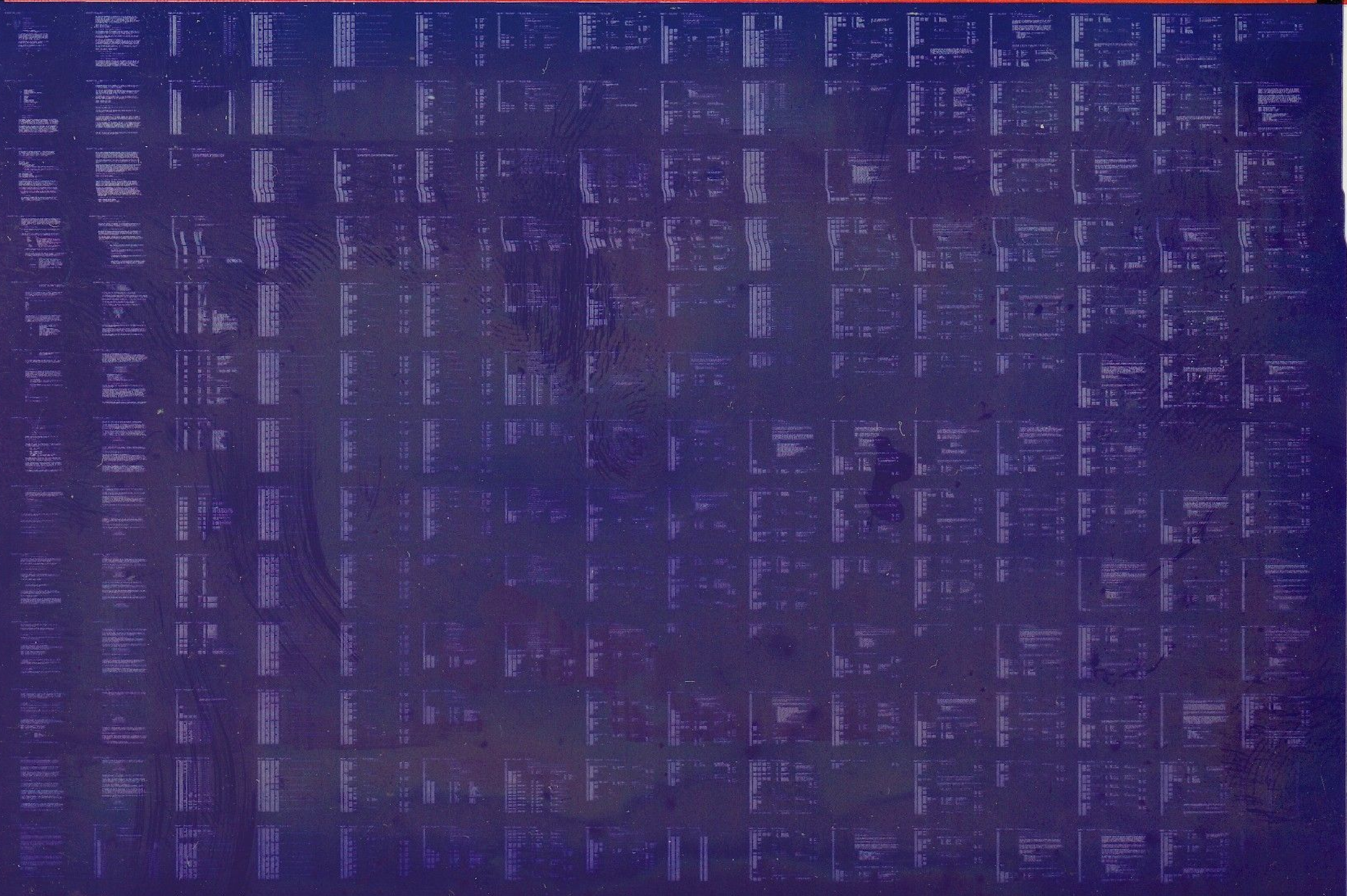
DEUNA

REPAIR DIAG
CZUAAA0

AH-T364A-MC
FICHE 1 OF 3

MAY 1983
COPYRIGHT© 1983
MADE IN USA



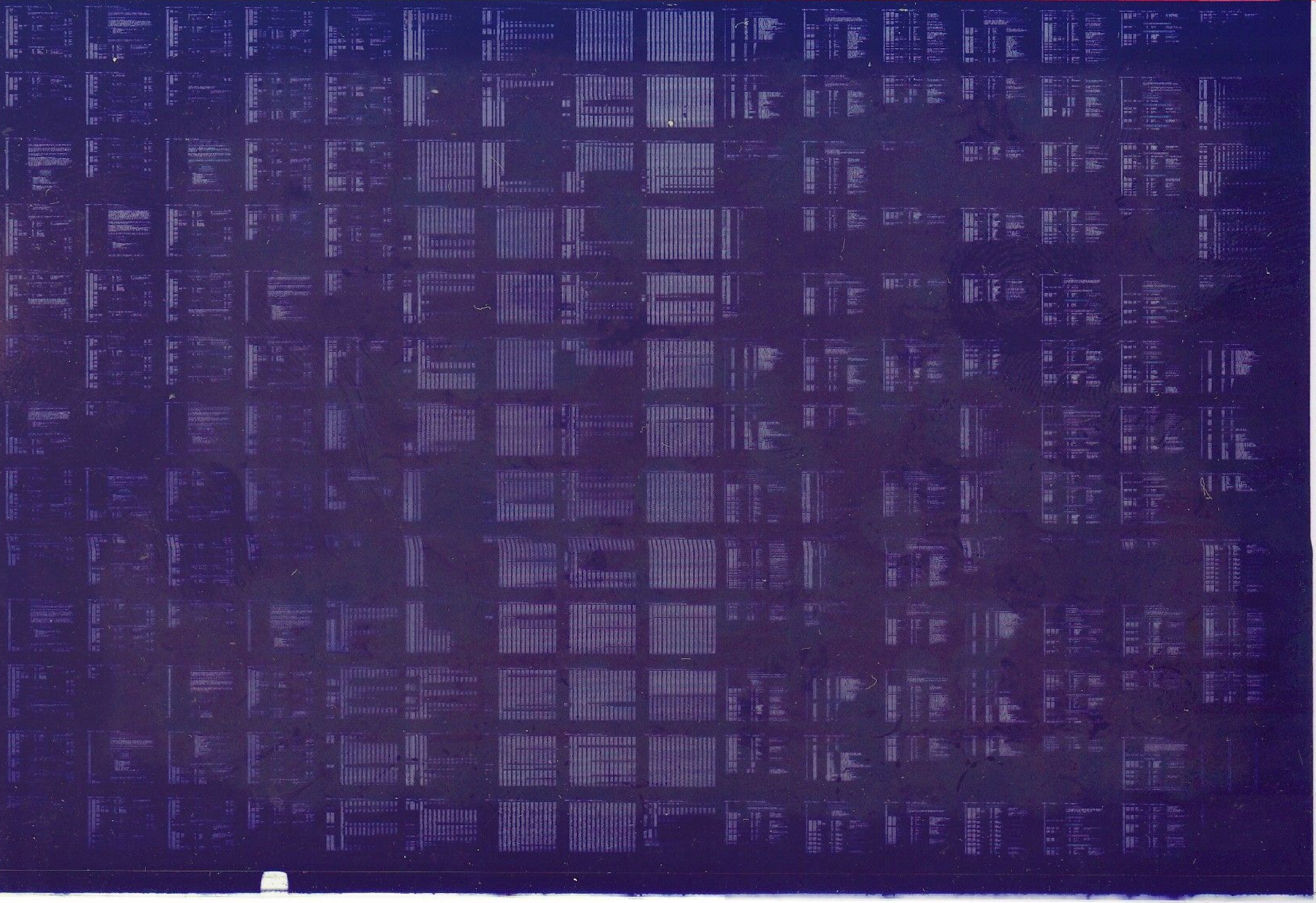
DEUNA

REPAIR DIAG
CZUAAA0

AH-T364A-MC
FICHE 2 OF 3

MAY 1983

MADE IN USA



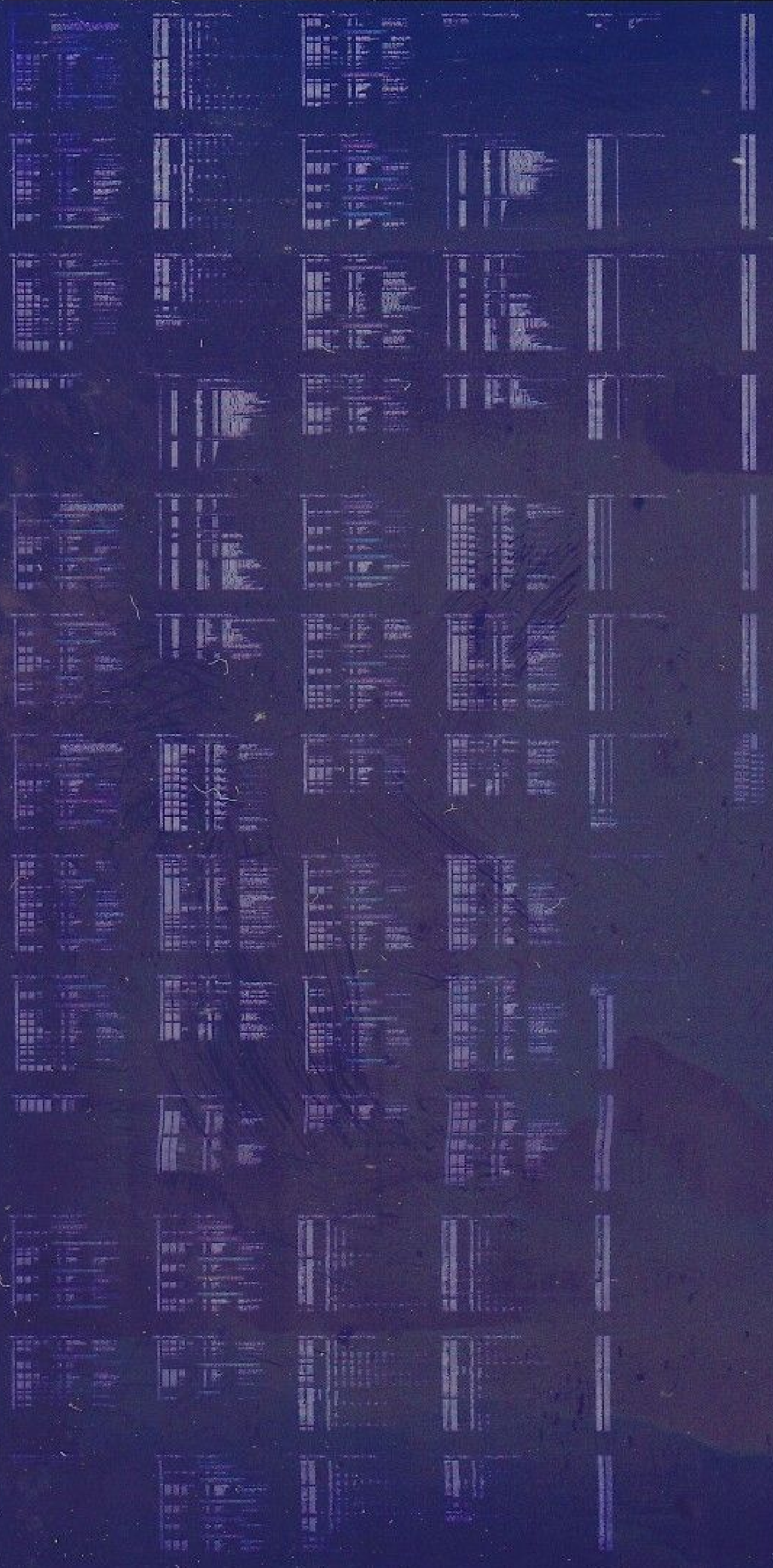
DEUNA

REPAIR DIAG
CZUAAA0

AH-T364A-MC
FICHE 3 OF 3

MAY 1983

MADE IN USA



```
                                        .REM &

                IDENTIFICATION
                --------------

        PRODUCT CODE:   AC-T363A-MC

        PRODUCT NAME:   CZUAAA0 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 AVAIL-
                ABLE 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 COM-
                MITTMENT 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             PDP             UNIBUS          MASSBUS
                DECUS           DECTAPE         VAX
```

TABLE OF CONTENTS

.PAGE
&
.REM &

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 DIAGNOSITC 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 CHQUS
XXDP+ PROGRAMMERS MANUAL
DEUNA LINK BOARD FUNCTIONAL SPECIFICATION
DEUNA PORT BOARD FUNCTIONAL SPECIFICATION
DEUNA PROGRAMMING SPECIFICATION

1.4 DIAGNOSTIC HIERARCY 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 DECX-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 VISABILITY 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 (CHQUS).

2.1 COMMANDS

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

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT 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 "DDDDD".

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

/UNITS:LIST   TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

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

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

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

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

2.3 FLAGS

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

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR

```
        UAM             UNATTENDED MODE (NO MANUAL INTERVENTION)
        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  ER: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 ERRCR 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 DEUNA BEING TESTED, INFORMATION FOR THAT DEUNA 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: PCSR0 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 CPC 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 INTERUPT 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: PCSR0 INTERRUPT BIT TEST

THIS TEST WILL VERIFY THAT EACH OF THE INTERRUPT BITS IN REGISTER PCSR0 CAN CAUSE AN INTERRUPT.

EACH OF THE INTERRUPTS OF REGISTER PCSR0 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 PCSR0.

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.

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.

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 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 PCSR0 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. 1111111111111111000000000000000011...) 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 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 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 BACKGRAOUND 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 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 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 RECFIVER 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 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 CONFLICKS BETWEEN PROCESSES AND ASSURES THAT HIGHER PRIORITY PROCESSES GET PRECEDENCE.

THIS TEST WILL VERIFY THE ABLILITY 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.

```
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 ARITRATION 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  !
                    +-----------------------+
 PCBB+6:            !   DATA TRANSFERRED    !
    !               !        FROM           !
    !               !     LINK MEMORY       !
    !               !       BY THE          !
    !               !     DMA ENGINE        !
    V               !                       !
 PCBB+4000:         +-----------------------+
```

TEST 34: STATION ADDRESS PATTERN TEST

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 THE TEST WAS SUCCESSFUL

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

```
                +-------------------------------+
PCBB+0:         !     STATION ADDRESS LOW       !
                +-------------------------------+
PCBB+2:         .   STATION ADDRESS MIDDLE      !
                +-------------------------------+
PCBB+4:         !     STATION ADDRESS HIGH      !
                +-------------------------------+
```

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 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 POSTION  !
                +-----------------------+
```

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 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

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 DEUNA 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.

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!
                +--------------------------------+
```

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 CONTATINS DIAGNOSTIC LOGIC THAT ALLOWS COLLSIONS 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
LOOPBACK STEP #	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 FOLOWS:

```
                +-----------------------+
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 TAHT 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 WIAT 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.

```
                                  .TITLE PROGRAM HEADER AND TABLES
                                  .SBTTL  PROGRAM HEADER

                                        .ENABL  AMA

000000'                                 BGNMOD

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

000000'                                 POINTER BGNRPT,BGNAU,BGNDU

000000'                                 HEADER  UNAREP,A,0,0,0,340
000000'                                                                   L$NAME::
000000'     125                                                                   .ASCII /U/
000001'     116                                                                   .ASCII /N/
000002'     101                                                                   .ASCII /A/
000003'     122                                                                   .ASCII /R/
000004'     105                                                                   .ASCII /E/
000005'     120                                                                   .ASCII /P/
000006'     000                                                                   .BYTE   0
000007'     000                                                                   .BYTE   0
000010'                                                                   L$REV::
000010'     101                                                                   .ASCII  /A/
000011'                                                                   L$DEPO::
000011'     060                                                                   .ASCII  /0/
000012'                                                                   L$UNIT::
000012' 000000                                                                    .WORD   0
000014'                                                                   L$TIML::
000014' 000000                                                                    .WORD   0
000016'                                                                   L$HPCP::
000016' 053464'                                                                   .WORD   L$HARD
000020'                                                                   L$SPCP::
000020' 000000                                                                    .WORD   0
000022'                                                                   L$HPTP::
000022' 000262'                                                                   .WORD   L$HW
000024'                                                                   L$SPTP::
000024' 000000                                                                    .WORD   0
000026'                                                                   L$LADP::
000026' 000000G                                                                   .WORD   L$LAST
000030'                                                                   L$STA::
000030' 000000                                                                    .WORD   0
000032'                                                                   L$CO::
000032' 000000                                                                    .WORD   0
000034'                                                                   L$DTYP::
000034' 000000                                                                    .WORD   0
000036'                                                                   L$APT::
000036' 000000                                                                    .WORD   0
000040'                                                                   L$DTP::
000040' 000124'                                                                   .WORD   L$DISPATCH
000042'                                                                   L$PRIO::
000042' 000340                                                                    .WORD   340
000044'                                                                   L$ENVI::
000044' 000000                                                                    .WORD   0
```

```
000046'                 L$EXP1::
000046' 000000                  .WORD   0
000050'                 L$MREV::
000050'    003                  .BYTE   C$REVISION
000051'    003                  .BYTE   C$EDIT
000052'                 L$EF::
000052' 000000                  .WORD   0
000054' 000000                  .WORD   0
000056'                 L$SPC::
000056' 000000                  .WORD   0
000060'                 L$DEVP::
000060' 000676'                 .WORD   L$DVTYP
000062'                 L$REPP::
000062' 021110'                 .WORD   L$RPT
000064'                 L$EXP4::
000064' 000000                  .WORD   0
000066'                 L$EXP5::
000066' 000000                  .WORD   0
000070'                 L$AUT::
000070' 021640'                 .WORD   L$AU
000072'                 L$DUT::
000072' 021632'                 .WORD   L$DU
000074'                 L$LUN::
000074' 000000                  .WORD   0
000076'                 L$DESP::
000076' 000704'                 .WORD   L$DESC
000100'                 L$LOAD::
000100' 104035                  EMT     E$LOAD
000102'                 L$ETP::
000102' 000000                  .WORD   0
000104'                 L$ICP::
000104' 021124'                 .WORD   L$INIT
000106'                 L$CCP::
000106' 021472'                 .WORD   L$CLEAN
000110'                 L$ACP::
000110' 021470'                 .WORD   L$AUTO
000112'                 L$PRT::
000112' 021116'                 .WORD   L$PROT
000114'                 L$TEST::
000114' 000000                  .WORD   0
000116'                 L$DLY::
000116' 000000                  .WORD   0
000120'                 L$HIME::
000120' 000000                  .WORD   0

```

```
                                  .SBTTL  DISPATCH TABLE

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

000122'                                 DISPATCH 46
000122' 000056                                                               .WORD   46
000124'                                                                 L$DISPATCH::
000124' 021710'                                                              .WORD   T1
000126' 022122'                                                              .WORD   T2
000130' 022454'                                                              .WORD   T3
000132' 022566'                                                              .WORD   T4
000134' 022704'                                                              .WORD   T5
000136' 023366'                                                              .WORD   T6
000140' 023450'                                                              .WORD   T7
000142' 023546'                                                              .WORD   T8
000144' 023770'                                                              .WORD   T9
000146' 027332'                                                              .WORD   T10
000150' 027746'                                                              .WORD   T11
000152' 030534'                                                              .WORD   T12
000154' 031214'                                                              .WORD   T13
000156' 031632'                                                              .WORD   T14
000160' 032122'                                                              .WORD   T15
000162' 032630'                                                              .WORD   T16
000164' 033146'                                                              .WORD   T17
000166' 033614'                                                              .WORD   T18
000170' 034172'                                                              .WORD   T19
000172' 034552'                                                              .WORD   T20
000174' 035100'                                                              .WORD   T21
000176' 035410'                                                              .WORD   T22
000200' 035720'                                                              .WORD   T23
000202' 036300'                                                              .WORD   T24
000204' 036660'                                                              .WORD   T25
000206' 037254'                                                              .WORD   T26
000210' 040016'                                                              .WORD   T27
000212' 040414'                                                              .WORD   T28
000214' 041016'                                                              .WORD   T29
000216' 041336'                                                              .WORD   T30
000220' 041726'                                                              .WORD   T31
000222' 042312'                                                              .WORD   T32
000224' 042676'                                                              .WORD   T33
000226' 043336'                                                              .WORD   T34
000230' 043666'                                                              .WORD   T35
000232' 044206'                                                              .WORD   T36
000234' 044556'                                                              .WORD   T37
000236' 045076'                                                              .WORD   T38
000240' 045436'                                                              .WORD   T39
000242' 046026'                                                              .WORD   T40
000244' 046410'                                                              .WORD   T41
000246' 047042'                                                              .WORD   T42
000250' 047446'                                                              .WORD   T43
000252' 050536'                                                              .WORD   T44
000254' 051260'                                                              .WORD   T45
000256' 051674'                                                              .WORD   T46
```

```
                                .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.
                                ;--

000260'                                 BGNHW   DFPTBL
000260' 000002                                                                    .WORD   L10000-L$HW/2
000262'                                                                       L$HW::
000262'                                                                       DFPTBL::

000262' 000000                          .WORD   0       ;CSR
000264' 000000                          .WORL   0       ;VECTOR

000266'                                 ENDHW
000266'                                                                       L10000:
```

```
                                        .TITLE GLOBAL AREAS
                                        .SBTTL  GLOBAL EQUATES SECTION



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

        000266'                                 EQUALS
                                        ;
                                        ; BIT DIFINITIONS
                                        ;
                000000                  BIT15== 100000
                040000                  BIT14== 40000
                020000                  BIT13== 20000
                010000                  BIT12== 10000
                004000                  BIT11== 4000
                002000                  BIT10== 2000
                001000                  BIT09== 1000
                000400                  BIT08== 400
                000200                  BIT07== 200
                000100                  BIT06== 100
                000040                  BIT05== 40
                000020                  BIT04== 20
                000010                  BIT03== 10
                000004                  BIT02== 4
                000002                  BIT01== 2
                000001                  BIT00== 1
                                        ;
                001000                  BIT9==  BIT09
                000400                  BIT8==  BIT08
                000200                  BIT7==  BIT07
                000100                  BIT6==  BIT06
                000040                  BIT5==  BIT05
                000020                  BIT4==  BIT04
                000010                  BIT3==  BIT03
                000004                  BIT2==  BIT02
                000002                  BIT1==  BIT01
                000001                  BIT0==  BIT00
                                        ;
                                        ; EVENT FLAG DEFINITIONS
                                        ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                        ;
                000040                  EF.START==      32.                             ; START COMMAND WAS ISSUED
                000037                  EF.RESTART==    31.                             ; RESTART COMMAND WAS ISSUED
                000036                  EF.CONTINUE==   30.                             ; CONTINUE COMMAND WAS ISSUED
                000035                  EF.NEW==        29.                             ; A NEW PASS HAS BEEN STARTED
                000034                  EF.PWR==        28.                             ; A POWER-FAIL/POWER-UP OCCURRED
                                        ;
                                        ;
                                        ; PRIORITY LEVEL DEFINITIONS
                                        ;
                000340                  PRI07== 340
                000300                  PRI06== 300
        000240          PRI05== 240
        000200          PRI04== 200
        000140          PRI03== 140
        000100          PRI02== 100
        000040          PRI01== 40
        000000          PRI00== 0
                        ;
                        ;OPERATOR FLAG BITS
                        ;
        000004          EVL==        4
        000010          LOT==       10
        000020          ADR==       20
        000040          IDU==       40
        000100          ISR==      100
        000200          UAM==      200
        000400          BOE==      400
        001000          PNT==     1000
        002000          PRI==     2000
        004000          IXE==     4000
        010000          IBE==    10000
        020000          IER==    20000
        040000          LOE==    40000
        100000          HOE==   100000

        000077          SECOND==63.             ;63 LINE CLOCK TICKS = APROX. 1 SECOND
        000001          SET==   1
        000000          CLEAR== 0
        004000          SIZ1K== 4000            ;1K WORDS
        010000          SIZ2K== SIZ1K*2         ;2K WORDS
        020000          SIZ4K== SIZ2K*2         ;4K WORDS
        040000          SIZ8K== SIZ4K*2         ;8K WORDS
        020000          WCSSIZ==SIZ4K           ;SIZE OF THE DEUNA WRITEABLE CONTROL STORE
        020000          IOSIZ==SIZ4K            ;SIZE OF THE DEUNA I/O PAGE
        040000          ROMSIZ==SIZ8K           ;SIZE OF THE DEUNA ROM IN WORDS
        077774          LINSIZ==SIZ8K*2-4       ;SIZE OF THE DEUNA LINK MEMORY
        000000          WCSADR==0               ;INTERNAL BASE ADDRESS OF WCS
        020000          IOADR==WCSADR+WCSSIZ    ;INTERNAL BASE ADDRESS OF THE I/O PAGE FOR THE T11
        040000          ROMADR==IOADR+IOSIZ     ;INTERNAL BASE ADDRESS OF THE DEUNA ROM
        100000          LINADR==ROMADR+ROMSIZ   ;INTERNAL BASE ADDRESS OF THE DEUNA LINK MEMORY
        000100          IE==    BIT6            ;INTERRUPT ENABLE
        177777          INITH== -1              ;INITIAL HIGH WORD FOR 32 BIT CRC CALCULATOR
        177777          INITL== -1              ;INITIAL LOW WORD FOR 32 BIT CRC CALCULATOR
        166670          POLYH== 166670          ;FUNCTION POLYNOMIAL HIGH WORD FOR 32 BIT CRC
        101440          POLYL== 101440          ;FUNCTION POLYNOMIAL LOW WORD FOR 32 BIT CRC
        120001          POLY16== 120001         ;FUNCTION POLYNOMIAL FOR 16 BIT CRC
        000000          DATERR==0               ;DATA ERROR INDICATER FOR LINK MEMORY TESTS
        000001          ADRERR==1               ;ADDRESS ERROR INDICATOR FOR LINK MEMORY TESTS
        002756          MAXBYT==1518.           ;MAXIMUM NUMBER OF BYTES RECEIVER CAN HANDLE
        000100          MINBYT==64.             ;MINIMUM NUMBER OF BYTES RECEIVER CAN HANDLE
        000004          CRCSIZ==4.              ;SIZE OF CRC
        000001          INMON=1                 ;IN MICROMONITOR STATE
        000002          INTST=2                 ;IN A TEST STATF
        000003          INERR=3                 ;IN ERROR STATE
```

```

                        ; P  RO - PORT CONTROL AND STATUS  REGISTER 0 (OPERATIONAL MICROCODE DEFINITIONS)

100000                  SERI    ==      BIT15           ; STATUS ERROR INTERRUPT
040000                  PCEI    ==      BIT14           ; PORT COMMAND ERROR INTERRUPT
020000                  RXI     ==      BIT13           ; RECEIVE RING INTERRUPT
010000                  TXI     ==      BIT12           ; TRANSMIT RING INTERRUPT
004000                  DNI     ==      BIT11           ; DONE INTERRUPT
002000                  RCEI    =-      BIT10           ; RECEIVE BUFFER UNAVAILABLE
                        ;
000400                  FATI    ==      BIT08           ; FATAL ERROR INTERERUPT
000200                  INTR    ==      BIT07           ; INTERRUPT SUMMARY <15:08>
000100                  INTE    ==      BIT06           ; INTERRUPT ENABLE
000040                  RSET    ==      BIT05           ; UNA RESET

                        ;
                        ;PCSR0 - PORT CONTROL AND STATUS REGISTER 0 (DIAGNOSTIC MICROCODE DEFINITIONS)
                        ;
100000                  NPRERR  ==      BIT15           ;T11 NPR TIMEOUT INTERRUPT OCCURRED
040000                  NXMERR  ==      BIT14           ;T11 NON-EXISTANT MEMORY TIMEOUT OCCURRED
020000                  UNIERR  ==      BIT13           ;T11 UNEXPECTED INTERRUPT OCCURRED
010000                  PARERR  ==      BIT12           ;T11 LINK MEMORY PARITY ERROR OCCURRED
                        ;
                        ;PORT COMMANDS <03:00>
000001                  GETPCB  ==      BIT00
000002                  GETCMD  ==      BIT01
000006                  PNOP    ==      BIT01!BIT02
000003                  SLFT    ==      BIT00!BIT01
                        ;
                        ;PCSR1 - PORT CONTROL AND STATUS REGISTER 1
100000                  XPWR    ==      BIT15           ; TRANSCEIVER POWER BAD
040000                  ICAB    ==      BIT14           ; PORT/LINK CABLING OK
                        ;
                        ;SELF TEST ERROR CODE <13:08>
                        ;
000200                  PCTO    ==      BIT07           ; PORT COMMAND TIMEOUT
                        ;
000010                  RMTC    ==      BIT03           ; REMOTE CONSOLE RESERVED
000400                  SFTB0   ==      BIT8            ;FIRST BIT OF SELF TEST FIELD
001000                  SFTB1   ==      BIT9            ;SECOND
002000                  SFTB2   ==      BIT10           ;THIRD
004000                  SFTB3   ==      BIT11           ;FOURTH
010000                  SFTB4   ==      BIT12           ;FIFTH
020000                  SFTB5   ==      BIT13           ;SIXTH
037400                  SFT     ==      BIT8!BIT9!BIT10!BIT11!BIT12!BIT13       ;SELF TEST FIELD
                        ;
                        ;PORT STATE <02:00>
                        ;
000007                  PSTATE  ==      BIT00!BIT01!BIT02 ;PORT STATE BITS OF PCSR1
000000                  RESET   ==      0               ;RESET STATE (NOT REALLY A STATE)
000001                  PRILD   ==      BIT00           ; PRIMARY LOAD STATE
000002                  READY   ==      BIT01           ;READY STATE
000003                  RUN     ==      BIT00!BIT01     ;RUNNING STATE
000005                  UNIHLT  ==      BIT00!BIT02     ;UNIBUS HALTED STATE
000006                  NIHLT   ==      BIT01!BIT02     ;NI HALTED STATE
000007                  NIUNI   ==      BIT00!BIT01!BIT02 ;UNIBUS AND NI HALTED STATE
```

```
                              ;
                              ;DESCRIPTOR RING DEFINITIONS
          100000              OWN     ==      BIT15
          040000              ERRS    ==      BIT14
          001000              STP     ==      BIT09
          000400              ENP     ==      BIT08
                              ;
                              ;PORT FUNCTION CODES
                              ;

          000000              PFNOP1  ==      0        ;NOP PORT FUNCTION #1
          000001              LASM    ==      1        ;LOAD AND START MICROADDRESS
          000002              RDPA    ==      2        ;READ DEFAULT PHYSICAL ADDRESS
          000003              PFNOP2  ==      3        ;NOP PORT FUNCTION #2
          000004              RPA     ==      4        ;READ PHYSICAL ADDRESS
          000005              WPA     ==      5        ;WRITE PHYSICAL ADDRESS
          000006              RMAL    ==      6        ;READ MULTICAST ADDRESS LIST
          000007              WMAL    ==      7        ;WRITE MULTICAST ADDRESS LIST
          000010              RRF     ==      10       ;READ RING FORMAT
          000011              WRF     ==      11       ;WRITE RING FORMAT
          000012              RC      ==      12       ;READ COUNTERS
          000013              RACC    ==      13       ;READ AND CLEAR COUNTERS
          000014              RTM     ==      14       ;READ THE MODE
          000015              WTM     ==      15       ;WRITE THE MODE
          000016              RPS     ==      16       ;READ PORT STATUS
          000017              RACPS   ==      17       ;READ AND CLEAR PORT STATUS
          000020              DIM     ==      20       ;DUMP INTERNAL MEMORY
          000021              LIM     ==      21       ;LOAD INTERNAL MEMORY
          000022              RSIDP   ==      22       ;READ SYSTEM ID PARAMETERS
          000023              WSIDP   ==      23       ;WRITE SYSTEM ID PARAMETERS
          000024              RLSA    ==      24       ;READ LOAD SERVER ADDRESS
          000025              WLSA    ==      25       ;WRITE LOAD SERVER ADDRESS
```

```


                               .SBTTL  GLOBAL DATA SECTION

                               ;++
                               ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                               ; IN MORE THAN ONE TEST.
                               ;--

 000266' 000000                UNACSR:::.WORD   0       ;CSR
 000270' 000000                UNAVEC:::.WORD   0       ;VECTOR
 000272' 000000                UNAPRI:::.WORD   0       ;PRIORITY

 000274'                       CLKTAB::
 000274' 000000                CLKCSR:::.WORD   0       ;LINE CLOCK STATUS REGISTER
 000276' 000000                CLKBR:: .WORD    0       ;LINE CLOCK PRIORITY
 000300' 000000                CLKVEC:::.WORD   0       ;LINE CLOCK VECTOR
 000302' 000000                CLKFRE:::.WORD   0       ;LINE CLOCK FREQUENCY

 000304' 000000                CSRNUM:::.WORD   0       ;PCSR NUMBER
 000306' 000000                BITNUM:::.WORD   0       ;BIT NUMBER
 000310' 000000                BITNAM:::.WORD   0       ;POINTER TO BIT NAME ASCII STRING
 000312' 000000                BITSTA:::.WORD   0       ;POINTER TO BIT STATUS ASCII STRING
 000314' 000000                PWHEN:: .WORD    0       ;POINTER TO 'BEFORE' OR 'AFTER' ASCII STRING
 000316' 000000                PCOMND:::.WORD   0       ;POINTER TO PORT COMMAND ASCII STRING
 000320' 000000                MICMOD:::.WORD   0       ;POINTER TO MICROCODE MODULE # LOADED

 000322' 000000                FRESIZ:::.WORD   0       ;POINTER TO WORD CONTAINING SIZE OF FREE MEMORY
 000324' 000000                FREMEM:::.WORD   0       ;POINTER TO FREE MEMORY SPACE

 000326' 000000                MICRO:: .WORD    0       ;CURRENT MICROCODE MODULE LOADED
 000330' 000000                UNIT::  .WORD    0       ;CURRENT UNIT NUMBER BEING TESTED
 000332' 000000                METER:: .WORD    0       ;CLOCK TICKS
 000334' 021040                SWADDR:::.WORD   21040   ;INTERNAL ADDRESS OF SWITCH PACK

 000336' 000000                PCSR0:: .WORD    0       ;PORT CONTROL AND STATUS REGISTER 0
 000340' 000000                PCSR1:: .WORD    0       ;PORT CONTROL AND STATUS REGISTER 1
 000342' 000000                PCSR2:: .WORD    0       ;PORT CONTROL AND STATUS REGISTER 2
 000344' 000000                PCSR3:: .WORD    0       ;PORT CONTROL AND STATUS REGISTER 3

 000346' 000000                PCSR0C:::.WORD   0       ;PCSR0 CONTENTS
 000350' 000000                PCSR1C:::.WORD   0       ;PCSR1 CONTENTS
 000352' 000000                PCSR2C:::.WORD   0       ;PCSR2 CONTENTS
 000354' 000000                PCSR3C:::.WORD   0       ;PCSR3 CONTENTS

 000356'                       BNAMT0::                 ;TABLE OF POINTERS TO BIT NAME MNEMONICS FOR PCSR0
 000356' 001267'                       $BIT0
 000360' 001261'                       $BIT1
 000362' 001253'                       $BIT2
 000364' 001245'                       $BIT3
 000366' 001237'                       $BIT4
 000370' 001052'                       $RSET
 000372' 001041'                       $INTE
 000374' 001030'                       $INTR
 000376' 001017'                       $FATI
 000400' 001201'                       $BIT9
```

```
000402' 001006'                           $RCEI
000404' 000776'                           $DNI
000406' 000766'                           $TXI
000410' 000756'                           $RXI
000412' 000745'                           $PCEI
000414' 000734'                           $SERI
000416'                         BNAMT1::                        ;TABLE OF POINTERS TO BIT NAME MNEMONICS FOR PCSR1
000416' 000000                            .WORD    0
000420' 000000                            .WORD    0
000422' 000000                            .WORD    0
000424' 001116'                           $RMTC
000426' 001237'                           $BIT4
000430' 001231'                           $BIT5
000432' 001223'                           $BIT6
000434' 001105'                           $PCTO
000436' 000000                            .WORD    0
000440' 000000                            .WORD    0
000442' 000000                            .WORD    0
000444' 000000                            .WORD    0
000446' 000000                            .WORD    0
000450' 000000                            .WORD    0
000452' 001074'                           $ICAB
000454' 001063'                           $XPWR

000456'                         BNAMT2::                        ;TABLE OF GENERIC BIT NAME MNEMONICS
000456' 001267'                           $BIT0
000460' 001261'                           $BIT1
000462' 001253'                           $BIT2
000464' 001245'                           $BIT3
000466' 001237'                           $BIT4
000470' 001231'                           $BIT5
000472' 001223'                           $BIT6
000474' 001215'                           $BIT7
000476' 001207'                           $BIT8
000500' 001201'                           $BIT9
000502' 001172'                           $BIT10
000504' 001163'                           $BIT11
000506' 001154'                           $BIT12
000510' 001145'                           $BIT13
000512' 001136'                           $BIT14
000514' 001127'                           $BIT15

000516'                         PATERN::
000516' 125252                  PAT1::    .WORD    ^B1010101010101010
000520' 052525                  PAT2::    .WORD    ^B0101010101010101
000522' 031463                  PAT3::    .WORD    ^B0011001100110011
000524' 007417                  PAT4::    .WORD    ^B0000111100001111
000526' 000377                  PAT5::    .WORD    ^B0000000011111111
000530' 177777                  PAT6::    .WORD    ^B1111111111111111

000532' 125252                  SPAT1::   .WORD    ^B1010101010101010
000534' 125252                            .WORD    ^B1010101010101010
000536' 125252                            .WORD    ^B1010101010101010
000540' 052525                  SPAT2::   .WORD    ^B0101010101010101
000542' 052525                            .WORD    ^B0101010101010101
000544' 052525                            .WORD    ^B0101010101010101
```

```
000546' 031463              SPAT3:: .WORD    ^B0011001100110011
000550' 031463                      .WORD    ^B0011001100110011
000552' 031463                      .WORD    ^B0011001100110011
000554' 007417              SPAT4:: .WORD    ^B0000111100001111
000556' 007417                      .WORD    ^B0000111100001111
000560' 007417                      .WORD    ^B0000111100001111
000562' 000377              SPAT5:: .WORD    ^B0000000011111111
000564' 000377                      .WORD    ^B0000000011111111
000566' 000377                      .WORD    ^B0000000011111111
000570' 177777              SPAT6:: .WORD    ^B1111111111111111
000572' 000000                      .WORD    ^B0000000000000000
000574' 177777                      .WORD    ^B1111111111111111
000576' 177777              SPAT7:: .WORD    ^B1111111111111111
000600' 000377                      .WORD    ^B0000000011111111
000602' 000000                      .WORD    ^B0000000000000000

000604' 000000              PCBB::  .WORD    0       ;PORT CONTROL BLOCK
000606' 000000                      .WORD    0       ;+2
000610' 000000                      .WORD    0       ;+4
000612' 000000                      .WORD    0       ;+6

000614' 000000              UDBB::  .WORD    0       ;UNIBUS DATA BLOCK BASE
000616' 000000                      .WORD    0       ;+2
000620' 000000                      .WORD    0       ;+4
000622' 000000                      .WORD    0       ;+6

000624' 000020              CNTTAB:::.BLKW   16.     ;HOLDS THE 16 RETRY WAIT COUNTS FOR TEST 39

000664' 000000              NEXMEM:::.WORD   0       ;NON-EXISTANT MEMORY FLAG
000666' 000000              ERRINT:::.WORD   0       ;HOLDS PCSRO INTERRUPT FLAG BITS FROM MICROCODE
000670' 000000              UNAINT:::.WORD   0       ;UNA INTERRUPT FLAG
000672' 000000              FRSTIM:::.WORD   0       ;FLAG INDICATING FIRST TIME THROUGH TESTS FOR ALL UNITS
000674' 000000              CPUPRI:::.WORD   0       ;CPU PRIORITY BEFORE INTERRUPT OCCURRED

                            .EVEN
```

```
                                        .SBTTL  GLOBAL TEXT SECTION

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

                                        ;
                                        ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                        ;
000676'                                         DEVTYP  <DEUNA>
000676'
000676' 042504  047125  000101                                                          L$DVTYP::
                                                                                                .ASCIZ  /DEUNA/
                                                                                                .EVEN

                                        ;
                                        ; TEST DESCRIPTION
                                        ;
000704'                                         DESCRIPT        <DEUNA REPAIR DIAGNOSTIC>
000704'                                                                                 L$DESC::
000704' 042504  047125  020101                                                                  .ASCIZ  /DFUNA REPAIR DI
000712' 042522  040520  051111
000720' 042040  040511  047107
000726' 051517  044524  000103
                                                                                                .EVEN

000734' 042523  044522  041040  $SERI:: .ASCIZ  /SERI BIT/
000742' 052111     000
000745'    120  042503  020111  $PCEI:: .ASCIZ  /PCEI BIT/
000752' 044502  000124
000756' 054122  020111  044502  $RXI::  .ASCIZ  /RXI BIT/
000764' 000124
000766' 054124  020111  044502  $TXI::  .ASCIZ  /TXI BIT/
000774' 000124
000776' 047104  020111  044502  $DNI::  .ASCIZ  /DNI BIT/
001004' 000124
001006' 041522  044505  041040  $RCEI:: .ASCIZ  /RCEI BIT/
001014' 052111     000
001017'    106  052101  020111  $FATI:: .ASCIZ  /FATI BIT/
001024' 044502  000124
001030' 047111  051124  041040  $INTR:: .ASCIZ  /INTR BIT/
001036' 052111     000
001041'    111  052116  020105  $INTE:: .ASCIZ  /INTE BIT/
001046' 044502  000124
001052' 051522  052105  041040  $RSET:: .ASCIZ  /RSET BIT/
001060' 052111     000
001063'    130  053520  020122  $XPWR:: .ASCIZ  /XPWR BIT/
001070' 044502  000124
001074' 041511  041101  041040  $ICAB:: .ASCIZ  /ICAB BIT/
001102' 052111     000
001105'    120  052103  020117  $PCTO:: .ASCIZ  /PCTO BIT/
```

```
001112' 044502  000124
001116' 046522  041524  041040  $RMTC:: .ASCIZ  /RMTC BIT/
001124' 052111     000
001127'    102  052111  030440  $BIT15:::.ASCIZ  /BIT 15/
001134' 000065
001136' 044502  020124  032061  $BIT14:::.ASCIZ  /BIT 14/
001144'    000
001145'    102  052111  030440  $BIT13:::.ASCIZ  /BIT 13/
001152' 000063
001154' 044502  020124  031061  $BIT12:::.ASCIZ  /BIT 12/
001162'    000
001163'    102  052111  030440  $BIT11:::.ASCIZ  /BIT 11/
001170' 000061
001172' 044502  020124  030061  $BIT10:::.ASCIZ  /BIT 10/
001200'    000
001201'    102  052111  034440  $BIT9:: .ASCIZ  /BIT 9/
001206'    000
001207'    102  052111  034040  $BIT8:: .ASCIZ  /BIT 8/
001214'    000
001215'    102  052111  033440  $BIT7:: .ASCIZ  /BIT 7/
001222'    000
001223'    102  052111  033040  $BIT6:: .ASCIZ  /BIT 6/
001230'    000
001231'    102  052111  032440  $BIT5:: .ASCIZ  /BIT 5/
001236'    000
001237'    102  052111  032040  $BIT4:: .ASCIZ  /BIT 4/
001244'    000
001245'    102  052111  031440  $BIT3:: .ASCIZ  /BIT 3/
001252'    000
001253'    102  052111  031040  $BIT2:: .ASCIZ  /BIT 2/
001260'    000
001261'    102  052111  030440  $BIT1:: .ASCIZ  /BIT 1/
001266'    000
001267'    102  052111  030040  $BIT0:: .ASCIZ  /BIT 0/
001274'    000
001275'    116  052117  051440  $NSET:: .ASCIZ  /NOT SET/
001302' 052105     000
001305'    123  052105     000  $SET::   .ASCIZ  /SET/
001311'    116  052117  041440  $NCLR:: .ASCIZ  /NOT CLEAR/
001316' 042514  051101     000
001323'    103  042514  051101  $CLR::   .ASCIZ  /CLEAR/
001330'    000
001331'    102  043105  051117  $BEFOR:::.ASCIZ  /BEFORE/
001336' 000105
001340' 043101  042524  000122  $AFTER:::.ASCIZ  /AFTER/
001346' 042507  050124  041103  $GTPCB:::.ASCIZ  /GETPCB/
001354'    000
001355'    107  052105  041440  $GTCMD:::.ASCIZ  /GET COMMAND/
001362' 046517  040515  042116
001370'    000
001371'    123  046105  020106  $SLFT:: .ASCIZ  /SELF TEST/
001376' 042524  052123     000
001403'    116  050117     000  $NOP::   .ASCIZ  /NOP/
001407'    123  040524  052122  $STRT:: .ASCIZ  /START/
001414'    000
001415'    104  046505  047101  $PDNDM:::.ASCIZ  /DEMAND POLL/
```

```
2024  001422' 020104  047520  046114
2025  001430'    000
2026  001431'    123  047524  000120  $STOP:: .ASCIZ  /STOP/
2027  001436' 042522  042523  000124  $RESET:::.ASCIZ  /RESET/
2028  001444' 040504  040524  042440  $DATER:::.ASCIZ  /DATA ERROR/
2029  001452' 051122  051117     000
2030  001457'    101  042104  042522  $ADRER:::.ASCIZ  /ADDRESSING ERROR/
2031  001464' 051523  047111  020107
2032  001472' 051105  047522  000122
2033  001500' 040520  044522  054524  $PARER:::.ASCIZ  /PARITY ERROR/
2034  001506' 042440  051122  051117
2035  001514'    000
2036
2037                                  ;
2038                                  ; FORMAT STATEMENTS USED IN PRINT CALLS
2039                                  ;
2040
2041  001515'    045  022516  052501  UNLOD:: .ASCIZ  /%N%AUNABLE TO LOAD MICROCODE MODULE %T%N/
2042  001522' 040516  046102  020105
2043  001530' 047524  046040  040517
2044  001536' 020104  044515  051103
2045  001544' 041517  042117  020105
2046  001552' 047515  052504  042514
2047  001560' 022440  022524  000116
2048  001566' 040503  047116  052117  NOCLK:: .ASCIZ  /CANNOT TEST DEUNA BECAUSE SYSTEM DOES NOT HAVE A LINE CLOCK!/
2049  001574' 052040  051505  020124
2050  001602' 042504  047125  020101
2051  001610' 042502  040503  051525
2052  001616' 020105  054523  052123
2053  001624' 046505  042040  042517
2054  001632' 020123  047516  020124
2055  001640' 040510  042526  040440
2056  001646' 046040  047111  020105
2057  001654' 046103  041517  020513
2058  001662'    000
2059  001663'    122  043505  051511  RACERR:::.ASCIZ  /REGISTER ACCESS TEST FAILED/
2060  001670' 042524  020122  041501
2061  001676' 042503  051523  052040
2062  001704' 051505  020124  040506
2063  001712' 046111  042105     000
2064  001717'    122  051505  052105  RSETER:::.ASCIZ  /RESET TEST FAILED/
2065  001724' 052040  051505  020124
2066  001732' 040506  046111  042105
2067  001740'    000
2068  001741'    122  043505  051511  RRWER:: .ASCIZ  'REGISTER READ/WRITE TEST FAILED'
2069  001746' 042524  020122  042522
2070  001754' 042101  053457  044522
2071  001762' 042524  052040  051505
2072  001770' 020124  040506  046111
2073  001776' 042105     000
2074  002001'    116  050117  052040  NOPERR:::.ASCIZ  /NOP TEST FAILED/
2075  002006' 051505  020124  040506
2076  002014' 046111  042105     000
2077  002021'    123  046105  020106  SLFTST:::.ASCIZ  /SELF TEST FAILED/
2078  002026' 042524  052123  043040
2079  002034' 044501  042514  000104
```

```
 2080  002042' 047522  020115  052504  ROMDMP::.ASCIZ  /ROM DUMP TEST FAILED/
 2081  002050' 050115  052040  051505
 2082  002056' 020124  040506  046111
 2083  002064' 042105     000
 2084  002067'    127  051503  046040  DATALD::.ASCIZ  'WCS LOAD/DUMP TEST FAILED'
 2085  002074' 040517  027504  052504
 2086  002102' 050115  052040  051505
 2087  002110' 020124  040506  046111
 2088  002116' 042105     000
 2089  002121'    114  040517  020104  LASFT:: .ASCIZ  /LOAD AND START FUNCTION TEST FAILED/
 2090  002126' 047101  020104  052123
 2091  002134' 051101  020124  052506
 2092  002142' 041516  044524  047117
 2093  002150' 052040  051505  020124
 2094  002156' 040506  046111  042105
 2095  002164'    000
 2096  002165'    127  051503  046440  WCSMEM::.ASCIZ  /WCS MEMORY TEST FAILED/
 2097  002172' 046505  051117  020131
 2098  002200' 042524  052123  043040
 2099  002206' 044501  042514  000104
 2100  002214' 047111  042524  051122  INTVEC::.ASCIZ  /INTERRUPT VECTOR TEST FAILED/
 2101  002222' 050125  020124  042526
 2102  002230' 052103  051117  052040
 2103  002236' 051505  020124  040506
 2104  002244' 046111  042105     000
 2105  002251'    120  051503  030122  INTBIT::.ASCIZ  /PCSRO INTERRUPT BIT TEST FAILED/
 2106  002256' 044440  052116  051105
 2107  002264' 052522  052120  041040
 2108  002272' 052111  052040  051505
 2109  002300' 020124  040506  046111
 2110  002306' 042105     000
 2111  002311'    124  046511  051105  TIMTST::.ASCIZ  /TIMER TEST FAILED/
 2112  002316' 052040  051505  020124
 2113  002324' 040506  046111  042105
 2114  002332'    000
 2115  002333'    114  047111  020113  LNKMEM::.ASCIZ  /LINK MEMORY TEST FAILED/
 2116  002340' 042515  047515  054522
 2117  002346' 052040  051505  020124
 2118  002354' 040506  046111  042105
 2119  002362'    000
 2120  002363'    104  040515  023440  DMATO:: .ASCIZ  /DMA 'TO' ADDRESS REGISTER TEST FAILED/
 2121  002370' 047524  020047  042101
 2122  002376' 051104  051505  020123
 2123  002404' 042522  044507  052123
 2124  002412' 051105  052040  051505
 2125  002420' 020124  040506  046111
 2126  002426' 042105     000
 2127  002431'    104  040515  023440  DMAFRM::.ASCIZ  /DMA 'FROM' ADDRESS REGISTER TEST FAILED/
 2128  002436' 051106  046517  020047
 2129  002444' 042101  051104  051505
 2130  002452' 020123  042522  044507
 2131  002460' 052123  051105  052040
 2132  002466' 051505  020124  040506
 2133  002474' 046111  042105     000
 2134  002501'    104  040515  041040  DMABLK::.ASCIZ  /DMA BLOCK TRANSFER TEST FAILED/
 2135  002506' 047514  045503  052040
```

```
002514' 040522  051516  042506
002522' 020122  042524  052123
002530' 043040  044501  042514
002536' 000104
002540' 051124  047101  046523  TRNDON::.ASCIZ  /TRANSMIT DONE TEST FAILED/
002546' 052111  042040  047117
002554' 020105  042524  052123
002562' 043040  044501  042514
002570' 000104
002572' 042522  042503  053111  RCVDON::.ASCIZ  /RECEIVER DONE TEST FAILED/
002600' 051105  042040  047117
002606' 020105  042524  052123
002614' 043040  044501  042514
002622' 000104
002624' 040504  040524  041040  DBFRAM::.ASCIZ  /DATA BYTE FRAMING TEST FAILED/
002632' 052131  020105  051106
002640' 046501  047111  020107
002646' 042524  052123  043040
002654' 044501  042514  000104
002662' 040504  040524  053440  DWFRAM::.ASCIZ  /DATA WORD FRAMING TEST FAILED/
002670' 051117  020104  051106
002676' 046501  047111  020107
002704' 042524  052123  043040
002712' 044501  042514  000104
002720' 040504  040524  050040  DPPAT:: .ASCIZ  /DATA PATH PATTERN TEST FAILED/
002726' 052101  020110  040520
002734' 052124  051105  020116
002742' 042524  052123  043040
002750' 044501  042514  000104
002756' 052123  052101  051525  STAMUX::.ASCIZ  /STATUS MUX VERIFICATION TEST FAILED/
002764' 046440  054125  053040
002772' 051105  043111  041511
003000' 052101  047511  020116
003006' 042524  052123  043040
003014' 044501  042514  000104
003022' 044514  045516  041040  LNKBYT::.ASCIZ  /LINK BYTE COUNTER TEST FAILED/
003030' 052131  020105  047503
003036' 047125  042524  020122
003044' 042524  052123  043040
003052' 044501  042514  000104
003060' 044514  045516  047440  ODDBYT::.ASCIZ  /LINK ODD BYTE COUNTER TEST FAILED/
003066' 042104  041040  052131
003074' 020105  047503  047125
003102' 042524  020122  042524
003110' 052123  043040  044501
003116' 042514  000104
003122' 044514  045516  046440  MAXCNT::.ASCIZ  /LINK MAXIMUM BYTE COUNT TEST FAILED/
003130' 054101  046511  046525
003136' 041040  052131  020105
003144' 047503  047125  020124
003152' 042524  052123  043040
003160' 044501  042514  000104
003166' 044506  047506  052040  FIFTST::.ASCIZ  /FIFO TEST FAILED/
003174' 051505  020124  040506
003202' 046111  042105     000
003207'    122  041505  044505  RLNKAD::.ASCIZ  /RECEIVER LINK MEMORY ADDRESS TEST FAILED/
```

```
  2192  003214' 042526  020122  044514
  2193  003222' 045516  046440  046505
  2194  003230' 051117  020131  042101
  2195  003236' 051104  051505  020123
  2196  003244' 042524  052123  043040
  2197  003252' 044501  042514  000104
  2198  003260' 051124  047101  046523  TLNKAD::.ASCIZ  /TRANSMITTER LINK MEMORY ADDRESS TEST FAILED/
  2199  003266' 052111  042524  020122
  2200  003274' 044514  045516  046440
  2201  003302' 046505  051117  020131
  2202  003310' 042101  051104  051505
  2203  003316' 020123  042524  052123
  2204  003324' 043040  044501  042514
  2205  003332' 000104
  2206  003334' 044514  045516  046440  LNKARB::.ASCIZ  /LINK MEMORY ARBITRATION TEST FAILED/
  2207  003342' 046505  051117  020131
  2208  003350' 051101  044502  051124
  2209  003356' 052101  047511  020116
  2210  003364' 042524  052123  043040
  2211  003372' 044501  042514  000104
  2212  003400' 052123  052101  047511  STAPAT::.ASCIZ  /STATION ADDRESS PATTERN TEST FAILED/
  2213  003406' 020116  042101  051104
  2214  003414' 051505  020123  040520
  2215  003422' 052124  051105  020116
  2216  003430' 042524  052123  043040
  2217  003436' 044501  042514  000104
  2218  003444' 052123  052101  047511  STAREJ::.ASCIZ  /STATION ADDRESS REJECTION TEST FAILED/
  2219  003452' 020116  042101  051104
  2220  003460' 051505  020123  042522
  2221  003466' 042512  052103  047511
  2222  003474' 020116  042524  052123
  2223  003502' 043040  044501  042514
  2224  003510' 000104
  2225  003512' 052123  052101  047511  STAPOS::.ASCIZ  /STATION ADDRESS RAM POSITION TEST FAILED/
  2226  003520' 020116  042101  051104
  2227  003526' 051505  020123  040522
  2228  003534' 020115  047520  044523
  2229  003542' 044524  047117  052040
  2230  003550' 051505  020124  040506
  2231  003556' 046111  042105     000
  2232  003563'    115  046125  044524  MUCAST::.ASCIZ  /MULTICAST ADDRESS TEST FAILED/
  2233  003570' 040503  052123  040440
  2234  003576' 042104  042522  051523
  2235  003604' 052040  051505  020124
  2236  003612' 040506  046111  042105
  2237  003620'    000
  2238  003621'    103  041522  042040  CRCDAT::.ASCIZ  /CRC DATA PATTERN TEST FAILED/
  2239  003626' 052101  020101  040520
  2240  003634' 052124  051105  020116
  2241  003642' 042524  052123  043040
  2242  003650' 044501  042514  000104
  2243  003656' 051103  020103  051105  CRCERR::.ASCIZ  /CRC ERROR TEST FAILED/
  2244  003664' 047522  020122  042524
  2245  003672' 052123  043040  044501
  2246  003700' 042514  000104
  2247  003704' 051103  020103  040520  CRCPAT::.ASCIZ  /CRC PATTERN LENGTH TEST FAILED/
```

```
 2248  003712' 052124  051105  020116
 2249  003720' 042514  043516  044124
 2250  003726' 052040  051505  020124
 2251  003734' 040506  046111  042105
 2252  003742'    000
 2253  003743'    122  041505  044505  RBRRUN::.ASC    /RECEIVE BUFFER RECOVERY - RUNT TEST FAILED/
 2254  003750' 042526  041040  043125
 2255  003756' 042506  020122  042522
 2256  003764' 047503  042526  054522
 2257  003772' 026440  051040  047125
 2258  004000' 020124  042524  052123
 2259  004006' 043040  044501  042514
 2260  004014' 000104
 2261  004016' 040510  043114  042055  HAFDUP::.ASCIZ  /HALF-DUPLEX TEST FAILED/
 2262  004024' 050125  042514  020130
 2263  004032' 042524  052123  043040
 2264  004040' 044501  042514  000104
 2265  004046' 047503  046114  051511  COLTST::.ASCIZ  /COLLISION TEST FAILED/
 2266  004054' 047511  020116  042524
 2267  004062' 052123  043040  044501
 2268  004070' 042514  000104
 2269  004074' 042124  020122  047503  TDRCNT::.ASCIZ  /TDR COUNTER TEST FAILED/
 2270  004102' 047125  042524  020122
 2271  004110' 042524  052123  043040
 2272  004116' 044501  042514  000104
 2273  004124' 042522  051124  020131  RETLOG::.ASCIZ  /RETRY LOGIC TEST FAILED/
 2274  004132' 047514  044507  020103
 2275  004140' 042524  052123  043040
 2276  004146' 044501  042514  000104
 2277  004154' 047125  041101  042514  PRTPAR::.ASCIZ  /UNABLE TO PRINT DEVICE PARAMETERS FOR THIS UNIT/
 2278  004162' 052040  020117  051120
 2279  004170' 047111  020124  042504
 2280  004176' 044526  042503  050040
 2281  004204' 051101  046501  052105
 2282  004212' 051105  020123  047506
 2283  004220' 020122  044124  051511
 2284  004226' 052440  044516  000124
 2285  004234' 040445  041520  051123  FORM1:: .ASCIZ  /%APCSR%D1%A DOES NOT EXIST%N/
 2286  004242' 042045  022461  020101
 2287  004250' 047504  051505  047040
 2288  004256' 052117  042440  044530
 2289  004264' 052123  047045     000
 2290  004271'    045  050101  051503  FORM2:: .ASCIZ  /%APCSR%D1%A BIT %Z2%A IS %T%N/
 2291  004276' 022522  030504  040445
 2292  004304' 041040  052111  022440
 2293  004312' 031132  040445  044440
 2294  004320' 020123  052045  047045
 2295  004326'    000
 2296  004327'    045  050101  051503  FORM3:: .ASCIZ  /%APCSR%D1%A FAILED DATA PATTERN TEST%N/
 2297  004334' 022522  030504  040445
 2298  004342' 043040  044501  042514
 2299  004350' 020104  040504  040524
 2300  004356' 050040  052101  042524
 2301  004364' 047122  052040  051505
 2302  004372' 022524  000116
 2303  004376' 040445  040504  040524  FORM4:: .ASCIZ  /%ADATA WRITTEN: %06%A DATA READ: %06%N/
```

```
  2304  004404' 053440  044522  052124
  2305  004412' 047105  020072  047445
  2306  004420' 022466  020101  040504
  2307  004426' 040524  051040  040505
  2308  004434' 035104  022440  033117
  2309  004442' 047045     000
  2310  004445'    045  041501  047101  FORM5:: .ASCIZ  /%ACANNOT CLEAR INTE BIT IN PCSR0%N/
  2311  004452' 047516  020124  046103
  2312  004460' 040505  020122  047111
  2313  004466' 042524  041040  052111
  2314  004474' 044440  020116  041520
  2315  004502' 051123  022460  000116
  2316  004510' 052045  051445  022461  FORM6:: .ASCIZ  /%T%S1%T%S1%T%S1%T%A PORT COMMAND WAS ISSUED%N/
  2317  004516' 022524  030523  052045
  2318  004524' 051445  022461  022524
  2319  004532' 020101  047520  052122
  2320  004540' 041440  046517  040515
  2321  004546' 042116  053440  051501
  2322  004554' 044440  051523  042525
  2323  004562' 022504  000116
  2324  004566' 040445  051103  020103  FORM7:: .ASCIZ  /%ACRC CHECK ERROR ON DATA DUMPED FROM DEUNA ROM%N/
  2325  004574' 044103  041505  020113
  2326  004602' 051105  047522  020122
  2327  004610' 047117  042040  052101
  2328  004616' 020101  052504  050115
  2329  004624' 042105  043040  047522
  2330  004632' 020115  042504  047125
  2331  004640' 020101  047522  022515
  2332  004646' 000116
  2333  004650' 040445  047104  020111  FORM8:: .ASCIZ  /%ADNI BIT DID NOT CLEAR AFTER WRITING A 1 TO IT%N/
  2334  004656' 044502  020124  044504
  2335  004664' 020104  047516  020124
  2336  004672' 046103  040505  020122
  2337  004700' 043101  042524  020122
  2338  004706' 051127  052111  047111
  2339  004714' 020107  020101  020061
  2340  004722' 047524  044440  022524
  2341  004730' 000116
  2342  004732' 040445  041520  051123  FORM9:: .ASCIZ  /%APCSR%D1%S1%T%S1%T%S1%T%A RESET WAS ISSUED%N/
  2343  004740' 042045  022461  030523
  2344  004746' 052045  051445  022461
  2345  004754' 022524  030523  052045
  2346  004762' 040445  051040  051505
  2347  004770' 052105  053440  051501
  2348  004776' 044440  051523  042525
  2349  005004' 022504  000116
  2350  005010' 040445  042504  047125  FORM10::.ASCIZ  /%ADEUNA NOT IN READY STATE AFTER RESET WAS ISSUED%N/
  2351  005016' 020101  047516  020124
  2352  005024' 047111  051040  040505
  2353  005032' 054504  051440  040524
  2354  005040' 042524  040440  052106
  2355  005046' 051105  051040  051505
  2356  005054' 052105  053440  051501
  2357  005062' 044440  051523  042525
  2358  005070' 022504  000116
  2359  005074' 040445  040504  040524  FORM11::.ASCIZ  /%ADATA COMPARE ERROR%N/
```

```
 2360  005102' 041440  046517  040520
 2361  005110' 042522  042440  051122
 2362  005116' 051117  047045     000
 2363  005123'    045  041101  043125  FORM12::.ASCIZ  /%ABUFFER ADDRESS: %06%S1%ADATA SB: %06%S1%ADATA WAS: %06%N/
 2364  005130' 042506  020122  042101
 2365  005136' 051104  051505  035123
 2366  005144' 022440  033117  051445
 2367  005152' 022461  042101  052101
 2368  005160' 020101  041123  020072
 2369  005166' 047445  022466  030523
 2370  005174' 040445  040504  040524
 2371  005202' 053440  051501  020072
 2372  005210' 047445  022466  000116
 2373  005216' 040445  033061  041040  FORM13::.ASCIZ  /%A16 BIT CRC SHOULD BE: 0%S2%AWAS: %06%N/
 2374  005224' 052111  041440  041522
 2375  005232' 051440  047510  046125
 2376  005240' 020104  042502  020072
 2377  005246' 022460  031123  040445
 2378  005254' 040527  035123  022440
 2379  005262' 033117  047045     000
 2380  005267'    045  042101  052101  FORM15::.ASCIZ  /%ADATA WRITTEN TO PCSR1 FROM T11 WAS: %06%N/
 2381  005274' 020101  051127  052111
 2382  005302' 042524  020116  047524
 2383  005310' 050040  051503  030522
 2384  005316' 043040  047522  020115
 2385  005324' 030524  020061  040527
 2386  005332' 035123  022440  033117
 2387  005340' 047045     000
 2388  005343'    045  042101  052101  FORM16::.ASCIZ  /%ADATA READ FROM PCSR1 FROM UNIBUS WAS: %06%N/
 2389  005350' 020101  042522  042101
 2390  005356' 043040  047522  020115
 2391  005364' 041520  051123  020061
 2392  005372' 051106  046517  052440
 2393  005400' 044516  052502  020123
 2394  005406' 040527  035123  022440
 2395  005414' 033117  047045     000
 2396  005421'    045  047101  020117  FORM17::.ASCIZ  /%ANO INTERRUPT AFTER NOP PORT COMMAND WAS ISSUED%N/
 2397  005426' 047111  042524  051122
 2398  005434' 050125  020124  043101
 2399  005442' 042524  020122  047516
 2400  005450' 020120  047520  052122
 2401  005456' 041440  046517  040515
 2402  005464' 042116  053440  051501
 2403  005472' 044440  051523  042525
 2404  005500' 022504  000116
 2405  005504' 040445  047125  020101  FORM18::.ASCIZ  /%AUNA DID NOT INTERRUPT AT CORRECT PRIORITY%N/
 2406  005512' 044504  020104  047516
 2407  005520' 020124  047111  042524
 2408  005526' 051122  050125  020124
 2409  005534' 052101  041440  051117
 2410  005542' 042522  052103  050040
 2411  005550' 044522  051117  052111
 2412  005556' 022531  000116
 2413  005562' 040445  047516  044440  FORM19::.ASCIZ  /%ANO INTERRUPT AFTER T11 SET %I%A IN PCSR0%N/
 2414  005570' 052116  051105  052522
 2415  005576' 052120  040440  052106
```

```
005604' 051105  052040  030461
005612' 051440  052105  022440
005620' 022524  020101  047111
005626' 050040  051503  030122
005634' 047045     000
005637'    045  052101  046511  FORM20::.ASCIZ  /%ATIMER DID NOT INTERRUPT T11%N/
005644' 051105  042040  042111
005652' 047040  052117  044440
005660' 052116  051105  052522
005666' 052120  052040  030461
005674' 047045     000
005677'    045  052101  046511  FORM21::.ASCIZ  /%ATIMER DID NOT INTERRUPT WHEN EXPECTED%N/
005704' 051105  042040  042111
005712' 047040  052117  044440
005720' 052116  051105  052522
005726' 052120  053440  042510
005734' 020116  054105  042520
005742' 052103  042105  047045
005750'    000
005751'    045  042501  050130  FORM22::.ASCIZ  /%AEXPECTED INTERRUPT BETWEEN 8 AND 12 SECONDS%N/
005756' 041505  042524  020104
005764' 047111  042524  051122
005772' 050125  020124  042502
006000' 053524  042505  020116
006006' 020070  047101  020104
006014' 031061  051440  041505
006022' 047117  051504  047045
006030'    000
006031'    045  051101  041505  FORM23::.ASCIZ  /%ARECIEVED INTERRUPT AT %D1%A SECONDS%N/
006036' 042511  042526  020104
006044' 047111  042524  051122
006052' 050125  020124  052101
006060' 022440  030504  040445
006066' 051440  041505  047117
006074' 051504  047045     000
006101'    045  046501  041511  FORM24::.ASCIZ  /%AMICRO TEST %D2%A HUNG%N/
006106' 047522  052040  051505
006114' 020124  042045  022462
006122' 020101  052510  043516
006130' 047045     000
006133'    045  042101  052101  FORM25::.ASCIZ  /%ADATA WRITTEN TO 'DMATO' REGISTER = %06%N/
006140' 020101  051127  052111
006146' 042524  020116  047524
006154' 023440  046504  052101
006162' 023517  051040  043505
006170' 051511  042524  020122
006176' 020075  047445  022466
006204' 000116
006206' 040445  040504  040524  FORM26::.ASCIZ  /%ADATA READ FROM 'DMATO' REGISTER  = %06%N/
006214' 051040  040505  020104
006222' 051106  046517  023440
006230' 046504  052101  023517
006236' 051040  043505  051511
006244' 042524  020122  036440
006252' 022440  033117  047045
006260'    000
```

```
 2472  006261'    045  042101  052101  FORM27:::.ASCIZ  /%ADATA SHOULD BE: %O6%A DATA WAS: %O6%N/
 2473  006266' 020101  044123  052517
 2474  006274' 042114  041040  035105
 2475  006302' 022440  033117  040445
 2476  006310' 042040  052101  020101
 2477  006316' 040527  035123  022440
 2478  006324' 033117  047045     000
 2479  006331'    045  022524  000116  FORM28:::.ASCIZ  /%T%N/
 2480  006336' 040445  042101  051104  FORM29:::.ASCIZ  /%AADDRESS = %O6%S2/
 2481  006344' 051505  020123  020075
 2482  006352' 047445  022466  031123
 2483  006360'    000
 2484  006361'    045  051101  041505  FORM30:::.ASCIZ  /%ARECEIVER STATUS:%O6%N/
 2485  006366' 044505  042526  020122
 2486  006374' 052123  052101  051525
 2487  006402' 022472  033117  047045
 2488  006410'    000
 2489  006411'    045  041101  043125  FORM31:::.ASCIZ  /%ABUFFER OFFSET:%O6%N/
 2490  006416' 042506  020122  043117
 2491  006424' 051506  052105  022472
 2492  006432' 033117  047045     000
 2493  006437'    045  052101  040522  FORM32:::.ASCIZ  /%ATRANSMIT STATUS WORD %D1%S1%T%A NOT CLEAR AFTER TRANSMIT%N/
 2494  006444' 051516  044515  020124
 2495  006452' 052123  052101  051525
 2496  006460' 053440  051117  020104
 2497  006466' 042045  022461  030523
 2498  006474' 052045  040445  047040
 2499  006502' 052117  041440  042514
 2500  006510' 051101  040440  052106
 2501  006516' 051105  052040  040522
 2502  006524' 051516  044515  022524
 2503  006532' 000116
 2504  006534' 040445  051124  047101  FORM33:::.ASCIZ  /%ATRANSMIT BYTE COUNT = %D4%N/
 2505  006542' 046523  052111  041040
 2506  006550' 052131  020105  047503
 2507  006556' 047125  020124  020075
 2508  006564' 042045  022464  000116
 2509  006572' 040445  042522  042503  FORM34:::.ASCIZ  /%ARECEIVE BYTE COUNT SHOULD BE: %D4%S3%AWAS: %D4%N/
 2510  006600' 053111  020105  054502
 2511  006606' 042524  041440  052517
 2512  006614' 052116  051440  047510
 2513  006622' 046125  020104  042502
 2514  006630' 020072  042045  022464
 2515  006636' 031523  040445  040527
 2516  006644' 035123  022440  032104
 2517  006652' 047045     000
 2518  006655'    045  040501  052103  FORM35:::.ASCIZ  /%AACTUAL NUMBER OF BYTES RECEIVED SHOULD BE: %D4%A   WAS: %D4%N/
 2519  006662' 040525  020114  052516
 2520  006670' 041115  051105  047440
 2521  006676' 020106  054502  042524
 2522  006704' 020123  042522  042503
 2523  006712' 053111  042105  051440
 2524  006720' 047510  046125  020104
 2525  006726' 042502  020072  042045
 2526  006734' 022464  020101  020040
 2527  006742' 040527  035123  022440
 2528  006750' 032104  047045     000
 2529  006755'    045  046101  047111  FORM36::.ASCIZ  /%ALINK MEMORY ADDRESS = %06%N/
 2530  006762' 020113  042515  047515
 2531  006770' 054522  040440  042104
 2532  006776' 042522  051523  036440
 2533  007004' 022440  033117  047045
 2534  007012'    000
 2535  007013'    045  052101  040522  FORM37::.ASCII  /%ATRANSMIT STATUS INFORMATION INCORRECT AFTER LOOPBACK STEP /
 2536  007020' 051516  044515  020124
 2537  007026' 052123  052101  051525
 2538  007034' 044440  043116  051117
 2539  007042' 040515  044524  047117
 2540  007050' 044440  041516  051117
 2541  007056' 042522  052103  040440
 2542  007064' 052106  051105  046040
 2543  007072' 047517  041120  041501
 2544  007100' 020113  052123  050105
 2545  007106'    040
 2546  007107'    116  046525  042502          .ASCIZ  /NUMBER %D2%N/
 2547  007114' 020122  042045  022462
 2548  007122' 000116
 2549  007124' 040445  051124  047101  FORM38::.ASCIZ  /%ATRANSMIT STATUS WORD %D1%A SHOULD BE: %06%A WAS: %06%N/
 2550  007132' 046523  052111  051440
 2551  007140' 040524  052524  020123
 2552  007146' 047527  042122  022440
 2553  007154' 030504  040445  051440
 2554  007162' 047510  046125  020104
 2555  007170' 042502  020072  047445
 2556  007176' 022466  020101  040527
 2557  007204' 035123  022440  033117
 2558  007212' 047045     000
 2559  007215'    045  052101  051104  FORM39::.ASCIZ  /%ATDR COUNTER NOT INCREMENTING%N/
 2560  007222' 041440  052517  052116
 2561  007230' 051105  047040  052117
 2562  007236' 044440  041516  042522
 2563  007244' 042515  052116  047111
 2564  007252' 022507  000116
 2565  007256' 040445  042522  051124  FORM40::.ASCIZ  /%ARETRY BACKOFF WAIT TIME INTERVAL NOT VARYING%N%N/
 2566  007264' 020131  040502  045503
 2567  007272' 043117  020106  040527
 2568  007300' 052111  052040  046511
 2569  007306' 020105  047111  042524
 2570  007314' 053122  046101  047040
 2571  007322' 052117  053040  051101
 2572  007330' 044531  043516  047045
 2573  007336' 047045     000
 2574  007341'    045  051101  052105  FORM41::.ASCIZ  /%ARETRY #%S4%AWAIT INTERVAL COUNT%N%N/
 2575  007346' 054522  021440  051445
 2576  007354' 022464  053501  044501
 2577  007362' 020124  047111  042524
 2578  007370' 053122  046101  041440
 2579  007376' 052517  052116  047045
 2580  007404' 047045     000
 2581  007407'    045  032504  051445  FORM42::.ASCIZ  /%D5%S7/
 2582  007414' 000067
 2583  007416' 042045  022465  000116  FORM43::.ASCIZ  /%D5%N/
```

```
007424' 040445  044124  020105  FORM44:::.ASCIZ  /%ATHE 48 BIT DESTINATION ADDRESS PATTERN IS:%N/
007432' 034064  041040  052111
007440' 042040  051505  044524
007446' 040516  044524  047117
007454' 040440  042104  042522
007462' 051523  050040  052101
007470' 042524  047122  044440
007476' 035123  047045     000
007503'    045  046101  053517  FORM45:::.ASCIZ  /%ALOWER ORDER = %06%N/
007510' 051105  047440  042122
007516' 051105  036440  022440
007524' 033117  047045     000
007531'    045  046501  042111  FORM46:::.ASCIZ  /%AMIDDLE ORDER = %06%N/
007536' 046104  020105  051117
007544' 042504  020122  020075
007552' 047445  022466  000116
007560' 040445  050125  042520  FORM47:::.ASCIZ  /%AUPPER ORDER  = %06%N/
007566' 020122  051117  042504
007574' 020122  036440  022440
007602' 033117  047045     000
007607'    045  042101  052505  FORM48:::.ASCIZ  /%ADEUNA FAILED TO RECOGNIZE A STATION ADDRESS THAT MATCHES%N/
007614' 040516  043040  044501
007622' 042514  020104  047524
007630' 051040  041505  043517
007636' 044516  042532  040440
007644' 051440  040524  044524
007652' 047117  040440  042104
007660' 042522  051523  052040
007666' 040510  020124  040515
007674' 041524  042510  022523
007702' 000116
007704' 040445  044124  020105  FORM49:::.ASCIZ  /%ATHE STATION ADDRESS IN RAM POSITION %D2%N/
007712' 052123  052101  047511
007720' 020116  042101  051104
007726' 051505  020123  047111
007734' 051040  046501  050040
007742' 051517  052111  047511
007750' 020116  042045  022462
007756' 000116
007760' 040445  042522  042503  FORM50:::.ASCIZ  /%ARECEIVER STATUS WORD 0 SHOULD BE: %06%S3%AWAS: %06%N/
007766' 053111  051105  051440
007774' 040524  052524  020123
010002' 047527  042122  030040
010010' 051440  047510  046125
010016' 020104  042502  020072
010024' 047445  022466  031523
010032' 040445  040527  035123
010040' 022440  033117  047045
010046'    000
010047'    045  041501  041522  FORM51:::.ASCIZ  /%ACRC ERROR BIT NOT SET%N/
010054' 042440  051122  051117
010062' 041040  052111  047040
010070' 052117  051440  052105
010076' 047045     000
010101'    045  042501  051122  FORM52:::.ASCIZ  /%AERROR SUMMARY BIT NOT SET%N/
010106' 051117  051440  046525
```

```
 2640  010114' 040515  054522  041040
 2641  010122' 052111  047040  052117
 2642  010130' 051440  052105  047045
 2643  010136'    000
 2644  010137'    045  047101  046525  FORM53::.ASCIZ  /%ANUMBER OF DATA BYTES TRANSMITTED: %D4%N/
 2645  010144' 042502  020122  043117
 2646  010152' 042040  052101  020101
 2647  010160' 054502  042524  020123
 2648  010166' 051124  047101  046523
 2649  010174' 052111  042524  035104
 2650  010202' 022440  032104  047045
 2651  010210'    000
 2652  010211'    045  042101  052101  FORM54::.ASCIZ  /%ADATA PATTERN: %O6%N/
 2653  010216' 020101  040520  052124
 2654  010224' 051105  035116  022440
 2655  010232' 033117  047045     000
 2656  010237'    045  042101  052505  FORM55::.ASCIZ  /%ADEUNA FAILED TO REJECT A RUNT PACKET%N/
 2657  010244' 040516  043040  044501
 2658  010252' 042514  020104  047524
 2659  010260' 051040  045105  041505
 2660  010266' 020124  020101  052522
 2661  010274' 052116  050040  041501
 2662  010302' 042513  022524  000116
 2663  010310' 040445  042504  047125  FORM56::.ASCIZ  /%ADEUNA FAILED TO RECOVER RECEIVE BUFFER%N/
 2664  010316' 020101  040506  046111
 2665  010324' 042105  052040  020117
 2666  010332' 042522  047503  042526
 2667  010340' 020122  042522  042503
 2668  010346' 053111  020105  052502
 2669  010354' 043106  051105  047045
 2670  010362'    000
 2671  010363'    045  047101  046525  FORM57::.ASCIZ  /%ANUMBER OF BYTES IN RUNT PACKET: %D2%N/
 2672  010370' 042502  020122  043117
 2673  010376' 041040  052131  051505
 2674  010404' 044440  020116  052522
 2675  010412' 052116  050040  041501
 2676  010420' 042513  035124  022440
 2677  010426' 031104  047045     000
 2678  010433'    045  047101  046525  FORM58::.ASCIZ  /%ANUMBER OF BYTES IN LEGITIMATE PACKET: %D4%N/
 2679  010440' 042502  020122  043117
 2680  010446' 041040  052131  051505
 2681  010454' 044440  020116  042514
 2682  010462' 044507  044524  040515
 2683  010470' 042524  050040  041501
 2684  010476' 042513  035124  022440
 2685  010504' 032104  047045     000
 2686  010511'    045  051101  041505  FORM59::.ASCIZ  /%ARECEIVE BUFFER ADDRESS AVAILABLE BEFORE RECEPTION: %O6%N/
 2687  010516' 044505  042526  041040
 2688  010524' 043125  042506  020122
 2689  010532' 042101  051104  051505
 2690  010540' 020123  053101  044501
 2691  010546' 040514  046102  020105
 2692  010554' 042502  047506  042522
 2693  010562' 051040  041505  050105
 2694  010570' 044524  047117  020072
 2695  010576' 047445  022466  000116
```

```
010604' 040445  042522  042503  FORM60::.ASCIZ  /%ARECEIVE BUFFER ADDRESS COMPLETED AFTER RECEPTION%N/
010612' 053111  020105  052502
010620' 043106  051105  040440
010626' 042104  042522  051523
010634' 041440  046517  046120
010642' 052105  042105  040440
010650' 052106  051105  051040
010656' 041505  050105  044524
010664' 047117  047045     000
010671'    045  042101  052101  FORM61::.ASCIZ  /%ADATA COMPARE ERROR IN RECOVERED RECEIVE BUFFER%N/
010676' 020101  047503  050115
010704' 051101  020105  051105
010712' 047522  020122  047111
010720' 051040  041505  053117
010726' 051105  042105  051040
010734' 041505  044505  042526
010742' 041040  043125  042506
010750' 022522  000116
010754' 052045  040445  042040  FORM62::.ASCIZ  /%T%A DID NOT CLEAR AFTER WRITING 1 TO IT%N/
010762' 042111  047040  052117
010770' 041440  042514  051101
010776' 040440  052106  051105
011004' 053440  044522  044524
011012' 043516  030440  052040
011020' 020117  052111  047045
011026'    000
011027'    045  051501  046105  FORM63::.ASCIZ  /%ASELF TEST ERROR CODE = %O2%N/
011034' 020106  042524  052123
011042' 042440  051122  051117
011050' 041440  042117  020105
011056' 020075  047445  022462
011064' 000116
011066' 047045  040445  047522  FORM64::.ASCIZ  /%N%AROM MICROCODE VERSION (DECIMAL): %D2/
011074' 020115  044515  051103
011102' 041517  042117  020105
011110' 042526  051522  047511
011116' 020116  042050  041505
011124' 046511  046101  035051
011132' 022440  031104     000
011137'    045  022516  051501  FORM65::.ASCIZ  /%N%ASWITCH PACK = %O6/
011144' 044527  041524  020110
011152' 040520  045503  036440
011160' 022440  033117     000
011165'    045  047101  020117  FORM66::.ASCIZ  /%ANO INTERRUPT FROM TRANSMIT STATE MACHINE TO T-11%N/
011172' 047111  042524  051122
011200' 050125  020124  051106
011206' 046517  052040  040522
011214' 051516  044515  020124
011222' 052123  052101  020105
011230' 040515  044103  047111
011236' 020105  047524  052040
011244' 030455  022461  000116
011252' 040445  047516  044440  FORM67::.ASCIZ  /%ANO INTERRUPT FROM RECEIVE STATE MACHINE TO T-11%N/
011260' 052116  051105  052522
011266' 052120  043040  047522
011274' 020115  042522  042503
```

```
  2752  011302' 053111  020105  052123
  2753  011310' 052101  020105  040515
  2754  011316' 044103  047111  020105
  2755  011324' 047524  052040  030455
  2756  011332' 022461  000116
  2757  011336' 040445  042522  042503  FORM68::.ASCIZ  /%ARECEIVER BYTE COUNTER FAILED TO LOCK UP AT MAXIMUM VALUE%N/
  2758  011344' 053111  051105  041040
  2759  011352' 052131  020105  047503
  2760  011360' 047125  042524  020122
  2761  011366' 040506  046111  042105
  2762  011374' 052040  020117  047514
  2763  011402' 045503  052440  020120
  2764  011410' 052101  046440  054101
  2765  011416' 046511  046525  053040
  2766  011424' 046101  042525  047045
  2767  011432'    000
  2768  011433'    045  042101  052505  FORM69::.ASCII  /%ADEUNA DID NOT REJECT A PACKET TRANSMITTED TO ITSELF/
  2769  011440' 040516  042040  042111
  2770  011446' 047040  052117  051040
  2771  011454' 045105  041505  020124
  2772  011462' 020101  040520  045503
  2773  011470' 052105  052040  040522
  2774  011476' 051516  044515  052124
  2775  011504' 042105  052040  020117
  2776  011512' 052111  042523  043114
  2777  011520' 040445  044440  020116          .ASCIZ  /%A IN HALF-DUPLEX MODE%N/
  2778  011526' 040510  043114  042055
  2779  011534' 050125  042514  020130
  2780  011542' 047515  042504  047045
  2781  011550'    000
  2782  011551'    045  052101  040522  FORM70::.ASCIZ  /%ATRANSMIT BUFFER ADDRESS = %06%N/
  2783  011556' 051516  044515  020124
  2784  011564' 052502  043106  051105
  2785  011572' 040440  042104  042522
  2786  011600' 051523  036440  022440
  2787  011606' 033117  047045     000
  2788  011613'    045  051101  041505  FORM71::.ASCIZ  /%ARECEIVE BUFFER ADDRESS  = %06%N/
  2789  011620' 044505  042526  041040
  2790  011626' 043125  042506  020122
  2791  011634' 042101  051104  051505
  2792  011642' 020123  036440  022440
  2793  011650' 033117  047045     000
  2794  011655'    045  052101  030461  FORM72::.ASCIZ  /%AT11 LINK MEMORY PARITY ERROR OCCURRED%N/
  2795  011662' 046040  047111  020113
  2796  011670' 042515  047515  054522
  2797  011676' 050040  051101  052111
  2798  011704' 020131  051105  047522
  2799  011712' 020122  041517  052503
  2800  011720' 051122  042105  047045
  2801  011726'    000
  2802  011727'    045  052101  030461  FORM73::.ASCIZ  /%AT11 NPR TIMEOUT ERROR OCCURRED%N/
  2803  011734' 047040  051120  052040
  2804  011742' 046511  047505  052125
  2805  011750' 042440  051122  051117
  2806  011756' 047440  041503  051125
  2807  011764' 042522  022504  000116
```

```
012772' 040445  030524  020061  FORM74::.ASCIZ  /%AT11 NON-EXISTANT MEMORY TIMEOUT OCCURRED%N/
012000' 047516  026516  054105
012006' 051511  040524  052116
012014' 046440  046505  051117
012022' 020131  044524  042515
012030' 052517  020124  041517
012036' 052503  051122  042105
012044' 047045     000
012047'    045  052101  030461  FORM75::.ASCIZ  /%AT11 UNEXPECTED INTERRUPT OCCURRED%N/
012054' 052440  042516  050130
012062' 041505  042524  020104
012070' 047111  042524  051122
012076' 050125  020124  041517
012104' 052503  051122  042105
012112' 047045     000
012115'    045  046501  052101  FORM76::.ASCIZ  /%AMATCH BIT FAILED TO SET%N/
012122' 044103  041040  052111
012130' 043040  044501  042514
012136' 020104  047524  051440
012144' 052105  047045     000
012151'    045  046501  052101  FORM77::.ASCIZ  /%AMATCH BIT SET BUT NO RECEIVER INTERRUPT%N/
012156' 044103  041040  052111
012164' 051440  052105  041040
012172' 052125  047040  020117
012200' 042522  042503  053111
012206' 051105  044440  052116
012214' 051105  052522  052120
012222' 047045     000
012225'    045  051501  047510  FORM78::.ASCIZ  /%ASHOULD BE: %06%S2%AWAS: %06%N/
012232' 046125  020104  042502
012240' 020072  047445  022466
012246' 031123  040445  040527
012254' 035123  022440  033117
012262' 047045     000
012265'    045  042101  052505  FORM79::.ASCIZ  /%ADEUNA FAILED TO REJECT A DATAGRAM.%N/
012272' 040516  043040  044501
012300' 042514  020104  047524
012306' 051040  045105  041505
012314' 020124  020101  040504
012322' 040524  051107  046501
012330' 022456  000116
012334' 040445  042504  052123  FORM80::.ASCIZ  /%ADESTINATION ADDRESS OF DATAGRAM IS ALL 1'S%N/
012342' 047111  052101  047511
012350' 020116  042101  051104
012356' 051505  020123  043117
012364' 042040  052101  043501
012372' 040522  020115  051511
012400' 040440  046114  030440
012406' 051447  047045     000
012413'    045  051501  040524  FORM81::.ASCIZ  /%ASTATION ADDRESS RAM IS ALL 0'S%N/
012420' 044524  047117  040440
012426' 042104  042522  051523
012434' 051040  046501  044440
012442' 020123  046101  020114
012450' 023460  022523  000116
012456' 040445  044524  042515  FORM82::.ASCIZ  /%ATIMEOUT WAITING FOR MICROCODE TO ENTER MICROMONITOR%N/
```

```
  2864  012464' 052517  020124  040527
  2865  012472' 052111  047111  020107
  2866  012500' 047506  020122  044515
  2867  012506' 051103  041517  042117
  2868  012514' 020105  047524  042440
  2869  012522' 052116  051105  046440
  2870  012530' 041511  047522  047515
  2871  012536' 044516  047524  022522
  2872  012544' 000116
  2873                                          .EVEN
  2874
  2875
```

```
                                  .SBTTL  GLOBAL ERROR REPORT SECTION

                                  ;++
                                  ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
                                  ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
                                  ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
                                  ;--




012546'                           BGNMSG  RACMG1
012546'                                                                       RACMG1::
012546'                                   PRINTB  #FORM1,CSRNUM
012546' 013746  000304'                                                               MOV     CSRNUM,-(SP)
012552' 012746  004234'                                                               MOV     #FORM1,-(SP)
012556' 012746  000002                                                                MOV     #2,-(SP)
012562' 010600                                                                        MOV     SP,R0
012564' 104414                                                                        TRAP    C$PNTB
012566' 062706  000006                                                                ADD     #6,SP
012572'                           ENDMSG
012572'                                                                       L10001:
012572' 104423                                                                        TRAP    C$MSG

012574'                           BGNMSG  RACMG2
012574'                                                                       RACMG2::
012574'                                   PRINTB  #FORM2,CSRNUM,BITNUM,BITSTA
012574' 013746  000312'                                                               MOV     BITSTA,-(SP)
012600' 013746  000306'                                                               MOV     BITNUM,-(SP)
012604' 013746  000304'                                                               MOV     CSRNUM,-(SP)
012610' 012746  004271'                                                               MOV     #FORM2,-(SP)
012614' 012746  000004                                                                MOV     #4,-(SP)
012620' 010600                                                                        MOV     SP,R0
012622' 104414                                                                        TRAP    C$PNTB
012624' 062706  000012                                                                ADD     #12,SP
012630'                           ENDMSG
012630'                                                                       L10002:
012630' 104423                                                                        TRAP    C$MSG

012632'                           BGNMSG  RACMG3
012632'                                                                       RACMG3::
012632'                                   PRINTB  #FORM3,CSRNUM
012632' 013746  000304'                                                               MOV     CSRNUM,-(SP)
012636' 012746  004327'                                                               MOV     #FORM3,-(SP)
012642' 012746  000002                                                                MOV     #2,-(SP)
012646' 010600                                                                        MOV     SP,R0
012650' 104414                                                                        TRAP    C$PNTB
012652' 062706  000006                                                                ADD     #6,SP
012656'                                   PRINTB  #FORM4,R3,R4
012656' 010446                                                                        MOV     R4,-(SP)
012660' 010346                                                                        MOV     R3,-(SP)
012662' 012746  004376'                                                               MOV     #FORM4,-(SP)
012666' 012746  000003                                                                MOV     #3,-(SP)
012672' 010600                                                                        MOV     SP,R0
012674' 104414                                                                        TRAP    C$PNTB
012676' 062706  000010                                                                ADD     #10,SP
```

```
012702'                          ENDMSG
012702'                                                                L10003:
012702' 104423                                                                 TRAP    C$MSG

012704'                          BGNMSG  RACMG4
012704'                                                                RACMG4::
012704'                                  PRINTB  #FORM5
012704' 012746  004445'                                                        MOV     #FORM5,-(SP)
012710' 012746  000001                                                         MOV     #1,-(SP)
012714' 010600                                                                 MOV     SP,R0
012716' 104414                                                                 TRAP    C$PNTB
012720' 062706  000004                                                         ADD     #4,SP
012724'                          ENDMSG
012724'                                                                L10004:
012724' 104423                                                                 TRAP    C$MSG

012726'                          BGNMSG  RACMG7
012726'                                                                RACMG7::
012726'                                  PRINTB  #FORM8
012726' 012746  004650'                                                        MOV     #FORM8,-(SP)
012732' 012746  000001                                                         MOV     #1,-(SP)
012736' 010600                                                                 MOV     SP,R0
012740' 104414                                                                 TRAP    C$PNTB
012742' 062706  000004                                                         ADD     #4,SP
012746'                          ENDMSG
012746'                                                                L10005:
012746' 104423                                                                 TRAP    C$MSG

012750'                          BGNMSG  MSG1
012750'                                                                MSG1::
012750'                                  PRINTB  #FORM6,BITNAM,BITSTA,PWHEN,PCOMND
012750' 013746  000316'                                                        MOV     PCOMND,-(SP)
012754' 013746  000314'                                                        MOV     PWHEN,-(SP)
012760' 013746  000312'                                                        MOV     BITSTA,-(SP)
012764' 013746  000310'                                                        MOV     BITNAM,-(SP)
012770' 012746  004510'                                                        MOV     #FORM6,-(SP)
012774' 012746  000005                                                         MOV     #5,-(SP)
013000' 010600                                                                 MOV     SP,R0
013002' 104414                                                                 TRAP    C$PNTB
013004' 062706  000014                                                         ADD     #14,SP
013010'                          ENDMSG
013010'                                                                L10006:
013010' 104423                                                                 TRAP    C$MSG

013012'                          BGNMSG  MSG2
013012'                                                                MSG2::
013012'                                  PRINTB  #FORM7
013012' 012746  004566'                                                        MOV     #FORM7,-(SP)
013016' 012746  000001                                                         MOV     #1,-(SP)
013022' 010600                                                                 MOV     SP,R0
013024' 104414                                                                 TRAP    C$PNTB
013026' 062706  000004                                                         ADD     #4,SP
013032'                                  PRINTB  #FORM13,R4
013032' 010446                                                                 MOV     R4,-(SP)
013034' 012746  005216'                                                        MOV     #FORM13,-(SP)
013040' 012746  000002                                                         MOV     #2,-(SP)
```

```
013044' 010600                                                                MOV     SP,R0
013046' 104414                                                                TRAP    C$PNTB
013050' 062706  000006                                                        ADD     #6,SP
013054'                   ENDMSG
013054'                                                               L10007:
013054' 104423                                                                TRAP    C$MSG

013056'                   BGNMSG  MSG3
013056'                                                               MSG3::
013056'                           PRINTB  #FORM9,CSRNUM,BITNAM,BITSTA,PWHEN
013056' 013746  000314'                                                       MOV     PWHEN,-(SP)
013062' 013746  000312'                                                       MOV     BITSTA,-(SP)
013066' 013746  000310'                                                       MOV     BITNAM,-(SP)
013072' 013746  000304'                                                       MOV     CSRNUM,-(SP)
013076' 012746  004732'                                                       MOV     #FORM9,-(SP)
013102' 012746  000005                                                        MOV     #5,-(SP)
013106' 010600                                                                MOV     SP,R0
013110' 104414                                                                TRAP    C$PNTB
013112' 062706  000014                                                        ADD     #14,SP
013116'                   ENDMSG
013116'                                                               L10010:
013116' 104423                                                                TRAP    C$MSG

013120'                   BGNMSG  MSG4
013120'                                                               MSG4::
013120'                           PRINTB  #FORM10
013120' 012746  005010'                                                       MOV     #FORM10,-(SP)
013124' 012746  000001                                                        MOV     #1,-(SP)
013130' 010600                                                                MOV     SP,R0
013132' 104414                                                                TRAP    C$PNTB
013134' 062706  000004                                                        ADD     #4,SP
013140'                   ENDMSG
013140'                                                               L10011:
013140' 104423                                                                TRAP    C$MSG

013142'                   BGNMSG  MSG5
013142'                                                               MSG5::
013142'                           PRINTB  #FORM11
013142' 012746  005074'                                                       MOV     #FORM11,-(SP)
013146' 012746  000001                                                        MOV     #1,-(SP)
013152' 010600                                                                MOV     SP,R0
013154' 104414                                                                TRAP    C$PNTB
013156' 062706  000004                                                        ADD     #4,SP
013162'                           PRINTB  #FORM12,R1,(R4),(R1)
013162' 011146                                                                MOV     (R1),-(SP)
013164' 011446                                                                MOV     (R4),-(SP)
013166' 010146                                                                MOV     R1,-(SP)
013170' 012746  005123'                                                       MOV     #FORM12,-(SP)
013174' 012746  000004                                                        MOV     #4,-(SP)
013200' 010600                                                                MOV     SP,R0
013202' 104414                                                                TRAP    C$PNTB
013204' 062706  000012                                                        ADD     #12,SP
013210'                   ENDMSG
013210'                                                               L10012:
013210' 104423                                                                TRAP    C$MSG
```

```
013212'                        BGNMSG  MSG6
013212'                                                        MSG6::
013212'                                PRINTB  #FORM11
013212' 012746  005074'                                                MOV     #FORM11,-(SP)
013216' 012746  000001                                                 MOV     #1,-(SP)
013222' 010600                                                         MOV     SP,R0
013224' 104414                                                         TRAP    C$PNTB
013226' 062706  000004                                                 ADD     #4,SP
013232'                                PRINTB  #FORM15,R1
013232' 010146                                                         MOV     R1,-(SP)
013234' 012746  005267'                                                MOV     #FORM15,-(SP)
013240' 012746  000002                                                 MOV     #2,-(SP)
013244' 010600                                                         MOV     SP,R0
013246' 104414                                                         TRAP    C$PNTB
013250' C62706  000006                                                 ADD     #6,SP
013254'                                PRINTB  #FORM16,R2
013254' 010246                                                         MOV     R2,-(SP)
013256' 012746  005343'                                                MOV     #FORM16,-(SP)
013262' 012746  000002                                                 MOV     #2,-(SP)
013266' 010600                                                         MOV     SP,R0
013270' 104414                                                         TRAP    C$PNTB
013272' 062706  000006                                                 ADD     #6,SP
013276'                        ENDMSG
013276'                                                        L10013:
013276' 104423                                                         TRAP    C$MSG

013300'                        BGNMSG  MSG7
013300'                                                        MSG7::
013300'                                PRINTB  #FORM17
013300' 012746  005421'                                                MOV     #FORM17,-(SP)
013304' 012746  000001                                                 MOV     #1,-(SP)
013310' 010600                                                         MOV     SP,R0
013312' 104414                                                         TRAP    C$PNTB
013314' 062706  000004                                                 ADD     #4,SP
013320'                        ENDMSG
013320'                                                        L10014:
013320' 104423                                                         TRAP    C$MSG

013322'                        BGNMSG  MSG8
013322'                                                        MSG8::
013322'                                PRINTB  #FORM18
013322' 012746  005504'                                                MOV     #FORM18,-(SP)
013326' 012746  000001                                                 MOV     #1,-(SP)
013332' 010600                                                         MOV     SP,R0
013334' 104414                                                         TRAP    C$PNTB
013336' 062706  000004                                                 ADD     #4,SP
013342'                        ENDMSG
013342'                                                        L10015:
013342' 104423                                                         TRAP    C$MSG

013344'                        BGNMSG  MSG9
013344'                                                        MSG9::
013344'                                PRINTB  #FORM19,@BITNAM
013344' 017746  164740                                                 MOV     @BITNAM,-(SP)
013350' 012746  005562'                                                MOV     #FORM19,-(SP)
013354' 012746  000002                                                 MOV     #2,-(SP)
```

```
013360' 010600                                                    MOV     SP,R0
013362' 104414                                                    TRAP    C$PNTB
013364' 062706  000006                                            ADD     #6,SP
013370'                   ENDMSG
013370'                                                 L10016:
013370' 104423                                                    TRAP    C$MSG

013372'                   BGNMSG  MSG10
013372'                                                 MSG10::
013372'                           PRINTB  #FORM20
013372' 012746  005637'                                           MOV     #FORM20,-(SP)
013376' 012746  000001                                            MOV     #1,-(SP)
013402' 010600                                                    MOV     SP,R0
013404' 104414                                                    TRAP    C$PNTB
013406' 062706  000004                                            ADD     #4,SP
013412'                   ENDMSG
013412'                                                 L10017:
013412' 104423                                                    TRAP    C$MSG

013414'                   BGNMSG  MSG11
013414'                                                 MSG11::
013414'                           PRINTB  #FORM21
013414' 012746  005677'                                           MOV     #FORM21,-(SP)
013420' 012746  000001                                            MOV     #1,-(SP)
013424' 010600                                                    MOV     SP,R0
013426' 104414                                                    TRAP    C$PNTB
013430' 062706  000004                                            ADD     #4,SP
013434'                           PRINTB  #FORM22
013434' 012746  005751'                                           MOV     #FORM22,-(SP)
013440' 012746  000001                                            MOV     #1,-(SP)
013444' 010600                                                    MOV     SP,R0
013446' 104414                                                    TRAP    C$PNTB
013450' 062706  000004                                            ADD     #4,SP
013454'                           PRINTB  #FORM23,R1
013454' 010146                                                    MOV     R1,-(SP)
013456' 012746  006031'                                           MOV     #FORM23,-(SP)
013462' 012746  000002                                            MOV     #2,-(SP)
013466' 010600                                                    MOV     SP,R0
013470' 104414                                                    TRAP    C$PNTB
013472' 062706  000006                                            ADD     #6,SP
013476'                   ENDMSG
013476'                                                 L10020:
013476' 104423                                                    TRAP    C$MSG

013500'                   BGNMSG  MSG12
013500'                                                 MSG12::
013500'                           PRINTB  #FORM24,R2
013500' 010246                                                    MOV     R2,-(SP)
013502' 012746  006101'                                           MOV     #FORM24,-(SP)
013506' 012746  000002                                            MOV     #2,-(SP)
013512' 010600                                                    MOV     SP,R0
013514' 104414                                                    TRAP    C$PNTB
013516' 062706  000006                                            ADD     #6,SP
013522'                   ENDMSG
013522'                                                 L10021:
013522' 104423                                                    TRAP    C$MSG
```

```

013524'                   BGNMSG  MSG13
013524'                                           MSG13::
013524'                           PRINTB  #FORM25,R3
013524' 010346                                            MOV     R3,-(SP)
013526' 012746  006133'                                   MOV     #FORM25,-(SP)
013532' 012746  000002                                    MOV     #2,-(SP)
013536' 010600                                            MOV     SP,R0
013540' 104414                                            TRAP    C$PNTB
013542' 062706  000006                                    ADD     #6,SP
013546'                           PRINTB  #FORM26,R4
013546' 010446                                            MOV     R4,-(SP)
013550' 012746  006206'                                   MOV     #FORM26,-(SP)
013554' 012746  000002                                    MOV     #2,-(SP)
013560' 010600                                            MOV     SP,R0
013562' 104414                                            TRAP    C$PNTB
013564' 062706  000006                                    ADD     #6,SP
013570'                   ENDMSG
013570'                                           L10022:
013570' 104423                                            TRAP    C$MSG

013572'                   BGNMSG  MSG14
013572'                                           MSG14::
013572'                           PRINTB  #FORM27,R3,R4
013572' 010446                                            MOV     R4,-(SP)
013574' 010346                                            MOV     R3,-(SP)
013576' 012746  006261'                                   MOV     #FORM27,-(SP)
013602' 012746  000003                                    MOV     #3,-(SP)
013606' 010600                                            MOV     SP,R0
013610' 104414                                            TRAP    C$PNTB
013612' 062706  000010                                    ADD     #10,SP
013616'                   ENDMSG
013616'                                           L10023:
013616' 104423                                            TRAP    C$MSG

013620'                   BGNMSG  MSG15
013620'                                           MSG15::
013620'                           PRINTB  #FORM12,R2,(R1),(R2)
013620' 011246                                            MOV     (R2),-(SP)
013622' 011146                                            MOV     (R1),-(SP)
013624' 010246                                            MOV     R2,-(SP)
013626' 012746  005123'                                   MOV     #FORM12,-(SP)
013632' 012746  000004                                    MOV     #4,-(SP)
013636' 010600                                            MOV     SP,R0
013640' 104414                                            TRAP    C$PNTB
013642' 062706  000012                                    ADD     #12,SP
013646'                   ENDMSG
013646'                                           L10024:
013646' 104423                                            TRAP    C$MSG

013650'                   BGNMSG  MSG16
013650'                                           MSG16::
013650'                           PRINTB  #FORM28,R2
013650' 010246                                            MOV     R2,-(SP)
013652' 012746  006331'                                   MOV     #FORM28,-(SP)
013656' 012746  000002                                    MOV     #2,-(SP)
```

```
013662' 010600                                                  MOV     SP,R0
013664' 104414                                                  TRAP    C$PNTB
013666' 062706  000006                                          ADD     #6,SP
013672'                         PRINTB  #FORM29,R1
013672' 010146                                                  MOV     R1,-(SP)
013674' 012746  006336'                                         MOV     #FORM29,-(SP)
013700' 012746  000002                                          MOV     #2,-(SP)
013704' 010600                                                  MOV     SP,R0
013706' 104414                                                  TRAP    C$PNTB
013710' 062706  000006                                          ADD     #6,SP
013714'                         PRINTB  #FORM27,R3,R4
013714' 010446                                                  MOV     R4,-(SP)
013716' 010346                                                  MOV     R3,-(SP)
013720' 012746  006261'                                         MOV     #FORM27,-(SP)
013724' 012746  000003                                          MOV     #3,-(SP)
013730' 010600                                                  MOV     SP,R0
013732' 104414                                                  TRAP    C$PNTB
013734' 062706  000010                                          ADD     #10,SP
013740'                 ENDMSG
013740'                                         L10025:
013740' 104423                                                  TRAP    C$MSG

013742'                 BGNMSG  MSG17
013742'                                         MSG17::
013742'                         PRINTB  #FORM30,R4
013742' 010446                                                  MOV     R4,-(SP)
013744' 012746  006361'                                         MOV     #FORM30,-(SP)
013750' 012746  000002                                          MOV     #2,-(SP)
013754' 010600                                                  MOV     SP,R0
013756' 104414                                                  TRAP    C$PNTB
013760' 062706  000006                                          ADD     #6,SP
013764'                         PRINTB  #FORM31,R1
013764' 010146                                                  MOV     R1,-(SP)
013766' 012746  006411'                                         MOV     #FORM31,-(SP)
013772' 012746  000002                                          MOV     #2,-(SP)
013776' 010600                                                  MOV     SP,R0
014000' 104414                                                  TRAP    C$PNTB
014002' 062706  000006                                          ADD     #6,SP
014006'                         PRINTB  #FORM27,R2,R3
014006' 010346                                                  MOV     R3,-(SP)
014010' 010246                                                  MOV     R2,-(SP)
014012' 012746  006261'                                         MOV     #FORM27,-(SP)
014016' 012746  000003                                          MOV     #3,-(SP)
014022' 010600                                                  MOV     SP,R0
014024' 104414                                                  TRAP    C$PNTB
014026' 062706  000010                                          ADD     #10,SP
014032'                 ENDMSG
014032'                                         L10026:
014032' 104423                                                  TRAP    C$MSG

014034'                 BGNMSG  MSG18
014034'                                         MSG18::
014034'                         PRINTB  #FORM32,R1,BITNAM
014034' 013746  000310'                                         MOV     BITNAM,-(SP)
014040' 010146                                                  MOV     R1,-(SP)
014042' 012746  006437'                                         MOV     #FORM32,-(SP)
```

```
3268  014046' 012746  000003                                                    MOV     #3,-(SP)
3269  014052' 010600                                                            MOV     SP,R0
3270  014054' 104414                                                            TRAP    C$PNTB
3271  014056' 062706  000010                                                    ADD     #10,SP
3272  014062'                      ENDMSG
3273  014062'                                                           L10027:
3274  014062' 104423                                                            TRAP    C$MSG
3275
3276  014064'                      BGNMSG  MSG19
3277  014064'                                                           MSG19::
3278  014064'                              PRINTB  #FORM33,R1
3279  014064' 010146                                                            MOV     R1,-(SP)
3280  014066' 012746  006534'                                                   MOV     #FORM33,-(SP)
3281  014072' 012746  000002                                                    MOV     #2,-(SP)
3282  014076' 010600                                                            MOV     SP,R0
3283  014100' 104414                                                            TRAP    C$PNTB
3284  014102' 062706  000006                                                    ADD     #6,SP
3285  014106'                              PRINTB  #FORM34,R2,R3
3286  014106' 010346                                                            MOV     R3,-(SP)
3287  014110' 010246                                                            MOV     R2,-(SP)
3288  014112' 012746  006572'                                                   MOV     #FORM34,-(SP)
3289  014116' 012746  000003                                                    MOV     #3,-(SP)
3290  014122' 010600                                                            MOV     SP,R0
3291  014124' 104414                                                            TRAP    C$PNTB
3292  014126' 062706  000010                                                    ADD     #10,SP
3293  014132'                              PRINTB  #FORM35,R1,R4
3294  014132' 010446                                                            MOV     R4,-(SP)
3295  014134' 010146                                                            MOV     R1,-(SP)
3296  014136' 012746  006655'                                                   MOV     #FORM35,-(SP)
3297  014142' 012746  000003                                                    MOV     #3,-(SP)
3298  014146' 010600                                                            MOV     SP,R0
3299  014150' 104414                                                            TRAP    C$PNTB
3300  014152' 062706  000010                                                    ADD     #10,SP
3301  014156'                      ENDMSG
3302  014156'                                                           L10030:
3303  014156' 104423                                                            TRAP    C$MSG
3304
3305  014160'                      BGNMSG  MSG20
3306  014160'                                                           MSG20::
3307  014160'                              PRINTB  #FORM11
3308  014160' 012746  005074'                                                   MOV     #FORM11,-(SP)
3309  014164' 012746  000001                                                    MOV     #1,-(SP)
3310  014170' 010600                                                            MOV     SP,R0
3311  014172' 104414                                                            TRAP    C$PNTB
3312  014174' 062706  000004                                                    ADD     #4,SP
3313  014200'                              PRINTB  #FORM36,R3
3314  014200' 010346                                                            MOV     R3,-(SP)
3315  014202' 012746  006755'                                                   MOV     #FORM36,-(SP)
3316  014206' 012746  000002                                                    MOV     #2,-(SP)
3317  014212' 010600                                                            MOV     SP,R0
3318  014214' 104414                                                            TRAP    C$PNTB
3319  014216' 062706  000006                                                    ADD     #6,SP
3320  014222'                              PRINTB  #FORM27,R1,R2
3321  014222' 010246                                                            MOV     R2,-(SP)
3322  014224' 010146                                                            MOV     R1,-(SP)
3323  014226' 012746  006261'                                                   MOV     #FORM27,-(SP)
```

```
014232' 012746  000003                                                             MOV     #3,-(SP)
014236' 010600                                                                     MOV     SP,R0
014240' 104414                                                                     TRAP    C$PNTB
014242' 062706  000010                                                             ADD     #10,SP
014246'                 ENDMSG
014246'                                                                  L10031:
014246' 104423                                                                     TRAP    C$MSG

014250'                 BGNMSG  MSG21
014250'                                                                  MSG21::
014250'                         PRINTB  #FORM11
014250' 012746  005074'                                                            MOV     #FORM11,-(SP)
014254' 012746  000001                                                             MOV     #1,-(SP)
014260' 010600                                                                     MOV     SP,R0
014262' 104414                                                                     TRAP    C$PNTB
014264' 062706  000004                                                             ADD     #4,SP
014270'                         PRINTB  #FORM29,R1
014270' 010146                                                                     MOV     R1,-(SP)
014272' 012746  006336'                                                            MOV     #FORM29,-(SP)
014276' 012746  000002                                                             MOV     #2,-(SP)
014302' 010600                                                                     MOV     SP,R0
014304' 104414                                                                     TRAP    C$PNTB
014306' 0627^   000006                                                             ADD     #6,SP
014312'                         PRINTB  #FORM27,R2,R3
014312' 010346                                                                     MOV     R3,-(SP)
014314' 010246                                                                     MOV     R2,-(SP)
014316' 012746  006261'                                                            MOV     #FORM27,-(SP)
014322' 012746  000003                                                             MOV     #3,-(SP)
014326' 010600                                                                     MOV     SP,R0
014330' 104414                                                                     TRAP    C$PNTB
014332' 062706  000010                                                             ADD     #10,SP
014336'                 ENDMSG
014336'                                                                  L10032:
014336' 104423                                                                     TRAP    C$MSG

014340'                 BGNMSG  MSG22
014340'                                                                  MSG22::
014340' 010146                  MOV     R1,-(SP)
014342'                         PRINTB  #FORM37,R2
014342' 010246                                                                     MOV     R2,-(SP)
014344' 012746  007013'                                                            MOV     #FORM37,-(SP)
014350' 012746  000002                                                             MOV     #2,-(SP)
014354' 010600                                                                     MOV     SP,R0
014356' 104414                                                                     TRAP    C$PNTB
014360' 062706  000006                                                             ADD     #6,SP
014364' 005001                  CLR     R1                  ;TRANSMIT STATUS WORD 0
014366'                         PRINTB  #FORM38,R1,R3,PCBB+2
014366' 013746  000606'                                                            MOV     PCBB+2,-(SP)
014372' 010346                                                                     MOV     R3,-(SP)
014374' 010146                                                                     MOV     R1,-(SP)
014376' 012746  007124'                                                            MOV     #FORM38,-(SP)
014402' 012746  000004                                                             MOV     #4,-(SP)
014406' 010600                                                                     MOV     SP,R0
014410' 104414                                                                     TRAP    C$PNTB
014412' 062706  000012                                                             ADD     #12,SP
014416' 005201                  INC     R1                  ;TRANSMIT STATUS WORD 1
```

```
014420'                                 PRINTB  #FORM38,R1,R4,PCBB+4
014420' 013746  000610'                                                             MOV     PCBB+4,-(SP)
014424' 010446                                                                      MOV     R4,-(SP)
014426' 010146                                                                      MOV     R1,-(SP)
014430' 012746  007124'                                                             MOV     #FORM38,-(SP)
014434' 012746  000004                                                              MOV     #4,-(SP)
014440' 010600                                                                      MOV     SP,R0
014442' 104414                                                                      TRAP    C$PNTB
014444' 062706  000012                                                              ADD     #12,SP
014450' 012601                          MOV     (SP)+,R1
014452'                         ENDMSG
014452'                                                                     L10033:
014452' 104423                                                                      TRAP    C$MSG

014454'                         BGNMSG  MSG23
014454'                                                                     MSG23::
014454'                                 PRINTB  #FORM39
014454' 012746  007215'                                                             MOV     #FORM39,-(SP)
014460' 012746  000001                                                              MOV     #1,-(SP)
014464' 010600                                                                      MOV     SP,R0
014466' 104414                                                                      TRAP    C$PNTB
014470' 062706  000004                                                              ADD     #4,SP
014474'                         ENDMSG
014474'                                                                     L10034:
014474' 104423                                                                      TRAP    C$MSG

014476'                         BGNMSG  MSG24
014476'                                                                     MSG24::
014476'                                 PRINTB  #FORM40
014476' 012746  007256'                                                             MOV     #FORM40,-(SP)
014502' 012746  000001                                                              MOV     #1,-(SP)
014506' 010600                                                                      MOV     SP,R0
014510' 104414                                                                      TRAP    C$PNTB
014512' 062706  000004                                                              ADD     #4,SP
014516'                                 PRINTB  #FORM41
014516' 012746  007341'                                                             MOV     #FORM41,-(SP)
014522' 012746  000001                                                              MOV     #1,-(SP)
014526' 010600                                                                      MOV     SP,R0
014530' 104414                                                                      TRAP    C$PNTB
014532' 062706  000004                                                              ADD     #4,SP
014536' 012701  000001                  MOV     #1,R1
014542' 012702  000624'                 MOV     #CNTTAB,R2
014546' 012203                  10$:    MOV     (R2)+,R3
014550'                                 PRINTB  #FORM42,R1
014550' 010146                                                                      MOV     R1,-(SP)
014552' 012746  007407'                                                             MOV     #FORM42,-(SP)
014556' 012746  000002                                                              MOV     #2,-(SP)
014562' 010600                                                                      MOV     SP,R0
014564' 104414                                                                      TRAP    C$PNTB
014566' 062706  000006                                                              ADD     #6,SP
014572'                                 PRINTB  #FORM43,R3
014572' 010346                                                                      MOV     R3,-(SP)
014574' 012746  007416'                                                             MOV     #FORM43,-(SP)
014600' 012746  000002                                                              MOV     #2,-(SP)
014604' 010600                                                                      MOV     SP,R0
014606' 104414                                                                      TRAP    C$PNTB
```

```
014610' 062706 000006                                                                  ADD    #6,SP
014614' 005201                      INC    R1
014616' 020127 000021               CMP    R1,#17.
014622' 001351                      BNE    10$
014624'                      ENDMSG
014624'                                                                        L10035:
014624' 104423                                                                         TRAP   C$MSG

014626'                      BGNMSG MSG25
014626'                                                                        MSG25::
014626' 122777 000003 163504        CMPB   #INERR,@PCSR1  ;WAS MATCH BIT SET?
014634' 001011                      BNE    10$            ;YES
014636'                             PRINTB #FORM76        ;NO
014636' 012746 012115'                                                                 MOV    #FORM76,-(SP)
014642' 012746 000001                                                                  MOV    #1,-(SP)
014646' 010600                                                                         MOV    SP,R0
014650' 104414                                                                         TRAP   C$PNTB
014652' 062706 000004                                                                  ADD    #4,SP
014656' 000410                      BR     20$
014660'                      10$:   PRINTB #FORM77
014660' 012746 012151'                                                                 MOV    #FORM77,-(SP)
014664' 012746 000001                                                                  MOV    #1,-(SP)
014670' 010600                                                                         MOV    SP,R0
014672' 104414                                                                         TRAP   C$PNTB
014674' 062706 000004                                                                  ADD    #4,SP
014700'                      20$:   PRINTB #FORM44
014700' 012746 007424'                                                                 MOV    #FORM44,-(SP)
014704' 012746 000001                                                                  MOV    #1,-(SP)
014710' 010600                                                                         MOV    SP,R0
014712' 104414                                                                         TRAP   C$PNTB
014714' 062706 000004                                                                  ADD    #4,SP
014720'                             PRINTB #FORM45,(R1)
014720' 011146                                                                         MOV    (R1),-(SP)
014722' 012746 007503'                                                                 MOV    #FORM45,-(SP)
014726' 012746 000002                                                                  MOV    #2,-(SP)
014732' 010600                                                                         MOV    SP,R0
014734' 104414                                                                         TRAP   C$PNTB
014736' 062706 000006                                                                  ADD    #6,SP
014742'                             PRINTB #FORM46,2(R1)
014742' 016146 000002                                                                  MOV    2(R1),-(SP)
014746' 012746 007531'                                                                 MOV    #FORM46,-(SP)
014752' 012746 000002                                                                  MOV    #2,-(SP)
014756' 010600                                                                         MOV    SP,R0
014760' 104414                                                                         TRAP   C$PNTB
014762' 062706 000006                                                                  ADD    #6,SP
014766'                             PRINTB #FORM47,4(R1)
014766' 016146 000004                                                                  MOV    4(R1),-(SP)
014772' 012746 007560'                                                                 MOV    #FORM47,-(SP)
014776' 012746 000002                                                                  MOV    #2,-(SP)
015002' 010600                                                                         MOV    SP,R0
015004' 104414                                                                         TRAP   C$PNTB
015006' 062706 000006                                                                  ADD    #6,SP
015012'                      ENDMSG
015012'                                                                        L10036:
015012' 104423                                                                         TRAP   C$MSG

```

C
C

```
015014'                         BGNMSG  MSG26
015014'                                                 MSG26::
015014'                                 PRINTB  #FORM48
015014' 012746  007607'                                         MOV     #FORM48,-(SP)
015020' 012746  000001                                          MOV     #1,-(SP)
015024' 010600                                                  MOV     SP,R0
015026' 104414                                                  TRAP    C$PNTB
015030' 062706  000004                                          ADD     #4,SP
015034'                                 PRINTB  #FORM49,R1
015034' 010146                                                  MOV     R1,-(SP)
015036' 012746  007704'                                         MOV     #FORM49,-(SP)
015042' 012746  000002                                          MOV     #2,-(SP)
015046' 010600                                                  MOV     SP,R0
015050' 104414                                                  TRAP    C$PNTB
015052' 062706  000006                                          ADD     #6,SP
015056'                         ENDMSG
015056'                                                 L10037:
015056' 104423                                                  TRAP    C$MSG

015060'                         BGNMSG  MSG27
015060'                                                 MSG27::
015060'                                 PRINTB  #FORM51
015060' 012746  010047'                                         MOV     #FORM51,-(SP)
015064' 012746  000001                                          MOV     #1,-(SP)
015070' 010600                                                  MOV     SP,R0
015072' 104414                                                  TRAP    C$PNTB
015074' 062706  000004                                          ADD     #4,SP
015100'                                 PRINTB  #FORM50,R4,R3
015100' 010346                                                  MOV     R3,-(SP)
015102' 010446                                                  MOV     R4,-(SP)
015104' 012746  007760'                                         MOV     #FORM50,-(SP)
015110' 012746  000003                                          MOV     #3,-(SP)
015114' 010600                                                  MOV     SP,R0
015116' 104414                                                  TRAP    C$PNTB
015120' 062706  000010                                          ADD     #10,SP
015124'                         ENDMSG
015124'                                                 L10040:
015124' 104423                                                  TRAP    C$MSG

015126'                         BGNMSG  MSG28
015126'                                                 MSG28::
015126'                                 PRINTB  #FORM52
015126' 012746  010101'                                         MOV     #FORM52,-(SP)
015132' 012746  000001                                          MOV     #1,-(SP)
015136' 010600                                                  MOV     SP,R0
015140' 104414                                                  TRAP    C$PNTB
015142' 062706  000004                                          ADD     #4,SP
015146'                                 PRINTB  #FORM50,R4,R3
015146' 010346                                                  MOV     R3,-(SP)
015150' 010446                                                  MOV     R4,-(SP)
015152' 012746  007760'                                         MOV     #FORM50,-(SP)
015156' 012746  000003                                          MOV     #3,-(SP)
015162' 010600                                                  MOV     SP,R0
015164' 104414                                                  TRAP    C$PNTB
015166' 062706  000010                                          ADD     #10,SP
015172'                         ENDMSG
```

```
015172'                                                     L10041:
015172' 104423                                                      TRAP    C$MSG

015174'                         BGNMSG  MSG29
015174'                                                     MSG29::
015174'                                 PRINTB  #FORM50,#0,PCBB+4
015174' 013746  000610'                                             MOV     PCBB+4,-(SP)
015200' 012746  000000                                              MOV     #0,-(SP)
015204' 012746  007760'                                             MOV     #FORM50,-(SP)
015210' 012746  000003                                              MOV     #3,-(SP)
015214' 010600                                                      MOV     SP,R0
015216' 104414                                                      TRAP    C$PNTB
015220' 062706  000010                                              ADD     #10,SP
015224'                                 PRINTB  #FORM53,PCBB+2
015224' 013746  000606'                                             MOV     PCBB+2,-(SP)
015230' 012746  010137'                                             MOV     #FORM53,-(SP)
015234' 012746  000002                                              MOV     #2,-(SP)
015240' 010600                                                      MOV     SP,R0
015242' 104414                                                      TRAP    C$PNTB
015244' 062706  000006                                              ADD     #6,SP
015250'                                 PRINTB  #FORM54,PCBB
015250' 013746  000604'                                             MOV     PCBB,-(SP)
015254' 012746  010211'                                             MOV     #FORM54,-(SP)
015260' 012746  000002                                              MOV     #2,-(SP)
015264' 010600                                                      MOV     SP,R0
015266' 104414                                                      TRAP    C$PNTB
015270' 062706  000006                                              ADD     #6,SP
015274'                         ENDMSG
015274'                                                     L10042:
015274' 104423                                                      TRAP    C$MSG

015276'                         BGNMSG  MSG30
015276'                                                     MSG30::
015276'                                 PRINTB  #FORM55
015276' 012746  010237'                                             MOV     #FORM55,-(SP)
015302' 012746  000001                                              MOV     #1,-(SP)
015306' 010600                                                      MOV     SP,R0
015310' 104414                                                      TRAP    C$PNTB
015312' 062706  000004                                              ADD     #4,SP
015316'                                 PRINTB  #FORM57,PCBB
015316' 013746  000604'                                             MOV     PCBB,-(SP)
015322' 012746  010363'                                             MOV     #FORM57,-(SP)
015326' 012746  000002                                              MOV     #2,-(SP)
015332' 010600                                                      MOV     SP,R0
015334' 104414                                                      TRAP    C$PNTB
015336' 062706  000006                                              ADD     #6,SP
015342'                         ENDMSG
015342'                                                     L10043:
015342' 104423                                                      TRAP    C$MSG

015344'                         BGNMSG  MSG31
015344'                                                     MSG31::
015344'                                 PRINTB  #FORM56
015344' 012746  010310'                                             MOV     #FORM56,-(SP)
015350' 012746  000001                                              MOV     #1,-(SP)
015354' 010600                                                      MOV     SP,R0
```

```
3604  015356' 104414                                                   TRAP    C$PNTB
3605  015360' 062706  000004                                           ADD     #4,SP
3606  015364'                          PRINTB  #FORM57,R1
3607  015364' 010146                                                   MOV     R1,-(SP)
3608  015366' 012746  010363'                                          MOV     #FORM57,-(SP)
3609  015372' 012746  000002                                           MOV     #2,-(SP)
3610  015376' 010600                                                   MOV     SP,R0
3611  015400' 104414                                                   TRAP    C$PNTB
3612  015402' 062706  000006                                           ADD     #6,SP
3613  015406'                          PRINTB  #FORM58,#104.
3614  015406' 012746  000150                                           MOV     #104.,-(SP)
3615  015412' 012746  010433'                                          MOV     #FORM58,-(SP)
3616  015416' 012746  000002                                           MOV     #2,-(SP)
3617  015422' 010600                                                   MOV     SP,R0
3618  015424' 104414                                                   TRAP    C$PNTB
3619  015426' 062706  000006                                           ADD     #6,SP
3620  015432'                          PRINTB  #FORM59,#100000
3621  015432' 012746  100000                                           MOV     #100000,-(SP)
3622  015436' 012746  010511'                                          MOV     #FORM59,-(SP)
3623  015442' 012746  000002                                           MOV     #2,-(SP)
3624  015446' 010600                                                   MOV     SP,R0
3625  015450' 104414                                                   TRAP    C$PNTB
3626  015452' 062706  000006                                           ADD     #6,SP
3627  015456'                          PRINTB  #FORM60,#100000,PCBB+2
3628  015456' 013746  000606'                                          MOV     PCBB+2,-(SP)
3629  015462' 012746  100000                                           MOV     #100000,-(SP)
3630  015466' 012746  010604'                                          MOV     #FORM60,-(SP)
3631  015472' 012746  000003                                           MOV     #3,-(SP)
3632  015476' 010600                                                   MOV     SP,R0
3633  015500' 104414                                                   TRAP    C$PNTB
3634  015502' 062706  000010                                           ADD     #10,SP
3635  015506'                  ENDMSG
3636  015506'                                                  L10044:
3637  015506' 104423                                                   TRAP    C$MSG
3638
3639  015510'                  BGNMSG  MSG32
3640  015510'                                                  MSG32::
3641  015510'                          PRINTB  #FORM61
3642  015510' 012746  010671'                                          MOV     #FORM61,-(SP)
3643  015514' 012746  000001                                           MOV     #1,-(SP)
3644  015520' 010600                                                   MOV     SP,R0
3645  015522' 104414                                                   TRAP    C$PNTB
3646  015524' 062706  000004                                           ADD     #4,SP
3647  015530'                          PRINTB  #FORM27,#52525,PCBB+4
3648  015530' 013746  000610'                                          MOV     PCBB+4,-(SP)
3649  015534' 012746  052525                                           MOV     #52525,-(SP)
3650  015540' 012746  006261'                                          MOV     #FORM27,-(SP)
3651  015544' 012746  000003                                           MOV     #3,-(SP)
3652  015550' 010600                                                   MOV     SP,R0
3653  015552' 104414                                                   TRAP    C$PNTB
3654  015554' 062706  000010                                           ADD     #10,SP
3655  015560'                          PRINTB  #FORM57,PCBB
3656  015560' 013746  000604'                                          MOV     PCBB,-(SP)
3657  015564' 012746  010363'                                          MOV     #FORM57,-(SP)
3658  015570' 012746  000002                                           MOV     #2,-(SP)
3659  015574' 010600                                                   MOV     SP,R0
```

```
015576' 104414                                                        TRAP    C$PNTB
015600' 062706  000006                                                ADD     #6,SP
015604'                   ENDMSG
015604'                                                     L10045:
015604' 104423                                                        TRAP    C$MSG

015606'                   BGNMSG  MSG33
015606'                                                     MSG33::
015606'                           PRINTB  #FORM62,@BITNAM
015606' 017746  162476                                                MOV     @BITNAM,-(SP)
015612' 012746  010754'                                               MOV     #FORM62,-(SP)
015616' 012746  000002                                                MOV     #2,-(SP)
015622' 010600                                                        MOV     SP,R0
015624' 104414                                                        TRAP    C$PNTB
015626' 062706  000006                                                ADD     #6,SP
015632'                   ENDMSG
015632'                                                     L10046:
015632' 104423                                                        TRAP    C$MSG

015634'                   BGNMSG  MSG34
015634'                                                     MSG34::
015634'                           PRINTB  #FORM63,R4
015634' 010446                                                        MOV     R4,-(SP)
015636' 012746  011027'                                               MOV     #FORM63,-(SP)
015642' 012746  000002                                                MOV     #2,-(SP)
015646' 010600                                                        MOV     SP,R0
015650' 104414                                                        TRAP    C$PNTB
015652' 062706  000006                                                ADD     #6,SP
015656'                           PRINTB  #FORM28,STMSG
015656' 013746  024202'                                               MOV     STMSG,-(SP)
015662' 012746  006331'                                               MOV     #FORM28,-(SP)
015666' 012746  000002                                                MOV     #2,-(SP)
015672' 010600                                                        MOV     SP,R0
015674' 104414                                                        TRAP    C$PNTB
015676' 062706  000006                                                ADD     #6,SP
015702'                   ENDMSG
015702'                                                     L10047:
015702' 104423                                                        TRAP    C$MSG

015704'                   BGNMSG  MSG35
015704'                                                     MSG35::
015704'                           PRINTB  #FORM59,PCBB
015704' 013746  000604'                                               MOV     PCBB,-(SP)
015710' 012746  010511'                                               MOV     #FORM59,-(SP)
015714' 012746  000002                                                MOV     #2,-(SP)
015720' 010600                                                        MOV     SP,R0
015722' 104414                                                        TRAP    C$PNTB
015724' 062706  000006                                                ADD     #6,SP
015730'                           PRINTB  #FORM60
015730' 012746  010604'                                               MOV     #FORM60,-(SP)
015734' 012746  000001                                                MOV     #1,-(SP)
015740' 010600                                                        MOV     SP,R0
015742' 104414                                                        TRAP    C$PNTB
015744' 062706  000004                                                ADD     #4,SP
015750'                           PRINTB  #FORM78,PCBB,PCBB+2
015750' 013746  000606'                                               MOV     PCBB+2,-(SP)
```

```
3716  015754' 013746  000604'                                                  MOV     PCBB,-(SP)
3717  015760' 012746  012225'                                                  MOV     #FORM78,-(SP)
3718  015764' 012746  000003                                                   MOV     #3,-(SP)
3719  015770' 010600                                                           MOV     SP,RC
3720  015772' 104414                                                           TRAP    C$PNTB
3721  015774' 062706  000010                                                   ADD     #10,SP
3722  016000'                         ENDMSG
3723  016000'                                                          L10050:
3724  016000' 104423                                                           TRAP    C$MSG
3725
3726  016002'                         BGNMSG  MSG36
3727  016002'                                                          MSG36::
3728  016002'                                 PRINTB  #FORM66
3729  016002' 012746  011165'                                                  MOV     #FORM66,-(SP)
3730  016006' 012746  000001                                                   MOV     #1,-(SP)
3731  016012' 010600                                                           MOV     SP,R0
3732  016014' 104414                                                           TRAP    C$PNTB
3733  016016' 062706  000004                                                   ADD     #4,SP
3734  016022'                         ENDMSG
3735  016022'                                                          L10051:
3736  016022' 104423                                                           TRAP    C$MSG
3737
3738  016024'                         BGNMSG  MSG37
3739  016024'                                                          MSG37::
3740  016024'                                 PRINTB  #FORM67
3741  016024' 012746  011252'                                                  MOV     #FORM67,-(SP)
3742  016030' 012746  000001                                                   MOV     #1,-(SP)
3743  016034' 010600                                                           MOV     SP,R0
3744  016036' 104414                                                           TRAP    C$PNTB
3745  016040' 062706  000004                                                   ADD     #4,SP
3746  016044'                         ENDMSG
3747  016044'                                                          L10052:
3748  016044' 104423                                                           TRAP    C$MSG
3749
3750  016046'                         BGNMSG  MSG38
3751  016046'                                                          MSG38::
3752  016046'                                 PRINTB  #FORM68
3753  016046' 012746  011336'                                                  MOV     #FORM68,-(SP)
3754  016052' 012746  000001                                                   MOV     #1,-(SP)
3755  016056' 010600                                                           MOV     SP,R0
3756  016060' 104414                                                           TRAP    C$PNTB
3757  016062' 062706  000004                                                   ADD     #4,SP
3758  016066'                         ENDMSG
3759  016066'                                                          L10053:
3760  016066' 104423                                                           TRAP    C$MSG
3761
3762  016070'                         BGNMSG  MSG39
3763  016070'                                                          MSG39::
3764  016070'                                 PRINTB  #FORM69
3765  016070' 012746  011433'                                                  MOV     #FORM69,-(SP)
3766  016074' 012746  000001                                                   MOV     #1,-(SP)
3767  016100' 010600                                                           MOV     SP,R0
3768  016102' 104414                                                           TRAP    C$PNTB
3769  016104' 062706  000004                                                   ADD     #4,SP
3770  016110'                         ENDMSG
3771  016110'                                                          L10054:
```

```
016110' 104423                                          TRAP    C$MSG

016112'                          BGNMSG  MSG40
016112'                                         MSG40::
016112'                                  PRINTB  #FORM56
016112' 012746  010310'                                 MOV     #FORM56,-(SP)
016116' 012746  000001                                  MOV     #1,-(SP)
016122' 010600                                          MOV     SP,R0
016124' 104414                                          TRAP    C$PNTB
016126' 062706  000004                                  ADD     #4,SP
016132'                                  PRINTB  #FORM59,#100000
016132' 012746  100000                                  MOV     #100000,-(SP)
016136' 012746  010511'                                 MOV     #FORM59,-(SP)
016142' 012746  000002                                  MOV     #2,-(SP)
016146' 010600                                          MOV     SP,R0
016150' 104414                                          TRAP    C$PNTB
016152' 062706  000006                                  ADD     #6,SP
016156'                                  PRINTB  #FORM60,#100000,PCBB
016156' 013746  000604'                                 MOV     PCBB,-(SP)
016162' 012746  100000                                  MOV     #100000,-(SP)
016166' 012746  010604'                                 MOV     #FORM60,-(SP)
016172' 012746  000003                                  MOV     #3,-(SP)
016176' 010600                                          MOV     SP,R0
016200' 104414                                          TRAP    C$PNTB
016202' 062706  000010                                  ADD     #10,SP
016206'                          ENDMSG
016206'                                         L10055:
016206' 104423                                          TRAP    C$MSG

016210'                          BGNMSG  MSG41
016210'                                         MSG41::
016210'                                  PRINTB  #FORM61
016210' 012746  010671'                                 MOV     #FORM61,-(SP)
016214' 012746  000001                                  MOV     #1,-(SP)
016220' 010600                                          MOV     SP,R0
016222' 104414                                          TRAP    C$PNTB
016224' 062706  000004                                  ADD     #4,SP
016230'                                  PRINTB  #FORM27,#52525,PCBB+2
016230' 013746  000606'                                 MOV     PCBB+2,-(SP)
016234' 012746  052525                                  MOV     #52525,-(SP)
016240' 012746  006261'                                 MOV     #FORM27,-(SP)
016244' 012746  000003                                  MOV     #3,-(SP)
016250' 010600                                          MOV     SP,R0
016252' 104414                                          TRAP    C$PNTB
016254' 062706  000010                                  ADD     #10,SP
016260'                          ENDMSG
016260'                                         L10056:
016260' 104423                                          TRAP    C$MSG

016262'                          BGNMSG  MSG42
016262'                                         MSG42::
016262'                                  PRINTB  #FORM70,PCBB+2
016262' 013746  000606'                                 MOV     PCBB+2,-(SP)
016266' 012746  011551'                                 MOV     #FORM70,-(SP)
016272' 012746  000002                                  MOV     #2,-(SP)
016276' 010600                                          MOV     SP,R0
```

```
016300' 104414                                                                 TRAP    C$PNTB
016302' 062706  000006                                                         ADD     #6,SP
016306'                            PRINTB  #FORM71,PCBB
016306' 013746  000604'                                                        MOV     PCBB,-(SP)
016312' 012746  011613'                                                        MOV     #FORM71,-(SP)
016316' 012746  000002                                                         MOV     #2,-'SP)
016322' 010600                                                                 MOV     SP,R0
016324' 104414                                                                 TRAP    C$PNTB
016326' 062706  000006                                                         ADD     #6,SP
016332'                            PRINTB  #FORM33,R1
016332' 010146                                                                 MOV     R1,-(SP)
016334' 012746  006534'                                                        MOV     #FORM33,-(SP)
016340' 012746  000002                                                         MOV     #2,-(SP)
016344' 010600                                                                 MOV     SP,R0
016346' 104414                                                                 TRAP    C$PNTB
016350' 062706  000006                                                         ADD     #6,SP
016354'                            PRINTB  #FORM36,PCBB+4
016354' 013746  000610'                                                        MOV     PCBB+4,-(SP,
016360' 012746  006755'                                                        MOV     #FORM36,-(SP)
016364' 012746  000002                                                         MOV     #2,-(SP)
016370' 010600                                                                 MOV     SP,R0
016372' 104414                                                                 TRAP    C$PNTB
016374' 062706  000006                                                         ADD     #6,SP
016400'                            PRINTB  #FORM27,PCBB+6,PCBB+10
016400' 013746  000614'                                                        MOV     PCBB+10,-(SP)
016404' 013746  000612'                                                        MOV     PCBB+6,-(SP)
016410' 012746  006261'                                                        MOV     #FORM27,-(SP)
016414' 012746  000003                                                         MOV     #3,-(SP)
016420' 010600                                                                 MOV     SP,R0
016422' 104414                                                                 TRAP    C$PNTB
016424' 062706  000010                                                         ADD     #10,SP
016430'                    ENDMSG
016430'                                                                L10057:
016430' 104423                                                                 TRAP    C$MSG

016432'                    BGNMSG  MSG43
016432'                                                                MSG43::
016432'                            PRINTB  #FORM72
016432' 012746  011655'                                                        MOV     #FORM72,-(SP)
016436' 012746  000001                                                         MOV     #1,-(SP)
016442' 010600                                                                 MOV     SP,R0
016444' 104414                                                                 TRAP    C$PNTB
016446' 062706  000004                                                         ADD     #4,SP
016452'                    ENDMSG
016452'                                                                L10060:
016452' 104423                                                                 TRAP    C$MSG

016454'                    BGNMSG  MSG44
016454'                                                                MSG44::
016454' 032737  100000  000666'    BIT     #NPRERR,ERRINT          ;NPR ERROR OCCUR?
016462' 001410                     BEQ     10$                     ;NO
016464'                            PRINTB  #FORM73                 ;YES, PRINT NPR ERROR MESSAGE
016464' 012746  011727'                                                        MOV     #FORM73,-(SP)
016470' 012746  000001                                                         MOV     #1,-(SP)
016474' 010600                                                                 MOV     SP,R0
016476' 104414                                                                 TRAP    C$PNTB
```

```
016500' 062706  000004                                                                          ADD     #4,SP
016504' 032737  040000  000666' 10$:    BIT     #NXMERR,ERRINT          ;NON-EXISTANT MEMORY OCCUR?
016512' 001410                          BEQ     20$                     ;NO
016514'                                 PRINTB  #FORM74                 ;YES, PRINT NON-EXISTANT MEMORY MESSAGE
016514' 012746  011772'                                                                         MOV     #FORM74,-(SP)
016520' 012746  000001                                                                          MOV     #1,-(SP)
016524' 010600                                                                                  MOV     SP,R0
016526' 104414                                                                                  TRAP    C$PNTB
016530' 062706  000004                                                                          ADD     #4,SP
016534' 032737  020000  000666' 20$:    BIT     #UNIERR,ERRINT          ;UNEXPECTED INTERRUPT OCCUR?
016542' 001410                          BEQ     30$                     ;NO
016544'                                 PRINTB  #FORM75                 ;YES, PRINT UNEXPECTED INTERRUPT MESSAGE
016544' 012746  012047'                                                                         MOV     #FORM75,-(SP)
016550' 012746  000001                                                                          MOV     #1,-(SP)
016554' 010600                                                                                  MOV     SP,R0
016556' 104414                                                                                  TRAP    C$PNTB
016560' 062706  000004                                                                          ADD     #4,SP
016564' 032737  010000  000666' 30$:    BIT     #PARERR,ERRINT          ;PARITY ERROR OCCUR?
016572' 001410                          BEQ     40$                     ;NO
016574'                                 PRINTB  #FORM72                 ;YES, PRINT PARITY ERROR MESSAGE
016574' 012746  011655'                                                                         MOV     #FORM72,-(SP)
016600' 012746  000001                                                                          MOV     #1,-(SP)
016604' 010600                                                                                  MOV     SP,R0
016606' 104414                                                                                  TRAP    C$PNTB
016610' 062706  000004                                                                          ADD     #4,SP
016614'                         40$:
016614'                         ENDMSG
016614'                                                                                 L10061:
016614' 104423                                                                                  TRAP    C$MSG

016616'                         BGNMSG  MSG45
016616'                                                                                 MSG45::
016616'                                 PRINTB  #FORM79
016616' 012746  012265'                                                                         MOV     #FORM79,-(SP)
016622' 012746  000001                                                                          MOV     #1,-(SP)
016626' 010600                                                                                  MOV     SP,R0
016630' 104414                                                                                  TRAP    C$PNTB
016632' 062706  000004                                                                          ADD     #4,SP
016636'                                 PRINTB  #FORM80
016636' 012746  012334'                                                                         MOV     #FORM80,-(SP)
016642' 012746  000001                                                                          MOV     #1,-(SP)
016646' 010600                                                                                  MOV     SP,R0
016650' 104414                                                                                  TRAP    C$PNTB
016652' 062706  000004                                                                          ADD     #4,SP
016656'                                 PRINTB  #FORM81
016656' 012746  012413'                                                                         MOV     #FORM81,-(SP)
016662' 012746  000001                                                                          MOV     #1,-(SP)
016666' 010600                                                                                  MOV     SP,R0
016670' 104414                                                                                  TRAP    C$PNTB
016672' 062706  000004                                                                          ADD     #4,SP
016676'                         ENDMSG
016676'                                                                                 L10062:
016676' 104423                                                                                  TRAP    C$MSG

016700'                         BGNMSG  MSG46
016700'                                                                                 MSG46::
 3940  016700'                                    PRINTB  #FORM82
 3941  016700' 012746  012456'                                                                MOV     #FORM82,-(SP)
 3942  016704' 012746  000001                                                                 MOV     #1,-(SP)
 3943  016710' 010600                                                                         MOV     SP,R0
 3944  016712' 104414                                                                         TRAP    C$PNTB
 3945  016714' 062706  000004                                                                 ADD     #4,SP
 3946  016720'                              ENDMSG
 3947  016720'                                                                          L10063:
 3948  016720' 104423                                                                         TRAP    C$MSG
 3949
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

                                ;++
                                ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
                                ; THAT ARE USED IN MORE THAN ONE TEST.
                                ;--
                                .SBTTL  32 BIT CRC CALCULATOR

                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;    SUBROUTINE TO CALCULATE A 32 BIT CRC ON A BLOCK OF DATA
                                ;
                                ; INPUTS:
                                ;       R0 = ADDRESS OF DATA BLOCK
                                ;       R2 = BYTE COUNT
                                ;
                                ; IMPLICIT INPUTS: NONE
                                ;
                                ; OUTPUTS:
                                ;       R4 = CRC HIGH WORD
                                ;       R5 = CRC LOW WORD
                                ;
                                ; SUBORDINATE ROUTINES USED: GETCRC
                                ;
                                ; FUNCTIONAL SIDE EFFECTS: NONE
                                ;
                                ; CALLING SEQUENCE. PUT ADDRESS OF DATA TO PERFORM CRC ON IN R0
                                ;               PUT NUMBER OF BYTES TO CHECK IN R2
                                ;               JSR     PC,BLKCRC
                                ;
                                ;--

016722'                         CRC32::
016722' 010046                          MOV     R0,-(SP)        ;SAVE REGISTERS 0-3
016724' 010146                          MOV     R1,-(SP)
016726' 010246                          MOV     R2,-(SP)
016730' 010346                          MOV     R3,-(SP)
016732' 012704  177777                  MOV     #INITH,R4       ;INITIAL CRC HIGH WORD
016736' 012705  177777                  MOV     #INITL,R5       ;INITIAL CRC LOW WORD
016742' 112001                  10$:    MOVB    (R0)+,R1        ;GET NEXT BYTE OF DATA
016744' 004737  016764'                 JSR     PC,GETCRC       ;CALCULATE THE CRC
016750' 077204                          SOB     R2,10$          ;LOOP TILL DONE
016752' 012603                          MOV     (SP)+,R3        ;RESTORE REGISTERS
016754' 012602                          MOV     (SP)+,R2
016756' 012601                          MOV     (SP)+,R1
016760' 012600                          MOV     (SP)+,R0
016762' 000207                          RTS     PC              ;RETURN
```

```
                                 ;++
                                 ;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
                                 ;
                                 ;--

016764'                          GETCRC:
016764' 010146                           MOV     R1,-(SP)        ;SAVE R1-3
016766' 010246                           MOV     R2,-(SP)
016770' 010346                           MOV     R3,-(SP)
016772' 042701  177400                   BIC     #^C377,R1       ;CLEAR HIGH BYTE
016776' 074105                           XOR     R1,R5           ;MERGE NEW BYTE WITH OLD CRC
017000' 012702  166670                   MOV     #POLYH,R2       ;GET CRC POLYNOMIAL HIGH WORD
017004' 012703  101440                   MOV     #POLYL,R3       ;GET CRC POLYNOMIAL LOW WORD
017010' 012701  000010                   MOV     #8.,R1          ;LOOP COUNT
017014' 000241                   1$:     CLC                     ;CLEAR THE CARRY
017016' 006004                           ROR     R4              ;SHIFT RIGHT THE CRC
017020' 006005                           ROR     R5              ;32 BITS WORTH
017022' 103002                           BCC     2$              ;SKIP IF BIT 0 NOT SET
017024' 074204                           XOR     R2,R4           ;EXCLUSIVE OR IN THE POLY
017026' 074305                           XOR     R3,R5           ;BOTH HIGH AND LOW WORDS
017030' 077107                   2$:     SOB     R1,1$           ;AND LOOP ON ALL 8 BITS
017032' 012603                           MOV     (SP)+,R3        ;RESTORE REGISTERS
017034' 012602                           MOV     (SP)+,R2
017036' 012601                           MOV     (SP)+,R1
017040' 000207                           RTS     PC              ;RETURN
```

```

                                 .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
                                 ;
                                 ;*******************************************************************

 017042'                         CRC16::
 017042' 010046                          MOV     R0,-(SP)                ; SAVE R0
 017044' 010146                          MOV     R1,-(SP)                ; SAVE R1
 017046' 010246                          MOV     R2,-(SP)                ; SAVE R2
 017050' 010346                          MOV     R3,-(SP)                ; SAVE R3
 017052' 010546                          MOV     R5,-(SP)                ; SAVE R5
 017054' 010203                          MOV     R2,R3                   ; R3 = BYTE COUNT
 017056' 010002                          MOV     R0,R2                   ; R2 = ADDRESS OF DATA BLOCK
 017060' 012705  120001                  MOV     #POLY16,R5              ; CRC POLYNOMIAL
 017064' 112200                  1$:     MOVB    (R2)+,R0                ; GET NEXT BYTE
 017066' 042700  177400                  BIC     #^C377,R0               ; CLEAR HIGH BYTE
 017072' 074004                          XOR     R0,R4                   ; MERGE BYTE WITH OLD CRC
 017074' 012701  000010                  MOV     #8.,R1                  ; LOOP COUNT
 017100' 000241                  2$:     CLC                             ; CLEAR CARRY
 017102' 006004                          ROR     R4                      ; SHIFT RIGHT THE CRC
 017104' 103001                          BCC     3$                      ; SKIP IF BIT ZERO NOT SET
 017106' 074504                          XOR     R5,R4                   ; EXCLUSIVE OR IN THE POLY
 017110' 077105                  3$:     SOB     R1,2$                   ; AND LOOP ON ALL 8 BITS
 017112' 077314                          SOB     R3,1$
 017114' 012605                          MOV     (SP)+,R5                ; RESTORE R5
 017116' 012603                          MOV     (SP)+,R3                ; RESTORE R3
 017120' 012602                          MOV     (SP)+,R2                ; RESTORE R2
 017122' 012601                          MOV     (SP)+,R1                ; RESTORE R1
 017124' 012600                          MOV     (SP)+,R0                ; RESTORE R0
 017126' 000207                          RTS     PC                      ; AND RETURN

```

```

                                    .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
                                    ;
                                    ;*******************************************************************

 017130'                            HEXDPA::
 017130' 010046                             MOV     R0,-(SP)                ; SAVE R0
 017132' 010346                             MOV     R3,-(SP)                ; SAVE R3
 017134' 010546                             MOV     R5,-(SP)                ; SAVE R5
                                    ;
 017136' 012700  000006                     MOV     #6,R0                   ; DO LOOP = 6 BYTES
 017142' 012703  053033'                    MOV     #DEFADR,R3              ; POINT TO ASCII MESSAGE
 017146' 012705  052750'                    MOV     #DPA,R5                 ; POINT TO DEFAULT PHYSICAL ADDR
                                    ;
 017152' 112537  052766'            10$:    MOVB    (R5)+,HEXDAT            ; LOAD BYTE FOR CONVERSION
 017156' 004737  017212'                    JSR     PC,HEXH                 ; CONVERT HIGH NIBBLE
 017162' 113723  052767'                    MOVB    HEXVAL,(R3)+            ; LOAD INTO ASCII MESSAGE
 017166' 004737  017250'                    JSR     PC,HEXL                 ; CONVERT LOW NIBBLE
 017172' 113723  052767'                    MOVB    HEXVAL,(R3)+            ; LOAD INTO ASCII MESSAGE
 017176' 105723                             TSTB    (R3)+                   ; SKIP OVER HYPHEN IN MESSAGE
 017200' 077014                             SOB     R0,10$                  ; LOOP TIL  ALL 6 BYTES ARE DONE
                                    ;
 017202' 012605                             MOV     (SP)+,R5                ; RESTORE R5
 017204' 012603                             MOV     (SP)+,R3                ; RESTORE R3
 017206' 012600                             MOV     (SP)+,R0                ; RESTORE R0
 017210' 000207                             RTS     PC                      ; AND RETURN
```

```

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

 017212'                        HEXH::
 017212' 010146                         MOV     R1,-(SP)                ; SAVE R1
                                ;
 017214' 013701  052766'                MOV     HEXDAT,R1               ; LOAD DATA FOR CONVERSION
 017220' 042701  177417                 BIC     #177417,R1              ; MASK HIGH NIBBLE
                                ;
 017224' 006201                         ASR     R1                      ; SHIFT RIGHT
 017226' 006201                         ASR     R1
 017230' 006201                         ASR     R1
 017232' 006201                         ASR     R1
                                ;
 017234' 062701  053057'                ADD     #HEXTBL,R1              ; GET INDEX INTO HEXTBL
 017240' 111137  052767'                MOVB    (R1),HEXVAL             ; AND LOAD HEXVAL
                                ;
 017244' 012601                         MOV     (SP)+,R1                ; RESTORE R1
 017246' 000207                         RTS     PC                      ; AND RETURN
```

```

                                 ;************************************************************************
                                 ;
                                 ;       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
                                 ;
                                 ;************************************************************************

017250'                          HEXL::
017250' 010146                          MOV     R1,-(SP)                 ; SAVE R1
                                 ;
017252' 013701  052766'                 MOV     HEXDAT,R1                ; LOAD DATA FOR CONVERSION
017256' 042701   177760                 BIC     #177760,R1               ; MASK LOW NIBBLE
                                 ;
017262' 062701  053057'                 ADD     #HEXTBL,R1               ; GET INDEX INTO HEXTBL
017266' 111137  052767'                 MOVB    (R1),HEXVAL              ; AND LOAD HEXVAL
                                 ;
017272' 012601                          MOV     (SP)+,R1                 ; RESTORE R1
017274' 000207                          RTS     PC                       ; AND RETURN
```

```

                                 .SBTTL  TURN ON THE CLOCK

                                 ;**********************************************************************
                                 ;
                                 ;THIS ROUTINE TURNS ON THE CLOCK
                                 ;
                                 ;**********************************************************************

017276'                          TIMON:: SETPRI  #PRI05                 ;SET PROCESSOR PRIORITY TO 5
017276' 012700  000240                                                                        MOV     #PRI05,R0
017302' 104441                                                                                TRAP    C$SPRI
017304' 012777  000100  160762           MOV     #IE,@CLKCSR            ;ENABLE CLOCK INTERRUPTS
017312' 000207                           RTS     PC


                                 ;**********************************************************************
                                 ;
                                 ;THIS ROUTINE TURNS THE CLOCK OFF
                                 ;
                                 ;**********************************************************************

017314' 005077  160754           TIMOFF::CLR     @CLKCSR                ;CLEAR INTERRUPT ENABLE
017320'                                  SETPRI  #PRI07                 ;PUT PRIORITY BACK UP
017320' 012700  000340                                                                        MOV     #PRI07,R0
017324' 104441                                                                                TRAP    C$SPRI
017326' 000207                           RTS     PC
```

```
                                    .SBTTL  CHECK FOR DONE INTERRUPT

                                    ;*********************************************************************************
                                    ;
                                    ;FUNCTIONAL DESCRIPTION:
                                    ;       ROUTINE TO WAIT FOR THE 'DONE INTERRUPT' BIT TO SET IN PCSRO
                                    ;
                                    ;INPUTS: NONE
                                    ;
                                    ;IMPLICIT INPUTS: METER
                                    ;
                                    ;OUTPUTS: C BIT SET IN PSW IF 'DNI' NOT SET
                                    ;         C BIT CLEAR IN PSW IF 'DNI' SET
                                    ;
                                    ;SUBORDINATE ROUTINES USED: TIMON, TIMOFF
                                    ;
                                    ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK INTERRUPTS ARE ENABLED
                                    ;
                                    ;CALLING SEQUENCE: PUT NUMBER OF CLOCK TICKS TO WAIT FOR IN--->METER
                                    ;                   JSR PC,CHKDNI
                                    ;
                                    ;****************************************************************************

017330' 004737  017276'             CHKDNI::JSR     PC,TIMON                   ;TURN ON THE LINE CLOCK
017334' 032777  004000  160774  10$:    BIT     #DNI,@PCSRO                ;IS 'DNI' SET?
017342' 001010                          BNE     20$                        ;YES
017344'                                 BREAK                              ;RETURN TO THE DRS FOR A MOMENT
017344' 104422                                                                                                 TRAP     C$BRK
017346' 005737  000332'                 TST     METER                      ;HAS THE TIME EXPIRED?
017352' 001370                          BNE     10$                        ;NO, KEEP WAITING
017354' 004737  017314'                 JSR     PC,TIMOFF                  ;TURN THE CLOCK OFF
017360' 000261                          SEC                                ;INDICATE 'DNI' DID NOT SET
017362' 000403                          BR      30$                        ;LEAVE
017364' 004737  017314'         20$:    JSR     PC,TIMOFF                  ;STOP THE CLOCK
017370' 000241                          CLC                                ;INDICATE 'DNI' SET
017372' 000207                  30$:    RTS     PC

```

```
                                        .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 SUCCESFULLY
                                        ;
                                        ;SUBORDINATE ROUTINES CALLED: NONE
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED
                                        ;CALLING SEQUENCE: JSR PC,CLRDNI
                                        ;
                                        ;**********************************************************

 017374' 012777  004000  160734  CLRDNI::MOV     #DNI,@PCSRO              ;CLEAR 'DNI' BIT
 017402' 032777  004000  160726          BIT     #DNI,@PCSRO              ;DID IS CLEAR?
 017410' 001402                          BEQ     10$                      ;YES
 017412' 000261                          SEC                              ;NO, INDICATE ERROR
 017414' 000401                          BR      20$
 017416' 000241                  10$:    CLC                              ;YES, INDICATE SUCCESS
 017420' 000207                  20$:    RTS     PC
```

```
                                     .SBTTL  CLEAR OUTSTANDING INTERRUPT BITS

                                     ;**********************************************************************8
                                     ;
                                     ;FUNCTIONAL DESCRIPTION:
                                     ;       ROUTINE TC CLEAR ALL INTERRUPT BITS IN PCSR0
                                     ;
                                     ;INPUTS: NONE
                                     ;
                                     ;IMPLICIT INPUTS: NONE
                                     ;
                                     ;OUTPUTS: C BIT SET IF UNABLE TO CLEAR A INTERRUPT 3IT
                                     ;         C BIT CLEARED IF SUCCESSFUL
                                     ;
                                     ;SUBORDINATE ROUTINES USED: NONE
                                     ;
                                     ;FUNCTIONAL SIDE EFFECTS: ANY OUTSTANDING INTERRUPT IS CLEARED
                                     ;
                                     ;CALLING SEQUENCE: JSR PC,CLRINT
                                     ;
                                     ;*******************************************************************

017422' 032777  100000  160706  CLRINT: BIT     #SERI,@PCSR0            ;IS 'SERI' BIT SET?
017430' 001413                          BEQ     10$                     ;NO
017432' 012777  100000  160676          MOV     #SERI,@PCSR0            ;WRITE ONE TO CLEAR 'SERI'
017440' 032777  100000  160670          BIT     #SERI,@PCSR0            ;DID IT CLEAR?
017446' 001404                          BEQ     10$                     ;YES
017450' 012737  000734' 000310'         MOV     #$SERI,BITNAM           ;NO GET POINTER TO BIT NAME STRING
017456' 000531                          BR      70$                     ;LEAVE
017460' 032777  040000  160650  10$:    BIT     #PCEI,@PCSR0            ;IS 'PCEI' BIT SET?
017466' 001413                          BEQ     20$                     ;NO
017470' 012777  040000  160640          MOV     #PCEI,@PCSR0            ;YES, WRITE ONE TO CLEAR 'PCEI'
017476' 032777  040000  160632          BIT     #PCEI,@PCSR0            ;DID IT CLEAR?
017504' 001404                          BEQ     20$                     ;YES
017506' 012737  000745' 000310'         MOV     #$PCEI,BITNAM           ;NO, GET POINTER TO BIT NAME STRING
017514' 000512                          BR      70$                     ;LEAVE
017516' 032777  020000  160612  20$:    BIT     #RXI,@PCSR0             ;IS 'RXI' BIT SET?
017524' 001413                          BEQ     30$                     ;NO
017526' 012777  020000  160602          MOV     #RXI,@PCSR0             ;YES, WRITE ONE TO CLEAR 'RXI'
017534' 032777  020000  160574          BIT     #RXI,@PCSR0             ;DID IT CLEAR?
017542' 001404                          BEQ     30$                     ;YES
017544' 012737  000756' 000310'         MOV     #$RXI,BITNAM            ;NO, GET POINTER TO BIT NAME STRING
017552' 000473                          BR      70$                     ;LEAVE
017554' 032777  010000  160554  30$:    BIT     #TXI,@PCSR0             ;IS 'TXI' BIT SET?
017562' 001413                          BEQ     40$                     ;NO
017564' 012777  010000  160544          MOV     #TXI,@PCSR0             ;YES, WRITE ONE TO CLEAR 'TXI'
017572' 032777  010000  160536          BIT     #TXI,@PCSR0             ;DID IT CLEAR?
017600' 001404                          BEQ     40$                     ;YES
017602' 012737  000766' 000310'         MOV     #$TXI,BITNAM            ;NO, GET POINTER TO BIT NAME STRING
017610' 000454                          BR      70$                     ;LEAVE
017612' 032777  004000  160516  40$:    BIT     #DNI,@PCSR0             ;IS 'DNI' BIT SET?
017620' 001413                          BEQ     50$                     ;NO
017622' 012777  004000  160506          MOV     #DNI,@PCSR0             ;YES, WRITE ONE TO CLEAR 'DNI'
017630' 032777  004000  160500          BIT     #DNI,@PCSR0             ;DID IT CLEAR?
017636' 001404                          BEQ     50$                     ;YES
017640' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;NO, GET POINTER TO BIT NAME STRING
```

```
017646' 000435                          BR      70$                     ;LEAVE
017650' 032777  002000  160460  50$:    BIT     #RCEI,@PCSRO            ;IS 'RCEI' BIT SET?
017656' 001413                          BEQ     60$                     ;NO
017660' 012777  002000  160450          MOV     #RCEI,@PCSRO            ;WRITE ONE TO CLEAR 'RCEI'
017666' 032777  002000  160442          BIT     #RCEI,@PCSRO            ;DID IT CLEAR?
017674' 001404                          BEQ     60$                     ;YES
017676' 012737  001006' 000310'         MOV     #$RCEI,BITNAM           ;NO, GET POINTER TO BIT NAME STRING
017704' 000416                          BR      70$                     ;LEAVE
017706' 032777  000400  160422  60$:    BIT     #FATI,@PCSRO            ;IS 'FATI' BIT SET?
017714' 001426                          BEQ     80$                     ;NO
017716' 012777  000400  160412          MOV     #FATI,@PCSRO            ;WRITE ONE TO CLEAR 'FATI'
017724' 032777  000400  160404          BIT     #FATI,@PCSRO            ;DID IS CLEAR?
017732' 001417                          BEQ     80$                     ;YES
017734' 012737  001017' 000310'         MOV     #$FATI,BITNAM           ;NO, GET POINTER TO BIT NAME STRING
017742'                         70$:    PRINTF  #FORM62,BITNAM          ;PRINT ERROR MESSAGE
017742' 013746  000310'                                                                         MOV     BITNAM,-(SP)
017746' 012746  010754'                                                                         MOV     #FORM62,-(SP)
017752' 012746  000002                                                                          MOV     #2,-(SP)
017756' 010600                                                                                  MOV     SP,R0
017760' 104417                                                                                  TRAP    C$PNTF
017762' 062706  000006                                                                          ADD     #6,SP
017766' 000261                          SEC                             ;INDICATE ERROR TO CALLER
017770' 000401                          BR      100$                    ;LEAVE
017772' 000241                  80$:    CLC                             ;INDICATE SUCCESS
017774' 000207                  100$:   RTS     PC
```

```
                                        .SBTTL  CHECK MICROMONITOR

                                        ;************************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: PCSR1
                                        ;
                                        ;OUTPUTS: C BIT SET IN PSW IF TIMEOUT WAITING FOR MICROMONITOR
                                        ;         C BIT CLEAR IN PSW IF MICROCODE IN MICROMONITOR
                                        ;
                                        ;SUBORDINATE ROUTINES CALLED: TIMON, TIMOFF
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK INTERRUPTS ARE ENABLED
                                        ;
                                        ;CALLING SEQUENCE: JSR PC,CHKMON
                                        ;
                                        ;***********************************************************************

017776' 012737  000077  000332' CHKMON::MOV     #1*SECOND,METER         ;TIMEOUT PERIOD IS 1 SECOND
020004' 004737  017276'                 JSR     PC,TIMON                ;TURN ON THE CLOCK
020010' 122777  000001  160322  10$:    CMPB    #INMON,@PCSR1           ;IS THE MICROCODE IN THE MICROMONITOR?
020016' 001410                          BEQ     20$                     ;YES
020020'                                 BREAK                           ;RETURN TO DRS FOR A MOMENT
020020' 104422                                                                          TRAP    C$BRK
020022' 005737  000332'                 TST     METER                   ;HAS THE TIMER EXPIRED?
020026' 001370                          BNE     10$                     ;NOT YET, KEEP CHECKING MICROCODE
020030' 004737  017314'                 JSR     PC,TIMOFF               ;TIMER HAS EXPIRED TURN OFF THE TIMER
020034' 000261                          SEC                             ;INDICATE ERROR TO CALLER
020036' 000403                          BR      30$                     ;LEAVE
020040' 004737  017314'         20$:    JSR     PC,TIMOFF               ;STOP THE CLOCK
020044' 000241                          CLC                             ;INDICATE TO CALLER MICROCODE IS IN
                                                                        ;MICROMONITOR
020046' 000207                  30$:    RTS     PC
```

```
                                        .SBTTL  CHECK INTERRUPT ERROR BITS

                                        ;************************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO CHECK FOR ANY T11 INTERRUPT ERROR BITS IN PCSR0
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: PCSR0
                                        ;
                                        ;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 PCSR0
                                        ;
                                        ;SUBORDINATE ROUTINES CALLED:  NONE
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: ANY INTERRUPT BIT SET IS CLEARED
                                        ;
                                        ;CALLING SEQUENCE: JSR PC,CHKINT
                                        ;
                                        ;*******************************************************************

020050' 017737  160262  000666' CHKINT::MOV     @PCSR0,ERRINT           ;GET PCSR0 CONTENTS
020056' 032737  170000  000666'         BIT     #NPRERR!NXMERR!UNIERR!PARERR,ERRINT ;ANY INTERRUPT ERRORS SET?
020064' 001405                          BEQ     10$                     ;NO
020066' 012777  170000  160242          MOV     #NPRERR!NXMERR!UNIERR!PARERR,@PCSR0 ;CLEAR ANY ERROR INTERRUPT
020074' 000261                          SEC                             ;INDICATE ERROR
020076' 000401                          BR      20$
020100' 000241                  10$:    CLC                             ;INDICATE NO ERRORS
020102' 000207                  20$:    RTS     PC
```

```



                                                .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

020104' 012777  000040  160224  REUNA:: MOV     #RSET,@PCSR0            ;RESET DEUNA BACK TO OPERATIONAL MICRO
020112' 012737  000770  000332'         MOV     #10*SECOND,METER        ;PUT SOME TIME ON THE METER
020120' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR 'DNI'
020124' 103035                          BCC     10$                     ;OK
                                                                        ;ERROR DNI NOT SET AFTER RESET!
020126' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
020134' 012737  001275' 000312'         MOV     #$NSET,BITSTA
020142' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
020150' 005037  000304'                 CLR     CSRNUM
020154'                                 PRINTF  #FORM9,CSRNUM,BITNAM,BITSTA,PWHEN
020154' 013746  000314'                                                                 MOV     PWHEN,-(SP)
020160' 013746  000312'                                                                 MOV     BITSTA,-(SP)
020164' 013746  000310'                                                                 MOV     BITNAM,-(SP)
020170' 013746  000304'                                                                 MOV     CSRNUM,-(SP)
020174' 012746  004732'                                                                 MOV     #FORM9,-(SP)
020200' 012746  000005                                                                  MOV     #5,-(SP)
020204' 010600                                                                          MOV     SP,R0
020206' 104417                                                                          TRAP    C$PNTF
020210' 062706  000014                                                                  ADD     #14,SP
020214' 000261                          SEC                             ;INDICATE ERROR
020216' 000414                          BR      20$
020220' 004737  017374'         10$:    JSR     PC,CLRDNI               ;GO CLEAR DNI
020224' 103011                          BCC     20$                     ;OK
020226'                                 PRINTF  #FORM8
020226' 012746  004650'                                                                 MOV     #FORM8,-(SP)
020232' 012746  000001                                                                  MOV     #1,-(SP)
020236' 010600                                                                          MOV     SP,R0
020240' 104417                                                                          TRAP    C$PNTF
020242' 062706  000004                                                                  ADD     #4,SP
020246' 000261                          SEC
```

```
  4488  020250' 000207                 20$:     RTS     PC
```

```
                                      .SBTTL  LOAD MICROMODULE

                                      ;**********************************************************************
                                      ;
                                      ;THIS SUBROUTINE LOADS THE MICROMODULE FOR TESTS THAT REQUIRE CUSTOM
                                      ;MICROCODE.
                                      ;
                                      ;**********************************************************************

020252' 012737  000326' 000320'  LODMIC: MOV     #MICRO,MICMOD           ;POINT TO MICRO MODULE NAME
020260' 004737  020104'                  JSR     PC,REUNA                ;GO START OPERATION MICROCODE
020264' 103015                           BCC     5$                      ;OK
020266'                                  PRINTF  #UNLOD,MICMOD           ;PRINT ERROR MESSAGE
020266' 013746  000320'                                                                          MOV     MICMOD,-(SP)
020272' 012746  001515'                                                                          MOV     #UNLOD,-(SP)
020276' 012746  000002                                                                           MOV     #2,-(SP)
020302' 010600                                                                                   MOV     SP,R0
020304' 104417                                                                                   TRAP    C$PNTF
020306' 062706  000006                                                                           ADD     #6,SP
020312' 000261                           SEC                             ;INDICATE ERROR OCCURRED
020314' 000137  021106'                  JMP     200$                    ;LEAVE
020320' 023727  000326' 000101   5$:     CMP     MICRO,#'A                       ;LOAD MICROCODE MODULE A?
020326' 001007                           BNE     10$                     ;NO
                                                                         ;YES
020330' 013737  000000G 000614'          MOV     MICASZ,UDBB             ;SIZE OF MODULE A
020336' 012737  000000G 000616'          MOV     #MICROA,UDBB+2          ;BASE ADDRESS OF MODULE A
020344' 000475                           BR      70$

020346' 023727  000326' 000102   10$:    CMP     MICRO,#'B                       ;LOAD MICROCODE MODULE B?
020354' 001007                           BNE     20$                     ;NO
                                                                         ;YES
020356' 013737  000000G 000614'          MOV     MICBSZ,UDBB             ;SIZE OF MODULE B
020364' 012737  000000G 000616'          MOV     #MICROB,UDBB+2          ;BASE ADDRESS OF MODULE B
020372' 000462                           BR      70$

020374' 023727  000326' 000103   20$:    CMP     MICRO,#'C                       ;LOAD MICROCODE MODULE C?
020402' 001007                           BNE     30$                     ;NO
                                                                         ;YES
020404' 013737  000000G 000614'          MOV     MICCSZ,UDBB             ;SIZE OF MODULE C
020412' 012737  000000G 000616'          MOV     #MICROC,UDBB+2          ;BASE ADDRESS OF MODULE C
020420' 000447                           BR      70$

020422' 023727  000326' 000104   30$:    CMP     MICRO,#'D                       ;LOAD MICROCODE MODULE D?
020430' 001007                           BNE     40$                     ;NO
                                                                         ;YES
020432' 013737  000000G 000614'          MOV     MICDSZ,UDBB             ;SIZE OF MODULE D
020440' 012737  000000G 000616'          MOV     #MICROD,UDBB+2          ;BASE ADDRESS OF MODULE D
020446' 000434                           BR      70$

020450' 023727  000326' 000105   40$:    CMP     MICRO,#'E                       ;LOAD MICROCODE MODULE E?
020456' 001007                           BNE     50$                     ;NO
                                                                         ;YES
020460' 013737  000000G 000614'          MOV     MICESZ,UDBB             ;SIZE OF MODULE E
020466' 012737  000000G 000616'          MOV     #MICROE,UDBB+2          ;BASE ADDRESS OF MODULE E
020474' 000421                           BR      70$
```

```

020476' 023727  000326' 000106  50$:    CMP     MICRO,#'F                       ;LOAD MICROCODE MODULE F?
020504' 001007                          BNE     60$                     ;NO
                                                                        ;YES
020506' 013737  000000G 000614'         MOV     MICFSZ,UDBB             ;SIZE OF MODULE F
020514' 012737  000000G 000616'         MOV     #MICROF,UDBB+2          ;BASE ADDRESS OF MODULE F
020522' 000406                          BR      70$

020524' 013737  000000G 000614' 60$:    MOV     MICGSZ,UDBB             ;SIZE OF MODULE G
020532' 012737  000000G 000616'         MOV     #MICROG,UDBB+2          ;BASE ADDRESS OF MODULE G

020540' 005037  000620'         70$:    CLR     UDBB+4
020544' 012737  010000  000622'         MOV     #WCSADR+<WCSSIZ/2>,UDBB+6 ;LOAD INTO TOP HALF OF WCS

                                                                        ;SETUP PCB
020552' 012737  000021  000604'         MOV     #LIM,PCBB               ;'LOAD INTERNAL MEMORY' FUNCTION
020560' 012737  000614' 000606'         MOV     #UDBB,PCBB+2            ;SET ADDRESS OF UDBB
020566' 005037  000610'                 CLR     PCBB+4
020572' 012777  000604' 157542          MOV     #PCBB,@PCSR2            ;TELL DEUNA WHERE PCBB IS
020600' 005077  157540                  CLR     @PCSR3

020604' 004737  017422'                 JSR     PC,CLRINT               ;CLEAR ANY OUTSTANDING INTERRUPT BITS
020610' 103014                          BCC     75$                     ;OK
020612'                                 PRINTF  #UNLOD,MICMOD           ;CAN'T CONTINUE WITH INTERRUPT BITS SET
020612' 013746  000320'                                                                 MOV     MICMOD,-(SP)
020616' 012746  001515'                                                                 MOV     #UNLOD,-(SP)
020622' 012746  000002                                                                  MOV     #2,-(SP)
020626' 010600                                                                          MOV     SP,R0
020630' 104417                                                                          TRAP    C$PNTF
020632' 062706  000006                                                                  ADD     #6,SP
020636' 000261                          SEC
020640' 000522                          BR      200$

020642' 012777  000001  157466  75$:    MOV     #GETPCB,@PCSR0          ;ISSUE 'GET PCB' PORT COMMAND
020650' 012737  000770  000332'         MOV     #10*SECOND,METER        ;SETUP TIMER
020656' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR 'DNI' TO SET
020662' 103014                          BCC     80$                     ;OK
                                                                        ;ERROR DNI NOT SET!
020664'                                 PRINTF  #UNLOD,MICMOD           ;PRINT MESSAGE
020664' 013746  000320'                                                                 MOV     MICMOD,-(SP)
020670' 012746  001515'                                                                 MOV     #UNLOD,-(SP)
020674' 012746  000002                                                                  MOV     #2,-(SP)
020700' 010600                                                                          MOV     SP,R0
020702' 104417                                                                          TRAP    C$PNTF
020704' 062706  000006                                                                  ADD     #6,SP
020710' 000261                          SEC                             ;INDICATE ERROR
020712' 000475                          BR      200$
020714'                         80$:
020714' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR 'DNI'
020720' 103014                          BCC     90$                     ;OK
                                                                        ;ERROR 'DNI' NOT CLEAR!
020722'                                 PRINTF  #UNLOD,MICMOD           ;PRINT MESSAGE
020722' 013746  000320'                                                                 MOV     MICMOD,-(SP)
020726' 012746  001515'                                                                 MOV     #UNLOD,-(SP)
020732' 012746  000002                                                                  MOV     #2,-(SP)
020736' 010600                                                                          MOV     SP,R0
```

```
020740' 104417                                                                                  TRAP    C$PNTF
020742' 062706  000006                                                                          ADD     #6,SP
020746' 000261                          SEC                             ;INDICATE ERROR
020750' 000456                          BR      200$
020752'                         90$:
020752' 012777  000002  157356          MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
020760' 012737  000770  000332'         MOV     #10*SECOND,METER        ;SETUP TIMER
020766' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR 'DNI'
020772' 103014                          BCC     100$                    ;OK
                                                                        ;ERROR 'DNI' NOT SET:
020774'                                 PRINTF  #UNLOD,MICMOD           ;PRINT MESSAGE
020774' 013746  000320'                                                                         MOV     MICMOD,-(SP)
021000' 012746  001515'                                                                         MOV     #UNLOD,-(SP)
021004' 012746  000002                                                                          MOV     #2,-(SP)
021010' 010600                                                                                  MOV     SP,R0
021012' 104417                                                                                  TRAP    C$PNTF
021014' 062706  000006                                                                          ADD     #6,SP
021020' 000261                          SEC                             ;INDICATE ERROR
021022' 000431                          BR      200$
021024'                         100$:
021024' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR 'DNI'
021030' 103014                          BCC     110$                    ;OK
                                                                        ;ERROR 'DNI' NOT CLEAR
021032'                                 PRINTF  #UNLOD,MICMOD           ;PRINT ERROR MESSAGE
021032' 013746  000320'                                                                         MOV     MICMOD,-(SP)
021036' 012746  001515'                                                                         MOV     #UNLOD,-(SP)
021042' 012746  000002                                                                          MOV     #2,-(SP)
021046' 010600                                                                                  MOV     SP,R0
021050' 104417                                                                                  TRAP    C$PNTF
021052' 062706  000006                                                                          ADD     #6,SP
021056' 000261                          SEC                             ;INDICATE ERROR
021060' 000412                          BR      200$
021062'                         110$:
021062' 012737  000001  000604'         MOV     #LASM,PCBB              ;LOAD PCBB WITH 'LOAD AND START' FUNCTION CODE
021070' 012737  010000  000606'         MOV     #WCSADR+<WCSSIZ/2>,PCBB+2 ;STARTING MICROADDRESS
021076' 012777  000002  157232          MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
021104' 000241                          CLC                             ;INDICATE SUCCESS

021106' 000207                  200$:   RTS     PC                      ;RETURN

```

```
                                 .TITLE MISCELLANEOUS SECTIONS
                                 .SBTTL  REPORT CODING SECTION
021110'                                  BGNRPT
021110'                                                                      L$RPT::
021110'                                  EXIT    RPT
021110' 000167                                                                        .WORD   J$JMP
021112' 000000                                                                        .WORD   L10064-2-.
021114'                                  ENDRPT
021114'                                                                      L10064:
021114' 104425                                                                        TRAP    C$RPT


                                 .SBTTL  PROTECTION TABLE

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

021116'                                  BGNPROT
021116'                                                                      L$PROT::

021116' 177777                           -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
021120' 177777                           -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
021122' 177777                           -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

021124'                                  ENDPROT

```

```
                                      .SBTTL  INITIALIZE SECTION

                                      ;++
                                      ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                      ; AT THE BEGINNING OF EACH PASS.
                                      ;--

021124'                                     BGNINIT
021124'                                                                            L$INIT::



021124'                                     READEF  #EF.CONTINUE           ;WAS A CONTINUE COMMAND ENTERED
021124' 012700  000036                                                                         MOV     #EF.CONTINUE,R0
021130' 104447                                                                                 TRAP    C$REFG
021132'                                     BCOMPLETE       30$            ;YES, LEAVE INIT CODE
021132' 103555                                                                                 BCS     30$
021134'                                     READEF  #EF.PWR                ;WAS THERE A POWER FAILURE?
021134' 012700  000034                                                                         MOV     #EF.PWR,R0
021140' 104447                                                                                 TRAP    C$REFG
021142'                                     BCOMPLETE       30$            ;YES, LEAVE INIT CODE
021142' 103551                                                                                 BCS     30$
021144'                                     READEF  #EF.NEW                ;IS THIS A NEW PASS?
021144' 012700  000035                                                                         MOV     #EF.NEW,R0
021150' 104447                                                                                 TRAP    C$REFG
021152'                                     BNCOMPLETE      10$            ;NO SKIP THE NEW PASS TIME STUFF
021152' 103072                                                                                 BCC     10$
021154'                                     READEF  #EF.START              ;IS THIS THE VERY FIRST TIME THOUGH?
021154' 012700  000040                                                                         MOV     #EF.START,R0
021160' 104447                                                                                 TRAP    C$REFG
021162'                                     BNCOMPLETE      6$             ;NO SKIP THE FIRST TIME STUFF
021162' 103061                                                                                 BCC     6$
021164' 012737  000001  000672'             MOV     #1,FRSTIM              ;SET THE FIRST TIME FLAG
021172'                                     MEMORY  FRESIZ                 ;GET FREE MEMORY INFO
021172' 104431                                                                                 TRAP    C$MEM
021174' 010037  000322'                                                                        MOV     R0,FRESIZ
021200' 013737  000322' 000324'             MOV     FRESIZ,FREMEM          ;SIZE OF FREE MEMORY IN FRESIZ
021206' 062737  000002  000324'             ADD     #2,FREMEM              ;START OF FREE MEMORY IN FREMEM
021214'                                     CLOCK   L,R1                   ;GET LINE CLOCK INFO
021214' 012700  000114                                                                         MOV     #'L,R0
021220' 104462                                                                                 TRAP    C$CLCK
021222' 010001                                                                                 MOV     R0,R1
021224'                                     BCOMPLETE       1$
021224' 103411                                                                                 BCS     1$
021226'                                     PRINTF  #NOCLK                 ;ERROR MESSAGE
021226' 012746  001566'                                                                        MOV     #NOCLK,-(SP)
021232' 012746  000001                                                                         MOV     #1,-(SP)
021236' 010600                                                                                 MOV     SP,R0
021240' 104417                                                                                 TRAP    C$PNTF
021242' 062706  000004                                                                         ADD     #4,SP
021246' 000504                              BR       20$                   ;CANNOT CONTINUE
021250' 012137  000274'             1$:     MOV     (R1)+,CLKCSR           ;LINE CLOCK CSR
021254' 012102                              MOV     (R1)+,R2               ;GET CLOCK PRIORITY
021256' 072227  000005                      ASH     #5,R2
021262' 010237  000276'                     MOV     R2,CLKBR
021266' 012137  000300'                     MOV     (R1)+,CLKVEC           ;VECTOR
```

```
021272' 012137  000302'                   MOV     (R1)+,CLKFRE            ;FREQUENCY
021276'                                   SETVEC  CLKVEC,#CLKSRV,CLKBR    ;SETUP CLOCK INTERRUPT VECTOR
021276' 013746  000276'                                                                         MOV     CLKBR,-(SP)
021302' 012746  021674'                                                                         MOV     #CLKSRV,-(SP)
021306' 013746  000300'                                                                         MOV     CLKVEC,-(SP)
021312' 012746  000003                                                                          MOV     #3,-(SP)
021316' 104437                                                                                  TRAP    C$SVEC
021320' 062706  000010                                                                          ADD     #10,SP
021324' 000402                            BR      5$
021326' 005037  000672'           6$:     CLR     FRSTIM                  ;INDICATE NOT THE FIRST TIME THROUGH
021332' 012737  177777  000330'   5$:     MOV     #-1,UNIT                ;YES, INITIALIZE UNIT #
021340' 005237  000330'           10$:    INC     UNIT                    ;SETUP FOR NEXT UNIT
021344' 023737  000330' 000012'           CMP     UNIT,L$UNIT             ;WE TESTED ALL AVAILABLE UNITS?
021352' 003042                            BGT     20$                     ;YES, LEAVE
021354'                                   GPHARD  UNIT,R1                 ;GET P-TAB POINTER FOR THIS UNIT
021354' 013700  000330'                                                                         MOV     UNIT,R0
021360' 104442                                                                                  TRAP    C$GPHRD
021362' 010001                                                                                  MOV     R0,R1
021364'                                   BNCOMPLETE      10$             ;THIS ONE IS NOT AVAILABLE
021364' 103365                                                                                  BCC     10$
021366' 012137  000266'                   MOV     (R1)+,UNACSR            ;SAVE CSR
021372' 012137  000270'                   MOV     (R1)+,UNAVEC            ;SAVE VECTOR
021376' 012737  000240  000272'           MOV     #PRI05,UNAPRI           ;SAVE PRIORITY
021404' 013737  000266' 000336'           MOV     UNACSR,PCSR0            ;PCSR0
021412' 013737  000336' 000340'           MOV     PCSR0,PCSR1
021420' 062737  000002  000340'           ADD     #2,PCSR1                ;PCSR1
021426' 013737  000340' 000342'           MOV     PCSR1,PCSR2
021434' 062737  000002  000342'           ADD     #2,PCSR2                ;PCSR2
021442' 013737  000342' 000344'           MOV     PCSR2,PCSR3
021450' 062737  000002  000344'           ADD     #2,PCSR3                ;PCSR3
021456' 000403                            BR      30$                     ;LEAVE
021460' 005037  000672'           20$:    CLR     FRSTIM                  ;CLEAR FIRST TIME THROUGH FLAG
021464'                                   DOCLN                           ;ABORT PASS
021464' 104444                                                                                  TRAP    C$DCLN
021466'                           30$:
021466'                           ENDINIT
021466'                                                                                L10066:
021466' 104411                                                                                  TRAP    C$INIT



                                          .EVEN

```

```
                                            .SBTTL  AUTODROP SECTION

       021470'                                      BGNAUTO
       021470'                                                                                    L$AUTO::
       021470'                                      ENDAUTO
       021470'                                                                                    L10067:
       021470' 104461                                                                                     TRAP    C$AUTO

                                            .SBTTL  CLEANUP CODING SECTION

                                            ;++
                                            ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                            ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
                                            ;--
       021472'                              BGNCLN
       021472'                                                                                    L$CLEAN::
       021472' 005737  000664'                      TST     NEXMEM                  ;DOES PCSRO EXIST?
       021476' 001023                               BNE     20$                     ;NO SKIP RESET
       021500' 012777  000040  156630               MOV     #RSET,@PCSRO            ;RESTORE DEUNA TO OPERATIONAL
                                                                                    ;STATE
       021506' 012737  000770  000332'              MOV     #10*SECOND,METER                ;SETUP TIMER
       021514' 004737  017330'                      JSR     PC,CHKDNI               ;DNI SET WHEN DONE
       021520' 103010                               BCC     10$
       021522'                                      PRINTF  #CLNERR                 ;ERROR
       021522' 012746  021556'                                                                            MOV     #CLNERR,-(SP)
       021526' 012746  000001                                                                             MOV     #1,-(SP)
       021532' 010600                                                                                     MOV     SP,R0
       021534' 104417                                                                                     TRAP    C$PNTF
       021536' 062706  000004                                                                             ADD     #4,SP
       021542' 004737  017374'              10$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
       021546' 004737  017314'              20$:    JSR     PC,TIMOFF               ;TURN OFF THE LINE CLOCK        ;CC
       021552'                                      EXIT    CLN
       021552' 104432                                                                                     TRAP    C$EXIT
       021554' 000054                                                                                     .WORD   L10070-.

       021556' 047045  040445  051105       CLNERR: .ASCIZ  /%N%AERROR OCCURRED DURING DEVICE RESET%N/
       021564' 047522  020122  041517
       021572' 052503  051122  042105
       021600' 042040  051125  047111
       021606' 020107  042504  044526
       021614' 042503  051040  051505
       021622' 052105  047045     000
               021630'                              .EVEN

       021630'                              ENDCLN
       021630'                                                                                    L10070:
       021630' 104412                                                                                     TRAP    C$CLEAN
```

```
                                        .SBTTL  DROP UNIT SECTION

021632'                                         BGNDU
021632'                                                                                 L$DU::
021632'                                         EXIT    DU
021632' 000167                                                                                  .WORD   J$JMP
021634' 000000                                                                                  .WORD   L10071-2-.
021636'                                         ENDDU
021636'                                                                                 L10071:
021636' 104453                                                                                  TRAP    C$DU

                                        .SBTTL  ADD UNIT SECTION

021640'                                         BGNAU
021640'                                                                                 L$AU::
021640'                                         EXIT    AU
021640' 000167                                                                                  .WORD   J$JMP
021642' 000000                                                                                  .WORD   L10072-2-.
021644'                                         ENDAU
021644'                                                                                 L10072:
021644' 104452                                                                                  TRAP    C$AU

                                        .TITLE  GLOBAL INTERRUPT SERVICE ROUTINES

                                        .SBTTL  NON-EXISTANT MEMORY INTERRUPT SERVICE ROUTINE

                                        ;*****************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;
                                        ;       THIS ROUTINE IS ASSIGNED TO VECTOR 4 BY THE ACCESS TESTS
                                        ;       IT SETS THE NEXMEM FLAG SIGNALING THAT AN ACCESS WAS
                                        ;       ATTEMPTED ON NON-EXISTANT MEMORY.
                                        ;
                                        ;*****************************************************

021646'                                 BGNSRV  TRAP4
021646'                                                                                 TRAP4::
021646' 012737  000001  000664'                 MOV     #1,NEXMEM               ;SET FLAG
021654'                                 ENDSRV
021654'                                                                                 L10073:
021654' 000002                                                                                  RTI
```

```
                                            .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 'UNAINT:' AND A COPY OF THE
                                            ;       PSW AT THE TIME OF THE INTERRUPT IS STORED AT 'CPUPRI:'.
                                            ;
                                            ;***********************************************************
                                            ;
 021656'                                    BGNSRV  UNASRV
 021656'                                                                                    UNASRV::
 021656' 017737  156454  000670'                    MOV     @PCSRO,UNAINT           ;INDICATE UNA INTERRUPTED
 021664' 016637  000002  000674'                    MOV     2(SP),CPUPRI            ;GET CPU PRIORITY AT TIME OF INTERRUPT
 021672'                                    ENDSRV
 021672'                                                                                    L10074:
 021672' 000002                                                                                     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
                                            ;
                                            ;****************************************************************
                                            ;
 021674'                                    BGNSRV  CLKSRV
 021674'                                                                                    CLKSRV::
 021674' 005737  000332'                            TST     METER           ;HAS THE METER EXPIRED?
 021700' 001402                                     BEQ     20$             ;YES, STOP COUNTING
 021702' 005337  000332'                            DEC     METER           ;COUNT TICKS
 021706'                                    20$:
 021706'                                    ENDSRV
 021706'                                                                                    L10075:
 021706' 000002                                                                                     RTI
```

```
                                .TITLE HARDWARE TESTS

                                .SBTTL  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 SEQUENCE: 1-READ PCSRO
                                ;               2-VERIFY BITS 9 AND 4 - 0
                                ;
                                ;**************************************************************

021710'                         BGNTST
021710'                                                                             T1::
021710'                                 SETVEC  #4,#TRAP4,#PRI07         ;SETUP TIME-OUT TRAP
021710' 012746  000340                                                                      MOV     #PRI07,-(SP)
021714' 012746  021646'                                                                     MOV     #TRAP4,-(SP)
021720' 012746  000004                                                                      MOV     #4,-(SP)
021724' 012746  000003                                                                      MOV     #3,-(SP)
021730' 104437                                                                              TRAP    C$SVEC
021732' 062706  000010                                                                      ADD     #10,SP
021736'                          BGNSUB ;#1
021736'                                                                             T1.1:
021736' 104402                                                                              TRAP    C$BSUB

                                ;
                                ;CHECK TO SEE IF PCSRO EXISTS
                                ;
021740' 005037  000664'                 CLR     NEXMEM                  ;CLEAR NON-EXISTANT MEMORY FLAG
021744' 005037  000304'                 CLR     CSRNUM                  ;HOLDS WHICH PCSR WE ARE DOING
021750' 005777  156362                  TST     @PCSRO                  ;DOES PSCRO EXIST?
021754' 005737  000664'                 TST     NEXMEM
021760' 001414                          BEQ     10$                     ;YES
021762'                                 ERRDF   001,RACERR,RACMG1       ;NO, PRINT FATAL ERROR MESSAGE
021762' 104455                                                                              TRAP    C$ERDF
021764' 000001                                                                              .WORD   1
021766' 001663'                                                                             .WORD   RACERR
021770' 012546'                                                                             .WORD   RACMG1
021772'                                 CKLOOP                          ;LOOP BACK FROM HERE IF ERROR
021772' 104406                                                                              TRAP    C$CLP1
021774'                                 CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
021774' 012700  000004                                                                      MOV     #4,R0
022000' 104436                                                                              TRAP    C$CVEC
022002'                                 DODU    UNIT                    ;DROP UNIT
022002' 013700  000330'                                                                     MOV     UNIT,R0
022006' 104451                                                                              TRAP    C$DODU
022010'                                 DOCLN                           ;ABORT SUB-PASS
022010' 104444                                                                              TRAP    C$DCLN
022012'                         10$:
022012'                          ENDSUB ;#1
022012'                                                                             L10077:
022012' 104403                                                                              TRAP    C$ESUB
022014'                          BGNSUB ;#2
022014'                                                                             T1.2:
```

```
022014' 104402                                                                      TRAP    C$BSUB
                                     ;
                                     ;OK, PCSRO EXISTS NOW CHECK SOME BITS
                                     ;
022016'                                BGNSEG
022016' 104404                                                                      TRAP    C$BSEG
                                     ;
                                     ;CHECK BIT 4 = 0
                                     ;
022020' 012737  001311' 000312'              MOV     #$NCLR,BITSTA          ;TESTING FOR CLEARED BITS
022026' 012737  000004  000306'              MOV     #4,BITNUM              ;TESTING BIT 4
022034' 032777  000020  156274               BIT     #BIT4,@PCSRO           ;IS BIT 4=0?
022042' 001404                               BEQ     20$                    ;YES
022044'                                      ERRHRD  002,RACERR,RACMG2      ;NO,PRINT HARD ERROR MESSAGE
022044' 104456                                                                      TRAP    C$ERHRD
022046' 000002                                                                      .WORD   2
022050' 001663'                                                                     .WORD   RACERR
022052' 012574'                                                                     .WORD   RACMG2
022054'                              20$:
022054'                                ENDSEG
022054'                                                                     10000$:
022054' 104405                                                                      TRAP    C$ESEG
022056'                                BGNSEG
022056' 104404                                                                      TRAP    C$BSEG
                                     ;
                                     ;CHECK BIT 9 = 0
                                     ;
022060' 012737  000011  000306'              MOV     #9.,BITNUM             ;TESTING BIT 9 NOW
022066' 032777  001000  156242               BIT     #BIT9,@PCSRO           ;IS BIT 9=0?
022074' 001404                               BEQ     30$                    ;YES
022076'                                      ERRHRD  003,RACERR,RACMG2      ;NO,PRINT HARD ERROR MESSAGE
022076' 104456                                                                      TRAP    C$ERHRD
022100' 000003                                                                      .WORD   3
022102' 001663'                                                                     .WORD   RACERR
022104' 012574'                                                                     .WORD   RACMG2
022106'                              30$:
022106'                                ENDSEG
022106'                                                                     10001$:
022106' 104405                                                                      TRAP    C$ESEG
022110'                               ENDSUB ;#2
022110'                                                                     L10100:
022110' 104403                                                                      TRAP    C$ESUB
022112'                                      CLRVEC  #4                     ;FREE TRAP 4 VECTOR
022112' 012700  000004                                                              MOV     #4,R0
022116' 104436                                                                      TRAP    C$CVEC
022120'                              ENDTST
022120'                                                                     L10076:
022120' 104401                                                                      TRAP    C$ETST
```

```
                                    .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
                                    ;
                                    ;*******************************************************

022122'                             BGNTST
022122'                                                                                        T2::
022122'                                         SETVEC  #4,#TRAP4,#PRI07        ;SETUP TIMEOUT TRAP
022122'  012746  000340                                                                                    MOV     #PRI07,-(SP)
022126'  012746  021646'                                                                                   MOV     #TRAP4,-(SP)
022132'  012746  000004                                                                                    MOV     #4,-(SP)
022136'  012746  000003                                                                                    MOV     #3,-(SP)
022142'  104437                                                                                            TRAP    C$SVEC
022144'  062706  000010                                                                                    ADD     #10,SP
022150'                              BGNSUB ;#1
022150'                                                                                        T2.1:
022150'  104402                                                                                            TRAP    C$BSUB
                                    ;
                                    ;CHECK TO SEE IF PCSR1 EXISTS
                                    ;
022152'  005037  000664'                        CLR     NEXMEM                  ;CLEAR NON-EXISTANT MEMORY FLAG
022156'  012737  000001  000304'                MOV     #1,CSRNUM               ;TESTING PCSR1
022164'  005777  156150                         TST     @PCSR1                  ;DOES PCSR1 EXIST?
022170'  005737  000664'                        TST     NEXMEM
022174'  001414                                 BEQ     10$                     ;YES
022176'                                         ERRDF   004,RACERR,RACMG1       ;NO,PRINT FATAL ERROR MESSAGE
022176'  104455                                                                                            TRAP    C$ERDF
022200'  000004                                                                                            .WORD   4
022202'  001663'                                                                                           .WORD   RACERR
022204'  012546'                                                                                           .WORD   RACMG1
022206'                                         CKLOOP                          ;LOOP FROM HERE IF ERROR
022206'  104406                                                                                            TRAP    C$CLP1
022210'                                         CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
022210'  012700  000004                                                                                    MOV     #4,R0
022214'  104436                                                                                            TRAP    C$CVEC
022216'                                         DODU    UNIT                    ;DROP THE UNIT
022216'  013700  000330'                                                                                   MOV     UNIT,R0
022222'  104451                                                                                            TRAP    C$DODU
022224'                                         DOCLN                           ;ABORT SUB-PASS
022224'  104444                                                                                            TRAP    C$DCLN
022226'                             10$:
022226'                              ENDSUB ;#1
022226'                                                                                        L10102:
022226'  104403                                                                                            TRAP    C$ESUB
022230'                              BGNSUB ;#2
022230'                                                                                        T2.2:
022230'  104402                                                                                            TRAP    C$BSUB
```

```
                                  ;
                                  ;OK, PCSR1 EXISTS NOW CHECK SOME BITS
                                  ;
022232'                             BGNSEG
022232' 104404                                                                                       TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 4 = 0
                                  ;
022234' 012737  001311' 000312'           MOV     #$NCLR,BITSTA             ;TESTING CLEARED BITS
022242' 012737  000004  000306'           MOV     #4,BITNUM                 ;TESTING BIT 4
022250' 032777  000020  156062            BIT     #BIT4,@PCSR1              ;IS BIT 4=0?
022256' 001404                            BEQ     20$                       ;YES
022260'                                   ERRHRD  005,RACERR,RACMG2         ;NO, PRINT HARD ERROR MESSAGE
022260' 104456                                                                                       TRAP    C$ERHRD
022262' 000005                                                                                       .WORD   5
022264' 001663'                                                                                      .WORD   RACERR
022266' 012574'                                                                                      .WORD   RACMG2
022270'                           20$:
022270'                             ENDSEG
022270'                                                                                      10000$:
022270' 104405                                                                                       TRAP    C$ESEG
022272'                             BGNSEG
022272' 104404                                                                                       TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 5 = 0
                                  ;
022274' 012737  000005  000306'           MOV     #5,BITNUM                 ;TESTING BIT 5
022302' 032777  000040  156030            BIT     #BIT5,@PCSR1              ;IS BIT 5=0?
022310' 001404                            BEQ     30$                       ;YES
022312'                                   ERRHRD  006,RACERR,RACMG2         ;NO,PRINT HARD ERROR MESSAGE
022312' 104456                                                                                       TRAP    C$ERHRD
022314' 000006                                                                                       .WORD   6
022316' 001663'                                                                                      .WORD   RACERR
022320' 012574'                                                                                      .WORD   RACMG2
022322'                           30$:
022322'                             ENDSEG
022322'                                                                                      10001$:
022322' 104405                                                                                       TRAP    C$ESEG
022324'                             BGNSEG
022324' 104404                                                                                       TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 6 = 0
                                  ;
022326' 012737  000006  000306'           MOV     #6,BITNUM                 ;TESTING BIT 6
022334' 032777  000100  155776            BIT     #BIT6,@PCSR1              ;IS BIT 6=0?
022342' 001404                            BEQ     40$                       ;YES
022344'                                   ERRHRD  007,RACERR,RACMG2         ;NO,PRINT HARD ERROR MESSAGE
022344' 104456                                                                                       TRAP    C$ERHRD
022346' 000007                                                                                       .WORD   7
022350' 001663'                                                                                      .WORD   RACERR
022352' 012574'                                                                                      .WORD   RACMG2
022354'                           40$:
022354'                             ENDSEG
022354'                                                                                      10002$:
022354' 104405                                                                                       TRAP    C$ESEG
022356'                             BGNSEG
```

```
022356' 104404                                                                                       TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 14 = 0
                                  ;
022360' 012737  000016  000306'          MOV     #14.,BITNUM           ;TESTING BIT 14
022366' 032777  040000  155744           BIT     #BIT14,@PCSR1         ;IS BIT 14 =0?
022374' 001404                           BEQ     50$                   ;YES
022376'                                  ERRHRD  008,RACERR,RACMG2     ;NO, PRINT ERROR MESSAGE
022376' 104456                                                                                       TRAP    C$ERHRD
022400' 000010                                                                                       .WORD   8
022402' 001663'                                                                                      .WORD   RACERR
022404' 012574'                                                                                      .WORD   RACMG2
022406'                           50$:
022406'                             ENDSEG
022406'                                                                                     10003$:
022406' 104405                                                                                       TRAP    C$ESEG
022410'                             BGNSEG
022410' 104404                                                                                       TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 15 = 0
                                  ;
022412' 012737  0C0017  000306'          MOV     #15.,BITNUM           ;TESTING BIT 15
022420' 032777  100000  155712           BIT     #BIT15,@PCSR1         ;IS BIT 15 = 0?
022426' 001404                           BEQ     60$                   ;YES
022430'                                  ERRHRD  009,RACERR,RACMG2     ;NO, PRINT ERROR MESSAGE
022430' 104456                                                                                       TRAP    C$ERHRD
022432' 000011                                                                                       .WORD   9
022434' 001663'                                                                                      .WORD   RACERR
022436' 012574'                                                                                      .WORD   RACMG2
022440'                           60$:
022440'                             ENDSEG
022440'                                                                                     10004$:
022440' 104405                                                                                       TRAP    C$ESEG
022442'                            ENDSUB ;#2
022442'                                                                                     L10103:
022442' 104403                                                                                       TRAP    C$ESUB
022444'                                  CLRVEC  #4                    ;FREE VECTOR 4
022444' 012700  000004                                                                               MOV     #4,R0
022450' 104436                                                                                       TRAP    C$CVEC
022452'                           ENDTST
022452'                                                                                     L10101:
022452' 104401                                                                                       TRAP    C$ETST

```

```
                                .SBTTL  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 SEQUENCE: 1-READ PCSR2
                                ;
                                ;*****************************************************************

022454'                         BGNTST
022454'                                                                T3::
022454'                                 SETVEC  #4,#TRAP4,PRI07         ;SETUP TIMFOUT TRAP VECTOR
022454' 013746  000340                                                                  MOV     PRI07,-(SP)
022460' 012746  021646'                                                                 MOV     #TRAP4,-(SP)
022464' 012746  000004                                                                  MOV     #4,-(SP)
022470' 012746  000003                                                                  MOV     #3,-(SP)
022474' 104437                                                                          TRAP    C$SVEC
022476' 062706  000010                                                                  ADD     #10,SP
022502' 005037  000664'                 CLR     NEXMEM                  ;CLEAR NON-EXISTANT MEMORY FLAG
022506' 012737  000002  000304'         MOV     #2,CSRNUM               ;TESTING PCSR2
022514' 005777  155622                  TST     @PCSR2                  ;DOES PCSR2 EXIST?
022520' 005737  000664'                 TST     NEXMEM
022524' 001414                          BEQ     10$                     ;YES
022526'                                 ERRDF   010,RACERR,RACMG1       ;NO,PRINT FATAL ERROR MESSAGE
022526' 104455                                                                          TRAP    C$ERDF
022530' 000012                                                                          .WORD   10
022532' 001663'                                                                         .WORD   RACERR
022534' 012546'                                                                         .WORD   RACMG1
022536'                                 CKLOOP                          ;LOOP BACK FROM HERE IF ERROR
022536' 104406                                                                          TRAP    C$CLP1
022540'                                 CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
022540' 012700  000004                                                                  MOV     #4,R0
022544' 104436                                                                          TRAP    C$CVEC
022546'                                 DODU    UNIT                    ;DROP THE UNIT
022546' 013700  000330'                                                                 MOV     UNIT,R0
022552' 104451                                                                          TRAP    C$DODU
022554'                                 DOCLN                           ;ABORT SUBPASS
022554' 104444                                                                          TRAP    C$DCLN
022556'                         10$:    CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
022556' 012700  000004                                                                  MOV     #4,R0
022562' 104436                                                                          TRAP    C$CVEC

022564'                         ENDTST
022564'                                                                L10104:
022564' 104401                                                                          TRAP    C$ETST

```

```
                                        .SBTTL  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 SEQUENCE: 1-READ PCSR3
                                        ;
                                        ;*****************************************************************

022566'                                 BGNTST
022566'                                                                                 T4::
022566'                                         SETVEC  #4,#TRAP4,#PRI07        ;SETUP VECTOR FOR TIME-OUT
022566' 012746  000340                                                                          MOV     #PRI07,-(SP)
022572' 012746  021646'                                                                         MOV     #TRAP4,-(SP)
022576' 012746  000004                                                                          MOV     #4,-(SP)
022602' 012746  000003                                                                          MOV     #3,-(SP)
022606' 104437                                                                                  TRAP    C$SVEC
022610' 062706  000010                                                                          ADD     #10,SP
022614'                                   BGNSEG
022614' 104404                                                                                  TRAP    C$BSEC
022616' 005037  000664'                         CLR     NEXMEM                  ;CLEAR NON-EXISTANT MEMORY FLAG
022622' 012737  000003  000304'                 MOV     #3,CSRNUM               ;TESTING PCSR3
022630' 005777  155510                          TST     @PCSR3                  ;DOES PCSR3 EXIST?
022634' 005737  000664'                         TST     NEXMEM
022640' 001414                                  BEQ     10$                     ;YES
022642'                                         ERRDF   011,RACERR,RACMG1       ;NO,PRINT FATAL ERROR MESSAGE
022642' 104455                                                                                  TRAP    C$ERDF
022644' 000013                                                                                  .WORD   11
022646' 001663'                                                                                 .WORD   RACERR
022650' 012546'                                                                                 .WORD   RACMG1
022652'                                         CKLOOP                          ;LOOP BACK FROM HERE IF ERROR
022652' 104406                                                                                  TRAP    C$CLP1
022654'                                         CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
022654' 012700  000004                                                                          MOV     #4,R0
022660' 104436                                                                                  TRAP    C$CVEC
022662'                                         DODU    UNIT                    ;DROP UNIT
022662' 013700  000330'                                                                         MOV     UNIT,R0
022666' 104451                                                                                  TRAP    C$DODU
022670'                                         DOCLN                           ;ABORT SUB-PASS
022670' 104444                                                                                  TRAP    C$DCLN
022672'                                 10$:
022672'                                   ENDSEG
022672'                                                                               10000$:
022672' 104405                                                                                  TRAP    C$ESEG
022674'                                         CLRVEC  #4                      ;RETURN TIME-OUT TRAP
022674' 012700  000004                                                                          MOV     #4,R0
022700' 104436                                                                                  TRAP    C$CVEC
022702'                                 ENDTST
022702'                                                                               L10105:
022702' 104401                                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 5: RESET TEST

                                ;*************************************************************
                                ;
                                ;THIS TEST WILL VERIFY THE RESET STATE FOR ALL DEUNA UNIBUS REGISTERS
                                ;
                                ;TEST SEQUENCE:
                                ;       1-WRITE A 1 TO PCSR0 BIT 5
                                ;       2-READ PCSR0
                                ;        -VERIFY DNI SET
                                ;        -VERIFY INTR SET
                                ;       3-VERIFY PCSR0 BITS 15:12 AND 10:08 AND 06:00 FOR LOGICAL 0
                                ;       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
                                ;
                                ;*******************************************************************

022704'                         BGNTST
022704'                                                                 T5::
022704' 005037  000304'                 CLR     CSRNUM                  ;TESTING PCSR0 FIRST
022710' 012737  001340' 000314'         MOV     #$AFTER,PWHEN           ;SETUP PART OF ERROR MESSAGE
                                                                        ;FOR THE FOLLOWING TEST SEGMENTS
022716'                          BGNSUB ;#1
022716'                                                                 T5.1:
022716' 104402                                                                  TRAP    C$BSUB

                                ;
                                ;SET THE RESET BIT IN PCSR0 AND WAIT FOR 'DNI' TO SET
                                ;
022720' 012777  000040  155410          MOV     #RSET,@PCSR0            ;RESET DEUNA
022726' 012737  000770  000332'         MOV     #10*SECOND,METER        ;PUT SOME TIME ON THE METER
022734' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
022740' 103014                          BCC     10$                     ;OK
                                                                        ;ERROR 'DNI' DID NOT SET
022742' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
022750' 012737  001275' 000312'         MOV     #$NSET,BITSTA
022756'                                 ERRHRD  012,RSETER,MSG3         ;PRINT ERROR MESSAGE
022756' 104456                                                                  TRAP    C$ERHRD
022760' 000014                                                                  .WORD   12
022762' 001717'                                                                 .WORD   RSETER
022764' 013056'                                                                 .WORD   MSG3
022766'                                 ESCAPE  TST                     ;NO POINT IN CONTINUING TEST
022766' 104410                                                                  TRAP    C$ESCAPE
022770' 000374                                                                  .WORD   L10106-.
022772'                         10$:
022772'                          ENDSUB ;#1
022772'                                                                 L10107:
022772' 104403                                                                  TRAP    C$ESUB

022774'                          BGNSUB ;#2
022774'                                                                 T5.2:
022774' 104402                                                                  TRAP    C$BSUB

                                ;
                                ;CHECK ALL THE BITS AFFECTED BY RESET
                                ;
022776'                           BGNSEG
```

```
022776' 104404                                                                                  TRAP    C$BSEG
                                            ;
                                            ;CHECK THE INTERRUPT SUMMARY BIT = 0
                                            ;
023000' 032777  000200  155330                      BIT     #INTR,@PCSR0            ;IS THE INTERRUPT SUMMARY BIT SET?
023006' 001012                                      BNE     11$                     ;YES
                                                                                    ;ERROR, INTERRUPT SUMMARY NOT SET!
023010' 012737  001030' 000310'                     MOV     #$INTR,BITNAM           ;PREPARE ERROR MESSAGE
023016' 012737  001275' 000312'                     MOV     #$NSET,BITSTA
023024'                                             ERRHRD  013,RSETER,MSG3         ;PRINT ERROR MESSAGE
023024' 104456                                                                                  TRAP    C$ERHRD
023026' 000015                                                                                  .WORD   13
023030' 001717'                                                                                 .WORD   RSETER
023032' 013056'                                                                                 .WORD   MSG3
023034'                                     11$:
023034'                                       ENDSEG
023034'                                                                                 10000$:
023034' 104405                                                                                  TRAP    C$ESEG
                                            ;
                                            ;CHECK BITS 15:12 AND 10:08 AND 06:00 OF PCSR0 FOR 0
                                            ;
023036' 0'2701  000356'                             MOV     #BNAMT0,R1              ;POINT TO BIT NAME MNEMONICS
023042' 012702  000001                              MOV     #BIT0,R2                ;START TESTING AT LSB
023046' 012737  001311' 000312'                     MOV     #$NCLR,BITSTA           ;TESTING FOR CLEARED BITS
023054'                                     15$:
023054'                                       BGNSEG
023054' 104404                                                                                  TRAP    C$BSEG
023056' 030277  155254                              BIT     R2,@PCSR0               ;IS THIS BIT CLEARED?
023062' 001406                                      BEQ     20$                     ;YES
                                                                                    ;ERROR BIT FAILED TO CLEAR
023064' 011137  000310'                             MOV     (R1),BITNAM             ;GFT MNEMONIC FOR THIS BIT
023070'                                             ERRHRD  014,RSETER,MSG3         ;PR:NT ERROR MESSAGE
023070' 104456                                                                                  TRAP    C$ERHRD
023072' 000016                                                                                  .WORD   14
023074' 001717'                                                                                 .WORD   RSETER
023076' 013056'                                                                                 .WORD   MSG3
023100'                                     20$:
023100'                                       ENDSEG
023100'                                                                                 10001$:
023100' 104405                                                                                  TRAP    C$ESEG
023102' 005702                              21$:    TST     R2                      ;WE TESTED ALL BITS IN PCSR0?
023104' 100411                                      BM!     25$                     ;YES
023106' 006302                                      ASL     R2                      ;POINT TO NEXT BIT TO TEST
023110' 062701  000002                              ADD     #2,R1                   ;POINT TO NEXT MNEMONIC
023114' 105702                                      TSTB    R2                      ;ARE WE POINTING TO 'INTR' BIT?
023116' 100771                                      BMI     21$                     ;YES, SKIP IT
023120' 022702  004000                              CMP     #DNI,R2                 ;ARE WE POINTING TO 'DNI' BIT?
023124' 001766                                      BEQ     21$                     ;YES SKIP IT
023126' 000752                                      BR      15$                     ;CONTINUE TESTING

023130'                                     25$:
023130'                                       BGNSEG
023130' 104404                                                                                  TRAP    C$BSEG
                                            ;
                                            ;CHECK PCSR1 BITS 02:00 FOR THE READY STATE
                                            ;
```

```
023132' 012737  000001  000304'           MOV       #1,CSRNUM             ;TESTING PCSR1
023140' 017701  155174                    MOV       @PCSR1,R1             ;GET PCSR1 CONTENTS
023144' 042701  177770                    BIC       #^CPSTATE,R1          ;MASK ALL BUT PORT STATUS BITS
023150' 022701  000002                    CMP       #READY,R1             ;IS THE DEUNA IN THE READY STATE?
023154' 001404                            BEQ       30$                   ;YES
                                                                          ;ERROR, DEUNA NOT IN READY STATE!
023156'                                   ERRHRD    015,RSETER,MSG4       ;PRINT ERROR MESSAGE
023156' 104456                                                                                        TRAP      C$ERHRD
023160' 000017                                                                                        .WORD     15
023162' 001717'                                                                                       .WORD     RSETER
023164' 013120'                                                                                       .WORD     MSG4
023166'                           30$:
023166'                             ENDSEG
023166'                                                                                       10002$:
023166' 104405                                                                                        TRAP      C$ESEG

                                  ;
                                  ;CHECK PCSR1 BITS 07:03 FOR 0
                                  ;
023170' 012701  000416'                   MOV       #BNAMT1,R1            ;POINT TO MNEMONIC TABLE FOR PCSR1
023174' 062701  000006                    ADD       #3*2,R1               ;INDEX PAST STATE BITS
023200' 012737  001311' 000312'           MOV       #$NCLR,BITSTA         ;PREPARE ERROR MESSAGE
023206' 012737  001340' 000314'           MOV       #$AFTER,PWHEN
023214' 012702  000010                    MOV       #BIT3,R2              ;POINT TO BIT 3 TO START TESTING
023220'                           40$:
023220'                             BGNSEG
023220' 104404                                                                                        TRAP      C$BSEG
023222' 030277  155112                    BIT       R2,@PCSR1             ;IS THIS BIT A 0?
023226' 001406                            BEQ       45$                   ;YES
                                                                          ;ERROR, BIT IS NOT A 0 AFTER RESET!
023230' 011137  000310'                   MOV       (R1),BITNAM           ;GET MNEMONIC FOR THIS BIT
023234'                                   ERRHRD    016,RSETER,MSG3       ;PRINT ERROR MESSAGE
023234' 104456                                                                                        TRAP      C$ERHRD
023236' 000020                                                                                        .WORD     16
023240' 001717'                                                                                       .WORD     RSETER
023242' 013056'                                                                                       .WORD     MSG3
023244'                           45$:
023244'                             ENDSEG
023244'                                                                                       10003$:
023244' 104405                                                                                        TRAP      C$ESEG
023246' 105702                            TSTB      R2                    ;WE TESTED BITS 07:03?
023250' 100404                            BMI       50$                   ;YES
023252' 006302                            ASL       R2                    ;NO, POINT TO NEXT BIT
023254' 062701  000002                    ADD       #2,R1                 ;POINT TO MNEMONIC FOR NEXT BIT
023260' 000757                            BR        40$                   ;CONTINUE TESTING
023262'                           50$:
023262'                             BGNSEG
023262' 104404                                                                                        TRAP      C$BSEG
                                  ;
                                  ;CHECK PCSR1 BIT 14 = 0
                                  ;
023264' 032777  040000  155046            BIT       #ICAB,@PCSR1          ;IS PORT/CABLING BIT CLEAR?
023272' 001412                            BEQ       60$                   ;YES
                                                                          ;ERROR, PORT/CABLING BIT NOT CLEAR!
023274' 012737  001311' 000312'           MOV       #$NCLR,BITSTA         ;PREPARE ERROR MESSAGE
023302' 012737  001074' 000310'           MOV       #$ICAB,BITNAM         ;GET MNEMONIC
023310'                                   ERRHRD    017,RSETER,MSG3       ;PRINT ERROR MESSAGE
```

```
023310' 104456                                                                              TRAP    C$ERHRD
023312' 000021                                                                              .WORD   17
023314' 001717'                                                                             .WORD   RSETER
023316' 013056'                                                                             .WORD   MSG3
023320'                         60$:
023320'                           ENDSEG
023320'                                                                          10004$:
023320' 104405                                                                              TRAP    C$ESEG
023322'                           BGNSEG
023322' 104404                                                                              TRAP    C$BSEG
                                ;
                                ;CHECK PCSR1 BIT 15 = 0
                                ;
023324' 032777  100000  155006          BIT     #XPWR,@PCSR1            ;IS TRANSCEIVER POWER BIT CLEAR?
023332' 001412                          BEQ     70$                     ;YES
                                                                        ;ERROR, TRANSCEIVER POWER BIT NOT CLEAR!
023334' 012737  001311' 000312'         MOV     #$NCLR,BITSTA           ;PREPARE ERROR MESSAGE
023342' 012737  001063' 000310'         MOV     #$XPWR,BITNAM           ;GET MNEMONIC
023350'                                 ERRHRD  018,RSETER,MSG3         ;PRINT ERROR MESSAGE
023350' 104456                                                                              TRAP    C$ERHRD
023352' 000022                                                                              .WORD   18
023354' 001717'                                                                             .WORD   RSETER
023356' 013056'                                                                             .WORD   MSG3
023360'                         70$.
023360'                           ENDSEG
023360'                                                                          10005$:
023360' 104405                                                                              TRAP    C$ESEG
023362'                          ENDSUB ;#2
023362'                                                                          L10110:
023362' 104403                                                                              TRAP    C$ESUB
023364'                         ENDTST
023364'                                                                          L10106:
023364' 104401                                                                              TRAP    C$ETST
```

```
                                          .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
                                          ;
                                          ;**********************************************************************

023366'                                   BGNTST
023366'                                                                                   T6::
023366' 012737  000002  000304'                   MOV     #2,CSRNUM              ;TESTING PCSR2
023374' 012701  000516'                           MOV     #PATERN,R1             ;GET ADDRESS OF DATA PATTERNS
023400' 012705  000005                            MOV     #5,R5                  ;5 DATA PATTERNS
023404' 012103                            10$:    MOV     (R1)+,R3               ;GET DATA PATTERN
023406'                                     BGNSEG
023406' 104404                                                                                 TRAP    C$BSEG
023410' 010377  154726                            MOV     R3,@PCSR2              ;WRITE PATTERN TO PCSR2
023414' 017704  154722                            MOV     @PCSR2,R4              ;READ PCSR2
023420' 042703  000001                            BIC     #BIT0,R3               ;MASK BIT 00
023424' 020304                                    CMP     R3,R4                  ;COMPARE WHAT WAS WRITTEN TO...
                                                                                 ;WHAT WAS READ
023426' 001404                                    BEQ     20$                    ;COMPARED OK
023430'                                           ERRHRD  019,RRWER,RACMG3       ;PRINT ERROR MESSAGE
023430' 104456                                                                                 TRAP    C$ERHRD
023432' 000023                                                                                 .WORD   19
023434' 001741'                                                                                .WORD   RRWER
023436' 012632'                                                                                .WORD   RACMG3
023440'                                   20$:
023440'                                     ENDSEG
023440'                                                                                  10000$:
023440' 104405                                                                                 TRAP    C$ESEG
023442' 005305                                    DEC     R5                     ;DONE ALL DATA PATTERNS?
023444' 001357                                    BNE     10$                    ;NO, CONTINUE TESTING
023446'                                   ENDTST
023446'                                                                                  L10111:
023446' 104401                                                                                 TRAP    C$ETST

```

```
                                          .SBTTL  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 SEQUENCE:
                                          ;        1-WRITE PATTERN 000001 TO PCSR3
                                          ;        2-VERIFY PATTERN IN PCSR3
                                          ;        3-WRITE PATTERN 000002 TO PCSR3
                                          ;        4-VERIFY PATTERN IN PCSR3
                                          ;
                                          ;*******************************************************************

023450'                                   BGNTST
023450'                                                                                  T7::
023450' 012737  000003  000304'                   MOV     #3,CSRNUM               ;TESTING PCSR3
023456' 012703  000001                            MOV     #BIT00,R3               ;DATA PATTERN =1
023462'                                     BGNSEG
023462' 104404                                                                                   TRAP    C$BSEG
023464' 010377  154654                            MOV     R3,@PCSR3               ;WRITE PATTERN TO PCSR3
023470' 017704  154650                            MOV     @PCSR3,R4               ;READ PCSR3
023474' 020304                                    CMP     R3,R4                   ;COMPARE
023476' 001404                                    BEQ     10$                     ;IF == GOOD
023500'                                           ERRHRD  020,RRWER,RACMG3        ;ELSE PRINT ERROR MESSAGE
023500' 104456                                                                                   TRAP    C$ERHRD
023502' 000024                                                                                   .WORD   20
023504' 001741'                                                                                  .WORD   RRWER
023506' 012632'                                                                                  .WORD   RACMG3
023510'                                   10$:
023510'                                     ENDSEG
023510'                                                                                  10000$:
023510' 104405                                                                                   TRAP    C$ESEG
023512' 006303                                    ASL     R3                      ;DATA PATTERN =2
023514'                                     BGNSEG
023514' 104404                                                                                   TRAP    C$BSEG
023516' 010377  154622                            MOV     R3,@PCSR3               ;WRITE PATTERN TO PCSR3
023522' 017704  154616                            MOV     @PCSR3,R4               ;READ PCSR3
023526' 020304                                    CMP     R3,R4                   ;COMPARE
023530' 001404                                    BEQ     20$                     ;IF = GOOD
023532'                                           ERRHRD  021,RRWER,RACMG3        ;ELSE PRINT ERROR MESSAGE
023532' 104456                                                                                   TRAP    C$ERHRD
023534' 000025                                                                                   .WORD   21
023536' 001741'                                                                                  .WORD   RRWER
023540' 012632'                                                                                  .WORD   RACMG3
023542'                                   20$:
023542'                                     ENDSEG
023542'                                                                                  10001$:
023542' 104405                                                                                   TRAP    C$ESEG
023544'                                   ENDTST
023544'                                                                                  L10112:
023544' 104401                                                                                   TRAP    C$ETST
```

```
                                        .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 PCSR0 BITS 3:0 AND WILL WAIT FOR THE 'DNI' BIT TO
                                        ;SET IN PCSR0.
                                        ;
                                        ;THIS NOP PORT COMMAND USES A MINIMUM OF HARDWARE BUT FORCES THE T11
                                        ;TO EXECUTE THE PORT COMMAND SEQUENCE.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-READ REGISTER PCSR0
                                        ;       2-VERIFY INTERRUPT ENABLE BIT 06 = LOGIC 0
                                        ;       3-VERIFY DONE INTERRUPT BIT 11 = LOGIC 0
                                        ;       4-WRITE PORT COMMAND NOP TO PCSR0 BITS 3:0
                                        ;       5-READ REGISTER PCSR0
                                        ;       6-VERIFY DONE INTERRUPT BIT 11 = LOGIC 1
                                        ;       7-WRITE REGISTER PCSR0 BIT 11 WITH LOGIC 1
                                        ;       8-READ REGISTER PCSR0
                                        ;       9-VERIFY PCSR0 BIT 11 = 0
                                        ;
                                        ;*****************************************************************

        023546'                         BGNTST
        023546'                                                                         T8::
        023546'                          BGNSUB ;#1
        023546'                                                                         T8.1:
        023546' 104402                                                                          TRAP    C$BSUB
                                        ;
                                        ;CHECK THE INTERRUPT ENABLE BIT; IF SET CLEAR IT
                                        ;
        023550' 013701  000336'                 MOV     PCSR0,R1                ;GET PCSR0 ADDRESS
        023554' 011102                          MOV     (R1),R2                 ;GET CONTENTS
        023556' 032702  000100                  BIT     #INTE,R2                ;IS INTERRUPT ENABLE SET?
        023562' 001412                          BEQ     10$                     ;NO
        023564' 112711  000000                  MOVB    #0,(R1)                 ;CLEAR INTERRUPT ENABLE
        023570' 011102                          MOV     (R1),R2                 ;GET PCSR0 CONTENTS
        023572' 032702  000100                  BIT     #INTE,R2                ;IS INTERRUPT ENABLE CLEAR NOW?
        023576' 001404                          BEQ     10$                     ;YES
        023600'                                 ERRHRD  022,NOPERR,RACMG4       ;NO,PRINT ERROR MESSAGE
        023600' 104456                                                                          TRAP    C$ERHRD
        023602' 000026                                                                          .WORD   22
        023604' 002001'                                                                         .WORD   NOPERR
        023606' 012704'                                                                         .WORD   RACMG4
        023610'                         10$:
        023610'                          ENDSUB ;#2
        023610'                                                                         L10114:
        023610' 104403                                                                          TRAP    C$ESUB
        023612'                          BGNSUB ;#2
        023612'                                                                         T8.2:
        023612' 104402                                                                          TRAP    C$BSUB
                                        ;
                                        ;CHECK THE DONE INTERRUPT BIT; IF SET WRITE ONE TO CLEAR IT
                                        ;
```

```
023614' 032702  004000                  BIT     #DNI,R2                 ;IS DONE INTERRUPT CLEAR?
023620' 001413                          BEQ     20$                     ;YES
023622' 000302                          SWAB    R2                      ;NO,ORIENT UPPER & LOWER BYTES
023624' 110261  000001                  MOVB    R2,1(R1)                ;SO WE CAN CLEAR UPPER BITS
023630' 011102                          MOV     (R1),R2                 ;GET PCSRO CONTENTS
023632' 032702  004000                  BIT     #DNI,R2                 ;IS DNI CLEAR NOW?
023636' 001404                          BEQ     20$                     ;YES
023640'                                 ERRHRD  023,NOPERR,RACMG7       ;NO,PRINT ERROR MESSAGE
023640' 104456                                                                                  TRAP    C$ERHRD
023642' 000027                                                                                  .WORD   23
023644' 002001'                                                                                 .WORD   NOPERR
023646' 012726'                                                                                 .WORD   RACMG7
023650'                         20$:
023650'                          ENDSUB ;#2
023650'                                                                                 L10115:
023650' 104403                                                                                  TRAP    C$ESUB
023652'                          BGNSUB ;#3
023652'                                                                                 T8.3:
023652' 104402                                                                                  TRAP    C$BSUB

                                ;
                                ;ISSUE NOP PORT COMMAND AND CHECK FOR DNI
                                ;
023654' 012777  000006  154454          MOV     #PNOP,@PCSRO            ;ISSUE NOP PORT COMMAND
023662' 012737  000176  000332'         MOV     #2*SECOND,METER         ;SETUP TIMER
023670' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
023674' 103022                          BCC     30$                     ;OK
                                                                        ;ERROR DNI DID NOT SET!
                                                                        ;SETUP TO PRINT ERROR MESSAGE
023676' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;POINT TO 'DNI' ASCII STRING
023704' 012737  001275' 000312'         MOV     #$NSET,BITSTA           ;POINT TO 'NOT SET' ASCII STRING
023712' 012737  001340' 000314'         MOV     #$AFTER,PWHEN           ;POINT TO 'AFTER' ASCII STRING
023720' 012737  001403' 000316'         MOV     #$NOP,PCOMND            ;POINT TO 'NOP' ASCII STRING
023726'                                 ERRHRD  024,NOPERR,MSG1         ;PRINT ERROR MESSAGE
023726' 104456                                                                                  TRAP    C$ERHRD
023730' 000030                                                                                  .WORD   24
023732' 002001'                                                                                 .WORD   NOPERR
023734' 012750'                                                                                 .WORD   MSG1
023736'                                 ESCAPE  TST                     ;DON'T CONTINUE TEST IF ERROR OCCURRED
023736' 104410                                                                                  TRAP    C$ESCAPE
023740' 000026                                                                                  .WORD   L10113-.
023742'                         30$:
023742'                           BGNSEG
023742' 104404                                                                                  TRAP    C$BSEG
                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
023744' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR DNI BIT
023750' 103004                          BCC     40$                     ;IT CLEARED OK
                                                                        ;ERROR DNI DID NOT CLEAR!
023752'                                 ERRHRD  025,NOPERR,RACMG7       ;NO,PRINT ERROR MESSAGE
023752' 104456                                                                                  TRAP    C$ERHRD
023754' 000031                                                                                  .WORD   25
023756' 002001'                                                                                 .WORD   NOPERR
023760' 012726'                                                                                 .WORD   RACMG7
023762'                         40$:
023762'                           ENDSEG
```

```
023762'                                                  10000$:
023762' 104405                                                   TRAP    C$ESEG
023764'                 ENDSUB ;#3
023764'                                                  L10116:
023764' 104403                                                   TRAP    C$ESUB
023766'                ENDTST
023766'                                                  L10113:
023766' 104401                                                   TRAP    C$ETST
```

```
                                            .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
                                            ;
                                            ;*******************************************************************

 023770'                                    BGNTST
 023770'                                                                            T9::
 023770' 004737  017422'                            JSR     PC,CLRINT               ;CLEAR ANY OUTSTANDING INTERRUPTS
 023774' 103006                                     BCC     10$                     ;OK
 023776'                                            ERRHRD  026,SLFTST              ;ERROR OCCURRED TRYING TO CLEAR
 023776' 104456                                                                                             TRAP    C$ERHRD
 024000' 000032                                                                                             .WORD   26
 024002' 002021'                                                                                            .WORD   SLFTST
 024004' 000000                                                                                             .WORD   0
 024006'                                            ESCAPE  TST                     ;LEAVE THIS TEST
 024006' 104410                                                                                             TRAP    C$ESCAPE
 024010' 000170                                                                                             .WORD   L10117-.
 024012' 012737  000770  000332'  10$:              MOV     #10*SECOND,METER        ;PUT SOME TIME ON THE METER
 024020' 012777  000003  154310                     MOV     #SLFT,@PCSR0            ; RUN SELF TEST
 024026' 004737  017330'                            JSR     PC,CHKDNI               ; DNI ?
 024032' 103022                                     BCC     30$                     ; YES
                                                                                    ;ERROR DNI FAILED TO SET!
 024034' 012737  000776' 000310'                    MOV     #$DNI,BITNAM
 024042' 012737  001275' 000312'                    MOV     #$NSET,BITSTA
 024050' 012737  001340' 000314'                    MOV     #$AFTER,PWHEN
 024056' 012737  001371' 000316'                    MOV     #$SLFT,PCOMND
 024064'                                            ERRHRD  027,SLFTST,MSG1 ;PRINT ERROR MESSAGE
 024064' 104456                                                                                             TRAP    C$ERHRD
 024066' 000033                                                                                             .WORD   27
 024070' 002021'                                                                                            .WORD   SLFTST
 024072' 012750'                                                                                            .WORD   MSG1
 024074'                                            ESCAPE  TST                     ; AND ABORT TEST
 024074' 104410                                                                                             TRAP    C$ESCAPE
 024076' 000102                                                                                             .WORD   L10117-.

 024100' 017703  154234           30$:              MOV     @PCSR1,R3               ;GET PCSR1 CONTENTS
 024104' 032703  037400                             BIT     #SFT,R3                 ;WAS SELF TEST SUCCESSFUL?
 024110' 001424                                     BEQ     40$                     ;YES
 024112' 005103                                     COM     R3                      ;GET TEST NUMBER CORRECT POLARITY
 024114' 042703  140377                             BIC     #^CSFT,R3               ;MASK ALL BUT SELF TEST FIELD
 024120' 012701  000010                             MOV     #8.,R1                  ;SHIFT RESULT OVER 8 BIT POSTIONS
 024124' 006203                   35$:              ASR     R3
 024126' 005301                                     DEC     R1
 024130' 001375                                     BNE     35$
```

```
024132' 010304                         MOV     R3,R4                   ;SAVE ERROR NUMBER
024134' 006303                         ASL     R3                      ;MAKE NUMBER A BYTE OFFSET
024136' 062703  024204'                ADD     #STTBL,R3               ; INDEX INTO SELF TEST TABLE
024142' 011337  024202'                MOV     (R3),STMSG              ; LOAD INTO SELF TEST MESSAGE
024146'                                ERRHRD  028,SLFTST,MSG34        ; REPORT SELF TEST FAILURE
024146' 104456                                                                                  TRAP    C$ERHRD
024150' 000034                                                                                  .WORD   28
024152' 002021'                                                                                 .WORD   SLFTST
024154' 015634'                                                                                 .WORD   MSG34
024156'                                ESCAPE  TST                     ; AND ABORT TEST
024156' 104410                                                                                  TRAP    C$ESCAPE
024160' 000020                                                                                  .WORD   L10117-.
                               ;
024162' 004737  017374'        40$:    JSR     PC,CLRDNI               ; WRITE ONF TO CLEAR DNI
                                                                       ; ERROR?
024166' 103004                         BCC     50$                     ; NO
                                                                       ;ERROR DNI FAILED TO CLEAR!
024170'                                ERRHRD  029,SLFTST,RACMG7       ;PRINT ERROR MESSAGE
024170' 104456                                                                                  TRAP    C$ERHRD
024172' 000035                                                                                  .WORD   29
024174' 002021'                                                                                 .WORD   SLFTST
024176' 012726'                                                                                 .WORD   RACMG7
024200'                        50$:
024200'                                ENDTST
024200'                                                                                L10117:
024200' 104401                                                                                  TRAP    C$ETST
```

```
                              ;
                              ;LOCAL STORAGE FOR TEST 9
                              ;
024202' 000000                STMSG::         .WORD   0               ; SELF TEST MESSAGE ADDRESS
                              ;
                              ;SELF TEST MESSAGE TABLE
                              ;
024204' 024404'               STTBL::         .WORD   SMSG00
024206' 024432'                               .WORD   SMSG01
024210' 024457'                               .WORD   SMSG02
024212' 024470'                               .WORD   SMSG03
024214' 024525'                               .WORD   SMSG04
024216' 024552'                               .WORD   SMSG05
024220' 024613'                               .WORD   SMSG06
024222' 024644'                               .WORD   SMSG07
024224' 024660'                               .WORD   SMSG10
024226' 024710'                               .WORD   SMSG11
024230' 024721'                               .WORD   SMSG12
024232' 024734'                               .WORD   SMSG13
024234' 024746'                               .WORD   SMSG14
024236' 024760'                               .WORD   SMSG15
024240' 024772'                               .WORD   SMSG16
024242' 025004'                               .WORD   SMSG17
024244' 025016'                               .WORD   SMSG20
024246' 025037'                               .WORD   SMSG21
024250' 025051'                               .WORD   SMSG22
024252' 025063'                               .WORD   SMSG23
024254' 025075'                               .WORD   SMSG24
024256' 025107'                               .WORD   SMSG25
024260' 025121'                               .WORD   SMSG26
024262' 025133'                               .WORD   SMSG27
024264' 025145'                               .WORD   SMSG30
024266' 025220'                               .WORD   SMSG31
024270' 025270'                               .WORD   SMSG32
024272' 025341'                               .WORD   SMSG33
024274' 025403'                               .WORD   SMSG34
024276' 025452'                               .WORD   SMSG35
024300' 025513'                               .WORD   SMSG36
024302' 025562'                               .WORD   SMSG37
024304' 025623'                               .WORD   SMSG40
024306' 025711'                               .WORD   SMSG41
024310' 025774'                               .WORD   SMSG42
024312' 026060'                               .WORD   SMSG43
024314' 026135'                               .WORD   SMSG44
024316' 026217'                               .WORD   SMSG45
024320' 026273'                               .WORD   SMSG46
024322' 026305'                               .WORD   SMSG47
024324' 026317'                               .WORD   SMSG50
024326' 026403'                               .WORD   SMSG51
024330' 026464'                               .WORD   SMSG52
024332' 026546'                               .WORD   SMSG53
024334' 026621'                               .WORD   SMSG54
024336' 026701'                               .WORD   SMSG55
024340' 026753'                               .WORD   SMSG56
024342' 026765'                               .WORD   SMSG57
024344' 026777'                               .WORD   SMSG60
```

```
024346' 027010'                                  .WORD   SMSG61
024350' 027031'                                  .WORD   SMSG62
024352' 027043'                                  .WORD   SMSG63
024354' 027070'                                  .WORD   SMSG64
024356' 027102'                                  .WORD   SMSG65
024360' 027121'                                  .WORD   SMSG66
024362' 027133'                                  .WORD   SMSG67
024364' 027145'                                  .WORD   SMSG70
024366' 027164'                                  .WORD   SMSG71
024370' 027215'                                  .WORD   SMSG72
024372' 027237'                                  .WORD   SMSG73
024374' 027251'                                  .WORD   SMSG74
024376' 027263'                                  .WORD   SMSG75
024400' 027275'                                  .WORD   SMSG76
024402' 027307'                                  .WORD   SMSG77
                                 ;ASCII MESSAGES
024404' 047503  050115  042514   SMSG00::        .ASCIZ  /COMPLETED - NO ERRORS/
024412' 042524  020104  020055
024420' 047516  042440  051122
024426' 051117  000123
024432' 050103  020125  047111   SMSG01::        .ASCIZ  /CPU INSTRUCTION TEST/
024440' 052123  052522  052103
024446' 047511  020116  042524
024454' 052123     000
024457'    122  046517  052040   SMSG02::        .ASCIZ  /ROM TEST/
024464' 051505  000124
024470' 051127  052111  040505   SMSG03::        .ASCIZ  /WRITEABLE CONTROL STORE TEST/
024476' 046102  020105  047503
024504' 052116  047522  020114
024512' 052123  051117  020105
024520' 042524  052123     000
024525'    120  054510  044523   SMSG04::        .ASCIZ  /PHYSICAL ADDRESS ROM/
024532' 040503  020114  042101
024540' 051104  051505  020123
024546' 047522  000115
024552' 030524  020061  047125   SMSG05::        .ASCIZ  /T11 UNIBUS ADDRESS REGISTER TEST/
024560' 041111  051525  040440
024566' 042104  042522  051523
024574' 051040  043505  051511
024602' 042524  020122  042524
024610' 052123     000
024613'    122  041505  044505   SMSG06::        .ASCIZ  /RECEIVER UNIBUS DMA TEST/
024620' 042526  020122  047125
024626' 041111  051525  042040
024634' 040515  052040  051505
024642' 000124
024644' 047125  041111  051525   SMSG07::        .ASCIZ  /UNIBUS TEST/
024652' 052040  051505  000124
024660' 044515  051103  050117   SMSG10::        .ASCIZ  /MICROPROCESSOR DMA TEST/
024666' 047522  042503  051523
024674' 051117  042040  040515
024702' 052040  051505  000124
024710' 051503  020122  042524   SMSG11::        .ASCIZ  /CSR TEST/
024716' 052123     000
024721'    124  046511  051105   SMSG12::        .ASCIZ  /TIMER TEST/
024726' 052040  051505  000124
```

```
 5883  024734' 047125  042504  044506  SMSG13::        .ASCIZ  /UNDEFINED/
 5884  024742' 042516  000104
 5885  024746' 047125  042504  044506  SMSG14::        .ASCIZ  /UNDEFINED/
 5886  024754' 042516  000104
 5887  024760' 047125  042504  044506  SMSG15::        .ASCIZ  /UNDEFINED/
 5888  024766' 042516  000104
 5889  024772' 047125  042504  044506  SMSG16::        .ASCIZ  /UNDEFINED/
 5890  025000' 042516  000104
 5891  025004' 047125  042504  044506  SMSG17::        .ASCIZ  /UNDEFINED/
 5892  025012' 042516  000104
 5893  025016' 044514  045516  046440  SMSG20::        .ASCIZ  /LINK MEMORY TEST/
 5894  025024' 046505  051117  020131
 5895  025032' 042524  052123     000
 5896  025037'    125  042116  043105  SMSG21::        .ASCIZ  /UNDEFINED/
 5897  025044' 047111  042105     000
 5898  025051'    125  042116  043105  SMSG22::        .ASCIZ  /UNDEFINED/
 5899  025056' 047111  042105     000
 5900  025063'    125  042116  043105  SMSG23::        .ASCIZ  /UNDEFINED/
 5901  025070' 047111  042105     000
 5902  025075'    125  042116  043105  SMSG24::        .ASCIZ  /UNDEFINED/
 5903  025102' 047111  042105     000
 5904  025107'    125  042116  043105  SMSG25::        .ASCIZ  /UNDEFINED/
 5905  025114' 047111  042105     000
 5906  025121'    125  042116  043105  SMSG26::        .ASCIZ  /UNDEFINED/
 5907  025126' 047111  042105     000
 5908  025133'    125  042116  043105  SMSG27::        .ASCIZ  /UNDEFINED/
 5909  025140' 047111  042105     000
 5910  025145'    114  041517  046101  SMSG30::        .ASCIZ  /LOCAL LOOPBACK TEST -  TRANSMITTER TIMEOUT/
 5911  025152' 046040  047517  041120
 5912  025160' 041501  020113  042524
 5913  025166' 052123  026440  020040
 5914  025174' 051124  047101  046523
 5915  025202' 052111  042524  020122
 5916  025210' 044524  042515  052517
 5917  025216' 000124
 5918  025220' 047514  040503  020114  SMSG31::        .ASCIZ  /LOCAL LOOPBACK TEST -  RECEIVER TIMEOUT/
 5919  025226' 047514  050117  040502
 5920  025234' 045503  052040  051505
 5921  025242' 020124  020055  051040
 5922  025250' 041505  044505  042526
 5923  025256' 020122  044524  042515
 5924  025264' 052517  000124
 5925  025270' 047514  040503  020114  SMSG32::        .ASCIZ  /LOCAL LOOPBACK TEST -  BUFFER COMPARSION/
 5926  025276' 047514  050117  040502
 5927  025304' 045503  052040  051505
 5928  025312' 020124  020055  041040
 5929  025320' 043125  042506  020122
 5930  025326' 047503  050115  051101
 5931  025334' 044523  047117     000
 5932  025341'    114  041517  046101  SMSG33::        .ASCIZ  /LOCAL LOOPBACK TEST -  BYTE COUNT/
 5933  025346' 046040  047517  041120
 5934  025354' 041501  020113  042524
 5935  025362' 052123  026440  020040
 5936  025370' 054502  042524  041440
 5937  025376' 052517  052116     000
 5938  025403'    114  041517  046101  SMSG34::        .ASCIZ  /LOCAL LOOPBACK TEST -  RECEIVER STATUS/
```

```
025410' 046040  047517  041120
025416' 041501  020113  042524
025424' 052123  026440  020040
025432' 042522  042503  053111
025440' 051105  051440  040524
025446' 052524  000123
025452' 047514  040503  020114  SMSG35::        .ASCIZ  /LOCAL LOOPBACK TEST -  CRC ERROR/
025460' 047514  050117  040502
025466' 045503  052040  051505
025474' 020124  020055  041440
025502' 041522  042440  051122
025510' 051117     000
025513'    114  041517  046101  SMSG36::        .ASCIZ  /LOCAL LOOPBACK TEST -  MATCH BIT ERROR/
025520' 046040  047517  041120
025526' 041501  020113  042524
025534' 052123  026440  020040
025542' 040515  041524  020110
025550' 044502  020124  051105
025556' 047522  000122
025562' 047514  040503  020114  SMSG37::        .ASCIZ  /LOCAL LOOPBACK TEST -  TDR ERROR/
025570' 047514  050117  040502
025576' 045503  052040  051505
025604' 020124  020055  052040
025612' 051104  042440  051122
025620' 051117     000
025623'    124  040522  051516  SMSG40::        .ASCIZ  /TRANSMITTER BUFFER ADDRESS TEST - TRANSMITTER TIMEOUT/
025630' 044515  052124  051105
025636' 041040  043125  042506
025644' 020122  042101  051104
025652' 051505  020123  042524
025660' 052123  026440  052040
025666' 040522  051516  044515
025674' 052124  051105  052040
025702' 046511  047505  052125
025710'    000
025711'    124  040522  051516  SMSG41::        .ASCIZ  /TRANSMITTER BUFFER ADDRESS TEST - RECEIVER TIMEOUT/
025716' 044515  052124  051105
025724' 041040  043125  042506
025732' 020122  042101  051104
025740' 051505  020123  042524
025746' 052123  026440  051040
025754' 041505  044505  042526
025762' 020122  044524  042515
025770' 052517  000124
025774' 051124  047101  046523  SMSG42::        .ASCIZ  /TRANSMITTER BUFFER ADDRESS TEST - BUFFER COMPARSION/
026002' 052111  042524  020122
026010' 052502  043106  051105
026016' 040440  042104  042522
026024' 051523  052040  051505
026032' 020124  020055  052502
026040' 043106  051105  041440
026046' 046517  040520  051522
026054' 047511  000116
026060' 051124  047101  046523  SMSG43::        .ASCIZ  /TRANSMITTER BUFFER ADDRESS TEST - BYTE COUNT/
026066' 052111  042524  020122
026074' 052502  043106  051105
```

```
 5995  026102' 040440  042104  042522
 5996  026110' 051523  052040  051505
 5997  026116' 020124  020055  054502
 5998  026124' 042524  041440  052517
 5999  026132' 052116     000
 6000  026135'    124  040522  051516  SMSG44::         .ASCIZ  /TRANSMITTER BUFFER ADDRESS TEST - RECEIVER STATUS/
 6001  026142' 044515  052124  051105
 6002  026150' 041040  043125  042506
 6003  026156' 020122  042101  051104
 6004  026164' 051505  020123  042524
 6005  026172' 052123  026440  051040
 6006  026200' 041505  044505  042526
 6007  026206' 020122  052123  052101
 6008  026214' 051525     000
 6009  026217'    124  040522  051516  SMSG45::         .ASCIZ  /TRANSMITTER BUFFER ADDRESS TEST - CRC ERROR/
 6010  026224' 044515  052124  051105
 6011  026232' 041040  043125  042506
 6012  026240' 020122  042101  051104
 6013  026246' 051505  020123  042524
 6014  026254' 052123  026440  041440
 6015  026262' 041522  042440  051122
 6016  026270' 051117     000
 6017  026273'    125  042116  043105  SMSG46::         .ASCIZ  /UNDEFINED/
 6018  026300' 047111  042105     000
 6019  026305'    125  042116  043105  SMSG47::         .ASCIZ  /UNDEFINED/
 6020  026312' 047111  042105     000
 6021  026317'    122  041505  044505  SMSG50::         .ASCIZ  /RECEIVER BUFFER ADDRESS TEST -  TRANSMITTER TIMEOUT/
 6022  026324' 042526  020122  052502
 6023  026332' 043106  051105  040440
 6024  026340' 042104  042522  051523
 6025  026346' 052040  051505  020124
 6026  026354' 020055  052040  040522
 6027  026362' 051516  044515  052124
 6028  026370' 051105  052040  046511
 6029  026376' 047505  052125     000
 6030  026403'    122  041505  044505  SMSG51::         .ASCIZ  /RECEIVER BUFFER ADDRESS TEST -  RECEIVER TIMEOUT/
 6031  026410' 042526  020122  052502
 6032  026416' 043106  051105  040440
 6033  026424' 042104  042522  051523
 6034  026432' 052040  051505  020124
 6035  026440' 020055  051040  041505
 6036  026446' 044505  042526  020122
 6037  026454' 044524  042515  052517
 6038  026462' 000124
 6039  026464' 042522  042503  053111  SMSG52::         .ASCIZ  /RECEIVER BUFFER ADDRESS TEST -  BUFFER COMPARSION/
 6040  026472' 051105  041040  043125
 6041  026500' 042506  020122  042101
 6042  026506' 051104  051505  020123
 6043  026514' 042524  052123  026440
 6044  026522' 020040  052502  043106
 6045  026530' 051105  041440  046517
 6046  026536' 040520  051522  047511
 6047  026544' 000116
 6048  026546' 042522  042503  053111  SMSG53::         .ASCIZ  /RECEIVER BUFFER ADDRESS TEST -  BYTE COUNT/
 6049  026554' 051105  041040  043125
 6050  026562' 042506  020122  042101
```

```
026570' 051104  051505  020123
026576' 042524  052123  026440
026604' 020040  054502  042524
026612' 041440  052517  052116
026620'    000
026621'    122  041505  044505  SMSG54::        .ASCIZ  /RECEIVER BUFFER ADDRESS TEST -  RECEIVER STATUS/
026626' 042526  020122  052502
026634' 043106  051105  040440
026642' 042104  042522  051523
026650' 052040  051505  020124
026656' 020055  051040  041505
026664' 044505  042526  020122
026672' 052123  052101  051525
026700'    000
026701'    122  041505  044505  SMSG55::        .ASCIZ  /RECEIVER BUFFER ADDRESS TEST -  CRC ERROR/
026706' 042526  020122  052502
026714' 043106  051105  040440
026722' 042104  042522  051523
026730' 052040  051505  020124
026736' 020055  041440  041522
026744' 042440  051122  051117
026752'    000
026753'    125  042116  043105  SMSG56::        .ASCIZ  /UNDEFINED/
026760' 047111  042105     000
026765'    125  042116  043105  SMSG57::        .ASCIZ  /UNDEFINED/
026772' 047111  042105     000
026777'    103  041522  052040  SMSG60::        .ASCIZ  /CRC TEST/
027004' 051505  000124
027010' 052522  052116  050040  SMSG61::        .ASCIZ  /RUNT PACKET TEST/
027016' 041501  042513  020124
027024' 042524  052123     000
027031'    125  042116  043105  SMSG62::        .ASCIZ  /UNDEFINED/
027036' 047111  042105     000
027043'    117  042526  051522  SMSG63::        .ASCIZ  /OVERSIZE PACKET TEST/
027050' 055111  020105  040520
027056' 045503  052105  052040
027064' 051505  000124
027070' 047125  042504  044506  SMSG64::        .ASCIZ  /UNDEFINED/
027076' 042516  000104
027102' 047503  046114  051511  SMSG65::        .ASCIZ  /COLLISION TEST/
027110' 047511  020116  042524
027116' 052123     000
027121'    125  042116  043105  SMSG66::        .ASCIZ  /UNDEFINED/
027126' 047111  042105     000
027133'    125  042116  043105  SMSG67::        .ASCIZ  /UNDEFINED/
027140' 047111  042105     000
027145'    115  046125  044524  SMSG70::        .ASCIZ  /MULTICAST TEST/
027152' 040503  052123  052040
027160' 051505  000124
027164' 042101  051104  051505  SMSG71::        .ASCIZ  /ADDRESS RECOGNITION TEST/
027172' 020123  042524  047503
027200' 047107  052111  047511
027206' 020116  042524  052123
027214'    000
027215'    105  052130  051105  SMSG72::        .ASCIZ  /EXTERNAL LOOPBACK/
027222' 040516  020114  047514
```

```
  6107  027230' 050117  040502  045503
  6108  027236'    000
  6109  027237'    125  042116  043105  SMSG73::         .ASCIZ  /UNDEFINED/
  6110  027244' 047111  042105     000
  6111  027251'    125  042116  043105  SMSG74::         .ASCIZ  /UNDEFINED/
  6112  027256' 047111  042105     000
  6113  027263'    125  042116  043105  SMSG75::         .ASCIZ  /UNDEFINED/
  6114  027270' 047111  042105     000
  6115  027275'    125  042116  043105  SMSG76::         .ASCIZ  /UNDEFINED/
  6116  027302' 047111  042105     000
  6117  027307'    116  053105  051105  SMSG77::         .ASCIZ  /NEVER GOT STARTED/
  6118  027314' 043440  052117  051440
  6119  027322' 040524  052122  042105
  6120  027330'    000
  6121          027332'                                  .EVEN
```

```
                                         .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 PCSR0
                                         ;       3-READ PCSR0 AND VERIFY DNI BIT SET
                                         ;       4-WRITE PCSR0 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 PCSR0
                                         ;       12-READ PCSR0 AND VERIFY DNI SET
                                         ;       13-CALCULATE CRC ON DUMPED DATA
                                         ;       14-REPEAT STEPS 5-13 ON EACH 1K OF ROM
                                         ;       15-VERIFY CRC =0
                                         ;
                                         ;*****************************************************************************

 027332'                                BGNTST
 027332'                                                                              T10::
 027332'                                 BGNSUB ;#1
 027332'                                                                              T10.1:
 027332' 104402                                                                               TRAP    C$BSUB
                                         ;
                                         ;LOAD PCSR2+3; ISSUE GET PORT CONTROL BLOCK PORT COMMAND; AND WAIT FOR 'DNI'
                                         ;
 027334'                                  BGNSEG
 027334' 104404                                                                               TRAP    C$BSEG
 027336' 012777  000604' 150776                MOV     #PCBB,@PCSR2            ;LOAD PCSR2+3 WITH PORT CONTROL BLOCK..
 027344' 005077  150774                        CLR     @PCSR3                  ;ADDRESS
 027350' 012777  000001  150760                MOV     #GETPCB,@PCSR0          ;ISSUE GET PORT CONTROL BLOCK COMMAND
 027356' 012737  000176  000332'               MOV     #2*SECOND,METER ;PUT SOME TIME IN THE METER
 027364' 004737  017330'                       JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
 027370' 103022                                BCC     20$                     ;OK
```

```
                                                                        ;ERROR DNI DID NOT SET!
027372' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;NO, FORMAT ERROR MESSAGE
027400' 012737  001275' 000312'         MOV     #$NSET,BITSTA
027406' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
027414' 012737  001346' 000316'         MOV     #$GTPCB,PCOMND
027422'                                 ERRHRD  030,ROMDMP,MSG1 ;PRINT ERROR MESSAGE
027422' 104456                                                                                 TRAP    C$ERHRD
027424' 000036                                                                                 .WORD   30
027426' 002042'                                                                                .WORD   ROMDMP
027430' 012750'                                                                                .WORD   MSG1
027432'                                 ESCAPE  TST                     ;CAN NOT CONTINUE TESTING
027432' 104410                                                                                 TRAP    C$ESCAPE
027434' 000310                                                                                 .WORD   L10120-.
027436'                         20$:
027436'                           ENDSEG
027436'                                                                                10000$:
027436' 104405                                                                                 TRAP    C$ESEG
027440'                           BGNSEG
027440' 104404                                                                                 TRAP    C$BSEG
                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
027442' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
027446' 103004                          BCC     25$                     ;OK
                                                                        ;ERROR DNI FAILED TO CLEAR!
027450'                                 ERRHRD  031,ROMDMP,RACMG7       ;PRINT ERROR MESSAGE
027450' 104456                                                                                 TRAP    C$ERHRD
027452' 000037                                                                                 .WORD   31
027454' 002042                                                                                 .WORD   ROMDMP
027456' 012726'                                                                                .WORD   RACMG7
027460'                         25$:
027460'                           ENDSEG
027460'                                                                                10001$:
027460' 104405                                                                                 TRAP    C$ESEG
027462'                          ENDSUB ;#1
027462'                                                                                L10121:
027462' 104403                                                                                 TRAP    C$ESUB
027464'                          BGNSUB ;#2
027464'                                                                                T10.2:
027464' 104402                                                                                 TRAP    C$BSUB
                                ;
                                ;FILL BUFFER WITH PATTERN; FORMAT PCBB AND UDBB; ISSUE GET COMMAND PORT COMMAND
                                ;AND WAIT FOR 'DNI'
                                ;
027466' 012737  000020  000604'         MOV     #DIM,PCBB               ;LOAD PCB WITH DUMP INTERNAL MEMORY FUNCTION
027474' 012737  000614' 000606'         MOV     #UDBB,PCBB+2            ;LOAD UNIBUS DATA BLOCK BASE ADDRESS...
027502' 005037  000610'                 CLR     PCBB+4                  ;INTO PCB
027506' 012737  040000  000622'         MOV     #ROMADR,UDBB+6          ;INTERNAL BASE ADDRESS OF ROM
027514' 012705  000010                  MOV     #8.,R5                  ;DUMP ROM IN 8. 1K CHUNKS
027520' 005004                          CLR     R4                      ;INITIAL CRC VALUE
027522'                         26$:
027522'                           BGNSEG
027522' 104404                                                                                 TRAP    C$BSEG
027524' 013701  000324'                 MOV     FREMEM,R1               ;GET POINTER TO BUFFER
027530' 012702  002000                  MOV     #SIZ1K/2,R2             ;GET 1K WORD COUNT
027534' 012703  000516'                 MOV     #PAT1,R3                ;GET A BACKGROUND PATTERN
```

```
027540' 011321                  27$:    MOV     (R3),(R1)+              ;FILL BUFFER WITH BACKGROUND PATTERN
027542' 005302                          DEC     R2
027544' 001375                          BNE     27$
027546' 012737  004000  000614'         MOV     #ROMSIZ/8.,UDBB         ;1K BYTE COUNT
027554' 013737  000324' 000616'         MOV     FREMEM,UDBB+2           ;AND BUFFER ADDRESS
027562' 005037  000620'                 CLR     UDBB+4
027566' 012737  000176  000332'         MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
027574' 012777  000002  150534          MOV     #GETCMD,@PCSR0          ;LOAD 'GET COMMAND' PORT COMMAND
027602' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
027606' 103022                          BCC     30$                     ;OK
                                                                        ;ERROR, DNI DID NOT SET!
027610' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;FORMAT ERROR MESSAGE
027616' 012737  001275' 000312'         MOV     #$NSET,BITSTA
027624' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
027632' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
027640'                                 ERRHRD  032,ROMDMP,MSG1
027640' 104456                                                                                  TRAP    C$ERHRD
027642' 000040                                                                                  .WORD   32
027644' 002042'                                                                                 .WORD   ROMDMP
027646' 012750'                                                                                 .WORD   MSG1
027650'                                 ESCAPE  TST
027650' 104410                                                                                  TRAP    C$ESCAPE
027652' 000072                                                                                  .WORD   L10120-.
027654'                         30$:
027654'                           ENDSEG
027654'                                                                                 10000$:
027654' 104405                                                                                  TRAP    C$ESEG
027656'                           BGNSEG
027656' 104404                                                                                  TRAP    C$BSEG
                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
027660' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR DNI BIT
027664' 103004                          BCC     33$                     ;OK
                                                                        ;ERROR DNI FAILED TO CLEAR
027666'                                 ERRHRD  033,ROMDMP,RACMG7       ;PRINT ERROR MESSAGE
027666' 104456                                                                                  TRAP    C$ERHRD
027670' 000041                                                                                  .WORD   33
027672' 002042'                                                                                 .WORD   ROMDMP
027674' 012726'                                                                                 .WORD   RACMG7
027676'                         33$:
027676'                           ENDSEG
027676'                                                                                 10001$:
027676' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;CALCULATE CRC ON 1K OF DATA DUMPED FROM ROM
                                ;
027700' 013700  000324'                 MOV     FREMEM,R0               ;GET BUFFER ADDRESS
027704' 012702  004000                  MOV     #SIZ1K,R2               ;GET BYTE COUNT
027710' 004737  017042'                 JSR     PC,CRC16                ;CALC CRC ON 1K BUFFER
027714' 062737  004000  000622'         ADD     #ROMSIZ/8.,UDBB+6       ;POINT TO NEXT 1K OF ROM
027722' 005305                          DEC     R5                      ;HAVE WE DONE ALL 8K?
027724' 001276                          BNE     26$                     ;NO
027726' 005704                          TST     R4                      ;YES, IS CRC = 0?
027730' 001404                          BEQ     40$                     ;YES, OK
027732'                                 ERRHRD  034,ROMDMP,MSG2         ;PRINT ERROR MESSAGE
```

```
027732' 104456                                                          TRAP    C$ERHRD
027734' 000042                                                          .WORD   34
027736' 002042'                                                         .WORD   ROMDMP
027740' 013012'                                                         .WORD   MSG2
027742'                      40$:
027742'                       ENDSUB ;#2
027742'                                                         L10122:
027742' 104403                                                          TRAP    C$ESUB
027744'                      ENDTST
027744'                                                         L10120:
027744' 104401                                                          TRAP    C$ETST



```

```

                                   .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
                                   ;
                                   ;***********************************************************************

 027746'                           BGNTST
 027746'                                                                    T11::
 027746' 012704  000516'                   MOV     #PATERN,R4              ;POINT TO LIST OF DATA PATTERNS
 027752'                            BGNSUB ;#1
 027752'                                                                    T11.1:
 027752' 104402                                                                     TRAP    C$BSUB
                                   ;
                                   ;LOAD WCS
                                   ;
 027754'                             BGNSEG
 027754' 104404                                                                     TRAP    C$BSEG
                                   ;
                                   ;FILL SOURCE BUFFER; FORMAT UDBB AND PCBB; ISSUE GET PORT CONTROL BLOCK PORT
                                   ;COMMAND AND WAIT FOR 'DNI'
                                   ;
 027756' 013702  000324'                   MOV     FREMEM,R2               ;GET UNIBUS BUFFER ADDRESS
 027762' 012703  004000                    MOV     #WCSSIZ/4,R3            ;GET HALF SIZE OF WCS IN WORDS
 027766' 011422                    10$:    MOV     (R4),(R2)+              ;FILL BUFFER WITH DATA PATTERN
 027770' 005303                            DEC     R3
 027772' 001375                            BNE     10$

                                                                           ;SETUP UNIBUS DATA BLOCK
```

```
027774' 012737  010000  000614'         MOV     #WCSSIZ/2,UDBB          ;BYTE COUNT
030002' 013737  000324' 000616'         MOV     FREMEM,UDBB+2           ;BUFFER ADDRESS
030010' 005037  000620'                 CLR     UDBB+4
030014' 012737  010000  000622'         MOV     #WCSADR+<WCSSIZ/2>,UDBB+6 ;BASE ADDRESS OF TOP HALF OF WCS
                                                                        ;SETUP PORT CONTROL BLOCK
030022' 012737  000021  000604'         MOV     #LIM,PCBB               ;'LOAD INTERNAL MEMORY' FUNCTION
030030' 012737  000614' 000606'         MOV     #UDBB,PCBB+2            ;ADDRESS OF UNIBUS DATA BLOCK
030036' 005037  000610'                 CLR     PCBB+4
030042' 012777  000604' 150272          MOV     #PCBB,@PCSR2            ;LOAD PCSR2+3 WITH PCB ADDRESS
030050' 005077  150270                  CLR     @PCSR3

030054' 012777  000001  150254          MOV     #GETPCB,@PCSR0          ;ISSUE 'GET PORT CONTROL BLOCK' PORT COMMAND
030062' 012737  000176  000332'         MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
030070' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
030074' 103022                          BCC     20$                     ;OK
                                                                        ;ERROR DNI DID NOT SET!
030076' 012737  000776' 000310'         MOV     #$DNI,BITNAM            ;POINT TO 'DNI' ASCII STRING
030104' 012737  001275' 000312'         MOV     #$NSET,BITSTA           ;POINT TO 'NOT SET' ASCII STRING
030112' 012737  001340' 000314'         MOV     #$AFTER,PWHEN           ;POINT TO 'AFTER' ASCII STRING
030120' 012737  001346' 000316'         MOV     #$GTPCB,PCOMND          ;POINT TO 'GET PORT CONTROL BLOCK' ASCII STRING
030126'                                 ERRHRD  035,DATALD,MSG1         ;PRINT ERROR MESSAGE
030126' 104456                                                                                  TRAP    C$ERHRD
030130' 000043                                                                                  .WORD   35
030132' 002067'                                                                                 .WORD   DATALD
030134' 012750'                                                                                 .WORD   MSG1
030136'                                 ESCAPE  TST
030136' 104410                                                                                  TRAP    C$ESCAPE
030140' 000372                                                                                  .WORD   L10123-.
030142'                         20$:
030142'                           ENDSEG
030142'                                                                                 10000$:
030142' 104405                                                                                  TRAP    C$ESEG
030144'                           BGNSEG
030144' 104404                                                                                  TRAP    C$BSEG
                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
030146' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
030152' 103004                          BCC     25$                     ;OK
                                                                        ;ERROR DNI FAILED TO CLEAR
030154'                                 ERRHRD  036,DATALD,RACMG7       ;PRINT ERROR MESSAGE
030154' 104456                                                                                  TRAP    C$ERHRD
030156' 000044                                                                                  .WORD   36
030160' 002067'                                                                                 .WORD   DATALD
030162' 012726'                                                                                 .WORD   RACMG7
030164'                         25$:
030164'                           ENDSEG
030164'                                                                                 10001$:
030164' 104405                                                                                  TRAP    C$ESEC
030166'                           BGNSEG
030166' 104404                                                                                  TRAP    C$BSEG
                                ;
                                ;ISSUE GET COMMAND PORT COMMAND AND WAIT FOR 'DNI'
                                ;
030170' 012777  000002  150140          MOV     #GETCMD,@PCSR0          ;ISSUE GET COMMAND PORT COMMAND
030176' 012737  000275  000332'         MOV     #3*SECOND,METER         ;PUT SOME TIME ON THE METER
```

```
030204'   004737    017330'                       JSR       PC,CHKDNI                     ;GO WAIT FOR DNI
030210'   103022                                  BCC       30$                           ;OK
                                                                                          ;ERROR DNI DID NOT SET!
030212'   012737    000776'   000310'             MOV       #$DNI,BITNAM
030220'   012737    001275'   000312'             MOV       #$NSET,BITSTA
030226'   012737    001340'   000314'             MOV       #$AFTER,PWHEN
030234'   012737    001355'   000316'             MOV       #$GTCMD,PCOMND
030242'                                           ERRHRD    037,DATALD,MSG1               ;PRINT ERROR MESSAGE
030242'   104456                                                                                              TRAP      C$ERHRD
030244'   000045                                                                                              .WORD     37
030246'   002067'                                                                                             .WORD     DATALD
030250'   012750'                                                                                             .WORD     MSG1
030252'                                           ESCAPE TST
030252'   104410                                                                                              TRAP      C$ESCAPE
030254'   000256                                                                                              .WORD     L10123-.
030256'                                 30$:
030256'                                   ENDSEG
030256'                                                                                             10002$:
030256'   104405                                                                                              TRAP      C$ESEG
030260'                                   BGNSEG
030260'   104404                                                                                              TRAP      C$BSEG
                                        ;
                                        ;WRITE ONE TO CLEAR 'DNI'
                                        ;
030262'   004737    017374'                       JSR       PC,CLRDNI                     ;GO CLEAR DNI
030266'   103004                                  BCC       35$                           ;OK
                                                                                          ;ERROR DNI FAILED TO CLEAR
030270'                                           ERRHRD    038,DATALD,RACMG7             ;PRINT ERROR MESSAGE
030270'   104456                                                                                              TRAP      C$ERHRD
030272'   000046                                                                                              .WORD     38
030274'   002067'                                                                                             .WORD     DATALD
030276'   012726'                                                                                             .WORD     RACMG7
030300'                                 35$:
030300'                                   ENDSEG
030300'                                                                                             10003$:
030300'   104405                                                                                              TRAP      C$ESEG
030302'                                  ENDSUB ;#1
030302'                                                                                             L10124:
030302'   104403                                                                                              TRAP      C$ESUB
030304'                                  BGNSUB ;#2
030304'                                                                                             T11.2:
030304'   104402                                                                                              TRAP      C$BSUB
                                        ;
                                        ;DUMP WCS
                                        ;
030306'                                   BGNSEG
030306'   104404                                                                                              TRAP      C$BSEG
                                        ;
                                        ;CLEAR DESTINATION BUFFER; FORMAT PCBB AND UDBB; ISSUE GET COMMAND PORT COMMAND
                                        ;AND WAIT FOR 'DNI'
                                        ;
030310'   013702    000324'                       MOV       FREMEM,R2                     ;GET ADDRESS OF FREE MEMORY
030314'   012703    004000                        MOV       #WCSSIZ/4,R3                  ;SIZE IN WORDS OF BUFFER
030320'   005022                        40$:      CLR       (R2)+                         ;FILL DESTINATION BUFFER WITH ZEROS
030322'   005303                                  DEC       R3
030324'   001375                                  BNE       40$
```

```

030326'  013737   000324'  000616'           MOV      FREMEM,UDBB+2
030334'  005037   000620'                    CLR      UDBB+4
030340'  012737   000020   000604'           MOV      #DIM.PCBB                  ;LOAD 'DUMP INTERNAL MEMORY' FUNCTION
030346'  012777   000002   147762            MOV      #GETCMD,@PCSR0             ;ISSUE GET COMMAND PORT COMMAND
030354'  012737   000275   000332'           MOV      #3*SECOND,METER            ;PUT SOME TIME ON THE METER
030362'  004737   017330'                    JSR      PC,CHKDNI                  ;WAIT FOR DNI TO SET
030366'  103022                              BCC      50$                        ;OK
030370'  012737   000776'  000310'           MOV      #$DNI,BITNAM
030376'  012737   001275'  000312'           MOV      #$NSET,BITSTA
030404'  012737   001340'  000314'           MOV      #$AFTER,PWHEN
030412'  012737   001355'  000316'           MOV      #$GTCMD,PCOMND
030420'                                      ERRHRD   039,DATALD,MSG1
030420'  104456                                                                                    TRAP     C$ERHRD
030422'  000047                                                                                    .WORD    39
030424'  002067'                                                                                   .WORD    DATALD
030426'  012750'                                                                                   .WORD    MSG1
030430'                                      ESCAPE TST
030430'  104410                                                                                    TRAP     C$ESCAPE
030432'  000100                                                                                    .WORD    L10123-.
030434'                             50$:
030434'                               ENDSEG
030434'                                                                                   10000$:
030434'  104405                                                                                    TRAP     C$ESEG
030436'                               BGNSEG
030436'  104404                                                                                    TRAP     C$BSEG
030440'  004737   017374'                    JSR      PC,CLRDNI                  ;GO CLEAR DNI
030444'  103004                              BCC      60$                        ;OK
                                                                                 ;ERROR DNI FAILED TO CLEAR
030446'                                      ERRHRD   040,DATALD,RACMG7          ;PRINT ERROR MESSAGE
030446'  104456                                                                                    TRAP     C$ERHRD
030450'  000050                                                                                    .WORD    40
030452'  002067'                                                                                   .WORD    DATALD
030454'  012726'                                                                                   .WORD    RACMG7
030456'                             60$:
030456'                               ENDSEG
030456'                                                                                   10001$:
030456'  104405                                                                                    TRAP     C$ESEG
                                    ;
                                    ;COMPARE DUMPED DATA TO WRITTEN PATTERN
                                    ;
030460'  013701   000324'                    MOV      FREMEM,R1                  ;SOURCE BUFFER ADDRESS
030464'  012703   004000                     MOV      #<WCSSIZ/2>/2,R3           ;# OF WORDS TO COMPARE
030470'  021114                     70$:     CMP      (R1`,(R4)                  ;IS WHAT WAS LOADED SAME AS...
                                                                                 ;WHAT WAS DUMPED?
030472'  001404                              BEQ      80$                        ;YES
030474'                                      ERRHRD   041,DATALD,MSG5 ;ERROR DATA COMPARE
030474'  104456                                                                                    TRAP     C$ERHRD
030476'  000051                                                                                    .WORD    41
030500'  002067'                                                                                   .WORD    DATALD
030502'  013142'                                                                                   .WORD    MSG5
030504'  005721                     80$:     TST      (R1)+                      ;POINT TO NEXT LOCATION
030506'  005303                              DEC      R3                         ;WE DONE ALL WORDS?
030510'  001367                              BNE      70$                        ;NOT YET
030512'                              ENDSUB ;#2
030512'                                                                                   L10125:
```

```
030512' 104403                                                                      TRAP    C$ESUB
                                      ;
                                      ;CKECK TO SEE IF ALL PATTERNS HAVE BEEN RUN THROUGH
                                      ;
030514' 062704  000002                        ADD     #2,R4                 ;POINT TO NEXT DATA PATTERN
030520' 020427  000530'                       CMP     R4,#PAT6              ;WE DONE ALL DATA PATTERNS?
030524' 001402                                BEQ     90$                   ;YES END OF TEST
030526' 000137  027752'                       JMP     T11.1                 ;NO CONTINUE WITH NEW DATA PATTERN
030532'                               90$:
030532'                               ENDTST
030532'                                                                L10123:
030532' 104401                                                                      TRAP    C$ETST
```

```

                                .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 ADDESS OF UNIBUS DATA BLOCK AND
                                ;          LOAD INTERNAL MEMORY FUNCTION CODE.
                                ;        3-WRITE PCSR2 AND PCSR3 WITH ADDRESS OF THE PORT CONTROL BLOCK.
                                ;        4-WRITE PCSR0 WITH <GET PCB> PORT COMMAND
                                ;        5-READ PCSR0 AND VERIFY 'DNI'
                                ;        6-WRITE PCSR0 'DNI' TO CLEAR
                                ;        7-WRITE PCSR0 WITH <GET COMMAND> PORT COMMAND
                                ;        8-READ PCSR0 AND VERIFY 'DNI'
                                ;        9-WRITE PCSR0 'DNI' TO CLEAR
                                ;        10-WRITE PORT CONTROL BLOCK WITH START ADDRESS OF WCS MICROCODE
                                ;           AND WITH LOAD AND START FUNCTION CODE
                                ;        11-WRITE PCSR0 WITH <GET COMMAND> PORT COMMAND
                                ;        12-READ PCSR0 AND VERIFY 'DNI' SET
                                ;        13-READ PCSR1 AND VERIFY BITS 13:8 ARE PATTERN WRITTEN BY MICROCODE
                                ;        14-ISSUE RESET TO PCSR0 TO RESTART OPERATIONAL MICROCODE
                                ;        15-READ PCSR0 AND VERIFY 'DNI' SET
                                ;        16-WRITE PCSR0 WITH 'DNI' TO CLEAR
                                ;
                                ;*****************************************************************************

030534'                         BGNTST
030534'                                                                        T12::

                                ;
                                ;LOAD MICROCODE MODULE 'A' INTO THE TOP HALF OF WCS
                                ;
030534'                           BGNSEG
030534' 104404                                                                        TRAP    C$BSEG

                                ;
                                ;FORMAT THE UNIBUS DATA BLOCK AND THE PORT CONTROL BLOCK FOR THE LOAD
                                ;INTERNAL MEMORY FUNCTION
                                ;
                                                                    ;SETUP UDBB
030536' 013737  000000G 000614'        MOV     MICASZ,UDBB          ;SIZE OF MICROCODE MODULE TO LOAD
030544' 012737  000000G 000616'        MOV     #MICROA,UDBB+2       ;BASE ADDRESS OF MICROCODE MODULE
030552' 005037  000620'                CLR     UDBB+4
030556' 012737  010000  000622'        MOV     #WCSADR+<WCSSIZ/2>,UDBB+6 ;LOAD INTO TOP HALF OF WCS

                                                                    ;SETUP PCB
030564' 012737  000021  000604'        MOV     #LIM,PCBB            ;'LOAD INTERNAL MEMORY' FUNCTION
```

```
030572' 012737  000614' 000606'           MOV     #UDBB,PCBB+2            ;SET ADDRESS OF UDBB
030600' 005037  000610'                   CLR     PCBB+4
030604' 012777  000604' 147530            MOV     #PCBB,@PCSR2            ;TELL DEUNA WHERE PCBB IS
030612' 005077  147526                    CLR     @PCSR3

030616' 012777  000001  147512            MOV     #GETPCB,@PCSR0          ;ISSUE 'GET PCB' PORT COMMAND
030624' 012737  000176  000332'           MOV     #2*SECOND,METER         ;SETUP TIMER
030632' 004737  017330'                   JSR     PC,CHKDNI               ;WAIT FOR 'DNI' TO SET
030636' 103022                            BCC     10$                     ;OK
                                                                          ;ERROR DNI NOT SET!
030640' 012737  000776' 000310'           MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
030646' 012737  001275' 000312'           MOV     #$NSET,BITSTA
030654' 012737  001340' 000314'           MOV     #$AFTER,PWHEN
030662' 012737  001346' 000316'           MOV     #$GTPCB,PCOMND
030670'                                   ERRHRD  042,LASFT,MSG1          ;PRINT ERROR MESSAGE
030670' 104456                                                                                    TRAP    C$ERHRD
030672' 000052                                                                                    .WORD   42
030674' 002121'                                                                                   .WORD   LASFT
030676' 012750'                                                                                   .WORD   MSG1
030700'                                   ESCAPE  TST                     ;NO POINT IN CONTINUING TEST
030700' 104410                                                                                    TRAP    C$ESCAPE
030702' 000310                                                                                    .WORD   L10126-.
030704'                           10$:
030704'                             ENDSEG
030704'                                                                                   10000$:
030704' 104405                                                                                    TRAP    C$ESEG
030706'                             BGNSEG
030706' 104404                                                                                    TRAP    C$BSEG
                                  ;
                                  ;WRITE ONE TO CLEAR 'DNI'
                                  ;
030710' 004737  017374'                   JSR     PC,CLRDNI               ;GO CLEAR 'DNI'
030714' 103006                            BCC     20$                     ;OK
                                                                          ;ERROR 'DNI' NOT CLEAR!
030716'                                   ERRHRD  043,LASFT,RACMG7        ;PRINT ERROR MESSAGE
030716' 104456                                                                                    TRAP    C$ERHRD
030720' 000053                                                                                    .WORD   43
030722' 002121'                                                                                   .WORD   LASFT
030724' 012726'                                                                                   .WORD   RACMG7
030726'                                   ESCAPE  TST                     ;DO NOT CONTINUE TEST
030726' 104410                                                                                    TRAP    C$ESCAPE
030730' 000262                                                                                    .WORD   L10126-.
030732'                           20$:
030732'                             ENDSEG
030732'                                                                                   10001$:
030732' 104405                                                                                    TRAP    C$ESEG
030734'                             BGNSEG
030734' 104404                                                                                    TRAP    C$BSEG
                                  ;
                                  ;NOW THAT THE UNA KNOWS WHERE THE MICROCODE IS, ISSUE THE GET COMMAND PORT
                                  ;COMMAND SO THE MICROCODE CAN GET LOADED INTO WCS
                                  ;
030736' 012777  000002  147372            MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
030744' 012737  000176  000332'           MOV     #2*SECOND,METER         ;SETUP TIMER
030752' 004737  017330'                   JSR     PC,CHKDNI               ;WAIT FOR 'DNI'
030756' 103022                            BCC     30$                     ;OK
```

```
                                                                       ;ERROR 'DNI' NOT SET!
030760' 012737  000776' 000310'          MOV     #$DNI,BITNAM           ;SETUP ERROR MESSAGE
030766' 012737  001275' 000312'          MOV     #$NSET,BITSTA
030774' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
031002' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
031010'                                  ERRHRD  044,LASFT,MSG1         ;PRINT ERROR MESSAGE
031010' 104456                                                                                  TRAP    C$ERHRD
031012' 000054                                                                                  .WORD   44
031014' 002121'                                                                                 .WORD   LASFT
031016' 012750'                                                                                 .WORD   MSG1
031020'                                  ESCAPE TST                     ;LEAVE TEST
031020' 104410                                                                                  TRAP    C$ESCAPE
031022' 000170                                                                                  .WORD   L10126-.
031024'                         30$:
031024'                           ENDSEG
031024'                                                                             10002$:
031024' 104405                                                                                  TRAP    C$ESEG
031026'                           BGNSEG
031026' 104404                                                                                  TRAP    C$BSEG
                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
031030' 004737  017374'                  JSR     PC,CLRDNI              ;GO CLEAR 'DNI'
031034' 103006                           BCC     40$                    ;OK
                                                                        ;ERROR 'DNI' NOT CLEAR
031036'                                  ERRHRD  045,LASFT,RACMG7       ;PRINT ERROR MESSAGE
031036' 104456                                                                                  TRAP    C$ERHRD
031040' 000055                                                                                  .WORD   45
031042' 002121'                                                                                 .WORD   LASFT
031044' 012726'                                                                                 .WORD   RACMG7
031046'                                  ESCAPE TST                     ;DO NOT CONTINUE TEST
031046' 104410                                                                                  TRAP    C$ESCAPE
031050' 000142                                                                                  .WORD   L10126-.
031052'                         40$:
031052'                           ENDSEG
031052'                                                                             10003$:
031052' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;OK, MICROCODE MODULE 'A' IS LOADED INTO WCS. NOW START IT AND CHECK PCSR1.
                                ;
031054'                           BGNSEG
031054' 104404                                                                                  TRAP    C$BSEG
031056' 012737  000001  000604'          MOV     #LASM,PCBB             ;LOAD PCBB WITH 'LOAD AND START' FUNCTION CODE
031064' 012737  010000  000606'          MOV     #WCSADR+<WCSSIZ/2>,PCBB+2 ;STARTING MICROADDRESS
031072' 012777  000002  147236           MOV     #GETCMD,@PCSR0         ;ISSUE <GET COMMAND> PORT COMMAND
031100' 012737  000176  000332'          MOV     #2*SECOND,METER        ;SETUP TIMER
031106' 004737  017330'                  JSR     PC,CHKDNI              ;GO WAIT FOR 'DNI' TO SET
031112' 103020                           BCC     50$                    ;OK
                                                                        ;ERROR 'DNI' NOT SET!
031114' 012737  000776' 000310'          MOV     #$DNI,BITNAM           ;SETUP ERROR MESSAGE
031122' 012737  001275' 000312'          MOV     #$NSET,BITSTA
031130' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
031136' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
031144'                                  ERRHRD  046,LASFT,MSG1         ;PRINT ERROR MESSAGE
031144' 104456                                                                                  TRAP    C$ERHRD
031146' 000056                                                                                  .WORD   46
```

```
031150' 002121'                                                                                    .WORD   LASFT
031152' 012750'                                                                                    .WORD   MSG1
031154' 017702  147160          50$:    MOV     @PCSR1,R2               ;GET PCSR1 CONTENTS
031160' 042702  140370                  BIC     #^C<SFT!PSTATE>,R2      ;CLEAR ALL BUT SELF TEST AND STATE BITS
031164' 012701  012402                  MOV     #SFTB0!SFTB2!SFTB4!INTST,R1 ;PATTERN THAT SHOULD BE IN PCSR1
031170' 020102                          CMP     R1,R2                   ;IS PCSR1 PATTERN CORRECT?
031172' 001404                          BEQ     60$                     ;YES
031174'                                 ERRHRD  047,LASFT,MSG6          ;NO, PRINT ERROR MESSAGE
031174' 104456                                                                                     TRAP    C$ERHRD
031176' 000057                                                                                     .WORD   47
031200' 002121'                                                                                    .WORD   LASFT
031202' 013212'                                                                                    .WORD   MSG6
031204'                         60$:
031204'                           ENDSEG
031204'                                                                                    10004$:
031204' 104405                                                                                     TRAP    C$ESEG
                                ;
                                ;EVERYTHING WORKED JUST FINE, NOW WE HAVE TO GET THE OPERATIONAL MICROCODE
                                ;GOING AGAIN BEFORE WE LEAVE OTHERWISE EVERYTHING WILL BE SCREWED UP.
                                ;
031206' 004737  020104'                 JSR     PC,REUNA                ;RESET DEUNA TO RESTORE OPERATIONAL
                                                                        ;MICROCODE
031212'                         ENDTST
031212'                                                                                    L10126:
031212' 104401                                                                                     TRAP    C$ETST
```

```
                                        .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
                                        ;
                                        ;*******************************************************************

031214'                                 BGNTST
031214'                                                                                   T13::
                                        ;
                                        ;CHECK TO SEE IF MICROCODE MODULE 'B' HAS BEEN LOADED. IF NOT, LOAD IT INTO
                                        ;THE TOP HALF OF WCS, :TART IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
031214'                                   BGNSEG
031214' 104404                                                                                  TRAP    C$BSEG
031216' 022737  000102  000326'                 CMP     #'B,MICRO               ;HAS MICROCODE MODULE 'B' BEEN LOADED?
031224' 001004                                  BNE     5$                      ;NO
031226' 122777  000001  147104                  CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
031234' 001435                                  BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
031236' 012737  000102  000326' 5$:             MOV     #'B,MICRO               ;GO LOAD MICROCODE MODULE B
031244' 004737  020252'                         JSR     PC,LODMIC
031250' 103564                                  BCS     55$                     ;ERROR
031252' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
031260' 004737  017330'                         JSR     PC,CHKDNI
031264' 103021                                  BCC     20$                     ;OK
031266' 012737  000776' 000310'                 MOV     #$DNI,BITNAM
031274' 012737  001275' 000312'                 MOV     #$NSET,BITSTA
031302' 012737  001340' 000314'                 MOV     #$AFTER,PWHEN
031310' 012737  001355' 000316'                 MOV     #$GTCMD,PCOMND
031316'                                         ERRHRD  048,WCSMEM,MSG1         ;PRINT ERROR MESSAGE
031316' 104456                                                                                  TRAP    C$ERHRD
031320' 000060                                                                                  .WORD   48
031322' 002165'                                                                                 .WORD   WCSMEM
031324' 012750'                                                                                 .WORD   MSG1
```

```
031326' 000535                          BR      55$
031330' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI
031334' 103005                          BCC     25$
031336'                                 ERRHRD  049,WCSMEM,RACMG7       ;DNI DID NOT CLEAR!
031336' 104456                                                                          TRAP    C$ERHRD
031340' 000061                                                                          .WORD   49
031342' 002165'                                                                         .WORD   WCSMEM
031344' 012726'                                                                         .WORD   RACMG7
031346' 000525                          BR      55$
031350'                         25$:
031350'                           ENDSEG
031350'                                                                         10000$:
031350' 104405                                                                          TRAP    C$ESEG

                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE THEN START MICROTEST #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.
                                ;
031352'                           BGNSEG
031352' 104404                                                                          TRAP    C$BSEG
031354' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
031360' 103006                          BCC     30$                     ;OK
031362'                                 ERRHRD  050,WCSMEM,MSG46        ;PRINT ERROR
031362' 104456                                                                          TRAP    C$ERHRD
031364' 000062                                                                          .WORD   50
031366' 002165'                                                                         .WORD   WCSMEM
031370' 016700'                                                                         .WORD   MSG46
031372'                                 ESCAPE  TST                     ;LEAVE TEST
031372' 104410                                                                          TRAP    C$ESCAPE
031374' 000234                                                                          .WORD   L10127-.
031376' 012777  000604' 146736  30$:    MOV     #PCBB,@PCSR2            ;TELL MICROCODE TEST WHERE PCBB IS
031404' 005077  146734                  CLR     @PCSR3
031410' 005037  000604'                 CLR     PCBB+0                  ;CLEAR OUT THE PCBB
031414' 005037  000606'                 CLR     PCBB+2
031420' 005037  000610'                 CLR     PCBB+4
031424' 005037  000612'                 CLR     PCBB+6
031430' 012777  000001  146700          MOV     #1,@PCSR0               ;TELL MICROMONITOR TO EXECUTE...
                                                                        ;MICROTEST #1
031436' 012737  000770  000332'         MOV     #10*SECOND,METER        ;PUT SOME TIME ON THE METER
031444' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR MICROTEST TO FINISH
031450' 103017                          BCC     40$                     ;OK, IT FINISHED
031452' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
031456' 103005                          BCC     35$                     ;NO, OK
031460'                                 ERRHRD  051,WCSMEM,MSG44        ;PRINT ERROR MESSAGE
031460' 104456                                                                          TRAP    C$ERHRD
031462' 000063                                                                          .WORD   51
031464' 002165'                                                                         .WORD   WCSMEM
031466' 016454'                                                                         .WORD   MSG44
031470' 000454                          BR      55$                     ;LEAVE
031472' 012702  000003          35$:    MOV     #3,R2                   ;MICROTEST # THAT IS HUNG
031476'                                 ERRHRD  052,WCSMEM,MSG12        ;TELL MICROTEST HUNG
031476' 104456                                                                          TRAP    C$ERHRD
031500' 000064                                                                          .WORD   52
031502' 002165'                                                                         .WORD   WCSMEM
031504' 013500'                                                                         .WORD   MSG12
```

```
6845  031506' 000445                           BR      55$
6846  031510' 122777  000003  146622  40$:     CMPB    #INERR,@PCSR1          ;DID AN ERROR OCCUR?
6847  031516' 001032                           BNE     50$                    ;NO
6848  031520' 122737  000000  000605'          CMPB    #DATERR,PCBB+1         ;YES, WAS IT A DATA ERROR?
6849  031526' 001003                           BNE     45$                    ;NO
6850  031530' 012702  001444'                  MOV     #$DATER,R2             ;YES, POINT TO DATA ERROR STRING
6851  031534' 000411                           BR      46$
6852  031536' 122737  000001  000605' 45$:     CMPB    #ADRERR,PCBB+1         ;WAS IT AN ADDRESS ERROR
6853  031544' 001003                           BNE     47$                    ;NO
6854  031546' 012702  001457'                  MOV     #$ADRER,R2             ;POINT TO ADDRESS ERPOR STRING
6855  031552' 000402                           BR      46$
6856  031554' 012702  001500'         47$:     MOV     #$PARER,R2             ;POINT TO PARITY ERROR STRING
6857  031560' 013701  000606'         46$:     MOV     PCBB+2,R1              ;GET FAILING ADDRESS
6858  031564' 013703  000610'                  MOV     PCBB+4,R3              ;GET GOOD DATA
6859  031570' 013704  000612'                  MOV     PCBB+6,R4              ;GET BAD DATA
6860  031574'                                  ERRHRD  053,WCSMEM,MSG16       ;PRINT ERROR MESSAGE
6861  031574' 104456                                                                              TRAP    C$ERHRD
6862  031576' 000065                                                                              .WORD   53
6863  031600' 002165'                                                                             .WORD   WCSMEM
6864  031602' 013650'                                                                             .WORD   MSG16
6865  031604' 004737  017374'         50$:     JSR     PC,CLRDNI              ;CLEAR DNI
6866  031610' 103004                           BCC     55$
6867  031612'                                  ERRHRD  054,WCSMEM,RACMG7      ;DNI DID NOT CLEAR!
6868  031612' 104456                                                                              TRAP    C$ERHRD
6869  031614' 000066                                                                              .WORD   54
6870  031616' 002165'                                                                             .WORD   WCSMEM
6871  031620' 012726'                                                                             .WORD   RACMG7
6872  031622' 004737  020104'         55$:     JSR     PC,REUNA               ;RESTORE OPERATIONAL MICROCODE
6873  031626'                           ENDSEG
6874  031626'                                                                             10001$:
6875  031626' 104405                                                                              TRAP    C$ESEG
6876  031630'                         ENDTST
6877  031630'                                                                             L10127:
6878  031630' 104401                                                                              TRAP    C$ETST
6879
```

```
                                        .SBTTL  TEST 14: INTERRUPT VECTOR TEST

                                        ;*****************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE INTERRUPT INTERFACE LOGIC OF THE DEUNA
                                        ;IS CAPABLE OF GENERATING AN INTERUPT 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 SEQUENCE:
                                        ;       1-SETUP INTERRUPT VECTOR
                                        ;       2-CLEAR INTERRUPT FLAG
                                        ;       3-SET INTERRUPT ENABLE IN PCSR0
                                        ;       4-ISSUE NOP PORT COMMAND
                                        ;       5-WAIT FOR DNI SET IN PCSR0
                                        ;       6-VERIFY INTERRUPT FLAG SET
                                        ;       7-VERIFY INTERRUPT AT CORRECT PRIORITY
                                        ;       8-RELEASE INTERRUPT VECTOR
                                        ;       9-CLEAR INTERRUPT ENABLE IN PCSR0
                                        ;       10-WRITE DNI TO CLEAR
                                        ;
                                        ;*****************************************************************

 031632'                                BGNTST
 031632'                                                                          T14::

                                        ;
                                        ;SETUP INTERRUPT VECTOR, CLEAR INTERRUPT FLAG, SET INTERRUPT ENABLE,
                                        ;ISSUE NOP PORT COMMAND AND WAIT FOR 'DNI'
                                        ;
 031632'                                  BGNSEG
 031632' 104404                                                                         TRAP    C$BSEG
 031634'                                        SETVEC  UNAVEC,#UNASRV,UNAPRI   ;SETUP DEUNA INTERRUPT VECTOR
 031634' 013746  000272'                                                                MOV     UNAPRI,-(SP)
 031640' 012746  021656'                                                                MOV     #UNASRV,-(SP)
 031644' 013746  000270'                                                                MOV     UNAVEC,-(SP)
 031650' 012746  000003                                                                 MOV     #3,-(SP)
 031654' 104437                                                                         TRAP    C$SVEC
 031656' 062706  000010                                                                 ADD     #10,SP
 031662' 005037  000670'                CLR     UNAINT                  ;CLEAR UNA INTERRUPTED FLAG
 031666' 012777  000100  146442         MOV     #IE,@PCSR0              ;SET INTERRUPT ENABLE
 031674' 012777  000106  146434         MOV     #IE!PNOP,@PCSR0         ;ISSUE NOP PORT COMMAND
 031702' 012737  000176  000332'        MOV     #2*SECOND,METER         ;SETUP TIMER
 031710' 004737  017330'                JSR     PC,CHKDNI               ;GO WAIT FOR DNI
 031714' 103020                         BCC     10$                     ;OK
                                                                        ;ERROR DNI NOT SET!
 031716' 012737  000776' 000310'        MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
 031724' 012737  001275' 000312'        MOV     #$NSET,BITSTA
 031732' 012737  001340' 000314'        MOV     #$AFTER,PWHEN
 031740' 012737  001403' 000316'        MOV     #$NOP,PCOMND
 031746'                                ERRHRD  055,INTVEC,MSG1         ;PRINT ERROR MESSAGE
 031746' 104456                                                                         TRAP    C$ERHRD
 031750' 000067                                                                         .WORD   55
 031752' 002214'                                                                        .WORD   INTVEC
 031754' 012750'                                                                        .WORD   MSG1
```

```
031756'                          10$:
031756'                            ENDSEG
031756'                                                                     10000$:
031756' 104405                                                                        TRAP    C$ESEG
                                 ;
                                 ;VERIFY THAT INTERRUPT OCCURRED AT CORRECT PRIORITY
                                 ;
031760'                            BGNSEG
031760' 104404                                                                        TRAP    C$BSEG
031762' 012701  000006                   MOV     #6,R1               ;START CPU PRIORITY LOWERING
                                                                     ;FROM PRIORITY 7
031766' 010102                   20$:    MOV     R1,R2               ;GET INTEGER PRIORITY
031770' 072227  000005                   ASH     #5,R2               ;PUT PRIORITY IN CORRECT POSITION
031774'                                  SETPRI  R2                  ;SET NEW PRIORITY
031774' 010200                                                                        MOV     R2,R0
031776' 104441                                                                        TRAP    C$SPRI
032000' 000240                           NOP                         ;LET INTERRUPT OCCUR HERE IF PENDING
032002' 005301                           DEC     R1                  ;LOWER PRIORITY
032004' 100370                           BPL     20$                 ;IF DONE FROM 6-->0 THEN DONE

032006' 005737  000670'                  TST     UNAINT              ;UNA SHOULD HAVE INTERRUPTED BY NOW
032012' 001006                           BNE     30$                 ;OK, GO CHECK THE PRIORITY OF THE INTERUPT
                                                                     ;ERROR! UNA DID NOT INTERRUPT
032014'                                  ERRHRD  056,INTVEC,MSG7     ;PRINT ERROR MESSAGE
032014' 104456                                                                        TRAP    C$ERHRD
032016' 000070                                                                        .WORD   56
032020' 002214'                                                                       .WORD   INTVEC
032022' 013300'                                                                       .WORD   MSG7
032024'                                  ESCAPE TST                  ;LEAVE TEST
032024' 104410                                                                        TRAP    C$ESCAPE
032026' 000072                                                                        .WORD   L10130-.
032030'                          30$:
032030' 013701  000272'                  MOV     UNAPRI,R1           ;GET UNA PRIORITY
032034' 072127  177773                   ASH     #-5,R1              ;MAKE IT AN INTEGER
032040' 005301                           DEC     R1                  ;THIS IS THE CPU PRIORITY WHEN THE...
032042' 072127  000005                   ASH     #5,R1               ;UNA SHOULD HAVE INTERRUPTED
032046' 020137  000674'                  CMP     R1,CPUPRI           ;DID UNA INTERRUPT AT CORRECT PRIORITY?
032052' 001404                           BEQ     40$                 ;YES.
                                                                     ;ERROR! UNA INTERRUPT PRIORITY INCORRECT
032054'                                  ERRHRD  057,INTVEC,MSG8 ;PRINT ERROR MESSAGE
032054' 104456                                                                        TRAP    C$ERHRD
032056' 000071                                                                        .WORD   57
032060' 002214'                                                                       .WORD   INTVEC
032062' 013322'                                                                       .WORD   MSG8
032064'                          40$:
032064'                            ENDSEG
032064'                                                                     10001$:
032064' 104405                                                                        TRAP    C$ESEG
032066'                                  CLRVEC  UNAVEC              ;RELEASE INTERRUPT VECTOR
032066' 013700  000270'                                                               MOV     UNAVEC,R0
032072' 104436                                                                        TRAP    C$CVEC
032074' 012777  000000  146234           MOV     #0,@PCSR0           ;CLEAR INTERRUPT ENABLE
032102' 004737  017374'                  JSR     PC,CLRDNI           ;CLEAR DNI
032106' 103004                           BCC     50$
032110'                                  ERRHRD  058,INTVEC,RACMG7   ;ERROR! DNI DID NOT CLEAR
032110' 104456                                                                        TRAP    C$ERHRD
```

```
  6992  032112' 000072                                              .WORD   58
  6993  032114' 002214'                                             .WORD   INTVEC
  6994  032116' 012726'                                             .WORD   RACMG7
  6995  032120'                 50$:
  6996  032120'                 ENDTST
  6997  032120'                                              L10130:
  6998  032120' 104401                                              TRAP    C$ETST
```

```
                                .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
                                ;
                                ;***********************************************************************

032122'                         BGNTST
032122'                                                                                 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.
                                ;
032122'                           BGNSEG
032122' 104404                                                                                  TRAP    C$BSEG
032124' 022737  000103  000326'         CMP     #'C,MICRO               ;HAS MICROCODE MODULE C BEEN LOADED?
032132' 001004                          BNE     5$                      ;NO
032134' 122777  000001  146176          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
032142' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
032144' 012737  000103  000326' 5$:     MOV     #'C,MICRO               ;SETUP TO LOAD MODULE C
032152' 004737  020252'                 JSR     PC,LODMIC               ;GO LOAD MICROMODULE C
032156' 103002                          BCC     10$                     ;SUCCESS
032160'                                 ESCAPE  TST                     ;ERROR OCCURRED LEAVE
032160' 104410                                                                                  TRAP    C$ESCAPE
032162' 000444                                                                                  .WORD   L10131-.
032164' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
032172' 004737  017330'                 JSR     PC,CHKDNI               ;GO WAIT FOR DNI TO SET AFTER THE
```

```
                                                                            ;LOAD AND START FUNCTION
032176' 103022                            BCC      20$                      ;OK
032200' 012737   000776' 000310'          MOV      #$DNI,BITNAM             ;SETUP ERROR MESSAGE
032206' 012737   001275' 000312'          MOV      #$NSET,BITSTA
032214' 012737   001340' 000314'          MOV      #$AFTER,PWHEN
032222' 012737   001355' 000316'          MOV      #$GTCMD,PCOMND
032230'                                   ERRHRD   059,INTBIT,MSG1          ;PRINT ERROR MESSAGE
032230' 104456                                                                                       TRAP     C$ERHRD
032232' 000073                                                                                       .WORD    59
032234' 002251'                                                                                      .WORD    INTBIT
032236' 012750'                                                                                      .WORD    MSG1
032240'                                   ESCAPE TST
032240' 104410                                                                                       TRAP     C$ESCAPE
032242' 000364                                                                                       .WORD    L10131-.
032244' 004737   017374'          20$:    JSR      PC,CLRDNI                ;CLEAR DNI
032250' 103006                            BCC      25$
032252'                                   ERRHRD   060,INTBIT,RACMG7        ;ERROR! DNI DID NOT CLEAR
032252' 104456                                                                                       TRAP     C$ERHRD
032254' 000074                                                                                       .WORD    60
032256' 002251'                                                                                      .WORD    INTBIT
032260' 012726'                                                                                      .WORD    RACMG7
032262'                                   ESCAPE TST
032262' 104410                                                                                       TRAP     C$ESCAPE
032264' 000342                                                                                       .WORD    L10131-.
032266'                           25$:    SETVEC   UNAVEC,#UNASRV,UNAPRI    ;SETUP DEUNA INTERRUPT VECTOR
032266' 013746   000272'                                                                             MOV      UNAPRI,-(SP)
032272' 012746   021656'                                                                             MOV      #UNASRV,-(SP)
032276' 013746   000270'                                                                             MOV      UNAVEC,-(SP)
032302' 012746   000003                                                                              MOV      #3,-(SP)
032306' 104437                                                                                       TRAP     C$SVEC
032310' 062706   000010                                                                              ADD      #10,SP
032314'                             ENDSEG
032314'                                                                                      10000$:
032314' 104405                                                                                       TRAP     C$ESEG
                                  ;
                                  ;THE FOLLOWING LOOP WILL BE EXECUTED 6 TIMES- ONCE FOR EACH BIT 10 THRU 15
                                  ;OF PCSR0. IT WAITS FOR THE MICROMONITOR TO BECOME ACTIVE THEN CALLS
                                  ;MICROTEST #1 TO SET A BIT IN PCSR0 AS DEFINED BY THE PARAMETER IN PCBB+0.
                                  ;THIS OPERATION SHOULD CAUSE AN INTERRUPT WHICH WILL BE REFLECTED BY THE
                                  ;VARIABLE 'UNAINT' WHICH IS SET BY THE UNA INTERRUPT SERVICE ROUTINE. UNAINT
                                  ;IS LOADED WITH THE VALUE OF PCSR0 AT THE TIME OF THE INTERRUPT. THIS WAY
                                  ;THE BIT THAT CAUSED THE INTERRUPT CAN BE CHECKED.
                                  ;
032316' 005037   000304'                  CLR      CSRNUM                   ;CHECKING PCSR0
032322' 012737   001275' 000312'          MOV      #$NSET,BITSTA            ;CHECKING FOR SET BITS
032330' 012701   002000                   MOV      #BIT10,R1                ;START WITH BIT 10
032334' 012737   000012  000306'          MOV      #10.,BITNUM
032342'                           26$:
032342'                             BGNSEG
032342' 104404                                                                                       TRAP     C$BSEG
032344' 004737   017776'                  JSR      PC,CHKMON                ;WAIT FOR MICROMONITOR
032350' 103006                            BCC      30$                      ;OK
032352'                                   ERRHRD   061,INTBIT,MSG46         ;PRINT ERROR
032352' 104456                                                                                       TRAP     C$ERHRD
032354' 000075                                                                                       .WORD    61
032356' 002251'                                                                                      .WORD    INTBIT
```

```
032360' 016700'                                                                         .WORD   MSG46
032362'                                  ESCAPE  TST                   ;LEAVE TEST
032362' 104410                                                                          TRAP    C$ESCAPE
032364' 000242                                                                          .WORD   L10131-.
032366' 005037  000670'          30$:    CLR     UNAINT                ;CLEAR UNA INTERRUPTED FLAG
032372' 010137  000604'                  MOV     R1,PCBB+0             ;TELL MICROMONITOR WHICH BIT TO SET
032376' 012737  000077  000332'          MOV     #1*SECOND,METER       ;PUT SOME TIME ON THE METER
032404' 012777  000100  145724           MOV     #IE,@PCSRO            ;SET INTERRUPT ENABLE AND...
                                                                       ;CLEAR SELF TEST BITS IN PCSR1
032412' 012777  000101  145716           MOV     #IE!1,@PCSRO          ;INVOKE INTERRUPT BIT MICROTEST
032420' 004737  017276'                  JSR     PC,TIMON              ;TURN ON THE TIMER
032424'                                  SETPRI  #PRI04                ;LOWER PRIORITY TO ALLOW UNA INTERRUPT
032424' 012700  000200                                                                  MOV     #PRI04,R0
032430' 104441                                                                          TRAP    C$SPRI
032432' 005737  000670'          35$:    TST     UNAINT                ;HAS INTERRUPT OCCURRED YET?
032436' 001022                           BNE     40$                   ;YES
032440' 005737  000332'                  TST     METER                 ;NO, HAS METER EXPIRED?
032444' 001372                           BNE     35$                   ;NOT YET
032446' 004737  017314'                  JSR     PC,TIMOFF             ;TIMER HAS EXPIRED, SHUT IT OFF
                                                                       ;ERROR, NO UNA INTERRUPT!
032452' 013703  000306'                  MOV     BITNUM,R3             ;GET WHICH BIT
032456' 006303                           ASL     R3                    ;MAKE IT A BYTE OFFSET
032460' 012737  000356' 000310'          MOV     #BNAMTO,BITNAM        ;POINT TO TABLE OF BIT MNEMONICS
032466' 060337  000310'                  ADD     R3,BITNAM             ;INDEX INTO TABLE OF BIT MNEMONICS
032472'                                  ERRHRD  062,INTBIT,MSG9       ;PRINT ERROR MESSAGE
032472' 104456                                                                          TRAP    C$ERHRD
032474' 000076                                                                          .WORD   62
032476' 002251'                                                                         .WORD   INTBIT
032500' 013344'                                                                         .WORD   MSG9
032502' 000402                           BR      45$
032504' 004737  017314'          40$:    JSR     PC,TIMOFF             ;INTERRUPT OCCURRED, SHUT OFF THE TIMER
032510' 013703  000670'          45$:    MOV     UNAINT,R3             ;GET SAVED PCSRO CONTENTS
032514' 042703  000377                   BIC     #377,R3               ;CLEAR UNWANTED BITS
032520' 020103                           CMP     R1,R3                 ;DID CORRECT BIT CAUSE INTERRUPT?
032522' 001404                           BEQ     50$                   ;YES, OK
                                                                       ;ERROR, INCORRECT BIT CAUSED INTERRUPT!
032524'                                  ERRHRD  063,INTBIT,RACMG2     ;PRINT ERROR MESSAGE
032524' 104456                                                                          TRAP    C$ERHRD
032526' 000077                                                                          .WORD   63
032530' 002251'                                                                         .WORD   INTBIT
032532' 012574'                                                                         .WORD   RACMG2
032534'                          50$:
032534'                            ENDSEG
032534'                                                                        10001$:
032534' 104405                                                                          TRAP    C$ESEG
                                 ;
                                 ;NOW WRITE '1' TO CLEAR THE BIT THAT CAUSED THE INTERRUPT
                                 ;
032536'                            BGNSEG
032536' 104404                                                                          TRAP    C$BSEG
032540' 010377  145572                   MOV     R3,@PCSRO             ;WRITE '1' TO CLEAR INTERRUPT BIT
032544' 017704  145566                   MOV     @PCSRO,R4             ;READ IT BACK
032550' 030403                           BIT     R4,R3                 ;IS BIT CLEARED?
032552' 001420                           BEQ     55$                   ;YES
                                                                       ;ERROR, BIT DID NOT CLEAR!
032554' 012705  000017                   MOV     #15.,R5               ;
```

```
032560' 006303                  52$:    ASL     R3                      ;IS THIS THE BIT WE SET?
032562' 103402                          BCS     53$                     ;YES
032564' 005305                          DEC     R5                      ;NO
032566' 000774                          BR      52$
032570' 006305                  53$:    ASL     R5                      ;MAKE IT A BYTE OFFSET
032572' 012737  000356' 000310'         MOV     #BNAMTO,BITNAM          ;GET POINTER TO BIT NAME MNEMONICS
032600' 060537  000310'                 ADD     R5,BITNAM               ;INDEX INTO BIT NAME TABLE
032604'                                 ERRHRD  064,INTBIT,MSG33        ;PRINT ERROR MESSAGE
032604' 104456                                                                                  TRAP    C$ERHRD
032606' 000100                                                                                  .WORD   64
032610' 002251'                                                                                 .WORD   INTBIT
032612' 015606'                                                                                 .WORD   MSG33
032614'                         55$:
032614'                           ENDSEG
032614'                                                                                 10002$:
032614' 104405                                                                                  TRAP    C$ESEG
032616' 005237  000306'                 INC     BITNUM                  ;SETUP FOR NEXT BIT
032622' 006301                          ASL     R1                      ;POINT TO NEXT HIGHER BIT
032624' 103246                          BCC     26$                     ;IF BIT 15 NOT DONE YET GO ON
                                                                        ;ELSE ALL DONE
032626'                         ENDTST
032626'                                                                                 L10131:
032626' 104401                                                                                  TRAP    C$ETST
```

```
                                 .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
                                 ;
                                 ;*******************************************************************************

032630'                          BGNTST
032630'                                                                                    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.
                                 ;
032630'                            BGNSEG
032630' 104404                                                                                      TRAP    C$BSEG
032632' 022737  000103  000326'           CMP      #'C,MICRO               ;HAS MICROCODE MODULE C BEEN LOADED?
032640' 001004                            BNE      5$                      ;NO
032642' 122777  000001  145470            CMPB     #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
032650' 001440                            BEQ      20$                     ;YES SKIP LOADING THE MICROMODULE
032652' 012737  000103  000326' 5$:       MOV      #'C,MICRO               ;NO
032660' 004737  020252'                   JSR      PC,LODMIC               ;GO LOAD MICROCODE MODULE C
032664' 103002                            BCC      10$                     ;OK
032666'                                   ESCAPE TST                       ;ERROR OCCURRED LOADING MICROCODE
032666' 104410                                                                                      TRAP    C$ESCAPE
032670' 000254                                                                                      .WORD   L10132-.
032672' 012737  000176  000332' 10$:      MOV      #2*SECOND,METER         ;WAIT FOR MICROMONITOR TO TAKE OVER
032700' 004737  017330'                   JSR      PC,CHKDNI
032704' 103022                            BCC      20$
032706' 012737  000776' 000310'           MOV      #$DNI,BITNAM
032714' 012737  001275' 000312'           MOV      #$NSET,BITSTA
032722' 012737  001340' 000314'           MOV      #$AFTER,PWHEN
032730' 012737  001355' 000316'           MOV      #$GTCMD,PCOMND
032736'                                   ERRHRD   065,TIMTST,MSG1
032736' 104456                                                                                      TRAP    C$ERHRD
032740' 000101                                                                                      .WORD   65
032742' 002311'                                                                                     .WORD   TIMTST
```

```
032744' 012750'                                                                                      .WORD   MSG1
032746'                                  ESCAPE TST
032746' 104410                                                                                       TRAP    C$ESCAPE
032750' 000174                                                                                       .WORD   L10132-.
032752' 004737  017374'         20$:     JSR     PC,CLRDNI               ;CLEAR DNI
032756' 103006                           BCC     25$
032760'                                  ERRHRD  066,TIMTST,RACMG7       ;ERROR DNI DID NOT CLEAR!
032760' 104456                                                                                       TRAP    C$ERHRD
032762' 000102                                                                                       .WORD   66
032764' 002311'                                                                                      .WORD   TIMTST
032766' 012726'                                                                                      .WORD   RACMG7
032770'                                  ESCAPE TST
032770' 104410                                                                                       TRAP    C$ESCAPE
032772' 000152                                                                                       .WORD   L10132-.
032774'                         25$:
032774'                           ENDSEG
032774'                                                                                     10000$:
032774' 104405                                                                                       TRAP    C$ESEG
                                ;
                                ;WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR, SETUP OUR TIMEOUT TO BE
                                ;12 SECONDS (THIS GIVES A BETTER RESOLUTION THAN 1 SECOND), START THE MICROTEST
                                ;BY LOADING THE COMMAND FIELD OF PCSR0 WITH THE MICROTEST # TO EXECUTE.
                                ;CHECK FOR 'DNI' TO BE SET IN LESS THAN 12 SECONDS.
                                ;
032776'                           BGNSEG
032776' 104404                                                                                       TRAP    C$BSEG
033000' 004737  017776'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
033004' 103006                           BCC     30$                     ;OK
033006'                                  ERRHRD  068,TIMTST,MSG46        ;PRINT ERROR
033006' 104456                                                                                       TRAP    C$ERHRD
033010' 000104                                                                                       .WORD   68
033012' 002311'                                                                                      .WORD   TIMTST
033014' 016700'                                                                                      .WORD   MSG46
033016'                                  ESCAPE  TST                     ;LEAVE TEST
033016' 104410                                                                                       TRAP    C$ESCAPE
033020' 000124                                                                                       .WORD   L10132-.
033022' 012737  001356  000332' 30$:     MOV     #750.,METER             ;TIMEOUT = 12 SECONDS
033030' 012777  000002  145300           MOV     #2,@PCSR0               ;START MICROTEST #2
033036' 004737  017330'                  JSR     PC,CHKDNI               ;WAIT FOR DNI
033042' 103006                           BCC     40$                     ;OK IT FINISHED IN TIME
033044'                                  ERRHRD  069,TIMTST,MSG10        ;NO TIMER INTERRUPT
033044' 104456                                                                                       TRAP    C$ERHRD
033046' 000105                                                                                       .WORD   69
033050' 002311'                                                                                      .WORD   TIMTST
033052' 013372'                                                                                      .WORD   MSG10
033054'                                  ESCAPE TST
033054' 104410                                                                                       TRAP    C$ESCAPE
033056' 000066                                                                                       .WORD   L10132-.
                                ;
                                ;OK THE TIMER INTERRUPTED IN LESS THAN 12 SECONDS, SO NOW CHECK TO SEE IF IT
                                ;HAPPENED IN LESS THAN 8.
                                ;
033060' 023727  000332' 000764  40$:     CMP     METER,#500.             ;DID INTERRUPT OCCUR BEFORE 8 SECS.?
033066' 003416                           BLE     60$                     ;NO, OK
033070' 012700  001161                   MOV     #625.,R0
033074' 163700  000332'                  SUB     METER,R0                ;CALC HOW MUCH TIME
```

```
033100' 005001                   CLR     R1                      ;CONTAINS SECS
033102' 162700  000077   50$:    SUB     #63.,R0                 ;FIND OUT HOW MAY SECS IT TOOK
033106' 103402                   BCS     55$
033110' 005201                   INC     R1
033112' 000773                   BR      50$
033114'                  55$:    ERRHRD  070,TIMTST,MSG11        ;PRINT ERROR MESSAGE
033114' 104456                                                                       TRAP     C$ERHRD
033116' 000106                                                                       .WORD    70
033120' 002311'                                                                      .WORD    TIMTST
033122' 013414'                                                                      .WORD    MSG11
033124' 004737  017374'  60$:    JSR     PC,CLRDNI               ;CLEAR DNI
033130' 103004                   BCC     70$
033132'                          ERRHRD  071,TIMTST,RACMG7       ;ERROR DNI DID NOT CLEAR
033132' 104456                                                                       TRAP     C$ERHRD
033134' 000107                                                                       .WORD    71
033136' 002311'                                                                      .WORD    TIMTST
033140' 012726'                                                                      .WORD    RACMG7
033142'                  70$:
033142'                    ENDSEG
033142'                                                                    10001$:
033142' 104405                                                                       TRAP     C$ESEG
033144'                  ENDTST
033144'                                                                    L10132:
033144' 104401                                                                       TRAP     C$ETST
```

```
                                        .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
                                        ;
                                        ;*********************************************************************

 033146'                                BGNTST
 033146'                                                                             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.
                                        ;
 033146'                                 BGNSEG
 033146' 104404                                                                             TRAP    C$BSEG
 033150' 022737  000103  000326'                CMP     #'C,MICRO               ;HAS MICROCODE MODULE 'C' BEEN LOADED?
 033156' 001004                                 BNE     5$                      ;NO
 033160' 122777  000001  145152                 CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 033166' 001440                                 BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 033170' 012737  000103  000326' 5$:            MOV     #'C,MICRO               ;GO LOAD MICROCODE MODULE C
 033176' 004737  020252'                        JSR     PC,LODMIC
 033202' 103002                                 BCC     10$                     ;OK
 033204'                                        ESCAPE  TST
 033204' 104410                                                                             TRAP    C$ESCAPE
 033206' 000404                                                                             .WORD   L10133-.
 033210' 012737  000176  000332' 10$:           MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
```

```
033216' 004737  017330'                  JSR     PC,CHKDNI
033222' 103022                           BCC     20$                     ;OK
033224' 012737  000776' 000310'          MOV     #$DNI,BITNAM
033232' 012737  001275' 000312'          MOV     #$NSET,BITSTA
033240' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
033246' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
033254'                                  ERRHRD  072,LNKMEM,MSG1 ;PRINT ERROR MESSAGE
033254' 104456                                                                              TRAP    C$ERHRD
033256' 000110                                                                              .WORD   72
033260' 002333'                                                                             .WORD   LNKMEM
033262' 012750'                                                                             .WORD   MSG1
033264'                                  ESCAPE TST
033264' 104410                                                                              TRAP    C$ESCAPE
033266' 000324                                                                              .WORD   L10133-.
033270' 004737  017374'          20$:    JSR     PC,CLRDNI               ;CLEAR DNI
033274' 103006                           BCC     25$
033276'                                  ERRHRD  073,LNKMEM,RACMG7
033276' 104456                                                                              TRAP    C$ERHRD
033300' 000111                                                                              .WORD   73
033302' 002333'                                                                             .WORD   LNKMEM
033304' 012726'                                                                             .WORD   RACMG7
033306'                                  ESCAPE TST
033306' 104410                                                                              TRAP    C$ESCAPE
033310' 000302                                                                              .WORD   L10133-.
033312'                          25$:
033312'                            ENDSEG
033312'                                                                            10000$:
033312' 104405                                                                              TRAP    C$ESEG

                                 ;
                                 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. CLEAR THE
                                 ;LOCATIONS FOR ERROR INFORMATION, LOAD COMMAND FIELD OF PCSR0 WITH A 3
                                 ;CAUSING THE MICROMONITOR TO EXECUTE MICROTEST #3. THIS WILL START THE
                                 ;EXECUTION OF THE MICROSUBTEST SEQUENCE OF MEMORY TESTS. DNI WILL SET
                                 ;WHEN THE TEST IS COMPLETE
                                 ;
033314'                            BGNSEG
033314' 104404                                                                              TRAP    C$BSEG
033316' 004737  017776'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
033322' 103006                           BCC     30$                     ;OK
033324'                                  ERRHRD  074,LNKMEM,MSG46        ;PRINT ERROR
033324' 104456                                                                              TRAP    C$ERHRD
033326' 000112                                                                              .WORD   74
033330' 002333'                                                                             .WORD   LNKMEM
033332' 016700'                                                                             .WORD   MSG46
033334'                                  ESCAPE  TST                     ;LEAVE TEST
033334' 104410                                                                              TRAP    C$ESCAPE
033336' 000254                                                                              .WORD   L10133-.
033340' 012777  000604' 144774   30$:    MOV     #PCBB,@PCSR2            ;TELL MICROCODE TEST WHERE PCBB IS
033346' 005077  144772                   CLR     @PCSR3
033352' 005037  000604'                  CLR     PCBB+0                  ;CLEAR OUT THE PCBB
033356' 005037  000606'                  CLR     PCBB+2
033362' 005037  000610'                  CLR     PCBB+4
033366' 005037  000612'                  CLR     PCBB+6
033372' 012777  000003  144736           MOV     #3,@PCSR0               ;TELL MICROMONITOR TO EXECUTE...
                                                                         ;MICROTEST #3
033400' 012737  000770  000332'          MOV     #10*SECOND,METER        ;PUT SOME TIME ON THE METER
```

```
033406' 004737  017330'                  JSR     PC,CHKDNI               ;WAIT FOR MICROTEST TO FINISH
033412' 103021                           BCC     40$                     ;OK, IT FINISHED
033414' 004737  020050'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
033420' 103006                           BCC     35$                     ;NO, OK
033422'                                  ERRHRD  075,LNKMEM,MSG44        ;PRINT ERROR MESSAGE
033422' 104456                                                                                TRAP     C$ERHRD
033424' 000113                                                                                .WORD    75
033426' 002333'                                                                               .WORD    LNKMEM
033430' 016454'                                                                               .WORD    MSG44
033432'                                  ESCAPE  TST                     ;LEAVE TEST
033432' 104410                                                                                TRAP     C$ESCAPE
033434' 000156                                                                                .WORD    L10133-.
033436' 012702  000003           35$:    MOV     #3,R2                   ;MICROTEST #
033442'                                  ERRHRD  076,LNKMEM,MSG12        ;TELL MICROTEST HUNG
033442' 104456                                                                                TRAP     C$ERHRD
033444' 000114                                                                                .WORD    76
033446' 002333'                                                                               .WORD    LNKMEM
033450' 013500'                                                                               .WORD    MSG12
033452'                                  ESCAPE  TST
033452' 104410                                                                                TRAP     C$ESCAPE
033454' 000136                                                                                .WORD    L10133-.
                                 ;
                                 ;MICROTEST IS COMPLETE, NOW CHECK FOR AN ERROR CONDITION
                                 ;
033456' 122777  000003  144654   40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
033464' 001027                           BNE     47$                     ;NO
033466' 122737  000000  000605'          CMPB    #DATERR,PCBB+1          ;YES, WAS IT A DATA ERROR?
033474' 001003                           BNE     45$                     ;NO
033476' 012702  001444'                  MOV     #$DATER,R2              ;YES, POINT TO DATA ERROR STRING
033502' 000406                           BR      46$
033504' 122737  000001  000605'  45$:    CMPB    #ADRERR,PCBB+1          ;WAS IT AN ADDRESS ERROR
033512' 001014                           BNE     47$                     ;NO
033514' 012702  001457'                  MOV     #$ADRER,R2              ;POINT TO ADDRESS ERROR STRING
033520' 013701  000606'          46$:    MOV     PCBB+2,R1               ;GET FAILING ADDRESS
033524' 013703  000610'                  MOV     PCBB+4,R3               ;GET GOOD DATA
033530' 013704  000612'                  MOV     PCBB+6,R4               ;GET BAD DATA
033534'                                  ERRHRD  077,LNKMEM,MSG16        ;PRINT ERROR MESSAGE
033534' 104456                                                                                TRAP     C$ERHRD
033536' 000115                                                                                .WORD    77
033540' 002333'                                                                               .WORD    LNKMEM
033542' 013650'                                                                               .WORD    MSG16
033544' 032777  010000  144564   47$:    BIT     #PARERR,@PCSR0          ;DID A PARITY ERROR OCCUR?
033552' 001407                           BEQ     50$                     ;NO
033554'                                  ERRHRD  078,LNKMEM,MSG43        ;YES PRINT ERROR
033554' 104456                                                                                TRAP     C$ERHRD
033556' 000116                                                                                .WORD    78
033560' 002333'                                                                               .WORD    LNKMEM
033562' 016432'                                                                               .WORD    MSG43
033564' 012777  010000  144544           MOV     #PARERR,@PCSR0          ;CLEAR PARITY ERROR FLAG


                                 ;
                                 ;WRITE ONE TO CLEAR THE DNI BIT
                                 ;
033572' 004737  017374'          50$:    JSR     PC,CLRDNI               ;CLEAR DNI
033576' 103004                           BCC     55$
033600'                                  ERRHRD  079,LNKMEM,RACMG7
```

```
 7494  033600' 104456                                                          TRAP    C$ERHRD
 7495  033602' 000117                                                          .WORD   79
 7496  033604' 002333'                                                         .WORD   LNKMEM
 7497  033606' 012726'                                                         .WORD   RACMG7
 7498  033610'                     55$:
 7499  033610'                       ENDSEG
 7500  033610'                                                         10001$:
 7501  033610' 104405                                                          TRAP    C$FSEG
 7502  033612'                     ENDTST
 7503  033612'                                                         L10123:
 7504  033612' 104401                                                          TRAP    C$ETST
 7505
```

```
                                  .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 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 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
                                  ;
                                  ;**********************************************************************

033614'                           BGNTST
033614'                                                                                   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.
                                  ;
033614'                            BGNSEG
033614' 104404                                                                                    TRAP   C$BSEG
033616' 022737  000103  000326'           CMP     #'C,MICRO       ;HAS MICROCODE MODULE 'C' BEEN LOADED?
033624' 001004                            BNE     5$                      ;NO
033626' 122777  000001  144504            CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
033634' 001440                            BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
033636' 012737  000103  000326' 5$:       MOV     #'C,MICRO       ;NO
033644' 004737  020252'                   JSR     PC,LODMIC       ;GO LOAD MICROCODE MODULE C
033650' 103002                            BCC     10$             ;OK
033652'                                   ESCAPE TST
033652' 104410                                                                                    TRAP   C$ESCAPE
033654' 000314                                                                                    .WORD  L10134-.
033656' 012737  000176  000332' 10$:      MOV     #2*SECOND,METER
033664' 004737  017330'                   JSR     PC,CHKDNI       ;WAIT FOR MICROMONITOR TO TAKE OVER
033670' 103022                            BCC     20$             ;OK
033672' 012737  000776' 000310'           MOV     #$DNI,BITNAM
033700' 012737  001275' 000312'           MOV     #$NSET,BITSTA
033706' 012737  001340' 000314'           MOV     #$AFTER,PWHEN
033714' 012737  001355' 000316'           MOV     #$GTCMD,PCOMND
033722'                                   ERRHRD  080,DMATO,MSG1
033722' 104456                                                                                    TRAP   C$ERHRD
033724' 000120                                                                                    .WORD  80
033726' 002363'                                                                                   .WORD  DMATO
033730' 012750'                                                                                   .WORD  MSG1
033732'                                   ESCAPE TST
033732' 104410                                                                                    TRAP   C$ESCAPE
033734' 000234                                                                                    .WORD  L10134-.
033736' 004737  017374'         20$:      JSR     PC,CLRDNI               ;CLEAR DNI
```

```
033742' 103006                          BCC     25$
033744'                                 ERRHRD  081,DMATO,RACMG7
033744' 104456                                                                               TRAP    C$ERHRD
033746' 000121                                                                               .WORD   81
033750' 002363'                                                                              .WORD   DMATO
033752' 012726'                                                                              .WORD   RACMG7
033754'                                 ESCAPE TST
033754' 104410                                                                               TRAP    C$ESCAPE
033756' 000212                                                                               .WORD   L10134-.
033760'                         25$:
033760'                           ENDSEG
033760'                                                                               10000$:
033760' 104405                                                                               TRAP    C$ESEG

                                ;
                                ;POINT TO LIST OF DATA PATTERNS TO USE, THERE ARE FIVE ENTRIES IN THE LIST
                                ;SO THE LOOP WILL BE EXECUTED 5 TIMES ONCE FOR EACH DATA PATTERN.
                                ;
033762' 012701  000516'                 MOV     #PATERN,R1      ;GET ADDRESS OF DATA PATTERNS
033766' 012705  000005                  MOV     #5,R5           ;NUMBER OF DATA PATTERNS
033772'                         30$:
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE, LOAD PCBB+0 WITH
                                ;A DATA PATTERN, LOAD THE COMMAND FIELD OF PCSRO WITH 4 TO START THE EXECUTION
                                ;OF MICROTEST #4, DNI SETS WHEN IT IS COMPLETE
                                ;
033772'                           BGNSEG
033772' 104404                                                                               TRAP    C$BSEG
033774' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
034000' 103006                          BCC     35$                     ;OK
034002'                                 ERRHRD  082,DMATO,MSG46         ;PRINT ERROR
034002' 104456                                                                               TRAP    C$ERHRD
034004' 000122                                                                               .WORD   82
034006' 002363'                                                                              .WORD   DMATO
034010' 016700'                                                                              .WORD   MSG46
034012'                                 ESCAPE  TST                     ;LEAVE TEST
034012' 104410                                                                               TRAP    C$ESCAPE
034014' 000154                                                                               .WORD   L10134-.
034016' 012137  000604'         35$:    MOV     (R1)+,PCBB+0    ;GET A DATA PATTERN
034022' 012777  000004  144306          MOV     #4,@PCSRO       ;START MICROTEST #4
034030' 012737  000176  000332'         MOV     #2*SECOND,METER ;PUT SOME TIME ON THE METER
034036' 004737  017330'                 JSR     PC,CHKDNI       ;WAIT FOR DNI
034042' 103021                          BCC     40$             ;OK
034044' 004737  020050'                 JSR     PC,CHKINT       ;SEE IF ANY ERROR INTERRUPTS OCCURRED
034050' 103006                          BCC     36$             ;NO, OK
034052'                                 ERRHRD  083,DMATO,MSG44 ;PRINT ERROR MESSAGE
034052' 104456                                                                               TRAP    C$ERHRD
034054' 000123                                                                               .WORD   83
034056' 002363'                                                                              .WORD   DMATO
034060' 016454'                                                                              .WORD   MSG44
034062'                                 ESCAPE  TST             ;LEAVE TEST
034062' 104410                                                                               TRAP    C$ESCAPE
034064' 000104                                                                               .WORD   L10134-.
034066' 012702  000004          36$:    MOV     #4,R2           ;MICROTEST #
034072'                                 ERRHRD  084,DMATO,MSG12 ;TELL MICROTEST HUNG
034072' 104456                                                                               TRAP    C$ERHRD
034074' 000124                                                                               .WORD   84
```

```
034076' 002363'                                                                          .WORD   DMATO
034100' 013500'                                                                          .WORD   MSG12
034102'                                 ESCAPE  TST
034102' 104410                                                                           TRAP    C$ESCAPE
034104' 000064                                                                           .WORD   L10134-.
                                ;
                                ;OK NOW CHECK TO SEE IF DATA READ IS SAME AS THE DATA WRITTEN
                                ;REMEMBER BIT 0 OF DMATO IS NOT USED
                                ;
034106' 013703  000604'         40$:    MOV     PCBB+0,R3       ;GET ORIGINAL DATA PATTERN
034112' 042703  000001                  BIC     #BIT0,R3        ;STRIP LSB
034116' 020337  000606'                 CMP     R3,PCBB+2       ;SEE IF DATA WRITTEN = DATA READ
034122' 001407                          BEQ     50$             ;YES
034124' 013704  000606'                 MOV     PCBB+2,R4       ;NO, ERROR
034130'                                 ERRHRD  085,DMATO,MSG13 ;PRINT ERROR
034130' 104456                                                                           TRAP    C$ERHRD
034132' 000125                                                                           .WORD   85
034134' 002363'                                                                          .WORD   DMATO
034136' 013524'                                                                          .WORD   MSG13
034140'                          ENDSEG
034140'                                                                          10001$:
034140' 104405                                                                           TRAP    C$ESEG
                                ;
                                ;WRITE ONE TO CLEAR DNI BIT
                                ;
034142' 004737  017374'         50$:    JSR     PC,CLRDNI       ;CLEAR DNI
034146' 103006                          BCC     55$
034150'                                 ERRHRD  086,DMATO,RACMG7;ERROR DNI DID NOT CLEAR
034150' 104456                                                                           TRAP    C$ERHRD
034152' 000126                                                                           .WORD   86
034154' 002363'                                                                          .WORD   DMATO
034156' 012726'                                                                          .WORD   RACMG7
034160'                                 ESCAPE  TST
034160' 104410                                                                           TRAP    C$ESCAPE
034162' 000006                                                                           .WORD   L10134-.
034164' 005305                  55$:    DEC     R5              ;ANY MORE DATA PATTERNS?
034166' 001301                          BNE     30$             ;YES
034170'                         ENDTST
034170'                                                                          L10134:
034170' 104401                                                                           TRAP    C$ETST
```

```
                                        .SBTTL  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 SEQUENCE:
                                        ;       1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO
                                        ;       2-LOAD PCBB+0 WITH ADDRESS OF LINK MEMORY TO LOAD
                                        ;       3-START MICROCODE
                                        ;       4-WAIT FOR 'DNI'
                                        ;       5-VERIFY PCBB+2 = PCBB+0
                                        ;       6-CHANGE PCBB+0 BY 1K CHUNKS
                                        ;       7-REPEAT 2-6
                                        ;
                                        ;*********************************************************************

 034172'                                BGNTST
 034172'                                                                                T19::

                                        ;
                                        ;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.
                                        ;
 034172'                                  BGNSEG
 034172' 104404                                                                          TRAP    C$BSEG
 034174' 022737  000103  000326'              CMP     #'C,MICRO              ;HAS MICRO MODULE C BEEN LOADED?
 034202' 001004                               BNE     5$                     ;NO
 034204' 122777  000001  144126               CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
 034212' 001440                               BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
 034214' 012737  000103  000326' 5$:          MOV     #'C,MICRO              ;NO, LOAD MICRO MODULE C
 034222' 004737  020252'                      JSR     PC,LODMIC
 034226' 103002                               BCC     10$                    ;OK
 034230'                                      ESCAPE TST
 034230' 104410                                                                          TRAP    C$ESCAPE
 034232' 000316                                                                          .WORD   L10135-.
 034234' 012737  000176  000332' 10$:         MOV     #2*SECOND,METER        ;PUT SOME TIME ON THE METER
 034242' 004737  017330'                      JSR     PC,CHKDNI              ;WAIT FOR THE MICROMONITOR
 034246' 103022                               BCC     20$                    ;OK
 034250' 012737  000776' 000310'              MOV     #$DNI,BITNAM
 034256' 012737  001275' 000312'              MOV     #$NSET,BITSTA
 034264' 012737  001340' 000314'              MOV     #$AFTER,PWHEN
 034272' 012737  001355' 000316'              MOV     #$GTCMD,PCOMND
 034300'                                      ERRHRD  087,DMAFRM,MSG1
 034300' 104456                                                                          TRAP    C$ERHRD
```

```
034302' 000127                                                                          .WORD   87
034304' 002431'                                                                         .WORD   DMAFRM
034306' 012750'                                                                         .WORD   MSG1
034310'                                 ESCAPE  TST
034310' 104410                                                                          TRAP    C$ESCAPE
034312' 000236                                                                          .WORD   L10135-.
034314' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
034320' 103006                          BCC     25$
034322'                                 ERRHRD  088,DMAFRM,RACMG7       ;DNI BIT DID NOT CLEAR!
034322' 104456                                                                          TRAP    C$ERHRD
034324' 000130                                                                          .WORD   88
034326' 002431'                                                                         .WORD   DMAFRM
034330' 012726'                                                                         .WORD   RACMG7
034332'                                 ESCAPE  TST
034332' 104410                                                                          TRAP    C$ESCAPE
034334' 000214                                                                          .WORD   L10135-.
034336'                         25$:
034336'                           ENDSEG
034336'                                                                       10000$:
034336' 104405                                                                          TRAP    C$ESEG
034340' 005037  000604'                 CLR     PCBB+0                  ;TELL T11 TO START LOAD AT BASE
                                                                        ;OF LINK MEMORY
034344'                         30$:
                                ;
                                ;WAIT FOR THE MICRMONITOR TO ENTER 'IN MONITOR' STATE, LOAD COMMAND FIELD
                                ;OF PCSRO WITH 5 TO START THE EXECUTION OF MICROTEST #5, WAIT FOR DNI
                                ;
034344'                           BGNSEG
034344' 104404                                                                          TRAP    C$BSEG
034346' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
034352' 103006                          BCC     40$                     ;OK
034354'                                 ERRHRD  089,DMAFRM,MSG46        ;PRINT ERROR
034354' 104456                                                                          TRAP    C$ERHRD
034356' 000131                                                                          .WORD   89
034360' 002431'                                                                         .WORD   DMAFRM
034362' 016700'                                                                         .WORD   MSG46
034364'                                 ESCAPE  TST                     ;LEAVE TEST
034364' 104410                                                                          TRAP    C$ESCAPE
034366' 000162                                                                          .WORD   L10135-.
034370' 012777  000005  143740  40$:    MOV     #5,@PCSRO               ;TELL T11 TO START MICROTEST #5
034376' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT A WHILE FOR DNI
034404' 004737  017330'                 JSR     PC,CHKDNI
034410' 103021                          BCC     50$
034412' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
034416' 103006                          BCC     45$                     ;NO, OK
034420'                                 ERRHRD  090,DMAFRM,MSG44        ;PRINT ERROR MESSAGE
034420' 104456                                                                          TRAP    C$ERHRD
034422' 000132                                                                          .WORD   90
034424' 002431'                                                                         .WORD   DMAFRM
034426' 016454'                                                                         .WORD   MSG44
034430'                                 ESCAPE  TST                     ;LEAVE TEST
034430' 104410                                                                          TRAP    C$ESCAPE
034432' 000116                                                                          .WORD   L10135-.
034434' 012702  000005          45$:    MOV     #5,R2                   ;WE WERE EXECUTING TEST 5
034440'                                 ERRHRD  091,DMAFRM,MSG12        ;TELL HIM IT HUNG
034440' 104456                                                                          TRAP    C$ERHRD
```

```
034442' 000133                                                                                  .WORD   91
034444' 002431'                                                                                 .WORD   DMAFRM
034446' 013500'                                                                                 .WORD   MSG12
034450'                                 ESCAPE  TST
034450' 104410                                                                                  TRAP    C$ESCAPE
034452' 000076                                                                                  .WORD   L10135-.
                                ;
                                ;OK GET THE 'FROM ADDRESS' THAT I PASSED TO THE MICROCODE AND MAKE IT AN
                                ;ACTUAL LINK MEMORY ADDRESS BY FORCING BIT 15 THIS MAPS IT TO THE 16-32K RANGE
                                ;ALSO FORCE BIT 2 BECAUSE THE FIRST 2 WORDS OF EACH BUFFER CAN NOT BE DMA'ED
                                ;FROM LINK MEMORY TO THE UNIBUS. CHECK THAT THE DATA TRANSFERRED IS GOOD
                                ;
034454' 013703  000604'         50$:    MOV     PCBB+0,R3               ;GET ORIGINAL 'FROM' ADDRESS
034460' 052703  100004                  BIS     #BIT15!BIT2,R3          ;MAKE IT ACTUAL LINK MEMORY ADDRESS
034464' 013704  000606'                 MOV     PCBB+2,R4               ;GET WHAT WAS READ FROM LINK MEMORY
034470' 020304                          CMP     R3,R4                   ;IS DATA CORRECT?
034472' 001404                          BEQ     55$                     ;YES
034474'                                 ERRHRD  092,DMAFRM,MSG14        ;NO
034474' 104456                                                                                  TRAP    C$ERHRD
034476' 000134                                                                                  .WORD   92
034500' 002431'                                                                                 .WORD   DMAFRM
034502' 013572'                                                                                 .WORD   MSG14
034504'                         55$:
034504'                          ENDSEG
034504'                                                                                 10001$:
034504' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;WRITE ONE TO CLEAR DNI
                                ;
034506' 004737  017374'                 JSR     PC,CLRDNI
034512' 103006                          BCC     57$
034514'                                 ERRHRD  093,DMAFRM,RACMG7       ;ERROR DNI DID NOT CLEAR!
034514' 104456                                                                                  TRAP    C$ERHRD
034516' 000135                                                                                  .WORD   93
034520' 002431'                                                                                 .WORD   DMAFRM
034522' 012726'                                                                                 .WORD   RACMG7
034524'                                 ESCAPE  TST
034524' 104410                                                                                  TRAP    C$ESCAPE
034526' 000022                                                                                  .WORD   L10135-.
034530' 022737  074000  000604' 57$:    CMP     #74000,PCBB+0           ;HAVE WE CHECKED ALL 1K CHUNKS?
034536' 001404                          BEQ     60$                     ;YES
034540' 062737  004000  000604'         ADD     #4000,PCBB+0            ;NO, CHECK NEXT 1K
034546' 000676                          BR      30$                     ;DO AGAIN
034550'                         60$:
034550'                         ENDTST
034550'                                                                                 L10135:
034550' 104401                                                                                  TRAP    C$ETST
```

```
                                .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 BUFFER ADDRESS
                                ;       3-START MICROCODE
                                ;       4-WAIT FOR DNI
                                ;       5-VERIFY ALL 1776 WORDS STARTING AT BUFFER ADDRESS BASE
                                ;
                                ;*************************************************************************

034552'                         BGNTST
034552'                                                                                  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.
                                ;
034552'                           BGNSEG
034552' 104404                                                                                  TRAP    C$BSEG
034554' 022737  000103  000326'         CMP     #'C,MICRO               ;HAS MICRO MODULE C BEEN LOADED?
034562' 001004                          BNE     5$                      ;NO
034564' 122777  000001  043546          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
034572' 001435                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
034574' 012737  000103  000326' 5$:     MOV     #'C,MICRO               ;NO, LOAD MICRO MODULE C
034602' 004737  020252'                 JSR     PC,LODMIC
034606' 103530                          BCS     70$                     ;ERROR
034610' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
034616' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR THE MICROMONITOR
034622' 103021                          BCC     20$                     ;OK
034624' 012737  000776' 000310'         MOV     #$DNI,BITNAM
034632' 012737  001275' 000312'         MOV     #$NSET,BITSTA
034640' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
034646' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
034654'                                 ERRHRD  094,DMABLK,MSG1
034654' 104456                                                                                  TRAP    C$ERHRD
034656' 000136                                                                                  .WORD   94
034660' 002501'                                                                                 .WORD   DMABLK
034662' 012750'                                                                                 .WORD   MSG1
034664' 000501                          BR      70$
034666' 004737  017374'         20$:    JSR     PC,CLRDNI               ;GO CLEAR THE DNI BIT
034672' 103005                          BCC     25$
034674'                                 ERRHRD  095,DMABLK,RACMG7       ;DNI DID NOT CLEAR!
034674' 104456                                                                                  TRAP    C$ERHRD
034676' 000137                                                                                  .WORD   95
```

```
034700' 002501'                                                                                 .WORD   DMABLK
034702' 012726'                                                                                 .WORD   RACMG7
034704' 000471                          BR      70$
034706'                         25$:
034706'                           ENDSEG
034706'                                                                                 10000$:
034706' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;TELL MICROCODE TO DMA TO A BUFFER IN FREE MEMORY LOCATED ABOVE THIS DIAGNOSTIC
                                ;BY LOADING PCBB+0 WITH THE ADDRESS TO DMA TO. WAIT FOR THE MICROMONITOR TO
                                ;ENTER THE 'IN MONITOR' STATE AND LOAD THE COMMAND FIELD OF PCSR0 WITH A 6
                                ;TO START THE EXECUTION OF MICROTEST #6. WAIT FOR DNI TO SET INDICATING IT IS
                                ;COMPLETE
                                ;
034710'                           BGNSEG
034710' 104404                                                                                  TRAP    C$BSEG
034712' 013737  000324' 000604'         MOV     FREMEM,PCBB+0           ;GET ADDRESS OF FREE MEMORY
034720' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
034724' 103006                          BCC     30$                     ;OK
034726'                                 ERRHRD  096,DMABLK,MSG46        ;PRINT ERROR
034726' 104456                                                                                  TRAP    C$ERHRD
034730' 000140                                                                                  .WORD   96
034732' 002501'                                                                                 .WORD   DMABLK
034734' 016700'                                                                                 .WORD   MSG46
034736'                                 ESCAPE  TST                     ;LEAVE TEST
034736' 104410                                                                                  TRAP    C$ESCAPE
034740' 000136                                                                                  .WORD   L10136-.
034742' 012777  000006  143366  30$:    MOV     #6,@PCSR0               ;START MICROTEST #6
034750' 012737  000473  000332'         MOV     #5*SECOND,METER         ;PUT SOME TIME ON THE METER
034756' 004737  017330'                 JSR     PC,CHKDNI               ;WAIT FOR MICROTEST FINISH
034762' 103020                          BCC     40$                     ;OK
034764' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
034770' 103006                          BCC     35$                     ;NO, OK
034772'                                 ERRHRD  097,DMABLK,MSG44        ;PRINT ERROR MESSAGE
034772' 104456                                                                                  TRAP    C$ERHRD
034774' 000141                                                                                  .WORD   97
034776' 002501'                                                                                 .WORD   DMABLK
035000' 016454'                                                                                 .WORD   MSG44
035002'                                 ESCAPE  TST                     ;LEAVE TEST
035002' 104410                                                                                  TRAP    C$ESCAPE
035004' 000072                                                                                  .WORD   L10136-.
035006' 012702  000006          35$:    MOV     #6,R2                   ;MICROTEST NEVER FINISHED!
035012'                                 ERRHRD  098,DMABLK,MSG12
035012' 104456                                                                                  TRAP    C$ERHRD
035014' 000142                                                                                  .WORD   98
035016' 002501'                                                                                 .WORD   DMABLK
035020' 013500'                                                                                 .WORD   MSG12
035022' 000422                          BR      70$
                                ;
                                ;OK NOW CHECK THE DATA TRANSFERRED FROM LINK MEMORY TO UNIBUS MEMORY.
                                ;THE DATA PATTERN IS ACTUALLY THE ADDRESS FROM WHICH THE DATA CAME FROM.
                                ;THE FIRST LOCATION WILL BE 4 BYTES FROM THE BEGINING OF LINK MEMORY BECAUSE
                                ;THE DMA FROM ADDRESS REGISTER DOES NOT USE BITS 0 AND 1.
                                ;
035024' 013702  000604'         40$:    MOV     PCBB+0,R2               ;GET ADDRESS OF BUFFER
035030' 012746  100004                  MOV     #100004,-(SP)           ;STARTING DATA PATTERN
```

```
 7929  035034' 021216                  50$:    CMP     (R2),(SP)               ;IS DATA PATTERN CORRECT?
 7930  035036' 001405                          BEQ     60$                     ;YES
 7931  035040' 010601                          MOV     SP,R1                   ;NO GET GOOD DATA
 7932  035042'                                 ERRHRD  099,DMABLK,MSG15        ;PRINT ERROR MESSAGE
 7933  035042' 104456                                                                                  TRAP    C$ERHRD
 7934  035044' 000143                                                                                  .WORD   99
 7935  035046' 002501'                                                                                 .WORD   DMABLK
 7936  035050' 013620'                                                                                 .WORD   MSG15
 7937  035052' 021627  103776          60$:    CMP     (SP),#103776            ;DONE ALL DATA?
 7938  035056' 002004                          BGE     70$                     ;YES
 7939  035060' 0627:6  000002                  ADD     #2,(SP)                 ;NEXT GOOD DATA PATTERN
 7940  035064' 005722                          TST     (R2)+                   ;NEXT BUFFER ADDRESS
 7941  035066' 000762                          BR      50$                     ;CONTINUE CHECKING
 7942  035070' 004737  020104'         70$:    JSR     PC,REUNA                ;RESTORE OPERATIONAL MICROCODE
 7943  035074'                           ENDSEG
 7944  035074'                                                                                 10001$:
 7945  035074' 104405                                                                                  TRAP    C$ESEG
 7946  035076'                         ENDTST
 7947  035076'                                                                                 L10136:
 7948  035076' 104401                                                                                  TRAP    C$ETST
```

```
                                        .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 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 BF 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 #1
                                        ;       4-VERIFY 'DNI' BIT IN PCSRO AFTER A RESONABLE PERIOD OF TIME
                                        ;       5-WRITE ONE TO CLEAR DNI
                                        ;
                                        ;***********************************************************************

035100'                                 BGNTST
035100'                                                                                 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.
                                        ;
035100'                                   BGNSEG
035100' 104404                                                                                  TRAP    C$BSEG
035102' 022737  000104  000326'                 CMP     #'D,MICRO                   ;HAS MICROCODE MODULE 'D' BEEN LOADED
035110' 001004                                  BNE     5$                          ;NO
035112' 122777  000001  143220                  CMPB    #INMON,@PCSR1               ;YES, IS THE MICROMONITOR ACTIVE?
035120' 001440                                  BEQ     20$                         ;YES SKIP LOADING THE MICROMODULE
035122' 012737  000104  000326' 5$:             MOV     #'D,MICRO                   ;GO LOAD MICRO MODULE 'D'
035130' 004737  020252'                         JSR     PC,LODMIC
035134' 103002                                  BCC     10$                         ;OK
035136'                                         ESCAPE  TST
035136' 104410                                                                                  TRAP    C$ESCAPE
035140' 000246                                                                                  .WORD   L10137-.
035142' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER             ;WAIT FOR THE MICROMONITOR
035150' 004737  017330'                         JSR     PC,CHKDNI
035154' 103022                                  BCC     20$                         ;OK
035156' 012737  000776' 000310'                 MOV     #$DNI,BITNAM
035164' 012737  001275' 000312'                 MOV     #$NSET,BITSTA
035172' 012737  001340' 000314'                 MOV     #$AFTER,PWHEN
035200' 012737  001355' 000316'                 MOV     #$GTCMD,PCOMND
035206'                                         ERRHRD  100,TRNDON,MSG1
035206' 104456                                                                                  TRAP    C$ERHRD
035210' 000144                                                                                  .WORD   100
035212' 002540'                                                                                 .WORD   TRNDON
035214' 012750'                                                                                 .WORD   MSG1
```

```
035216'                                     ESCAPE  TST
035216' 104410                                                                                TRAP    C$ESCAPE
035220' 000166                                                                                .WORD   L10137-.
035222' 004737  017374'             20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
035226' 103006                              BCC     25$
035230'                                     ERRHRD  101,TRNDON,RACMG7       ;DNI DID NOT CLEAR!
035230' 104456                                                                                TRAP    C$ERHRD
035232' 000145                                                                                .WORD   101
035234' 002540'                                                                               .WORD   TRNDON
035236' 012726'                                                                               .WORD   RACMG7
035240'                                     ESCAPE TST
035240' 104410                                                                                TRAP    C$ESCAPE
035242' 000144                                                                                .WORD   L10137-.
035244'                             25$:
035244'                               ENDSEG
035244'                                                                            10000$:
035244' 104405                                                                                TRAP    C$ESEG
                                    ;
                                    ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. LOAD THE COMMAND
                                    ;FIELD BITS OF PCSRO WITH A 1 TO START THE EXECUTION OF MICROTEST #1.
                                    ;WAIT ABOUT 1 SECOND FOR IT TO FINISH. IF NO 'DNI' SET PRINT ERROR.
                                    ;
035246'                               BGNSEG
035246' 104404                                                                                TRAP    C$BSEG
035250' 004737  017776'                     JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
035254' 103006                              BCC     30$                     ;OK
035256'                                     ERRHRD  102,TRNDON,MSG46        ;PRINT ERROR
035256' 104456                                                                                TRAP    C$ERHRD
035260' 000146                                                                                .WORD   102
035262' 002540'                                                                               .WORD   TRNDON
035264' 016700'                                                                               .WORD   MSG46
035266'                                     ESCAPE  TST                     ;LEAVE TEST
035266' 104410                                                                                TRAP    C$ESCAPE
035270' 000116                                                                                .WORD   L10137-.
035272' 012777  000001  143036      30$:    MOV     #1,@PCSRO               ;TELL T11 TO EXECUTE FIRST MICROTEST
035300' 012737  000077  000332'             MOV     #1*SECOND,METER         ;WAIT FOR DNI
035306' 004737  017330'                     JSR     PC,CHKDNI
035312' 103025                              BCC     40$                     ;OK-'DNI' SET
                                                                            ;ERROR 'DNI' DID NOT SET! NO TRANSMIT
                                                                            ;INTERRUPT!
035314'                                     ERRHRD  103,TRNDON,MSG36        ;PRINT ERROR MESSAGE
035314' 104456                                                                                TRAP    C$ERHRD
035316' 000147                                                                                .WORD   103
035320' 002540'                                                                               .WORD   TRNDON
035322' 016002'                                                                               .WORD   MSG36
035324' 004737  020050'                     JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
035330' 103006                              BCC     35$                     ;NO, OK
035332'                                     ERRHRD  103,TRNDON,MSG44        ;PRINT ERROR MESSAGE
035332' 104456                                                                                TRAP    C$ERHRD
035334' 000147                                                                                .WORD   103
035336' 002540'                                                                               .WORD   TRNDON
035340' 016454'                                                                               .WORD   MSG44
035342'                                     ESCAPF  TST                     ;LEAVE TEST
035342' 104410                                                                                TRAP    C$ESCAPE
035344' 000042                                                                                .WORD   L10137-.
035346' 012702  000001              35$:    MOV     #1,R2                   ;MICROTEST #1 IS HUNG
```

```
035352'                                        ERRHRD  104,TRNDON,MSG12        ;PRINT MESSAGE ABOUT HUNG MICROTEST
035352' 104456                                                                                         TRAP    C$ERHRD
035354' 000150                                                                                         .WORD   104
035356' 002540'                                                                                        .WORD   TRNDON
035360' 013500'                                                                                        .WORD   MSG12
035362'                                        ESCAPE  TST
035362' 104410                                                                                         TRAP    C$ESCAPE
035364' 000022                                                                                         .WORD   L10137-.
                                ;
                                ;WRITE 1 TO CLEAR DNI BIT
                                ;
035366' 004737  017374'         40$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
035372' 103004                          BCC     55$
035374'                                 ERRHRD  105,TRNDON,RACMG7       ;ERROR DNI DID NOT CLEAR!
035374' 104456                                                                                         TRAP    C$ERHRD
035376' 000151                                                                                         .WORD   105
035400' 002540'                                                                                        .WORD   TRNDON
035402' 012726'                                                                                        .WORD   RACMG7
035404'                         55$:
035404'                           ENDSEG
035404'                                                                                        10001$:
035404' 104405                                                                                         TRAP    C$ESEG
035406'                         ENDTST
035406'                                                                                        L10137:
035406' 104401                                                                                         TRAP    C$ETST
```

```
                                 .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 RESONABLE PERIOD OF TIME
                                 ;       5-WRITE ONE TO CLEAR DNI
                                 ;
                                 ;********************************************************************

035410'                          BGNTST
035410'                                                                              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.
                                 ;
035410'                            BGNSEG
035410' 104404                                                                                   TRAP    C$BSEG
035412' 022737  000104  000326'         CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
035420' 001004                          BNE     5$                      ;NO
035422' 122777  000001  142710          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
035430' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
035432' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
035440' 004737  020252'                 JSR     PC,LODMIC
035444' 103002                          BCC     10$                     ;OK
035446'                                 ESCAPE  TST
035446' 104410                                                                                   TRAP    C$ESCAPE
035450' 000246                                                                                   .WORD   L10140-.
035452' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
035460' 004737  017330'                 JSR     PC,CHKDNI
035464' 103022                          BCC     20$                     ;OK
035466' 012737  000776' 000310'         MOV     #$DNI,BITNAM
035474' 012737  001275' 000312'         MOV     #$NSET,BITSTA
035502' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
035510' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
035516'                                 ERRHRD  106,RCVDON,MSG1
035516' 104456                                                                                   TRAP    C$ERHRD
035520' 000152                                                                                   .WORD   106
035522' 002572'                                                                                  .WORD   RCVDON
```

```
035524' 012750'                                                                       .WORD   MSG1
035526'                                    ESCAPE  TST
035526' 104410                                                                        TRAP    C$ESCAPE
035530' 000166                                                                        .WORD   L10140-.
035532' 004737  017374'         20$:       JSR     PC,CLRDNI             ;CLEAR DNI BIT
035536' 103006                             BCC     25$
035540'                                    ERRHRD  107,RCVDON,RACMG7     ;DNI DID NOT CLEAR!
035540' 104456                                                                        TRAP    C$ERHRD
035542' 000153                                                                        .WORD   107
035544' 002572'                                                                       .WORD   RCVDON
035546' 012726'                                                                       .WORD   RACMG7
035550'                                    ESCAPE TST
035550' 104410                                                                        TRAP    C$ESCAPE
035552' 000144                                                                        .WORD   L10140-.
035554'                         25$:
035554'                           ENDSEG
035554'                                                                          10000$:
035554' 104405                                                                        TRAP    C$ESEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. LOAD THE COMMAND
                                ;FIELD BITS OF PCSRO WITH A 2 TO START THE EXECUTION OF MICROTEST #2.
                                ;WAIT ABOUT 1 SECOND FOR IT TO FINISH. IF NO 'DNI' SET PRINT ERROR.
                                ;
035556'                           BGNSEG
035556' 104404                                                                        TRAP    C$BSEG
035560' 004737  017776'                    JSR     PC,CHKMON             ;WAIT FOR MICROMONITOR
035564' 103006                             BCC     30$                   ;OK
035566'                                    ERRHRD  108,RCVDON,MSG46      ;PRINT ERROR
035566' 104456                                                                        TRAP    C$ERHRD
035570' 000154                                                                        .WORD   108
035572' 002572'                                                                       .WORD   RCVDON
035574' 016700'                                                                       .WORD   MSG46
035576'                                    ESCAPE  TST                   ;LEAVE TEST
035576' 104410                                                                        TRAP    C$ESCAPE
035600' 000116                                                                        .WORD   L10140-.
035602' 012777  000002  142526  30$:       MOV     #2,@PCSRO             ;TELL T11 TO EXECUTE MICROTEST #2
035610' 012737  000077  000332'            MOV     #1*SECOND,METER       ;WAIT FOR DNI
035616' 004737  017330'                    JSR     PC,CHKDNI
035622' 103025                             BCC     40$                   ;OK DNI SET
                                                                         ;ERROR-DNI NOT SET, NO RECEIVER
                                                                         ;INTERRUPT
035624'                                    ERRHRD  109,RCVDON,MSG37      ;PRINT ERROR MESSAGE
035624' 104456                                                                        TRAP    C$ERHRD
035626' 000155                                                                        .WORD   109
035630' 002572'                                                                       .WORD   RCVDON
035632' 016024'                                                                       .WORD   MSG37
035634' 004737  020050'                    JSR     PC,CHKINT             ;SEE IF ANY ERROR INTERRUPTS OCCURRED
035640' 103006                             BCC     35$                   ;NO, OK
035642'                                    ERRHRD  110,RCVDON,MSG44      ;PRINT ERROR MESSAGE
035642' 104456                                                                        TRAP    C$ERHRD
035644' 000156                                                                        .WORD   110
035646' 002572'                                                                       .WORD   RCVDON
035650' 016454'                                                                       .WORD   MSG44
035652'                                    ESCAPE  TST
035652' 104410                                                                        TRAP    C$ESCAPE
035654' 000042                                                                        .WORD   L10140-.
```

```
035656' 012702  000002        35$:    MOV     #2,R2                   ;MICROTEST #2 IS HUNG
035662'                               ERRHRD  111,RCVDON,MSG12        ;PRINT ERROR MESSAGE
035662' 104456                                                                                  TRAP    C$ERHRD
035664' 000157                                                                                  .WORD   111
035666' 002572'                                                                                 .WORD   RCVDON
035670' 013500'                                                                                 .WORD   MSG12
035672'                               ESCAPE  TST
035672' 104410                                                                                  TRAP    C$ESCAPE
035674' 000022                                                                                  .WORD   L10140-.
                              ;
                              ;WRITE 1 TO CLEAR DNI BIT
                              ;
035676' 004737  017374'       40$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
035702' 103004                        BCC     55$
035704'                               ERRHRD  112,RCVDON,RACMG7       ;ERROR DNI DID NOT CLEAR!
035704' 104456                                                                                  TRAP    C$ERHRD
035706' 000160                                                                                  .WORD   112
035710' 002572'                                                                                 .WORD   RCVDON
035712' 012726'                                                                                 .WORD   RACMG7
035714'                       55$:
035714'                         ENDSEG
035714'                                                                                 10001$:
035714' 104405                                                                                  TRAP    C$ESEG
035716'                       ENDTST
035716'                                                                                 L10140:
035716' 104401                                                                                  TRAP    C$ETST
```

```

                                .SBTTL  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. 1111111110000000011...) 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 SEQUENCE:
                                ;       1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO
                                ;       2-VERIFY MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;       3-CLEAR OUT THE PCBB+,+2,+4,+6
                                ;       4-SELECT MICROTEST #3
                                ;       5-VERIFY 'DNI' BIT SET IN PCSRO
                                ;       6-CHECK FOR AN ERROR CONDITION IN PCSR1
                                ;       7-WRITE ONE TO CLEAR THE DNI BIT
                                ;
                                ;*************************************************************************

035720'                         BGNTST
035720'                                                                           T23::

                                ;
                                ;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.
                                ;
035720'                           BGNSEG
035720' 104404                                                                          TRAP    C$BSEG
035722' 022737  000104  000326'        CMP     #'D,MICRO              ;HAS MICROCODE MODULE 'D' BEEN LOADED
035730' 001004                         BNE     5$                     ;NO
035732' 122777  000001  142400         CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
035740' 001440                         BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
035742' 012737  000104  000326' 5$:    MOV     #'D,MICRO              ;GO LOAD MICRO MODULE 'D'
035750' 004737  020252'                JSR     PC,LODMIC
035754' 103002                         BCC     10$                    ;OK
035756'                                ESCAPE  TST
035756' 104410                                                                          TRAP    C$ESCAPE
035760' 000316                                                                          .WORD   L10141-.
035762' 012737  000176  000332' 10$:   MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
```

```
035770' 004737  017330'                  JSR     PC,CHKDNI
035774' 103022                           BCC     20$                    ;OK
035776' 012737  000776' 000310'          MOV     #$DNI,BITNAM           ;ERROR 'DNI' BIT NEVER SET
036004' 012737  001275' 000312'          MOV     #$NSET,BITSTA
036012' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
036020' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
036026'                                  ERRHRD  113,DBFRAM,MSG1
036026' 104456                                                                                  TRAP    C$ERHRD
036030' 000161                                                                                  .WORD   113
036032' 002624'                                                                                 .WORD   DBFRAM
036034' 012750'                                                                                 .WORD   MSG1
036036'                                  ESCAPE  TST
036036' 104410                                                                                  TRAP    C$ESCAPE
036040' 000236                                                                                  .WORD   L10141-.
036042' 004737  017374'          20$:    JSR     PC,CLRDNI              ;CLEAR DNI BIT
036046' 103006                           BCC     25$
036050'                                  ERRHRD  114,DBFRAM,RACMG7      ;DNI DID NOT CLEAR.
036050' 104456                                                                                  TRAP    C$ERHRD
036052' 000162                                                                                  .WORD   114
036054' 002624'                                                                                 .WORD   DBFRAM
036056' 012726'                                                                                 .WORD   RACMG7
036060'                                  ESCAPE TST
036060' 104410                                                                                  TRAP    C$ESCAPE
036062' 000214                                                                                  .WORD   L10141-.
036064'                          25$:
036064'                            ENDSEG
036064'                                                                                 10000$:
036064' 104405                                                                                  TRAP    C$ESEG

                                 ;
                                 ;WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE. CLEAR THE PCBB. LOAD THE
                                 ;COMMAND FIELD OF PCSRO WITH A 3 TO START THE EXECUTION OF MICROTEST #3.
                                 ;WAIT FOR THE 'DNI' BIT. IF NO 'DNI' OR IF ANY EXTRANEOUS INTERRUPTS HAPPEN
                                 ;PRINT ERROR
                                 ;

036066'                            BGNSEG
036066' 104404                                                                                  TRAP    C$BSEG
036070' 004737  017776'                  JSR     PC,CHKMON              ;WAIT FOR MICROMONITOR
036074' 103006                           BCC     30$                    ;OK
036076'                                  ERRHRD  115,DBFRAM,MSG46       ;PRINT ERROR
036076' 104456                                                                                  TRAP    C$ERHRD
036100' 000163                                                                                  .WORD   115
036102' 002624'                                                                                 .WORD   DBFRAM
036104' 016700'                                                                                 .WORD   MSG46
036106'                                  ESCAPE  TST                    ;LEAVE TEST
036106' 104410                                                                                  TRAP    C$ESCAPE
036110' 000166                                                                                  .WORD   L10141-.
036112' 005037  000604'          30$:    CLR     PCBB                   ;THIS IS WHERE MICROCODE WILL PUT...
                                                                        ;RECEIVE BUFFER STATUS WORD
036116' 005037  000606'                  CLR     PCBB+2                 ;HERE IS WHERE THE GOOD DATA GOES
036122' 005037  000610'                  CLR     PCBB+4                 ;HERE IS WHERE THE BAD DATA GOES
036126' 005037  000612'                  CLR     PCBB+6                 ;HERE IS WHERE THE BUFFER OFFSET GOES
036132' 012777  000003  142176           MOV     #3,@PCSRO              ;TELL T11 TO EXECUTE MICROTEST #3
036140' 012737  000176  000332'          MOV     #2*SECOND,METER        ;WAIT FOR DNI
036146' 004737  017330'                  JSR     PC,CHKDNI
036152' 103021                           BCC     40$
```

```
036154' 004737  020050'                         JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
036160' 103006                                  BCC     35$                     ;NO, OK
036162'                                         ERRHRD  116,DBFRAM,MSG44        ;PRINT ERROR MESSAGE
036162' 104456                                                                                          TRAP    C$ERHRD
036164' 000164                                                                                          .WORD   116
036166' 002624'                                                                                         .WORD   DBFRAM
036170' 016454'                                                                                         .WORD   MSG44
036172'                                         ESCAPE  TST
036172' 104410                                                                                          TRAP    C$ESCAPE
036174' 000102                                                                                          .WORD   L10141-.
036176' 012702  000003                  35$:    MOV     #3,R2                   ;MICROTEST #3 IS HUNG
036202'                                         ERRHRD  117,DBFRAM,MSG12
036202' 104456                                                                                          TRAP    C$ERHRD
036204' 000165                                                                                          .WORD   117
036206' 002624'                                                                                         .WORD   DBFRAM
036210' 013500'                                                                                         .WORD   MSG12
036212'                                         ESCAPE  TST
036212' 104410                                                                                          TRAP    C$ESCAPE
036214' 000062                                                                                          .WORD   L10141-.

                                        ;
                                        ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF THE T-11 FOUND A BOGUS BYTE IN THE
                                        ;RECEIVED DATA. IF SO, GET THE INFORMATION FROM THE PCBB AND PRINT ERROR
                                        ;
036216' 122777  000003  142114  40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
036224' 001014                                  BNE     50$                     ;NO
036226' 013701  000612'                         MOV     PCBB+6,R1               ;GET BUFFER OFFSET FOR ERROR REPORT
036232' 013702  000606'                         MOV     PCBB+2,R2               ;THIS IS THE GOOD DATA FOR ERROR REPORT
036236' 013703  000610'                         MOV     PCBB+4,R3               ;THIS IS THE BAD DATA FOR ERROR REPORT
036242' 013704  000604'                         MOV     PCBB,R4                 ;THIS IS THE RECEIVER STATUS WORD
036246'                                         ERRHRD  118,DBFRAM,MSG17        ;PRINT ERROR MESSAGE
036246' 104456                                                                                          TRAP    C$ERHRD
036250' 000166                                                                                          .WORD   118
036252' 002624'                                                                                         .WORD   DBFRAM
036254' 013742'                                                                                         .WORD   MSG17
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI' BIT
                                        ;
036256' 004737  017374'                 50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
036262' 103004                                  BCC     55$
036264'                                         ERRHRD  119,DBFRAM,RACMG7       ;ERROR DNI DID NOT CLEAR!
036264' 104456                                                                                          TRAP    C$ERHRD
036266' 000167                                                                                          .WORD   119
036270' 002624'                                                                                         .WORD   DBFRAM
036272' 012726'                                                                                         .WORD   RACMG7
036274'                                 55$:
036274'                                   ENDSEG
036274'                                                                                         10001$:
036274' 104405                                                                                          TRAP    C$ESEG
036276'                                 ENDTST
036276'                                                                                         L10141:
036276' 104401                                                                                          TRAP    C$ETST
```

```
                                .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. 11111111111111110000000000000000011...) 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 SEQUENCE:
                                ;       1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO
                                ;       2-VERIFY MICROMONITOR IS IN THE 'INMON' STATE
                                ;       3-CLEAR OUT THE PCBB+0,+2,+4,+6
                                ;       4-SELECT MICROTEST #4
                                ;       5-VERIFY 'DNI' BIT SET IN PCSPO
                                ;       6-CHECK FOR AN ERROR CONDITION IN PCSR1
                                ;       7-WRITE ONE TO CLEAR THE DNI BIT
                                ;
                                ;*************************************************************************

036300'                         BGNTST
036300'                                                                          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.
                                ;
036300'                           BGNSEG
036300' 104404                                                                           TRAP    C$BSEG
036302' 022737  000104  000326'         CMP     #'D,MICRO              ;HAS MICROCODE MODULE 'D' BEEN LOADED
036310' 001004                          BNE     5$                     ;NO
036312' 122777  000001  142020          CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
036320' 001440                          BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
036322' 012737  000104  000326' 5$:     MOV     #'D,MICRO              ;GO LOAD MICRO MODULE 'D'
036330' 004737  020252'                 JSR     PC,LODMIC
036334' 103002                          BCC     10$                    ;OK
036336'                                 ESCAPE  TST
036336' 104410                                                                           TRAP    C$ESCAPE
036340' 000316                                                                           .WORD   L10142-.
036342' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
036350' 004737  017330'                 JSR     PC,CHKDNI
```

```
036354' 103022                          BCC     20$                     ;OK
036356' 012737  000776' 000310'         MOV     #$DNI,BITNAM
036364' 012737  001275' 000312'         MOV     #$NSET,BITSTA
036372' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
036400' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
036406'                                 ERRHRD  120,DWFRAM,MSG1
036406' 104456                                                                  TRAP    C$ERHRD
036410' 000170                                                                  .WORD   120
036412' 002662'                                                                 .WORD   DWFRAM
036414' 012750'                                                                 .WORD   MSG1
036416'                                 ESCAPE  TST
036416' 104410                                                                  TRAP    C$ESCAPE
036420' 000236                                                                  .WORD   L10142-.
036422' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
036426' 103006                          BCC     25$
036430'                                 ERRHRD  121,DWFRAM,RACMG7       ;DNI DID NOT CLEAR!
036430' 104456                                                                  TRAP    C$ERHRD
036432' 000171                                                                  .WORD   121
036434' 002662'                                                                 .WORD   DWFRAM
036436' 012726'                                                                 .WORD   RACMG7
036440'                                 ESCAPE TST
036440' 104410                                                                  TRAP    C$ESCAPE
036442' 000214                                                                  .WORD   L10142-.
036444'                         25$:
036444'                           ENDSEG
036444'                                                                 10000$:
036444' 104405                                                                  TRAP    C$ESEG

                                ;
                                ;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 EXTRANEOUS INTERRUPTS HAPPEN
                                ;PRINT ERROR
                                ;
036446'                           BGNSEG
036446' 104404                                                                  TRAP    C$BSEG
036450' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
036454' 103006                          BCC     30$                     ;OK
036456'                                 ERRHRD  122,DWFRAM,MSG46        ;PRINT ERROR
036456' 104456                                                                  TRAP    C$ERHRD
036460' 000172                                                                  .WORD   122
036462' 002662'                                                                 .WORD   DWFRAM
036464' 016700'                                                                 .WORD   MSG46
036466'                                 ESCAPE  TST                     ;LEAVE TEST
036466' 104410                                                                  TRAP    C$ESCAPE
036470' 000166                                                                  .WORD   L10142-.
036472' 005037  000604'         30$:    CLR     PCBB                    ;THIS IS WHERE MICROCODE WILL PUT...
                                                                        ;RECEIVE BUFFER STATUS WORD
036476' 005037  000606'                 CLR     PCBB+2                  ;HERE IS WHERE THE GOOD DATA GOES
036502' 005037  000610'                 CLR     PCBB+4                  ;HERE IS WHERE THE BAD DATA GOES
036506' 005037  000612'                 CLR     PCBB+6                  ;HERE IS WHERE THE BUFFER OFFSET GOES
036512' 012777  000004  141615          MOV     #4,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #4
036520' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
036526' 004737  017330'                 JSR     FC,CHKDNI
036532' 103021                          BCC     40$
036534' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
036540' 103006                          BCC     35$                     ;NO, OK
```

```
036542'                                  ERRHRD  123,DWFRAM,MSG44        ;PRINT ERROR MESSAGE
036542' 104456                                                                                 TRAP    C$ERHRD
036544' 000173                                                                                 .WORD   123
036546' 002662'                                                                                .WORD   DWFRAM
036550' 016454'                                                                                .WORD   MSG44
036552'                                  ESCAPE  TST
036552' 104410                                                                                 TRAP    C$ESCAPE
036554' 000102                                                                                 .WORD   L10142-.
036556' 012702  000004          35$:     MOV     #4,R2                   ;MICROTEST #4 IS HUNG
036562'                                  ERRHRD  124,DWFRAM,MSG12
036562' 104456                                                                                 TRAP    C$ERHRD
036564' 000174                                                                                 .WORD   124
036566' 002662'                                                                                .WORD   DWFRAM
036570' 013500'                                                                                .WORD   MSG12
036572'                                  ESCAPE  TST
036572' 104410                                                                                 TRAP    C$ESCAPE
036574' 000062                                                                                 .WORD   L10142-.
                                ;
                                ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF THE T-11 DETECTED A BOGUS WORD
                                ;IN THE RECEIVER BUFFER. IF SO, PRINT ERROR.
                                ;
036576' 122777  000003  141534  40$:     CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
036604' 001014                           BNE     50$                     ;NO
036606' 013701  000612'                  MOV     PCBB+6,R1               ;GET BUFFER OFFSET FOR ERROR REPORT
036612' 013702  000606'                  MOV     PCBB+2,R2               ;THIS IS THE GOOD DATA FOR ERROR REPORT
036616' 013703  000610'                  MOV     PCBB+4,R3               ;THIS IS THE BAD DATA FOR ERROR REPORT
036622' 013704  000604'                  MOV     PCBB,R4                 ;THIS IS THE RECEIVER STATUS WORD
036626'                                  ERRHRD  125,DWFRAM,MSG17        ;PRINT ERROR MESSAGE
036626' 104456                                                                                 TRAP    C$ERHRD
036630' 000175                                                                                 .WORD   125
036632' 002662'                                                                                .WORD   DWFRAM
036634' 013742'                                                                                .WORD   MSG17
                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
036636' 004737  017374'         50$:     JSR     PC,CLRDNI               ;CLEAR DNI BIT
036642' 103004                           BCC     55$
036644'                                  ERRHRD  126,DWFRAM,RACMG7       ;ERROR DNI DID NOT CLEAR!
036644' 104456                                                                                 TRAP    C$ERHRD
036646' 000176                                                                                 .WORD   126
036650' 002662'                                                                                .WORD   DWFRAM
036652' 012726'                                                                                .WORD   RACMG7
036654'                         55$:
036654'                           ENDSEG
036654'                                                                                10001$:
036654' 104405                                                                                 TRAP    C$ESEG
036656'                         ENDTST
036656'                                                                                L10142:
036656' 104401                                                                                 TRAP    C$ETST
```

```
                                          .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 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
                                          ;
                                          ;***********************************************************************

      036660'                             BGNTST
      036660'                                                                         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.
                                          ;
      036660'                               BGNSEG
      036660' 104404                                                                                  TRAP    C$BSEG
      036662' 022737  000104  000326'             CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
      036670' 001004                              BNE     5$                      ;NO
      036672' 122777  000001  141440              CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
      036700' 001440                              BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
      036702' 012737  000104  000326' 5$:         MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
      036710' 004737  020252'                     JSR     PC,LODMIC
      036714' 103002                              BCC     10$                     ;OK
      036716'                                     ESCAPE  TST
      036716' 104410                                                                                  TRAP    C$ESCAPE
      036720' 000332                                                                                  .WORD   L10143-.
      036722' 012737  000176  000332' 10$:        MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
      036730' 004737  017330'                     JSR     PC,CHKDNI
      036734' 103022                              BCC     20$                     ;OK
      036736' 012737  000776' 000310'             MOV     #$DNI,BITNAM
      036744' 012737  001275' 000312'             MOV     #$NSET,BITSTA
      036752' 012737  001340' 000314'             MOV     #$AFTER,PWHEN
      036760' 012737  001355' 000316'             MOV     #$GTCMD,PCOMND
      036766'                                     ERRHRD  127,DPPAT,MSG1
      036766' 104456                                                                                  TRAP    C$ERHRD
      036770' 000177                                                                                  .WORD   127
```

```
036772' 002720'                                                                                     .WORD   DPPAT
036774' 012750'                                                                                     .WORD   MSG1
036776'                                   ESCAPE  TST
036776' 104410                                                                                      TRAP    C$ESCAPE
037000' 000252                                                                                      .WORD   L10143-.
037002' 004737   017374'         20$:     JSR      PC,CLRDNI                ;CLEAR DNI BIT
037006' 103006                            BCC      25$
037010'                                   ERRHRD   128,DPPAT,RACMG7         ;DNI DID NOT CLEAR!
037010' 104456                                                                                      TRAP    C$ERHRD
037012' 000200                                                                                      .WORD   128
037014' 002720'                                                                                     .WORD   DPPAT
037016' 012726'                                                                                     .WORD   RACMG7
037020'                                   ESCAPE TST
037020' 104410                                                                                      TRAP    C$ESCAPE
037022' 000230                                                                                      .WORD   L10143-.
037024'                          25$:
037024'                            ENDSEG
037024'                                                                                    10000$:
037024' 104405                                                                                      TRAP    C$ESEG
                                 ;
                                 ;CLEAR OUT THE PCBB+2,+4,+6 AND LOAD PCBB+0 WITH A DATA PATTERN SELECTED
                                 ;FROM A LIST OF STANDARD DATA PATTERNS
                                 ;
037026' 005037   000606'                  CLR      PCBB+2                   ;HERE IS WHERE MICROCODE WILL PUT...
                                                                            ;STATUS WORD
037032' 005037   000610'                  CLR      PCBB+4                   ;HERE IS WHERE BAD DATA GOES
037036' 005037   000612'                  CLR      PCBB+6                   ;HERE IS WHERE BUFFER OFFSET GOES
037042' 012701   000516'                  MOV      #PATERN,R1               ;GET ADDRESS OF DATA PATTERN TABLE
037046' 012705   000005                   MOV      #5,R5                    ;# OF DATA PATTERNS
037052' 012137   000604'         27$:     MOV      (R1)+,PCBB               ;LOAD PCBB WITH A DATA PATTERN
037056'                            BGNSEG
037056' 104404                                                                                      TRAP    C$BSEG
                                 ;
                                 ;WAIT FOR THE T-11 TO ENTER THE 'INMON' STATE AT WHICH TIME SELECT MICROTEST #5
                                 ;BY LOADING PCSRO COMMAND FIELD BITS WITH A 5. WAIT FOR 'DNI'.
                                 ;
037060' 004737   017776'                  JSR      PC,CHKMON                ;WAIT FOR MICROMONITOR
037064' 103006                            BCC      30$                      ;OK
037066'                                   ERRHRD   129,DPPAT,MSG46          ;PRINT ERROR
037066' 104456                                                                                      TRAP    C$ERHRD
037070' 000201                                                                                      .WORD   129
037072' 002720'                                                                                     .WORD   DPPAT
037074' 016700'                                                                                     .WORD   MSG46
037076'                                   ESCAPE  TST                      ;LEAVE TEST
037076' 104410                                                                                      TRAP    C$ESCAPE
037100' 000152                                                                                      .WORD   L10143-.
037102' 012777   000005  141226  30$:     MOV      #5,@PCSRO                ;TELL T11 TO EXECUTE MICROTEST #5
037110' 012737   000176  000332'          MOV      #2*SECOND,METER          ;WAIT FOR DNI
037116' 004737   017330'                  JSR      PC,CHKDNI
037122' 103021                            BCC      40$
037124' 004737   020030'                  JSR      PC,CHKINT                ;SEE IF ANY ERROR INTERRUPTS OCCURRED
037130' 103006                            BCC      35$                      ;NO, OK
037132'                                   ERRHRD   130,DPPAT,MSG44          ;PRINT ERROR MESSAGE
037132' 104456                                                                                      TRAP    C$ERHRD
037134' 000202                                                                                      .WORD   130
037136' 002720'                                                                                     .WORD   DPPAT
```

```
037140'   016454'                                                                                                       .WORD     MSG44
037142'                                           ESCAPE    TST
037142'   104410                                                                                                        TRAP      C$ESCAPE
037144'   000106                                                                                                        .WORD     L10143-.
037146'   012702    000005          35$:      MOV       #5,R2                         ;MICROTEST #5 IS HUNG
037152'                                           ERRHRD    131,DPPAT,MSG12
037152'   104456                                                                                                        TRAP      C$ERHRD
037154'   000203                                                                                                        .WORD     131
037156'   002720'                                                                                                       .WORD     DPPAT
037160'   013500'                                                                                                       .WORD     MSG12
037162'                                           ESCAPE    TST
037162'   104410                                                                                                        TRAP      C$ESCAPE
037164'   000066                                                                                                        .WORD     L10143-.
                                        ;
                                        ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF AN ERROR HAPPENED, IF SO, PRINT
                                        ;PERTINENT INFORMATION
                                        ;
037166'   122777    000003    141144    40$:      CMPB      #INERR,@PCSR1                 ;DID AN ERROR OCCUR?
037174'   001014                                  BNE       50$                           ;NO
037176'   013701    000612'                       MOV       PCBB+6,R1                     ;GET BUFFER OFFSET FOR ERROR REPORT
037202'   013702    000604'                       MOV       PCBB,R2                       ;THIS IS THE GOOD DATA FOR ERROR REPORT
037206'   013703    000610'                       MOV       PCBB+4,R3                     ;THIS IS THE BAD DATA FOR ERROR REPORT
037212'   013704    000606'                       MOV       PCBB+2,R4                     ;THIS IS THE RECEIVER STATUS WORD
037216'                                           ERRHRD    132,DPPAT,MSG17 ;PRINT ERROR MESSAGE
037216'   104456                                                                                                        TRAP      C$ERHRD
037220'   000204                                                                                                        .WORD     132
037222'   002720'                                                                                                       .WORD     DPPAT
037224'   013742'                                                                                                       .WORD     MSG17
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI' BIT
                                        ;
037226'   004737    017374'             50$:      JSR       PC,CLRDNI                     ;CLEAR DNI BIT
037232'   103004                                  BCC       55$
037234'                                           ERRHRD    133,DPPAT,RACMG7              ;ERROR DNI DID NOT CLEAR!
037234'   104456                                                                                                        TRAP      C$ERHRD
037236'   000205                                                                                                        .WORD     133
037240'   002720'                                                                                                       .WORD     DPPAT
037242'   012726'                                                                                                       .WORD     RACMG7
037244'                                 55$:
037244'                                   ENDSEG
037244'                                                                                                       10001$:
037244'   104405                                                                                                        TRAP      C$ESEG
037246'   005305                                  DEC       R5                            ;HAVE WE TESTED WITH ALL DATA PATTERNS?
037250'   001300                                  BNE       27$                           ;NOT YET
037252'                                 ENDTST
037252'                                                                                                       L10143:
037252'   104401                                                                                                        TRAP      C$ETST
```

```
                                           .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 BACKGRAOUND 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'
                                           ;
                                           ;*****************************************************************************

 037254'                                  BGNTST
 037254'                                                                                   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.
                                           ;
 037254'                                    BGNSEG
 037254' 104404                                                                                     TRAP    C$BSEG
 037256' 022737  000104  000326'                 CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
 037264' 001004                                  BNE     5$                      ;NO
```

```
037266' 122777  000001  141044          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
037274' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
037276' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
037304' 004737  020252'                 JSR     PC,LODMIC
037310' 103002                          BCC     10$                     ;OK
037312'                                 ESCAPE  TST
037312' 104410                                                                                  TRAP    C$ESCAPE
037314' 000500                                                                                  .WORD   L10144-.
037316' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
037324' 004737  017330'                 JSR     PC,CHKDNI
037330' 103022                          BCC     20$                     ;OK
037332' 012737  000776' 000310'         MOV     #$DNI,BITNAM
037340' 012737  001275' 000312'         MOV     #$NSET,BITSTA
037346' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
037354' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
037362'                                 ERRHRD  134,STAMUX,MSG1
037362' 104456                                                                                  TRAP    C$ERHRD
037364' 000206                                                                                  .WORD   134
037366' 002756'                                                                                 .WORD   STAMUX
037370' 012750'                                                                                 .WORD   MSG1
037372'                                 ESCAPE  TST
037372' 104410                                                                                  TRAP    C$ESCAPE
037374' 000420                                                                                  .WORD   L10144-.
037376' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
037402' 103006                          BCC     25$
037404'                                 ERRHRD  135,STAMUX,RACMG7       ;DNI DID NOT CLEAR!
037404' 104456                                                                                  TRAP    C$ERHRD
037406' 000207                                                                                  .WORD   135
037410' 002756'                                                                                 .WORD   STAMUX
037412' 012726'                                                                                 .WORD   RACMG7
037414'                                 ESCAPE TST
037414' 104410                                                                                  TRAP    C$ESCAPE
037416' 000376                                                                                  .WORD   L10144-.
037420'                         25$:
037420'                           ENDSEG
037420'                                                                                 10000$:
037420' 104405                                                                                  TRAP    C$ESEG

                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, THEN CLEAR OUT THE
                                ;FIRST TWO WORDS OF THE PCBB. EXECUTE MICROTEST #6 BY LOADING THE COMMAND
                                ;FIELD BITS OF PCSR0 WITH A 6, WAIT FOR 'DNI'.
                                ;
037422'                           BGNSEG
037422' 104404                                                                                  TRAP    C$BSEG
037424' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
037430' 103006                          BCC     30$                     ;OK
037432'                                 ERRHRD  136,STAMUX,MSG46        ;PRINT ERROR
037432' 104456                                                                                  TRAP    C$ERHRD
037434' 000210                                                                                  .WORD   136
037436' 002756'                                                                                 .WORD   STAMUX
037440' 016700'                                                                                 .WORD   MSG46
037442'                                 ESCAPE  TST                     ;LEAVE TEST
037442' 104410                                                                                  TRAP    C$ESCAPE
037444' 000350                                                                                  .WORD   L10144-.
037446' 005037  000604'         30$:    CLR     PCBB                    ;THIS IS WHERE THE MICROCODE WILL...
                                                                        ;PUT THE CONTENTS OF THE FIRST WORD
                                                                        ;OF THE TRANSMIT BUFFER
037452' 005037  000606'                 CLR     PCBB+2                  ;HERE IS WHERE THE SECOND WORD GOES
037456' 012777  000006  140652          MOV     #6,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #6
037464' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
037472' 004737  017330'                 JSR     PC,CHKDNI
037476' 103021                          BCC     45$
037500' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
037504' 103006                          BCC     35$                     ;NO, OK
037506'                                 ERRHRD  137,STAMUX,MSG44        ;PRINT ERROR MESSAGE
037506' 104456                                                                                  TRAP    C$ERHRD
037510' 000211                                                                                  .WORD   137
037512' 002756'                                                                                 .WORD   STAMUX
037514' 016454'                                                                                 .WORD   MSG44
037516'                                 ESCAPE  TST
037516' 104410                                                                                  TRAP    C$ESCAPE
037520' 000274                                                                                  .WORD   L10144-.
037522' 012702  000006          35$:    MOV     #6,R2                   ;MICROTEST #6 IS HUNG
037526'                                 ERRHRD  138,STAMUX,MSG12
037526' 104456                                                                                  TRAP    C$ERHRD
037530' 000212                                                                                  .WORD   138
037532' 002756'                                                                                 .WORD   STAMUX
037534' 013500'                                                                                 .WORD   MSG12
037536'                                 ESCAPE  TST
037536' 104410                                                                                  TRAP    C$ESCAPE
037540' 000254                                                                                  .WORD   L10144-.
                                ;
                                ;OK, NOW PCBB+0 SHOULD CONTAIN THE TRANSMIT STATUS WORD 0 AND PCBB+2 SHOULD
                                ;CONTAIN TRANSMIT STATUS WORD 1. CHECK THAT PCBB+0 CONTAINS STATUS BITS
                                ;FOR A GOOD TRANSMIT I.E. BITS 09:00 SHOULD ALL BE 0 BIT 13 SHOULD BE A 1
                                ;AND BIT 15 SHOULD BE A 0
                                ;
037542' 005001                  45$:    CLR     R1                      ;WE ARE GOING TO CHECK TRANSMIT WORD 0
037544' 012703  000456'                 MOV     #BNAMT2,R3              ;POINT TO A TABLE OF BIT MNEMONICS
037550' 012704  000012                  MOV     #10.,R4                 ;FIRST 10 BITS SHOULD BE 0
037554' 012702  000001                  MOV     #1,R2                   ;R2 POINTS TO THE BIT WE ARE TESTING
037560' 030237  000604'         50$:    BIT     R2,PCBB                 ;IS THIS BIT A 0?
037564' 001406                          BEQ     55$                     ;YES
037566' 011337  000310'                 MOV     (R3),BITNAM             ;NO, GET POINTER TO BIT NAME ASCII STRING
037572'                                 ERRHRD  139,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037572' 104456                                                                                  TRAP    C$ERHRD
037574' 000213                                                                                  .WORD   139
037576' 002756'                                                                                 .WORD   STAMUX
037600' 014034'                                                                                 .WORD   MSG18
037602' 062703  000002          55$:    ADD     #2,R3                   ;POINT TO NEXT BIT MNEMONIC
037606' 006302                          ASL     R2                      ;POINT TO NEXT BIT
037610' 005304                          DEC     R4                      ;HAVE WE DONE ALL 10 BITS?
037612' 001362                          BNE     50$                     ;NO

037614' 032737  020000  000604'         BIT     #BIT13,PCBB             ;IS BIT 13 A 1?
037622' 001007                          BNE     60$                     ;YES
037624' 012737  001145' 000310'         MOV     #$BIT13,BITNAM          ;NO, POINT TO ASCII STRING
037632'                                 ERRHRD  140,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037632' 104456                                                                                  TRAP    C$ERHRD
037634' 000214                                                                                  .WORD   140
037636' 002756'                                                                                 .WORD   STAMUX
037640' 014034'                                                                                 .WORD   MSG18
```

```

037642' 032737  100000  000604' 60$:    BIT     #BIT15,PCBB             ;IS BIT 15 A 0?
037650' 001407                          BEQ     65$                     ;YES
037652' 012737  001127' 000310'         MOV     #$BIT15,BITNAM          ;NO, POINT TO ASCII STRING
037660'                                 ERRHRD  141,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037660' 104456                                                                  TRAP    C$ERHRD
037662' 000215                                                                  .WORD   141
037664' 002756'                                                                 .WORD   STAMUX
037666' 014034'                                                                 .WORD   MSG18
                                ;
                                ;NOW CHECK TRANSMIT WORD 1 BITS 15,14 AND 13 TO BE ALL 0
                                ;
037670' 005201                  65$:    INC     R1                      ;WE ARE CHECKING TRNASMIT WORD 1 NOW
037672' 032737  020000  000606'         BIT     #BIT13,PCBB+2           ;IS BIT 13 A 0?
037700' 001407                          BEQ     70$                     ;YES
037702' 012737  001145' 000310'         MOV     #$BIT13,BITNAM          ;NO, POINT TO ASCII STRING
037710'                                 ERRHRD  142,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037710' 104456                                                                  TRAP    C$ERHRD
037712' 000216                                                                  .WORD   142
037714' 002756'                                                                 .WORD   STAMUX
037716' 014034'                                                                 .WORD   MSG18

037720' 032737  040000  000606' 70$:    BIT     #BIT14,PCBB+2           ;IS BIT 14 A 0?
037726' 001407                          BEQ     75$                     ;YES
037730' 012737  001136' 000310'         MOV     #$BIT14,BITNAM          ;NO, POINT TO ASCII STRING
037736'                                 ERRHRD  143,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037736' 104456                                                                  TRAP    C$ERHRD
037740' 000217                                                                  .WORD   143
037742' 002756'                                                                 .WORD   STAMUX
037744' 014034'                                                                 .WORD   MSG18

037746' 032737  100000  000606' 75$:    BIT     #BIT15,PCBB+2           ;IS BIT 15 A 0?
037754' 001407                          BEQ     80$                     ;YES
037756' 012737  001127' 000310'         MOV     #$BIT15,BITNAM          ;NO, POINT TO ASCII STRING
037764'                                 ERRHRD  144,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037764' 104456                                                                  TRAP    C$ERHRD
037766' 000220                                                                  .WORD   144
037770' 002756'                                                                 .WORD   STAMUX
037772' 014034'                                                                 .WORD   MSG18
                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
037774' 004737  017374'         80$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
040000' 103004                          BCC     85$
040002'                                 ERRHRD  145,STAMUX,RACMG7       ;ERROR DNI DID NOT CLEAR!
040002' 104456                                                                  TRAP    C$ERHRD
040004' 000221                                                                  .WORD   145
040006' 002756'                                                                 .WORD   STAMUX
040010' 012726'                                                                 .WORD   RACMG7
040012'                         85$:
040012'                           ENDSEG
040012'                                                                 10001$:
040012' 104405                                                                  TRAP    C$ESEG
040014'                         ENDTST
040014'                                                                 L10144:
040014' 104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  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 SEQUENCE:
                                ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                ;       2-WRITE MINIMUM 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
                                ;
                                ;*************************************************************************

040016'                         BGNTST
040016'                                                                 T27::
                                ;
                                ;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.
                                ;
040016'                          BGNSEG
```

```
040016' 104404                                                                          TRAP    C$BSEG
040020' 022737  000104  000326'         CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
040026' 001004                          BNE     5$                      ;NO
040030' 122777  000001  140302          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
040036' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
040040' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
040046' 004737  020252'                 JSR     PC,LODMIC
040052' 103002                          BCC     10$                     ;OK
040054'                                 ESCAPE  TST
040054' 104410                                                                          TRAP    C$ESCAPE
040056' 000334                                                                          .WORD   L10145-.
040060' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
040066' 004737  017330'                 JSR     PC,CHKDNI
040072' 103022                          BCC     20$                     ;OK
040074' 012737  000776' 000310'         MOV     #$DNI,BITNAM
040102' 012737  001275' 000312'         MOV     #$NSET,BITSTA
040110' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
040116' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
040124'                                 ERRHRD  146,LNKBYT,MSG1
040124' 104456                                                                          TRAP    C$ERHRD
040126' 000222                                                                          .WORD   146
040130' 003022'                                                                         .WORD   LNKBYT
040132' 012750'                                                                         .WORD   MSG1
040134'                                 ESCAPE  TST
040134' 104410                                                                          TRAP    C$ESCAPE
040136' 000254                                                                          .WORD   L10145-.
040140' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
040144' 103006                          BCC     25$
040146'                                 ERRHRD  147,LNKBYT,RACMG7       ;DNI DID NOT CLEAR!
040146' 104456                                                                          TRAP    C$ERHRD
040150' 000223                                                                          .WORD   147
040152' 003022'                                                                         .WORD   LNKBYT
040154' 012724'                                                                         .WORD   RACMG7
040156'                                 ESCAPE TST
040156' 104410                                                                          TRAP    C$ESCAPE
040160' 000232                                                                          .WORD   L10145-.
040162'                         25$:
040162'                           ENDSEG
040162'                                                                         10000$:
040162' 104405                                                                          TRAP    C$ESEG

                                ;
                                ;LOAD PCBB+0 WITH THE MINIMUM BYTE COUNT
                                ;
040164' 012737  000100  000604'         MOV     #MINBYT,PCBB            ;BEGIN WITH MINIMUM BYTE COUNT

                                ;
                                ;CLEAR PCBB+2 AND PCBB+4, AIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;EXECUTE MICROTEST #7 BY LOADING THE COMMAND FIELD OF PCSR0 WITH A 7.
                                ;WAIT FOR 'DNI'
                                ;
040172'                         30$:
040172'                           BGNSEG
040172' 104404                                                                          TRAP    C$BSEG
040174' 005037  000606'                 CLR     PCBB+2                  ;THIS IS WHERE MICROCODE WILL PUT...
                                                                        ;RECEIVER BYTE COUNT
040200' 005037  000610'                 CLR     PCBB+4                  ;HERE IS WHERE MICROCODE WILL PUT...
                                                                        ;ACTUAL NUMBER OF BYTES RECEIVED
```

```
040204' 004737  017776'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
040210' 103006                           BCC     35$                     ;OK
040212'                                  ERRHRD  148,LNKBYT,MSG46        ;PRINT ERROR
040212' 104456                                                                                   TRAP    C$ERHRD
040214' 000224                                                                                   .WORD   148
040216' 003022'                                                                                  .WORD   LNKBYT
040220' 016700'                                                                                  .WORD   MSG46
040222'                                  ESCAPE  TST                     ;LEAVE TEST
040222' 104410                                                                                   TRAP    C$ESCAPE
040224' 000166                                                                                   .WORD   L10145-.
040226' 012777  000007  140102  35$:     MOV     #7,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #7
040234' 012737  000176  000332'          MOV     #2*SECOND,METER         ;WAIT FOR DNI
040242' 004737  017330'                  JSR     PC,CHKDNI
040246' 103021                           BCC     40$
040250' 004737  020050'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
040254' 103006                           BCC     36$                     ;NO, OK
040256'                                  ERRHRD  149,LNKBYT,MSG44        ;PRINT ERROR MESSAGE
040256' 104456                                                                                   TRAP    C$ERHRD
040260' 000225                                                                                   .WORD   149
040262' 003022'                                                                                  .WORD   LNKBYT
040264' 016454'                                                                                  .WORD   MSG44
040266'                                  ESCAPE  TST
040266' 104410                                                                                   TRAP    C$ESCAPE
040270' 000122                                                                                   .WORD   L10145-.
040272' 012702  000007          36$:     MOV     #7,R2                   ;MICROTEST #7 IS HUNG
040276'                                  ERRHRD  150,LNKBYT,MSG12
040276' 104456                                                                                   TRAP    C$ERHRD
040300' 000226                                                                                   .WORD   150
040302' 003022'                                                                                  .WORD   LNKBYT
040304' 013500'                                                                                  .WORD   MSG12
040306'                                  ESCAPE  TST
040306' 104410                                                                                   TRAP    C$ESCAPE
040310' 000102                                                                                   .WORD   L10145-.
                                ;
                                ;OK, CHECK PCBB+2, WHICH CONTAINS THE BYTE COUNT WRITTEN INTO THE RECEIVER
                                ;BUFFER BY THE LINK, TO SEE IF IT IS THE SAME AS THE TRANSMIT BYTE COUNT. ALSO
                                ;CHECK PCBB+4, WHICH IS THE ACTUAL NUMBER OF BYTES RECEIVED IN THE RECEIVER
                                ;BUFFER, TO SEE IF IT IS THE SAME AS THE RECEIVE BYTE COUNT
                                ;IF NOT PRINT AN ERROR
                                ;
040312' 013701  000604'         40$:     MOV     PCBB,R1                 ;GET TRANSMIT BYTE COUNT
040316' 010102                           MOV     R1,R2
040320' 013703  000606'                  MOV     PCBB+2,R3               ;GET RECEIVE BYTE COUNT
040324' 013704  000610'                  MOV     PCBB+4,R4               ;GET ACTUAL NUMBER OF BYTES TRANSMITTED
040330' 020203                           CMP     R2,R3                   ;IS THE RECEIVE BYTE COUNT CORRECT?
040332' 001002                           BNE     45$                     ;NO, ERROR
040334' 020104                           CMP     R1,R4                   ;IS THE ACTUAL NUMBER OF BYTES CORRECT?
040336' 001404                           BEQ     50$                     ;YES
040340'                         45$:     ERRHRD  151,LNKBYT,MSG19        ;PRINT ERROR MESSAGE
040340' 104456                                                                                   TRAP    C$ERHRD
040342' 000227                                                                                   .WORD   151
040344' 003022'                                                                                  .WORD   LNKBYT
040346' 014064'                                                                                  .WORD   MSG19
040350'                         50$:
040350'                           ENDSEG
040350'                                                                                 10001$:
```

```
040350' 104405                                                                                  TRAP    C$ESEG
                                   ;
                                   ;WRITE '1' TO CLEAR THE 'DNI' BIT
                                   ;
040352' 004737  017374'                    JSR     PC,CLRDNI               ;CLEAR DNI BIT
040356' 103006                             BCC     60$
040360'                                    ERRHRD  152,LNKBYT,RACMG7       ;ERROR DNI DID NOT CLEAR!
040360' 104456                                                                                  TRAP    C$ERHRD
040362  000230                                                                                  .WORD   152
040364' 003022'                                                                                 .WORD   LNKBYT
040366' 012726'                                                                                 .WORD   RACMG7
040370'                                    ESCAPE  TST
040370' 104410                                                                                  TRAP    C$ESCAPE
040372' 000020                                                                                  .WORD   L10145-.
                                   ;
                                   ;CONTINUE CALLING MICROTEST #7 EACH TIME PASSING A LARGER (EVEN) BYTE COUNT
                                   ;UNTIL THE BYTE COUNT REACHES THE MAXIMUM COUNT ALLOWED
                                   ;
040374' 062737  000002  000604' 60$:       ADD     #2,PCBB                 ;UP THE BYTE COUNT
040402' 023727  000604' 002756             CMP     PCBB,#MAXBYT            ;HAVE WE DONE ALL EVEN BYTE COUNTS?
040410' 003670                             BLE     30$                     ;NO, CONTINE TESTING WITH NEW BYTE COUNT
040412'                            ENDTST
040412'                                                                             L10145:
040412' 104401                                                                                  TRAP    C$ETST
```

```
                                                .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
                                                ;
                                                ;***********************************************************************

 040414'                                        BGNTST
 040414'                                                                                    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.
                                                ;
 040414'                                          BGNSEG
 040414' 104404                                                                                    TRAP    C$BSEG
 040416' 022737  000104  000326'        CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
 040424' 001004                         BNE     5$                      ;NO
 040426' 122777  000001  137704         CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 040434' 001440                         BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 040436' 012737  000104  000326' 5$:    MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
 040444' 004737  020252'                JSR     PC,LODMIC
 040450' 103002                         BCC     10$                     ;OK
 040452'                                ESCAPE  TST
 040452' 104410                                                                                    TRAP    C$ESCAPE
 040454' 000340                                                                                    .WORD   L10146-.
 040456' 012737  000176  000332' 10$:   MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
 040464' 004737  017330'                JSR     PC,CHKDNI
 040470' 103022                         BCC     20$                     ;OK
 040472' 012737  000776' 000310'        MOV     #$DNI,BITNAM
 040500' 012737  001275' 000312'        MOV     #$NSET,BITSTA
 040506' 012737  001340' 000314'        MOV     #$AFTER,PWHEN
 040514' 012737  001355' 000316'        MOV     #$GTCMD,PCOMND
 040522'                                ERRHRD  153,ODDBYT,MSG1
 040522' 104456                                                                                    TRAP    C$ERHRD
 040524' 000231                                                                                    .WORD   153
 040526' 003060'                                                                                   .WORD   ODDBYT
 040530' 012750'                                                                                   .WORD   MSG1
```

```
040532'                                 ESCAPE  TST
040532' 104410                                                                                  TRAP    C$ESCAPE
040534' 000260                                                                                  .WORD   L10146-.
040536' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
040542' 103006                          BCC     25$
040544'                                 ERRHRD  154,ODDBYT,RACMG7       ;DNI DID NOT CLEAR!
040544' 104456                                                                                  TRAP    C$ERHRD
040546' 000232                                                                                  .WORD   154
040550' 003060'                                                                                 .WORD   ODDBYT
040552' 012726'                                                                                 .WORD   RACMG7
040554'                                 ESCAPE  TST
040554' 104410                                                                                  TRAP    C$ESCAPE
040556' 000236                                                                                  .WORD   L10146-.
040560'                         25$:
040560'                           ENDSEG
040560'                                                                                 10000$:
040560' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;LOAD PCBB+0 WITH THE MINIMUM ODD BYTE COUNT
                                ;
040562' 012737  000101  000604'         MOV     #MINBYT+1,PCBB          ;BEGIN WITH MINIMUM ODD BYTE COUNT
040570'                         30$:
                                ;
                                ;CLEAR PCBB+2 AND PCBB+4. WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON'
                                ;STATE. EXECUTE MICROTEST #7 BY LOADING THE COMMAND FIELD OF PCSRO WITH A
                                ;'7'. WAIT FOR 'DNI'
                                ;
040570'                           BGNSEG
040570' 104404                                                                                  TRAP    C$BSEG
040572' 005037  000606'                 CLR     PCBB+2                  ;THIS IS WHERE MICROCODE WILL PUT...
                                                                        ;RECEIVER BYTE COUNT
040576' 005037  000610'                 CLR     PCBB+4                  ;HERE IS WHERE MICROCODE WILL PUT...
                                                                        ;ACTUAL NUMBER OF BYTES RECEIVED
040602' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
040606' 103006                          BCC     35$                     ;OK
040610'                                 ERRHRD  156,ODDBYT,MSG46        ;PRINT ERROR
040610' 104456                                                                                  TRAP    C$ERHRL
040612' 000234                                                                                  .WORD   156
040614' 003060'                                                                                 .WORD   ODDBYT
040616' 016700'                                                                                 .WORD   MSG46
040620'                                 ESCAPE  TST                     ;LEAVE TEST
040620' 104410                                                                                  TRAP    C$ESCAPE
040622' 000172                                                                                  .WORD   L10146-.
040624' 012777  000007  137504  35$:    MOV     #7,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #7
040632' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
040640' 004737  017330'                 JSR     PC,CHKDNI
040644' 103021                          BCC     40$
040646' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
040652' 103006                          BCC     36$                     ;NO, OK
040654'                                 ERRHRD  157,ODDBYT,MSG44        ;PRINT ERROR MESSAGE
040654' 104456                                                                                  TRAP    C$ERHRD
040656' 000235                                                                                  .WORD   157
040660' 003060'                                                                                 .WORD   ODDBYT
040662' 016454'                                                                                 .WORD   MSG44
040664'                                 ESCAPE  TST
040664' 104410                                                                                  TRAP    C$ESCAPE
```

```
040666' 000126                                                                                  .WORD   L10146-.
040670' 012702  000007          36$:    MOV     #7,R2                   ;MICROTEST #7 IS HUNG
040674'                                 ERRHRD  158,ODDBYT,MSG12
040674' 104456                                                                                  TRAP    C$ERHRD
040676' 000236                                                                                  .WORD   158
040700' 003060'                                                                                 .WORD   ODDBYT
040702' 013500'                                                                                 .WORD   MSG12
040704'                                 ESCAPE  TST
040704' 104410                                                                                  TRAP    C$ESCAPE
040706' 000106                                                                                  .WORD   L10146-.
                                ;
                                ;CHECK PCBB+2 TO SEE IF IT IS THE SAME AS PCBB+0, THIS WILL VERIFY THE
                                ;THE TRANSMIT AND RECEIVE BYTE COUNTS ARE THE SAME. THEN CHECK PCBB+4
                                ;TO SEE IF IT IS ONE LARGER THAN THE RECEIVE BYTE COUNT. THIS IS BECAUSE
                                ;THE RECEIVE CAN ONLY ADDRESS LINK MEMORY IN WORD BOUNDARIES SO THE ACTUAL
                                ;NUMBER OF BYTES THAT ARE CHANGED IS ONE MORE THAN WHAT WAS TRANSMITTED
                                ;
040710' 0 3701  000604'         40$:    MOV     PCBB,R1                 ;GET TRANSMIT BYTE COUNT
040714' 010102                          MOV     R1,R2
040716' 013703  000606'                 MOV     PCBB+2,R3               ;GET RECEIVE BYTE COUNT
040722' 013704  000610'                 MOV     PCBB+4,R4               ;GET ACTUAL NUMBER OF BYTES TRANSMITTED
040726' 020203                          CMP     R2,R3                   ;IS THE RECEIVE BYTE COUNT CORRECT?
040730' 001004                          BNE     45$                     ;NO, ERROR
040732' 162704  000001                  SUB     #1,R4                   ;ACCOUNT FOR THE GARBAGE BYTE
040736' 020104                          CMP     R1,R4                   ;IS THE ACTUAL NUMBER OF BYTES CORRECT?
040740' 001404                          BEQ     50$                     ;YES
040742'                         45$:    ERRHRD  159,ODDBYT,MSG19        ;PRINT ERROR MESSAGE
040742' 104456                                                                                  TRAP    C$ERHRD
040744' 000237                                                                                  .WORD   159
040746' 003060'                                                                                 .WORD   ODDBYT
040750' 014064'                                                                                 .WORD   MSG19
040752'                         50$:
040752'                           ENDSEG
040752'                                                                                 10001$:
040752' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
040754' 004737  017374'                 JSR     PC,CLRDNI               ;CLEAR DNI BIT
040760' 103006                          BCC     60$
040762'                                 ERRHRD  160,ODDBYT,RACMG7       ;ERROR DNI DID NOT CLEAR!
040762' 104456                                                                                  TRAP    C$ERHRD
040764' 000240                                                                                  .WORD   160
040766' 003060'                                                                                 .WORD   ODDBYT
040770' 012726'                                                                                 .WORD   RACMG7
040772'                                 ESCAPE  TST
040772' 104410                                                                                  TRAP    C$ESCAPE
040774' 000020                                                                                  .WORD   L10146-.
                                ;
                                ;REPEAT MICROTEST #7 EACH TIME CIVING IT A LARGER ODD BYTE COUNT. CONTINUE
                                ;UNTIL THE MAXIMUM BYTE COUNT IS REACHED
                                ;
040776' 062737  000002  000604' 60$:    ADD     #2,PCBB                 ;UP THE BYTE COUNT
041004' 023727  000604' 002756          CMP     PCBB,#MAXBYT            ;HAVE WE DONE ALL ODD BYTE COUNTS?
041012' 003666                          BLE     30$                     ;NO, CONTINUE TEST WITH NEW BYTE COUNT
041014'                         ENDTST
```

```
 9291  041014'                                                              L10146:
 9292  041014' 104401                                                               TRAP    C$ETST
```

```
                                .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 PCSR1
                                ;       6-WRITE '1' TO CLEAR 'DNI'
                                ;
                                ;*******************************************************************

041016'                         BGNTST
041016'                                                                            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.
                                ;
041016'                          BGNSEG
041016' 104404                                                                                TRAP    C$BSEG
041020' 022737  000104  000326'         CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED?
041026' 001004                          BNE     5$                      ;NO
041030' 122777  000001  137302          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
041036' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
041040' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICROCODE MODULE D
041046' 004737  020252'                 JSR     PC,LODMIC
041052' 103002                          BCC     10$                     ;OK
041054'                                 ESCAPE  TST
041054' 104410                                                                                TRAP    C$ESCAPE
041056' 000256                                                                                .WORD   L10147-.
041060' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
041066' 004737  017330'                 JSR     PC,CHKDNI
041072' 103022                          BCC     20$                     ;OK
041074' 012737  000776' 000310'         MOV     #$DNI,BITNAM
041102' 012737  001275' 000312'         MOV     #$NSET,BITSTA
041110' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
041116' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
041124'                                 ERRHRD  161,MAXCNT,MSG1 ;PRINT ERROR MESSAGE
041124' 104456                                                                                TRAP    C$ERHRD
```

```
041126' 000241                                                                        .WORD   161
041130' 003122'                                                                       .WORD   MAXCNT
041132' 012750'                                                                       .WORD   MSG1
041134'                                  ESCAPE  TST
041134' 104410                                                                        TRAP    C$ESCAPE
041136' 000176                                                                        .WORD   L10147-.
041140' 004737  017374'         20$:     JSR     PC,CLRDNI               ;CLEAR DNI
041144' 103006                           BCC     25$
041146'                                  ERRHRD  162,MAXCNT,RACMG7
041146' 104456                                                                        TRAP    C$ERHRD
041150' 000242                                                                        .WORD   162
041152' 003122'                                                                       .WORD   MAXCNT
041154' 012726'                                                                       .WORD   RACMG7
041156'                                  ESCAPE  TST
041156' 104410                                                                        TRAP    C$ESCAPE
041160' 000154                                                                        .WORD   L10147-.
041162'                         25$:
041162'                           ENDSEG
041162'                                                                      10000$:
041162' 104405                                                                        TRAP    C$ESEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE,  LOAD COMMAND
                                ;FIELD OF PCSRO WITH A 9 CAUSING THE MICROMONITOR TO EXECUTE MICROTEST #9.
                                ;WAIT FOR DNI
                                ;
041164'                           BGNSEG
041164' 104404                                                                        TRAP    C$BSEG
041166' 004737  017776'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
041172' 103006                           BCC     30$                     ;OK
041174'                                  ERRHRD  163,MAXCNT,MSG46        ;PRINT ERROR
041174' 104456                                                                        TRAP    C$ERHRD
041176' 000243                                                                        .WORD   163
041200' 003122'                                                                       .WORD   MAXCNT
041202' 016700'                                                                       .WORD   MSG46
041204'                                  ESCAPE  TST                     ;LEAVE TEST
041204' 104410                                                                        TRAP    C$ESCAPE
041206' 000126                                                                        .WORD   L10147-.
041210' 012777  000011  137120  30$:     MOV     #9.,@PCSRO              ;TELL MICROMONITOR TO EXECUTE...
                                                                         ;MICROTEST #9
041216' 012737  000176  000332'          MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
041224' 004737  017330'                  JSR     PC,CHKDNI               ;WAIT FOR MICROTEST TO FINISH
041230' 103021                           BCC     40$                     ;OK, IT FINISHED
041232' 004737  020050'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
041236' 103006                           BCC     35$                     ;NO, OK
041240'                                  ERRHRD  164,MAXCNT,MSG44        ;PRINT ERROR MESSAGE
041240' 104456                                                                        TRAP    C$ERHRD
041242' 000244                                                                        .WORD   164
041244' 003122'                                                                       .WORD   MAXCNT
041246' 016454'                                                                       .WORD   MSG44
041250'                                  ESCAPE  TST
041250' 104410                                                                        TRAP    C$ESCAPE
041252' 000062                                                                        .WORD   L10147-.
041254' 012702  000011          35$:     MOV     #9.,R2                  ;MICROTEST #
041260'                                  ERRHRD  165,MAXCNT,MSG12        ;TELL MICROTEST HUNG
041260' 104456                                                                        TRAP    C$ERHRD
041262' 000245                                                                        .WORD   165
```

```
041264' 003122'                                                                           .WORD   MAXCNT
041266' 013500'                                                                           .WORD   MSG12
041270'                                 ESCAPE  TST
041270' 104410                                                                            TRAP    C$ESCAPE
041272' 000042                                                                            .WORD   L10147-.
                                ;
                                ;MICROTEST IS COMPLETE, NOW CHECK TO SEE IF IT WAS SUCCESSFULL
                                ;
041274' 122777  000003  137036  40$:    CMPB    #INERR,@PCSR1     ;DID AN ERROR OCCUR?
041302' 001004                          BNE     50$               ;NO
041304'                                 ERRHRD  166,MAXCNT,MSG38  ;PRINT ERROR MESSAGE
041304' 104456                                                                            TRAP    C$ERHRD
041306' 000246                                                                            .WORD   166
041310' 003122'                                                                           .WORD   MAXCNT
041312' 016046'                                                                           .WORD   MSG38
                                ;
                                ;WRITE ONE TO CLEAR THE DNI BIT
                                ;
041314' 004737  017374'         50$:    JSR     PC,CLRDNI         ;CLEAR DNI
041320' 103004                          BCC     55$
041322'                                 ERRHRD  167,MAXCNT,RACMG7
041322' 104456                                                                            TRAP    C$ERHRD
041324' 000247                                                                            .WORD   167
041326' 003122'                                                                           .WORD   MAXCNT
041330' 012726'                                                                           .WORD   RACMG7
041332'                         55$:
041332'                            ENDSEG
041332'                                                                          10001$:
041332' 104405                                                                            TRAP    C$ESEG
041334'                         ENDTST
041334'                                                                          L10147:
041334' 104401                                                                            TRAP    C$ETST
```

```
                                        .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
                                        ;
                                        ;*******************************************************************************

 041336'                                BGNTST
 041336'                                                                      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.
                                        ;
```

```
041336'                                  BGNSEG
041336' 104404                                                                                  TRAP    C$BSEG
041340' 022737  000104  000326'                 CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
041346' 001004                                  BNE     5$                      ;NO
041350' 122777  000001  136762                  CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
041356' 001440                                  BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
041360' 012737  000104  000326' 5$:             MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
041366' 004737  020252'                         JSR     PC,LODMIC
041372' 103002                                  BCC     10$                     ;OK
041374'                                         ESCAPE
041374' 104410                                                                                  TRAP    C$ESCAPE
041376' 000326                                                                                  .WORD   L10150-.
041400' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
041406' 004737  017330'                         JSR     PC,CHKDNI
041412' 103022                                  BCC     20$                     ;OK
041414' 012737  000776' 000310'                 MOV     #$DNI,BITNAM
041422' 012737  001275' 000312'                 MOV     #$NSET,BITSTA
041430' 012737  001340' 000314'                 MOV     #$AFTER,PWHEN
041436' 012737  001355' 000316'                 MOV     #$GTCMD,PCOMND
041444'                                         ERRHRD  168,FIFTST,MSG1
041444' 104456                                                                                  TRAP    C$ERHRD
041446' 000250                                                                                  .WORD   168
041450' 003166'                                                                                 .WORD   FIFTST
041452' 012750'                                                                                 .WORD   MSG1
041454'                                         ESCAPE  TST
041454' 104410                                                                                  TRAP    C$ESCAPE
041456' 000246                                                                                  .WORD   L10150-.
041460' 004737  017374'         20$:            JSR     PC,CLRDNI               ;CLEAR DNI BIT
041464' 103006                                  BCC     25$
041466'                                         ERRHRD  169,FIFTST,RACMG7       ;DNI DID NOT CLEAR!
041466' 104456                                                                                  TRAP    C$ERHRD
041470' 000251                                                                                  .WORD   169
041472' 003166'                                                                                 .WORD   FIFTST
041474' 012726'                                                                                 .WORD   RACMG7
041476'                                         ESCAPE TST
041476' 104410                                                                                  TRAP    C$ESCAPE
041500' 000224                                                                                  .WORD   L10150-.
041502'                         25$:
041502'                           ENDSEG
041502'                                                                                 10000$:
041502' 104405                                                                                  TRAP    C$ESEG

                                ;
                                ;LINK MEMORY STARTS AT MICROMEMORY ADDRESS 100000, THE MICROTEST WILL USE
                                ;100000 AS THE TRANSMIT BUFFER ADDRESS SO THE FIRST RECEIVE BUFFER ADDRESS
                                ;WE WILL PASS TO THE MICROCODE WILL BE 104000
                                ;
041504' 012701  104000                          MOV     #LINADR+SIZ1K,R1        ;FIRST RECEIVE BUFFER STARTS 1K FROM
                                                                                ;BASE OF LINK MEMORY
041510'                         30$:
                                ;
                                ;PUT THE BUFFER ADDRESS WE WANT THE MICROCODE TO LOAD INTO THE AVAILABLE BUFFER
                                ;INTO PCBB AND CLEAR PCBB+2. WAIT FOR THE MICROMONITOR TO BECOME READY, THEN
                                ;SELECT MICROTEST #10 BY LOADING THE COMMAND FIELD OF PCSR0 WITH A 10. WAIT FOR
                                ;'DNI'
                                ;
041510'                           BGNSEG
```

```
041510' 104404                                                                                          TRAP      C$BSEG
041512' 010137   000604'                  MOV     R1,PCBB                 ;PASS AVAILABLE RECEIVER BUFFER TO DEUNA
041516' 005037   000606'                  CLR     PCBB+2                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                          ;'DONE' RECEIVE BUFFER FIFO ADDRESS
041522' 004737   017776'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
041526' 103006                            BCC     35$                     ;OK
041530'                                   ERRHRD  170,FIFTST,MSG46        ;PRINT ERROR
041530' 104456                                                                                          TRAP      C$ERHRD
041532' 000252                                                                                          .WORD     170
041534' 003166'                                                                                         .WORD     FIFTST
041536' 016700'                                                                                         .WORD     MSG46
041540'                                   ESCAPE  TST                     ;LEAVE TEST
041540' 104410                                                                                          TRAP      C$ESCAPE
041542' 000162                                                                                          .WORD     L10150-.
041544' 012777   000012   136564   35$:   MOV     #10.,@PCSR0             ;TELL T11 TO EXECUTE MICROTEST #10
041552' 012737   000176   000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
041560' 004737   017330'                  JSR     PC,CHKDNI
041564' 103021                            BCC     45$                     ;OK
041566' 004737   020050'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
041572' 103006                            BCC     40$                     ;NO, OK
041574'                                   ERRHRD  171,FIFTST,MSG44        ;PRINT ERROR MESSAGE
041574' 104456                                                                                          TRAP      C$ERHRD
041576' 000253                                                                                          .WORD     171
041600' 003166'                                                                                         .WORD     FIFTST
041602' 016454'                                                                                         .WORD     MSG44
041604'                                   ESCAPE  TST
041604' 104410                                                                                          TRAP      C$ESCAPE
041606' 000116                                                                                          .WORD     L10150-.
041610' 012702   000012            40$:   MOV     #10.,R2                 ;MICROTEST #10 IS HUNG
041614'                                   ERRHRD  172,FIFTST,MSG12        ;PRINT ERROR MESSAGE
041614' 104456                                                                                          TRAP      C$ERHRD
041616' 000254                                                                                          .WORD     172
041620' 003166'                                                                                         .WORD     FIFTST
041622' 013500'                                                                                         .WORD     MSG12
041624'                                   ESCAPE TST
041624' 104410                                                                                          TRAP      C$ESCAPE
041626' 000076                                                                                          .WORD     L10150-.
                                   ;
                                   ;OK, THE MICROCODE HAS FILLED PCBB+2 WITH THE OUTPUT OF THE RECEIVER BUFFER
                                   ;DONE FIFO HOWEVER, THIS VALUE IS NOT IN ADDRESS FORM BECAUSE THERE ARE SOME
                                   ;UNUSED BITS (15,10:00) THAT COULD BE FLOATING SO, STRIP THESE BITS FIRST.
                                   ;THEN SET BIT 15 TO FORCE THE ADDRESS TO BE A LINK MEMORY ADDRESS. COMPARE
                                   ;THE ADDRESS GIVEN TO THE MICROCODE WITH THE ONE JUST GENERATED, THEY SHOULD
                                   ;BE IDENTICAL, IF NOT, PRINT ERROR.
                                   ;
041630' 042737   103777   000606'  45$:   BIC     #103777,PCBB+2          ;STRIP OFF THE FLOATING BITS
041636' 052737   100000   000606'         BIS     #100000,PCBB+2          ;MAKE IT A LINK MEMORY ADDRESS
041644' 023737   000604'  000606'         CMP     PCBB,PCBB+2             ;DID THE DEUNA USE THE BUFFER AVAILABLE
041652' 001404                            BEQ     50$                     ;YES
041654'                                   ERRHRD  173,FIFTST,MSG35        ;NO, PRINT ERROR MESSAGE
041654' 104456                                                                                          TRAP      C$ERHRD
041656' 000255                                                                                          .WORD     173
041660' 003166'                                                                                         .WORD     FIFTST
041662' 015704'                                                                                         .WORD     MSG35
041664'                            50$:
041664'                              ENDSEG
```

```
041664'                                                                      10001$:
041664' 104405                                                                          TRAP    C$ESEG
                          ;
                          ;WRITE '1' TO CLEAR 'DNI' BIT
                          ;
041666' 004737  017374'           JSR     PC,CLRDNI                   ;GO CLEAR DNI
041672' 103006                    BCC     60$                         ;OK
041674'                           ERRHRD  174,FIFTST,RACMG7           ;ERROR DNI DID NOT CLEAR!
041674' 104456                                                                          TRAP    C$ERHRD
041676' 000256                                                                          .WORD   174
041700' 003166'                                                                         .WORD   FIFTST
041702' 012726'                                                                         .WORD   RACMG7
041704'                           ESCAPE TST
041704' 104410                                                                          TRAP    C$ESCAPE
041706' 000016                                                                          .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
                          ;
041710' 020127  174000    60$:    CMP     R1,#LINADR+<SIZ1K*15.>      ;HAVE WE TRIED ALL 15 BUFFERS
041714' 001403                    BEQ     70$                         ;YES ALL DONE
041716' 062701  004000            ADD     #SIZ1K,R1                   ;NO POINT TO NEXT RECEIVER BUFFER
041722' 000672                    BR      30$                         ;DO AGAIN
041724'                   70$:
041724'                   ENDTST
041724'                                                                      L10150:
041724' 104401                                                                          TRAP    C$ETST
```

```
                                        .SBTTL  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 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
                                        ;
                                        ;************************************************************************

 041726'                                BGNTST
 041726'                                                                            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.
                                        ;
 041726'                                 BGNSEG
 041726' 104404                                                                        TRAP    C$BSEG
 041730' 022737  000104  000326'               CMP     #'D,MICRO              ;HAS MICROCODE MODULE 'D' BEEN LOADED
 041736' 001004                                BNE     5$                     ;NO
 041740' 122777  000001  136372                CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
 041746' 001440                                BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
```

TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

```
041750' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
041756' 004737  020252'                 JSR     PC,LODMIC
041762' 103002                          BCC     10$                     ;OK
041764'                                 ESCAPE  TST
041764' 104410                                                                                  TRAP     C$ESCAPE
041766' 000322                                                                                  .WORD    L10151-.
041770' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
041776' 004737  017330'                 JSR     PC,CHKDNI
042002' 103022                          BCC     20$                     ;OK
042004' 012737  000776' 000310'         MOV     #$DNI,BITNAM
042012' 012737  001275' 000312'         MOV     #$NSET,BITSTA
042020' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
042026' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
042034'                                 ERRHRD  175,RLNKAD,MSG1
042034' 104456                                                                                  TRAP     C$ERHRD
042036' 000257                                                                                  .WORD    175
042040' 003207'                                                                                 .WORD    RLNKAD
042042' 012750'                                                                                 .WORD    MSG1
042044'                                 ESCAPE  TST
042044' 104410                                                                                  TRAP     C$ESCAPE
042046' 000242                                                                                  .WORD    L10151-.
042050' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
042054' 103006                          BCC     25$
042056'                                 ERRHRD  176,RLNKAD,RACMG7       ;DNI DID NOT CLEAR!
042056' 104456                                                                                  TRAP     C$ERHRD
042060' 000260                                                                                  .WORD    176
042062' 003207'                                                                                 .WORD    RLNKAD
042064' 012726'                                                                                 .WORD    RACMG7
042066'                                 ESCAPE TST
042066' 104410                                                                                  TRAP     C$ESCAPE
042070' 000220                                                                                  .WORD    L10151-.
042072'                         25$:
042072'                           ENDSEG
042072'                                                                                 10000$:
042072' 104405                                                                                  TRAP     C$ESEG
                                ;
                                ;LOAD PCBB+2 WITH THE ADDRESS OF THE FIRST TRANSMIT BUFFER
                                ;LOAD PCBB+0 WITH THE FIRST RECEIVER BUFFER ADDRESS
                                ;
042074' 012701  002756                  MOV     #MAXBYT,R1              ;TRANSMIT BYTE COUNT
042100' 012737  100000  000606'         MOV     #LINADR,PCBB+2          ;SET TRANSMIT BUFFER AT BASE OF LINK
                                                                        ;MEMORY
042106' 012737  104000  000604'         MOV     #LINADR+SIZ1K,PCBB      ;FIRST RECEIVE BUFFER STARTS 1K FROM
                                                                        ;BASE OF LINK MEMORY
042114'                         30$:
042114'                           BGNSEG
042114' 104404                                                                                  TRAP     C$BSEG
                                ;
                                ;WAIT FOR THE MICROMONITOR, THEN SELECT MICROTEST #11 BY LOADING THE COMMAND
                                ;FIELD OF PCSR0 WITH 11 , WAIT FOR 'DNI'
                                ;
042116' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
042122' 103006                          BCC     35$                     ;OK
042124'                                 ERRHRD  177,RLNKAD,MSG46        ;PRINT ERROR
042124' 104456                                                                                  TRAP     C$ERHRD
042126' 000261                                                                                  .WORD    177
```

```
042130' 003207'                                                                                 .WORD   RLNKAD
042132' 016700'                                                                                 .WORD   MSG46
042134'                                 ESCAPE  TST                     ;LEAVE TEST
042134' 104410                                                                                  TRAP    C$ESCAPE
042136' 000152                                                                                  .WORD   L10151-.
042140' 012777  000013  136170  35$:    MOV     #11.,@PCSR0             ;TELL T11 TO EXECUTE MICROTEST #11
042146' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
042154' 004737  017330'                 JSR     PC,CHKDNI
042160' 103021                          BCC     45$                     ;OK
042162' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
042166' 103006                          BCC     40$                     ;NO, OK
042170'                                 ERRHRD  178,RLNKAD,MSG44        ;PRINT ERROR MESSAGE
042170' 104456                                                                                  TRAP    C$ERHRD
042172' 000262                                                                                  .WORD   178
042174' 003207'                                                                                 .WORD   RLNKAD
042176' 016454'                                                                                 .WORD   MSG44
042200'                                 ESCAPE  TST
042200' 104410                                                                                  TRAP    C$ESCAPE
042202' 000106                                                                                  .WORD   L10151-.
042204' 012702  000013          40$:    MOV     #11.,R2                 ;MICROTEST #11 IS HUNG
042210'                                 ERRHRD  179,RLNKAD,MSG12        ;PRINT ERROR MESSAGE
042210' 104456                                                                                  TRAP    C$ERHRD
042212' 000263                                                                                  .WORD   179
042214' 003207'                                                                                 .WORD   RLNKAD
042216' 013500'                                                                                 .WORD   MSG12
042220'                                 ESCAPE TST
042220' 104410                                                                                  TRAP    C$ESCAPE
042222' 000066                                                                                  .WORD   L10151-.
                                ;
                                ;IF THE MICROCODE FOUND AN ERROR IT SETS PCSR1 TO THE ERROR STATE SO, CHECK
                                ;PCSR1 TO SEE IF AN ERROR OCCURRED
                                ;IF SO PRINT ERROR MESSAGE
                                ;
042224' 122777  000003  136106  45$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
042232' 001004                          BNE     50$                     ;NO
042234'                                 ERRHRD  180,RLNKAD,MSG42        ;YES, PRINT ERROR MESSAGE
042234' 104456                                                                                  TRAP    C$ERHRD
042236' 000264                                                                                  .WORD   180
042240' 003207'                                                                                 .WORD   RLNKAD
042242' 016262'                                                                                 .WORD   MSG42
042244'                         50$:
042244'                           ENDSEG
042244'                                                                                 10001$:
042244' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
042246' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
042252' 103006                          BCC     60$                     ;OK
042254'                                 ERRHRD  181,RLNKAD,RACMG7       ;ERROR DNI DID NOT CLEAR!
042254' 104456                                                                                  TRAP    C$ERHRD
042256' 000265                                                                                  .WORD   181
042260' 003207'                                                                                 .WORD   RLNKAD
042262' 012726'                                                                                 .WORD   RACMG7
042264'                                 ESCAPE TST
042264' 104410                                                                                  TRAP    C$ESCAPE
```

```
042266' 000022                                                                                  .WORD   L10151-.
                                        ;
                                        ;THIS TEST IS TO BE REPEATED 15 TIMES ONECE 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
                                        ;
042270' 023727  000604' 174000  60$:    CMP     PCBB,#LINADR+<SIZ1K*15.> ;HAVE WE TRIED ALL 15 BUFFERS
042276' 001404                          BEQ     70$                      ;YES ALL DONE
042300' 062737  004000  000604'         ADD     #SIZ1K,PCBB              ;NO POINT TO NEXT RECEIVER BUFFER
042306' 000702                          BR      30$                      ;DO AGAIN
042310'                         70$:
042310'                         ENDTST
042310'                                                                          L10151:
042310' 104401                                                                          TRAP    C$ETST
```

```
                                        .SBTTL  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 SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                        ;       2-SET PCBB+0 TO 100000 (BASE OF LINK MEMORY) AS RECEIVER BUFFER ADDRESS
                                        ;       3-SET PCBB+2 WITH A TRANSMITTER BUFFER ADDRESS
                                        ;       4-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;       5-SELECT MICROTEST #11
                                        ;       6-WAIT FOR 'DNI'
                                        ;       7-CHECK PCSR1 FOR AN ERROR CONDITION IF SO PRINT ERROR MESSAGE
                                        ;       8-WRITE '1' TO CLEAR 'DNI'
                                        ;       9-REPEAT STEPS 2-8 FOR ALL TRANSMIT BUFFERS 1-15
                                        ;
                                        ;*************************************************************************

 042312'                                BGNTST
 042312'                                                                                 T32::
                                        ;
                                        ;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.
                                        ;
 042312'                                 BGNSEG
 042312' 104404                                                                                        TRAP    C$BSEG
 042314' 022737  000104  000326'               CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
 042322' 001004                                BNE     5$                      ;NO
 042324' 122777  000001  136006                CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 042332' 001440                                BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 042334' 012737  000104  000326' 5$:           MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
 042342' 004737  020252'                       JSR     PC,LODMIC
 042346' 103002                                BCC     10$                     ;OK
 042350'                                       ESCAPE  TST
 042350' 104410                                                                                        TRAP    C$ESCAPE
 042352' 000322                                                                                        .WORD   L10152-.
 042354' 012737  000176  000332' 10$:          MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
 042362' 004737  017330'                       JSR     PC,CHKDNI
 042366' 103022                                BCC     20$                     ;OK
 042370' 012737  000776' 000310'               MOV     #$DNI,BITNAM
 042376' 012737  001275' 000312'               MOV     #$NSET,BITSTA
 042404' 012737  001340' 000314'               MOV     #$AFTER,PWHEN
 042412' 012737  001355' 000316'               MOV     #$GTCMD,PCOMND
 042420'                                       ERRHRD  182,TLNKAD,MSG1
 042420' 104456                                                                                        TRAP    C$ERHRD
 042422' 000266                                                                                        .WORD   182
 042424' 003260'                                                                                       .WORD   TLNKAD
 042426' 012750'                                                                                       .WORD   MSG1
 042430'                                       ESCAPE  TST
 042430' 104410                                                                                        TRAP    C$ESCAPE
 042432' 000242                                                                                        .WORD   L10152-.
```

```
042434' 004737  017374'           20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
042440' 103006                            BCC     25$
042442'                                   ERRHRD  183,TLNKAD,RACMG7       ;DNI DID NOT CLEAR!
042442' 104456                                                                                     TRAP    C$ERHRD
042444' 000267                                                                                     .WORD   183
042446' 003260'                                                                                    .WORD   TLNKAD
042450' 012726'                                                                                    .WORD   RACMG7
042452'                                   ESCAPE  TST
042452' 104410                                                                                     TRAP    C$ESCAPE
042454' 000220                                                                                     .WORD   L10152-.
042456'                           25$:
042456'                             ENDSEG
042456'                                                                                  10000$:
042456' 104405                                                                                     TRAP    C$ESEG

                                  ;
                                  ;LOAD PCBB+0 WITH THE BASE ADDRESS OF LINK MEMORY. THIS IS WHERE THE RECEIVER
                                  ;BUFFER WILL BE FIXED AT. LOAD PCBB+2 WITH THE ADDRESS OF BUFFER #1 (104000)
                                  ;
042460' 012701  002756                    MOV     #MAXBYT,R1              ;TRANSMIT BYTE COUNT
042464' 012737  100000  000604'           MOV     #LINADR,PCBB            ;SET RECEIVE BUFFER AT BASE OF LINK
                                                                          ;MEMORY
042472' 012737  104000  000606'           MOV     #LINADR+SIZ1K,PCBB+2    ;TRANSMIT BUFFER STARTS 1K FROM
                                                                          ;BASE OF LINK MEMORY
042500'                           30$:
                                  ;
                                  ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE. THEN EXECUTE MICROTEST
                                  ;#11 BY LOADING PCSR0 COMMAND FIELD BITS WITH A 11. WAIT FOR 'DNI'
                                  ;
042500'                             BGNSEG
042500' 104404                                                                                     TRAP    C$BSEG
042502' 004737  017776'                   JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
042506' 103006                            BCC     35$                     ;OK
042510'                                   ERRHRD  184,TLNKAD,MSG46        ;PRINT ERROR
042510' 104456                                                                                     TRAP    C$ERHRD
042512' 000270                                                                                     .WORD   184
042514' 003260'                                                                                    .WORD   TLNKAD
042516' 016700'                                                                                    .WORD   MSG46
042520'                                   ESCAPE  TST                     ;LEAVE TEST
042520' 104410                                                                                     TRAP    C$ESCAPE
042522' 000152                                                                                     .WORD   L10152-.
042524' 012777  000013  135604    35$:    MOV     #11.,@PCSR0             ;TELL T11 TO EXECUTE MICROTEST #11
042532' 012737  000176  000332'           MOV     #2*SECOND,METER         ;WAIT FOR DNI
042540' 004737  017330'                   JSR     PC,CHKDNI
042544' 103021                            BCC     45$                     ;OK
042546' 004737  020050'                   JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
042552' 103006                            BCC     40$                     ;NO, OK
042554'                                   ERRHRD  185,TLNKAD,MSG44        ;PRINT ERROR MESSAGE
042554' 104456                                                                                     TRAP    C$ERHRD
042556' 000271                                                                                     .WORD   185
042560' 003260'                                                                                    .WORD   TLNKAD
042562' 016454'                                                                                    .WORD   MSG44
042564'                                   ESCAPE  TST
042564' 104410                                                                                     TRAP    C$ESCAPE
042566' 000106                                                                                     .WORD   L10152-.
042570' 012702  000013            40$:    MOV     #11.,R2                 ;MICROTEST #1' IS HUNG
042574'                                   ERRHRD  186,TLNKAD,MSG12        ;PRINT ERROR MESSAGE
```

```
042574' 104456                                                                           TRAP    C$ERHRD
042576' 000272                                                                           .WORD   186
042600' 003260'                                                                          .WORD   TLNKAD
042602' 013500'                                                                          .WORD   MSG12
042604'                                 ESCAPE TST
042604' 104410                                                                           TRAP    C$ESCAPE
042606' 000066                                                                           .WORD   L10152-.
                                ;
                                ;THE MICROCODE WILL SET PCSR1 TO AN ERROR CONDITION IF IT DETECTED AN ERROR.
                                ;SO CHECK PCSR1 IF ERROR, PRINT ERROR MESSAGE
                                ;
042610' 122777  000003  135522  45$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
042616' 001004                          BNE     50$                     ;NO
042620'                                 ERRHRD  187,TLNKAD,MSG42        ;YES, PRINT ERROR MESSAGE
042620' 104456                                                                           TRAP    C$ERHRD
042622' 000273                                                                           .WORD   187
042624' 003260'                                                                          .WORD   TLNKAD
042626' 016262'                                                                          .WORD   MSG42
042630'                         50$:
042630'                           ENDSEG
042630'                                                                          10001$:
042630' 104405                                                                           TRAP    C$ESEG
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
042632' 004737  017374'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
042636' 103006                          BCC     60$                     ;OK
042640'                                 ERRHRD  188,TLNKAD,RACMG7       ;ERROR DNI DID NOT CLEAR!
042640' 104456                                                                           TRAP    C$ERHRD
042642' 000274                                                                           .WORD   188
042644' 003260'                                                                          .WORD   TLNKAD
042646' 012726'                                                                          .WORD   RACMG7
042650'                                 ESCAPE TST
042650' 104410                                                                           TRAP    C$ESCAPE
042652' 000022                                                                           .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
                                ;
042654' 023727  000606' 174000  60$:    CMP     PCBB+2,#LINADR+<SIZ1K*15.> ;TRIED BUFFERS 1-15 FOR TRANSMITTER?
042662' 001404                          BEQ     70$                     ;YES ALL DONE
042664' 062737  004000  000606'         ADD     #SIZ1K,PCBB+2           ;NO POINT TO NEXT TRANSMIT BUFFER
042672' 000702                          BR      30$                     ;DO AGAIN
042674'                         70$:
042674'                         ENDTST
042674'                                                                          L10152:
042674' 104401                                                                           TRAP    C$ETST
```

```
.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 CONFLICKS BETWEEN PROCESSES AND ASSURES THAT HIGHER
;PRIORITY PROCESSES GET PRECEDENCE.
;
;THIS TEST WILL VERIFY THE ABLILITY 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.
;
;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 ARITRATION 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  !
                                          ;               +----------------------+
                                          ; PCBB+6:       !   DATA TRANSFERRED   !
                                          ;    !          !         FROM         !
                                          ;    !          !      LINK MEMORY     !
                                          ;    !          !        BY THE        !
                                          ;    !          !      DMA ENGINE      !
                                          ;    V          !                      !
                                          ; PCBB+4000:    +----------------------+
                                          ;
                                          ;
                                          ;TEST SEQUENCE:
                                          ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                          ;       2-LOAD PCSR2 AND PCSR3 WITH ADDRESS OF PCBB
                                          ;       3-CLEAR DMA ENGINE 'TO' BUFFER
                                          ;       4-WAIT FOR MICROMONITOR TO ENTER 'INMON' STATE
                                          ;       5-SELECT MICROTEST #8
                                          ;       6-WAIT FOR 'DNI'
                                          ;       7-CHECK FOR ERROR IN PCSR1
                                          ;       8-VERIFY DATA IN DMA 'TO' BUFFER
                                          ;       9-WRITE '1' TO CLEAR 'DNI'
                                          ;
                                          ;*************************************************************************

042676'                                   BGNTST
042676'                                                                                 T33::

                                          ;
                                          ;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.
                                          ;
042676'                                     BGNSEG
042676' 104404                                                                                  TRAP    C$BSEG
042700' 022737  000104  000326'                   CMP     #'D,MICRO            ;HAS MICROCODE MODULE 'D' BEEN LOADED
042706' 001004                                    BNE     5$                   ;NO
042710' 122777  000001  135422                    CMPB    #INMON,@PCSR1        ;YES, IS THE MICROMONITOR ACTIVE?
```

```
042716' 001440                           BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
042720' 012737  000104  000326'  5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
042726' 004737  020252'                  JSR     PC,LODMIC
042732' 103002                           BCC     10$                     ;OK
042734'                                  ESCAPE  TST
042734' 104410                                                                                   TRAP    C$ESCAPE
042736' 000376                                                                                   .WORD   L10153-.
042740' 012737  000176  000332'  10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
042746' 004737  017330'                  JSR     PC,CHKDNI
042752' 103022                           BCC     20$                     ;OK
042754' 012737  000776' 000310'          MOV     #$DNI,BITNAM
042762' 012737  001275' 000312'          MOV     #$NSET,BITSTA
042770' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
042776' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
043004'                                  ERRHRD  189,LNKARB,MSG1
043004' 104456                                                                                   TRAP    C$ERHRD
043006' 000275                                                                                   .WORD   189
043010' 003334'                                                                                  .WORD   LNKARB
043012' 012750'                                                                                  .WORD   MSG1
043014'                                  ESCAPE  TST
043014' 104410                                                                                   TRAP    C$ESCAPE
043016' 000316                                                                                   .WORD   L10153-.
043020' 004737  017374'          20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
043024' 103006                           BCC     25$
043026'                                  ERRHRD  190,LNKARB RACMG7       ;DNI DID NOT CLEAR!
043026' 104456                                                                                   TRAP    C$ERHRD
043030' 000276                                                                                   .WORD   190
043032' 003334'                                                                                  .WORD   LNKARB
043034' 012726'                                                                                  .WORD   RACMG7
043036'                                  ESCAPE TST
043036' 104410                                                                                   TRAP    C$ESCAPE
043040' 000274                                                                                   .WORD   L10153-.
043042'                          25$:
043042'                            ENDSEG
043042'                                                                                10000$:
043042' 104405                                                                                   TRAP    C$ESEG
                                 ;
                                 ;SETUP PCBB TO BE JUST AFTER LAST LOCATION USED BY THIS DIAGNOSTIC
                                 ;TELL MICROCODE WHERE PCBB IS BY LOADING PCSR2 AND PCSR3 WITH THE PCBB ADDRESS
                                 ;
043044' 013777  000324' 135270           MOV     FREMEM,@PCSR2           ;TELL MICROCODE WHERE PCBB IS
043052' 005077  135266                   CLR     @PCSR3
                                 ;
                                 ;FIRST CLEAR THE 3 WORDS THAT WILL BE USED BY THE MICROCODE IF AN ERROR OCCURS.
                                 ;THEN CLEAR OUT THE BUFFER TO BE USED AS THE DMA ENGINE'S 'TO' BUFFER.
                                 ;
043056' 013700  000324'                  MOV     FREMEM,R0               ;GET A POINTER TO PCBB
043062' 005020                           CLR     (R0)+                   ;HERE IS WHERE MICROCODE WILL PUT
                                                                         ;EXPECTED DATA PATTERN IF ERROR
043064' 005020                           CLR     (R0)+                   ;HERE, THE BAD DATA
043066' 005020                           CLR     (R0)+                   ;AND HERE THE LINK MEMORY ADDRESS

043070' 012701  002000                   MOV     #SIZ1K/2,R1             ;SIZE OF DMA ENGINE 'TO' BUFFER
043074' 005020                   30$:    CLR     (R0)+                   ;CLEAR THE AREA WHERE THE DMA ENGINE...
043076' 005301                           DEC     R1                      ;WILL LOAD DATA
043100' 001375                           BNE     30$
```

```
043102'                         35$:
043102'                           BGNSEG
043102' 104404                                                                                              TRAP    C$BSEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;SELECT MICROTEST #8 BY LOADING PCSR0 COMMAND FIELD BITS WITH A #8.
                                ;WAIT FOR 'DNI'
                                ;
043104' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
043110' 103006                          BCC     36$                     ;OK
043112'                                 ERRHRD  191,LNKARB,MSG46        ;PRINT ERROR
043112' 104456                                                                                              TRAP    C$ERHRD
043114' 000277                                                                                              .WORD   191
043116' 003334'                                                                                             .WORD   LNKARB
043120' 016700'                                                                                             .WORD   MSG46
043122'                                 ESCAPE  TST                     ;LEAVE TEST
043122' 104410                                                                                              TRAP    C$ESCAPE
043124' 000210                                                                                              .WORD   L10153-.
043126' 012777  000010  135202  36$:    MOV     #8.,@PCSR0              ;TELL T11 TO EXECUTE MICROTEST #8
043134' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
043142' 004737  017330'                 JSR     PC,CHKDN'
043146' 103021                          BCC     40$
043150' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
043154' 103006                          BCC     37$                     ;NO, OK
043156'                                 ERRHRD  192,LNKARB,MSG44        ;PRINT ERROR MESSAGE
043156' 104456                                                                                              TRAP    C$ERHRD
043160' 000300                                                                                              .WORD   192
043162' 003334'                                                                                             .WORD   LNKARB
043164' 016454'                                                                                             .WORD   MSG44
043166'                                 ESCAPE  TST
043166' 104410                                                                                              TRAP    C$ESCAPE
043170' 000144                                                                                              .WORD   L10153-.
043172' 012702  000010          37$:    MOV     #8.,R2                  ;MICROTEST #8 IS HUNG
043176'                                 ERRHRD  193,LNKARB,MSG12
043176' 104456                                                                                              TRAP    C$ERHRD
043200' 000301                                                                                              .WORD   193
043202' 003334'                                                                                             .WORD   LNKARB
043204' 013500'                                                                                             .WORD   MSG12
043206'                                 ESCAPE  TST
043206' 104410                                                                                              TRAP    C$ESCAPE
043210' 000124                                                                                              .WORD   L10153-.
                                ;
                                ;CHECK MICROCODE FINISH STATUS BY CHECKING PCSR1 STATE BITS FOR AN ERROR
                                ;IF ERROR, PRINT ERROR MESSAGE
                                ;
043212' 122777  000003  135120  40$:    CMPB    #INERR,@PCSR1           ;DID AND ERROR OCCUR?
043220' 001013                          BNE     50$                     ;NO
043222' 013701  000324'                 MOV     FREMEM,R1               ;YES, GET EXPECTED DATA PATTERN
043226' 013702  000326'                 MOV     FREMEM+2,R2             ;GET BAD DATA
043232' 013703  000330'                 MOV     FREMEM+4,R3             ;GET LINK MEMORY ADDRESS
043236'                                 ERRHRD  194,LNKARB,MSG20        ;PRINT ERROR MESSAGE
043236' 104456                                                                                              TRAP    C$ERHRD
043240' 000302                                                                                              .WORD   194
043242' 003334'                                                                                             .WORD   LNKARB
043244' 014160'                                                                                             .WORD   MSG20
043246' 000410                          BR      60$
```

```
                                        ;
                                        ;CHECK DATA THAT THE DMA ENGINE TRANSFERRED TO UNIBUS MEMORY
                                        ;
 043250' 013701  000324'                50$:    MOV     FREMEM,R1               ;GET POINTER TO DMA ENGINE BUFFER
 043254' 062701  000006                         ADD     #6,R1
 043260' 012704  003774                         MOV     #3774,R4                ;NUMBER OF BYTES
 043264' 012702  000017                         MOV     #17,R2                  ;DATA THAT SHOULD BE IN BUFFER
 043270' 111103                         60$:    MOVB    (R1),R3                 ;GET DATA FROM BUFFER
 043272' 120211                                 CMPB    R2,(R1)                 ;IS DATA CORRECT?
 043274' 001404                                 BEQ     70$                     ;YES
 043276'                                        ERRHRD  195,LNKARB,MSG21        ;NO, PRINT ERROR
 043276' 104456                                                                                 TRAP    C$ERHRD
 043300' 000303                                                                                 .WORD   195
 043302' 003334'                                                                                .WORD   LNKARB
 043304' 014250'                                                                                .WORD   MSG21
 043306' 005201                         70$:    INC     R1
 043310' 005304                                 DEC     R4
 043312' 001366                                 BNE     60$                     ;CONTINUE CHECKING
                                                                                ;ALL DONE
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI'
                                        ;
 043314' 004737  017374'                80$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
 043320' 103004                                 BCC     90$
 043322'                                        ERRHRD  196,LNKARB,RACMG7       ;ERROR DNI DID NOT CLEAR!
 043322' 104456                                                                                 TRAP    C$ERHRD
 043324' 000304                                                                                 .WORD   196
 043326' 003334'                                                                                .WORD   LNKARB
 043330' 012726'                                                                                .WORD   RACMG7
 043332'                                90$:
 043332'                                  ENDSEG
 043332'                                                                                 10001$:
 043332' 104405                                                                                 TRAP    C$ESEG
 043334'                                ENDTST
 043334'                                                                                L10153:
 043334' 104401                                                                                 TRAP    C$ETST
```

```
.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 THE
;TEST WAS SUCCESSFUL
;
;THE PCBB WILL BE USED TO PASS THE 48 BIT STATION ADDRESS PATTERN:
;
;                    +--------------------------------+
;PCBB+0:             !      STATION ADDRESS LOW       !
;                    +--------------------------------+
;PCBB+2:             !   STATION ADDRESS MIDDLE       !
;                    +--------------------------------+
;PCBB+4:             !      STATION ADDRESS HIGH      !
;                    +--------------------------------+
;
;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 MICRMODULE '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
```

```
                                          ;       7-IF NO DNI AND PCSR1 INDICATES NO ERROR THEN PRINT MATCH BIT SET
                                          ;         BUT DATAGRAM WAS NOT RECEIVED.
                                          ;       8-WRITE '1' TO CLEAR 'DNI'
                                          ;       9-REPEAT STEPS 2-8 FOR ALL SEVEN DATA PATTERNS
                                          ;
                                          ;**************************************************************************

043336'                                   BGNTST
043336'                                                                              T34::
                                          ;
                                          ;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.
                                          ;
043336'                                     BGNSEG
043336' 104404                                                                           TRAP    C$BSEG
043340' 022737  000105  000326'                  CMP     #'E,MICRO            ;HAS MICROCODE MODULE 'E' BEEN LOADED
043346' 001004                                   BNE     5$                   ;NO
043350' 122777  000001  134762                   CMPB    #INMON,@PCSR1        ;YES, IS THE MICROMONITOR ACTIVE?
043356' 001440                                   BEQ     20$                  ;YES SKIP LOADING THE MICROMODULE
043360' 012737  000105  000326' 5$:              MOV     #'E,MICRO            ;GO LOAD MICRO MODULE 'E'
043366' 004737  020252'                          JSR     PC,LODMIC
043372' 103002                                   BCC     10$                  ;OK
043374'                                          ESCAPE  TST
043374' 104410                                                                           TRAP    C$ESCAPE
043376' 000266                                                                           .WORD   L10154-.
043400' 012737  000176  000332' 10$:             MOV     #2*SECOND,METER      ;WAIT FOR THE MICROMONITOR
043406' 004737  017330'                          JSR     PC,CHKDNI
043412' 103022                                   BCC     20$                  ;OK
043414' 012737  000776' 000310'                  MOV     #$DNI,BITNAM
043422' 012737  001275' 000312'                  MOV     #$NSET,BITSTA
043430' 012737  001340' 000314'                  MOV     #$AFTER,PWHEN
043436' 012737  001355' 000316'                  MOV     #$GTCMD,PCOMND
043444'                                          ERRHRD  197,STAPAT,MSG1
043444' 104456                                                                           TRAP    C$ERHRD
043446' 000305                                                                           .WORD   197
043450' 003400'                                                                          .WORD   STAPAT
043452' 012750'                                                                          .WORD   MSG1
043454'                                          ESCAPE  TST
043454' 104410                                                                           TRAP    C$ESCAPE
043456' 000206                                                                           .WORD   L10154-.
043460' 004737  017374'         20$:             JSR     PC,CLRDNI            ;CLEAR DNI BIT
043464' 103006                                   BCC     25$
043466'                                          ERRHRD  198,STAPAT,RACMG7    ;DNI DID NOT CLEAR!
043466' 104456                                                                           TRAP    C$ERHRD
043470' 000306                                                                           .WORD   198
043472' 003400'                                                                          .WORD   STAPAT
043474' 012726'                                                                          .WORD   RACMG7
043476'                                          ESCAPE  TST
043476' 104410                                                                           TRAP    C$ESCAPE
043500' 000164                                                                           .WORD   L10154-.
043502'                         25$:
043502'                                     ENDSEG
043502'                                                                          10000$:
043502' 104405                                                                           TRAP    C$ESEG
                                          ;
```

```
                                   ;SELECT ONE OF THE PATTERNS FROM THE STATION ADDRESS PATTERN TABLE AND LOAD
                                   ;IT INTO THE PCBB
                                   ;
043504' 012701  000532'                    MOV     #SPAT1,R1               ;POINT TO STATION ADDRESS PATTERN TABLE
043510' 012705  000007                     MOV     #7,R5                   ;# OF ADDRESS PATTERNS
043514' 011137  000604'            27$:    MOV     (R1),PCBB               ;LOAD PCBB WITH AN ADDRESS PATTERN
043520' 016137  000002  000606'            MOV     2(R1),PCBB+2            ;MIDDLE ORDER
043526' 016137  000004  000610'            MOV     4(R1),PCBB+4            ;UPPER ORDER
043534'                              BGNSEG
043534' 104404                                                                                     TRAP    C$BSEG
                                   ;
                                   ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE. SELECT MICROTEST #1
                                   ;BY LOADING PCSR0 COMMAND FIELD BITS WITH A 1. WAIT FOR DNI.
                                   ;
043536' 004737  017776'                    JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
043542' 103006                             BCC     30$                     ;OK
043544'                                    ERRHRD  199,STAPAT,MSG46        ;PRINT ERROR
043544' 104456                                                                                     TRAP    C$ERHRD
043546' 000307                                                                                     .WORD   199
043550' 003400'                                                                                    .WORD   STAPAT
043552' 016700'                                                                                    .WORD   MSG46
043554'                                    ESCAPE  TST                     ;LEAVE TEST
043554' 104410                                                                                     TRAP    C$ESCAPE
043556' 000106                                                                                     .WORD   L10154-.
043560' 012777  000001  134550     30$:    MOV     #1,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #1
043566' 012737  000176  000332'            MOV     #2*SECOND,METER         ;WAIT FOR DNI
043574' 004737  017330'                    JSR     PC,CHKDNI
043600' 103015                             BCC     40$
                                   ;
                                   ;DNI DID NOT SET WHICH INDICATES THAT THE RECEIVE  INTERRUPT NEVER HAPPENED.
                                   ;THE REASON MIGHT BE BECAUSE THE MATCH BIT NEVER SET IN THE TRANSMIT
                                   ;STATUS REGISTER INDICATING THAT THE STATION ADDRESS COMPARATOR FAILED
                                   ;TO RECOGNIZE THE DATAGRAM DESTINATION ADDRESS. THE MICROCODE CHECKS
                                   ;FOR THE MATCH BIT AND IF IT IS NOT SET IT PUTS AN ERROR STATUS IN PCSR1.
                                   ;IF PCSR1 DOES NOT HAVE AN ERROR STATE THEN THE RECEIVER INTERRUPT FAILED
                                   ;FOR SOME OTHER REASON AND ITS NOT THE STATION ADDRESS RECOGNITION'S FAULT.
                                   ;
043602' 004737  020050'                    JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
043606' 103006                             BCC     35$                     ;NO
043610'                                    ERRHRD  200,STAPAT,MSG44        ;PRINT ERROR MESSAGE
043610' 104456                                                                                     TRAP    C$ERHRD
043612' 000310                                                                                     .WORD   200
043614' 003400'                                                                                    .WORD   STAPAT
043616' 016454'                                                                                    .WORD   MSG44
043620'                                    ESCAPE  TST                     ;LEAVE TST
043620' 104410                                                                                     TRAP    C$ESCAPE
043622' 000042                                                                                     .WORD   L10154-.
043624'                            35$:    ERRHRD  201,STAPAT,MSG25        ;STATION ADDRESS PATTERN WAS NOT
043624' 104456                                                                                     TRAP    C$ERHRD
043626' 000311                                                                                     .WORD   201
043630' 003400'                                                                                    .WORD   STAPAT
043632' 014626'                                                                                    .WORD   MSG25
                                                                           ;RECOGNIZED! ERROR!
                                   ;
                                   ;WRITE '1' TO CLEAR 'DNI'
                                   ;
```

```
043634' 004737  017374'         40$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
043640' 103004                          BCC     50$
043642'                                 ERRHRD  202,STAPAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
043642' 104456                                                                                  TRAP    C$ERHRD
043644' 000312                                                                                  .WORD   202
043646' 003400'                                                                                 .WORD   STAPAT
043650' 012726'                                                                                 .WORD   RACMG7
043652'                         50$:
043652'                           ENDSEG
043652'                                                                                 10001$:
043652' 104405                                                                                  TRAP    C$ESEG

                                ;
                                ;REPEAT THE TEST WITH ALL SEVEN PATTERNS
                                ;
043654' 062701  000006                  ADD     #6,R1                   ;POINT TO NEXT PATTERN
043660' 005305                          DEC     R5                      ;TESTED WITH ALL ADDRESS PATTERNS?
043662' 001314                          BNE     27$                     ;NOT YET

043664'                         ENDTST
043664'                                                                                 L10154:
043664' 104401                                                                                  TRAP    C$ETST
```

```
                                        .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 MICROTEST #2
                                        ;       4-WAIT FOR 'DNI'
                                        ;       5-CHECK PCSR1 FOR AN ERROR CONDITION
                                        ;       6-WRITE '1' TO CLEAR 'DNI'
                                        ;
                                        ;*************************************************************************

 043666'                                BGNTST
 043666'                                                                                  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.
                                        ;
 043666'                                 BGNSEG
 043666' 104404                                                                                 TRAP    C$BSEG
 043670' 022737  000105  000326'               CMP     #'E,MICRO               ;HAS MICROCODE MODULE 'E' BEEN LOADED
 043676' 001004                                BNE     5$                      ;NO
 043700' 122777  000001  134432                CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 043706' 001440                                BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 043710' 012737  000105  000326' 5$:           MOV     #'E,MICRO               ;GO LOAD MICRO MODULE 'E'
 043716' 004737  020252'                       JSR     PC,LODMIC
 043722' 103002                                BCC     10$                     ;OK
 043724'                                       ESCAPE  TST
 043724' 104410                                                                                 TRAP    C$ESCAPE
 043726' 000256                                                                                 .WORD   L10155-.
 043730' 012737  000176  000332' 10$:          MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
 043736' 004737  017330'                       JSR     PC,CHKDNI
 043742' 103022                                BCC     20$                     ;OK
 043744' 012737  000776' 000310'               MOV     #$DNI,BITNAM
 043752' 012737  001275' 000312'               MOV     #$NSET,BITSTA
 043760' 012737  001340' 000314'               MOV     #$AFTER,PWHEN
 043766' 012737  001355' 000316'               MOV     #$GTCMD,PCOMND
 043774'                                       ERRHRD  203,STAREJ,MSG1
```

```
043774' 104456                                                                                                          TRAP      C$ERHRD
043776' 000313                                                                                                          .WORD     203
044000' 003444'                                                                                                         .WORD     STAREJ
044002' 012750'                                                                                                         .WORD     MSG1
044004'                                           ESCAPE    TST
044004' 104410                                                                                                          TRAP      C$ESCAPE
044006' 000176                                                                                                          .WORD     L10155-.
044010' 004737    017374'           20$:          JSR       PC,CLRDNI                     ;CLEAR DNI BIT
044014' 103006                                    BCC       25$
044016'                                           ERRHRD    204,STAREJ,RACMG7             ;DNI DID NOT CLEAR!
044016' 104456                                                                                                          TRAP      C$ERHRD
044020' 000314                                                                                                          .WORD     204
044022' 003444'                                                                                                         .WORD     STAREJ
044024' 012726'                                                                                                         .WORD     RACMG7
044026'                                           ESCAPE    TST
044026' 104410                                                                                                          TRAP      C$ESCAPE
044030' 000154                                                                                                          .WORD     L10155-.
044032'                             25$:
044032'                               ENDSEG
044032'                                                                                                       10000$:
044032' 104405                                                                                                          TRAP      C$ESEG

                                    ;
                                    ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, SELECT MICROTEST #2
                                    ;BY LOADING PCSRO WITH A 2
                                    ;
044034'                               BGNSEG
044034' 104404                                                                                                          TRAP      C$BSEG
044036' 004737    017776'                         JSR       PC,CHKMON                     ;WAIT FOR MICROMONITOR
044042' 103006                                    BCC       30$                           ;OK
044044'                                           ERRHRD    205,STAREJ,MSG46              ;PRINT ERROR
044044' 104456                                                                                                          TRAP      C$ERHRD
044046' 000315                                                                                                          .WORD     205
044050' 003444'                                                                                                         .WORD     STAREJ
044052' 016700'                                                                                                         .WORD     MSG46
044054'                                           ESCAPE    TST                           ;LEAVE TEST
044054' 104410                                                                                                          TRAP      C$ESCAPE
044056' 000126                                                                                                          .WORD     L10155-.
044060' 012737    000275    000332' 30$:          MOV       #3*SECOND,METER               ;TIMEOUT MUST BE GREATER THAN TIMER
                                                                                        ;PERIOD IN THE MICROCODE.
044066' 012777    000002    134242                MOV       #2,@PCSRO                     ;TELL T11 TO EXECUTE MICROTEST #2
044074' 004737    017330'                         JSR       PC,CHKDNI                     ;WAIT FOR DNI

044100' 103021                                    BCC       40$
044102' 004737    020050'                         JSR       PC,CHKINT                     ;SEE IF ANY ERROR INTERRUPTS OCCURRED
044106' 103006                                    BCC       35$                           ;NO, OK
044110'                                           ERRHRD    206,STAREJ,MSG44              ;PRINT ERROR MESSAGE
044110' 104456                                                                                                          TRAP      C$ERHRD
044112' 000316                                                                                                          .WORD     206
044114' 003444'                                                                                                         .WORD     STAREJ
044116' 016454'                                                                                                         .WORD     MSG44
044120'                                           ESCAPE    TST
044120' 104410                                                                                                          TRAP      C$ESCAPE
044122' 000062                                                                                                          .WORD     L10155-.
044124' 012702    000002            35$:          MOV       #2,R2                         ;MICROTEST #2 IS HUNG
044130'                                           ERRHRD    207,STAREJ,MSG12
044130' 104456                                                                                                          TRAP      C$ERHRD
```

```
044132' 000317                                                                                  .WORD   207
044134' 003444'                                                                                 .WORD   STAREJ
044136' 013500'                                                                                 .WORD   MSG12
044140'                                 ESCAPE TST
044140' 104410                                                                                  TRAP    C$ESCAPE
044142' 000042                                                                                  .WORD   L10155-.
                                ;
                                ;OK 'DNI' SET INDICATING EITHER THE RECEIVER INTERRUPT HAPPENED OR THE TIMER
                                ;BROKE THE LOOP, CHECK PCSR1 TO FIND OUT WHICH
                                ;
044144' 122777  000003  134166  40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
044152' 001004                          BNE     50$                     ;NO
044154'                                 ERRHRD  208,STAREJ              ;YES, PRINT ERROR MESSAGE
044154' 104456                                                                                  TRAP    C$ERHRD
044156' 000320                                                                                  .WORD   208
044160' 003444'                                                                                 .WORD   STAREJ
044162' 000000                                                                                  .WORD   0
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
044164' 004737  017374'         50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
044170' 103004                          BCC     60$
044172'                                 ERRHRD  209,STAREJ,RACMG7       ;ERROR DNI DID NOT CLEAR!
044172' 104456                                                                                  TRAP    C$ERHRD
044174' 000321                                                                                  .WORD   209
044176' 003444'                                                                                 .WORD   STAREJ
044200' 012726'                                                                                 .WORD   RACMG7
044202'                         60$:
044202'                           ENDSEG
044202'                                                                                 10001$:
044202' 104405                                                                                  TRAP    C$ESEG
044204'                         ENDTST
044204'                                                                                 L10155:
044204' 104401                                                                                  TRAP    C$ETST
```

```
                                        .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
                                        ;
                                        ;***************************************************************************

 044206'                                BGNTST
 044206'                                                                               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.
                                        ;
 044206'                                  BGNSEG
 044206' 104404                                                                              TRAP    C$BSEG
 044210' 022737  000105  000326'               CMP     #'E,MICRO            ;HAS MICROCODE MODULE 'E' BEEN LOADED
 044216' 001004                                BNE     5$                   ;NO
 044220' 122777  000001  134112                CMPB    #INMON,@PCSR1        ;YES, IS THE MICROMONITOR ACTIVE?
 044226' 001440                                BEQ     20$                  ;YES SKIP LOADING THE MICROMODULE
```

```
044230' 012737  000105  000326' 5$:     MOV     #'E,MICRO               ;GO LOAD MICRO MODULE 'E'
044236' 004737  020252'                 JSR     PC,LODMIC
044242' 103002                          BCC     10$                     ;OK
044244                                  ESCAPE  TST
044244' 104410                                                                                  TRAP    C$ESCAPE
044246' 000306                                                                                  .WORD   L10156-.
044250' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
044256' 004737  017330'                 JSR     PC,CHKDNI
044262' 103022                          BCC     20$                     ;OK
044264' 012737  000776' 000310'         MOV     #$DNI,BITNAM
044272' 012737  001275' 000312'         MOV     #$NSET,BITSTA
044300' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
044306' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
044314'                                 ERRHRD  210,STAPOS,MSG1
044314' 104456                                                                                  TRAP    C$ERHRD
044316' 000322                                                                                  .WORD   210
044320' 003512'                                                                                 .WORD   STAPOS
044322' 012750'                                                                                 .WORD   MSG1
044324'                                 ESCAPE  TST
044324' 104410                                                                                  TRAP    C$ESCAPE
044326' 000226                                                                                  .WORD   L10156-.
044330' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
044334' 103006                          BCC     25$
044336'                                 ERRHRD  211,STAPOS,RACMG7       ;DNI DID NOT CLEAR!
044336' 104456                                                                                  TRAP    C$ERHRD
044340' 000323                                                                                  .WORD   211
044342' 003512'                                                                                 .WORD   STAPOS
044344' 012725'                                                                                 .WORD   RACMG7
044346'                                 ESCAPE TST
044346' 104410                                                                                  TRAP    C$ESCAPE
044350' 000204                                                                                  .WORD   L10156-.
044352'                         25$:
044352'    :                      ENDSEG
044352'                                                                         10000$:
044352' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;R1 WILL CONTAIN THE RAM POSTION TO BE TESTED
                                ;
044354' 012701  000001                  MOV     #1,R1                   ;BEGIN WITH RAM POSTION #1
044360'                         30$:
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, LOAD PCBB WITH
                                ;THE RAM POSTION TO BE TESTED, SELECT MICROTEST #3 BY LOADING PCSR0 COMMAND
                                ;FIELD WITH A 3, AND WAIT FOR 'DNI' TO SET.
                                ;
044360'                           BGNSEG
044360' 104404                                                                                  TRAP    C$BSEG
044362' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
044366' 103006                          BCC     35$                     ;OK
044370'                                 ERRHRD  212,STAPOS,MSG46        ;PRINT ERROR
044370' 104456                                                                                  TRAP    C$ERHRD
044372' 000324                                                                                  .WORD   212
044374' 003512'                                                                                 .WORD   STAPOS
044376' 016700'                                                                                 .WORD   MSG46
044400'                                 ESCAPE  TST                     ;LEAVE TEST
044400' 104410                                                                                  TRAP    C$ESCAPE
```

```
044402' 000152                                                                              .WORD   L10156-.
044404' 010137  000604'          35$:    MOV     R1,PCBB                 ;TELL MICROCODE WHICH RAM POSTION TO USE
044410' 012777  000003  133720           MOV     #3,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #3
044416' 012737  000275  000332'          MOV     #3*SECOND,METER         ;WAIT FOR DNI
044424' 004737  017330'                  JSR     PC,CHKDNI
044430' 103021                           BCC     40$
044432' 004737  020050'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
044436' 103006                           BCC     36$                     ;NO, OK
044440'                                  ERRHRD  213,STAPOS,MSG44        ;PRINT ERROR MESSAGE
044440' 104456                                                                              TRAP    C$ERHRD
044442' 000325                                                                              .WORD   213
044444' 003512'                                                                             .WORD   STAPOS
044446' 016454'                                                                             .WORD   MSG44
044450'                                  ESCAPE  TST
044450' 104410                                                                              TRAP    C$ESCAPE
044452' 000102                                                                              .WORD   L10156-.
044454' 012702  000003           36$:    MOV     #3,R2                   ;MICROTEST #3 IS HUNG
044460'                                  ERRHRD  214,STAPOS,MSG12
044460' 104456                                                                              TRAP    C$ERHRD
044462' 000326                                                                              .WORD   214
044464' 003512'                                                                             .WORD   STAPOS
044466' 013500'                                                                             .WORD   MSG12
044470'                                  ESCAPE  TST
044470' 104410                                                                              TRAP    C$ESCAPE
044472' 000062                                                                              .WORD   L10156-.
                                 ;
                                 ;CHECK PCSR1 TO SEE IF THE DATAGRAM WAS RECEIVED
                                 ;
044474' 122777  000003  133636   40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
044502' 001004                           BNE     45$                     ;NO
044504'                                  ERRHRD  215,STAPOS,MSG26        ;PRINT ERROR MESSAGE
044504' 104456                                                                              TRAP    C$ERHRD
044506' 000327                                                                              .WORD   215
044510' 003512'                                                                             .WORD   STAPOS
044512' 015014'                                                                             .WORD   MSG26
044514'                          45$:
044514'                            ENDSEG
044514'                                                                             10001$:
044514' 104405                                                                              TRAP    C$ESEG
                                 ;
                                 ;WRITE '1' TO CLEAR 'DNI'
                                 ;
044516'                            BGNSEG
044516' 104404                                                                              TRAP    C$BSEG
044520' 004737  017374'                  JSR     PC,CLRDNI               ;GO CLEAR DNI
044524' 103006                           BCC     50$                     ;OK
044526'                                  ERRHRD  216,STAPOS,RACMG7       ;ERROR OCCURRED PRINT MESSAGE
044526' 104456                                                                              TRAP    C$ERHRD
044530' 000330                                                                              .WORD   216
044532' 003512'                                                                             .WORD   STAPOS
044534' 012726'                                                                             .WORD   RACMG7
044536'                                  ESCAPE  TST
044536' 104410                                                                              TRAP    C$ESCAPE
044540' 000014                                                                              .WORD   L10156-.
044542'                          50$:
044542'                            ENDSEG
```

```
 10740  044542'                                                                                     10002$:
 10741  044542' 104405                                                                                      TRAP    C$ESEG
 10742                                             ;
 10743                                             ;REPEAT TEST FOR ALL 12 RAM POSTIONS
 10744                                             ;
 10745  044544' 005201                                     INC     R1                          ;DO NEXT RAM POSTION
 10746  044546' 020127  000014                             CMP     R1,#12.                     ;HAVE WE DONE ALL 12 RAM POSTIONS?
 10747  044552' 103702                                     BLO     30$                         ;NO KEEP GOING
 10748  044554'                                    ENDTST
 10749  044554'                                                                                     L10156:
 10750  044554' 104401                                                                                      TRAP    C$ETST
```

```
                                        .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'
                                        ;
                                        ;*************************************************************************

 044556'                                BGNTST
 044556'                                                                      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' TO CLEAR IT.
                                        ;
 044556'                                  BGNSEG
 044556' 104404                                                                         TRAP    C$BSEG
 044560' 022737  000105  000326'               CMP     #'E,MICRO             ;HAS MICROCODE MODULE 'E' BEEN LOADED
 044566' 001004                                BNE     5$                    ;NO
 044570' 122777  000001  133542                CMPB    #INMON,@PCSR1         ;YES, IS THE MICROMONITOR ACTIVE?
 044576' 001440                                BEQ     20$                   ;YES SKIP LOADING THE MICROMODULE
 044600' 012737  000105  000326' 5$:           MOV     #'E,MICRO             ;GO LOAD MICRO MODULE 'E'
 044606  004737  020252'                       JSR     PC,LODMIC
 044612' 103002                                BCC     10$                   ;OK
 044614'                                       ESCAPE  TST
 044614' 104410                                                                         TRAP    C$ESCAPE
 044616' 000256                                                                         .WORD   L10157-.
 044620' 012737  000176  000332' 10$:          MOV     #2*SECOND,METER       ;WAIT FOR THE MICROMONITOR
 044626' 004737  017330'                       JSR     PC,CHKDNI
 044632' 103022                                BCC     20$                   ;OK
 044634' 012737  000776' 000310'               MOV     #$DNI,BITNAM
 044642' 012737  001275' 000312'               MOV     #$NSET,BITSTA
 044650' 012737  001340' 000314'               MOV     #$AFTER,PWHEN
```

```
044656' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
044664'                                  ERRHRD  217,MJCAST,MSG1
044664' 104456                                                                                   TRAP    C$ERHRD
044666' 000331                                                                                   .WORD   217
044670' 003563'                                                                                  .WORD   MUCAST
044672' 012750'                                                                                  .WORD   MSG1
044674'                                  ESCAPE  TST
044674' 104410                                                                                   TRAP    C$ESCAPE
044676' 000176                                                                                   .WORD   L10157-.
044700' 004737  017374'          20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
044704' 103006                           BCC     25$
044706'                                  ERRHRD  218,MUCAST,RACMG7       ;DNI DID NOT CLEAR!
044706' 104456                                                                                   TRAP    C$ERHRD
044710' 000332                                                                                   .WORD   218
044712' 003563'                                                                                  .WORD   MUCAST
044714' 012726'                                                                                  .WORD   RACMG7
044716'                                  ESCAPE  TST
044716' 104410                                                                                   TRAP    C$ESCAPE
044720' 000154                                                                                   .WORD   L10157-.
044722'                          25$:
044722'                            ENDSEG
044722'                                                                                  10000$:
044722' 104405                                                                                   TRAP    C$ESEG

                                 ;
                                 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                 ;
044724'                            BGNSEG
044724' 104404                                                                                   TRAP    C$BSEG
044726' 004737  017776'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
044732' 103006                           BCC     30$                     ;OK
044734'                                  ERRHRD  219,MUCAST,MSG46        ;PRINT ERROR
044734' 104456                                                                                   TRAP    C$ERHRD
044736' 000333                                                                                   .WORD   219
044740' 003563'                                                                                  .WORD   MUCAST
044742' 016700'                                                                                  .WORD   MSG46
044744'                                  ESCAPE  TST                     ;LEAVE TEST
044744' 104410                                                                                   TRAP    C$ESCAPE
044746' 000126                                                                                   .WORD   L10157-.
044750'                          30$:
                                 ;
                                 ;EXECUTE MICROTEST #4 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 4
                                 ;
044750' 012777  000004  133360           MOV     #4,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #4
                                 ;
                                 ;WAIT AT LEAST 3 SECONDS FOR TEST TO FINISH BECAUSE MICROCODE WAITS 2 SECONDS
                                 ;FOR THE RECEIVER INTERRUPT
                                 ;
044756' 012737  000275  000332'          MOV     #3*SECOND,METER         ;WAIT FOR DNI
044764' 004737  017330'                  JSR     PC,CHKDNI
044770' 103021                           BCC     40$
044772' 004737  020050'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
044776' 103006                           BCC     35$                     ;NO, OK
045000'                                  ERRHRD  220,MUCAST,MSG44        ;PRINT ERROR MESSAGE
045000' 104456                                                                                   TRAP    C$ERHRD
045002' 000334                                                                                   .WORD   220
045004' 003563'                                                                                  .WORD   MUCAST
```

```
045006' 016454'                                                                   .WORD   MSG44
045010'                             ESCAPE  TST
045010' 104410                                                                    TRAP    C$ESCAPE
045012' 000062                                                                    .WORD   L10157-.
045014' 012702  000004      35$:    MOV     #4,R2               ;MICROTEST #4 IS HUNG
045020'                             ERRHRD  221,MUCAST,MSG12
045020' 104456                                                                    TRAP    C$ERHRD
045022' 000335                                                                    .WORD   221
045024' 003563'                                                                   .WORD   MUCAST
045026' 013500'                                                                   .WORD   MSG12
045030'                             ESCAPE TST
045030' 104410                                                                    TRAP    C$ESCAPE
045032' 000042                                                                    .WORD   L10157-.
                            ;
                            ;OK, EITHER THE TIMER BROKE THE LOOP OR THE RECEIVER INTERRUPT DID.
                            ;WHICH WAS IT?
                            ;
045034' 122777  000003  133276  40$:    CMPB    #INERR,@PCSR1       ;DID AN ERROR OCCUR?
045042' 001004                          BNE     50$                 ;NO
045044'                                 ERRHRD  222,MUCAST          ;YES, PRINT ERROR MESSAGE
045044' 104456                                                                    TRAP    C$ERHRD
045046' 000336                                                                    .WORD   222
045050' 003563'                                                                   .WORD   MUCAST
045052' 000000                                                                    .WORD   0
                            ;
                            ;WRITE '1' TO CLEAR 'DNI'
                            ;
045054' 004737  017374'     50$:    JSR     PC,CLRDNI           ;CLEAR DNI BIT
045060' 103004                      BCC     60$
045062'                             ERRHRD  223,MUCAST,RACMG7   ;ERROR DNI DID NOT CLEAR!
045062' 104456                                                                    TRAP    C$ERHRD
045064' 000337                                                                    .WORD   223
045066' 003563'                                                                   .WORD   MUCAST
045070' 012726'                                                                   .WORD   RACMG7
045072'                     60$:
045072'                       ENDSEG
045072'                                                                 10001$:
045072' 104405                                                                    TRAP    C$ESEG
045074'                     ENDTST
045074'                                                                 L10157:
045074' 104401                                                                    TRAP    C$ETST
```

```
                        .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

045076'                 BGNTST
045076'                                                                 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.
```

```
                                  ;
045076'                            BGNSEG
045076' 104404                                                                      TRAP    C$BSEG
045100' 022737  000106  000326           CMP     #'F,MICRO              ;HAS MICROCODE MODULE 'F' BEEN LOADED
045106' 001004                           BNE     5$                     ;NO
045110' 122777  000001  133222           CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
045116' 001440                           BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
045120' 012737  000106  000326' 5$:      MOV     #'F,MICRO              ;GO LOAD MICRO MODULE 'F'
045126' 004737  020252'                  JSR     PC,LODMIC
045132' 103002                           BCC     10$                    ;OK
045134'                                  ESCAPE  TST
045134' 104410                                                                      TRAP    C$ESCAPE
045136' 000276                                                                      .WORD   L10160-.
045140' 012737  000176  000332' 10$:     MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
045146' 004737  017330'                  JSR     PC,CHKDNI
045152' 103022                           BCC     20$                    ;OK
045154' 012737  000776' 000310'          MOV     #$DNI,BITNAM
045162' 012737  001275' 000312'          MOV     #$NSET,BITSTA
045170' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
045176' 012737  001355' 000316'          MOV     #$GTCMD,PCOMND
045204'                                  ERRHRD  224,CRCDAT,MSG1
045204' 104456                                                                      TRAP    C$ERHRD
045206' 000340                                                                      .WORD   224
045210' 003621'                                                                     .WORD   CRCDAT
045212' 012750'                                                                     .WORD   MSG1
045214'                                  ESCAPE  TST
045214' 104410                                                                      TRAP    C$ESCAPE
045216' 000216                                                                      .WORD   L10160-.
045220' 004737  017374'         20$:     JSR     PC,CLRDNI              ;CLEAR DNI BIT
045224' 103006                           BCC     25$
045226'                                  ERRHRD  225,CRCDAT,RACMG7      ;DNI DID NOT CLEAR!
045226' 104456                                                                      TRAP    C$ERHRD
045230' 000341                                                                      .WORD   225
045232' 003621'                                                                     .WORD   CRCDAT
045234' 012726'                                                                     .WORD   RACMG7
045236'                                  ESCAPE TST
045236' 104410                                                                      TRAP    C$ESCAPE
045240' 000174                                                                      .WORD   L10160-.
045242'                         25$:
045242'                          ENDSEG
045242'                                                                   10000$:
045242' 104405                                                                      TRAP    C$ESEG
                                ;
                                ;SELECT A DATA PATTERN FROM A TABLE OF PREDEFINED DATA PATTERNS
                                ;AND LOAD IT INTO PCBB+0 FOR THE MICROCODE
                                ;
045244' 012701  000516'                  MOV     #PATERN,R1             ;GET ADDRESS OF DATA PATTERN TABLE
045250' 012705  000005                   MOV     #5,R5                  ;# OF DATA PATTERNS
045254' 012137  000604'         27$:     MOV     (R1)+,PCBB             ;LOAD PCBB WITH A DATA PATTERN
045260'                          BGNSEG
045260' 104404                                                                      TRAP    C$BSEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;
045262' 004737  017776'                  JSR     PC,CHKMON              ;WAIT FOR MICROMONITOR
045266' 103006                           BCC     30$                    ;OK
```

```
045270'                                   ERRHRD  227,CRCDAT,MSG46        ;PRINT ERROR
045270' 104456                                                                                      TRAP    C$ERHRD
045272' 000343                                                                                      .WORD   227
045274' 003621'                                                                                     .WORD   CRCDAT
045276' 016700'                                                                                     .WORD   MSG46
045300'                                   ESCAPE  TST                     ;LEAVE TEST
045300' 104410                                                                                      TRAP    C$ESCAPE
045302' 000132                                                                                      .WORD   L10160-.
045304'                           30$:
                                  ;
                                  ;EXECUTE MICROTEST #1 BY LOADING PCSR0 COMMAND FIELD WITH A 1
                                  ;WAIT FOR DNI
                                  ;
045304' 012777  000001  133024            MOV     #1,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #1
045312' 012737  000176  000332'           MOV     #2*SECOND,METER         ;WAIT FOR DNI
045320' 004737  017330'                   JSR     PC,CHKDNI
045324' 103021                            BCC     40$
045326' 004737  020050'                   JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
045332' 103006                            BCC     35$                     ;NO, OK
045334'                                   ERRHRD  228,CRCDAT,MSG44        ;PRINT ERROR MESSAGE
045334' 104456                                                                                      TRAP    C$ERHRD
045336' 000344                                                                                      .WORD   228
045340' 003621'                                                                                     .WORD   CRCDAT
045342' 016454'                                                                                     .WORD   MSG44
045344'                                   ESCAPE  TST
045344' 104410                                                                                      TRAP    C$ESCAPE
045346' 000066                                                                                      .WORD   L10160-.
045350' 012702  000005            35$:    MOV     #5,R2                   ;MICROTEST #1 IS HUNG
045354'                                   ERRHRD  229,CRCDAT,MSG12
045354' 104456                                                                                      TRAP    C$ERHRD
045356' 000345                                                                                      .WORD   229
045360' 003621'                                                                                     .WORD   CRCDAT
045362' 013500'                                                                                     .WORD   MSG12
045364'                                   ESCAPE  TST
045364' 104410                                                                                      TRAP    C$ESCAPE
045366' 000046                                                                                      .WORD   L10160-.
                                  ;
                                  ;CHECK PCSR1 TO SEE IF THE MICROCODE DETECTED AN ERROR IN THE CRC CALCULATION
                                  ;
045370' 122777  000003  132742    40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
045376' 001004                            BNE     50$                     ;NO
045400'                                   ERRHRD  230,CRCDAT              ;PRINT ERROR MESSAGE
045400' 104456                                                                                      TRAP    C$ERHRD
045402' 000346                                                                                      .WORD   230
045404' 003621'                                                                                     .WORD   CRCDAT
045406' 000000                                                                                      .WORD   0
                                  ;
                                  ;WRITE '1' TO CLEAR 'DNI'
                                  ;
045410' 004737  017374'           50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
045414' 103004                            BCC     55$
045416'                                   ERRHRD  231,CRCDAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
045416' 104456                                                                                      TRAP    C$ERHRD
045420' 000347                                                                                      .WORD   231
045422' 003621'                                                                                     .WORD   CRCDAT
045424' 012726'                                                                                     .WORD   RACMG7
```

```
11072
11073  045426'                          55$:
11074  045426'                            ENDSEG
11075  045426'                                                                               10001$:
11076  045426' 104405                                                                                TRAP    C$ESEG
11077                                   ;
11078                                   ;REPEAT THE TEST EACH TIME WITH A NEW DATA PATTERN
11079                                   ;
11080  045430' 005305                          DEC     R5                        ;HAVE WE TESTED WITH ALL DATA PATTERNS?
11081  045432' 001310                          BNE     27$                       ;NOT YET
11082  045434'                          ENDTST
11083  045434'                                                                               L10160:
11084  045434' 104401                                                                                TRAP    C$ETST
```

```
                                 .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 DEUNA 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'
                                 ;
                                 ;****************************************************************************

045436'                          BGNTST
045436'                                                                                  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.
                                 ;
045436'                            BGNSEG
045436' 104404                                                                                   TRAP    C$BSEG
045440' 022737  000106  000326'         CMP      #'F,MICRO               ;HAS MICROCODE MODULE 'F' BEEN LOADED
045446' 001004                          BNE      5$                      ;NO
045450' 122777  000001  132662          CMPB     #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
045456' 001440                          BEQ      20$                     ;YES SKIP LOADING THE MICROMODULE
045460' 012737  000106  000326' 5$:     MOV      #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
045466' 004737  020252'                 JSR      PC,LODMIC
045472' 103002                          BCC      10$                     ;OK
045474'                                 ESCAPE   TST
045474' 104410                                                                                   TRAP    C$ESCAPE
045476' 000326                                                                                   .WORD   L10161-.
```

```
045500' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
045506' 004737  017330'                 JSR     PC,CHKDNI
045512' 103022                          BCC     20$                     ;OK
045514' 012737  000776' 000310'         MOV     #$DNI,BITNAM
045522' 012737  001275' 000312'         MOV     #$NSET,BITSTA
045530' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
045536' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
045544'                                 ERRHRD  232,CRCERR,MSG1
045544' 104456                                                                                  TRAP    C$ERHRD
045546' 000350                                                                                  .WORD   232
045550' 003656'                                                                                 .WORD   CRCERR
045552' 012750'                                                                                 .WORD   MSG1
045554'                                 ESCAPE  TST
045554' 104410                                                                                  TRAP    C$ESCAPE
045556' 000246                                                                                  .WORD   L10161-.
045560' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
045564' 103006                          BCC     25$
045566'                                 ERRHRD  233,CRCERR,RACMG7       ;DNI DID NOT CLEAR!
045566' 104456                                                                                  TRAP    C$ERHRD
045570' 000351                                                                                  .WORD   233
045572' 003656'                                                                                 .WORD   CRCERR
045574' 012726'                                                                                 .WORD   RACMG7
045576'                                 ESCAPE TST
045576' 104410                                                                                  TRAP    C$ESCAPE
045600' 000224                                                                                  .WORD   L10161-.
045602'                         25$:
045602'                           ENDSEG
045602'                                                                                 10000$:
045602' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;CLEAR PCBB+2, GET A DATA PATTERN FROM A LIST OF PRESELECTED DATA PATTERNS
                                ;AND PLACE IT IN PCBB+0 FOR THE MICROCODE
                                ;
045604' 005037  000606'                 CLR     PCBB+2                  ;HERE IS WHERE MICROCODE WILL PUT...
                                                                        ;STATUS WORD
045610' 012701  000516'                 MOV     #PATERN,R1              ;GET ADDRESS OF DATA PATTERN TABLE
045614' 012705  000005                  MOV     #5,R5                   ;# OF DATA PATTERNS
045620' 012137  000604'         27$:    MOV     (R1)+,PCBB              ;LOAD PCBB WITH A DATA PATTERN
                                ;
                                ;WAIT FOR THE MICROMONITOR TO BECOME READY
                                ;
045624'                           BGNSEG
045624' 104404                                                                                  TRAP    C$BSEG
045626' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
045632' 103006                          BCC     30$                     ;OK
045634'                                 ERRHRD  234,CRCERR,MSG46        ;PRINT ERROR
045634' 104456                                                                                  TRAP    C$ERHRD
045636' 000352                                                                                  .WORD   234
045640' 003656'                                                                                 .WORD   CRCERR
045642' 016700'                                                                                 .WORD   MSG46
045644'                                 ESCAPE  TST                     ;LEAVE TEST
045644' 104410                                                                                  TRAP    C$ESCAPE
045646' 000156                                                                                  .WORD   L10161-.
045650'                         30$:
                                ;
                                ;EXECUTE MICROTEST #2 BY LOADING THE COMMAND FIELD OF PCSR0 WITH A 2
```

```
                                         ;WAIT FOR 'DNI'
                                         ;
045650' 012777  000002  132460                   MOV     #2,@PCSR0                ;TELL T11 TO EXECUTE MICROTEST #2
045656' 012737  000176  000332'                  MOV     #2*SECOND,METER          ;WAIT FOR DNI
045664' 004737  017330'                          JSR     PC,CHKDNI
045670' 103021                                   BCC     40$
045672' 004737  020050'                          JSR     PC,CHKINT                ;SEE IF ANY ERROR INTERRUPTS OCCURRED
045676' 103006                                   BCC     35$                      ;NO, OK
045700'                                          ERRHRD  235,CRCERR,MSG44         ;PRINT ERROR MESSAGE
045700' 104456                                                                                            TRAP    C$ERHRD
045702' 000353                                                                                            .WORD   235
045704' 003656'                                                                                           .WORD   CRCERR
045706' 016454'                                                                                           .WORD   MSG44
045710'                                          ESCAPE  TST
045710' 104410                                                                                            TRAP    C$ESCAPE
045712' 000112                                                                                            .WORD   L10161-.
045714' 012702  000002           35$:            MOV     #2,R2                    ;MICROTEST #2 IS HUNG
045720'                                          ERRHRD  236,CRCERR,MSG12
045720' 104456                                                                                            TRAP    C$ERHRD
045722' 000354                                                                                            .WORD   236
045724' 003656'                                                                                           .WORD   CRCERR
045726' 013500'                                                                                           .WORD   MSG12
045730'                                          ESCAPE  TST
045730' 104410                                                                                            TRAP    C$ESCAPE
045732' 000072                                                                                            .WORD   L10161-.
                                         ;
                                         ;OK, TEST IS COMPLETED, NOW CHECK PCBB+2. PCBB+2 CONTAINS THE RECEIVER STATUS
                                         ;WORD, IT SHOULD HAVE THE CRC ERROR BIT SET AND THE ERROR SUMMARY BIT SET
                                         ;
045734' 013704  000606'          40$:            MOV     PCBB+2,R4                ;THIS IS THE RECEIVER STATUS WORD
045740' 012703  004000                           MOV     #BIT11,R3                ;BIT CRC SHOULD BE SET
045744' 030304                                   BIT     R3,R4                    ;IS CRC BIT SET?
045746' 001004                                   BNE     45$                      ;YES, OK
045750'                                          ERRHRD  237,CRCERR,MSG27         ;NO, PRINT ERROR MESSAGE
045750' 104456                                                                                            TRAP    C$ERHRD
045752' 000355                                                                                            .WORD   237
045754' 003656'                                                                                           .WORD   CRCERR
045756' 015060'                                                                                           .WORD   MSG27
045760' 012703  040000           45$:            MOV     #BIT14,R3                ;ERROR SUMMARY BIT SHOULD BE SET
045764' 030304                                   BIT     R3,R4                    ;IS ERROR SUMMARY BIT SET?
045766' 001004                                   BNE     50$                      ;YES, GOOD
045770'                                          ERRHRD  238,CRCERR,MSG28         ;NO, PRINT ERROR MESSAGE
045770' 104456                                                                                            TRAP    C$ERHRD
045772' 000356                                                                                            .WORD   238
045774' 003656'                                                                                           .WORD   CRCERR
045776' 015126'                                                                                           .WORD   MSG28
                                         ;
                                         ;WRITE '1' TO CLEAR 'DNI'
                                         ;
046000' 004737  017374'          50$:            JSR     PC,CLRDNI                ;CLEAR DNI BIT
046004' 103004                                   BCC     55$
046006'                                          ERRHRD  239,CRCERR,RACMG7        ;ERROR DNI DID NOT CLEAR!
046006' 104456                                                                                            TRAP    C$ERHRD
046010' 000357                                                                                            .WORD   239
046012' 003656'                                                                                           .WORD   CRCERR
046014' 012726'                                                                                           .WORD   RACMG7
```

```
 11253  046016'                         55$:
 11254  046016'                           ENDSEG
 11255  046016'                                                                          10001$:
 11256  046016' 104405                                                                            TRAP    C$ESEG
 11257                                  ;
 11258                                  ;REPEAT TEST WITH ALL DATA PATTERNS
 11259                                  ;
 11260  046020' 005305                          DEC     R5                      ;HAVE WE TESTED WITH ALL DATA PATTERNS?
 11261  046022' 001276                          BNE     27$                     ;NOT YET
 11262
 11263  046024'                         ENDTST
 11264  046024'                                                                          L10161:
 11265  046024' 104401                                                                            TRAP    C$ETST
```

```
                                          .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 MICROTEST #3
                                          ;       7-WAIT FOR 'DNI'
                                          ;       8-VERIFY NO ERRORS IN PCBB+4
                                          ;       9-WRITE '1' TO CLEAR 'DNI'
                                          ;       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
                                          ;
                                          ;*********************************************************************

 046026'                                  BGNTST
 046026'                                                                                   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.
                                          ;
 046026'                                   BGNSEG
 046026' 104404                                                                                    TRAP    C$BSEG
 046030' 022737  000106  000326'                 CMP     #'F,MICRO              ;HAS MICROCODE MODULE 'F' BEEN LOADED
 046036' 001004                                  BNE     5$                     ;NO
```

```
046040' 122777  000001  132272          CMPB    #INMON,SPCSW+1              ;YES, IS THE MICROMONITOR ACTIVE?
046046' 001440                          BEQ     20$                         ;YES SKIP LOADING THE MICROMODULE
046050' 012737  000106  000326' 5$:     MOV     #'F,MICRO                   ;GO LOAD MICRO MODULE 'F'
046056' 004737  020252'                 JSR     PC,LODMIC
046062' 103002                          BCC     10$                         ;OK
046064'                                 ESCAPE  TST
046064' 104410                                                                                 TRAP    C$ESCAPE
046066' 000320                                                                                 .WORD   L10162-.
046070' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER             ;WAIT FOR THE MICROMONITOR
046076' 004737  017330'                 JSR     PC,CHKDNI
046102' 103022                          BCC     20$                         ;OK
046104' 012737  000776' 000310'         MOV     #$DNI,BITNAM
046112' 012737  001275' 000312'         MOV     #$NSET,BITSTA
046120' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
046126' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
046134'                                 ERRHRD  240,CRCPAT,MSG1
046134' 104456                                                                                 TRAP    C$ERHRD
046136' 000360                                                                                 .WORD   240
046140' 003704'                                                                                .WORD   CRCPAT
046142' 012750'                                                                                .WORD   MSG1
046144'                                 ESCAPE  TST
046144' 104410                                                                                 TRAP    C$ESCAPE
046146' 000240                                                                                 .WORD   L10162-.
046150' 004737  017374'         20$:    JSR     PC,CLRDNI                   ;CLEAR DNI BIT
046154' 103006                          BCC     25$
046156'                                 ERRHRD  241,CRCPAT,RACMG7           ;DNI DID NOT CLEAR!
046156' 104456                                                                                 TRAP    C$ERHRD
046160' 000361                                                                                 .WORD   241
046162' 003704'                                                                                .WORD   CRCPAT
046164' 012726'                                                                                .WORD   RACMG7
046166'                                 ESCAPE TST
046166' 104410                                                                                 TRAP    C$ESCAPE
046170' 000216                                                                                 .WORD   L10162-.
046172'                         25$:
046172'                           ENDSEG
046172'                                                                                10000$:
046172' 104405                                                                                 TRAP    C$ESEG
                                ;
                                ;POINT TO LIST OF DATA PATTERNS
                                ;
046174' 012701  000516'                 MOV     #PATERN,R1                  ;GET ADDRESS OF DATA PATTERN TABLE
046200' 012705  000005                  MOV     #5,R5                       ;# OF DATA PATTERNS
                                ;
                                ;GET A DATA PATTERN FROM THE LIST AND PLACE IT IN PCBB+0
                                ;
046204'                         27$:
046204' 012137  000604'                 MOV     (R1)+,PCBB                  ;LOAD PCBB WITH A DATA PATTERN
046210'                           BGNSEG
046210' 104404                                                                                 TRAP    C$BSEG
                                ;
                                ;GENERATE A MINIMUM BYTE COUNT
                                ;
046212' 012703  000100                  MOV     #BIT6,R3                    ;R3 WILL BE BYTE COUNT
046216'                         28$:
                                ;
                                ;LOAD BYTE COUNT INTO THE PCBB+4
                                ;
046216'                          BGNSEG
046216' 104404                                                                            TRAP     C$BSEG
046220' 010337  000606'                 MOV     R3,PCBB+2           ;LOAD BYTE COUNT INTO PCBB
                                ;
                                ;WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE
                                ;EXECUTE MICROTEST #3 BY LOADING THE COMMAND FIELD OF PCSR0 WITH A 3
                                ;WAIT FOR 'DNI'
                                ;
046224' 004737  017776'                 JSR     PC,CHKMON           ;WAIT FOR MICROMONITOR
046230' 103006                          BCC     30$                 ;OK
046232'                                 ERRHRD  242,CRCPAT,MSG46    ;PRINT ERROR
046232' 104456                                                                            TRAP     C$ERHRD
046234' 000362                                                                            .WORD    242
046236' 003704'                                                                           .WORD    CRCPAT
046240' 016700'                                                                           .WORD    MSG46
046242'                                 ESCAPE  TST                 ;LEAVE TEST
046242' 104410                                                                            TRAP     C$ESCAPE
046244' 000142                                                                            .WORD    L10162-.
046246' 012777  000003  132062  30$:    MOV     #3,@PCSR0           ;TELL T11 TO EXECUTE MICROTEST #3
046254' 012737  000176  000332'         MOV     #2*SECOND,METER     ;WAIT FOR DNI
046262' 004737  017330'                 JSR     PC,CHKDNI
046266' 103021                          BCC     40$
046270' 004737  020050'                 JSR     PC,CHKINT           ;SEE IF ANY ERROR INTERRUPTS OCCURRED
046274' 103006                          BCC     35$                 ;NO, OK
046276'                                 ERRHRD  243,CRCPAT,MSG44    ;PRINT ERROR MESSAGE
046276' 104456                                                                            TRAP     C$ERHRD
046300' 000363                                                                            .WORD    243
046302' 003704'                                                                           .WORD    CRCPAT
046304' 016454'                                                                           .WORD    MSG44
046306'                                 ESCAPE  TST
046306' 104410                                                                            TRAP     C$ESCAPE
046310' 000076                                                                            .WORD    L10162-.
046312' 012702  000003          35$:    MOV     #3,R2               ;MICROTEST #3 IS HUNG
046316'                                 ERRHRD  244,CRCPAT,MSG12
046316' 104456                                                                            TRAP     C$ERHRD
046320' 000364                                                                            .WORD    244
046322' 003704'                                                                           .WORD    CRCPAT
046324' 013500'                                                                           .WORD    MSG12
046326'                                 ESCAPE  TST
046326' 104410                                                                            TRAP     C$ESCAPE
046330' 000056                                                                            .WORD    L10162-.
                                ;
                                ;CHECK THE RECEIVER STATUS WORD, WHICH THE MICROCODE PLACED IN PCBB+4, FOR ANY
                                ;ERRORS
                                ;
046332' 005737  000610'         40$:    TST     PCBB+4              ;ANYTHING SET IN RECEIVER STATUS WORD 0
046336' 001404                          BEQ     50$                 ;NO, OK
046340'                                 ERRHRD  245,CRCPAT,MSG29    ;PRINT ERROR MESSAGE
046340' 104456                                                                            TRAP     C$ERHRD
046342' 000365                                                                            .WORD    245
046344' 003704'                                                                           .WORD    CRCPAT
046346' 015174'                                                                           .WORD    MSG29
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
```

```
046350' 004737  017374'         50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
046354' 103004                          BCC     55$
046356'                                 ERRHRD  246,CRCPAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
046356' 104456                                                                                  TRAP    C$ERHRD
046360' 000366                                                                                  .WORD   246
046362' 003704'                                                                                 .WORD   CRCPAT
046364' 012726'                                                                                 .WORD   RACMG7
046366'                         55$:
046366'                           ENDSEG
046366'                                                                                 10002$:
046366' 104405                                                                                  TRAP    C$ESEG

                                ;
                                ;GENERATE A NEW BYTE COUNT BY SHIFTING IT OVER ONE PLACE TO THE LEFT.
                                ;THIS EFFECTIVELY MULTIPLIES THE NUMBER BY TWO.
                                ;CHECK TO SEE IF THE NEW NUMBER IS NOT TOO LARGE.
                                ;REPEAT THE TEST WITH THE NEW BYTE COUNT
                                ;
046370' 006303                          ASL     R3                      ;UP THE BYTE COUNT TO NEXT BIT POSTION
046372' 020327  002000                  CMP     R3,#BIT10               ;HAVE WE CHECKED ALL BIT POSTIONS IN
                                                                        ;THE BYTE COUNT REGISTER?
046376' 001307                          BNE     28$                     ;NOT YET
046400'                           ENDSEG
046400'                                                                                 10001$:
046400' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;GET A NEW DATA PATTERN FROM THE LIST OF PRESELECTED DATA PATTERNS
                                ;REPEAT TEST WITH NEW DATA PATTERN
                                ;
046402' 005305                          DEC     R5                      ;HAVE WE TESTED WITH ALL DATA PATTERNS?
046404' 001277                          BNE     27$                     ;NOT YET

046406'                         ENDTST
046406'                                                                                 L10162:
046406' 104401                                                                                  TRAP    C$ETST
```

```
                        .SBTTL  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.
                        ;
                        ;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
                        ;
                        ;*************************************************************************

 046410'                BGNTST
 046410'                                                    T41::

                        ;
                        ;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.
                                ;
046410'                           BGNSEG
046410' 104404                                                                                  TRAP    C$BSEG
046412' 022737  000106  000326'         CMP     #'F,MICRO               ;HAS MICROCODE MODULE 'F' BEEN LOADED
046420' 001004                          BNE     5$                      ;NO
046422' 122777  000001  131710          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
046430' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
046432' 012737  000106  000326' 5$:     MOV     #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
046440' 004737  020252'                 JSR     PC,LODMIC
046444' 103002                          BCC     10$                     ;OK
046446'                                 ESCAPE  TST
046446' 104410                                                                                  TRAP    C$ESCAPE
046450' 000370                                                                                  .WORD   L10163-.
046452' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
046460' 004737  017330'                 JSR     PC,CHKDNI
046464' 103022                          BCC     20$                     ;OK
046466' 012737  000776' 000310'         MOV     #$DNI,BITNAM
046474' 012737  001275' 000312'         MOV     #$NSET,BITSTA
046502' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
046510' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
046516'                                 ERRHRD  247,RBRRUN,MSG1
046516' 104456                                                                                  TRAP    C$ERHRD
046520' 000367                                                                                  .WORD   247
046522' 003743'                                                                                 .WORD   RBRRUN
046524' 012750'                                                                                 .WORD   MSG1
046526'                                 ESCAPE  TST
046526' 104410                                                                                  TRAP    C$ESCAPE
046530' 000310                                                                                  .WORD   L10163-.
046532' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
046536' 103006                          BCC     25$
046540'                                 ERRHRD  248,RBRRUN,RACMG7       ;DNI DID NOT CLEAR!
046540' 104456                                                                                  TRAP    C$ERHRD
046542' 000370                                                                                  .WORD   248
046544' 003743'                                                                                 .WORD   RBRRUN
046546' 012726'                                                                                 .WORD   RACMG7
046550'                                 ESCAPE TST
046550' 104410                                                                                  TRAP    C$ESCAPE
046552' 000266                                                                                  .WORD   L10163-.
046554'                         25$:
046554'                           ENDSEG
046554'                                                                                 10000$:
046554' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;R1 WILL CONTAIN THE RUNT BYTE COUNT TO BE USED BY THE MICROCODE
                                ;
046556' 012701  000001                  MOV     #1,R1                   ;BEGIN WITH BYTE COUNT OF 1
046562'                         30$:
                                ;
                                ;PASS THE BYTE COUNT TO MICROCODE THROUGH PCBB+, CLEAR PCBB+2 AND PCBB+4
                                ;
046562'                           BGNSEG
046562' 104404                                                                                  TRAP    C$BSEG
046564' 010137  000604'                 MOV     R1,PCBB                 ;PASS BYTE COUNT TO MICROCODE
046570' 005037  000606'                 CLR     PCBB+2                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                        ;'DONE' RECEIVE BUFFER FIFO ADDRESS
```

```
046574'  005037   000610'                    CLR      PCBB+4                      ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                                  ;FIRST WORD OF DATA FROM THE RECEIVE BUFFER
                                    ;
                                    ;WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE, START MICROTEST #4 BY
                                    ;LOADING THE COMMAND FIELD OF PCSR0 WITH A 4, WAIT FOR 'DNI'
                                    ;
046600'  004737   017776'                    JSR      PC,CHKMON                   ;WAIT FOR MICROMONITOR
046604'  103006                              BCC      35$                         ;OK
046606'                                      ERRHRD   249,RBRRUN,MSG46            ;PRINT ERROR
046606'  104456                                                                                               TRAP     C$ERHRD
046610'  000371                                                                                               .WORD    249
046612'  003743'                                                                                              .WORD    RBRRUN
046614'  016700'                                                                                              .WORD    MSG46
046616'                                      ESCAPE   TST                         ;LEAVE TEST
046616'  104410                                                                                               TRAP     C$ESCAPE
046620'  000220                                                                                               .WORD    L10163-.
046622'  012777   000004   131506   35$:     MOV      #4,@PCSR0                   ;TELL T11 TO EXECUTE MICROTEST #4
046630'  012737   000275   000332'           MOV      #3*SECOND,METER             ;WAIT FOR DNI
046636'  004737   017330'                    JSR      PC,CHKDNI
046642'  103021                              BCC      40$                         ;OK
046644'  004737   020050'                    JSR      PC,CHKINT                   ;SEE IF ANY ERROR INTERRUPTS OCCURRED
046650'  103006                              BCC      36$                         ;NO, OK
046652'                                      ERRHRD   250,RBRRUN,MSG44            ;PRINT ERROR MESSAGE
046652'  104456                                                                                               TRAP     C$ERHRD
046654'  000372                                                                                               .WORD    250
046656'  003743'                                                                                              .WORD    RBRRUN
046660'  016454'                                                                                              .WORD    MSG44
046662'                                      ESCAPE   TST
046662'  104410                                                                                               TRAP     C$ESCAPE
046664'  000154                                                                                               .WORD    L10163-.
046666'  012702   000004            36$:     MOV      #4,R2                       ;MICROTEST #4 IS HUNG
046672'                                      ERRHRD   251,RBRRUN,MSG12            ;PRINT ERROR MESSAGE
046672'  104456                                                                                               TRAP     C$ERHRD
046674'  000373                                                                                               .WORD    251
046676'  003743'                                                                                              .WORD    RBRRUN
046700'  013500'                                                                                              .WORD    MSG12
046702'                                      ESCAPE TST
046702'  104410                                                                                               TRAP     C$ESCAPE
046704'  000134                                                                                               .WORD    L10163-.
                                    ;
                                    ;'DNI' SET INDICATING THE TEST IS FINISHED, NOW CHECK PCSR1 FOR AN ERROR
                                    ;CONDITION. THIS CONDITION WILL BE SET IF, AFTER THE MICROCODE TRANSMITTED THE
                                    ;RUNT PACKET, A RECEIVER INTERRUPT OCCURRED. THIS SHOULD NOT HAPPEN BECAUSE
                                    ;THE RECEIVER STATE MACHINE SHOULD GO THROUGH A BAD PACKET STATE AND NOT
                                    ;CAUSE A RECEIVER INTERRUPT
                                    ;
046706'  122777   000003   131424   40$:     CMPB     #INERR,@PCSR1               ;DID AN ERROR OCCUR?
046714'  001005                              BNE      45$                         ;NO
046716'                                      ERRHRD   252,RBRRUN,MSG30            ;YES, PRINT ERROR MESSAGE
046716'  104456                                                                                               TRAP     C$ERHRD
046720'  000374                                                                                               .WORD    252
046722'  003743'                                                                                              .WORD    RBRRUN
046724'  015276'                                                                                              .WORD    MSG30
046726'  000425                              BR       55$
                                    ;
                                    ;OK, NO RECEIVER INTERRUPT. NOW CHECK PCBB+2, WHICH IS THE CONTENTS OF THE
```

```
                                        ;BUFFER DONE FIFO AFTER THE SECOND PACKET WAS RECEIVED. BITS 14:11 SHOULD
                                        ;ALL BE ZERO INDICATING BUFFER 0 WAS RECOVERED BY THE RECEIVER STATE MACHINE.
                                        ;
046730' 042737  103777  000606' 45$:          BIC     #103777,PCBB+2          ;STRIP OFF THE FLOATING BITS
046736' 005737  000606'                       TST     PCBB+2                  ;DID DEUNA RECOVER RECEIVE BUFFER?
046742' 001407                                BEQ     50$                     ;YES
046744' 052737  100000  000606'               BIS     #100000,PCBB+2          ;NO, MAKE IT A LINK MEMORY ADDRESS
046752'                                       ERRHRD  253,RBRRUN,MSG31        ;PRINT ERROR MESSAGE
046752' 104456                                                                                        TRAP    C$ERHRD
046754' 000375                                                                                        .WORD   253
046756' 003743'                                                                                       .WORD   RBRRUN
046760' 015344'                                                                                       .WORD   MSG31
                                        ;
                                        ;NOW CHECK THE DATA THAT WAS RECEIVED INTO THE RECEIVER BUFFER. IT SHOULD
                                        ;BE AN ALTERNATING 1'S AND 0'S PATTERN.
                                        ;
046762' 022737  052525  000610' 50$:          CMP     #52525,PCBB+4           ;IS DATA GOOD IN LEGIT RECEIVE BUFFER?
046770' 001404                                BEQ     55$                     ;YES
046772'                                       ERRHRD  254,RBRRUN,MSG32        ;NO,PRINT ERROR MESSAGE
046772' 104456                                                                                        TRAP    C$ERHRD
046774' 000376                                                                                        .WORD   254
046776' 003743'                                                                                       .WORD   RBRRUN
047000' 015510'                                                                                       .WORD   MSG32
047002'                                 55$:
047002'                                   ENDSEG
047002'                                                                                       10001$:
047002' 104405                                                                                        TRAP    C$ESEG
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI'
                                        ;
047004' 004737  017374'                       JSR     PC,CLRDNI               ;GO CLEAR DNI
047010' 103006                                BCC     60$                     ;OK
047012'                                       ERRHRD  255,RBRRUN,RACMG7       ;ERROR DNI DID NOT CLEAR!
047012' 104456                                                                                        TRAP    C$ERHRD
047014' 000377                                                                                        .WORD   255
047016' 003743'                                                                                       .WORD   RBRRUN
047020' 012726'                                                                                       .WORD   RACMG7
047022'                                       ESCAPE TST
047022' 104410                                                                                        TRAP    C$ESCAPE
047024' 000014                                                                                        .WORD   L10163-.
                                        ;
                                        ;GENERATE A NEW RUNT PACKET BYTE COUNT. WE WILL JUST SLIDE A BIT THROUGH
                                        ;THE COUNTER, UP TO THE LAST RUNT PACKET SIZE OF 63
                                        ;
047026' 006301                          60$:          ASL     R1                      ;MOVE OVER THE ALREADY SET BITS
047030' 005201                                        INC     R1                      ;SET LSB
047032' 020127  000100                                CMP     R1,#MINBYT              ;STOP WITH A BYTE COUNT GREATER THAN
                                                                                      ;MINIMUM SIZE
047036' 002651                                        BLT     30$
047040'                                 ENDTST
047040'                                                                                       L10163:
047040' 104401                                                                                        TRAP    C$ETST
```

```
                                        .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' 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!
                                        ;               +--------------------------------+
                                        ;
                                        ;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 'INMON' 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
                                        ;
                                        ;*****************************************************************

 047042'                               BGNTST
 047042'                                                                                T42::
                                        ;
                                        ;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.
                                        ;
 047042'                                 BGNSEG
 047042' 104404                                                                             TRAP    C$BSEG
 047044' 022737  000106  000326'              CMP     #'F,MICRO               ;HAS MICROCODE MODULE 'F' BEEN LOADED
```

```
047052' 001004                            BNE     5$                      ;NO
047054' 122777  000001  131256            CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
047062' 001440                            BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
047064' 012737  000106  000326' 5$:       MOV     #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
047072' 004737  020252'                   JSR     PC,LODMIC
047076' 103002                            BCC     10$                     ;OK
047100'                                   ESCAPE  TST
047100' 104410                                                                                    TRAP    C$ESCAPE
047102' 000342                                                                                    .WORD   L10164-.
047104' 012737  000176  000332' 10$:      MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
047112' 004737  017330'                   JSR     PC,CHKDNI
047116' 103022                            BCC     20$                     ;OK
047120' 012737  000776' 000310'           MOV     #$DNI,BITNAM
047126' 012737  001275' 000312'           MOV     #$NSET,BITSTA
047134' 012737  001340' 000314'           MOV     #$AFTER,PWHEN
047142' 012737  001355' 000316'           MOV     #$GTCMD,PCOMND
047150'                                   ERRHRD  256,HAFDUP,MSG1
047150' 104456                                                                                    TRAP    C$ERHRD
047152' 000400                                                                                    .WORD   256
047154' 004016'                                                                                   .WORD   HAFDUP
047156' 012750'                                                                                   .WORD   MSG1
047160'                                   ESCAPE  TST
047160' 104410                                                                                    TRAP    C$ESCAPE
047162' 000262                                                                                    .WORD   L10164-.
047164' 004737  017374'         20$:      JSR     PC,CLRDNI               ;CLEAR DNI BIT
047170' 103006                            BCC     25$
047172'                                   ERRHRD  257,HAFDUP,RACMG7       ;DNI DID NOT CLEAR!
047172' 104456                                                                                    TRAP    C$ERHRD
047174' 000401                                                                                    .WORD   257
047176' 004016'                                                                                   .WORD   HAFDUP
047200' 012726'                                                                                   .WORD   RACMG7
047202'                                   ESCAPE    T
047202' 104410                                                                                    TRAP    C$ESCAPE
047204' 000240                                                                                    .WORD   L10164-.
047206'                         25$:
047206'                           ENDSEG
047206'                                                                                  10000$:
047206' 104405                                                                                    TRAP    C$ESEG
                                ;
                                ;CLEAR PCBB+0 AND PCBB+2 THESE LOCATIONS ARE WHERE THE MICROCODE WILL PLACE
                                ;INFORMATION ABOUT THE TEST SUCCESS
                                ;
047210'                           BGNSEG
047210' 104404                                                                                    TRAP    C$BSEG
047212' 005037  000604'                   CLR     PCBB+0                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                          ;'DONE' RECEIVE BUFFER FIFO ADDRESS
047216' 005037  000606'                   CLR     PCBB+2                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                          ;FIRST WORD OF DATA FROM THE RECEIVE BUFFER
                                ;
                                ;WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE
                                ;THEN EXECUTE MICROTEST #5 BY LOADING PCSR0 WITH A 5
                                ;WAIT FOR 'DNI'
                                ;
047222' 004737  017776'                   JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
047226' 103006                            BCC     35$                     ;OK
047230'                                   ERRHRD  258,HAFDUP,MSG46        ;PRINT ERROR
```

```
047230' 104456                                                                                          TRAP    C$ERHRD
047232' 000402                                                                                          .WORD   258
047234' 004016'                                                                                         .WORD   HAFDUP
047236' 016700'                                                                                         .WORD   MSG46
047240'                                 ESCAPE  TST                             ;LEAVE TEST
047240' 104410                                                                                          TRAP    C$ESCAPE
047242' 000202                                                                                          .WORD   L10164-.
047244' 012777  000005  131064  35$:    MOV     #5,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #5
047252' 012737  000473  000332'         MOV     #5*SECOND,METER         ;WAIT FOR DNI
047260' 004737  017330'                 JSR     PC,CHKDNI
047264' 103021                          BCC     40$                     ;OK
047266' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
047272' 103006                          BCC     36$                     ;NO, OK
047274'                                 ERRHRD  259,HAFDUP,MSG44        ;PRINT ERROR MESSAGE
047274' 104456                                                                                          TRAP    C$ERHRD
047276' 000403                                                                                          .WORD   259
047300' 004016'                                                                                         .WORD   HAFDUP
047302' 016454'                                                                                         .WORD   MSG44
047304'                                 ESCAPE  TST
047304' 104410                                                                                          TRAP    C$ESCAPE
047306' 000136                                                                                          .WORD   L10164-.
047310' 012702  000005          36$:    MOV     #5,R2                   ;MICROTEST #5 IS HUNG
047314'                                 ERRHRD  260,HAFDUP,MSG12        ;PRINT ERROR MESSAGE
047314' 104456                                                                                          TRAP    C$ERHRD
047316' 000404                                                                                          .WORD   260
047320' 004016'                                                                                         .WORD   HAFDUP
047322' 013500'                                                                                         .WORD   MSG12
047324'                                 ESCAPE TST
047324' 104410                                                                                          TRAP    C$ESCAPE
047326' 000116                                                                                          .WORD   L10164-.
                                ;
                                ;AN ERROR IN PCSR1 MEANS THAT THE FIRST DATAGRAM SENT, WHICH IS THE ONE SENT
                                ;IN HALF-DUPLEX MODE, CAUSED A RECEIVER INTERRUPT.
                                ;
047330' 122777  000003  131002  40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
047336' 001005                          BNE     45$                     ;NO
047340'                                 ERRHRD  261,HAFDUP,MSG39        ;YES, PRINT ERROR MESSAGE
047340' 104456                                                                                          TRAP    C$ERHRD
047342' 000405                                                                                          .WORD   261
047344' 004016'                                                                                         .WORD   HAFDUP
047346' 016070'                                                                                         .WORD   MSG39
047350' 000425                          BR      55$
                                ;
                                ;PCBB+0 CONTAINS THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM
                                ;WAS RECEIVED IT SHOULD CONTAIN THE BUFFER ADDRESS OF THE FIRST BUFFER IN LINK
                                ;MEMORY
                                ;
047352' 042737  103777  000604' 45$:    BIC     #103777,PCBB            ;STRIP OFF THE FLOATING BITS
047360' 005737  000604'                 TST     PCBB                    ;DID DEUNA RECOVER RECEIVE BUFFER?
047364' 001407                          BEQ     50$                     ;YES
047366' 052737  100000  000604'         BIS     #100000,PCBB            ;MAKE IT A LINK MEMORY ADDRESS
047374'                                 ERRHRD  262,HAFDUP,MSG40        ;NO, PRINT ERROR MESSAGE
047374' 104456                                                                                          TRAP    C$ERHRD
047376' 000406                                                                                          .WORD   262
047400' 004016'                                                                                         .WORD   HAFDUP
047402' 016112'                                                                                         .WORD   MSG40
```

```
                                 ;
                                 ;PCBB+2 CONTAINS THE FIRST DATA WORD FROM THE FIRST BUFFER IN LINK MEMORY
                                 ;IT SHOULD BE ALTERNATING 1'S AND 0'S IF THE DATA WAS PLACED IN THIS BUFFER
                                 ;
047404' 022737  052525  000606'  50$:     CMP     #52525,PCBB+2           ;IS DATA GOOD IN LEGIT RECEIVE BUFFER?
047412' 001404                            BEQ     55$                     ;YES
047414'                                   ERRHRD  263,HAFDUP,MSG41        ;NO,PRINT ERROR MESSAGE
047414' 104456                                                                                        TRAP     C$ERHRD
047416' 000407                                                                                        .WORD    263
047420' 004016'                                                                                       .WORD    HAFDUP
047422' 016210'                                                                                       .WORD    MSG41
047424'                          55$:
047424'                            ENDSEG
047424'                                                                                     10001$:
047424' 104405                                                                                        TRAP     C$ESEG
                                 ;
                                 ;WRITE '1' TO CLEAR 'DNI'
                                 ;
047426' 004737  017374'                   JSR     PC,CLRDNI               ;GO CLEAR DNI
047432' 103004                            BCC     60$                     ;OK
047434'                                   ERRHRD  264,HAFDUP,RACMG7       ;ERROR DNI DID NOT CLEAR!
047434' 104456                                                                                        TRAP     C$ERHRD
047436' 000410                                                                                        .WORD    264
047440' 004016'                                                                                       .WORD    HAFDUP
047442' 012726'                                                                                       .WORD    RACMG7
047444'                          60$:
047444'                          ENDTST
047444'                                                                                     L10164:
047444' 104401                                                                                        TRAP     C$ETST
```

```
.SBTTL  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 CONTATINS DIAGNOSTIC LOGIC THAT ALLOWS COLLSIONS 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
;               --------------------    ------
;               !                  !    !    !
;               !                  !    !    !
;LOOPBACK       ERRS    MORE    ONE     RETRY
;  STEP #       (14)    (12)    (11)     (10)
;--------       ----    ----    ----    -----
;
; 1              0       0       1        0
;
; 2-15           0       1       0        0
;
; 16             1       0       0        1
;
;
;THE PCBB IS FORMATTED AS FOLOWS:
;
;               +------------------------+
;PCBB+0:        !     LINK MODE WORD     !
;               +------------------------+
;PCBB+2:        ! TRANSMIT STATUS WORD 0!
;               +------------------------+
;PCBB+4:        ! TRANSMIT STATUS WORD 1!
;               +------------------------+
;
```

```
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'G' IF NOT ALREADY SONE 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
                                        ;       COLLSIONS
                                        ;       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

 047446'                                BGNTST
 047446'                                                                                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.
                                        ;
 047446'                                 BGNSEG
 047446' 104404                                                                                 TRAP    C$BSEG
 047450' 022737  000107  000326'                CMP     #'G,MICRO               ;HAS MICROCODE MODULE 'G' BEEN LOADED
 047456' 001004                                 BNE     5$                      ;NO
 047460' 122777  000001  130652                 CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 047466' 001440                                 BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 047470' 012737  000107  000326' 5$:            MOV     #'G,MICRO               ;GO LOAD MICRO MODULE 'G'
 047476' 004737  020252'                        JSR     PC,LODMIC
 047502' 103002                                 BCC     10$                     ;OK
 047504'                                        ESCAPE  TST
 047504' 104410                                                                                 TRAP    C$ESCAPE
 047506' 001026                                                                                 .WORD   L10165-.
 047510' 012737  000176  000332' 10$:           MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
 047516' 004737  017330'                        JSR     PC,CHKDNI
 047522' 103022                                 BCC     20$                     ;OK
 047524' 012737  000776' 000310'                MOV     #$DNI,BITNAM
 047532' 012737  001275' 000312'                MOV     #$NSET,BITSTA
 047540' 012737  001340' 000314'                MOV     #$AFTER,PWHEN
 047546' 012737  001355' 000316'                MOV     #$GTCMD,PCOMND
 047554'                                        ERRHRD  265,COLTST,MSG1
```

```
047554' 104456                                                                                  TRAP    C$ERHRD
047556' 000411                                                                                  .WORD   265
047560' 004046'                                                                                 .WORD   COLTST
047562' 012750'                                                                                 .WORD   MSG1
047564'                                 ESCAPE  TST
047564' 104410                                                                                  TRAP    C$ESCAPE
047566' 000746                                                                                  .WORD   L10165-.
047570' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
047574' 103006                          BCC     25$
047576'                                 ERRHRD  266,COLTST,RACMG7       ;DNI DID NOT CLEAR!
047576' 104456                                                                                  TRAP    C$ERHRD
047600' 000412                                                                                  .WORD   266
047602' 004046'                                                                                 .WORD   COLTST
047604' 012726'                                                                                 .WORD   RACMG7
047606'                                 ESCAPE TST
047606' 104410                                                                                  TRAP    C$ESCAPE
047610' 000724                                                                                  .WORD   L10165-.
047612'                         25$:
047612'                           ENDSEG
047612'                                                                                 10000$:
047612' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;LOAD PCSR2 WITH ADDRESS OF PORT CONTROL BLOCK
                                ;LOAD REGISTER 5 WITH BITS TO BE LOADED INTO THE LINK MODE REGISTER BY THE
                                ;MICROCODE
                                ;
047614' 012777  000604' 130520          MOV     #PCBB,@PCSR2            ;TELL DEUNA WHERE PCBB IS
047622' 005077  130516                  CLR     @PCSR3
047626' 012705  100024                  MOV     #BIT15!BIT4!BIT2,R5     ;TELL MICROCODE TO LOAD THE FOLLOWING
                                                                        ;INTO THE LINK MODE REGISTER:
                                                                        ;PROMISCUOUS MODE, INTERNAL LOOPBACK,
                                                                        ;AND FORCE COLLISIONS
                                ;
                                ;BEGIN FIRST LOOPBACK
                                ;
047632'                           BGNSEG
047632' 104404                                                                                  TRAP    C$BSEG
047634' 012702  000001                  MOV     #1,R2                   ;START LOOPBACK STEP #1
047640' 010537  000604'                 MOV     R5,PCBB                 ;LOAD PCBB WITH LINK MODE REGISTER DATA
047644' 005037  000606'                 CLR     PCBB+2                  ;MICROCODE WILL PUT XMIT STAT WORD 0
047650' 005037  000610'                 CLR     PCBB+4                  ;MICROCODE WILL PUT XMIT STAT WORD 1
047654' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
047660' 103006                          BCC     30$                     ;OK
047662'                                 ERRHRD  267,COLTST,MSG46        ;PRINT ERROR
047662' 104456                                                                                  TRAP    C$ERHRD
047664' 000413                                                                                  .WORD   267
047666' 004046'                                                                                 .WORD   COLTST
047670' 016700'                                                                                 .WORD   MSG46
047672'                                 ESCAPE  TST                     ;LEAVE TEST
047672' 104410                                                                                  TRAP    C$ESCAPE
047674' 000640                                                                                  .WORD   L10165-.
047676' 012777  000001  130432  30$:    MOV     #1,@PCSR0               ;EXECUTE MICROTEST #1
047704' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
047712' 004737  017330'                 JSR     PC,CHKDNI
047716' 103021                          BCC     40$
047720' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
```

```
047724' 103006                          BCC     35$                     ;NO, OK
047726'                                 ERRHRD  268,COLTST,MSG44        ;PRINT ERROR MESSAGE
047726' 104456                                                                                   TRAP    C$ERHRD
047730' 000414                                                                                   .WORD   268
047732' 004046'                                                                                  .WORD   COLTST
047734' 016454'                                                                                  .WORD   MSG44
047736'                                 ESCAPE  TST
047736' 104410                                                                                   TRAP    C$ESCAPE
047740' 000574                                                                                   .WORD   L10165-.
047742' 012702  000001          35$:    MOV     #1,R2                   ;MICROTEST #1 IS HUNG
047746'                                 ERRHRD  269,COLTST,MSG12
047746' 104456                                                                                   TRAP    C$ERHRD
047750' 000415                                                                                   .WORD   269
047752' 004046'                                                                                  .WORD   COLTST
047754' 013500'                                                                                  .WORD   MSG12
047756'                                 ESCAPE TST
047756' 104410                                                                                   TRAP    C$ESCAPE
047760' 000554                                                                                   .WORD   L10165-.
                                ;
                                ;THE RESULT OF THE FIRST LOOPBACK SHOULD BE TX WORD 0 'ONE' BIT SET AND NO
                                ;OTHERS
                                ;
047762' 012703  004000          40$:    MOV     #BIT11,R3               ;'ONE' BIT SHOULD BE SET IN TX WORD 0
047766' 005004                          CLR     R4                      ;NO BITS SHOULD BE SET IN TX WORD 1
047770' 032737  040000  000606'         BIT     #BIT14,PCBB+2           ;IS 'ERROR SUMMARY' SET IN WORD 0?
047776' 001014                          BNE     45$                     ;YES, ERROR
050000' 032737  010000  000606'         BIT     #BIT12,PCBB+2           ;IS 'MORE' BIT SET IN WORD 0?
050006' 001010                          BNE     45$                     ;YES, ERROR
050010' 032737  004000  000606'         BIT     #BIT11,PCBB+2           ;IS 'ONE' BIT SET IN WORD 0?
050016' 001404                          BEQ     45$                     ;NO, ERROR
050020' 032737  002000  000610'         BIT     #BIT10,PCBB+4           ;IS 'RETRY' BIT SET IN WORD 1?
050026' 001404                          BEQ     50$                     ;NO
050030'                         45$:    ERRHRD  270,COLTST,MSG22        ;PRINT ERROR MESSAGE
050030' 104456                                                                                   TRAP    C$ERHRD
050032' 000416                                                                                   .WORD   270
050034' 004046'                                                                                  .WORD   COLTST
050036' 014340'                                                                                  .WORD   MSG22

                                ;
                                ;WRITE 'ONE' TO CLEAR 'DNI'
                                ;
050040' 004737  017374'         50$:    JSR     PC,CLRDNI               ;GO CLEAR DNI
050044' 103004                          BCC     55$                     ;OK
050046'                                 ERRHRD  271,COLTST,RACMG7       ;PRINT ERROR MESSAGE
050046' 104456                                                                                   TRAP    C$ERHRD
050050' 000417                                                                                   .WORD   271
050052' 004046'                                                                                  .WORD   COLTST
050054' 012726'                                                                                  .WORD   RACMG7
050056'                         55$:
050056'                           ENDSEG
050056'                                                                                 10001$:
050056' 104405                                                                                   TRAP    C$ESEG
                                ;
                                ;BEGIN LOOPBACKS 2-15
                                ;
050060'                           BGNSEG
```

```
050060' 104404                                                                           TRAP    C$BSEG
050062' 012702  000002                   MOV     #2,R2                  ;START LOOPS 2-15

050066' 010537  000604'          60$:    MOV     R5,PCBB                ;LOAD PCBB WITH LINK MODE REGISTER DATA
050072' 005037  000606'                  CLR     PCBB+2                 ;MICROCODE WILL PUT XMIT STAT WORD 0
050076' 005037  000610'                  CLR     PCBB+4                 ;MICROCODE WILL PUT XMIT STAT WORD 1
050102' 004737  017776'                  JSR     PC,CHKMON              ;WAIT FOR MICROMONITOR
050106' 103006                           BCC     70$                    ;OK
050110'                                  ERRHRD  272,COLTST,MSG46       ;PRINT ERROR
050110' 104456                                                                           TRAP    C$ERHRD
050112' 000420                                                                           .WORD   272
050114' 004046'                                                                          .WORD   COLTST
050116' 016700'                                                                          .WORD   MSG46
050120'                                  ESCAPE  TST                    ;LEAVE TEST
050120' 104410                                                                           TRAP    C$ESCAPE
050122' 000412                                                                           .WORD   L10165-.
050124' 012777  000001  130204   70$:    MOV     #1,@PCSR0              ;EXECUTE MICROTEST #1
050132' 012737  000176  000332'          MOV     #2*SECOND,METER        ;WAIT FOR DNI
050140' 004737  017330'                  JSR     PC,CHKDNI
050144' 103017                           BCC     80$
050146' 004737  020050'                  JSR     PC,CHKINT              ;SEE IF ANY ERROR INTERRUPTS OCCURRED
050152' 103004                           BCC     75$                    ;NO, OK
050154'                                  ERRHRD  273,COLTST,MSG44       ;PRINT ERROR MESSAGE
050154' 104456                                                                           TRAP    C$ERHRD
050156' 000421                                                                           .WORD   273
050160' 004046'                                                                          .WORD   COLTST
050162' 016454'                                                                          .WORD   MSG44
050164' 012702  000001           75$:    MOV     #1,R2                  ;MICROTEST #1 IS HUNG
050170'                                  ERRHRD  274,COLTST,MSG12
050170' 104456                                                                           TRAP    C$ERHRD
050172' 000422                                                                           .WORD   274
050174' 004046'                                                                          .WORD   COLTST
050176' 013500'                                                                          .WORD   MSG12
050200'                                  ESCAPE TST
050200' 104410                                                                           TRAP    C$ESCAPE
050202' 000332                                                                           .WORD   L10165-.
                                 ;
                                 ;THE RESULT OF LOOPBACKS 2-15 SHOULD BE THE 'MORE' BIT IN TX 0 AND NO OTHERS
                                 ;
050204' 012703  010000           80$:    MOV     #BIT12,R3              ;'MORE' BIT SHOULD BE SET IN TX0
050210' 032737  040000  000606'          BIT     #BIT14,PCBB+2          ;IS 'ERROR SUMMARY' SET IN TX 0?
050216' 001014                           BNE     90$                    ;YES, ERROR
050220' 032737  010000  000606'          BIT     #BIT12,PCBB+2          ;IS 'MORE' BIT SET IN TX 0?
050226' 001410                           BEQ     90$                    ;NO, ERROR
050230' 032737  004000  000606'          BIT     #BIT11,PCBB+2          ;IS 'ONE' BIT SET IN TX 0?
050236' 001004                           BNE     90$                    ;YES, ERROR
050240' 032737  002000  000610'          BIT     #BIT10,PCBB+4          ;IS 'RETRY' BIT SET IN TX 1?
050246' 001404                           BEQ     100$                   ;NO
050250'                          90$:    ERRHRD  275,COLTST,MSG22       ;PRINT ERROR MESSAGE
050250' 104456                                                                           TRAP    C$ERHRD
050252' 000423                                                                           .WORD   275
050254' 004046'                                                                          .WORD   COLTST
050256' 014340'                                                                          .WORD   MSG22

                                 ;
                                 ;WRITE '1' TO CLEAR 'DNI'
                                 ;
050260' 004737  017374'         100$:   JSR     PC,CLRDNI               ;GO CLEAR DNI
050264' 103004                          BCC     105$                    ;OK
050266'                                 ERRHRD  276,COLTST,RACMG7       ;PRINT ERROR MESSAGE
050266' 104456                                                                                  TRAP    C$ERHRD
050270' 000424                                                                                  .WORD   276
050272' 004046'                                                                                 .WORD   COLTST
050274' 012726'                                                                                 .WORD   RACMG7
050276' 005202                  105$:   INC     R2                      ;LOOP COUNT
050300' 022702  000020                  CMP     #16.,R2                 ;HAVE WE DONE LOOP STEPS 2-15?
050304' 001270                          BNE     60$                     ;NO CONTINUE
050306'                           ENDSEG
050306'                                                                             10002$:
050306' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;BEGIN LOOPBACK #16
                                ;
050310'                           BGNSEG
050310' 104404                                                                                  TRAP    C$BSEG
050312' 010537  000604'                 MOV     R5,PCBB                 ;LOAD PCBB WITH LINK MODE REGISTER DATA
050316' 005037  000606'                 CLR     PCBB+2                  ;MICROCODE WILL PUT XMIT STAT WORD 0
050322' 005037  000610'                 CLR     PCBB+4                  ;MICROCODE WILL PUT XMIT STAT WORD 1
050326' 004737  017776'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
050332' 103006                          BCC     110$                    ;OK
050334'                                 ERRHRD  277,COLTST,MSG46        ;PRINT ERROR
050334' 104456                                                                                  TRAP    C$ERHRD
050336' 000425                                                                                  .WORD   277
050340' 004046'                                                                                 .WORD   COLTST
050342' 016700'                                                                                 .WORD   MSG46
050344'                                 ESCAPE  TST                     ;LEAVE TEST
050344' 104410                                                                                  TRAP    C$ESCAPE
050346' 000166                                                                                  .WORD   L10165-.
050350' 012777  000001  127760  110$:   MOV     #1,@PCSR0               ;EXECUTE MICROTEST #1
050356' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
050364' 004737  017330'                 JSR     PC,CHKDNI
050370' 103021                          BCC     120$
050372' 004737  020050'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
050376' 103006                          BCC     115$                    ;NO, OK
050400'                                 ERRHRD  278,COLTST,MSG44        ;PRINT ERROR MESSAGE
050400' 104456                                                                                  TRAP    C$ERHRD
050402' 000426                                                                                  .WORD   278
050404' 004046'                                                                                 .WORD   COLTST
050406' 016454'                                                                                 .WORD   MSG44
050410'                                 ESCAPE  TST
050410' 104410                                                                                  TRAP    C$ESCAPE
050412' 000122                                                                                  .WORD   L10165-.
050414' 012702  000001          115$:   MOV     #1,R2                   ;MICROTEST #1 IS HUNG
050420'                                 ERRHRD  279,COLTST,MSG12
050420' 104456                                                                                  TRAP    C$ERHRD
050422' 000427                                                                                  .WORD   279
050424' 004046'                                                                                 .WORD   COLTST
050426' 013500'                                                                                 .WORD   MSG12
050430'                                 ESCAPE  TST
050430' 104410                                                                                  TRAP    C$ESCAPE
050432' 000102                                                                                  .WORD   L10165-.
```

```
                                  ;
                                  ;PCBB+2 HAS TX WORD 0, IT SHOULD HAVE ERROR SUMMARY SET. PCBB+4 HAS TX WORD 1
                                  ;IT SHOULD HAVE RETRY SET
                                  ;
050434' 012703  040000            120$:   MOV     #BIT14,R3               ;'ERROR SUMMARY' SHOULD BE SET IN TX 0
050440' 012704  002000                    MOV     #BIT10,R4               ;'RETRY' BIT SHOULD BE SET IN TX 1

050444' 032737  040000  000606'           BIT     #BIT14,PCBB+2           ;IS 'ERROR SUMMARY' BIT SET?
050452' 001414                            BEQ     130$                    ;NO, ERROR
050454' 032737  010000  000606'           BIT     #BIT12,PCBB+2           ;IS 'MORE' BIT SET?
050462' 001010                            BNE     130$                    ;YES, EPROR
050464' 032737  004000  000606'           BIT     #BIT11,PCBB+2           ;IS 'ONE' 9IT SET?
050472' 001004                            BNE     130$                    ;YES, ERROR
050474' 032737  002000  000610'           BIT     #BIT10,PCBB+4           ;IS 'RETRY' BIT SET?
050502' 001004                            BNE     140$                    ;YES
050504'                           130$:   ERRHRD  280,COLTST,MSG22        ;PRINT ERROR MESSAGE
050504' 104456                                                                                    TRAP    C$ERHRD
050506' 000430                                                                                    .WORD   280
050510' 004046'                                                                                   .WORD   COLIST
050512' 014340'                                                                                   .WORD   MSG22
                                  ;
                                  ;WRITE '1' TO CLEAR 'DNI'
                                  ;
050514' 004737  017374'           140$:   JSR     PC,CLRDNI               ;GO CLEAR DNI BIT
050520' 103004                            BCC     150$                    ;OK
050522'                                   ERRHRD  281,COLTST,RACMG7       ;PRINT ERROR MESSAGE
050522' 104456                                                                                    TRAP    C$FRHRD
050524' 000431                                                                                    .WORD   281
050526' 004046'                                                                                   .WORD   COLTST
050530' 012726'                                                                                   .WORD   RACMG7
050532'                           150$:
050532'                             ENDSEG
050532'                                                                                   10003$:
050532' 104405                                                                                    TRAP    C$ESEG
050534'                           ENDTST
050534'                                                                                   L10165:
050534' 104401                                                                                    TRAP    C$ETST
```

```
                        .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 TAHT 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 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 'INMON' 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 'INMON' 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
                        ;
                        ;*************************************************************************

      050536'           BGNTST
```

```
050536'                                                                                 T44::
                                ;
                                ;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.
                                ;
050536'                           BGNSEG
050536' 104404                                                                                  TRAP    C$BSEG
050540' 022737  000107  000326'         CMP     #'G,MICRO               ;HAS MICROCODE MODULE 'G' BEEN LOADED
050546' 001004                          BNE     5$                      ;NO
050550' 122777  000001  127562          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
050556' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
050560' 012737  000107  000326' 5$:     MOV     #'G,MICRO               ;GO LOAD MICRO MODULE 'G'
050566' 004737  020252'                 JSR     PC,LODMIC
050572' 103002                          BCC     10$                     ;OK
050574'                                 ESCAPE  TST
050574' 104410                                                                                  TRAP    C$ESCAPE
050576' 000460                                                                                  .WORD   L10166-.
050600' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
050606' 004737  017330'                 JSR     PC,CHKDNI
050612' 103022                          BCC     20$                     ;OK
050614' 012737  000776' 000310'         MOV     #$DNI,BITNAM
050622' 012737  001275' 000312'         MOV     #$NSET,BITSTA
050630' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
050636' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
050644'                                 ERRHRD  282,TDRCNT,MSG1
050644' 104456                                                                                  TRAP    C$ERHRD
050646' 000432                                                                                  .WORD   282
050650' 004074'                                                                                 .WORD   TDRCNT
050652' 012750'                                                                                 .WORD   MSG1
050654'                                 ESCAPE  TST
050654' 104410                                                                                  TRAP    C$ESCAPE
050656' 000400                                                                                  .WORD   L10166-.
050660' 004737  017374'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
050664' 103006                          BCC     25$
050666'                                 ERRHRD  283,TDRCNT,RACMG7       ;DNI DID NOT CLEAR!
050666' 104456                                                                                  TRAP    C$ERHRD
050670' 000433                                                                                  .WORD   283
050672' 004074'                                                                                 .WORD   TDRCNT
050674' 012726'                                                                                 .WORD   RACMG7
050676'                                 ESCAPE TST
050676' 104410                                                                                  TRAP    C$ESCAPE
050700' 000356                                                                                  .WORD   L10166-.
050702'                         25$:
050702'                           ENDSEG
050702'                                                                                 10000$:
050702' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;LOAD MINIMUM BYTE COUNT INTO PCBB+0, CLEAR PCBB+2, WAIT FOR MICROMONITOR
                                ;EXECUTE MICROTEST #2 BY LOADING PCSR0 WITH A 2
                                ;WAIT FOR 'DNI'
                                ;
050704'                           BGNSEG
050704' 104404                                                                                  TRAP    C$BSEG
050706' 012737  000100  000604'         MOV     #MINBYT,PCBB            ;BEGIN WITH MINIMUM BYTE COUNT
050714' 005037  000606'                 CLR     PCBB+2                  ;THIS IS WHERE MICROCODE WILL PUT...
```

```
                                                                        ;THE TDR COUNTER VALUE
050720' 004737  017776'                  JSR     PC,CHKMON              ;WAIT FOR MICROMONITOR
050724' 103006                           BCC     35$                    ;OK
050726'                                  ERRHRD  284,TDRCNT,MSG46       ;PRINT ERROR
050726' 104456                                                                                  TRAP    C$ERHRD
050730' 000434                                                                                  .WORD   284
050732' 004074'                                                                                 .WORD   TDRCNT
050734' 016700'                                                                                 .WORD   MSG46
050736'                                  ESCAPE  TST                    ;LEAVE TEST
050736' 104410                                                                                  TRAP    C$ESCAPE
050740' 000316                                                                                  .WORD   L10166-.
050742' 012777  000002  127366  35$:     MOV     #2,@PCSR0              ;TELL T11 TO EXECUTE MICROTEST #2
050750' 012737  000176  000332'          MOV     #2*SECOND,METER        ;WAIT FOR DNI
050756' 004737  017330'                  JSR     PC,CHKDNI
050762' 103021                           BCC     40$
050764' 004737  020050'                  JSR     PC,CHKINT              ;SEE IF ANY ERROR INTERRUPTS OCCURRED
050770' 103006                           BCC     36$                    ;NO, OK
050772'                                  ERRHRD  285,TDRCNT,MSG44       ;PRINT ERROR MESSAGE
050772' 104456                                                                                  TRAP    C$ERHRD
050774' 000435                                                                                  .WORD   285
050776' 004074'                                                                                 .WORD   TDRCNT
051000' 016454'                                                                                 .WORD   MSG44
051002'                                  ESCAPE  TST
051002' 104410                                                                                  TRAP    C$ESCAPE
051004' 000252                                                                                  .WORD   L10166-.
051006' 012702  000002          36$:     MOV     #2,R2                  ;MICROTEST #2 IS HUNG
051012'                                  ERRHRD  286,TDRCNT,MSG12
051012' 104456                                                                                  TRAP    C$ERHRD
051014' 000436                                                                                  .WORD   286
051016' 004074'                                                                                 .WORD   TDRCNT
051020' 013500'                                                                                 .WORD   MSG12
051022'                                  ESCAPE  TST
051022' 104410                                                                                  TRAP    C$ESCAPE
051024' 000232                                                                                  .WORD   L10166-.

                                ;
                                ;CHECK THAT THE TRANSMIT STATUS WORD 1 HAS A NON-ZERO TDR VALUE IN IT
                                ;
051026' 013701  000606'         40$:     MOV     PCBB+2,R1              ;GET MINIMUM TDR VALUE
051032' 001004                           BNE     45$                    ;SHOULD BE A NON-ZERO VALUE
051034'                                  ERRHRD  287,TDRCNT,MSG23       ;ERROR TDR COUNTER NOT COUNTING
051034' 104456                                                                                  TRAP    C$ERHRD
051036' 000437                                                                                  .WORD   287
051040' 004074'                                                                                 .WORD   TDRCNT
051042' 014454'                                                                                 .WORD   MSG23
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
051044' 004737  017374'         45$:     JSR     PC,CLRDNI              ;GO CLEAR DNI
051050' 103004                           BCC     46$                    ;OK
051052'                                  ERRHRD  288,TDRCNT,RACMG7      ;ERROR OCCURRED PRINT ERROR MESSAGE
051052' 104456                                                                                  TRAP    C$ERHRD
051054' 000440                                                                                  .WORD   288
051056' 004074'                                                                                 .WORD   TDRCNT
051060' 012726'                                                                                 .WORD   RACMG7
051062'                         46$:
```

```
051062'                                 ENDSEG
051062'                                                                                10001$:
051062' 104405                                                                                   TRAP    C$ESEG
                                        ;
                                        ;INCREASE THE BYTE COUNT
                                        ;
051064' 005237  000604'                 47$:    INC     PCBB                    ;INCREASE BYTE COUNT BY 1
051070'                                   BGNSEG
051070' 104404                                                                                   TRAP    C$BSEG
                                        ;
                                        ;WAIT FOR THE MICROMONITOR, EXECUTE MICROTEST #2, AND WAIT FOR 'DNI'
                                        ;
051072' 004737  017776'                         JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
051076' 103006                                  BCC     50$                     ;OK
051100'                                         ERRHRD  289,TDRCNT,MSG46        ;PRINT ERROR
051100' 104456                                                                                   TRAP    C$ERHRD
051102' 000441                                                                                   .WORD   289
051104' 004074'                                                                                  .WORD   TDRCNT
051106' 016700'                                                                                  .WORD   MSG46
051110'                                         ESCAPE  TST                     ;LEAVE TEST
051110' 104410                                                                                   TRAP    C$ESCAPE
051112' 000144                                                                                   .WORD   L10166-.
051114' 012777  000002  127214  50$:    MOV     #2,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #2
051122' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
051130' 004737  017330'                         JSR     PC,CHKDNI
051134' 103021                                  BCC     55$
051136' 004737  020050'                         JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
051142' 103006                                  BCC     51$                     ;NO  OK
051144'                                         ERRHRD  290,TDRCNT,MSG44        ;PRINT ERROR MESSAGE
051144' 104456                                                                                   TRAP    C$ERHRD
051146' 000442                                                                                   .WORD   290
051150' 004074'                                                                                  .WORD   TDRCNT
051152' 016454'                                                                                  .WORD   MSG44
051154'                                         ESCAPE  TST
051154' 104410                                                                                   TRAP    C$ESCAPE
051156' 000100                                                                                   .WORD   L10166-.
051160' 012702  000002          51$:    MOV     #2,R2                   ;MICROTEST #2 IS HUNG
051164'                                         ERRHRD  291,TDRCNT,MSG12
051164' 104456                                                                                   TRAP    C$ERHRD
051166' 000443                                                                                   .WORD   291
051170' 004074'                                                                                  .WORD   TDRCNT
051172' 013500'                                                                                  .WORD   MSG12
051174'                                         ESCAPE  TST
051174' 104410                                                                                   TRAP    C$ESCAPE
051176' 000060                                                                                   .WORD   L10166-.
                                        ;
                                        ;VERIFY THAT THE TRANSMIT STATUS WORD 1 TDR VALUE IS LARGER THAN THE PREVIOUS
                                        ;VALUE
                                        ;
051200' 013702  000606'         55$:    MOV     PCBB+2,R2               ;GET NEW TDR VALUE
051204' 020201                                  CMP     R2,R1                   ;IS TDR GETTING LARGER?
051206' 101006                                  BHI     60$                     ;YES
051210'                                         ERRHRD  292,TDRCNT,MSG23        ;NO ERROR
051210' 104456                                                                                   TRAP    C$ERHRD
051212' 000444                                                                                   .WORD   292
051214' 004074'                                                                                  .WORD   TDRCNT
```

```
051216' 014454'                                                                           .WORD   MSG23
051220'                                 ESCAPE  TST
051220' 104410                                                                            TRAP    C$ESCAPE
051222' 000034                                                                            .WORD   L10166-.
                                 ;
                                 ;WRITE '1' TO CLEAR 'DNI'
                                 ;
051224' 004737  017374'          60$:   JSR     PC,CLRDNI               ;GO CLEAR DNI
051230' 103004                          BCC     65$                     ;OK
051232'                                 ERRHRD  293,TDRCNT,RACMG7       ;ERROR OCCURRED PRINT ERROR MESSAGE
051232' 104456                                                                            TRAP    C$ERHRD
051234' 000445                                                                            .WORD   293
051236' 004074'                                                                           .WORD   TDRCNT
051240' 012726'                                                                           .WORD   RACMG7
051242'                          65$:
051242'                            ENDSEG
051242'                                                                          10002$:
051242' 104405                                                                            TRAP    C$ESEG
                                 ;
                                 ;CONTINUE THE LOOP UNTIL THE BYTE COUNT IS TOO LARGE FOR THE COUNTER MODULUS
                                 ;
051244' 010201                          MOV     R2,R1                   ;HOLD NEW TDR VALUE FOR LATER COMPARE
051246' 023727  000606' 000164          CMP     PCBB+2,#116.            ;HAVE WE DONE ENOUGH PACKETS?
051254' 103703                          BLO     47$                     ;NO KEEP GOING
051256'                          ENDTST
051256'                                                                          L10166:
051256' 104401                                                                            TRAP    C$ETST
```

```
                                        .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 RFTRANSMIT.
                                        ;
                                        ;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 WIAT 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 ACCUMULATE' 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 FFOM THE PCBB AND STORED IN A TABLE. 'HE TABLE WILL BE SCANNED TO
                                        ;VERIFY THA' 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.
                                        ;*****************************************************************************

        051260'                         BGNTST
        051260'                                                                 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'.
```

```
                                         ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                         ;
051260'                                   BGNSEG
051260' 104404                                                                                    TRAP     C$BSEG
051262' 022737  000107  000326'                 CMP     #'G,MICRO             ;HAS MICROCODE MODULE 'G' BEEN LOADED
051270' 001004                                  BNE     5$                    ;NO
051272' 122777  000001  127040                  CMPB    #INMON,@PCSR1         ;YES, IS THE MICROMONITOR ACTIVE?
051300' 001440                                  BEQ     20$                   ;YES SKIP LOADING THE MICROMODULE
051302' 012737  000107  000326' 5$:             MOV     #'G,MICRO             ;GO LOAD MICRO MODULE 'G'
051310' 004737  020252'                         JSR     PC,LODMIC
051314' 103002                                  BCC     10$                   ;OK
051316'                                         ESCAPE  TST
051316' 104410                                                                                    TRAP     C$ESCAPE
051320' 000352                                                                                    .WORD    L10167-.
051322' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER       ;WAIT FOR THE MICROMONITOR
051330' 004737  017330'                         JSR     PC,CHKDNI
051334' 103022                                  BCC     20$                   ;OK
051336' 012737  000776' 000310'                 MOV     #$DNI,BITNAM
051344' 012737  001275' 000312'                 MOV     #$NSET,BITSTA
051352' 012737  001340' 000314'                 MOV     #$AFTER,PWHEN
051360' 012737  001355' 000316'                 MOV     #$GTCMD,PCOMND
051366'                                         ERRHRD  294,RETLOG,MSG1
051366' 104456                                                                                    TRAP     C$ERHRD
051370' 000446                                                                                    .WORD    294
051372' 004124'                                                                                   .WORD    RETLOG
051374' 012750'                                                                                   .WORD    MSG1
051376'                                         ESCAPE  TST
051376' 104410                                                                                    TRAP     C$ESCAPE
051400' 000272                                                                                    .WORD    L10167-.
051402' 004737  017374'         20$:            JSR     PC,CLRDNI             ;CLEAR DNI BIT
051406' 103006                                  BCC     25$
051410'                                         ERRHRD  295,RETLOG,RACMG7     ;DNI DID NOT CLEAR!
051410' 104456                                                                                    TRAP     C$ERHRD
051412' 000447                                                                                    .WORD    295
051414' 004124'                                                                                   .WORD    RETLOG
051416' 012726'                                                                                   .WORD    RACMG7
051420'                                         ESCAPE TST
051420' 104410                                                                                    TRAP     C$ESCAPE
051422' 000250                                                                                    .WORD    L10167-.
051424'                         25$:
051424'                                   ENDSEG
051424'                                                                                10000$:
051424' 104405                                                                                    TRAP     C$ESEG
                                         ;
                                         ;WRITE A BYTE COUT TO PCBB+0, CLEAR PCBB+2, GET POINTER TO TOP OF TABLE
                                         ;INITALIZE A COUNTER TO 16
                                         ;
051426' 012737  000100  000604'                 MOV     #MINBYT,PCBB          ;BEGIN WITH MINIMUM BYTE COUNT
051434' 005037  000606'                         CLR     PCBB+2                ;THIS IS WHERE MICROCODE WILL PUT...
                                                                              ;THE RETRY WAIT INTERVAL
051440' 012702  000624'                         MOV     #CNTTAB,R2            ;GET POINTER TO WAIT INTERVAL STORAGE AREA
051444' 012701  000020                          MOV     #16.,R1               ;NUMBER OF VALUES TO STORE
                                         ;
                                         ;WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR. EXECUTE MICROTEST #3 BY
                                         ;LOADING THE COMMAND FIELD OF PCSRO WITH A 3. WAIT FOR 'DNI'
                                         ;
```

```
051450'                                BGNSEG
051450' 104404                                                                                 TRAP    C$BSEG
051452' 004737  017776'                        JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
051456' 103006                                 BCC     35$                     ;OK
051460'                                        ERRHRD  296,RETLOG,MSG46        ;PRINT ERROR
051460' 104456                                                                                 TRAP    C$ERHRD
051462' 000450                                                                                 .WORD   296
051464' 004124'                                                                                .WORD   RETLOG
051466' 016700'                                                                                .WORD   MSG46
051470'                                        ESCAPE  TST                     ;LEAVE TEST
051470' 104410                                                                                 TRAP    C$ESCAPE
051472' 000200                                                                                 .WORD   L10167-.
051474' 012777  000003  126634  35$:           MOV     #3,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #3
051502' 012737  000770  000332'                MOV     #10*SECOND,METER        ;WAIT FOR DNI
051510' 004737  017330'                        JSR     PC,CHKDNI
051514' 103021                                 BCC     40$
051516' 004737  020050'                        JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
051522' 103006                                 BCC     36$                     ;NO, OK
051524'                                        ERRHRD  297,RETLOG,MSG44        ;PRINT ERROR MESSAGE
051524' 104456                                                                                 TRAP    C$ERHRD
051526' 000451                                                                                 .WORD   297
051530' 004124'                                                                                .WORD   RETLOG
051532' 016454'                                                                                .WORD   MSG44
051534'                                        ESCAPE  TST
051534' 104410                                                                                 TRAP    C$ESCAPE
051536' 000134                                                                                 .WORD   L10167-.
051540' 012702  000003          36$:           MOV     #3,R2                   ;MICROTEST #3 IS HUNG
051544'                                        ERRHRD  298,RETLOG,MSG12
051544' 104456                                                                                 TRAP    C$ERHRD
051546' 000452                                                                                 .WORD   298
051550' 004124'                                                                                .WORD   RETLOG
051552' 013500'                                                                                .WORD   MSG12
051554'                                        ESCAPE  TST
051554' 104410                                                                                 TRAP    C$ESCAPE
051556' 000114                                                                                 .WORD   L10167-.
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
051560' 004737  017374'         40$:           JSR     PC,CLRDNI               ;GO CLEAR DNI
051564' 103006                                 BCC     45$                     ;OK
051566'                                        ERRHRD  299,RETLOG,RACMG7       ;ERROR
051566' 104456                                                                                 TRAP    C$ERHRD
051570' 000453                                                                                 .WORD   299
051572' 004124'                                                                                .WORD   RETLOG
051574' 012726'                                                                                .WORD   RACMG7
051576'                                        ESCAPE TST                      ;LEAVE
051576' 104410                                                                                 TRAP    C$ESCAPE
051600' 000072                                                                                 .WORD   L10167-.
                                ;
                                ;STORE THE VALUE FORM PCBB+2 IN THE TABLE AND BUMP THE POINTER TO NEXT ENTRY
                                ;REPEAT THE TEST UNTIL ALL 16 ENTRIES ARE OBTAINED
                                ;
051602' 013722  000606'         45$:           MOV     PCBB+2,(R2)+            ;STORE COUNTER VALUE IN THE TABLE
051606' 005301                                 DEC     R1                      ;HAVE WE STORED 16 VALUES?
051610' 001331                                 BNE     35$                     ;NOT YET
                                ;
```

```
                                        ;TREAT THE TABLE AS 7 GROUPS OF 10 ENTRIES. VERIFY THAT ALL 10 ENTRIES OF EACH
                                        ;GROUP ARE NOT THE SAME.
                                        ;
051612' 012701  000624'                         MOV     #CNTTAB,R1              ;POINT TO TOP OF TABLE
051616' 010102                          50$:    MOV     R1,R2                   ;GET POINTER TO FIRST ELEMENT IN THIS
                                                                                ;GROUP
051620' 010204                                  MOV     R2,R4                   ;GET POINTER TO SECOND ELEMENT IN THIS
                                                                                ;GROUP
051622' 062704  000002                          ADD     #2,R4
051626' 012703  000007                          MOV     #7.,R3                  ;NUMBER OF GROUPS OF 10 TO COMPARE
051632' 022224                          55$:    CMP     (R2)+,(R4)+             ;IS THIS PAIR THE SAME?
051634' 001010                                  BNE     60$                     ;NO, START A NEW GROUP TO CHECK
051636' 005303                                  DEC     R3                      ;HAVE WE CHECKED ALL 10 VALUESIN THIS
                                                                                ;GROUP?
051640' 001374                                  BNE     55$                     ;NOT YET
051642'                                         ERRHRD  300,RETLOG,MSG24        ;YES, PRINT ERROR MESSAGE AND DUMP TABLE
051642' 104456                                                                                          TRAP    C$ERHRD
051644' 000454                                                                                          .WORD   300
051646' 004124'                                                                                         .WORD   RETLOG
051650' 014476'                                                                                         .WORD   MSG24
051652'                                         ESCAPE  TST
051652' 104410                                                                                          TRAP    C$ESCAPE
051654' 000016                                                                                          .WORD   L10167-.
051656' 062701  000002                  60$:    ADD     #2,R1                   ;POINT TO NEXT GROUP OF 10 ELEMENTS
051662' 020127  000642'                         CMP     R1,#CNTTAB+16           ;CHECKED ALL GROUPS OF 10?
051666' 001353                                  BNE     50$                     ;NOT YET
051670'                                   ENDSEG
051670'                                                                                         10001$:
051670' 104405                                                                                          TRAP    C$ESEG
051672'                                 ENDTST
051672'                                                                                         L10167:
051672' 104401                                                                                          TRAP    C$ETST
```

```

                                 .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 #
                                 ;
                                 ;*******************************************************************

051674'                          BGNTST
051674'                                                                                  T46::
051674' 005737  000672'                  TST     FRSTIM                  ; RUN THIS TEST ?
051700' 001002                           BNE     5$                      ; YES
051702'                                  EXIT    TST                     ; NO, EXIT
051702' 104432                                                                                   TRAP    C$EXIT
051704' 001042                                                                                   .WORD   L10170-.
                                 ;
051706' 004737  020104'          5$:     JSR     PC,REUNA                ;GO RESET UNA
051712' 103002                           BCC     20$                     ;OK
051714'                                  ESCAPE  TST                     ; ABORT TEST
051714' 104410                                                                                   TRAP    C$ESCAPE
051716' 001030                                                                                   .WORD   L10170-.
                                 ;
051720' 012777  000604' 126414   20$:    MOV     #PCBB,@PCSR2            ;LOAD PCSR2 WITH PORT CONTROL BLOCK ADR
051726' 005077  126412                   CLR     @PCSR3                  ;LOAD PCSR3 WITH 0
051732' 012777  000001  126376           MOV     #GETPCB,@PCSR0          ; ISSUE GET PCBB PORT COMMAND
051740' 012737  000077  000332'          MOV     #1*SECOND,METER         ;PUT SOME TIME ON THE METER
051746' 004737  017330'                  JSR     PC,CHKDNI               ; DNI?
051752' 103022                           BCC     40$                     ; YES
                                                                         ;ERROR DNI FAILED TO SET!
                                                                         ;FORMAT ERROR MESSAGE
051754' 012737  000776' 000310'          MOV     #$DNI,BITNAM
051762' 012737  001275' 000312'          MOV     #$NSET,BITSTA
051770' 012737  001340' 000314'          MOV     #$AFTER,PWHEN
051776' 012737  001346' 000316'          MOV     #$GTPCB,PCOMND
052004'                                  ERRHRD  301,PRTPAR,MSG1
052004' 104456                                                                                   TRAP    C$ERHRD
052006' 000455                                                                                   .WORD   301
052010' 004154'                                                                                  .WORD   PRTPAR
052012' 012750'                                                                                  .WORD   MSG1
052014'                                  ESCAPE  TST                     ; AND ABORT TEST
052014' 104410                                                                                   TRAP    C$ESCAPE
052016' 000730                                                                                   .WORD   L10170-.
                                 ;
052020' 004737  017374'          40$:    JSR     PC,CLRDNI               ; WRITE ONE TO CLEAR DNI
                                                                         ; ERROR ?
052024' 103006                           BCC     50$                     ; NO
```

```
                                                                        ;ERROR DNI FAILED TO CLEAR!
052026'                                 ERRHRD  302,PRTPAR,RACMG7
052026' 104456                                                                                  TRAP    C$ERHRD
052030' 000456                                                                                  .WORD   302
052032' 004154'                                                                                 .WORD   PRTPAR
052034' 012726'                                                                                 .WORD   RACMG7
052036'                                 ESCAPE  TST                     ; AND ABORT TEST
052036' 104410                                                                                  TRAP    C$ESCAPE
052040' 000706                                                                                  .WORD   L10170-.
                                ;
                                ;READ DEFAULT PHYSICAL ADDRESS
                                ;
052042' 012737  000002  000604' 50$:    MOV     #RDPA,PCBB              ;LOAD PCBB WITH READ DEFAULT PHY ADR
052050' 012737  000176  000332'         MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
052056' 012777  000002  126252          MOV     #GETCMD,@PCSR0          ; ISSUE GET_CMD PORT COMMAND
052064' 004737  017330'                 JSR     PC,CHKDNI               ; DNI ?
052070' 103022                          BCC     60$                     ; YES
                                                                        ;ERROR DNI FAILED TO SET!
052072' 012737  000776' 000310'         MOV     #$DNI,BITNAM
052100' 012737  001275' 000312'         MOV     #$NSET,BITSTA
052106' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
052114' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
052122'                                 ERRHRD  303,PRTPAR,MSG1 ;PRINT ERROR MESSAGE
052122' 104456                                                                                  TRAP    C$ERHRD
052124' 000457                                                                                  .WORD   303
052126' 004154'                                                                                 .WORD   PRTPAR
052130' 012750'                                                                                 .WORD   MSG1
052132'                                 ESCAPE  TST                     ; AND ABORT TEST
052132' 104410                                                                                  TRAP    C$ESCAPE
052134' 000612                                                                                  .WORD   L10170-.
                                ;
052136' 004737  017374'         60$:    JSR     PC,CLRDNI               ; WRITE ONE TO CLEAR DNI
                                                                        ; ERROR ?
052142' 103006                          BCC     70$                     ; NO
                                                                        ;ERROR DNI FAILED TO CLEAR!
052144'                                 ERRHRD  304,PRTPAR,RACMG7       ;PRINT ERROR MESSAGE
052144' 104456                                                                                  TRAP    C$ERHRD
052146' 000460                                                                                  .WORD   304
052150' 004154'                                                                                 .WORD   PRTPAR
052152' 012726'                                                                                 .WORD   RACMG7
052154'                                 ESCAPE  TST                     ; AND ABORT TEST
052154' 104410                                                                                  TRAP    C$ESCAPE
052156' 000570                                                                                  .WORD   L10170-.
                                ;
                                ;MOVE DEFAULT PHYSICAL ADDRESS FROM PCBB -> DPA
                                ;
052160' 013737  000606' 052750' 70$:    MOV     PCBB+2,DPA
052166' 013737  000610' 052752'         MOV     PCBB+4,DPA+2
052174' 013737  000612' 052754'         MOV     PCBB+6,DPA+4
                                ;
                                ;LOAD ASCII MESSAGE (DEFADR)
                                ;
052202' 004737  017130'                 JSR     PC,HEXDPA               ; CONVERT TO ASCII HEX
                                ;
                                ;READ MICROCODE REVISION
                                ;
```

```
052206' 012737  000016  000604' 100$:   MOV     #RPS,PCBB               ;LOAD PCBB WITH READ PORT STATUS FUNCTION
052214' 012737  000176  000332'         MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
052222' 012777  000002  126106          MOV     #GETCMD,@PCSRO          ; ISSUE GET_CMD PORT COMMAND
052230' 004737  017330'                 JSR     PC,CHKDNI               ; DNI ?
052234' 103022                          BCC     110$                    ; YES
                                                                        ;ERROR DNI FAILED TO SET
052236' 012737  000776' 000310'         MOV     #$DNI,BITNAM
052244' 012737  001275' 000312'         MOV     #$NSET,BITSTA
052252' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
052260' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
052266'                                 ERRHRD  305,PRTPAR,MSG1 ;PRINT ERROR MESSAGE
052266' 104456                                                                          TRAP    C$ERHRD
052270' 000461                                                                          .WORD   305
052272' 004154'                                                                         .WORD   PRTPAR
052274' 012750'                                                                         .WORD   MSG1
052276'                                 ESCAPE  TST                     ; AND ABORT TEST
052276' 104410                                                                          TRAP    C$ESCAPE
052300' 000446                                                                          .WORD   L10170-.
                                ;
052302' 004737  017374'         110$:   JSR     PC,CLRDNI               ; WRITE ONE TO CLEAR DNI
                                                                        ; ERROR ?
052306' 103006                          BCC     120$                    ; NO
                                                                        ;ERROR DNI FAILED TO SET
052310'                                 ERRHRD  306,PRTPAR,RACMG7       ;PRINT ERROR MESSAGE
052310' 104456                                                                          TRAP    C$ERHRD
052312' 000462                                                                          .WORD   306
052314' 004154'                                                                         .WORD   PRTPAR
052316' 012726'                                                                         .WORD   RACMG7
052320'                                 ESCAPE  TST                     ; AND ABORT TEST
052320' 104410                                                                          TRAP    C$ESCAPE
052322' 000424                                                                          .WORD   L10170-.
                                ;
                                ;MOVE MICROCODE REVISION FROM PCBB -> RREV
                                ;
052324' 013737  000606' 052756' 120$:   MOV     PCBB+2,RREV
052332' 042737  177700  052756'         BIC     #177700,RREV            ; MASK RREV
                                ;
                                ;READ SWITCH PACK
                                ;
052340' 012737  000020  000604' 130$:   MOV     #DIM,PCBB               ;LOAD DUMP INTERNAL MEMORY FUNCTION
052346' 012737  000614' 000606'         MOV     #UDBB,PCBB+2            ;GIVE THE UNIBUS DATA BLOCK BASE ADR
052354' 005037  000610'                 CLR     PCBB+4
052360' 012737  000002  000614'         MOV     #2,UDBB                 ;SETUP TO LOAD 2 BYTES
052366' 012737  052760' 000616'         MOV     #SWPACK,UDBB+2          ; LOAD ADDRESS
052374' 005037  000620'                 CLR     UDBB+4
052400' 013737  000334' 000622'         MOV     SWADDR,UDBB+6           ; LOAD INTERNAL ADDRESS
052406' 012737  000176  000332'         MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
052414' 012777  000002  125714          MOV     #GETCMD,@PCSRO          ; ISSUE GET COMMAND PORT COMMAND
052422' 004737  017330'                 JSR     PC,CHKDNI               ; DNI ?
052426' 103022                          BCC     140$                    ; YES
                                                                        ;ERROR DNI FAILED TO SET
052430' 012737  000776' 000310'         MOV     #$DNI,BITNAM
052436' 012737  001275' 000312'         MOV     #$NSET,BITSTA
052444' 012737  001340' 000314'         MOV     #$AFTER,PWHEN
052452' 012737  001355' 000316'         MOV     #$GTCMD,PCOMND
052460'                                 ERRHRD  307,PRTPAR,MSG1
```

```
052460' 104456                                                                     TRAP    C$ERHRD
052462' 000463                                                                     .WORD   307
052464' 004154'                                                                    .WORD   PRTPAR
052466' 012750'                                                                    .WORD   MSG1
052470'                            ESCAPE  TST                 ; AND ABORT TEST
052470' 104410                                                                     TRAP    C$ESCAPE
052472' 000254                                                                     .WORD   L10170-.
                          ;
052474' 004737  017374'   140$:    JSR     PC,CLRDNI           ; WRITE ONE TO CLEAR DNI
                                                               ; ERROR ?
052500' 103006                     BCC     150$                ; NO
052502'                            ERRHRD  308,PRTPAR,RACMG7
052502' 104456                                                                     TRAP    C$ERHRD
052504' 000464                                                                     .WORD   308
052506' 004154'                                                                    .WORD   PRTPAR
052510' 012726'                                                                    .WORD   RACMG7
052512'                            ESCAPE  TST                 ; AND ABORT TEST
052512' 104410                                                                     TRAP    C$ESCAPE
052514' 000232                                                                     .WORD   L10170-.
                          ;
                          ;GET SWITCH PACK INFO READY TO PRINT
                          ;
052516' 013704  052760'   150$:    MOV     SWPACK,R4           ; SWITCH PACK -> R4
052522' 042704  167777             BIC     #167777,R4          ; MASK BIT 12
052526' 012700  000013             MOV     #11.,R0             ; SHIFT BIT FOR INDEX
052532' 006204            160$:    ASR     R4
052534' 005300                     DEC     R0
052536' 001375                     BNE     160$
052540' 062704  053100'            ADD     #LPTBL,R4           ; INDEX INTO LOOP TABLE
052544' 011437  052762'            MOV     (R4),LPMSG          ; LOAD INTO LOOP MESSAGE
052550' 013704  052760'   170$:    MOV     SWPACK,R4           ; SWITCH PACK -> R4
052554' 042704  171777             BIC     #171777,R4          ; MASK BITS 10 AND 11
052560' 012700  000011             MOV     #9.,R0              ; SHIFT BITS FOR INDEX
052564' 006204            180$:    ASR     R4
052566' 005300                     DEC     R0
052570' 001375                     BNE     180$
052572' 062704  053104'            ADD     #BTTBL,R4           ; INDEX INTO BOOT TABLE
052576' 011437  052764'            MOV     (R4),BTMSG          ; LOAD INTO BOOT MESSAGE
                          ;
                          ; PRINT
                          ;
052602'                            PRINTB  #FORM28,#DEFHDR     ; PRINT DEFAULT PHYSICAL ADDPESS
052602' 012746  052770'                                                            MOV     #DEFHDR,-(SP)
052606' 012746  006331'                                                            MOV     #FORM28,-(SP)
052612' 012746  000002                                                             MOV     #2,-(SP)
052616' 010600                                                                     MOV     SP,R0
052620' 104414                                                                     TRAP    C$PNTB
052622' 062706  000006                                                             ADD     #6,SP
052626'                            PRINTB  #FORM64,RREV        ; PRINT MICROCODE REV
052626' 013746  052756'                                                            MOV     RREV,-(SP)
052632' 012746  011066'                                                            MOV     #FORM64,-(SP)
052636' 012746  000002                                                             MOV     #2,-(SP)
052642' 010600                                                                     MOV     SP,R0
052644' 104414                                                                     TRAP    C$PNTB
052646' 062706  000006                                                             ADD     #6,SP
052652'                            PRINTB  #FORM28,#SWHDR      ; PRINT SWITCH PACK HEADER
```

```
052652' 012746  053114'                                                                 MOV     #SWHDR,-(SP)
052656' 012746  006331'                                                                 MOV     #FORM28,-(SP)
052662' 012746  000002                                                                  MOV     #2,-(SP)
052666' 010600                                                                          MOV     SP,R0
052670' 104414                                                                          TRAP    C$PNTB
052672' 062706  000006                                                                  ADD     #6,SP
052676'                          PRINTB  #FORM28,LPMSG           ; PRINT LOOPBACK MESSAGE
052676' 013746  052762'                                                                 MOV     LPMSG,-(SP)
052702' 012746  006331'                                                                 MOV     #FORM28,-(SP)
052706' 012746  000002                                                                  MOV     #2,-(SP)
052712' 010600                                                                          MOV     SP,R0
052714' 104414                                                                          TRAP    C$PNTB
052716' 062706  000006                                                                  ADD     #6,SP
052722'                          PRINTB  #FORM28,BTMSG           ; PRINT BOOT MESSAGE
052722' 013746  052764'                                                                 MOV     BTMSG,-(SP)
052726' 012746  006331'                                                                 MOV     #FORM28,-(SP)
052732' 012746  000002                                                                  MOV     #2,-(SP)
052736' 010600                                                                          MOV     SP,R0
052740' 104414                                                                          TRAP    C$PNTB
052742' 062706  000006                                                                  ADD     #6,SP
                        ;
052746'                 250$:
052746'                          ENDTST
052746'                                                                         L10170:
052746' 104401                                                                          TRAP    C$ETST
```

```
                                         ;LOCAL STORAGE FOR TEST 41
052750' 000000                           DPA::           .WORD   0               ; DEFAULT PHYSICAL ADDRESS (15:00)
052752' 000000                                           .WORD   0               ; DEFAULT PHYSICAL ADDRESS (31:16)
052754' 000000                                           .WORD   0               ; DEFAULT PHYSICAL ADDRESS (47:32)
                                         ;
052756' 000000                           RREV::          .WORD   0               ; MICROCODE REVISION
                                         ;
052760' 000000                           SWPACK::        .WORD   0               ; SWITCH PACK CONTENTS
052762' 000000                           LPMSG::         .WORD   0               ; LOOPBACK MESSAGE ADDRESS
052764' 000000                           BTMSG::         .WORD   0               ; BOOT MESSAGE ADDRESS
                                         ;
052766'    000                           HEXDAT::        .BYTE   0               ; HEX DATA FOR CONVERSION
052767'    000                           HEXVAL::        .BYTE   0               ; ASCII HEX VALUE
                                         ;
052770' 005015  052105  042510           DEFHDR::        .ASCII  <15><12>/ETHERNET DEFAULT ADDRESS (HEX):  /
052776' 047122  052105  042040
053004' 043105  052501  052114
053012' 040440  042104  042522
053020' 051523  024040  042510
053026' 024530  020072     040
053033'    040     040                   DEFADR::        .ASCII  /  /
053035'    055                                           .ASCII  /-/
053036' 020040                                           .ASCII  /  /
053040'    055                                           .ASCII  /-/
053041'    040     040                                   .ASCII  /  /
053043'    055                                           .ASCII  /-/
053044' 020040                                           .ASCII  /  /
053046'    055                                           .ASCII  /-/
053047'    040     040                                   .ASCII  /  /
053051'    055                                           .ASCII  /-/
053052' 020040                                           .ASCII  /  /
053054' 005015     000                                   .ASCIZ  <15><12>
                                         ;
053057'    060                           HEXTBL::        .ASCII  /0/
053060'    061                                           .ASCII  /1/
053061'    062                                           .ASCII  /2/
053062'    063                                           .ASCII  /3/
053063'    064                                           .ASCII  /4/
053064'    065                                           .ASCII  /5/
053065'    066                                           .ASCII  /6/
053066'    067                                           .ASCII  /7/
053067'    070                                           .ASCII  /8/
053070'    071                                           .ASCII  /9/
053071'    101                                           .ASCII  /A/
053072'    102                                           .ASCII  /B/
053073'    103                                           .ASCII  /C/
053074'    104                                           .ASCII  /D/
053075'    105                                           .ASCII  /E/
053076'    106                                           .ASCII  /F/
        053100'                                          .EVEN


                                         ;
                                         ;LOOP MESSAGE TABLE
053100' 053146'                          LPTBL::         .WORD   LPMSG0
053102' 053205'                                          .WORD   LPMSG1
                                         ;BOOT MESSAGE TABLE
```

```
13014  053104' 053243'                          BTTBL::          .WORD   BTMSG0
13015  053106' 053301'                                           .WORD   BTMSG1
13016  053110' 053334'                                           .WORD   BTMSG2
13017  053112' 053400'                                           .WORD   BTMSG3
13018                                           ;ASCII MESSAGES
13019  053114' 005015  053523  052111  SWHDR::          .ASCII  <15><12>/SWITCH PACK SET FOR :/
13020  053122' 044103  050040  041501
13021  053130' 020113  042523  020124
13022  053136' 047506  020122     072
13023  053143'    015  000012                            .ASCIZ  <15><12>
13024  053146' 020040  020040  051440  LPMSG0::         .ASCII  /      SELF TEST LOOP DISABLED/
13025  053154' 046105  020106  042524
13026  053162' 052123  046040  047517
13027  053170' 020120  044504  040523
13028  053176' 046102  042105
13029  053202' 005015     000                            .ASCIZ  <15><12>
13030  053205'    040  020040  020040  LPMSG1::         .ASCII  /      SELF TEST LOOP ENABLED/
13031  053212' 042523  043114  052040
13032  053220' 051505  020124  047514
13033  053226' 050117  042440  040516
13034  053234' 046102  042105
13035  053240' 005015     000                            .ASCIZ  <15><12>
13036  053243'    040  020040  020040  BTMSG0::         .ASCII  /      NO REMOTE BOOT ENABLED/
13037  053250' 047516  051040  046505
13038  053256' 052117  020105  047502
13039  053264' 052117  042440  040516
13040  053272' 046102  042105
13041  053276' 005015     000                            .ASCIZ  <15><12>
13042  053301'    040  020040  020040  BTMSG1::         .ASCII  /      REMOTE BOOT ENABLED/
13043  053306' 042522  047515  042524
13044  053314' 041040  047517  020124
13045  053322' 047105  041101  042514
13046  053330'    104
13047  053331'    015  000012                            .ASCIZ  <15><12>
13048  053334' 020040  020040  051040  BTMSG2::         .ASCII  /      REMOTE BOOT ENABLED WITH ROM/
13049  053342' 046505  052117  020105
13050  053350' 047502  052117  042440
13051  053356' 040516  046102  042105
13052  053364' 053440  052111  020110
13053  053372' 047522     115
13054  053375'    015  000012                            .ASCIZ  <15><12>
13055  053400' 020040  020040  051040  BTMSG3::         .ASCII  /      REMOTE BOOT AND POWER UP BOOT BOTH ENABLED/
13056  053406' 046505  052117  020105
13057  053414' 047502  052117  040440
13058  053422' 042116  050040  053517
13059  053430' 051105  052440  020120
13060  053436' 047502  052117  041040
13061  053444' 052117  020110  047105
13062  053452' 041101  042514     104
13063  053457'    015  000012                            .ASCIZ  <15><12>
13064                                                    .EVEN
```

```
                                        .TITLE PARAMETER CODING

                                        .SBTTL  HARDWARE PARAMETER CODING SECTION

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



053462'                                         BGNHRD
053462' 000010                                                                                  .WORD L10171-L$HARD/2
053464'                                                                                 L$HARD::
053464'                                         GPRMA   ASKCSR,0,0,160000,177776,NO     ;FIRST P-TABLE QUESTION
053464' 000021                                                                                  .WORD   T$CODE
053466' 053504'                                                                                 .WORD   ASKCSR
053470' 160000                                                                                  .WORD   T$LOLIM
053472' 177776                                                                                  .WORD   T$HILIM
053474'                                         GPRMA   ASKVEC,2,0,0,776,NO             ;SECOND P-TABLE QUESTION
053474' 001021                                                                                  .WORD   T$CODE
053476' 053537'                                                                                 .WORD   ASKVEC
053500' 000000                                                                                  .WORD   T$LOLIM
053502' 000776                                                                                  .WORD   T$HILIM
053504'                                         ENDHRD

                                                                                                .EVEN
053504'                                                                                 L10171:


053504' 044127  052101  044440  ASKCSR: .ASCIZ  /WHAT IS THE PCSRO ADDRESS?/
053512' 020123  044124  020105
053520' 041520  051123  020060
053526' 042101  051104  051505
053534' 037523     000
053537'    127  040510  020124  ASKVEC: .ASCIZ  /WHAT IS THE VECTOR ADDRESS?/
053544' 051511  052040  042510
053552' 053040  041505  047524
053560' 020122  042101  051104
053566' 051505  037523     000
        053574'                 .EVEN
```

```
13109  053574' 000100           $PATCH::.BLKW   100
13110
13111
13112          000000'                  .CSECT  MICRA
13113          000000'                  .CSECT  MICRB
13114          000000'                  .CSECT  MICRC
13115          000000'                  .CSECT  MICRD
13116          000000'                  .CSECT  MICRE
13117          000000'                  .CSECT  MICRF
13118          000000'                  .CSECT  MICRG
13120  000000'                          .CSECT  NOMORE
13121          000001                   ENDMOD
                                .END
```

```
ADR   = 000020 G       1632#
ADRERR= 000001 G       1667#   6852    7468
ASKCSR  053504R   002 13085   13098#
ASKVEC  053537R   002 13090   13103#
ASSEMB= 000010         1389
BITNAM  000310RG  002  1786#   2966    3000    3097    3265    3669    4309*   4316*   4323*   4330*   4337*   4344*   4351*
                       4353    4463*   4470    5300*   5329*   5352*   5407*   5432*   5451*   5652*   5723*   6179*   6245*
                       6376*   6419*   6480*   6606*   6653*   6701*   6780*   6927*   7057*   7133*   7134*   7172*   7173*
                       7238*   7384*   7549*   7708*   7858*   7996*   8134*   8282*   8444*   8597*   8774*   8856*   8869*
                       8878*   3890*   8899*   8908*   9001*   9170*   9343*   9508*   9697*   9860*  10099*  10322*  10477*
                      10637*  10804*  10976*  11144*  11333*  11541*  11756*  11992*  12335*  12582*  12749*  12783*  12827*
                      12872*
BITNUM  000306RG  002  1785#   2904    4973*   4990*   5076*   5093*   5110*   5127*   5144*   7101*   7131    7183*
BITSTA  000312RG  002  1787#   2903    2965    2999    4464*   4469    4972*   5075*   5301*   5330*   5345*   5398*   5431*
                       5450*   5653*   5724*   6180*   6246*   6377*   6420*   6481*   6607*   6654*   6702*   6781*   6928*
                       7058*   7099*   7239*   7385*   7550*   7709*   7859*   7997*   8135*   8283*   8445*   8598*   8775*
                       9002*   9171*   9344*   9509*   9698*   9861*  10100*  10323*  10478*  10638*  10805*  10977*  11145*
                      11334*  11542*  11757*  11993*  12336*  12583*  12750*  12784*  12828*  12873*
BIT0  = 000001 G       1605#   5344    5496    7628
BIT00 = 000001 G       1594#   1605    1699    1702    1723    1725    1727    1728    1730    5533
BIT01 = 000002 G       1593#   1604    1700    1701    1702    1723    1726    1727    1729    1730
BIT02 = 000004 G       1592#   1603    1701    1723    1728    1729    1730
BIT03 = 000010 G       1591#   1602    1712
BIT04 = 000020 G       1590#   1601
BIT05 = 000040 G       1589#   1600    1688
BIT06 = 000100 G       1588#   1599    1687
BIT07 = 000200 G       1587#   1598    1686    1710
BIT08 = 000400 G       1586#   1597    1685    1736
BIT09 = 001000 G       1585#   1596    1735
BIT1  = 000002 G       1604#
BIT10 = 002000 G       1584#   1683    1715    1719    7100   11452   12083   12155   12226   12234
BIT11 = 004000 G       1583#   1682    1716    1719   11227   12075   12081   12153   12232
BIT12 = 010000 G       1582#   1681    1696    1717    1719   12079   12148   12151   12230
BIT13 = 020000 G       1581#   1680    1695    1718    1719    8867    8888
BIT14 = 040000 G       1580#   1679    1694    1706    1734    5128    8897   11235   12077   12149   12225   12228
BIT15 = 100000 G       1579#   1678    1693    1705    1733    5145    7783    8876    8906   12025
BIT2  = 000004 G       1603#   7783   12025
BIT3  = 000010 G       1602#   5400
BIT4  = 000020 G       1601#   4974    5077   12025
BIT5  = 000040 G       1600#   5094
BIT6  = 000100 G       1599#   1660    5111   11374
BIT7  = 000200 G       1598#
BIT8  = 000400 G       1597#   1713    1719
BIT9  = 001000 G       1596#   1714    1719    4991
BNAMT0  000356RG  002  1810#   5343    7133    7172
BNAMT1  000416RG  002  1827#   5396
BNAMT2  000456RG  002  1845#   8851
BOE   = 000400 G       1636#
BTMSG   052764RG  002 12914*  12947   12967#
BTMSG0  053243RG  002 13014   13036#
BTMSG1  053301RG  002 13015   13042#
BTMSG2  053334RG  002 13016   13048#
BTMSG3  053400RG  002 13017   13055#
BTTBL   053104RG  002 12913   13014#
CHKDNI  017330RG  002  4240#   4460    4580    4608    4789    5297    5648    5720    6176    6242    6373    6416    6478
                       6603    6650    6698    6778    6829    6924    7054    7236    7284    7382    7438    7547    7602
```

```
                            7706     7755     7856     7902     7994     8041     8132     8179     8280     8334     8442     8495     8595
                            8652     8772     8823     8999     9055     9168     9224     9341     9389     9506     9565     9695     9751
                            9858     9914    10097    10165    10320    10375    10475    10522    10635    10688    10802    10855    10974
                           11031    11142    11201    11331    11399    11539    11598    11754    11809    11990    12050    12127    12199
                           12333    12383    12450    12580    12635    12745    12780    12824    12869
CHKINT   020050RG   002     4424#    6831     7440     7604     7757     7904     8050     8188     8336     8497     8654     8825     9057
                            9226     9391     9567     9753     9916    10167    10386    10525    10690    10857    11033    11203    11401
                           11600    11811    12052    12129    12201    '2385    12452    12637
CHKMON   017776RG   002     4386#    6810     7105     7272     7419     7589     7743     7890     8029     8167     8317     8478     8640
                            8807     9043     9212     9376     9553     9739     9902    10153    10363    1050°    10675    10835    11014
                           11184    11387    11586    11797    12038    12115    12187    12371    12438    12623
CLEAR =  000000 G           1647#
CLKBR    000276RG   002     1780#    4723*    4727
CLKCSR   000274RG   002     1779#    4202*    4212*    4720*
CLKFRE   000302RG   002     1782#    4725*
CLKSRV   021674RG   002     4728     4897#
CLKTPB   000274RG   002     1778#
CLKVEC   000300RG   002     1781#    4724*    4729
CLNERR   021556R    002     4792     4803#
CLRDNI   017374RG   002     4275#    4479     4593     4621     4797     5670     5758     6200     6266     6397     6440     6498     6627
                            6674     6790     6865     6988     7069     7250     7312     7396     7491     7561     7643     7720     7799
                            7868     8008     8072     8146     8210     8294     8373     8456     8534     8609     8691     8786     8917
                            9013     9103     9182     9273     9355     9423     9520     9610     9709     9791     9872     9953    10111
                           10223    10334    10405    10489    10558    10649    10728    10816    10890    10988    11065    1'156    11246
                           11345    11434    11553    11666    11768    11874    12004    12094    12166    12244    12347    12418    12489
                           12594    12659    12762    12796    12840    12885
CLRINT   017422R    002     4304#    4566     5708
CNTTAB   000624RG   002     1903#    3421    12615    12680    12701
COLTST   004046RG   002     2265#   11999    12009    12043    12057    12066    12088    12099    12120    12134    12140    12160    12171
                           12192    12206    12215    12239    12249
CPUPRI   000674RG   002     1909#    4873*    6972
CRCDAT   003621RG   002     2238#   10983    10993    11019    11038    11047    11060    11070
CRCERR   003656RG   002     2243#   11151    11161    11189    11208    11217    11233    11241    11251
CRCPAT   003704RG   002     2247#   '1340    11350    11392    11406    11415    11429    11439
CRCSIZ=  000004 C           1670#
CRC16    017042FG   002     4054#    6283
CRC32    016722RG   002     3982#
CSRNUM   000304RG   002     1784#    2890     2905     2918     3001     4466*    4471     4938*    5041*    5187*    5237*    5286*    5378*
                            5488*    5532*    7098*
C$AU   = 000052             1389#    4835
C$AUTO=  000061             1389#    4774
C$BRK  = 0J0022             1389#    4244     4391
C$BSEG=  000004             1389#    4968     4986     5071     5089     5106     5123     5140     5235     5322     5348     5374     5403
                            5424     5443     5493     5535     5551     5666     6171     6196     6230     6262     6348     6393     6410
                            6436     6462     6497     6583     6623     6643     6670     6693     6769     6809     6912     6944     7042
                            7104     7160     7224     7271     7370     7418     7535     7588     7694     7742     7847     7888     7982
                            8028     8120     '166     8?68     8316     8430     8477     8583     8635     8760     8806     8987     9038
                            9156     9207     9329     9375     9494     9549     9683     9734     9846     9901    10085    10147    10308
                           10358    10463    10508    10623    10674    10727    10790    10834    10962    11010    11130    11183    11319
                           11370    11380    11527    11576    11742    11787    11978    12033    12109    12183    12321    12367    12434
                           12568    12622
C$BSUB=  000002             1389#    4933     4963     5036     5066     5291     5317     5597     5620     5642     6166     6217     6343
                            6457
C$CEFG=  000045             1389#
C$CLCK=  000062             1389#    4709
C$CLEA-  000012             1389#    4814
```

```
C$CLOS= 000035        1389#
C$CLP1= 000006        1389#    4948    5051    5197    5247
C$CVEC= 000036        1389#    4951    5007    5054    5161    5200    5208    5250    5262    6986
C$DCLN= 000044        1389#    4758    4956    5059    5205    5255
C$DODU= 000051        1389#    4954    5057    5203    5253
C$DRP1= 000024        1389#
C$DU  = 000053        1389#    4824
C$EDIT= 000003        1389#    1447
C$ERDF= 000055        1389#    4943    5046    5192    5242
C$ERHR= 000056        1389#    4977    4994    5080    5097    5114    5131    5148    5303    5332    5354    5385    5409
                      5434     5453    5501    5541    5557    5610    5632    5657    5674    5711    5728    5750    5763
                      6184     6204    6250    6270    6290    6381    6401    6424    6444    6485    6502    6519    6611
                      6631     6658    6678    6706    6716    6785    6793    6813    6834    6841    6861    6868    6932
                      6960     6976    6991    7062    7072    7108    7136    7148    7175    7243    7253    7275    7287
                      7308     7315    7389    7399    7422    7443    7452    7475    7482    7494    7554    7564    7592
                      7607     7616    7633    7646    7713    7723    7746    7760    7769    7788    7802    7863    7871
                      7893     7907    7916    7933    8001    8011    8032    8046    8053    8062    8075    8139    8149
                      8170     8184    8191    8200    8213    8287    8297    8320    8339    8348    8366    8376    8449
                      8459     8481    8500    8509    8527    8537    8602    8612    8643    8657    8666    8684    8694
                      8779     8789    8810    8828    8837    8858    8871    8880    8892    8901    8910    8920    9006
                      9016     9046    9060    9069    9092    9106    9175    9185    9215    9229    9238    9262    9276
                      9348     9358    9379    9394    9403    9416    9426    9513    9523    9556    9570    9579    9599
                      9613     9702    9712    9742    9756    9765    9780    9794    9865    9875    9905    9919    9928
                      9942     9956   10104   10114   10156   10170   10179   10196   10212   10226   10327   10337   10366
                     10389    10397   10408   10482   10492   10512   10528   10537   10551   10561   10642   10652   10678
                     10693    10702   10715   10731   10809   10819   10838   10860   10869   10883   10893   10981   10991
                     11017    11036   11045   11058   11068   11149   11159   11187   11206   11215   11231   11239   11249
                     11338    11348   11390   11404   11413   11427   11437   11546   11556   11589   11603   11612   11629
                     11644    11655   11669   11761   11771   11800   11814   11823   11837   11852   11863   11877   11997
                     12007    12041   12055   12064   12086   12097   12118   12132   12138   12158   12169   12190   12204
                     12213    12237   12247   12340   12350   12374   12388   12397   12411   12421   12441   12455   12464
                     12479    12492   12587   12597   12626   12640   12649   12662   12693   12754   12767   12788   12801
                     12832    12845   12877   12889
C$ERRO= 000060        1389#
C$ERSF= 000054        1389#
C$ERSO= 000057        1389#
C$ESCA= 000010        1389#    5308    5662    5716    5733    5755    6189    6255    6386    6429    6490    6616    6636
                      6663     6683    6818    6935    7051    7067    7457    7077    7113    7233    7248    7258    7280    7292
                      7379     7394    7404    7427    7448    7457    7544    7559    7569    7597    7612    7621    7651
                      7703     7718    7728    7751    7765    7774    7807    7898    7912    7991    8006    8016    8037
                      8056     8067    8129    8144    8154    8175    8196    8205    8277    8292    8302    8325    8344
                      8353     8439    8454    8464    8486    8505    8514    8592    8607    8617    8648    8662    8671
                      8769     8784    8794    8815    8833    8842    8996    9011    9021    9051    9065    9074    9111
                      9165     9180    9190    9220    9234    9243    9281    9338    9353    9363    9384    9399    9408
                      9503     9518    9528    9561    9575    9584    9618    9692    9707    9717    9747    9761    9770
                      9799     9855    9870    9880    9910    9924    9933    9961   10094   10109   10119   10161   10175
                     10184    10317   10332   10342   10371   10394   10472   10487   10497   10517   10533   10542   10632
                     10647    10657   10683   10698   10707   10736   10799   10814   10824   10843   10865   10874   10971
                     10986    10996   11022   11041   11050   11139   11154   11164   11192   11211   11220   11328   11343
                     11353    11395   11409   11418   11536   11551   11561   11594   11608   11617   11674   11751   11766
                     11776    11805   11819   11828   11987   12002   12012   12046   12060   12069   12123   12143   12195
                     12209    12218   12330   12345   12355   12379   12393   12402   12446   12460   12469   12484   12577
                     12592    12602   12631   12645   12654   12667   12698   12738   12759   12772   12793   12806   12837
                     12850    12882   12894
C$ESEG= 000005        1389#    4984    5001    5087    5104    5121    5138    5155    5259    5339    5361    5392    5416
                      5441     5460    5508    5548    5564    5681    6194    6211    6260    6277    6391    6408    6434
```

```
                            6451    6495    6509    6621    6641    6668    6688    6723    6801    6875    6939    6983    7088
                            7155    7182    7263    7322    7409    7501    7574    7639    7733    7795    7879    7945    8021
                            8082    8159    8220    8307    8383    8469    8544    8622    8701    8799    8927    9026    9099
                            9195    9269    9368    9433    9533    9606    9722    9787    9885    9949   10124   10233   10347
                           10415   10502   10568   10662   10722   10741   10829   10900   11001   11076   11169   11256   11358
                           11444   11457   11566   11662   11781   11870   12017   12104   12178   12254   12360   12428   12499
                           12607   12705
C$ESUB= 000003             1389#    4960    5004    5063    5158    5313    5463    5617    5639    5684    6214    6297    6454
                            6528
C$ETST= 000001             1389#    5010    5164    5212    5265    5466    5515    5567    5687    5770    6300    6539    6732
                            6878    6998    7189    7325    7504    7657    7816    7948    8085    8223    8386    8547    8706
                            8930    9122    9292    9436    9631    9814    9975   10236   10425   10571   10750   10903   11084
                           11265   11467   11687   11884   12257   12508   12708   12957
C$EXIT= 000032             1389#    4800   12732
C$GETB= 000026             1389#
C$GETW= 000027             1389#
C$GMAN= 000043             1389#
C$GPHR= 000042             1389#    4741
C$GPLO= 000030             1389#
C$GPRI= 000040             1389#
C$INIT= 000011             1389#    4762
C$INLP= 000020             1389#
C$MANI= 000050             1389#
C$MEM = 000031             1389#    4703
C$MSG = 000023             1389#    2898    2913    2934    2946    2958    2974    2993    3009    3021    3032    3068    3080
                            3092    3105    3117    3142    3155    3175    3189    3204    3232    3260    3274    3303    3330
                            3357    3392    3404    3442    3490    3509    3529    3549    3577    3596    3637    3664    3677
                            3697    3724    3736    3748    3760    3772    3799    3819    3861    3873    3912    3936    3948
C$OPEN= 000034             1389#
C$PNTB= 000014             1389#    2894    2909    2922    2930    2942    2954    2970    2982    2989    3005    3017    3029
                            3038    3050    3057    3064    3076    3088    3101    3113    3125    3131    3138    3151    3164
                            3171    3185    3200    3213    3220    3228    3241    3248    3256    3270    3283    3291    3299
                            3311    3318    3326    3338    3345    3353    3367    3377    3387    3400    3412    3418    3428
                            3435    3452    3459    3465    3472    3479    3486    3498    3505    3517    3525    3537    3545
                            3559    3566    3573    3585    3592    3604    3611    3618    3625    3633    3645    3653    3660
                            3673    3686    3693    3706    3712    3720    3732    3744    3756    3768    3780    3787    3795
                            3807    3815    3828    3835    3842    3849    3857    3869    3883    3891    3899    3907    3920
                            3926    3932    3944   12923   12930   12937   12944   12951
C$PNTF= 000017             1389#    4357    4475    4485    4507    4573    4588    4601    4616    4629    4717    4795
C$PNTS= 000016             1389#
C$PNTX= 000015             1389#
C$QIO = 000377             1389#
C$RDBU= 000007             1389#
C$REFG= 000047             1389#    4683    4688    4693    4698
C$RESE= 000033             1389#
C$REVI= 000003             1389#    1446
C$RFLA= 000021             1389#
C$RPT = 000025             1389#    4650
C$SEFG= 000046             1389#
C$SPRI= 000041             1389#    4201    4215    6951    7124
C$SVEC= 000037             1389#    4731    4929    5032    5184    5232    6918    7084
C$TPRI= 000013             1389#
DATALD  002067RG    002    2084#    6383    6403    6426    6446    6487    6504    6521
DATERR= 000000 G           1666#    6848    7464
DBFRAM  002624RG    002    2150#    8289    8299    8322    8341    8350    8368    8378
DEFADR  053053RG    002     4108   12978#
```

```
DEFHDR  052770RG   002  12919   12972#
DFPTBL  000262RG   002   1557#
DIAGMC= 000000           1389
DIM   = 000020 G         1757#   6222    6475    12860
DMABLK  002501RG   002   2134#   7865    7873    7895    7909    7918    7935
DMAFRM  002431RG   002   2127#   7715    7725    7748    7762    7771    7790    7804
DMATO   002363RG   002   2120#   7556    7566    7594    7609    7618    7635    7648
DNI   = 004000 G         1682#   4241    4275    4276    4332    4334    4335    5368    5624    5629
DPA     052750RG   002   4109   12811*  12812*  12813*  12959#
DPPAT   002720RG   002   2160#   8604    8614    8645    8659    8668    8686    8696
DWFRAM  002662RG   002   2155#   8451    8461    8483    8502    8511    8529    8539
EF.CON= 000036 G         1612#   4682
EF.NEW= 000035 G         1613#   4692
EF.PWR= 000034 G         1614#   4687
EF.RES= 000037 G         1611#
EF.STA= 000040 G         1610#   4697
ENP   = 000400 G         1736#
ERRINT  000666RG   002   1906#   3877    3885    3893    3901    4424*   4425
ERRS  = 040000 G         1734#
EVL   = 000004 G         1630#
E$END = 002100           1389#
E$LOAD= 000035           1389#   1470
FATI  = 000400 G         1685#   4346    4348    4349
FIFTST  003166RG   002   2188#   9515    9525    9558    9572    9581    9601    9615
FORM1   004234RG   002   2285#   2891
FORM10  005010RG   002   2350#   3014
FORM11  005074RG   002   2359#   3026    3047    3308    3335
FORM12  005123RG   002   2363#   3035    3197
FORM13  005216RG   002   2373#   2986
FORM15  005267RG   002   2380#   3054
FORM16  005343RG   002   2388#   3061
FORM17  005421RG   002   2396#   3073
FORM18  005504RG   002   2405#   3085
FORM19  005562RG   002   2413#   3098
FORM2   004271RG   002   2290#   2906
FORM20  005637RG   002   2421#   3110
FORM21  005677RG   002   2427#   3122
FORM22  005751RG   002   2435#   3128
FORM23  006031RG   002   2444#   3135
FORM24  006101RG   002   2451#   3148
FORM25  006133RG   002   2456#   3161
FORM26  006206RG   002   2464#   3168
FORM27  006261RG   002   2472#   3182    3225    3253    3323    3350    3650    3812    3854
FORM28  006331RG   002   2479#   3210    3690    12920   12934   12941   12948
FORM29  006336RG   002   2480#   3217    3342
FORM3   004327RG   002   2296#   2919
FORM30  006361RG   002   2484#   3238
FORM31  006411RG   002   2489#   3245
FORM32  006437RG   002   2493#   3267
FORM33  006534RG   002   2504#   3280    3839
FORM34  006572RG   002   2509#   3288
FORM35  006655RG   002   2518#   3296
FORM36  006755RG   002   2529#   3315    3846
FORM37  007013RG   002   2535#   3364
FORM38  007124RG   002   2549#   3374    3384
FORM39  007215RG   002   2559#   3397
```

```
FORM4    004376RG    002    2303#    2927
FORM40   007256RG    002    2565#    3409
FORM41   007341RG    002    2574#    3415
FORM42   007407RG    002    2581#    3425
FORM43   007416RG    002    2583#    3432
FORM44   007424RG    002    2584#    3462
FORM45   007503RG    002    2592#    3469
FORM46   007531RG    002    2596#    3476
FORM47   007560RG    002    2600#    3483
FORM48   007607RG    002    2604#    3495
FORM49   007704RG    002    2615#    3502
FORM5    004445RG    002    2310#    2939
FORM50   007760RG    002    2623#    3522     3542     3556
FORM51   010047RG    002    2633#    3514
FORM52   010101RG    002    2638#    3534
FORM53   010137RG    002    2644#    3563
FORM54   010211RG    002    2652#    3570
FORM55   010237RG    002    2656#    3582
FORM56   010310RG    002    2663#    3601     3777
FORM57   010363RG    002    2671#    3589     3608     3657
FORM58   010433RG    002    2678#    3615
FORM59   010511RG    002    2686#    3622     3703     3784
FORM6    004510RG    002    2316#    2967
FORM60   010604RG    002    2696#    3630     3709     3792
FORM61   010671RG    002    2705#    3642     3804
FORM62   010754RG    002    2714#    3670     4354
FORM63   011027RG    002    2722#    3683
FORM64   011066RG    002    2728#   12927
FORM65   011137RG    002    2735#
FORM66   011165RG    002    2739#    3729
FORM67   011252RG    002    2748#    3741
FORM68   011336RG    002    2757#    3753
FORM69   011433RG    002    2768#    3765
FORM7    004566RG    002    2324#    2979
FORM70   011551RG    002    2782#    3825
FORM71   011613RG    002    2788#    3832
FORM72   011655RG    002    2794#    3866     3904
FORM73   011727RG    002    2802#    3880
FORM74   011772RG    002    2808#    3888
FORM75   012047RG    002    2816#    3896
FORM76   012115RG    002    2823#    3449
FORM77   012151RG    002    2828#    3456
FORM78   012225RG    002    2836#    3717
FORM79   012265RG    002    2842#    3917
FORM8    004650RG    002    2333#    2951     4482
FORM80   012334RG    002    2849#    3923
FORM81   012413RG    002    2857#    3929
FORM82   012456RG    002    2863#    3941
FORM9    004732RG    002    2342#    3002     4472
FREMEM   000324RG    002    1793#    4705*    4706*    6231     6238     6281     6353     6361     6467     6473     6513     7889    10129
                           10135    10192    10193    10194    10204
FRESIZ   000322RG    002    1792#    4704*    4705
FRSTIM   000672RG    002    1908#    4701*    4734*    4756*   12729
F$AU   = 000015             1389#    4829     4834
F$AUTO=  000020             1389#    4771     4773
F$BGN  = 000040             1389#    1393     2888     2901     2916     2937     2949     2961     2977     2996     3012     3024     3045
```

```
                          3071     3083     3095     3108     3120     3145     3158     3178     3192     3207     3235     3263     3277
                          3306     3333     3360     3395     3407     3445     3493     3512     3532     3552     3580     3599     3640
                          3667     3680     3700     3727     3739     3751     3763     3775     3802     3822     3864     3876     3915
                          3939     4644     4648     4661     4677     4771     4783     4800     4818     4822     4829     4833     4852
                          4871     4897     4923     4932     4959     4962     4968     4986     5003     5009     5026     5035     5062
                          5065     5071     5089     5106     5123     5140     5157     5163     5178     5211     5226     5235     5264
                          5285     5290     5308     5312     5316     5322     5348     5374     5403     5424     5443     5462     5465
                          5487     5493     5512     5531     5535     5551     5566     5594     5595     5616     5619     5638     5641
                          5662     5666     5683     5686     5707     5716     5733     5755     5769     6163     6165     6171     6189
                          6196     6213     6216     6230     6255     6262     6296     6299     6339     6342     6348     6386     6393
                          6410     6429     6436     6453     6456     6462     6490     6497     6527     6538     6578     6583     6616
                          6623     6636     6643     6663     6670     6683     6693     6731     6762     6769     6809     6818     6877
                          6906     6912     6944     6965     6997     7035     7042     7051     7067     7077     7104     7113     7160
                          7188     7217     7224     7233     7248     7258     7271     7280     7292     7324     7363     7370     7379
                          7394     7404     7418     7427     7448     7457     7503     7528     7535     7544     7559     7569     7588
                          7597     7612     7621     7651     7656     7687     7694     7703     7718     7728     7742     7751     7765
                          7774     7807     7815     7840     7847     7888     7898     7912     7947     7975     7982     7991     8006
                          8016     8028     8037     8058     8067     8084     8113     8120     8129     8144     8154     8166     8175
                          8196     8205     8222     8261     8268     8277     8292     8302     8316     8325     8344     8353     8385
                          8423     8430     8439     8454     8464     8477     8486     8505     8514     8546     8576     8583     8592
                          8607     8617     8635     8648     8662     8671     8705     8753     8760     8769     8784     8794     8806
                          8815     8833     8842     8929     8980     8987     8996     9011     9021     9038     9051     9065     9074
                          9111     9121     9149     9156     9165     9180     9190     9207     9220     9234     9243     9281     9291
                          9322     9329     9338     9353     9363     9375     9384     9399     9408     9435     9487     9494     9503
                          9518     9528     9549     9561     9575     9584     9618     9630     9676     9683     9692     9707     9717
                          9734     9747     9761     9770     9799     9813     9839     9846     9855     9870     9880     9901     9910
                          9924     9933     9961     9974    10078    10085    10094    10109    10119    10147    10161    10175    10184
                         10235    10301    10308    10317    10332    10342    10358    10371    10394    10424    10456    10463    10472
                         10487    10497    10508    10517    10533    10542    10570    10616    10623    10632    10647    10657    10674
                         10683    10698    10707    10727    10736    10749    10783    10790    10799    10814    10824    10834    10843
                         10865    10874    10902    10955    10962    10971    10986    10996    11010    11022    11041    11050    11083
                         11123    11130    11139    11154    11164    11183    11192    11211    11220    11264    11312    11319    11328
                         11343    11353    11370    11380    11395    11409    11418    11466    11520    11527    11536    11551    11561
                         11576    11594    11608    11617    11674    11686    11735    11742    11751    11766    11776    11787    11805
                         11819    11828    11885    11971    11978    11987    12002    12012    12033    12046    12060    12069    12109
                         12123    12143    12183    12195    12209    12218    12256    12314    12321    12330    12345    12355    12367
                         12379    12393    12402    12434    12446    12460    12469    12484    12507    12561    12568    12577    12592
                         12602    12622    12631    12645    12654    12667    12698    12707    12728    12732    12738    12759    12772
                         12793    12806    12837    12850    12882    12894    12956    13081    13121
F$CLEA= 000007            1389#    4783     4813
F$DU  = 000016            1389#    4818     4823
F$END = 000041            1389#    1393     2899     2914     2935     2947     2959     2975     2994     3010     3022     3043     3069
                          3081     3093     3106     3118     3143     3156     3176     3190     3205     3233     3261     3275     3304
                          3331     3358     3393     3405     3443     3491     3510     3530     3550     3578     3597     3638     3665
                          3678     3698     3725     3737     3749     3761     3773     3800     3820     3862     3874     3913     3937
                          3949     4646     4651     4763     4775     4800     4815     4820     4825     4831     4836     4857     4877
                          4905     4923     4932     4959     4961     4962     4985     5002     5003     5005     5009     5011     5026
                          5035     5062     5064     5065     5088     5105     5122     5139     5156     5157     5159     5163     5165
                          5178     5211     5213     5226     5260     5264     5266     5285     5290     5308     5312     5314     5316
                          5340     5362     5393     5417     5442     5461     5462     5464     5465     5467     5487     5509     5512
                          5514     5531     5549     5565     5566     5568     5594     5596     5616     5618     5619     5638     5640
                          5641     5662     5682     5683     5685     5686     5688     5707     5716     5733     5755     5769     5771
                          6163     6165     6189     6195     6212     6213     6215     6216     6255     6261     6278     6296     6298
                          6299     6301     6339     6342     6386     6392     6409     6429     6435     6452     6453     6455     6456
                          6490     6496     6510     6527     6529     6538     6540     6578     6616     6622     6636     6642     6663
                          6669     6683     6689     6724     6731     6733     6762     6802     6818     6876     6877     6879     6906
                         6940    6965    6984    6997    6999    7035    7051    7067    7077    7089    7113    7156    7183
                         7188    7190    7217    7233    7248    7258    7264    7280    7292    7323    7324    7326    7363
                         7379    7394    7404    7410    7427    7448    7457    7502    7503    7505    7528    7544    7559
                         7569    7575    7597    7612    7621    7640    7651    7656    7658    7687    7703    7718    7728
                         7734    7751    7765    7774    7796    7807    7815    7817    7840    7880    7898    7912    7946
                         7947    7949    7975    7991    8006    8016    8022    8037    8058    8067    8083    8084    8086
                         8113    8129    8144    8154    8160    8175    8196    8205    8221    8222    8224    8261    8277
                         8292    8302    8308    8325    8344    8353    8384    8385    8387    8423    8439    8454    8464
                         8470    8486    8505    8514    8545    8546    8548    8576    8592    8607    8617    8623    8648
                         8662    8671    8702    8705    8707    8753    8769    8784    8794    8800    8815    8833    8842
                         8928    8929    8931    8980    8996    9011    9021    9027    9051    9065    9074    9100    9111
                         9121    9123    9149    9165    9180    9190    9196    9220    9234    9243    9270    9281    9291
                         9293    9322    9338    9353    9363    9369    9384    9399    9408    9434    9435    9437    9487
                         9503    9518    9528    9534    9561    9575    9584    9607    9618    9630    9632    9676    9692
                         9707    9717    9723    9747    9761    9770    9788    9799    9813    9815    9839    9855    9870
                         9880    9886    9910    9924    9933    9950    9961    9974    9976   10078   10094   10109   10119
                        10125   10161   10175   10184   10234   10235   10237   10301   10317   10332   10342   10348   10371
                        10394   10416   10424   10426   10456   10472   10487   10497   10503   10517   10533   10542   10569
                        10570   10572   10616   10632   10647   10657   10663   10683   10698   10707   10723   10736   10742
                        10749   10751   10783   10799   10814   10824   10830   10843   10865   10874   10901   10902   10904
                        10955   10971   10986   10996   11002   11022   11041   11050   11077   11083   11085   11123   11139
                        11154   11164   11170   11192   11211   11220   11257   11264   11266   11312   11328   11343   11353
                        11359   11395   11409   11418   11458   11466   11468   11520   11536   11551   11561   11567   11594
                        11608   11617   11663   11674   11686   11688   11735   11751   11766   11776   11782   11805   11819
                        11828   11871   11883   11885   11971   11987   12002   12012   12018   12046   12060   12069   12105
                        12123   12143   12179   12195   12209   12218   12255   12256   12258   12314   12330   12345   12355
                        12361   12379   12393   12402   12429   12446   12460   12469   12484   12500   12507   12509   12561
                        12577   12592   12602   12608   12631   12645   12654   12667   12698   12706   12707   12709   12728
                        12732   12738   12759   12772   12793   12806   12837   12850   12882   12894   12956   12958   13096
                        13121
F$HARD= 000004           1389#  13081   13094
F$HW  = 000013           1389#   1555    1563
F$INIT= 000006           1389#   4677    4761
F$JMP = 000050           1389#   4646    4800    4820    4831   12732
F$MOD = 000000           1389#   1393   13121
F$MSG = 000011           1389#   2888    2897    2901    2912    2916    2933    2937    2945    2949    2957    2961    2973
                         2977    2992    2996    3008    3012    3020    3024    3041    3045    3067    3071    3079    3083
                         3091    3095    3104    3108    3116    3120    3141    3145    3154    3158    3174    3178    3188
                         3192    3203    3207    3231    3235    3259    3263    3273    3277    3302    3306    3329    3333
                         3356    3360    3391    3395    3403    3407    3441    3445    3489    3493    3508    3512    3528
                         3532    3548    3552    3576    3580    3595    3599    3636    3640    3663    3667    3676    3680
                         3696    3700    3723    3727    3735    3739    3747    3751    3759    3763    3771    3775    3798
                         3802    3818    3822    3860    3864    3872    3876    3911    3915    3935    3939    3947
F$PROT= 000021           1389#   4661    4668
F$PWR = 000017           1389#
F$RPT = 000012           1389#   4644    4649
F$SEG = 000003           1389#   4968    4983    4986    5000    5071    5086    5089    5103    5106    5120    5123    5137
                         5140    5154    5235    5258    5322    5338    5348    5360    5374    5391    5400    5415    5424
                         5440    5443    5459    5493    5507    5535    5547    5551    5563    5666    5680    6171    6193
                         6196    6210    6230    6259    6262    6276    6348    6390    6393    6407    6410    6433    6436
                         6450    6462    6494    6497    6508    6583    6620    6623    6640    6643    6667    6670    6687
                         6693    6722    6769    6800    6809    6874    6912    6938    6944    6982    7042    7087    7104
                         7154    7160    7181    7224    7262    7271    7321    7370    7408    7418    7500    7535    7573
                         7588    7638    7694    7732    7742    7794    7847    7878    7888    7944    7982    8020    8028
                         8081    8120    8158    8166    8219    8268    8306    8316    8382    8430    8468    8477    8543
                         8583    8621    8635    8700    8760    8798    8806    8926    8987    9025    9038    9098    9156
```

```
                            9194     9207     9268     9329     9367     9375     9432     9494     9532     9549     9605     9683     9721
                            9734     9786     9846     9884     9901     9948    10085    10123    10147    10232    10308    10346    10358
                           10414    10463    10501    10508    10567    10623    10661    10674    10721    10727    10740    10790    10828
                           10834    10899    10962    11000    11010    11075    11130    11168    11183    11255    11319    11357    11370
                           11380    11443    11456    11527    11565    11576    11661    11742    11780    11787    11869    11978    12016
                           12033    12103    12109    12177    12183    12253    12321    12359    12367    12427    12434    12498    12568
                           12606    12622    12704
F$SOFT= 000005              1389#
F$SRV = 000010              1389#    4852     4855     4871     4875     4897     4903
F$SUB = 000002              1389#    4933     4959     4963     5003     5036     5062     5066     5157     529;     5312     5317     5462
                            5597     5616     5620     5638     5642     5683     6166     6213     6217     6296     6343     6453     6457
                            6527
F$SW  = 000014              1389#
F$TEST= 000001              1389#    4924     5009     5027     5163     5179     5211     5227     5264     5286     5465     5488     5512
                            5532     5566     5595     5686     5708     5769     6164     6299     6340     6538     6579     6731     6763
                            6877     6907     6997     7036     7188     7218     7324     7364     7503     7529     7656     7688     7815
                            7841     7947     7976     8084     8114     8222     8262     8385     8424     8546     8577     8705     8754
                            8929     8981     9121     9150     9291     9323     9435     9488     9630     9677     9813     9840     9974
                           10079    10235    10302    10424    10457    10570    10617    10749    10784    10902    10956    11083    11124
                           11264    11313    11466    11521    11686    11736    11883    11972    12256    12315    12507    12562    12707
                           12729    12956
GETCMD= 000002 G            1700#    4606     4636     6241     6414     6476     6648     6696    12779    12823    12868
GETCRC  016764R     002     3990     4013#
GETPCB= 000001 G            1699#    4578     6174     6371     6601    12743
G$CNTO= 000200              1389#
G$DELM= 000372              1389#
G$DISP= 000003              1389#
G$EXCP= 000400              1389#
G$HILI= 000002              1389#
G$LOLI= 000001              1389#
G$NO  = 000000              1389#   13084    13089
G$OFFS= 000400              1389#   13084    13089
G$OFSI= 000376              1389#   13084    13089
G$PRMA= 000001              1389#   13084    13089
G$PRMD= 000002              1389#
G$PRML= 000000              1389#
G$RADA= 000140              1389#
G$RADB= 000000              1389#
G$RADD= 000040              1389#
G$RADL= 000120              1389#
G$RADO= 000020              1389#   13084    13089
G$XFER= 000004              1389#
G$YES = 000010              1389#
HAFDUP  004016RG    002     2261#   11763    11773    11802    11816    11825    11839    11854    11865    11879
HELP  = 000000              1387#    1389     1400     1488     1545     1559     1565#    1568     4641#    4669     4679     4765     4908#
                           13066#   13079    13097
HEXDAT  052766RG    002     4111*    4146     4182    12969#
HEXDPA  017130RG    002     4102#   12817
HEXH    017212RG    002     4112     4143#
HEXL    017250RG    002     4114     4179#
HEXTBL  053057RG    002     4154     4185    12991#
HEXVAL  052767RG    002     4113     4115     4155*    4186*   12970#
HOE   = 100000 G            1643#
IBE   = 010000 G            1640#
ICAB  = 040000 G            1706#    5428
IDU   = 000040 G            1633#
```

```
IE    = 000100 G          1660#    4202     6921     6922     7118     7120
IER   = 020000 G          1641#
INERR = 000003            1673#    3446     6846     7462     8359     8520     8677     9413     9777     9939     10190    10548    10712
                          10880    11055    11626    11834
INITH = 177777 G          1661#    3987
INITL = 177777 G          1662#    3988
INMON = 000001            1671#    4388     6772     7045     7227     7373     7538     7697     7850     7985     8123     8271     8433
                          8586     8763     8990     9159     9332     9497     9686     9849     10088    10311    10466    10626    10793
                          10965    11133    11322    11530    11745    11981    12324    12571
INTBIT  002251RG    002   2105#    7064     7074     7110     7138     7150     7177
INTE  = 000100 G          1687#    5603     5607
INTR  = 000200 G          1686#    5326
INTST = 000002            1672#    6712
INTVEC  002214RG    002   2100#    6934     6962     6978     6993
IOADR = 020000 G          1657#    1658
IOSIZ = 020000 G          1653#    1658
ISR   = 000100 G          1634#
IXE   = 004000 G          1639#
I$AU  = 000041            1389#    4829*    4836#
I$AUTO= 000041            1389#    4771#    4775#
I$CLN = 000041            1389#    4783#    4800     4815#
I$DU  = 000041            1389#    4818#    4825#
I$HRD - 000041            13081#   13096#
I$INIT= 000041            1389#    4677#    4763#
I$MOD = 000041            1389#    1393#    13121#
I$MSG = 000041            1389#    2888#    2899#    2901#    2914#    2916#    2935#    2937#    2947#    2949#    2959#    2961#    2975#
                          2977#    2994#    2996#    3010#    3012#    3022#    3024#    3043#    3045#    3069#    3071#    3081#    3083#
                          3093#    3095#    3106#    3108#    3118#    3120#    3143#    3145#    3156#    3158#    3176#    3178#    3190#
                          3192#    3205#    3207#    3233#    3235#    3261#    3263#    3275#    3277#    3304#    3306#    3331#    3333#
                          3358#    3360#    3393#    3395#    3405#    3407#    3443#    3445#    3491#    3493#    3510#    3512#    3530#
                          3532#    3550#    3552#    3578#    3580#    3597#    3599#    3638#    3640#    3665#    3667#    3678#    3680#
                          3698#    3700#    3725#    3727#    3737#    3739#    3749#    3751#    3761#    3763#    3773#    3775#    3800#
                          3802#    3820#    3822#    3862#    3864#    3874#    3876#    3913#    3915#    3937#    3939#    3949#
I$PROT= 000040            1389#    4661#
I$PTAB= 000041            1389#
I$PWR = 000041            1389#
I$RPT = 000041            1389#    4644#    4651#
I$SEG - 000041            1389#    4923     4932     4962     4968#    4985#    4986#    5002#    5026     5035     5065     5071#    5088#
                          5089#    5105#    5106#    5122#    5123#    5139#    5140#    5156#    5178     5226     5235#    5260#    5285
                          5290     5316     5322#    5340#    5348#    5362#    5374#    5393#    5403#    5417#    5424#    5442#    5443#
                          5461#    5487     5493#    5509#    5531     5535#    5549#    5551#    5565#    5594     5596     5619     5641
                          5666#    5682#    5707     6163     6165     6171#    6195#    6196#    6212#    6216     6230#    6261#    6262#
                          6278#    6339     6342     6348#    6392#    6393#    6409#    6410#    6435#    6436#    6452#    6456     6462#
                          6496#    6497#    6510#    6578     6583#    6622#    6623#    6642#    6643#    6669#    6670#    6689#    6693#
                          6724#    6762     6769#    6802#    6809#    6876#    6906     6912#    6940#    6944#    6984#    7035     7042#
                          7089#    7104#    7156#    7160#    7183#    7217     7224#    7264#    7271#    7323#    7363     7370#    7410#
                          7418#    7502#    7528     7535#    7575#    7588#    7640#    7687     7694#    7734#    7742#    7796#    7840
                          7847#    7880#    7888#    7946#    7975     7982#    8022#    8028#    8083#    8113     8120#    8160#    8166#
                          8221#    8261     8268#    8308#    8316#    8384#    8423     8430#    8470#    8477#    8545#    8576     8583#
                          8623#    8635#    8702#    8753     8760#    8800#    8806#    8928#    8980     8987#    9027#    9038#    9100#
                          9149     9156#    9196#    9207#    9270#    9322     9329#    9369#    9375#    9434#    9487     9494#    9534#
                          9549#    9607#    9676     9683#    9723#    9734#    9788#    9839     9846#    9880#    9901#    9950#    10078
                          10085#   10125#   10147#   10234#   10301    10308#   10348#   10358#   10416#   10456    10463#   10503#   10508#
                          10569#   10616    10623#   10663#   10674#   10723#   10727#   10742#   10783    10790#   10830#   10834#   10901#
                          10955    10962#   11002#   11010#   11077#   11123    11130#   11170#   11183#   11257#   11312    11319#   11359#
                          11370#   11380#   11458#   11520    11527#   11567#   11576#   11663#   11735    11742#   11782#   11787#   11871#
```

```
                                     11971    11978#   12018#   12033#   12105#   12109#   12179#   12183#   12255#   12314    12321#   12361#   12367#
                                     12429#   12434#   12500#   12561    12568#   12608#   12622#   12706#   12728
I$SETU= 000041                        1389#
I$SRV = 000041                        1389#    4852#    4857#    4871#    4877#    4897#    4905#
I$SUB = 000041                        1389#    4923     4932#    4959#    4961#    4962#    5003#    5005#    5026     5035#    5062#    5064#    5065#
                                      5157#    5159#    5178     5226     5285     5290#    5312#    5314#    5316#    5462#    5464#    5487     5531
                                      5594     5596#    5616#    5618#    5619#    5638#    5640#    5641#    5683#    5685#    5707     6163     6165#
                                      6213#    6215#    6216#    6296#    6298#    6339     6342#    6453#    6455#    6456#    6527#    6529#    6578
                                      6762     6906     7035     7217     7363     7528     7687     7840     7975     8113     8261     8423     8576
                                      8753     8980     9149     9322     9487     9676     9839    10078    10301    10456    10616    10783    10955
                                     11123    11312    11520    11735    11971    12314    12561    12728
I$TST = 000041                        1389#    4923#    4932     4962     5009#    5011#    5026#    5035     5065     5163#    5165#       3#    5211#
                                      5213#    5226#    5264#    5266#    5285#    5290     5308     5316     5465#    5467#    5487#       2#    5514#
                                      5531#    5566#    5568#    5594#    5596     5619     5641     5662     5686#    5688#    5707#     /16     5733
                                      5755     5769#    5771#    6163#    6165     6189     6216     6255     6299#    6301#    6339#    6342     6386
                                      6429     6456     6490     6538#    6540#    6578#    6616     6636     6663     6683     6731#    6733#    6762#
                                      6818     6877#    6879#    6906#    6965     6997#    6999#    7035#    7051     7067     7077     7113     7188#
                                      7190#    7217#    7233     7248     7258     7280     7292     7324#    7326#    7363#    7379     7394     7404
                                      7427     7448     7457     7503#    7505#    7528#    7544     7559     7569     7597     7612     7621     7651
                                      7656#    7658#    7687#    7703     7718     7728     7751     7765     7774     7807     7815#    7817#    7840#
                                      7898     7912     7947#    7949#    7975#    7991     8006     8016     8037     8058     8067     8084#    8086#
                                      8113#    8129     8144     8154     8175     8196     8205     8222#    8224#    8261#    8277     8292     8302
                                      8325     8344     8353     8385#    8387#    8423#    8439     8454     8464     8486     8505     8514     8546#
                                      8548#    8576#    8592     8607     8617     8648     8662     8671     8705#    8707#    8753#    8769     8784
                                      8794     8815     8833     8842     8929#    8931#    8980#    8996     9011     9021     9051     9065     9074
                                      9111     9121#    9123#    9149#    9165     9180     9190     9220     9234     9243     9281     9291#    9293#
                                      9322#    9338     9353     9363     9384     9399     9408     9435#    9437#    9487#    9503     9518     9528
                                      9561     9575     9584     9618     9630#    9632#    9676#    9692     9707     9717     9747     9761     9770
                                      9799     9813#    9815#    9839#    9855     9870     9880     9910     9924     9933     9961     9974#    9976#
                                     10078#   10094    10109    10119    10161    10175    10184    10235#   10237#   10301#   10317    10332    10342
                                     10371    10394    10424#   10426#   10456#   10472    10487    10497    10517    10533    10542    10570#   10572#
                                     10616#   10632    10647    10657    10683    10698    10707    10736    10749#   10751#   10783#   10799    10814
                                     10824    10843    10865    10874    10902#   10904#   10955#   10971    10986    10996    11022    11041    11050
                                     11083#   11085#   11123#   11139    11154    11164    11192    11211    11220    11264#   11266#   11312#   11328
                                     11343    11353    11395    11409    11418    11466#   11468#   11520#   11536    11551    11561    11594    11608
                                     11617    11674    11686#   11688#   11735#   11751    11766    11776    11805    11819    11828    11883#   11885#
                                     11971#   11987    12002    12012    12046    12060    12069    12123    12143    12195    12209    12218    12256#
                                     12258#   12314#   12330    12345    12355    12379    12393    12402    12446    12460    12469    12484    12507#
                                     12509#   12561#   12577    12592    12602    12631    12645    12654    12667    12698    12707#   12709#   12728#
                                     12732    12738    12759    12772    12793    12806    12837    12850    12882    12894    12956#   12958#
J$JMP = 000167                        1389#    4646     4820     4831
LASFT   002121RG    002               2089#    6613     6633     6660     6680     6708     6718
LASM  = 000001 G                      1742#    4634     6694
LIM   = 000021 G                      1758#    4560     6365     6595
LINADR= 100000 G                      1659#    9539     9624     9728     9730     9807     9891     9893     9968
LINSIZ= 077774 G                      1655#
LNKARB  003334RG    002               2206#   10106    10116    10158    10172    10181    10198    10214    10228
LNKBYT  003022RG    002               2171#    9008     9018     9048     9062     9071     9094     9108
LNKMEM  002333RG    002               2115#    7391     7401     7424     7445     7454     7477     7484     7496
LODMIC  020252R     002               4499#    6775     7048     7230     7376     7541     7700     7853     7988     8126     8274     8436     8589
                                      8766     8993     9162     9335     9500     9689     9852    10091    10314    10469    10629    10796    10968
                                     11136    11325    11533    11748    11984    12327    12574
LOE   = 040000 G                      1642#
LOT   = 000010 G                      1631#
LPMSG   052762RG    002              12906*   12940    12966#
LPMSGO  053146RG    002              13011    13024#
```

```
LPMSG1   053205RG    002   13012    13030#
LPTBL    053100RG    002   12905    13011#
L$ACP    000110RG    002    1477#
L$APT    000036RG    002    1435#
L$AU     021640RG    002    1462     4829#
L$AUT    000070RG    002    1461#
L$AUTO   021470RG    002    1478     4771#
L$CCP    000106RG    002    1475#
L$CLEA   021472RG    002    1476     4783#
L$CO     000032RG    002    1431#
L$DEPO   000011RG    002    1413#
L$DESC   000704RG    002    1468     1936#
L$DESP   000076RG    002    1457#
L$DEVP   000060RG    002    1453#
L$DISP   000124RG    002    1438     1497#
L$DLY    000116RG    002    1483#
L$DTP    000040RG    002    1437#
L$DTYP   000034RG    002    1433#
L$DU     021632RG    002    1464     4818#
L$DUT    000072RG    002    1463#
L$DVTY   000676RG    002    1454     1928#
L$EF     000052RG    002    1448#
L$ENVI   000044RG    002    1441#
L$ETP    000102RG    002    1471#
L$EXP1   000046RG    002    1443#
L$EXP4   000064RG    002    1457#
L$EXP5   000066RG    002    1459#
L$HARD   053464RG    002    1420    13081    13082#
L$HIME   000120RG    002    1485#
L$HPCP   000016RG    002    1419#
L$HPTP   000022RG    002    1423#
L$HW     000262RG    002    1424     1555     1556#
L$ICP    000104RG    002    1473#
L$INIT   021124RG    002    1474     4677#
L$LADP   000026RG    002    1427#
L$LAST=  ****** GX          1428
L$LOAD   000100RG    002    1469#
L$LUN    000074RG    002    1465#
L$MREV   000050RG    002    1445#
L$NAME   000000RG    002    1402#
L$PRIO   000042RG    002    1439#
L$PROT   021116RG    002    1480     4661#
L$PRT    000112RG    002    1479#
L$REPP   000062RG    002    1455#
L$REV    000010RG    002    1411#
L$RPT    021110RG    002    1456     4644#
L$SPC    000056RG    002    1451#
L$SPCP   000020RG    002    1421#
L$SPTP   000024RG    002    1425#
L$STA    000030RG    002    1429#
L$TEST   000114RG    002    1481#
L$TIML   000014RG    002    1417#
L$UNIT   000012RG    002    1415#    4737
L10000   000266R     002    1555     1563#
L10001   012572R     002    2897#
L10002   012630R     002    2912#
```

```
L10003  012702R    002    2933#
L10004  012724R    002    2945#
L10005  012746R    002    2957#
L10006  013010R    002    2973#
L10007  013054R    002    2992#
L10010  013116R    002    3008#
L10011  013140R    002    3020#
L10012  013210R    002    3041#
L10013  013276R    002    3067#
L10014  013320R    002    3079#
L10015  013342R    002    3091#
L10016  013370R    002    3104#
L10017  013412R    002    3116#
L10020  013476R    002    3141#
L10021  013522R    002    3154#
L10022  013570R    002    3174#
L10023  013616R    002    3188#
L10024  013646R    002    3203#
L10025  013740R    002    3231#
L10026  014032R    002    3259#
L10027  014062R    002    3273#
L10030  014156R    002    3302#
L10031  014246R    002    3329#
L10032  014336R    002    3356#
L10033  014452R    002    3391#
L10034  014474R    002    3403#
L10035  014624R    002    3441#
L10036  015012R    002    3489#
L10037  015056R    002    3508#
L10040  015124R    002    3528#
L10041  015172R    002    3548#
L10042  015274R    002    3576#
L10043  015342R    002    3595#
L10044  015506R    002    3636#
L10045  015604R    002    3663#
L10046  015632R    002    3676#
L10047  015702R    002    3696#
L10050  016000R    002    3723#
L10051  016022R    002    3735#
L10052  016044R    002    3747#
L10053  016066R    002    3759#
L10054  016110R    002    3771#
L10055  016206R    002    3798#
L10056  016260R    002    3818#
L10057  016430R    002    3860#
L10060  016452R    002    3872#
L10061  016614R    002    3911#
L10062  016676R    002    3935#
L10063  016720R    002    3947#
L10064  021114R    002    4647      4649#
L10066  021466R    002    4761#
L10067  021470R    002    4773#
L10070  021630R    002    4801      4813#
L10071  021636R    002    4821      4823#
L10072  021644R    002    4832      4834#
L10073  021654R    002    4855#
```

```
L10074  021672R    002     4875#
L10075  021706R    002     4903#
L10076  022120R    002     5009#
L10077  022012R    002     4959#
L10100  022110R    002     5003#
L10101  022452R    002     5163#
L10102  022226R    002     5062#
L10103  022442R    002     5157#
L10104  022564R    002     5211#
L10105  022702R    002     5264#
L10106  023364R    002     5309      5465#
L10107  022772R    002     5312#
L10110  023362R    002     5462#
L10111  023446R    002     5512#
L10112  023544R    002     5566#
L10113  023766R    002     5663      5686#
L10114  023610R    002     5616#
L10115  023650R    002     5638#
L10116  023764R    002     5683#
L10117  024200R    002     5717      5734      5756      5769#
L10120  027744R    002     6190      6256      6299#
L10121  027462R    002     6213#
L10122  027742R    002     6296#
L10123  030532R    002     6387      6430      6491      6538#
L10124  030302R    002     6453#
L10125  030512R    002     6527#
L10126  031212R    002     6617      6637      6664      6684      6731#
L10127  031630R    002     6819      6877#
L10130  032120R    002     6966      6997#
L10131  032626R    002     7052      7068      7078      7114      7188#
L10132  033144R    002     7234      7249      7259      7281      7293      7324#
L10133  033612R    002     7380      7395      7405      7428      7449      7458      7503#
L10134  034170R    002     7545      7560      7570      7598      7613      7622      7652      7656#
L10135  034550R    002     7704      7719      7729      7752      7766      7775      7808      7815#
L10136  035076R    002     7899      7913      7947#
L10137  035406R    002     7992      8007      8017      8038      8059      8068      8084#
L10140  035716R    002     8130      8145      8155      8176      8197      8206      8222#
L10141  036276R    002     8278      8293      8303      8326      8345      8354      8385#
L10142  036656R    002     8440      8455      8465      8487      8506      8515      8546#
L10143  037252R    002     8593      8608      8618      8649      8663      8672      8705#
L10144  040014R    002     8770      8785      8795      8816      8834      8843      8929#
L10145  040412R    002     8997      9012      9022      9052      9066      9075      9112      9121#
L10146  041014R    002     9166      9181      9191      9221      9235      9244      9282      9291#
L10147  041334R    002     9339      9354      9364      9385      9400      9409      9435#
L10150  041724R    002     9504      9519      9529      9562      9576      9585      9619      9630#
L10151  042310R    002     9693      9708      9718      9748      9762      9771      9800      9813#
L10152  042674R    002     9856      9871      9881      9911      9925      9934      9962      9974#
L10153  043334R    002    10095     10110     10120     10162     10176     10185     10235#
L10154  043664R    002    10318     10333     10343     10372     10395     10424#
L10155  044204R    002    10473     10488     10498     10518     10534     10543     10570#
L10156  044554R    002    10633     10648     10658     10684     10699     10708     10737     10749#
L10157  045074R    002    10800     10815     10825     10844     10866     10875     10902#
L10160  045434R    002    10972     10987     10997     11023     11042     11051     11083#
L10161  046024R    002    11140     11155     11165     11193     11212     11221     11264#
L10162  046406R    002    11329     11344     11354     11396     11410     11419     11466#
L10163  047040R    002    11537     11552     11562     11595     11609     11618     11675     11686#
```

```
L10164   047444R    002   11752    11767    11777    11806    11820    11829    11883#
L10165   050534R    002   11988    12003    12013    12047    12061    12070    12124    12144    12196    12210    12219    12256#
L10166   051256R    002   12331    12346    12356    12380    12394    12403    12447    12461    12470    12485    12507#
L10167   051672R    002   12578    12593    12603    12632    12646    12655    12668    12699    12707#
L10170   052746R    002   12733    12739    12760    12773    12794    12807    12838    12851    12883    12895    12956#
L10171   053504R    002   13081    13095#
MAXBYT=  002756 G          1668#    9118     9288     9727     9890
MAXCNT   003122RG   002    2182#    9350     9360     9381     9396     9405     9418     9428
METER    000332RG   002    1797#    4245     4386*    4392     4459*    4579*    4607*    4788*    4898     4900*    5296*    5647*    5718*
                           6175*    6240*    6372*    6415*    6477*    6602*    6649*    6697*    6777*    6828*    6923*    7053*    7117*
                           7127     7235*    7282*    7298     7301     7381*    7437*    7546*    7601*    7705*    7754*    7855*    7901*
                           7993*    8040*    8131*    8178*    8279*    8333*    8441*    8494*    8594*    8651*    8771*    8822*    8998*
                           9054*    9167*    9223*    9340*    9388*    9505*    9564*    9694*    9750*    9857*    9913*   10096*   10164*
                          10319*   10374*   10474*   10519*   10634*   10687*   10801*   10854*   10973*   11030*   11141*   11200*   11330*
                          11398*   11538*   11597*   11753*   11808*   11989*   12049*   12126*   12198*   12332*   12382*   12449*   12579*
                          12634*   12744*   12778*   12822*   12867*
MICASZ=  ****** GX         4514     6589
MICBSZ=  ****** GX         4521
MICCSZ=  ****** GX         4528
MICDSZ=  ****** GX         4535
MICESZ=  ****** GX         4542
MICFSZ=  ****** GX         4549
MICGSZ=  ****** GX         4553
MICMOD   000320RG   002    1790#    4499*    4503     4569     4584     4597     4612     4625
MICRO    000326RG   002    1795#    4499     4511     4518     4525     4532     4539     4546     6770     6774*    7043     7047*    7225
                           7229*    7371     7375*    7536     7540*    7695     7699*    7848     7852*    7983     7987*    8121     8125*
                           8269     8273*    8431     8435*    8584     8588*    8761     8765*    8988     8992*    9157     9161*    9330
                           9334*    9495     9499*    9684     9688*    9847     9851*   10086    10090*   10309    10313*   10464    10468*
                          10624    10628*   10791    10795*   10963    10967*   11131    11135*   11320    11324*   11528    11532*   11743
                          11747*   11979    11983*   12322    12326*   12569    12573*
MICROA=  ****** GX         4515     6590
MICROB=  ****** GX         4522
MICROC=  ****** GX         4529
MICROD=  ****** GX         4536
MICROE=  ****** GX         4543
MICROF=  ****** GX         4550
MICROG=  ****** GX         4554
MINBYT=  000100 G          1669#    9030     9199    11682    12368    12612
MSG1     012750RG   002    2961#    5660     5731     6187     6253     6384     6427     6488     6614     6661     6709     6788     6935
                           7065     7246     7392     7557     7716     7866     8004     8142     8290     8452     8605     8782     9009
                           9178     9351     9516     9705     9868    10107    10330    10485    10645    10812    10984    11152    11341
                          11549    11764    12000    12343    12590    12757    12791    12835    12880
MSG10    013372RG   002    3108#    7290
MSG11    013414RG   002    3120#    7311
MSG12    013500RG   002    3145#    6844     7455     7619     7772     7919     8065     8203     8351     8512     8669     8840     9072
                           9241     9406     9582     9768     9931    10182    10540    10705    10872    11048    11218    11416    11615
                          11826    12067    12141    12216    12400    12467    12652
MSG13    013524RG   002    3158#    7636
MSG14    013572RG   002    3178#    7791
MSG15    013620RG   002    3192#    7936
MSG16    013650RG   002    3207#    6864     7478
MSG17    013742RG   002    3235#    8369     8530     8687
MSG18    014034RG   002    3263#    8861     8874     8883     8895     8904     8913
MSG19    014064RG   002    3277#    9095     9265
MSG2     013012RG   002    2977#    6293
MSG20    014160RG   002    3306#   10199
```

```
MSG21    014250RG   002    3333#  10215
MSG22    014340RG   002    3360#  12089    12161    12240
MSG23    014454RG   002    3395#  12414    12482
MSG24    014476RG   002    3407#  12696
MSG25    014626RG   002    3445#  10400
MSG26    015014RG   002    3493#  10718
MSG27    015060RG   002    3512#  11234
MSG28    015126RG   002    3532#  11242
MSG29    015174RG   002    3552#  11430
MSG3     013056RG   002    2996#   5306     5335     5357     5412     5437     5456
MSG30    015276RG   002    3580#  11632
MSG31    015344RG   002    3599#  11647
MSG32    015510RG   002    3640#  11658
MSG33    015606RG   002    3667#   7178
MSG34    015634RG   002    3680#   5753
MSG35    015704RG   002    3700#   9602
MSG36    016002RG   002    3727#   8049
MSG37    016024RG   002    3739#   8187
MSG38    016046RG   002    3751#   9419
MSG39    016070RG   002    3763#  11840
MSG4     013120RG   002    3012#   5388
MSG40    016112RG   002    3775#  11855
MSG41    016210RG   002    3802#  11866
MSG42    016262RG   002    3822#   9783     9945
MSG43    016432RG   002    3864#   7485
MSG44    016454RG   002    3876#   6837     7446     7610     7763     7910     8056     8194     8342     8503     8660     8831     9063
                           9232    9397     9573     9759     9922    10173    10392    10531    10696    10863    11039    11209    11407
                          11606   11817    12058    12135    12207    12391    12458    12643
MSG45    016616RG   002    3915#
MSG46    016700RG   002    3939#   6816     7111     7278     7425     7595     7749     7896     8035     8173     8323     8484     8646
                           8813    9049     9218     9382     9559     9745     9908    10159    10369    10515    10681    10841    11020
                          11190   11393    11592    11803    12044    12121    12193    12377    12444    12629
MSG5     013142RG   002    3024#   6522
MSG6     013212RG   002    3045#   6719
MSG7     013300RG   002    3071#   6963
MSG8     013322RG   002    3083#   6979
MSG9     013344RG   002    3095#   7139
MUCAST   003563RG   002    2232#  10811    10821    10840    10862    10871    10885    10895
NEXMEM   000664RG   002    1905#   4784     4853*    4937*    4940     5040*    5043     5186*    5189     5236*    5239
NIHLT  = 000006 G          1729#
NIUNI  = 000007 G          1730#
NCCLK    001566RG   002    2048#   4714
NOPERR   002001RG   002    2074#   5612     5634     5659     5676
NPRERR=  100000 G          1693#   3877     4425     4427
NXMERR=  040000 G          1694#   3885     4425     4427
ODDBYT   003060RG   002    2176#   9177     9187     9217     9231     9240     9264     9278
ONEFIL=  000001               1#   1387#    1390     1565#    1567     4641#    4651     4907     4908#   13066#   13068
OWN    = 100000 G          1733#
O$APTS=  000000            1389#   1429
O$AU   = 000001            1389#   1400#    1461
O$BGNR=  000001            1389#   1400#    1455
O$BGNS=  000000            1389#   1421
O$DU   = 000001            1389#   1400#    1463
O$ERRT=  000000            1389#   1471
O$GNSW=  000000            1389#   1425
O$POIN=  000001            1389#   1400#    1487
```

```
O$SETU= 000000              1389#    1415
PARERR= 010000 G            1696#    3901     4425     4427     7479     7486
PATERN  000516RG    002     1863#    5489     6340     7579     8631    11006    11176    11362
PAT1    000516RG    002     1864#    6233
PAT2    000520RG    002     1865#
PAT3    000522RG    002     1866#
PAT4    000524RG    002     1867#
PAT5    000526RG    002     1868#
PAT6    000530RG    002     1869#    6533
PCBB    000604RG    002     1893#    3371     3381     3554     3562     3569     3588     3628     3648     3656     3702     3715     3716
                            3790     3810     3824     3831     3845     3852     3853     4560*    4561*    4562*    4563     4634*    4635*
                            6172     6222*    6223*    6224*    6365*    6366*    6367*    6368     6475*    6595*    6596*    6597*    6598
                            6694*    6695*    6820     6822*    6823*    6824*    6825*    6848     6852     6857     6858     6859     7116*
                            7429     7431*    7432*    7433*    7434*    7464     7468     7471     7472     7473     7599*    7627     7629
                            7631     7734*    7782     7784     7809     7811*    7889*    7927     8327*    8329*    8330*    8331*    8361
                            8362     8363     8364     8488*    8490*    8491*    8492*    8522     8523     8524     8525     8627*    8629*
                            8630*    8633*    8679     8680     8681     8682     8817*    8820*    8854     8867     8876     8888     8897
                            8906     9030*    9039*    9041*    9083     9085     9086     9117*    9118     9199*    9208*    9210*    9252
                            9254     9255     9287*    9288     9550*    9551*    9594*    9595*    9596     9728*    9730*    9807     9809*
                            9891*    9893*    9968     9970*   10354*   10355*   10356*   10685*   11008*   11174*   11178*   11226    11368*
                           11381*   11424    11577*   11578*   11580*   11639*   11640    11642*   11652    11788*   11790*   11847*   11848
                           11850*   11860    12023    12035*   12036*   12037*   12077    12079    12081    12083    12112*   12113*   12114*
                           12149    12151    12153    12155    12184*   12185*   12186*   12228    12230    12232    12234    12368*   12369*
                           12408    12432*   12475    12504    12612*   12613*   12673    12741    12777*   12811    12812    12813    12821*
                           12855    12860*   12861*   12862*
PCEI  = 040000 G            1679#    4311     4313     4314
PCOMND  000316RG    002     1789#    2963     5655*    5726*    6182*    6248*    6379*    6422*    6483*    6609*    6656*    6704*    6783*
                            6930*    7060*    7241*    7387*    7552*    7711*    7861*    7999*    8137*    8285*    8447*    8600*    8777*
                            9004*    9173*    9346*    9511*    9700*    9863*   10102*   10325*   10480*   10640*   10807*   10979*   11147*
                           11336*   11544*   11759*   11995*   12338*   12585*   12752*   12786*   12830*   12875*
PCSR0   000336RG    002     1800#    4241     4275*    4276     4304     4306*    4307     4311     4313*    4314     4318     4320*    4321
                            4325     4327*    4328     4332     4334*    4335     4339     4341*    4342     4346     4348*    4349     4424
                            4427*    4458*    4578*    4606*    4636*    4748*    4749     4786*    4872     4939     4974     4991     5295*
                            5326     5349     5601     5646*    5719*    6174*    6241*    6371*    6414*    6476*    6601*    6648*    6696*
                            6826*    6921*    6922*    6987*    7118*    7120*    7161*    7162     7283*    7435*    7479     7486*    7600*
                            7753*    7900*    8039*    8177*    8332*    8493*    8650*    8821*    9053*    9222*    9386*    9563*    9749*
                            9912*   10163*   10373*   10521*   10686*   10849*   11020*   11199*   11397*   11596*   11807*   12048*   12125*
                           12197*   12381*   12448*   12633*   12743*   12779*   12823*   12868*
PCSR0C  000346RG    002     1805#
PCSR1   000340RG    002     1801#    3446     4388     4749*    4750*    4751     5042     5077     5094     5111     5128     5145     5379
                            5404     5428     5447     5736     6710     6772     6846     7045     7227     7373     7462     7538     7697
                            7850     7985     8123     8271     8359     8433     8520     8586     8677     8763     8990     9159     9332
                            9413     9497     9686     9777     9849     9939    10088    10190    10311    10466    10548    10626    10712
                           10793    10880    10965    11055    11133    11322    11530    11626    11745    11834    11981    12324    12571
PCSR1C  000350RG    002     1806#
PCSR2   000342RG    002     1802#    4563*    4751*    4752*    4753     5188     5494*    5495     6172*    6368*    6598*    6820*    7429*
                           10129*   12023*   12741*
PCSR2C  000352RG    002     1807#
PCSR3   000344RG    002     1803#    4564*    4753*    4754*    5238     5536*    5537     5552*    5553     6173*    6369*    6599*    6821*
                            7430*   10130*   12024*   12742*
PCSR3C  000354RG    002     1808#
PCTO  = 000200 G            1710#
PFNOP1= 000000 G            1741#
PFNOP2= 000003 G            1744#
PNOP  = 000006 G            1701#    5646     6922
PNT   = 001000 G            1637#
```

```
POLYH = 166670 G             1663#   4019
POLYL = 101440 G             1664#   4020
POLY16= 120001 G             1665#   4062
PRI   = 002000 G             1638#
PRILD = 000001 G             1725#
PRI00 = 000000 G             1626#
PRI01 = 000040 G             1625#
PRI02 = 000100 G             1624#
PRI03 = 000140 G             1623#
PRI04 = 000200 G             1622#   7123
PRI05 = 000240 G             1621#   4200    4747
PRI06 = 000300 G             1620#
PRI07 = 000340 G             1619#   4214    4925    5028    5180    5228
PRTPAR  004154RG     002     2277#  12756   12769   12790   12803   12834   12847   12879   12891
PSTATE= 000007 G             1723#   5380    6711
PWHEN   000314RG     002     1788#   2964    2998    4465*   4468    5287*   5399*   5654*   5725*   6181*   6247*   6378*   6421*
                             6462*   6608*   6655*   6703*   6782*   6929*   7059*   7240*   7386*   7551*   7710*   7860*   7998*
                             8136*   8284*   8446*   8599*   8776*   9003*   9172*   9345*   9510*   9699*   9862*  10101*  10324*
                            10479*  10639*  10806*  10978*  11146*  11335*  11543*  11758*  11994*  12337*  12584*  12751*  12785*
                            12829*  12874*
RACC  = 000013 G             1752#
RACERR  001663RG     002     2059#   4945    4979    4996    5048    5082    5099    5116    5133    5150    5194    5244
RACMG1  012546RG     002     2888#   4946    5049    5195    5245
RACMG2  012574RG     002     2901#   4980    4997    5083    5100    5117    5134    5151    7151
RACMG3  012632RG     002     2916#   5504    5544    5560
RACMG4  012704RG     002     2937#   5613
RACMG7  012726RG     002     2949#   5635    5677    5766    6207    6273    6404    6447    6505    6634    6681    6796    6871
                             6994    7075    7256    7318    7402    7497    7567    7649    7726    7805    7874    8014    8078
                             8152    8216    8300    8379    8462    8540    8615    8697    8792    8923    9019    9109    9188
                             9279    9361    9429    9526    9616    9715    9797    9878    9959   10117   10229   10340   10411
                            10495   10564   10655   10734   10822   10896   10994   11071   11162   11252   11351   11440   11559
                            11672   11774   11880   12010   12100   12172   12250   12353   12424   12495   12600   12665   12770
                            12804   12848   12892
RACPS = 000017 G             1756#
RBRRUN  003743RG     002     2253#  11548   11558   11591   11605   11614   11631   11646   11657   11671
RC    = 000012 G             1751#
RCEI  = 002000 G             1683#   4339    4341    4342
RCVDON  002572RG     002     2145#   8141    8151    8172    8186    8193    8202    8215
RDPA  = 000002 G             1743#  12777
READY = 000002 G             1726#   5381
RESET = 000000 G             1724#
RETLOG  004124RG     002     2273#  12589   12599   12628   12642   12651   12664   12695
REUNA   020104RG     002     4458#   4500    6728    6872    7942   12735
RLNKAD  003207RG     002     2191#   9704    9714    9744    9758    9767    9782    9796
RLSA  = 000024 G             1761#
RMAL  = 000006 G             1747#
RMTC  = 000010 G             1712#
ROMADR= 040000 G             1658#   1659    6225
ROMDMP  002042RG     002     2080#   6186    6206    6252    6272    6292
ROMSIZ= 040000 G             1654#   1659    6237    6284
RPA   = 000004 G             1745#
RPS   = 000016 G             1755#  12821
RREV    052756RG     002    12855*  12856*  12926   12963#
RRF   = 000010 G             1749#
RRWER   001741RG     002     2068#   5503    5543    5559
RSET  = 000040 G             1688#   4458    4786    5295
```

```
RSETER  001717RG   002   2064#   5305    5334    5356    5387    5411    5436    5455
RSIDP = 000022 G         1759#
RTM   = 000014 G         1753#
RUN   = 000003 G         1727#
RXI   = 020000 G         1680#   4318    4320    4321
SECOND= 000077 G         1645#   4386    4459    4579    4607    4788    5296    5647    5718    6175    6240    6372    6415
                         6477    6602    6649    6697    6777    6828    6923    7053    7117    7235    7381    7437    7546
                         7601    7705    7754    7855    7901    7993    8040    8131    8178    8279    8333    8441    8494
                         8594    8651    8771    8822    8998    9054    9167    9223    9340    9388    9505    9564    9694
                         9750    9857    9913    10096   10164   10319   10374   10474   10519   10634   10687   10801   10854
                         10973   11030   11141   11200   11330   11398   11538   11597   11753   11808   11989   12049   12126
                         12198   12332   12382   12449   12579   12634   12744   12778   12822   12867
SERI  = 100000 G         1678#   4304    4306    4307
SET   = 000001 G         1646#
SFT   = 037400 G         1719#   5737    5740    6711
SFTB0 = 000400 G         1713#   6712
SFTB1 = 001000 G         1714#
SFTB2 = 002000 G         1715#   6712
SFTB3 = 004000 G         1716#
SFTB4 = 010000 G         1717#   6712
SFTB5 = 020000 G         1718#
SIZ1K = 004000 G         1648#   1649    6232    6282    9539    9624    9626    9730    9807    9809    9893    9968    9970
                         10141
SIZ2K = 010000 G         1649#   1650
SIZ4K = 020000 G         1650#   1651    1652    1653
SIZ8K = 040000 G         1651#   1654    1655
SLFT  = 000003 G         1702#   5719
SLFTST  002021RG   002   2077#   5713    5730    5752    5765
SMSG00  024404RG   002   5778    5843#
SMSG01  024432RG   002   5779    5847#
SMSG02  024457RG   002   5780    5851#
SMSG03  024470RG   002   5781    5853#
SMSG04  024525RG   002   5782    5858#
SMSG05  024552RG   002   5783    5862#
SMSG06  024613RG   002   5784    5868#
SMSG07  024644RG   002   5785    5873#
SMSG10  024660RG   002   5786    5875#
SMSG11  024710RG   002   5787    5879#
SMSG12  024721RG   002   5788    5881#
SMSG13  024734RG   002   5789    5883#
SMSG14  024746RG   002   5790    5885#
SMSG15  024760RG   002   5791    5887#
SMSG16  024772RG   002   5792    5889#
SMSG17  025004RG   002   5793    5891#
SMSG20  025016RG   002   5794    5893#
SMSG21  025037RG   002   5795    5896#
SMSG22  025051RG   002   5796    5898#
SMSG23  025063RG   002   5797    5900#
SMSG24  025075RG   002   5798    5902#
SMSG25  025107RG   002   5799    5904#
SMSG26  025121RG   002   5800    5906#
SMSG27  025133RG   002   5801    5908#
SMSG30  025145RG   002   5802    5910#
SMSG31  025220RG   002   5803    5918#
SMSG32  025270RG   002   5804    5925#
SMSG33  025341RG   002   5805    5932#
```

```
SMSG34  025403RG   002   5806     5938#
SMSG35  025452RG   002   5807     5945#
SMSG36  025513RG   002   5808     5951#
SMSG37  025562RG   002   5809     5958#
SMSG40  025623RG   002   5810     5964#
SMSG41  025711RG   002   5811     5974#
SMSG42  025774RG   002   5812     5983#
SMSG43  026060RG   002   5813     5992#
SMSG44  026135RG   002   5814     6000#
SMSG45  026217RG   002   5815     6009#
SMSG46  026273RG   002   5816     6017#
SMSG47  026305RG   002   5817     6019#
SMSG50  026317RG   002   5818     6021#
SMSG51  026403RG   002   5819     6030#
SMSG52  026464RG   002   5820     6039#
SMSG53  026546RG   002   5821     6048#
SMSG54  026621RG   002   5822     6056#
SMSG55  026701RG   002   5823     6065#
SMSG56  026753RG   002   5824     6073#
SMSG57  026765RG   002   5825     6075#
SMSG60  026777RG   002   5826     6077#
SMSG61  027010RG   002   5827     6079#
SMSG62  027031RG   002   5828     6082#
SMSG63  027043RG   002   5829     6084#
SMSG64  027070RG   002   5830     6088#
SMSG65  027102RG   002   5831     6090#
SMSG66  027121RG   002   5832     6093#
SMSG67  027133RG   002   5833     6095#
SMSG70  027145RG   002   5834     6097#
SMSG71  027164RG   002   5835     6100#
SMSG72  027215RG   002   5836     6105#
SMSG73  027237RG   002   5837     6109#
SMSG74  027251RG   002   5838     6111#
SMSG75  027263RG   002   5839     6113#
SMSG76  027275RG   002   5840     6115#
SMSG77  027307RG   002   5841     6117#
SPAT1   000532RG   002   1871#   10352
SPAT2   000540RG   002   1874#
SPAT3   000546RG   002   1877#
SPAT4   000554RG   002   1880#
SPAT5   000562RG   002   1883#
SPAT6   000570RG   002   1886#
SPAT7   000576RG   002   1889#
STAMUX  002756RG   002   2165#    8781     8791     8812     8830     8839     8860     8873     8882     8894     8903     8912     8922
STAPAT  003400RG   002   2212#   10329    10339    10368    10391    10399    10410
STAPOS  003512RG   002   2225#   10644    10654    10680    10695    10704    10717    10733
STAREJ  003444RG   002   2218#   10484    10494    10514    10530    10539    10553    10563
STMSG   024202RG   002   3689     5748*    5774#
STP   = 001000 G         1735#
STTBL   024204RG   002   5747     5778#
SVCGBL= 000001           1389#    1402     1411     1413     1415     1417     1419     1421     1423     1425     1427     1429     1431
                         1433     1435     1437     1439     1441     1443     1445     1448     1451     1453     1455     1457     1459
                         1461     1463     1465     1467     1469     1471     1473     1475     1477     1479     1481     1483     1485
                         1497     1556     1557     1928     1936     2888     2901     2916     2937     2949     2961     2977     2996
                         3012     3024     3045     3071     3083     3095     3108     3120     3145     3158     3178     3192     3207
                         3235     3263     3277     3306     3333     3360     3395     3407     3445     3493     3512     3532     3552
```

```
                              3580    3599    3640    3667    3680    3700    3727    3739    3751    3763    3775    3802    3822
                              3864    3876    3915    3939    4644    4661    4677    4771    4783    4818    4829    4852    4871
                              4897   13082
SVCINS= 000001                1389#   1403    1404    1405    1406    1407    1408    1409    1410    1412    1414    1416    1418
                              1420    1422    1424    1426    1428    1430    1432    1434    1436    1438    1440    1442    1444
                              1446    1447    1449    1450    1452    1454    1456    1458    1460    1462    1464    1466    1468
                              1470    1472    1474    1476    1478    1480    1482    1484    1486    1496    1498    1499    1500
                              1501    1502    1503    1504    1505    1506    1507    1508    1509    1510    1511    1512    1513
                              1514    1515    1516    1517    1518    1519    1520    1521    1522    1523    1524    1525    1526
                              1527    1528    1529    1530    1531    1532    1533    1534    1535    1536    1537    1538    1539
                              1540    1541    1542    1543    1555    1929    1930    1937    1941    2890    2891    2892    2893
                              2894    2895    2898    2903    2904    2905    2906    2907    2908    2909    2910    2913    2918
                              2919    2920    2921    2922    2923    2925    2926    2927    2928    2929    2930    2931    2934
                              2939    2940    2941    2942    2943    2946    2951    2952    2953    2954    2955    2958    2963
                              2964    2965    2966    2967    2968    2969    2970    2971    2974    2979    2980    2981    2982
                              2983    2985    2986    2987    2988    2989    2990    2993    2998    2999    3000    3001    3002
                              3003    3004    3005    3006    3009    3014    3015    3016    3017    3018    3021    3026    3027
                              3028    3029    3030    3032    3033    3034    3035    3036    3037    3038    3039    3042    3047
                              3048    3049    3050    3051    3053    3054    3055    3056    3057    3058    3060    3061    3062
                              3063    3064    3065    3068    3073    3074    3075    3076    3077    3080    3085    3086    3087
                              3088    3089    3092    3097    3098    3099    3100    3101    3102    3105    3110    3111    3112
                              3113    3114    3117    3122    3123    3124    3125    3126    3128    3129    3130    3131    3132
                              3134    3135    3136    3137    3138    3139    3142    3147    3148    3149    3150    3151    3152
                              3155    3160    3161    3162    3163    3164    3165    3167    3168    3169    3170    3171    3172
                              3175    3180    3181    3182    3183    3184    3185    3186    3189    3194    3195    3196    3197
                              3198    3199    3200    3201    3204    3209    3210    3211    3212    3213    3214    3216    3217
                              3218    3219    3220    3221    3223    3224    3225    3226    3227    3228    3229    3232    3237
                              3238    3239    3240    3241    3242    3244    3245    3246    3247    3248    3249    3251    3252
                              3253    3254    3255    3256    3257    3260    3265    3266    3267    3268    3269    3270    3271
                              3274    3279    3280    3281    3282    3283    3284    3286    3287    3288    3289    3290    3291
                              3292    3294    3295    3296    3297    3298    3299    3300    3303    3308    3309    3310    3311
                              3312    3314    3315    3316    3317    3318    3319    3321    3322    3323    3324    3325    3326
                              3327    3330    3335    3336    3337    3338    3339    3341    3342    3343    3344    3345    3346
                              3348    3349    3350    3351    3352    3353    3354    3357    3363    3364    3365    3366    3367
                              3368    3371    3372    3373    3374    3375    3376    3377    3378    3381    3382    3383    3384
                              3385    3386    3387    3388    3392    3397    3398    3399    3400    3401    3404    3409    3410
                              3411    3412    3413    3415    3416    3417    3418    3419    3424    3425    3426    3427    3428
                              3429    3431    3432    3433    3434    3435    3436    3442    3449    3450    3451    3452    3453
                              3456    3457    3458    3459    3460    3462    3463    3464    3465    3466    3468    3469    3470
                              3471    3472    3473    3475    3476    3477    3478    3479    3480    3482    3483    3484    3485
                              3486    3487    3490    3495    3496    3497    3498    3499    3501    3502    3503    3504    3505
                              3506    3509    3514    3515    3516    3517    3518    3520    3521    3522    3523    3524    3525
                              3526    3529    3534    3535    3536    3537    3538    3540    3541    3542    3543    3544    3545
                              3546    3549    3554    3555    3556    3557    3558    3559    3560    3562    3563    3564    3565
                              3566    3567    3569    3570    3571    3572    3573    3574    3577    3582    3583    3584    3585
                              3586    3588    3589    3590    3591    3592    3593    3596    3601    3602    3603    3604    3605
                              3607    3608    3609    3610    3611    3612    3614    3615    3616    3617    3618    3619    3621
                              3622    3623    3624    3625    3626    3628    3629    3630    3631    3632    3633    3634    3637
                              3642    3643    3644    3645    3646    3648    3649    3650    3651    3652    3653    3654    3656
                              3657    3658    3659    3660    3661    3664    3669    3670    3671    3672    3673    3674    3677
                              3682    3683    3684    3685    3686    3687    3689    3690    3691    3692    3693    3694    3697
                              3702    3703    3704    3705    3706    3707    3709    3710    3711    3712    3713    3715    3716
                              3717    3718    3719    3720    3721    3724    3729    3730    3731    3732    3733    3736    3741
                              3742    3743    3744    3745    3748    3753    3754    3755    3756    3757    3760    3765    3766
                              3767    3768    3769    3772    3777    3778    3779    3780    3781    3783    3784    3785    3786
                              3787    3788    3790    3791    3792    3793    3794    3795    3796    3799    3804    3805    3806
```

```
3807   3808   3810   3811   3812   3813   3814   3815   3816   3819   3824   3825   3826
3827   3828   3829   3831   3832   3833   3834   3835   3836   3838   3839   3840   3841
3842   3843   3845   3846   3847   3848   3849   3850   3852   3853   3854   3855   3856
3857   3858   3861   3866   3867   3868   3869   3870   3873   3880   3881   3882   3883
3884   3888   3889   3890   3891   3892   3896   3897   3898   3899   3900   3904   3905
3906   3907   3908   3912   3917   3918   3919   3920   3921   3923   3924   3925   3926
3927   3929   3930   3931   3932   3933   3936   3941   3942   3943   3944   3945   3948
4200   4201   4214   4215   4244   4353   4354   4355   4356   4357   4358   4391   4468
4469   4470   4471   4472   4473   4474   4475   4476   4482   4483   4484   4485   4486
4503   4504   4505   4506   4507   4508   4569   4570   4571   4572   4573   4574   4584
4585   4586   4587   4588   4589   4597   4598   4599   4600   4601   4602   4612   4613
4614   4615   4616   4617   4625   4626   4627   4628   4629   4630   4646   4647   4650
4682   4683   4685   4687   4688   4690   4692   4693   4695   4697   4698   4700   4703
4704   4708   4709   4710   4712   4714   4715   4716   4717   4718   4727   4728   4729
4730   4731   4732   4740   4741   4742   4744   4758   4762   4774   4792   4793   4794
4795   4796   4800   4801   4814   4820   4821   4824   4831   4832   4835   4856   4876
4904   4925   4926   4927   4928   4929   4930   4933   4943   4944   4945   4946   4948
4950   4951   4953   4954   4956   4960   4963   4968   4977   4978   4979   4980   4984
4986   4994   4995   4996   4997   5001   5004   5006   5007   5010   5028   5029   5030
5031   5032   5033   5036   5046   5047   5048   5049   5051   5053   5054   5056   5057
5059   5063   5066   5071   5080   5081   5082   5083   5087   5089   5097   5098   5099
5100   5104   5106   5114   5115   5116   5117   5121   5123   5131   5132   5133   5134
5138   5140   5148   5149   5150   5151   5155   5158   5160   5161   5164   5180   5181
5182   5183   5184   5185   5192   5193   5194   5195   5197   5199   5200   5202   5203
5205   5207   5208   5212   5228   5229   5230   5231   5232   5233   5235   5242   5243
5244   5245   5247   5249   5250   5252   5253   5255   5259   5261   5262   5265   5291
5303   5304   5305   5306   5308   5309   5313   5317   5322   5332   5333   5334   5335
5339   5348   5354   5355   5356   5357   5361   5374   5385   5386   5387   5388   5392
5403   5409   5410   5411   5412   5416   5424   5434   5435   5436   5437   5441   5443
5453   5454   5455   5456   5460   5463   5466   5493   5501   5502   5503   5504   5508
5513   5535   5541   5542   5543   5544   5548   5551   5557   5558   5559   5560   5564
5567   5597   5610   5611   5612   5613   5617   5620   5632   5633   5634   5635   5639
5642   5657   5658   5659   5660   5662   5663   5666   5674   5675   5676   5677   5681
5684   5687   5711   5712   5713   5714   5716   5717   5728   5729   5730   5731   5733
5734   5750   5751   5752   5753   5755   5756   5763   5764   5765   5766   5770   6166
6171   6184   6185   6186   6187   6189   6190   6194   6196   6204   6205   6206   6207
6211   6214   6217   6230   6250   6251   6252   6253   6255   6256   6260   6262   6270
6271   6272   6273   6277   6290   6291   6292   6293   6297   6300   6343   6348   6381
6382   6383   6384   6386   6387   6391   6393   6401   6402   6403   6404   6408   6410
6424   6425   6426   6427   6429   6430   6434   6436   6444   6445   6446   6447   6451
6454   6457   6462   6485   6486   6487   6488   6490   6491   6495   6497   6502   6503
6504   6505   6509   6519   6520   6521   6522   6528   6539   6583   6611   6612   6613
6614   6616   6617   6621   6623   6631   6632   6633   6634   6636   6637   6641   6643
6658   6659   6660   6661   6663   6664   6668   6670   6678   6679   6680   6681   6683
6684   6688   6693   6706   6707   6708   6709   6716   6717   6718   6719   6723   6732
6769   6785   6786   6787   6788   6793   6794   6795   6796   6801   6809   6813   6814
6815   6816   6818   6819   6834   6835   6836   6837   6841   6842   6843   6844   6861
6862   6863   6864   6868   6869   6870   6871   6875   6878   6912   6914   6915   6916
6917   6918   6919   6932   6933   6934   6935   6939   6944   6950   6951   6960   6961
6962   6963   6965   6966   6976   6977   6978   6979   6983   6985   6986   6991   6992
6993   6994   6998   7042   7051   7052   7062   7063   7064   7065   7067   7068   7072
7073   7074   7075   7077   7078   7080   7081   7082   7083   7084   7085   7088   7104
7108   7109   7110   7111   7113   7114   7123   7124   7136   7137   7138   7139   7148
7149   7150   7151   7155   7160   7175   7176   7177   7178   7182   7189   7224   7233
7234   7243   7244   7245   7246   7248   7249   7253   7254   7255   7256   7258   7259
7263   7271   7275   7276   7277   7278   7280   7281   7287   7288   7289   7290   7292
```

```
    7293   7308   7309   7310   7311   7315   7316   7317   7318   7322   7325   7370   7379
    7380   7389   7390   7391   7392   7394   7395   7399   7400   7401   7402   7404   7405
    7409   7418   7422   7423   7424   7425   7427   7428   7443   7444   7445   7446   7448
    7449   7452   7453   7454   7455   7457   7458   7475   7476   7477   7478   7482   7483
    7484   7485   7494   7495   7496   7497   7501   7504   7535   7544   7545   7554   7555
    7556   7557   7559   7560   7564   7565   7566   7567   7569   7570   7574   7588   7592
    7593   7594   7595   7597   7598   7607   7608   7609   7610   7612   7613   7616   7617
    7618   7619   7621   7622   7633   7634   7635   7636   7639   7646   7647   7648   7649
    7651   7652   7657   7694   7703   7704   7713   7714   7715   7716   7718   7719   7723
    7724   7725   7726   7728   7729   7733   7742   7746   7747   7748   7749   7751   7752
    7760   7761   7762   7763   7765   7766   7769   7770   7771   7772   7774   7775   7788
    7789   7790   7791   7795   7802   7803   7804   7805   7807   7808   7816   7847   7863
    7864   7865   7866   7871   7872   7873   7874   7879   7888   7893   7894   7895   7896
    7898   7899   7907   7908   7909   7910   7912   7913   7916   7917   7918   7919   7933
    7934   7935   7936   7945   7948   7982   7991   7992   8001   8002   8003   8004   8006
    8007   8011   8012   8013   8014   8016   8017   8021   8028   8032   8033   8034   8035
    8037   8038   8046   8047   8048   8049   8053   8054   8055   8056   8058   8059   8062
    8063   8064   8065   8067   8068   8075   8076   8077   8078   8082   8085   8120   8129
    8130   8139   8140   8141   8142   8144   8145   8149   8150   8151   8152   8154   8155
    8159   8166   8170   8171   8172   8173   8175   8176   8184   8185   8186   8187   8191
    8192   8193   8194   8196   8197   8200   8201   8202   8203   8205   8206   8213   8214
    8215   8216   8220   8223   8268   8277   8278   8287   8288   8289   8290   8292   8293
    8297   8298   8299   8300   8302   8303   8307   8316   8320   8321   8322   8323   8325
    8326   8339   8340   8341   8342   8344   8345   8348   8349   8350   8351   8353   8354
    8366   8367   8368   8369   8376   8377   8378   8379   8383   8386   8430   8439   8440
    8449   8450   8451   8452   8454   8455   8459   8460   8461   8462   8464   8465   8469
    8477   8481   8482   8483   8484   8486   8487   8500   8501   8502   8503   8505   8506
    8509   8510   8511   8512   8514   8515   8527   8528   8529   8530   8537   8538   8539
    8540   8544   8547   8583   8592   8593   8602   8603   8604   8605   8607   8608   8612
    8613   8614   8615   8617   8618   8622   8635   8643   8644   8645   8646   8648   8649
    8657   8658   8659   8660   8662   8663   8666   8667   8668   8669   8671   8672   8684
    8685   8686   8687   8694   8695   8696   8697   8701   8706   8760   8769   8770   8779
    8780   8781   8782   8784   8785   8789   8790   8791   8792   8794   8795   8799   8806
    8810   8811   8812   8813   8815   8816   8828   8829   8830   8831   8833   8834   8837
    8838   8839   8840   8842   8843   8858   8859   8860   8861   8871   8872   8873   8874
    8880   8881   8882   8883   8892   8893   8894   8895   8901   8902   8903   8904   8910
    8911   8912   8913   8920   8921   8922   8923   8927   8930   8987   8996   8997   9006
    9007   9008   9009   9011   9012   9016   9017   9018   9019   9021   9022   9026   9038
    9046   9047   9048   9049   9051   9052   9060   9061   9062   9063   9065   9066   9069
    9070   9071   9072   9074   9075   9092   9093   9094   9095   9099   9106   9107   9108
    9109   9111   9112   9122   9156   9165   9166   9175   9176   9177   9178   9180   9181
    9185   9186   9187   9188   9190   9191   9195   9207   9215   9216   9217   9218   9220
    9221   9229   9230   9231   9232   9234   9235   9238   9239   9240   9241   9243   9244
    9262   9263   9264   9265   9269   9276   9277   9278   9279   9281   9282   9292   9329
    9338   9339   9348   9349   9350   9351   9353   9354   9358   9359   9360   9361   9363
    9364   9368   9375   9379   9380   9381   9382   9384   9385   9394   9395   9396   9397
    9399   9400   9403   9404   9405   9406   9408   9409   9416   9417   9418   9419   9426
    9427   9428   9429   9433   9436   9494   9503   9504   9513   9514   9515   9516   9518
    9519   9523   9524   9525   9526   9528   9529   9533   9549   9556   9557   9558   9559
    9561   9562   9570   9571   9572   9573   9575   9576   9579   9580   9581   9582   9584
    9585   9599   9600   9601   9602   9606   9613   9614   9615   9616   9618   9619   9631
    9683   9692   9693   9702   9703   9704   9705   9707   9708   9712   9713   9714   9715
    9717   9718   9722   9734   9742   9743   9744   9745   9747   9748   9756   9757   9758
    9759   9761   9762   9765   9766   9767   9768   9770   9771   9780   9781   9782   9783
    9787   9794   9795   9796   9797   9799   9800   9814   9846   9855   9856   9865   9866
    9867   9868   9870   9871   9875   9876   9877   9878   9880   9881   9885   9901   9905
    9906    9907    9908    9910    9911    9919    9920    9921    9922    9924    9925    9928    9929
    9930    9931    9933    9934    9942    9943    9944    9945    9949    9956    9957    9958    9959
    9961    9962    9975   10085   10094   10095   10104   10105   10106   10107   10109   10110   10114
   10115   10116   10117   10119   10120   10124   10147   10156   10157   10158   10159   10161   10162
   10170   10171   10172   10173   10175   10176   10179   10180   10181   10182   10184   10185   10196
   10197   10198   10199   10212   10213   10214   10215   10226   10227   10228   10229   10233   10236
   10308   10317   10318   10327   10328   10329   10330   10332   10333   10337   10338   10339   10340
   10342   10343   10347   10358   10366   10367   10368   10369   10371   10372   10389   10390   10391
   10392   10394   10395   10397   10398   10399   10400   10408   10409   10410   10411   10415   10425
   10463   10472   10473   10482   10483   10484   10485   10487   10488   10492   10493   10494   10495
   10497   10498   10502   10508   10512   10513   10514   10515   10517   10518   10528   10529   10530
   10531   10533   10534   10537   10538   10539   10540   10542   10543   10551   10552   10553   10554
   10561   10562   10563   10564   10568   10571   10623   10632   10633   10642   10643   10644   10645
   10647   10648   10652   10653   10654   10655   10657   10658   10662   10674   10678   10679   10680
   10681   10683   10684   10693   10694   10695   10696   10698   10699   10702   10703   10704   10705
   10707   10708   10715   10716   10717   10718   10722   10727   10731   10732   10733   10734   10736
   10737   10741   10750   10790   10799   10800   10809   10810   10811   10812   10814   10815   10819
   10820   10821   10822   10824   10825   10829   10834   10838   10839   10840   10841   10843   10844
   10860   10861   10862   10863   10865   10866   10869   10870   10871   10872   10874   10875   10883
   10884   10885   10886   10893   10894   10895   10896   10900   10903   10962   10971   10972   10981
   10982   10983   10984   10986   10987   10991   10992   10993   10994   10996   10997   11001   11010
   11017   11018   11019   11020   11022   11023   11036   11037   11038   11039   11041   11042   11045
   11046   11047   11048   11050   11051   11058   11059   11060   11061   11068   11069   11070   11071
   11076   11084   11130   11139   11140   11149   11150   11151   11152   11154   11155   11159   11160
   11161   11162   11164   11165   11169   11183   11187   11188   11189   11190   11192   11193   11206
   11207   11208   11209   11211   11212   11215   11216   11217   11218   11220   11221   11231   11232
   11233   11234   11239   11240   11241   11242   11249   11250   11251   11252   11256   11265   11319
   11328   11329   11338   11339   11340   11341   11343   11344   11348   11349   11350   11351   11353
   11354   11358   11370   11380   11390   11391   11392   11393   11395   11396   11404   11405   11406
   11407   11409   11410   11413   11414   11415   11416   11418   11419   11427   11428   11429   11430
   11437   11438   11439   11440   11444   11457   11467   11527   11536   11537   11546   11547   11548
   11549   11551   11552   11556   11557   11558   11559   11561   11562   11566   11576   11589   11590
   11591   11592   11594   11595   11603   11604   11605   11606   11608   11609   11612   11613   11614
   11615   11617   11618   11629   11630   11631   11632   11644   11645   11646   11647   11655   11656
   11657   11658   11662   11669   11670   11671   11672   11674   11675   11687   11742   11751   11752
   11761   11762   11763   11764   11766   11767   11771   11772   11773   11774   11776   11777   11781
   11787   11800   11801   11802   11803   11805   11806   11814   11815   11816   11817   11819   11820
   11823   11824   11825   11826   11828   11829   11837   11838   11839   11840   11852   11853   11854
   11855   11863   11864   11865   11866   11870   11877   11878   11879   11880   11884   11978   11987
   11988   11997   11998   11999   12000   12002   12003   12007   12008   12009   12010   12012   12013
   12017   12033   12041   12042   12043   12044   12046   12047   12055   12056   12057   12058   12060
   12061   12064   12065   12066   12067   12069   12070   12086   12087   12088   12089   12097   12098
   12099   12100   12104   12109   12118   12119   12120   12121   12123   12124   12132   12133   12134
   12135   12138   12139   12140   12141   12143   12144   12158   12159   12160   12161   12169   12170
   12171   12172   12178   12183   12190   12191   12192   12193   12195   12196   12204   12205   12206
   12207   12209   12210   12213   12214   12215   12216   12218   12219   12237   12238   12239   12240
   12247   12248   12249   12250   12254   12257   12321   12330   12331   12340   12341   12342   12343
   12345   12346   12350   12351   12352   12353   12355   12356   12360   12367   12374   12375   12376
   12377   12379   12380   12388   12389   12390   12391   12393   12394   12397   12398   12399   12400
   12402   12403   12411   12412   12413   12414   12421   12422   12423   12424   12428   12434   12441
   12442   12443   12444   12446   12447   12455   12456   12457   12458   12460   12461   12464   12465
   12466   12467   12469   12470   12479   12480   12481   12482   12484   12485   12492   12493   12494
   12495   12499   12508   12568   12577   12578   12587   12588   12589   12590   12592   12593   12597
   12598   12599   12600   12602   12603   12607   12622   12626   12627   12628   12629   12631   12632
   12640   12641   12642   12643   12645   12646   12649   12650   12651   12652   12654   12655   12662
   12663   12664   12665   12667   12668   12693   12694   12695   12696   12698   12699   12705   12708
```

```
                        12732   12733   12738   12739   12754   12755   12756   12757   12759   12760   12767   12768   12769
                        12770   12772   12773   12788   12789   12790   1279    12793   12794   12801   12802   12803   12804
                        12806   12807   12832   12833   12834   12835   12837   12838   12845   12846   12847   12848   12850
                        12851   12877   12878   12879   12880   12882   12883   12889   12890   12891   12892   12894   12895
                        12919   12920   12921   12922   12923   12924   12926   12927   12928   12929   12930   12931   12933
                        12934   12935   12936   12937   12938   12940   12941   12942   12943   12944   12945   12947   12948
                        12949   12950   12951   12952   12957   13081   13084   13085   13086   13087   13089   13090   13091
                        13092   13094
SVCSUB= 000001           1389#   4932    4962    5035    5065    _90    5316    5596    5619    5641    6165    62'6    6342
                         6456
SVCTAG= 000001           1389#   1563    2897    2912    2933    2945    2957    2973    2992    3008    3020    3041    3067
                         3079    3091    3104    3116    3141    3154    3174    3188    3203    3231    3259    3273    3302
                         3329    3356    3391    3403    3441    3489    3508    3528    3548    3576    3595    3636    3663
                         3676    3696    3723    3735    3747    3759    3771    3798    3818    3860    3872    3911    3935
                         3947    4649    4761    4773    4813    4823    4834    4855    4875    4903    4959    4983    5000
                         5003    5009    5062    5086    5103    5120    5137    5154    5157    5163    5211    5258    5264
                         5312    5338    5360    5391    5415    5440    5459    5462    5465    5507    5512    5547    5563
                         5566    5616    5638    5680    5683    5686    5769    6193    6210    6213    6259    6276    6296
                         6299    6390    6407    6433    6450    6453    6494    6508    6527    6538    6620    6640    6667
                         6687    6722    6731    6800    6874    6877    6938    6982    6997    7087    7154    7181    7188
                         7262    7321    7324    7408    7500    7503    7573    7638    7656    7732    7794    7815    7878
                         7944    7947    8020    8081    8084    8158    8219    8222    8306    8382    8385    8468    8543
                         8546    8621    8700    8705    8798    8926    8929    9025    9098    9121    9194    9268    9291
                         9367    9432    9435    9532    9605    9630    9721    9786    9813    9884    9948    9974   10123
                        10232   10235   10346   10414   10424   10501   10567   10570   10661   10721   10740   10749   10828
                        10899   10902   11000   11075   11083   11168   11255   11264   11357   11443   11456   11466   11565
                        11661   11686   11780   11869   11883   12016   12103   12177   12253   12256   12359   12427   12498
                        12507   12606   12704   12707   12956   13095
SVCTST= 000001           1389#   4923    5026    5178    5226    5285    5487    5531    5594    5707    6163    6339    6578
                         6762    6906    7035    7217    7363    7528    7687    7840    7975    8113    8261    8423    8576
                         8753    8980    9149    9322    9487    9676    9839   10078   10701   10456   10616   10783   10955
                        11123   11312   11520   11735   11971   12314   12561   12728
SWADDR   000334RG   002  1798#  12866
SWHDR    053114RG   002 12933   13019#
SWPACK   052760RG   002 12864   12899   12907   12965#
S$LSYM= 010000           1389#   1564#   2898#   2913#   2934#   2946#   2958#   2974#   2993#   3009#   3021#   3042#   3068#
                         3080#   3092#   3105#   3117#   3142#   3155#   3175#   3189#   3204#   3232#   3260#   3274#   3303#
                         3330#   3357#   3392#   3404#   3442#   3490#   3509#   3529#   3549#   3577#   3596#   3637#   3664#
                         3677#   3697#   3724#   3736#   3748#   3760#   3772#   3799#   3819#   3861#   3873#   3912#   3936#
                         3948#   4650#   4762#   4774#   4814#   4824#   4835#   4856#   4876#   4904#   4960#   4968#   4986#
                         5004#   5010#   5063#   5071#   5089#   5106#   5123#   5140#   5158#   5164#   5212#   5235#   5265#
                         5313#   5322#   5348#   5374#   5403#   5424#   5443#   5463#   5466#   5493#   5513#   5535#   5551#
                         5567#   5617#   5639#   5666#   5684#   5687#   5770#   6171#   6196#   6214#   6230#   6262#   6297#
                         6300#   6348#   6393#   6410#   6436#   6454#   6462#   6497#   6528#   6539#   6583#   6623#   6643#
                         6670#   6693#   6732#   6769#   6809#   6878#   6912#   6944#   6998#   7042#   7104#   7160#   7189#
                         7224#   7271#   7325#   7370#   7418#   7504#   7535#   7588#   7657#   7694#   7742#   7816#   7847#
                         7888#   7948#   7982#   8028#   8 85#   8120#   8166#   8223#   8268#   8316#   8386#   8430#   8477#
                         8547#   8583#   8635#   8706#   8760#   8806#   8930#   8987#   9038#   9122#   9156#   9207#   9292#
                         9329#   9375#   9436#   9494#   9549#   9631#   9683#   9734#   9814#   9846#   9901#   9975#  10085#
                        10147#  10236#  10308#  10358#  10425#  10463#  10508#  10571#  10623#  10674#  10727#  10750#  10790#
                        10834#  10903#  10962#  11010#  11084#  11130#  11183#  11265#  11319#  11370#  11380#  11467#  11527#
                        11576#  11687#  11742#  11787#  11884#  11978#   2033#  12109#  12183#  12257#  12321#  12367#  12434#
                        12508#  12568#  12622#  12708#  12957#  13096#
TDRCNT   004074RG   002  2269#  12342   12352   12376   12390   12399   12413   12423   12443   12457   12466   12481   12494
TIMOFF   017314RG   002  4212#   4247    4250    4394    4397    4798    7129    7141
TIMON    017276RG   002  4199#   4240    4387    7121
```

```
TIMTST  002311RG    002     2111#   7245    7255    7277    7289    7310    7317
TLNKAD  003260RG    002     2198#   9867    9877    9907    9921    9930    9944    9958
TRAP4   021646RG    002     4852#   4926    5029    5181    5229
TRNDON  002540RG    002     2140#   8003    8013    8034    8048    8055    8064    8077
TXI   = 010000 G            1681#   4325    4327    4328
T$ARGC= 000002              1403#   1404#   1405#   1406#   1407#   1408#   1409#   2890#   2895    2903#   2910    2918#   2923
                            2925#   2931    2939#   2943    2951#   2955    2963#   2971    2979#   2983    2985#   2990    2998#
                            3006    3014#   3018    3026#   3030    3032#   3039    3047#   3051    3053#   3058    3060#   3065
                            3073#   3077    3085#   3089    3097#   3102    3110#   3114    3122#   3126    3128#   3132    3134#
                            3139    3147#   3152    3160#   3165    3167#   3172    3180#   3186    3194#   3201    3209#   3214
                            3216#   3221    3223#   3229    3237#   3242    3244#   3249    3251#   3257    3265#   3271    3279#
                            3284    3286#   3292    3294#   3300    3308#   3312    3314#   3319    3321#   3327    3335#   3339
                            3341#   3346    3348#   3354    3363#   3368    3371#   3378    3381#   3388    3397#   3401    3409#
                            3413    3415#   3419    3424#   3429    3431#   3436    3449#   3453    3456#   3460    3462#   3466
                            3468#   3473    3475#   3480    3482#   3487    3495#   3499    3501#   3506    3514#   3518    3520#
                            3526    3534#   3538    3540#   3546    3554#   3560    3562#   3567    3569#   3574    3582#   3586
                            3588#   3593    3601#   3605    3607#   3612    3614#   3619    3621#   3626    3628#   3634    3642#
                            3646    3648#   3654    3656#   3661    3669#   3674    3682#   3687    3689#   3694    3702#   3707
                            3709#   3713    3715#   3721    3729#   3733    3741#   3745    3753#   3757    3765#   3769    3777#
                            3781    3783#   3788    3790#   3796    3804#   3808    3810#   3816    3824#   3829    3831#   3836
                            3838#   3843    3845#   3850    3852#   3858    3866#   3870    3880#   3884    3888#   3892    3896#
                            3900    3904#   3908    3917#   3921    3923#   3927    3929#   3933    3941#   3945    4353#   4358
                            4468#   4476    4482#   4486    4503#   4508    4569#   4574    4584#   4589    4597#   4602    4612#
                            4617    4625#   4630    4714#   4713    4792#   4796   12919#  12924   12926#  12931   12933#  12938
                           12940#  12945   12947#  12952
T$CODE= 001021             13084#  13089#
T$ERRN= 000464              1389#   4944#   4978#   4995#   5047#   5081#   5098#   5115#   5132#   5149#   5193#   5243#   5304#
                            5333#   5355#   5386#   5410#   5435#   5454#   5502#   5542#   5558#   5611#   5633#   5658#   5675#
                            5712#   5729#   5751#   5764#   6185#   6205#   6251#   6271#   6291#   6382#   6402#   6425#   6445#
                            6486#   6503#   6520#   6612#   6632#   6659#   6679#   6707#   6717#   6786#   6794#   6814#   6835#
                            6842#   6862#   6869#   6935#   6961#   6977#   6992#   7063#   7073#   7109#   7137#   7149#   7176#
                            7244#   7254#   7276#   7288#   7309#   7316#   7390#   7400#   7423#   7444#   7453#   7476#   7483#
                            7495#   7555#   7565#   7593#   7608#   7617#   7634#   7647#   7714#   7724#   7747#   7761#   7770#
                            7789#   7803#   7864#   7872#   7894#   7908#   7917#   7934#   8002#   8012#   8033#   3047#   8054#
                            8063#   8076#   8140#   8150#   8171#   8185#   8192#   8201#   8214#   8288#   8298#   8321#   8340#
                            8349#   8367#   8377#   8450#   8460#   8482#   8501#   8510#   8528#   8538#   8603#   8613#   8644#
                            8658#   8667#   8685#   8695#   8780#   8790#   8811#   8829#   8838#   8859#   8872#   8881#   8893#
                            8902#   8911#   8921#   9007#   9017#   9047#   9061#   9070#   9093#   9107#   9176#   9186#   9216#
                            9230#   9239#   9263#   9277#   9349#   9359#   9380#   9395#   9404#   9417#   9427#   9514#   9524#
                            9557#   9571#   9580#   9600#   9614#   9703#   9713#   9743#   9757#   9766#   9781#   9795#   9866#
                            9876#   9906#   9920#   9929#   9943#   9957#  10105#  10115#  10157#  10171#  10180#  10197#  10213#
                           10227#  10328#  10338#  10367#  10390#  10398#  10409#  10483#  10493#  10513#  10529#  10538#  10552#
                           10562#  10643#  10653#  10679#  10694#  10703#  10716#  10732#  10810#  10820#  10839#  10861#  10870#
                           10884#  10894#  10982#  10992#  11018#  11037#  11046#  11059#  11069#  11150#  11160#  11188#  11207#
                           11216#  11232#  11240#  11250#  11339#  11349#  11391#  11405#  11414#  11428#  11438#  11547#  11557#
                           11590#  11604#  11613#  11630#  11645#  11656#  11670#  11762#  11772#  11801#  11815#  11824#  11838#
                           11853#  11864#  11878#  11998#  12008#  12042#  12056#  12065#  12087#  12098#  12119#  12133#  12139#
                           12159#  12170#  12191#  12205#  12214#  12238#  12248#  12341#  12351#  12375#  12389#  12398#  12412#
                           12422#  12442#  12456#  12465#  12480#  12493#  12588#  12598#  12627#  12641#  12650#  12663#  12694#
                           12755#  12768#  12789#  12802#  12833#  12846#  12878#  12890#
T$EXCP  000000             13084#  13088   13089#  13093
T$FLAG- 000040              4646#   4648    4800#   4820#   4822    4831#   4833    5308#   5662#   5716#   5733#   5755#   6189#
                            6255#   6386#   6429#   6490#   6616#   6636#   6663#   6683#   6818#   6965#   7051#   7067#   7077#
                            7113#   7233#   7248#   7258#   7280#   7292#   7379#   7394#   7404#   7427#   7448#   7457#   7544#
                            7559#   7569#   7597#   7612#   7621#   7651#   7703#   7718#   7728#   7751#   7765#   7774#   7807#
                            7898#   7912#   7991#   8006#   8016#   8037#   8058#   8067#   8129#   8141#   8154#   8175#   8196#
```

```
                                                                 K  7
                                                                                                                                  SEQ 294
PARAMETER CODING       MACY11 30(1046)  11-JAN-83  10:13  PAGE 299
CZUAAA.P11    11-JAN-83 09:29           CROSS REFERENCE TABLE -- USER SYMBOLS

                           8205#   8277#   8292#   8302#   8325#   8344#   8353#   8439#   8454#   8464#   8486#   8505#   8514#
                           8592#   8607#   8617#   8648#   8662#   8671#   8769#   8784#   8794#   8815#   8833#   8842#   8996#
                           9011#   9021#   9051#   9065#   9074#   9111#   9165#   9180#   9190#   9220#   9234#   9243#   9281#
                           9338#   9353#   9363#   9384#   9399#   9408#   9503#   9518#   9528#   9561#   9575#   9584#   9618#
                           9692#   9707#   9717#   9747#   9761#   9770#   9799#   9855#   9870#   9880#   9910#   9924#   9933#
                           9961#  10094#  10109#  10119#  10161#  10175#  10184#  10317#  10332#  10342#  10371#  10394#  10472#
                          10487#  10497#  10517#  10533#  10542#  10632#  10647#  10657#  10683#  10698#  10707#  10736#  10799#
                          10814#  10824#  10843#  10865#  10874#  10971#  10986#  10996#  11022#  11041#  11050#  11139#  11154#
                          11164#  11192#  11211#  11220#  11328#  11343#  11353#  11395#  11409#  11418#  11536#  11551#  11561#
                          11594#  11608#  11617#  11674#  11751#  11766#  11776#  11805#  11819#  11828#  11987#  12002#  12012#
                          12046#  12060#  12069#  12123#  12143#  12195#  12209#  12218#  12330#  12345#  12355#  12379#  12393#
                          12402#  12446#  12460#  12469#  12484#  12577#  12592#  12602#  12631#  12645#  12654#  12667#  12698#
                          12732#  12738#  12759#  12772#  12793#  12806#  12837#  12850#  12882#  12894#
T$GMAN= 000000             1389#
T$HILI= 000776            13084#  13087   13089#  13092
T$LAST= 000000             1389#
T$LOLI= 000000            13084#  13086   13089#  13091
T$LSYM= 010000             1389#   1564    2898    2913    2934    2946    2958    2974    2993    3009    3021    3042    3068
                           3080    3092    3105    3117    3142    3155    3175    3189    3204    3232    3260    3274    3303
                           3330    3357    3392    3404    3442    3490    3509    3529    3549    3577    3596    3637    3664
                           3677    3697    3724    3736    3748    3760    3772    3799    3819    3861    3873    3912    3936
                           3948    4650    4762    4774    4814    4824    4835    4856    4876    4904    4960    5004    5010
                           5063    5158    5164    5212    5265    5313    5463    5466    5513    5567    5617    5639    5684
                           5687    5770    6214    6297    6300    6454    6528    6539    6732    6878    6998    7189    7325
                           7504    7657    7816    7948    8085    8223    8386    8547    8706    8930    9122    9292    9436
                           9631    9814    9975   10236   10425   10571   10750   10903   11084   11265   11467   11687   11884
                          12257   12508   12708   12957   13096
T$NEST= 177777             1389#   1393#   1555#   1563#   2888#   2897#   2901#   2912#   2916#   2933#   2937#   2945#   2949#
                           2957#   2961#   2973#   2977#   2992#   2996#   3008#   3012#   3020#   3024#   3041#   3045#   3067#
                           3071#   3079#   3083#   3091#   3095#   3104#   3108#   3116#   3120#   3141#   3145#   3154#   3158#
                           3174#   3178#   3188#   3192#   3203#   3207#   3231#   3235#   3259#   3263#   3273#   3277#   3302#
                           3306#   3329#   3333#   3356#   3360#   3391#   3395#   3403#   3407#   3441#   3445#   3489#   3493#
                           3508#   3512#   3528#   3532#   3548#   3552#   3576#   3580#   3595#   3599#   3636#   3640#   3663#
                           3667#   3676#   3680#   3696#   3700#   3723#   3727#   3735#   3739#   3747#   3751#   3759#   3763#
                           3771#   3775#   3798#   3802#   3818#   3822#   3860#   3864#   3872#   3876#   3911#   3915#   3935#
                           3939#   3947#   4644#   4649#   4661#   4668#   4677#   4761#   4771#   4773#   4783#   4813#   4818#
                           4823#   4829#   4834#   4852#   4855#   4871#   4875#   4897#   4903#   4924#   4933#   4959#   4963#
                           4968#   4983#   4986#   5000#   5003#   5009#   5027#   5036#   5062#   5066#   5071#   5086#   5089#
                           5103#   5106#   5120#   5123#   5137#   5140#   5154#   5157#   5163#   5179#   5211#   5227#   5235#
                           5258#   5264#   5286#   5291#   5312#   5317#   5322#   5338#   5348#   5360#   5374#   5391#   5403#
                           5415#   5424#   5440#   5443#   5459#   5462#   5465#   5488#   5493#   5507#   5512#   5532#   5535#
                           5547#   5551#   5563#   5566#   5595#   5597#   5616#   5620#   5638#   5642#   5666#   5680#   5683#
                           5686#   5708#   5769#   6164#   6166#   6171#   6193#   6196#   6210#   6213#   6217#   6230#   6259#
                           6262#   6276#   6296#   6299#   6340#   6343#   6348#   6390#   6393#   6407#   6410#   6433#   6436#
                           6450#   6453#   6457#   6462#   6494#   6497#   6508#   6527#   6538#   6579#   6583#   6620#   6623#
                           6640#   6643#   6667#   6670#   6687#   6693#   6722#   6731#   6763#   6769#   6800#   6809#   6874#
                           6877#   6907#   6912#   6938#   6944#   6982#   6997#   7036#   7042#   7087#   7104#   7154#   7160#
                           7181#   7188#   7218#   7224#   7262#   7271#   7321#   7324#   7364#   7370#   7408#   7418#   7500#
                           7503#   7529#   7535#   7573#   7588#   7638#   7656#   7688#   7694#   7732#   7742#   7794#   7815#
                           7841#   7847#   7878#   7888#   7944#   7947#   7976#   7982#   8020#   8028#   8081#   8084#   8114#
                           8120#   8158#   8166#   8219#   8222#   8262#   8268#   8306#   8316#   8382#   8385#   8424#   8430#
                           8468#   8477#   8543#   8546#   8577#   8583#   8621#   8635#   8700#   8705#   8754#   8760#   8798#
                           8806#   8926#   8929#   8981#   8987#   9025#   9038#   9098#   9121#   9150#   9156#   9194#   9207#
                           9268#   9291#   9323#   9329#   9367#   9375#   9432#   9435#   9488#   9494#   9532#   9549#   9605#
                           9630#   9677#   9683#   9721#   9734#   9786#   9813#   9840#   9846#   9884#   9901#   9948#   99[illegible]#
                          10079#  10085#  10123#  10147#  10232#  10235#  10302#  10308#  10346#  10358#  10414#  10424#  10457#
```

```
                        10463#  10501#  10508#  10567#  10570#  10617#  10623#  10661#  10674#  10721#  10727#  10740#  10749#
                        10784#  10790#  10828#  10834#  10899#  10902#  10956#  10962#  11000#  11010#  11075#  11083#  11124#
                        11130#  11168#  11183#  11255#  11264#  11313#  11319#  11357#  11370#  11380#  11443#  11456#  11466#
                        11521#  11527#  11565#  11576#  11661#  11686#  11736#  11742#  11780#  11787#  11869#  11883#  11972#
                        11978#  12016#  12033#  12103#  12109#  12177#  12183#  12253#  12256#  12315#  12321#  12359#  12367#
                        12427#  12434#  12498#  12507#  12562#  12568#  12606#  12622#  12704#  12707#  12729#  12956#  13081#
                        13094#  13121#
T$NS0 = 000000           1393#  13121
T$NS1 = 000004           1555#   1563    2888#   2897    2901#   2912    2916#   2933    2937#   2945    2949#   2957    2961#
                         2973    2977#   2992    2996#   3008    3012#   3020    3024#   3041    3045#   3067    3071#   3079
                         3083#   3091    3095#   3104    3108#   3116    3120#   3141    3145#   3154    3158#   3174    3178#
                         3188    3192#   3203    3207#   3231    3235#   3259    3263#   3273    3277#   3302    3306#   3329
                         3333#   3356    3360#   3391    3395#   3403    3307#   3441    3445#   3489    3493#   3508    3512#
                         3528    3532#   3548    3552#   3576    3580#   3595    3599#   3636    3640#   3663    3667#   3676
                         3680#   3696    3700#   3723    3727#   3735    3739#   3747    3751#   3759    3763#   3771    3775#
                         3798    3802#   3818    3822#   3860    3864#   3872    3876#   3911    3915#   3935    3939#   3947
                         4644#   4649    4661#   4668    4677#   4761    4771#   4773    4783#   4813    4818#   4823    4829#
                         4834    4852#   4855    4871#   4875    4897#   4903    4924#   5009    5027#   5163    5179#   5211
                         5227#   5264    5286#   5465    5488#   5512    5532#   5566    5595#   5686    5708#   5769    6164#
                         6299    6340#   6538    6579#   6731    6763#   6877    6907#   6997    7036#   7188    7218#   7324
                         7364#   7503    7529#   7656    7688#   7815    7841#   7947    7976#   8084    8114#   8222    8262#
                         8385    8424#   8546    8577#   8705    8754#   8929    8981#   9121    9150#   9291    9323#   9435
                         9488#   9630    9677#   9813    9840#   9974   10079#  10235   10302#  10424   10457#  10570   10617#
                        10749   10784#  10902   10956#  11083   11124#  11264   11313#  11466   11521#  11686   11736#  11883
                        11972#  12256   12315#  12507   12562#  12707   12729#  12956   13081#  13094
T$NS2 = 000003           4933#   4959    4963#   5003    5036#   5062    5066#   5157    5235#   5258    5291#   5312    5317#
                         5462    5493#   5507    5535#   5547    5551#   5563    5597#   5616    5620#   5638    5642#   5683
                         6166#   6213    6217#   6296    6343#   6453    6457#   6527    6583#   6620    6623#   6640    6643#
                         6667    6670#   6687    6693#   6722    6769#   6800    6809#   6874    6912#   6938    6944#   6982
                         7042#   7087    7104#   7154    7160#   7181    7224#   7262    7271#   7321    7370#   7408    7418#
                         7500    7535#   7573    7588#   7638    7694#   7732    7742#   7794    7847#   7878    7888#   7944
                         7982#   8020    8028#   8081    8120#   8158    8166#   8219    8268#   8306    8316#   8382    8430#
                         8468    8477#   8543    8583#   8621    8635#   8700    8760#   8798    8806#   8926    8987#   9025
                         9038#   9098    9156#   9194    9207#   9268    9329#   9367    9375#   9432    9494#   9532    9549#
                         9605    9683#   9721    9734#   9786    9846#   9884    9901#   9948   10085#  10123   10147#  10232
                        10308#  10346   10358#  10414   10463#  10501   10508#  10567   10623#  10661   10674#  10721   10727#
                        10740   10790#  10828   10834#  10899   10962#  11000   11010#  11075   11130#  11168   11183#  11255
                        11319#  11357   11370#  11456   11527#  11565   11576#  11661   11742#  11780   11787#  11869   11978#
                        12016   12033#  12103   12109#  12177   12183#  12253   12321#  12359   12367#  12427   12434#  12498
                        12568#  12606   12622#  12704
T$NS3 - 000003           4968#   4983    4986#   5000    5071#   5086    5089#   5103    5106#   5120    5123#   5137    5140#
                         5154    5322#   5338    5348#   5360    5374#   5391    5403#   5415    5424#   5440    5443#   5459
                         5666#   5680    6171#   6193    6196#   6210    6230#   6259    6262#   6276    6348#   6390    6393#
                         6407    6410#   6433    6436#   6450    6462#   6494    6497#   6508   11380#  11443
T$PTNU= 000000           1389#
T$SAVL= 177777           1389#
T$SEGL= 177777           1389#   4968#   4983#   4985    4986#   5000#   5002    5071#   5086#   5088    5089#   5103#   5105
                         5106#   5120#   5122    5123#   5137#   5139    5140#   5154#   5156    5235#   5258#   5260    5322#
                         5338#   5340    5348#   5360#   5362    5374#   5391#   5393    5403#   5415#   5417    5424#   5440#
                         5442    5443#   5459#   5461    5493#   5507#   5509    5535#   5547#   5549    5551#   5563#   5565
                         5666#   5680#   5682    6171#   6193#   6195    6196#   6210#   6212    6230#   6259#   6261    6262#
                         6276#   6278    6348#   6390#   6392    6393#   6407#   6409    6410#   6433#   6435    6436#   6450#
                         6452    6462#   6494#   6496    6497#   6508#   6510    6583#   6620#   6622    6623#   6640#   6642
                         6643#   6667#   6669    6670#   6687#   6689    6693#   6722#   6724    6769#   6800#   6802    6809#
                         6874#   6876    6912#   6938#   6940    6944#   6982#   6984    7042#   7087#   7089    7104#   7154#
                         7156    7160#   7181#   7183    7224#   7262#   7264    7271#   7321#   7323    7370#   7408#   7410
```

```
                      7418#    7500#    7502     7535#    7573#    7575     7588#    7638#    7640     7694#    7732#    7734     7742#
                      7794#    7796     7847#    7878#    7880     7888#    7944#    7946     7982#    8020#    8022     8028#    8081#
                      8083     8120#    8158#    8160     8166#    8219#    8221     8268#    8306#    8308     8316#    8382#    8384
                      8430#    8468#    8470     8477#    8543#    8545     8583#    8621#    8623     8635#    8700#    8702     8760#
                      8798#    8800     8806#    8926#    8928     8987#    9025#    9027     9038#    9098#    9100     9156#    9194#
                      9196     9207#    9268#    9270     9329#    9367#    9369     9375#    9432#    9434     9494#    9532#    9534
                      9549#    9605#    9607     9683#    9721#    9723     9734#    9786#    9788     9846#    9884#    9886     9901#
                      9948#    9950    10085#   10123#   10125    10147#   10232#   10234    10308#   10346#   10348    10358#   10414#
                     10416    10463#   10501#   10503    10508#   10567#   10569    10623#   10661#   10663    10674#   10721#   10723
                     10727#   10740#   10742    10790#   10828#   10830    10834#   10899#   10901    10962#   11000#   11002    11010#
                     11075#   11077    11130#   11168#   11170    11183#   11255#   11257    11319#   11357#   11359    11370#   11380#
                     11443#   11445    11456#   11458    11527#   11565#   11567    11576#   11661#   11663    11742#   11780#   11782
                     11787#   11869#   11871    11978#   12016#   12018    12033#   12103#   12105    12109#   12177#   12179    12183#
                     12253#   12255    12321#   12359#   12361    12367#   12427#   12429    12434#   12498#   12500    12568#   12606#
                     12608    12622#   12704#   12706
T$SEK0= 010001        4968#    4983     4986#    5000     5071#    5086     5089#    5103     5106#    5120     5123#    5137     5140#
                      5154     5235#    5258     5322#    5338     5348#    5360     5374#    5391     5403#    5415     5424#    5440
                      5443#    5459     5493#    5507     5535#    5547     5551#    5563     5666#    5680     6171#    6193     6196#
                      6210     6230#    6259     6262#    6276     6348#    6390     6393#    6407     6410#    6433     6436#    6450
                      6462#    6494     6497#    6508     6583#    6620     6623#    6640     6643#    6667     6670#    6687     6693#
                      6722     6769#    6800     6809#    6874     6912#    6938     6944#    6982     7042#    7087     7104#    7154
                      7160#    7181     7224#    7262     7271#    7321     7370#    7408     7418#    7500     7535#    7573     7588#
                      7638     7694#    7732     7742#    7794     7847#    7878     7888#    7944     7982#    8020     8028#    8081
                      8120#    8158     8166#    8219     8268#    8306     8316#    8382     8430#    8468     8477#    8543     8583#
                      8621     8635#    8700     8760#    8798     8806#    8926     8987#    9025     9038#    9098     9156#    9194
                      9207#    9268     9329#    9367     9375#    9432     9494#    9532     9549#    9605     9683#    9721     9734#
                      9786     9846#    9884     9901#    9948    10085#   10123    10147#   10232    10308#   10346    10358#   10414
                     10463#   10501    10508#   10567    10623#   10661    10674#   10721    10727#   10740    10790#   10828    10834#
                     10899    10962#   11000    11010#   11075    11130#   11168    11183#   11255    11319#   11357    11370#   11456
                     11527#   11565    11576#   11661    11742#   11780    11787#   11869    11978#   12016    12033#   12103    12109#
                     12177    12183#   12253    12321#   12359    12367#   12427    12434#   12498    12568#   12606    12622#   12704
T$SEK1= 010002       11380#   11443
T$SUBN= 000000        1389#    4923#    4932#    4962#    5026#    5035#    5065#    5178#    5226#    5285#    5290#    5316#    5487#
                      5531#    5594#    5596#    5619#    5641#    5707#    6163#    6165#    6216#    6339#    6342#    6456#    6578#
                      6762#    6906#    7035#    7217#    7363#    7528#    7687#    7840#    7975#    8113#    8261#    8423#    8576#
                      8753#    8980#    9149#    9322#    9487#    9676#    9839#   10078#   10301#   10456#   10616#   10783#   10955#
                     11123#   11312#   11520#   11735#   11971#   12314#   12561#   12728#
T$TAGL= 177777        1389#
T$TAGN= 010172        1389#    1555#    2888#    2901#    2916#    2937#    2949#    2961#    2977#    2996#    3012#    3024#    3045#
                      3071#    3083#    3095#    3108#    3120#    3145#    3158#    3178#    3192#    3207#    3235#    3263#    3277#
                      3306#    3333#    3360#    3395#    3407#    3445#    3493#    3512#    3532#    3552#    3580#    3599#    3640#
                      3667#    3680#    3700#    3727#    3739#    3751#    3763#    3775#    3802#    3822#    3864#    3876#    3915#
                      3939#    4644#    4661#    4677#    4771#    4783#    4818#    4829#    4852#    4871#    4897#    4924#    4933#
                      4963#    5027#    5036#    5066#    5179#    5227#    5286#    5291#    5317#    5488#    5532#    5595#    5597#
                      5620#    5642#    5708#    6164#    6166#    6217#    6340#    6343#    6457#    6579#    6763#    6907#    7036#
                      7218#    7364#    7529#    7688#    7841#    7976#    8114#    8262#    8424#    8577#    8754#    8981#    9150#
                      9323#    9488#    9677#    9840#   10079#   10302#   10457#   10617#   10784#   10956#   11124#   11313#   11521#
                     11736#   11972#   12315#   12562#   12729#   13081#
T$TEMP= 000000        1498#    1499#    1500#    1501#    1502#    1503#    1504#    1505#    1506#    1507#    1508#    1509#    1510#
                      1511#    1512#    1513#    1514#    1515#    1516#    1517#    1518#    1519#    1520#    1521#    1522#    1523#
                      1524#    1525#    1526#    1527#    1528#    1529#    1530#    1531#    1532#    1533#    1534#    1535#    1536#
                      1537#    1538#    1539#    1540#    1541#    1542#    1543#    1544#    1563#    2897#    2912#    2933#    2945#
                      2957#    2973#    2992#    3008#    3020#    3041#    3067#    3079#    3091#    3104#    3116#    3141#    3154#
                      3174#    3188#    3203#    3231#    3259#    3273#    3302#    3329#    3356#    3391#    3403#    3441#    3489#
                      3508#    3528#    3548#    3576#    3595#    3636#    3663#    3676#    3696#    3723#    3735#    3747#    3759#
                      3771#    3798#    3818#    3860#    3872#    3911#    3935#    3947#    4646#    4647     4649#    4668#    4761#
                      4773#    4800#    4801     4813#    4820#    4821     4823#    4831#    4832     4834#    4855#    4875#    4903#
                      4959#    4983#    5000#    5003#    5009#    5062#    5086#    5103#    5120#    5137#    5154#    5157#    5163#
                      5211#    5258#    5264#    5308#    5309     5312#    5338#    5360#    5391#    5415#    5440#    5459#    5462#
                      5465#    5507#    5512#    5547#    5563#    5566#    5616#    5638#    5662#    5663     5680#    5683#    5686#
                      5716#    5717     5733#    5734     5755#    5756     5769#    6189#    6190     6193#    6210#    6213#    6255#
                      6256     6259#    6276#    6296#    6299#    6386#    6387     6390#    6407#    6429#    6430     6433#    6450#
                      6453#    6490#    6491     6494#    6508#    6527#    6538#    6616#    6617     6620#    6636#    6637     6640#
                      6663#    6664     6667#    6683#    6684     6687#    6722#    6731#    6800#    6818#    6819     6874#    6877#
                      6938#    6965#    6966     6982#    6997#    7051#    7052     7067#    7068     7077#    7078     7087#    7113#
                      7114     7154#    7181#    7188#    7233#    7234     7248#    7249     7258#    7259     7262#    7280#    7281
                      7292#    7293     7321#    7324#    7379#    7380     7394#    7395     7404#    7405     7408#    7427#    7428
                      7448#    7449     7457#    7458     7500#    7503#    7544#    7545     7559#    7560     7569#    7570     7573#
                      7597#    7598     7612#    7613     7621#    7622     7638#    7651#    7652     7656#    7703#    7704     7718#
                      7719     7728#    7729     7732#    7751#    7752     7765#    7766     7774#    7775     7794#    7807#    7808
                      7815#    7878#    7898#    7899     7912#    7913     7944#    7947#    7991#    7992     8006#    8007     8016#
                      8017     8020#    8037#    8038     8058#    8059     8067#    8068     8081#    8084#    8129#    8130     8144#
                      8145     8154#    8155     8158#    8175#    8176     8196#    8197     8205#    8206     8219#    8222#    8277#
                      8278     8292#    8293     8302#    8303     8306#    8325#    8326     8344#    8345     8353#    8354     8382#
                      8385#    8439#    8440     8454#    8455     8464#    8465     8468#    8486#    8487     8505#    8506     8514#
                      8515     8543#    8546#    8592#    8593     8607#    8608     8617#    8618     8621#    8648#    8649     8662#
                      8663     8671#    8672     8700#    8705#    8769#    8770     8784#    8785     8794#    8795     8798#    8815#
                      8816     8833#    8834     8842#    8843     8926#    8929#    8996#    8997     9011#    9012     9021#    9022
                      9025#    9051#    9052     9065#    9066     9074#    9075     9098#    9111#    9112     9121#    9165#    9166
                      9180#    9181     9190#    9191     9194#    9220#    9221     9234#    9235     9243#    9244     9268#    9281#
                      9282     9291#    9338#    9339     9353#    9354     9363#    9364     9367#    9384#    9385     9399#    9400
                      9408#    9409     9432#    9435#    9503#    9504     9518#    9519     9528#    9529     9532#    9561#    9562
                      9575#    9576     9584#    9585     9605#    9618#    9619     9630#    9692#    9693     9707#    9708     9717#
                      9718     9721#    9747#    9748     9761#    9762     9770#    9771     9786#    9799#    9800     9813#    9855#
                      9856     9870#    9871     9880#    9881     9884#    9910#    9911     9924#    9925     9933#    9934     9948#
                      9961#    9962     9974#   10094#   10095    10109#   10110    10119#   10120    10123#   10161#   10162    10175#
                     10176    10184#   10185    10232#   10235#   10317#   10318    10332#   10333    10342#   10343    10346#   10371#
                     10372    10394#   10395    10414#   10424#   10472#   10473    10487#   10488    10497#   10498    10501#   10517#
                     10518    10533#   10534    10542#   10543    10567#   10570#   10632#   10633    10647#   10648    10657#   10658
                     10661#   10683#   10684    10698#   10699    10707#   10708    10721#   10736#   10737    10740#   10749#   10799#
                     10800    10814#   10815    10824#   10825    10828#   10843#   10844    10865#   10866    10874#   10875    10899#
                     10902#   10971#   10972    10986#   10987    10996#   10997    11000#   11022#   11023    11041#   11042    11050#
                     11051    11075#   11083#   11139#   11140    11154#   11155    11164#   11165    11168#   11192#   11193    11211#
                     11212    11220#   11221    11255#   11264#   11328#   11329    11343#   11344    11353#   11354    11357#   11395#
                     11396    11409#   11410    11418#   11419    11443#   11456#   11466#   11536#   11537    11551#   11552    11561#
                     11562    11565#   11594#   11595    11608#   11609    11617#   11618    11661#   11674#   11675    11686#   11751#
                     11752    11766#   11767    11776#   11777    11780#   11805#   11806    11819#   11820    11828#   11829    11869#
                     11883#   11987#   11988    12002#   12003    12012#   12013    12016#   12046#   12047    12060#   12061    12069#
                     12070    12103#   12123#   12124    12143#   12144    12177#   12195#   12196    12209#   12210    12218#   12219
                     12253#   12256#   12330#   12331    12345#   12346    12355#   12356    12359#   12379#   12380    12393#   12394
                     12402#   12403    12427#   12446#   12447    12460#   12461    12469#   12470    12484#   12485    12498#   12507#
                     12577#   12578    12592#   12593    12602#   12603    12606#   12631#   12632    12645#   12646    12654#   12655
                     12667#   12668    12698#   12699    12704#   12707#   12732#   12733    12738#   12739    12759#   12760    12772#
                     12773    12793#   12794    12806#   12807    12837#   12838    12850#   12851    12882#   12883    12894#   12895
                     12956#   13084#   13089#   13094#   13121#
T$TEST= 000056        1389#    4923#    4932     4962     5026#    5035     5065     5178#    5226#    5285#    5290     5316     5487#
                      5531#    5594#    5596     5619     5641     5707#    6163#    6165     6216     6339#    6342     6456     6578#
                      6762#    6906#    7035#    7217#    7363#    7528#    7687#    7840#    7975#    8113#    8261#    8423#    8576#
                      8753#    8980#    9149#    9322#    9487#    9676#    9839#   10078#   10301#   10456#   10616#   10783#   10955#
                     11123#   11312#   11520#   11735#   11971#   12314#   12561#   12728#
T$TSTM= 177777        1389#    2894     2898     2909     2913     2922     2930     2934     2942     2946     2954     2958     2970
                      2974     2982     2989     2993     3005     3009     3017     3021     3029     3038     3042     3050     3057
```

```
    3064     3068     3076     3080     3088     3092     3101     3105     3113     3117     3125     3131     3138
    3142     3151     3155     3164     3171     3175     3185     3189     3200     3204     3213     3220     3228
    3232     3241     3248     3256     3260     3270     3274     3283     3291     3299     3303     3311     3318
    3326     3330     3338     3345     3353     3357     3367     3377     3387     3392     3400     3404     3412
    3418     3428     3435     3442     3452     3459     3465     3472     3479     34       3490     3498     3505
    3509     3517     3525     3529     3537     3545     3549     3559     3566     35       3577     3585     3592
    3596     3604     3611     3618     3625     3633     3637     3645     3653     3660     3664     3673     3677
    3686     3693     3697     3706     3712     3720     3724     3732     3736     3744     3748     3756     3760
    3768     3772     3780     3787     3795     3799     3807     3815     3819     3828     3835     3842     3849
    3857     3861     3869     3873     3883     3891     3899     3907     3912     3920     3926     3932     3936
    3944     3948     4201     4215     4244     4357     4391     4475     4485     4507     4573     4588     4601
    4616     4629     4650     4683     4688     4693     4698     4703     4709     4717     4731     4741     4758
    4762     4774     4795     4800     4814     4824     4835     4929     4933     4943     4948     4951     4954
    4956     4960     4963     4968     4977     4984     4986     4994     5001     5004     5007     5010     5032
    5036     5046     5051     5054     5057     5059     5063     5066     5071     5080     5087     5089     5097
    5104     5106     5114     5121     5123     5131     5138     5140     5148     5155     5158     5161     5164
    5184     5192     5197     5200     5203     5205     5208     5212     5232     5235     5242     5247     5250
    5253     5255     5259     5262     5265     5291     5303     5308     5313     5317     5322     5332     5339
    5348     5354     5361     5374     5385     5392     5403     5409     5416     5424     5434     5441     5443
    5453     5460     5463     5466     5493     5501     5508     5513     5535     5541     5548     5551     5557
    5564     5567     5597     5610     5617     5620     5632     5639     5642     5657     5662     5666     5674
    5681     5684     5687     5711     5716     5728     5733     5750     5755     5763     5770     6166     6171
    6184     6189     6194     6196     6204     6211     6214     6217     6230     6250     6255     6260     6262
    6270     6277     6290     6297     6300     6343     6348     6381     6386     6391     6393     6401     6408
    6410     6424     6429     6434     6436     6444     6451     6454     6457     6462     6485     6490     6495
    6497     6502     6509     6519     6528     6539     6583     6611     6616     6621     6623     6631     6636
    6641     6643     6658     6663     6668     6670     6678     6683     6688     6693     6706     6716     6723
    6732     6769     6785     6793     6801     6809     6813     6818     6834     6841     6861     6868     6875
    6878     6912     6918     6932     6939     6944     6951     6960     6965     6976     6983     6986     6991
    6998     7042     7051     7062     7067     7072     7077     7084     7088     7104     7108     7113     7124
    7136     7148     7155     7160     7175     7182     7189     7224     7233     7243     7248     7253     7258
    7263     7271     7275     7280     7287     7292     7308     7315     7322     7325     7370     7379     7389
    7394     7399     7404     7409     7418     7422     7427     7443     7448     7452     7457     7475     7482
    7494     7501     7504     7535     7544     7554     7559     7564     7569     7574     7588     7592     7597
    7607     7612     7616     7621     7633     7639     7646     7651     7657     7694     7703     7713     7718
    7723     7728     7733     7742     7746     7751     7760     7765     7769     7774     7788     7795     7802
    7807     7816     7847     7863     7871     7879     7888     7893     7898     7907     7912     7916     7933
    7945     7948     7982     7991     8001     8006     8011     8016     8021     8028     8032     8037     8046
    8053     8058     8062     8067     8075     8082     8085     8120     8129     8139     8144     8149     8154
    8159     8166     8170     8175     8184     8191     8196     8200     8205     8213     8220     8223     8268
    8277     8287     8292     8297     8302     8307     8316     8320     8325     8339     8344     8348     8353
    8366     8376     8383     8386     8430     8439     8449     8454     8459     8464     8469     8477     8481
    8486     8500     8505     8509     8514     8527     8537     8544     8547     8583     8592     8602     8607
    8612     8617     8622     8635     8643     8648     8657     8662     8666     8671     8684     8694     8701
    8706     8760     8769     8779     8784     8789     8794     8799     8806     8810     8815     8828     8833
    8837     8842     8858     8871     8880     8892     8901     8910     8920     8927     8930     8987     8996
    9006     9011     9016     9021     9026     9038     9046     9051     9060     9065     9069     9074     9092
    9099     9106     9111     9122     9156     9165     9175     9180     9185     9190     9195     9207     9215
    9220     9229     9234     9238     9243     9262     9269     9276     9281     9292     9329     9338     9348
    9353     9358     9363     9368     9375     9379     9384     9394     9399     9403     9408     9416     9426
    9433     9436     9494     9503     9513     9518     9523     9528     9533     9549     9556     9561     9570
    9575     9579     9584     9599     9606     9613     9618     9631     9683     9692     9702     9707     9712
    9717     9722     9734     9742     9747     9756     9761     9765     9770     9780     9787     9794     9799
    9814     9846     9855     9865     9870     9875     9880     9885     9901     9905     9910     9919     9924
    9928     9933     9942     9949     9956     9961     9975    10085    10094    10104    10109    10114    10119
   10124    10147    10156    10161    10170    10175    10179    10184    10196    10212    10226    10233    10236
```

```
                        10308   10317   10327   10332   10337   10342   10347   10358   10366   10371   10389   10394   10397
                        10408   10415   10425   10463   10472   10482   10487   10492   10497   10502   10508   10512   10517
                        10528   10533   10537   10542   10551   10561   10568   10571   10623   10632   10642   10647   10652
                        10657   10662   10674   10678   10683   10693   10698   10702   10707   10715   10722   10727   10731
                        10736   10741   10750   10790   10799   10809   10814   10819   10824   10829   10834   10838   10843
                        10860   10865   10869   10874   10883   10893   10900   10903   10962   10971   10981   10986   10991
                        10996   11001   11010   11017   11022   11036   11041   11045   11050   11058   11068   11076   11084
                        11130   11139   11149   11154   11159   11164   11169   11183   11187   11192   11206   11211   11215
                        11220   11231   11239   11249   11256   11265   11319   11328   11338   11343   11348   11353   11358
                        11370   11380   11390   11395   11404   11409   11413   11418   11427   11437   11444   11457   11467
                        11527   11536   11546   11551   11556   11561   11566   11576   11589   11594   11603   11608   11612
                        11617   11629   11644   11655   11662   11669   11674   11687   11742   11751   11761   11766   11771
                        11776   11781   11787   11800   11805   11814   11819   11823   11828   11837   11852   11863   11870
                        11877   11884   11978   11987   11997   12002   12007   12012   12017   12033   12041   12046   12055
                        12060   12064   12069   12086   12097   12104   12109   12118   12123   12132   12138   12143   12158
                        12169   12178   12183   12190   12195   12204   12209   12213   12218   12237   12247   12254   12257
                        12321   12330   12340   12345   12350   12355   12360   12367   12374   12379   12388   12393   12397
                        12402   12411   12421   12428   12434   12441   12446   12455   12460   12464   12469   12479   12484
                        12492   12499   12508   12568   12577   12587   12592   12597   12602   12607   12622   12626   12631
                        12640   12645   12649   12654   12662   12667   12693   12698   12705   12708   12732   12738   12754
                        12759   12767   12772   12788   12793   12801   12806   12832   12837   12845   12850   12877   12882
                        12889   12894   12923   12930   12937   12944   12951   12957
T$TSTS= 000001           1389#   4924#   5027#   5179#   5227#   5286#   5488#   5532#   5595#   5708#   6164#   6340#   6579#
                         6763#   6907#   7036#   7218#   7364#   7529#   7688#   7841#   7976#   8114#   9262#   8424#   8577#
                         8754#   8981#   9150#   9323#   9488#   9677#   9840#  10079#  10302#  10457#  10617#  10784#  10956#
                        11124#  11313#  11521#  11736#  11972#  12315#  12562#  12729#
T$$AU  = 010072          4829#   4831    4834
T$$AUT= 010067           4771#   4773
T$$CLE= 010070           4783#   4800    4813
T$$DU  = 010071          4818#   4820    4823
T$$HAR= 010171          13081#  13095
T$$HW  = 010000          1555#   1563
T$$INI= 010066           4677#   4761
T$$MSG= 010063           2888#   2897    2901#   2912    2916#   2933    2937#   2945    2949#   2957    2961#   2973    2977#
                         2992    2996#   3008    3012#   3020    3024#   3041    3045#   3067    3071#   3079    3083#   3091
                         3095#   3104    3108#   3116    3120#   3141    3145#   3154    3158#   3174    3178#   3188    3192#
                         3203    3207#   3231    3235#   3259    3263#   3273    3277#   3302    3306#   3329    3333#   3356
                         3360#   3391    3395#   3403    3407#   344     3445#   3489    3493#   3508    3512#   3528    3532#
                         3548    3552#   3576    3580#   3595    3599#   3636    3640#   3663    3667#   3676    3680#   3696
                         3700#   3723    3727#   3735    3739#   3747    3751#   3759    3763#   3771    3775#   3798    3802#
                         3818    3822#   3860    3864#   3872    3876#   3911    3915#   3935    3939#   3947
T$$PRO= 010065           4661#
T$$RPT= 010064           4644#   4646    4649
T$$SEG= 010001           4968#   4983#   4986#   5000#   5071#   5086#   5089#   5103#   5106#   5120#   5123#   5137#   5140#
                         5154#   5235#   5258#   5322#   5338#   5348#   5360#   5374#   5391#   5403#   5415#   5424#   5440#
                         5443#   5459#   5493#   5507#   5535#   5547#   5551#   5563#   5666#   5680#   6171#   6193#   6196#
                         6210#   6230#   6259#   6262#   6276#   6348#   6390#   6393#   6407#   6410#   6433#   6436#   6450#
                         6462#   6494#   6497#   6508#   6583#   6620#   6623#   6640#   6643#   6667#   6670#   6687#   6693#
                         6722#   6769#   6800#   6809#   6874#   6912#   6938#   6944#   6982#   7042#   7087#   7104#   7154#
                         7160#   7181#   7224#   7262#   7271#   7321#   7370#   7408#   7418#   7500#   7535#   7573#   7588#
                         7638#   7694#   7732#   7742#   7794#   7847#   7878#   7888#   7944#   7982#   8020#   8028#   8081#
                         8120#   8158#   8166#   8219#   8268#   8306#   8316#   8382#   8430#   8468#   8477#   8543#   8583#
                         8621#   8635#   8700#   8760#   8798#   8806#   8926#   8987#   9025#   9038#   9098#   9156#   9194#
                         9207#   9268#   9329#   9367#   9375#   9432#   9494#   9532#   9549#   9605#   9683#   9721#   9734#
                         9786#   9846#   9884#   9901#   9948#  10085#  10123#  10147#  10232#  10308#  10346#  10358#  10414#
                        10463#  10501#  10508#  10567#  10623#  10661#  10674#  10721#  10727#  10740#  10790#  10828#  10834#
```

```
                         10899#  10962#  11000#  11010#  11075#  11:30#  11168#  11183#  11255#  11319#  11357#  11370#  11380#
                         11443#  11456#  11527#  11565#  11576#  11661#  11742#  11780#  11787#  11869#  11978#  12016#  12033#
                         12103#  12109#  12177#  12183#  12253#  12321#  12359#  12367#  12427#  12434#  12498#  12568#  12606#
                         12622#  12704#
T$$SRV= 010075            4852#   4855    4871#   4875    4897#   4903
T$$SUB= 010125            4933#   4959    4963#   5003    5036#   5062    5066#   5157    5291#   5312    5317#   5462    5597#
                          5616    5620#   5638    5642#   5683    6166#   6213    6217#   6296    6343#   6453    6457#   6527
T$$TES= U10170            4924#   5009    5027#   5163    5179#   5211    5227#   5264    5286#   5308    5465    5488#   5512
                          5532#   5566    5595#   5662    5686    5708#   5716    5733    5755    5769    6164#   6189    6255
                          6299    6340#   6386    6429    6490    6538    6579#   6616    6636    6663    6683    6731    6763#
                          6818    6877    6907#   6965    6997    7036#   7051    7067    7077    7113    7188    7218#   7233
                          7248    7258    7280    7292    7324    7364#   7379    7394    7404    7427    7448    7457    7503
                          7529#   7544    7559    7569    7597    7612    7621    7651    7656    7688#   7703    7718    7728
                          7751    7765    7774    7807    7815    7841#   7898    7912    7947    7976#   7991    8006    8016
                          8037    8058    8067    8084    8114#   8129    8144    8154    8175    8196    8205    8222    8262#
                          8277    8292    8302    8325    8344    8353    8385    8424#   8439    8454    8464    8486    8505
                          8514    8546    8577#   8592    8607    8617    8648    8662    8671    8705    8754#   8769    8784
                          8794    8815    8833    8842    8929    8981#   8996    9011    9021    9051    9065    9074    9111
                          9121    9150#   9165    9180    9190    9220    9234    9243    9281    9291    9323#   9338    9353
                          9363    9384    9399    9408    9435    9488#   9503    9518    9528    9561    9575    9584    9618
                          9630    9677#   9692    9707    9717    9747    9761    9770    9799    9813    9840#   9855    9870
                          9880    9910    9924    9933    9961    9974   10079#  10094   10109   10119   10161   10175   10184
                         10235   10302#  10317   10332   10342   10371   10394   10424   10457#  10472   '0487   10497   10517
                         10533   10542   10570   10617#  10632   10647   10657   10683   10698   10707   10736   10749   10784#
                         10799   10814   10824   10843   10865   10874   10902   10956#  10971   10986   10996   11022   11041
                         11050   11083   11124#  11139   11154   11164   11192   11211   11220   11264   11313#  11328   11343
                         11353   11395   11409   11418   11466   11521#  11536   11551   11561   11594   11608   11617   11674
                         11686   11736#  11751   11766   11776   11805   11819   11828   11883   11972#  11987   12002   12012
                         12046   12060   12069   12123   12143   12195   12209   12218   12256   12315#  12330   12345   12355
                         12379   12393   12402   12446   12460   12469   12484   12507   12562#  12577   12592   12602   12631      M
                         12645   12654   12667   12698   12707   12729#  12732   12738   12759   12772   12793   12806   12837
                         12850   12882   12894   12956
T1       021710RG   002   1498    4923#
T1.1     021736R    002   4932#
T1.2     022014R    002   4962#
T10      027332RG   002   1507    6163#
T10.1    027332R    002   6165#
T10.2    027464R    002   6216#                                                                                              M
T11      027746RG   002   1508    6339#                                                                                        M
T11.1    027752R    002   6342#   6535
T11.2    030304R    002   6456#
T12      030534RG   002   1509    6578#
T13      031214RG   002   1510    6762#
T14      031632RG   002   1511    6906#
T15      032122RG   002   1512    7035#
T16      032630RG   002   1513    7217#
T17      033146RG   002   1514    7363#
T18      033614RG   002   1515    7528#                                                                                        M
T19      034172RG   002   1516    7687#
T2       022122RG   002   1499    5026#
T2.1     022150R    002   5035#
T2.2     022230R    002   5065#                                                                                                M
T20      034552RG   002   1517    7840#                                                                                        M
T21      035100RG   002   1518    7975#                                                                                        M
T22      035410RG   002   1519    8113#
T23      035720RG   002   1520    8261#
```

```
T24     036300RG   002    1521      8423#
T25     036660RG   002    1522      8576#
T26     037254RG   002    1523      8753#
T27     040016RG   002    1524      8980#
T28     040414RG   002    1525      9149#
T29     041016RG   002    1526      9322#
T3      022454RG   002    1500      5178#
T30     041336RG   002    1527      9487#
T31     041726RG   002    1528      9676#
T32     042312RG   002    1529      9839#
T33     042676RG   002    1530     10078#
T34     043336RG   002    1531     10301#
T35     043666RG   002    1532     10456#
T36     044206RG   002    1533     10616#
T37     044556RG   002    1534     10783#
T38     045076RG   002    1535     10955#
T39     045436RG   002    1536     11123#
T4      022566RG   002    1501      5226#
T40     046026RG   002    1537     11312#
T41     046410RG   002    1538     11520#
T42     047042RG   002    1539     11735#
T43     047446RG   002    1540     11971#
T44     050536RG   002    1541     12314#
T45     051260RG   002    1542     12561#
T46     051674RG   002    1543     12728#
T5      022704RG   002    1502      5285#
T5.1    022716R    002    5290#
T5.2    022774R    002    5316#
T6      023366RG   002    1503      5487#
T7      023450PG   002    1504      5531#
T8      023546RG   002    1505      5594#
T8.1    023546R    002    5596#
T8.2    023612R    002    5619#
T8.3    023652R    002    5641#
T9      023770RG   002    1506      5707#
UAM   = 000200 G          1635#
UDBB    000614RG   002    1898#     4514*    4515*    4521*    4522*    4528*    4529*    4535*    4536*    4542*    4543*    4549*    4550*
                          4553*     4554*    4556*    4557*    4561     6223     6225*    6237*    6238*    6239*    6284*    6360*    6361*
                          6362*     6363*    6366     6473*    6474*    6589*    6590*    6591*    6592*    6596    12861    12863*   12864*
                         12865*    12866*
UNACSR  000266RG   002    1774#     4745*    4748
UNAINT  000670RG   002    1907#     4872*    6920*    6956     7115*    7125     7142
UNAPRI  000272RG   002    1776#     4747*    6914     6968     7080
UNASRV  021656RG   002    4871#     6915     7081
UNAVEC  000270RG   002    1775#     4746*    6916     6985     7082
UNIERR= 020000 G          1695#     3893     4425     4427
UNIHLT= 000005 G          1728#
UNIT    000330RG   002    1796#     4735*    4736*    4737     4740     4953     5056     5202     5252
UNLOD   001515RG   002    2041#     4504     4570     4585     4598     4613     4626
WCSADR= 000000 G          1656#     1657     4557     4635     6363     6592     6695
WCSMEM  002165RG   002    2096#     6787     6795     6815     6836     6843     6863     6870
WCSSIZ= 020000 G          1652#     1657     4557     4635     6354     6360     6363     6468     6514     6592     6695
WLSA  = 000025 G          1762#
WMAL  = 000007 G          1748#
WPA   = 000005 G          1746#
WRF     000011 G          1750#
```

```
PARAMETER CODING        MACY11 30(1046)  11-JAN-83  10:13  PAGE 307
CZUAAA.P11    11-JAN-83 09:29            CROSS REFERENCE TABLE -- USER SYMBOLS

WSIDP = 000023 G           1760#
WTM   = 000015 G           1754#
XPWR  = 100000 G           1705#    5447
X$ALWA= 000000             1389#
X$FALS= 000040             1389#
X$OFFS= 000400             1389#
X$TRUE= 000020             1389#
$ADRER  001457RG    002    2030#    6854     7470
$AFTER  001340RG    002    2012#    4465     5287     5399     5654     5725     6181     6247     6378     6421     6482     6608     6655
                           6703     6782     6929     7059     7240     7386     7551     7710     7860     7998     8136     8284     8446
                           8599     8776     9003     9172     9345     9510     9699     9862     10101    10324    10479    10639    10806
                           10978    11146    11335    11543    11758    11994    12337    12584    12751    12785    12829    12874
$BEFOR  001331RG    002    2010#
$BIT0   001267RG    002    1811     1846     2001#
$BIT1   001261RG    002    1812     1847     1999#
$BIT10  001172RG    002    1856     1981#
$BIT11  001163RG    002    1857     1979A
$BIT12  001154RG    002    1858     1977#
$BIT13  001145RG    002    1859     1975#    8869     8890
$BIT14  001136RG    002    1860     1973#    8899
$BIT15  001127RG    002    1861     1971#    8878     8908
$BIT2   001253RG    002    1813     1848     1997#
$BIT3   001245RG    002    1814     1849     1995#
$BIT4   001237RG    002    1815     1832     1850     1993#
$BIT5   001231RG    002    1833     1851     1991#
$BIT6   001223RG    002    1834     1852     1989#
$BIT7   001215RG    002    1853     1987#
$BIT8   001207RG    002    1854     1985#
$BIT9   001201RG    002    1820     1855     1983#
$CLR    001323RG    002    2008#
$DATER  001444RG    002    2028#    6850     7466
$DNI    000776RG    002    1822     1951#    4337     4463     5300     5652     5723     6179     6245     6376     6419     6480     6606
                           6653     6701     6780     6927     7057     7238     7384     7549     7708     7853     7996     8134     8282
                           8444     8597     8774     9001     9170     9343     9508     9697     9860     10099    10322    10477    10637
                           10804    10976    11144    11333    11541    11756    11992    12335    12582    12749    12783    12827    12872
$FATI   001017RG    002    1819     1955#    4351
$GTCMD  001355RG    002    2015#    6248     6422     6483     6656     6704     6783     7060     7241     7387     7552     7711     7861
                           7999     8137     8285     8447     8600     8777     9004     9173     9346     9511     9700     9863     10102
                           10325    10480    10640    10807    10979    11147    11336    11544    11759    11995    12338    12585    12786
                           12830    12875
$GTPCB  001346RG    002    2013#    6182     6379     6609     12752
$ICAB   001074RG    002    1842     1965#    5432
$INTE   001041RG    002    1817     1959#
$INTR   001030RG    002    1818     1957#    5329
$NCLR   001311RG    002    2006#    4972     5075     5345     5398     5431     5450
$NOP    001403RG    002    2020#    5655     6930
$NSET   001275RG    002    2003#    4464     5301     5330     5653     5724     6180     6246     6377     6420     6481     6607     6654
                           6702     6781     6928     7058     7099     7239     7385     7550     7709     7859     7997     8135     8283
                           8445     8598     8775     9002     9171     9344     9509     9698     9861     10100    10323    10478    10638
                           10805    10977    11145    11334    11542    11757    11993    12336    12583    12750    12784    12828    12873
$PARER  001500RG    002    2033#    6856
$PATCH  053574RG    002    13109#
$PCEI   000745RG    002    1825     1945#    4316
$PCTO   001105RG    002    1835     1967#
$PDNDM  001415RG    002    2023#
$RCEI   001006RG    002    1821     1953#    4344
```

```
$RESET   001436RG    002   2027#
$RMTC    001116RG    002   1831     1969#
$RSET    001052RG    002   1816     1961#
$RXI     000756RG    002   1824     1947#    4323
$SERI    000734RG    002   1826     1943#    4309
$SET     001305RG    002   2005#
$SLFT    001371RG    002   2018#    5726
$STOP    001431RG    002   2026#
$STRT    001407RG    002   2021#
$TXI     000766RG    002   1823     1949#    4330
$XPWR    001063RG    002   1843     1963#    5451
.      = 000000R     012   1903#    4647     4801     4810#    4821     4832     5309     5665     5717     5734     5756     6121#    6190
                           6256     6387     6430     6491     6617     6637     6664     6684     6819     6966     7052     7068     7078
                           7114     7234     7249     7259     7281     7293     7380     7395     7405     7428     7449     7458     7545
                           7560     7570     7598     7613     7622     7652     7704     7719     7729     7752     7766     7775     7808
                           7899     7913     7992     8007     8017     8038     8059     8068     8130     8145     8155     8176     8197
                           8206     8278     8293     8303     8326     8345     8354     8440     8455     8465     8487     8506     8515
                           8593     8608     8618     8649     8663     8672     8770     8785     8795     8816     8834     8843     8997
                           9012     9022     9052     9066     9075     9112     9166     9181     9191     9221     9235     9244     9282
                           9339     9354     9364     9385     9400     9409     9504     9519     9529     9562     9576     9585     9619
                           9693     9708     9718     9748     9762     9771     9800     9856     9871     9881     9911     9925     9934
                           9962    10095    10110    10120    10162    10176    10185    10318    10333    10343    10372    10395    10473
                          10488    10498    10518    10534    10543    10633    10648    10658    10684    10699    10708    10737    10800
                          10815    10825    10844    10866    10875    10972    10987    10997    11023    11042    11051    11140    11155
                          11165    11193    11212    11221    11329    11344    11354    11396    11410    11419    11537    11552    11562
                          11595    11609    11618    11675    11752    11767    11777    11806    11820    11829    11988    12003    12013
                          12047    12061    12070    12124    12144    12196    12210    12219    12331    12346    12356    12380    12394
                          12403    12447    12461    12470    12485    12578    12593    12603    12632    12646    12655    12668    12699
                          12733    12739    12760    12773    12794    12807    12838    12851    12883    12895    13007#   13108#   13109#
```

```
PARAMETER CODING        MACY11 30(1046)  11-JAN-83  10:13  PAGE 310
CZUAAA.P11    11-JAN-83 09:29           CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL        1#    1389#     4684     4689     4711
BERROR        1#    1389#
BGNAU         1#    1389#     4828
BGNAUT        1#    1389#     4770
BGNCLN        1#    1389#     4782
BGNDU         1#    1389#     4817
BGNHRD        1#    1389#    13080
BGNHW         1#    1389#     1554
BGNINI        1#    1389#     4676
BGNMOD        1#    1389#     1392
BGNMSG        1#    1389#     2887     2900     2915     2936     2948     2960     2976     2995     3011     3023     3044     3070     3082
            3094     3107     3119     3144     3157     3177     3191     3206     3234     3262     3276     3305     3332     3359     3394
            3406     3444     3492     3511     3531     3551     3579     3598     3639     3666     3679     3699     3726     3738     3750
            3762     3774     3801     3821     3863     3875     3914     3938
BGNPRO        1#    1389#     4660
BGNPTA        1#    1389#
BGNRPT        1#    1389#     4643
BGNSEG        1#    1389#     4967     4985     5070     5088     5105     5122     5139     5234     5321     5347     5373     5402     5423
            5442     5492     5534     5550     5665     6170     6195     6229     6261     6347     6392     6409     6435     6461     6496
            6582     6622     6642     6669     6692     6768     6808     6911     6943     7041     7103     7159     7223     7270     7369
            7417     7534     7587     7693     7741     7846     7887     7981     8027     8119     8165     8267     8315     8429     8476
            8582     8634     8759     8805     8986     9037     9155     9206     9326     9374     9493     9548     9682     9733     9845
            9900    10084    10146    10307    10357    10462    10507    10622    10673    10726    10789    10833    10961    11009    11129
           11182    11318    11369    11379    11526    11575    11741    11786    11977    12032    12108    12182    12320    12366    12433
           12567    12621
BGNSET        1#    1389#
BGNSFT        1#    1389#
BGNSRV        1#    1389#     4851     4870     4996
BGNSUB        1#    1389#     4931     4961     5034     5064     5289     5315     5595     5618     5640     6164     6215     6341     6455
BGNSW         1#    1389#
BGNTST        1#    1389#     4922     5025     5177     5225     5284     5486     5530     5593     5706     6162     6338     6577     6761
            6905     7034     7216     7362     7527     7686     7839     7974     8112     8260     8422     8575     8752     8979     9148
            9321     9486     9675     9838    10077    10300    10455    10615    10782    10954    11122    11311    11519    11734    11970
           12313    12560    12727
BNCOMP        1#    1389#     4694     4699     4743
BNERRO        1#    1389#
BREAK         1#    1389#     4243     4390
BRESET        1#    1389#
CKLOOP        1#    1389#     4947     5050     5196     5246
CLOCK         1#    1389#     4707
CLOSE         1#    1389#
CLRVEC        1#    1389#     4949     5005     5052     5159     5198     5206     5248     5260     6984
COMMEN        1#    1389#
DELAY         1#    1389#
DESCRI        1#    1389#     1935
DEVTYP        1#    1389#     1927
DISPAT        1#    1389#     1495
DISPLA        1#    1389#
DOCLN         1#    1389#     4757     4955     5058     5204     5254
DODU          1#    1389#     4952     5055     5201     5251
DORPT         1#    1389#
ENDAU         1#    1389#     4833
ENDAUT        1#    1389#     4772
ENDCLN        1#    1389#     4812
ENDCOM        1#    1389#
ENDDU         1#    1389#     4822
```

```
ENDHRD         1#    1389#    13093
ENDHW          1#    1389#     1562
ENDUNI         1#    1389#     4760
ENDMOD         1#    1389#    13120
ENDMSG         1#    1389#     2896     2911     2932     2944     2956     2972     2991     3007     3019     3040     3066     3078     3090
             3103     3115     3140     3153     3173     3187     3202     3230     3258     3272     3301     3328     3355     3390     3402
             3440     3488     3507     3527     3547     3575     3594     3635     3662     3675     3695     3722     3734     3746     3758
             3770     3797     3817     3859     3871     3910     3934     3946
ENDPRO         1#    1389#     4667
ENDPTA         1#    1389#
ENDRPT         1#    1389#     4648
ENDSEG         1#    1389#     4982     4999     5085     5102     5119     5136     5153     5257     5337     5359     5390     5414     5439
             5458     5506     5546     5562     5679     6192     6209     6258     6275     6389     6406     6432     6449     6493     6507
             6619     6639     6666     6686     6721     6799     6873     6937     6981     7086     7153     7180     7261     7320 [illegible]
             7499     7572     7637     7731     7793     7877     7943     8019     8080     8157     8218     8305     8381     8467 [illegible]
             8620     8699     8797     8925     9024     9097     9193     9267     9366     9431     9531     9604     9720     9785     9855
             9947    10122    10231    10345    10413    10500    10566    10660    10720    10739    10827    10898    10999    11074    11167
            11254    11356    11442    11455    11564    11660    11779    11868    12015    12102    12176    12252    12358    12426    12497
            12605    12703
ENDSET         1#    1389#
ENDSFT         1#    1389#
ENDSRV         1#    1389#     4854     4874     4902
ENDSUB         1#    1389#     4958     5002     5061     5156     5311     5461     5615     5637     5682     6212     6395     6452     6526
ENDSW          1#    1389#
ENDTST         1#    1389#     5008     5162     5210     5263     5464     5511     5565     5685     5768     6298     6537     6730     6876
             6996     7187     7323     7502     7655     7814     7946     8083     8221     8384     8545     8704     8928     9120     9290
             9434     9629     9812     9973    10234    10423    10569    10748    10901     1082    11263    11465    11685    11882    12255
            12506    12706    12955
EQUALS         1#    1389#     1575
ERRDF          1#    1389#     4942     5045     5191     5241
ERRHRD         1#    1389#     4976     4993     5079     5096     5113     5130     5147     5302     5331     5353     5384     5408     5433
             5452     5500     5540     5556     5609     5631     5656     5673     5710     5727     5749     5762     6183     6203     6249
             6269     6289     6380     6400     6423     6443     6484     6501     6518     6610     6630     6657     6677     6705     6715
             6784     6792     6812     6833     6840     6860     6867     6931     6959     6975     6990     7061     7071     7107     7135
             7147     7174     7242     7252     7274     7286     7307     7314     7388     7398     7421     7442     7451     7474     7481
             7493     7553     7563     7591     7606     7615     7632     7645      771     7722     7745     7759     7768     7787     7801
             7862     7870     7892     7906     7915     7932     8000     8010     8031     8045     8052     8061     8074     8138     8148
             8169     8183     8190     8199     8212     8286     8296     8319 [illegible]    8347     8365     8375     8448     8458     8480
             8499     8508     8526     8536     8601     8611     8642     8656     8665     8683     8693     8778     8788     8809     8827
             8836     8857     8870     8879     8891     8900     8909     8919     9005     9015     9045     9059     9068     9091     9105
             9174     9184     9214     9228     9237     9261     9275     9347     9357     9378     9393     9402     9415     9425     9512
             9522     9555     9569     9578     9598     9612     9701     9711     9741     9755     9764     9779     9793     9864     9874
             9904     9918     9927     9941     9955    10103    10113    10155    10169    10178    10195    10211    10225    10326    10336
            10365    10388    10396    10407    10481    10491    10511    10527    10536    10550    10560    10641    10651    10677    10692
            10701    10714    10730    10808    10818    10837    10859    10868    10882    10892    10980    10990    11016    11035    11044
            11057    11067    11148    11158    11186    11205    11214    11230    11238    11248    11337    11347    11389    11403    11412
            11426    11436    11545    11555    11588    11602    11611    11628    11643    11654    11668    11760    11770    11799    11813
            11822    11836    11851    11862    11876    11996    12006    12040    12054    12063    12085    12096    12117    12131    12137
            12157    12168    12189    12203    12212    12236    12246    12339    12349    12373    12387    12396    12410    12420    12440
            12454    12463    12478    12491    12586    12596    12625    12639    12648    12661    12692    12753    12766    12787    12800
            12831    12844    12876    12888
ERROR          1#    1389#
ERRSF          1#    1389#
ERRSOF         1#    1389#
ERRTBL         1#    1389#
ESCAPE         1#    1389#     5307     5661     5715     5732     5754     6188     6254     6385     6428     6489     6615     6635     6662
```

```
           6682    6817    6964    7050    7066    7076    7112    7232    7247    7257    7279    7291    7378    7393    7403
           7426    7447    7456    7543    7558    7568    7596    7611    7620    7650    7702    7717    7727    7750    7764
           7773    7806    7897    7911    7990    8005    8015    8036    8057    8066    8128    8143    8153    8174    8195
           8204    8276    8291    8301    8324    8343    8352    8438    8453    8463    8485    8504    8513    8591    8606
           8616    8647    8661    8670    8768    8783    8793    8814    8832    8841    8995    9010    9020    9050    9064
           9073    9110    9164    9179    9189    9219    9233    9242    9280    9337    9352    9362    9383    9398    9407
           9502    9517    9527    9560    9574    9583    9617    9691    9706    9716    9746    9760    9769    9798    9854
           9869    9879    9909    9923    9932    9960   10093   10108   10118   10160   10174   10183   10316   10331   10341
          10370   10393   10471   10486   10496   10516   10532   10541   10631   10646   10656   10682   10697   10706   10735
          10798   10813   10823   10842   10864   10873   10970   10985   10995   11021   11040   11049   11138   11153   11163
          11191   11210   11219   11327   11342   11352   11394   11408   11417   11535   11550   11560   11593   11607   11616
          11673   11750   11765   11775   11804   11818   11827   11986   12001   12011   12045   12059   12068   12122   12142
          12194   12208   12217   12329   12344   12354   12378   12392   12401   12445   12459   12468   12483   12576   12591
          12601   12630   12644   12653   12666   12697   12737   12758   12771   12792   12805   12836   12849   12881   12893
EXIT          1#   1389#   4645    4799    4819    4830   12731
FEQUAL        1#   1389#
GETBYT        1#   1389#
GETPRI        1#   1389#
GETWOR        1#   1389#
GMANIA        1#   1389#
GMANID        1#   1389#
GMANIL        1#   1389#
GPHARD        1#   1389#   4739
GPRMA         1#   1389#  13083   13088
GPRMD         1#   1389#
GPRML         1#   1389#
HEADER        1#   1389#   1401
IOLOOP        1#   1389#
IOSETU        1#   1389#
IOSTAR        1#   1389#
KT11          1#   1389#
LASTAD        1#   1389#
MANUAL        1#   1389#
MEMORY        1#   1389#   4702
MSBYTE        1#   1389#   1402#   1409    1410
MSCHEC        1#   1389#   4646#   4800#   4820#   4831#  12732#
MSCNTO        1#   1389#  13084#  13059#
MSCOUN        1#   1389#   2890#   2903#   2918#   2925#   2939#   2951#   2963#   2979#   2985#   2998#   3014#   3026#   3032#
           3047#   3053#   3060#   3073#   3085#   3097#   3110#   3122#   3128#   3134#   3147#   3160#   3167#   3180#   3194#
           3209#   3216#   3223#   3237#   3244#   3251#   3265#   3279#   3286#   3294#   3308#   3314#   3321#   3335#   3341#
           3348#   3363#   3371#   3381#   3397#   3409#   3415#   3424#   3431#   3449#   3456#   3462#   3468#   3475#   3482#
           3495#   3501#   3514#   3520#   3534#   3540#   3554#   3562#   3569#   3582#   3588#   3601#   3607#   3614#   3621#
           3628#   3642#   3648#   3656#   3669#   3682#   3689#   3702#   3709#   3715#   3729#   3741#   3753#   3765#   3777#
           3783#   3795#   3804#   3810#   3824#   3831#   3838#   3845#   3852#   3866#   3880#   3888#   3896#   3904#   3917#
           3923#   3929#   3941#   4353#   4468#   4482#   4503#   4569#   4584#   4597#   4612#   4625#   4714#   4792#  12919#
          12926#  12933#  12940#  12947#
MSDATA        1#   1389#   1402#   1411    1413    1415    1417    1419    1421    1423    1425    1427    1429    1431    1433
           1435    1437    1439    1441#   1443    1445    1448    1451    1453    1455    1457    1459    1461    1463    1465
           1467    1469    1471    1473    1475    1477    1479    1481    1483    1485    1928#   1936#
MSDECR        1#   1389#   1563#   2897#   2912#   2933#   2945#   2957#   2973#   2992#   3008#   3020#   3041#   3067#   3079#
           3091#   3104#   3116#   3141#   3154#   3174#   3188#   3203#   3231#   3259#   3273#   3302#   3329#   3356#   3391#
           3403#   3441#   3489#   3508#   3528#   3548#   3576#   3595#   3636#   3663#   3676#   3696#   3723#   3735#   3747#
           3759#   3771#   3798#   3818#   3860#   3872#   3911#   3935#   3947#   4649#   4668#   4761#   4773#   4813#   4823#
           4834#   4855#   4875#   4903#   4959#   4983#   5000#   5003#   5009#   5062#   5086#   5103#   5120#   5137#   5154#
           5157#   5163#   5211#   5258#   5264#   5312#   5338#   5360#   5391#   5415#   5440#   5459#   5462#   5465#   5507#
           5512#   5547#   5563#   5566#   5616#   5638#   5680#   5683#   5686#   5769#   6193#   6210#   6213#   6259#   6276#
          6296#   6299#   6390#   6407#   6433#   6450#   6453#   6494#   6508#   6527#   6538#   6620#   6640#   6667#   6687#
          6722#   6731#   6800#   6874#   6877#   6938#   6982#   6997#   7087#   7154#   7181#   7188#   7262#   7321#   7324#
          7408#   7500#   7503#   7573#   7638#   7656#   7732#   7794#   7815#   7878#   7944#   7947#   8020#   8081#   8084#
          8158#   8219#   8222#   8306#   8382#   8385#   8468#   8543#   8546#   8621#   8700#   8705#   8798#   8926#   8929#
          9025#   9098#   9121#   9194#   9268#   9291#   9367#   9432#   9435#   9532#   9605#   9630#   9721#   9786#   9813#
          9884#   9948#   9974#  10123#  10232#  10235#  10346#  10414#  10424#  10501#  10567#  10570#  10661#  10721#  10740#
         10749#  10828#  10899#  10902#  11000#  11075#  11083#  11168#  11255#  11264#  11357#  11443#  11456#  11466#  11565#
         11661#  11686#  11780#  11869#  11883#  12016#  12103#  12177#  12253#  12256#  12359#  12427#  12498#  12507#  12606#
         12704#  12707#  12956#  13094#  13121#
M$DEFA       1#   1389#  13084#  13089#
M$ENDE       1#   1389#   1563#   2897#   2912#   2933#   2945#   2957#   2973#   2990#   3008#   3020#   3041#   3067#   3079#
          3091#   3104#   3116#   3141#   3154#   3174#   3188#   3203#   3231#   3259#   3273#   3302#   3329#   3356#   3391#
          3403#   3441#   3489#   3508#   3528#   3548#   3576#   3595#   3636#   3663#   3676#   3696#   3723#   3735#   3747#
          3759#   3771#   3798#   3818#   3860#   3872#   3911#   3935#   3947#   4649#   4761#   4773#   4813#   4823#   4834#
          4855#   4875#   4903#   4959#   4983#   5000#   5003#   5009#   5062#   5086#   5103#   5120#   5137#   5154#   5157#
          5163#   5211#   5258#   5264#   5312#   5338#   5360#   5391#   5415#   5440#   5459#   5462#   5465#   5507#   5512#
          5547#   5563#   5566#   5616#   5638#   5680#   5683#   5686#   5769#   6193#   6210#   6213#   6259#   6276#   6296#
          6299#   6390#   6407#   6433#   6450#   6453#   6494#   6508#   6527#   6538#   6620#   6640#   6667#   6687#   6722#
          6731#   6800#   6874#   6877#   6938#   6982#   6997#   7087#   7154#   7181#   7188#   7262#   7321#   7324#   7408#
          7500#   7503#   7573#   7638#   7656#   7732#   7794#   7815#   7878#   7944#   7947#   8020#   8081#   8084#   8158#
          8219#   8222#   8306#   8382#   8385#   8468#   8543#   8546#   8621#   8700#   8705#   8798#   8926#   8929#   9025#
          9098#   9121#   9194#   9268#   9291#   9367#   9432#   9435#   9532#   9605#   9630#   9721#   9786#   9813#   9884#
          9948#   9974#  10123#  10232#  10235#  10346#  10414#  10424#  10501#  10567#  10570#  10661#  10721#  10740#  10749#
         10828#  10899#  10902#  11000#  11075#  11083#  11168#  11255#  11264#  11357#  11443#  11456#  11466#  11565#  11661#
         11686#  11780#  11869#  11883#  12016#  12103#  12177#  12253#  12256#  12359#  12427#  12498#  12507#  12606#  12704#
         12707#  12956#  13094#  13121#
M$ERRI       1#   1389#   4943#   4977#   4994#   5046#   5080#   5097#   5114#   5131#   5148#   5192#   5242#   5303#   5332#
          5354#   5385#   5409#   5434#   5453#   5501#   5541#   5557#   5610#   5632#   5657#   5674#   5711#   5728#   5750#
          5763#   6184#   6204#   6250#   6270#   6290#   6381#   6401#   6424#   6444#   6485#   6502#   6519#   6611#   6631#
          6658#   6678#   6706#   6716#   6785#   6793#   6813#   6834#   6841#   6861#   6868#   6932#   6960#   6976#   6991#
          7062#   7072#   7108#   7136#   7148#   7175#   7243#   7253#   7275#   7287#   7308#   7315#   7389#   7399#   7422#
          7443#   7452#   7465#   7482#   7494#   7554#   7564#   7592#   7607#   7616#   7633#   7646#   7713#   7723#   7746#
          7760#   7769#   7788#   7802#   7863#   7871#   7893#   7907#   7916#   7933#   8001#   8011#   8032#   8046#   8053#
          8062#   8075#   8139#   8149#   8170#   8184#   8191#   8200#   8213#   8287#   8297#   8320#   8339#   8348#   8366#
          8376#   8449#   8459#   8481#   8500#   8509#   8527#   8537#   8602#   8612#   8643#   8657#   8666#   8684#   8694#
          8779#   8789#   8810#   8828#   8837#   8858#   8871#   8880#   8892#   8901#   8910#   8920#   9006#   9016#   9046#
          9060#   9069#   9092#   9106#   9175#   9185#   9215#   9229#   9238#   9262#   9276#   9348#   9358#   9379#   9394#
          9403#   9416#   9426#   9513#   9523#   9556#   9570#   9579#   9599#   9613#   9702#   9712#   9742#   9756#   9765#
          9780#   9797#   9865#   9875#   9905#   9919#   9928#   9942#   9956#  10104#  10114#  10156#  10170#  10179#  10196#
         10212#  10226#  10327#  10337#  10366#  10389#  10397#  10408#  10482#  10492#  10512#  10528#  10537#  10551#  10561#
         10642#  10652#  10676#  10693#  10702#  10715#  10731#  10809#  10819#  10838#  10860#  10869#  10883#  10893#  10981#
         10991#  11017#  11036#  11045#  11058#  11068#  11149#  11159#  11187#  11206#  11215#  11231#  11239#  11249#  11338#
         11348#  11390#  11404#  11413#  11427#  11437#  11546#  11556#  11589#  11603#  11612#  11629#  11644#  11655#  11669#
         11761#  11771#  11800#  11814#  11823#  11837#  11852#  11863#  11877#  11997#  12007#  12041#  12055#  12064#  12086#
         12097#  12118#  12132#  12138#  12158#  12169#  12190#  12204#  12213#  12237#  12247#  12340#  12350#  12374#  12388#
         12397#  12411#  12421#  12441#  12455#  12464#  12479#  12492#  12587#  12597#  12626#  12640#  12649#  12662#  12693#
         12754#  12767#  12788#  12801#  12832#  12845#  12877#  12889#
M$ESCA       1#   1389#   5308#   5309    5662#   5663    5716#   5717    5733#   5734    5755#   5756    6189#   6190    6255#
          6256    6386#   6387    6429#   6430    6490#   6491    6616#   6617    6636#   6637    6663#   6664    6683#   6684
          6818#   6819    6965#   6966    7051#   7052    7067#   7068    7077#   7078    7113#   7114    7233#   7234    7248#
          7249    7258#   7259    7280#   7281    7292#   7293    7379#   7380    7394#   7395    7404#   7405    7427#   7428
          7448#   7449    7457#   7458    7544#   7545    7559#   7560    7569#   7570    7597#   7598    7612#   7613    7621#
          7622    7651#   7652    7703#   7704    7718#   7719    7728#   7729    7751#   7752    7765#   7766    7774#   7775
          7807#   7808    7898#   7899    7912#   7913    7991#   7992    8006#   8007    8016#   8017    8037#   8038    8058#
          8059    8067#   8068    8129#   8130    8144#   8145    8154#   8155    8175#   8176    8196#   8197    8205#   8206
          8277#   8278    8292#   8293    8302#   8303    8325#   8326    8344#   8345    8353#   8354    8439#   8440    8454#
```

```
           8455    8464#   8465    8486#   8487    8505#   8506    8514#   8515    8592#   8593    8607#   8608    8617#   8618
           8648#   8649    8662#   8663    8671#   8672    8769#   8770    8784#   8785    8794#   8795    8815#   8816    8833#
           8834    8842#   8843    8996#   8997    9011#   9012    9021#   9022    9051#   9052    9065#   9066    9074#   9075
           9111#   9112    9165#   9166    9180#   9181    9190#   9191    9220#   9221    9234#   9235    9243#   9244    9281#
           9282    9338#   9339    9353#   9354    9363#   9367    9384#   9385    9399#   9400    9408#   9409    9503#   9504
           9518#   9519    9528#   9529    9561#   9562    9575#   9576    9584#   9585    9618#   9619    9692#   9693    9707#
           9708    9717#   9718    9747#   9748    9761#   9762    9770#   9771    9799#   9800    9855#   9856    9870#   9871
           9880#   9881    9910#   9911    9924#   9925    9933#   9934    9961#   9962   10094#  10095   10109#  10110   10119#
          10120   10161#  10162   10175#  10176   10184#  10185   10317#  10318   10332#  10333   10342#  10343   10371#  10372
          10394#  10395   10472#  10473   10487#  10488   10497#  10498   10517#  10518   10533#  10534   10542#  10543   10632#
          10633   10647#  10648   10657#  10658   10683#  10684   10698#  10699   10707#  10708   10736#  10737   10799#  10800
          10814#  10815   10824#  10825   10843#  10844   10865#  10866   10874#  10875   10971#  10972   10986#  10987   10996#
          10997   11022#  11023   11041#  11042   11050#  11051   11139#  11140   11154#  11155   11164#  11165   11192#  11193
          11211#  11212   11220#  11221   11328#  11329   11343#  11344   11353#  11354   11395#  11396   11409#  11410   11418#
          11419   11536#  11537   11551#  11552   11561#  11562   11594#  11595   11608#  11609   11617#  11618   11674#  11675
          11751#  11752   11766#  11767   11776#  11777   11805#  11806   11819#  11820   11828#  11829   11987#  11988   12002#
          12003   12012#  12013   12046#  12047   12060#  12061   12069#  12070   12123#  12124   12143#  12144   12195#  12196
          12209#  12210   12218#  12219   12330#  12331   12345#  12346   12355#  12356   12379#  12380   12393#  12394   12402#
          12403   12446#  12447   12460#  12461   12469#  12470   12484#  12485   12577#  12578   12592#  12593   12602#  12603
          12631#  12632   12645#  12646   12654#  12655   12667#  12668   12698#  12699   12738#  12739   12759#  12760   12772#
          12773   12793#  12794   12806#  12807   12837#  12838   12850#  12851   12882#  12883   12894#  12895
M$ESCS        1#   1389#   5308#   5662#   5716#   5733#   5755#   6189#   6255#   6386#   6429#   6490#   6616#   6636#   6663#
           6683#   6818#   6965#   7051#   7067#   7077#   7113#   7233#   7248#   7258#   7280#   7292#   7379#   7394#   7404#
           7427#   7448#   7457#   7544#   7559#   7569#   7597#   7612#   7621#   7651#   7703#   7718#   7728#   7751#   7765#
           7774#   7807#   7898#   7912#   7991#   8006#   8016#   8037#   8058#   8067#   8129#   8144#   8154#   8175#   8196#
           8205#   8277#   8292#   8302#   8325#   8344#   8353#   8439#   8454#   8464#   8486#   8505#   8514#   8592#   8607#
           8617#   8648#   8662#   8671#   8769#   8784#   8794#   8815#   8833#   8842#   8996#   9011#   9021#   9051#   9065#
           9074#   9111#   9165#   9180#   9190#   9220#   9234#   9243#   9281#   9338#   9353#   9363#   9384#   9399#   9408#
           9503#   9518#   9528#   9561#   9575#   9584#   9618#   9692#   9707#   9717#   9747#   9761#   9770#   9799#   9855#
           9870#   9880#   9910#   9924#   9933#   9961#  10094#  10109#  10119#  10161#  10175#  10184#  10317#  10332#  10342#
          10371#  10394#  10472#  10487#  10497#  10517#  10533#  10542#  10632#  10647#  10657#  10683#  10698#  10707#  10736#
          10799#  10814#  10824#  10843#  10865#  10874#  10971#  10986#  10996#  11022#  11041#  11050#  11139#  11154#  11164#
          11192#  11211#  11220#  11328#  11343#  11353#  11395#  11409#  11418#  11536#  11551#  11561#  11594#  11608#  11617#
          11674#  11751#  11766#  11776#  11805#  11819#  11828#  11987#  12002#  12012#  12046#  12060#  12069#  12123#  12143#
          12195#  12209#  12218#  12330#  12345#  12355#  12379#  12393#  12402#  12446#  12460#  12469#  12484#  12577#  12592#
          12602#  12631#  12645#  12654#  12667#  12698#  12738#  12759#  12772#  12793#  12806#  12837#  12850#  12882#  12894#
M$EXCP        1#   1389#  13084#  13089#
M$EXIT        1#   1389#   4646#   4800#   4801    4820#   4831#  12732#  12733
M$EXSE        1#   1389#   4646#   4800#   4820#   4831#  12732#
M$EXTJ        1#   1389#   4646#   4647    4800#   4820#   4821    4831#   4832   12732#
M$GEN         1#   1389#   1402#   1411#   1413#   1415#   1417#   1419#   1421#   1423#   1425#   1427#   1429#   1431#   1433#
           1435#   1437#   1439#   1441#   1443#   1445#   1448#   1451#   1453#   1455#   1457#   1459#   1461#   1463#   1465#
           1467#   1469#   1471#   1473#   1475#   1477#   1479#   1481#   1483#   1485#   1497#   1556#   1557#   1563#   1928#
           1936#   2888#   2897#   2901#   2912#   2916#   2933#   2937#   2945#   2949#   2957#   2961#   2973#   2977#   2992#
           2996#   3008#   3012#   3020#   3024#   3041#   3045#   3067#   3071#   3079#   3083#   3091#   3095#   3104#   3108#
           3116#   3120#   3141#   3145#   3154#   3158#   3174#   3178#   3188#   3192#   3203#   3207#   3231#   3235#   3259#
           3263#   3273#   3277#   3302#   3306#   3329#   3333#   3356#   3360#   3391#   3395#   3403#   3407#   3441#   3445#
           3489#   3493#   3508#   3512#   3528#   3532#   3548#   3552#   3576#   3580#   3595#   3599#   3636#   3640#   3663#
           3667#   3676#   3680#   3696#   3700#   3723#   3727#   3735#   3739#   3747#   3751#   3759#   3763#   3771#   3775#
           3798#   3802#   3818#   3822#   3860#   3864#   3872#   3876#   3911#   3915#   3935#   3939#   3947#   4644#   4649#
           4661#   4677#   4761#   4771#   4773#   4783#   4813#   4818#   4823#   4829#   4834#   4852#   4855#   4871#   4875#
           4897#   4903#   4923#   4932#   4959#   4962#   4983#   5000#   5003#   5009#   5026#   5035#   5062#   5065#   5086#
           5103#   5120#   5137#   5154#   5157#   5163#   5178#   5211#   5226#   5258#   5264#   5285#   5290#   5312#   5316#
           5338#   5360#   5391#   5415#   5440#   5459#   5462#   5465#   5487#   5507#   5512#   5531#   5547#   5563#   5566#
           5594#   5596#   5616#   5619#   5638#   5641#   5680#   5683#   5686#   5707#   5769#   6163#   6165#   6193#   6210#
           6213#   6216#   6259#   6276#   6296#   6299#   6339#   6342#   6390#   6407#   6433#   6450#   6453#   6456#   6494#
           6508#    6527#    6538#    6578#    6620#    6640#    6667#    6687#    6722#    6731#    6762#    6800#    6874#    6877#    6906#
           6938#    6982#    6997#    7035#    7087#    7154#    7181#    7188#    7217#    7262#    7321#    7324     7363#    7408#    7500#
           7503#    7528#    7573#    7638#    7656#    7687#    7732#    7794#    7815#    7840#    7878#    7944     7947#    7975#    8020#
           8081#    8084#    8113#    8158#    8219#    8222#    8261#    8306#    8382#    8385#    8423#    8468     8543#    8546#    8576#
           8621#    8700#    8705#    8753#    8798#    8926#    8929#    8980#    9025#    9098#    9121#    9149     9194#    9208#    9291#
           9322#    9367#    9432#    9435#    9487#    9532#    9605#    9630#    9676#    9721#    9786#    9813#    9839#    9884#    9948#
           9974#   10078#   10123#   10232#   10235#   10301#   10346#   10414#   10424#   10456#   10501#   10567#   10570    10616#   10661#
          10721#   10740#   10749#   10783#   10828#   10899#   10902#   10955#   11000#   11075#   11083#   11123#   11168    11255#   11264#
          11312#   11357#   11443#   11456#   11466#   11520#   11565#   11661#   11686#   11735#   11780#   11869#   11883    11971#   12016#
          12103#   12177#   12253#   12256#   12314#   12359#   12427#   12498#   12507#   12561#   12606#   12704#   12707#   12728#   12956#
          13082#   13095#
M$GENB        1#    1389#
M$GETS        1#    1389#    1563#    2897#    2912#    2933#    2945#    2957#    2973#    2992#    3008#    3020#    3041#    3067#    3079#
           3091#    3104#    3116#    3141#    3154#    3174#    3188#    3203#    3231#    3259#    3273#    3302#    3329#    3356#    3391#
           3403#    3441#    3489#    3508#    3528#    3548#    3576#    3595#    3636#    3663#    3676#    3696#    3723#    3735#    3747#
           3759#    3771#    3798#    3818#    3860#    3872#    3911#    3935#    3947#    4649#    4668#    4761#    4773#    4813#    4823#
           4834#    4855#    4875#    4903#    4959#    4983#    5000#    5003#    5009#    5062#    5086#    5103#    5120#    5137#    5154#
           5157#    5163#    5211#    5258#    5264#    5312#    5338#    5360#    5391#    5415#    5440#    5459#    5462#    5465#    5507#
           5512#    5547#    5563#    5566#    5616#    5638#    5680#    5683#    5686#    5769#    6193#    6210#    6213#    6259#    6276#
           6296#    6299#    6390#    6407#    6433#    6450#    6453#    6494#    6508#    6527#    6538#    6620#    6640#    6667#    6687#
           6722#    6731#    6800#    6874#    6877#    6938#    6982#    6997#    7087#    7154#    7181#    7188#    7262#    7321#    7324#
           7408#    7500#    7503#    7573#    7638#    7656#    7732#    7794#    7815#    7878#    7944#    7947#    8020#    8081#    8084#
           8158#    8219#    8222#    8306#    8382#    8385#    8468#    8543#    8546#    8621#    8700#    8705#    8798#    8926#    8929#
           9025#    9098#    9121#    9194#    9268#    9291#    9367#    9432#    9435#    9532#    9605#    9630#    9721#    9786#    9813#
           9884#    9948#    9974#   10123#   10232#   10235#   10346#   10414#   10424#   10501#   10567#   10570#   10661#   10721#   10740#
          10749#   10828#   10899#   10902#   11000#   11075#   11083#   11168#   11255#   11264#   11357#   11443#   11456#   11466#   11565#
          11661#   11686#   11780#   11869#   11883#   12016#   12103#   12177#   12253#   12256#   12359#   12427#   12498#   12507#   12606#
          12704#   12707#   12956#   13094#   13121
M$GETT        1#    1389#    4646#    4800#    4820#    4831#    5308#    5662#    5716#    5733#    5755#    6189#    6255#    6386#    6429#
           6490#    6616#    6636#    6663#    6683#    6818#    6965#    7051#    7067#    7077#    7113#    7233#    7248#    7258#    7280#
           7292#    7379#    7394#    7404#    7427#    7448#    7457#    7544#    7559#    7569#    7597#    7612#    7621#    7651#    7703#
           7718#    7728#    7751#    7765#    7774#    7807#    7898#    7912#    7991#    8006#    8016#    8037#    8058#    8067#    8129#
           8144#    8154#    8175#    8196#    8205#    8277#    8292#    8302#    8325#    8344#    8353#    8439#    8454#    8464#    8486#
           8505#    8514#    8592#    8607#    8617#    8648#    8662#    8671#    8769#    8784#    8794#    8815#    8833#    8842#    8996#
           9011#    9021#    9051#    9065#    9074#    9111#    9165#    9180#    9190#    9220#    9234#    9243#    9281#    9338#    9353#
           9363#    9384#    9399#    9408#    9503#    9518#    9528#    9561#    9575#    9584#    9618#    9692#    9707#    9717#    9747#
           9761#    9770#    9799#    9855#    9870#    9880#    9910#    9924#    9933#    9961#   10094#   10109#   10119#   10161#   10175#
          10184#   10317#   10332#   10342#   10371#   10394#   10472#   10487#   10497#   10517#   10533#   10542#   10632#   10647#   10657#
          10683#   10698#   10707#   10736#   10799#   10814#   10824#   10843#   10865#   10874#   10971#   10986#   10996#   11022#   11041#
          11050#   11139#   11154#   11164#   11192#   11211#   11220#   11328#   11343#   11353#   11395#   11409#   11418#   11536#   11551#
          11561#   11594#   11608#   11617#   11674#   11751#   11766#   11776#   11805#   11819#   11828#   11987#   12002#   12012#   12046#
          12060#   12069#   12123#   12143#   12195#   12209#   12218#   12330#   12345#   12355#   12379#   12393#   12402#   12446#   12460#
          12469#   12484#   12577#   12592#   12602#   12631#   12645#   12654#   12667#   12698#   12732#   12738#   12759#   12772#   12793#
          12806#   12837#   12850#   12882#   12894#
M$GNGB        1#    1389#    1393#    1402#    1411#    1413#    1415#    1417#    1419#    1421#    1423#    1425#    1427#    1429#    1431#
           1433#    1435#    1437#    1439#    1441#    1443#    1445#    1448#    1451#    1453#    1455#    1457#    1459#    1461#    1463#
           1465#    1467#    1469#    1471#    1473#    1475#    1477#    1479#    1481#    1483#    1485#    1496#    1497     1555#    1556
           1557     1928#    1936#    2888#    2901#    2916#    2937#    2949#    2961#    2977#    2996#    3012#    3024#    3045#    3071#
           3083#    3095#    3108#    3120#    3145#    3158#    3178#    3192#    3207#    3235#    3263#    3277#    3306#    3333#    3360#
           3395#    3407#    3445#    3493#    3512#    3532#    3552#    3580#    3599#    3640#    3667#    3680#    3700#    3727#    3739#
           3751#    3763#    3775#    3802#    3822#    3864#    3876#    3915#    3939#    4644#    4661#    4677#    4771#    4783#    4818#
           4829#    4852#    4871#    4897#   13081#   13082
M$GNIN        1#    1389#    1402#    1403     1404     1405     1406     1407     1408     1409#    1410#    1411#    1412     1413#    1414
           1415#    1416     1417#    1418     1419#    1420     1421#    1422     1423#    1424     1425#    1426     1427#    1428     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    1496#   1498#   1499#
1500#   1501#   1502#   1503#   1504#   1505#   1506#   1507#   1508#   1509#   1510#   1511#   1512#   1513#   1514#
1515#   1516#   1517#   1518#   1519#   1520#   1521#   1522#   1523#   1524#   1525#   1526#   1527#   1528#   1529#
1530#   1531#   1532#   1533#   1534#   1535#   1536#   1537#   1538#   1539#   1540#   1541#   1542#   1543#   1555#
1928#   1929    1930    1936#   1937    1941    2890#   2891#   2892#   2893    2894#   2895    2898#   2903#   2904#
2905#   2906#   2907#   2908    2909#   2910    2913#   2918#   2919#   2920#   2921    2922#   2923    2925#   2926#
2927#   2928#   2929    2930#   2931    2934#   2939#   2940#   2941    2942#   2943    2946#   2951#   2952#   2953
2954#   2955    2958#   2963#   2964#   2965#   2966#   2967#   2968#   2969    2970#   2971    2974#   2979#   2980#
2981    2982#   2983    2985#   2986#   2987#   2988    2989#   2990    2993#   2998#   2999#   3000#   3001#   3002#
3003#   3004    3005#   3006    3009#   3014#   3015#   3016    3017#   3018    3021#   3026#   3027#   3028    3029#
3030    3032#   3033#   3034#   3035#   3036#   3037    3038#   3039    3042#   3047#   3048#   3049    3050#   3051
3053#   3054#   3055#   3056    3057#   3058    3060#   3061#   3062#   3063    3064#   3065    3068#   3073#   3074#
3075    3076#   3077    3080#   3085#   3086#   3087    3088#   3089    3092#   3097#   3098#   3099#   3100    3101#
3102    3105#   3110#   3111#   3112    3113#   3114    3117#   3122#   3123#   3124    3125#   3126    3128#   3129#
3130    3131#   3132    3134#   3135#   3136#   3137    3138#   3139    3142#   3147#   3148#   3149#   3150    3151#
3152    3155#   3160#   3161#   3162#   3163    3164#   3165    3167#   3168#   3169#   3170    3171#   3172    3175#
3180#   3181#   3182#   3183#   3184    3185#   3186    3189#   3194#   3195#   3196#   3197#   3198#   3199    3200#
3201    3204#   3209#   3210#   3211#   3212    3213#   3214    3216#   3217#   3218#   3219    3220#   3221    3223#
3224#   3225#   3226#   3227    3228#   3229    3232#   3237#   3238#   3239#   3240    3241#   3242    3244#   3245#
3246#   3247    3248#   3249    3251#   3252#   3253#   3254#   3255    3256#   3257    3260#   3265#   3266#   3267#
3268#   3269    3270#   3271    3274#   3279#   3280#   3281#   3282    3283#   3284    3286#   3287#   3288#   3289#
3290    3291#   3292    3294#   3295#   3296#   3297#   3298    3299#   3300    3303#   3308#   3309#   3310    3311#
3312    3314#   3315#   3316#   3317    3318#   3319    3321#   3322#   3323#   3324#   3325    3326#   3327    3330#
3335#   3336#   3337    3338#   3339    3341#   3342#   3343#   3344    3345#   3346    3348#   3349#   3350#   3351#
3352    3353#   3354    3357#   3363#   3364#   3365#   3366    3367#   3368    3371#   3372#   3373#   3374#   3375#
3376    3377#   3378    3381#   3382#   3383#   3384#   3385#   3386    3387#   3388    3392#   3397#   3398#   3399
3400#   3401    3404#   3409#   3410#   3411    3412#   3413    3415#   3416#   3417    3418#   3419    3424#   3425#
3426#   3427    3428#   3429    3431#   3432#   3433#   3434    3435#   3436    3442#   3449#   3450#   3451    3452#
3453    3456#   3457#   3458    3459#   3460    3462#   3463#   3464    3465#   3466    3468#   3469#   3470#   3471
3472#   3473    3475#   3476#   3477#   3478    3479#   3480    3482#   3483#   3484#   3485    3486#   3487    3490#
3495#   3496#   3497    3498#   3499    3501#   3502#   3503#   3504    3505#   3506    3509#   3514#   3515#   3516
3517#   3518    3520#   3521#   3522#   3523#   3524    3525#   3526    3529#   3534#   3535#   3536    3537#   3538
3540#   3541#   3542#   3543#   3544    3545#   3546    3549#   3554#   3555#   3556#   3557#   3558    3559#   3560
3562#   3563#   3564#   3565    3566#   3567    3569#   3570#   3571#   3572    3573#   3574    3577#   3582#   3583#
3584    3585#   3586    3588#   3589#   3590#   3591    3592#   3593    3596#   3601#   3602#   3603    3604#   3605
3607#   3608#   3609#   3610    3611#   3612    3614#   3615#   3616#   3617    3618#   3619    3621#   3622#   3623#
3624    3625#   3626    3628#   3629#   3630#   3631#   3632    3633#   3634    3637#   3642#   3643#   3644    3645#
3646    3648#   3649#   3650#   3651#   3652    3653#   3654    3656#   3657#   3658#   3659    3660#   3661    3664#
3669#   3670#   3671#   3672    3673#   3674    3677#   3682#   3683#   3684#   3685    3686#   3687    3689#   3690#
3691#   3692    3693#   3694    3697#   3702#   3703#   3704#   3705    3706#   3707    3709#   3710#   3711    3712#
3713    3715#   3716#   3717#   3718#   3719    3720#   3721    3724#   3729#   3730#   3731    3732#   3733    3736#
3741#   3742#   3743    3744#   3745    3748#   3753#   3754#   3755    3756#   3757    3760#   3765#   3766#   3767
3768#   3769    3772#   3777#   3778#   3779    3780#   3781    3783#   3784#   3785#   3786    3787#   3788    3790#
3791#   3792#   3793#   3794    3795#   3796    3799#   3804#   3805#   3806    3807#   3808    3810#   3811#   3812#
3813#   3814    3815#   3816    3819#   3824#   3825#   3826#   3827    3828#   3829    3831#   3832#   3833#   3834
3835#   3836    3838#   3839#   3840#   3841    3842#   3843    3845#   3846#   3847#   3848    3849#   3850    3852#
3853#   3854#   3855#   3856    3857#   3858    3861#   3866#   3867#   3868    3869#   3870    3873#   3880#   3881#
3882    3883#   3884    3888#   3889#   3890    3891#   3892    3896#   3897#   3898    3899#   3900    3904#   3905#
3906    3907#   3908    3912#   3917#   3918#   3919    3920#   3921    3923#   3924#   3925    3926#   3927    3929#
3930#   3931    3932#   3933    3936#   3941#   3942#   3943    3944#   3945    3948#   4200#   4201#   4214#   4215#
4244#   4353#   4354#   4355#   4356    4357#   4358    4391#   4468#   4469#   4470#   4471#   4472#   4473#   4474
4475#   4476    4482#   4483#   4484    4485#   4486    4503#   4504#   4505#   4506    4507#   4508    4569#   4570#
4571#   4572    4573#   4574    4584#   4585#   4586#   4587    4588#   4589    4597#   4598#   4599#   4600    4601#
4602    4612#   4613#   4614#   4615    4616#   4617    4625#   4626#   4627#   4628    4629#   4630    4646#   4647#
4650#   4682#   4683#   4685#   4687#   4688#   4690#   4692#   4693#   4695#   4697#   4698#   4700#   4703#   4704#
```

```
4708#   4709#   4710#   4712#   4714#   4715#   4716    4717#   4718    4727#   4728#   4729#   4730#   4731#   4732
4740#   4741#   4742#   4744#   4758#   4762#   4774#   47[illegible]   4793#   4794    4795#   4796    4800#   4801#   4814#
4820#   4821#   4824#   4831#   4832#   4835#   4855#   48[illegible]   4875#   4876    4903#   4904    4925#   4926#   4927#
4928#   4929#   4930    4933#   4943#   4944#   4945#   49[illegible]   4948#   4950#   4951#   4953#   4954#   4956#   4960#
4963#   4968#   4977#   4978#   4979#   4980#   4984#   498[illegible]#   4994#   4995#   4996#   4997#   5001#   5004#   5006#
5007#   5010#   5028#   5029#   5030#   5031#   5032#   5033    5036#   5046#   5047#   5048#   5049#   5051#   5053#
5054#   5056#   5057#   5059#   5063#   5066#   5071#   5080#   5081#   5082#   5083#   5087#   5089#   5097#   5098#
5099#   5100#   5104#   5106#   5114#   5115#   5116#   5117#   5121#   5123#   5131#   5132#   5133#   5134#   5138#
5140#   5148#   5149#   5150#   5151#   5155#   5158#   5160#   5161#   5164#   5180#   5181#   5182#   5183#   5184#
5185    5192#   5193#   5194#   5195#   5197#   5199#   5200#   5202#   5203#   5205#   5207#   5208#   5212#   5228#
5229#   5230#   5231#   5232#   5233    5235#   5242#   5243#   5244#   5245#   5247#   5249#   5250#   5252#   5253#
5255#   5259#   5261#   5262#   5265#   5291#   5303#   5304#   5305#   5306#   5308#   5309#   5313#   5317#   5322#
5332#   5333#   5334#   5335#   5339#   5348#   5354#   5355#   5356#   5357#   5361#   5374#   5385#   5386#   5387#
5388#   5392#   5403#   5409#   5410#   5411#   5412#   5416#   5424#   5434#   5435#   5436#   5437#   5441#   5443#
5453#   5454#   5455#   5456#   5460#   5463#   5466#   5493#   5501#   5502#   5503#   5504#   5508#   5513#   5535#
5541#   5542#   5543#   5544#   5548#   5551#   5557#   5558#   5559#   5560#   5564#   5567#   5597#   5610#   5611#
5612#   5613#   5617#   5620#   5632#   5633#   5634#   5635#   5639#   5642#   5657#   5658#   5659#   5660#   5662#
5663#   5666#   5674#   5675#   5676#   5677#   5681#   5684#   5687#   5711#   5712#   5713#   5714#   5716#   5717#
5728#   5729#   5730#   5731#   5733#   5734#   5750#   5751#   5752#   5753#   5755#   5756#   5763#   5764#   5765#
5766#   5770#   6166#   6171#   6184#   6185#   6186#   6187#   6189#   6190#   6194#   6196#   6204#   6205#   6206#
6207#   6211#   6214#   6217#   6230#   6250#   6251#   6252#   6253#   6255#   6256#   6260#   6262#   6270#   6271#
6272#   6273#   6277#   6290#   6291#   6292#   6293#   6297#   6300#   6343#   6348#   6381#   6382#   6383#   6384#
6386#   6387#   6391#   6393#   6401#   6402#   6403#   6404#   6408#   6410#   6424#   6425#   6426#   6427#   6429#
6430#   6434#   6436#   6444#   6445#   6446#   6447#   6451#   6454#   6457#   6462#   6485#   6486#   6487#   6488#
6490#   6491#   6495#   6497#   6502#   6503#   6504#   6505#   6509#   6519#   6520#   6521#   6522#   6528#   6539#
6583#   6611#   6612#   6613#   6614#   6616#   6617#   6621#   6623#   6631#   6632#   6633#   6634#   6636#   6637#
6641#   6643#   6658#   6659#   6660#   6661#   6663#   6664#   6668#   6670#   6678#   6679#   6680#   6681#   6683#
6684#   6688#   6693#   6706#   6707#   6708#   6709#   6716#   6717#   6718#   6719#   6723#   6732#   6769#   6785#
6786#   6787#   6788#   6793#   6794#   6795#   6796#   6801#   6809#   6813#   6814#   6815#   6816#   6818#   6819#
6834#   6835#   6836#   6837#   6841#   6842#   6843#   6844#   6861#   6862#   6863#   6864#   6868#   6869#   6870#
6871#   6875#   6878#   6912#   6914#   6915#   6916#   6917#   6918#   6919    6932#   6933#   6934#   6935#   6939#
6944#   6950#   6951#   6960#   6961#   6962#   6963#   6965#   6966#   6976#   6977#   6978#   6979#   6983#   6985#
6986#   6991#   6992#   6993#   6994#   6998#   7042#   7051#   7052#   7062#   7063#   7064#   7065#   7067#   7068#
7072#   7073#   7074#   7075#   7077#   7078#   7080#   7081#   7082#   7083#   7084#   7085    7088#   7104#   7108#
7109#   7110#   7111#   7113#   7114#   7123#   7124#   7136#   7137#   7138#   7139#   7148#   7149#   7150#   7151#
7155#   7160#   7175#   7176#   7177#   7178#   7182#   7189#   7224#   7233#   7234#   7243#   7244#   7245#   7246#
7248#   7249#   7253#   7254#   7255#   7256#   7258#   7259#   7263#   7271#   7275#   7276#   7277#   7278#   7280#
7281#   7287#   7288#   7289#   7290#   7292#   7293#   7308#   7309#   7310#   7311#   7315#   7316#   7317#   7318#
7322#   7325#   7370#   7379#   7380#   7389#   7390#   7391#   7392#   7394#   7395#   7399#   7400#   7401#   7402#
7404#   7405#   7409#   7418#   7422#   7423#   7424#   7425#   7427#   7428#   7443#   7444#   7445#   7446#   7448#
7449#   7452#   7453#   7454#   7455#   7457#   7458#   7475#   7476#   7477#   7478#   7482#   7483#   7484#   7485#
7494#   7495#   7496#   7497#   7501#   7504#   7535#   7544#   7545#   7554#   7555#   7556#   7557#   7559#   7560#
7564#   7565#   7566#   7567#   7569#   7570#   7574#   7588#   7592#   7593#   7594#   7595#   7597#   7598#   7607#
7608#   7609#   7610#   7612#   7613#   7616#   7617#   7618#   7619#   7621#   7622#   7633#   7634#   7635#   7636#
7639#   7646#   7647#   7648#   7649#   7651#   7652#   7657#   7694#   7703#   7704#   7713#   7714#   7715#   7716#
7718#   7719#   7723#   7724#   7725#   7726#   7728#   7729#   7733#   7742#   7746#   7747#   7748#   7749#   7751#
7752#   7760#   7761#   7762#   7763#   7765#   7766#   7769#   7770#   7771#   7772#   7774#   7775#   7788#   7789#
7790#   7791#   7795#   7802#   7803#   7804#   7805#   7807#   7808#   7816#   7847#   7863#   7864#   7865#   7866#
7871#   7872#   7873#   7874#   7879#   7888#   7893#   7894#   7895#   7896#   7898#   7899#   7907#   7908#   7909#
7910#   7912#   7913#   7916#   7917#   7918#   7919#   7933#   7934#   7935#   7936#   7945#   7948#   7982#   7991#
7992#   8001#   8002#   8003#   8004#   8006#   8007#   8011#   8012#   8013#   8014#   8016#   8017#   8021#   8028#
8032#   8033#   8034#   8035#   8037#   8038#   8046#   8047#   8048#   8049#   8053#   8054#   8055#   8056#   8058#
8059#   8062#   8063#   8064#   8065#   8067#   8068#   8075#   8076#   8077#   8078#   8082#   8085#   8120#   8129#
8130#   8139#   8140#   8141#   8142#   8144#   8145#   8149#   8150#   8151#   8152#   8154#   8155#   8159#   8166#
8170#   8171#   8172#   8173#   8175#   8176#   8184#   8185#   8186#   8187#   8191#   8192#   8193#   8194#   8196#
8197#   8200#   8201#   8202#   8203#   8205#   8206#   8213#   8214#   8215#   8216#   8220#   8223#   8268#   8277#
```

```
   8278#    8287#    8288#    8289#    8290#    8292#    8293#    8297#    8298#    8299#    8300#    8302#    8303#    8307#    8316#
   8320#    8321#    8322#    8323#    8325#    8326#    8339#    8340#    8341#    8342#    8344#    8345#    8348#    8349#    8350#
   8351#    8353#    8354#    8366#    8367#    8368#    8369#    8376#    8377#    8378#    8379#    8383#    8386#    8430#    8439#
   8440#    8449#    8450#    8451#    8452#    8454#    8455#    8459#    8460#    8461#    8462#    8464#    8465#    8469#    8477#
   8481#    8482#    8483#    8484#    8486#    8487#    8500#    8501#    8502#    8503#    8505#    8506#    8509#    8510#    8511#
   8512#    8514#    8515#    8527#    8528#    8529#    8530#    8537#    8538#    8539#    8540#    8544#    8547#    8583#    8592#
   8593#    8600#    8603#    8604#    8605#    8607#    8608#    8612#    8613#    8614#    8615#    8617#    8618#    8622#    8635#
   8643#    8644#    8645#    8646#    8648#    8649#    8657#    8658#    8659#    8660#    8662#    8663#    8666#    8667#    8668#
   8669#    8671#    8672#    8684#    8685#    8686#    8687#    8694#    8695#    8696#    8697#    8701#    8706#    8760#    8769#
   8770#    8779#    8780#    8781#    8782#    8784#    8785#    8789#    8790#    8791#    8792#    8794#    8795#    8799#    8806#
   8810#    8811#    8812#    8813#    8815#    8816#    8828#    8829#    8830#    8831#    8833#    8834#    8837#    8838#    8839#
   8840#    8842#    8843#    8858#    8859#    8860#    8861#    8371#    8872#    8873#    8874#    8880#    8881#    8882#    8883#
   8892#    8893#    8894#    8895#    8901#    8902#    8903#    8904#    8910#    8911#    8912#    8913#    8920#    8921#    8922#
   8923#    8927#    8930#    8987#    8996#    8997#    9006#    9007#    9008#    9009#    9011#    9012#    9016#    9017#    9018#
   9019#    9021#    9022#    9026#    9038#    9046#    9047#    9048#    9049#    9051#    9052#    9060#    9061#    9062#    9063#
   9065#    9066#    9069#    9070#    9071#    9072#    9074#    9075#    9092#    9093#    9094#    9095#    9099#    9106#    9107#
   9108#    9109#    9111#    9112#    9122#    9156#    9165#    9166#    9175#    9176#    9177#    9178#    9180#    9181#    9185#
   9186#    9187#    9188#    9190#    9191#    9195#    9207#    9215#    9216#    9217#    9218#    9220#    9221#    9229#    9230#
   9231#    9232#    9234#    9235#    9238#    9239#    9240#    9241#    9243#    9244#    9262#    9263#    9264#    9265#    9269#
   9276#    9277#    9278#    9279#    9281#    9282#    9292#    9329#    9338#    9339#    9348#    9349#    9350#    9351#    9353#
   9354#    9358#    9359#    9360#    9361#    9363#    9364#    9368#    9375#    9379#    9380#    9381#    9382#    9384#    9385#
   9394#    9395#    9396#    9397#    9399#    9400#    9403#    9404#    9405#    9406#    9408#    9409#    9416#    9417#    9418#
   9419#    9426#    9427#    9428#    9429#    9433#    9436#    9494#    9503#    9504#    9513#    9514#    9515#    9516#    9518#
   9519#    9523#    9524#    9525#    9526#    9528#    9529#    9533#    9549#    9556#    9557#    9558#    9559#    9561#    9562#
   9570#    9571#    9572#    9573#    9575#    9576#    9579#    9580#    9581#    9582#    9584#    9585#    9599#    9600#    9601#
   9602#    9606#    9613#    9614#    9615#    9616#    9618#    9619#    9631#    9683#    9692#    9693#    9702#    9703#    9704#
   9705#    9707#    9708#    9712#    9713#    9714#    9715#    9717#    9718#    9722#    9734#    9742#    9743#    9744#    9745#
   9747#    9748#    9756#    9757#    9758#    9759#    9761#    9762#    9765#    9766#    9767#    9768#    9770#    9771#    9780#
   9781#    9782#    9783#    9787#    9794#    9795#    9796#    9797#    9799#    9800#    9814#    9846#    9855#    9856#    9865#
   9866#    9867#    9868#    9870#    9871#    9875#    9876#    9877#    9878#    9880#    9881#    9885#    9901#    9905#    9906#
   9907#    9908#    9910#    9911#    9919#    9920#    9921#    9922#    9924#    9925#    9928#    9929#    9930#    9931#    9933#
   9934#    9942#    9943#    9944#    9945#    9949#    9956#    9957#    9958#    9959#    9961#    9962#    9975#   10085#   10094#
  10095#   10104#   10105#   10106#   10107#   10109#   10110#   10114#   101[illegible]   10116#   10117#   10119#   10120#   10124#   10147#
  10156#   10157#   10158#   10159#   10161#   10162#   10170#   10171#   101[illegible]#   10173#   10175#   10176#   10179#   10180#   10181#
  10182#   10184#   10185#   10196#   10197#   10198#   10199#   10212#   10213#   10214#   10215#   10226#   10227#   10228#   10229#
  10233#   10236#   10308#   10317#   10318#   10327#   10328#   10329#   10330#   10332#   10333#   10337#   10338#   10339#   10340#
  10342#   10343#   10347#   10358#   10366#   10367#   10368#   10369#   10371#   10372#   10389#   10390#   10391#   10392#   10394#
  10395#   10397#   10398#   10399#   10400#   10408#   10409#   10410#   10411#   10415#   10425#   10463#   10472#   10473#   10482#
  10483#   10484#   10485#   10487#   10488#   10492#   10493#   10494#   10495#   10497#   10498#   10502#   10508#   10512#   10513#
  10514#   10515#   10517#   10518#   10528#   10529#   10530#   10531#   10533#   10534#   10537#   10538#   10539#   10540#   10542#
  10543#   10551#   10552#   10553#   10554#   10561#   10562#   10563#   [illegible]   10568#   10571#   10623#   10632#   10633#   10642#
  10643#   10644#   10645#   10647#   10648#   10652#   10653#   10654#   [illegible]   10657#   10658#   10662#   10674#   10678#   10679#
  10680#   10681#   10683#   10684#   10693#   10694#   10695#   10696#   [illegible]   10699#   10702#   10703#   10704#   10705#   10707#
  10708#   10715#   10716#   10717#   10718#   10722#   10727#   10731#   [illegible]   10733#   10734#   10736#   10737#   10741#   10750#
  10790#   10799#   10800#   10809#   10810#   10811#   10812#   10814#   10815    10819#   10820#   10821#   10822#   10824#   10825#
  10829#   10834#   10838#   10839#   10840#   10841#   10843#   10844#   10860#   10861#   10862#   10863#   10865#   10866#   10869#
  10870#   10871#   10872#   10874#   10875#   10883#   10884#   10885#   10886#   10893#   10894#   10895#   10896#   10900#   10903#
  10962#   10971#   10972#   10981#   10982#   10983#   10984#   10986#   10987#   10991#   10992#   10993#   10994#   10996#   10997#
  11001#   11010#   11017#   11018#   11019#   11020#   11022#   11023#   11036#   11037#   11038#   11039#   11041#   11042#   11045#
  11046#   11047#   11048#   11050#   11051#   11058#   11059#   11060#   11061#   11068#   11069#   11070#   11071#   11076#   11084#
  11130#   11139#   11140#   11149#   11150#   11151#   11152#   11154#   11155#   11159#   11160#   11161#   11162#   11164#   11165#
  11169#   11183#   11187#   11188#   11189#   11190#   11192#   11193#   11206#   11207#   11208#   11209#   11211#   11212#   11215#
  11216#   11217#   11218#   11220#   11221#   11231#   11232#   11233#   11234#   11239#   11240#   11241#   11242#   11249#   11250#
  11251#   11252#   11256#   11265#   11319#   11328#   11329#   11338#   11339#   11340#   11341#   11343#   11344#   11348#   11349#
  11350#   11351#   11353#   11354#   11358#   11370#   11380#   11390#   11391#   11392#   11393#   11395#   11396#   11404#   11405#
  11406#   11407#   11409#   11410#   11413#   11414#   11415#   11416#   11418#   11419#   11427#   11428#   11429#   11430#   11437#
```

```
           11438#  11439#  11440#  11444#  11457#  11467#  11527#  11536#  11537#  11546#  11547#  11548#  11549#  11551#  11552#
           11556#  11557#  11558#  11559#  11561#  11562#  11566#  11576#  11589#  11590#  11591#  11592#  11594#  11595#  11603#
           11604#  11605#  11606#  11608#  11609#  11612#  11613#  11614#  11615#  11617#  11618#  11629#  11630#  11631#  11632#
           11644#  11645#  11646#  11647#  11655#  11656#  11657#  11658#  11662#  11669#  11670#  11671#  11672#  11674#  11675#
           11687#  11742#  11751#  11752#  11761#  11762#  11763#  11764#  11766#  11767#  11771#  11772#  11773#  11774#  11776#
           11777#  11781#  11787#  11800#  11801#  11802#  11803#  11805#  11806#  11814#  11815#  11816#  11817#  11819#  11820#
           11823#  11824#  11825#  11826#  11828#  11829#  11837#  11838#  11839#  11840#  11852#  11853#  11854#  11855#  11863#
           11864#  11865#  11866#  11870#  11877#  11878#  11879#  11880#  11884#  11978#  11987#  11988#  11997#  11998#  11999#
           12000#  12002#  12003#  12007#  12008#  12009#  12010#  12012#  12013#  12017#  12033#  12041#  12042#  12043#  12044#
           12046#  12047#  12055#  12056#  12057#  12058#  12060#  12061#  12064#  12065#  12066#  12067#  12069#  12070#  12086#
           12087#  12088#  12089#  12097#  12098#  12099#  12100#  12104#  12109#  12118#  12119#  12120#  12121#  12123#  12124#
           12132#  12133#  12134#  12135#  12138#  12139#  12140#  12141#  12143#  12144#  12158#  12159#  12160#  12161#  12169#
           12170#  12171#  12172#  12178#  12183#  12190#  12191#  12192#  12193#  12195#  12196#  12204#  12205#  12206#  12207#
           12209#  12210#  12213#  12214#  12215#  12216#  12218#  12219#  12237#  12238#  12239#  12240#  12247#  12248#  12249#
           12250#  12254#  12257#  12321#  12330#  12331#  12340#  12341#  12342#  12343#  12345#  12346#  12350#  12351#  12352#
           12353#  12355#  12356#  12360#  12367#  12374#  12375#  12376#  12377#  12379#  12380#  12388#  12389#  12390#  12391#
           12393#  12394#  12397#  12398#  12399#  12400#  12402#  12403#  12411#  12412#  12413#  12414#  12421#  12422#  12423#
           12424#  12428#  12434#  12441#  12442#  12443#  12444#  12446#  12447#  12455#  12456#  12457#  12458#  12460#  12461#
           12464#  12465#  12466#  12467#  12469#  12470#  12479#  12480#  12481#  12482#  12484#  12485#  12492#  12493#  12494#
           12495#  12499#  12508#  12568#  12577#  12578#  12587#  12588#  12589#  12590#  12592#  12593#  12597#  12598#  12599#
           12600#  12602#  12603#  12607#  12622#  12626#  12627#  12628#  12629#  12631#  12632#  12640#  12641#  12642#  12643#
           12645#  12646#  12649#  12650#  12651#  12652#  12654#  12655#  12662#  12663#  12664#  12665#  12667#  12668#  12693#
           12694#  12695#  12696#  12698#  12699#  12705#  12708#  12732#  12733#  12738#  12739#  12754#  12755#  12756#  12757#
           12759#  12760#  12767#  12768#  12769#  12770#  12772#  12773#  12788#  12789#  12790#  12791#  12793#  12794#  12801#
           12802#  12803#  12804#  12806#  12807#  12832#  12833#  12834#  12835#  12837#  12838#  12845#  12846#  12847#  12848#
           12850#  12851#  12877#  12878#  12879#  12880#  12882#  12883#  12889#  12890#  12891#  12892#  12894#  12895#  12919#
           12920#  12921#  12922   12923#  12924   12926#  12927#  12928#  12929   12930#  12931   12933#  12934#  12935#  12936
           12937#  12938   12940#  12941#  12942#  12943   12944#  12945   12947#  12948#  12949#  12950   12951#  12952   12957#
           13081#  13084#  13085   13086   13087   13089#  13090   13091   13092   13094#
M$GNLS         1#   1389#   4983#   5000#   5086#   5103#   5120#   5137#   5154#   5258#   5338#   5360#   5391#   5415#   5440#
            5459#   5507#   5547#   5563#   5680#   6193#   6210#   6259#   6276#   6390#   6407#   6433#   6450#   6494#   6508#
            6620#   6640#   6667#   6687#   6722#   6800#   6874#   6938#   6982#   7087#   7154#   7181#   7262#   7321#   7408#
            7500#   7573#   7638#   7732#   7794#   7878#   7944#   8020#   8081#   8158#   8219#   8306#   8382#   8468#   8543#
            8621#   8700#   8798#   8926#   9025#   9098#   9194#   9268#   9367#   9432#   9532#   9605#   9721#   9786#   9884#
            9948#  10123#  10232#  10346#  10414#  10501#  10567#  10661#  10721#  10740#  10828#  10899#  11000#  11075#  11168#
           11255#  11357#  11443#  11456#  11565#  11661#  11780#  11869#  12016#  12103#  12177#  12253#  12359#  12427#  12498#
           12606#  12704#
M$GNSU         1#   1389#   4932#   4962#   5035#   5065#   5290#   5316#   5596#   5619#   5641#   6165#   6216#   6342#   6456#
M$GNTA         1#   1389#   1563#   2897#   2912#   2933#   2945#   2957#   2973#   2992#   3008#   3020#   3041#   3067#   3079#
            3091#   3104#   3116#   3141#   3154#   3174#   3188#   3203#   3231#   3259#   3273#   3302#   3329#   3356#   3391#
            3403#   3441#   3489#   3508#   3528#   3548#   3576#   3595#   3636#   3663#   3676#   3696#   3723#   3735#   3747#
            3759#   3771#   3798#   3818#   3860#   3872#   3911#   3935#   3947#   4649#   4761#   4773#   4813#   4823#   4834#
            4855#   4875#   4903#   4959#   5003#   5009#   5062#   5157#   5163#   5211#   5264#   5312#   5462#   5465#   5512#
            5566#   5616#   5638#   5683#   5686#   5769#   6213#   6296#   6299#   6453#   6527#   6538#   6731#   6877#   6997#
            7188#   7324#   7503#   7656#   7815#   7947#   8084#   8222#   8385#   8546#   8705#   8929#   9121#   9291#   9435#
            9630#   9813#   9974#  10235#  10424#  10570#  10749#  10902#  11083#  11264#  11466#  11686#  11883#  12256#  12507#
           12707#  12956#  13094#  13095
M$GNTE         1#   1389#   4923#   5026#   5178#   5226#   5285#   5487#   5531#   5594#   5707#   6163#   6339#   6578#   6762#
            6906#   7035#   7217#   7363#   7528#   7687#   7840#   7975#   8113#   8261#   8423#   8576#   8753#   8980#   9149#
            9322#   9487#   9676#   9839#  10078#  10301#  10456#  10616#  10783#  10955#  11123#  11312#  11520#  11735#  11971#
           12314#  12561#  12728#
M$HAPT         1#   1389#   1402#
M$HNAP         1#   1389#   1402#   144
M$INCR         1#   1389#   1393#   1555#   2888#   2894#   2898#   2901#   2909#   2913#   2916#   2922#   2930#   2934#   2937#
            2942#   2946#   2949#   2954#   2958#   2961#   2970#   2974#   2977#   2982#   2989#   2993#   2996#   3005#   3009#
            3012#   3017#   3021#   3024#   3029#   3038#   3042#   3045#   3050#   3057#   3064#   3068#   3071#   3076#   3080#
```

```
        3083#    3089#    3092#    3095#    3101#    3105#    3108#    3113#    3117#    3120#    3125#    3131#    3138#    3142#    3145#
        3151#    3155#    3158#    3164#    3171#    3175#    3178#    3185#    3189#    3192#    3200#    3204#    3207#    3213#    3220#
        3228#    3232#    3235#    3241#    3248#    3256#    3260#    3263#    3270#    3274#    3277#    3283#    3291#    3299#    3303#
        3306#    3311#    3318#    3326#    3330#    3333#    3338#    3345#    3353#    3357#    3360#    3367#    3377#    3387#    3392#
        3395#    3400#    3404#    3407#    3412#    3418#    3428#    3435#    3442#    3445#    3452#    3459#    3465#    3472#    3479#
        3486#    3490#    3493#    3498#    3505#    3509#    3512#    3517#    3525#    3529#    3532#    3537#    3545#    3549#    3552#
        3559#    3566#    3573#    3577#    3580#    3585#    3592#    3596#    3599#    3604#    3611#    3618#    3625#    3633#    3637#
        3640#    3645#    3653#    3660#    3664#    3667#    3673#    3677#    3680#    3686#    3693#    3697#    3700#    3706#    3712#
        3720#    3724#    3727#    3732#    3736#    3739#    3744#    3748#    3751#    3756#    3760#    3763#    3768#    3772#    3775#
        3780#    3787#    3795#    3799#    3802#    3807#    3815#    3819#    3822#    3828#    3835#    3842#    3849#    3857#    3861#
        3864#    3869#    3873#    3876#    3883#    3891#    3899#    3907#    3912#    3915#    3920#    3926#    3932#    3936#    3939#
        3944#    3948#    4201#    4215#    4244#    4357#    4391#    4475#    4485#    4507#    4573#    4588#    4601#    4616#    4629#
        4644#    4650#    4661#    4677#    4683#    4688#    4693#    4698#    4703#    4709#    4717#    4731#    4741#    4758#    4762#
        4771#    4774#    4783#    4795#    4800#    4814#    4818#    4824#    4829#    4835#    4852#    4871#    4897#    4923#    4924#
        4929#    4932#    4933#    4943#    4948#    4951#    4954#    4956#    4960#    4962#    4963#    4968#    4977#    4984#    4986#
        4994#    5001#    5004#    5007#    5010#    5026#    5027#    5032#    5035#    5036#    5046#    5051#    5054#    5057#    5059#
        5063#    5065#    5066#    5071#    5080#    5087#    5089#    5097#    5104#    5106#    5114#    5121#    5123#    5131#    5138#
        5140#    5148#    5155#    5158#    5161#    5164#    5178#    5179#    5184#    5192#    5197#    5200#    5203#    5205#    5208#
        5212#    5226#    5227#    5232#    5235#    5242#    5247#    5250#    5253#    5255#    5259#    5262#    5265#    5285#    5286#
        5290#    5291#    5303#    5308#    5313#    5316#    5317#    5322#    5332#    5339#    5348#    5354#    5361#    5374#    5385#
        5392#    5403#    5409#    5416#    5424#    5434#    5441#    5443#    5453#    5460#    5463#    5466#    5487#    5488#    5493#
        5501#    5508#    5513#    5531#    5532#    5535#    5541#    5548#    5551#    5557#    5564#    5567#    5594#    5595#    5596#
        5597#    5610#    5617#    5619#    5620#    5632#    5639#    5641#    5642#    5657#    5662#    5666#    5674#    5681#    5684#
        5687#    5707#    5708#    5711#    5716#    5728#    5733#    5750#    5755#    5763#    5770#    6163#    6164#    6165#    6166#
        6171#    6184#    6189#    6194#    6196#    6204#    6211#    6214#    6216#    6217#    6230#    6250#    6255#    6260#    6262#
        6270#    6277#    6290#    6297#    6300#    6339#    6340#    6342#    6343#    6348#    6381#    6386#    6391#    6393#    6401#
        6408#    6410#    6424#    6429#    6434#    6436#    6444#    6451#    6454#    6456#    6457#    6462#    6485#    6490#    6495#
        6497#    6502#    6509#    6519#    6528#    6539#    6578#    6579#    6583#    6611#    6616#    6621#    6623#    6631#    6636#
        6641#    6643#    6658#    6663#    6668#    6670#    6678#    6683#    6688#    6693#    6706#    6716#    6723#    6732#    6762#
        6763#    6769#    6785#    6793#    6801#    6809#    6813#    6818#    6834#    6841#    6861#    6868#    6875#    6878#    6906#
        6907#    6912#    6918#    6932#    6939#    6944#    6951#    6960#    6965#    6976#    6983#    6986#    6991#    6998#    7035#
        7036#    7042#    7051#    7062#    7067#    7072#    7077#    7084#    7088#    7104#    7108#    7113#    7124#    7136#    7148#
        7155#    7160#    7175#    7182#    7189#    7217#    7218#    7224#    7233#    7243#    7248#    7253#    7258#    7263#    7271#
        7275#    7280#    7287#    7292#    7308#    7315#    7322#    7325#    7363#    7364#    7370#    7379#    7389#    7394#    7399#
        7404#    7409#    7418#    7422#    7427#    7443#    7448#    7452#    7457#    7475#    7482#    7494#    7501#    7504#    7528#
        7529#    7535#    7544#    7554#    7559#    7564#    7569#    7574#    7588#    7592#    7597#    7607#    7612#    7616#    7621#
        7633#    7639#    7646#    7651#    7657#    7687#    7688#    7694#    7703#    7713#    7718#    7723#    7728#    7733#    7742#
        7746#    7751#    7760#    7765#    7769#    7774#    7788#    7795#    7802#    7807#    7816#    7840#    7841#    7847#    7863#
        7871#    7879#    7888#    7893#    7898#    7907#    7912#    7916#    7933#    7945#    7948#    7975#    7976#    7982#    7991#
        8001#    8006#    8011#    8016#    8021#    8028#    8032#    8037#    8046#    8053#    8058#    8062#    8067#    8075#    8082#
        8085#    8113#    8114#    8120#    8129#    8139#    8144#    8149#    8154#    8159#    8166#    8170#    8175#    8184#    8191#
        8196#    8200#    8205#    8213#    8220#    8223#    8261#    8262#    8268#    8277#    8287#    8292#    8297#    8302#    8307#
        8316#    8320#    8325#    8339#    8344#    8348#    8353#    8366#    8376#    8383#    8386#    8423#    8424#    8430#    8439#
        8449#    8454#    8459#    8464#    8469#    8477#    8481#    8486#    8500#    8505#    8509#    8514#    8527#    8537#    8544#
        8547#    8576#    8577#    8583#    8592#    8602#    8607#    8612#    8617#    8622#    8635#    8643#    8648#    8657#    8662#
        8666#    8671#    8684#    8694#    8701#    8706#    8753#    8754#    8760#    8769#    8779#    8784#    8789#    8794#    8799#
        8806#    8810#    8815#    8828#    8833#    8837#    8842#    8858#    8871#    8880#    8892#    8901#    8910#    8920#    8927#
        8930#    8980#    8981#    8987#    8996#    9006#    9011#    9016#    9021#    9026#    9038#    9046#    9051#    9060#    9065#
        9069#    9074#    9092#    9099#    9106#    9111#    9122#    9149#    9150#    9156#    9165#    9175#    9180#    9185#    9190#
        9195#    9207#    9215#    9220#    9229#    9234#    9238#    9243#    9262#    9269#    9276#    9281#    9292#    9322#    9323#
        9329#    9338#    9348#    9353#    9358#    9363#    9368#    9375#    9379#    9384#    9394#    9399#    9403#    9408#    9416#
        9426#    9433#    9436#    9487#    9488#    9494#    9503#    9513#    9518#    9523#    9528#    9533#    9549#    9556#    9561#
        9570#    9575#    9579#    9584#    9599#    9606#    9613#    9618#    9631#    9676#    9677#    9683#    9692#    9702#    9707#
        9712#    9717#    9722#    9734#    9742#    9747#    9756#    9761#    9765#    9770#    9780#    9787#    9794#    9799#    9814#
        9839#    9840#    9846#    9855#    9865#    9870#    9875#    9880#    9885#    9901#    9905#    9910#    9919#    9924#    9928#
        9933#    9942#    9949#    9956#    9961#    9975#   10078#   10079#   10085#   10094#   10104#   10109#   10114#   10119#   10124#
```

```
           10147#   10156#   10161#   10170#   10175#   10179#   10184#   10196#   10212#   10226#   10233#   10236#   10301#   10302#   10308#
           10317#   10327#   10332#   10337#   10342#   10347#   10358#   10366#   10371#   10389#   10394#   10397#   10408#   10415#   10425#
           10456#   10457#   10463#   10472#   10482#   10487#   10492#   10497#   10502#   10508#   10512#   10517#   10528#   10533#   10537#
           10542#   10551#   10561#   10568#   10571#   10616#   10617#   10623#   10632#   10642#   10647#   10652#   10657#   10662#   10674#
           10678#   10683#   10693#   10698#   10702#   10707#   10715#   10722#   10727#   10731#   10736#   10741#   10750#   10783#   10784#
           10790#   10799#   10809#   10814#   10819#   10824#   10829#   10834#   10838#   10843#   10860#   10865#   10869#   10874#   10883#
           10893#   10900#   10903#   10955#   10956#   10962#   10971#   10981#   10986#   10991#   10996#   11001#   11010#   11017#   11022#
           11036#   11041#   11045#   11050#   11058#   11068#   11076#   11084#   11123#   11124#   11130#   11139#   11149#   11154#   11159#
           11164#   11169#   11183#   11187#   11192#   11206#   11211#   11215#   11220#   11231#   11239#   11249#   11256#   11265#   11312#
           11313#   11319#   11328#   11338#   11343#   11348#   11353#   11358#   11370#   11380#   11390#   11395#   11404#   11409#   11413#
           11418#   11427#   11437#   11444#   11457#   11467#   11520#   11521#   11527#   11536#   11546#   11551#   11556#   11561#   11566#
           11576#   11589#   11594#   11603#   11608#   11612#   11617#   11629#   11644#   11655#   11662#   11669#   11674#   11687#   11735#
           11736#   11742#   11751#   11761#   11766#   11771#   11776#   11781#   11787#   11800#   11805#   11814#   11819#   11823#   11828#
           11837#   11852#   11863#   11870#   11877#   11884#   11971#   11972#   11978#   11987#   11997#   12002#   12007#   12012#   12017#
           12033#   12041#   12046#   12055#   12060#   12064#   12069#   12086#   12097#   12104#   12109#   12118#   12123#   12132#   12138#
           12143#   12158#   12169#   12178#   12183#   12190#   12195#   12204#   12209#   12213#   12218#   12237#   12247#   12254#   12257#
           12314#   12315#   12321#   12330#   12340#   12345#   12350#   12355#   12360#   12367#   12374#   12379#   12388#   12393#   12397#
           12402#   12411#   12421#   12428#   12434#   12441#   12446#   12455#   12460#   12464#   12469#   12479#   12484#   12492#   12499#
           12508#   12561#   12562#   12568#   12577#   12587#   12592#   12597#   12602#   12607#   12622#   12626#   12631#   12640#   12645#
           12649#   12654#   12662#   12667#   12693#   12698#   12705#   12708#   12728#   12729#   12732#   12738#   12754#   12759#   12767#
           12772#   12788#   12793#   12801#   12806#   12832#   12837#   12845#   12850#   12877#   12882#   12889#   12894#   12923#   12930#
           12937#   12944#   12951#   12957#   13081#
M$IOSE         1#    1389#
M$LDRO         1#    1389#    4200#    4214#    4682#    4687#    4692#    4697#    4708#    4740#    4950#    4953#    5006#    5053#    5056#
            5160#    5199#    5202#    5207#    5249#    5252#    5261#    6950#    6985#    7123#
M$MASK         1#    1389#
M$MCHI         1#    1389#
M$MCLO         1#    1389#
M$MSK1         1#    1389#
M$POP          1#    1389#    1563#    2897#    2912#    2933#    2945#    2957#    2973#    2992#    3008#    3020#    3041#    3067#    3079#
            3091#    3104#    3116#    3141#    3154#    3174#    3188#    3203#    3231#    3259#    3273#    3302#    3329#    3356#    3391#
            3403#    3441#    3489#    3508#    3528#    3548#    3576#    3595#    3636#    3663#    3676#    3696#    3723#    3735#    3747#
            3759#    3771#    3798#    3818#    3860#    3872#    3911#    3935#    3947#    4649#    4668#    4761#    4773#    4813#    4823#
            4834#    4855#    4875#    4903#    4959#    4983#    5000#    5003#    5009# [illegible]    5086#    5103#    5120#    5137#    5154#
            5157#    5163#    5211#    5258#    5264#    5312#    5338#    5360#    5391#    5415#    5440#    5459#    5462#    5465#    5507#
            5512#    5547#    5563#    5566#    5616#    5638#    5680#    5683#    5686#    5769#    6193#    6210#    6213#    6259#    6276#
            6296#    6299#    6390#    6407#    6433#    6450#    6453#    6494#    6508#    6527#    6538#    6620#    6640#    6667#    6687#
            6722#    6731#    6800#    6874#    6877#    6938#    6982#    6997#    7087#    7154#    7181#    7188#    7262#    7321#    7324#
            7408#    7500#    7503#    7573#    7638#    7656#    7732#    7794#    7815#    7878# [illegible]    7947#    8020#    8081#    8084#
            8158#    8219#    8222#    8306#    8382#    8385#    8468#    8543#    8546#    8621# [illegible]    8705#    8798#    8926#    8929#
            9025#    9098#    9121#    9194#    9268#    9291#    9367#    9432#    9435#    9532#    9605#    9630#    9721#    9786#    9813#
            9884#    9948#    9974#   10123#   10232#   10235#   10346#   10414#   10424#   10501#   10567#   10570#   10661#   10721#   10740#
           10749#   10828#   10899#   10902#   11000#   11075#   11083#   11168#   11255#   11264#   11357#   11443#   11456#   11466#   11565#
           11661#   11686#   11780#   11869#   11883#   12016#   12103#   12177#   12253#   12256#   12359#   12427#   12498#   12507#   12606#
           12704#   12707#   12956#   13094#   13121#
M$PRIN         1#    1389#    2890#    2903#    2918#    2925#    2939#    2951#    2963#    2979#    2985#    2998#    3014#    3026#    3032#
            3047#    3053#    3060#    3073#    3085#    3097#    3110#    3122#    3128#    3134#    3147#    3160#    3167#    3180#    3194#
            3209#    3216#    3223#    3237#    3244#    3251#    3265#    3279#    3286#    3294#    3308#    3314#    3321#    3335#    3341#
            3348#    3363#    3371#    3381#    3397#    3409#    3415#    3424#    3431#    3449#    3456#    3462#    3468#    3475#    3482#
            3495#    3501#    3514#    3520#    3534#    3540#    3554#    3562#    3569#    3582#    3588#    3601#    3607#    3614#    3621#
            3628#    3642#    3648#    3656#    3669#    3682#    3689#    3702#    3709#    3715#    3729#    3741#    3753#    3765#    3777#
     [illegible]    3790#    3804#    3810#    3824#    3831#    3838#    3845#    3852#    3866#    3880#    3888#    3896#    3904#    3917#
            3923#    3929#    3941#    4353#    4468#    4482#    4503#    4569#    4584#    4597#    4612#    4625#    4714#    4792#   12919#
           12926#   12933#   12940#   12947#
M$PUSH         1#    1389#    1393#    1555#    2888#    2901#    2916#    2937#    2949#    2961#    2977#    2996#    3012#    3024#    3045#
            3071#    3083#    3095#    3108#    3120#    3145#    3158#    3178#    3192#    3207#    3235#    3263#    3277#    3306#    3333#
```

```
          3360#   3395#   3407#   3445#   3493#   3512#   3532#   3552#   3580#   3599#   3640#   3667#   3680#   3700#   3727#
          3739#   3751#   3763#   3775#   3802#   3822#   3864#   3876#   3915#   3939#   4644#   4661#   4677#   4771#   4783#
          4818#   4829#   4852#   4871#   4897#   4923#   4924    4932#   4933    4962#   4963    4968#   4986#   5026#   5027
          5035#   5036    5065#   5066    5071#   5089#   5106#   5123#   5140#   5178#   5179    5226#   5227    5235#   5285#
          5286    5290#   5291    5316#   5317    5322#   5348#   5374#   5403#   5424#   5443#   5487#   5488    5493#   5531#
          5532    5535#   5551#   5594#   5595    5596#   5597    5619#   5620    5641#   5642    5666#   5707#   5708    6163#
          6164    6165#   6166    6171#   6196#   6216#   6217    6230#   6262#   6339#   6340    6342#   6343    6348#   6393#
          6410#   6436#   6456#   6457    6462#   6497#   6578#   6579    6583#   6623#   6643#   6670#   6693#   6762#   6763
          6769#   6879#   6906#   6907    6912#   6944#   7035#   7036    7042#   7104#   7160#   7217#   7218    7224#   7271#
          7363#   7364    7370#   7478#   7528#   7529    7535#   7588#   7687#   7688    7694#   7742#   7840#   7841    7847#
          7888#   7975#   7976    7982#   8028#   8113#   8114    8120#   8166#   8261#   8262    8268#   8316#   8423#   8424
          8430#   8477#   8576#   8577    8583#   8635#   8753#   8754    8760#   8806#   8980#   8981    8987#   9038#   9149#
          9150    9156#   9207#   9322#   9323    9329#   9375#   9487#   9488    9494#   9549#   9676#   9677    9683#   9734#
          9839#   9840    9846#   9901#  10078#  10079   10085#  10147#  10301#  10302   10308#  10358#  10456#  10457   10463#
         10508#  10616#  10617   10623#  10674#  10727#  10783#  10784   10790#  10834#  10955#  10956   10962#  11010#  11123#
         11124   11130#  11183#  11312#  11313   11319#  11370#  11380#  11520#  11521   11527#  11576#  11735#  11736   11742#
         11787#  11971#  11972   11978#  12033#  12109#  12183#  12314#  12315   12321#  12367#  12434#  12561#  12562   12568#
         12622#  12728#  12729   13081#
M$PUT        1#   1389#   2890#   2903#   2918#   2925#   2939#   2951#   2963#   2979#   2985#   2998#   3014#   3026#   3032#
          3047#   3053#   3060#   3073#   3085#   3097#   3110#   3122#   3128#   3134#   3147#   3160#   3167#   3180#   3194#
          3209#   3216#   3223#   3237#   3244#   3251#   3265#   3279#   3286#   3294#   3308#   3314#   3321#   3335#   3341#
          3348#   3363#   3371#   3381#   3397#   3409#   3415#   3424#   3431#   3449#   3456#   3462#   3468#   3475#   3482#
          3495#   3501#   3514#   3520#   3534#   3540#   3554#   3562#   3569#   3582#   3588#   3601#   3607#   3614#   3621#
          3628#   3642#   3648#   3656#   3669#   3682#   3689#   3702#   3709#   3715#   3729#   3741#   3753#   3765#   3777#
          3783#   3790#   3804#   3810#   3824#   3831#   3838#   3845#   3852#   3866#   3880#   3888#   3896#   3904#   3917#
          3923#   3929#   3941#   4353#   4468#   4482#   4503#   4569#   4584#   4597#   4612#   4625#   4714#   4727#   4792#
          4925#   5028#   5180#   5228#   6914#   7080#  12919#  12926#  12933#  12940#  12947#
M$PUT1       1#   1389#   2890#   2891    2892    2903#   2904    2905    2906    2907    2918#   2919    2920    2925#   2926
          2927    2928    2939#   2940    2951#   2952    2963#   2964    2965    2966    2967    2968    2979#   2980    2985#
          2986    2987    2998#   2999    3000    3001    3002    3003    3014#   3015    3026#   3027    3032#   3033    3034
          3035    3036    3047#   3048    3053#   3054    3055    3060#   3061    3062    3073#   3074    3085#   3086    3097#
          3098    3099    3110#   3111    3122#   3123    3128#   3129    3134#   3135    3136    3147#   3148    3149    3160#
          3161    3162    3167#   3168    3169    3180#   3181    3182    3183    3194#   3195    3196    3197    3198    3209#
          3210    3211    3216#   3217    3218    3223#   3224    3225    3226    3237#   3238    3239    3244#   3245    3246
          3251#   3252    3253    3254    3265#   3266    3267    3268    3279#   3280    3281    3286#   3287    3288    3289
          3294#   3295    3296    3297    3308#   3309    3314#   3315    3316    3321#   3322    3323    3324    3335#   3336
          3341#   3342    3343    3348#   3349    3350    3351    3363#   3364    3365    3371#   3372    3373    3374    3375
          3381#   3382    3383    3384    3385    3397#   3398    3409#   3410    3415#   3416    3424#   3425    3426    3431#
          3432    3433    3449#   3450    3456#   3457    3462#   3463    3468#   3469    3470    3475#   3476    3477    3482#
          3483    3484    3495#   3496    3501#   3502    3503    3514#   3515    3520#   3521    3522    3523    3534#   3535
          3540#   3541    3542    3543    3554#   3555    3556    3557    3562#   3563    3564    3569#   3570    3571    3582#
          3583    3588#   3589    3590    3601#   3602    3607#   3608    3609    3614#   3615    3616    3621#   3622    3623
          3628#   3629    3630    3631    3642#   3643    3648#   3649    3650    3651    3656#   3657    3658    3669#   3670
          3671    3682#   3683    3684    3689#   3690    3691    3702#   3703    3704    3709#   3710    3715#   3716    3717
          3718    3729#   3730    3741#   3742    3753#   3754    3765#   3766    3777#   3778    3783#   3784    3785    3790#
          3791    3792    3793    3804#   3805    3810#   3811    3812    3813    3824#   3825    3826    3831#   3832    3833
          3838#   3839    3840    3845#   3846    3847    3852#   3853    3854    3855    3866#   3867    3880#   3881    3888#
          3889    3896#   3897    3904#   3905    3917#   3918    3923#   3924    3929#   3930    3941#   3942    4353#   4354
          4355    4468#   4469    4470    4471    4472    4473    4482#   4483    4503#   4504    4505    4569#   4570    4571
          4584#   4585    4586    4597#   4598    4599    4612#   4613    4614    4625#   4626    4627    4714#   4715    4727#
          4728    4729    4730    4792#   4793    4925#   4926    4927    4928    5028#   5029    5030    5031    5180#   5181
          5182    5183    5228#   5229    5230    5231    6914#   6915    6916    6917    7080#   7081    7082    7083   12919#
         12920   12921   12926#  12927   12928   12933#  12934   12935   12940#  12941   12942   12947#  12948   12949
M$RADI       1#   1389#  13084#  13089#
M$RBRO       1#   1389#
M$RNRO       1#   1389#   4703#   4704    4708#   4710    4740#   4742
```

```
M$SETS        1#   1389#   1393#   1555#   2888#   2901#   2916#   2937#   2949#   2961#   2977#   2996#   3012#   3024#   3045#
           3071#   3083#   3095#   3108#   3120#   3145#   3158#   3178#   3192#   3207#   3235#   3263#   3277#   3306#   3333#
           3360#   3395#   3407#   3445#   3493#   3512#   3532#   3552#   3580#   3599#   3640#   3667#   3680#   3700#   3727#
           3739#   3751#   3763#   3775#   3802#   3822#   3864#   3876#   3915#   3939#   4644#   4661#   4677#   4771#   4783#
           4818#   4829#   4852#   4871#   4897#   4924#   4933#   4963#   4968#   4986#   5027#   5036#   5066#   5071#   5089#
           5106#   5123#   5140#   5179#   5227#   5235#   5286#   5291#   5317#   5322#   5348#   5374#   5403#   5424#   5443#
           5488#   5493#   5532#   5535#   5551#   5595#   5597#   5620#   5642#   5666#   5708#   6164#   6166#   6171#   6196#
           6217#   6230#   6262#   6340#   6343#   6348#   6393#   6410#   6436#   6457#   6462#   6497#   6579#   6583#   6623#
           6643#   6670#   6693#   6763#   6769#   6809#   6907#   6912#   6944#   7036#   7042#   7104#   7160#   7218#   7224#
           7271#   7364#   7370#   7418#   7529#   7535#   7588#   7688#   7694#   7742#   7841#   7847#   7888#   7976#   7982#
           8028#   8114#   8120#   8166#   8262#   8268#   8316#   8424#   8430#   8477#   8577#   8583#   8635#   8754#   8760#
           8806#   8981#   8987#   9038#   9150#   9156#   9207#   9323#   9329#   9375#   9488#   9494#   9549#   9677#   9683#
           9734#   9840#   9846#   9901#  10079#  10085#  10147#  10302#  10308#  10358#  10457#  10463#  10508#  10617#  10623#
          10674#  10727#  10784#  10790#  10834#  10956#  10962#  11010#  11124#  11130#  11183#  11313#  11319#  11370#  11380#
          11521#  11527#  11576#  11736#  11742#  11787#  11972#  11978#  12033#  12109#  12183#  12315#  12321#  12367#  12434#
          12562#  12568#  12622#  12729#  13081#
M$STAR        1#   1389#
M$SVC         1#   1389#   2890#   2894    2897#   2898    2903#   2909    2912#   2913    2918#   2922    2925#   2930    2933#
           2934    2939#   2942    2945#   2946    2951#   2954    2957#   2958    2963#   2970    2973#   2974    2979#   2982
           2985#   2989    2992#   2993    2998#   3005    3008#   3009    3014#   3017    3020#   3021    3026#   3029    3032#
           3038    3041#   3042    3047#   3050    3053#   3057    3060#   3064    3067#   3068    3073#   3076    3079#   3080
           3085#   3088    3091#   3092    3097#   3101    3104#   3105    3110#   3113    3116#   3117    3122#   3125    3128#
           3131    3134#   3138    3141#   3142    3147#   3151    3154#   3155    3160#   3164    3167#   3171    3174#   3175
           3180#   3185    3188#   3189    3194#   3200    3203#   3204    3209#   3213    3216#   3220    3223#   3228    3231#
           3232    3237#   3241    3244#   3248    3251#   3256    3259#   3260    3265#   3270    3273#   3274    3279#   3283
           3286#   3291    3294#   3299    3302#   3303    3308#   3311    3314#   3318    3321#   3326    3329#   3330    3335#
           3338    3341#   3345    3348#   3353    3356#   3357    3363#   3367    3371#   3377    3381#   3387    3391#   3392
           3397#   3400    3403#   3404    3409#   3412    3415#   3418    3424#   3428    3431#   3435    3441#   3442    3449#
           3452    3456#   3459    3462#   3465    3468#   3472    3475#   3479    3482#   3486    3489#   3490    3495#   3498
           3501#   3505    3508#   3509    3514#   3517    3520#   3525    3528#   3529    3534#   3537    3540#   3545    3548#
           3549    3554#   3559    3562#   3566    3569#   3573    3576#   3577    3582#   3585    3588#   3592    3595#   3596
           3601#   3604    3607#   3611    3614#   3618    3621#   3625    3628#   3633    3636#   3637    3642#   3645    3648#
           3653    3656#   3660    3663#   3664    3669#   3673    3676#   3677    3682#   3686    3689#   3693    3696#   3697
           3702#   3706    3709#   3712    3715#   3720    3723#   3724    3729#   3732    3735#   3736    3741#   3744    3747#
           3748    3753#   3756    3759#   3760    3765#   3768    3771#   3772    3777#   3780    3783#   3787    3790#   3795
           3798#   3799    3804#   3807    3810#   3815    3818#   3819    3824#   3828    3831#   3835    3838#   3842    3845#
           3849    3852#   3857    3860#   3861    3866#   3869    3872#   3873    3880#   3883    3888#   3891    3896#   3899
           3904#   3907    3911#   3912    3917#   3920    3923#   3926    3929#   3932    3935#   3936    3941#   3944    3947#
           3948    4200#   4201    4214#   4215    4244#   4353#   4357    4391#   4468#   4475    4482#   4485    4503#   4507
           4569#   4573    4584#   4588    4597#   4601    4612#   4616    4625#   4629    4646#   4649#   4650    4682#   4683
           4687#   4688    4692#   4693    4697#   4698    4703#   4708#   4709    4714#   4717    4727#   4731    4740#   4741
           4758#   4761#   4762    4773#   4774    4792#   4795    4800#   4813#   4814    4820#   4823#   4824    4831#   4834#
           4835    4925#   4929    4932#   4933    4943    4948#   4950#   4951    4953#   4954    4956#   4959#   4960    4962#
           4963    4968#   4977    4983#   4984    4986#   4994    5000#   5001    5003#   5004    5006#   5007    5009#   5010
           5028#   5032    5035#   5036    5046    5051#   5053#   5054    5056#   5057    5059#   5062#   5063    5065#   5066
           5071#   5080    5086#   5087    5089#   5097    5103#   5104    5106#   5114    5120#   5121    5123#   5131    5137#
           5138    5140#   5148    5154#   5155    5157#   5158    5160#   5161    5163#   5164    5180#   5184    5192    5197#
           5199#   5200    5202#   5203    5205#   5207#   5208    5211#   5212    5228#   5232    5235#   5242    5247#   5249#
           5250    5252#   5253    5255#   5258#   5259    5261#   5262    5264#   5265    5290#   5291    5303    5308#   5312#
           5313    5316#   5317    5322#   5332    5338#   5339    5348#   5354    5360#   5361    5374#   5385    5391#   5392
           5403#   5409    5415#   5416    5424#   5434    5440#   5441    5443#   5453    5459#   5460    5462#   5463    5465#
           5466    5493#   5501    5507#   5508    5512#   5513    5535#   5541    5547#   5548    5551#   5557    5563#   5564
           5566#   5567    5596#   5597    5610    5616#   5617    5619#   5620    5632    5638#   5639    5641#   5642    5657
           5662#   5666#   5674    5680#   5681    5683#   5684    5686#   5687    5711    5716#   5728    5733#   5750    5755#
           5763    5769#   5770    6165#   6166    6171#   6184    6189#   6193#   6194    6196#   6204    6210#   6211    6213#
           6214    6216#   6217    6230#   6250    6255#   6259#   6260    6262#   6270    6276#   6277    6290    6296#   6297
```

```
 6299#   6300    6342#   6343    6348#   6381    6386#   6390#   6391    6393#   6401    6407#   6408    6410#   6424
 6429#   6433#   6434    6436#   6444    6450#   6451    6453#   6454    6456#   6457    6462#   6485    6490#   6494#
 6495    6497#   6502    6508#   6509    6519    6527#   6528    6538#   6539    6583#   6611    6616#   6620#   6621
 6623#   6631    6636#   6640#   6641    6643#   6658    6663#   6667#   6668    6670#   6678    6683#   6687#   6688
 6693#   6706    6716    6722#   6723    6731#   6732    6769#   6785    6793    6800#   6801    6809#   6813    6818#
 6834    6841    6861    6868    6874#   6875    6877#   6878    6912#   6914#   6918    6932    6938#   6939    6944#
 6950#   6951    6960    6965#   6976    6982#   6983    6985#   6986    6991    6997#   6998    7042#   7051#   7062
 7067#   7072    7077#   7080#   7084    7087#   7088    7104#   7108    7113#   7123#   7124    7136    7148    7154#
 7155    7160#   7175    7181#   7182    7138#   7189    7224#   7233#   7243    7248#   7253    7258#   7262#   7263
 7271#   7275    7280#   7287    7292#   7308    7315    7321#   7322    7324#   7325    7370#   7379#   7389    7394#
 7399    7404#   7408#   7409    7418#   7422    7427#   7443    7448#   7452    7457#   7475    7482    7494    7500#
 7501    7503#   7504    7535#   7544#   7554    7559#   7564    7569#   7573#   7574    7588#   7592    7597#   7607
 7612#   7616    7621#   7633    7638#   7639    7646    7651#   7656#   7657    7694#   7703#   7713    7718#   7723
 7728#   7732#   7733    7742#   7746    7751#   7760    7765#   7769    7774#   7738    7794#   7795    7802    7807#
 7815#   7816    7847#   7863    7871    7878#   7879    7888#   7893    7898#   7907    7912#   7916    7933    7944#
 7945    7947#   7948    7982#   7991#   8001    8006#   8011    8016#   8020#   8021    8028#   8032    8037#   8046
 8053    8058#   8062    8067#   8075    8081#   8082    8084#   8085    8120#   8129#   8139    8144#   8149    8154#
 8158#   8159    8166#   8170    8175#   8184    8191    8196#   8200    8205#   8213    8219#   8220    8222#   8223
 8268#   8277#   8287    8292#   8297    8302#   8306#   8307    8316#   8320    8325#   8339    8344#   8348    8353#
 8366    8376    8382#   8383    8385#   8386    8430#   8439#   8449    8454#   8459    8464#   8468#   8469    8477#
 8481    8486#   8500    8505#   8509    8514    8527    8537    8543#   8544    8546#   8547    8583#   8592#   8602
 8607#   8612    8617#   8621#   8622    8635#   8643    8648#   8657    8662#   8666    8671#   8684    8694    8700#
 8701    8705#   8706    8760#   8769#   8779    8784#   8789    8794#   8798#   8799    8806#   8810    8815#   8828
 8833#   8837    8842#   8858    8871    8880    8892    8901    8910    8920    8926#   8927    8929#   8930    8987#
 8996#   9006    9011#   9016    9021#   9025#   9026    9038#   9046    9051#   9060    9065#   9069    9074#   9092
 9098#   9099    9106    9111#   9121#   9122    9156#   9165#   9175    9180#   9185    9190#   9194#   9195    9207#
 9215    9220#   9229    9234#   9238    9243#   9262    9268#   9269    9276    9281#   9291#   9292    9329#   9338#
 9348    9353#   9358    9363#   9367#   9368    9375#   9379    9384#   9394    9399#   9403    9408#   9416    9426
 9432#   9433    9435#   9436    9494#   9503#   9513    9518#   9523    9528#   9532#   9535    9549#   9556    9561#
 9570    9575#   9579    9584#   9599    9605#   9606    9613    9618#   9630#   9631    9683#   9692#   9702    9707#
 9712    9717#   9721#   9722    9734#   9742    9747#   9756    9761#   9765    9770#   9780    9786#   9787    9794
 9799#   9813#   9814    9846#   9855#   9865    9870#   9875    9880#   9884#   9885    9901#   9905    9910#   9919
 9924#   9928    9933#   9942    9948#   9949    9956    9961#   9974#   9975   10085#  10094#  10104   10109#  10114
10119#  10123#  10124   10147#  10156   10161#  10170   10175#  10179   10184#  10196   10212   10226   10232#  10233
10235#  10236   10308#  10317#  10327   10332#  10337   10342#  10346#  10347   10358#  10366   10371#  10389   10394#
10397   10408   10414#  10415   10424#  10425   10463#  10472#  10482   10487#  10492   10497#  10501#  10502   10508#
10512   10517#  10528   10533#  10537   10542#  10551   10561   10567#  10568   10570#  10571   10623#  10632#  10642
10647#  10652   10657#  10661#  10662   10674#  10678   10683#  10693   10698#  10702   10707#  10715   10721#  10722
10727#  10731   10736#  10740#  10741   10749#  10750   10790#  10799#  10809   10814#  10819   10824#  10828#  10829
10834#  10838   10843#  10860   10865#  10869   10874#  10883   10893   10899#  10900   10902#  10903   10962#  10971#
10981   10986#  10991   10996#  11000#  11001   11010#  11017   11022#  11036   11041#  11045   11050#  11058   11068
11075#  11076   11080#  11084   11130#  11139#  11140   11154#  11159   11164#  11168#  11169   11183#  11187   11192#
11206   11211#  11215   11220#  11231   11239   11249   11255#  11256   11264#  11265   11319#  11328#  11338   11343#
11348   11353#  11357#  11358   11370#  11380#  11390   11395#  11404   11409#  11413   11418#  11427   11437   11443#
11444   11456#  11457   11466#  11467   11527#  11536#  11546   11551#  11556   11561#  11565#  11566   11576#  11589
11594#  11603   11608#  11612   11617#  11629   11644   11655   11661#  11662   11669   11674#  11686#  11687   11742#
11751#  11761   11766#  11771   11776#  11780#  11781   11787#  11800   11805#  11814   11819#  11823   11828#  11837
11852   11863   11869#  11870   11877   11883#  11884   11978#  11987#  11997   12002#  12007   12012#  12016#  12017
12033#  12041   12046#  12055   12060#  12064   12069#  12086   12097   12103#  12104   12109#  12118   12123#  12132
12138   12143#  12158   12169   12177#  12178   12183#  12190   12195#  12204   12209#  12213   12218#  12237   12247
12253#  12254   12256#  12257   12321#  12330#  12340   12345#  12350   12355#  12359#  12360   12367#  12374   12379#
12388   12393#  12397   12402#  12411   12421   12427#  12428   12434#  12441   12446#  12455   12460#  12464   12469#
12479   12484#  12492   12498#  12499   12507#  12508   12568#  12577#  12587   12592#  12597   12602#  12606#  12607
12622#  12626   12631#  12640   12645#  12649   12654#  12662   12667#  12693   12698#  12704#  12705   12707#  12708
12732#  12738#  12754   12759#  12767   12772#  12788   12793#  12801   12806#  12832   12837#  12845   12850#  12877
12882#  12889   12894#  12919#  12923   12926#  12930   12933#  12937   12940#  12944   12947#  12951   12956#  12957
```

```
M$TLAB        1#    1389#    2894#    2898#    2909#    2915#    2922#    2930#    2934#    2942#    2946#    2954#    2958#    2970#    2974#
           2982#    2989#    2993#    3005#    3009#    3017#    3021#    3029#    3038#    3042#    3050#    3057#    3064#    3068#    3076#
           3080#    3088#    3092#    3101#    3105#    3113#    3117#    3125#    3131#    3138#    3142#    3151#    3155#    3164#    3171#
           3175#    3185#    3189#    3200#    3204#    3213#    3220#    3228#    3232#    3241#    3248#    3256#    3260#    3270#    3274#
           3283#    3291#    3299#    3303#    3311#    3318#    3326#    3330#    3338#    3345#    3353#    3357#    3367#    3377#    3387#
           3392#    3400#    3404#    3412#    3418#    3428#    3435#    3442#    3452#    3459#    3465#    3472#    3479#    3486#    3490#
           3498#    3505#    3509#    3517#    3525#    3529#    3537#    3545#    3549#    3559#    3566#    3573#    3577#    3585#    3592#
           3596#    3604#    3611#    3618#    3625#    3633#    3637#    3645#    3653#    3660#    3664#    3673#    3677#    3686#    3693#
           3697#    3706#    3712#    3720#    3724#    3732#    3736#    3744#    3748#    3756#    3760#    3768#    3772#    3780#    3787#
           3795#    3799#    3807#    3815#    3819#    3828#    3835#    3842#    3849#    3857#    3861#    3869#    3873#    3883#    3891#
           3899#    3907#    3912#    3920#    3926#    3932#    3936#    3944#    3948#    4201#    4215#    4244#    4357#    4391#    4475#
           4485#    4507#    4573#    4588#    4601#    4616#    4629#    4650#    4683#    4688#    4693#    4698#    4703#    4709#    4717#
           4731#    4741#    4758#    4762#    4774#    4795#    4800#    4814#    4824#    4835#    4929#    4933#    4943#    4948#    4951#
           4954#    4956#    4960#    4963#    4968#    4977#    4984#    4986#    4994#    5001#    5004#    5007#    5010#    5032#    5036#
           5046#    5051#    5054#    5057#    5059#    5063#    5066#    5071#    5080#    5087#    5089#    5097#    5104#    5106#    5114#
           5121#    5123#    5131#    5138#    5140#    5148#    5155#    5158#    5161#    5164#    5184#    5192#    5197#    5200#    5203#
           5205#    5208#    5212#    5232#    5235#    5242#    5247#    5250#    5253#    5255#    5259#    5262#    5265#    5291#    5303#
           5308#    5313#    5317#    5322#    5332#    5339#    5348#    5354#    5361#    5374#    5385#    5392#    5403#    5409#    5416#
           5424#    5434#    5441#    5443#    5453#    5460#    5463#    5466#    5493#    5501#    5508#    5513#    5535#    5541#    5548#
           5551#    5557#    5564#    5567#    5597#    5610#    5617#    5620#    5632#    5639#    5642#    5657#    5662#    5666#    5674#
           5681#    5684#    5687#    5711#    5716#    5728#    5733#    5750#    5755#    5763#    5770#    6166#    6171#    6184#    6189#
           6194#    6196#    6204#    6211#    6214#    6217#    6230#    6250#    6255#    6260#    6262#    6270#    6277#    6290#    6297#
           6300#    6343#    6348#    6381#    6386#    6391#    6393#    6401#    6408#    6410#    6424#    6429#    6434#    6436#    6444#
           6451#    6454#    6457#    6462#    6485#    6490#    6495#    6497#    6502#    6509#    6519#    6528#    6539#    6583#    6611#
           6616#    6621#    6623#    6631#    6636#    6641#    6643#    6658#    6663#    6668#    6670#    6678#    6683#    6688#    6693#
           6706#    6716#    6723#    6732#    6769#    6785#    6793#    6801#    6809#    6813#    6810#    6834#    6841#    6861#    6868#
           6875#    6878#    6912#    6918#    6932#    6939#    6944#    6951#    6960#    69[illegible]5#    6976#    6983#    6986#    6991#    6998#
           7042#    7051#    7062#    7067#    7072#    7077#    7084#    7088#    7104#    7108#    7113#    7124#    7136#    7148#    7155#
           7160#    7175#    7182#    7189#    7224#    7233#    7243#    7248#    7253#    7258#    7263#    7271#    7275#    7280#    7287#
           7292#    7308#    7315#    7322#    7325#    7370#    7379#    7389#    7394#    7399#    7404#    7409#    7418#    7422#    7427#
           7443#    7448#    7452#    7457#    7475#    7482#    7494#    7501#    7504#    7535#    7544#    7554#    7559#    7564#    7569#
           7574#    7588#    7592#    7597#    7607#    7612#    7616#    7621#    7633#    7639#    7646#    7651#    7657#    7694#    7703#
           7713#    7718#    7723#    7728#    7733#    7742#    7746#    7751#    7760#    7765#    7769#    7774#    7788#    7795#    7802#
           7807#    7816#    7847#    7863#    7871#    7879#    7888#    7893#    7898#    7907#    7912#    7916#    7933#    7945#    7948#
           7982#    7991#    8001#    8006#    8011#    8016#    8021#    8028#    8032#    8037#    8046#    8053#    8058#    8062#    8067#
           8075#    8082#    8085#    8120#    8129#    8139#    8144#    8149#    8154#    8159#    8166#    8170#    8175#    8184#    8191#
           8196#    8200#    8205#    8213#    8220#    8223#    8268#    8277#    8287#    8292#    8297#    8302#    8307#    8316#    8320#
           8325#    8339#    8344#    8348#    8353#    8366#    8376#    8383#    8386#    8430#    8439#    8449#    8454#    8459#    8464#
           8469#    8477#    848[illegible]#    8486#    8500#    8505#    8509#    8514#    8527#    8537#    8544#    8547#    8583#    8592#    8602#
           8607#    8612#    8617#    8622#    8635#    8643#    8648#    8657#    8662#    8666#    8671#    8684#    8694#    8701#    8706#
           8760#    8769#    8779#    8784#    8789#    8794#    8799#    8806#    8810#    8815#    8828#    8833#    8837#    8842#    8858#
           8871#    8880#    8892#    8901#    8910#    8920#    8927#    8930#    8987#    8996#    9006#    9011#    9016#    9021#    9026#
           9038#    9046#    9051#    9060#    9065#    9069#    9074#    9092#    9099#    9106#    9111#    9122#    9156#    9165#    9175#
           9180#    9185#    9190#    9195#    9207#    9215#    9220#    9229#    9234#    9238#    9243#    9262#    9269#    9276#    9281#
           9292#    9329#    9338#    9348#    9353#    9358#    9363#    9368#    9375#    9379#    9384#    9394#    9399#    9403#    9408#
           9416#    9426#    9433#    9436#    9494#    9503#    9513#    9518#    9523#    9528#    9533#    9549#    9556#    9561#    9570#
           9575#    9579#    9584#    9599#    9606#    9613#    9618#    9631#    9683#    9692#    9702#    9707#    9712#    9717#    9722#
           9734#    9742#    9747#    9756#    9761#    9765#    9770#    9780#    9787#    9794#    9799#    9814#    9846#    9855#    9865#
           9870#    9875#    9880#    9885#    9901#    9905#    9910#    9919#    9924#    9928#    9933#    9942#    9949#    9956#    9961#
           9975#   10085#   10094#   10104#   10109#   10114#   10119#   10124#   10147#   10156#   10161#   10170#   10175#   10179#   10184#
          10196#   10212#   10226#   10233#   10236#   10308#   10317#   10327#   10332#   10337#   10342#   10347#   10358#   10366#   10371#
          10389#   10394#   10397#   10408#   10415#   10425#   10463#   10472#   10482#   10487#   10492#   10497#   10502#   10508#   10512#
          10517#   10528#   10533#   10537#   10542#   10551#   10561#   10568#   10571#   10623#   10632#   10642#   10647#   10652#   10657#
          10662#   10674#   10678#   10683#   10693#   10698#   10702#   10707#   10715#   10722#   10727#   10731#   10736#   10741#   10750#
          10790#   10799#   10809#   10814#   10819#   10824#   10829#   10834#   10838#   10843#   10860#   10865#   10869#   10874#   10883#
          10893#   10900#   10903#   10962#   10971#   10981#   10986#   10991#   10996#   1100[illegible]#   11010#   11017#   11022#   11036#   11041#
```

```
         11045#  11050#  11058#  11068#  11076#  11084#  11130#  11139#  11149#  11154#  11159#  11164#  11169#  11183#  11187#
         11192#  11206#  11211#  11215#  11220#  11231#  11239#  11249#  11256#  11265#  11319#  11328#  11338#  11343#  11348#
         11353#  11358#  11370#  11380#  11390#  11395#  11404#  11409#  11413#  11418#  11427#  11437#  11444#  11457#  11467#
         11527#  11536#  11546#  11551#  11556#  11561#  11566#  11576#  11589#  11594#  11603#  11608#  11612#  11617#  11629#
         11644#  11655#  11662#  11669#  11674#  11687#  11742#  11751#  11761#  11766#  11771#  11776#  11781#  11787#  11800#
         11805#  11814#  11819#  11823#  11828#  11837#  11850#  11862#  11870#  11877#  11884#  11978#  11987#  11997#  12002#
         12007#  12012#  12017#  12033#  12041#  12046#  12055#  12060#  12064#  12069#  12086#  12097#  12104#  12109#  12118#
         12123#  12132#  12138#  12143#  12158#  12169#  12178#  12183#  12190#  12195#  12204#  12209#  12213#  12218#  12237#
         12247#  12254#  12257#  12321#  12330#  12340#  12345#  12350#  12355#  12360#  12367#  12374#  12379#  12388#  12393#
         12397#  12402#  12411#  12421#  12428#  12434#  12441#  12446#  12455#  12460#  12464#  12469#  12479#  12484#  12492#
         12499#  12508#  12568#  12577#  12587#  12592#  12597#  12602#  12607#  12622#  12626#  12631#  12640#  12645#  12649#
         12654#  12662#  12667#  12693#  12696   12705#  12708#  12732#  12738#  12754#  12759#  12767#  12772#  12788#  12793#
         12801#  12806#  12832#  12837#  12845#  12850#  12877#  12882#  12389#  12894#  12923#  12930#  12937#  12944#  12951#
         12957#
M$TSTL       1#   1389#   2894#   2898#   2909#   2913#   2922#   2930#   2934#   2942#   2946#   2954#   2958#   2970#   2974#
          2982#   2989#   2993#   3005#   3009#   3017#   3021#   3029#   3033#   3042#   3050#   3057#   3064#   3068#   3076#
          3080#   3088#   3092#   3101#   3105#   3113#   3117#   3125#   3131#   3138#   3142#   3151#   3155#   3164#   3171#
          3175#   3185#   3189#   3200#   3204#   3213#   3220#   3228#   3232#   3241#   3248#   3256#   3260#   3270#   3274#
          3283#   3291#   3299#   3303#   3311#   3318#   3326#   3330#   3338#   3345#   3353#   3357#   3367#   3377#   3387#
          3392#   3400#   3404#   3412#   3418#   3428#   3435#   3442#   3452#   3459#   3465#   3472#   3479#   3486#   3490#
          3498#   3505#   3509#   3517#   3525#   3529#   3537#   3545#   3549#   3559#   3566#   3573#   3577#   3585#   3592#
          3596#   3604#   3611#   3618#   3625#   3633#   3637#   3645#   3653#   3660#   3664#   3673#   3677#   3686#   3693#
          3697#   3706#   3712#   3720#   3724#   3732#   3736#   3744#   3748#   3756#   3760#   3768#   3772#   3780#   3787#
          3795#   3799#   3807#   3815#   3819#   3828#   3835#   3842#   3849#   3857#   3861#   3869#   3873#   3883#   3891#
          3899#   3907#   3912#   3920#   3926#   3932#   3936#   3944#   3948#   4201#   4215#   4244#   4357#   4391#   4475#
          4485#   4507#   4573#   4588#   4601#   4616#   4629#   4650#   4683#   4688#   4693#   4698#   4703#   4709#   4717#
          4731#   4741#   4758#   4762#   4774#   4795#   4800#   4814#   4824#   4835#   4929#   4933#   4943#   4948#   4951#
          4954#   4956#   4960#   4963#   4968#   4977#   4984#   4986#   4994#   5001#   5004#   5007#   5010#   5032#   5036#
          5046#   5051#   5054#   5057#   5059#   5063#   5066#   5071#   5080#   5087#   5089#   5097#   5104#   5106#   5114#
          5121#   5123#   5131#   5138#   5140#   5148#   5155#   5158#   5161#   5164#   5184#   5192#   5197#   5200#   5203#
          5205#   5208#   5212#   5232#   5235#   5242#   5247#   5250#   5253#   5255#   5259#   5262#   5265#   5291#   5303#
          5308#   5313#   5317#   5322#   5332#   5339#   5348#   5354#   5361#   5374#   5385#   5392#   5403#   5409#   5416#
          5424#   5434#   5441#   5443#   5453#   5460#   5463#   5466#   5493#   5501#   5508#   5513#   5535#   5541#   5548#
          5551#   5557#   5564#   5567#   5597#   5610#   5617#   5620#   5632#   5639#   5642#   5657#   5662#   5666#   5674#
          5681#   5684#   5687#   5711#   5716#   5728#   5733#   5750#   5755#   5763#   5770#   6166#   6171#   6184#   6189#
          6194#   6196#   6204#   6211#   6214#   6217#   6230#   6250#   6255#   6260#   6262#   6270#   6277#   6290#   6297#
          6300#   6343#   6348#   6381#   6386#   6391#   6393#   6401#   6408#   6410#   6424#   6429#   6434#   6436#   6444#
          6451#   6454#   6457#   6462#   6485#   6490#   6495#   6497#   6502#   6509#   6519#   6528    6539#   6583#   6611#
          6616#   6621#   6623#   6631#   6636#   6641#   6643#   6658#   6663#   6668#   6670#   6678#   6683#   6688#   6693#
          6706#   6716#   6723#   6732#   6769#   6785#   6793#   6801#   6809#   6813#   6818#   6834#   6841#   6861#   6868#
          6875#   6878#   6912#   6918#   6932#   6939#   6944#   6951#   6960#   6965#   6976#   6983#   6986#   6991#   6998#
          7042#   7051#   7062#   7067#   7072#   7077#   7084#   7088#   7104#   7108#   7113#   7124#   7136#   7148#   7155#
          7160#   7175#   7182#   7189#   7224#   7233#   7243#   7248#   7253#   7258#   7263#   7271#   7275#   7280#   7287#
          7292#   7308#   7315#   7322#   7325#   7370#   7379#   7389#   7394#   7399#   7404#   7409#   7418#   7422#   7427#
          7443#   7448#   7452#   7457#   7475#   7482#   7494#   7501#   7504#   7535#   7544#   7554#   7559#   7564#   7569#
          7574#   7588#   7592#   7597#   7607#   7612#   7616#   7621#   7633#   7639#   7646#   7651#   7657#   7694#   7703#
          7713#   7718#   7723#   7728#   7733#   7742#   7746#   7751#   7760#   7765#   7769#   7774#   7788#   7795#   7802#
          7807#   7816#   7847#   7863#   7871#   7879#   7888#   7893#   7898#   7907#   7912#   7916#   7933#   7945#   7948#
          7982#   7991#   8001#   8006#   8011#   8016#   8021#   8028#   8032#   8037#   8046#   8053#   8058#   8062#   8067#
          8075#   8082#   8085#   8120#   8129#   8139#   8144#   8149#   8154#   8159#   8166#   8170#   8175#   8184#   8191#
          8196#   8200#   8205#   8213#   8220#   8223#   8258#   8277#   8287#   8292#   8297#   8302#   8307#   8316#   8320#
          8325#   8339#   8344#   8348#   8353#   8366#   8376#   8383#   8386#   8430#   8439#   8449#   8454#   8459#   8464#
          8469#   8477#   8481#   8486#   8500#   8505#   8509#   8514#   8527#   8537#   8544#   8547#   8583#   8592#   8602#
          8607#   8612#   8617#   8622#   8635#   8643#   8648#   8657#   8662#   8666#   8671#   8684#   8694#   8701#   8706#
          8760#   8769#   8779#   8784#   8789#   8794#   8799#   8806#   8810#   8815#   8828#   8833#   8837#   8842#   8858#
          8871#   8880#   8892#   8901#   8910#   8920#   8927#   8930#   8987#   8996#   9006#   9011#   9016#   9021#   9026#
```

```
             9038#   9046#   9051#   9060#   9065#   9069#   9074#   9092#   9099#   9106#   9111#   9122#   9156#   9165#   9175#
             9180#   9185#   9190#   9195#   9207#   9215#   9220#   9229#   9234#   9238#   9243#   9262#   9269#   9276#   9281#
             9292#   9329#   9338#   9348#   9353#   9358#   9363#   9368#   9375#   9379#   9384#   9394#   9399#   9403#   9408#
             9416#   9426#   9433#   9436#   9494#   9503#   9513#   9518#   9523#   9528#   9533#   9549#   9556#   9561#   9570#
             9575#   9579#   9584#   9599#   9606#   9613#   9618#   9631#   9683#   9692#   9702#   9707#   9712#   9717#   9722#
             9734#   9742#   9747#   9756#   9761#   9765#   9770#   9780#   9787#   9794#   9799#   9814#   9846#   9855#   9865#
             9870#   9875#   9880#   9885#   9901#   9905#   9910#   9919#   9924#   9928#   9933#   9942#   9949#   9956#   9961#
             9975#  10085#  10094#  10104#  10109#  10114#  10119#  10124#  10147#  10156#  10161#  10170#  10175#  10179#  10184#
            10196#  10212#  10226#  10233#  10236#  10308#  10317#  10327#  10332#  10337#  10342#  10347#  10358#  10366#  10371#
            10389#  10394#  10397#  10408#  10415#  10425#  10463#  10472#  10482#  10487#  10492#  10497#  10502#  10508#  10512#
            10517#  10528#  10533#  10537#  10542#  10551#  10561#  10568#  10571#  10623#  10632#  10642#  10647#  10652#  10657#
            10662#  10674#  10678#  10683#  10693#  10698#  10702#  10707#  10715#  10722#  10727#  10731#  10736#  10741#  10750#
            10790#  10799#  10809#  10814#  10819#  10824#  10829#  10834#  10838#  10843#  10860#  10865#  10869#  10874#  10883#
            10893#  10900#  10903#  10962#  10971#  10981#  10986#  10991#  10996#  11001#  11010#  11017#  11022#  11036#  11041#
            11045#  11050#  11058#  11068#  11076#  11084#  11130#  11139#  11149#  11154#  11159#  11164#  11169#  11183#  11187#
            11192#  11206#  11211#  11215#  11220#  11231#  11239#  11249#  11256#  11265#  11319#  11328#  11338#  11343#  11348#
            11353#  11358#  11370#  11380#  11390#  11395#  11404#  11409#  11413#  11418#  11427#  11437#  11444#  11457#  11467#
            11527#  11536#  11546#  11551#  11556#  11561#  11566#  11576#  11589#  11594#  11603#  11608#  11612#  11617#  11629#
            11644#  11655#  11662#  11669#  11674#  11687#  11742#  11751#  11761#  11766#  11771#  11776#  11781#  11787#  11800#
            11805#  11814#  11819#  11823#  11828#  11837#  11852#  11863#  11870#  11877#  11884#  11978#  11987#  11997#  12002#
            12007#  12012#  12017#  12033#  12041#  12046#  12055#  12060#  12064#  12069#  12086#  12097#  12104#  12109#  12118#
            12123#  12132#  12138#  12143#  12158#  12169#  12178#  12183#  12190#  12195#  12204#  12209#  12213#  12218#  12237#
            12247#  12254#  12257#  12321#  12330#  12340#  12345#  12350#  12355#  12360#  12367#  12374#  12379#  12388#  12393#
            12397#  12402#  12411#  12421#  12428#  12434#  12441#  12446#  12455#  12460#  12464#  12469#  12479#  12484#  12492#
            12499#  12508#  12568#  12577#  12587#  12592#  12597#  12602#  12607#  12622#  12626#  12631#  12640#  12645#  12649#
            12654#  12662#  12667#  12693#  12698#  12705#  12708#  12732#  12738#  12754#  12759#  12767#  12772#  12788#  12793#
            12801#  12806#  12832#  12837#  12845#  12850#  12877#  12882#  12889#  12894#  12923#  12930#  12937#  12944#  12951#
            12957#
M$WORD          1#   1389#   1441#   1450    1496#   1498    1499    1500    1501    1502    1503    1504    1505    1506    1507
             1508    1509    1510    1511    1512    1513    1514    1515    1516    1517    1518    1519    1520    1521    1522
             1523    1524    1525    1526    1527    1528    1529    1530    1531    1532    1533    1534    1535    1536    1537
             1538    1539    1540    1541    1542    1543    4646#   4800#   4820#   4831#   4943#   4944    4945    4946    4977#
             4978    4979    4980    4994#   4995    4996    4997    5046#   5047    5048    5049    5080#   5081    5082    5083
             5097#   5098    5099    5100    5114#   5115    5116    5117    5131#   5132    5133    5134    5148#   5149    5150
             5151    5192#   5193    5194    5195    5242#   5243    5244    5245    5303#   5304    5305    5306    5332#   5333
             5334    5335    5354#   5355    5356    5357    5385#   5386    5387    5388    5409#   5410    5411    5412    5434#
             5435    5436    5437    5453#   5454    5455    5456    5501#   5502    5503    5504    5541#   5542    5543    5544
             5557#   5558    5559    5560    5610#   5611    5612    5613    5632#   5633    5634    5635    5657#   5658    5659
             5660    5674#   5675    5676    5677    5711#   5712    5713    5714    5728#   5729    5730    5731    5750#   5751
             5752    5753    5763#   5764    5765    5766    6184#   6185    6186    6187    6204#   6205    6206    6207    6250#
             6251    6252    6253    6270#   6271    6272    6273    6290#   6291    6292    6293    6381#   6382    6383    6384
             6401#   6402    6403    6404    6424#   6425    6426    6427    6444#   6445    6446    6447    6485#   6486    6487
             6488    6502#   6503    6504    6505    6519#   6520    6521    6522    6611#   6612    6613    6614    6631#   6632
             6633    6634    6658#   6659    6660    6661    6678#   6679    6680    6681    6706#   6707    6708    6709    6716#
             6717    6718    6719    6785#   6786    6787    6788    6793#   6794    6795    6796    6813#   6814    6815    6816
             6834#   6835    6836    6837    6841#   6842    6843    6844    6861#   6862    6863    6864    6868#   6869    6870
             6871    6932#   6933    6934    6935    6960#   6961    6962    6963    6976#   6977    6978    6979    6991#   6992
             6993    6994    7062#   7063    7064    7065    7072#   7073    7074    7075    7108#   7109    7110    7111    7136#
             7137    7138    7139    7148#   7149    7150    7151    7175#   7176    7177    7178    7243#   7244    7245    7246
             7253#   7254    7255    7256    7275#   7276    7277    7278    7287#   7288    7289    7290    7308#   7309    7310
             7311    7315#   7316    7317    7318    7389#   7390    7391    7392    7399#   7400    7401    7402    7422#   7423
             7424    7425    7443#   7444    7445    7446    7452#   7453    7454    7455    7475#   7476    7477    7478    7482#
             7483    7484    7485    7494#   7495    7496    7497    7554#   7555    7556    7557    7564#   7565    7566    7567
             7592#   7593    7594    7595    7607#   7608    7609    7610    7616#   7617    7618    7619    7633#   7634    7635
             7636    7646#   7647    7648    7649    7713#   7714    7715    7716    7723#   7724    7725    7726    7746#   7747
             7748    7749    7760#   7761    7762    7763    7769#   7770    7771    7772    7788#   7789    7790    7791    7802#
```

```
       7803    7804    7805    7863#   7864    7865    7866    7871#   7872    7873    7874    7893#   7894    7895    7896
       7907#   7908    7909    7910    7916#   7917    7918    7919    7933#   7934    7935    7936    8001#   8002    8003
       8004    8011#   8012    8013    8014    8032#   8033    8034    8035    8046#   8047    8048    8049    8053#   8054
       8055    8056    8062#   8063    8064    8065    8075#   8076    8077    8078    8139#   8140    8141    8142    8149#
       8150    8151    8152    8170#   8171    8172    8173    8184#   8185    8186    8187    8191#   8192    8193    8194
       8200#   8201    8202    8203    8213#   8214    8215    8216    8287#   8288    8289    8290    8297#   8298    8299
       8300    8320#   8321    8322    8323    8339#   8340    8341    8342    8348#   8349    8350    8351    8366#   8367
       8368    8369    8376#   8377    8378    8379    8449#   8450    8451    8452    8459#   8460    8461    8462    8481#
       8482    8483    8484    8500#   8501    8502    8503    8509#   8510    8511    8512    8527#   8528    8529    8530
       8537#   8538    8539    8540    8602#   8603    8604    8605    8612#   8613    8614    8615    8643#   8644    8645
       8646    6657#   8658    9659    8660    8666#   8667    8668    8669    8684#   8695    8686    8687    8694#   8695
       8696    8697    8779#   8780    8781    8782    8789#   8790    8791    8792    8810#   8811    8812    8813    8828#
       8829    8830    8831    8837#   8838    8839    8840    8858#   8859    8860    8861    8871#   8872    8873    8874
       8880#   8881    8882    8883    8892#   8893    8894    8895    8901#   8902    8903    8904    8910#   8911    8912
       8913    8920#   8921    8922    8923    9006#   9007    9008    9009    9016#   9017    9018    9019    9046#   9047
       9048    9049    9060#   9061    9062    9063    9069#   9070    9071    9072    9092#   9093    9094    9095    9106#
       9107    9108    9109    9175#   9176    9177    9178    9185#   9186    9187    9188    9215#   9216    9217    9218
       9229#   9230    9231    9232    9238#   9239    9240    9241    9262#   9263    9264    9265    9276#   9277    9278
       9279    9348#   9349    9350    9351    9358#   9359    9360    9361    9379#   9380    9381    9382    9394#   9395
       9396    9397    9403#   9404    9405    9406    9416#   9417    9418    9419    9426#   9427    9428    9429    9513#
       9514    9515    9516    9523#   9524    9525    9526    9556#   9557    9558    9559    9570#   9571    9572    9573
       9579#   9580    9581    9582    9599#   9600    9601    9602    9613#   9614    9615    9616    9702#   9703    9704
       9705    9712#   9713    9714    9715    9742#   9743    9744    9745    9756#   9757    9758    9759    9765#   9766
       9767    9768    9780#   9781    9782    9783    9794#   9795    9796    9797    9865#   9866    9867    9868    9875#
       9876    9877    9878    9905#   9906    9907    9908    9919#   9920    9921    9922    9928#   9929    9930    9931
       9942#   9943    9944    9945    9956#   9957    9958    9959   10104#  10105   10106   10107   10114#  10115   10116
      10117   10156#  10157   10158   10159   10170#  10171   10172   10173   10179#  10180   10181   10182   10196#  10197
      10198   10199   10212#  10213   10214   10215   10226#  10227   10228   10229   10327#  10328   10329   10330   10337#
      10338   10339   10340   10366#  10367   10368   10369   10389#  10390   10391   10392   10397#  10398   10399   10400
      10408#  10409   10410   10411   10482#  10483   10484   10485   10492#  10493   10494   10495   10512#  10513   10514
      10515   10528#  10529   10530   10531   10537#  10538   10539   10540   10551#  10552   10553   10554   10561#  10562
      10563   10564   10642#  10643   10644   10645   10652#  10653   10654   10655   10678#  10679   10680   10681   10693#
      10694   10695   10696   10702#  10703   10704   10705   10715#  10716   10717   10718   10731#  10732   10733   10734
      10809#  10810   10811   10812   10819#  10820   10821   10822   10838#  10839   10840   10841   10860#  10861   10862
      10863   10869#  10870   10871   10872   10883#  10884   10885   10886   10893#  10894   10895   10896   10981#  10982
      10983   10984   10991#  10992   10993   10994   11017#  11018   11019   11020   11036#  11037   11038   11039   11045#
      11046   11047   11048   11058#  11059   11060   11061   11068#  11069   11070   11071   11149#  11150   11151   11152
      11159#  11160   11161   11162   11187#  11188   11189   11190   11206#  11207   11208   11209   11215#  11216   11217
      11218   11231#  11232   11233   11234   11239#  11240   11241   11242   11249#  11250   11251   11252   11338#  11339
      11340   11341   11348#  11349   11350   11351   11390#  11391   11392   11393   11404#  11405   11406   11407   11413#
      11414   11415   11416   11427#  11428   11429   11430   11437#  11438   11439   11440   11546#  11547   11548   11549
      11556#  11557   11558   11559   11589#  11590   11591   11592   11603#  11604   11605   11606   11612#  11613   11614
      11615   11629#  11630   11631   11632   11644#  11645   11646   11647   11655#  11656   11657   11658   11669#  11670
      11671   11672   11761#  11762   11763   11764   11771#  11772   11773   11774   11800#  11801   11802   11803   11814#
      11815   11816   11817   11823#  11824   11825   11826   11837#  11838   11839   11840   11852#  11853   11854   11855
      11863#  11864   11865   11866   11877#  11878   11879   11880   11997#  11998   11999   12000   12007#  12008   12009
      12010   12041#  12042   12043   12044   12055#  12056   12057   12058   12064#  12065   12066   12067   12086#  12087
      12088   12089   12097#  12098   12099   12100   12118#  12119   12120   12121   12132#  12133   12134   12135   12138#
      12139   12140   12141   12158#  12159   12160   12161   12169#  12170   12171   12172   12190#  12191   12192   12193
      12204#  12205   12206   12207   12213#  12214   12215   12216   12237#  12238   12239   12240   12247#  12248   12249
      12250   12340#  12341   12342   12343   12350#  12351   12352   12353   12374#  12375   12376   12377   12388#  12389
      12390   12391   12397#  12398   12399   12400   12411#  12412   12413   12414   12421#  12422   12423   12424   12441#
      12442   12443   12444   12455#  12456   12457   12458   12464#  12465   12466   12467   12479#  12480   12481   12482
      12492#  12493   12494   12495   12587#  12588   12589   12590   12597#  12598   12599   12600   12626#  12627   12628
      12629   12640#  12641   12642   12643   12649#  12650   12651   12652   12662#  12663   12664   12665   12693#  12694
      12695   12696   12732#  12754#  12755   12756   12757   12767#  12768   12769   12770   12788#  12789   12790   12791
```

```
          12801#   12802    12803    12804    12832#   12833    12834    12835    12845#   12846    12847    12848    12877#   12878    12879
          12880    12889#   12890    12891    12892    13084#   13089#
M$XFER        1#    1389#
OPEN          1#    1389#
POINTE        1#    1389#    1399
PRINTB        1#    1389#    2889     2902     2917     2924     2938     2950     2962     2978     2984     2997     3013     3025     3031
           3046     3052     3059     3072     3084     3096     3109     3121     3127     3133     3146     3159     3166     3179     3193
           3208     3215     3222     3236     3243     3250     3264     3278     3285     3293     3307     3313     3320     3334     3340
           3347     3362     3370     3380     3396     3408     3414     3423     3430     3448     3455     3461     3467     3474     3481
           3494     3500     3513     3519     3533     3539     3553     3561     3568     3581     3587     3600     3606     3613     3620
           3627     3641     3647     3655     3668     3681     3688     3701     3708     3714     3728     3740     3752     3764     3776
           3782     3789     3803     3809     3823     3830     3837     3844     3851     3865     3879     3887     3895     3903     3916
           3922     3928     3940    12918    12925    12932    12939    12946
PRINTF        1#    1389#    4352     4467     4481     4502     4568     4583     4596     4611     4624     4713     4791
PRINTS        1#    1389#
PRINTX        1#    1389#
READBU        1#    1389#
READEF        1#    1389#    4681     4686     4691     4696
RFLAGS        1#    1389#
SETPRI        1#    1389#    4199     4213     6949     7122
SETVEC        1#    1389#    4726     4924     5027     5179     5227     6913     7079
SLASH         1#    1389#
STARS         1#    1389#
SVC           1#    1389#
XFER          1#    1389#    4646#    4800#    4820#    4831#   12732#
XFERF         1#    1389#
XFERT         1#    1389#


. ABS.  000000      000
        000000      001
UNAREP  053774      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

 CZUAAA.OBJ,CZUAAA.LST/CRF/SOL/NL:TOC=SVC34R.P11,CZUAAA.P11
 RUN-TIME: 77 91 11 SECONDS'
 RUN-TIME RATIO: 361/181=1.9
 CORE USED:  33K  (66 PAGES)
```

```

                                        .TITLE  MICROA - MICROCODE MODULE A

        000000'                         .CSECT MICRA

                                        .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000                          IPCSR0   =        21000   ;INTERNAL PCSR0 ADDRESS
        021002                          DMACSR   =        21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004                          DMAT0    =        21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006                          DMAT1    =        21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010                          MDMA0    =        21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012                          MDMA1    =        21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014                          MDMAR0   =        21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016                          MDMAR1   =        21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020                          IPCSR1   =        21020   ;INTERNAL PCSR1 ADDRESS
        021022                          DMAF     =        21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024                          DMAWC    =        21024   ;DMA ENGINE WORD COUNT REGISTER
        021026                          MDMAW0   =        21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030                          LTAC     =        21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032                          LFRBUF   =        21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034                          CLRFIF   =        21034   ;CLEAR FIFO
        021036                          MDMAW1   =        21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040                          PCSRSW   =        21040   ;SWITCH PACK REGISTER
        021042                          MDMSR    =        21042   ;MICROCPU DMA STATUS REGISTER
        021044                          LRBUF    =        21044   ;LINK RECIEVE BUFFER COMPLETED
        021060                          PHYAD0   =        21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062                          PHYAD1   =        21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064                          PHYAD2   =        21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066                          PHYAD3   =        21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070                          PHYAD4   =        21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072                          PHYAD5   =        21072   ;PHYSICAL ADDRESS ROM BYTE 5

                                        .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000                          BIT15    =        100000
        040000                          BIT14    =        40000
        020000                          BIT13    =        20000
        010000                          BIT12    =        10000
        004000                          BIT11    =        4000
        002000                          BIT10    =        2000
        001000                          BIT9     =        1000
        000400                          BIT8     =        400
        000200                          BIT7     =        200
        000100                          BIT6     =        100
        000040                          BIT5     =        40
        000020                          BIT4     =        20
        000010                          BIT3     =        10
        000004                          BIT2     =        4
        000002                          BIT1     =        2
        000001                          BIT0     =        1
                                        ;
        012400                          LASFTP   =        BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340                          PRI07    =        340
        000300                          PRI06    =        300
        000240                          PRI05    =        240
```

```
000200          PRI04   =       200
000140          PRI03   =       140
000100          PRI02   =       100
000040          PRI01   =       40
000000          PRI00   =       0
                ;
                ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                ;
100000          SERI    =       BIT15
040000          PCEI    =       BIT14
020000          RXI     =       BIT13
010000          TXI     =       BIT12
004000          DNI     =       BIT11
002000          RCEI    =       BIT10
000400          FATI    =       BIT8
                ;
000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
000150          NXMVEC= 150             ;VECTOR ADDRESS FOR THE NON-EXISTANT MEMORY TIMOUT
000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
000150          NXMVEC= 150             ;VECTOR ADDRESS FOR NON-EXISTANT UNIBUS ADDRESS
001000          STACK=  1000            ;STACK LOCATION
000001          INMON=  1               ;IN MICROMONITOR STATE
000002          INTST=  2               ;IN A TEST STATE
000003          INERR=  3               ;IN ERROR STATE
000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
020000          SIZ4K=  20000           ;4K WORDS
040000          SIZ8K=  SIZ4K*2         ;8K WORDS
020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
077774          LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
000000          WCSADR= 0               ;BASE ADDRESS OF WCS
020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
000002          PARERR= 2               ;FLAG INDICATING PARITY ERROR OCCURRED
177774          MODREG= LINADR+LINSIZ   ;LINK MODE REGISTER
177774          ADDREG= MODREG          ;LINK STATION ADDRESS RAM REGISTER
177776          CMDREG= MODREG+2        ;LINK COMMAND REGISTER
```

```


                                    .SBTTL  MICROCODE MODULE A

                                    ;*****************************************************************
                                    ;
                                    ;THIS MODULE CONTAINS MICROCODE THAT IS USED FOR THE LOAD AND START
                                    ;FUNCTION TEST
                                    ;
                                    ;*****************************************************************


 000000' 106427  000340             MICFOA::MTPS    #PRI07                 ;DISABLE INTERRUPTS
 000004' 012737  012402  021020             MOV     #LASFTP!INTST,@#IPCSR1 ;SET TEST PATTERN AND IN TEST STATE BITS
 000012' 012737  004000  021000             MOV     #DNI,@#IPCSR0          ;SET DONE BIT
 000020' 000777                             BR      .                      ;HANG HERE UNTIL HOST RESETS US

 000022' 000024                     MICASZ::MICASZ-MICROA+2                ;SIZE OF MICROCODE MODULE A
         000001                     .END
```

```
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#
CL  IF= 021034                    22#
CMDREG= 177776                   102#
CSRFLG= 000001                    83#
CSRVEC= 000064                    75#
DATERR= 000000                    97#
DMACSR= 021002                     9#
DMAF  = 021022                    17#
DMAT0 = 021004                    10#
DMAT1 = 021006                    11#
DMAVEC= 000114                    76#
DMAWC = 021024                    18#
DNI   = 004000                    69#    118
ERRFLG= 000002                    84#
FATI  = 000400                    71#
INERR = 000003                    82#
INMON = 000001                    80#
INTST = 000002                    81#    117
IOADR = 020000                    94#     95
IOSIZ = 020000                    90#     95
IPCSR0= 021000                     8#    118*
IPCSR1= 021020                    16#    117*
LASFTP= 012400                    53#    117
LFRBUF= 021032                    21#
LINADR= 100000                    96#    100
LINSIZ= 077774                    92#    100
LRBUF = 021044                    26#
LTAC  = 021030                    20#
MDMAR0= 021014                    14#
MDMAR1= 021016                    15#
MDMAW0= 021026                    19#
MDMAW1= 021036                    23#
MDMA0 = 021010                    12#
MDMA1 = 021012                    13#
MDMSR = 021042                    25#
MICASZ  000022RG        002      121#
MICROA  000000RG        002      116#    121
MODREG= 177774                   100#    101      102
NXMFLG= 000010                    86#
```

```
NXMVEC= 000150                 74#      78#
PARERR= 000002                 99#
PARFLG= 000004                 85#
PARVEC= 000140                 77#
PCEI  = 040000                 66#
PCSRSW= 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
SIZ8K = 040000                 88#      91       92
STACK = 001000                 79#
TXI   = 010000                 68#
WCSADR= 000000                 93#      94
WCSSIZ= 020000                 89#      94
.     = 000024R     002       119


. ABS.  000000      000
        000000      001
MICRA   000024      002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROA.OBJ,MICROA.LST/CRF/DOC/NL:TOC/SOL=MICROA.MAC
 RUN-TIME: .5 .2 .2 SECONDS
 RUN-TIME RATIO: 59/1=55.7
 CORE USED:  8K  (16 PAGES)

 DOCUMENT PAGES:  5
```

```

                        .TITLE  MICROB - MICROCODE MODULE B

        000000'         .CSECT MICRB

                        .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000          IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002          DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004          DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006          DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010          MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012          MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014          MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016          MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020          IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022          DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024          DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026          MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030          LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032          LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034          CLRFIF  =       21034   ;CLEAR FIFO
        021036          MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040          PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042          MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044          LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060          PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062          PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064          PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066          PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070          PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072          PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5

                        .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000          BIT15   =       100000
        040000          BIT14   =       40000
        020000          BIT13   =       20000
        010000          BIT12   =       10000
        004000          BIT11   =       4000
        002000          BIT10   =       2000
        001000          BIT9    =       1000
        000400          BIT8    =       400
        000200          BIT7    =       200
        000100          BIT6    =       100
        000040          BIT5    =       40
        000020          BIT4    =       20
        000010          BIT3    =       10
        000004          BIT2    =       4
        000002          BIT1    =       2
        000001          BIT0    =       1
                        ;
        012400          LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340          PRI07   =       340
        000300          PRI06   =       300
        000240          PRI05   =       240
```

```
000200          PRIO4   =       200
000140          PRIO3   =       140
000100          PRIO2   =       100
000040          PRIO1   =       40
000000          PRIO0   =       0
                ;
                ;PCSRO - PORT CONTROL STATUS REGISTER 0
                ;
100000          SERI    =       BIT15
040000          PCEI    =       BIT14
020000          RXI     =       BIT13
010000          TXI     =       BIT12
004000          DNI     =       BIT11
002000          RCEI    =       BIT10
000400          FATI    =       BIT8
                ;
000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
001000          STACK=  1000            ;STACK LOCATION
000001          INMON=  1               ;IN MICROMONITOR STATE
000002          INTST=  2               ;IN A TEST STATE
000003          INERR=  3               ;IN ERROR STATE
000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
100000          NPRERR= BIT15           ;PCSRO FLAG INDICATING NPR ERROR OCCURRED
040000          NXMERR= BIT14           ;PCSRO FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
020000          UNIERR= BIT13           ;PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
010000          PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
020000          SIZ4K=  20000           ;4K WORDS
040000          SIZ8K=  SIZ4K*2         ;8K WORDS
020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
077774          LINSIZ= SIZ8K*2-4               ;SIZE OF LINK MEMORY
000000          WCSADR= 0               ;BASE ADDRESS OF WCS
020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
177774          MODREG= LINADR+LINSIZ   ;LINK MODE REGISTER
177774          ADDREG= MODREG          ;LINK STATION ADDRESS RAM REGISTER
177776          CMDREG= MODREG+2        ;LINK COMMAND REGISTER
```

```


                                            .SBTTL  MICROCODE MODULE B

                                            ;*****************************************************************
                                            ;
                                            ;THIS MODULE CONTAINS A MICROMONITOR AND MICROCODE FOR:
                                            ;
                                            ;       1-WCS MEMORY TEST
                                            ;
                                            ;*****************************************************************


000000' 106427  000340                      MICROB::MTPS    #PRI07                          ;DISABLE INTERRUPTS
000004' 012737  000000  177776                      MOV     #0,@#CMDREG                     ;TURN OFF THE LINK
000012' 012706  001000                              MOV     #STACK,SP                       ;SETUP STACK
000016' 012767  000001  000510                      MOV     #INMON,PCSR1                    ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000504  021020                      MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000                      MOV     #DNI,@#IPCSR0                   ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                                      MOV     PC,R0                           ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000344                              ADD     #ERRINT-.,R0                    ;HANDLER
000046' 005001                                      CLR     R1                              ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                              10$:    MOV     R0,(R1)+                        ;HANDLER
000052' 012721  000340                              MOV     #PRI07,(R1)+
000056' 020127  001000                              CMP     R1,#1000
000062' 002772                                      BLT     10$

000064' 010700                                      MOV     PC,R0                           ;SETUP PARITY TRAP VECTOR
000066' 062700  000422                              ADD     #PARINT-.,R0
000072' 010037  000140                              MOV     R0,@#PARVEC
000076' 012737  000340  000142                      MOV     #PRI07,@#PARVEC+2

000104' 010700                                      MOV     PC,R0                           ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000324                              ADD     #DMAINT-.,R0
000112' 010037  000114                              MOV     R0,@#DMAVEC
000116' 012737  000340  000116                      MOV     #PRI07,@#DMAVEC+2

000124' 010700                                      MOV     PC,R0                           ;SETUP CSR WRITE VECTOR
000126' 062700  000250                              ADD     #CSRWRT-.,R0
000132' 010037  000064                              MOV     R0,@#CSRVEC
000136' 012737  000200  000066                      MOV     #PRI04,@#CSRVEC+2

000144' 010700                                      MOV     PC,R0                           ;SETUP SANTITY TIMER VECTOR
000146' 062700  000256                              ADD     #TIMINT-.,R0
000152' 010037  000134                              MOV     R0,@#SANVEC
000156' 012737  000240  000136                      MOV     #PRI05,@#SANVEC+2

000164' 013700  021040                              MOV     @#PCSRSW,R0                     ;GET SWITCH PACK BITS
000170' 052700  176000                              BIS     #176000,R0                      ;MAP THEM INTO HOST I/O PAGE
000174' 006300                                      ASL     R0                              ;SHIFT OVER TO POSITION CORRECTLY
000176' 006300                                      ASL     R0
000200' 006300                                      ASL     R0
000202' 062700  000004                              ADD     #4,R0                           ;PCSR2 IS PCSR0+4
000206' 010067  000324                              MOV     R0,IPCSR2                       ;SAVE PCSR2 ADDRESS
000212' 012767  000003  000320                      MOV     #3,IPCSR2+2                     ;HIGH ORDER BITS 17:16
000220' 005067  000304                              CLR     FLG2                            ;INITIALIZE FLAG WORD
000224' 106427  000000                      15$:    MTPS    #PRI0                           ;ALLOW INTERRUPTS
```

```
000230' 005767  000274          20$:    TST     FLG2                    ;WAIT FOR A COMMAND FROM HOST
000234' 001775                          BEQ     20$

000236' 106427  000340                  MTPS    #PRI07                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000242' 032767  000001  000260          BIT     #CSRFLG,FLG2            ;DID HOST GIVE US A COMMAND?
000250' 001001                          BNE     30$                     ;YES
000252' 000777                          BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                        ;FOR LACK OF ANYTHING BETTER TO DO

000254' 013700  021000          30$:    MOV     @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000260' 042700  177760                  BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000264' 001004                          BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000266' 012737  000001  021020          MOV     #INMON,@#IPCSR1         ;YES, CLEAR SELF TEST BITS
000274' 000432                          BR      50$
000276' 022700  000017          35$:    CMP     #17,R0                  ;RETURN TO OPERATIONAL MICROCODE?
000302' 001432                          BEQ     60$                     ;YES
000304' 162700  000001                  SUB     #1,R0                   ;WILL FORM A TABLE INDEX WITH THIS
000310' 010701                          MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000312' 062701  000024                  ADD     #TBLB-.,R1
000316' 006300                          ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000320' 060001                          ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000322' 061101                          ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000324' 004711                          JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000326' 103404                          BCS     40$                     ;ERROR OCCURRED
000330' 112767  000001  000176          MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000336' 000403                          BR      45$                     ;MICROMONITR
000340' 112767  000003  000166  40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000346' 016737  000162  021020  45$:    MOV     PCSR1,@#IPCSR1
000354' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000362' 005067  000142          50$:    CLR     FLG2                    ;RESET FLAG WORD
000366' 000716                          BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000370' 005000                  60$:    CLR     R0                      ;FAKE SUCCESSFULL SELF TEST DONE
000372' 000137  040006                  JMP     @#40006                 ;START OPERATIONAL MICROCODE

000376' 052767  000001  000124  CSRWRT: BIS     #CSRFLG,FLG2            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000404' 000002                          RTI

000406  052767  000002  000114  ERRINT: BIS     #ERRFLG,FLG2            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000414' 012737  020000  021000          MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                        ;HAPPENED
000422' 000777                          BR      .                       ;JUST SIT HERE AND SPIN WHEELS

000424' 005267  000102          TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000430' 000002                          RTI

000432' 013767  021002  000106  DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000440' 032767  040000  000100          BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000446' 001404                          BEQ     10$                     ;NO
000450' 012737  040000  021000          MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                        ;LOCATION WAS ADDRESSED
000456' 000407                          BR      20$
000460' 032767  100000  000060  10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
000466' 001407                          BEQ     30$                     ;NO
000470' 012737  100000  021000          MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
```

```
000476' 012737  100000  021002  20$:     MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000504' 000777                           BR      .                       ;SIT HERE AND SPIN WHEELS
000506' 000002                  30$:     RTI

000510' 052767  000004  000012  PARINT:  BIS     #PARFLG,FLG2            ;SET PARITY ERROR OCCURRED
000516' 012737  010000  021000           MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                         ;OCCURRED
000524' 000777                           BR      .                       ;WAIT FOR HOST TO TIME OUT

000526' 000024                  TBLB:    .WORD   MICB1-.                 ;WCS MEMORY TEST

000530' 000000                  FLG2:    .WORD   0                       ;FLAG WORD
000532' 000000                  SANTIM:  .WORD   0                       ;COUNT FOR SANITY TIMER
000534' 000000                  PCSR1:   .WORD   0                       ;COPY OF PCSR1(BECAUSE WE CAN'T DO BYTE)
000536' 000000  000000          IPCSR2:  .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000542' 000000  000000          PCBADR:  .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000546' 000000                  DMDONE:  .WORD   0                       ;DMA INTERRUPT HAPPENED
000550' 000377                  BAKPAT:  .WORD   377                     ;BACKGROUND PATTERN FOR MEMORY TESTS

```

```


                                             .SBTTL  MODULE B, MICROTEST #1

                                             ;***************************************************************8
                                             ;
                                             ;THIS MICROTEST CHECKS THE WCS MEMORY. IT DOES SO BY RUNNING A SERIES
                                             ;OF MICROSUBTESTS ON THE MEMORY BETWEEN 0 AND 2K(000000-10000) AND THEN
                                             ;RELOCATES ITSELF TO THE LOWER HALF OF WCS AND TESTS THE UPPER HALF.
                                             ;
                                             ;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

000552' 112767  000002  177754    MICB1:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000556' 016737  177750  021020            MOV     PCSR1,@#IPCSR1
000566' 005067  000320                    CLR     BOTH                    ;START TESTING BOTTOM HALF FIRST
000572' 012767  001000  000304            MOV     #1000,LOWLIM            ;SET LOW LIMIT JUST ABOVE VECTORS
000600' 012767  010000  000300            MOV     #WCSSIZ/2,HILIM         ;HIGH LIMIT IS HALF WAY INTO WCS

000606' 010700                    5$:     MOV     PC,R0                   ;GET ADDRESS OF SUBTEST LIST
000610' 062700  000256                    ADD     #B1STBL-.,R0

000614' 005067  177710            10$:    CLR     FLG2                    ;CLEAR THE FLAG WORD
000620' 011001                            MOV     (R0),R1                 ;GET OFFSET FROM ENTRY TO SUBTEST
000622' 001437                            BEQ     30$                     ;NO MORE ENTRIES
000624' 060001                            ADD     R0,R1                   ;CALC SUBTEST ADDRESS
000626' 004711                            JSR     PC,(R1)                 ;GO EXECUTE SUBTEST
000630' 103403                            BCS     20$                     ;ERROR OCCURRED IN SUBTEST
000632' 062700  000002                    ADD     #2,R0                   ;POINT TO NEXT SUBTEST IN LIST
000636' 000766                            BR      10$

000640' 016737  177672  021010    20$:    MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2 AND PCSR3
000646' 016737  177666  021012            MOV     IPCSR2+2,@#MDMA1
000654' 013700  021014                    MOV     @#MDMAR0,R0             ;R0=CONTENTS OF PCSR2
000660' 013702  021014                    MOV     @#MDMAR0,R2             ;R2=CONTENTS OF PCSR3
000664' 010037  021010                    MOV     R0,@#MDMA0              ;POINT TO PCBB+0
000670' 010237  021012                    MOV     R2,@#MDMA1
000674' 016737  000210  021026            MOV     SUBNUM,@#MDMAW0          LOAD PCBB+0 WITH SUBTEST #...
                                                                          ;AND PCBB+1 WITH ERROR TYPE
000702' 010137  021026                    MOV     R1,@#MDMAW0             ;LOAD PCBB+2 WITH FAILING ADDRESS
000706' 010337  021026                    MOV     R3,@#MDMAW0             ;LOAD PCBB+4 WITH GOOD DATA
000712' 010437  021026                    MOV     R4,@#MDMAW0             ;LOAD PCBB+6 WITH BAD DATA
000716' 000261                            SEC                             ;INDICATE THIS MICROTEST FAILED
000720' 000434                            BR      45$

000722' 005767  000164            30$:    TST     BOTH                    ;HAVE WE DONE BOTH HALVES YET?
000726' 001030                            BNE     40$                     ;YES
000730' 005267  000156                    INC     BOTH                    ;NO, WE ARE ABOUT TO DO OTHER HALF
000734' 012767  010000  000142            MOV     #WCSSIZ/2,LOWLIM        ;SET LOW LIMIT AT BOTTOM OF UPPER HALF
000742' 012767  020000  000136            MOV     #WCSSIZ,HILIM           ;HIGH LIMIT IS TOP OF UPPER HALF
```

```
000750' 010700                            MOV     PC,R0                   ;GET BASE ADDRESS OF THIS MICROTEST
000752' 062700  177026                    ADD     #MICROB-.,R0
000756' 016701  001330                    MOV     MICBSZ,R1               ;GET ITS SIZE
000762' 012702  001000                    MOV     #1000,R2                ;RELOCATE TO JUST ABOVE STACK
000766' 112022                    35$:    MOVB    (R0)+,(R2)+             ;RELOCATE THIS MICROTEST
000770' 005301                            DEC     R1
000772' 001375                            BNE     35$
000774' 010700                            MOV     PC,R0
000776' 062700  177610                    ADD     #5$-.,R0
001002' 162700  007000                    SUB     #WCSSIZ/2-1000,R0
001006' 000110                            JMP     (R0)                    ;GO DO THE UPPER HALF TEST
001010' 000241                    40$:    CLC                             ;INDICATE THIS MICROTEST SUCCESS
001012' 106746                    45$:    MFPS    -(SP)                   ;SAVE RESULT STATUS
001014' 005767  000072                    TST     BOTH                    ;WE DONE BOTH HALVES?
001020' 001412                            BEQ     55$                     ;NO
001022' 010700                            MOV     PC,R0                   ;YES, WE HAVE TO RELOCATE BACK TO...
001024' 062700  176754                    ADD     #MICROB-.,R0            ;WHERE WF WERE
001030' 016701  001256                    MOV     MICBSZ,R1
001034' 012702  010000                    MOV     #WCSSIZ/2,R2
001040' 112022                    50$:    MOVB    (R0)+,(R2)+
001042' 005301                            DEC     R1
001044' 001375                            BNE     50$
001046' 106426                    55$:    MTPS    (SP)+                   ;RESTORE RESULT STATUS
001050' 112767  000001  177457            MOVB    #1,PCSR1+1              ;TELL HOST WHICH TEST T.IS WAS
001056' 016737  177452  021020            MOV     PCSR1,@#IPCSR'
001064' 000207                            RTS     PC                      ;RETURN TO MICROMONITOR

001066' 000026                    B1STBL: MICB1A-.
001070' 000162                            MICB1B-.
001072' 000320                            MICB1C-.
001074' 000432                            MICB1D-.
001076' 000670                            MICB1E-.
001100' 001002                            MICB1F-.
001102' 000000                            .WORD   0                       ;END OF LIST

001104' 000000                    LOWLIM: .WORD   0                       ;LOW LIMIT CF MEMORY TESTS
001106' 000000                    HILIM:  .WORD   0                       ;HIGH LIMIT OF MEMORY TESTS
001110'    000                    SUBNUM: .BYTE   0                       ;CURRENT SUBTEST # BE.NG EXECUTED
001111'    000                    ERRTYP: .BYTE   0                       ;TYPE OF ERROR, DATA =0, ADDRESS-1
001112' 000000                    BOTH:   .WORD   0                       ;FLAG INDICATING WHEN BOTH HALVES OF
                                                                          ;WCS HAVE BEEN TESTED
```

```
                                        .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.
                                        ;
                                        ;***********************************************************************

001114' 010046                          MICB1A: MOV     R0,-(SP)                ;SAVE R0
001116' 112767  000001  177764                  MOVB    #1,SUBNUM               ;TELL WE ARE IN SUBTEST A
001124' 106427  000300                          MTPS    #PRI06                  ;ALLOW PARITY ERRORS
001130' 012703  177777                          MOV     #177777,R3              ;GOOD DATA = ALL ONES
001134' 016700  177744                  5$:     MOV     LOWLIM,R0               ;GET BASE ADDRESS OF MEMORY
001140' 010320                          10$:    MOV     R3,(R0)+                ;WRITE ALL OVER MEMORY
001142' 020067  177740                          CMP     R0,HILIM
001146' 103774                                  BLO     10$

001150' 016701  177730                          MOV     LOWLIM,R1               ;POINT BACK TO BASE
001154' 011104                          12$:    MOV     (R1),R4                 ;READ DATA
001156' 032767  000004  177344                  BIT     #PARFLG,FLG2            ;DID A PARITY ERROR OCCUR?
001164' 001407                                  BEQ     15$                     ;NO
001166' 020304                                  CMP     R3,R4                   ;YES, WAS DATA READ GOOD?
001170' 001007                                  BNE     20$                     ;NO, DATA ERROR
001172' 112767  010000  177711                  MOVB    #PARERR,ERRTYP          ;TELL MICROMONITOR TYPE OF ERROR
001200' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001202' 000421                                  BR      40$                     ;LEAVE

001204' 020304                          15$:    CMP     R3,R4                   ;WAS DATA GOOD?
001206' 001405                                  BEQ     30$                     ;YES
001210' 112767  000000  177673          20$:    MOVB    #DATERR,ERRTYP          ;TELL MICROMONITOR TYPE OF ERROR
001216' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001220' 000412                                  BR      40$

001222' 062701  000002                  30$:    ADD     #2,R1                   ;POINT TO NEXT MEMORY ADDRESS
001226' 020167  177654                          CMP     R1,HILIM                ;DONE ALL MEMORY?
001232' 001350                                  BNE     12$                     ;NOT YET
001234' 005703                                  TST     R3                      ;DONE BOTH DATA TYPES?
001236' 001402                                  BEQ     35$                     ;YES
001240' 005103                                  COM     R3                      ;NO, WRITE ONES NOW
001242' 000734                                  BR      5$

001244' 000241                          35$:    CLC                             ;INDICATE SUCCESS
001246' 012600                          40$:    MOV     (SP)+,R0                ;RESTORE R0
001250' 000207                                  RTS     PC
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST B

                                        ;*********************************************************************
                                        ;
                                        ;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
                                        ;
                                        ;*********************************************************************

001252' 010046                          MICB1B: MOV     R0,-(SP)                        ;SAVE R0
001254' 112767  000002  177626                  MOVB    #2,SUBNUM                       ;TELL HIM WE ARE IN THE SECOND SUBTEST

001262' 016700  177616                          MOV     LOWLIM,R0                       ;SET 'WRITTEN TO' ADDRESS
001266' 016701  177612                  10$:    MOV     LOWLIM,R1                       ;SET 'READ FROM' ADDRESS
001272' 012710  177777                          MOV     #177777,(R0)                    ;WRITE DATA INTO LOCATION AT BASE

001276' 011104                          20$:    MOV     (R1),R4                         ;READ DATA FROM MEMORY WHICH IS AN EVEN
                                                                                        ;INCREMENT AWAY
001300' 020001                                  CMP     R0,R1                           ;IS 'READ FROM AND 'WRITTEN TO'
                                                                                        ;ADDRESSES THE SAME?
001302' 001413                                  BEQ     70$                             ;YES, SO DATA IN BOTH SHOULD BE THE SAME
001304' 005003                                  CLR     R3                              ;GOOD DATA IS ZEROS
001306' 005704                                  TST     R4                              ;DATA IN 'READ FROM' MUST BE 0'S
001310' 001422                                  BEQ     50$                             ;OK GO CHANGE 'READ FROM' ADDRESS
                                                                                        ;ERROR OCCURRED BUT WE DON'T KNOW IF...
                                                                                        ;IT IS A DATA ERROR OR AN ADDRESS ERROR
001312' 022704  177777                          CMP     #177777,R4                      ;WAS DATA READ ALL 1'S?
001316' 001012                                  BNE     40$                             ;NO, SO IT WAS A DATA ERROR
001320' 112767  000001  177563                  MOVB    #ADRERR,ERRTYP                  ;YES, IT WAS AN ADDRESS ERROR
001326' 000261                                  SEC                                     ;INDICATE FAILURE
001330' 000426                                  BR      60$                             ;LEAVE THIS SUBTEST
001332' 022704  177777                  70$:    CMP     #177777,R4                      ;DATA READ MUST BE ALL 1'S
001336' 001407                                  BEQ     50$                             ;IT IS GOOD
001340' 012703  177777                          MOV     #177777,R3                      ;GOOD DATA IS ONES
001344' 112767  000000  177537          40$:    MOVB    #DATERR,ERRTYP                  ;IT WAS A DATA ERROR
001352' 000261                                  SEC                                     ;INDICATE FAILURE
001354' 000414                                  BR      60$                             ;LEAVE THIS SUBTEST

001356' 062701  001000                  50$:    ADD     #1000,R1                        ;CHANGE 'READ FROM' ADDRESS BY .25K
001362' 020167  177520                          CMP     R1,HILIM                        ;WE GO PAST BOUNDARY?
001366' 103743                                  BLO     20$                             ;NO CONTINUE WITH SAME 'WRITTEN TO'
001370' 005010                                  CLR     (R0)                            ;CLEAR OLD 'WRITTEN TO' ADDRESS
001372' 062700  001000                          ADD     #1000,R0                        ;CHANGE 'WRITTEN TO' ADDRESS BY .25K
001376' 020067  177504                          CMP     R0,HILIM                        ;WE PAST BOUNDARY?
001402' 103731                                  BLO     10$                             ;NO CONTINUE THIS SUBTEST
001404' 000241                                  CLC                                     ;INDICATE SUCCESS OF THIS SUBTEST
001406' 012600                          60$:    MOV     (SP)+,R0                        ;RESTORE R0
001410' 000207                                  RTS     PC
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST C

                                        ;*****************************************************************
                                        ;
                                        ;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.
                                        ;
                                        ;*****************************************************************

001412' 010046                          MICB1C: MOV     R0,-(SP)                ;SAVE R0
001414' 112767  000003  177466                  MOVB    #3,SUBNUM               ;TELL WEA ARE IN SUBTEST 'C'
001422' 016701  177456                          MOV     LOWLIM,R1               ;GET BASE ADDRESS OF MEMORY
001426' 012700  000001                  1$:     MOV     #1,R0                   ;DATA = 1 IN LEAST SIGNIFICANT BIT
001432' 010002                                  MOV     R0,R2                   ;INDICATE WE ARE SHIFTING A '1'
001434' 010011                          2$:     MOV     R0,(R1)                 ;WRITE LOCATION WITH GOOD DATA
001436' 011104                                  MOV     (R1),R4                 ;READ DATA FROM SAME LOCATION
001440' 020004                          3$:     CMP     R0,R4                   ;IS DATA THE SAME AS WRITTEN?
001442' 001406                                  BEQ     4$                      ;YES, OK
001444' 112767  000000  177437                  MOVB    #DATERR,ERRTYP          ;ERROR IS DATA ERROR
001452' 010003                                  MOV     R0,R3                   ;GOOD DATA
001454' 000261                                  SEC                             ;INDICATE THIS SUBTEST FAILED
001456' 000421                                  BR      6$                      ;LEAVE THIS SUBTEST
001460' 005702                          4$:     TST     R2                      ;ARE WE SHIFTING A 1 OR A 0?
001462' 001406                                  BEQ     5$                      ;ZERO
001464' 006300                                  ASL     R0                      ;SHIFT THE ONE OVER
001466' 103362                                  BCC     2$                      ;IF THE '1' HAS NOT BEEN SHIFTED...
                                                                                ;THRU THE 16 POSITIONS CONTINUE WITH 1
001470' 005002                                  CLR     R2                      ;ELSE START SHIFTING A '0'
001472' 012700  177776                          MOV     #177776,R0              ;START WITH LSB = 0 ALL OTHERS 1'S
001476' 000756                                  BR      2$                      ;CONTINUE WITH SHIFTING A 0
001500' 000261                          5$:     SEC                             ;MOVE '0' OVER ONE BIT POSITION
001502' 006100                                  ROL     R0                      ;HAS '0' BEEN IN ALL POSITIONS?
001504' 103753                                  BCS     2$                      ;NO CONTINUE WITH SHIFTING A 0
001506' 062701  001000                          ADD     #1000,R1                ;CONTINUE TEST AT NEXT BOUNDARY
001512' 020167  177370                          CMP     R1,HILIM                ;DONE ALL OF MEMORY?
001516' 103743                                  BLO     1$                      ;NO
001520' 000241                                  CLC                             ;INDICATE THIS SUBTEST SUCCESSFUL
001522' 012600                          6$:     MOV     (SP)+,R0                ;RESTORE R0
001524' 000207                                  RTS     PC
```

```
                                          .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST D

                                          ;*******************************************************************
                                          ;
                                          ;ADDRESS BIT SHIFT #1
                                          ;
                                          ;THIS TEST CHECKS FOR DUAL ADDRESS PROBLEMS BY FIRST WRITING A BACKGROUND
                                          ;PATTERN THROUGHOUT 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 BANK.
                                          ;
                                          ;*******************************************************************

 001526' 010046                           MICB1D: MOV     R0,-(SP)                ;SAVE R0
 001530' 112767  000004  177352                   MOVB    #4,SUBNUM               ;TELL WE ARE IN SUBTEST 4
 001536' 012767  000377  177004                   MOV     #377,BAKPAT             ;LOAD BAKPAT CONSTANT
 001544' 005003                           1$:     CLR     R3                      ;CONTAINS BANK ADDRESS
 001546' 016700  177332                   2$:     MOV     LOWLIM,R0               ;WRITE LINK MEMORY WITH BACKGROUND...
 001552' 016720  176772                   3$:     MOV     BAKPAT,(R0)+            ;PATTERN
 001556' 020067  177324                           CMP     R0,HILIM
 001562' 103773                                   BLO     3$
 001564' 016700  176760                           MOV     BAKPAT,R0               ;R0 CONTAINS GOOD DATA

 001570' 005002                           4$:     CLR     R2                      ;R2 WILL BE OUR 'WRITTEN TO' ADDRESS
 001572' 050302                           6$:     BIS     R3,R2                   ;INDEX INTO THIS BANK
 001574' 020267  177304                           CMP     R2,LOWLIM               ;IS RESULT LESS THAN MEM BASE?
 001600' 103450                                   BLO     16$                     ;YES, DON'T USE THIS ADDRESS
 001602' 020267  177300                           CMP     R2,HILIM                ;IS RESULT LARGER THAN MEM TOP?
 001606' 103061                                   BHIS    20$                     ;YES, DON'T USE THIS ADDRESS EITHER
 001610' 000312                                   SWAB    (R2)                    ;WRITE COMPLEMENT OF PATTERN
 001612' 005001                                   CLR     R1                      ;R1 WILL BE OUR 'READ FROM' ADDRESS
 001614' 050301                           7$:     BIS     R3,R1                   ;INDEX INTO THIS BANK
 001616' 020167  177262                           CMP     R1,LOWLIM               ;IS RESULT LESS THAN  MEM BASE?
 001622' 103426                                   BLO     12$                     ;YES, DON'T USE THIS ADDRESS
 001624' 020167  177256                           CMP     R1,HILIM                ;IS RESULT LARGER THAN MEM TOP?
 001630' 103033                                   BHIS    15$                     ;YES, DON'T USE THIS ADDRESS EITHER
 001632' 011104                                   MOV     (R1),R4                 ;READ DATA
 001634' 020102                                   CMP     R1,R2                   ;IS 'READ FROM' AND 'WRITTEN TO' SAME?
 001636' 001414                                   BEQ     10$                     ;YES, GO CHECK DATA
 001640' 020004                                   CMP     R0,R4                   ;NO, DATA READ SHOULD BE SAME AS BAKPAT
 001642' 001416                                   BEQ     12$                     ;IF SO CONTINUE WITH NEW INDEX

 001644' 112767  000001  177237           8$:     MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
 001652' 010003                                   MOV     R0,R3                   ;GET GOOD DATA
 001654' 000261                                   SEC                             ;INDICATE THIS SUBTEST FAILED
 001656' 000441                                   BR      25$                     ;LEAVE THIS SUBTEST
 001660' 020067  176664                           CMP     R0,BAKPAT               ;DOES R0 CONTAIN SWAPPED DATA?
 001664' 001405                                   BEQ     12$                     ;NO
 001666' 000403                                   BR      11$                     ;YES

 001670' 000300                           10$:    SWAB    R0                      ;MAKE GOOD DATA LIKE SWAPPED BAKPAT
 001672' 020004                                   CMP     R0,R4                   ;IS DATA READ SAME AS DATA WRITTEN?
 001674' 001363                                   BNE     8$                      ;NO, ERROR
 001676' 000300                           11$:    SWAB    R0                      ;RESTORE GOOD DATA TO BAKPAT
```

```
001700' 040301          12$:    BIC     R3,R1           ;REMOVE BANK SELECT FROM 'READ FROM'
001702' 005701                  TST     R1              ;DID AN INDEX EXIST?
001704' 001001                  BNE     13$             ;YES
001706' 005201                  INC     R1              ;NO, MAKE ONE
001710' 006101          13$:    ROL     R1              ;SHIFT INDEX OVER
001712' 020127  001000          CMP     R1,#1000        ;IS IT LESS THAN .25K
001716' 103736                  BLO     7$              ;YES, CONTINUE CHECKING FOR BOGUS DATA
                                                        ;IN THIS BANK
001720' 000312          15$:    SWAB    (R2)            ;RESTORE CONTENTS OF 'WRITTEN TO'
001722' 040302          16$:    BIC     R3,R2           ;REMOVE BANK SELECT FROM 'WRITTEN TO'
001724' 005702                  TST     R2              ;DOES 'WRITTEN TO' CONTAIN AN ADDRESS?
001726' 001001                  BNE     18$             ;YES
001730' 005202                  INC     R2              ;NO, MAKE ONE
001732' 006102          18$:    ROL     R2              ;SHIFT 'WRITTEN TO' ADDRESS
001734' 020227  001000          CMP     R2,#1000        ;IS 'WRITTEN TO' LARGER THAN .25K?
001740' 103714                  BLO     6$              ;NO
001742' 060203                  ADD     R2,R3           ;MAKE 'WRITTEN TO' POINT TO NEXT .25K
001744' 020367  177136          CMP     R3,HILIM        ;IS RESULT LARGER THAN MEM TOP?
001750' 103707                  BLO     4$              ;NO CONTINUE TESTING
001752' 000367  176572  20$:    SWAB    BAKPAT          ;REPEAT TEST WITH COMPLEMENTED PATTERN
001756' 001672                  BEQ     1$
001760' 000241                  CLC                     ;INDICATE THIS SUBTEST SUCCESS
001762' 012600          25$:    MOV     (SP)+,R0        ;RESTORE R0
001764' 000207                  RTS     PC
```

```
                                           .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST E

                                           ;*********************************************************************
                                           ;
                                           ;ADDRESS BIT SHIFT #2'
                                           ;
                                           ;THIS TEST CHECKS FOR DUAL ADDRESSING PROBLEMS BY WRITING EACH LOCATION
                                           ;OF MEMORY WITH ITS ADDRESS AND THEN VERIFYING EACH LOCATION.
                                           ;IT THEN DOES THE SAME THING BUT USES THE COMPLEMENT OF THE ADDRESS
                                           ;AS DATA.
                                           ;
                                           ;*******************************************************************

 001766' 010046                            MICB1E: MOV     R0,-(SP)                ;SAVE R0
 001770' 112767  000005  177112                    MOVB    #5,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
 001776' 005003                                    CLR     R3                      ;FLAG INDICATING ADDRESS IS COMPLEMENTED
 002000' 016701  177100                            MOV     LOWLIM,R1               ;GET STARTING ADDRESS OF MEMORY
 002004' 010100                            1$:     MOV     R1,R0                   ;GET ADDRESS TO WORK WITH
 002006' 005703                                    TST     R3                      ;SHOULD WE COMPLEMENT THE DATA TO STORE ?
 002010' 001401                                    BEQ     2$                      ;NO STORE AS IS
 002012' 005100                                    COM     R0                      ;COMPLEMENT DATA
 002014' 010021                            2$:     MOV     R0,(R1)+                ;WRITE DATA
 002016' 020167  177064                            CMP     R1,HILIM                ;IS NEW ADDRESS LARGER THAN MEM?
 002022' 103770                                    BLO     1$                      ;NO, KEEP FILLING MEMORY

 002024' 014104                            3$:     MOV     -(R1),R4                ;READ DATA STORED
 002026' 020004                                    CMP     R0,R4                   ;IS DATA READ SAME AS WRITTEN?
 002030' 001406                                    BEQ     4$                      ;YES
 002032' 112767  000001  177051                    MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
 002040' 010003                                    MOV     R0,R3                   ;GET GOOD DATA
 002042' 000261                                    SEC                             ;INDICATE FAILURE
 002044' 000414                                    BR      10$                     ;LEAVE THIS SUBTEST
 002046' 010100                            4$:     MOV     R1,R0                   ;CALC GOOD DATA FOR NEXT LOCATION
 002050' 162700  000002                            SUB     #2,R0
 002054' 005703                                    TST     R3
 002056' 001401                                    BEQ     5$
 002060' 005100                                    COM     R0
 002062' 020167  177016                    5$:     CMP     R1,LOWLIM               ;HAVE WE CHECKED ALL LOCATIONS
 002066' 101356                                    BHI     3$                      ;NOT YET
 002070' 005103                                    COM     R3                      ;HAVE WE DONE IT COMPLEMENTED?
 002072' 001344                                    BNE     1$                      ;NO, REPEAT WITH COMPLEMENT
 002074' 000241                                    CLC                             ;INDICATE SUCCESS
 002076' 012600                            10$:    MOV     (SP)+,R0                ;RESTORE R0
 002100' 000207                                    RTS     PC
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST F

                                        ;*******************************************************************
                                        ;
                                        ;MARCH TEST
                                        ;
                                        ;THIS TEST WRITES A BACKGROUND PATTERN IN ALL OF MEMORY
                                        ;
                                        ;1-READ EVERY LOCATION FOR CORRECT DATA, SWAPS BYTES AT EACH LOCATION
                                        ;AND PROCEED IN MAX TO MIN DIRECTION
                                        ;2-READ EVERY LOCATION FOR SWAPPED DATA, WRITES ORIGINAL PATTERN IN EACH
                                        ;LOCATION AND PROCEEDS IN MIN TO MAX DIRECTION
                                        ;3-REPEAT 1 GOING IN MIN TO MAX DIRECTION
                                        ;4-REPEAT 2 GOING IN MAX TO MIN DIRECTION
                                        ;
                                        ;*******************************************************************

002102' 010046                          MICB1F: MOV     R0,-(SP)                ;SAVE R0
002104' 112767  000006  176776                  MOVB    #6,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
002112' 005003                                  CLR     R3                      ;ADDRESS DIRECTION FLAG 0 = MIN.->MAX
002114' 016701  176764                  10$:    MOV     LOWLIM,R1               ;FILL  MEMORY WITH BACKGROUND PATTERN
002120' 012700  000377                          MOV     #377,R0                 ;BACKGROUND PATTERN=LOW BYTE ALL 1'S
002124' 010021                          12$:    MOV     R0,(R1)+
002126' 020167  176754                          CMP     R1,HILIM
002132' 103774                                  BLO     12$

002134' 014104                          20$:    MOV     -(R1),R4                ;STARTING FROM THE TOP, READ DATA
002136' 020004                                  CMP     R0,R4                   ;R0 = GOOD DATA, R4 = DATA READ
002140' 001406                                  BEQ     30$                     ;IF SAME OK
002142' 112767  000000  176741                  MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002150' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
002152' 000261                                  SEC                             ;INDICATE FAILURE
002154' 000454                                  B       200$                    ;LEAVE THIS SUBTEST
002156' 000300                          30$:    SWAB    R0                      ;NEW GOOD DATA PATTERN
002160' 010011                                  MOV     R0,(R1)                 ;STORE AT SAME PLACE
002162' 011104                                  MOV     (R1),R4                 ;READ IT BACK
002164' 020400                                  CMP     R4,R0                   ;R0=GOOD DATA, R4=DATA READ
002166' 001406                                  BEQ     40$                     ;IF SAME OK
002170' 112767  000000  176713                  MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002176' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
002200' 000261                                  SEC                             ;FAILURE
002202' 000441                                  BR      200$                    ;LEAVE THIS SUBTEST
002204' 000300                          40$:    SWAB    R0                      ;SWITCH GOOD DATA AGAIN
002206' 001  7                                  BNE     90$                     ;IF ORIGINAL PATTERN THEN WE ARE...
                                                                                ;READING THE MEMORY TO CONTAIN A...
                                                                                ;BACKGROUND OF LOW BYTE = ALL 1'S...
                                                                                ;AND WRITING IT BACK SWAPPED AND...
                                                                                ;VERIFING SWAPPED DATA
002210' 005703                          50$:    TST     R3                      ;
002212' 001027                                  BNE     100$                    ;
002214' 062701  000002                  60$:    ADD     #2,R1                   ;WE ARE GOING MIN->MAX SO ADJUST POINTER
002220' 020167  176662                          CMP     R1,HILIM                ;WE AT MAX?
002224' 103012                                  BHIS    80$                     ;YES
002226' 011104                          70$:    MOV     (R1),R4                 ;READ DATA
002230' 020004                                  CMP     R0,R4                   ;R0=GOOD DATA, R4 DATA READ
002232' 001751                                  BEQ     30$                     ;OK IF SAME
```

```
002234' 112767  000000  176647           MOVB    #DATERR,ERRTYP            ;INDICATE DATA ERROR
002242' 010003                           MOV     R0,R3                     ;GET GOOD DATA
002244' 000261                           SEC                               ;INDICATE FAILURE
002246' 000417                           BR      200$                      ;LEAVE THIS SUBTEST
002250' 000742                           BR      30$
002252' 000300                   80$:    SWAB    R0                        ;SWITCH GOOD DATA
002254' 001727                           BEQ     20$                       ;IF LOW BYTE = ALL 0'S GO IN MAX->MIN
002256' 005103                           COM     R3                        ;ELSE GO IN MIN->MAX DIRECTION
002260' 016701  176620                   MOV     LOWLIM,R1                 ;SET ADDRESS POINTER TO MIN
002264' 000760                           BR      70$

002266' 005703                   90$:    TST     R3                        ;ARE WE GOING MIN->MAX?
002270' 001347                           BNE     50$                       ;YES

002272' 020167  176606           100$:   CMP     R1,LOWLIM                 ;NO, CHECK TO SEE IF WE ARE AT MIN
002276' 101316                           BHI     20$                       ;NOT YET
002300' 000300                           SWAB    R0                        ;WE ARE AT MIN SO SWITCH GOOD DATA
002302' 001751                           BEQ     70$                       ;IF LOW BYTE = ALL 0'S ELSE WE ARE DONE
002304' 000241                           CLC                               ;INDICATE SUCCESS
002306' 012600                   200$:   MOV     (SP)+,R0                  ;RESTORE R0
002310' 000207                           RTS     PC

002312' 002314                   MICBSZ::MICBSZ-MICROB+2                   ;SIZE OF MICROCODE MODULE B
        000001                   .END
```

```
MICRGB - MICROCODE MODULE B     MACY11 30(1046)  11-JAN-83  09:50  PAGE 17
MICROB.MAC    10-JAN-83 17:52             CROSS REFERENCE TABLE -- USER SYMBOLS

ADDREG= 177774              103#
ADRERR= 000001              101#     414      524      589
BAKPAT  000552R     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#
CMDREG= 177776              104#     119*
CSRFLG= 000001               81#     167      198
CSRVEC= 000064               74#     144*     145*
CSRWRT  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#
DMAT0  = 021004              10#
DMAT1  = 021006              11#
DMAVEC= 000114               75#     139*     140*
DMAWC  = 021024              18#
DMDON.  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*
FATI   = 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
INMON  = 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
```

```
                                  626     670     676
LRBUF  = 021044                     #
LTAC   = 021030                   20#
MDMAR0= 021014                    14#     276     277
MDMAR1= 021016                    15#
MDMAW0= 021026                    19#     280*    282*    283*    284*
MDMAW1= 021036                    23#
MDMA0  = 021010                   12#     274*    278*
MDMA1  = 021012                   13#     275*    279*
MDMSR  = 021042                   25#
MICBSZ  002312RG     002         295      310     684#
MICB1   000552R      002         227      256#
MICB1A  001114R      002         320      344#
MICB1B  001252R      002         321      395#
MICB1C  001412R      002         322      450#
MICB1D  001526R      002         323      495#
MICB1E  001766R      002         324      574#
MICB1F  002102R      002         325      623#
MICROB  000000RG     002         118#     294     309     684
MODREG= 177774                   102#     103     104
NPRERR= 100000                    86#     217
NPRFLG= 000020                    85#
NXMERR= 040000                    87#     212
NXMFLG= 000010                    84#
PARERR= 010000                    89#     223     359
PARFLG= 000004                    83#     222     355
PARINT  000510R      002         133      222#
PARVEC= 000140                     #      134*    135*
PCBADR  000542R      002         233#
PCEI   - 040000                   66#
PCSRSW= 021040                    24#     152
PCSR1   000534R      002         121*     122     187*    189*    190     231#    256*    257     316*    317
PHYAD0= 021060                    27#
PHYAD1= 021062                    28#
PHYAD2= 021064                    29#
PHYAD3= 021066                    30#
PHYAD4= 021070                    31#
PHYAD5= 021072                    32#
PRI00  = 000000                   61#     161*
PRI01  = 000040                   60#
PRI02  = 000100                   59#
PRI03  = 000140                   58#
PRI04  = 000200                   57#     145
PRI05  = 000240                   56#     150
PRI06  = 000300                   55#     345*
PRI07  = 000340                   54#     118*    128     135     140     166*
RCEI   = 002000                   70#
ROMADR= 040000                    98#      99
ROMSIZ= 040000                    94#      99
RXI    = 020000                   67#
SANTIM  000532R      002         206*     230#
SANVEC= 000134                    73#     149*    150*
SERI   = 100000                   65#
SIZ4K  = 020000                   90#      91      92      93
SIZ8K  = 040000                   91#      94      95
STACK  = 001000                   77#     120
```

```
SUBNUM  001110R     002     280      330#     345*     396*     451*     496*     575*     624*
TBLB    000526R     002     181      227#
TIMINT  000424R     002     148      206#
TXI   = 010000               68#
UNIERR= 020000               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


. ABS.  000000      000
        000000      001
MICRB   002314      002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROB.OBJ,MICROB.LST/CRF/DOC/NL:TOC/SOL=MICROB.MAC
 RUN-TIME: .9 1 .3 SECONDS
 RUN-TIME RATIO: 43/2=15.4
 CORE USED:  8K  (16 PAGES)

 DOCUMENT PAGES:  18
```

```

                                .TITLE  MICROC - MICROCODE MODULE C

        000000'                 .CSECT MICRC

                                .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000                  IPCSRO  =       21000   ;INTERNAL PCSRO ADDRESS
        021002                  DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004                  DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006                  DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010                  MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012                  MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014                  MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016                  MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020                  IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022                  DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024                  DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026                  MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030                  LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032                  LTRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034                  CLRFIF  =       21034   ;CLEAR FIFO
        021036                  MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040                  PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042                  MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044                  LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060                  PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062                  PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064                  PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066                  PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070                  PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072                  PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5

                                .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000                  BIT15   =       100000
        040000                  BIT14   =       40000
        020000                  BIT13   =       20000
        010000                  BIT12   =       10000
        004000                  BIT11   =       4000
        002000                  BIT10   =       2000
        001000                  BIT9    =       1000
        000400                  BIT8    =       400
        000200                  BIT7    =       200
        000100                  BIT6    =       100
        000040                  BIT5    =       40
        000020                  BIT4    =       20
        000010                  BIT3    =       10
        000004                  BIT2    =       4
        000002                  BIT1    =       2
        000001                  BIT0    =       1
                                ;
        012400                  LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340                  PRI07   =       340
        000300                  PRI06   =       300
        000240                  PRI05   =       240
        000200                  PRI04   =       200
        000140                  PRI03   =       140
        000100                  PRI02   =       100
        000040                  PRI01   =       40
        000000                  PRI00   =       0
                                ;
                                ;PCSRO - PORT CONTROL STATUS REGISTER 0
                                ;
        100000                  SERI    =       BIT15
        040000                  PCEI    =       BIT14
        020000                  RXI     =       BIT13
        010000                  TXI     =       BIT12
        004000                  DNI     =       BIT11
        002000                  RCEI    =       BIT10
        000400                  FATI    =       BIT8
                                ;
        000134                  SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064                  CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114                  DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140                  PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000                  STACK=  1000            ;STACK LOCATION
        000001                  INMON=  1               ;IN MICROMONITOR STATE
        000002                  INTST=  2               ;IN A TEST STATE
        000003                  INERR=  3               ;IN ERROR STATE
        000001                  CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002                  ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004                  PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010                  NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020                  NPRFLG= BIT4            ;NPR TIMEOUT ERROR OCCURRED
        100000                  NPRERR= BIT15           ;PCSRO FLAG INDICATING NPR ERROR OCCURRED
        040000                  NXMERR= BIT14           ;PCSRO FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000                  UNIERR= BIT13           ;PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
        010000                  PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        020000                  SIZ4K=  20000           ;4K WORDS
        040000                  SIZ8K=  SIZ4K*2         ;8K WORDS
        020000                  WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000                  IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000                  ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774                  LINSIZ= SIZ8K*2-4               ;SIZE OF LINK MEMORY
        000000                  WCSADR= 0               ;BASE ADDRESS OF WCS
        020000                  IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000                  ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000                  LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        000000                  DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000001                  ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
        177774                  MODREG= LINADR+LINSIZ   ;LINK MODE REGISTER
        177774                  ADDREG= MODREG          ;LINK STATION ADRESS RAM REGISTER
        177776                  CMDREG= MODREG+2        ;LINK COMMAND REGISTER
```

```

                                        .SBTTL    MICROCODE MODULE C

                                        ;*****************************************************************
                                        ;
                                        ;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
                                        ;
                                        ;*****************************************************************


000000' 106427  000340          MICROC::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776          MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                  MOV     #STACK,SP               ;SETUP STACK
000016' 012767  000001  000502          MOV     #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000476  021020          MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                          MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000320                  ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                          CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                  10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                  MOV     #PRI07,(R1)+
000056' 020127  001000                  CMP     R1,#1000
000062' 002772                          BLT     10$

000064' 010700                          MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000376                  ADD     #PARINT-.,R0
000072' 010037  000140                  MOV     R0,@#PARVEC
000076' 012737  000340  000142          MOV     #PRI07,@#PARVEC+2

000104' 010700                          MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000300                  ADD     #DMAINT-.,R0
000112' 010037  000114                  MOV     R0,@#DMAVEC
000116' 012737  000300  000116          MOV     #PRI06,@#DMAVEC+2

000124' 010700                          MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000224                  ADD     #CSRWRT-.,R0
000132' 010037  000064                  MOV     R0,@#CSRVEC
000136' 012737  000200  000066          MOV     #PRI04,@#CSRVEC+2

000144' 010700                          MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000232                  ADD     #TIMINT-.,R0
000152' 010037  000134                  MOV     R0,@#SANVEC
000156' 012737  000240  000136          MOV     #PRI05,@#SANVEC+2

000164' 013700  021040                  MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000170' 052700  176000                  BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000174' 006300                          ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000176' 006300                          ASL     R0
```

```
000200' 006300                         ASL     R0
000202' 062700  000004                 ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000206' 010067  000316                 MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000212' 012767  000003  000312         MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000220' 005067  000276                 CLR     FLG3                    ;INITIALIZE FLAG WORD
000224' 106427  000000          15$:   MTPS    #PRI00                  ;ALLOW INTERRUPTS

000230' 005767  000266          20$:   TST     FLG3                    ;WAIT FOR AN INTERRUPT
000234' 001775                         BEQ     20$

000236' 106427  000340                 MTPS    #PRI07                  ;RAISE CPU PRIORITY
000242' 032767  000001  000252         BIT     #CSRFLG,FLG3            ;DID HOST GIVE US A COMMAND?
000250' 001001                         BNE     30$                     ;YES
000252' 000777                         BR      .                       ;NO, ERROR SO JUST SIT HERE ..
                                                                       ;FOR LACK OF ANYTHING BETTER TO DO

000254' 113700  021000          30$:   MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000260' 042700  177760                 BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000264' 022700  000017                 CMP     #17,R0                  ;RETURN TO OPERATIONAL MICROCODE?
000270' 001425                         BEQ     60$                     ;YES
000272' 010701                         MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000274' 062701  000206                 ADD     #TBLC-.,R1
000300' 006300                         ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000302' 060001                         ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000304' 061101                         ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000306' 004711                         JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000310' 103404                         BCS     40$                     ;ERROR OCCURRED
000312' 112767  000001  000206         MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000320' 000403                         BR      45$                     ;MICROMONITR
000322' 112767  000003  000176  40$:   MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000330' 016737  000172  021020  45$:   MOV     PCSR1,@#IPCSR1
000336' 005067  000160                 CLR     FLG3                    ;RESET FLAG WORD
000342' 000730                         BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000344' 005000                  60$:   CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000346' 000137  040006                 JMP     @#40006                 ;START OPERATIONAL MICROCODE

000352' 052767  000001  000142  CSRWRT: BIS    #CSRFLG,FLG3            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000360' 000002                         RTI

000362' 052767  000002  000132  ERRINT: BIS    #ERRFLG,FLG3            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000370' 012737  020000  021000         MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                       ;HAPPENED
000376' 000777                         BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                       ;COUNT ON THE HOST TO TIMEOUT WAITING

000400' 005267  000120          TIMINT: INC    SANTIM                  ;COUNT TICKS AS THEY OCCUR
000404' 000002                         RTI

000406' 013767  021002  000124  DMAINT: MOV    @#DMACSR,DMDONE         ;GET DMA STATUS
000414' 032767  040000  000116         BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000422' 001404                         BEQ     10$                     ;NO
000424' 012737  040000  021000         MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                       ;LOCATION WAS ADDRESSED
000432' 000407                         BR      20$
000434' 032767  100000  000076  10$:   BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000442' 001407                            BEQ     30$                     ;NO
000444' 012737  100000  021000            MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
000452' 012737  100000  021002  20$:      MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000460' 000777                            BR      .                       ;SIT HERE AND SPIN WHEELS
000462' 000002                  30$:      RTI


000464' 052767  000004  000030  PARINT:   BIS     #PARFLG,FLG3            ;SET PARITY ERROR OCCURRED
000472' C12737  010000  021000            MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                          ;OCCURRED
000500' 000002                            RTI

000502' 000042                  TBLC:     .WORD   CLRC-.                  ;CLEAR COMMAND
000504' 000050                            .WORD   MICC1-.                 ;PCSR0 INTERRUPT BIT TEST
000506' 000136                            .WORD   MICC2-.                 ;SANITY TIMER TEST
000510' 000222                            .WORD   MICC3-.                 ;COMPREHENSIVE LINK MEMORY TEST
000512' 001662                            .WORD   MICC4-.                 ;DMA 'TO' ADDRESS REGISTER TEST
000514' 001772                            .WORD   MICC5-.                 ;DMA 'FROM' ADDRESS REGISTER TEST
000516' 002166                            .WORD   MICC6-.                 ;DMA BLOCK TRANSFER TEST
000520' 002346                            .WORD   MICC7-.                 ;RIPPLE TEST

000522' 000000                  FLG3:     .WORD   0                       ;FLAG WORD
000524' 000000                  SANTIM:   .WORD   0                       ;COUNT FOR SANITY TIMER
000526' 000000                  PCSR1:    .WORD   0                       ;COPY OF WHAT GOES TO IPCSR1
000530' 000000  000000          IPCSR2:   .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000534' 000000  000000          PCBADR:   .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000540' 000000                  DMDONE:   .WORD   0                       ;DMA INTERRUPT HAPPENED
000542' 000377                  BAKPAT:   .WORD   377                     ;BACKGROUND PATTERN FOR MEMORY TESTS
```

```

                                .SBTTL  MODULE C, MICRO CLEAR FUNCTION


000544' 105067  177757          CLRC:   CLRB    PCSR1+1                 ;CLEAR THE SELF TEST FIELD
000550' 000241                          CLC
000552' 000207                          RTS     PC
```

```
                                          .SBTTL  MODULE C, MICROTEST #1

000554' 112767  000002  177744  MICC1:   MOVB    #INTST,PCSR1
000562' 016737  177740  021020           MOV     PCSR1,@#IPCSR1
000570' 016737  177734  021010           MOV     IPCSR2,@#MDMA0  ;PICK UP ADDRESS OF PCBB
000576' 016737  177730  021012           MOV     IPCSR2+2,@#MDMA1
000604' 013700  021014                   MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
000610' 013701  021014                   MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
000614' 010037  021010                   MOV     R0,@#MDMA0              ;FETCH CONTENTS OF PCBB+0
000620' 010137  021012                   MOV     R1,@#MDMA1

                                 ;AT THIS POINT MDMAR0 WILL CONTAIN THE INTERRUPT BIT THAT WE ARE TO SET

000624' 013737  021014  021000           MOV     @#MDMAR0,@#IPCSR0       ;SET CORRESPONDING INTERRUPT BIT
000632' 112767  000001  177667           MOVB    #1,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
000640' 000241                           CLC                             ;SUCCESS
000642' 000207                           RTS     PC
```

```
                                          .SBTTL  MODULE C, MICROTEST #2


000644' 112767  000002  177654  MICC2:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000652' 016737  177650  021020          MOV     PCSR1,@#IPCSR1
000660' 005067  177640                  CLR     SANTIM                  ;CLEAR TIMER COUNT
000664' 106427  000200                  MTPS    #PRI04                  ;LOWER PRIORITY TO ALLOW TIMER INTERRUPT
000670' 026727  177630  000012  10$:    CMP     SANTIM,#10.             ;WAIT FOR TEN TICKS OF THE TIMER
000676' 002774                          BLT     10$
000700' 106427  000340                  MTPS    #PRI07                  ;RAISE PRIORITY TO STOP TIMER
000704' 112767  000002  177615          MOVB    #2,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
000712' 016737  177610  021020          MOV     PCSR1,@#IPCSR1
000720' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST WE ARE DONE
000726' 000241                          CLC
000730' 000207                          RTS     PC
```

```
                                        .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

000732' 112767  000002  177566  MICC3:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000740' 016737  177562  021020          MOV     PCSR1,@#IPCSR1
000746' 010700                          MOV     PC,R0                   ;GET ADDRESS OF SUBTEST LIST
000750' 062700  000134                  ADD     #C3STBL-.,R0

000754' 011001                  10$:    MOV     (R0),R1                 ;GET OFFSET FROM ENTRY TO SUBTEST
000756' 001437                          BEQ     30$                     ;NO MORE ENTRIES
000760' 060001                          ADD     R0,R1                   ;CALC ACTUAL SUBTEST ADDRESS
000762' 004711                          JSR     PC,(R1)                 ;GO EXECUTE SUBTEST
000764' 103403                          BCS     20$                     ;ERROR OCCURRED IN SUBTEST
000766' 062700  000002                  ADD     #2,R0                   ;POINT TO NEXT SUBTEST IN LIST
000772' 000770                          BR      10$

000774' 016737  177530  021010  20$:    MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2 AND PCSR3
001002' 016737  177524  021012          MOV     IPCSR2+2,@#MDMA1
001010' 013700  021014                  MOV     @#MDMAR0,R0             ;R0=CONTENTS OF PCSR2
001014' 013702  021014                  MOV     @#MDMAR0,R2             ;R2=CONTENTS OF PCSR3
001020' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
001024' 010237  021012                  MOV     R2,@#MDMA1
001030' 016737  000066  021026          MOV     SUBNUM,@#MDMAW0         ;LOAD PCBB+0 WITH SUBTEST #...
                                                                        ;AND PCBB+1 WITH ERROR TYPE
001036' 010137  021026                  MOV     R1,@#MDMAW0             ;LOAD PCBB+2 WITH FAILING ADDRESS
001042' 010337  021026                  MOV     R3,@#MDMAW0             ;LOAD PCBB+4 WITH GOOD DATA
001046' 010437  021026                  MOV     R4,@#MDMAW0             ;LOAD PCBB+6 WITH BAD DATA
001052' 000261                          SEC                             ;TELL MICROMONITOR THIS TEST FAILED
001054' 000401                          BR      40$

001056' 000241                  30$:    CLC                             ;TELL MICROMONITOR THIS TEST WAS SUCCESSFUL
001060' 112767  000003  177441  40$:    MOVB    #3,PCSR1+1              ;TELL HOST WHICH TEST THIS WAS
001066' 016737  177434  021020          MOV     PCSR1,@#IPCSR1
001074' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST IS FINISHED
001102' 000207                          RTS     PC

001104' 000020                  C3STBL: MICC3A-.
001106' 000160                          MICC3B-.
001110' 000322                          MICC3C-.
001112' 000436                          MICC3D-.
001114' 000704                          MICC3E-.
001116' 001024                          MICC3F-.
001120' 000000                          .WORD   0                       ;END OF LIST
```

```
   342
   3.3  001122'     000              SUBNUM: .BYTE   0                        ;CURRENT SUBTEST # BEING EXECUTED
   344  001123'     000              ERRTYP: .BYTE   0                        ;TYPE OF ERROR, DATA =0, ADDRESS=1
   345
```

```
                                                .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST A

                                                ;*****************************************************************
                                                ;
                                                ;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.
                                                ;
                                                ;*****************************************************************

001124' 010046                          MICC3A: MOV     R0,-(SP)                ;SAVE R0
001126' 112767  000001  177766                  MOVB    #1,SUBNUM               ;TELL WE ARE IN SUBTEST A
001134' 106427  000300                          MTPS    #PRI06                  ;ALLOW PARITY ERRORS

001140' 010600                                  MOV     SP,R0                   ;SAVE THE STACK POINTER FOR A MOMENT
001142' 012706  177774                          MOV     #LINADR+LINSIZ,SP       ;POINT STACK TO TOP OF LINK MEMORY
001146' 012705  037776                          MOV     #LINSIZ/2,R5            ;NUMBER OF WORDS IN LINK MEMORY
001152' 004767  000000                  5$:     JSR     PC,10$                  ;WRITE ADDRESS OF NEXT INSTUCTION
001156' 077503                          10$:    SOB     R5,5$                   ;ALL OVER LINK MEMORY

001160' 010006                                  MOV     R0,SP                   ;RESTORE STACK
001162' 010703                                  MOV     PC,R3                   ;GET DATA THAT WAS WRITTEN TO LINK MEMORY
001164' 062703  177772                          ADD     #10$-.,R3

001170' 012701  100000                          MOV     #LINADR,R1              ;POINT TO BASE OF LINK MEMORY
001174' 011104                          12$:    MOV     (R1),R4                 ;READ DATA
001176' 032767  000004  177316                  BIT     #PARFLG,FLG3            ;DID A PARITY ERROR OCCUR?
001204' 001002                                  BNE     20$                     ;YES

001206' 020304                          15$:    CMP     R3,R4                   ;NO, WAS DATA GOOD?
001210' 001410                                  BEQ     30$                     ;YES
001212' 112767  000003  177306          20$:    MOVB    #INERR,PCSR1            ;TELL HOST THIS SUBTEST FAILED
001220' 112767  000000  177675                  MOVB    #DATERR,ERRTYP          ;TELL MICROMONITOR TYPE OF ERROR
001226' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001230' 000414                                  BR      40$

001232' 062701  000002                  30$:    ADD     #2,R1                   ;POINT TO NEXT ADDRESS
001236' 020127  177774                          CMP     R1,#LINADR+LINSIZ       ;DONE ALL OF LINK MEMORY?
001242' 001354                                  BNE     12$                     ;NOT YET

001244' 012701  100000                          MOV     #LINADR,R1              ;OK NOW FILL WITH ZEROES
001250' 005021                          32$:    CLR     (R1)+
001252' 020127  177774                          CMP     R1,#LINADR+LINSIZ
001256' 001374                                  BNE     32$

001260' 000241                          35$:    CLC                             ;INDICATE SUCCESS
001262' 012600                          40$:    MOV     (SP)+,R0                ;RESTORE R0
001264' 000207                                  RTS     PC

```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST B

                                        ;***************************************************************************
                                        ;
                                        ;ADDRESS SHIFT TEST
                                        ;
                                        ;THIS TEST ASSUMES ALL MEMORY BETWEEN 100000 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                          MICC3B: MOV     R0,-(SP)                        ;SAVE R0
001270' 112767  000002  177624                  MOVB    #2,SUBNUM                       ;TELL HIM WE ARE IN THE SECOND SUBTEST

001276' 012700  100000                          MOV     #LINADR,R0                      ;SET 'WRITTEN TO' ADDRESS
001302' 012701  100000                  10$:    MOV     #LINADR,R1                      ;SET 'READ FROM' ADDERSS
001306' 012710  177777                          MOV     #177777,(R0)                    ;WRITE DATA INTO LOCATION AT BASE...
                                                                                        ;OF A 4K BOUNDARY
001312' 011104                          20$:    MOV     (R1),R4                         ;READ DATA FROM MEMORY ADDRESS WHICH...
                                                                                        ;IS AN EVEN 4K INCREMENT AWAY
001314' 020001                                  CMP     R0,R1                           ;IS 'READ FROM' AND 'WRITTEN TO'
                                                                                        ;ADDRESSES THE SAME?
001316' 001416                                  BEQ     70$                             ;YES, SO DATA IN BOTH SHOULD BE THE SAME
001320' 005003                                  CLR     R3                              ;GOOD DATA IS ZEROS
001322' 005704                                  TST     R4                              ;DATA IN 'READ FROM' MUST BE 0'S
001324' 001430                                  BEQ     50$                             ;OK GO CHANGE 'READ FROM' ADDRESS
                                                                                        ;ERROR OCCURRED BUT WE DON'T KNOW IF...
                                                                                        ;IT IS A DATA ERROR OR AN ADDRESS ERROR
001326' 022704  177777                          CMP     #177777,R4                      ;WAS DATA READ ALL 1'S?
001332' 001015                                  BNE     40$                             ;NO, SO IT WAS A DATA ERROR
001334' 112767  000001  177561                  MOVB    #ADRERR,ERRTYP                  ;YES, IT WAS AN ADDRESS ERROR
001342' 112767  000003  177156                  MOVB    #INERR,PCSR1                    ;TELL HOST THIS TEST FAILED
001350' 000261                                  SEC                                     ;INDICATE FAILURE
001352' 000425                                  BR      60$                             ;LEAVE THIS SUBTEST
001354' 022704  177777                  70$:    CMP     #177777,R4                      ;DATA READ MUST BE ALL 1'S
001360' 001412                                  BEQ     50$                             ;IT IS GOOD
001362' 012703  177777                          MOV     #177777,R3                      ;GOOD DATA IS ONES
001366' 112767  000000  177527          40$:    MOVB    #DATERR,ERRTYP                  ;IT WAS A DATA ERROR
001374' 112767  000003  177124                  MOVB    #INERR,PCSR1                    ;TELL HOST THIS TEST FAILED
001402' 000261                                  SEC                                     ;INDICATE FAILURE
001404' 000410                                  BR      60$                             ;LEAVE THIS SUBTEST

001406' 062701  020000                  50$:    ADD     #20000,R1                       ;CHANGE 'READ FROM' ADDRESS BY 4K
001412' 103337                                  BCC     20$                             ;CONTINUE SAME 'WRITTEN TO' IF NOT PAST 32K
001414' 005010                                  CLR     (R0)                            ;CLEAR OLD 'WRITTEN TO' ADDRESS
001416' 062700  020000                          ADD     #20000,R0                       ;CHANGE 'WRITTEN TO' ADDRESS BY 4K
001422' 103327                                  BCC     10$                             ;CONTINUE IF NOT PAST 32K
001424' 000241                                  CLC                                     ;INDICATE SUCCESS OF THIS SUBTEST
001426' 012600                          60$:    MOV     (SP)+,R0                        ;RESTORE R0
001430' 000207                                  RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST C

                                        ;*****************************************************************************
                                        ;
                                        ;DATA LATCH TEST
                                        ;
                                        ;AT THE FIRST LOCATION OF EACH 4K 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.
                                        ;
                                        ;*****************************************************************************


001432' 010046                          MICC3C: MOV     R0,-(SP)                ;SAVE R0
001434' 112767  000003  177460                  MOVB    #3,SUBNUM               ;TELL WEA ARE IN SUBTEST 'C'
001442' 012701  100000                          MOV     #LINADR,R1              ;GET BASE ADDRESS OF LINK MEMORY
001446' 012700  000001                  1$:     MOV     #1,R0                   ;DATA = 1 IN LEAST SIGNIFICANT BIT
001452' 010002                                  MOV     R0,R2                   ;INDICATE WE ARE SHIFTING A '1'
001454' 010011                          2$:     MOV     R0,(R1)                 ;WRITE LOCATION WITH GOOD DATA
001456' 011104                                  MOV     (R1),R4                 ;READ DATA FROM SAME LOCATION
001460' 020004                          3$:     CMP     R0,R4                   ;IS DATA THE SAME AS WRITTEN?
001462' 001411                                  BEQ     4$                      ;YES, OK
001464' 112767  000000  177431                  MOVB    #DATERR,ERRTYP          ;ERROR IS DATA ERROR
001472' 010003                                  MOV     R0,R3                   ;GOOD DATA
001474' 112767  000003  177024                  MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
001502' 000261                                  SEC                             ;INDICATE THIS SUBTEST FAILED
001504' 000417                                  BR      6$                      ;LEAVE THIS SUBTEST
001506' 005702                          4$:     TST     R2                      ;ARE WE SHIFTING A 1 OR A 0?
001510' 001406                                  BEQ     5$                      ;ZERO
001512' 006300                                  ASL     R0                      ;SHIFT THE ONE OVER
001514' 103357                                  BCC     2$                      ;IF THE '1' HAS NOT BEEN SHIFTED...
                                                                                ;THRU THE 16 POSITIONS CONTINUE WITH 1
001516' 005002                                  CLR     R2                      ;ELSE START SHIFTING A '0'
001520' 012700  177776                          MOV     #177776,R0              ;START WITH LSB = 0 ALL OTHERS 1'S
001524' 000753                                  BR      2$                      ;CONTINUE WITH SHIFTING A 0
001526' 000261                          5$:     SEC                             ;MOVE '0' OVER ONE BIT POSITION
001530' 006100                                  ROL     R0                      ;HAS '0' BEEN IN ALL POSITIONS?
001532' 103750                                  BCS     2$                      ;NO CONTINUE WITH SHIFTING A 0
001534' 062701  020000                          ADD     #SIZ4K,R1               ;CONTINUE TEST AT NEXT 4K BOUNDARY...
001540' 103342                                  BCC     1$                      ;IF NOT PAST 32K
001542' 000241                                  CLC                             ;INDICATE THIS SUBTEST SUCCESSFUL
001544' 012600                          6$:     MOV     (SP)+,R0                ;RESTORE R0
001546' 000207                                  RTS     PC
```

```
                                        .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                          MICC3D: MOV     R0,-(SP)                ;SAVE R0
001552' 112767  000004  177342                  MOVB    #4,SUBNUM               ;TELL WE ARE IN SUBTEST 4
001560' 012767  000377  176754                  MOV     #377,BAKPAT             ;LOAD BAKPAT CONSTANT
001566' 005003                          1$:     CLR     R3                      ;CONTAINS 4K BANK ADDRESS
001570' 012700  100000                  2$:     MOV     #LINADR,R0              ;WRITE LINK MEMORY WITH BACKGROUND...
001574' 016720  176742                  3$:     MOV     BAKPAT,(R0)+            ;PATTERN
001600' 020027  177774                          CMP     R0,#LINADR+LINSIZ
001604' 103773                                  BLO     3$
001606' 016700  176730                          MOV     BAKPAT,R0               ;R0 CONTAINS GOOD DATA

001612' 005002                          4$:     CLR     R2                      ;R2 WILL BE OUR 'WRITTEN TO' ADDRESS
001614' 050302                          6$:     BIS     R3,R2                   ;INDEX INTO THIS 4K BANK
001616' 020227  100000                          CMP     R2,#LINADR              ;IS RESULT LESS THAN LINK MEM BASE
001622' 103453                                  BLO     16$                     ;YES, DON'T USE THIS ADDRESS
001624' 020227  177774                          CMP     R2,#LINADR+LINSIZ       ;IS RESULT LARGER THAN LINK MEM TOP?
001630' 103065                                  BHIS    20$                     ;YES, DON'T USE THIS ADDRESS EITHER
001632' 000312                                  SWAB    (R2)                    ;WRITE COMPLEMENT OF PATTERN
001634' 005001                                  CLR     R1                      ;R1 WILL BE OUR 'READ FROM' ADDRESS
001636' 050301                          7$:     BIS     R3,R1                   ;INDEX INTO THIS 4K BANK
001640' 020127  100000                          CMP     R1,#LINADR              ;IS RESULT LESS THAN LINK MEM BASE?
001644' 103431                                  BLO     12$                     ;YES, DON'T USE THIS ADDRESS
001646' 020127  177774                          CMP     R1,#LINADR+LINSIZ       ;IS RESULT LARGER THAN LINK MEM TOP?
001652' 103036                                  BHIS    15$                     ;YES, DON'T USE THIS ADDRESS EITHER
001654' 011104                                  MOV     (R1),R4                 ;READ DATA
001656' 020102                                  CMP     R1,R2                   ;IS 'READ FROM' AND 'WRITTEN TO' SAME?
001660' 001417                                  BEQ     10$                     ;YES, GO CHECK DATA
001662' 020004                                  CMP     R0,R4                   ;NO, DATA READ SHOULD BE SAME AS BAKPAT
001664' 001421                                  BEQ     12$                     ;IF SO CONTINUE WITH NEW INDEX

001666' 112767  000001  177227  8$:     MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
001674' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
001676' 000261                                  SEC                             ;INDICATE THIS SUBTEST FAILED
001700' 112767  000003  176620                  MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
001706' 000442                                  BR      25$                     ;LEAVE THIS SUBTEST
001710' 020067  176626                          CMP     R0,BAKPAT               ;DOES R0 CONTAIN SWAPPED DATA?
001714' 001405                                  BEQ     12$                     ;NO
001716' 000403                                  BR      11$                     ;YES

001720' 000300                          10$:    SWAB    R0                      ;MAKE GOOD DATA LIKE SWAPPED BAKPAT
001722' 020004                                  CMP     R0,R4                   ;IS DATA READ SAME AS DATA WRITTEN?
001724' 001360                                  BNE     8$                      ;NO, ERROR
```

```
001726' 000300                  11$:    SWAB    R0                      ;RESTORE GOOD DATA TO BAKPAT
001730' 040301                  12$:    BIC     R3,R1                   ;REMOVE 4K BANK SELECT FROM 'READ FROM'
001732' 005701                          TST     R1                      ;DID AN INDEX EXIST?
001734' 001001                          BNE     13$                     ;YES
001736' 005201                          INC     R1                      ;NO, MAKE ONE
001740' 006101                  13$:    ROL     R1                      ;SHIFT INDEX OVER
001742' 020127  020000                  CMP     R1,#SIZ4K               ;IS IT LESS THAN 4K
001746' 103733                          BLO     7$                      ;YES, CONTINUE CHECKING FOR BOGUS DATA
                                                                        ;IN THIS 4K BANK
001750' 000312                  15$:    SWAB    (R2)                    ;RESTORE CONTENTS OF 'WRITTEN TO'
001752' 040302                  16$:    BIC     R3,R2                   ;REMOVE BANK SELECT FROM 'WRITTEN TO'
001754' 005702                          TST     R2                      ;DOES 'WRITTEN TO' CONTAIN AN ADDRESS?
001756' 001001                          BNE     18$                     ;YES
001760' 005202                          INC     R2                      ;NO, MAKE ONE
001762' 006102                  18$:    ROL     R2                      ;SHIFT 'WRITTEN TO' ADDRESS
001764' 020227  020000                  CMP     R2,#SIZ4K               ;IS 'WRITTEN TO' LARGER THAN 4K?
001770' 103711                          BLO     6$                      ;NO
001772' 060203                          ADD     R2,R3                   ;MAKE 'WRITTEN TO' POINT TO NEXT 4K...
001774' 103403                          BCS     20$                     ;IF IT HAS WRAPPED AROUND DO COMPLEMENTED
001776' 020327  177774                  CMP     R3,#LINADR+LINSIZ       ;IS RESULT LARGER THAN LINK MEMORY TOP?
002002' 103703                          BLO     4$                      ;NO
002004' 000367  176532          20$:    SWAB    BAKPAT                  ;REPEAT TEST WITH COMPLEMENTED PATTERN
002010' 001666                          BEQ     1$
002012' 000241                          CLC                             ;INDICATE THIS SUBTEST SUCCESS
002014' 012600                  25$:    MOV     (SP)+,R0                ;RESTORE R0
002016' 000207                          RTS     PC

```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST E

                                        ;****************************************** ********************************
                                        ;
                                        ;ADDRESS BIT SHIFT #2
                                        ;
                                        ;THIS TEST CHECKS FOR DUAL ADDRESSING PROBLEMS BY WRITING EACH LOCATION
                                        ;OF LINK MEMORY WITH ITS ADDRESS AND THEN VERIFYING EACH LOCATION.
                                        ;IT THEN DOES THE SAME THING BUT USES THE COMPLEMENT OF THE ADDRESS
                                        ;AS DATA.
                                        ;
                                        ;***************************************************************************

002020' 010046                          MICC3E: MOV     R0,-(SP)                ;SAVE R0
002022' 112767  000005  177072                  MOVB    #5,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
002030' 005003                                  CLR     R3                      ;FLAG INDICATING ADDRESS IS COMPLEMENTED
002032' 012701  100000                          MOV     #LINADR,R1              ;GET STARTING ADDRESS OF LINK MEMORY
002036' 010100                          1$:     MOV     R1,R0                   ;GET ADDRESS TO WORK WITH
002040' 005703                                  TST     R3                      ;SHOULD WE COMPLEMENT THE DATA TO STORE?
002042' 001401                                  BEQ     2$                      ;NO STORE AS IS
002044' 005100                                  COM     R0                      ;COMPLEMENT DATA
002046' 010021                          2$:     MOV     R0,(R1)+                ;WRITE DATA
002050' 020127  177774                          CMP     R1,#LINADR+LINSIZ       ;IS NEW ADDRESS LARGER THAN LINK MEM?
002054' 103770                                  BLO     1$                      ;NO, KEEP FILLING LINK MEMORY

002056' 014104                          3$:     MOV     -(R1),R4                ;READ DATA STORED
002060' 020004                                  CMP     R0,R4                   ;IS DATA READ SAME AS WRITTEN?
002062' 001411                                  BEQ     4$                      ;YES
002064' 112767  000001  177031                  MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
002072' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
002074' 000261                                  SEC                             ;INDICATE FAILURE
002076' 112767  000003  176422                  MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
002104' 000414                                  BR      10$                     ;LEAVE THIS SUBTEST
002106' 010100                          4$:     MOV     R1,R0                   ;CALC GOOD DATA FOR NEXT LOCATION
002110' 162700  000002                          SUB     #2,R0
002114' 005703                                  TST     R3
002116' 001401                                  BEQ     5$
002120' 005100                                  COM     R0
002122' 020127  100000                  5$:     CMP     R1,#LINADR              ;HAVE WE CHECKED ALL LOCATIONS
002126' 101353                                  BHI     3$                      ;NOT YET
002130' 005103                                  COM     R3                      ;HAVE WE DONE IT COMPLEMENTED?
002132' 001341                                  BNE     1$                      .NO, REPEAT WITH COMPLEMENT
002134' 000241                                  CLC                             ;INDICATE SUCCESS
002136' 012600                          10$:    MOV     (SP)+,R0                ;RESTORE R0
002140' 000207                                  RTS     PC
```

```
                                       .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST F

                                       ;**********************************************************************
                                       ;
                                       ;MARCH TEST
                                       ;
                                       ;THIS TEST WRITES A BACKGROUND PATTERN IN ALL OF LINK MEMORY
                                       ;
                                       ;1-READ EVERY LOCATION FOR CORRECT DATA, SWAPS BYTES AT EACH LOCATION
                                       ;AND PROCEED IN MAX TO MIN DIRECTION
                                       ;2-READ EVERY LOCATION FOR SWAPPED DATA, WRITES ORIGINAL PATTERN IN EACH
                                       ;LOCATION AND PROCEEDS IN MIN TO MAX DIRECTION
                                       ;3-REPEAT 1 GOING IN MIN TO MAX DIRECTION
                                       ;4-REPEAT 2 GOING IN MAX TO MIN DIRECTION
                                       ;
                                       ;**********************************************************************

002142' 010046                         MICC3F: MOV     R0,-(SP)                ;SAVE R0
002144' 112767  000006  176750                 MOVB    #6,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
002152' 005003                                 CLR     R3                      ;ADDRESS DIRECTION FLAG 0 = MIN.->MAX
002154' 012701  100000                 10$:    MOV     #LINADR,R1              ;FILL LINK MEMORY WITH BACKGROUND PATTERN
002160' 012700  000377                         MOV     #377,R0                 ;BACKGROUND PATTERN=LOW BYTE ALL 1'S
002164' 010021                         12$:    MOV     R0,(R1)+
002166' 020127  177774                         CMP     R1,#LINADR+LINSIZ
002172' 103774                                 BLO     12$

002174' 014104                         20$:    MOV     -(R1),R4                ;STARTING FROM THE TOP, READ DATA
002176' 020004                                 CMP     R0,R4                   ;R0 = GOOD DATA, R4 = DATA READ
002200' 001411                                 BEQ     30$                     ;IF SAME OK
002202' 112767  000000  176713                 MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002210' 010003                                 MOV     R0,R3                   ;GET GOOD DATA
002212' 000261                                 SEC                             ;INDICATE FAILURE
002214' 112767  000003  176304                 MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
002222' 000462                                 BR      200$                    ;LEAVE THIS SUBTEST
002224' 000300                         30$:    SWAB    R0                      ;NEW GOOD DATA PATTERN
002226' 010011                                 MOV     R0,(R1)                 ;STORE AT SAME PLACE
002230' 011104                                 MOV     (R1),R4                 ;READ IT BACK
002232' 020400                                 CMP     R4,R0                   ;R0=GOOD DATA, R4=DATA READ
002234' 001411                                 BEQ     40$                     ;IF SAME OK
002236' 112767  000000  176657                 MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002244' 010003                                 MOV     R0,R3                   ;GET GOOD DATA
002246' 000261                                 SEC                             ;FAILURE
002250' 112767  000003  176250                 MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
002256' 000444                                 BR      200$                    ;LEAVE THIS SUBTEST
002260' 000300                         40$:    SWAB    R0                      ;SWITCH GOOD DATA AGAIN
002262' 001032                                 BNE     90$                     ;IF ORIGINAL PATTERN THEN WE ARE...
                                                                               ;READING THE MEMORY TO CONTAIN A...
                                                                               ;BACKGROUND OF LOW BYTE = ALL 1'S...
                                                                               ;AND WRITING IT BACK SWAPPED AND...
                                                                               ;VERIFING SWAPPED DATA

002264' 005703                         50$:    TST     R3                      ;
002266' 001032                                 BNE     100$                    ;
002270' 062701  000002                 60$:    ADD     #2,R1                   ;WE ARE GOING MIN->MAX SO ADJUST POINTER
002274' 020127  177774                         CMP     R1,#LINADR+LINSIZ       ;WE AT MAX?
002300' 103015                                 BHIS    80$                     ;YES
002302' 011104                         70$:    MOV     (R1),R4                 ;READ DATA
```

```
002304' 020004                             CMP     R0,R4                   ;R0=GOOD DATA, R4=DATA READ
002306' 001746                             BEQ     30$                     ;OK IF SAME
002310' 112767  000000  176605             MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002316' 010003                             MOV     R0,R3                   ;GET GOOD DATA
002320' 000261                             SEC                             ;INDICATE FAILURE
002322' 112767  000003  176176             MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
002330' 000417                             BR      200$                    ;LEAVE THIS SUBTEST
002332' 000734                             BR      30$
002334' 000300                     80$:    SWAB    R0                      ;SWITCH GOOD DATA
002336' 001716                             BEQ     20$                     ;IF LOW BYTE = ALL 0'S GO IN MAX->MIN
002340' 005103                             COM     R3                      ;ELSE GO IN MIN->MAX DIRECTION
002342' 012701  100000                     MOV     #LINADR,R1              ;SET ADDRESS POINTER TO MIN
002346' 000755                             BR      70$

002350' 005703                     90$:    TST     R3                      ;ARE WE GOING MIN->MAX?
002352' 001344                             BNE     50$                     ;YES

002354' 020127  100000             100$:   CMP     R1,#LINADR              ;NO, CHECK TO SEE IF WE ARE AT MIN
002360' 101305                             BHI     20$                     ;NOT YET
002362' 000300                             SWAB    R0                      ;WE ARE AT MIN SO SWITCH GOOD DATA
002364' 001746                             BEQ     70$                     ;IF LOW BYTE = ALL 0'S ELSE WE ARE DONE
002366' 000241                             CLC                             ;INDICATE SUCCESS
002370' 012600                     200$:   MOV     (SP)+,R0                ;RESTORE R0
002372' 000207                             RTS     PC

```

```
                                        .SBTTL  MODULE C, MICROTEST #4

002374' 112767  000002  176124  MICC4:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
002402' 016737  176120  021020          MOV     PCSR1,@#IPCSR1
002410' 016737  176114  021010          MOV     IPCSR2,@#MDMA0          ;GET PCBB ADDRESS THROUGH HOST'S PCSR2
002416' 016737  176110  021012          MOV     IPCSR2+2,@#MDMA1
002424' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
002430' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
002434' 010037  021010                  MOV     R0,@#MDMA0              ;SETUP TO READ PCBB+0
002440' 010137  021012                  MOV     R1,@#MDMA1
002444' 013737  021014  021004          MOV     @#MDMAR0,@#DMAT0        ;GET DATA PATTERN AND WRITE IT TO...
                                                                        ;DMA 'TO' ADDRESS REGISTER
002452' 013737  021004  021026          MOV     @#DMAT0,@#MDMAW0        ;READ DATA PATTERN BACK AND WRITE...
                                                                        ;BACK INTO HOST MEMORY
002460' 112767  000004  176041          MOVB    #4,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
002466' 016737  176034  021020          MOV     PCSR1,@#IPCSR1
002474' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST WE ARE DONE
002502' 000241                          CLC
002504' 000207                          RTS     PC
```

```
                                         .SBTTL  MODULE C, MICROTEST #5


        002506' 112767  000002  176012  MICC5:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
        002514' 016737  176006  021020          MOV     PCSR1,@#IPCSR1
        002522' 012700  100000                  MOV     #LINADR,R0              ;BASE ADDRESS OF LINK MEMORY
        002526' 012701  037776                  MOV     #LINSIZ/2,R1            ;NUMBER OF WORDS IN LINK MEMORY
        002532' 010010                  10$:    MOV     R0,(R0)                 ;FILL LINK MEMORY WITH THE ADDRESS
        002534' 005720                          TST     (R0)+
        002536' 005301                          DEC     R1                      ;OF EACH LOCATION
        002540' 001374                          BNE     10$
        002542' 016737  175762  021010          MOV     IPCSR2,@#MDMA0          ;GET PCBB THROUGH HOST'S PCSR2
        002550' 016737  175756  021012          MOV     IPCSR2+2,@#MDMA1
        002556' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
        002562' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
        002566' 010037  021010                  MOV     R0,@#MDMA0              ;R0 HAS ADDRESS OF PCBB+0
        002572' 010137  021012                  MOV     R1,@#MDMA1              ;R1 HAS ITS HIGH ORDER BITS
        002576' 013737  021014  021022          MOV     @#MDMAR0,@#DMAF         ;LOAD DMA 'FROM' ADDRESS REGISTER
                                                                                ;WITH ADDRESS SUPPLIED FROM HOST
        002604' 062700  000002                  ADD     #2,R0                   ;LOAD DMA 'TO' ADDRESS REGISTER
        002610' 005501                          ADC     R1                      ;WITH PCBB+2 ADDRESS
        002612' 010037  021004                  MOV     R0,@#DMAT0
        002616' 010137  021006                  MOV     R1,@#DMAT1
        002622' 012737  000002  021024          MOV     #2,@#DMAWC              ;LOAD WORD COUNT TO TRANSFER 1 WORD
        002630' 005067  175704                  CLR     DMDONE                  ;CLEAR DMA DONE FLAG
        002634' 005237  021002                  INC     @#DMACSR                        ;START DMA ENGINE
        002640' 106427  000240                  MTPS    #PRI05                  ;LOWER CPU PRIORITY
        002644' 005767  175670          20$:    TST     DMDONE                  ;IS THE TRANSFER COMPLETE?
        002650' 001775                          BEQ     20$                     ;NO, WAIT FOR THE INTERRUPT
        002652' 106427  000340                  MTPS    #PRI07
        002656' 112767  000005  175643          MOVB    #5,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
        002664' 016737  175636  021020          MOV     PCSR1,@#IPCSR1
        002672' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST WE ARE DONE
        002700' 000241                          CLC
        002702' 000207                          RTS     PC
```

```
                                         .SBTTL  MODULE C, MICROTEST #6


002704' 112767  000002  175614  MICC6:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
002712' 016737  175610  021020          MOV     PCSR1,@#IPCSR1
002720' 012700  100000                  MOV     #LINADR,R0              ;GET BASE ADDRESS OF LINK MEMORY
002724' 012701  037776                  MOV     #LINSIZ/2,R1            ;SIZE OF LINK MEMORY IN WORDS
002730' 010010                  10$:    MOV     R0,(R0)                 ;FILL LINK MEMORY WITH ADDRESS OF
002732' 005720                          TST     (R0)+
002734' 005301                          DEC     R1                      ;EACH LOCATION
002736' 001374                          BNE     10$
002740' 016737  175564  021010          MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2
002746' 016737  175560  021012          MOV     IPCSR2+2,@#MDMA1
002754' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
002760' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS O' HOST'S PCSR3
002764' 010037  021010                  MOV     R0,@#MDMA0              ;SETUP TO READ CONTENTS OF PCBB+0
002770' 010137  021012                  MOV     R1,@#MDMA1
002774' 013737  021014  021004          MOV     @#MDMAR0,@#DMATO        ;SETUP DMA 'TO' REGISTER FROM PCBB+0
003002' 012737  100000  021022          MOV     #LINADR,@#DMAF          ;SETUP DMA 'FROM' WITH LINK MEM BASE
003010' 012737  003774  021024          MOV     #3774,@#DMAWC           ;SETUP MAXIMUM BYTE COUNT
003016' 005067  175516                  CLR     DMDONE
003022' 005237  021002                  INC     @#DMACSR                ;TURN ON DMA ENGINE
003026' 106427  000240                  MTPS    #PRI05                  ;ALLOW A DMA DONE INTERRUPT
003032' 005767  175502          20$:    TST     DMDONE                  ;WAIT FOR THE INTERRUPT FROM THE DMA
003036' 001775                          BEQ     20$                     ;ENGINE
003040' 112767  000006  175461          MOVB    #6,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
003046' 016737  175454  021020          MOV     PCSR1,@#IPCSR1
003054' 01273.  004000  021000          MOV     #CNI,@#IPCSR0           ;TELL HOST THAT THIS TEST IS DONE
003062' 000241                          CLC
003064' 000207                          RTS     PC
```

```
                                        ;
                                        .SBTTL  <DMA "TO" REGISTER RIPPLE TEST>
                                        ;
                                        ;
003066' 112767  000002  175432          MICC7:: MOVB    #INTST,PCSR1
003074' 016737  175426  021020                  MOV     PCSR1,@#IPCSR1
                                        ;
                                        ;
003102' 012700  100004                          MOV     #LINADR+4,R0
003106' 012720  052525                          MOV     #052525,(R0)+
003112' 012710  031463                          MOV     #031463,(R0)
                                        ;
                                        ;
003116' 016737  175406  021010                  MOV     IPCSR2,@#MDMA0
003124' 016737  175402  021012                  MOV     IPCSR2+2,@#MDMA1
003132' 013700  021014                          MOV     @#MDMAR0,R0
003136' 013701  021014                          MOV     @#MDMAR0,R1
                                        ;
                                        ;
003142' 010037  021004                          MOV     R0,@#DMAT0
003146' 012737  100004  021022                  MOV     #LINADR+4,@#DMAF
003154' 012737  000004  021024                  MOV     #4,@#DMAWC
003162' 010700                                  MOV     PC,R0
003164' 062700  175222                          ADD     #DMAINT-.,R0
003170' 010037  000114                          MOV     R0,@#DMAVEC
003174' 012737  000300  000116                  MOV     #PRI06,@#DMAVEC+2
003202' 005067  175332                          CLR     DMDONE
003206' 005237  021002                          INC     @#DMACSR
003212' 106427  000240                          MTPS    #PRI05
003216' 005767  175316                  20$:    TST     DMDONE
003222' 001775                                  BEQ     20$
                                        ;
                                        ;
003224' 112767  000007  175275                  MOVB    #7,PCSR1+1
003232' 016737  175270  021020                  MOV     PCSR1,@#IPCSR1
003240' 012737  004000  021000                  MOV     #DNI,@#IPCSR0
003246' 000241                                  CLC
003250' 000207                                  RTS     PC

003252' 003254                          MICCSZ::MICCSZ-MICROC+2                 ;SIZE OF MICROCODE MODULE C

        000001                          .END
```

```
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#      150*      151*
CSRWRT  000352R         002       149       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       144       211#      812
DMAT0 = 021004                     10#      714*      716       745*      776*      808*
DMAT1 = 021006                     11#      746*
DMAVEC= 000114                     75#      145*      146*      813*      814*
DMAWC = 021024                     16#      747*      778*      810*
DMDONE  000540R         002       211*      212       217       244#      748*      751       779*      782       815*      818
DNI   = 004000                     69#      129       283       332       720       756       786       824
ERRFLG= 000002                     82#      202
ERRINT  000362R         002       131       202#
ERRTYP  001123R         002       344#      378*      428*      435*      473*      538*      605*      652*      662*      681*
FATI  = 000400                     71#
FLG3    000522R         002       166*      169       173       193*      199*      202*      225*      239#      372
INERR = 000003                     80#      191       377       429       436       475       541       608       655       665       684
INMON = 000001                     78#      127       189
INTST = 000002                     79#      256       274       302       706       727       762       793
IOADR = 020000                     97#       98
IOSIZ = 020000                     93#       98
IPCSR0= 021000                      8#      129*      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       164*      165*      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
```

```
                             764     777     797     809
LINSIZ= 077774                95#    102     361     362     383     388     515     523     530     569     599     646     676
                             730     765
LRBUF = 021044                26#
LTAC  = 021030                20#
MDMAR0= 021014                14#    260     261     267     317     318     710     711     714     737     738     741     772
                             773     776     804     805
MDMAR1= 021016                15#
MDMAW0= 021026                19#    321*    323*    324*    325*    716*
MDMAW1= 021036                23#
MDMA0 = 021010                12#    258*    262*    315*    319*    708*    712*    735*    739*    770*    774*    802*
MDMA1 = 021012                13#    259*    263*    316*    320*    709*    713*    736*    740*    771*    775*    803*
MDMSR = 021042                25#
MICCSZ  003252RG    002      828#
MICC1   000554R     002      231     256#
MICC2   000644R     002      232     274#
MICC3   000732R     002      233     302#
MICC3A  001124R     002      335     356#
MICC3B  001266R     002      336     409#
MICC3C  001432R     002      337     464#
MICC3D  001550R     002      338     509#
MICC3E  002020R     002      339     590#
MICC3F  002142R     002      340     640#
MICC4   002374R     002      234     706#
MICC5   002506R     002      235     727#
MICC6   002704R     002      236     762#
MICC7   003066RG    002      237     793#
MICROC  000000RG    002      124#    828
MODREG= 177774               102#    103     104
NPRERR= 100000                86#    219
NPRFLG= 000020                85#
NXMERR= 040000                87#    214
NXMFLG= 000010                84#
PARERR= 010000                89#    226
PARFLG= 000004                83#    225     372
PARINT  000464R     002      139     225#
PARVEC= 000140                76#    140*    141*
PCBADR  000534R     002      243#
PCEI  = 040000                66#
PCSRSW= 021040                24#    158
PCSR1   000526R     002      127*    128     189*    191*    192     241#    251*    256*    257     268*    274*    275     281*
                             282     302*    303     330*    331     377*    429*    436*    475*    541*    608*    655*    665*
                             684*    706*    707     718*    719     727*    728     754*    755     762*    763     784*    785
                             793*    794     822*    823
PHYAD0= 021060                27#
PHYAD1= 021062                28#
PHYAD2= 021064                29#
PHYAD3= 021066                30#
PHYAD4= 021070                31#
PHYAD5= 021072                32#
PRI00 = 000000                61#    167*
PRI01 = 000040                60#
PRI02 = 000100                59#
PRI03 = 000140                58#
PRI04 = 000200                57#    151     277*
PRI05 - 000240                56#    156     750*    781*    817*
```

```
PRI06 = 000300            55#     146     358*    814
PRI07 = 000340            54#     124*    134     141     172*    280*    753*
RCEI  = 002000            70#
ROMADR= 040000            98#      99
ROMSIZ= 040000            94#      99
RXI   = 020000            67#
SANTIM  000524R    002   208*     240#    276*    278
SANVEC= 000134            73#     155*    156*
SERI  = 100000            65#
SIZ4K = 020000            90#      91      92      93     489     556     565
SIZ8K = 040000            91#      94      95
STACK = 001000            77#     126
SUBNUM  001122R    002   321      343#    357*    410*    465*    510*    591*    641*
TBLC    000502R    002   183      230#
TIMINT  000400R    002   154      208#
TXI   = 010000            68#
UNIERR= 020000            88#     203
WCSADR= 000000            96#      97
WCSSIZ= 020000            92#      97
.     = 003254R    002   131      139     144     149     154     175     183     205     221     230     231     232     233
                         234      235     236     237     305     335     336     337     338     339     340     368     812

. ABS.  000000     000
        000000     001
MICRC   003254     002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROC.OBJ,MICROC.LST/CRF/DOC/NL:TOC/SOL=MICROC.MAC
 RUN-TIME: 1 1 .3 SECONDS
 RUN-TIME RATIO: 31/3=9.6
 CORE USED:  8K  (16 PAGES)

 DOCUMENT PAGES:  25
```

```
                                .TITLE  MICROD - MICROCODE MODULE D

        000000'                 .CSECT MICRD

                                .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000                  IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002                  DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004                  DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006                  DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010                  MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012                  MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014                  MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016                  MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020                  IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022                  DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024                  DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026                  MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030                  LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032                  LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034                  CLRFIF  =       21034   ;CLEAR FIFO
        021036                  MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040                  PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042                  MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044                  LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060                  PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062                  PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064                  PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066                  PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070                  PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072                  PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774                  MODREG  =       177774  ;LINK MODE REGISTER
        177774                  ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776                  CMDREG  =       177776  ;LINK COMMAND REGISTER

                                .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000                  BIT15   =       100000
        040000                  BIT14   =       40000
        020000                  BIT13   =       20000
        010000                  BIT12   =       10000
        004000                  BIT11   -       4000
        002000                  BIT10   -       2000
        001000                  BIT9    -       1000
        000400                  BIT8    =       400
        000200                  BIT7    =       200
        000100                  BIT6    =       100
        000040                  BIT5    =       40
        000020                  BIT4    -       20
        000010                  BIT3    =       10
        000004                  BIT2    =       4
        000002                  BIT1    =       2
        000001                  BIT0    -       1
                                ;
        012400                  LASFTP  -       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340          PRI07   =       340
        000300          PRI06   =       300
        000240          PRI05   -       240
        000200          PRI04   =       200
        000140          PRI03   =       140
        000100          PRI02   =       100
        000040          PRI01   =       40
        000000          PRI00   =       0
                        ;
                        ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                        ;
        100000          SERI    =       BIT15
        040000          PCEI    =       BIT14
        020000          RXI     -       BIT13
        010000          TXI     =       BIT12
        004000          DNI     =       BIT11
        002000          RCEI    =       BIT10
        000400          FATI    =       BIT8
                        ;
                        ;LINK COMMAND REGISTER
                        ;
        100000          ENABLE  =       BIT15   ;ENABLE LINK MODULE
        000200          MODE    =       BIT7    ;ENABLE MODE REGISTER
        000100          ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                        ;
                        ;LINK MODE REGISTER
                        ;
        100000          PROM    =       BIT15   ;PROMISCUIOUS MODE
        040000          ENAL    =       BIT14   ;ENABLE MULTICAST
        004000          ENCR    =       BIT11   ;ENABLE COLLISION TEST
        002000          ACLO    =       BIT10   ;ENABLE ACLO
        000040          DRTY    -       BIT5    ;DISABLE RETRY LOGIC
        000020          COLL    =       BIT4    ;SIMULATE A COLLISION
        000010          DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
        000004          LOOP    =       BIT2    ;ENABLE LOOPBACK

        000070          TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
        000120          RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
        000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000          STACK=  1000            ;STACK LOCATION
        000001          INMON=  1               ;IN MICROMONITOR STATE
        000002          INTST=  2               ;IN A TEST STATE
        000003          INERR=  3               ;IN ERROR STATE
        000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
        000040          TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
        000100          RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
        100000          NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR ERROR OCCURRED
        040000          NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000          UNIERR= BIT13           ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
        010000          PARERR= BIT12           ;PCSR0 FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        004000          SIZ1K=  4000            ;1K WORDS
        010000          SIZ2K=  SIZ1K*2         ;2K WORDS
        014000          SIZ3K=  SIZ1K*3         ;3K WORDS
        020000          SIZ4K=  SIZ2K*2         ;4K WORDS
        040000          SIZ8K=  SIZ4K*2         ;8K WORDS
        020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774          LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
        000000          WCSADR= 0               ;BASE ADDRESS OF WCS
        020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        002756          MAXBC=  1518.           ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
        007777          MXMTBC= 7777            ; MAXIMUM COUNT IN LINK XMIT BYTE COUNT FIELD
        007777          MRECBC= 7777            ; MAXIMUM COUNT IN LINK RECEIVE "MLEN" FIELD
        000004          CRCSIZ= 4               ;NUMBER OF BYTES IN THE CRC
        000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED

```

```

                                        .SBTTL  MICROCODE MODULE D

 000000' 106427  000340                 MICROD::MTPS    #PRI07                  ;DISABLE INTERRUPTS
 000004' 012737  000000  177776                 MOV     #0,@#CMDREG             ;TURN OFF THE LINK
 000012' 012706  001000                         MOV     #STACK,SP               ;SETUP STACK
 000016' 112767  000001  000620                 MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
 000024' 016737  000614  021020                 MOV     PCSR1,@#IPCSR1
 000032' 012737  004000  021000                 MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
 000040' 010700                                 MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
 000042' 062700  000404                         ADD     #ERRINT-.,R0            ;HANDLER
 000046' 005001                                 CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
 000050' 010021                         10$:    MOV     R0,(R1)+                ;HANDLER
 000052' 012721  000340                         MOV     #PRI07,(R1)+
 000056' 020127  001000                         CMP     R1,#1000
 000062' 002772                                 BLT     10$

 000064' 010700                                 MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
 000066' 062700  000462                         ADD     #PARINT-.,R0
 000072' 010037  000140                         MOV     R0,@#PARVEC
 000076' 012737  000340  000142                 MOV     #PRI07,@#PARVEC+2

 000104' 010700                                 MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
 000106' 062700  000364                         ADD     #DMAINT-.,R0
 000112' 010037  000114                         MOV     R0,@#DMAVEC
 000116' 012737  000340  000116                 MOV     #PRI07,@#DMAVEC+2

 000124' 010700                                 MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
 000126' 062700  000310                         ADD     #CSRWRT-.,R0
 000132' 010037  000064                         MOV     R0,@#CSRVEC
 000136' 012737  000200  000066                 MOV     #PRI04,@#CSRVEC+2

 000144' 010700                                 MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
 000146' 062700  000316                         ADD     #TIMINT-.,R0
 000152' 010037  000134                         MOV     R0,@#SANVEC
 000156' 012737  000240  000136                 MOV     #PRI05,@#SANVEC+2

 000164' 010700                                 MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
 000166' 062700  000414                         ADD     #TRNINT-.,R0
 000172' 010037  000070                         MOV     R0,@#TRNVEC
 000176' 012737  000200  000072                 MOV     #PRI04,@#TRNVEC+2

 000204' 010700                                 MOV     PC,R0                   ;SETUP RECEIVER VECTOR
 000206' 062700  000360                         ADD     #RCVINT-.,R0
 000212' 010037  000120                         MOV     R0,@#RCVVEC
 000216' 012737  000240  000122                 MOV     #PRI05,@#RCVVEC+2

 000224' 013700  021040                         MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
 000230' 052700  176000                         BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
 000234' 006300                                 ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
 000236' 006300                                 ASL     R0
 000240' 006300                                 ASL     R0
 000242' 062700  000004                         ADD     #4,R0                   ;PCSR2 IS PCSR0+4
 000246' 010067  000374                         MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
 000252' 012767  000003  000370                 MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
 000260' 005067  000354                         CLR     FLG4                    ;INITIALIZE FLAG WORD
```

```
000264' 106427  000000          15$:    MTPS    #PRI00          ;ALLOW INTERRUPTS

000270' 005767  000344          20$:    TST     FLG4            ;WAIT FOR A COMMAND FROM HOST
000274' 001775                          BEQ     20$

000276' 106427  000340                  MTPS    #PRI07          ;RAISE CPU PRIORITY TO SERVICE COMMAND
000302' 032767  000001  000330          BIT     #CSRFLG,FLG4    ;DID HOST GIVE US A COMMAND?
000310' 001001                          BNE     30$             ;YES
000312' 000777                          BR      .               ;NO, ERROR SO JUST SIT HERE...
                                                                ;FOR LACK OF ANYTHING BETTER TO DO

000314' 113700  021000          30$:    MOVB    @#IPCSR0,R0     ;GET WHAT HOST WROTE TO PCSR0
000320' 042700  177760                  BIC     #177760,R0      ;STRIP ALL BUT COMMAND BITS
000324' 001004                          BNE     35$             ;WAS IT THE CLEAR FUNCTION?
000326' 012737  000001  021020          MOV     #INMON,@#IPCSR1 ;YES, CLEAR OUT THE TEST # BITS
000334' 000432                          BR      50$
000336' 022700  000017          35$:    CMP     #17,R0          ;START OPERATIONAL MICROCODE?
000342' 001432                          BEQ     60$
000344' 162700  000001                  SUB     #1,R0
000350' 010701                          MOV     PC,R1           ;GET ADDRESS OF OUR COMMAND TABLE
000352' 062701  000240                  ADD     #TBLD-.,R1
000356' 006300                          ASL     R0              ;MAKE COMMAND A BYTE OFFSET
000360' 060001                          ADD     R0,R1           ;USE IT TO INDEX INTO COMMAND TABLE
000362' 061101                          ADD     (R1),R1         ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000364' 004711                          JSR     PC,(R1)         ;EXECUTE AS COMMANDED FROM HOST
000366' 103404                          BCS     40$             ;ERROR OCCURRED
000370' 112767  000001  000246          MOVB    #INMON,PCSR1    ;INDICATE TO HOST WE ARE BACK IN...
000376' 000403                          BR      45$             ;MICROMONITR
000400' 112767  000003  000236  40$:    MOVB    #INERR,PCSR1    ;INDICATE TO HOST ERROR OCCURRED
000406' 016737  000232  021020  45$:    MOV     PCSR1,@#IPCSR1
000414' 012737  004000  021000          MOV     #DNI,@#IPCSR0   ;TELL HOST THIS MICROTEST FINISHED
000422' 005067  000212          50$:    CLR     FLG4            ;RESET FLAG WORD
000426' 000716                          BR      15$             ;GO WAIT FOR ANOTHER COMMAND

000430' 005000                  60$:    CLR     R0              ;FAKE SELF TEST RESULTS
000432' 000137  040006                  JMP     @#40006         ;START OPERATIONAL MICROCODE

000436' 052767  000001  000174  CSRWRT: BIS     #CSRFLG,FLG4    ;INDICATE A CSR WRITE INTERRUPT OCCURED
000444' 000002                          RTI

000446' 052767  000002  000164  ERRINT: BIS     #ERRFLG,FLG4    ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000454' 012737  020000  021000          MOV     #UNIERR,@#IPCSR0 ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                ;HAPPENED
000462' 000777                          BR      .               ;JUST SIT HERE AND SPIN WHEELS
                                                                ;COUNT ON HOST TO TIME OUT

000464' 005267  000152          TIMINT: INC     SANTIM          ;COUNT TICKS AS THEY OCCUR
000470' 000002                          RTI

000472' 013767  021002  000156  DMAINT: MOV     @#DMACSR,DMDONE ;GET DMA STATUS
000500' 032767  040000  000150          BIT     #BIT14,DMDONE   ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000506' 001404                          BEQ     10$             ;NO
000510' 012737  040000  021000          MOV     #NXMERR,@#IPCSR0 ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                ;LOCATION WAS ADDRESSED
000516' 000407                          BR      20$
000520' 032767  100000  000130  10$:    BIT     #BIT15,DMDONE   ;DID A NPR TIMEOUT OCCUR?
```

```
000526' 001407                           BEQ    30$                   ;NO
000530' 012737  100000  021000           MOV    #NPRERR,@#IPCSR0      ;TELL HOST NPR TIMEOUT HAPPENED
000536' 012737  100000  021002  20$:     MOV    #BIT15,@#DMACSR       ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000544' 000777                           BR     .                     ;SIT HERE AND SPIN WHEELS
000546' 000002                  30$:     RTI


000550' 052767  000004  000062  PARINT:  BIS    #PARFLG,FLG4          ;SET PARITY ERROR OCCURRED
000556' 012737  010000  021000           MOV    #PARERR,@#IPCSR0      ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                      ;OCCURRED
000564' 000777                           BR     .                     ;SIT HERE AND SPIN WHEELS

000566' 005737  021044          RCVINT:  TST    @#LRBUF               ;READ BUFFER DONE REGISTER...
                                                                      ;WHICH CLEARS THE INTERRUPT
000572' 052767  000100  000040           BIS    #RCVFLG,FLG4          ;SET RECEIVER INTERRUPT OCCURRED
000600' 000002                           RTI

000602' 052767  000040  000030  TRNINT:  BIS    #TRNFLG,FLG4          ;SET TRANSMITTER INTERRUPT OCCURRED
000610' 000002                           RTI

000612' 000056                  TBLD:    .WORD  MICD1-.               ;TRANSMITTER DONE TEST
000614' 000236                           .WORD  MICD2-.               ;RECEIVER DONE TESTS
000616' 000432                           .WORD  MICD3-.               ; DATA BYTE FRAMING TEST
000620' 000734                           .WORD  MICD4-.               ; DATA WORD FRAMING TEST
000622' 001244                           .WORD  MICD5-.               ; DATA PATH PATTERN
000624' 001604                           .WORD  MICD6-.               ; STATUS MUX TEST
000626' 002046                           .WORD  MICD7-.               ; LINK BYTE COUNT TEST
000630' 002354                           .WORD  MICD8-.               ;LINK MEMORY ARBITRATION TEST
000632' 003014                           .WORD  MICD9-.               ;LINK BYTE COUNTER MAXIMUM TEST
000634' 003206                           .WORD  MICD10-.              ;FIFO TEST
000636' 003460                           .WORD  MICD11-.              ;LINK MEMORY ADDRESS TEST

000640' 000000                  FLG4:    .WORD  0                     ;FLAG WORD
000642' 000000                  SANTIM:  .WORD  0                     ;COUNT FOR SANITY TIMER
000644' 000000                  PCSR1:   .WORD  0                     ;COPY OF WHAT GOES TO PCSR1
000646' 000000  000000          IPCSR2:  .WORD  0,0                   ;ADDRESS IN HOST MEMORY FOR PCSR2
000652' 000000  000000          PCBADR:  .WORD  0,0                   ;ADDRESS IN HOST MEMORY FOR PCB
000656' 000000                  DMDONE:  .WORD  0
000660' 000000                  RBUFP:   .WORD  0                     ;POINTER TO RECIEVE BUFFFR
000662' 000000                  TBUFP:   .WORD  0                     ;POINTER TO XMIT BUFFER
000664' 000000                  DBUFP:   .WORD  0                     ;POINTER TO DMA ENGINE BUFFER
000666' 000000                  MBUFP:   .WORD  0                     ;POINTER TO MICROCPU DMA BUFFER
```

```
000670' 112767  000002  177746  MICD1:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000676' 016737  177742  021020          MOV     PCSR1,@#IPCSR1
000704' 012703  177777                  MOV     #177777,R3              ;FILL RECIEVE BUFFER WITH 1'S
000710' 012700  100000                  MOV     #LINADR,R0              ;RECIEVE BUFFER STARTS HERE
000714' 010067  177740                  MOV     R0,RBUFP
000720' 010320                  10$:    MOV     R3,(R0)+
000722' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ;FILL ENTIRE BUFFER
000726' 103774                          BLO     10$
000730' 005003                          CLR     R3                      ;FILL XMIT BUFFER WITH 0'S
000732' 010067  177724                  MOV     R0,TBUFP                ;XMIT BUFFER STARTS 'K AWAY FROM RECIEVE
000736' 010320                  20$:    MOV     R3,(R0)+
000740' 020027  110000                  CMP     R0,#LINADR+SIZ2K
000744' 103774                          BLO     20$

000746' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
000754' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ;SET PROMISCUIOUS MODE AND ENABLE LOOPBACK

000762' 016701  177674                  MOV     TBUFP,R1                ;POINT TO XMIT BUFFER
000766' 005021                          CLR     (R1)+                   ;CLEAR OUT STATUS WORD
000770' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ;SET BYTE COUNT TO MAXIMUM ALLOWED
000774' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001000' 005067  177634                  CLR     FLG4                    ;CLEAR THE INTERRUPT FLAG
001004' 016737  177650  021032          MOV     RBUFP,@#LFRBUF          ;TELL LINK WHERE RECIEVE BUFF IS
001012' 016737  177644  021030          MOV     TBUFP,@#LTAC            ;TELL LINK WHERE TRANSMIT BUFF IS...
                                                                        ;WHICH WILL START A XMIT OPERATION
001020' 106427  000140                  MTPS    #PRI03                  ;ALLOW XMITTER TO INTERRUPT
001024' 032767  000040  177606  30$:    BIT     #TRNFLG,FLG4            ;WAIT FOR XMIT INTERRUPT
001032' 001774                          BEQ     30$
001034' 106427  000340                  MTPS    #PRI07                  ;DISABLE INTERRUPTS
001040' 112767  000001  177577          MOVB    #1,PCSR1+1              ;TELL HOST THIS IS TEST #1
001046' 000241                          CLC
001050' 000207                          RTS     PC
```

```
001C52' 112767  000002  177564  MICD2:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
001060' 016737  177560  021020          MOV     PCSR1,@#IPCSR1
001066' 012703  177777                  MOV     #177777,R3              ;FILL RECIEVE BUFFER WITH 1'S
001072' 012700  100000                  MOV     #LINADR,R0              ;RECIEVE BUFFER STARTS HERE
001076' 010067  177556                  MOV     R0,RBUFP
001102' 010320                  10$:    MOV     R3,(R0)+
001104' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ;FILL ENTIRE BUFFER
001110' 103774                          BLO     10$
001112' 005003                          CLR     R3                      ;FILL XMIT BUFFER WITH 0'S
001114' 010067  177542                  MOV     R0,TBUFP                ;XMIT BUFFER STARTS 1K AWAY FROM RECIEVE
001120' 010320                  20$:    MOV     R3,(R0)+
001122' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001126' 103774                          BLO     20$

001130' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
001136' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ;SET PROMISCUIOUS MODE AND ENABLE LOOPBACK

001144' 016701  177512                  MOV     TBUFP,R1                ;POINT TO XMIT BUFFER
001150' 005021                          CLR     (R1)+                   ;CLEAR OUT STATUS WORD
001152' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ;SET BYTE COUNT TO MAXIMUM ALLOWED
001156' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001162' 005067  177452                  CLR     FLG4                    ;CLEAR THE INTERRUPT FLAG
001166' 016737  177466  021032          MOV     RBUFP,@#LFRBUF          ;TELL LINK WHERE RECIEVE BUFF IS
001174' 016737  177462  021030          MOV     TBUFP,@#LTAC            ;TELL LINK WHERE TRANSMIT BUFF IS...
                                                                        ;WHICH WILL START A XMIT OPERATION
001202' 106427  000200                  MTPS    #PRI04                  ;ALLOW RECEIVER TO INTERRUPT
001206' 032767  000100  177424  30$:    BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVE INTERRUPT
001214' 001774                          BEQ     30$

001216' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMTR TO INTERRUPT
001222' 032767  000040  177410  40$:    BIT     #TRNFLG,FLG4            ; WAIT FOR XMIT INTERRUPT
001230' 001774                          BEQ     40$

001232' 106427  000340                  MTPS    #PRI07                  ;DISABLE INTERRUPTS
001236' 112767  000002  177401          MOVB    #2,PCSR1+1              ;TELL HOST THIS IS TEST #2
001244' 000241                          CLC
001246' 000207                          RTS     PC
```

```

                                          ; DATA BYTE FRAMING TEST
                                          ;
                                          ; THIS IS "MICROCODE" FOR DATA BYTE FRAMING TEST. FILLS XMIT BUFFER
                                          ; WITH PATTERN 0000000011111111 (BINARY) AND TRANSMITS OVER 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 #3
                                          ;
 001250' 112767  000002  177366  MICD3:   MOVB    #INTST,PCSR1             ; TELL HOST WE ARE TESTING
 001256' 016737  177362  021020           MOV     PCSR1,@#IPCSR1

                                          ; ***********************************************
                                          ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                          ; ***********************************************

 001264' 012700  100000                   MOV     #LINADR,R0               ; RECEIVE BUFFER STARTS HERE
 001270' 010067  177364                   MOV     R0,RBUFP
 001274' 005020                  10$:     CLR     (R0)+                    ; FILL RECEIVE BUFFER WITH ZEROS
 001276' 020027  104000                   CMP     R0,#LINADR+SIZ1K         ; FILL ENTIRE BUFFER
 001302' 103774                           BLO     10$


                                          ; **********************************************
                                          ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                          ; **********************************************

 001304' 012703  000377                   MOV     #0377,R3                 ; WORST CASE FOR CLOCKING
 001310' 010067  177346                   MOV     R0,TBUFP
 001314' 010320                  20$:     MOV     R3,(R0)+                 ; FILL XMIT BUFFER WITH PATTERN
 001316' 020027  110000                   CMP     R0,#LINADR+SIZ2K                 ; STOP AT TOP
 001322' 103774                           BLO     20$


                                          ; ***********************  ***************************
                                          ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                          ; ****************************************************

 001324' 012737  100200  177776           MOV     #MODE!ENABLE,@#CMDREG    ; ENABLE LINK, SELECT MODE REG
 001332' 012737  100004  177774           MOV     #PROM!LOOP,@#MODREG      ; PROM MODE AND LOOPBACK

 001340' 016701  177316                   MOV     TBUFP,R1                 ; POINT AT XMIT BUFFER
 001344' 005021                           CLR     (R1)+                    ; CLEAR OUT STATUS WORD
 001346' 012721  002752                   MOV     #MAXBC-CRCSIZ,(R1)+      ; SET BYTE COUNT TO MAX ALLOWED
 001352' 005037  021034                   CLR     @#CLRFIF                 ; CLEAR THE FIFO
 001356' 005067  177256                   CLR     FLG4                     ; CLEAR INTERRUPT FLAG
 001362' 016737  177272  021032           MOV     RBUFP,@#LFRBUF           ; TELL UNA WHERE RECEIVE BUFF IS
 001370' 016737  177266  021030           MOV     TBUFP,@#LTAC             ; TELL UNA WHERE XMIT BUFF IS

 001376' 106427  000140                   MTPS    #PRI03                   ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
 001402' 032767  000100  177230  30$:     BIT     #RCVFLG,FLG4             ; WAIT FOR RECEIVER INTERRUPT
 001410' 001774                           BEQ     30$

 001412' 032767  000040  177220  35$:     BIT     #TRNFLG,FLG4             ;WAIT FOR TRANSMIT INTERRUPT TOO
```

```
001420' 001774                          BEQ     35$
 
001422' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
 
 
                                ; ********************************************
                                ; ***** VERIFY PATTERN IN RECEIVE BUFFER *****
                                ; ********************************************
 
001426' 012700  100000                  MOV     #LINADR,R0              ; VERIFY RECEIVE BUFFER CONTENT
001432' 012005                          MOV     (R0)+,R5                ; SAVE STATUS IN CASE ERROR
001434' 005004                          CLR     R4                      ; TRACK OFFSET IN CASE ERROR
001436' 062700  000002                  ADD     #2,R0                   ; DON'T NEED "LENGTH" IN BUFFER
 
001442' 011001                  40$:    MOV     (R0),R1                 ; READ THE BUFFER
001444' 020103                          CMP     R1,R3                   ; R3 CONTAINS ORIGINAL PATTERN
001446' 001012                          BNE     70$                     ; GO TO ERROR EXIT
001450' 005200                          INC     R0                      ; COULDN'T BUMP TILL AFTER TEST
001452' 005204                          INC     R4                      ; TRACK OFFSET FROM BUFFER START
001454' 022704  002752                  CMP     #MAXBC-CRCSIZ,R4        ; COMPARE ALL BUFFER ENTRIES
001460' 001370                          BNE     40$
001462' 000241                          CLC                             ;TELL MICROMONITOR SUCCESS
 
                                ; *********************************************
                                ; ***** EXIT POINT FOR NO ERROR CONDITION *****
                                ; *********************************************
 
001464' 112767  000003  177153  50$:    MOVB    #3,PCSR1+1              ; TELL HOST TEST 3 DONE
001472' 000207                          RTS     PC                      ; RETURN TO SENDER
 
 
                                ; ******************************************
                                ; ***** EXIT POINT FOR ERROR CONDITION *****
                                ; ******************************************
 
001474' 016737  177146  021010  70$:    MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
001502' 016737  177142  021012          MOV     IPCSR2+2,@#MDMA1
001510' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
001514' 013702  021014                  MOV     @#MDMAR0,R2             ; R2=CONTENTS OF PCSR3
001520' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO PCBB+0
001524' 010237  021012                  MOV     R2,@#MDMA1
 
001530' 010537  021026                  MOV     R5,@#MDMAW0             ; WRITE STATUS WORD TO HOST
001534' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE ORIGNAL PATTERN BACK
001540' 010137  021026                  MOV     R1,@#MDMAW0             ; WRITE ERRENT PATTERN TO HOST
001544' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE ERROR OFFSET TO HOST
 
001550' 000261                          SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001552' 000744                          BR      50$                     ; GO EXIT THROUGH NORMAL
 
```

```

                                      ; DATA WORD FRAMING TEST
                                      ;
                                      ; THIS IS "MICROCODE" FOR DATA WORD FRAMING TEST. FILLS XMIT BUFFER
                                      ; WITH PATTERN 00000000000000001111111111111111 (BINARY) AND TRANSMITS
                                      ; 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 #4

                                      ; ********************************
                                      ; ***** TELL HOST WE ARE BUSY *****
                                      ; ********************************

  001554' 112767  000002  177062  MICD4:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
  0C1562' 016737  177056  021020          MOV     PCSR1,@#IPCSR1


                                      ; ******************************************************
                                      ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                      ; ******************************************************

  001570' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS H:RE
  001574' 010067  177060                  MOV     R0,RBUFP
  001600' 005020                  10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
  001602' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL ENTIRE BUFFER
  001606' 103774                          BLO     10$


                                      ; *********************************************
                                      ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                      ; *********************************************

  001610' 012703  177777                  MOV     #177777,R3              ; WORST CASE FOR CLOCKING
  001614' 010067  177042                  MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
  001620' 010320                  20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
  001622' 005103                          CGM     R3                      ; FLIP IT OVER
  001624' 020027  110000                  CMP     R0,#LINADR+SIZ2K                ; STOP AT TOP
  001^30' 103773                          BLO     20$


                                      ; ********************************************
                                      ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                      ; ********************************************

  001632' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK, SELECT MODE REG
  001640' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ; PROM MODE AND LOOPBACK

  001646' 016701  177010                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
  001652' 005021                          CLR     (R1)+                   ; 'LEAR OUT STATUS WORD
  001654' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
  001660' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
  001664' 005067  176750                  CLR     FLG4                    ; CLEAR INTERRUPT FLAG
```

```
001670' 016737  176764  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUFF IS
001676' 016737  176760  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUFF IS

001704' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
001710' 032767  000100  176722  30$:    BIT     #RCVFLG,FLG4            ; WAIT FOR RECEIVER INTERRUPT
001716' 001774                          BEQ     30$

001720' 032767  000040  176712  35$:    BIT     #TRNFLG,FLG4            ;WAIT FOR XMIT INTERRUPT TOO
001726' 001774                          BEQ     35$

001730' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS



                                ; ***********************************************
                                ; ***** VERIFY THE CONTENTS OF RECEIVE BUFFER *****
                                ; ***********************************************

001734' 012700  100000                  MOV     #LINADR,R0              ; VERIFY RECEIVE BUFFER CONTENT
001740' 012005                          MOV     (R0)+,R5                ; SAVE STATUS IN CASE ERROR
001742' 005004                          CLR     R4                      ; TRACK OFFSET IN CASE ERROR
001744' 062700  000002                  ADD     #2,R0                   ; DON'T NEED ''LENGTH'' IN BUFFER

001750' 005003                          CLR     R3                      ; NEED A ZERO
001752' 005103                  40$:    COM     R3                      ; FLIP IT OVER
001754' 011001                          MOV     (R0),R1                 ; READ DATA BACK
001756' 020103                          CMP     R1,R3
001760' 001012                          BNE     70$                     ; GO TO ERROR EXIT
001762' 005200                          INC     R0                      ; NEW ADDRESS AFTER TESTING
001764' 005204                          INC     R4                      ; NEXT OFFSET
001766' 022704  002752                  CMP     #MAXBC-CRCSIZ,R4
001772' 103767                          BLO     40$                     ; END OF LOOP
001774' 000241                          CLC                             ;TELL MICROMONITOR SUCCESS

                                ; *****************************************
                                ; ***** FALLTHROUGH EXIT IF NO ERROR *****
                                ; *****************************************

001776' 112767  000004  176641  50$:    MOVB    #4,PCSR1+1              ; TELL HIM TEST FINISHED
002004' 000207                          RTS     PC                      ; RETURN TO SENDER

                                ; *********************
                                ; ***** ERROR EXIT *****
                                ; *********************

002006' 016737  176634  021010  70$:    MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
002014' 016737  176630  021012          MOV     IPCSR2+2,@#MDMA1
002022' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
002026' 013702  021014                  MOV     @#MDMAR0,R2             ; R2=CONTENTS OF PCSR3
002032' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO PCBB+0
002036' 010237  021012                  MOV     R2,@#MDMA1

002042' 010537  021026                  MOV     R5,@#MDMAW0             ; WRITE STATUS WORD TO HOST
002046' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE ORIGNAL TO HOST
002052' 010137  021026                  MOV     R1,@#MDMAW0             ; WRITE ERRENT PATTERN TO HOST
002056' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE ERROR OFFSET TO HOST
```

```
  577  002062' 000261                       SEC                             ;TELL MICROMONITOR ERROR OCCURRED
  578  002064' 000744                       BR      50$                     ; GO EXIT THROUGH NORMAL
  579
```

```

                                          ; 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 *****
                                          ; **********************************
                                          ;
 002066' 112767  000002  176550   MICD5:  MOVB    #INTST,PCSR1              ; TELL HOST WE ARE TESTING
 002074' 016737  176544  021020           MOV     PCSR1,@#IPCSR1

                                          ; ********************************************
                                          ; ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                          ; ********************************************
                                          ;
 002102' 016737  176540  021010           MOV     IPCSR2,@#MDMA0            ; SET TO GET HOST PCBB ADDRESS
 002110' 016737  176534  021012           MOV     IPCSR2+2,@#MDMA1
 002116' 013700  021014                   MOV     @#MDMAR0,R0               ; R0 NOW CONTAINS PCBB LOW
 002122' 013701  021014                   MOV     @#MDMAR0,R1               ; R1 NOW CONTAINS PCBB HIGH
 002126' 010037  021010                   MOV     R0,@#MDMA0                ; POINT AT PCBB
 002132' 010137  021012                   MOV     R1,@#MDMA1
 002136' 013703  021014                   MOV     @#MDMAR0,R3               ; R3 NOW HOLDS DATA PATTERN


                                          ; ******************************************************
                                          ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                          ; ******************************************************
                                          ;
 002142' 012700  100000                   MOV     #LINADR,R0                ; RECEIVE BUFFER STARTS HERE
 002146' 010067  176506                   MOV     R0,RBUFP
 002152' 005020                   10$:    CLR     (R0)+                     ; FILL RECEIVE BUFFER WITH ZEROS
 002154' 020027  104000                   CMP     R0,#LINADR+SIZ1K          ; FILL ENTIRE BUFFER
 002160' 103774                           BLO     10$


                                          ; *********************************************
                                          ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                          ; *********************************************
                                          ;
 002162' 010067  176474                   MOV     R0,TBUFP                  ; SAVE COPY OF ADDRESS
 002166' 010320                   20$:    MOV     R3,(R0)+                  ; FILL XMIT BUFFER WITH PATTERN
 002170' 020027  110000                   CMP     R0,#LINADR+SIZ2K                  ; STOP AT TOP
 002174' 103774                           BLO     20$


                                          ; ********************************************
                                          ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                          ; *********************************************

002176' 012737  100200  177776            MOV     #MODE!ENABLE,@#CMDREG    ; ENABLE LINK, SELECT MODE REG
002204' 012737  100004  177774            MOV     #PROM!LOOP,@#MODREG      ; PROM MODE AND LOOPBACK

002212' 016701  176444                    MOV     TBUFP,R1                 ; POINT AT XMIT BUFFER
002216' 005021                            CLR     (R1)+                    ; CLEAR OUT STATUS WORD
002220' 012721  002752                    MOV     #MAXBC-CRCSIZ,(R1)+      ; SET BYTE COUNT TO MAX ALLOWED
002224' 005037  021034                    CLR     @#CLRFIF                 ; CLEAR THE FIFO
002230' 005067  176404                    CLR     FLG4                     ; CLEAR INTERRUPT FLAG
002234' 016737  176420  021032            MOV     RBUFP,@#LFRBUF           ; TELL UNA WHERE RECEIVE BUFF IS
002242' 016737  176414  021030            MOV     TBUFP,@#LTAC             ; TELL UNA WHERE XMIT BUFF IS

002250' 106427  000140                    MTPS    #PRI03                   ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
002254' 032767  000100  176356   30$:     BIT     #RCVFLG,FLG4             ; WAIT FOR RECEIVER INTERRUPT
002262' 001774                            BEQ     30$

002264' 032767  000040  176346   35$:     BIT     #TRNFLG,FLG4             ;WAIT FOR XMIT INTERRUPT TOO
002272' 001774                            BEQ     35$

002274' 106427  000340                    MTPS    #PRI07                   ; DISABLE INTERRUPTS


                                          ; ***************************************************
                                          ; ***** VERIFY THE CONTENTS OF RECEIVE BUFFER *****
                                          ; ***************************************************

002300' 012700  100000                    MOV     #LINADR,R0               ; VERIFY RECEIVE BUFFER CONTENT
002304' 012002                            MOV     (R0)+,R2                 ; SAVE STATUS IN CASE ERROR
002306' 005004                            CLR     R4                       ; TRACK OFFSET IN CASE ERROR
002310' 062700  000002                    ADD     #2,R0                    ; DON'T NEED "LENGTH" IN BUFFER

002314' 011005                   40$:     MOV     (R0),R5                  ; READ DATA BACK
002316' 020503                            CMP     R5,R3                    ; R3 HOLDS ORIGINAL PATTERN
002320' 001012                            BNE     70$                      ; GO TO ERROR EXIT
002322' 005200                            INC     R0                       ; NEW ADDRESS AFTER TESTING
002324' 005204                            INC     R4                       ; NEXT OFFSET
002326' 022704  002752                    CMP     #MAXBC-CRCSIZ,R4
002332' 103770                            BLO     40$                      ; END OF LOOP
002334' 000241                            CLC                              ;TELL MICROMONITOR SUCCESS

                                          ; *****************************************
                                          ; ***** FALLTHROUGH EXIT IF NO ERROR *****
                                          ; *****************************************

002336' 112767  000005  176301   50$:     MOVB    #5,PCSR1+1               ; EXIT HERE IF NO ERROR
002344' 000207                            RTS     PC                       ; RETURN TO SENDER


                                          ; *********************
                                          ; ***** ERROR EXIT *****
                                          ; *********************

002346' 016737  176274  021010   70$:     MOV     IPCSR2,@#MDMA0           ; SET TO GET HOST PCBB ADDRESS
002354' 016737  176270  021012            MOV     IPCSR2+2,@#MDMA1
002362' 013700  021014                    MOV     @#MDMAR0,R0              ; R0 NOW CONTAINS PCBB LOW
```

```
002366' 013701  021014                  MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HIGH

002372' 062700  000002                  ADD     #2,R0                   ; INDEX DOWN TO PCBB+2
002376' 005501                          ADC     R1

002400' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO PCBB+0
002404' 010137  021012                  MOV     R1,@#MDMA1

002410' 010237  021026                  MOV     R2,@#MDMAW0             ; WRITE STATUS WORD TO HOST
002414' 010537  021026                  MOV     R5,@#MDMAW0             ; WRITE ERRENT PATTERN TO HOST
002420' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE ERROR OFFSET TO HOST

002424' 000261                          SEC                             ;TELL MICROMONITOR ERROR OCCURRED
002426' 000743                          BR      50$                     ; GO EXIT THROUGH NORMAL

```

```

                                        ; STATUS MUX VERIFICATION TEST
                                        ;
                                        ; THIS TEST INSURES THAT THE STATUS MULTIPLEXER WRITES INTO THE FIRST
                                        ; TWO LOCATIONS TO THE LINK TRANSMIT BUFFER. THE DEUNA PROCESSOR WILL
                                        ; FILL A TRANSMIT BUFFER AND A RECEIVE BUFFER AND WILL TRANSMIT A
                                        ; DATAGRAM OVER THE LOOPBACK. THE FIRST TWO WORDS OF THE RECEIVE BUFFER
                                        ; WILL BE COPIED TO HOST MEMORY FOR VERIFICATION.
                                        ;
                                        ; HOST WILL VERIFY THAT PREDETERMINED BITS OF THE STATUS WORD ARE ZERO.


                                        ; ********************************
                                        ; ***** TELL HOST WE ARE BUSY *****
                                        ; ********************************
 002430' 112767  000002  176206  MICD6:  MOVB    #INTST,PCSR1       ; TELL HOST WE ARE TESTING
 002436' 016737  176202  021020          MOV     PCSR1,@#IPCSR1


                                        ; ******************************
                                        ; ***** FILL RECEIVE BUFFER *****
                                        ; ******************************

 002444' 012703  177777                  MOV     #177777,R3         ; FILL RECEIVE BUFFER WITH ONES
 002450' 012700  100000                  MOV     #LINADR,R0         ; RECEIVE BUFFER STARTS HERE
 002454' 010067  176200                  MOV     R0,RBUFP           ; SAVE A COPY

 002460' 010320                  10$:    MOV     R3,(R0)+           ; FILL THE BUFFER
 002462' 020027  104000                  CMP     R0,#LINADR+SIZ1K
 002466' 103774                          BLO     10$

                                        ; *****************************************
                                        ; ***** FILL TRANSMIT BUFFER WITH ZERO *****
                                        ; *****************************************

 002470' 010067  176166                  MOV     R0,TBUFP           ; SAVE A COPY OF TRANSMIT ADDRESS
 002474' 005003                          CLR     R3                 ; NEED A ZERO
 002476' 010320                  20$:    MOV     R3,(R0)+           ; XMIT BUFFER STARTS 1K FROM RECEIVE
 002500' 020027  110000                  CMP     R0,#LINADR+SIZ2K
 002504' 103774                          BLO     20$


                                        ; **********************************************
                                        ; ***** FILL DATAGRAM FOR DATAGRAM LOOPBACK *****
                                        ; **********************************************

 002506' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK MODE, SEL MODE REG
 002514' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ; PROMIS, ENABLE LOOPBACK

 002522' 016701  176134                  MOV     TBUFP,R1                ; POINT TO XMIT BUFFER
 002526' 005103                          COM     R3                      ; NEED SOME 1'S
 002530' 010321                          MOV     R3,(R1)+                ; BACKGROUND- SHOW STATUS OVRLAY
 002532' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
 002536' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE INTERRUPT FLAG
 002542' 005067  176072                  CLR     FLG4                    ; CLEAR THE INTERRUPT FLAG
```

```
002546' 016737  176106  021032          MOV     RBUFP,@#LFRBUF          ; TELL LINK WHERE RECEIVE IS
002554' 016737  176102  021030          MOV     TBUFP,@#LTAC            ; TELL LINK WHERE XMIT IS
002562' 106427  000140                  MTPS    #PRI03                  ; WAIT FOR AN INTERRUPT
002566' 032767  000040  176044  30$:    BIT     #TRNFLG,FLG4            ; WAIT FOR XMIT INTERRUPT
002574' 001774                          BEQ     30$
002576' 032767  000100  176034  35$:    BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVER BEFORE GOING ANY FURTHER
002604' 001774                          BEQ     35$



                                ; **************************************************
                                ; ***** GET STATUS WORDS FROM TRANSMIT BUFFER *****
                                ; **************************************************

002606' 016700  176050                  MOV     TBUFP,R0                ; POINT AT XMIT BUFFER
002612' 012003                          MOV     (R0)+,R3                ; R5 NOW HOLDS TX0 STATUS WORD
002614' 011004                          MOV     (R0),R4                 ; R6 NOW HOLDS TX1 STATUS WORD


                                ; ************************************************************
                                ; ***** GET HOST MEMORY ADDRESS AND WRITE STATUS WORDS *****
                                ; ************************************************************

002616' 016737  176024  021010          MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
002624' 016737  176020  021012          MOV     IPCSR2+2,@#MDMA1
002632' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
002636' 013702  021014                  MOV     @#MDMAR0,R2             ; R2=CONTENTS OF PCSR3
002642' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB+0
002646' 010237  021012                  MOV     R2,@#MDMA1

002652' 010337  021026                  MOV     R3,@#MDM'W0             ; WRITE TX0 TO HOST MEMORY
002656' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE TX1 TO HOST MEMORY

002662' 112767  000006  175755          MOVB    #6,PCSR1+1
002670' 000241                          CLC
002672' 000207                          RTS     PC

```

```

                                  ; LINK BYTE COUNTER TEST
                                  ;
                                  ; THIS IS "MICROCODE" FOR THE LINK BYTE COUNTER TEST.
                                  ; GETS A BYTE COUNT FROM THE HOST MEMORY. LOOPS BACK A DATAGRAM
                                  ; WITH THE BYTE COUNT TO THE RECEIVE BUFFER. VERIFIES THE RECEIVE
                                  ; BUFFER TO:
                                  ;
                                  ;       1. MAKE SURE RECEIVE BYTE COUNT IS THE SAME AS THAT DESIGNATED
                                  ;               BY THE TRANSMIT BUFFER BYTE COUNT ENTRY
                                  ;
                                  ;       2. MAKE SURE RECEIVE BUFFER WAS ACTUALLY OVERWRITTEN WITH THE
                                  ;               BYTE COUNT WRITTEN IN THE TRANSMIT BYTE COUNT BUFFER
                                  ;               ENTRY.
                                  ;
                                  ; WRITES THE VALUE OF THE RECEIVE BUFFER BYTE COUNT TO HOST MEMORY.

                                  .SBTTL  LINK BYTE COUNTER TEST

                                  ; *********************************
                                  ; ***** TELL HOST WE ARE BUSY *****
                                  ; *********************************

 002674' 112767  000002  175742   MICD7:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
 002702' 016737  175736  021020           MOV     PCSR1,@#IPCSR1


                                  ; *****************************************************
                                  ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                  ; *****************************************************

 002710' 012700  100000                   MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
 002714' 012703  177777                   MOV     #177777,R3              ; GET ALL ONES
 002720' 010067  175734                   MOV     R0,RBUFP                ; SAVE ADDRESS OF RECV BUFFER

 002724' 010320                   10$:    MOV     R3,(R0)+                ; FILL UP THE BUFFER
 002726' 020027  104000                   CMP     R0,#LINADR+SIZ1K        ; OVERFILL- SHOW RECV OVERLAY
 002732' 103774                           BLO     10$


                                  ; ************************************************
                                  ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                  ; ************************************************

 002734' 010067  175722                   MOV     R0,TBUFP                ; SAVE ADDRESS XMIT BUFFER
 002740' 005020                   20$:    CLR     (R0)+                   ; ZEROS FOR OVERLAY
 002742' 020027  110000                   CMP     R0,#LINADR+SIZ2K        ; FILL XMIT BUFFER WITH PATTERN
 002746' 103774                           BLO     20$                     ; STOP AT THE TOP


                                  ; *********************************************
                                  ; ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
                                  ; *********************************************

 002750' 016737  175672  021010           MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
 002756' 016737  175666  021012           MOV     IPCSR2+2,@#MDMA1
 002764' 013700  021014                      MOV     @#MDMARO,RO             ; RO NOW CONTAINS PCBB LO
 002770' 013701  021014                      MOV     @#MDMARO,R1             ; R1 NOW CONTAINS PCBB HI
 002774' 010037  021010                      MOV     RO,@#MDMAO              ; POINT AT PCBB
 003000' 010137  021012                      MOV     R1,@#MDMA1
 003004' 013703  021014                      MOV     @#MDMARO,R3             ; R3 NOW CONTAINS BYTE COUNT



                                  ; ******************************************
                                  ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                  ; ******************************************

 003010' 012737  100200  177776              MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK, SELECT MODE REG
 003016' 012737  100014  177774              MOV     #PROM!LOOP!DTCR,@#MODREG        ; PROM MODE AND LOOPBACK

 003024' 016704  175632                      MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
 003030' 005024                              CLR     (R4)+                   ; CLEAR OUT STATUS WORD
 003032' 010324                              MOV     R3,(R4)+                ; WRITE PASSED BYTE COUNT
 003034' 005037  021034                      CLR     @#CLRFIF                ; CLEAR THE FIFO
 003040' 005067  175574                      CLR     FLG4                    ; CLEAR INTERRUPT FLAG
 003044' 016737  175610  021032              MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECIEVE BUFF IS
 003052' 016737  175604  021030              MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUFF IS

 003060' 106427  000140                      MTPS    #PRIO3                  ; ALLOW XMITTER TO INTERRUPT
 003064' 032767  000040  175546  30$:        BIT     #TRNFLG,FLG4            ; WAIT FOR INTERRUPT
 003072' 001774                              BEQ     30$
 003074' 032767  000100  175536  35$:        BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVER INTERRUPT TOO
 003102' 001774                              BEQ     35$


 003104' 106427  000340                      MTPS    #PRIO7                  ; DISABLE INTERRUPTS



                                  ; ******************************************
                                  ; ***** COUNT ZEPOS IN RECEIVE BUFFER *****
                                  ; ******************************************

 003110' 016704  175544                      MOV     RBUFP,R4                ; VERIFY RECEIVE BUFFER CONTENTS
 003114' 062704  000002                      ADD     #2,R4                   ; && COULD REPORT STATUS &&
 003120' 012402                              MOV     (R4)+,R2                ; SAVE "MLEN" BUFFER LENGTH
 003122' 005005                              CLR     R5                      ; COUNT DATA WORDS TRANSFERRED

 003124' 112403                  40$:        MOVB    (R4)+,R3                ; READ DATA BACK
 003126' 122703  000000                      CMPB    #0,R3                   ; IS IT ZERO?
 003132' 001004                              BNE     45$                     ; EXIT IF AT THE EDGE
 003134' 005205                              INC     R5                      ; BUMP THE TALLY
 003136' 020527  002756                      CMP     R5,#MAXBC               ; DONE YET?
 003142' 103770                              BLO     40$                     ; IF NOT, KEEP GOING

 003144' 062700  000002          45$:        ADD     #2,RO                   ; INDEX DOWN TO PCBB+2
 003150' 005501                              ADC     R1

 003152' 010037  021010                      MOV     RO,@#MDMAO              ; POINT TO PCBB+2
 003156' 010137  021012                      MOV     R1,@#MDMA1

 003162' 010237  021026                      MOV     R2,@#MDMAWO             ; WRITE RECEIVE BYTF COUNT
 003166' 010537  021026                      MOV     R5,@#MDMAWO             ; WRITE BUFFER BYTE COUNT
```

```

003172' 112767  000007  175445          MOVB    #7,PCSR1+1              ; TELL HIM WHAT TEST IT IS
003200' 000241                          CLC
003202' 000207                          RTS     PC
```

```




                                          ; LINK MEMORY ARBITRATION TEST

                                          .SBTTL  LINK MEMORY ARBITRATION TEST

                                          ; THIS IS MICROCODE FOR THE LINK ARBITRATION TEST.
                                          ; THIS MICROCODE WILL HAVE THE DMA ENGINE, THE T-11, AND BOTH STATE
                                          ; MACHINES ATTEMPTING TO ACCESS LINK MEMORY AT THE SAME TIME.
                                          ;
                                          ; EACH PROCESS OPERATES ON DATA THAT IS 'IN!QUE FOR IDENTIFICATION.
                                          ;
                                          ; ***********************************
                                          ; ***** TELL HOST WE ARE TESTING ****
                                          ; ***********************************

003204' 112737  000002  021020  MICD8:  MOVB    #INTST,@#IPCSR1         ; TELL HOST WE ARE TESTING
003212' 016737  175426  021020          MOV     PCSR1,@#IPCSR1


                                          ; ******************************************
                                          ; ***** RETRIEVE HOST MEMORY FRC'' PCBB *****
                                          ; ******************************   *********

003220' 016737  175422  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
003226' 016737  175416  021012          MOV     IPCSR2+2,@#MDMA1
003234' 013704  021014                  MOV     @#MDMAR0,R4             ; R4 NOW HOLDS PCBB LOW
003240' 013705  021014                  MOV     @#MDMAR0,R5             ; R5 NOW HOLDS PCBB HIGH


                                          ; ******************************************
                                          ; ***** FILL RECEIVE BUFFER WITH ZEROS *****
                                          ; ******************************************

003244' 012700  100000                  MOV     #LINADR,R0              ; FILL RECEIVE BUFFER
003250' 010067  175404                  MOV     R0,RBUFP                ; SAVE A COPY OF POINTER
003254' 005020                  10$:    CLR     (R0)+                   ; CLEAR IT OUT
003256' 020027  104000                  CMP     R0,#LINADR+SIZ1K
003262' 103774                          BLO     10$


                                          ; *****************************************
                                          ; ***** FILL TRANSMIT BUFFER WITH 33 HEX **
                                          ; ***** THATS 31463 OCTAL *****************
                                          ; *****************************************

003264' 010067  175372                  MOV     R0,TBUFP                ; FILL TRANSMIT BUFFER
003270' 012720  031463          20$:    MOV     #31463,(R0)+            ; 31463 = 3333 HEX (THE REAL THING)
003274' 020027  110000                  CMP     R0,#LINADR+SIZ2K
003300' 103773                          BLO     20$


                                          ; ****************************************
                                          ; ***** FILL DMA BUFFER WITH 0F HEX ***
```

```
                                        ; ***** THATS 7417 OCTAL **************
                                        ; ************************************

003302' 010067  175356                         MOV     R0,DBUFP                ; SAVE COPY OF POINTER
003306' 012720  007417                 30$:    MOV     #7417,(R0)+
003312' 020027  114000                         CMP     R0,#LINADR+SIZ3K
003316' 103773                                 BLO     30$


                                        ; ********************************
                                        ; ***** FILL BUFFER WITH FF HEX ***
                                        ; ***** THATS 177777 OCTAL ********
                                        ; ********************************

003320' 010067  175342                         MOV     R0,MBUFP                ; T-11 BUFFER POINTER
003324' 012720  177777                 40$:    MOV     #177777,(R0)+
003330' 020027  120000                         CMP     R0,#LINADR+SIZ4K
003334' 103773                                 BLO     40$


                                        ; **************************************
                                        ; ***** GET DMA ENGINE READY TO GO *****
                                        ; **************************************

003336' 010402                                 MOV     R4,R2                   ; GET HOST ADDRESS OF PCBB
003340' 010503                                 MOV     R5,R3

003342' 062702  000006                         ADD     #6,R2                   ; DUMP BUFFER AFTER ERROR CODES
003346' 005503                                 ADC     R3

003350' 010237  021004                         MOV     R2,@#DMATO              ; "TO" REGISTERS INCREMENTS
003354' 010337  021006                         MOV     R3,@#DMAT1
003360' 016737  175300  021022                 MOV     DBUFP,@#DMAF            ; WHERE "FROM" REG IS LOCATED
003366' 012737  003774  021024                 MOV     #3774,@#DMAWC           ; WILL XFER 1K-2 WORDS

003374' 010700                                 MOV     PC,R0                   ; CALCULATE THE INTERRUPT VECTOR
003376' 062700  175074                         ADD     #DMAINT-.,R0            ; THROW IN THE OFFSET
003402' 010037  000114                         MOV     R0,@#DMAVEC
003406' 012737  000300  000116                 MOV     #PRI06,@#DMAVEC+2       ;PRIORITY OF INTERRUPT SERVICE ROUTINE
003414' 005067  175236                         CLR     DMDONE                  ; FLAG

                                        ; *************************************
                                        ; ***** SET UP THE STATE MACHINES *****
                                        ; *************************************

003420' 012737  100200  177776                 MOV     #MODE!ENABLE,@#CMDREG
003426' 012737  100004  177774                 MOV     #PROM!LOOP,@#MODREG

003434' 016701  175222                         MOV     TBUFP,R1                ; GET ADDRESS OF XMIT BUFFER
003440' 062701  000002                         ADD     #2,R1                   ; SKIP TO BYTE COUNT
003444' 012721  002752                         MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
003450' 005037  021034                         CLR     @#CLRFIF                ; CLEAR INTERRUPT FLAG
003454' 005067  175160                         CLR     FLG4                    ; CLEAR INTERRUPT FLAG
003460' 016737  175174  021032                 MOV     RBUFP,@#LFRBUF          ; TELL LINK WHERE RECEIVE BUF IS
003466' 016700  175174                         MOV     MBUFP,R0                ;POINT TO MICROCPU BUFFER
003472' 012702  002000                         MOV     #SIZ1K/2,R2             ;WORD COUNT FOR MICROCPU LOOP
```

```
003476' 012701  125252                      MOV     #125252,R1                  ;DATA FOR MICROCPU LOOP

                                    ; ******************************
                                    ; ***** STATE THE MACHINES *****
                                    ; ******************************
                                                                                ; IS EVERYBODY READY?

003502' 016737  175154  021030              MCV     TBUFP,@#LTAC                ; THIS STARTS THE STATE MACHINES

003510' 005237  021002                      INC     @#DMACSR                    ; THIS STARTS DMA ENGINE

003514' 106427  000140                      MTPS    #PRI03                      ; ALLOW INTERRUPTS

003520' 010120                      50$:    MOV     R1,(R0)+                    ;START FILLING LINK MEMORY...
003522' 077202                              SOB     R2,50$                      ;WITH MICROCPU

                                                                                ;THE MICROCPU IS DONE

003524' 032767  000100  175106      60$:    BIT     #RCVFLG,FLG4                ;IS THE RECEIVE DONE?
003532' 001774                              BEQ     60$                         ;NOT YET

003534' 032767  000040  175076      70$:    BIT     #TRNFLG,FLG4                ;IS THE TRANSMIT DONE?
003542' 001774                              BEQ     70$                         ;NOT YET

003544' 005767  175106              80$:    TST     DMDONE                      ;IS THE DMA ENGINE DONE?
003550' 001775                              BEQ     80$                         ;NOT YET

                                                                                ;EVERYBODY IS DONE, SO NOW CHECK THE...
                                                                                ;DATA

003552' 016700  175102                      MOV     RBUFP,R0                    ;POINT TO RECEIVE BUFFER
003556' 062700  000004                      ADD     #4,R0                       ;INDEX DOWN TO DATA PART
003562' 012702  001365                      MOV     #<MAXBC-CRCSIZ>/2,R2        ;AMOUNT OF DATA TO CHECK
003566' 022710  031463              90$:    CMP     #31463,(R0)                 ;IS THE DATA CORRECT?
003572' 001005                              BNE     100$                        ;NO
003574' 062700  000002                      ADD     #2,R0                       ;YES, POINT TO NEXT WORD OF DATA
003600' 077206                              SOB     R2,90$                      ;CONTINUE CHECKING DATA UNTIL DONE
003602' 000241                              CLC                                 ;TELL HOST THIS TEST SUCCESS
003604' 000414                              BR      110$

003606' 010437  021010              100$:   MOV     R4,@#MDMA0                  ;POINT TO HOST PCBB
003612' 010537  021012                      MOV     R5,@#MDMA1
003616' 012737  031463  021026              MOV     #31463,@#MDMAW0             ;GIVE HOST EXPECTED PATTERN IN PCBB
003624' 011037  021026                      MOV     (R0),@#MDMAW0               ;GIVE HOST ACTUAL PATTERN IN PCBB+2
003630' 010037  021026                      MOV     R0,@#MDMAW0                 ;GIVE HOST LINK MEMORY ADDRESS
003634' 000261                              SEC                                 ;TELL MICROMONITOR ERROR OCCURRED

003636' 112767  000010  175001      110$:   MOVB    #8.,PCSR1+1                 ;TELL HOST WHAT TEST THIS IS
003644' 000207                              RTS     PC                          ;RETURN TO MICROMONITOR

```

```
                                     .SBTTL  LINK BYTE COUNTER MAXIMUM TEST
                                     ;
                                     ; ****************************************
                                     ; ***** TELL THE HOST WE ARE TESTING *****
                                     ; ****************************************
                                     ;
003646' 012767  000002  174770  MICD9:  MOV     #INTST,PCSR1           ; TELL HOST WE ARE TESTING
003654' 016737  174764  021020          MOV     PCSR1,@#IPCSR1
                                     ;
                                     ;
                                     ; **********************************************
                                     ; ***** CALCULATE BUFFER ADDRESS POINTERS *****
                                     ; **********************************************
                                     ;
003662' 012700  100000                  MOV     #LINADR,R0             ; RELATIVE TO LINK MEMORY
003666' 010067  174766                  MOV     R0,RBUFP
003672' 062700  004000                  ADD     #SIZ1K,R0
003676' 010067  174760                  MOV     R0,TBUFP
                                     ;
                                     ;
                                     ; ***************************************
                                     ; ***** SET UP LINK FOR A LOOPBACK *****
                                     ; ***************************************
                                     ;
003702' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG
003710' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG
                                     ;
003716' 016704  174740                  MOV     TBUFP,R4               ; SET UP XMIT BUFFER
003722' 005024                          CLR     (R4)+
003724' 012724  007777                  MOV     #MXMTBC,(R4)+
003730' 005037  021034                  CLR     @#CLRFIF
003734' 005067  174700                  CLR     FLG4
003740' 016737  174714  021032          MOV     RBUFP,@#LFRBUF
003746' 016737  174710  021030          MOV     TBUFP,@#LTAC
                                     ;
                                     ;
                                     ; ********************************
                                     ; ***** WAIT FOR INTERRUPTS *****
                                     ; ********************************
                                     ;
003754' 106427  000100                  MTPS    #PRI02
003760' 032767  000040  174652  10$:    BIT     #TRNFLG,FLG4
003766' 001774                          BEQ     10$
                                     ;
003770' 032767  000100  174642  20$:    BIT     #RCVFLG,FLG4
003776' 001774                          BEQ     20$
                                     ;
004000' 106427  000340                  MTPS    #PRI07
                                     ;
                                     ;
                                     ; ******************************************
                                     ; ***** CHECK RECEIVE BUFFER BYTE COUNT *****
                                     ; ******************************************
                                     ;
004004' 016704  174650                  MOV     RBUFP,R4
```

```
004010' 062704  000002                  ADD     #2,R4           ; POINT AT MLEN
004014' 012402                          MOV     (R4)+,R2
004016' 022702  007777                  CMP     #MRECBC,R2
004022' 001005                          BNE     40$
004024' 000241                          CLC
004026' 112767  000011  174611  30$:    MOVB    #9.,PCSR1+1
004034' 000207                          RTS     PC
                                ;
004036' 000261                  40$:    SEC                     ; ERROR EXIT
004040' 000772                          BR      30$
```

```
                                    .SBTTL  MODULE D, MICROTEST #10 - FIFO TEST

                                    ;**************************************************************************
                                    ;
                                    ;THIS IS THE MICROCODE FOR THE FIFO TEST.
                                    ;IT TRANSMITS A PACKET FROM BUFFER 0, SETS UP THE RECEIVER TO RECEIVE TO
                                    ;THE BUFFER SPECIFIED BY THE HOST IN PCBB+0. AFTER THE INTERRUPT THE BUFFER
                                    ;DONE FIFO CONTENTS ARE READ AND PASSED TO THE HOST IN PCBB+2.
                                    ;
                                    ;**************************************************************************

004042' 112767  000002  174574      MICD10: MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
004050' 016737  174570  021020              MOV     PCSR1,@#IPCSR1
                                    ;
                                    ;GET RECEIVE BUFFER ADDRESS FROM HOST MEMORY
                                    ;
004056' 016737  174564  021010              MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2+3
004064' 016737  174560  021012              MOV     IPCSR2+2,@#MDMA1
004072' 013700  021014                      MOV     @#MDMAR0,R0             ;R0 = CONTENTS OF HOST'S PCSR2
004076' 013701  021014                      MOV     @#MDMAR0,R1             ;R1 = CONTENTS OF HOST'S PCSR3
004102' 010037  021010                      MOV     R0,@#MDMA0              ;POINT TO PCBB+0
004106' 010137  021012                      MOV     R1,@#MDMA1
004112' 013767  021014  174540              MOV     @#MDMAR0,RBUFP          ;GET RECEIVE BUFFER ADDRESS
                                    ;
                                    ;CLEAR ALL OF LINK MEMORY
                                    ;
004120' 012702  100000                      MOV     #LINADR,R2
004124' 005022                      10$:    CLR     (R2)+
004126' 020227  177774                      CMP     R2,#LINADR+LINSIZ
004132' 103774                              BLO     10$
                                    ;
                                    ;FILL THE TRANSMIT BUFFER WITH 1'S
                                    ;
004134' 012702  100000                      MOV     #LINADR,R2              ;USE BUFFER 0 FOR TRANSMIT
004140' 010267  174516                      MOV     R2,TBUFP                ;SAVE IT
004144' 012722  177777              20$:    MOV     #177777,(R2)+
004150' 020227  104000                      CMP     R2,#LINADR+SIZ1K
004154' 103773                              BLO     20$
                                    ;
                                    ;SET UP LINK FOR PROMISCUOUS MODE AND INTERNAL LOOPBACK. TRANSMIT THE MAX
                                    ;SIZE PACKET. CLEAR THE FIFO AND GIVE THE LINK A RECEIVER BUFFER AND A TRANSMIT
                                    ;BUFFER TO START THE OPERATION.
                                    ;
004156' 012737  100200  177776              MOV     #MODE!ENABLE,@#CMDREG   ;TURN ON THE LINK AND SELECT MODE REG
004164' 012737  100004  177774              MOV     #PROM!LOOP,@#MODREG     ;SET PROMISCUOUS MODE AND LOOPBACK
004172' 016702  174464                      MOV     TBUFP,R2                ;GET TRANSMIT BUFFER
004176' 005022                              CLR     (R2)+                   ;CLEAR THE STATUS WORD
004200' 012712  002752                      MOV     #MAXBC-CRCSIZ,(R2)      ;SET TO MAX BYTE COUNT
004204' 005037  021034                      CLR     @#CLRFIF                ;CLEAR THE FIFO
004210' 005067  174424                      CLR     FLG4                    ;CLEAR THE INTERRUPT FLAG
004214' 016737  174440  021032              MOV     RBUFP,@#LFRBUF          ;LOAD THE FIFO WITH A RECEIVE BUFFER
004222' 016737  174434  021030              MOV     TBUFP,@#LTAC            ;GIVE TRANSMIT BUFFER TO START TRANSMIT
                                    ;
                                    ;LOWER THE PROCESSOR PRIORITY AND WAIT FOR THE INTERRUPTS
                                    ;
004230' 106427  000140                      MTPS    #PRI03                  ;ALLOW BOTH TRANSMIT AND RECEIVER INTERRUPT
```

```
004234' 032767  000100  174376  30$:    BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVER FIRST
004242' 001774                          BEQ     30$

004244' 032767  000040  174366  40$:    BIT     #TRNFLG,FLG4            ;THEN THE TRANSMITTER
004252' 001774                          BEQ     40$

004254' 106427  000340                  MTPS    #PRI07                  ;DISABLE FURTHER INTERRUPTS
                                ;
                                ;READ RECEEVER BUFFER DONE FIFO AND PASS BACK TO HOST IN PCBB+2
                                ;
004260' 062700  000002                  ADD     #2,R0                   ;POINT TO PCBB+2
004264' 005501                          ADC     R1
004266' 010037  021010                  MOV     R0,@#MDMA0
004272' 010137  021012                  MOV     R1,@#MDMA1
004276' 013737  021044  021026          MOV     @#LRBUF,@#MDMAW0        ;PASS BUFFER DONE DATA TO HOST

004304' 112767  000012  174333          MOVB    #10.,PCSR1+1            ;INDICATE WHAT TEST WE JUST FINISHED
004312' 000241                          CLC                             ;INDICATE SUCCESS
004314' 000207                          RTS     PC
```

```
                                        .SBTTL  MODULE D, MICROTEST #11 - LINK ADDRESS TEST

                                        ;*****************************************************************    ****
                                        ;
                                        ;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.
                                        ;
                                        ;*****************************************************************************

004316' 112767  000002  174320  MICD11: MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
004324' 016737  174314  021020          MOV     PCSR1,@#IPCSR1

                                        ;
                                        ;GET THE RECEIVER BUFFER FROM THE HOST'S PCBB+0 AND THE TRANSMIT BUFFER FROM
                                        ;THE HOST'S PCBB+2
                                        ;
004332' 016737  174310  021010          MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2+3
004340' 016737  174304  021012          MOV     IPCSR2+2,@#MDMA1
004346' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 = CONTENTS OF HOST'S PCSR2
004352' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 = CONTENTS OF HOST'S PCSR3
004356' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
004362' 010137  021012                  MOV     R1,@#MDMA1
004366' 013767  021014  174264          MOV     @#MDMAR0,RBUFP          ;GET RECEIVER BUFFER FROM PCBB+0
004374' 013767  021014  174260          MOV     @#MDMAR0,TBUFP          ;GET TRANSMIT BUFFER FROM PCBB+2
                                        ;
                                        ;FILL ALL OF LINK MEMORY WITH 0'S
                                        ;
004402' 012702  100000                  MOV     #LINADR,R2
004406' 005022                  10$:    CLR     (R2)+
004410' 020227  177774                  CMP     R2,#LINADR+LINSIZ
004414' 103774                          BLO     10$

004416' 016702  174240                  MOV     TBUFP,R2                ;POINT TO BASE OF TRANSMIT BUFFER
004422' 005022                          CLR     (R2)+                   ;CLEAR STATUS WORD
004424' 012722  002756                  MOV     #MAXBC,(R2)+            ;PUT IN BYTE COUNT
                                        ;
                                        ;FILL TRANSMIT BUFFER WITH 1'S
                                        ;
004430' 012701  002756                  MOV     #MAXBC,R1
004434' 006201                          ASR     R1
004436' 012722  177777          20$:    MOV     #177777,(R2)+
004442' 005301                          DEC     R1
004444' 001374                          BNE     20$


                                        ;
                                        ;SET UP LINK FOR PROMISCIOUS MODE AND INTERNAL LOOPBACK. TRANSMIT THE PACKET
                                        ;
004446' 012737  000200  177776          MOV     #MODE,@#CMDREG          ;SELECT THE MODE REGISTER
004454' 012737  100014  177774          MOV     #PROM!LOOP!DTCR,@#MODREG ;ENABLE PROMISCUIOUS MODE AND LOOPBACK
```

```
                                                                          ;AND DISABLE TRANSMIT CRC
004462' 012737  100000  177776             MOV     #ENABLE,@#CMDREG       ;TURN ON THE LINK
004470' 005037  021034                     CLR     @#CLRFIF               ;CLEAR THE FIFO
004474' 005067  174140                     CLR     FLG4                   ;CLEAR THE INTERRUPT WORD
004500' 016737  174154  021032             MOV     RBUFP,@#LFRBUF         ;GIVE THE LINK A RECEIVE BUFFER
004506' 016737  174150  021030             MOV     TBUFP,@#LTAC           ;GIVE THE LINK A TRANSMIT BUFFER
                                                                          ;TO START THE OPERATION
                                   ;
                                   ;LOWER THE PROCESSOR PRIORITY AND WAIT FOR THE INTERRUPT
                                   ;
004514' 106427  000140                     MTPS    #PRI03                 ;ALLOW BOTH TRANSMIT AND RECEIVE INTERRUPT
004520' 032767  000100  174112     30$:    BIT     #RCVFLG,FLG4           ;WAIT FOR THE RECEIVE INTERRUPT
004526' 001774                             BEQ     30$
004530' 032767  000040  174102     40$:    BIT     #TRNFLG,FLG4
004536' 001774                             BEQ     40$

004540' 106427  000340                     MTPS    #PRI07                 ;DISABLE FURTHER INTERRUPTS
                                   ;
                                   ;CHECK RECEIVE BUFFER TO MAKE SURE IT RECEIVED NON-ZERO DATA
                                   ;
004544' 012703  177777                     MOV     #177777,R3             ;PASS THIS NON-ZERO DATA IF FAILURE
004550' 016702  174104                     MOV     RBUFP,R2               ;GET POINTER TO RECEIVER BUFFER
004554' 062702  000004                     ADD     #4,R2                  ;SKIP STATUS AND BYTE COUNT
004560' 012701  002756                     MOV     #MAXBC,R1              ;NUMBER OF BYTES WE SENT
004564' 006201                             ASR     R1                     ;MAKE IT WORDS
004566' 005712                     45$:    TST     (R2)                   ;IS THERE NON-ZERO DATA HERE?
004570' 001432                             BEQ     90$                    ;NO, ERROR!
004572' 062702  000002                     ADD     #2,R2                  ;POINT TO NEXT WORD IN RECEIVER BUFFER
004576' 005301                             DEC     R1
004600' 001372                             BNE     45$
                                   ;
                                   ;CHECK ALL LINK MEMORY EXCEPT THE TRANSMIT AND RECEIVE BUFFERS FOR ANY NON-ZERO
                                   ;DATA
                                   ;
004602' 005003                             CLR     R3                     ;PASS THIS ZERO DATA IF FAILURE
004604' 012702  100000                     MOV     #LINADR,R2             ;START AT BASE OF LINK MEMORY
004610' 020267  174046             50$:    CMP     R2,TBUFP               ;ARE WE AT THE TRANSMIT BUFFER?
004614' 001002                             BNE     60$                    ;NO
004616' 062702  002762                     ADD     #MAXBC+4,R2            ;YES, SKIP OVER THE HEADER AND THE DATA
004622' 020267  174032             60$:    CMP     R2,RBUFP               ;ARE WE AT THE RECEIVE BUFFER?
004626' 001002                             BNE     70$                    ;NO
004630' 062702  002762                     ADD     #MAXBC+4,R2            ;YES, SKIP OVER THE HEADER AND THE DATA
004634' 005712                     70$:    TST     (R2)                   ;IS ANY NON-ZERO DATA IN HERE?
004636' 001007                             BNE     90$                    ;YES, ERROR!
004640' 062702  000002                     ADD     #2,R2                  ;POINT TO NEXT WORD IN LINK MEMORY
004644' 020227  177774                     CMP     R2,#LINADR+LINSIZ      ;HAVE WE CHECKED ALL OF LINK MEMORY?
004650' 103757                             BLO     50$                    ;NOT YET
                                   ;
                                   ;TEST WAS SUCESSFULL
                                   ;
004652' 000241                             CLC                            ;INDICATE SUCCESS
004654' 000407                             BR      100$                   ;LEAVE
                                   ;
                                   ;PASS THE ADDRESS BACK TO THE HOST IN PCBB+4 AND THE DATA IN PCBB+6
                                   ;
004656' 010237  021026             90$:    MOV     R2,@#MDMALO            ;PASS ADDRESS TO PCBB+4
```

```
  1333  004662' 010337  021026                   MOV     R3,@#MDMAWO              ;PASS GOOD DATA TO PCBB+6
  1334  004666' 011237  021026                   MOV     (R2),@#MDMAWO            ;PASS DATA TO PCBB+10
  1335  004672' 000261                           SEC                              ;INDICATE FAILURE
  1336
  1337  004674' 112767  000013  173743  100$:    MOVB    #11.,PCSR1+1             ;TELL HOST WHAT TEST JUST FINISHED
  1338  004702' 000207                           RTS     PC
  1339
  1340
  1341  004704' 004706                   MICDSZ::MICDSZ-MICROD+2
  1342          000001                   .END
```

```
ACLO  = 002000              87#
ADRERR= 000001             132#
ADRREG= 177774              34#
ARAM  = 000100              80#
BIT0  = 000001              54#    103
BIT1  = 000002              53#    104
BIT10 = 002000              44#     56      73      87
BIT11 = 004000              43#     72      86
BIT12 = 010000              42#     56      71     113
BIT13 = 020000              41#     70     112
BIT14 = 040000              40#     69      85     111     240
BIT15 = 100000              39#     68      78      84     110     245     248
BIT2  = 000004              52#     91     105
BIT3  = 000010              51#     90     106
BIT4  = 000020              50#     89     107
BIT5  = 000040              49#     88     108
BIT6  = 000100              48#     80     109
BIT7  = 000200              47#     79
BIT8  = 000400              46#     56      74
BIT9  = 001000              45#
CLRFIF= 021034              22#    309*    342*    405*    519*    644*    761*    874*   1024*   1110*   1193*   1279*
CMDREG= 177776              35#    138*    303*    336*    399*    513*    638*    754*    868*   1018*   1104*   1188*   1275*
                          1278*
COLL  = 000020              89#
CRCSIZ= 000004             130#    303     341     404     434     518     550     643     673     760    1023    1061    1192
CSRFLG= 000001             103#    196     227
CSRVEC= 000064              96#    163*    164*
CSRWRT  000436R    002     162     227#
DATERR= 000000             131#
DBUFP   000664R    002     286#    976*   1005
DMACSR= 021002               9#    239     248*   1038*
DMAF  = 021022              17#   1005*
DMAINT  000472R    002     157     239#   1009
DMAT0 = 021004              10#   1003*
DMAT1 = 021006              11#   1004*
DMAVEC= 000114              97#    158*    159*   1010*   1011*
DMAWC = 021024              18#   1006*
DMDONE  000656R    002     239*    240     245     283#   1012*   1053
DNI   = 004000              72#    142     220
DRTY  = 000040              88#
DTCR  = 000010              90#    869    1276
ENABLE= 100000              78#    303     336     399     513     638     754     868    1018    1104    1188    1278
ENAL  = 040000              85#
ENCR  = 004000              86#
ERRFLG= 000002             104#    230
ERRINT  000446R    002     144     230#
FATI  = 000400              74#
FLG4    000640R    002     189*    192     196     221*    227*    230*    253*    260*    263*    278#    310*    315     343*
                           348     352     406*    411     414     520*    525     528     645*    650     653     762*    766
                           768     875*    880     882    1025*   1047    1050    1111*   1121    1124    1194*   1201    1204
                          1280*   1288    1290
INERR = 000003             102#    218
INMON = 000001             100#    140     204     216
INTST = 000002             101#    289     322     370     482     597     723     824     935    1086    1156    1237
IOADR = 020000             124#    125
IOSIZ = 020000             120#    125
```

```
MICROD - MICROCODE MODULE D     MACY11 30(1046)  11-JAN-83  09:52  PAGE 34
MICROD.MAC     10-JAN-83 17:53            CROSS REFERENCE TABLE -- USER SYMBOLS

IPCSR0= 021000                  8#     142*     201      220*     231*     242*     247*     254*
IPCSR1= 021020                 16#     141*     204*     219*     290*     323*     371*     483*     598*     724*     825*     935*     936*
                             1087*    1157*    1238*
IPCSR2  000646R     002       187*     188*     281#     450      451      565      566      604      605      689      690      786      787
                              855      856      943      944     1161     1162     1244     1245
LASFTP= 012400                 56#
LFRBUF= 021032                 21#     311*     344*     407*     521*     646*     763*     876*    1026*    1112*    1195*    1281*
LINADR= 100000                126#     292      295      300      325      328      333      377      380      391      424      490      493
                              505      538      617      620      630      663      732      736      746      832      837      847      953
                              956      967      978      989     1094     1171     1173     1178     1181     1255     1257     1312     1322
LINSIZ= 077774                122#    1173     1257     1322
LOOP  = 000004                 91#     304      337      400      514      639      755      869     1019     1105     1189     1276
LRBUF = 021044                 26#     258     1215
LTAC  = 021030                 20#     312*     345*     408*     522*     647*     764*     877*    1036*    1113*    1196*    1282*
MAXBC = 002756                127#     308      341      404      434      518      550      643      673      760      902     1023     1061
                             1192     1262     1266     1300     1315     1318
MBUFP   000666R     002       287#     987*    1027
MDMAR0= 021014                 14#     452      453      567      568      606      607      610      691      692      788      739      857
                              858      861      945      946     1163     1164     1167     1246     1247     1250     1251
MDMAR1= 021016                 15#
MDMAW0= 021026                 19#     457*     458*     459*     460*     572*     573*     574*     575*     700*     701*     702*     793*
                              794*     911*     912*    1071*    1072*    1073*    1215*    1332*    1333*    1334*
MDMAW1= 021036                 23#
MDMA0 = 021010                 12#     450*     454*     565*     569*     604*     608*     689*     697*     786*     790*     855*     859*
                              908*     943*    1069*    1161*    1165*    1213*    1244*    1248*
MDMA1 = 021012                 13#     451*     455*     566*     570*     605*     609*     690*     698*     787*     791*     856*     860*
                              909*     944*    1070*    1162*    1166*    1214*    1245*    1249*
MDMSR = 021042                 25#
MICDSZ  004704RG    002      1341#
MICD1   000670R     002       266      289#
MICD10  004042R     002       275     1156#
MICD11  004316R     002       276     1237#
MICD2   001052R     002       267      322#
MICD3   001250R     002       268      370#
MICD4   001554R     002       269      482#
MICD5   002066R     002       270      597#
MICD6   002430R     002       271      723#
MICD7   002674R     002       272      824#
MICD8   003204R     002       273      935#
MICD9   003646R     002       274     1086#
MICROD  000000RG    002       137#    1341
MODE  = 000200                 79#     303      336      399      513      638      754      868     1018     1104     1188     1275
MODREG= 177774                 33#     304*     337*     400*     514*     639*     755*     869*    1019*    1105*    1189*    1276*
MRECBC= 007777                129#    1137
MXMTBC= 007777                128#    1109
NPRERR= 100000                110#     247
NPRFLG= 000020                107#
NXMERR= 040000                111#     242
NXMFLG= 000010                106#
PARERR= 010000                113#     254
PARFLG= 000004                105#     253
PARINT  000550R     002       152      253#
PARVEC= 000140                 98#     153*     154*
PCBADR  000652R     002       282#
PCEI  = 040000                 69#
PCSRSW= 021040                 24#     181
```

```
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
RBUFP    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#
TBUFP    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

. ABS.   000000     000
```

```
        000000      001
MICRD   004706      002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROD.OBJ,MICROD.LST/CRF/DOC/NL:TOC/SOL=MICROD.MAC
 RUN-TIME: 1 2 .4 SECONDS
 RUN-TIME RATIO: 35/4=7.7
 CORE USED:  9K  (18 PAGE.)

 DOCUMENT PAGES:  35
```

```
                                .TITLE  MICROE - MICROCODE MODULE E
                                ; && DEDICATED THE CRC CIRCUITRY TO THE RECEIVE SIDE OF THE LINK

        000000'                 .CSECT MICRE

                                .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000                  IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002                  DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004                  DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006                  DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010                  MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012                  MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014                  MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016                  MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020                  IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022                  DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024                  DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026                  MDMAW0          21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030                  LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032                  LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034                  CLRFIF  =       21034   ;CLEAR FIFO
        021036                  MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040                  PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042                  MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044                  LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060                  PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062                  PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064                  PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066                  PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070                  PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072                  PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774                  MODREG  =       177774  ;LINK MODE REGISTER
        177774                  ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776                  CMDREG  =       177776  ;LINK COMMAND REGISTER

                                .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000                  BIT15   =       100000
        040000                  BIT14   =       40000
        020000                  BIT13   =       20000
        010000                  BIT12   =       10000
        004000                  BIT11   =       4000
        002000                  BIT10   =       2000
        001000                  BIT9    =       1000
        000400                  BIT8    =       400
        000200                  BIT7    =       200
        000100                  BIT6    =       100
        000040                  BIT5    =       40
        000020                  BIT4    =       20
        000010                  BIT3    =       10
        000004                  BIT2    =       4
        000002                  BIT1    =       2
        000001                  BIT0    =       1
                                ;
        012400                  LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340          PRIO7   =       340
        000300          PRIO6   =       300
        000240          PRIO5   =       240
        000200          PRIO4   =       200
        000140          PRIO3   =       140
        000100          PRIO2   =       100
        000040          PRIO1   =       40
        000000          PRIO0   =       0
                        ;
                        ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                        ;
        100000          SERI    =       BIT15
        040000          PCEI    =       BIT14
        020000          RXI     =       BIT13
        010000          TXI     =       BIT12
        004000          DNI     =       BIT11
        002000          RCEI    =       BIT10
        000400          FATI    =       BIT8
                        ;
                        ;LINK COMMAND REGISTER
                        ;
        100000          ENABLE  =       BIT15   ;ENABLE LINK MODULE
        000200          MODE    =       BIT7    ;ENABLE MODE REGISTER
        000100          ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                        ;
                        ;LINK MODE REGISTER
                        ;
        100000          PROM    =       BIT15   ;PROMISCUIOUS MODE
        040000          ENAL    =       BIT14   ;ENABLE MULTICAST
        004000          ENCR    =       BIT11   ;ENABLE COLLISION TEST
        002000          ACLO    =       BIT10   ;ENABLE ACLO
        000040          DRTY    =       BIT5    ;DISABLE RETRY LOGIC
        000020          COLL    =       BIT4    ;SIMULATE A COLLISION
        000010          DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
        000004          LOOP    =       BIT2    ;ENABLE LOOPBACK

        000070          TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
        000120          RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
        000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000          STACK=  1000            ;STACK LOCATION
        000001          INMON=  1               ;IN MICROMONITOR STATE
        000002          INTST=  2               ;IN A TEST STATE
        000003          INERR=  3               ;IN ERROR STATE
        000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
        000040          TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
        000100          RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
        100000          NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
        040000          NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000          UNIERR= BIT13           ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
```

```
010000                 PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
020000                 MTCH=   BIT13           ;MATCH BIT
004000                 SIZ1K=  4000            ;1K WORDS
010000                 SIZ2K=  SIZ1K*2         ;2K WORDS
020000                 SIZ4K=  SIZ2K*2         ;4K WORDS
040000                 SIZ8K=  SIZ4K*2         ;8K WORDS
020000                 WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
020000                 IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
040000                 ROMSIZ= SIZ8K           ;SIZE OF ROM
077774                 LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
000000                 WCSADR= 0               ;BASE ADDRESS OF WCS
020000                 IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
040000                 ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
100000                 LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
000100                 MINBC=  64.             ; 64 BYTES
002752                 MAXBC=  1518.-4.        ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
000000                 DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
000001                 ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
```

```

                                                 .SBTTL  MICROCODE MODULE E

000000' 106427  000340                  MICROE::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776                  MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                          MOV     #STACK,SP               ;SETUP STACK
000016' 112767  000001  000602                  MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000576  021020                  MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000                  MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                                  MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000404                          ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                                  CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                          10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                          MOV     #PRI07,(R1)+
000056' 020127  001000                          CMP     R1,#1000
000062' 002772                                  BLT     10$

000064' 010700                                  MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000462                          ADD     #PARINT-.,R0
000072' 010037  000140                          MOV     R0,@#PARVEC
000076' 012737  000340  000142                  MOV     #PRI07,@#PARVEC+2

000104' 010700                                  MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000364                          ADD     #DMAINT-.,R0
000112' 010037  000114                          MOV     R0,@#DMAVEC
000116' 012737  000340  000116                  MOV     #PRI07,@#DMAVEC+2

000124' 010700                                  MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000310                          ADD     #CSRWRT-.,R0
000132' 010037  000064                          MOV     R0,@#CSRVEC
000136' 012737  000200  000066                  MOV     #PRI04,@#CSRVEC+2

000144' 010700                                  MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000316                          ADD     #TIMINT-.,R0
000152' 010037  000134                          MOV     R0,@#SANVEC
000156' 012737  000240  000136                  MOV     #PRI05,@#SANVEC+2

000164' 010700                                  MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
000166' 062700  000414                          ADD     #TRNINT-.,R0
000172' 010037  000070                          MOV     R0,@#TRNVEC
000176' 012737  000200  000072                  MOV     #PRI04,@#TRNVEC+2

000204' 010700                                  MOV     PC,R0                   ;SETUP RECEIVER VECTOR
000206' 062700  000360                          ADD     #RCVINT-.,R0
000212' 010037  000120                          MOV     R0,@#RCVVEC
000216' 012737  000240  000122                  MOV     #PRI05,@#RCVVEC+2

000224' 013700  021040                          MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000230' 052700  176000                          BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000234' 006300                                  ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000236' 006300                                  ASL     R0
000240' 006300                                  ASL     R0
000242' 062700  000004                          ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000246' 010067  000356                          MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000252' 012767  000003  000352                  MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000260' 005067  000336                          CLR     FLGS                    ;INITIALIZE FLAG WORD
```

```
000264' 106427  000000          15$:    MTPS    #PR100                  ;ALLOW INTERRUPTS

000270' 005767  000326          20$:    TST     FLG5                    ;WAIT FOR A COMMAND FROM HOST
000274' 001775                          BEQ     20$

000276' 106427  000340                  MTPS    #PR107                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000302' 032767  000001  000312          BIT     #CSRFLG,FLG5            ;DID HOST GIVE US A COMMAND?
000310' 001001                          BNE     30$                     ;YES
000312' 000777                          BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                        ;FOR LACK OF ANYTHING BETTER TO DO

000314' 113700  021000          30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000320' 042700  177760                  BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000324' 001004                          BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000326' 012737  000001  021020          MOV     #INMON,@#IPCSR1         ;YES, CLEAR OUT THE TEST # BITS
000334' 000432                          BR      50$
000336' 022700  000017          35$:    CMP     #17,R0                  ;RESTART OPERATIONAL MICROCODE?
000342' 001432                          BEQ     60$                     ;YES
000344' 162700  000001                  SUB     #1,R0
000350' 010701                          MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000352' 062701  000240                  ADD     #TBLD-.,R1
000356' 006300                          ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000360' 060001                          ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000362' 061101                          ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000364' 004711                          JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000366' 103404                          BCS     40$                     ;ERROR OCCURRED
000370' 112767  000001  000230          MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000376' 000403                          BR      45$                     ;MICROMONITR
000400' 112767  000003  000220  40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000406' 016737  000214  021020  45$:    MOV     PCSR1,@#IPCSR1
000414' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000422' 005067  000174          50$:    CLR     FLG5                    ;RESET FLAG WORD
000426' 000716                          BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000430' 005000                  60$:    CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000432' 000137  040006                  JMP     @#40006                 ;START OPERATIONAL MICROCODE

000436' 052767  000001  000156  CSRWRT: BIS     #CSRFLG,FLG5            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000444' 000002                          RTI

000446' 052767  000002  000146  ERRINT: BIS     #ERRFLG,FLG5            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000454' 012737  020000  021000          MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                        ;HAPPENED
000462' 000777                          BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                        ;COUNT ON HOST TO TIMEOUT

000464' 005267  000134          TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000470' 000002                          RTI

000472' 013767  021002  000140  DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000500' 032767  040000  000132          BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000506' 001404                          BEQ     10$                     ;NO
000510' 012737  040000  021000          MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                        ;LOCATION WAS ADDRESSED
000516' 000407                          BR      20$
000520' 032767  100000  000112  10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000526' 001407                          BEQ     30$                             ;NO
000530' 012737  100000  021000          MOV     #NPRERR,@#IPCSRO                ;TELL HOST NPR TIMEOUT HAPPENED
000536' 012737  100000  021002  20$:    MOV     #BIT15,@#DMACSR                 ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000544' 000777                          BR      .                               ;SIT HERE AND SPIN WHEELS
000546' 000002                  30$:    RTI

000550' 052767  000004  000044  PARINT: BIS     #PARFLG,FLG5                    ;SET PARITY ERROR OCCURRED
000556' 012737  010000  021000          MOV     #PARERR,@#IPCSRO                ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                                ;OCCURRED
000564' 000777                          BR      .                               ;SIT HERE AND SPIN WHEELS

000566' 005737  021044          RCVINT: TST     @#LRBUF                         ;READ BUFFER DONE REGISTER...
                                                                                ;WHICH CLEARS THE INTERRUPT
000572' 052767  000100  000022          BIS     #RCVFLG,FLG5                    ;SET RECEIVER INTERRUPT OCCURRED
000600' 000002                          RTI

000602' 052767  000040  000012  TRNINT: BIS     #TRNFLG,FLG5                    ;SET TRANSMITTER INTERRUPT OCCURRED
000610' 000002                          RTI

000612' 000034                  TBLD:   .WORD   MICE1-.                         ; STATION ADDRESS PATTERN
000614' 000614                          .WORD   MICE2-.                         ; STATION ADDRESS REJECTION
000616' 001064                          .WORD   MICE3-.                         ; STATION ADDRESS POSITION
000620' 001620                          .WORD   MICE4-.                         ; MULTICAST ADDRESS TEST

000622' 000000                  FLG5:   .WORD   0                               ;FLAG WORD
000624' 000000                  SANTIM: .WORD   0                               ;COUNT FOR SANITY TIMER
000626' 000000                  PCSR1:  .WORD   0                               ;COPY OF WHAT GOES TO PCSR1
000630' 000000  000000          IPCSR2: .WORD   0,0                             ;ADDRESS IN HOST MEMORY FOR PCSR2
000634' 000000  000000          PCBADR: .WORD   0,0                             ;ADDRESS IN HOST MEMORY FOR PCB
000640' 000000                  DMDONE: .WORD   0
000642' 000000                  RBUFP:  .WORD   0                               ;POINTER TO RECIEVE BUFFER
000644' 000000                  TBUFP:  .WORD   0                               ;POINTER TO XMIT BUFFER
```

```
                                  ;
                                  .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 *****
                                  ; **********************************
                                  ;
 [illegible]  000002  177752      MICE1:  MOVB    #INTST,PCSR1              ; TELL HOST WE ARE TESTING
 [illegible]  177746  021020              MOV     PCSR1,@#IPCSR1
                                  ;
                                  ;
                                  ; **************************************
                                  ; ***** PICK UP HOST ADDRESS OF PCBB *****
                                  ; **************************************
                                  ;
 000662' 016737  177742  021010           MOV     IPCSR2,@#MDMAR0           ; PICK UP ADDRESS OF PCBB
 000670' 016737  177736  021012           MOV     IPCSR2+2,@#MDMAR1
 000676' 013700  021014                   MOV     @#MDMAR0,R0               ; R0=CONTENTS OF PCSR2
 000702' 013701  021014                   MOV     @#MDMAR0,R1               ; R1=CONTENTS OF PCSR3
                                  ;
 000706' 010037  021010                   MOV     R0,@#MDMAR0               ; POINT TO HOST PCBB
 000712' 010137  021012                   MOV     R1,@#MDMAR1
                                  ;
 000716' 013702  021014                   MOV     @#MDMAR0,R2               ; R2 NOW HOLDS LS ADDRESS WORD
                                  ;
 000722' 062700  000002                   ADD     #2,R0                     ; INDEX TO NEXT HOST WORD
 000726' 005501                           ADC     R1
                                  ;
 000730' 013703  021014                   MOV     @#MDMAR0,R3               ; R3 NOW HOLDS MIDDLE PATTERN
                                  ;
 000734' 062700  000002                   ADD     #2,R0                     ; INDEX TO NEXT HOST WORD
 000740' 005501                           ADC     R1
                                  ;
 000742' 013704  021014                   MOV     @#MDMAR0,R4               ; R4 NOW HOLDS MS PATTERN WORD
                                  ;
 000746' 010446                           MOV     R4,-(SP)                  ; SAVE THEM FOR LATER
 000750' 010346                           MOV     R3,-(SP)
 000752' 010246                           MOV     R2,-(SP)
                                  ;
                                  ;
                                  ; ***********************************************************
                                  ; ***** MAKE A STATION ADDRESS FILE FROM THE PASSED PATTERN *****
                                  ; ***********************************************************
                                  ;
 000754' 010700                           MOV     PC,R0                     ; POINT AT SA FILE
 000756' 062700  000342                   ADD     #AFILE-.,R0               ;
                                  ;
 000762' 012705  000014                   MOV     #12.,R5                   ; NEED TOTAL OF 12 ENTRIES
 000766' 010220                   10$:    MOV     R2,(R0)+                  ; LS WORD OF PATTERN
```

```
000770' 010320                             MOV     R3,(R0)+                ; MIDDLE WORD OF PATTERN
000772' 010420                             MOV     R4,(R0)+                ; MS WORD OF PATTERN
000774' 005305                             DEC     R5                      ; END OF LOOP
000776' 001373                             BNE     10$
                                   ;
                                   ;
                                   ; ***************************************
                                   ; ***** LOAD THE STATION ADDRESS RAM *****
                                   ; ***************************************
                                   ;
001000' 012737  000200  177776             MOV     #MODE,@#CMDREG          ; SET LOOPBACK BEFORE LOADING
001006' 012737  000004  177774             MOV     #LOOP,@#CMDREG
                                   ;
001014' 010701                             MOV     PC,R1                   ; FORM POS/IND POINTER
001016' 062701  000302                     ADD     #AFILE-.,R1
                                   ;
001022' 005000                             CLR     R0                      ; THIS WILL CLEAR HIGH BITS
                                   ;
001024' 012746  000120                     MOV     #ARAM+20.,-(SP)         ; SA RAM STARTS AT +20 LOCATIONS
                                   ;
001030' 012702  000003                     MOV     #3,R2                   ; 3 WORDS PER ADDRESS/PATTERN
001034' 012704  000020             20$:    MOV     #16.,R4                 ; SIXTEEN BITS PER WORD
001040' 012705  000014             30$:    MOV     #12.,R5                 ; 12 POSITIONS IN SA RAM
                                   ;
001044' 010103                             MOV     R1,R3                   ; COPY THE POINTER
                                   ;
001046' 006013                     40$:    ROR     (R3)                    ; GET LSB OF ALL 12 ADDRESSES
001050' 006100                             ROL     R0                      ; R0 WILL HOLD ORTHOGONAL WORD
001052' 062703  000006                     ADD     #6,R3                   ; 6 BYTES PER ADDRES/PATTERN
001056' 077505                             SOB     R5,40$                  ; GO TILL DONE
                                   ;
001060' 011637  177776                     MOV     (SP),@#CMDREG           ; SET MODE TO WRITE SA RAM
001064' 010037  177774                     MOV     R0,@#ADRREG             ; ORTHOGONAL WORD TO SA RAM
001070' 005216                             INC     (SP)                    ; BUMP STATION ADDRESS
001072' 077416                             SOB     R4,30$                  ; DO ANOTHER ONE
                                   ;
001074' 062701  000002                     ADD     #2,R1                   ; ADVANCE TO NEXT WORD
001100' 077223                             SOB     R2,20$                  ; LOOP TILL DONE
                                   ;
001102' 012600                             MOV     (SP)+,R0                ; R0 IS A JUNK REGISTER
                                   ;
                                   ;
                                   ; *****************************************************
                                   ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                   ; **** *************************************************
                                   ;
001104' 012700  100000                     MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001110' 010067  177526                     MOV     R0,RBUFP
001114' 005020                     50$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001116' 020027  104000                     CMP     R0,#LINADR+SIZ1K        ; FILL THE BUFFER
001122' 103774                             BLO     50$                     ; FILL ENTIRE BUFFER
                                   ;
                                   ;
                                   ; ****************************************************
                                   ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                   ; ****************************************************
```

```
                                ;
001124' 012703  000377                  MOV     #0377,R3                ; WORST CASE FOR CLOCKING
001130' 010067  177510                  MOV     R0,TBUFP
001134' 010320                  60$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001136' 020027  110000                  CMP     R0,#LINADR+SIZ2K        ; STOP AT TOP
001142' 103774                          BLO     60$
                                ;
                                ;
                                ; *******************************************************
                                ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                ; *******************************************************
                                ;
001144' 012737  100020  177776          MOV     #ENABLE+20,@#CMDREG     ; LEAVE 20 IN COMMAND REGISTER
                                ;
001152' 016701  177466                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001156' 005021                          CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001160' 012721  000100                  MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                ;
001164' 012621                          MOV     (SP)+,(R1)+             ; GET BACK HIGH ADDRESS PATTERN
001166' 012621                          MOV     (SP)+,(R1)+
001170' 012621                          MOV     (SP)+,(R1)+             ; GET BACK LOW ADDRESS PATTERN
                                ;
001172' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
001176' 005067  177420                  CLR     FLG5                    ; CLEAR INTERRUPT FLAG
001202' 016737  177434  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
001210' 016737  177430  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                ;
001216' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
001222' 112767  000003  177376          MOVB    #INERR,PCSR1            ; TELL HOST IN CASE OF FAILURE
001230' 016737  177372  021020          MOV     PCSR1,@#IPCSR1
                                ;
001236' 016701  177402                  MOV     TBUFP,R1                ; WAIT FOR MATCH BIT FIRST
001242' 011102                  65$:    MOV     (R1),R2
001244' 032702  020000                  BIT     #MTCH,R2
001250' 001774                          BEQ     65$
                                ;
001252' 112767  000002  177346          MOVB    #INTST,PCSR1            ; TELL HOST WE GOT BY IT
001260' 016737  177342  021020          MOV     PCSR1,@#IPCSR1
                                ;
001266' 032767  000100  177326  70$:    BIT     #RCVFLG,FLG5            ; WAIT FOR RECEIVE INTER
001274' 001774                          BEQ     70$
                                ;
001276' 032767  000040  177316  80$:    BIT     #TRNFLG,FLG5            ; WAIT FOR XMIT INT TOO!
001304' 001774                          BEQ     80$
                                ;
001306' 112767  000001  177313          MOVB    #1,PCSR1+1              ; TELL HOST WE ARE DONE
001314' 000241                          CLC
001316' 000207                          RTS     PC
                                ;
                                ;
001320' 000000  000000  000000  AFILE:  .WORD   0,0,0           ; WORD #1
001326' 000000  000000  000000          .WORD   0,0,0
001334' 000000  000000  000000          .WORD   0,0,0
001342' 000000  000000  000000          .WORD   0,0,0
001350' 000000  000000  000000          .WORD   0,0,0
001356' 000000  000000  000000          .WORD   0,0,0
```

```
   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
```

```
                                ;
                                .SBTTL  STATION ADDRESS REJECTION TEST
                                ;
                                ;       THIS TEST WILL VERIFY THAT THE STATION ADDRESS LOGIC WILL NOT
                                ;       ACCEPT A STATION ADDRESS IF THAT STATION ADDRESS IS NOT PRESENT
                                ;       IN THE STATION ADDRESS RAM. STATION ADDRESS RAM WILL BE FILLED
                                ;       WITH AN ADDRESS. DIFFERENT ADDRESS WILL BE PUT IN DATAGRAM FIELD
                                ;       AND LOOPED BACK. LINK SHOULD NOT RECOGNIZE THE DATAGRAM.
                                ;
                                ; ***********************************
                                ; ***** TELL HIM WE ARE TESTING *****
                                ; ***********************************
                                ;
001430' 112767  000002  177170  MICE2:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001436' 016737  177164  021020          MOV     PCSR1,@#IPCSR1
                                ;
001444' 005067  177154                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                ;
                                ;
                                ; ************************************************************
                                ; ***** FILL STATION ADDRESS RAM WITH KNOWN (PHONEY) ADDRESS *****
                                ; ************************************************************
                                ;
001450' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; SET LOOP TO LOAD SA RAM
001456' 012737  000004  177774          MOV     #LOOP,@#MODREG
                                ;
001464' 012704  000060                  MOV     #48.,R4                 ; COUNTER
001470' 005000                          CLR     R0
                                ;
001472' 012701  000120                  MOV     #ARAM+20,R1             ; STATION ADDRESS STARTS AT +20
001476' 010137  177776          10$:    MOV     R1,@#CMDREG             ; SELECT STATION ADDRESS RAM
001502' 010037  177774                  MOV     R0,@#ADRREG             ; PARK IN RAM
001506' 005201                          INC     R1                      ; NEXT LOCATION
001510' 077406                          SOB     R4,10$                  ; DO THEM ALL
                                ;
                                ;
                                ; **************************************************
                                ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                ; **************************************************
                                ;
001512' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001516' 010067  177120                  MOV     R0,RBUFP
001522' 005020                  20$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001524' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL THE BUFFER
001530' 103774                          BLO     20$                     ; FILL ENTIRE BUFFER
                                ;
                                ;
                                ; ************************************************
                                ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                ; ************************************************
                                ;
001532' 012703  000377                  MOV     #0377,R3                ; WORST CASE FOR CLOCKING
001536' 010067  177102                  MOV     R0,TBUFP
001542' 010320                  30$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001544' 020027  110000                  CMP     R0,#LINADR+SIZ2K        ; STOP AT TOP
001550' 103774                          BLO     30$
```

```
                                        ;
                                        ;
                                        ;   *****************************************************
                                        ;   ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                        ;   *****************************************************
                                        ;
001552' 012737  100020  177776                  MOV     #ENABLE+20,@#CMDREG     ; POINTER TO FRONT OF RAM
                                        ;
001560' 016701  177060                          MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001564' 005021                                  CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001566' 012721  000100                          MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                        ;
                                        ;
                                        ;   ********************************************
                                        ;   ***** GET PHYSICAL ADDRESS INTO DATAGRAM *****
                                        ;   ********************************************
                                        ;
001572' 012700  177777                          MOV     #177777,R0              ; && STAND-IN FOR PHYSICAL ADDR
001576' 010021                                  MOV     R0,(R1)+
001600' 010021                                  MOV     R0,(R1)+
001602' 010021                                  MOV     R0,(R1)+
                                        ;
                                        ;
001604' 005037  021034                          CLR     @#CLRFIF                ; CLEAR THE FIFO
001610' 005067  177006                          CLR     FLG5                    ; CLEAR INTERRUPT FLAG
001614' 016737  177022  021032                  MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
001622' 016737  177016  021030                  MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                        ;
001630' 106427  000140                          MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
001634' 026727  176764  000002  40$:            CMP     SANTIM,#2               ; WAIT FOR 2 SECONDS
001642' 002012                                  BGE     50$                     ; EXIT NORMALLY IF TIMER DONE
001644' 032767  000100  176750                  BIT     #RCVFLG,FLG5            ; WAIT FOR RECEIVER INTERRUPT
001652' 001770                                  BEQ     40$                     ; IF NONE, WE'RE OK
                                        ;
                                        ;
                                        ;   ****************************
                                        ;   ***** ERROR FALLTHROUGH *****
                                        ;   ****************************
                                        ;
001654' 032767  000040  176740  45$:            BIT     #TRNFLG,FLG5            ; WAIT FOR XMIT INT TOO!
001662' 001774                                  BEQ     45$
                                        ;
001664' 000261                                  SEC                             ;INDICATE ERROR
001666' 000401                                  BR      60$
                                        ;
                                        ;
                                        ;   ****************************
                                        ;   ***** EXIT FOR NO ERROR *****
                                        ;   ****************************
                                        ;
001670' 000241                          50$:    CLC                             ;INDICATE SUCESS
001672' 112767  000002  176727  60$:            MOVB    #2,PCSR1+1              ; TELL HOST WE ARE DONE
001700' 000207                                  RTS     PC
                                        ;
                                        ;
                                        ;
                                        .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 PHSYICAL ADDRESS
                                        ;       WILL BE LOOPED AROUND. THE TEST WILL VERIFY THE DATAGRAM IS
                                        ;       RECEIVED.
                                        ;
                                        ; ************************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; ************************************
                                        ;
001702' 112767  000002  176716  MICE3:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001710' 016737  176712  021020          MOV     PCSR1,@#IPCSR1
                                        ;
001716' 005067  176702                  CLR     SANTIM                  ; CLEAR THE FLAG
                                        ;
                                        ;
                                        ; ****************************************
                                        ; ***** PICK UP HOST ADDRESS OF PCBB *****
                                        ; ****************************************
                                        ;
001722' 016737  176702  021010          MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
001730' 016737  176676  021012          MOV     IPCSR2+2,@#MDMA1
001736' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
001742' 013701  021014                  MOV     @#MDMAR0,R1             ; R1=CONTENTS OF PCSR3
                                        ;
001746' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO HOST PCBB
001752' 010137  021012                  MOV     R1,@#MDMA1
                                        ;
001756' 013702  021014                  MOV     @#MDMAR0,R2             ; R2 NOW HOLDS SA RAM POSITION
                                        ;
                                        ;
                                        ; **********************************************************
                                        ; ***** FILL THE BUFFER WITH PHONEY STATION ADDRESSES *****
                                        ; **********************************************************
                                        ;
001762' 010700                          MOV     PC,R0                   ; CALCULATE POS/IND ADDRESS
001764' 062700  000344                  ADD     #BFILE-.,R0
                                        ;
001770' 012705  000044                  MOV     #36.,R5                 ; THERE ARE 12 ADDRESSES
001774' 005020                  10$:    CLR     (R0)+                   ; FILL WITH ZERO (ADDRESS)
001776' 005305                          DEC     R5
002000' 001375                          BNE     10$
                                        ;
                                        ;
                                        ; ************************************************
                                        ; ***** PUT PHYSICAL ADDRESS IN RAM POSITION *****
                                        ; ************************************************
                                        ;
002002' 010700                          MOV     PC,R0                   ; CALCULATE POS/IND ADDRESS
002004' 062700  000324                  ADD     #BFILE-.,R0
                                        ;
                                        ;
002010' 005302                          20$:    DEC     R2                      ; PARAMTER IS NEVER > 1
002012' 001403                                  BEQ     30$                     ; GO TO WORK
002014' 062700  000006                          ADD     #6,R0
002020' 000773                                  BR      20$                     ; GO ROUND
                                        ;
002022' 012701  177777                  30$:    MOV     #177777,R1              ; && STAND-IN FOR PHSICAL ADDR
002026' 010120                                  MOV     R1,(R0)+
002030' 010120                                  MOV     R1,(R0)+
002032' 010120                                  MOV     R1,(R0)+
                                        ;
                                        ;
                                        ; ***************************************
                                        ; ***** LOAD THE STATION ADDRESS RAM *****
                                        ; ***************************************
                                        ;
002034' 012737  000200  177776                  MOV     #MODE,@#CMDREG          ; SET LOOP TO LOAD SA RAM
002042' 012737  000004  177774                  MOV     #LOOP,@#MODREG
                                        ;
002050' 010701                                  MOV     PC,R1                   ; FORM A POS/IND ADDRES
002052' 062701  000256                          ADD     #BFILE-.,R1             ; ADD THE OFFSET
                                        ;
002056' 005000                                  CLR     R0                      ; THIS CLEARS HIGH BITS
                                        ;
002060' 012746  000120                          MOV     #ARAM+20,-(SP)          ; FOR MODE REG/NEED A REGISTER
                                        ;
002064' 012702  000003                          MOV     #3,R2                   ; 3 WORDS PER ADDRESS/PATTERN
002070' 012704  000020                  40$:    MOV     #16.,R4                 ; SIXTEEN BITS PER WORD
002074' 012705  000014                  50$:    MOV     #12.,R5                 ; 12 POSITIONS IN SA RAM
                                        ;
002100' 010103                                  MOV     R1,R3                   ; COPY THE POINTER
                                        ;
002102' 006013                          60$:    ROR     (R3)                    ; GET LSB OF ALL 12 ADDRESSES
002104' 006100                                  ROL     R0                      ; R0 WILL HOLD ORTHOGONAL WORD
002106' 062703  000006                          ADD     #6,R3                   ; 6 BYTES PER ADDRES/PATTERN
002112' 077505                                  SOB     R5,60$                  ; GO TILL DONE
                                        ;
002114' 011637  177776                          MOV     (SP),@#CMDREG           ; SET MODE TO WRITE SA RAM
002120' 010037  177774                          MOV     R0,@#ADRREG             ; ORTHOGONAL WORD TO SA RAM
002124' 005216                                  INC     (SP)                    ; BUMP STATION ADDRESS
002126' 077416                                  SOB     R4,50$                  ; DO ANOTHER ONE
                                        ;
002130' 062701  000002                          ADD     #2,R1                   ; ADVANCE TO NEXT WORD
002134' 077223                                  SOB     R2,40$                  ; LOOP TILL DONE
                                        ;
002136' 012600                                  MOV     (SP)+,R0                ; POP TO BYTE BUCKET
                                        ;
                                        ;
                                        ; ***************************************************
                                        ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                        ; ***************************************************
                                        ;
002140' 012700  100000                          MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
002144' 010067  176472                          MOV     R0,RBUFP
002150' 005020                          70$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
002152' 020027  104000                          CMP     R0,#LINADR+SIZ1K        ; FILL THE BUFFER
```

```
002156' 103774                           BLO     70$                       ; FILL ENTIRE BUFFER
                                ;
                                ;
                                ; ***************************************************
                                ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                ; ***************************************************
                                ;
002160' 012703  000377                   MOV     #0377,R3                  ; WORST CASE FOR CLOCKING
002164' 010067  176454                   MOV     R0,TBUFP
002170' 010320                  80$:     MOV     R3,(R0)+                  ; FILL XMIT BUFFER WITH PATTERN
002172' 020027  110000                   CMP     R0,#LINADR+SIZ2K          ; STOP AT TOP
002176' 103774                           BLO     80$
                                ;
                                ;
                                ; ********************************************************
                                ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                ; ********************************************************
                                ;
002200' 012737  100020  177776           MOV     #ENABLE+20,@#CMDREG       ; ENABLE LINK
                                ;
002206' 016701  176432                   MOV     TBUFP,R1                  ; POINT AT XMIT BUFFER
002212' 005021                           CLR     (R1)+                     ; CLEAR OUT STATUS WORD
002214' 012721  000100                   MOV     #MINBC,(R1)+              ; SET BYTE COUNT TO MIN ALLOWED
                                ;
002220' 012700  177777                   MOV     #177777,R0                ; && STAND IN PHYSICAL ADDRESS
002224' 010021                           MOV     R0,(R1)+
002226' 010021                           MOV     R0,(R1)+
002230' 010021                           MOV     R0,(R1)+
                                ;
002232' 005037  021034                   CLR     @#CLRFIF                  ; CLEAR THE FIFO
002236' 005067  176360                   CLR     FLG5                      ; CLEAR INTERRUPT FLAG
002242' 016737  176374  021032           MOV     RBUFP,@#LFRBUF            ; TELL UNA WHERE RECEIVE BUF IS
002250' 016737  176370  021030           MOV     TBUFP,@#LTAC              ; TELL UNA WHERE XMIT BUF IS
                                ;
002256' 106427  000140                   MTPS    #PRI03                    ; ALLOW XMIT AND REC TO INTER
002262' 026727  176336  000002  90$:     CMP     SANTIM,#2                 ; WAIT FOR COUNTS TO ACCUMULATE
002270' 002015                           BGE     120$
002272' 032767  000100  176322           BIT     #RCVFLG,FLG5              ; THIS SETS IF DATAGRAM LOOPS
002300' 001770                           BEQ     90$
                                ;
002302' 032767  000040  176312  100$:    BIT     #TRNFLG,FLG5              ; WAIT FOR XMIT TOO
002310' 001774                           BEQ     100$
                                ;
002312' 000241                           CLC
002314' 112767  000003  176305  110$:    MOVB    #3,PCSR1+1                ; TELL HIM WHAT TEST WE'RE IN
002322' 000207                           RTS     PC
                                ;
                                ;
002324' 000261                  120$:    SEC                               ;INDICATE ERROR
002326' 000772                           BR      110$                      ; GO BACK TO MICROMONITOR
                                ;
                                ;
002330' 000000  000000  000000  BFILE:: .WORD    0,0,0
002336' 000000  000000  000000          .WORD    0,0,0
002344' 000000  000000  000000          .WORD    0,0,0
002352' 000000  000000  000000          .WORD    0,0,0
```

```
   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
```

```
                                        ;
                                        .SBTTL  MULTICAST ADDRESS TEST
                                        ;
                                        ; ***********************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; ***********************************
                                        ;
002440' 112767  000002  176160  MICE4:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
002446' 016737  176154  021020          MOV     PCSR1,@#IPCSR1
                                        ;
002454' 005067  176144                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                        ;
                                        ;
                                        ; ****************************************************************
                                        ; ***** FILL STATION ADDRESS RAM WITH KNOWN (PHONEY) ADDRESS *****
                                        ; ****************************************************************
                                        ;
002460' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; SET LOOP TO LOAD SA RAM
002466' 012737  000004  177774          MOV     #LOOP,@#MODREG
                                        ;
002474' 012704  000060                  MOV     #48.,R4                 ; COUNTER
002500' 005000                          CLR     R0
                                        ;
002502' 012701  000120                  MOV     #ARAM+20,R1             ; STATION ADDRESS STARTS AT +20
002506' 010137  177776          10$:    MOV     R1,@#CMDREG             ; SELECT STATION ADDRESS RAM
002512' 010037  177774                  MOV     R0,@#ADRREG             ; PARK IN RAM
002516' 005201                          INC     R1                      ; NEXT LOCATION
002520' 077406                          SOB     R4,10$                  ; DO THEM ALL
                                        ;
                                        ;
                                        ; ***************************************************
                                        ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                        ; ***************************************************
                                        ;
002522' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
002526' 010067  176110                  MOV     R0,RBUFP
002532' 005020                  20$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
002534' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL THE BUFFER
002540' 103774                          BLO     20$                     ; FILL ENTIRE BUFFER
                                        ;
                                        ;
                                        ; **************************************************
                                        ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                        ; **************************************************
                                        ;
002542' 012703  000377                  MOV     #0377,R3                ; WORST CASE FOR CLOCKING
002546' 010067  176072                  MOV     R0,TBUFP
002552' 010320                  30$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
002554' 020027  110000                  CMP     R0,#LINADR+SIZ2K        ; STOP AT TOP
002560' 103774                          BLO     30$
                                        ;
                                        ;
                                        ; *******************************************************
                                        ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                        ; *******************************************************
                                        ;
```

```
002562' 012737  000200  177776               MOV     #MODE,@#CMDREG          ; ENABLE LINK, SELECT MODE
002570' 012737  040004  177774               MOV     #ENAL!LOOP,@#MODREG     ; ENABLE LOOPBACK
002576' 012737  100020  177776               MOV     #ENABLE+20,@#CMDREG     ; POINTER TO FRONT OF RAM
                                     ;
002604' 016701  176034                       MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
002610' 005021                               CLR     (R1)+                   ; CLEAR OUT STATUS WORD
002612' 012721  000100                       MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                     ;
                                     ;
                                     ; ***********************************************
                                     ; ***** GET PHYSICAL ADDRESS INTO DATAGRAM *****
                                     ; ***********************************************
                                     ;
002616' 012700  177777                       MOV     #177777,R0              ; && STAND-IN FOR PHYSICAL ADDR
002622' 010011                               MOV     R0,(R1)
002624' 052721  000001                       BIS     #01,(R1)+               ; SET MULTICAST ADDRESS BIT
002630' 010021                               MOV     R0,(R1)+
002632' 010021                               MOV     R0,(R1)+
                                     ;
                                     ;
002634' 005037  021034                       C.R     @#CLRFIF                ; CLEAR THE FIFO
002640' 005067  175756                       CLR     FLG5                    ; CLEAR INTERRUPT FLAG
002644' 016737  175772  021032               MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002652' 016737  175766  021030               MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                     ;
002660' 106427  000140                       MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
002664' 026727  175734  000002       40$:    CMP     SANTIM,#2               ; && LEAVE IT 2 SECONDS
002672' 002015                               BGE     60$                     ; EXIT IF TIMER DONE
002674' 032767  000100  175720               BIT     #RCVFLG,FLG5            ; WAIT FOR RECEIVER INTERRUPT
002702' 001770                               BEQ     40$                     ;
                                     ;
002704' 032767  000040  175710       45$:    BIT     #TRNFLG,FLG5            ; WAIT FOR XMIT INT TOO!
002712' 001774                               BEQ     45$
                                     ;
002714' 000241                               CLC                             ;INDICATE SUCCESS
002716' 112767  000004  175703       50$:    MOVB    #4,PCSR1+1              ; TELL HOST WE ARE DONE
002724' 000207                               RTS     PC
                                     ;
                                     ;
002726' 000261                       60$:    SEC                             ;INDICATE ERROR
002730' 000772                               BR      50$                     ; RETURN TO MICROMONITOR
                                     ;
                                     ;
```

```
   837                              ;
   838  002732' 002734              MICESZ::MICESZ-MICROE+2
   839                              ;
   840          000001              .END
```

```
ACLO  = 002000             87#
ADRERR= 000001            130#
ADRREG= 177774             34#     365*     482*     656*     763*
AFILE   001320R   002     329      347      439#
ARAM  = 000100             80#     351      480      642      761
BFILE   002330RG  002     603      616      638      726#
BIT0  = 000001             54#     103
BIT1  = 000002             53#     104
BIT10 = 002000             44#      56       73       87
BIT11 = 004000             43#      72       86
BIT12 = 010000             42#      56       71      113
BIT13 = 020000             41#      70      112      114
BIT14 = 040000             40#      69       85      111      238
BIT15 = 100000             39#      68       78       84      110      243      246
BIT2  = 000004             52#      91      105
BIT3  = 000010             51#      90      106
BIT4  = 000020             50#      89      107
BIT5  = 000040             49#      88      108
BIT6  = 000100             48#      80      109
BIT7  = 000200             47#      79
BIT8  = 000400             46#      56       74
BIT9  = 001000             45#
CLRFIF= 021034             22#     411*     530*     703*     814*
CMDREG= 177776             35#     136*     343*     364*     401*     474*     481*     513*     634*     655*     692*     755*     762*
                          794*     796*

COLL  = 000020             89#
CSRFLG= 000001            103#     194      225
CSRVEC= 000064             96#     161*     162*
CSRWRT  000436R   002     160      225#
DATERR= 000000            129#
DMACSR= 021002              9#     237      246*
DMAF  = 021022             17#
DMAINT  000472R   002     155      237#
DMAT0 = 021004             10#
DMAT1 = 021006             11#
DMAVEC= 000114             97#     156*     157*
DMAWC = 021024             18#
DMDONE  000640R   002     237*     238      243      273#
DNI   = 004000             72#     140      218
DRTY  = 000040             88#
DTCR  = 000010             90#
ENABLE= 100000             78#     401      513      692      796
ENAL  = 040000             85#     795
ENCR  = 004000             86#
ERRFLG= 000002            104#     228
ERRINT  000446R   002     142      228#
FATI  = 000400             74#
FLG5    000622R   002     187*     190      194      219*     225*     228*     250*     257*     260*     268#     412*     428      431
                          531*     538      546      704*     711      714      815*     822      825

INERR = 000003            102#     216      417
INMON = 000001            100#     138      202      214
INTST = 000002            101#     291      425      464      577      745
IOADR = 020000            124#     125
IOSIZ = 020000            120#     125
IPCSR0= 021000              8#     140*     199      218*     229*     240*     245*     251*
IPCSR1= 021020             16#     139*     202*     217*     292*     418*     426*     465*     578*     746*
```

```
IPCSR2  000630R     002     185*    186*    271#    299     300     587     588
LASFTP= 012400               56#
LFRBUF= 021032               21#    413*    532*    705*    816*
LINADR= 100000              126#    379     382     393     491     494     505     670     673     684     772     775     786
LINSIZ= 077774              122#
LOOP  = 000004               91#    344     475     635     756     795
LRBUF = 021044               26#    255
LTAC  = 021030               20#    414*    533*    706*    817*
MAXBC = 002752              128#
MDMAR0= 021014               14#    301     302     307     312     317     589     590     595
MDMAR1= 021016               15#
MDMAW0= 021026               19#
MDMAW1= 021036               23#
MDMA0 = 021010               12#    299*    304*    587*    592*
MDMA1 = 021012               13#    300*    305*    588*    593*
MDMSR = 021042               25#
MICESZ  002732RG    002     838#
MICE1   000646R     002     263     291#
MICE2   001430R     002     264     464#
MICE3   001702R     002     265     577#
MICE4   002440R     002     266     745#
MICROE  000000RG    002     135#    838
MINBC = 000100              127#    405     517     696     800
MODE  = 000200               79#    343     474     634     755     794
MODREG= 177774               33#    344*    475*    635*    756*    795*
MTCH  = 020000              114#    422
NPRERR= 100000              110#    245
NPRFLG= 000020              107#
NXMERR= 040000              111#    240
NXMFLG= 000010              106#
PARERR= 010000              113#    251
PARFLG= 000004              105#    250
PARINT  000550R     002     150     250#
PARVEC= 000140               98#    151*    152*
PCBADR  000634R     002     272#
PCEI  = 040000               69#
PCSRSW= 021040               24#    179
PCSR1   000626R     002     138*    139     214*    216*    217     270#    291*    292     417*    418     425*    426     434*
                            464*    465     558*    577*    578     718*    745*    746     829*
PHYAD0= 021060               27#
PHYAD1= 021062               28#
PHYAD2= 021064               29#
PHYAD3= 021066               30#
PHYAD4= 021070               31#
PHYAD5= 021072               32#
PRI00 = 000000               64#    188*
PRI01 = 000040               63#
PRI02 = 000100               62#
PRI03 = 000140               61#    416*    535*    708*    819*
PRI04 = 000200               60#    162     172
PRI05 = 000240               59#    167     177
PRI06 = 000300               58#
PRI07 = 000340               57#    135*    145     152     157     193*
PROM  = 100000               84#
RBUFP   000642R     002     274#    380*    413     492*    532     671*    705     773*    816
RCEI  = 002000               73#
```

```
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
S1ACK = 001000               99#    137
TBLD    000612R     002     208     263#
TBUFP   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
WCSSIZ= 020000              119#    124
.     = 002734R     002     142     150     155     160     165     170     175     196     208     231     247     253     263
                            264     265     266     329     347     603     616     638


. ABS.  000000      000
        000000      001
MICRE   002734      002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROE.OBJ,MICROE.LST/CRF/DOC/NL:TOC/SOL=MICROE.MAC
 RUN-TIME: .9 1 .3 SECONDS
 RUN-TIME RATIO: 86/3=28.7
 CORE USED:  9K  (18 PAGES)

 DOCUMENT PAGES:  22
```

```
                        .TITLE  MICROF - MICROCODE MODULE F
                        ; && DEDICATED THE CRC CIRCUITRY TO THE RECEIVE SIDE OF THE LINK

        000000'         .CSECT MICRF

                        .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000          IPCSRO  =       21000   ;INTERNAL PCSRO ADDRESS
        021002          DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004          DMATO   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006          DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010          MDMAO   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012          MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014          MDMARO  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016          MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020          IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022          DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024          DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026          MDMAWO  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030          LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032          LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034          CLRFIF  =       21034   ;CLEAR FIFO
        021036          MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040          PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042          MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044          LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060          PHYADO  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062          PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064          PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066          PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070          PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072          PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774          MODREG  =       177774  ;LINK MODE REGISTER
        177774          ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776          CMDREG  =       177776  ;LINK COMMAND REGISTER

                        .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000          BIT15   =       100000
        040000          BIT14   =       40000
        020000          BIT13   =       20000
        010000          BIT12   =       10000
        004000          BIT11   =       4000
        002000          BIT10   =       2000
        001000          BIT9    =       1000
        000400          BIT8    =       400
        000200          BIT7    =       200
        000100          BIT6    =       100
        000040          BIT5    =       40
        000020          BIT4    =       20
        000010          BIT3    =       10
        000004          BIT2    =       4
        000002          BIT1    =       2
        000001          BIT0    =       1
                        ;
        012400          LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340                  PRI07   =       340
        000300                  PRI06   =       300
        000240                  PRI05   =       240
        000200                  PRI04   =       200
        000140                  PRI03   =       140
        000100                  PRI02   =       100
        000040                  PRI01   =       40
        000000                  PRI00   =       0
                                ;
                                ;PCSRO - PORT CONTROL STATUS REGISTER 0
                                ;
        100000                  SERI    =       BIT15
        040000                  PCEI    =       BIT14
        020000                  RXI     =       BIT13
        010000                  TXI     =       BIT12
        004000                  DNI     =       BIT11
        002000                  RCEI    =       BIT10
        000400                  FATI    =       BIT8
                                ;
                                ;LINK COMMAND REGISTER
                                ;
        100000                  ENABLE  =       BIT15   ;ENABLE LINK MODULE
        000200                  MODE    =       BIT7    ;ENABLE MODE REGISTER
        000100                  ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                                ;
                                ;LINK MODE REGISTER
                                ;
        100000                  PROM    =       BIT15   ;PROMISCUIOUS MODE
        040000                  ENAL    =       BIT14   ;ENABLE MULTICAST
        004000                  ENCR    =       BIT11   ;ENABLE COLLISION TEST
        002000                  ACLO    =       BIT10   ;ENABLE ACLO
        000040                  DRTY    =       BIT5    ;DISABLE RETRY LOGIC
        000020                  COLL    =       BIT4    ;SIMULATE A COLLISION
        000010                  DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
        000004                  LOOP    =       BIT2    ;ENABLE LOOPBACK
        000001                  HDX     =       BIT0    ;HALF DUPLEX BIT

        000070                  TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
        000120                  RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
        000134                  SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064                  CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114                  DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140                  PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000                  STACK=  1000            ;STACK LOCATION
        000001                  INMON=  1               ;IN MICROMONITOR STATE
        000002                  INTST=  2               ;IN A TEST STATE
        000003                  INERR=  3               ;IN ERROR STATE
        000001                  CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002                  ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004                  PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010                  NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020                  NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
        000040                  TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
        000100                  RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
        100000                  NPRERR= BIT15           ;PCSRO FLAG INDICATING NPR ERROR OCCURRED
        040000                  NXMERR= BIT14           ;PCSRO FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000                  UNIERR= BIT13           ;PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
        010000                  PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        004000                  SIZ1K=  4000            ;1K WORDS
        010000                  SIZ2K=  SIZ1K*2         ;2K WORDS
        020000                  SIZ4K=  SIZ2K*2         ;4K WORDS
        040000                  SIZ8K=  SIZ4K*2         ;8K WORDS
        020000                  WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000                  IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000                  ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774                  LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
        000000                  WCSADR= 0               ;BASE ADDRESS OF WCS
        020000                  IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000                  ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000                  LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        000100                  MINBC=  64.             ; 64 BYTES
        002752                  MAXBC=  1518.-4.        ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
        000000                  DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000001                  ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
                                ;
        177777                  INITH   =       -1      ; MINUS ONE     ( INITIAL CRC VALUE)
        177777                  INITL   =       -1
        166670                  POLYH   =       166670          ; FUNCTION POLYNOMIAL HIGH WORD
        101440                  POLYL   =       101440          ; FUNCTION POLYNOMIAL LOW WORD
        157273                  EXCRCH  =       157273          ; EXPECTED CRC HIGH
        020343                  EXCRCL  =       020343          ; EXPECTED CRC LOW
```

```

                                                .SBTTL  MICROCODE MODULE F

000000' 106427  000340                  MICROF::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776                  MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                          MOV     #STACK,SP               ;SETUP STACK
000016' 112767  000001  000604                  MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000600  021020                  MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000                  MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                                  MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000404                          ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                                  CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                          10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                          MOV     #PRI07,(R1)+
000056' 020127  001000                          CMP     R1,#1000
000062' 002772                                  BLT     10$

000064' 010700                                  MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000462                          ADD     #PARINT-.,R0
000072' 010037  000140                          MOV     R0,@#PARVEC
000076' 012737  000340  000142                  MOV     #PRI07,@#PARVEC+2

000104' 010700                                  MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000364                          ADD     #DMAINT-.,R0
000112' 010037  000114                          MOV     R0,@#DMAVEC
000116' 012737  000340  000116                  MOV     #PRI07,@#DMAVEC+2

000124' 010700                                  MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000310                          ADD     #CSRWRT-.,R0
000132' 010037  000064                          MOV     R0,@#CSRVEC
000136' 012737  000200  000066                  MOV     #PRI04,@#CSRVEC+2

000144' 010700                                  MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000316                          ADD     #TIMINT-.,R0
000152' 010037  000134                          MOV     R0,@#SANVEC
000156' 012737  000240  000136                  MOV     #PRI05,@#SANVEC+2

000164' 010700                                  MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
000166' 062700  000414                          ADD     #TRNINT-.,R0
000172' 010037  000070                          MOV     R0,@#TRNVEC
000176' 012737  000200  000072                  MOV     #PRI04,@#TRNVEC+2

000204' 010700                                  MOV     PC,R0                   ;SETUP RECEIVER VECTOR
000206' 062700  000360                          ADD     #RCVINT-.,R0
000212' 010037  000120                          MOV     R0,@#RCVVEC
000216' 012737  000240  000122                  MOV     #PRI05,@#RCVVEC+2

000224' 013700  021040                          MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000230' 052700  176000                          BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000234' 006300                                  ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000236' 006300                                  ASL     R0
000240' 006300                                  ASL     R0
000242' 062700  000004                          ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000246' 010067  000360                          MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000252' 012767  000003  000354                  MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000260' 005067  000340                          CLR     FLG6                    ;INITIALIZE FLAG WORD
```

```
000264' 106427  000000            15$:    MTPS    #PRI00                  ;ALLOW INTERRUPTS

000270' 005767  000330            20$:    TST     FLG6                    ;WAIT FOR A COMMAND FROM HOST
000274' 001775                            BEQ     20$

000276' 106427  000340                    MTPS    #PRI07                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000302' 032767  000001  000314            BIT     #CSRFLG,FLG6            ;DID HOST GIVE US A COMMAND?
000310' 001001                            BNE     30$                     ;YES
000312' 000777                            BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                          ;FOR LACK OF ANYTHING BETTER TO DO

000314' 113700  021000            30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000320' 042700  177760                    BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000324' 001004                            BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000326' 012737  000001  021020            MOV     #INMON,@#IPCSR1         ;YES, CLEAR OUT THE TEST # BITS
000334' 000432                            BR      50$
000336' 022700  000017            35$:    CMP     #17,R0                  ;RESTART OPERATIONAL MICROCODE?
000342' 001432                            BEQ     60$                     ;YES
000344' 162700  000001                    SUB     #1,R0
000350' 010701                            MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000352' 062701  000240                    ADD     #TBLD-.,R1
000356' 006300                            ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000360' 060001                            ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000362' 061101                            ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000364' 004711                            JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000366' 103404                            BCS     40$                     ;ERROR OCCURRED
000370' 112767  000001  000232            MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000376' 000403                            BR      45$                     ;MICROMONITR
000400' 112767  000003  000222    40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000406' 016737  000216  021020    45$:    MOV     PCSR1,@#IPCSR1
000414' 012737  004000  021000            MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000422' 005067  000176            50$:    CLR     FLG6                    ;RESET FLAG WORD
000426' 000716                            BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000430' 005000                    60$:    CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000432' 000137  040006                    JMP     @#40006                 ;START OPERATIONAL MICROCODE

000436' 052767  000001  000160    CSRWRT: BIS     #CSRFLG,FLG6            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000444' 000002                            RTI

000446' 052767  000002  000150    ERRINT: BIS     #ERRFLG,FLG6            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000454' 012737  020000  021000            MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                          ;HAPPENED
                                                                          ;JUST SIT HERE AND SPIN WHEELS
000462' 000777                            BR      .                       ;COUNT ON HOST TO TIMEOUT

000464' 005267  000136            TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000470' 000002                            RTI

000472' 013767  021002  000142    DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000500' 032767  040000  000134            BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000506' 001404                            BEQ     10$                     ;NO
000510' 012737  040000  021000            MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                          ;LOCATION WAS ADDRESSED
000516' 000407                            BR      20$
000520' 032767  100000  000114    10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000526' 001407                          BEQ     30$                     ;NO
000530' 012737  100000  021000          MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
000536' 012737  100000  021002  20$:    MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000544' 000777                          BR      .                       ;SIT HERE AND SPIN WHEELS
000546' 000002                  30$:    RTI

000550' 052767  000004  000046  PARINT: BIS     #PARFLG,FLG6            ;SET PARITY ERROR OCCURRED
000556' 012737  010000  021000          MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                        ;OCCURRED
000564' 000777                          BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                        ;COUNT ON HOST TO TIMEOUT

000566' 005737  021044          RCVINT: TST     @#LRBUF                 ;READ BUFFER DONE REGISTER...
                                                                        ;WHICH CLEARS THE INTERRUPT
000572' 052767  000100  000024          BIS     #RCVFLG,FLG6            ;SET RECEIVER INTERRUPT OCCURRED
000600' 000002                          RTI

000602' 052767  000040  000014  TRNINT: BIS     #TRNFLG,FLG6            ;SET TRANSMITTER INTERRUPT OCCURRED
000610' 000002                          RTI

000612' 000170                  TBLD:   .WORD   MICF1-.                 ; CRC DATA PATTERN
000614' 000460                          .WORD   MICF2-.                 ; CRC ERROR TEST
000616' 001032                          .WORD   MICF3-.                 ; CRC PATTERN LENGTH
000620' 001544                          .WORD   MICF4-.                 ; BUFFER RECOVERY TEST
000622' 002222                          .WORD   MICF5-.                 ;HALF-DUPLEX TEST

000624' 000000                  FLG6:   .WORD   0                       ;FLAG WORD
000626' 000000                  SANTIM: .WORD   0                       ;COUNT FOR SANITY TIMER
000630' 000000                  PCSR1:  .WORD   0                       ;COPY OF WHAT GOES TO PCSR1
000632' 000000  000000          IPCSR2: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000636' 000000  000000          PCBADR: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000642' 000000                  DMDONE: .WORD   0
000644' 000000                  RBUFP:  .WORD   0                       ;POINTER TO RECIEVE BUFFER
000646' 000000                  TBUFP:  .WORD   0                       ;POINTER TO XMIT BUFFER
000650' 000000                  CRC1:   .WORD   0                       ;HOLDS LAST DATA BYTE AND 1ST CRC BYTE
000652' 000000                  CRC23:  .WORD   0                       ;HOLDS 2ND AND 3RD CRC BYTES
000654' 000000                  CRC4:   .WORD   0                       ;HOLDS 4TH CRC BYTE
```

```

                                        ;
                                        ;
                                        ; ***********************
                                        ; ***** SUBROUTINES *****
                                        ; ***********************
                                        ;
 000656' 106746                         BLKCRC: MFPS    -(SP)                           ; SAVE PSW
 000660' 010046                                 MOV     R0,-(SP)                        ; SAVE R0
 000662' 010146                                 MOV     R1,-(SP)
 000664' 010246                                 MOV     R2,-(SP)
 000666' 012704  177777                         MOV     #INITH,R4                       ; INITIAL CRC HIGH WORD
 000672' 012705  177777                         MOV     #INITL,R5                       ; INITIAL CRC LOW WORD
 000676' 112001                         10$:    MOVB    (R0)+,R1                        ; GET NEXT BYTE
 000700' 004767  000014                         JSR     PC,GETCRC                       ; CALCULATE CRC
 000704' 077204                         20$:    SOB     R2,10$                          ; LOOP TILL DONE
 000706' 012602                                 MOV     (SP)+,R2
 000710' 012601                                 MOV     (SP)+,R1
 000712' 012600                                 MOV     (SP)+,R0
 000714' 106426                                 MTPS    (SP)+
 000716' 000207                                 RTS     PC
                                        ;
                                        ;
 000720' 106746                         GETCRC: MFPS    -(SP)                           ; SAVE REGISTERS
 000722' 010146                                 MOV     R1,-(SP)
 000724' 010246                                 MOV     R2,-(SP)
 000726' 010346                                 MOV     R3,-(SP)
 000730' 042701  177400                         BIC     #^C377,R1                       ; CLEAR HIGH BYTE
 000734' 074105                                 XOR     R1,R5                           ; MERGE NEW BYTE WITH OLD CRC
 000736' 012702  166670                         MOV     #POLYH,R2                       ; GET POLYNOMIAL HIGH WORD
 000742' 012703  101440                         MOV     #POLYL,R3                       ; GET CRC POLYNOMIAL LOW WORD
 000746' 012701  000010                         MOV     #8.,R1                          ; LOOP COUNT
 000752' 000241                         10$:    CLC                                     ; CLEAR THE CARRY
 000754' 006004                                 ROR     R4                              ; SHIFT RIGHT THE CRC
 000756' 006005                                 ROR     R5                              ; 32 BITS WORTH
 000760' 103002                                 BCC     20$                             ; SKIP IF BIT 0 NOT SET
 000762' 074204                                 XOR     R2,R4                           ; EXCLUSIVE OR IN THE POLY
 000764' 074305                                 XOR     R3,R5                           ; ROTH HIGH AND LOW WORDS
 000766' 077107                         20$:    SOB     R1,10$                          ;  ND LOOP ON ALL 8 BITS
 000770' 012603                                 MOV     (SP)+,R3                        ; RESTORE REGISTERS
 000772' 012602                                 MOV     (SP)+,R2
 000774' 012601                                 MOV     (SP)+,R1
 000776' 106426                                 MTPS    (SP)+
 001000' 000207                                 RTS     PC
                                        ;
                                        ;
```

```
                                      ;
                                      .SBTTL  CRC DATA PATTERN TEST
                                      ;
                                      ; ***********************************
                                      ; ***** TELL HIM WE ARE TESTING *****
                                      ; ***********************************
                                      ;
001002' 112767  000002  177620        MICF1:  MOVB    #INTST,PCSR1              ; TELL HOST WE ARE TESTING
001010' 016737  177614  021020                MOV     PCSR1,@#IPCSR1
                                      ;
                                      ; ********************************************
                                      ; ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                      ; ********************************************
                                      ;
001016' 016737  177610  021010                MOV     IPCSR2,@#MDMA0            ; SET TO GET HOST PCBB ADDRESS
001024' 016737  177604  021012                MOV     IPCSR2+2,@#MDMA1
001032' 013700  021014                        MOV     @#MDMAR0,R0               ; R5 NOW CONTAINS PCBB LOW
001036' 013701  021014                        MOV     @#MDMAR0,R1               ; R1 NOW CONTAINS PCBB HIGH
001042' 010037  021010                        MOV     R0,@#MDMA0                ; POINT AT PCBB
001046' 010137  021012                        MOV     R1,@#MDMA1
001052' 013703  021014                        MOV     @#MDMAR0,R3               ; R3 NOW HOLDS DATA PATTERN
                                      ;
                                      ;
                                      ; *****************************************************
                                      ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                      ; *****************************************************
                                      ;
001056' 012700  100000                        MOV     #LINADR,R0                ; RECEIVE BUFFER STARTS HERE
001062' 010067  177556                        MOV     R0,RBUFP
001066' 005020                        10$:    CLR     (R0)+                     ; FILL RECEIVE BUFFER WITH ZEROS
001070' 020027  104000                        CMP     R0,#LINADR+SIZ1K          ; STOP AT THE TOP
001074' 103774                                BLO     10$
                                      ;
                                      ;
                                      ; *********************************************
                                      ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                      ; *********************************************
                                      ;
001076' 010067  177544                        MOV     R0,TBUFP                  ; SAVE COPY OF ADDRESS
001102' 010320                        20$:    MOV     R3,(R0)+                  ; FILL XMIT BUFFER WITH PATTERN
001104' 020027  110000                        CMP     R0,#LINADR+SIZ2K
001110' 103774                                BLO     20$
                                      ;
                                      ;
                                      ; ********************************************
                                      ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                      ; ********************************************
                                      ;
001112' 012737  000200  177776                MOV     #MODE,@#CMDREG            ; ENABLE LINK, SELECT MODE REG
001120' 012737  100004  177774                MOV     #PROM!LOOP,@#MODREG       ; PROM MODE AND LOOPBACK
001126' 012737  100000  177776                MOV     #ENABLE,@#CMDREG          ; ENABLE IT
                                      ;
001134' 016701  177506                        MOV     TBUFP,R1                  ; POINT AT XMIT BUFFER
001140' 005021                                CLR     (R1)+                     ; CLEAR OUT STATUS WORD
001142' 012721  002752                        MOV     #MAXBC,(R1)+              ; CLEAR OUT THE FIFO
001146' 005037  021034                        CLR     @#CLRFIF                  ; CLEAR THE FIFO
```

```
001152' 005067  177446                          CLR     FLG6
001156' 016737  177462  021032                  MOV     RBUFP,@#LFRBUF          ; TELL DEUNA RECEIVE BUF LOC
001164' 016737  177456  021030                  MOV     TBUFP,@#LTAC            ; TELL DEUNA WHERE XMIT BUF IS
                                        ;
001172' 106427  000140                          MTPS    #PRI03                  ; ALLOW INTERRUPTS
001176' 032767  000100  177420  30$:            BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
001204' 001774                                  BEQ     30$
                                        ;
001206' 032767  000040  177410  40$:            BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INTERRUPT TOO
001214' 001774                                  BEQ     40$
                                        ;
001216' 106427  000340                          MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                        ;
                                        ;
                                        ; ************************************************************
                                        ; ***** CALCULATE THE CRC ON THE RECEIVER BUFFER CONTENTS *****
                                        ; ************************************************************
                                        ;
001222' 016700  177416                          MOV     RBUFP,R0                ; POINT AT RECEIVE BUFFER
001226' 062700  000004                          ADD     #4,R0                   ; BY-PASS THE STATUS AND BC
001232' 012702  002756                          MOV     #MAXBC+4,R2             ; INCLUDE CRC BYTES
001236' 004767  177414                          JSR     PC,BLKCRC
001242' 020427  157273                          CMP     R4,#EXCRCH
001246' 001010                                  BNE     50$                     ; ERROR IF NOT
001250' 020527  020343                          CMP     R5,#EXCRCL
001254' 001005                                  BNE     50$
001256' 000241                                  CLC                             ;INDICATE SUCCESS
001260' 112767  000001  177343  45$:            MOVB    #1,PCSR1+1              ; TELL WHAT TEST WE JUST FINISHED
001266' 000207                                  RTS     PC
                                        ;
                                        ;
                                        ; *********************
                                        ; ***** ERROR EXIT *****
                                        ; *********************
                                        ;
001270' 000261                          50$:    SEC                             ; TELL HIM WE MADE ERROR
001272' 000772                                  BR      45$                     ; LEAVE

```

```
                                ;
                                .SBTTL  CRC ERROR TEST
                                ;
                                ; ***********************************
                                ; ***** TELL HIM WE ARE TESTING *****
                                ; ***********************************
                                ;
001274' 112767  000002  177326  MICF2:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001302' 016737  177322  021020          MOV     PCSR1,@#IPCSR1
                                ;
                                ;
                                ; *********************************************
                                ; ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                ; *********************************************
                                ;
001310' 016737  177316  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
001316' 016737  177312  021012          MOV     IPCSR2+2,@#MDMA1
001324' 013704  021014                  MOV     @#MDMAR0,R4             ; R4 NOW CONTAINS PCBB LOW
001330' 013705  021014                  MOV     @#MDMAR0,R5             ; R5 NOW CONTAINS PCBB HIGH
001334' 010437  021010                  MOV     R4,@#MDMA0              ; POINT AT PCBB
001340' 010537  021012                  MOV     R5,@#MDMA1
001344' 013703  021014                  MOV     @#MDMAR0,R3             ; R3 NOW HOLDS DATA PATTERN
                                ;
                                ;
                                ; *******************************************************
                                ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                ; *******************************************************
                                ;
001350' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001354' 010067  177264                  MOV     R0,RBUFP
001360' 005020                  10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001362' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
001366' 103774                          BLO     10$
                                ;
                                ;
                                ; **********************************************
                                ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                ; **********************************************
                                ;
001370' 010067  177252                  MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
001374' 010320                  20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001376' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001402' 103774                          BLO     20$
                                ;
                                ;
                                ; ********************************************
                                ; ***** CALCULATE CRC ON TRANSMIT BUFFER *****
                                ; ********************************************
                                ;
001404' 016700  177236                  MOV     TBUFP,R0                ; POINT AT RECEIVE BUFFER
001410' 062700  000004                  ADD     #4,R0                   ; SKIP 1ST WORD AND BYTE COUNT
001414' 012702  002746                  MOV     #MAXBC-4,R2             ; DO A BUNCH
001420' 004767  177232                  JSR     PC,BLKCRC
001424' 016700  177216                  MOV     TBUFP,R0                ; POINT AT XMIT BUFFER
001430' 062700  000004                  ADD     #4,R0                   ; BY-PASS STATUS AND BC
001434' 062700  002746                  ADD     #MAXBC-4,R0             ; ADD BUFFER OFFSET
```

```
001440' 005104                                  COM     R4                      ; MUST SEND THE COMPLEMENT
001442' 005105                                  COM     R5
001444' 005205                                  INC     R5                      ; INTRODUCE AN ERROR IN CRC!
001446' 010520                                  MOV     R5,(R0)+                ; APPEND THE CRC WORD
001450' 010420                                  MOV     R4,(R0)+                ; APPEND THE OTHER CRC WORD
                                        ;
                                        ;
                                        ; **********************************************
                                        ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                        ; **********************************************
                                        ;
001452' 012737  000200  177776                  MOV     #MODE,@#CMDREG                  ; ENABLE, SELECT MODE
001460' 012737  100014  177774                  MOV     #PROM!LOOP!DTCR,@#MODREG        ; PROM MODE AND LOOPBACK
001466' 012737  100000  177776                  MOV     #ENABLE,@#CMDREG                ; ENABLE THE XMTR
                                        ;
001474' 016701  177146                          MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001500' 005021                                  CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001502' 012721  002752                          MOV     #MAXBC,(R1)+            ; CLEAR OUT THE FIFO
001506' 005037  021034                          CLR     @#CLRFIF                ; CLEAR THE FIFO
001512' 005067  177106                          CLR     FLG6
001516' 016737  177122  021032                  MOV     RBUFP,@#LFRBUF          ; TELL DEUNA RECEIVE BUF LOC
001524' 016737  177116  021030                  MOV     TBUFP,@#LTAC            ; TELL DEUNA WHERE XMIT BUF IS
                                        ;
001532' 106427  000140                          MTPS    #PRI03                  ; ALLOW INTERRUPTS
001536' 032767  000100  177060  30$:            BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
001544' 001774                                  BEQ     30$
                                        ;
001546' 032767  000040  177050  40$:            BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INTERRUPT TOO
001554' 001774                                  BEQ     40$
                                        ;
001556' 106427  000340                          MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                        ;
                                        ;
                                        ; ************************************************
                                        ; ***** WRITE STATUS REGISTER TO HOST MEMORY *****
                                        ; ************************************************
                                        ;
001562' 012700  100000                          MOV     #LINADR,R0              ; GET RECEIVE STATUS WORD
001566' 011003                                  MOV     (R0),R3                 ; GET STATUS WORD
                                        ;
001570' 016737  177036  021010                  MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
001576' 016737  177032  021012                  MOV     IPCSR2+2,@#MDMA1
001604' 013700  021014                          MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LOW
001610' 013701  021014                          MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
001614' 062700  000002                          ADD     #2,R0                   ; BUMP TO NEXT HOST WORD
001620' 005501                                  ADC     R1
001622' 010037  021010                          MOV     R0,@#MDMA0              ; POINT TO PCBB
001626' 010137  021012                          MOV     R1,@#MDMA1
001632' 010337  021026                          MOV     R3,@#MDMAW0             ; WRITE STATUS WORD TO HOST
                                        ;
                                        ;
001636' 112767  000002  176765                  MOVB    #2,PCSR1+1              ; TELL HIM WHICH TEST IT IS
001644' 000241                                  CLC
001646' 000207                                  RTS     PC
```

```
                                        ;
                                        .SBTTL  CRC PATTERN LENGTH TEST
                                        ;
                                        ; ***********************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; ***********************************
                                        ;
001650' 112767  000002  176752  MICF3:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001656' 016737  176746  021020          MOV     PCSR1,@#IPCSR1
                                        ;
                                        ;
                                        ; ********************************************
                                        ; ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                        ; ********************************************
                                        ;
001664' 016737  176742  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
001672' 016737  176736  021012          MOV     IPCSR2+2,@#MDMA1
001700' 013704  021014                  MOV     @#MDMAR0,R4             ; R4 NOW CONTAINS PCBB LOW
001704' 013705  021014                  MOV     @#MDMAR0,R5             ; R5 NOW CONTAINS PCBB HIGH
001710' 010437  021010                  MOV     R4,@#MDMA0              ; POINT AT PCBB
001714' 010537  021012                  MOV     R5,@#MDMA1
001720' 013703  021014                  MOV     @#MDMAR0,R3             ; R3 NOW HOLDS DATA PATTERN
                                        ;
                                        ;
                                        ; *******************************************************
                                        ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                        ; *******************************************************
                                        ;
001724' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001730' 010067  176710                  MOV     R0,RBUFP
001734' 005020                  10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001736' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
001742' 103774                          BLO     10$
                                        ;
                                        ;
                                        ; ********************************************
                                        ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                        ; ********************************************
                                        ;
001744' 010067  176676                  MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
001750' 010320                  20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001752' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001756' 103774                          BLO     20$
                                        ;
                                        ;
                                        ; *********************************************
                                        ; ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
                                        ; *********************************************
                                        ;
001760' 016737  176646  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
001766' 016737  176642  021012          MOV     IPCSR2+2,@#MDMA1
001774' 013704  021014                  MOV     @#MDMAR0,R4             ; R4 NOW CONTAINS PCBB LO
002000' 013705  021014                  MOV     @#MDMAR0,R5             ; R5 NOW CONTAINS PCBB HI
002004' 062704  000002                  ADD     #2,R4                   ; BUMP LOW BY TWO
002010' 005505                          ADC     R5
002012' 010437  021010                  MOV     R4,@#MDMA0              ; POINT AT PCBB
```

```
002016' 010537  021012                    MOV     R5,@#MDMA1
002022' 013703  021014                    MOV     @#MDMAR0,R3             ; R3 NOW HOLDS BYTE COUNT
                                  ;
                                  ;
                                  ; ********************************************
                                  ; ***** CALCULATE CRC ON TRANSMIT BUFFER *****
                                  ; ********************************************
                                  ;
002026' 016700  176614                    MOV     TBUFP,R0                ; POINT AT RECEIVE BUFFER
002032' 062700  000004                    ADD     #4,R0                   ; BY-PASS STATUS AND BC
002036' 010302                            MOV     R3,R2                   ; R3 CONTAINS BYTE COUNT
002040' 004767  176612                    JSR     PC,BLKCRC
002044' 016700  176576                    MOV     TBUFP,R0                ; POINT AT XMIT BUFFER
002050' 062700  000004                    ADD     #4,R0                   ; BY-PASS STATUS AND BC
002054' 060300                            ADD     R3,R0                   ; ADD BUFFER BYTE COUNT
002056' 005104                            COM     R4                      ; COMPLEMENT BEFORE SENDING
002060' 005105                            COM     R5
002062' 032700  000001                    BIT     #1,R0                   ;ARE WE LOOKING AT AN ODD BOUNDARY?
002066' 001427                            BEQ     25$                     ;NO, GOOD IT MAKES THINGS EASIER
                                  ;
002070' 162700  000001                    SUB     #1,R0                   ;POINT BACK TO EVEN BOUNDARY
002074' 011067  176550                    MOV     (R0),CRC1               ;GET DATA BYTE AND JUNK
002100' 110567  176545                    MOVB    R5,CRC1+1               ;REPLACE JUNK WITH 1ST CRC BYTE
002104' 000305                            SWAB    R5                      ;POSITION 2ND CRC BYTE
002106' 110567  176540                    MOVB    R5,CRC23                ;GET 2ND CRC BYTE
002112' 110467  176535                    MOVB    R4,CRC23+1              ;GET 3RD CRC BYTE
002116' 000304                            SWAB    R4                      ;POSITION 4TH CRC BYTE
002120' 110467  176530                    MOVB    R4,CRC4                 ;GET 4TH CRC BYTE
002124' 105067  176525                    CLRB    CRC4+1                  ;CLEAR JUNK FOR THE HELL OF IT
002130' 016720  176514                    MOV     CRC1,(R0)+              ;STORE DATA AND 1ST CRC BYTE IN BUFFER
002134' 016720  176512                    MOV     CRC23,(R0)+             ;STORE 2ND AND 3RD CRC BYTES IN BUFFER
002140' 016710  176510                    MOV     CRC4,(R0)               ;STORE 4TH CRC BYTE IN BUFFER
002144' 000402                            BR      26$                     ;CARRY ON

002146' 010520                    25$:    MOV     R5,(R0)+                ; APPEND THE CRC WORD
002150' 010420                            MOV     R4,(R0)+                ; APPEND THE OTHER CRC WORD
002152'                           26$:
                                  ;
                                  ;
                                  ; ***********************************************
                                  ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                  ; ***********************************************
                                  ;
002152' 012737  000200  177776            MOV     #MODE,@#CMDREG                  ; ENABLE, SELECT MODE
002160' 012737  100014  177774            MOV     #PROM!LOOP!DTCR,@#MODREG        ; PROM MODE AND LOOPBACK
002166' 012737  100000  177776            MOV     #ENABLE,@#CMDREG                ; LET HER GO!
                                  ;
002174' 016701  176446                    MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
002200' 005021                            CLR     (R1)+                   ; CLEAR OUT STATUS WORD
002202' 062703  000004                    ADD     #4,R3                   ; ACCOUNT FOR CRC BYTES
002206' 010321                            MOV     R3,(R1)+                ; WRITE THE BYTE COUNT
002210' 005037  021034                    CLR     @#CLRFIF                ; CLEAR THE FIFO
002214' 005067  176404                    CLR     FLG6
002220' 016737  176420  021032            MOV     RBUFP,@#LFRBUF          ; TELL DEUNA RECEIVE BUF LOC
002226' 016737  176414  021030            MOV     TBUFP,@#LTAC            ; TELL DEUNA WHERE XMIT BUF IS
                                  ;
```

```
002234' 106427  000140                         MTPS    #PRI03                  ; ALLCW INTERRUPTS
002240' 032767  000100  176356  30$:           BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
002246' 001774                                 BEQ     30$
                                ;
002250' 032767  000040  176346  40$:           BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INTERRUPT TOO
002256' 001774                                 BEQ     40$
                                ;
002260' 106427  000340                         MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                ;
                                ;
                                ; ************************************************
                                ; ***** WRITE STATUS REGISTER TO HOST MEMORY *****
                                ; ************************************************
                                ;
002264' 016700  176354                         MOV     RBUFP,R0                ; GET RECEIVE STATUS WORD
002270' 011003                                 MOV     (R0),R3                 ; GET STATUS WORD
                                ;
002272' 016737  176334  021010                 MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
002300' 016737  176330  021012                 MOV     IPCSR2+2,@#MDMA1
002306' 013700  021014                         MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LOW
002312' 013701  021014                         MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
002316' 062700  000004                         ADD     #4,R0                   ; BUMP TO NEXT HOST WORD
002322' 005501                                 ADC     R1
002324' 010037  021010                         MOV     R0,@#MDMA0              ; POINT TO PCBB
002330' 010137  021012                         MOV     R1,@#MDMA1
002334' 010337  021026                         MOV     R3,@#MDMAW0             ; WRITE STATUS WORD TO HOST
                                ;
                                ;
002340' 016702  176302                         MOV     TBUFP,R2                ; GET XMIT BUFFER STATUS
002344' 011203                                 MOV     (R2),R3
002346' 010337  021026                         MOV     R3,@#MDMAW0             ; WRITE XMIT STATUS TO HOST
002352' 112767  000003  176251                 MOVB    #3,PCSR1+1              ; TELL HIM WHICH TEST IT IS
002360' 000241                                 CLC
002362' 000207                                 RTS     PC
```

```
                                        ;
                                        .SBTTL  RECEIVE BUFFER RECOVERY - RUNT TEST
                                        ;
                                        ; **********************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; **********************************
                                        ;
002364' 112767  000002  176236  MICF4:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
002372' 016737  176232  021020          MOV     PCSR1,@#IPCSR1
                                        ;
002400' 005067  176222                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                        ;
                                        ;
                                        ; ***********************************
                                        ; ***** CLEAR THE RECEIVE BUFFER *****
                                        ; ***********************************
                                        ;
002404' 012700  100000                  MOV     #LINADR,R0              ; RECEIVER BUFFER STARTS HERE
002410' 010067  176230                  MOV     R0,RBUFP
002414' 005020                  10$:    CLR     (R0)+                   ; FILL BUFFER WITH ZEROS
002416' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
002422' 103774                          BLO     10$
                                        ;
                                        ;
                                        ; *************************************
                                        ; ***** FILL XMIT WITH RUNT MARKER *****
                                        ; *************************************
                                        ;
002424' 010067  176216                  MOV     R0,TBUFP                ; SAVE COPY OF THE ADDRESS
002430' 012720  125252          20$:    MOV     #125252,(R0)+
002434' 020027  110000                  CMP     R0,#LINADR+SIZ2K
002440' 103773                          BLO     20$
                                        ;
                                        ;
                                        ; ***********************************************
                                        ; ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
                                        ; ***********************************************
                                        ;
002442' 016737  176164  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GE' HOST PCBB ADDRESS
002450' 016737  176160  021012          MOV     IPCSR2+2,@#MDMA1
002456' 013700  021014                  MOV     @#MDMAR0,R0             ; R4 NOW CONTAINS PCBB LOW
002462' 013701  021014                  MOV     @#MDMAR0,R1             ; P5 NOW CONTAINS PCBB HIGH
002466' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB
002472' 010137  021012                  MOV     R1,@#MDMA1
002476' 013703  021014                  MOV     @#MDMAR0,R3             ; R3 NOW HOLDS BYTE COUNT
                                        ;
                                        ;
                                        ; ************************************************
                                        ; ***** SET UP LINK FOR RUNT DATAGRAM LOOPBACK *****
                                        ; ************************************************
                                        ;
002502' 012737  000200  177776          MOV     #MODE,@#CMDREG  ; DISABLE XMIT CRC - TRUE BC
002510' 012737  100014  177774          MOV     #PROM!LOOP!DTCR,@#MCDREG
002516' 012737  100000  177776          MOV     #ENABLE,@#CMDREG
                                        ;
002524' 016704  176116                  MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
```

```
002530' 005024                                 CLR     (R4)+                   ; CLEAR OUT STATUS WORD
002532' 010324                                 MOV     R3,(R4)+                ; WRITE PASSED BYTE COUNT
                                       ;
002534' 005037  021034                         CLR     @#CLRFIF                ; CLEAR THE FIFO
002540' 005067  176060                         CLR     FLG6                    ; CLEAR THE INTERRUPT FLAG
002544' 016737  176074  021032                 MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002552' 016737  176070  021030                 MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                       ;
002560' 106427  000140                         MTPS    #PRI03                  ; ALLOW XMITTER TO INTERRUPT
002564' 026727  176036  000002  30$:           CMP     SANTIM,#2               ; WAIT AT MOST 2 SECONDS
002572' 002012                                 BGE     50$                     ; EXIT NORMALLY IF TIMER DONE
002574' 032767  000100  176022                 BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
002602' 001770                                 BEQ     30$                     ; IF NONE, WERE OK
                                       ;
                                       ;
                                       ; ****************************
                                       ; ***** ERROR FALLTHROUGH *****
                                       ; ****************************
                                       ;
002604' 032767  000040  176012  40$:           BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
002612' 001774                                 BEQ     40$
                                       ;
002614' 000261                                  EC                             ; TELL HOST UNA SCREWED UP!
002616' 000506                                 BR      90$                     ; EXIT WITHOUT WRITING PTR
                                       ;
                                       ;
                                       ; *******************************************
                                       ; ***** NOW TRY LEGITIMATE BUFFER SIZE *****
                                       ; *******************************************
                                       ;
002620' 016700  176022          50$:           MOV     TBUFP,R0                ; GET TRANSMIT BUFFER POINTER
002624' 012720  052525          60$:           MOV     #52525,(R0)+
002630' 020027  110000                         CMP     R0,#LINADR+SIZ2K
002634' 103773                                 BLO     60$
                                       ;
                                       ;
                                       ; ****************************************************
                                       ; ***** SET UP LINK FOR MINIMUM DATAGRAM MESSAGE *****
                                       ; ****************************************************
                                       ;
002636' 012737  000200  177776                 MOV     #MODE,@#CMDREG          ; ENABLE LINK, SEL MODE REG
002644' 012737  100004  177774                 MOV     #PROM!LOOP,@#MODREG
002652' 012737  100000  177776                 MOV     #ENABLE,@#CMDREG
                                       ;
002660' 016704  175762                         MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
002664' 005024                                 CLR     (R4)+                   ; CLEAR STATUS WORD
002666' 012724  000100                         MOV     #MINBC,(R4)+            ; SEND SMALL DATAGRAM
                                       ;
002672' 005067  175726                         CLR     FLG6
002676' 016737  175742  021032                 MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002704' 016737  175736  021030                 MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                       ;
002712' 106427  000140                         MTPS    #PRI03                  ; ALLOW XMITR TO INTERRUPT
002716' 032767  000100  175700  70$:           BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
002724' 001774                                 BEQ     70$
                                       ;
```

```
002726' 032767  000040  175670  80$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
002734' 001774                          BEQ     80$
                                ;
002736' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                ;
                                ;
                                ; ****************************************
                                ; ***** WRITE PARAMETERS TO HOST PCBB *****
                                ; ****************************************
                                ;
002742' 013703  021044                  MOV     @#LRBUF,R3              ; GET LINK POINTER
                                ;
002746' 016700  175672                  MOV     RBUFP,R0                ; POINT AT RECEIVER BUFFER
002752' 062700  000004                  ADD     #4,R0
002756' 011004                          MOV     (R0),R4                 ; GET PATTERN WORD
                                ;
                                ;
002760' 016737  175646  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
002766' 016737  175642  021012          MOV     IPCSR2+2,@#MDMA1
002774' 013700  021014                  MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LO
003000' 013701  021014                  MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
003004' 062700  000002                  ADD     #2,R0
003010' 005501                          ADC     R1
003012' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB+2
003016' C10137  021012                  MOV     R1,@#MDMA1
                                ;
003022' 010337  021026                  MOV     P3,@#MDMAW0             ; WRITE LINK POINTER
003026' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE DATA BYTE
                                ;
                                ;
003032' 000241                          CLC
003034' 112767  000004  175567  90$:    MOVB    #4,PCSR1+1              ; TELL HOST WE ARE DONE
003042' 000207                          RTS     PC
```

```
                                        .SBTTL  HALF-DUPLEX TEST
                                        ;
                                        ; ***********************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; ***********************************
                                        ;
003044' 112767  000002  175556  MICF5:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
003052' 016737  175552  021020          MOV     PCSR1,@#IPCSR1
                                        ;
003060' 005067  175542                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                        ;
                                        ;
                                        ; ***********************************
                                        ; ***** CLEAR THE RECEIVE BUFFER *****
                                        ; ***********************************
                                        ;
003064' 012700  100000                  MOV     #LINADR,R0              ; RECEIVER BUFFER STARTS HERE
003070' 010067  175550                  MOV     R0,RBUFP
003074' 005020                  10$:    CLR     (R0)+                   ; FILL BUFFER WITH ZEROS
003076' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
003102' 103774                          BLO     10$
                                        ;
                                        ;
                                        ; **************************************
                                        ; ***** FILL XMIT WITH RUNT MARKER *****
                                        ; **************************************
                                        ;
003104' 010067  175536                  MOV     R0,TBUFP                ; SAVE COPY OF THE ADDRESS
003110' 012720  125252          20$:    MOV     #125252,(R0)+
003114' 020027  110000                  CMP     R0,#LINADR+SIZ2K
003120' 103773                          BLO     20$
                                        ;
                                        ;
                                        ; ***************************************************
                                        ; ***** SET UP LINK FOR RUNT DATAGRAM LOOPBACK *****
                                        ; ***************************************************
                                        ;
003122' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; DISABLE XMIT CRC - TRUE BC
003130' 012737  100005  177774          MOV     #PROM!LOOP!HDX,@#MODREG
003136' 012737  100000  177776          MOV     #ENABLE,@#CMDREG
                                        ;
003144' 016704  175476                  MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
003150' 005024                          CLR     (R4)+                   ; CLEAR OUT STATUS WORD
003152' 010324                          MOV     R3,(R4)+                ; WRITE PASSED BYTE COUNT
                                        ;
003154' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
003160' 005067  175440                  CLR     FLG6                    ; CLEAR THE INTERRUPT FLAG
003164' 016737  175454  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
003172' 016737  175450  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                        ;
003200' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITTER TO INTERRUPT
003204' 026727  175416  000002  30$:    CMP     SANTIM,#2               ; WAIT AT MOST 2 SECONDS
003212' 002012                          BGE     50$                     ; EXIT NORMALLY IF TIMER DONE
003214' 032767  000100  175402          BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
003222' 001770                          BEQ     30$                     ; IF NONE, WERE OK
                                        ;
                                        ;
                                        ; ****************************
                                        ; ***** ERROR FALLTHROUGH *****
                                        ; ****************************
                                        ;
003224' 032767  000040  175372  40$:         BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
003232' 001774                               BEQ     40$
                                        ;
003234' 000261                               SEC                             ; TELL HOST UNA SCREWED UP!
003236' 000503                               BR      90$                     ; EXIT WITHOUT WRITING PTR
                                        ;
                                        ;
                                        ; **************************************
                                        ; ***** NOW SEE IF BUFFER RECOVERS *****
                                        ; **************************************
                                        ;
003240' 016700  175402          50$:         MOV     TBUFP,R0                ; GET TRANSMIT BUFFER POINTER
003244' 012720  052525          60$:         MOV     #52525,(R0)+
003250' 020027  110000                       CMP     R0,#LINADR+SIZ2K
003254' 103773                               BLO     60$
                                        ;
                                        ;
                                        ; ***************************************************
                                        ; ***** SET UP LINK FOR MINIMUM DATAGRAM MESSAGE *****
                                        ; ***************************************************
                                        ;
003256' 012737  000200  177776               MOV     #MODE,@#CMDREG          ; ENABLE LINK, SEL MODE REG
003264' 012737  100004  177774               MOV     #PROM!LOOP,@#MODREG
003272' 012737  100000  177776               MOV     #ENABLE,@#CMDREG
                                        ;
003300' 016704  175342                       MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
003304' 005024                               CLR     (R4)+                   ; CLEAR STATUS WORD
003306' 012724  000100                       MOV     #MINBC,(R4)+            ; SEND SMALL DATAGRAM
                                        ;
003312' 005067  175306                       CLR     FLG6
003316' 016737  175322  021032               MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
003324' 016737  175316  021030               MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                        ;
003332' 106427  000140                       MTPS    #PRI03                  ; ALLOW XMITR TO INTERRUPT
003336' 032767  000100  175260  70$:         BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
003344' 001774                               BEQ     70$
                                        ;
003346' 032767  000040  175250  80$:         BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
003354' 001774                               BEQ     80$
                                        ;
003356' 106427  000340                       MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                        ;
                                        ;
                                        ; ****************************************
                                        ; ***** WRITE PARAMETERS TO HOST PCBB *****
                                        ; ****************************************
                                        ;
003362' 013703  021044                       MOV     @#LRBUF,R3              ; GET LINK POINTER
                                        ;
003366' 016700  175252                       MOV     RBUFP,R0                ; POINT AT RECEIVER BUFFER
003372' 062700  000004                       ADD     #4,R0
```

```
003376' 011004                                 MOV     (R0),R4                 ; GET PATTERN WORD
                                        ;
                                        ;
003400' 016737  175226  021010                 MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
003406' 016737  175222  021012                 MOV     IPCSR2+2,@#MDMA1
003414' 013700  021014                         MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LO
003420' 013701  021014                         MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
003424' 010037  021010                         MOV     R0,@#MDMA0              ; POINT AT PCBB+2
003430' 010137  021012                         MOV     R1,@#MDMA1
                                        ;
003434' 010337  021026                         MOV     R3,@#MDMAW0             ; WRITE LINK POINTER
003440' 010437  021026                         MOV     R4,@#MDMAW0             ; WRITE DATA BYTE
                                        ;
                                        ;
003444' 000241                                 CLC
003446' 112767  000005  175155  90$:           MOVB    #5,PCSR1+1              ; TELL HOST WE ARE DONE
003454' 000207                                 RTS     PC

003456' 003460                          MICFSZ::MICFSZ-MICROF+2
                                        ;
        000001                          .END
```

```
ACLO   = 002000               87#
ADRERR= 000001               130#
ADRREG= 177774                34#
ARAM   = 000100               80#
BIT0   = 000001               54#     92     104
BIT1   = 000002               53#    105
BIT10  = 002000               44#     56      73      87
BIT11  = 004000               43#     72      86
BIT12  = 010000               42#     56      71     114
BIT13  = 020000               41#     70     113
BIT14  = 040000               40#     69      85     112     244
BIT15  = 100000               39#     68      78      84     111     249     252
BIT2   = 000004               52#     91     106
BIT3   = 000010               51#     90     107
BIT4   = 000020               50#     89     108
BIT5   = 000040               49#     88     109
BIT6   = 000100               48#     80     110
BIT7   = 000200               47#     79
BIT8   = 000400               46#     56      74
BIT9   = 001000               45#
BLKCRC   000656R     002     295#    411     480     605
CLRFIF= 021034                22#    389*    502*    645*    743*    874*
CMDREG= 177776                35#    142*    382*    384*    495*    497*    637*    639*    735*    737*    780*    782*    866*
                             868*    911*    913*
COLL   = 000020               89#
CRC1     000650R     002     284#    615*    616*    623
CRC23    000652R     002     285#    618*    619*    624
CRC4     000654R     002     286#    621*    622*    625
CSRFLG= 000001               104#    200     231
CSRVEC= 000064                97#    167*    168*
CSRWRT   000436R     002     166     231#
DATERR= 000000               129#
DMACSR= 021002                 9#    243     252*
DMAF   = 021022               17#
DMAINT   000472R     002     161     243#
DMAT0  = 021004               10#
DMAT1  = 021006               11#
DMAVEC= 000114                98#    162*    163*
DMAWC  = 021024               18#
DMDONE   000642R     002     243*    244     249     281#
DNI    = 004000               72#    146     224
DRTY   = 000040               88#
DTCR   = 000010               90#    496     638     736
ENABLE= 100000                78#    384     497     639     737     782     868     913
ENAL   = 040000               85#
ENCR   = 004000               86#
ERRFLG= 000002               105#    234
ERRINT   000446R     002     148     234#
EXCRCH= 157273               136#    412
EXCRCL= 020343               137#    414
FATI   = 000400               74#
FLG6     000624R     002     193*    196     200     225*    231*    234*    256*    264*    267*    276#    390*    395     398
                             503*    508     511     646*    651     654     744*    751     759     788*    793     796     875*
                             882     890     919*    924     927
GETCRC   000720R     002     302     311#
HDX    = 000001               92#    867
```

```
INERR = 000003              103#    222
INITH = 177777              132#    299
INITL = 177777              133#    300
INMON = 000001              101#    144     20[illegible] 220
INTST = 000002              102#    341     435     545     691     835
IOADR = 020000              124#    125
IOSIZ = 020000              120#    125
IPCSR0= 021000                8#    146*    205     224*    235*    246*    251*    257*
IPCSR1= 021020               16#    145*    208*    223*    342*    436*    546*    692*    836*
IPCSR2  000632R     002     191*    192*    279#    348     349     443     444     524     525     553     554     587     588
                            667     668     722     723     813     814     944     945
LASFTP= 012400               56#
LFRBUF= 021032               21#    391*    504*    647*    745*    789*    876*    920*
LINADR= 100000              126#    361     364     374     456     459     469     521     566     569     579     701     704
                            714     772     845     848     858     903
LINSIZ= 077774              122#
LOOP  = 000004               91#    383     496     638     736     781     867     912
LRBUF = 021044               26#    262     806     937
LTAC  = 021030               20#    392*    505*    648*    746*    790*    877*    921*
MAXBC = 002752              128#    388     410     479     483     501
MDMAR0= 021014               14#    350     351     354     445     446     449     526     527     555     556     559     589
                            590     595     669     670     724     725     728     815     816     946     947
MDMAR1= 021016               15#
MDMAW0= 021026               19#    532*    675*    680*    822*    823*    951*    952*
MDMAW1= 021036               23#
MDMA0 = 021010               12#    348*    352*    443*    447*    524*    530*    553*    557*    587*    593*    667*    673*
                            722*    726*    813*    819*    944*    948*
MDMA1 = 021012               13#    349*    353*    444*    448*    525*    531*    554*    558*    588*    594*    668*    674*
                            723*    727*    814*    820*    945*    949*
MDMSR = 021042               25#
MICFSZ  003456RG    002     959#
MICF1   001002R     002     270     341#
MICF2   001274R     002     271     435#
MICF3   001650R     002     272     545#
MICF4   002364R     002     273     691#
MICF5   003044R     002     274     835#
MICROF  000000RG    002     141#    959
MINBC = 000100              127#    786     917
MODE  = 000200               79#    382     495     637     735     780     866     911
MODREG= 177774               33#    383*    496*    638*    736*    781*    867*    912*
NPRERR= 100000              111#    251
NPRFLG= 000020              108#
NXMERR= 040000              112#    246
NXMFLG= 000010              107#
PARERR= 010000              114#    257
PARFLG= 000004              106#    256
PARINT  000550R     002     156     256#
PARVEC= 000140               99#    157*    158*
PCBADR  000636R     002     280#
PCEi  = 040000               69#
PCSRSW= 021040               24#    185
PCSR1   000630R     002     144*    145     220*    222*    223     278#    341*    342     417*    435*    436     535*    545*
                            546     681*    691*    692     827*    835*    836     956*
PHYAD0= 021060               27#
PHYAD1= 021062               28#
PHYAD2= 021064               29#
```

```
PHYAD3= 021066             30#
PHYAD4= 021070             31#
PHYAD5= 021072             32#
POLYH = 166670            134#    317
POLYL = 101440            135#    318
PRI00 = 000000             64#    194*
PRI01 = 000040             63#
PRI02 = 000100             62#
PRI03 = 000140             61#    394*    507*    650*    748*    792*    879*    923*
PRI04 = 000200             60#    168     178
PRI05 = 000240             59#    173     183
PRI06 = 000300             58#
PRI07 = 000340             57#    141*    151     158     163     199*    401*    514*    657*    799*    930*
PROM  = 100000             84#    383     496     638     736     781     867     912
RBUFP   000644R    002    282#    362*    391     408     457*    504     567*    647     664     702*    745     789     808
                          846*    876     920     939
RCEI  = 002000             73#
RCVFLG= 000100            110#    264     395     508     651     751     793     882     924
RCVINT  000566R    002    181     262#
RCVVEC= 000120             95#    182*    183*
ROMADR= 040000            125#    126
ROMSIZ= 040000            121#    126
RXI   = 020000             70#
SANTIM  000626R    002    240*    277#    694*    749     838*    880
SANVEC= 000134             96#    172*    173*
SERI  = 100000             68#
SIZ1K = 004000            115#    116     364     459     569     704     848
SIZ2K = 010000            116#    117     374     469     579     714     772     858     903
SIZ4K = 020000            117#    118     119     120
SIZ8K = 040000            118#    121     122
STACK = 001000            100#    143
TBLD    000612R    002    214     270#
TBUFP   000646R    002    283#    372*    386     392     467*    477     481     499     505     577*    602     606     641
                          648     678     712*    739     746     770     784     790     856*    870     877     901     915
                          921
TIMINT  000464R    002    171     240#
TRNFLG= 000040            109#    267     398     511     654     759     796     890     927
TRNINT  000602R    002    176     267#
TRNVEC= 000070             94#    177*    178*
TXI   = 010000             71#
UNIERR= 020000            113#    235
WCSADR= 030000            123#    124
WCSSIZ= 020000            119#    124
.     = 003460R    002    148     156     161     166     171     176     181     202     214     238     253     259     270
                          271     272     273     274


. ABS.  000000     000
        000000     001
MICRF   003460     002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROF.OBJ,MICROF.LST/CRF/DOC/NL:TOC/SOL=MICROF.MAC
```

```
RUN-TIME: 1 1 .3 SECONDS
RUN-TIME RATIO: 40/3=11.5
CORE USED:  9K  (18 PAGES)

DOCUMENT PAGES:  24
```

```
                                  .TITLE  MICROG - MICROCODE MODULE G

         000000'                  .CSECT MICRG

                                  .SBTTL   REGISTER DEFINITIONS USED BY THE T11

         021000                   IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
         021002                   DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
         021004                   DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
         021006                   DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
         021010                   MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
         021012                   MDMA1   -       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
         021014                   MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
         021016                   MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
         021020                   IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
         021022                   DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
         021024                   DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
         021026                   MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
         021030                   LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
         021032                   LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
         021034                   CLRFIF  =       21034   ;CLEAR FIFO
         021036                   MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
         021040                   PCSRSW  =       21040   ;SWITCH PACK REGISTER
         021042                   MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
         021044                   LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
         021060                   PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
         021062                   PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
         021064                   PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
         021066                   PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
         021070                   PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
         021072                   PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
         177774                   MODREG  =       177774  ;LINK MODE REGISTER
         177774                   ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
         177776                   CMDREG  =       177776  ;LINK COMMAND REGISTER

                                  .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

         100000                   BIT15   =       100000
         040000                   BIT14   =       40000
         020000                   BIT13   =       20000
         010000                   BIT12   =       10000
         004000                   BIT11   =       4000
         002000                   BIT10   =       2000
         001000                   BIT9    =       1000
         000400                   BIT8    =       400
         000200                   BIT7    =       200
         000100                   BIT6    =       100
         000040                   BIT5    =       40
         000020                   BIT4    =       20
         000010                   BIT3    =       10
         000004                   BIT2    =       4
         000002                   BIT1    =       2
         000001                   BIT0    =       1
                                  ;
         012400                   LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
```

```
000340          PRI07   =       340
000300          PRI06   =       300
000240          PRI05   =       240
000200          PRI04   =       200
000140          PRI03   =       140
000100          PRI02   =       100
000040          PRI01   =       40
000000          PRI00   =       0
                ;
                ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                ;
100000          SERI    =       BIT15
040000          PCEI    =       BIT14
020000          RXI     =       BIT13
010000          TXI     =       BIT12
004000          DNI     =       BIT11
002000          RCEI    =       BIT10
000400          FATI    =       BIT8
                ;
                ;LINK COMMAND REGISTER
                ;
100000          ENABLE  =       BIT15   ;ENABLE LINK MODULE
000200          MODE    =       BIT7    ;ENABLE MODE REGISTER
000100          ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                ;
                ;LINK MODE REGISTER
                ;
100000          PROM    =       BIT15   ;PROMISCUIOUS MODE
040000          ENAL    =       BIT14   ;ENABLE MULTICAST
004000          ENCR    =       BIT11   ;ENABLE COLLISION TEST
002000          ACLO    =       BIT10   ;ENABLE ACLO
000040          DRTY    =       BIT5    ;DISABLE RETRY LOGIC
000020          COLL    =       BIT4    ;SIMULATE A COLLISION
000010          DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
000004          LOOP    =       BIT2    ;ENABLE LOOPBACK

000070          TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
000120          RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
001000          STACK=  1000            ;STACK LOCATION
000001          INMON=  1               ;IN MICROMONITOR STATE
000002          INTST=  2               ;IN A TEST STATE
000003          INERR=  3               ;IN ERROR STATE
000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
000040          TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
000100          RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
100000          NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
040000          NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
020000          UNIERR= BIT13           ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
```

```
        010000          PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        004000          SIZ1K=  4000            ;1K WORDS
        010000          SIZ2K=  SIZ1K*2         ;2K WORDS
        014000          SIZ3K=  SIZ1K*3         ;3K WORDS
        020000          SIZ4K=  SIZ2K*2         ;4K WORDS
        040000          SIZ8K=  SIZ4K*2         ;8K WORDS
        020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774          LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
        000000          WCSADR= 0               ;BASE ADDRESS OF WCS
        020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        002756          MAXBC=  1518.           ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
        000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000004          CRCSIZ= 4               ;NUMBER OF BYTES IN A CRC
        000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
```

```

                                        .SBTTL  MICROCODE MODULE G

000000' 106427  000340          MICROG::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012706  001000                  MOV     #STACK,SP               ;SETUP STACK
000010' 112767  000001  000600          MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000016' 016737  000574  021020          MOV     PCSR1,@#IPCSR1
000024' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000032' 010700                          MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000034' 062700  000404                  ADD     #ERRINT-.,R0            ;HANDLER
000040' 005001                          CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000042' 010021                  10$:    MOV     R0,(R1)+                ;HANDLER
000044' 012721  000340                  MOV     #PRI07,(R1)+
000050' 020127  001000                  CMP     R1,#1000
000054' 002772                          BLT     10$

000056' 010700                          MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000060' 062700  000462                  ADD     #PARINT-.,R0
000064' 010037  000140                  MOV     R0,@#PARVEC
000070' 012737  000340  000142          MOV     #PRI07,@#PARVEC+2

000076' 010700                          MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000100' 062700  000364                  ADD     #DMAINT-.,R0
000104' 010037  000114                  MOV     R0,@#DMAVEC
000110' 012737  000340  000116          MOV     #PRI07,@#DMAVEC+2

000116' 010700                          MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000120' 062700  000310                  ADD     #CSRWRT-.,R0
000124' 010037  000064                  MOV     R0,@#CSRVEC
000130' 012737  000200  000066          MOV     #PRI04,@#CSRVEC+2

000136' 010700                          MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000140' 062700  000316                  ADD     #TIMINT-.,R0
000144' 010037  000134                  MOV     R0,@#SANVEC
000150' 012737  000240  000136          MOV     #PRI05,@#SANVEC+2

000156' 010700                          MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
000160' 062700  000414                  ADD     #TRNINT-.,R0
000164' 010037  000070                  MOV     R0,@#TRNVEC
000170' 012737  000200  000072          MOV     #PRI04,@#TRNVEC+2

000176' 010700                          MOV     PC,R0                   ;SETUP RECEIVER VECTOR
000200' 062700  000360                  ADD     #RCVINT- ,R0
000204' 010037  000120                  MOV     R0,@#RCVVEC
000210' 012737  000240  000122          MOV     #PRI05,@#RCVVEC+2

000216' 013700  021040                  MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000222' 052700  176000                  BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000226' 006300                          ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000230' 006300                          ASL     R0
000232' 006300                          ASL     R0
000234' 062700  000004                  ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000240' 010067  000354                  MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000244' 012767  000003  000350          MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000252' 005067  000334                  CLR     FLG7                    ;INITIALIZE FLAG WORD
000256' 106427  000000          15$:    MTPS    #PRI00                  ;ALLOW INTERRUPTS
```

```
000262' 005767  000324                  20$:    TST     FLG7                    ;WAIT FOR A COMMAND FROM HOST
000266' 001775                                  BEQ     20$

000270' 106427  000340                          MTPS    #PRI07                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000274' 032767  000001  000310                  BIT     #CSRFLG,FLG7            ;DID HOST GIVE US A COMMAND?
000302' 001001                                  BNE     30$                     ;YES
000304' 000777                                  BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                                ;FOR LACK OF ANYTHING BETTER TO DO

000306' 113700  021000                  30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000312' 042700  177760                          BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000316' 001004                                  BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000320' 012737  000001  021020                  MOV     #INMON,@#IPCSR1         ;YES, CLEAR OUT THE TEST # BITS
000326' 000432                                  BR      50$
000330' 022700  000017                  35$:    CMP     #17,R0                  ;RESTART OPERATIONAL MICROCODE?
000334' 001432                                  BEQ     60$                     ;YES
000336' 162700  000001                          SUB     #1,R0
000342' 010701                                  MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000344' 062701  000240                          ADD     #TBLG-.,R1
000350' 006300                                  ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000352' 060001                                  ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000354' 061101                                  ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000356' 004711                                  JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000360' 103404                                  BCS     40$                     ;ERROR OCCURRED
000362' 112767  000001  000226                  MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000370' 000403                                  BR      45$                     ;MICROMONITR
000372' 112767  000003  000216          40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000400' 016737  000212  021020          45$:    MOV     PCSR1,@#IPCSR1
000406' 012737  004000  021000                  MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000414' 005067  000172                  50$:    CLR     FLG7                    ;RESET FLAG WORD
000420' 000716                                  BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000422' 005000                          60$:    CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000424' 000137  040006                          JMP     @#40006                 ;START OPERATIONAL MICROCODE

000430' 052767  000001  000154          CSRWRT: BIS     #CSRFLG,FLG7            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000436' 000002                                  RTI

000440' 052767  000002  000144          ERRINT: BIS     #ERRFLG,FLG7            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000446' 012737  020000  021000                  MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                                ;HAPPENED
000454' 000777                                  BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                                ;COUNT ON THE HOST TO TIMEOUT WAITING

000456' 005267  000132                  TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000462' 000002                                  RTI


000464' 013767  021002  000136          DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000472' 032767  040000  000130                  BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000500' 001404                                  BEQ     10$                     ;NO
000502' 012737  040000  021000                  MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                                ;LOCATION WAS ADDRESSED
000510' 000407                                  BR      20$
000512' 032767  100000  000110          10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000520' 001407                          BEQ     30$                     ;NO
000522' 012737  100000  021000          MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
000530' 012737  100000  021002  20$:    MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000536' 000777                          BR      .                       ;SIT HERE AND SPIN WHEELS
000540' 000002                  30$:    RTI

000542' 052767  000004  000042  PARINT: BIS     #PARFLG,FLG7            ;SET PARITY ERROR OCCURRED
000550' 012737  010000  021000          MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                        ;OCCURRED
000556' 000777                          BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                        ;COUNT ON HOST TO TIMEOUT

000560' 005737  021044          RCVINT: TST     @#LRBUF                 ;READ BUFFER DONE REGISTER...
                                                                        ;WHICH CLEARS THE INTERRUPT
000564' 052767  000100  000020          BIS     #RCVFLG,FLG7            ;SET RECEIVER INTERRUPT OCCURRED
000572' 000002                          RTI

000574' 052767  000040  000010  TRNINT: BIS     #TRNFLG,FLG7            ;SET TRANSMITTER INTERRUPT OCCURRED
000602' 000002                          RTI

000604' 000036                  TBLG:   .WORD   MICG1-.                 ;COLLISION TEST
000606' 000306                          .WORD   MICG2-.                 ;TDR COUNTER TEST
000610' 000556                          .WORD   MICG3-.                 ;RETRY TEST

000612' 000000                  FLG7:   .WORD   0                       ;FLAG WORD
000614' 000000                  SANTIM: .WORD   0                       ;COUNT FOR SANITY TIMER
000616' 000000                  PCSR1:  .WORD   0                       ;COPY OF WHAT GOES TO PCSR1
000620' 000000  000000          IPCSR2: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000624' 000000  000000          PCBADR: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000630' 000000                  DMDONE: .WORD   0
000632' 000000                  RBUFP:  .WORD   0                       ;POINTER TO RECIEVE BUFFER
000634' 000000                  TBUFP:  .WORD   0                       ;POINTER TO XMIT BUFFER
000636' 000000                  DBUFP:  .WORD   0                       ;POINTER TO DMA ENGINE BUFFER
000640' 000000                  MBUFP:  .WORD   0                       ;POINTER TO MICROCPU DMA BUFFER
```

```
NO
NO
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
MS
OP
PC
PR
PR
PR
PR
RE
RE
RF
SE
SE
SL
ST
SV
XF
XF
XF

.

NO
```

```
000642' 112767  000002  177746  MICG1:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE IN A TEST
000650' 016737  177742  021020          MOV     PCSR1,@#IPCSR1

000656' 012700  100000                  MOV     #LINADR,R0              ;FILL RECEIVE BUFFER WITH ZEROS
000662' 010067  177744                  MOV     R0,RBUFP
000666' 005020                  10$:    CLR     (R0)+
000670' 020027  104000                  CMP     R0,#LINADR+SIZ1K
000674' 103774                          BLO     10$

000676' 010067  177732                  MOV     R0,TBUFP                ;FILL TRANSMIT BUFFEP WITH 1'S
000702' 012720  177777          20$:    MOV     #177777,(R0)+
000706' 020027  110000                  CMP     R0,#LINADR+SIZ2K
000712' 103773                          BLO     20$

000714' 016737  177700  021010          MOV     IPCSR2,@#MDMA0          ;GET HOST'S PCBB ADDRESS
000722' 016737  177674  021012          MOV     IPCSR2+2,@#MDMA1
000730' 013704  021014                  MOV     @#MDMAR0,R4             ;R4 HOLDS PCBB LOW ADDRESS
000734' 013705  021014                  MOV     @#MDMAR0,R5             ;R5 HOLDS PCBB HIGH ADDRESS
000740' 010437  021010                  MOV     R4,@#MDMA0              ;POINT TO PCBB+0
000744' 010537  021012                  MOV     R5,@#MDMA1
000750' 013703  021014                  MOV     @#MDMAR0,R3             ;GET WHAT HOST WANTS TO GO INTO LINK
                                                                        ;MODE REGISTER

000754' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
000762' 010337  177774                  MOV     R3,@#MODREG             ;LOAD MODE REGISTER WITH HOST'S VALUE

000766' 016702  177642                  MOV     TBUFP,R2                ;GET BEGINNING OF TRANSMIT BUFFER
000772' 005722                          TST     (R2)+                   ;SKIP FIRST WORD
000774' 012722  002752                  MOV     #MAXBC-CRCSIZ,(R2)+     ;SET TO TRANSMIT MAX LENGTH PACKET
001000' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001004' 005067  177602                  CLR     FLG7                    ;CLEAR OUT THE FLAG WORD
001010' 016737  177616  021032          MOV     RBUFP,@#LFRBUF          ;TELL UNA WHERE RECEIVE BUFF IS
001016' 016737  177612  021030          MOV     TBUFP,@#LTAC            ;TELL UNA WHERE TRANSMIT BUFF IS
                                                                        ;START TRANSMIT OPERATION

001024' 106427  000140                  MTPS    #PRI03                  ;ALLOW INTERRUPTS
001030' 032767  000040  177554  30$:    BIT     #TRNFLG,FLG7            ;WAIT FOR TRANSMIT DONE
001036' 001774                          BEQ     30$

001040' 106427  000340                  MTPS    #PRI07                  ;DISABLE ANY MORE INTERRUPTS

001044' 016700  177564                  MOV     TBUFP,R0                ;POINT TO TRANSMIT BUFFER
001050' 012001                          MOV     (R0)+,R1                ;GET FIRST TRANSMIT STATUS WORD
001052' 012002                          MOV     (R0)+,R2                ;GET SECOND TRANSMIT STATUS WORD

001054' 062704  000002                  ADD     #2,R4                   ;POINT TO HOSTS PCBB+2
001060' 005505                          ADC     R5
001062' 010437  021010                  MOV     R4,@#MDMA0
001066' 010537  021012                  MOV     R5,@#MDMA1
001072' 010137  021026                  MOV     R1,@#MDMAW0             ;DUMP FIRST STATUS WORD TO PCBB+2
001076' 010237  021026                  MOV     R2,@#MDMAW0             ;DUMP SECOND STATUS WORD TO PCBB+4

001102' 112767  000001  177507          MOVB    #1,PCSP1+1              ;TELL HOST WHAT TEST JUST FINISHED
001110' 000241                          CLC
001112' 000207                          RTS     PC

```

```
001114' 112767  000002  177474  MICG2:  MOVB    #INTST,PCSR1            ;TELL HOST THAT WE ARE IN A TEST
001122' 016737  177470  021020          MOV     PCSR1,@#IPCSR1

001130' 012700  100000                  MOV     #LINADR,R0              ;FILL RECEIVE BUFFER WITH ALL 1'S
001134' 010067  177472                  MOV     R0,RBUFP
001140' 012720  177777          10$:    MOV     #177777,(R0)+
001144' 020027  104000                  CMP     R0,#LINADR+SIZ1K
001150' 103773                          BLO     10$

001152' 010067  177456                  MOV     R0,TBUFP                ;FILL TRANSMIT BUFFEP WITH ALL 0'S
001156' 005020                  20$:    CLR     (R0)+
001160' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001164' 103774                          BLO     20$

001166' 016737  177426  021010          MOV     IPCSR2,@#MDMA0          ;GET HOST'S PCBB ADDRESS
001174' 016737  177422  021012          MOV     IPCSR2+2,@#MDMA1
001202' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HOLDS HOST'S PCBB LOW ADDRESS
001206' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HOLDS HOST'S PCBB HI ADDRESS
001212' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
001216' 010137  021012                  MOV     R1,@#MDMA1
001222' 013703  021014                  MOV     @#MDMAR0,R3             ;GET BYTE COUNT FOR TRANSMIT OPERATION

001226' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
001234' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ;ACCEPT ALL PACKETS IN LOOP BACK

001242' 016704  177366                  MOV     TBUFP,R4                ;POINT TO TRANSMIT BUFFER
001246' 005724                          TST     (R4)+                   ;SKIP FIRST WORD
001250' 010324                          MOV     R3,(R4)+                ;SET BYTE COUNT TO HOST'S VALUE
001252' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001256' 005067  177330                  CLR     FLG7                    ;CLEAR THE FLAG WORD
001262' 016737  177344  021032          MOV     RBUFP,@#LFRBUF          ;TELL UNA WHERE RECEIVE BUFF IS
001270' 016737  177340  021030          MOV     TBUFP,@#LTAC            ;TELL UNA WHERE TRANSMIT BUFF IS AND
                                                                        ;START THE TRANSMIT OPERATION

001276' 106427  000140                  MTPS    #PRI03                  ;ALLOW INTERRUPTS
0C1302' 032767  000040  177302  30$:    BIT     #TRNFLG,FLG7            ;WAIT FOR TRANSMIT DONE INTERRUPT
001310' 001774                          BEQ     30$

001312' 106427  000340                  MTPS    #PRI07                  ;DISABLE INTERRUPTS

001316' 016702  177312                  MOV     TBUFP,R2                ;POINT TO TRANSMIT BUFFER
001322' 005722                          TST     (R2)+                   ;SKIP FIRST STATUS WORD
001324' 012203                          MOV     (R2)+,R3                ;GET SECOND STATUS WORD
001326' 042703  176000                  BIC     #176000,R3              ;STRIP ALL BUT TDR VALUE BITS
001332' 062700  000002                  ADD     #2,R0                   ;POINT TO HOST'S PCBB+2
001336' 005501                          ADC     R1
001340' 010037  021010                  MOV     R0,@#MDMA0
001344' 010137  021012                  MOV     R1,@#MDMA1
001350' 010337  021026                  MOV     R3,@#MDMAW0             ;DUMP TDR COUNTER VALUE TO PCBB+2

001354' 112767  000002  177235          MOVB    #2,PCSR1+1              ;TELL HOST WHAT TEST JUST FINISHED
001362' 000241                          CLC
001364' 000207                          RTS     PC

001366' 112767  000002  177222  MICG3:  MOVB    #INTST,PCSR1            ;TELL HOST THAT WE ARE IN A TEST
001374' 016737  177216  021020          MOV     PCSR1,@#IPCSR1
```

LN
#C
LC
TR
LO
HI
**
MO
SE
<.

```

001402' 012700  100000                  MOV     #LINADR,R0              ;FILL RECEIVE BUFFER WITH ALL 1'S
001406' 010067  177220                  MOV     R0,RBUFP
001412' 012720  177777          10$:    MOV     #177777,(R0)+
001416' 020027  104000                  CMP     R0,#LINADR+SIZ1K
001422' 103773                          BLO     10$

001424' 010067  177204                  MOV     R0,TBUFP                ;FILL TRANSMIT BUFFER WITH ALL 0'S
001430' 005020                  20$:    CLR     (R0)+
001432' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001436' 103774                          BLO     20$

001440' 016737  177154  021010          MOV     IPCSR2,@#MDMA0          ;GET HOST'S PCBB ADDRESS
001446' 016737  177150  021012          MOV     IPCSR2+2,@#MDMA1
001454' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HOLDS HOST'S PCBB LOW ADDRESS
001460' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HOLDS HOST'S PCBB HI ADDRESS
001464' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
001470' 010137  021012                  MOV     R1,@#MDMA1
001474' 013703  021014                  MOV     @#MDMAR0,R3             ;GET BYTE COUNT FOR TRANSMIT OPERATION

001500' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
001506' 012737  100024  177774          MOV     #PROM!LOOP!COLL,@#MODREG ;ACCEPT ALL PACKETS IN LOOP BACK AND
                                                                        ;FORCE A COLLISION

001514' 016704  177114                  MOV     TBUFP,R4                ;POINT TO TRANSMIT BUFFER
001520' 005724                          TST     (R4)+                   ;SKIP FIRST WORD
001522' 010324                          MOV     R3,(R4)+                ;SET BYTE COUNT TO HOST'S VALUE
001524' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001530' 005067  177056                  CLR     FLG7                    ;CLEAR THE FLAG WORD
001534' 016737  177072  021032          MOV     RBUFP,@#LFRBUF          ;TELL UNA WHERE RECEIVE BUFF IS
001542' 016737  177066  021030          MOV     TBUFP,@#LTAC            ;TELL UNA WHERE TRANSMIT BUFF IS AND
                                                                        ;START THE TRANSMIT OPERATION

001550' 005005                          CLR     R5                      ;CLEAR COUNTER
001552' 106427  000140                  MTPS    #PRI03                  ;ALLOW INTERRUPTS
001556' 032767  000040  177026  30$:    BIT     #TRNFLG,FLG7            ;IS TRANSMITTER DONE?
001564' 001002                          BNE     35$                     ;YES
001566' 005205                          INC     R5                      ;NO, COUNT TIME
001570' 000772                          BR      30$
001572' 106427  000340          35$:    MTPS    #PRI07                  ;DISABLE INTERRUPTS

001576' 062700  000002                  ADD     #2,R0                   ;POINT TO HOST'S PCBB+2
001602' 005501                          ADC     R1
001604' 010037  021010                  MOV     R0,@#MDMA0
001610' 010137  021012                  MOV     R1,@#MDMA1
001614' 010537  021026                  MOV     R5,@#MDMAW0             ;DUMP COUNTER VALUE TO PCBB+2

001620' 112767  000003  176771          MOVB    #3,PCSR1+1              ;TELL HOST WHAT TEST JUST FINISHED
001626' 000241                          CLC
001630' 000207                          RTS     PC

001632' 001634                  MICGSZ::MICGSZ-MICROG+2
        000001                  .END
```

```
ACLO   = 002000                 87#
ADRERR= 000001                 130#
ADRREG= 177774                  34#
ARAM   = 000100                 80#
BIT0   = 000001                 54#     103
BIT1   = 000002                 53#     104
BIT10  = 002000                 44#      56       73       87
BIT11  = 004000                 43#      72       86
BIT12  = 010000                 42#      56       71      113
BIT13  = 020000                 41#      70      112
BIT14  = 040000                 40#      69       85      111      238
BIT15  = 100000                 39#      68       78       84      110      243      246
BIT2   = 000004                 52#      91      105
BIT3   = 000010                 51#      90      106
BIT4   = 000020                 50#      89      107
BIT5   = 000040                 49#      88      108
BIT6   = 000100                 48#      80      109
BIT7   = 000200                 47#      79
BIT8   = 000400                 46#      56       74
BIT9   = 001000                 45#
CLRFIF= 021034                  22#     308*     363*     418*
CMDREG= 177776                  35#     302*     357*     411*
COLL   = 000020                 89#     412
CRCSIZ= 000004                 129#     307
CSRFLG= 000001                 103#     193      224
CSRVEC= 000064                  96#     160*     161*
CSRWRT  000430R      002       159      224#
DATERR= 000000                 128#
DBUFP   000636R      002       276#
DMACSR= 021002                   9#     237      246*
DMAF   = 021022                 17#
DMAINT  000464R      002       154      237#
DMAT0  = 021004                 10#
DMAT1  = 021006                 11#
DMAVEC= 000114                  97#     155*     156*
DMAWC  = 021024                 18#
DMDONE  000630R      002       237*     238      243      273#
DNI    = 004000                 72#     139      217
DRTY   = 000040                 88#
DTCR   = 000010                 90#
ENABLE= 100000                  78#     302      357      411
ENAL   = 040000                 85#
ENCR   = 004000                 86#
ERRFLG= 000002                 104#     227
ERRINT  000440R      002       141      227#
FATI   = 000400                 74#
FLG7    000612R      002       186*     189      193      218*     224*     227*     250*     258*     261*     268#     309*     315      364*
                               370      419*     426
INERR  = 000003                102#     215
INMON  = 000001                100#     137      201      213
INTST  = 000002                101#     279      335      389
IOADR  = 020000                124#     125
IOSIZ  = 020000                120#     125
IPCSR0= 021000                   8#     139*     198      217*     228*     240*     245*     251*
IPCSR1= 021020                  16#     138*     201*     216*     280*     336*     390*
IPCSR2  000620R      002       184*     185*     271#     293      294      349      350      403      404
```

```
LASFTP= 012400                 56#
LFRBUF= 021032                 21#    310*    365*    420*
LINADR= 100000                126#    282     285     290     338     341     346     392     395     400
LINSIZ= 077774                122#
LOOP  = 000004                 91#    358     412
LRBUF = 021044                 26#    256
LTAC  = 021030                 20#    311*    366*    421*
MAXBC = 002756                127#    307
MBUFP   000640R      002      277#
MDMAR0= 021014                 14#    295     296     299     351     352     355     405     406     409
MDMAR1= 021016                 15#
MDMAW0= 021026                 19#    328*    329*    383*    436*
MDMAW1= 021036                 23#
MDMA0 = 021010                 12#    293*    297*    326*    349*    353*    381*    403*    407*    434*
MDMA1 = 021012                 13#    294*    298*    327*    350*    354*    382*    404*    408*    435*
MDMSR = 021042                 25#
MICGSZ  001632RG     002      442#
MICG1   000642R      002      264     279#
MICG2   001114R      002      265     335#
MICG3   001366R      002      266     389#
MICROG  000000RG     002      135#    442
MODE  = 000200                 79#    302     357     411
MODREG= 177774                 33#    303*    358*    412*
NPRERR= 100000                110#    245
NPRFLG= 000020                107#
NXMERR= 040000                111#    240
NXMFLG= 000010                106#
PARERR= 010000                113#    251
PARFLG= 000004                105#    250
PARINT  000542R      002      149     250#
PARVEC= 000140                 98#    150*    151*
PCBADR  000624R      002      272#
PCEI  = 040000                 69#
PCSRSW= 021040                 24#    178
PCSR1   000616R      002      137*    138     213*    215*    216     270#    279*    280     331*    335*    336     385*    389*
                              390     438*
PHYAD0= 021060                 27#
PHYAD1= 021062                 28#
PHYAD2= 021064                 29#
PHYAD3= 021066                 30#
PHYAD4= 021070                 31#
PHYAD5= 021072                 32#
PRI00 = 000000                 64#    187*
PRI01 = 000040                 63#
PRI02 = 000100                 62#
PRI03 = 000140                 61#    314*    369*    425*
PRI04 = 000200                 60#    161     171
PRI05 = 000240                 59#    166     176
PRI06 = 000300                 58#
PRI07 = 000340                 57#    135*    144     151     156     192*    318*    373*    430*
PROM  = 100000                 84#    358     412
RBUFP   000632R      002      274#    283*    310     339*    365     393*    420
RCEI  = 002000                 73#
RCVFLG= 000100                109#    258
RCVINT  000560R      002      174     256#
RCVVEC= 000120                 94#    175*    176*
```

```
ROMADR= 040000               125#     126
ROMSIZ= 040000               121#     126
RXI   = 020000                70#
SANTIM  000614R     002      233*     269#
SANVEC= 000134                95#     165*     166*
SERI  = 100000                68#
SIZ1K = 004000               114#     115      116      285      341      395
SIZ2K = 010000               115#     117      290      346      400
SIZ3K = 014000               116#
SIZ4K = 020000               117#     118      119      120
SIZ8K = 040000               118#     121      122
STACK = 001000                99#     136
TBLG    000604R     002      207      264#
TBUFP   000634R     002      275#     288*     305      311      320      344*     360      366      375      398*     415      421
TIMINT  000456R     002      164      233#
TRNFLG= 000040               108#     261      315      370      426
TRNINT  000574R     002      169      261#
TRNVEC= 000070                93#     170*     171*
TXI   = 010000                71#
UNIERR= 020000               112#     228
WCSADR= 000000               123#     124
WCSSIZ= 020000               119#     124
.     = 001634R     002      141      149      154      159      164      169      174      195      207      230      247      253      264
                             265      266


. ABS.  000000      000
        000000      001
MICRG   001634      002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROG.OBJ,MICROG.LST/CRF/DOC/NL:TOC/SOL=MICROG.MAC
 RUN-TIME: .6 .9 .3 SECONDS
 RUN-TIME RATIO: 30/2=14.8
 CORE USED:  9K  (18 PAGES)

 DOCUMENT PAGES:  12
```

```
                                         .TITLE NO MORE MICROCODE MODULES
                000000'                  .CSECT  NOMORE
                                         .MCALL SVC
        000000'                          SVC
        000000'                          LASTAD
        000004'                          L$LAST::
                000001                   .END
```

```
ASSEMB= 000010            5
C$AU  = 000052            5#
C$AUTO= 000061            5#
C$BRK = 000022            5#
C$BSEG= 000004            5#
C$BSUB= 000002            5#
C$CEFG= 000045            5#
C$CLCK= 000062            5#
C$CLEA= 000012            5#
C$CLOS= 000035            5#
C$CLP1= 000006            5#
C$CVEC= 000036            5#
C$DCLN= 000044            5#
C$DODU= 000051            5#
C$DRPT= 000024            5#
C$DU  = 000053            5#
C$EDIT= 000003            5#
C$ERDF= 000055            5#
C$ERHR= 000056            5#
C$ERRO= 000060            5#
C$ERSF= 000054            5#
C$ERSO= 000057            5#
C$ESCA= 000010            5#
C$ESEG= 000005            5#
C$ESUB= 000003            5#
C$ETST= 000001            5#
C$EXIT= 000032            5#
C$GETB= 000026            5#
C$GETW= 000027            5#
C$GMAN= 000043            5#
C$GPHR= 000042            5#
C$GPLO= 000030            5#
C$GPRI= 000040            5#
C$INIT= 000011            5#
C$INLP= 000020            5#
C$MANI= 000050            5#
C$MEM = 000031            5#
C$MSG = 000023            5#
C$OPEN= 000034            5#
C$PNTB= 000014            5#
C$PNTF= 000017            5#
C$PNTS= 000016            5#
C$PNTX= 000015            5#
C$QIO = 000377            5#
C$RDBU= 000007            5#
C$REFG= 000047            5#
C$RESE= 000033            5#
C$REVI= 000003            5#
C$RFLA= 000021            5#
C$RPT = 000025            5#
C$SEFG= 000046            5#
C$SPRI= 000041            5#
C$SVEC= 000037            5#
C$TPRI= 000013            5#
DIAGMC= 000000            5
E$END = 002100            5#
```

```
E$LOAD= 000035                  5#
F$AU  = 000015                  5#
F$AUTO= 000020                  5#
F$BGN = 000040                  5#
F$CLEA= 000007                  5#
F$DU  = 000016                  5#
F$END = 000041                  5#
F$HARD= 000004                  5#
F$HW  = 000013                  5#
F$INIT= 000006                  5#
F$JMP = 000050                  5#
F$MOD = 000000                  5#
F$MSG = 000011                  5#
F$PROT= 000021                  5#
F$PWR = 000017                  5#
F$RPT = 000012                  5#
F$SEG = 000003                  5#
F$SOFT= 000005                  5#
F$SRV = 000010                  5#
F$SUB = 000002                  5#
F$SW  = 000014                  5#
F$TEST= 000001                  5#
G$CNTO= 000200                  5#
G$DELM= 000372                  5#
G$DISP= 000003                  5#
G$EXCP= 000400                  5#
G$HILI= 000002                  5#
G$LOLI= 000001                  5#
G$NO  = 000000                  5#
G$OFFS= 000400                  5#
G$OFSI= 000376                  5#
G$PRMA= 000001                  5#
G$PRMD= 000002                  5#
G$PRML= 000000                  5#
G$RADA= 000140                  5#
G$RADB= 000000                  5#
G$RADD= 000040                  5#
G$RADL= 000120                  5#
G$RADO= 000020                  5#
G$XFER= 000004                  5#
G$YES = 000010                  5#
I$AU  = 000041                  5#
I$AUTO= 000041                  5#
I$CLN = 000041                  5#
I$DU  = 000041                  5#
I$INIT= 000041                  5#
I$MOD = 000041                  5#
I$MSG = 000041                  5#
I$PROT= 000041                  5#
I$PTAB= 000041                  5#
I$PWR = 000041                  5#
I$RPT = 000041                  5#
I$SEG = 000041                  5#
I$SETU= 000041                  5#
I$SRV = 000041                  5#
I$SUB = 000041                  5#
```

```
I$TST = 000041              5#
J$JMP = 000167              5#
L$LAST  000004RG   002      6#
O$APTS= 000000              5#
O$AU  = 000000              5#
O$BGNR= 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$LSYM= 010000              5#
T$LINO= 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$TSTM= 177777              5#
T$TSTS= 000000              5#
X$ALWA= 000000              5#
X$FALS= 000040              5#
X$OFFS= 000400              5#
X$TRUE= 000020              5#
```

```
BCOMPL       1#       5#
BERROR       1#       5#
BGNAU        1#       5#
BGNAUT       1#       5#
BGNCLN       1#       5#
BGNDU        1#       5#
BGNHRD       1#       5#
BGNHW        1#       5#
BGNINI       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#
```

```
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#
GMANII       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#
M$BYTE       1#       5#
M$CHEC       1#       5#
M$CNTO       1#       5#
M$COUN       1#       5#
M$DATA       1#       5#
M$DECR       1#       5#
M$DEFA       1#       5#
M$ENDE       1#       5#
M$ERRI       1#       5#
M$ESCA       1#       5#
M$ESCS       1#       5#
M$EXCP       1#       5#                                                                                                  <MI
M$EXIT       1#       5#
M$EXSE       1#       5#
M$EXTJ       1#       5#                                                                                                  <MI
M$GEN        1#       5#       6#
M$GENB       1#       5#
M$GETS       1#       5#                                                                                                  <MI
M$GETT       1#       5#
M$GNGB       1#       5#       6#
M$GNIN       1#       5#       6#
M$GNLS       1#       5#                                                                                                  <M
M$GNSU       1#       5#
M$GNTA       1#       5#
M$GNTE       1#       5#                                                                                                  <M
M$HAPT       1#       5#
```

```
M$HNAP      1#      5#
M$INCR      1#      5#
M$IOSE      1#      5#
M$LDRO      1#      5#
M$MASK      1#      5#
M$MCHI      1#      5#
M$MCLO      1#      5#
M$MSK1      1#      5#
M$POP       1#      5#
M$PRIN      1#      5#
M$PUSH      1#      5#
M$PUT       1#      5#
M$PUT1      1#      5#
M$RADI      1#      5#
M$RBRO      1#      5#
M$RNRO      1#      5#
M$SETS      1#      5#
M$STAR      1#      5#
M$SVC       1#      5#
M$TLAB      1#      5#
M$TSTL      1#      5#
M$WORD      1#      5#      6
M$XFER      1#      5#
OPEN        1#      5#
POINTE      1#      5#
PRINTB      1#      5#
PRINTF      1#      5#
PRINTS      1#      5#
PRINTX      1#      5#
READBU      1#      5#
READEF      1#      5#
RFLAGS      1#      5#
SETPRI      1#      5#
SETVEC      1#      5#
SLASH       1#      5#
STARS       1#      5#
SVC         1#      3#      4
XFER        1#      5#
XFERF       1#      5#
XFERT       1#      5#


. ABS.   000000      000
         000000      001
NOMORE   000004      002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 NOMORE.OBJ,NOMORE.LST/CRF/DOC/NL:TOC/SOL SVC34R.P11,NOMORE.MAC
 RUN-TIME: 2 2 .4 SECONDS
 RUN-TIME RATIO: 56/5=11.0
 CORE USED:  17K  (34 PAGES)

 DOCUMENT PAGES:  7
```

```
<MI

<MI

<NO

***
MOD
SEC
<
***
MOD
SEC
<
***
MOD
SEC
<
***
MOD
SEC
<
***
MOD
SEC
<
***
MOD
SEC
<
***
MOD
SEC
<
***
MOD
SEC
<

 RU
 5M
```

```
LNKX11   V023   12-JAN-83   1:15

#CZUAAA.BIC/B:2000,CZUAAA.MAP=CZUAAA,MICROA,MICROB,MICROC,MICROD,MICROE,MICROF,MICROG,NOMORE/E


LOAD MAP

TRANSFER ADDRESS: 000001
LOW LIMIT: 002000
HIGH LIMIT: 100772
***********
MODULE  PARAME
SECTION ENTRY    ADDRESS SIZE
<. 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
        EF.CON   000036
        EF.NEW   000035
        EF.PWR   000034
        EF.RES   000037
        EF.STA   000040
        ENP      000400
        ERRS     040000
        EVL      000004
        FATI     000400
        GETCMD   000002
```

```
GETPCB  000001
HOE     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
LASM    000001
LIM     000021
LINADR  100000
LINSIZ  077774
LOE     040000
LOT     000010
MAXBYT  002756
MINBYT  000100
NIHLT   000006
NIUNI   000007
NPRERR  100000
NXMERR  040000
OWN     100000
PARERR  010000
PCEI    040000
PCTO    000200
PFNOP1  000000
PFNOP2  000005
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
RAC S   000017
RC      000012
RCEI    002000
RDPA    000002
READY   000002
RESET   000000
RLSA    000024
RMAL    000006
RMIC    000010
```

```
        ROMADR  040000
        ROMSIZ  040000
        RPA     000004
        RPS     000016
        RRF     000010
        RSET    000040
        RSIDP   000022
        RTM     000014
        RUN     000003
        RXI     020000
        SECOND  000077
        SERI    100000
        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
        UAM     000200
        UNIERR  020000
        UNIHLT  000005
        WCSADR  000000
        WCSSIZ  020000
        WLSA    000025
        WMAL    000007
        WPA     000005
        WRF     000011
        WSIDP   000023
        WTM     000015
        XPWR    100000
<      >                002000  000000
<UNAREP>                002000  053774
        BITNAM  002310
        BITNUM  002306
        BITSTA  002312
        BNAMT0  002356
        BNAMT1  002416
        BNAMT2  002456
        BTMSG   054764
        BTMSG0  055243
        BTMSG1  055301
        BTMSG2  055334
        BTMSG3  055400
        BTTBL   055104
        CHKDNI  021330
        CHKINT  022050
        CHKMON  021776
        CLKBR   002276
        CLKCSR  002274
        CLKFRE  002302
```

```
CLKSRV  023674
CLKTAB  002274
CLKVEC  002300
CLRDNI  021374
CNTTAB  002624
COLTST  006046
CPUPRI  002674
CRCDAT  005621
CRCERR  005656
CRCPAT  005704
CRC16   021042
CRC32   020722
CSRNUM  002304
DATALD  004067
DBFRAM  004624
DEFADR  055033
DEFHDR  054770
DFPTBL  002262
DMABLK  004501
DMAFRM  004431
DMATO   004363
DPA     054750
DPPAT   004720
DWFRAM  004662
ERRINT  002666
FIFTST  005166
FORM1   006234
FORM10  007010
FORM11  007074
FORM12  007123
FORM13  007216
FORM15  007267
FORM16  007343
FORM17  007421
FORM18  007504
FORM19  007562
FORM2   006271
FORM20  007637
FORM21  007677
FORM22  007751
FORM23  010031
FORM24  010101
FORM25  010133
FORM26  010206
FORM27  010261
FORM28  010331
FORM29  010336
FORM3   006327
FORM30  010361
FORM31  010411
FORM32  010437
FORM33  010534
FORM34  010572
FORM35  010655
FORM36  010755
FORM37  011013
FORM38  011124
FORM39  011215
FORM4   006376
```

```
FORM40  011256
FORM41  011341
FORM42  011407
FORM43  011416
FORM44  011424
FORM45  011503
FORM46  011531
FORM47  011560
FORM48  011607
FORM49  011704
FORM5   006445
FORM50  011760
FORM51  012047
FORM52  012101
FORM53  012137
FORM54  012211
FORM55  012237
FORM56  012310
FORM57  012363
FORM58  012433
FORM59  012511
FORM6   006510
FORM60  012604
FORM61  012671
FORM62  012754
FORM63  01302[illegible]
FORM64  013066
FORM65  013137
FORM66  013165
FORM67  013252
FORM68  013336
FORM69  013433
FORM7   006566
FORM70  013551
FORM71  013613
FORM72  013655
FORM73  013727
FORM74  013772
FORM75  014047
FORM76  014115
FORM77  014151
FORM78  014225
FORM79  014265
FORM8   006650
FORM80  014334
FORM81  014413
FORM82  014456
FORM9   006732
FREMEM  002324
FRESIZ  002322
FRSTIM  002672
HAFDUP  006016
HEXDAT  054766
HEXDPA  021130
HEXH    021212
HEXL    021250
HEXTBL  055057
HEXVAL  054767
INTBIT  004251
```

```
INTVEC  004214
LASFT   004121
LNKARB  005334
LNKBYT  005022
LNKMEM  004333
LPMSG   054762
LPMSG0  055146
LPMSG1  055205
LPTBL   055100
L$ACP   002110
L$APT   002036
L$AU    023640
L$AUT   002070
L$AUTO  023470
L$CCP   002106
L$CLEA  023472
L$CO    002032
L$DEPO  002011
L$DESC  002704
L$DESP  002076
L$DEVP  002060
L$DISP  002124
L$DLY   002116
L$DTP   002040
L$DTYP  002034
L$DU    023632
L$DUT   002072
L$DVTY  002676
L$EF    002052
L$ENVI  002044
L$ETP   002102
L$EXP1  002046
L$EXP4  002064
L$EXP5  002066
L$HARD  055464
L$HIME  002120
L$HPCP  002016
L$HPTP  002022
L$HW    002262
L$ICP   002104
L$INIT  023124
L$LADP  002026
L$LOAD  002100
L$LUN   002074
L$MREV  002050
L$NAME  002000
L$PR!O  002042
L$PROT  023116
L$PRT   002112
L$REPP  002062
L$REV   002010
L$RPT   023110
L$SPC   002056
L$SPCP  002020
L$SPTP  002024
L$STA   002030
L$TEST  002114
L$TIML  002014
L$UNIT  002012
```

```
MAXCNT  005122
METER   002332
MICMOD  002320
MICRO   002326
MSG1    014750
MSG10   015372
MSG11   015414
MSG12   015500
MSG13   015524
MSG14   015572
MSG15   015620
MSG16   015650
MSG17   015742
MSG18   016034
MSG19   016064
MSG2    015012
MSG20   016160
MSG21   016250
MSG22   016340
MSG23   016454
MSG24   016476
MSG25   016626
MSG26   017014
MSG27   017060
MSG28   017126
MSG29   017174
MSG3    015056
MSG30   017276
MSG31   017344
MSG32   017510
MSG33   017606
MSG34   017634
MSG35   017704
MSG36   020002
MSG37   020024
MSG38   020046
MSG39   020070
MSG4    015120
MSG40   020112
MSG41   020210
MSG42   020262
MSG43   020432
MSG44   020454
MSG45   020616
MSG46   020700
MSG5    015142
MSG6    015212
MSG7    015300
MSG8    015322
MSG9    015344
MUCAST  005563
NEXMEM  002664
NOCLK   003566
NOPERR  004001
ODDBYT  005060
PATERN  002516
PAT1    002516
PAT2    002520
PAT3    002522
```

```
PAT4     002524
PAT5     002526
PAT6     002530
PCBB     002604
PCOMND   002316
PCSR0    002336
PCSR0C   002346
PCSR1    002340
PCSR1C   002350
PCSR2    002342
PCSR2C   002352
PCSR3    002344
PCSR3C   002354
PRTPAR   006154
PWHEN    002314
RACERR   003663
RACMG1   014546
RACMG2   014574
RACMG3   014632
RACMG4   014704
RACMG7   014726
RBRRUN   005743
RCVDON   004572
RETLOG   006124
REUNA    022104
RLNKAD   005207
ROMDMP   004042
RREV     054756
RRWER    003741
RSETER   003717
SLFTST   004021
SMSG00   026404
SMSG01   026432
SMSG02   026457
SMSG03   026470
SMSG04   026525
SMSG05   026552
SMSG06   026613
SMSG07   026644
SMSG10   026660
SMSG11   026710
SMSG12   026721
SMSG13   026734
SMSG14   026746
SMSG15   026760
SMSG16   026772
SMSG17   027004
SMSG20   027016
SMSG21   027037
SMSG22   027051
SMSG23   027063
SMSG24   027075
SMSG25   027107
SMSG26   027121
SMSG27   027133
SMSG30   027145
SMSG31   027220
SMSG32   027270
SMSG33   027341
```

```
SMSG34  027403
SMSG35  027452
SMSG36  027513
SMSG37  027562
SMSG40  027623
SMSG41  027711
SMSG42  027774
SMSG43  030060
SMSG44  030135
SMSG45  030217
SMSG46  030273
SMSG47  030305
SMSG50  030317
SMSG51  030403
SMSG52  030464
SMSG53  030546
SMSG54  030621
SMSG55  030701
SMSG56  030753
SMSG57  030765
SMSG60  030777
SMSG61  031010
SMSG62  031031
SMSG63  031043
SMSG64  031070
SMSG65  031102
SMSG66  031121
SMSG67  031133
SMSG70  031145
SMSG71  031164
SMSG72  031215
SMSG73  031237
SMSG74  031251
SMSG75  031263
SMSG76  031275
SMSG77  031307
SPAT1   002532
SPAT2   002540
SPAT3   002546
SPAT4   002554
SPAT5   002562
SPAT6   002570
SPAT7   002576
STAMUX  004756
STAPAT  005400
STAPOS  005512
STAREJ  005444
STMSG   026202
STTBL   026204
SWADDR  002334
SWHDR   055114
SWPACK  054760
TDRCNT  006074
TIMOFF  021314
TIMON   021276
TIMTST  004311
TLNKAD  005260
TRAP4   023646
TRNDON  004540
```

```
T1        023710
T10       031332
T11       031746
T12       032534
T13       033214
T14       033632
T15       034122
T16       034630
T17       035146
T18       035614
T19       036172
T2        024122
T20       036552
T21       037100
T22       037410
T23       037720
T24       040300
T25       040660
T26       041254
T27       042016
T28       042414
T29       043016
T3        024454
T30       043336
T31       043726
T32       044312
T33       044676
T34       045336
T35       045666
T36       046206
T37       046556
T38       047076
T39       047436
T4        024566
T40       050026
T41       050410
T42       051042
T43       051446
T44       052536
T45       053260
T46       053674
T5        024704
T6        025366
T7        025450
T8        025546
T9        025770
UDBB      002614
UNACSR    002266
UNAINT    002670
UNAPRI    002272
UNASRV    023656
UNAVEC    002270
UNIT      002330
UNLOD     003515
WCSMEM    004165
$ADRER    003457
$AFTER    003340
$BEFOR    003351
$BITO     003267
```

```
           $BIT1     003261
           $BIT10    003172
           $BIT11    003163
           $BIT12    003154
           $BIT13    003145
           $BIT14    003136
           $BIT15    003127
           $BIT2     003253
           $BIT3     003245
           $BIT4     003237
           $BIT5     003231
           $BIT6     003223
           $BIT7     003215
           $BIT8     003207
           $BIT9     003201
           $CLR      003323
           $DATER    003444
           $DNI      002776
           $FATI     003017
           $GTCMD    003355
           $GTPCB    003346
           $ICAB     003074
           $INTE     003041
           $INTR     003030
           $NCLR     003311
           $NOP      003403
           $NSET     003275
           $PARER    003500
           $PATCH    055574
           $PCEI     002745
           $PCTO     003105
           $PDNDM    003415
           $RCEI     003006
           $RESET    003436
           $RMTC     003116
           $RSET     003052
           $RXI      002756
           $SERI     002734
           $SET      003305
           $SLFT     003371
           $STOP     003431
           $STRT     003407
           $TXI      002766
           $XPWR     003063
<MICRA >             055774   000024
           MICASZ    056016
           MICROA    055774
<MICRB >             056020   002314
           MICBSZ    060332
           MICROB    056020
<MICRC >             060334   003254
           MICCSZ    063606
           MICC7     063422
           MICROC    060334
<MICRD >             063610   004706
           MICDSZ    070514
           MICROD    063610
<MICRE >             070516   002734
           BFILE     073046
```

```
         MICESZ  073450
         MICROE  070516
<MICRF >         073452  003460
         MICFSZ  077130
         MICROF  073452
<MICRG >         077132  001634
         MICGSZ  100764
         MICROG  077132
<NOMORE>         100766  000004
         L$LAST  100772
**********
MODULE   MICROA
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   MICROB
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   MICROC
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   MICROD
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   MICROE
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   MICROF
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   MICROG
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000
**********
MODULE   NO
SECTION ENTRY    ADDRESS SIZE
<        >       100772  000000


  RUN-TIME    5 SECONDS
  5K CORE USED
```