTED INTERRUPT
236									:HAPPENED
237									:JUST SIT HERE AND SPIN WHEELS
238	000462'	000777				BR	.		:COUNT ON HOST TO TIMEOUT
239									
240	000464'	005267	000136		TIMINT:	INC	SANTIM		:COUNT TICKS AS THEY OCCUR
241	000470'	000002				RTI			
242									
243	000472'	013767	021002	000142	DMaint:	MOV	#DMACSR,DMDONE		:GET DMA STATUS
244	000500'	032767	040000	000134		BIT	#BIT14,DMDONE		:DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
245	000506'	001404				BEQ	10\$		:NO
246	000510'	012737	040000	021000		MOV	#NXMERR,#IPCSRO		:YES, TELL HOST A NON-EXISTANT MEMORY
247									:LOCATION WAS ADDRESSED
248	000516'	000407				BR	20\$		
249	000520'	032767	100000	000114	10%:	BIT	#BIT15,DMDONE		:DID A NPR TIMEOUT OCCUR?



```

288
289
290
291
292
293
294
295 000656' 106746
296 000660' 010046
297 000662' 010146
298 000664' 010246
299 000666' 012704 177777
300 000672' 012705 177777
301 000676' 112001
302 000700' 004767 000014
303 000704' 077204
304 000706' 012602
305 000710' 012601
306 000712' 012600
307 000714' 106426
308 000716' 000207
309
310
311 000720' 106746
312 000722' 010146
313 000724' 010246
314 000726' 010346
315 000730' 042701 177400
316 000734' 074105
317 000736' 012702 166670
318 000742' 012703 101440
319 000746' 012701 000010
320 000752' 000241
321 000754' 006004
322 000756' 006005
323 000760' 103002
324 000762' 074204
325 000764' 074305
326 000766' 077107
327 000770' 012603
328 000772' 012602
329 000774' 012601
330 000776' 106426
331 001000' 000207
332
333

```

```

:
: *****
: ***** SUBROUTINES *****
: *****
:
BLKCRC: MFPS      -(SP)          : SAVE PSW
          MOV      R0,-(SP)       : SAVE R0
          MOV      R1,-(SP)
          MOV      R2,-(SP)
          MOV      #INITH,R4      : INITIAL CRC HIGH WORD
          MOV      #INITL,R5      : INITIAL CRC LOW WORD
10$:     MOV      (R0)+,R1        : GET NEXT BYTE
          JSR      PC,GETCRC      : CALCULATE CRC
20$:     SOB      R2,10$         : LOOP TILL DONE
          MOV      (SP)+,R2
          MOV      (SP)+,R1
          MOV      (SP)+,R0
          MTPS    (SP)+
          RTS     PC
:
:
GETCRC: MFPS      -(SP)          : SAVE REGISTERS
          MOV      R1,-(SP)
          MOV      R2,-(SP)
          MOV      R3,-(SP)
          BIC      #^C377,R1      : CLEAR HIGH BYTE
          XOR      R1,R5          : MERGE NEW BYTE WITH OLD CRC
          MOV      #POLYH,R2      : GET POLYNOMIAL HIGH WORD
          MOV      #POLYL,R3      : GET CRC POLYNOMIAL LOW WORD
          MOV      #8.,R1         : LOOP COUNT
10$:     CLC
          ROR     R4
          ROR     R5
          BCC     20$
          XOR     R2,R4
          XOR     R3,R5
          : SKIP IF BIT 0 NOT SET
20$:     SOB     R1,10$
          MOV     (SP)+,R3
          MOV     (SP)+,R2
          MOV     (SP)+,R1
          MTPS   (SP)+
          RTS    PC
:
:

```

76MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 9  
F\_MODULE MICROCODE

```

334
335
336
337
338
339
340
341 001002' 112767 000002 177620 MICF1: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
342 001010' 016737 177614 021020 MOV PCSR1,@#IPCSR1
343
344
345
346
347
348 001016' 016737 177610 021010 MOV IPCSR2,@#MDMA0 ; SET TO GET HOST PCBB ADDRESS
349 001024' 016737 177604 021012 MOV IPCSR2+2,@#MDMA1
350 001032' 013700 021014 MOV @#MDMAR0,R0 ; R5 NOW CONTAINS PCBB LOW
351 001036' 013701 021014 MOV @#MDMAR0,R1 ; R1 NOW CONTAINS PCBB HIGH
352 001042' 010037 021010 MOV R0,@#MDMA0 ; POINT AT PCBB
353 001046' 010137 021012 MOV R1,@#MDMA1
354 001052' 013703 021014 MOV @#MDMAR0,R3 ; R3 NOW HOLDS DATA PATTERN
355
356
357
358
359
360
361 001056' 012700 100000 MOV #LINADR,R0 ; RECEIVE BUFFER STARTS HERE
362 001062' 010067 177556 MOV R0,RBUF
363 001066' 005020 104000 10$: CLR (R0)+ ; FILL RECEIVE BUFFER WITH ZEROS
364 001070' 020027 104000 CMP R0,#LINADR+SIZ1K ; STOP AT THE TOP
365 001074' 103774 BLO 10$
366
367
368
369
370
371
372 001076' 010067 177544 MOV R0,TBUF ; SAVE COPY OF ADDRESS
373 001102' 010320 110000 20$: MOV R3,(R0)+ ; FILL XMIT BUFFER WITH PATTERN
374 001104' 020027 110000 CMP R0,#LINADR+SIZ2K
375 001110' 103774 BLO 20$
376
377
378
379
380
381
382 001112' 012737 000200 177776 MOV #MODE,@#CMDREG ; ENABLE LINK, SELECT MODE REG
383 001120' 012737 100004 177774 MOV #PROM!LOOP,@#MODREG ; PROM MODE AND LOOPBACK
384 001126' 012737 100000 177776 MOV #ENABLE,@#CMDREG ; ENABLE IT
385
386 001134' 016701 177506 MOV TBUF,R1 ; POINT AT XMIT BUFFER
387 001140' 005021 CLR (R1)+ ; CLEAR OUT STATUS WORD
388 001142' 012721 002752 MOV #MAXBC,(R1)+ ; CLEAR OUT THE FIFO
389 001146' 005037 021034 CLR @#CLRFIF ; CLEAR THE FIFO

```

77MICROF - MICROCODE MODULE F MACY11 30A(1052) 07-APR-83 16:51 PAGE 10  
MICROF.MAC 07-APR-83 16:06 CRC DATA PATTERN TEST

```

390 001152' 005067 177446          CLR      FLG6
391 001156' 016737 177462 021032    MOV      RBUF,R0#LFRBUF      ; TELL DEUMA RECEIVE BUF LOC
392 001164' 016737 177456 021030    MOV      TBUF,R0#LTAC       ; TELL DEUMA WHERE XMIT BUF IS
393                                     ;
394 001172' 106427 000140          MTPS     #PRI03              ; ALLOW INTERRUPTS
395 001176' 032767 000100 177420 30$:  BIT      #RCVFLG,FLG6       ; WAIT FOR RECEIVER INTERRUPT
396 001204' 001774          BEQ      30$
397                                     ;
398 001206' 032767 000040 177410 40$:  BIT      #TRNFLG,FLG6       ; WAIT FOR XMIT INTERRUPT TOO
399 001214' 001774          BEQ      40$
400                                     ;
401 001216' 106427 000340          MTPS     #PRI07              ; DISABLE INTERRUPTS
402                                     ;
403                                     ;
404                                     ; *****
405                                     ; ***** CALCULATE THE CRC ON THE RECEIVER BUFFER CONTENTS *****
406                                     ; *****
407                                     ;
408 001222' 016700 177416          MOV      RBUF,R0              ; POINT AT RECEIVE BUFFER
409 001226' 062700 000004          ADD      #4,R0                ; BY-PASS THE STATUS AND BC
410 001232' 012702 002756          MOV      #MAXBC+4,R2         ; INCLUDE CRC BYTES
411 001236' 004767 177414          JSR      PC,BLKCRCH
412 001242' 020427 157273          CMP      R4,#EXCRCH
413 001246' 001010          BNE      50$                  ; ERROR IF NOT
414 001250' 020527 020343          CMP      R5,#EXCRCL
415 001254' 001005          BNE      50$
416 001256' 000241          CLC
417 001260' 112767 000001 177343 45$:  MOV      #1,PCSR1+1          ; INDICATE SUCCESS
418 001266' 000207          RTS      PC                  ; TELL WHAT TEST WE JUST FINISHED
419                                     ;
420                                     ; *****
421                                     ; ***** ERROR EXIT *****
422                                     ; *****
423                                     ;
424                                     ;
425 001270' 000261          50$:    SEC
426 001272' 000772          BR      45$                  ; TELL HIM WE MADE ERROR
427                                     ; LEAVE

```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 11  
CRC DATA PATTERN TEST

428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483

001274' 112767 000002 177326  
001302' 016737 177322 021020  
  
001310' 016737 177316 021010  
001316' 016737 177312 021012  
001324' 013704 021014  
001330' 013705 021014  
001334' 010437 021010  
001340' 010537 021012  
001344' 013703 021014  
  
001350' 012700 100000  
001354' 010067 177264  
001360' 005020  
001362' 020027 104000  
001366' 103774  
  
001370' 010067 177252  
001374' 010320  
001376' 020027 110000  
001402' 103774  
  
001404' 016700 177236  
001410' 062700 000004  
001414' 012702 002746  
001420' 004767 177232  
001424' 016700 177216  
001430' 062700 000004  
001434' 062700 002746

```
:
:SBTTL  CRC ERROR TEST
:
:*****
:***** TELL HIM WE ARE TESTING *****
:*****
MICF2:  MOVB  #INTST,PCSR1      ; TELL HOST WE ARE TESTING
        MOV   PCSR1,B#IPCSR1
:
:*****
:***** RETRIEVE PATTERN FROM HOST MEMORY *****
:*****
        MOV   IPCSR2,B#MDMA0    ; SET TO GET HOST PCBB ADDRESS
        MOV   IPCSR2+2,B#MDMA1
        MOV   B#MDMARI,R4      ; R4 NOW CONTAINS PCBB LOW
        MOV   B#MDMARI,R5      ; R5 NOW CONTAINS PCBB HIGH
        MOV   R4,B#MDMA0      ; POINT AT PCBB
        MOV   R5,B#MDMA1
        MOV   B#MDMARI,R3      ; R3 NOW HOLDS DATA PATTERN
:
:*****
:***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
:*****
        MOV   #LINADR,R0        ; RECEIVE BUFFER STARTS HERE
        MOV   R0,RBUF
10$:    CLR   (R0)+              ; FILL RECEIVE BUFFER WITH ZEROS
        CMP   R0,#LINADR+SIZ1K  ; STOP AT THE TOP
        BLO  10$
:
:*****
:***** FILL XMIT BUFFER WITH TEST PATTERN *****
:*****
20$:    MOV   R0,RBUF           ; SAVE COPY OF ADDRESS
        MOV   R3,(R0)+         ; FILL XMIT BUFFER WITH PATTERN
        CMP   R0,#LINADR+SIZ2X
        BLO  20$
:
:*****
:***** CALCULATE CRC ON TRANSMIT BUFFER *****
:*****
        MOV   TBUF,R0          ; POINT AT RECEIVE BUFFER
        ADD   #4,R0            ; SKIP 1ST WORD AND BYTE COUNT
        MOV   #MAXBC-4,R2     ; DO A BUNCH
        JSR   PC,BLKCRC
        MOV   TBUF,R0          ; POINT AT XMIT BUFFER
        ADD   #4,R0            ; BY-PASS STATUS AND BC
        ADD   #MAXBC-4,R0     ; ADD BUFFER OFFSET
```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 50A(1052) 07-APR-83 16:51 PAGE 12  
CRC ERROR TEST

```

484 001440' 005104          COM      R4          : MUST SEND THE COMPLEMENT
485 001442' 005105          COM      R5
486 001444' 005205          INC      R5          : INTRODUCE AN ERROR IN CRC!
487 001446' 010520          MOV      R5,(R0)+    : APPEND THE CRC WORD
488 001450' 010420          MOV      R4,(R0)+    : APPEND THE OTHER CRC WORD
489
490
491
492
493
494
495 001452' 012737 000200 177776      MOV      #MODE,#CMODREG      : ENABLE, SELECT MODE
496 001460' 012737 100014 177774      MOV      #PROM!LOOP!DTCR,#MODREG : PROM MODE AND LOOPBACK
497 001466' 012737 100000 177776      MOV      #ENABLE,#CMODREG    : ENABLE THE XMITR
498
499 001474' 016701 177146          MOV      TBUF,R1          : POINT AT XMIT BUFFER
500 001500' 005021          CLR      (R1)+          : CLEAR OUT STATUS WORD
501 001502' 012721 002752          MOV      #MAXBC,(R1)+    : CLEAR OUT THE FIFO
502 001506' 005037 021034          CLR      #CLRFIF        : CLEAR THE FIFO
503 001512' 005067 177106          CLR      FLG6
504 001516' 016737 177122 021032      MOV      RBUF,#LFRBUF    : TELL DEUNA RECEIVE BUF LOC
505 001524' 016737 177116 021030      MOV      TBUF,#LTAC      : TELL DEUNA WHERE XMIT BUF IS
506
507 001532' 106427 000140          MTPS    #PRI03          : ALLOW INTERRUPTS
508 001536' 032767 000100 177060 30$ : BIT     #RCVFLG,FLG6    : WAIT FOR RECEIVER INTERRUPT
509 001544' 001774          BEQ     30$
510
511 001546' 032767 000040 177050 40$ : BIT     #TRNFLG,FLG6    : WAIT FOR XMIT INTERRUPT TOO
512 001554' 001774          BEQ     40$
513
514 001556' 106427 000340          MTPS    #PRI07          : DISABLE INTERRUPTS
515
516
517
518
519
520
521 001562' 012700 100000          MOV      #LINADR,R0      : GET RECEIVE STATUS WORD
522 001566' 011003          MOV      (R0),R3        : GET STATUS WORD
523
524 001570' 016737 177036 021010      MOV      IPCSR2,#MDMA0   : SET TO GET HOST PCBB
525 001576' 016737 177032 021012      MOV      IPCSR2+2,#MDMA1
526 001604' 013700 021014          MOV      #MDMAR0,R0     : R0 NOW CONTAINS PCBB LOW
527 001610' 013701 021014          MOV      #MDMAR0,R1     : R1 NOW CONTAINS PCBB HI
528 001614' 062700 000002          ADD      #2,R0          : BUMP TO NEXT HOST WORD
529 001620' 005501          ADC      R1
530 001622' 010037 021010          MOV      R0,#MDMA0      : POINT TO PCBB
531 001626' 010137 021012          MOV      R1,#MDMA1
532 001632' 010337 021026          MOV      R3,#MDMA0      : WRITE STATUS WORD TO HOST
533
534
535 001636' 112767 000002 176765      MOVB    #2,PCSR1+1      : TELL HIM WHICH TEST IT IS
536 001644' 000241          CLC
537 001646' 000207          RTS      PC

```



77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 13  
CRC ERROR TEST

538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593

001650' 112767 000002 176752  
001656' 016737 176746 021020  
  
001664' 016737 176742 021010  
001672' 016737 176736 021012  
001700' 013704 021014  
001704' 013705 021014  
001710' 010437 021010  
001714' 010537 021012  
001720' 013703 021014  
  
001724' 012700 100000  
001730' 010067 176710  
001734' 005020  
001736' 020027 104000  
001742' 103774  
  
001744' 010067 176676  
001750' 010320  
001752' 020027 110000  
001756' 103774  
  
001760' 016737 176646 021010  
001766' 016737 176642 021012  
001774' 013704 021014  
002000' 013705 021014  
002004' 062704 000002  
002010' 005505  
002012' 010437 021010

```
.SBTTL CRC PATTERN LENGTH TEST
*****
**** TELL HIM WE ARE TESTING ****
*****
MICF3: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
      MOV  PCSR1,&#IPCSR1
*****
**** RETRIEVE PATTERN FROM HOST MEMORY ****
*****
      MOV  IPCSR2,&#MDMA0 ; SET TO GET HOST PCBB ADDRESS
      MOV  IPCSR2+2,&#MDMA1
      MOV  &#MDMAR0,R4 ; R4 NOW CONTAINS PCBB LOW
      MOV  &#MDMAR0,R5 ; R5 NOW CONTAINS PCBB HIGH
      MOV  R4,&#MDMA0 ; POINT AT PCBB
      MOV  R5,&#MDMA1
      MOV  &#MDMAR0,R3 ; R3 NOW HOLDS DATA PATTERN
*****
**** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN ****
*****
      MOV  #LINADR,R0 ; RECEIVE BUFFER STARTS HERE
      MOV  R0,RBUF
10$: CLR  (R0)+ ; FILL RECEIVE BUFFER WITH ZEROS
      CMP  R0,#LINADR+SIZ1K ; STOP AT THE TOP
      BLO  10$
*****
**** FILL XMIT BUFFER WITH TEST PATTERN ****
*****
      MOV  R0,TBUF ; SAVE COPY OF ADDRESS
20$: MOV  R3,(R0)+ ; FILL XMIT BUFFER WITH PATTERN
      CMP  R0,#LINADR+SIZ2K
      BLO  20$
*****
**** RETRIEVE BYTE COUNT FROM HOST MEMORY ****
*****
      MOV  IPCSR2,&#MDMA0 ; SET TO GET HOST PCBB
      MOV  IPCSR2+2,&#MDMA1
      MOV  &#MDMAR0,R4 ; R4 NOW CONTAINS PCBB LO
      MOV  &#MDMAR0,R5 ; R5 NOW CONTAINS PCBB HI
      ADD  #2,R4 ; BUMP LOW BY TWO
      ADC  R5
      MOV  R4,&#MDMA0 ; POINT AT PCBB
```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 14  
CRC PATTERN LENGTH TEST

```

594 002016' 010537 021012
595 002022' 013703 021014
596
597
598
599
600
601
602 002026' 016700 176614
603 002032' 062700 000004
604 002036' 010302
605 002040' 004767 176612
606 002044' 016700 176576
607 002050' 062700 000004
608 002054' 060300
609 002056' 005104
610 002060' 005105
611 002062' 032700 000001
612 002066' 001427
613
614 002070' 162700 000001
615 002074' 011067 176550
616 002100' 110567 176545
617 002104' 000305
618 002106' 110567 176540
619 002112' 110467 176535
620 002116' 000304
621 002120' 110467 176530
622 002124' 105067 176525
623 002130' 016720 176514
624 002134' 016720 176512
625 002140' 016710 176510
626 002144' 000402
627
628 002146' 010520
629 002150' 010420
630 002152'
631
632
633
634
635
636
637 002152' 012737 000200 177776
638 002160' 012737 100014 177774
639 002166' 012737 100000 177776
640
641 002174' 016701 176446
642 002200' 005021
643 002202' 062703 000004
644 002206' 010321
645 002210' 005037 021034
646 002214' 005067 176404
647 002220' 016737 176420 021032
648 002226' 016737 176414 021030
649

```

```

MOV R5,&#MDMA1
MOV &#MDMAR0,R3 ; R3 NOW HOLDS BYTE COUNT
:
:
***** CALCULATE CRC ON TRANSMIT BUFFER *****
:
MOV TBUF,R0 ; POINT AT RECEIVE BUFFER
ADD #4,R0 ; BY-PASS STATUS AND BC
MOV R3,R2 ; R3 CONTAINS BYTE COUNT
JSR PC,BLKCRC
MOV TBUF,R0 ; POINT AT XMIT BUFFER
ADD #4,R0 ; BY-PASS STATUS AND BC
ADD R3,R0 ; ADD BUFFER BYTE COUNT
COM R4 ; COMPLEMENT BEFORE SENDING
COM R5
BIT #1,R0 ; ARE WE LOOKING AT AN ODD BOUNDARY?
BEQ 258 ; NO, GOOD IT MAKES THINGS EASIER
:
SUB #1,R0 ; POINT BACK TO EVEN BOUNDARY
MOV (R0),CRC1 ; GET DATA BYTE AND JUNK
MOVB R5,CRC1+1 ; REPLACE JUNK WITH 1ST CRC BYTE
SWAB R5 ; POSITION 2ND CRC BYTE
MOVB R5,CRC23 ; GET 2ND CRC BYTE
MOVB R4,CRC23+1 ; GET 3RD CRC BYTE
SWAB R4 ; POSITION 4TH CRC BYTE
MOVB R4,CRC4 ; GET 4TH CRC BYTE
CLAB CRC4+1 ; CLEAR JUNK FOR THE HELL OF IT
MOV CRC1,(R0)+ ; STORE DATA AND 1ST CRC BYTE IN BUFFER
MOV CRC23,(R0)+ ; STORE 2ND AND 3RD CRC BYTES IN BUFFER
MOV CRC4,(R0) ; STORE 4TH CRC BYTE IN BUFFER
BR 268 ; CARRY ON
258: MOV R5,(R0)+ ; APPEND THE CRC WORD
MOV R4,(R0)+ ; APPEND THE OTHER CRC WORD
268:
:
***** SET UP LINK FOR DATAGRAM LOOPBACK *****
:
MOV &#MODE,&#CHDREG ; ENABLE, SELECT MODE
MOV &#PROM!LOOP!DTCR,&#MODREG ; PROM MODE AND LOOPBACK
MOV &#ENABLE,&#CHDREG ; LET HER GO!
:
MOV TBUF,R1 ; POINT AT XMIT BUFFER
CLR (R1)+ ; CLEAR OUT STATUS WORD
ADD #4,R3 ; ACCOUNT FOR CRC BYTES
MOV R3,(R1)+ ; WRITE THE BYTE COUNT
CLR &#CLRIFIF ; CLEAR THE FIFO
CLR FLG6
MOV RBUF,&#LFRBUF ; TELL DEUMA RECEIVE BUF LOC
MOV TBUF,&#LTAC ; TELL DEUMA WHERE XMIT BUF IS
:

```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 15  
CRC PATTERN LENGTH TEST

```

650 002234' 106427 000140
651 002240' 032767 000100 176356 308: MTPS #PRI03 ; ALLOW INTERRUPTS
652 002246' 001774 BIT #RCVFLG.FLG6 ; WAIT FOR RECEIVER INTERRUPT
653 BEQ 308
654 002250' 032767 000040 176346 408: BIT #TRNFLG.FLG6 ; WAIT FOR XMIT INTERRUPT TOO
655 002256' 001774 BEQ 408
656 :
657 002260' 106427 000340 MTPS #PRI07 ; DISABLE INTERRUPTS
658 :
659 :
660 :
661 : ***** WRITE STATUS REGISTER TO HOST MEMORY *****
662 : *****
663 :
664 002264' 016700 176354 MOV RBUF,R0 ; GET RECEIVE STATUS WORD
665 002270' 011003 MOV (R0),R3 ; GET STATUS WORD
666 :
667 002272' 016737 176334 021010 MOV IPCSR2,B#DMA0 ; SET TO GET HOST PCBB
668 002300' 016737 176330 021012 MOV IPCSR2+2,B#DMA1
669 002306' 013700 021014 MOV B#DMA0,R0 ; R0 NOW CONTAINS PCBB LOW
670 002312' 013701 021014 MOV B#DMA0,R1 ; R1 NOW CONTAINS PCBB HI
671 002316' 062700 000004 ADD #4,R0 ; BUMP TO NEXT HOST WORD
672 002322' 005501 ADC R1
673 002324' 010037 021010 MOV R0,B#DMA0 ; POINT TO PCBB
674 002330' 010137 021012 MOV R1,B#DMA1
675 002334' 010337 021026 MOV R3,B#DMA0 ; WRITE STATUS WORD TO HOST
676 :
677 :
678 002340' 016702 176302 MOV TBUF,R2 ; GET XMIT BUFFER STATUS
679 002344' 011203 MOV (R2),R3
680 002346' 010337 021026 MOV R3,B#DMA0 ; WRITE XMIT STATUS TO HOST
681 002352' 112767 000003 176251 MOVB #3,PCSR1+1 ; TELL HIM WHICH TEST IT IS
682 002360' 000241 CLC
683 002362' 000207 RTS PC

```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 16  
CRC PATTERN LENGTH TEST

```

684
685
686
687
688
689
690
691 002364' 112767 000002 176236 MICF4: MOVB #INTST,PCSR1 ; TELL HOST WE ARE TESTING
692 002372' 016737 176232 021020 MOV PCSR1,#IPCSR1
693
694 002400' 005067 176222 ; CLR SANTIM ; CLEAR FLAG FOR TIMER
695
696
697
698
699
700
701 002404' 012700 100000 ; MOV #LINADR,RO ; RECEIVER BUFFER STARTS HERE
702 002410' 010067 176230 MOV RO,RBUF
703 002414' 005020 10$: CLR (RO)+ ; FILL BUFFER WITH ZEROS
704 002416' 020027 104000 CFP RO,#LINADR+SIZ1K ; STOP AT THE TOP
705 002422' 103774 BLO 10$
706
707
708
709
710
711
712 002424' 010067 176216 ; MOV RO,TBUF ; SAVE COPY OF THE ADDRESS
713 002430' 012720 125252 20$: MOV #125252,(RO)+
714 002434' 020027 110000 CFP RO,#LINADR+SIZ2K
715 002440' 103773 BLO 20$
716
717
718
719
720
721
722 002442' 016737 176164 021010 ; MOV IPCSR2,#INDMA0 ; SET TO GET HOST PCBB ADDRESS
723 002450' 016737 176160 021012 MOV IPCSR2+2,#INDMA1
724 002456' 013700 021014 MOV #INDMA0,RO ; R4 NOW CONTAINS PCBB LOW
725 002462' 013701 021014 MOV #INDMA0,R1 ; R5 NOW CONTAINS PCBB HIGH
726 002466' 010037 021010 MOV RO,#INDMA0 ; POINT AT PCBB
727 002472' 010137 021012 MOV R1,#INDMA1
728 002476' 013703 021014 MOV #INDMA0,R3 ; R3 NOW HOLDS BYTE COUNT
729
730
731
732
733
734
735 002502' 012737 000200 177776 ; MOV #MODE,#PCNDREG ; DISABLE XMIT CRC - TRUE BC
736 002510' 012737 100014 177774 MOV #PROM!LOOP!DTCR,#PCNDREG
737 002516' 012737 100000 177776 MOV #ENABLE,#PCNDREG
738
739 002524' 016704 176116 ; MOV TBUF,R4 ; POINT AT XMIT BUFFER

```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 17  
RECEIVE BUFFER RECOVERY - RUNT TEST

```

740 002530' 005024          CLR      (R4)+          ; CLEAR OUT STATUS WORD
741 002532' 010324          MOV      R3,(R4)+      ; WRITE PASSED BYTE COUNT
742                          ;
743 002534' 005037 021034    CLR      @*CLRIF       ; CLEAR THE FIFO
744 002540' 005067 176060    CLR      FLG6          ; CLEAR THE INTERRUPT FLAG
745 002544' 016737 176074 021032  MOV     RBUF,@*LFRBUF  ; TELL UNA WHERE RECEIVE BUF IS
746 002552' 016737 176070 021030  MOV     TBUF,@*LTAC    ; TELL UNA WHERE XMIT BUF IS
747                          ;
748 002560' 106427 000140          MTPS     #PRI03        ; ALLOW XMITTER TO INTERRUPT
749 002564' 026727 176036 000002 30$:  CMP     SANTIM,#2      ; WAIT AT MOST 2 SECONDS
750 002572' 002012          BGE     50$           ; EXIT NORMALLY IF TIMER DONE
751 002574' 032767 000100 176022  BIT     #RCVFLG,FLG6  ; WAIT FOR RECEIVER INTERRUPT
752 002602' 001770          BEQ     30$           ; IF NONE, WERE OK
753                          ;
754                          ;
755                          ; *****
756                          ; ***** ERROR FALLTHROUGH *****
757                          ; *****
758                          ;
759 002604' 032767 000040 176012 40$:  BIT     #TRNFLG,FLG6  ; WAIT FOR XMIT INT TOO!
760 002612' 001774          BEQ     40$
761                          ;
762 002614' 000261          SEC
763 002616' 000506          BR      90$           ; TELL HOST UNA SCREWED UP!
764                          ; EXIT WITHOUT WRITING PTR
765                          ;
766                          ; *****
767                          ; ***** NOW TRY LEGITIMATE BUFFER SIZE *****
768                          ; *****
769                          ;
770 002620' 016700 176022 50$:  MOV     TBUF,R0          ; GET TRANSMIT BUFFER POINTER
771 002624' 012720 052525 60$:  MOV     #52525,(R0)+
772 002630' 020027 110000    CMP     R0,#LINADR+SIZ2K
773 002634' 103773          BLO     60$
774                          ;
775                          ;
776                          ; *****
777                          ; ***** SET UP LINK FOR MINIMUM DATAGRAM MESSAGE *****
778                          ; *****
779                          ;
780 002636' 012737 000200 177776    MOV     #MODE,@*CMREG  ; ENABLE LINK, SEL MODE REG
781 002644' 012737 100004 177774    MOV     #PROM!LOOP,@*MODREG
782 002652' 012737 100000 177776    MOV     #ENABLE,@*CMREG
783                          ;
784 002660' 016704 175762          MOV     TBUF,R4        ; POINT AT XMIT BUFFER
785 002664' 005024          CLR     (R4)+          ; CLEAR STATUS WORD
786 002666' 012724 000100          MOV     #MINBC,(R4)+  ; SEND SMALL DATAGRAM
787                          ;
788 002672' 005067 175726          CLR     FLG6          ;
789 002676' 016737 175742 021032  MOV     RBUF,@*LFRBUF  ; TELL UNA WHERE RECEIVE BUF IS
790 002704' 016737 175736 021030  MOV     TBUF,@*LTAC    ; TELL UNA WHERE XMIT BUF IS
791                          ;
792 002712' 106427 000140          MTPS     #PRI03        ; ALLOW XMITR TO INTERRUPT
793 002716' 032767 000100 175700 70$:  BIT     #RCVFLG,FLG6  ; WAIT FOR RECEIVER INTERRUPT
794 002724' 001774          BEQ     70$
795                          ;

```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 18  
RECEIVE BUFFER RECOVERY - RUNT TEST

```

796 002726' 032767 000040 175670 80$: BIT #TRNFLG,FLG6 ; WAIT FOR XMIT INT TOO!
797 002734' 001774 BEQ 80$
798 :
799 002736' 106427 000340 MTPS #PRI07 ; DISABLE INTERRUPTS
800 :
801 :
802 :
803 : *****
804 : ***** WRITE PARAMETERS TO HOST PCBB *****
805 : *****
806 002742' 013703 021044 MOV @LRBUF,R3 ; GET LINK POINTER
807 :
808 002746' 016700 175672 MOV RBUF,R0 ; POINT AT RECEIVER BUFFER
809 002752' 062700 000004 ADD #4,R0
810 002756' 011004 MOV (R0),R4 ; GET PATTERN WORD
811 :
812 :
813 002760' 016737 175646 021010 MOV IPCSR2,@MDMA0 ; SET TO GET HOST PCBB
814 002766' 016737 175642 021012 MOV IPCSR2+2,@MDMA1
815 002774' 013700 021014 MOV @MDMAR0,R0 ; R0 NOW CONTAINS PCBB LO
816 003000' 013701 021014 MOV @MDMAR0,R1 ; R1 NOW CONTAINS PCBB HI
817 003004' 062700 000002 ADD #2,R0
818 003010' 005501 ADC R1
819 003012' 010037 021010 MOV R0,@MDMA0 ; POINT AT PCBB+2
820 003016' 010137 021012 MOV R1,@MDMA1
821 :
822 003022' 010337 021026 MOV R3,@MDMAW0 ; WRITE LINK POINTER
823 003026' 010437 021026 MOV R4,@MDMAW0 ; WRITE DATA BYTE
824 :
825 :
826 003032' 000241 CLC
827 003034' 112767 000004 175567 90$: MOVB #4,PCSR1+1 ; TELL HOST WE ARE DONE
828 003042' 000207 RTS PC

```

```

829      .SBTTL HALF-DUPLEX TEST
830      :
831      : *****
832      : ***** TELL HIM WE ARE TESTING *****
833      : *****
834      :
835      003044' 112767 000002 175556 MICF5: MOVB   #INTST,PCSR1      ; TELL HOST WE ARE TESTING
836      003052' 016737 175552 021020      MOV     PCSR1,#IPCSR1
837      :
838      003060' 005067 175542      CLR     SANTIM      ; CLEAR FLAG FOR TIMER
839      :
840      :
841      : *****
842      : ***** CLEAR THE RECEIVE BUFFER *****
843      : *****
844      :
845      003064' 012700 100000      MOV     #LINADR,RO      ; RECEIVER BUFFER STARTS HERE
846      003070' 010067 175550      MOV     RO,RBUF
847      003074' 005020      10$:   CLR     (RO)+      ; FILL BUFFER WITH ZEROS
848      003076' 020027 104000      CMP     RO,#LINADR+SIZ1K ; STOP AT THE TOP
849      003102' 103774      BLO    10$
850      :
851      :
852      : *****
853      : ***** FILL XMIT WITH RUNT MARKER *****
854      : *****
855      :
856      003104' 010067 175536      MOV     RO,TBUF        ; SAVE COPY OF THE ADDRESS
857      003110' 012720 125252      20$:   MOV     #125252,(RO)+
858      003114' 020027 110000      CMP     RO,#LINADR+SIZ2K
859      003120' 103773      BLO    20$
860      :
861      :
862      : *****
863      : ***** SET UP LINK FOR RUNT DATAGRAM LOOPBACK *****
864      : *****
865      :
866      003122' 012737 000200 177776      MOV     #MODE,#CMDREG  ; DISABLE XMIT CRC - TRUE BC
867      003130' 012737 100005 177774      MOV     #PROM!LOOP!HDX,#MODREG
868      003136' 012737 100000 177776      MOV     #ENABLE,#CMDREG
869      :
870      003144' 016704 175476      MOV     TBUF,R4        ; POINT AT XMIT BUFFER
871      003150' 005024      CLR     (R4)+          ; CLEAR OUT STATUS WORD
872      003152' 010324      MOV     R3,(R4)+      ; WRITE PASSED BYTE COUNT
873      :
874      003154' 005037 021034      CLR     @#CLRFIF      ; CLEAR THE FIFO
875      003160' 005067 175440      CLR     FLG6          ; CLEAR THE INTERRUPT FLAG
876      003164' 016737 175454 021032      MOV     RBUF,@#LFRBUF ; TELL UNA WHERE RECEIVE BUF IS
877      003172' 016737 175450 021030      MOV     TBUF,@#LTAC   ; TELL UNA WHERE XMIT BUF IS
878      :
879      003200' 106427 000140      MTPS   #PRI03         ; ALLOW XMITTER TO INTERRUPT
880      003204' 026727 175416 000002 30$:   CMP     SANTIM,#2      ; WAIT AT MOST 2 SECONDS
881      003212' 002012      BGE    50$            ; EXIT NORMALLY IF TIMER DONE
882      003214' 032767 000100 175402      BIT     #RCVFLG,FLG6  ; WAIT FOR RECEIVER INTERRUPT
883      003222' 001770      BEQ    30$            ; IF NONE, WERE OK
884      :

```

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 20  
HALF-DUPLEX TEST

```

885
886
887
888
889
890 003224' 032767 000040 175372 40$: BIT #TRNFLG,FLG6 ; WAIT FOR XMIT INT TOO!
891 003232' 001774 BEQ 40$
892
893 003234' 000261 SEC ; TELL HOST UNA SCREWED UP!
894 003236' 000503 BR 90$ ; EXIT WITHOUT WRITING PTR
895
896
897
898
899
900
901 003240' 016700 175402 50$: MOV TBUF,RO ; GET TRANSMIT BUFFER POINTER
902 003244' 012720 052525 60$: MOV #52525,(RO)+
903 003250' 020027 110000 CMP RO,#LINADR+SIZ2K
904 003254' 103773 BLO 60$
905
906
907
908
909
910
911 003256' 012737 000200 177776 MOV #MODE,#CMDREG ; ENABLE LINK, SEL MODE REG
912 003264' 012737 100004 177774 MOV #PROM!LOOP,#MODREG
913 003272' 012737 100000 177776 MOV #ENABLE,#CMDREG
914
915 003300' 016704 175342 MOV TBUF,R4 ; POINT AT XMIT BUFFER
916 003304' 005024 CLR (R4)+ ; CLEAR STATUS WORD
917 003306' 012724 000100 MOV #MINBC,(R4)+ ; SEND SMALL DATAGRAM
918
919 003312' 005067 175306 CLR FLG6
920 003316' 016737 175322 021032 MOV RBUF,#LFRBUF ; TELL UNA WHERE RECEIVE BUF IS
921 003324' 016737 175316 021030 MOV TBUF,#LTAC ; TELL UNA WHERE XMIT BUF IS
922
923 003332' 106427 000140 MTPS #PRI03 ; ALLOW XMITR TO INTERRUPT
924 003336' 032767 000100 175260 70$: BIT #RCVFLG,FLG6 ; WAIT FOR RECEIVER INTERRUPT
925 003344' 001774 BEQ 70$
926
927 003346' 032767 000040 175250 80$: BIT #TRNFLG,FLG6 ; WAIT FOR XMIT INT TOO!
928 003354' 001774 BEQ 80$
929
930 003356' 106427 000340 MTPS #PRI07 ; DISABLE INTERRUPTS
931
932
933
934
935
936
937 003362' 013703 021044 MOV #LRFBUF,R3 ; GET LINK POINTER
938
939 003366' 016700 175252 MOV RBUF,RO ; POINT AT RECEIVER BUFFER
940 003372' 062700 000004 ADD #4,RO

```

\*\*\*\*\*  
\*\*\*\*\* ERROR FALLTHROUGH \*\*\*\*\*  
\*\*\*\*\*

\*\*\*\*\*  
\*\*\*\*\* NOW SEE IF BUFFER RECOVERS \*\*\*\*\*  
\*\*\*\*\*

\*\*\*\*\*  
\*\*\*\*\* SET UP LINK FOR MINIMUM DATAGRAM MESSAGE \*\*\*\*\*  
\*\*\*\*\*

\*\*\*\*\*  
\*\*\*\*\* WRITE PARAMETERS TO HOST PCBB \*\*\*\*\*  
\*\*\*\*\*



77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 21  
HALF-DUPLEX TEST

```

941 003376' 011004          MOV      (R0),R4          : GET PATTERN WORD
942
943          :
944 003400' 016737 175226 021010  MOV      I'PCSR2,@#MDMA0      : SET TO GET HOST PCBB
945 003406' 016737 175222 021012  MOV      I'PCSR2+2,@#MDMA1
946 003414' 013700 021014          MOV      @#MDMAR0,R0          : R0 NOW CONTAINS PCBB LO
947 003420' 013701 021014          MOV      @#MDMAR0,R1          : R1 NOW CONTAINS PCBB HI
948 003424' 010037 021010          MOV      R0,@#MDMA0
949 003430' 010137 021012          MOV      R1,@#MDMA1
950          :
951 003434' 010337 021026          MOV      R3,@#MDMAW0
952 003440' 010437 021026          MOV      R4,@#MDMAW0      : WRITE LINK POINTER
953          : WRITE DATA BYTE
954          :
955 003444' 000241          CLC
956 003446' 112767 000005 175155 90$: MOVB    #5,PCSR1+1      : TELL HOST WE ARE DONE
957 003454' 000207          RTS      PC
958
959 003456' 003460          MICFSZ::MICFSZ-MICROF+2
960          :
961          000001          .END

```







77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 27  
CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL	10
BERROR	10
BGAUJ	10
BGAUT	10
BGNCLN	10
BGNDU	10
BGNHRD	10
BGNHW	10
BGINI	10
BGNMOD	10
BGNMSG	10
BGNPRO	10
BGNPTA	10
BGNRPT	10
BGNSEG	10
BGNSET	10
BGNSFT	10
BGNSRV	10
BGNSUB	10
BGNSU	10
BGNTST	10
BNCOMP	10
BNERRO	10
BREAK	10
BRESET	10
CKLOOP	10
CLOCK	10
CLOSE	10
CLRVEC	10
COMREN	10
DELAY	10
DESCRI	10
DEVTYP	10
DISPAT	10
DISPLA	10
DOCLN	10
DODU	10
DORPT	10
ENDAU	10
ENDAUT	10
ENDCLN	10
ENDCON	10
ENDDU	10
ENDHRD	10
ENDHW	10
ENDINI	10
ENDMOD	10
ENDMSG	10
ENDPRO	10
ENDPTA	10
ENDRPT	10
ENDSEG	10
ENDSET	10
ENDSFT	10
ENDSRV	10
ENDSUB	10

77MICROF - MICROCODE MODULE F  
MICROF.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:51 PAGE 28  
CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	10
ENDTST	10
EQUALS	10
ERRDF	10
ERRHRD	10
ERROR	10
ERRSF	10
ERRSOF	10
ERRTBL	10
ESCAPE	10
EXIT	10
FEQUAL	10
GETBYT	10
GETPRI	10
GETWOR	10
GMANIA	10
GMANID	10
GMANIL	10
GPHARD	10
GPRMA	10
GPRMD	10
GPRML	10
HEADER	10
INLOOP	10
IOSETU	10
IOSTAR	10
KT11	10
LASTAD	10
MANUAL	10
MEMORY	10
MSBYTE	10
MSCHEC	10
MSCNTO	10
MSCOUN	10
MSDATA	10
MSDEC	10
MSDEFA	10
MSENDE	10
MSERRI	10
MSESCA	10
MSESCS	10
MSXCP	10
MSEXIT	10
MSXSE	10
MSXTJ	10
MSGEN	10
MSGENB	10
MSGETS	10
MSGETT	10
MSGNCB	10
MSGIN	10
MSGMLS	10
MSGNSU	10
MSGNTA	10
MSGNTS	10
MSHAPT	10

77MICROF - MICROCODE MODULE F MACY11 30A(1052) 07-APR-83 16:51 PAGE 29  
 MICROF.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

MSHMAP 1#  
 MSINCR 1#  
 MSIOSE 1#  
 MSLDRO 1#  
 MSMASK 1#  
 MSACHI 1#  
 MSACLO 1#  
 MSMSK1 1#  
 MSPOP 1#  
 MSPRIN 1#  
 MSPUSH 1#  
 MSPUT 1#  
 MSPUT1 1#  
 MSRAI 1#  
 MSRBRO 1#  
 MSRNRO 1#  
 MSSETS 1#  
 MSSTAR 1#  
 MSVC 1#  
 MSTLAB 1#  
 MSTSTL 1#  
 MSWORD 1#  
 MSXFER 1#  
 OPEN 1#  
 POINTE 1#  
 PRINTB 1#  
 PRINTF 1#  
 PRINTS 1#  
 PRINTX 1#  
 READBU 1#  
 READEF 1#  
 RFLAGS 1#  
 SETPRI 1#  
 SETVEC 1#  
 SLASH 1#  
 STARS 1#  
 SVC 1#  
 XFER 1#  
 XFERF 1#  
 XFERT 1#

. ABS. 000000 000  
 000000 001  
 MICRF 003460 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MICROF.OBJ,MICROF.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROF.MAC  
 RUN-TIME: 2 3 .4 SECONDS  
 RUN-TIME RATIO: 40/6=6.0  
 CODE USED: 31K (61 PAGES)

68SVC.MLB SOURCE FILE MACY11 30A(1052) 07-APR-83 16:52 PAGE 2  
 MICROG.MAC 07-APR-83 16:06

```

1
2
3
4          000000'
5
6          .SBTTL REGISTER DEFINITIONS USED BY THE T11
7
8          021000 IPCSRO = 21000 ;INTERNAL PCSRO ADDRESS
9          021002 DMACSR = 21002 ;DMA ENGINE CONTROL STATUS REGISTER
10         021004 DMATO = 21004 ;DMA ENGINE TO ADDRESS REGISTER #0
11         021006 DMAT1 = 21006 ;DMA ENGINE TO ADDRESS REGISTER #1
12         021010 MDMA0 = 21010 ;MICROCPU DMA TO ADDRESS REGISTER #0
13         021012 MDMA1 = 21012 ;MICROCPU DMA TO ADDRESS REGISTER #1
14         021014 MDMAR0 = 21014 ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
15         021016 MDMAR1 = 21016 ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
16         021020 IPCSR1 = 21020 ;INTERNAL PCSR1 ADDRESS
17         021022 DMAF = 21022 ;DMA ENGINE FROM ADDRESS REGISTER
18         021024 DMAWC = 21024 ;DMA ENGINE WORD COUNT REGISTER
19         021026 MDMAW0 = 21026 ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
20         021030 LTAC = 21030 ;LINK TRANSMIT ADDRESS COUNTER REGISTER
21         021032 LFRBUF = 21032 ;LINK RECIEVE BUFFER ADDRESS FIFO
22         021034 CLRFIF = 21034 ;CLEAR FIFO
23         021036 MDMAW1 = 21036 ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
24         021040 PCSRSW = 21040 ;SWITCH PACK REGISTER
25         021042 MDMSR = 21042 ;MICROCPU DMA STATUS REGISTER
26         021044 LRFBUF = 21044 ;LINK RECIEVE BUFFER COMPLETED
27         021060 PHYAD0 = 21060 ;PHYSICAL ADDRESS ROM BYTE 0
28         021062 PHYAD1 = 21062 ;PHYSICAL ADDRESS ROM BYTE 1
29         021064 PHYAD2 = 21064 ;PHYSICAL ADDRESS ROM BYTE 2
30         021066 PHYAD3 = 21066 ;PHYSICAL ADDRESS ROM BYTE 3
31         021070 PHYAD4 = 21070 ;PHYSICAL ADDRESS ROM BYTE 4
32         021072 PHYAD5 = 21072 ;PHYSICAL ADDRESS ROM BYTE 5
33         177774 MDREG = 177774 ;LINK MODE REGISTER
34         177774 ADDRREG = 177774 ;LINK STATION ADDRESS RAM REGISTER
35         177776 CMDREG = 177776 ;LINK COMMAND REGISTER
36
37          .SBTTL OTHER DEFINITIONS USED BY THE MICROCODE
38
39         100000 BIT15 = 100000
40         040000 BIT14 = 40000
41         020000 BIT13 = 20000
42         010000 BIT12 = 10000
43         004000 BIT11 = 4000
44         002000 BIT10 = 2000
45         001000 BIT9 = 1000
46         000400 BIT8 = 400
47         000200 BIT7 = 200
48         000100 BIT6 = 100
49         000040 BIT5 = 40
50         000020 BIT4 = 20
51         000010 BIT3 = 10
52         000004 BIT2 = 4
53         000002 BIT1 = 2
54         000001 BIT0 = 1
55
56         012400 LASFTP = BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN

```



76MICROG - MICROCODE MODULE G  
MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 3  
OTHER DEFINITIONS USED BY THE MICROCODE

57	000340	PRI07 =	340	
58	000300	PRI06 =	300	
59	000240	PRI05 =	240	
60	000200	PRI04 =	200	
61	000140	PRI03 =	140	
62	000100	PRI02 =	100	
63	000040	PRI01 =	40	
64	000000	PRI00 =	0	
65		:		
66		:PCSR0 - PORT CONTROL STATUS REGISTER 0		
67		:		
68	100000	SERI =	BIT15	
69	040000	PCEI =	BIT14	
70	020000	RXI =	BIT13	
71	010000	TXI =	BIT12	
72	004000	DNI =	BIT11	
73	002000	RCEI =	BIT10	
74	000400	FATI =	BIT8	
75		:		
76		:LINK COMMAND REGISTER		
77		:		
78	100000	ENABLE =	BIT15	:ENABLE LINK MODULE
79	000200	MODE =	BIT7	:ENABLE MODE REGISTER
80	000100	ARAM =	BIT6	:ENABLE STATION ADDRESS RAM
81		:		
82		:LINK MODE REGISTER		
83		:		
84	100000	PROM =	BIT15	:PROMISCUIOUS MODE
85	040000	EMAL =	BIT14	:ENABLE MULTICAST
86	004000	ENCR =	BIT11	:ENABLE COLLISION TEST
87	002000	ACLO =	BIT10	:ENABLE ACLO
88	000040	DRTY =	BIT5	:DISABLE RETRY LOGIC
89	000020	COLL =	BIT4	:SIMULATE A COLLISION
90	000010	DTCR =	BIT3	:DISABLE TRANSMIT CRC LOGIC
91	000004	LOOP =	BIT2	:ENABLE LOOPBACK
92		:		
93	000070	TRNVEC=	70	:VECTOR ADDRESS FOR THE TRANSMITTER
94	000120	RCVVEC=	120	:VECTOR ADDRESS FOR THE RECEIVER
95	000134	SANVEC=	134	:VECTOR ADDRESS FOR THE SANITY TIMER
96	000064	CSRVEC=	64	:VECTOR ADDRESS FOR CSR WRITE INTERRUPT
97	000114	DMAVEC=	114	:VECTOR ADDRESS FOR DMA DONE INTERRUPT
98	000140	PARVEC=	140	:VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
99	001000	STACK=	1000	:STACK LOCATION
100	000001	INMON=	1	:IN MICROMONITOR STATE
101	000002	INTST=	2	:IN A TEST STATE
102	000003	INERR=	3	:IN ERROR STATE
103	000001	CSRFLG=	BIT0	:CSR WRITE INTERRUPT OCCURED
104	000002	ERRFLG=	BIT1	:UNEXPECTED ERROR OCCURED
105	000004	PARFLG=	BIT2	:PARITY ERROR OCCURED
106	000010	NXMFLG=	BIT3	:NON-EXISTANT MEMORY ERROR OCCURED
107	000020	NPRFLG=	BIT4	:NPR TIMEOUT OCCURED
108	000040	TRNFLG=	BIT5	:TRANSMITTER INTERRUPT OCCURED
109	000100	RCVFLG=	BIT6	:RECEIVER INTERRUPT OCCURED
110	100000	NPRERR=	BIT15	:PCSR0 FLAG INDICATING NPR ERROR OCCURED
111	040000	NXMERR=	BIT14	:PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURED
112	020000	UNIERR=	BIT13	:PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURED

76MICROG - MICROCODE MODULE 6  
MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 4  
OTHER DEFINITIONS USED BY THE MICROCODE

113	010000	PARERR= BIT12	:PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
114	004000	SIZ1K= 4000	:1K WORDS
115	010000	SIZ2K= SIZ1K*2	:2K WORDS
116	014000	SIZ3K= SIZ1K*3	:3K WORDS
117	020000	SIZ4K= SIZ2K*2	:4K WORDS
118	040000	SIZ8K= SIZ4K*2	:8K WORDS
119	020000	WCSSIZ= SIZ4K	:SIZE OF WRITEABLE CONTROL STORE
120	020000	IOSIZ= SIZ4K	:SIZE OF I/O PAGE
121	040000	ROMSIZ= SIZ8K	:SIZE OF ROM
122	077774	LINSIZ= SIZ8K*2-4	:SIZE OF LINK MEMORY
123	000000	WCSADR= 0	:BASE ADDRESS OF WCS
124	020000	IOADR= WCSADR+WCSSIZ	:BASE ADDRESS OF I/O PAGE
125	040000	ROMADR= IOADR+IOSIZ	:BASE ADDRESS OF ROM
126	100000	LINADR= ROMADR+ROMSIZ	:BASE ADDRESS OF LINK MEMORY
127	002756	MAXBC= 1518.	:MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
128	000000	DATERR= 0	:FLAG INDICATING DATA ERROR OCCURRED
129	000004	CRCsiz= 4	:NUMBER OF BYTES IN A CRC
130	000001	ADRERR= 1	:FLAG INDICATING ADDRESS ERROR OCCURRED
131			

76MICROG - MICROCODE MODULE G  
MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 5  
OTHER DEFINITIONS USED BY THE MICROCODE

```

132
133
134
135 000000' 106427 000340          MICROG::MTPS      #PRI07           ;DISABLE INTERRUPTS
136 000004' 012706 001000          MOV              #STACK,SP          ;SETUP STACK
137 000010' 112767 000001 000600  MOV              #INMON,PCSR1       ;TELL HOST WE ARE IN MICROMONITOR
138 000016' 016737 000574 021020  MOV              PCSR1,#IPCSR1
139 000024' 012737 004000 021000  MOV              #DNI,#IPCSRO       ;TELL HOST THE LOAD AND START FINISHED
140 000032' 010700          MOV              PC,RO              ;GET ADDRESS OF UNEXPECTED ERROR...
141 000034' 062700 000404          ADD              #ERRINT-.,RO       ;HANDLER
142 000040' 005001          CLR              R1                 ;FILL ALL UNUSED VECTORS WITH TRAP...
143 000042' 010021          10$: MOV           RO,(R1)+            ;HANDLER
144 000044' 012721 000340          MOV              #PRI07,(R1)+
145 000050' 020127 001000          CMP              R1,#1000
146 000054' 002772          BLT              10$
147
148 000056' 010700          MOV              PC,RO              ;SETUP PARITY TRAP VECTOR
149 000060' 062700 000462          ADD              #PARINT-.,RO
150 000064' 010037 000140          MOV              RO,#PARVEC
151 000070' 012737 000340 000142  MOV              #PRI07,#PARVEC+2
152
153 000076' 010700          MOV              PC,RO              ;SETUP DMA INTERRUPT VECTOR
154 000100' 062700 000364          ADD              #DMAINT-.,RO
155 000104' 010037 000114          MOV              RO,#DMAVEC
156 000110' 012737 000340 000116  MOV              #PRI07,#DMAVEC+2
157
158 000116' 010700          MOV              PC,RO              ;SETUP CSR WRITE VECTOR
159 000120' 062700 000310          ADD              #CSRWRT-.,RO
160 000124' 010037 000064          MOV              RO,#CSRVEC
161 000130' 012737 000200 000066  MOV              #PRI04,#CSRVEC+2
162
163 000136' 010700          MOV              PC,RO              ;SETUP SANITY TIMER VECTOR
164 000140' 062700 000316          ADD              #TIMINT-.,RO
165 000144' 010037 000134          MOV              RO,#SANVEC
166 000150' 012737 000240 000136  MOV              #PRI05,#SANVEC+2
167
168 000156' 010700          MOV              PC,RO              ;SETUP TRANSMITTER VECTOR
169 000160' 062700 000414          ADD              #TRNINT-.,RO
170 000164' 010037 000070          MOV              RO,#TRNVEC
171 000170' 012737 000200 000072  MOV              #PRI04,#TRNVEC+2
172
173 000176' 010700          MOV              PC,RO              ;SETUP RECEIVER VECTOR
174 000200' 062700 000360          ADD              #RCVINT-.,RO
175 000204' 010037 000120          MOV              RO,#RCVVEC
176 000210' 012737 000240 000122  MOV              #PRI05,#RCVVEC+2
177
178 000216' 013700 021040          MOV              #PCSR5,RO          ;GET SWITCH PACK BITS
179 000222' 052700 176000          BIS              #176000,RO         ;MAP THEM INTO HOST I/O PAGE
180 000226' 006300          ASL              RO                ;SHIFT OVER TO POSITION CORRECTLY
181 000230' 006300          ASL              RO
182 000232' 006300          ASL              RO
183 000234' 062700 000004          ADD              #4,RO              ;PCSR2 IS PCSRO+4
184 000240' 010067 000354          MOV              RO,IPCSR2          ;SAVE PCSR2 ADDRESS
185 000244' 012767 000003 000350  MOV              #3,IPCSR2+2        ;HIGH ORDER BITS 17:16
186 000252' 005067 000334          CLR              FLG7              ;INITIALIZE FLAG WORD
187 000256' 106427 000000          15$: MTPS      #PRI00              ;ALLOW INTERRUPTS

```

76MICROG - MICROCODE MODULE G MACY11 30A(1052) 07-APR-83 16:52 PAGE 6  
 MICROG.MAC 07-APR-83 16:06 G\_MODULE MICROCODE

188										
189	000262'	005767	000324		20\$:	TST	FLG7			:WAIT FOR A COMMAND FROM HOST
190	000266'	001775				BEQ	20\$			
191										
192	000270'	106427	000340			MTPS	#PRI07			:RAISE CPU PRIORITY TO SERVICE COMMAND
193	000274'	032767	000001	000310		BIT	#CSRFLG,FLG7			:DID HOST GIVE US A COMMAND?
194	000302'	001001				BNE	30\$			:YES
195	000304'	000777				BR	.			:NO, ERROR SO JUST SIT HERE...
196										:FOR LACK OF ANYTHING BETTER TO DO
197										
198	000306'	113700	021000		30\$:	MOVB	#IPCSRO,R0			:GET WHAT HOST WROTE TO PCSRO
199	000312'	042700	177760			BIC	#177760,R0			:STRIP ALL BUT COMMAND BITS
200	000316'	001004				BNE	35\$			:WAS IT THE CLEAR FUNCTION?
201	000320'	012737	000001	021020		MOV	#INMON,#IPCSR1			:YES, CLEAR OUT THE TEST # BITS
202	000326'	000432				BR	50\$			
203	000330'	022706	000017		35\$:	CMP	#17,R0			:RESTART OPERATIONAL MICROCODE?
204	000334'	001432				BEQ	60\$			:YES
205	000336'	162700	000001			SUB	#1,R0			
206	000342'	010701				MOV	PC,R1			:GET ADDRESS OF OUR COMMAND TABLE
207	000344'	062701	000240			ADD	#TBLG-.,R1			
208	000350'	006300				ASL	R0			:MAKE COMMAND A BYTE OFFSET
209	000352'	060001				ADD	R0,R1			:USE IT TO INDEX INTO COMMAND TABLE
210	000354'	061101				ADD	(R1),R1			:R1 NOW HAS COMMAND ROUTINE ADDRESS
211	000356'	004711				JSR	PC,(R1)			:EXECUTE AS COMMANDED FROM HOST
212	000360'	103404				BCS	40\$			:ERROR OCCURRED
213	000362'	112767	000001	000226		MOVB	#INMON,PCSR1			:INDICATE TO HOST WE ARE BACK IN...
214	000370'	000403				BR	45\$			:MICROMONITR
215	000372'	112767	000003	000216	40\$:	MOVB	#INERR,PCSR1			:INDICATE TO HOST ERROR OCCURRED
216	000400'	016737	000212	021020	45\$:	MOV	PCSR1,#IPCSR1			
217	000406'	012737	004000	021000		MOV	#ONI,#IPCSRO			:TELL HOST THIS MICROTTEST FINISHED
218	000414'	005067	000172		50\$:	CLR	FLG7			:RESET FLAG WORD
219	000420'	000716				BR	15\$			:GO WAIT FOR ANOTHER COMMAND
220										
221	000422'	005000			60\$:	CLR	R0			:FAKE SUCCESSFUL SELF TEST RESULTS
222	000424'	000137	040006			JMP	#40006			:START OPERATIONAL MICROCODE
223										
224	000430'	052767	000001	000154	CSRWRT:	BIS	#CSRFLG,FLG7			:INDICATE A CSR WRITE INTERRUPT OCCURED
225	000436'	000002				RTI				
226										
227	000440'	052767	000002	000144	ERRINT:	BIS	#ERRFLG,FLG7			:INDICATE A UNEXPECTED INTERRUPT OCCURED
228	000446'	012737	020000	021000		MOV	#UNIERR,#IPCSRO			:TELL HOST AN UNEXPECTED INTERRUPT
229										:HAPPENED
230	000454'	000777				BR	.			:JUST SIT HERE AND SPIN WHEELS
231										:COUNT ON THE HOST TO TIMEOUT WAITING
232										
233	000456'	005267	000132		TIMINT:	INC	SANTIM			:COUNT TICKS AS THEY OCCUR
234	000462'	000002				RTI				
235										
236										
237	000464'	013767	021002	000136	DMAINT:	MOV	#DMACSR,DMDONE			:GET DMA STATUS
238	000472'	032767	040000	000130		BIT	#BIT14,DMDONE			:DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
239	000500'	001404				BEQ	10\$			:NO
240	000502'	012737	040000	021000		MOV	#NXMERR,#IPCSRO			:YES, TELL HOST A NON-EXISTANT MEMORY
241										:LOCATION WAS ADDRESSED
242	000510'	000407				BR	20\$			
243	000512'	032767	100000	000110	10\$:	BIT	#BIT15,DMDONE			:DID A NPR TIMEOUT OCCUR?

76MICROG - MICROCODE MODULE G MACY11 30A(1052) 07-APR-83 16:52 PAGE 7  
 MICROG.MAC 07-APR-83 16:06 G\_MODULE MICROCODE

244	000520'	001407			BEQ	30\$		:NO
245	000522'	012737	100000	021000	MOV	#NPRERR,@#IPCSRO		:TELL HOST NPR TIMEOUT HAPPENED
246	000530'	012737	100000	021002	20\$: MOV	#BIT15,@#DMACSR		:CLEAR THE INTERRUPT IN THE DMA ENGINE
247	000536'	000777			BR	.		:SIT HERE AND SPIN WHEELS
248	000540'	000002			30\$: RTI			
249								
250	000542'	052767	000004	000042	PARINT: BIS	#PARFLG,FLG7		:SET PARITY ERROR OCCURRED
251	000550'	012737	010000	021000	MOV	#PARERR,@#IPCSRO		:TELL HOST A LINK MEMORY PARITY ERROR
252								:OCCURRED
253	000556'	000777			BR	.		:JUST SIT HERE AND SPIN WHEELS
254								:COUNT ON HOST TO TIMEOUT
255								
256	000560'	005737	021044		RCVINT: TST	@#LRBUF		:READ BUFFER DONE REGISTER...
257								:WHICH CLEARS THE INTERRUPT
258	000564'	052767	000100	000020	BIS	#RCVFLG,FLG7		:SET RECEIVER INTERRUPT OCCURRED
259	000572'	000002			RTI			
260								
261	000574'	052767	000040	000010	TRNINT: BIS	#TRNFLG,FLG7		:SET TRANSMITTER INTERRUPT OCCURRED
262	000602'	000002			RTI			
263								
264	000604'	000036			TBLG: .WORD	MICG1-		:COLLISION TEST
265	000606'	000306			.WORD	MICG2-		:TDR COUNTER TEST
266	000610'	000556			.WORD	MICG3-		:RETRY TEST
267								
268	000612'	000000			FLG7: .WORD	0		:FLAG WORD
269	000614'	000000			SANTIM: .WORD	0		:COUNT FOR SANITY TIMER
270	000616'	000000			PCSR1: .WORD	0		:COPY OF WHAT GOES TO PCSR1
271	000620'	000000	000000		IPCSR2: .WORD	0,0		:ADDRESS IN HOST MEMORY FOR PCSR2
272	000624'	000000	000000		PCBADR: .WORD	0,0		:ADDRESS IN HOST MEMORY FOR PCB
273	000630'	000000			DMDONE: .WORD	0		
274	000632'	000000			RBUF: .WORD	0		:POINTER TO RECIEVE BUFFER
275	000634'	000000			TBUF: .WORD	0		:POINTER TO XMIT BUFFER
276	000636'	000000			DBUF: .WORD	0		:POINTER TO DMA ENGINE BUFFER
277	000640'	000000			MBUF: .WORD	0		:POINTER TO MICROCPU DMA BUFFER
278								

76MICROG - MICROCODE MODULE G  
MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 8  
G\_MODULE MICROCODE

```

279 000642' 112767 000002 177746 MICG1: MOV# #INTST,PCSR1      ;TELL HOST WE ARE IN A TEST
280 000650' 016737 177742 021020  MOV      PCSR1,#IPCSR1
281                                     MOV      #LINADR,R0      ;FILL RECEIVE BUFFER WITH ZEROS
282 000656' 012700 100000                                     R0,#RBUF
283 000662' 010067 177744                                     CLR      (R0)+
284 000666' 005020 108:                                     R0,#LINADR+SIZ1K
285 000670' 020027 104000                                     CMP      108
286 000674' 103774                                     BLO
287                                     MOV      R0,TBUF      ;FILL TRANSMIT BUFFER WITH 1'S
288 000676' 010067 177732 208:                                     #177777,(R0)+
289 000702' 012720 177777                                     CMP      R0,#LINADR+SIZ2K
290 000706' 020027 110000                                     BLO      208
291 000712' 103773
292
293 000714' 016737 177700 021010  MOV      IPCSR2,#MDMA0      ;GET HOST'S PCBB ADDRESS
294 000722' 016737 177674 021012  MOV      IPCSR2+2,#MDMA1
295 000730' 013704 021014  MOV      #MDMAR0,R4      ;R4 HOLDS PCBB LOW ADDRESS
296 000734' 013705 021014  MOV      #MDMAR0,R5      ;R5 HOLDS PCBB HIGH ADDRESS
297 000740' 010437 021010  MOV      R4,#MDMA0      ;POINT TO PCBB+0
298 000744' 010537 021010  MOV      R5,#MDMA1
299 000750' 013703 021014  MOV      #MDMAR0,R3      ;GET WHAT HOST WANTS TO GO INTO LINK
300                                     ;MODE REGISTER
301
302 000754' 012737 100200 177776  MOV      #MODE!ENABLE,#CMDREG ;ENABLE LINK MODULE AND SELECT MODE REG
303 000762' 010337 177774  MOV      R3,#MODREG      ;LOAD MODE REGISTER WITH HOST'S VALUE
304
305 000766' 016702 177642  MOV      TBUF,R2      ;GET BEGINNING OF TRANSMIT BUFFER
306 000772' 005722  TST      (R2)+      ;SKIP FIRST WORD
307 000774' 012722 002752  MOV      #MAXBC-CRCSIZ,(R2)+ ;SET TO TRANSMIT MAX LENGTH PACKET
308 001000' 005037 021034  CLR      #CLRFLIF      ;CLEAR THE FIFO
309 001004' 005067 177602  CLR      FLG7      ;CLEAR OUT THE FLAG WORD
310 001010' 016737 177616 021032  MOV      RBUF,#LFRBUF      ;TELL UNA WHERE RECEIVE BUFF IS
311 001016' 016737 177612 021030  MOV      TBUF,#LTAC      ;TELL UNA WHERE TRANSMIT BUFF IS
312                                     ;START TRANSMIT OPERATION
313
314 001024' 106427 000140 177554 308:  MTPS     #PRI03      ;ALLOW INTERRUPTS
315 001030' 032767 000040  BIT      #TRNFLG,FLG7 ;WAIT FOR TRANSMIT DONE
316 001036' 001774  BEQ      308
317
318 001040' 106427 000340  MTPS     #PRI07      ;DISABLE ANY MORE INTERRUPTS
319
320 001044' 016700 177564  MOV      TBUF,R0      ;POINT TO TRANSMIT BUFFER
321 001050' 012001  MOV      (R0)+,R1      ;GET FIRST TRANSMIT STATUS WORD
322 001052' 012002  MOV      (R0)+,R2      ;GET SECOND TRANSMIT STATUS WORD
323
324 001054' 062704 000002  ADD      #2,R4      ;POINT TO HOSTS PCBB+2
325 001060' 005505  ADC      R5
326 001062' 010437 021010  MOV      R4,#MDMA0
327 001066' 010537 021012  MOV      R5,#MDMA1
328 001072' 010137 021026  MOV      R1,#MDMAW0      ;DUMP FIRST STATUS WORD TO PCBB+2
329 001076' 010237 021026  MOV      R2,#MDMAW0      ;DUMP SECOND STATUS WORD TO PCBB+4
330
331 001102' 112767 000001 177507  MOV#B    #1,PCSR1+1      ;TELL HOST WHAT TEST JUST FINISHED
332 001110' 000241  CLC
333 001112' 000207  RTS      PC
334

```

76MICROG - MICROCODE MODULE G  
MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 9  
G\_MODULE MICROCODE

```

335 001114' 112767 000002 177474 MICG2: MOVB #INTST,PCSR1 ;TELL HOST THAT WE ARE IN A TEST
336 001122' 016737 177470 021020 MOV PCSR1,@#IPCSR1
337
338 001130' 012700 100000 MOV #1,INADR,R0 ;FILL RECEIVE BUFFER WITH ALL 1'S
339 001134' 010067 177472 RO,#RBUF
340 001140' 012720 177777 10$: MOV #177777,(R0)+
341 001144' 020027 104000 CMP RO,#LINADR+SIZ1K
342 001150' 103773 BLO 10$
343
344 001152' 010067 177456 20$: MOV RO,TBUF ;FILL TRANSMIT BUFFER WITH ALL 0'S
345 001156' 005020 CLR (R0)+
346 001160' 020027 110000 CMP RO,#LINADR+SIZ2K
347 001164' 103774 BLO 20$
348
349 001166' 016737 177426 021010 MOV IPCSR2,@#MDMA0 ;GET HOST'S PCBB ADDRESS
350 001174' 016737 177422 021012 MOV IPCSR2+2,@#MDMA1
351 001202' 013700 021014 @#MDMAR,R0 ;R0 HOLDS HOST'S PCBB LOW ADDRESS
352 001206' 013701 021014 @#MDMAR,R1 ;R1 HOLDS HOST'S PCBB HI ADDRESS
353 001212' 010037 021010 RO,@#MDMA0 ;POINT TO PCBB+0
354 001216' 010137 021012 R1,@#MDMA1
355 001222' 013703 021014 MOV @#MDMAR,R3 ;GET BYTE COUNT FOR TRANSMIT OPERATION
356
357 001226' 012737 100200 177776 MOV #MODE!ENABLE,@#CMDREG ;ENABLE LINK MODULE AND SELECT MODE REG
358 001234' 012737 100004 177774 MOV #PROM!LOOP,@#MODREG ;ACCEPT ALL PACKETS IN LOOP BACK
359
360 001242' 016704 177366 MOV TBUF,R4 ;POINT TO TRANSMIT BUFFER
361 001246' 005724 TST (R4)+ ;SKIP FIRST WORD
362 001250' 010324 MOV R3,(R4)+ ;SET BYTE COUNT TO HOST'S VALUE
363 001252' 005037 021034 CLR @#CLRIF ;CLEAR THE FIFO
364 001256' 005067 177330 CLR FLG7 ;CLEAR THE FLAG WORD
365 001262' 016737 177344 021032 MOV RBUF,@#LFRBUF ;TELL UNA WHERE RECEIVE BUFF IS
366 001270' 016737 177340 021030 MOV TBUF,@#LTAC ;TELL UNA WHERE TRANSMIT BUFF IS AND
367 ;START THE TRANSMIT OPERATION
368
369 001276' 106427 000140 177302 30$: MTPS #PRI03 ;ALLOW INTERRUPTS
370 001302' 032767 000040 BIT #TRNFLG,FLG7 ;WAIT FOR TRANSMIT DONE INTERRUPT
371 001310' 001774 BEQ 30$
372
373 001312' 106427 000340 MTPS #PRI07 ;DISABLE INTERRUPTS
374
375 001316' 016702 177312 MOV TBUF,R2 ;POINT TO TRANSMIT BUFFER
376 001322' 005722 TST (R2)+ ;SKIP FIRST STATUS WORD
377 001324' 012203 MOV (R2)+,R3 ;GET SECOND STATUS WORD
378 001326' 042703 176000 BIC #176000,R3 ;STRIP ALL BUT TDR VALUE BITS
379 001332' 062700 000002 ADD #2,R0 ;POINT TO HOST'S PCBB+2
380 001336' 005501 ADC R1
381 001340' 010037 021010 RO,@#MDMA0
382 001344' 010137 021012 R1,@#MDMA1
383 001350' 010337 021026 MOV R3,@#MDMA0 ;DUMP TDR COUNTER VALUE TO PCBB+2
384
385 001354' 112767 000002 177235 MOVB #2,PCSR1+1 ;TELL HOST WHAT TEST JUST FINISHED
386 001362' 000241 CLC
387 001364' 000207 RTS PC
388
389 001366' 112767 000002 177222 MICG3: MOVB #INTST,PCSR1 ;TELL HOST THAT WE ARE IN A TEST
390 001374' 016737 177216 021020 MOV PCSR1,@#IPCSR1

```

```

391
392 001402' 012700 100000      MOV      #LINADR,R0      :FILL RECEIVE BUFFER WITH ALL 1'S
393 001406' 010067 177220      MOV      R0,RBUF
394 001412' 012720 177777      10$:    MOV      #177777,(R0)+
395 001416' 020027 104000      CMP      R0,#LINADR+SIZ1K
396 001422' 103773      BLO
397
398 001424' 010067 177204      20$:    MOV      R0,TBUF
399 001430' 005020      CLR      (R0)+          :FILL TRANSMIT BUFFER WITH ALL 0'S
400 001432' 020027 110000      CMP      R0,#LINADR+SIZ2K
401 001436' 103774      BLO
402
403 001440' 016737 177154 021010      MOV      IPCSR2,&#MDMA0      :GET HOST'S PCBB ADDRESS
404 001446' 016737 177150 021012      MOV      IPCSR2+2,&#MDMA1
405 001454' 013700 021014      MOV      &#MDMAR0,R0      :R0 HOLDS HOST'S PCBB LOW ADDRESS
406 001460' 013701 021014      MOV      &#MDMAR0,R1      :R1 HOLDS HOST'S PCBB HI ADDRESS
407 001464' 010037 021010      MOV      R0,&#MDMA0
408 001470' 010137 021012      MOV      R1,&#MDMA1
409 001474' 013703 021014      MOV      &#MDMAR0,R3      :GET BYTE COUNT FOR TRANSMIT OPERATION
410
411 001500' 012737 100200 177776      MOV      #MODE!ENABLE,&#CMDREG :ENABLE LINK MODULE AND SELECT MODE REG
412 001506' 012737 100024 177774      MOV      #PROM!LOOP!COLL,&#MODREG :ACCEPT ALL PACKETS IN LOOP BACK AND
413                                          :FORCE A COLLISION
414
415 001514' 016704 177114      MOV      TBUF,R4          :POINT TO TRANSMIT BUFFER
416 001520' 005724      TST      (R4)+          :SKIP FIRST WORD
417 001522' 010324      MOV      R3,(R4)+        :SET BYTE COUNT TO HOST'S VALUE
418 001524' 005037 021034      CLR      &#CLRIFIF      :CLEAR THE FIFO
419 001530' 005067 177056      CLR      FLG7           :CLEAR THE FLAG WORD
420 001534' 016737 177072 021032      MOV      RBUF,&#LFRBUF    :TELL UNA WHERE RECEIVE BUFF IS
421 001542' 016737 177066 021030      MOV      TBUF,&#LTAC      :TELL UNA WHERE TRANSMIT BUFF IS AND
422                                          :START THE TRANSMIT OPERATION
423
424 001550' 005005      CLR      R5             :CLEAR COUNTER
425 001552' 106427 000140      MTPS     #PRI03        :ALLOW INTERRUPTS
426 001556' 032767 000040 177026 30$:    BIT      #TRNFLG,FLG7   :IS TRANSMITTER DONE?
427 001564' 001002      BNE      35$           :YES
428 001566' 005205      INC      R5             :NO, COUNT TIME
429 001570' 000772      BR       30$
430 001572' 106427 000340      35$:    MTPS     #PRI07        :DISABLE INTERRUPTS
431
432 001576' 062700 000002      ADD      #2,R0          :POINT TO HOST'S PCBB+2
433 001602' 005501      ADC      R1
434 001604' 010037 021010      MOV      R0,&#MDMA0
435 001610' 010137 021012      MOV      R1,&#MDMA1
436 001614' 010537 021026      MOV      R5,&#MDMA0
437                                          :DUMP COUNTER VALUE TO PCBB+2
438 001620' 112767 000003 176771      MOVB     #3,PCSR1+1    :TELL HOST WHAT TEST JUST FINISHED
439 001626' 000241      CLC
440 001630' 000207      RTS      PC
441
442 001632' 001634      MICGSZ: :MICGSZ-MICROG+2
443 000001      .END

```









77MICROG - MICROCODE MODULE 6  
MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 16  
CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL	1#
BERROR	1#
BGAUJ	1#
BGAUT	1#
BGNCLN	1#
BGNDU	1#
BGNHRD	1#
BGNHW	1#
BGINI	1#
BGNROD	1#
BGNPSG	1#
BGNPRO	1#
BGNPTA	1#
BGNRPT	1#
BGNSEG	1#
BGNSET	1#
BGNSFT	1#
BGNSRV	1#
BGNSUB	1#
BGNSW	1#
BGNTST	1#
BNCORP	1#
BNERRO	1#
BREAK	1#
BRESET	1#
CKLOOP	1#
CLOCK	1#
CLOSE	1#
CLVEC	1#
COMEN	1#
DELAY	1#
DESCRI	1#
DEVTYP	1#
DISPAT	1#
DISPLA	1#
DOCLN	1#
DODU	1#
DORPT	1#
ENDAU	1#
ENDAUT	1#
ENDCLN	1#
ENDCOM	1#
ENDDU	1#
ENDHRD	1#
ENDHW	1#
ENDINI	1#
ENDROD	1#
ENDPSG	1#
ENDPRO	1#
ENDPTA	1#
ENDRPT	1#
ENDSEG	1#
ENDSET	1#
ENDSFT	1#
ENDSRV	1#
ENDSUB	1#

77MICROG - MICROCODE MODULE G MACY11 30A(1052) 07-APR-83 16:52 PAGE 17  
 MICROG.MAC 07-APR-83 16:06 CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	10
ENDTST	10
EQUALS	10
ERRDF	10
ERRHRD	10
ERROR	10
ERRSF	10
ERRSOF	10
ERRTDL	10
ESCAPE	10
EXIT	10
FEQUAL	10
GETBYT	10
GETPRI	10
GETWOR	10
GMANIA	10
GMANID	10
GMANIL	10
GPHARD	10
GPRMA	10
GPRND	10
GPRNL	10
HEADER	10
INLOOP	10
IOSETU	10
IOSTAR	10
KT11	10
LASTAD	10
MANUAL	10
MEMORY	10
MSBYTE	10
MSCHEC	10
MSCNTO	10
MSCOLN	10
MSDATA	10
MSDECR	10
MSDEFA	10
MSDEME	10
MSERRI	10
MSESCA	10
MSESCS	10
MSXCP	10
MSXIT	10
MSXSE	10
MSXIJ	10
MSGEN	10
MSGENB	10
MSGETS	10
MSGETT	10
MSGNDB	10
MSGNIN	10
MSGNLS	10
MSGNSU	10
MSGNTA	10
MSGNTS	10
MSMPT	10

77MICROG - MICROCODL MODULE G  
 MICROG.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:52 PAGE 18  
 CROSS REFERENCE TABLE -- MACRO NAMES

MSHMAP 1#  
 MSIMCR 1#  
 MSIOSE 1#  
 MSLDRO 1#  
 MSMASK 1#  
 MSPCHI 1#  
 MSACLO 1#  
 MSRSK1 1#  
 MSPOP 1#  
 MSPRIN 1#  
 MSPUSH 1#  
 MSPUT 1#  
 MSPUT1 1#  
 MSRAJ 1#  
 MSRBRO 1#  
 MSRNRO 1#  
 MSSETS 1#  
 MSSTAR 1#  
 MSSVC 1#  
 MSTLAB 1#  
 MSTSTL 1#  
 MSWORD 1#  
 MSXFER 1#  
 OPEN 1#  
 POINTE 1#  
 PRINTB 1#  
 PRINTF 1#  
 PRINTS 1#  
 PRINTX 1#  
 READBU 1#  
 READEF 1#  
 RFLAGS 1#  
 SETPRI 1#  
 SETVEC 1#  
 SLASH 1#  
 STARS 1#  
 SVC 1#  
 XFER 1#  
 XFERF 1#  
 XFERT 1#

. ABS. 000000 000  
 000000 001  
 MICRG 001634 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MICROG.OBJ,MICROG.LST/CR/SOL/ML:TOC=SVC34R.P11,MICROG.MAC  
 RUN-TIME: 2 2 .3 SECONDS  
 RUN-TIME RATIO: 32/5=6.0  
 CORE USED: 31K (61 PAGES)

74NO MORE MICROCODE MODULES  
NOMORE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 2

SEQ 487

1		.TITLE NO MORE MICROCODE MODULES
2	000000'	.CSECT NOMORE
3		.RCALL SVC
4	000000'	SVC
5	000000'	LASTAD
6	000004'	LSLAST::
7	000001	.END

74NO MORE MICROCODE MODULES  
 NOMORE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 4  
 CROSS REFERENCE TABLE -- USER SYMBOLS

ASSEMB= 000010	5
CSAU = 000052	5#
CSAUTO= 000061	5#
CSBRK = 000022	5#
CSBSEG= 000004	5#
CSBSUB= 000002	5#
CSCEFG= 000045	5#
CSCLCK= 000062	5#
CSCLEA= 000012	5#
CSCLOS= 000035	5#
CSCLP1= 000006	5#
CSCVEC= 000036	5#
CSDCLN= 000044	5#
CSDODU= 000051	5#
CSDRPT= 000024	5#
CSDU = 000053	5#
CSEDIT= 000003	5#
CSERDF= 000055	5#
CSERHR= 000056	5#
CSERRO= 000060	5#
CSERSF= 000054	5#
CSERSO= 000057	5#
CSESCA= 000010	5#
CSSEEG= 000005	5#
CSESUB= 000003	5#
CSSTST= 000001	5#
CSEXIT= 000032	5#
CSGETB= 000026	5#
CSGETW= 000027	5#
CSGMAN= 000043	5#
CSGPHR= 000042	5#
CSGPLO= 000030	5#
CSGPRI= 000040	5#
CSINIT= 000011	5#
CSINLP= 000020	5#
CSMANI= 000050	5#
CSMEM = 000031	5#
CSMSG = 000023	5#
CSOPEN= 000034	5#
CSPTB= 000014	5#
CSPTF= 000017	5#
CSPTS= 000016	5#
CSPTX= 000015	5#
CSQIO = 000377	5#
CSRDBU= 000007	5#
CSREFG= 000047	5#
CSRESE= 000033	5#
CSREVI= 000003	5#
CSRFLA= 000021	5#
CSRPT = 000025	5#
CSSEFG= 000046	5#
CSSPRI= 000041	5#
CSVVEC= 000037	5#
CSTPRI= 000013	5#
DIAGMC= 000000	5
ESEND = 002100	5#



74NO MORE MICROCODE MODULES  
 NOMORE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 5  
 CROSS REFERENCE TABLE -- USER SYMBOLS

ESLOAD=	000035	SN
FSAU =	000015	SN
FSAUTO=	000020	SN
F SBGN =	000040	SN
FSCLEA=	000007	SN
FSDU =	000016	SN
FSEND =	000041	SN
FSHARD=	000004	SN
FSHW =	000013	SN
FSINIT=	000006	SN
FSJMP =	000050	SN
FSPOD =	000000	SN
FMSG =	000011	SN
FSPROT=	000021	SN
FSPWR =	000017	SN
FSRPT =	000012	SN
FSSEG =	000003	SN
FSSOFT=	000005	SN
FSSRV =	000010	SN
FSSUB =	000002	SN
FSSW =	000014	SN
FSTEST=	000001	SN
GBCNTO=	000200	SN
GDELTA=	000372	SN
GDISP=	000003	SN
GEXCP=	000400	SN
GHILI =	000002	SN
GLOLI =	000001	SN
GSNO =	000000	SN
GSOFFS=	000400	SN
GOSI =	000376	SN
GPRMA=	000001	SN
GPRMD=	000002	SN
GPRML=	000000	SN
GRADA=	000140	SN
GRADB=	000000	SN
GRADD=	000040	SN
GRADL=	000120	SN
GRADO=	000020	SN
GXFER=	000004	SN
GBYES =	000010	SN
ISAU =	000041	SN
ISAUTO=	000041	SN
ISCLN =	000041	SN
ISDU =	000041	SN
ISINIT=	000041	SN
ISMOD =	000041	SN
ISMSG =	000041	SN
ISPROT=	000041	SN
ISPTAB=	000041	SN
ISPR =	000041	SN
ISRPT =	000041	SN
ISSEG =	000041	SN
ISSETU=	000041	SN
ISSRV =	000041	SN
ISSUB =	000041	SN

74NO MORE MICROCODE MODULES  
 NOMORE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 6  
 CROSS REFERENCE TABLE -- USER SYMBOLS

1\$TST = 000041	5#		
J\$JMP = 000167	5#		
L\$LAST = 000004RG 002	6#		
O\$APTS = 000000	5#		
O\$AU = 000000	5#		
O\$BGR = 000000	5#		
O\$BGNS = 000000	5#		
O\$DU = 000000	5#		
O\$ERRT = 000000	5#		
O\$GNSW = 000000	5#		
O\$POIN = 000000	5#		
O\$SETU = 000000	5#	6	
SVCGBL = 000000	5#	6#	7
SVCINS = 177777	5#	6	
SVCSUB = 177777	5#		
SVCTAG = 177777	5#		
SVCTST = 177777	5#		
S\$LSYM = 010000	5#		
T\$ERRN = 000000	5#		
T\$GMAN = 000000	5#		
T\$LAST = 000001	5#	6#	
T\$LSYM = 010000	5#		
T\$LTNO = 000000	7#		
T\$NEST = 177777	5#		
T\$PTNU = 000000	5#		
T\$SAVL = 177777	5#		
T\$SEGL = 177777	5#		
T\$SUBN = 000000	5#		
T\$TAGL = 177777	5#		
T\$TAGN = 010000	5#		
T\$TEST = 000000	5#	7	
T\$TSM = 177777	5#		
T\$TSTS = 000000	5#		
X\$ALWA = 000000	5#		
X\$FALS = 000040	5#		
X\$OFFS = 000400	5#		
X\$TRUE = 000020	5#		

74NO MORE MICROCODE MODULES  
 NOMORE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 8  
 CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL	1#	5#
BERROR	1#	5#
BGNAU	1#	5#
BGNAUT	1#	5#
BGNCLN	1#	5#
BGNDU	1#	5#
BGNHRD	1#	5#
BGNHW	1#	5#
BGINI	1#	5#
BGNMOD	1#	5#
BGNMSG	1#	5#
BGNPRO	1#	5#
BGNPTA	1#	5#
BGNRPT	1#	5#
BGNSEG	1#	5#
BGNSET	1#	5#
BGNSFT	1#	5#
BGNSRV	1#	5#
BGNSUB	1#	5#
BGNSW	1#	5#
BGNTST	1#	5#
BNCOMP	1#	5#
BNERRO	1#	5#
BREAK	1#	5#
BRESET	1#	5#
CKLOOP	1#	5#
CLOCK	1#	5#
CLOSE	1#	5#
CLRVEC	1#	5#
COMMEN	1#	5#
DELAY	1#	5#
DESCRI	1#	5#
DEVTYP	1#	5#
DISPAT	1#	5#
DISPLA	1#	5#
DOCLN	1#	5#
DODU	1#	5#
DORPT	1#	5#
ENDAU	1#	5#
ENDAUT	1#	5#
ENDCLN	1#	5#
ENDCOM	1#	5#
ENDDU	1#	5#
ENDHRD	1#	5#
ENDHW	1#	5#
ENDINI	1#	5#
ENDMOD	1#	5#
ENDMSG	1#	5#
ENDPRO	1#	5#
ENDPTA	1#	5#
ENDRPT	1#	5#
ENDSEG	1#	5#
ENDSET	1#	5#
ENDSFT	1#	5#
ENDSRV	1#	5#
ENDSUB	1#	5#

74NO MORE MICROCODE MODULES  
NOMORE.MAC 07-APR-83 16:06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 9  
CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	1#	5#
ENDTST	1#	5#
EQUALS	1#	5#
ERRDF	1#	5#
ERRHRD	1#	5#
ERROR	1#	5#
ERRSF	1#	5#
ERRSOF	1#	5#
ERRTBL	1#	5#
ESCAPE	1#	5#
EXIT	1#	5#
FEQUAL	1#	5#
GETBYT	1#	5#
GETPRI	1#	5#
GETWOR	1#	5#
GMANIA	1#	5#
GMANID	1#	5#
GMANIL	1#	5#
GPHARD	1#	5#
GPRMA	1#	5#
GPRMD	1#	5#
GPRML	1#	5#
HEADER	1#	5#
INLOOP	1#	5#
IOSETU	1#	5#
IOSTAR	1#	5#
KT11	1#	5#
LASTAD	1#	5#
MANUAL	1#	5#
MEMORY	1#	5#
MSBYTE	1#	5#
MSCHEC	1#	5#
MSCNTO	1#	5#
MSCOLN	1#	5#
MSDATA	1#	5#
MSDECR	1#	5#
MSDEFA	1#	5#
MSDEDE	1#	5#
MSERRI	1#	5#
MSESCA	1#	5#
MSESCS	1#	5#
MSEXCP	1#	5#
MSEXIT	1#	5#
MSXSE	1#	5#
MSXTJ	1#	5#
MSGEN	1#	5#
MSGENB	1#	5#
MSGETS	1#	5#
MSGETT	1#	5#
MSGHGB	1#	5#
MSGNIN	1#	5#
MSGALS	1#	5#
MSGNSU	1#	5#
MSGNTA	1#	5#
MSGNTE	1#	5#
MSHAPT	1#	5#

6#

6#

6#

75NO MORE MICROCODE MODULES  
 NOMORE.MAC 07-APR-83 16.06

MACY11 30A(1052) 07-APR-83 16:53 PAGE 10  
 CROSS REFERENCE TABLE -- MACRC NAMES

MSHMAP	1#	5#
MSINCR	1#	5#
MSIOSE	1#	5#
MSLDRO	1#	5#
MSMASK	1#	5#
MSMCHI	1#	5#
MSMCLO	1#	5#
MSMSK1	1#	5#
MSPOP	1#	5#
MSPRIN	1#	5#
MSPUSH	1#	5#
MSPUT	1#	5#
MSPUT1	1#	5#
MSRADI	1#	5#
MSRBRO	1#	5#
MSRNRO	1#	5#
MSSETS	1#	5#
MSSTAR	1#	5#
MSVC	1#	5#
MSLAB	1#	5#
MSSTL	1#	5#
MSWORD	1#	5#
MSXFER	1#	5#
OPEN	1#	5#
POINTE	1#	5#
PRINTB	1#	5#
PRINTF	1#	5#
PRINTS	1#	5#
PRINTX	1#	5#
READBU	1#	5#
REDEF	1#	5#
RFLAGS	1#	5#
SETPRI	1#	5#
SETVEC	1#	5#
SLASH	1#	5#
STARS	1#	5#
SVC	1#	3#
XFER	1#	5#
XFERF	1#	5#
XFERT	1#	5#

6

4

. ABS.	000000	000
	000000	001
NOMORE	000004	002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

NOMORE.OBJ,NOMORE.LST/CR/SOL/ML:TOC=SVC34R.P11,NOMORE.MAC  
 RUN-TIME: 2 1 .3 SECONDS  
 RUN-TIME RATIO: 42/4=9.1  
 CORE USED: 31K (61 PAGES)

LNKX11 V023 7-APR-83 17:20

#CZUAAB.BIC/B:2000,CZUAAB.MAP=CZUAAB,MICROA,MICROB,MICROC,MICROD,MICROE,MICROF,MICROG,NOMORE/E

## LOAD MAP

TRANSFER ADDRESS: 000001

LOW LIMIT: 002000

HIGH LIMIT: 100564

\*\*\*\*\*

MODULE	PARAM	ADDRESS	SIZE
SECTION	ENTRY		
<. ABS.>		000000	000000
	ADR	000020	
	ADRERR	000001	
	BIT0	000001	
	BIT00	000001	
	BIT01	000002	
	BIT02	000004	
	BIT03	000010	
	BIT04	000020	
	BIT05	000040	
	BIT06	000100	
	BIT07	000200	
	BIT08	000400	
	BIT09	001000	
	BIT1	000002	
	BIT10	002000	
	BIT11	004000	
	BIT12	010000	
	BIT13	020000	
	BIT14	040000	
	BIT15	100000	
	BIT2	000004	
	BIT3	000010	
	BIT4	000020	
	BIT5	000040	
	BIT6	000100	
	BIT7	000200	
	BIT8	000400	
	BIT9	001000	
	BOE	000400	
	CLEAR	000000	
	CRCSIZ	000004	
	DATERR	000000	
	DIM	000020	
	DNI	004000	
	DNIB	000010	
	EF.COM	000036	
	EF.NEW	000035	
	EF.PWR	000034	
	EF.RES	000037	
	EF.SIA	000040	
	ENP	000400	
	ERRS	040000	
	EVL	000004	
	FATI	000400	

FATIB	000001
GETCMD	000002
GETPCB	000001
MOE	100000
IBE	010000
ICAB	040000
IDU	000040
IE	000100
IER	020000
INITH	177777
INITL	177777
INTE	000100
INTR	000200
IOADR	020000
IOSIZ	020000
ISR	000100
IXE	004000
LASH	000001
LIM	000021
LINADR	100000
LINSIZ	077774
LOE	040000
LOT	000010
MAXBYT	002756
MINBYT	000100
NIHLT	000006
NIUNI	000007
NPRERR	100000
NXPERR	040000
OWN	100000
PARERR	010000
PCEI	040000
PCEIB	000100
PL10	000200
PFNOP1	000000
PFNOP2	000003
PNOP	000006
PNT	001000
POLYH	166670
POLYL	101440
POLY16	120001
PRI	002000
PRILD	000001
PRI00	000000
PRI01	000040
PRI02	000100
PRI03	000140
PRI04	000200
PRI05	000240
PRI06	000300
PRI07	000340
PSTATE	000007
RACC	000013
RACPS	000017
RC	000012
RCEI	002000
RCEIB	000004
RDPA	000002
READY	000002
RESET	000000

RLSA	000024	
RMAL	000006	
RMTL	000010	
ROMADR	040000	
ROMSIZ	040000	
RPA	000004	
RPS	000016	
RRF	000010	
RSET	000040	
RSIDP	000022	
RTM	000014	
RUM	000003	
RXI	020000	
RXIB	000040	
SECOND	000077	
SERI	100000	
SERIB	000200	
SET	000001	
SFT	037400	
SFTB0	000400	
SFTB1	001000	
SFTB2	002000	
SFTB3	004000	
SFTB4	010000	
SFTB5	020000	
SIZ1K	004000	
SIZ2K	010000	
SIZ4K	020000	
SIZ8K	040000	
SLFT	000003	
STP	001000	
TXI	010000	
TXIB	000020	
UAM	000200	
UNIERR	020000	
UNHLT	000005	
WCSADR	000000	
WCSSIZ	020000	
WLSA	000025	
WMAL	000007	
WPA	000005	
WRF	000011	
WSIDP	000023	
WTM	000015	
XPWR	100000	
<UNAREP>	002000	000000
<UNAREP>	002000	053566
BITNAM	002310	
BITNUM	002306	
BITSTA	002312	
BNANT0	002360	
BNANT1	002420	
BNANT2	002460	
BMSG	054556	
BMSG0	055035	
BMSG1	055073	
BMSG2	055126	
BMSG3	055172	
BTBL	054676	
CHKCN1	021316	



CHKINT	022132
CHKMON	022060
CLKBR	002276
CLKCSR	002274
CLKFRE	002302
CLKSRV	024062
CLKTAB	002274
CLKVEC	002300
CLRWI	021362
CNTTAB	002626
COLTST	005753
CPUPRI	002676
CRCDAT	005526
CRCEER	005563
CRCPAT	005611
CRC16	021030
CRC32	020710
CSRNUM	002304
DATALD	003774
DBFRAM	004531
DEFADR	054625
DEFHOR	054562
DFPTBL	002262
DMABLK	004406
DMAFRM	004336
DMATO	004270
DPA	054542
DPPAT	004625
DWFRAM	004567
ERRINT	002670
FIFTST	005073
FORM1	006141
FORM10	006715
FORM11	007001
FORM12	007030
FORM13	007123
FORM15	007174
FORM16	007250
FORM17	007326
FORM18	007411
FORM19	007467
FORM2	006176
FORM20	007544
FORM21	007604
FORM22	007656
FORM23	007736
FORM24	010006
FORM25	010040
FORM26	010113
FORM27	010166
FORM28	010236
FORM29	010243
FORM3	006234
FORM30	010266
FORM31	010316
FORM32	010344
FORM33	010441
FORM34	010477
FORM35	010562
FORM36	010662

FORM37	010720
FORM38	011031
FORM39	011122
FORM40	006303
FORM40	011163
FORM41	011246
FORM42	011314
FORM43	011323
FORM44	011331
FORM45	011410
FORM46	011436
FORM47	011465
FORM48	011514
FORM49	011611
FORM50	006352
FORM50	011665
FORM51	011754
FORM52	012006
FORM53	012044
FORM54	012116
FORM55	012144
FORM56	012215
FORM57	012270
FORM58	012340
FORM59	012416
FORM60	006415
FORM60	012511
FORM61	012576
FORM62	012661
FORM63	012734
FORM64	012773
FORM65	013044
FORM66	013072
FORM67	013157
FORM68	013243
FORM69	013340
FORM70	006473
FORM70	013456
FORM71	013520
FORM72	013562
FORM73	013634
FORM74	013677
FORM75	013754
FORM76	014022
FORM77	014056
FORM78	014132
FORM79	014172
FORM80	006555
FORM80	014241
FORM81	014320
FORM82	014363
FORM83	014453
FORM89	006637
FREMEM	002324
FRESIZ	002322
FRSTIM	002674
NAFDUP	005723
HEXDAT	054560
HEXDPA	021116
HEXH	021200

HEXL	021236
HEXTBL	054651
HEXVAL	054561
INTBIT	004156
INTVEC	004121
LASFT	004026
LNKARB	005241
LNKBYT	004727
LNKMEM	004240
LPMSG	054554
LPMSG0	054740
LPMSG1	054777
LPTBL	054672
LSACP	002110
LSAPT	002036
LSAU	024026
LSAUT	002070
LSAUTO	023652
LSACP	002106
LSCLEA	023654
LSCO	002032
LSDEPO	002011
LSDESC	002706
LSDESP	002076
LSDEVP	002060
LSDISP	002124
LSDLY	002116
LSDTP	002040
LSDTYP	002034
LSDU	024020
LSDUT	002072
LSDVTY	002700
LSEF	002052
LSENVJ	002044
LSETP	002102
LSEXP1	002046
LSEXP4	002064
LSEXP5	002066
LSHARD	055256
LSHIRE	002120
LSMPCP	002016
LSMPTP	002022
LSHW	002262
LSICP	002104
LSINIT	023212
LSLADP	002026
LSLOAD	002100
LSLUN	002074
LSMREV	002050
LSNAME	002000
LSPRIO	002042
LSPROT	023204
LSPMT	002112
LSREPP	002062
LSREY	002010
LSRPT	023176
LSSPC	002056
LSSPCP	002020
LSSPTP	002024
LSSTA	002030

LSTEST	002114
LSTIML	002014
LSUMIT	002012
MAXCNT	005027
METER	002332
MICMOD	002320
MICRO	002326
MSG1	014716
MSG10	015360
MSG11	015402
MSG12	015466
MSG13	015512
MSG14	015560
MSG15	015606
MSG16	015636
MSG17	015730
MSG18	016022
MSG19	016052
MSG2	014764
MSG20	016146
MSG21	016236
MSG22	016326
MSG23	016442
MSG24	016464
MSG25	016614
MSG26	017002
MSG27	017046
MSG28	017114
MSG29	017162
MSG3	015030
MSG30	017264
MSG31	017332
MSG32	017476
MSG33	017574
MSG34	017622
MSG35	017672
MSG36	017770
MSG37	020012
MSG38	020034
MSG39	020056
MSG4	015076
MSG40	020100
MSG41	020176
MSG42	020250
MSG43	020420
MSG44	020442
MSG45	020604
MSG46	020666
MSG5	015124
MSG6	015174
MSG7	015262
MSG8	015310
MSG9	015332
MUCAST	005470
NEXPREM	002666
NOCLK	023610
NOPEER	003706
ODDBYT	004765
PATERN	002520
PAT1	002520

PAT2	002522
PAT3	002524
PAT4	002526
PAT5	002530
PAT6	002532
PCBB	002606
PCOMND	002316
PCSRO	002336
PCSROC	002350
PCSROU	002346
PCSR1	002340
PCSR1C	002352
PCSR2	002342
PCSR2C	002354
PCSR3	002344
PCSR3C	002356
PRTPAR	006061
PWHEN	002314
RACERR	003570
RACMG1	014500
RACMG2	014526
RACMG3	014570
RACMG4	014642
RACMG7	014670
RBRRUN	005650
RCVDON	004477
RETLOG	006031
REUNA	022166
RLNKAD	005114
ROMDMP	003747
RREV	054550
RRVER	003646
RSETER	003624
SLFTST	003726
SMSG00	026610
SMSG01	026632
SMSG02	026652
SMSG03	026656
SMSG04	026706
SMSG05	026742
SMSG06	026766
SMSG07	027030
SMSG10	027073
SMSG11	027140
SMSG12	027160
SMSG13	027166
SMSG14	027213
SMSG15	027214
SMSG16	027215
SMSG17	027216
SMSG20	027217
SMSG21	027233
SMSG22	027234
SMSG23	027235
SMSG24	027236
SMSG25	027237
SMSG26	027240
SMSG27	027241
SMSG30	027242
SMSG31	027304

SMSG32	027347
SMSG33	027413
SMSG34	027450
SMSG35	027512
SMSG36	027546
SMSG37	027610
SMSG40	027644
SMSG41	027721
SMSG42	027773
SMSG43	030046
SMSG44	030112
SMSG45	030163
SMSG46	030226
SMSG47	030227
SMSG50	030230
SMSG51	030307
SMSG52	030363
SMSG53	030440
SMSG54	030506
SMSG55	030561
SMSG56	030626
SMSG57	030627
SMSG60	030630
SMSG61	030644
SMSG62	030670
SMSG63	030714
SMSG64	030734
SMSG65	030740
SMSG66	030752
SMSG67	030764
SMSG70	031000
SMSG71	031012
SMSG72	031036
SMSG73	031060
SMSG74	031061
SMSG75	031062
SMSG76	031063
SMSG77	031064
SPAT1	002534
SPAT2	002542
SPAT3	002550
SPAT4	002556
SPAT5	002564
SPAT6	002572
SPAT7	002600
STAMUX	004663
STAPAT	005305
STAPDS	005417
STAREJ	005351
STMSG	026406
STTBL	026410
SWADDR	002334
SWHDR	054706
SMPACK	054552
TDRCNT	006001
TIMOFF	021302
TIMON	021264
TIMTST	004216
TLNKAD	005165
TRAP4	024034

TRNDON	004445
T1	024076
T10	031124
T11	031540
T12	032326
T13	033006
T14	033424
T15	033714
T16	034422
T17	034740
T18	035406
T19	035764
T2	024310
T20	036344
T21	036672
T22	037202
T23	037512
T24	040072
T25	040452
T26	041046
T27	041610
T28	042206
T29	042610
T3	024642
T30	043130
T31	043520
T32	044104
T33	044470
T34	045130
T35	045460
T36	046000
T37	046350
T38	046670
T39	047230
T4	024754
T40	047620
T41	050202
T42	050634
T43	051240
T44	052330
T45	053052
T46	053466
T5	025072
T6	025560
T7	025642
T8	025740
T9	026162
UDBB	002616
UNACSR	002266
UNAI NT	002672
UNAPRI	002272
UNASRV	024044
UNAVEC	002270
UNIT	002330
UNLOD	003517
UNUSED	031112
WCSMEM	004072
\$ADRER	003461
\$AFTER	003342
\$BEFOR	003333

\$BIT0	003271	
\$BIT1	003263	
\$BIT10	003174	
\$BIT11	003165	
\$BIT12	003156	
\$BIT13	003147	
\$BIT14	003140	
\$BIT15	003131	
\$BIT2	003255	
\$BIT3	003247	
\$BIT4	003241	
\$BIT5	003233	
\$BIT6	003225	
\$BIT7	003217	
\$BIT8	003211	
\$BIT9	003203	
\$CLR	003325	
\$DATER	003446	
\$DMI	003000	
\$FATI	003021	
\$GTCMD	003357	
\$GTPCB	003350	
\$ICAB	003076	
\$INTE	003043	
\$INTR	003032	
\$NCLR	003313	
\$NOP	003405	
\$NSET	003277	
\$PARER	003502	
\$PATCH	055366	
\$PCEI	002747	
\$PCTO	003107	
\$PDNDM	003417	
\$RCEI	003010	
\$RESET	003440	
\$RMTC	003120	
\$RSET	003054	
\$RXI	002760	
\$SERI	002736	
\$SET	003307	
\$SLFT	003373	
\$STOP	003433	
\$STRT	003411	
\$TXI	002770	
\$XPWR	003065	
<MICRA >	055566	000024
MICASZ	055610	
MICROA	055566	
<MICRB >	055612	002314
MICBSZ	060124	
MICROB	055612	
<MICRC >	060126	003254
MICCSZ	063400	
MICC7	063214	
MICROC	060126	
<MICRD >	063402	004706
MICDSZ	070306	
MICROD	063402	
<MICRE >	070310	002734
BFIL	072640	



```

MICESZ 073242
MICROE 070310
<MICRF > 073244 003460
MICFSZ 076722
MICROF 073244
<MICRG > 076724 001634
MICGSZ 100556
MICROG 076724
<NOMORE> 100560 000004
LSLAST 100564

```

```

*****
MODULE MICROA
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE MICROB
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE MICROC
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE MICROD
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE MICROE
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE MICROF
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE MICROG
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

*****
MODULE NO
SECTION ENTRY ADDRESS SIZE
< > 100564 000000

```

```

RUN-TIME: 5 SECONDS
5K CORE USED

```